

Standard ECMA-323
June 2001

ECMA

Standardizing Information and Communication Systems

**XML Protocol for Computer
Supported Telecommunications
Applications (CSTA) Phase III**

Standard ECMA-323
June 2001

ECMA

Standardizing Information and Communication Systems

**XML Protocol for Computer
Supported Telecommunications
Applications (CSTA) Phase III**

Brief History

This Standard defines an XML protocol 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 on practical experience of ECMA member companies and each one represents a pragmatic and widely-based consensus.

Phase III of CSTA extends the previous Phase II Standards in major theme directions as well as numerous details. This incorporates technology based upon the *versit* CTI Encyclopedia (Version 1.0), which was contributed to ECMA by *versit*. Major areas of advancement include:

- New categories of services and events such as capabilities exchange, charging, media attach services, call data recording (CDR), etc.
- Additional services and events for call and device control.
- Enhancement to existing services and events.
- Organization of services and events to reflect a grouping based on function (call control, device control, etc.).

XML Tools Acknowledgement

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

Table of Contents

1	Scope	1
2	Conformance	1
2.1	Static Requirements	1
2.2	Dynamic Requirements	1
2.3	PICS Requirement	1
3	References	1
3.1	ECMA Standards and Technical Reports	1
3.2	World Wide Web Consortium Specifications	2
4	Definitions and Abbreviations	2
5	CSTA Service Definition Model	2
5.1	Request/Response Interactions	2
5.2	Request/Response Protocol Requirements	2
5.3	Cross Referencing of Event Reports	2
5.4	Handling of Private Data	2
6	Interconnection service boundary	2
7	Association Management	3
7.1	Implicit association	3
7.2	Explicit association	3
8	XML Template Description	3
9	CSTA parameter types	5
9.1	Switching function objects	5
9.2	Device Identifiers	6
9.3	Call and connection identifiers	9
9.4	Connection states	11
9.5	Status reporting	12
9.6	Device and feature types and other parameters	16
9.7	Security	23
9.8	Common extensions	24
9.9	Call control	25
9.10	Capability Exchange	30
9.11	Call Detail Record	88
9.12	Charge information	91
9.13	Data call types	93

9.14	Escape types	94
9.15	Media services	95
9.16	Physical device features	98
9.17	Data Collection	101
9.18	Event Cause	102
9.19	Error Value	104
9.20	Empty (null element)	109
10	Top level schema	110
11	Capability exchange services	115
11.1	Services	115
11.1.1	Get logical device information	115
11.1.2	Get physical device information	117
11.1.3	Get switching function capabilities	118
11.1.4	Get switching function devices	122
11.1.5	Switching function devices	123
12	System services	124
12.1	Registration services	124
12.1.1	Change system status filter	124
12.1.2	System register	125
12.1.3	System register abort	126
12.1.4	System register cancel	127
12.2	Services	128
12.2.1	Request system status	128
12.2.2	System status	129
12.2.3	Switching function capabilities changed	130
12.2.4	Switching function devices changed	131
13	Monitoring services	132
13.1	Services	132
13.1.1	Change monitor filter	132
13.1.2	Monitor start	133
13.1.3	Monitor stop	134
14	Snapshot services	135
14.1	Services	135
14.1.1	Snapshot call	135
14.1.2	Snapshot device	136
14.1.3	Snapshot calldata	137
14.1.4	Snapshot devicedata	138

15	Call control services and events	139
15.1	Services	139
15.1.1	Accept call	139
15.1.2	Alternate call	140
15.1.3	Answer call	141
15.1.4	Call back call-related	142
15.1.5	Call back message call-related	143
15.1.6	Camp on call	144
15.1.7	Clear call	145
15.1.8	Clear Connection	146
15.1.9	Conference call	147
15.1.10	Consultation call	148
15.1.11	Deflect call	149
15.1.12	Dial digits	150
15.1.13	Directed pickup call	151
15.1.14	Group pickup call	152
15.1.15	Hold call	153
15.1.16	Intrude call	154
15.1.17	Join call	155
15.1.18	Make call	156
15.1.19	Make predictive call	157
15.1.20	Park call	159
15.1.21	Reconnect call	160
15.1.22	Retrieve call	161
15.1.23	Single step conference call	162
15.1.24	Single step transfer call	163
15.1.25	Transfer call	164
15.2	Events	165
15.2.1	Bridged	165
15.2.2	Call cleared	166
15.2.3	Conferenced	167
15.2.4	Connection cleared	168
15.2.5	Delivered	169
15.2.6	Digits dialed	170
15.2.7	Diverted	171
15.2.8	Established	172
15.2.9	Failed	173
15.2.10	Held	174
15.2.11	Network capabilities changed	175
15.2.12	Network reached	176
15.2.13	Offered	177
15.2.14	Originated	178
15.2.15	Queued	179
15.2.16	Retrieved	180
15.2.17	Service initiated	181

15.2.18	Transferred	182
16	Call associated features	183
16.1	Services	183
16.1.1	Associate data	183
16.1.2	Cancel telephony tones	184
16.1.3	Generate digits	185
16.1.4	Generate telephony tones	186
16.1.5	Send user information	187
16.2	Events	188
16.2.1	Call information	188
16.2.2	Charging	189
16.2.3	Digits generated	190
16.2.4	Telephony tones generated	191
16.2.5	Service completion failure	192
17	Media attachment services and events	193
17.1	Services	193
17.1.1	Attach media service	193
17.1.2	Detach media service	194
17.2	Events	195
17.2.1	Media attached	195
17.2.2	Media detached	196
18	Routeing services	197
18.1	Registration services	197
18.1.1	Route register	197
18.1.2	Route register abort	198
18.1.3	Route register cancel	199
18.2	Services	200
18.2.1	Re-Route	200
18.2.2	Route end	201
18.2.3	Route reject	202
18.2.4	Route request	203
18.2.5	Route select	204
18.2.6	Route used	205
19	Physical device features	206
19.1	Services	206
19.1.1	Button press	206
19.1.2	Get auditory apparatus information	207
19.1.3	Get button information	208
19.1.4	Get display	209
19.1.5	Get hookswitch status	210
19.1.6	Get lamp information	211

19.1.7	Get lamp mode	212
19.1.8	Get message waiting indicator	213
19.1.9	Get microphone gain	214
19.1.10	Get microphone mute	215
19.1.11	Get ringer status	216
19.1.12	Get speaker mute	217
19.1.13	Get speaker volume	218
19.1.14	Set button information	219
19.1.15	Set display	220
19.1.16	Set hookswitch status	221
19.1.17	Set lamp mode	222
19.1.18	Set message waiting indicator	223
19.1.19	Set microphone gain	224
19.1.20	Set microphone mute	225
19.1.21	Set ringer status	226
19.1.22	Set speaker mute	227
19.1.23	Set speaker volume	228
19.2	Events	229
19.2.1	Button information	229
19.2.2	Button press	230
19.2.3	Display updated	231
19.2.4	Hookswitch	232
19.2.5	Lamp mode	233
19.2.6	Message waiting	234
19.2.7	Microphone gain	235
19.2.8	Microphone mute	236
19.2.9	Ringer status	237
19.2.10	Speaker mute	238
19.2.11	Speaker volume	239
20	Logical device features	240
20.1	Services	240
20.1.1	Call back non-call-related	240
20.1.2	Call back message non-call-related	241
20.1.3	Cancel call back	242
20.1.4	Cancel call back message	243
20.1.5	Get agent state	244
20.1.6	Get auto answer	246
20.1.7	Get auto work mode	247
20.1.8	Get caller id status	248
20.1.9	Get do not disturb	249
20.1.10	Get forwarding	250
20.1.11	Get last number dialed	251
20.1.12	Get routeing mode	252
20.1.13	Set agent state	253

20.1.14	Set auto answer	254
20.1.15	Set auto work mode	255
20.1.16	Set caller id status	256
20.1.17	Set do not disturb	257
20.1.18	Set forwarding	258
20.1.19	Set routeing mode	259
20.2	Events	260
20.2.1	Agent busy	260
20.2.2	Agent logged off	261
20.2.3	Agent logged on	262
20.2.4	Agent not ready	263
20.2.5	Agent ready	264
20.2.6	Agent working after call	265
20.2.7	Auto answer	266
20.2.8	Auto work mode	267
20.2.9	Call back	268
20.2.10	Call back message	269
20.2.11	Caller id status	270
20.2.12	Do not disturb	271
20.2.13	Forwarding	272
20.2.14	Routeing mode	273
21	Device maintenance events	274
21.1	Events	274
21.1.1	Back in service	274
21.1.2	Device capabilities changed	275
21.1.3	Out of service	276
22	I/O services	277
22.1	Registration services	277
22.1.1	I/O register	277
22.1.2	I/O register abort	278
22.1.3	I/O register cancel	279
22.2	Services	280
22.2.1	Data path resumed	280
22.2.2	Data path suspended	281
22.2.3	Fast data	282
22.2.4	Resume data path	283
22.2.5	Send broadcast data	284
22.2.6	Send data	285
22.2.7	Send multicast data	286
22.2.8	Start data path	287
22.2.9	Stop data path	288
22.2.10	Suspend data path	289

23	Data Collection Services	290
23.1	Services	290
23.1.1	Data Collected	290
23.1.2	Data Collection Resumed	292
23.1.3	Data Collection Suspended	293
23.1.4	Resume Data Collection	294
23.1.5	Start Data Collection	295
23.1.6	Stop Data Collection	296
23.1.7	Suspend Data Collection	297
24	Voice unit services and events	298
24.1	Services	298
24.1.1	Concatenate message	298
24.1.2	Delete message	299
24.1.3	Play message	300
24.1.4	Query voice attribute	301
24.1.5	Record message	302
24.1.6	Reposition	303
24.1.7	Resume	304
24.1.8	Review	305
24.1.9	Set voice attribute	306
24.1.10	Stop	307
24.1.11	Suspend	308
24.1.12	Synthesize message	309
24.2	Events	310
24.2.1	Play	310
24.2.2	Record	311
24.2.3	Review	312
24.2.4	Stop	313
24.2.5	Suspend play	314
24.2.6	Suspend record	315
24.2.7	Voice attribute changed	316
25	Call detail record services	317
25.1	Services	317
25.1.1	Call detail records notification	317
25.1.2	Call detail records report	318
25.1.3	Send stored call detail records	319
25.1.4	Start call detail records transmission	320
25.1.5	Stop call detail records transmission	321
26	Vendor specific extensions services and events	322
26.1	Registration services	322
26.1.1	Escape register	322
26.1.2	Escape register abort	323

26.1.3	Escape register cancel	324
26.2	Services	325
26.2.1	Escape	325
26.2.2	Private data version selection	326
26.3	Events	327
26.3.1	Private event	327
Annex A Protocol Implementation Conformance Statement (PICS) Proforma		329
A.1	Introduction	329
A.2	Conformance	329
A.3	Instructions for completing the PICS proforma	329
A.4	Implementation identification	330
A.5	PICS proforma	330
Annex B Examples of Device Identifier Encodings		427
B.1	Simple DeviceID (no attributes)	427
B.2	DeviceID typed as a dialing number	427
B.3	DeviceID typed as a device number	427
B.4	DeviceID typed as a explicit national number	427
B.5	DeviceID typed as other type of numbering plan	427
B.6	DeviceID media characteristics	427
B.7	DeviceID including all attributes	427
Annex C Examples of Connection Identifier Encodings		429
C.1	Fully Specified Connection ID	429
C.2	CallID only ConnectionID	429
C.3	DeviceID only ConnectionID	429
Annex D Examples of CSTA-specific XML		431
D.1	Make Call service request - with only mandatory elements	431
D.2	Make Call service request - with optional elements	431
D.3	Make Call - Positive Acknowledgement	432
D.4	Make Call - Negative Acknowledgement	432
D.5	Delivered Event	432
Annex E CSTA XML with SOAP		433
E.1	CSTA service request	433
E.2	CSTA service response (positive acknowledgement)	433

E.3	CSTA service response (negative acknowledgement)	434
E.4	Delivered Event	434

Annex F CSTA XML with SOAP with HTTP 437

F.1	CSTA service request	437
F.2	CSTA service response (positive acknowledgement)	437
F.3	CSTA service response (negative acknowledgement)	438
F.4	Delivered Event	438

Annex G CSTA XML Over TCP 441

G.1	Message Format	441
G.2	TCP without SOAP	441
G.3	TCP with SOAP	442

1 Scope

This Standard specifies an XML protocol for the services described in ECMA-269, Services for Computer Supported Telecommunications Applications (CSTA) Phase III.

This Standard provides an alternative protocol to the ASN.1 based protocol specified in ECMA-285.

Clause 5 to clause 8 inclusive describes the concepts underlying the request/response model, application association, notation and service, and a description of the template used in this Standard.

Clause 9 to clause 26 inclusive contains CSTA-specific protocol details and forms the main part of this Standard.

Annex A specifies the PICS for this Standard.

Examples of CSTA XML encodings are provided in annex B through annex G.

2 Conformance

A manufacturer may select any part of the CSTA protocol, as specified in this Standard, for implementation on a system as long as it satisfies the minimum conformance requirements as specified in clause 2 of ECMA-269.

A Protocol Implementation Conformance Statement (PICS) shall be used to specify the operations which are provided by a particular implementation. A PICS shall also specify the parameter options which are used.

2.1 Static Requirements

To conform to this Standard, a system shall support the syntax as defined by the XML 1.0 specification and the structures and data types as defined in the XML Schema Structure and XML Schema Data Types specifications for the purpose of generating and interpreting CSTA protocol information for the operations supported.

2.2 Dynamic Requirements

To conform to this Standard, a system shall:

- a. follow the procedures as specified in this Standard, and ECMA-269, relevant to each CSTA operation that the system claims to implement; and
- b. satisfy the definitions, as specified in ECMA-269, relevant to each CSTA service that the system claims to implement.

2.3 PICS Requirement

To conform to this Standard, the following shall be stated by the implementer when defining a PICS corresponding to an application or implementation:

- a. which CSTA operations, as defined in ECMA-269, are supported by the system for the particular implementation; and
- b. which optional parameters are supported by each supported operations.

A PICS proforma is specified in annex A of this Standard.

3 References

3.1 ECMA/ISO Standards and Technical Reports

ECMA-269	Services for Computer Supported Telecommunications Applications (CSTA) Phase III, 4th edition (June 2000)
ECMA-285	Protocol for Computer Supported Telecommunications Applications (CSTA) Phase III, 2nd edition (June 2000)
ECMA TR/72	Glossary of Definitions and terminology for Computer Supported Telecommunications Applications (CSTA) Phase III, 3rd edition (June 2000)
ISO/IEC 8649:1996	Information technology - Open Systems Interconnection - Service definition for the Association Control Service Element (this corresponds to ITU-T Rec. X.217 1995)

ISO/IEC 8650-1:1996 Information technology - Open Systems Interconnection - Connection-oriented protocol for the Association Control Service Element: Protocol specification (this corresponds to ITU-T Rec. X.227 4/95)

3.2 World Wide Web Consortium Specifications

XML 1.0 XML 1.0. W3C Recommendation February 1998 (revised October 2000)

XML Schema Part 1 XML Schema Language Part 1: Structures. W3C Recommendation 2 May 2001

XML Schema Part 2 XML Schema Language Part 2: Datatypes. W3C Recommendation 2 May 2001

SOAP 1.1 Simple Object Access Protocol (SOAP) 1.1, W3C Note 08 May 2000

4 Definitions and Abbreviations

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

5 CSTA Service Definition Model

CSTA services specified in ECMA-269 are modeled as a request/response interaction.

Typically, one entity requests that a particular operation be performed; the other entity attempts to perform the operation and responds to the requestor.

5.1 Request/Response Interactions

CSTA services consist of the following types of request/response interactions:

- Services which specify a response in case of success or failure. The protocol description for these services includes a response element that represents the success (i.e. positive acknowledgement). In case of service failure, a fault element is used to provide the failure information (i.e. negative acknowledgement).
- Services which only specify a response in case of failure. The protocol description for these services do not include a response element. A fault element is used to provide the failure information (i.e. negative acknowledgement).

For CSTA event reports, no response is generated.

5.2 Request/Response Protocol Requirements

This Standard specifies CSTA-specific XML-encoded content for messages that are sent across a CSTA Service boundary. Typically this CSTA-specific XML is enveloped by a request/response protocol that provides a mechanism for correlating a CSTA request with a CSTA response, manages faults, defines header information, etc.

This Standard does not mandate that a specific request/response protocol be used. However, this Standard provides examples of how the CSTA-specific XML can be used with several industry request/response protocols and transports such as SOAP (Simple Object Access Protocol), HTTP, and TCP. These examples are provided beginning in annex E.

5.3 Cross Referencing of Event Reports

The Monitor Cross Reference Identifier in CSTA event reports is used to correlate events to an earlier Monitor Start service request (see 6.7.2.1 of ECMA-269).

5.4 Handling of Private Data

If an entity receives CSTA Private Data in a CSTA message, and it can not recognize the information contained, the CSTA Private Data shall be discarded, and the rest of the message shall be processed.

6 Interconnection service boundary

The protocol in this Standard is an OSI Application Layer protocol that assumes certain services are provided by the underlying layers, and these services are also assumed by the protocol for CSTA.

This protocol also provides a mechanism for secure transmission of CSTA messages as defined in this Standard.

7 Association Management

The protocol in this Standard operates in the context of an application association. This application association can be achieved by:

- an implicit association achieved via off-line agreement; or
- an explicit association that is achieved dynamically.

7.1 Implicit association

An a-priori agreement exists between switching and computing functions: the application context is implicit, dynamic negotiation is not possible.

7.2 Explicit association

An explicit application association is established by a process of exchanging messages that define the application context for the association.

This Standard does not require that a particular standard be used as part of the application association establishment procedure.

When ACSE (ISO/IEC 8649) is used to establish an application association per 7.2 of ECMA-269, the CSTA Association Information includes CSTA Version information that indicates the CSTA protocol versions supported by an implementation. The protocol defined in this Standard is indicated by setting the versionSix bit in the CSTAVersion parameter of the CSTA Association Information as defined in 8.2.2 of ECMA-285. Note that the ACSE messages are ASN.1 encoded while the CSTA messages are XML encoded as defined by this Standard.

8 XML Template Description

This clause describes the template format used to describe the XML schemas for CSTA services, events, and parameter types as specified in ECMA-269.

The template consists of the following parts:

- XML header - this consists of the text “<?xml version="1.0" encoding="UTF-8"?>” which identifies the XML version used.
- XML Schema header - this identifies the start of an XML schema and provides information such as:
 - the target namespace. In this Standard the target namespace is defined as “<http://www.ecma.ch/standards/ecma-323/csta>”.
 - the csta namespace is defined in the same way as the target namespace. XML names that refer to names in the CSTA namespace are prefixed with “csta:”.
 - the xsd namespace. This identifies the XML Schema. XML names that refer to names in the XML Schema namespace are prefixed with “xsd:”.
 - the element form default is specified as “qualified” which requires that all element names must be prefixed with a namespace (csta: or xsd:)
- documentation - this provides a comment that indicates the CSTA specific service, event report, or data-type set.
- included schemas - this provides a list of other CSTA schemas that are referenced by the service, event, or data type set.

- CSTA specific XML:
 - for CSTA services, this consists of a service request element and a service response element. (A comment is provided in some cases where a service response is not specified per ECMA-269.). This is followed, in some cases, by service-specific XML types that are referenced in the previous elements.
 - for CSTA event reports, this consists of an event report element followed, in some cases, by event-specific XML type definitions that are referenced in the event report element.
 - for CSTA data types, this consists of the data types that are referenced (included) by other CSTA XML schemas.
- XML Schema end tag - indicates the end of the schema.

Note that in general, each parameter defined in ECMA-269 is represented as an XML element and each parameter type in ECMA-269 is represented as an XML simple or complex type.

9 CSTA parameter types

9.1 Switching function objects

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-switching-function-objects</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-identifiers.xsd"/>
  <xsd:include schemaLocation="call-connection-identifiers.xsd"/>

  <xsd:complexType name="CSTAObject">
    <xsd:choice>
      <xsd:element name="deviceObject" type="csta:DeviceID"/>
      <xsd:element name="callObject" type="csta:ConnectionID"/>
    </xsd:choice>
  </xsd:complexType>

</xsd:schema>
```

9.2 Device Identifiers

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-device-identifiers</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="media-services.xsd"/>
  <xsd:include schemaLocation="empty.xsd"/>

  <xsd:complexType name="ExtendedDeviceID">
    <xsd:choice>
      <xsd:element name="deviceIdentifier" type="csta:DeviceID"/>
      <xsd:element name="notKnown" type="csta:Empty"/>
    </xsd:choice>
  </xsd:complexType>

  <xsd:complexType name="CallingDeviceID">
    <xsd:complexContent>
      <xsd:extension base="csta:ExtendedDeviceID"/>
    </xsd:complexContent>
  </xsd:complexType>

  <xsd:element name="callingDevice" type="csta:CallingDeviceID"/>

  <xsd:complexType name="CalledDeviceID">
    <xsd:complexContent>
      <xsd:extension base="csta:ExtendedDeviceID"/>
    </xsd:complexContent>
  </xsd:complexType>

  <xsd:element name="calledDevice" type="csta:CalledDeviceID"/>

  <xsd:complexType name="SubjectDeviceID">
    <xsd:complexContent>
      <xsd:extension base="csta:ExtendedDeviceID"/>
    </xsd:complexContent>
  </xsd:complexType>

  <xsd:complexType name="RedirectionDeviceID">
    <xsd:choice>
      <xsd:element name="numberDialed" type="csta:DeviceID"/>
      <xsd:element name="notKnown" type="csta:Empty"/>
      <xsd:element name="notRequired" type="csta:Empty"/>
      <xsd:element name="notSpecified" type="csta:Empty"/>
    </xsd:choice>
  </xsd:complexType>

  <xsd:element name="lastRedirectionDevice" type="csta:RedirectionDeviceID"/>

  <xsd:complexType name="AssociatedCallingDeviceID">
    <xsd:complexContent>
      <xsd:extension base="csta:ExtendedDeviceID"/>
    </xsd:complexContent>
  </xsd:complexType>

  <xsd:element name="associatedCallingDevice" type="csta:AssociatedCallingDeviceID"/>

  <xsd:complexType name="AssociatedCalledDeviceID">
    <xsd:complexContent>
      <xsd:extension base="csta:ExtendedDeviceID"/>
    </xsd:complexContent>
  </xsd:complexType>

```



```
</xsd:complexType>

<xsd:element name="associatedCalledDevice" type="csta:AssociatedCalledDeviceID"/>

<xsd:complexType name="NetworkCallingDeviceID">
  <xsd:complexContent>
    <xsd:extension base="csta:ExtendedDeviceID"/>
  </xsd:complexContent>
</xsd:complexType>

<xsd:element name="networkCallingDevice" type="csta:NetworkCallingDeviceID"/>

<xsd:complexType name="NetworkCalledDeviceID">
  <xsd:complexContent>
    <xsd:extension base="csta:ExtendedDeviceID"/>
  </xsd:complexContent>
</xsd:complexType>

<xsd:element name="networkCalledDevice" type="csta:NetworkCalledDeviceID"/>

<xsd:complexType name="DeviceID">
  <xsd:annotation>
    <xsd:documentation>uses flattened structure</xsd:documentation>
  </xsd:annotation>
  <xsd:simpleContent>
    <xsd:extension base="csta:DeviceIDType">
      <xsd:attribute name="typeOfNumber" default="dialingNumber">
        <xsd:simpleType>
          <xsd:restriction base="xsd:string">
            <xsd:enumeration value="dialingNumber"/>
            <xsd:enumeration value="deviceNumber"/>
            <xsd:enumeration value="implicitPublic"/>
            <xsd:enumeration value="explicitPublic:unknown"/>
            <xsd:enumeration value="explicitPublic:international"/>
            <xsd:enumeration value="explicitPublic:national"/>
            <xsd:enumeration value="explicitPublic:networkspecific"/>
            <xsd:enumeration value="explicitPublic:subscriber"/>
            <xsd:enumeration value="explicitPublic:abbreviated"/>
            <xsd:enumeration value="implicitPrivate"/>
            <xsd:enumeration value="explicitPrivate:unknown"/>
            <xsd:enumeration value="explicitPrivate:level3RegionalNumber"/>
            <xsd:enumeration value="explicitPrivate:level2RegionalNumber"/>
            <xsd:enumeration value="explicitPrivate:level1RegionalNumber"/>
            <xsd:enumeration value="explicitPrivate:pTNSpecificNumber"/>
            <xsd:enumeration value="explicitPrivate:localNumber"/>
            <xsd:enumeration value="explicitPrivate:abbreviated"/>
            <xsd:enumeration value="other"/>
          </xsd:restriction>
        </xsd:simpleType>
      </xsd:attribute>
      <xsd:attribute name="mediaClass" type="csta:MediaClassAttribute" default="notKnown"/>
      <xsd:attribute name="connectionRate" use="optional">
        <xsd:simpleType>
          <xsd:restriction base="xsd:long">
            <xsd:minInclusive value="0"/>
          </xsd:restriction>
        </xsd:simpleType>
      </xsd:attribute>
      <xsd:attribute name="bitRate" default="constant">
        <xsd:simpleType>
          <xsd:restriction base="xsd:string">
            <xsd:enumeration value="constant"/>
            <xsd:enumeration value="variable"/>
          </xsd:restriction>
        </xsd:simpleType>
      </xsd:attribute>
      <xsd:attribute name="delayTolerance" type="xsd:long" use="optional"/>
      <xsd:attribute name="switchingSubDomainCCIEType" use="optional">
```

```
<xsd:simpleType>
  <xsd:restriction base="xsd:string">
    <xsd:enumeration value="isdn"/>
    <xsd:enumeration value="atm"/>
    <xsd:enumeration value="isoEthernet"/>
    <xsd:enumeration value="rsvp"/>
    <xsd:enumeration value="other"/>
  </xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
<xsd:attribute name="switchingSubDomainInformationElements" use="optional">
  <xsd:simpleType>
    <xsd:restriction base="xsd:hexBinary"/>
  </xsd:simpleType>
</xsd:attribute>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>

<xsd:simpleType name="OtherPlan">
  <xsd:restriction base="xsd:hexBinary"/>
</xsd:simpleType>

<xsd:simpleType name="DeviceIDType">
  <xsd:union memberTypes="csta:OtherPlan csta:Number"/>
</xsd:simpleType>

<xsd:simpleType name="Number">
  <xsd:restriction base="xsd:string"/>
</xsd:simpleType>

<xsd:simpleType name="MediaClassComponents">
  <xsd:restriction base="xsd:string">
    <xsd:enumeration value="voice"/>
    <xsd:enumeration value="data"/>
    <xsd:enumeration value="image"/>
    <xsd:enumeration value="audio"/>
    <xsd:enumeration value="other"/>
    <xsd:enumeration value="notKnown"/>
  </xsd:restriction>
</xsd:simpleType>

<xsd:simpleType name="MediaClassAttribute">
  <xsd:list itemType="csta:MediaClassComponents"/>
</xsd:simpleType>

</xsd:schema>
```

9.3 Call and connection identifiers

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-call-connection-identifiers</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-identifiers.xsd"/>

  <xsd:complexType name="ConnectionID">
    <xsd:choice>
      <xsd:element name="callID" type="csta:CallID"/>
      <xsd:element name="deviceID" type="csta:LocalDeviceID"/>
      <xsd:sequence>
        <xsd:element name="callID" type="csta:CallID"/>
        <xsd:element name="deviceID" type="csta:LocalDeviceID"/>
      </xsd:sequence>
    </xsd:choice>
  </xsd:complexType>

  <xsd:simpleType name="CallID">
    <xsd:restriction base="xsd:string">
      <xsd:minLength value="0"/>
      <xsd:maxLength value="8"/>
      <xsd:pattern value="\d*"/>
    </xsd:restriction>
  </xsd:simpleType>

  <xsd:complexType name="LocalDeviceID">
    <xsd:annotation>
      <xsd:documentation>uses flattened structure - all types included here</xsd:documentation>
    </xsd:annotation>
    <xsd:simpleContent>
      <xsd:extension base="csta:DeviceIDType">
        <xsd:attribute name="typeOfNumber" default="dialingNumber">
          <xsd:simpleType>
            <xsd:restriction base="xsd:string">
              <xsd:enumeration value="dialingNumber"/>
              <xsd:enumeration value="deviceNumber"/>
              <xsd:enumeration value="implicitPublic"/>
              <xsd:enumeration value="explicitPublic:unknown"/>
              <xsd:enumeration value="explicitPublic:international"/>
              <xsd:enumeration value="explicitPublic:national"/>
              <xsd:enumeration value="explicitPublic:networkspecific"/>
              <xsd:enumeration value="explicitPublic:subscriber"/>
              <xsd:enumeration value="explicitPublic:abbreviated"/>
              <xsd:enumeration value="implicitPrivate"/>
              <xsd:enumeration value="explicitPrivate:unknown"/>
              <xsd:enumeration value="explicitPrivate:level3RegionalNumber"/>
              <xsd:enumeration value="explicitPrivate:level2RegionalNumber"/>
              <xsd:enumeration value="explicitPrivate:level1RegionalNumber"/>
              <xsd:enumeration value="explicitPrivate:pTNSpecificNumber"/>
              <xsd:enumeration value="explicitPrivate:localNumber"/>
              <xsd:enumeration value="explicitPrivate:abbreviated"/>
              <xsd:enumeration value="other"/>
              <xsd:enumeration value="dynamic"/>
            </xsd:restriction>
          </xsd:simpleType>
        </xsd:attribute>
        <xsd:attribute name="mediaClass" type="csta:MediaClassAttribute" default="notKnown"/>
        <xsd:attribute name="connectionRate" use="optional">
          <xsd:simpleType>
```

```
        <xsd:restriction base="xsd:long">
          <xsd:minInclusive value="0"/>
        </xsd:restriction>
      </xsd:simpleType>
    </xsd:attribute>
    <xsd:attribute name="bitRate" default="constant">
      <xsd:simpleType>
        <xsd:restriction base="xsd:string">
          <xsd:enumeration value="constant"/>
          <xsd:enumeration value="variable"/>
        </xsd:restriction>
      </xsd:simpleType>
    </xsd:attribute>
    <xsd:attribute name="delayTolerance" type="xsd:long" use="optional"/>
    <xsd:attribute name="switchingSubDomainCCIEType" use="optional">
      <xsd:simpleType>
        <xsd:restriction base="xsd:string">
          <xsd:enumeration value="isdn"/>
          <xsd:enumeration value="atm"/>
          <xsd:enumeration value="isoEthernet"/>
          <xsd:enumeration value="rsvp"/>
          <xsd:enumeration value="other"/>
        </xsd:restriction>
      </xsd:simpleType>
    </xsd:attribute>
    <xsd:attribute name="switchingSubDomainInformationElements" use="optional">
      <xsd:simpleType>
        <xsd:restriction base="xsd:hexBinary"/>
      </xsd:simpleType>
    </xsd:attribute>
  </xsd:extension>
</xsd:simpleContent>
</xsd:complexType>

</xsd:schema>
```

9.4 Connection states

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-connection-states</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="call-connection-identifiers.xsd"/>
  <xsd:include schemaLocation="device-identifiers.xsd"/>
  <xsd:include schemaLocation="media-services.xsd"/>
  <xsd:include schemaLocation="empty.xsd"/>

  <xsd:complexType name="ConnectionList">
    <xsd:sequence>
      <xsd:element name="connectionListItem" type="csta:ConnectionListItem"/>
    </xsd:sequence>
  </xsd:complexType>

  <xsd:simpleType name="LocalConnectionState">
    <xsd:restriction base="xsd:string">
      <xsd:enumeration value="null"/>
      <xsd:enumeration value="initiated"/>
      <xsd:enumeration value="alerting"/>
      <xsd:enumeration value="connected"/>
      <xsd:enumeration value="hold"/>
      <xsd:enumeration value="queued"/>
      <xsd:enumeration value="fail"/>
    </xsd:restriction>
  </xsd:simpleType>

  <xsd:element name="localConnectionInfo" type="csta:LocalConnectionState"/>

  <xsd:complexType name="ConnectionListItem">
    <xsd:sequence>
      <xsd:element name="newConnection" type="csta:ConnectionID" minOccurs="0"/>
      <xsd:element name="oldConnection" type="csta:ConnectionID" minOccurs="0"/>
      <xsd:element name="endpoint" minOccurs="0">
        <xsd:complexType>
          <xsd:choice>
            <xsd:element name="deviceID" type="csta:DeviceID"/>
            <xsd:element name="notKnown" type="csta:Empty"/>
          </xsd:choice>
        </xsd:complexType>
      </xsd:element>
      <xsd:element name="associatedNID" minOccurs="0">
        <xsd:complexType>
          <xsd:choice>
            <xsd:element name="deviceID" type="csta:DeviceID"/>
            <xsd:element name="notKnown" type="csta:Empty"/>
          </xsd:choice>
        </xsd:complexType>
      </xsd:element>
      <xsd:element name="resultingConnectionInfo" type="csta:ConnectionInformation"
minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>

</xsd:schema>
```

9.5 Status reporting

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-status-reporting</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="call-control.xsd"/>
  <xsd:include schemaLocation="switching-function-objects.xsd"/>
  <xsd:include schemaLocation="device-identifiers.xsd"/>
  <xsd:include schemaLocation="call-connection-identifiers.xsd"/>
  <xsd:include schemaLocation="connection-states.xsd"/>
  <xsd:include schemaLocation="media-services.xsd"/>

  <xsd:complexType name="MonitorObject">
    <xsd:complexContent>
      <xsd:extension base="csta:CSTAObject"/>
    </xsd:complexContent>
  </xsd:complexType>

  <xsd:simpleType name="MonitorCrossRefID">
    <xsd:restriction base="xsd:string">
      <xsd:minLength value="0"/>
      <xsd:maxLength value="4"/>
      <xsd:pattern value="\d*"/>
    </xsd:restriction>
  </xsd:simpleType>

  <xsd:element name="monitorCrossRefID" type="csta:MonitorCrossRefID"/>

  <xsd:complexType name="MonitorFilter">
    <xsd:sequence>
      <xsd:element name="callcontrol" type="csta:CallControlEvents" minOccurs="0"/>
      <xsd:element name="callAssociated" type="csta:CallAssociatedEvents" minOccurs="0"/>
      <xsd:element name="mediaAttachment" type="csta:MediaAttachmentEvents" minOccurs="0"/>
      <xsd:element name="physicalDeviceFeature" type="csta:PhysicalDeviceFeatureEvents"
minOccurs="0"/>
      <xsd:element name="logicalDeviceFeature" type="csta:LogicalDeviceFeatureEvents"
minOccurs="0"/>
      <xsd:element name="maintenance" type="csta:DeviceMaintenanceEvents" minOccurs="0"/>
      <xsd:element name="voiceUnit" type="csta:VoiceUnitEvents" minOccurs="0"/>
      <xsd:element name="private" type="csta:VendorSpecEvents" minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>

  <xsd:simpleType name="MonitorType">
    <xsd:restriction base="xsd:string">
      <xsd:enumeration value="call"/>
      <xsd:enumeration value="device"/>
    </xsd:restriction>
  </xsd:simpleType>

  <xsd:complexType name="MonitorMediaClass">
    <xsd:sequence>
      <xsd:element name="voice" type="xsd:boolean" minOccurs="0"/>
      <xsd:element name="data" type="xsd:boolean" minOccurs="0"/>
      <xsd:element name="image" type="xsd:boolean" minOccurs="0"/>
      <xsd:element name="audio" type="xsd:boolean" minOccurs="0"/>
      <xsd:element name="other" type="xsd:boolean" minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>
```

```
<xsd:complexType name="CallControlEvents">
  <xsd:sequence>
    <xsd:element name="bridged" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="callCleared" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="conferenced" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="connectionCleared" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="delivered" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="digitsDialed" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="diverted" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="established" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="failed" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="held" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="networkCapabilitiesChanged" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="networkReached" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="offered" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="originated" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="queued" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="retrieved" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="serviceInitiated" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="transferred" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="CallAssociatedEvents">
  <xsd:sequence>
    <xsd:element name="callInformation" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="charging" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="dtmfDigitsDetected" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="telephonyTonesDetected" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="serviceCompletionFailure" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="MediaAttachmentEvents">
  <xsd:sequence>
    <xsd:element name="mediaAttached" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="mediaDetached" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="PhysicalDeviceFeatureEvents">
  <xsd:sequence>
    <xsd:element name="buttonInformation" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="buttonPress" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="displayUpdated" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="hookswitch" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="lampMode" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="messageWaiting" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="microphoneGain" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="microphoneMute" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="ringerStatus" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="speakerMute" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="speakerVolume" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="LogicalDeviceFeatureEvents">
  <xsd:sequence>
    <xsd:element name="agentBusy" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="agentLoggedOn" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="agentLoggedOff" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="agentNotReady" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="agentReady" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="agentWorkingAfterCall" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="autoAnswer" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="autoWorkMode" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="callBack" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="callBackMessage" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

```
<xsd:element name="callerIDStatus" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="doNotDisturb" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="forwarding" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="routeingMode" type="xsd:boolean" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>

<xsd:complexType name="DeviceMaintenanceEvents">
  <xsd:sequence>
    <xsd:element name="backInService" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="deviceCapabilityChanged" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="outOfService" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="VoiceUnitEvents">
  <xsd:sequence>
    <xsd:element name="play" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="record" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="review" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="stop" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="suspendPlay" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="suspendRecord" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="voiceAttributesChange" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="VendorSpecEvents">
  <xsd:sequence>
    <xsd:element name="private" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="SnapshotDeviceData">
  <xsd:sequence>
    <xsd:element name="snapshotDeviceResponseInfo" type="csta:SnapshotDeviceResponseInfo"
      minOccurs="0" maxOccurs="unbounded"/>
  </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="SnapshotDeviceResponseInfo">
  <xsd:sequence>
    <xsd:element name="connectionIdentifier" type="csta:ConnectionID"/>
    <xsd:element name="localCallState" type="csta:CallState"/>
    <xsd:element ref="csta:servicesPermitted" minOccurs="0"/>
    <xsd:element name="mediaServiceInfoList" type="csta:DeviceMediaInfoList" minOccurs="0"/>
    <xsd:element ref="csta:mediaCallCharacteristics" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="DeviceMediaInfoList">
  <xsd:sequence>
    <xsd:element name="deviceMediaInfo" minOccurs="0" maxOccurs="unbounded">
      <xsd:complexType>
        <xsd:sequence>
          <xsd:element name="mediaStreamID" type="csta:MediaStreamID" minOccurs="0"/>
          <xsd:element name="connectionInformation" type="csta:ConnectionInformation"
            minOccurs="0"/>
        </xsd:sequence>
      </xsd:complexType>
    </xsd:element>
  </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="SnapshotCallData">
  <xsd:sequence>
    <xsd:element name="snapshotCallResponseInfo" type="csta:SnapshotCallResponseInfo"
      minOccurs="0" maxOccurs="unbounded"/>
  </xsd:sequence>
</xsd:complexType>
```



```
</xsd:sequence>
</xsd:complexType>

<xsd:complexType name="SnapshotCallResponseInfo">
  <xsd:sequence>
    <xsd:element name="deviceOnCall" type="csta:SubjectDeviceID"/>
    <xsd:element name="callIdentifier" type="csta:ConnectionID" minOccurs="0"/>
    <xsd:element ref="csta:localConnectionInfo" minOccurs="0"/>
    <xsd:element ref="csta:servicesPermitted" minOccurs="0"/>
    <xsd:element name="mediaServiceInfoList" type="csta:CallMediaInfoList" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="CallMediaInfoList">
  <xsd:sequence>
    <xsd:element name="callMediaInfo" minOccurs="0" maxOccurs="unbounded">
      <xsd:complexType>
        <xsd:sequence>
          <xsd:element name="mediaServiceType" type="csta:MediaServiceType"/>
          <xsd:element name="mediaServiceVersion" type="xsd:long" minOccurs="0"/>
          <xsd:element name="mediaServiceInstance" type="csta:MediaServiceInstanceID"
            minOccurs="0"/>
          <xsd:element name="mediaStreamID" type="csta:MediaStreamID" minOccurs="0"/>
          <xsd:element name="connectionInformation" type="csta:ConnectionInformation"
            minOccurs="0"/>
        </xsd:sequence>
      </xsd:complexType>
    </xsd:element>
  </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="CallState">
  <xsd:choice>
    <xsd:element name="compoundCallState" type="csta:CompoundCallState"/>
    <xsd:element name="simpleCallState" type="csta:SimpleCallState"/>
    <xsd:element name="unknown" type="csta:Empty"/>
  </xsd:choice>
</xsd:complexType>

<xsd:complexType name="CompoundCallState">
  <xsd:sequence>
    <xsd:element name="localConnectionState" type="csta:LocalConnectionState" minOccurs="0"
      maxOccurs="unbounded"/>
  </xsd:sequence>
</xsd:complexType>

<xsd:simpleType name="SimpleCallState">
  <xsd:restriction base="xsd:string">
    <xsd:enumeration value="callNull"/>
    <xsd:enumeration value="callPending"/>
    <xsd:enumeration value="callOriginated"/>
    <xsd:enumeration value="callDelivered"/>
    <xsd:enumeration value="callDeliveredHeld"/>
    <xsd:enumeration value="callReceived"/>
    <xsd:enumeration value="callEstablished"/>
    <xsd:enumeration value="callEstablishedHeld"/>
    <xsd:enumeration value="callReceivedOnHold"/>
    <xsd:enumeration value="callEstablishedOnHold"/>
    <xsd:enumeration value="callQueued"/>
    <xsd:enumeration value="callQueuedHeld"/>
    <xsd:enumeration value="callFailed"/>
    <xsd:enumeration value="callFailedHeld"/>
    <xsd:enumeration value="callBlocked"/>
  </xsd:restriction>
</xsd:simpleType>

</xsd:schema>
```

9.6 Device and feature types and other parameters

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-device-feature-types</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="call-connection-identifiers.xsd"/>
  <xsd:include schemaLocation="device-identifiers.xsd"/>

  <xsd:simpleType name="AccountInfo">
    <xsd:restriction base="xsd:hexBinary">
      <xsd:minLength value="0"/>
      <xsd:maxLength value="64"/>
    </xsd:restriction>
  </xsd:simpleType>

  <xsd:element name="accountCode" type="csta:AccountInfo"/>

  <xsd:simpleType name="AgentID">
    <xsd:restriction base="xsd:string">
      <xsd:minLength value="0"/>
      <xsd:maxLength value="32"/>
    </xsd:restriction>
  </xsd:simpleType>

  <xsd:simpleType name="AgentPassword">
    <xsd:restriction base="xsd:hexBinary">
      <xsd:minLength value="0"/>
      <xsd:maxLength value="64"/>
    </xsd:restriction>
  </xsd:simpleType>

  <xsd:simpleType name="AgentState">
    <xsd:restriction base="xsd:string">
      <xsd:enumeration value="agentNotReady"/>
      <xsd:enumeration value="agentNull"/>
      <xsd:enumeration value="agentReady"/>
      <xsd:enumeration value="agentBusy"/>
      <xsd:enumeration value="agentWorkingAfterCall"/>
    </xsd:restriction>
  </xsd:simpleType>

  <xsd:simpleType name="AuthCode">
    <xsd:restriction base="xsd:hexBinary">
      <xsd:minLength value="0"/>
      <xsd:maxLength value="64"/>
    </xsd:restriction>
  </xsd:simpleType>

  <xsd:element name="authCode" type="csta:AuthCode"/>

  <xsd:simpleType name="CorrelatorData">
    <xsd:restriction base="xsd:hexBinary">
      <xsd:minLength value="0"/>
      <xsd:maxLength value="64"/>
    </xsd:restriction>
  </xsd:simpleType>

  <xsd:element name="correlatorData" type="csta:CorrelatorData"/>

  <xsd:complexType name="IOCrossRefID">
    <xsd:choice>
```

```
<xsd:element name="switchProvided">
  <xsd:simpleType>
    <xsd:restriction base="xsd:string">
      <xsd:minLength value="0"/>
      <xsd:maxLength value="4"/>
      <xsd:pattern value="\d*"/>
    </xsd:restriction>
  </xsd:simpleType>
</xsd:element>
<xsd:element name="computerProvided">
  <xsd:simpleType>
    <xsd:restriction base="xsd:string">
      <xsd:minLength value="0"/>
      <xsd:maxLength value="4"/>
      <xsd:pattern value="\d*"/>
    </xsd:restriction>
  </xsd:simpleType>
</xsd:element>
</xsd:choice>
</xsd:complexType>

<xsd:simpleType name="IORegisterReqID">
  <xsd:restriction base="xsd:string">
    <xsd:minLength value="0"/>
    <xsd:maxLength value="4"/>
    <xsd:pattern value="\d*"/>
  </xsd:restriction>
</xsd:simpleType>

<xsd:simpleType name="MessageID">
  <xsd:restriction base="xsd:string">
    <xsd:pattern value="\d*"/>
  </xsd:restriction>
</xsd:simpleType>

<xsd:simpleType name="ParticipationType">
  <xsd:restriction base="xsd:string">
    <xsd:enumeration value="silent"/>
    <xsd:enumeration value="active"/>
  </xsd:restriction>
</xsd:simpleType>

<xsd:element name="routeRegisterReqID" type="csta:RouteRegisterReqID"/>

<xsd:simpleType name="RouteRegisterReqID">
  <xsd:restriction base="xsd:string">
    <xsd:minLength value="0"/>
    <xsd:maxLength value="4"/>
    <xsd:pattern value="\d*"/>
  </xsd:restriction>
</xsd:simpleType>

<xsd:element name="crossRefIdentifier" type="csta:RouteingCrossRefID"/>

<xsd:simpleType name="RouteingCrossRefID">
  <xsd:restriction base="xsd:string">
    <xsd:minLength value="0"/>
    <xsd:maxLength value="4"/>
    <xsd:pattern value="\d*"/>
  </xsd:restriction>
</xsd:simpleType>

<xsd:simpleType name="SelectValue">
  <xsd:restriction base="xsd:string">
    <xsd:enumeration value="normal"/>
    <xsd:enumeration value="leastCost"/>
    <xsd:enumeration value="emergency"/>
    <xsd:enumeration value="aCD"/>
  </xsd:restriction>
</xsd:simpleType>
```

```
<xsd:enumeration value="userDefined"/>
</xsd:restriction>
</xsd:simpleType>

<xsd:complexType name="RetryValue">
  <xsd:choice>
    <xsd:element name="noListAvailable" type="xsd:boolean"/>
    <xsd:element name="noCountAvailable" type="xsd:boolean"/>
    <xsd:element name="retryCount" type="xsd:long"/>
  </xsd:choice>
</xsd:complexType>

<xsd:simpleType name="SysStatRegisterID">
  <xsd:restriction base="xsd:string">
    <xsd:minLength value="0"/>
    <xsd:maxLength value="4"/>
    <xsd:pattern value="\d*" />
  </xsd:restriction>
</xsd:simpleType>

<xsd:element name="sysStatRegisterID" type="csta:SysStatRegisterID"/>

<xsd:simpleType name="SystemStatus">
  <xsd:restriction base="xsd:string">
    <xsd:enumeration value="disabled"/>
    <xsd:enumeration value="partiallyDisabled"/>
    <xsd:enumeration value="enabled"/>
    <xsd:enumeration value="initializing"/>
    <xsd:enumeration value="messagesLost"/>
    <xsd:enumeration value="normal"/>
    <xsd:enumeration value="overloadImminent"/>
    <xsd:enumeration value="overloadReached"/>
    <xsd:enumeration value="overloadRelieved"/>
  </xsd:restriction>
</xsd:simpleType>

<xsd:complexType name="CallOrigination">
  <xsd:sequence>
    <xsd:element name="internal" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="external" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="ForwardList">
  <xsd:sequence>
    <xsd:element name="forwardListItem" type="csta:ForwardListItem" maxOccurs="unbounded"/>
  </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="ForwardListItem">
  <xsd:sequence>
    <xsd:element name="forwardingType" type="csta:ForwardingType" minOccurs="0"/>
    <xsd:element name="forwardStatus" type="xsd:boolean"/>
    <xsd:element name="forwardDN" type="csta:DeviceID" minOccurs="0"/>
    <xsd:element name="forwardDefault" type="csta:ForwardDefault" minOccurs="0"/>
    <xsd:element name="ringCount" minOccurs="0">
      <xsd:simpleType>
        <xsd:restriction base="xsd:long">
          <xsd:minInclusive value="1"/>
          <xsd:maxInclusive value="100"/>
        </xsd:restriction>
      </xsd:simpleType>
    </xsd:element>
  </xsd:sequence>
</xsd:complexType>

<xsd:simpleType name="ForwardingType">
  <xsd:restriction base="xsd:string">
```

```
<xsd:enumeration value="forwardImmediate"/>
<xsd:enumeration value="forwardBusy"/>
<xsd:enumeration value="forwardNoAns"/>
<xsd:enumeration value="forwardDND"/>
<xsd:enumeration value="forwardBusyInt"/>
<xsd:enumeration value="forwardBusyExt"/>
<xsd:enumeration value="forwardNoAnsInt"/>
<xsd:enumeration value="forwardNoAnsExt"/>
<xsd:enumeration value="forwardImmInt"/>
<xsd:enumeration value="forwardImmExt"/>
<xsd:enumeration value="forwardDNDInt"/>
<xsd:enumeration value="forwardDNDExt"/>
</xsd:restriction>
</xsd:simpleType>

<xsd:simpleType name="ForwardDefault">
  <xsd:restriction base="xsd:string">
    <xsd:enumeration value="forwardingTypeAndForwardDN"/>
    <xsd:enumeration value="forwardingType"/>
    <xsd:enumeration value="forwardDN"/>
  </xsd:restriction>
</xsd:simpleType>

<xsd:simpleType name="PendingAgentState">
  <xsd:restriction base="xsd:string">
    <xsd:enumeration value="agentNotReady"/>
    <xsd:enumeration value="agentNull"/>
    <xsd:enumeration value="agentReady"/>
    <xsd:enumeration value="agentWorkingAfterCall"/>
  </xsd:restriction>
</xsd:simpleType>

<xsd:simpleType name="DataPathType">
  <xsd:restriction base="xsd:string">
    <xsd:enumeration value="text"/>
    <xsd:enumeration value="voice"/>
  </xsd:restriction>
</xsd:simpleType>

<xsd:complexType name="DisplayAttributeList">
  <xsd:sequence>
    <xsd:element name="physicalBaseRowNumber" type="xsd:long" minOccurs="0"/>
    <xsd:element name="physicalBaseColumnNumber" type="xsd:long" minOccurs="0"/>
    <xsd:element name="offset" type="xsd:long" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="TerminatingConditions">
  <xsd:sequence>
    <xsd:element name="durationExceeded" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="dtmfdigitDetected" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="endOfMessageDetected" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="speechDetected" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="AttributeInfo">
  <xsd:choice>
    <xsd:element name="encodingAlgorithm" type="csta:EncodingAlgorithm"/>
    <xsd:element name="samplingRate" type="xsd:long"/>
    <xsd:element name="duration" type="xsd:long"/>
    <xsd:element name="filename" type="xsd:string"/>
    <xsd:element name="currentPosition" type="xsd:long"/>
    <xsd:element name="currentSpeed" type="xsd:long"/>
    <xsd:element name="currentVolume">
      <xsd:simpleType>
        <xsd:restriction base="xsd:long">
          <xsd:minInclusive value="0"/>
        </xsd:restriction>
      </xsd:simpleType>
    </xsd:choice>
  </xsd:complexType>
```

```
        <xsd:maxInclusive value="100" />
      </xsd:restriction>
    </xsd:simpleType>
  </xsd:element>
  <xsd:element name="currentGain">
    <xsd:simpleType>
      <xsd:restriction base="xsd:long">
        <xsd:minInclusive value="0" />
        <xsd:maxInclusive value="100" />
      </xsd:restriction>
    </xsd:simpleType>
  </xsd:element>
  <xsd:element name="currentState" type="csta:CurrentState" />
</xsd:choice>
</xsd:complexType>

<xsd:simpleType name="EncodingAlgorithm">
  <xsd:restriction base="xsd:string">
    <xsd:enumeration value="aDPCM6K" />
    <xsd:enumeration value="aDPCM8K" />
    <xsd:enumeration value="muLawPCM6K" />
    <xsd:enumeration value="aLawPCM6K" />
  </xsd:restriction>
</xsd:simpleType>

<xsd:simpleType name="CurrentState">
  <xsd:restriction base="xsd:string">
    <xsd:enumeration value="stop" />
    <xsd:enumeration value="play" />
    <xsd:enumeration value="record" />
    <xsd:enumeration value="suspendPlay" />
    <xsd:enumeration value="suspendRecord" />
    <xsd:enumeration value="review" />
  </xsd:restriction>
</xsd:simpleType>

<xsd:complexType name="ControlData">
  <xsd:sequence>
    <xsd:element name="gender">
      <xsd:simpleType>
        <xsd:restriction base="xsd:string">
          <xsd:enumeration value="male" />
          <xsd:enumeration value="female" />
        </xsd:restriction>
      </xsd:simpleType>
    </xsd:element>
    <xsd:element name="language">
      <xsd:simpleType>
        <xsd:restriction base="xsd:string" />
      </xsd:simpleType>
    </xsd:element>
  </xsd:sequence>
</xsd:complexType>

<xsd:simpleType name="UserData">
  <xsd:restriction base="xsd:hexBinary">
    <xsd:minLength value="0" />
    <xsd:maxLength value="512" />
  </xsd:restriction>
</xsd:simpleType>

<xsd:element name="userData" type="csta:UserData" />

<xsd:simpleType name="TelephonyTone">
  <xsd:restriction base="xsd:string">
    <xsd:enumeration value="beep" />
    <xsd:enumeration value="billing" />
    <xsd:enumeration value="busy" />
  </xsd:restriction>
</xsd:simpleType>
```

```
<xsd:enumeration value="carrier"/>
<xsd:enumeration value="confirmation"/>
<xsd:enumeration value="dial"/>
<xsd:enumeration value="faxCNG"/>
<xsd:enumeration value="hold"/>
<xsd:enumeration value="howler"/>
<xsd:enumeration value="intrusion"/>
<xsd:enumeration value="modemCNG"/>
<xsd:enumeration value="park"/>
<xsd:enumeration value="recordWarning"/>
<xsd:enumeration value="reorder"/>
<xsd:enumeration value="ringback"/>
<xsd:enumeration value="silence"/>
<xsd:enumeration value="sitVC"/>
<xsd:enumeration value="sitIC"/>
<xsd:enumeration value="sitRO"/>
<xsd:enumeration value="sitNC"/>
<xsd:enumeration value="switchSpec0"/>
<xsd:enumeration value="switchSpec1"/>
<xsd:enumeration value="switchSpec2"/>
<xsd:enumeration value="switchSpec3"/>
<xsd:enumeration value="switchSpec4"/>
<xsd:enumeration value="switchSpec5"/>
<xsd:enumeration value="switchSpec6"/>
<xsd:enumeration value="switchSpec7"/>
<xsd:enumeration value="switchSpec8"/>
<xsd:enumeration value="switchSpec9"/>
<xsd:enumeration value="switchSpec10"/>
<xsd:enumeration value="switchSpec11"/>
<xsd:enumeration value="switchSpec12"/>
<xsd:enumeration value="switchSpec13"/>
<xsd:enumeration value="switchSpec14"/>
<xsd:enumeration value="switchSpec15"/>
<xsd:enumeration value="switchSpec16"/>
<xsd:enumeration value="switchSpec17"/>
<xsd:enumeration value="switchSpec18"/>
<xsd:enumeration value="switchSpec19"/>
<xsd:enumeration value="switchSpec20"/>
<xsd:enumeration value="switchSpec21"/>
<xsd:enumeration value="switchSpec22"/>
<xsd:enumeration value="switchSpec23"/>
<xsd:enumeration value="switchSpec24"/>
<xsd:enumeration value="switchSpec25"/>
<xsd:enumeration value="switchSpec26"/>
<xsd:enumeration value="switchSpec27"/>
<xsd:enumeration value="switchSpec28"/>
<xsd:enumeration value="switchSpec29"/>
<xsd:enumeration value="switchSpec30"/>
<xsd:enumeration value="switchSpec31"/>
<xsd:enumeration value="switchSpec32"/>
<xsd:enumeration value="switchSpec33"/>
<xsd:enumeration value="switchSpec34"/>
<xsd:enumeration value="switchSpec35"/>
<xsd:enumeration value="switchSpec36"/>
<xsd:enumeration value="switchSpec37"/>
<xsd:enumeration value="switchSpec38"/>
<xsd:enumeration value="switchSpec39"/>
<xsd:enumeration value="switchSpec40"/>
<xsd:enumeration value="switchSpec41"/>
<xsd:enumeration value="switchSpec42"/>
<xsd:enumeration value="switchSpec43"/>
<xsd:enumeration value="switchSpec44"/>
<xsd:enumeration value="switchSpec45"/>
<xsd:enumeration value="switchSpec46"/>
<xsd:enumeration value="switchSpec47"/>
<xsd:enumeration value="switchSpec48"/>
<xsd:enumeration value="switchSpec49"/>
<xsd:enumeration value="switchSpec50"/>
```

```
<xsd:enumeration value="switchSpec51"/>
<xsd:enumeration value="switchSpec52"/>
<xsd:enumeration value="switchSpec53"/>
<xsd:enumeration value="switchSpec54"/>
<xsd:enumeration value="switchSpec55"/>
<xsd:enumeration value="switchSpec56"/>
<xsd:enumeration value="switchSpec57"/>
<xsd:enumeration value="switchSpec58"/>
<xsd:enumeration value="switchSpec59"/>
<xsd:enumeration value="switchSpec60"/>
<xsd:enumeration value="switchSpec61"/>
<xsd:enumeration value="switchSpec62"/>
<xsd:enumeration value="switchSpec63"/>
<xsd:enumeration value="switchSpec64"/>
<xsd:enumeration value="switchSpec65"/>
<xsd:enumeration value="switchSpec66"/>
<xsd:enumeration value="switchSpec67"/>
<xsd:enumeration value="switchSpec68"/>
<xsd:enumeration value="switchSpec69"/>
<xsd:enumeration value="switchSpec70"/>
<xsd:enumeration value="switchSpec71"/>
<xsd:enumeration value="switchSpec72"/>
<xsd:enumeration value="switchSpec73"/>
<xsd:enumeration value="switchSpec74"/>
<xsd:enumeration value="switchSpec75"/>
<xsd:enumeration value="switchSpec76"/>
<xsd:enumeration value="switchSpec77"/>
<xsd:enumeration value="switchSpec78"/>
<xsd:enumeration value="switchSpec79"/>
<xsd:enumeration value="switchSpec80"/>
<xsd:enumeration value="switchSpec81"/>
<xsd:enumeration value="switchSpec82"/>
<xsd:enumeration value="switchSpec83"/>
<xsd:enumeration value="switchSpec84"/>
<xsd:enumeration value="switchSpec85"/>
<xsd:enumeration value="switchSpec86"/>
<xsd:enumeration value="switchSpec87"/>
<xsd:enumeration value="switchSpec88"/>
<xsd:enumeration value="switchSpec89"/>
<xsd:enumeration value="switchSpec90"/>
<xsd:enumeration value="switchSpec91"/>
<xsd:enumeration value="switchSpec92"/>
<xsd:enumeration value="switchSpec93"/>
<xsd:enumeration value="switchSpec94"/>
<xsd:enumeration value="switchSpec95"/>
<xsd:enumeration value="switchSpec96"/>
<xsd:enumeration value="switchSpec97"/>
<xsd:enumeration value="switchSpec98"/>
<xsd:enumeration value="switchSpec99"/>
<xsd:enumeration value="switchSpec100"/>
</xsd:restriction>
</xsd:simpleType>

</xsd:schema>
```


9.7 Security

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-security</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="empty.xsd"/>

  <xsd:complexType name="CSTASecurityData">
    <xsd:sequence>
      <xsd:element name="messageSequenceNumber" type="xsd:long" minOccurs="0"/>
      <xsd:element name="timeStamp" type="csta:TimeInfo" minOccurs="0"/>
      <xsd:element name="securityInfo" type="csta:SecurityInfo" minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>

  <xsd:complexType name="SecurityInfo">
    <xsd:choice>
      <xsd:element name="string">
        <xsd:simpleType>
          <xsd:restriction base="xsd:hexBinary"/>
        </xsd:simpleType>
      </xsd:element>
      <xsd:element name="private">
        <xsd:annotation>
          <xsd:documentation>private: the actual encoding has to be replaced here</xsd:documentation>
        </xsd:annotation>
        <xsd:complexType>
          <xsd:sequence>
            <xsd:any namespace="##any"/>
          </xsd:sequence>
        </xsd:complexType>
      </xsd:element>
    </xsd:choice>
  </xsd:complexType>

  <xsd:simpleType name="TimeInfo">
    <xsd:restriction base="csta:GeneralizedTime"/>
  </xsd:simpleType>

  <xsd:simpleType name="GeneralizedTime">
    <xsd:restriction base="xsd:string">
      <xsd:pattern value="\d{14}[,\.]?\d*[-]\d{4}" />
      <xsd:pattern value="\d{14}[,\.]?\d*Z" />
      <xsd:pattern value="\d{14}[,\.]?\d*" />
    </xsd:restriction>
  </xsd:simpleType>

</xsd:schema>
```

9.8 Common extensions

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-extension-types</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="security.xsd"/>

  <xsd:complexType name="CSTACCommonArguments">
    <xsd:sequence>
      <xsd:element name="security" type="csta:CSTASecurityData" minOccurs="0"/>
      <xsd:element name="privateData" type="csta:CSTAPrivateData" minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>

  <xsd:complexType name="CSTAPrivateData">
    <xsd:choice>
      <xsd:element name="string">
        <xsd:simpleType>
          <xsd:restriction base="xsd:hexBinary"/>
        </xsd:simpleType>
      </xsd:element>
      <xsd:element name="private">
        <xsd:annotation>
          <xsd:documentation>private: the actual encoding has to be replaced here</xsd:documentation>
        </xsd:annotation>
        <xsd:complexType>
          <xsd:sequence>
            <xsd:any namespace="##any"/>
          </xsd:sequence>
        </xsd:complexType>
      </xsd:element>
    </xsd:choice>
  </xsd:complexType>

  <xsd:element name="extensions" type="csta:CSTACCommonArguments"/>

</xsd:schema>
```

9.9 Call control

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-call-control</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="security.xsd"/>

  <xsd:simpleType name="AutoOriginate">
    <xsd:restriction base="xsd:string">
      <xsd:enumeration value="prompt"/>
      <xsd:enumeration value="doNotPrompt"/>
    </xsd:restriction>
  </xsd:simpleType>

  <xsd:complexType name="CallCharacteristics">
    <xsd:sequence>
      <xsd:element name="acdCall" type="xsd:boolean" minOccurs="0"/>
      <xsd:element name="priorityCall" type="xsd:boolean" minOccurs="0"/>
      <xsd:element name="maintainanceCall" type="xsd:boolean" minOccurs="0"/>
      <xsd:element name="directAgent" type="xsd:boolean" minOccurs="0"/>
      <xsd:element name="assistCall" type="xsd:boolean" minOccurs="0"/>
      <xsd:element name="voiceUnitCall" type="xsd:boolean" minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>

  <xsd:element name="callCharacteristics" type="csta:CallCharacteristics"/>

  <xsd:simpleType name="CallQualifyingData">
    <xsd:restriction base="xsd:hexBinary">
      <xsd:minLength value="0"/>
      <xsd:maxLength value="64"/>
    </xsd:restriction>
  </xsd:simpleType>

  <xsd:simpleType name="ConsultOptions">
    <xsd:restriction base="xsd:string">
      <xsd:enumeration value="unrestricted"/>
      <xsd:enumeration value="consultOnly"/>
      <xsd:enumeration value="transferOnly"/>
      <xsd:enumeration value="conferenceOnly"/>
    </xsd:restriction>
  </xsd:simpleType>

  <xsd:element name="consultOptions" type="csta:ConsultOptions"/>

  <xsd:complexType name="NetworkCapability">
    <xsd:sequence>
      <xsd:element name="networkType">
        <xsd:simpleType>
          <xsd:restriction base="xsd:string">
            <xsd:enumeration value="iSDNPublic"/>
            <xsd:enumeration value="nonISDNPublic"/>
            <xsd:enumeration value="iSDNPrivate"/>
            <xsd:enumeration value="nonISDNPrivate"/>
            <xsd:enumeration value="other"/>
          </xsd:restriction>
        </xsd:simpleType>
      </xsd:element>
      <xsd:element name="eventsProvided" minOccurs="0">
        <xsd:complexType>
```

```
<xsd:sequence>
  <xsd:element name="bridged" type="xsd:boolean" minOccurs="0"/>
  <xsd:element name="callCleared" type="xsd:boolean" minOccurs="0"/>
  <xsd:element name="conferenced" type="xsd:boolean" minOccurs="0"/>
  <xsd:element name="connectionCleared" type="xsd:boolean" minOccurs="0"/>
  <xsd:element name="delivered" type="xsd:boolean" minOccurs="0"/>
  <xsd:element name="digitsDialed" type="xsd:boolean" minOccurs="0"/>
  <xsd:element name="diverted" type="xsd:boolean" minOccurs="0"/>
  <xsd:element name="established" type="xsd:boolean" minOccurs="0"/>
  <xsd:element name="failed" type="xsd:boolean" minOccurs="0"/>
  <xsd:element name="held" type="xsd:boolean" minOccurs="0"/>
  <xsd:element name="networkCapabilitiesChange" type="xsd:boolean" minOccurs="0"/>
  <xsd:element name="networkReached" type="xsd:boolean" minOccurs="0"/>
  <xsd:element name="offered" type="xsd:boolean" minOccurs="0"/>
  <xsd:element name="originated" type="xsd:boolean" minOccurs="0"/>
  <xsd:element name="queued" type="xsd:boolean" minOccurs="0"/>
  <xsd:element name="retrieved" type="xsd:boolean" minOccurs="0"/>
  <xsd:element name="serviceInitiated" type="xsd:boolean" minOccurs="0"/>
  <xsd:element name="transferred" type="xsd:boolean" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>

<xsd:complexType name="ProgressIndicator">
  <xsd:sequence>
    <xsd:element name="progressLocation">
      <xsd:simpleType>
        <xsd:restriction base="xsd:string">
          <xsd:enumeration value="user"/>
          <xsd:enumeration value="privateNetServingLocal"/>
          <xsd:enumeration value="publicNetServingLocal"/>
          <xsd:enumeration value="transitNetwork"/>
          <xsd:enumeration value="publicNetServingRemote"/>
          <xsd:enumeration value="privateNetServingRemote"/>
          <xsd:enumeration value="localInterface"/>
          <xsd:enumeration value="internationalNetwork"/>
          <xsd:enumeration value="networkBeyondInterwk"/>
          <xsd:enumeration value="other"/>
        </xsd:restriction>
      </xsd:simpleType>
    </xsd:element>
    <xsd:element name="progressDescription">
      <xsd:simpleType>
        <xsd:restriction base="xsd:string">
          <xsd:enumeration value="iSDNProgressDesc"/>
          <xsd:enumeration value="qSIGProgressDesc"/>
          <xsd:enumeration value="other"/>
        </xsd:restriction>
      </xsd:simpleType>
    </xsd:element>
  </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="ServicesPermitted">
  <xsd:sequence>
    <xsd:element name="callControlServices" type="csta:CallControlServices"/>
    <xsd:element name="callAssociatedServices" type="csta:CallAssociatedServices"/>
    <xsd:element name="mediaAttachementServices" type="csta:MediaAttachmentServices"/>
    <xsd:element name="routeingServices" type="csta:RouteingServices"/>
    <xsd:element name="voiceUnitServices" type="csta:VoiceUnitServices"/>
  </xsd:sequence>
</xsd:complexType>

<xsd:element name="servicesPermitted" type="csta:ServicesPermitted"/>
```

```
<xsd:complexType name="CallLinkageDataList">
  <xsd:sequence>
    <xsd:element name="newCallLinkageData" type="csta:CallLinkageData"/>
    <xsd:element name="oldCallLinkageData" type="csta:CallLinkageData"/>
  </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="CallLinkageData">
  <xsd:sequence>
    <xsd:element name="globalCallData" type="csta:GlobalCallData"/>
    <xsd:element name="threadData" type="csta:ThreadData" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>

<xsd:element name="callLinkageData" type="csta:CallLinkageData"/>

<xsd:complexType name="GlobalCallData">
  <xsd:sequence>
    <xsd:element name="globalCallSwitchingSubDomainName" minOccurs="0">
      <xsd:simpleType>
        <xsd:restriction base="xsd:string">
          <xsd:minLength value="1"/>
          <xsd:maxLength value="64"/>
        </xsd:restriction>
      </xsd:simpleType>
    </xsd:element>
    <xsd:element name="globalCallLinkageID" type="csta:GlobalCallLinkageID"/>
    <xsd:element name="callLinkageIDTimestamp" type="csta:TimeInfo" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="GlobalCallLinkageID">
  <xsd:choice>
    <xsd:element name="subDomainCallLinkageID">
      <xsd:simpleType>
        <xsd:restriction base="xsd:string">
          <xsd:minLength value="1"/>
          <xsd:maxLength value="8"/>
          <xsd:pattern value="\d*"/>
        </xsd:restriction>
      </xsd:simpleType>
    </xsd:element>
    <xsd:element name="globallyUniqueCallLinkageID">
      <xsd:simpleType>
        <xsd:restriction base="xsd:string">
          <xsd:minLength value="1"/>
          <xsd:maxLength value="16"/>
          <xsd:pattern value="\d*"/>
        </xsd:restriction>
      </xsd:simpleType>
    </xsd:element>
  </xsd:choice>
</xsd:complexType>

<xsd:complexType name="ThreadData">
  <xsd:sequence>
    <xsd:element name="threadSwitchingSubDomainName" minOccurs="0">
      <xsd:simpleType>
        <xsd:restriction base="xsd:string">
          <xsd:minLength value="1"/>
          <xsd:maxLength value="64"/>
        </xsd:restriction>
      </xsd:simpleType>
    </xsd:element>
    <xsd:element name="threadLinkageID" type="csta:ThreadLinkageID"/>
    <xsd:element name="threadIDTimeStamp" type="csta:TimeInfo" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

```
<xsd:complexType name="ThreadLinkageID">
  <xsd:choice>
    <xsd:element name="subDomainThreadID">
      <xsd:simpleType>
        <xsd:restriction base="xsd:string">
          <xsd:minLength value="1"/>
          <xsd:maxLength value="8"/>
          <xsd:pattern value="\d*" />
        </xsd:restriction>
      </xsd:simpleType>
    </xsd:element>
    <xsd:element name="globallyUniqueThreadID">
      <xsd:simpleType>
        <xsd:restriction base="xsd:string">
          <xsd:minLength value="1"/>
          <xsd:maxLength value="16"/>
          <xsd:pattern value="\d*" />
        </xsd:restriction>
      </xsd:simpleType>
    </xsd:element>
  </xsd:choice>
</xsd:complexType>

<xsd:complexType name="CallControlServices">
  <xsd:sequence>
    <xsd:element name="acceptCall" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="alternateCall" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="answerCall" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="callBack" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="callBackMessage" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="campOnCall" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="clearCall" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="clearConnection" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="conferenceCall" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="consultationCall" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="deflectCall" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="dialDigits" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="directedPickupCall" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="groupPickupCall" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="holdCall" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="intrudeCall" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="joinCall" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="makeCall" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="makePredictiveCall" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="parkCall" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="reconnectCall" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="retrieveCall" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="singleStepConference" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="singleStepTransfer" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="transferCall" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="CallAssociatedServices">
  <xsd:sequence>
    <xsd:element name="associateData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="cancelTelephonyTones" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="generateDigits" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="generateTelephonyTones" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="sendUserInformation" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="MediaAttachmentServices">
  <xsd:sequence>
    <xsd:element name="attachMediaService" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="detachMediaService" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

```
</xsd:complexType>

<xsd:complexType name="RouteingServices">
  <xsd:sequence>
    <xsd:element name="routeRegister" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="routeRegisterCancel" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="routeRegisterAbort" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="reRoute" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="routeEnd" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="routeReject" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="routeRequest" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="routeSelect" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="routeUsed" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="VoiceUnitServices">
  <xsd:sequence>
    <xsd:element name="concatenateMessage" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="deleteMessage" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="playMessage" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="queryVoiceAttribute" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="recordMessage" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="reposition" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="resume" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="review" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="setVoiceAttribute" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="stop" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="suspend" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="synthesizeMessage" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>

</xsd:schema>
```

9.10 Capability Exchange

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

```
<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"  
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"  
xmlns:xsd="http://www.w3.org/2001/XMLSchema"  
elementFormDefault="qualified" attributeFormDefault="unqualified">
```

```
  <xsd:annotation>  
    <xsd:documentation>CSTA-capability-exchange</xsd:documentation>  
  </xsd:annotation>
```

```
  <xsd:include schemaLocation="media-services.xsd"/>
```

```
  <xsd:simpleType name="ServiceCrossRefID">  
    <xsd:restriction base="xsd:string">  
      <xsd:minLength value="0"/>  
      <xsd:maxLength value="4"/>  
      <xsd:pattern value="\d*"/>  
    </xsd:restriction>  
  </xsd:simpleType>
```

```
  <xsd:annotation>  
    <xsd:documentation>Capability Bitmaps</xsd:documentation>  
  </xsd:annotation>
```

```
  <xsd:complexType name="LogDevServList">  
    <xsd:sequence>  
      <xsd:element name="callControlServList" type="csta:CallControlServList" minOccurs="0"/>  
      <xsd:element name="callAssociatedServList" type="csta:CallAssociatedServList"  
minOccurs="0"/>  
      <xsd:element name="logicalServList" type="csta:LogicalServList" minOccurs="0"/>  
      <xsd:element name="mediaServList" type="csta:MediaServList" minOccurs="0"/>  
      <xsd:element name="iOservicesServList" type="csta:IOServicesServList" minOccurs="0"/>  
      <xsd:element name="dataCollectionServList" type="csta:DataCollectionServList"  
minOccurs="0"/>  
      <xsd:element name="voiceUnitServList" type="csta:VoiceUnitServList" minOccurs="0"/>  
    </xsd:sequence>  
  </xsd:complexType>
```

```
  <xsd:complexType name="LogDevEvtsList">  
    <xsd:sequence>  
      <xsd:element name="callControlEvtsList" type="csta:CallControlEvtsList" minOccurs="0"/>  
      <xsd:element name="callAssociatedEvtsList" type="csta:CallAssociatedEvtsList"  
minOccurs="0"/>  
      <xsd:element name="logicalEvtsList" type="csta:LogicalEvtsList" minOccurs="0"/>  
      <xsd:element name="mediaEvtsList" type="csta:MediaEvtsList" minOccurs="0"/>  
      <xsd:element name="voiceUnitEvtsList" type="csta:VoiceUnitEvtsList" minOccurs="0"/>  
    </xsd:sequence>  
  </xsd:complexType>
```

```
  <xsd:complexType name="CapExchangeServList">  
    <xsd:sequence>  
      <xsd:element name="getLogicalDeviceInformation" type="csta:GetLogicalDeviceInformationCaps"  
minOccurs="0"/>  
      <xsd:element name="getPhysicalDeviceInformation"  
type="csta:GetPhysicalDeviceInformationCaps" minOccurs="0"/>  
      <xsd:element name="getSwitchingFunctionCaps" type="csta:Empty"/>  
      <xsd:element name="getSwitchingFunctionDevices" type="csta:GetSwitchingFunctionDevicesCaps"  
minOccurs="0"/>  
      <xsd:element name="switchingFunctionDevices" type="csta:SwitchingFunctionDevicesCaps"  
minOccurs="0"/>  
    </xsd:sequence>  
  </xsd:complexType>
```

```
  <xsd:complexType name="GetLogicalDeviceInformationCaps">  
    <xsd:sequence>
```



```
<xsd:element name="privateDataInReq" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="namedDeviceTypeInAck" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="shortFormDeviceIDInAck" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="miscMonitorCapsInAck" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="maxCallBacksInAck" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="maxAutoAnswerRingsInAck" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="maxActiveCallsInAck" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="maxHeldCallsInAck" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="maxFwdSettingsInAck" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="maxDevicesInConfInAck" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="transAndConfSetupParameter" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="transAndConfSetupConsultationInAck" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="transAndConfSetupHoldMakeInAck" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="transAndConfSetupAlternateInAck" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="transAndConfSetupTwoCallsHoldInAck" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="transAndConfSetupTwoCallsConnectedInAck" type="xsd:boolean"
minOccurs="0"/>
<xsd:element name="mediaClassSupportInAck" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="connectionRateListInAck" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="delayToleranceListInAck" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="numberOfChannelsInAck" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="maxChannelBindInAck" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>

<xsd:complexType name="GetPhysicalDeviceInformationCaps">
  <xsd:sequence>
    <xsd:element name="privateDataInReq" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="namedDeviceTypesInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="otherLogicalDeviceListInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="deviceModelNameInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="maxDisplaysInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="maxButtonsInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="maxLampsInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="maxRingPatternsInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="GetSwitchingFunctionDevicesCaps">
  <xsd:sequence>
    <xsd:element name="requestedDeviceID" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="requestedDeviceCategoryACD" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="requestedDeviceCategoryACDGroup" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="requestedDeviceCategoryHuntGroup" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="requestedDeviceCategoryPickGroup" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="requestedDeviceCategoryOtherGroup" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="requestedDeviceCategoryNetwInterface" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="requestedDeviceCategoryPark" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="requestedDeviceCategoryRouteingDevice" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="requestedDeviceCategoryStation" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="requestedDeviceCategoryVoiceUnit" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="requestedDeviceCategoryOther" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="SwitchingFunctionDevicesCaps">
  <xsd:sequence>
    <xsd:element name="segmentID" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="deviceListDeviceCategory" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="deviceListNamedDeviceTypes" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="deviceListDeviceAttributes" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="deviceListDeviceModelName" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

```
</xsd:complexType>

<xsd:complexType name="SystemStatusServList">
  <xsd:sequence>
    <xsd:element name="changeSystemStatusFilter" type="csta:ChangeSystemStatusFilterCaps"
      minOccurs="0"/>
    <xsd:element name="systemRegister" type="csta:SystemRegisterCaps" minOccurs="0"/>
    <xsd:element name="systemStatusRegisterAbort" type="csta:SystemStatusRegisterAbortCaps"
      minOccurs="0"/>
    <xsd:element name="systemStatusRegisterCancel" type="csta:SystemStatusRegisterCancelCaps"
      minOccurs="0"/>
    <xsd:element name="requestSystemStatus" type="csta:RequestSystemStatusCaps" minOccurs="0"/>
    <xsd:element name="systemStatus" type="csta:SystemStatusCaps" minOccurs="0"/>
    <xsd:element name="swFunctionCapsChanged" type="csta:SwFunctionCapsChangedCaps"
      minOccurs="0"/>
    <xsd:element name="swFunctionDevicesChanged" type="csta:SwFunctionDevicesChangedCaps"
      minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="ChangeSystemStatusFilterCaps">
  <xsd:sequence>
    <xsd:element name="requestedStatusFilterInitializing" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="requestedStatusFilterEnabled" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="requestedStatusFilterNormal" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="requestedStatusFilterMessageLost" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="requestedStatusFilterDisabled" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="requestedStatusFilterPartiallyDisabled" type="xsd:boolean"
      minOccurs="0"/>
    <xsd:element name="requestedStatusFilterOverloadImminent" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="requestedStatusFilterOverloadReached" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="requestedStatusFilterOverloadRelieved" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="SystemRegisterCaps">
  <xsd:sequence>
    <xsd:element name="requestTypesSystemStatus" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="requestTypesRequestSystemStatus" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="requestTypesSwitchingFunctionCapsChanged" type="xsd:boolean"
      minOccurs="0"/>
    <xsd:element name="requestTypesSwitchingFuncionDevicesChanged" type="xsd:boolean"
      minOccurs="0"/>
    <xsd:element name="statusFilterInitializing" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="statusFilterEnabled" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="statusFilterNormal" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="statusFilterMessageLost" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="statusFilterDisabled" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="statusFilterPartiallyDisabled" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="statusFilterOverloadImminent" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="statusFilterOverloadReached" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="statusFilterOverloadRelieved" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateDataOctetString" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateDataOctetStringInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="SystemStatusRegisterCancelCaps">
  <xsd:sequence>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

```
<xsd:complexType name="SystemStatusRegisterAbortCaps">
  <xsd:sequence>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="RequestSystemStatusCaps">
  <xsd:sequence>
    <xsd:element name="privateDataInReq" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="systemStatusInitializing" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="systemStatusEnabled" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="systemStatusNormal" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="systemStatusMessageLost" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="systemStatusDisabled" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="systemStatusPartiallyDisabled" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="systemStatusOverloadImminent" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="systemStatusOverloadReached" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="systemStatusOverloadRelieved" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="switchingFunctionSupportsSending" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="switchingFunctionSupportsReceiving" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="SystemStatusCaps">
  <xsd:sequence>
    <xsd:element name="systemStatusInitializing" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="systemStatusEnabled" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="systemStatusNormal" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="systemStatusMessageLost" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="systemStatusDisabled" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="systemStatusPartiallyDisabled" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="systemStatusOverloadImminent" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="systemStatusOverloadReached" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="systemStatusOverloadRelieved" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="switchingFunctionSupportsSending" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="switchingFunctionSupportsReceiving" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="SwFunctionCapsChangedCaps">
  <xsd:sequence>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="SwFunctionDevicesChangedCaps">
  <xsd:sequence>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="MonitoringServList">
  <xsd:sequence>
    <xsd:element name="changeMonitorFilter" type="csta:ChangeMonitorFilterCaps" minOccurs="0"/>
    <xsd:element name="monitorStart" type="csta:MonitorStartCaps" minOccurs="0"/>
    <xsd:element name="monitorStop" type="csta:MonitorStopCaps" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="ChangeMonitorFilterCaps">
  <xsd:sequence>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

```
<xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>

<xsd:complexType name="MonitorStartCaps">
  <xsd:sequence>
    <xsd:element name="monitorObjectDevice" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="monitorObjectCall" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="requestedMonitorFilter" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="monitorTypeCall" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="monitorTypeDevice" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="monitorMediaClassParameter" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="monitorMediaClassAudio" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="monitorMediaClassData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="monitorMediaClassImage" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="monitorMediaClassVoice" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="monitorExistingCallsInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="callIDOnly" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="swDomainDefaultForMonitorTypeIsDevice" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="acceptsMonitorStartWhileOOS" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="MonitorStopCaps">
  <xsd:sequence>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="switchingFunctionSupportsSending" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="switchingFunctionSupportsReceiving" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="SnapshotServList">
  <xsd:sequence>
    <xsd:element name="snapshotCall" type="csta:SnapshotCallCaps" minOccurs="0"/>
    <xsd:element name="snapshotDevice" type="csta:SnapshotDeviceCaps" minOccurs="0"/>
    <xsd:element name="snapshotCallData" type="csta:SnapshotCallDataCaps" minOccurs="0"/>
    <xsd:element name="snapshotDeviceData" type="csta:SnapshotDeviceDataCaps" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="SnapshotCallCaps">
  <xsd:sequence>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="locaConnectionStateInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="mediaServiceInfoListInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="mediaServiceVersionInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="mediaServiceInstanceInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="mediaStreamID" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="connectionInformation" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="mediaCallCharacteristicsInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="callCharacteristicsInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="callingDeviceInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="calledDeviceInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="callIDOnly" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="reportsWithSnapsCallData" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="SnapshotDeviceCaps">
  <xsd:sequence>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="localCallStateCompoundCallState" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="localCallStateSimpleCallState" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="localCallStateUnknown" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="mediaServiceInfoListInAck" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

```
<xsd:element name="mediaServiceVersionInAck" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="mediaServiceInstanceInAck" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="mediaStreamID" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="connectionInformation" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="mediaCallCharacteristicsInAck" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="reportsWithSnapsDeviceData" type="xsd:boolean" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>

<xsd:complexType name="SnapshotCallDataCaps">
  <xsd:sequence>
    <xsd:element name="segmentID" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="localConnectionState" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="mediaServiceInformationList" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="mediaServiceVersion" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="mediaServiceInstance" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="mediaStreamID" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="connectionInformation" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="SnapshotDeviceDataCaps">
  <xsd:sequence>
    <xsd:element name="segmentID" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="localCallStateCompoundCallState" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="localCallStateSimpleCallState" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="localCallStateUnknown" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="mediaServiceInformationList" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="mediaServiceVersion" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="mediaServiceInstance" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="mediaStreamID" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="connectionInformation" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="mediaCallCharacteristics" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="CallControlServList">
  <xsd:sequence>
    <xsd:element name="acceptCall" type="csta:AcceptCallCaps" minOccurs="0"/>
    <xsd:element name="alternateCall" type="csta:AlternateCallCaps" minOccurs="0"/>
    <xsd:element name="answerCall" type="csta:AnswerCallCaps" minOccurs="0"/>
    <xsd:element name="callBack" type="csta:CallBackCaps" minOccurs="0"/>
    <xsd:element name="callBackMessage" type="csta:CallBackMessageCaps" minOccurs="0"/>
    <xsd:element name="campOnCall" type="csta:CampOnCallCaps" minOccurs="0"/>
    <xsd:element name="clearCall" type="csta:ClearCallCaps" minOccurs="0"/>
    <xsd:element name="clearConnection" type="csta:ClearConnectionCaps" minOccurs="0"/>
    <xsd:element name="conferenceCall" type="csta:ConferenceCallCaps" minOccurs="0"/>
    <xsd:element name="consultationCall" type="csta:ConsultationCallCaps" minOccurs="0"/>
    <xsd:element name="deflectCall" type="csta:DeflectCallCaps" minOccurs="0"/>
    <xsd:element name="dialDigits" type="csta:DialDigitsCaps" minOccurs="0"/>
    <xsd:element name="directedPickupCall" type="csta:DirectedPickupCallCaps" minOccurs="0"/>
    <xsd:element name="groupPickupCall" type="csta:GroupPickupCallCaps" minOccurs="0"/>
    <xsd:element name="holdCall" type="csta:HoldCallCaps" minOccurs="0"/>
    <xsd:element name="intrudeCall" type="csta:IntrudeCallCaps" minOccurs="0"/>
    <xsd:element name="joinCall" type="csta:JoinCallCaps" minOccurs="0"/>
    <xsd:element name="makeCall" type="csta:MakeCallCaps" minOccurs="0"/>
    <xsd:element name="makePredictiveCall" type="csta:MakePredictiveCallCaps" minOccurs="0"/>
    <xsd:element name="parkCall" type="csta:ParkCallCaps" minOccurs="0"/>
    <xsd:element name="reconnectCall" type="csta:ReconnectCallCaps" minOccurs="0"/>
    <xsd:element name="retrieveCall" type="csta:RetrieveCallCaps" minOccurs="0"/>
    <xsd:element name="singleStepConference" type="csta:SingleStepConferenceCaps"
      minOccurs="0"/>
    <xsd:element name="singleStepTransfer" type="csta:SingleStepTransferCaps" minOccurs="0"/>
    <xsd:element name="transferCall" type="csta:TransferCallCaps" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

```
</xsd:complexType>

<xsd:complexType name="AcceptCallCaps">
  <xsd:sequence>
    <xsd:element name="correlatorData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="userData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="deviceIDOnly" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="ackModelMultiStep" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="AlternateCallCaps">
  <xsd:sequence>
    <xsd:element name="alerting" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="hold" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="queued" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="connectionReservation" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="consultOptionConsultOnly" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="consultOptionTransferOnly" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="consultOptionConferenceOnly" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="consultOptionUnrestricted" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="deviceIDOnly" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="ackModelMultiStep" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="supportsOfferedModeOfAlerting" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="AnswerCallCaps">
  <xsd:sequence>
    <xsd:element name="alerting" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="initiated" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="queued" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="correlatorData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="userData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="deviceIDOnly" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="supportsOfferedModeOfAlerting" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="ackModelMultiStep" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="CallBackCaps">
  <xsd:sequence>
    <xsd:element name="alerting" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="null" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="failed" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="queued" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="callCharacteristics" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="targetDeviceInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="deviceIDOnly" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="moreCallBacksNegAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="ackModelMultiStep" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="CallBackMessageCaps">
  <xsd:sequence>
    <xsd:element name="alerting" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="null" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="failed" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="queued" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="targetDeviceInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="deviceIDOnly" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

```
<xsd:element name="moreCallbacksNegAck" type="xsd:boolean" minOccurs="0"/>
  <xsd:element name="ackModelMultiStep" type="xsd:boolean" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
<xsd:complexType name="CampOnCallCaps">
  <xsd:sequence>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="deviceIDOnly" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="ackModelMultiStep" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="ClearCallCaps">
  <xsd:sequence>
    <xsd:element name="alerting" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="connected" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="failed" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="queued" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="initiated" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="hold" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="userData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="deviceIDOnly" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="callIDOnly" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="ackModelMultiStep" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="ClearConnectionCaps">
  <xsd:sequence>
    <xsd:element name="alerting" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="connected" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="fail" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="queued" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="initiated" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="hold" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="correlatorData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="userData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="deviceIDOnly" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="ackModelMultiStep" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="ConferenceCallCaps">
  <xsd:sequence>
    <xsd:element name="activeCallConnected" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="activeCallHold" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="heldCallConnected" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="heldCallHold" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="connectionParameterInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="endpointDeviceID" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="resultingConnectionInfo" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="conferenceCallOnfoInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="protectedAgainstClearing" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="deviceIDOnly" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="ackModelMultiStep" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="ConsultationCallCaps">
  <xsd:sequence>
    <xsd:element name="connectionReservation" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="accountCode" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="authCode" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="correlatorData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="userData" type="xsd:boolean" minOccurs="0"/>
```

```
<xsd:element name="callCharacteristics" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="callCharacteristicsACDCall" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="callCharacteristicsPriorityCall" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="callCharacteristicsMaintenanceCall" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="callCharacteristicsDirectAgent" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="callCharacteristicsAssistCall" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="callCharacteristicsVoiceUnitCall" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="mediaCallCharacteristics" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="callingConnectionInfo" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="flowDirectionTransmit" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="flowDirectionReceive" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="flowDirectionTransmitAndReceive" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="numberOfChannels" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="consultOptionConsultOnly" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="consultOptionTransferOnly" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="consultOptionConferenceOnly" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="consultOptionUnrestricted" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="initiatedCallInfoInAck" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="deviceIDOnly" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="multiStage" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="supportsAdjustmentOfMediaCharacteristics" type="xsd:boolean"
minOccurs="0"/>
<xsd:element name="ackModelMultiStep" type="xsd:boolean" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
<xsd:complexType name="DeflectCallCaps">
  <xsd:sequence>
    <xsd:element name="alerting" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="connected" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="failed" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="hold" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="queued" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="correlatorData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="userData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="deviceIDOnly" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="ackModelMultiStep" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="DialDigitsCaps">
  <xsd:sequence>
    <xsd:element name="correlatorData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="deviceIDOnly" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="ackModelMultiStep" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="DirectedPickupCallCaps">
  <xsd:sequence>
    <xsd:element name="alerting" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="hold" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="queued" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="connected" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="correlatorData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="userData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="pickedCallInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="pickedCallInfoInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="deviceIDOnly" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="supportsOfferedModeOfAlerting" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="supportsPrompting" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="promptingMode" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="ackModelMultiStep" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```



```
</xsd:sequence>
</xsd:complexType>
<xsd:complexType name="GroupPickupCallCaps">
  <xsd:sequence>
    <xsd:element name="alerting" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="connected" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="hold" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="queued" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="pickGroup" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="correlatorData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="userData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="pickedCallInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="pickedCallInfoInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="supportsPrompting" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="promptingMode" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="supportsOfferedModeOfAlerting" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="ackModelMultiStep" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="HoldCallCaps">
  <xsd:sequence>
    <xsd:element name="connectionReservation" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="deviceIDOnly" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="ackModelMultiStep" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="IntrudeCallCaps">
  <xsd:sequence>
    <xsd:element name="participationTypeSilent" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="participationTypeActive" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="userData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="conferencedCallInfoInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="deviceIDOnly" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="supportsConference" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="supportsAlternate" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="ackModelMultiStep" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="JoinCallCaps">
  <xsd:sequence>
    <xsd:element name="autoOriginatePrompt" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="autoOriginateDoNotPrompt" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="participationTypeSilent" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="participationTypeActive" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="accountCode" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="authCode" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="correlatorData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="userData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="conferencedCallInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="conferencedCallInfoInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="deviceIDOnly" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="supportsPrompting" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="promptingMode" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="ackModelMultiStep" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="MakeCallCaps">
  <xsd:sequence>
    <xsd:element name="initiated" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="null" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="accountCode" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

```
<xsd:element name="authCode" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="autoOriginatePrompt" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="autoOriginateDoNotPrompt" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="correlatorData" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="userData" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="callCharacteristics" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="callCharacteristicsACDCall" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="callCharacteristicsPriorityCall" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="callCharacteristicsMaintenanceCall" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="callCharacteristicsDirectAgent" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="callCharacteristicsAssistCall" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="callCharacteristicsVoiceUnitCall" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="mediaCallCharacteristics" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="callingConnectionInfo" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="initiatedCallInfoInAck" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="multiStage" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="supportsPrompting" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="promptingMode" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="offHook" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="mediaCharacteristicsAdjustable" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="ackModelMultiStep" type="xsd:boolean" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
<xsd:complexType name="MakePredictiveCallCaps">
  <xsd:sequence>
    <xsd:element name="signallingDetection" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="signallingConditionCallDelivered" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="signallingConditionCallEstablished" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="signallingConditionActionDestinationDetection" type="xsd:boolean"
minOccurs="0"/>
    <xsd:element name="signallingConditionActionRemainConnected" type="xsd:boolean"
minOccurs="0"/>
    <xsd:element name="destinationDetection" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="destinationConditionHumanVoice" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="destinationConditionAnsweringMachine" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="destinationConditionFax" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="destinationActionClearConnection" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="destinationActionRemainConnected" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="defaultActionClearConnection" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="defaultActionRemainConnected" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="accountCode" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="authCode" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="autoOriginatePrompt" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="autoOriginateDoNotPrompt" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="alertTime" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="correlatorData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="callCharacteristics" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="callCharacteristicsACDCall" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="callCharacteristicsPriorityCall" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="callCharacteristicsMaintenanceCall" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="callCharacteristicsDirectAgent" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="callCharacteristicsAssistCall" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="callCharacteristicsVoiceUnitCall" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="userData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="initiatedCallInfoInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="deviceIDOnly" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="supportsPrompting" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="promptingMode" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="reservesCallingDevice" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="ackModelMultiStep" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="ParkCallCaps">
  <xsd:sequence>
```

```
<xsd:element name="hold" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="connected" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="correlatorData" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="parkedToInAck" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="deviceIDOnly" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="ackModelMultiStep" type="xsd:boolean" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
<xsd:complexType name="ReconnectCallCaps">
  <xsd:sequence>
    <xsd:element name="alerting" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="connected" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="fail" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="initiated" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="queued" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="deviceIDOnly" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="ackModelMultiStep" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="RetrieveCallCaps">
  <xsd:sequence>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="deviceIDOnly" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="ackModelMultiStep" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="SingleStepConferenceCaps">
  <xsd:sequence>
    <xsd:element name="participationTypeActive" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="participationTypeSilent" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="accountCode" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="authCode" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="correlatorData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="userData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="conferencedCallInfoInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="deviceIDOnly" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="ackModelMultiStep" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="SingleStepTransferCaps">
  <xsd:sequence>
    <xsd:element name="accountCode" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="authCode" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="correlatorData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="userData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="connectionParameterInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="endpointDeviceID" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="resolutionConnectionInformation" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="transferredCollInfoInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="deviceIDOnly" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="multipleDevices" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="ackModelMultiStep" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="TransferCallCaps">
  <xsd:sequence>
    <xsd:element name="activeCallConnected" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="activeCallHold" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="heldCallConnected" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

```
<xsd:element name="heldCallHold" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="connectionsParameterInAck" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="endpointDeviceID" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="resultingConnectionInformation" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="transferredCallInfoInAck" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="deviceIDOnly" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="multipleDevices" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="ackModelMultiStep" type="xsd:boolean" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
<xsd:complexType name="CallControlEvtsList">
  <xsd:sequence>
    <xsd:element name="bridged" type="csta:BridgedCaps" minOccurs="0"/>
    <xsd:element name="callCleared" type="csta:CallClearedCaps" minOccurs="0"/>
    <xsd:element name="conferenced" type="csta:ConferencedCaps" minOccurs="0"/>
    <xsd:element name="connectionCleared" type="csta:ConnectionClearedCaps" minOccurs="0"/>
    <xsd:element name="delivered" type="csta:DeliveredCaps" minOccurs="0"/>
    <xsd:element name="digitsDialed" type="csta:DigitsDialedCaps" minOccurs="0"/>
    <xsd:element name="diverted" type="csta:DivertedCaps" minOccurs="0"/>
    <xsd:element name="established" type="csta:EstablishedCaps" minOccurs="0"/>
    <xsd:element name="failed" type="csta:FailedCaps" minOccurs="0"/>
    <xsd:element name="held" type="csta:HeldCaps" minOccurs="0"/>
    <xsd:element name="netwCapsChanged" type="csta:NetwCapsChangedCaps" minOccurs="0"/>
    <xsd:element name="netwReached" type="csta:NetwReachedCaps" minOccurs="0"/>
    <xsd:element name="offered" type="csta:OfferedCaps" minOccurs="0"/>
    <xsd:element name="originated" type="csta:OriginatedCaps" minOccurs="0"/>
    <xsd:element name="queued" type="csta:QueuedCaps" minOccurs="0"/>
    <xsd:element name="retrieved" type="csta:RetrievedCaps" minOccurs="0"/>
    <xsd:element name="serviceInitiated" type="csta:ServiceInitiatedCaps" minOccurs="0"/>
    <xsd:element name="transferred" type="csta:TransferredCaps" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="BridgedCaps">
  <xsd:sequence>
    <xsd:element name="correlatorData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="userData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="servicesPermitted" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="mediaCallCharacteristics" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="callCharacteristics" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="bridgedConnectionInfo" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="callLinkageData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="CallClearedCaps">
  <xsd:sequence>
    <xsd:element name="correlatorData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="userData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="mediaCallCharacteristics" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="callCharacteristics" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="callLinkageData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="callIDOnly" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="ConferencedCaps">
  <xsd:sequence>
    <xsd:element name="confereceConnectionsEndpointDeviceID" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="confereceConnectionsresultingConnectionInfo" type="xsd:boolean"
    minOccurs="0"/>
    <xsd:element name="userData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="servicesPermitted" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="mediaCallCharacteristics" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="callCharacteristics" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="bridgedConnectionInfo" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

```
</xsd:sequence>
</xsd:complexType>
<xsd:complexType name="ConnectionClearedCaps">
  <xsd:sequence>
    <xsd:element name="correlatorData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="userData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="chargingInfo" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="numberUnitsNumberOfChargingUnits" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="numberUnitsTypeOfUnits" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="numberUnitsNumberOfCurrencyUnits" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="typeOfChargingInfoSubTotal" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="typeOfChargingInfoTotal" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="chargingMultiplierATHousandth" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="chargingMultiplierAHundredth" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="chargingMultiplierATenth" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="chargingMultiplierOne" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="chargingMultiplierTen" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="chargingMultiplierHundred" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="chargingMultiplierThousand" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="servicesPermitted" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="mediaCallCharacteristics" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="callCharacteristics" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="droppedConnectionInfo" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="callLinkageData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="DeliveredCaps">
  <xsd:sequence>
    <xsd:element name="originatingNIDConnection" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="userData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="servicesPermitted" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="netwCallingDevice" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="netwCalledDevice" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="mediaCallCharacteristics" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="callCharacteristics" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="connectionInfo" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="DigitsDialedCaps">
  <xsd:sequence>
    <xsd:element name="servicesPermitted" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="netwCallingDevice" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="netwCalledDevice" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="diallingConnectionInfo" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="callCharacteristics" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="DivertedCaps">
  <xsd:sequence>
    <xsd:element name="callingDevice" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="calledDevice" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="userData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="servicesPermitted" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="mediaCallCharacteristics" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="callCharacteristics" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="connectionInfo" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="netwCallingDevice" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="netwCalledDevice" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="sendsDivertedToAll" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="EstablishedCaps">
  <xsd:sequence>
    <xsd:element name="originatingNIDConnection" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

```
<xsd:element name="userData" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="servicesPermitted" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="netwCallingDevice" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="netwCalledDevice" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="mediaCallCharacteristics" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="callCharacteristics" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="establishedConnectionInfo" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
<xsd:complexType name="FailedCaps">
  <xsd:sequence>
    <xsd:element name="originatingNIDConnection" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="userData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="servicesPermitted" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="netwCallingDevice" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="netwCalledDevice" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="mediaCallCharacteristics" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="callCharacteristics" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="failedConnectionInfo" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="callIDOnly" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="HeldCaps">
  <xsd:sequence>
    <xsd:element name="correlatorData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="servicesPermitted" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="mediaCallCharacteristics" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="callCharacteristics" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="heldConnectionInfo" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="callLinkageData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="NetwCapsChangedCaps">
  <xsd:sequence>
    <xsd:element name="progressLocationUser" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="progressLocationPrivateServLocal" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="progressLocationPublicServLocal" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="progressLocationTransit" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="progressLocationPublicServRemote" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="progressLocationPrivateServRemote" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="progressLocationLocal" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="progressLocationInternational" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="progressLocationNetwBeyondInterworking" type="xsd:boolean"
minOccurs="0"/>
    <xsd:element name="progressLocationOther" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="progressDescriptionISDN" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="progressDescriptionQSIG" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="progressDescriptionOther" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="userData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="typeOfNetworkISDNPublic" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="typeOfNetworkNonISDNPublic" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="typeOfNetworkISDNPrivate" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="typeOfNetworkNonISDNPrivate" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="typeOfNetworkOther" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="eventsProvidedParameter" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="eventsProvidedBridged" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="eventsProvidedCallCleared" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="eventsProvidedConferenced" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="eventsProvidedConnectionCleared" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="eventsProvidedDelivered" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="eventsProvidedDigitsDialed" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="eventsProvidedDiverted" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="eventsProvidedEstablished" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="eventsProvidedFailed" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="eventsProvidedHeld" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

```
<xsd:element name="eventsProvidedNetwCapsChanged" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="eventsProvidedNetwReached" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="eventsProvidedOffered" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="eventsProvidedOriginated" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="eventsProvidedQueued" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="eventsProvidedRetrieved" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="eventsProvidedServiceInitiated" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="eventsProvidedTransferred" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="servicesPermitted" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="mediaCallCharacteristics" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="callCharacteristics" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="outboundConnectionInfo" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
<xsd:complexType name="NetwReachedCaps">
  <xsd:sequence>
    <xsd:element name="originatingNIDConnection" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="userData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="typeOfNetworkISDNPublic" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="typeOfNetworkNonISDNPublic" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="typeOfNetworkISDNPrivate" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="typeOfNetworkNonISDNPrivate" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="typeOfNetworkOther" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="eventsProvidedParameter" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="eventsProvidedBridged" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="eventsProvidedCallCleared" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="eventsProvidedConferenced" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="eventsProvidedConnectionCleared" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="eventsProvidedDelivered" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="eventsProvidedDigitsDialed" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="eventsProvidedDiverted" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="eventsProvidedEstablished" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="eventsProvidedFailed" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="eventsProvidedHeld" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="eventsProvidedNetwCapsChanged" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="eventsProvidedNetwReached" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="eventsProvidedOffered" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="eventsProvidedOriginated" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="eventsProvidedQueued" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="eventsProvidedRetrieved" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="eventsProvidedServiceInitiated" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="eventsProvidedTransferred" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="servicesPermitted" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="mediaCallCharacteristics" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="callCharacteristics" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="outboundConnectionInfo" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="netwCallingDevice" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="netwCalledDevice" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="OfferedCaps">
  <xsd:sequence>
    <xsd:element name="originatingNIDConnection" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="userData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="servicesPermitted" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="netwCallingDevice" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="netwCalledDevice" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="mediaCallCharacteristics" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="callCharacteristics" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="offeredConnectionInfo" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="OriginatedCaps">
  <xsd:sequence>
    <xsd:element name="originatingDevice" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

```
<xsd:element name="servicesPermitted" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="netwCallingDevice" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="netwCalledDevice" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="mediaCallCharacteristics" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="callCharacteristics" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="originatedConnectionInfo" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
<xsd:complexType name="QueuedCaps">
  <xsd:sequence>
    <xsd:element name="numberQueued" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="callsInFront" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="userData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="servicesPermitted" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="netwCallingDevice" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="netwCalledDevice" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="mediaCallCharacteristics" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="callCharacteristics" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="queuedConnectionInfo" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="RetrievedCaps">
  <xsd:sequence>
    <xsd:element name="correlatorData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="servicesPermitted" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="mediaCallCharacteristics" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="callCharacteristics" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="retrievedConnectionInfo" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="callLinkageData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="ServiceInitiatedCaps">
  <xsd:sequence>
    <xsd:element name="servicesPermitted" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="mediaCallCharacteristics" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="callCharacteristics" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="initiatedConnectionInfo" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="netwCallingDevice" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="netwCalledDevice" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="TransferredCaps">
  <xsd:sequence>
    <xsd:element name="transferredConnectionsEndpointDeviceID" type="xsd:boolean"
minOccurs="0"/>
    <xsd:element name="transferredConnectionsResultingConnectionInfo" type="xsd:boolean"
minOccurs="0"/>
    <xsd:element name="userData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="chargingInfo" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="numberUnitsNumberOfChargingUnits" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="numberUnitsTypeOfUnits" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="numberUnitsNumberOfCurrencyUnits" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="typeOfChargingInfoSubTotal" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="typeOfChargingInfoTotal" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="chargingMultiplierAThousandth" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="chargingMultiplierAHundredth" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="chargingMultiplierATenth" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="chargingMultiplierOne" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="chargingMultiplierTen" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="chargingMultiplierHundred" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="chargingMultiplierThousand" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="servicesPermitted" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="mediaCallCharacteristics" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="callCharacurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```



```
<xsd:element name="connectionInfo" type="xsd:boolean" minOccurs="0"/>
  <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
<xsd:complexType name="CallAssociatedServList">
  <xsd:sequence>
    <xsd:element name="associateData" type="csta:AssociateDataCaps" minOccurs="0"/>
    <xsd:element name="cancelTelephonyTones" type="csta:CancelTelephonyTonesCaps"
      minOccurs="0"/>
    <xsd:element name="generateDigits" type="csta:GenerateDigitsCaps" minOccurs="0"/>
    <xsd:element name="generateTelephonyTones" type="csta:GenerateTelephonyTonesCaps"
      minOccurs="0"/>
    <xsd:element name="sendUserInformation" type="csta:SendUserInformationCaps" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="AssociateDataCaps">
  <xsd:sequence>
    <xsd:element name="accountCode" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="authCode" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="correlatorData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="callQualifyingData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="deviceIDOnly" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="rejectsRequestsWithOldConnectionID" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="ackModelMultiStep" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="CancelTelephonyTonesCaps">
  <xsd:sequence>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="ackModelMultiStep" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="GenerateDigitsCaps">
  <xsd:sequence>
    <xsd:element name="digitModeDTMF" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="digitModePulse" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="toneDuration" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="pulseRate" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="pauseDuration" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="deviceIDOnly" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="supportsDTMFTonesABCD" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="supportsPauseToneChar" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="ackModelMultiStep" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="GenerateTelephonyTonesCaps">
  <xsd:sequence>
    <xsd:element name="toneToSendBeep" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="toneToSendBilling" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="toneToSendBusy" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="toneToSendCarrier" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="toneToSendConfirmation" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="toneToSendDial" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="toneToSendFaxCNG" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="toneToSendHold" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="toneToSendHowler" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="toneToSendIntrusion" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="toneToSendModemCNG" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="toneToSendPark" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="toneToSendRecordWarning" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="toneToSendReorder" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="toneToSendRingback" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="toneToSendSilence" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```



```
<xsd:element name="toneToSendSf64" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="toneToSendSf65" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="toneToSendSf66" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="toneToSendSf67" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="toneToSendSf68" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="toneToSendSf69" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="toneToSendSf70" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="toneToSendSf71" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="toneToSendSf72" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="toneToSendSf73" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="toneToSendSf74" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="toneToSendSf75" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="toneToSendSf76" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="toneToSendSf77" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="toneToSendSf78" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="toneToSendSf79" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="toneToSendSf80" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="toneToSendSf81" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="toneToSendSf82" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="toneToSendSf83" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="toneToSendSf84" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="toneToSendSf85" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="toneToSendSf86" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="toneToSendSf87" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="toneToSendSf88" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="toneToSendSf89" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="toneToSendSf90" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="toneToSendSf91" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="toneToSendSf92" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="toneToSendSf93" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="toneToSendSf94" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="toneToSendSf95" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="toneToSendSf96" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="toneToSendSf97" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="toneToSendSf98" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="toneToSendSf99" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="toneToSendSf100" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="toneDuration" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="deviceIDOnly" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="ackModelMultiStep" type="xsd:boolean" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
<xsd:complexType name="SendUserInformationCaps">
  <xsd:sequence>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="deviceIDOnly" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="ackModelMultiStep" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="CallAssociatedEvtsList">
  <xsd:sequence>
    <xsd:element name="callInformation" type="csta:CallInformationCaps" minOccurs="0"/>
    <xsd:element name="charging" type="csta:ChargingCaps" minOccurs="0"/>
    <xsd:element name="digitsGenerated" type="csta:DigitsGeneratedCaps" minOccurs="0"/>
    <xsd:element name="telephonyTonesGenerated" type="csta:TelephonyTonesGeneratedCaps"
      minOccurs="0"/>
    <xsd:element name="serviceCompletionFailure" type="csta:ServiceCompletionFailureCaps"
      minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="CallInformationCaps">
  <xsd:sequence>
    <xsd:element name="callingDevice" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="accountInfo" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="authorisatinonCode" type="xsd:boolean" minOccurs="0"/>
```

```
<xsd:element name="correlatorData" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="servicesPermitted" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="userData" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="callQualifyingData" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="connectionInfo" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="callLinkageData" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="genCallInfoForOutdatedConnID" type="xsd:boolean" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
<xsd:complexType name="ChargingCaps">
  <xsd:sequence>
    <xsd:element name="numberUnitsNumberOfChargingUnits" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="numberUnitsTypeOfUnits" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="numberUnitsNumberOfCurrencyUnits" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="typeOfChargingInfoSubTotal" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="typeOfChargingInfoTotal" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="chargingMultiplierATHousandth" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="chargingMultiplierAHundredth" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="chargingMultiplierATenth" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="chargingMultiplierOne" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="chargingMultiplierTen" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="chargingMultiplierHundred" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="chargingMultiplierThousand" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="DigitsGeneratedCaps">
  <xsd:sequence>
    <xsd:element name="digitsDurationList" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="pauseDurationList" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="connectionInfo" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="TelephonyTonesGeneratedCaps">
  <xsd:sequence>
    <xsd:element name="toneToSendBeep" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="toneToSendBilling" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="toneToSendBusy" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="toneToSendCarrier" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="toneToSendConfirmation" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="toneToSendDial" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="toneToSendFaxCNG" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="toneToSendHold" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="toneToSendHowler" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="toneToSendIntrusion" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="toneToSendModemCNG" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="toneToSendPark" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="toneToSendRecordWarning" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="toneToSendReorder" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="toneToSendRingback" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="toneToSendSilence" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="toneToSendSitVC" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="toneToSendSitIC" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="toneToSendSitRO" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="toneToSendSitNC" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="toneToSendSf0" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="toneToSendSf1" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="toneToSendSf2" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="toneToSendSf3" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="toneToSendSf4" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="toneToSendSf5" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="toneToSendSf6" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="toneToSendSf7" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="toneToSendSf8" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="toneToSendSf9" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="toneToSendSf10" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```



```
<xsd:element name="toneToSendSf79" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="toneToSendSf80" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="toneToSendSf81" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="toneToSendSf82" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="toneToSendSf83" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="toneToSendSf84" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="toneToSendSf85" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="toneToSendSf86" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="toneToSendSf87" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="toneToSendSf88" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="toneToSendSf89" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="toneToSendSf90" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="toneToSendSf91" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="toneToSendSf92" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="toneToSendSf93" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="toneToSendSf94" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="toneToSendSf95" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="toneToSendSf96" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="toneToSendSf97" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="toneToSendSf98" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="toneToSendSf99" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="toneToSendSf100" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="toneFrequency" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="toneDuration" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="pauseDurationList" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="connectionInfo" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
<xsd:complexType name="ServiceCompletionFailureCaps">
  <xsd:sequence>
    <xsd:element name="primaryCallConnectionInfo" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="secondaryCallConnectionInfo" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="otherDevicesPrimaryCallList" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="otherDevicesSecondaryCallList" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="mediaCallCharacteristics" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="MediaServList">
  <xsd:sequence>
    <xsd:element name="attachMediaService" type="csta:AttachMediaServiceCaps" minOccurs="0"/>
    <xsd:element name="detachMediaService" type="csta:DetachMediaServiceCaps" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="AttachMediaServiceCaps">
  <xsd:sequence>
    <xsd:element name="mediaServiceVersion" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="mediaServiceInstanceID" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="connectionModeConsultConference" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="connectionModeConsultConferenceHold" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="connectionModeDeflect" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="connectionModeDirectedPickup" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="connectionModeJoin" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="connectionModeSingleStepConference" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="connectionModeSingleStepConferenceHold" type="xsd:boolean"
minOccurs="0"/>
    <xsd:element name="connectionModeSingleStepTransfer" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="connectionModeTransfer" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="connectionModeDirect" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="requestedConnectionState" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="mediaServiceInstanceIDInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="mediaConnectionInfoInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="deviceIDOnly" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="ackModelMultiStep" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

```
</xsd:complexType>
<xsd:complexType name="DetachMediaServiceCaps">
  <xsd:sequence>
    <xsd:element name="alerting" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="connected" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="fail" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="hold" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="queued" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="deviceIDOnly" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="ackModelMultiStep" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="MediaEvtsList">
  <xsd:sequence>
    <xsd:element name="mediaAttached" type="csta:MediaAttachedCaps" minOccurs="0"/>
    <xsd:element name="mediaDetached" type="csta:MediaDetachedCaps" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="MediaAttachedCaps">
  <xsd:sequence>
    <xsd:element name="mediaServiceVersion" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="mediaServiceInstanceID" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="mediaStreamID" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="mediaCallCharacteristics" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="callCharacteristics" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="mediaConnectionInfo" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="MediaDetachedCaps">
  <xsd:sequence>
    <xsd:element name="mediaServiceVersion" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="mediaServiceInstanceID" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="mediaStreamID" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="mediaCallCharacteristics" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="callCharacteristics" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="mediaConnectionInfo" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="RouteingServList">
  <xsd:sequence>
    <xsd:element name="routeRegister" type="csta:RouteRegisterCaps" minOccurs="0"/>
    <xsd:element name="routeRegisterCancel" type="csta:RouteRegisterCancelCaps" minOccurs="0"/>
    <xsd:element name="routeRegisterAbort" type="csta:RouteRegisterAbortCaps" minOccurs="0"/>
    <xsd:element name="reRoute" type="csta:ReRouteCaps" minOccurs="0"/>
    <xsd:element name="routeEnd" type="csta:RouteEndCaps" minOccurs="0"/>
    <xsd:element name="routeReject" type="csta:RouteRejectCaps" minOccurs="0"/>
    <xsd:element name="routeRequest" type="csta:RouteRequestCaps" minOccurs="0"/>
    <xsd:element name="routeSelect" type="csta:RouteSelectCaps" minOccurs="0"/>
    <xsd:element name="routeUsed" type="csta:RouteUsedCaps" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="RouteRegisterCaps">
  <xsd:sequence>
    <xsd:element name="routeingDevice" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="requestedMonitorMediaClass" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="requestedMonitorMediaClassAudio" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="requestedMonitorMediaClassData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="requestedMonitorMediaClassImage" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="requestedMonitorMediaClassVoice" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="actualRouteingMediaClassInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="allRouteingDevices" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

```
</xsd:complexType>
<xsd:complexType name="RouteRegisterAbortCaps">
  <xsd:sequence>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="RouteRegisterCancelCaps">
  <xsd:sequence>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="ReRouteCaps">
  <xsd:sequence>
    <xsd:element name="replyTimeout" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="correlatorData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="RouteEndCaps">
  <xsd:sequence>
    <xsd:element name="errorValue" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="correlatorData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="supportsSending" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="supportsReceiving" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="RouteRejectCaps">
  <xsd:sequence>
    <xsd:element name="rejectCauseBusyOverflow" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="rejectCauseQueueTimeOverflow" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="rejectCauseCapacityOverflow" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="rejectCauseCalendarOverflow" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="rejectCauseUnknownOverflow" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="correlatorData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="RouteRequestCaps">
  <xsd:sequence>
    <xsd:element name="callingDevice" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="routeingDevice" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="routeSelAlgorithmACD" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="routeSelAlgorithmEmergency" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="routeSelAlgorithmLeastCost" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="routeSelAlgorithmNormal" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="routeSelAlgorithmUserDefined" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="priority" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="replyTimeout" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="correlatorData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="mediaCallCharacteristics" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="callCharacteristics" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="routedCallInfo" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="nonCallRelatedRouteing" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="RouteSelectCaps">
  <xsd:sequence>
    <xsd:element name="alternateRoutes" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="remainRetriesNoListAvailable" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="remainRetriesNoCountAvailable" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="remainRetriesRetryCount" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="routeUsed" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="correlatorData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```



```
</xsd:complexType>
<xsd:complexType name="RouteUsedCaps">
  <xsd:sequence>
    <xsd:element name="callingDevice" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="domain" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="correlatorData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="PhysDevServList">
  <xsd:sequence>
    <xsd:element name="buttonPress" type="csta:ButtonPressCaps" minOccurs="0"/>
    <xsd:element name="getAuditoryApparatusInfo" type="csta:GetAuditoryApparatusInfoCaps"
minOccurs="0"/>
    <xsd:element name="getButtonInformation" type="csta:GetButtonInformationCaps"
minOccurs="0"/>
    <xsd:element name="getDisplay" type="csta:GetDisplayCaps" minOccurs="0"/>
    <xsd:element name="getHookSwitchStatus" type="csta:GetHookSwitchStatusCaps" minOccurs="0"/>
    <xsd:element name="getLampInfo" type="csta:GetLampInfoCaps" minOccurs="0"/>
    <xsd:element name="getLampMode" type="csta:GetLampModeCaps" minOccurs="0"/>
    <xsd:element name="getMessageWaitingIndicator" type="csta:GetMessageWaitingIndicatorCaps"
minOccurs="0"/>
    <xsd:element name="getMicrophoneGain" type="csta:GetMicrophoneGainCaps" minOccurs="0"/>
    <xsd:element name="getMicrophoneMute" type="csta:GetMicrophoneMuteCaps" minOccurs="0"/>
    <xsd:element name="getRingerStatus" type="csta:GetRingerStatusCaps" minOccurs="0"/>
    <xsd:element name="getSpeakerMute" type="csta:GetSpeakerMuteCaps" minOccurs="0"/>
    <xsd:element name="getSpeakerVolume" type="csta:GetSpeakerVolumeCaps" minOccurs="0"/>
    <xsd:element name="setButtonInformation" type="csta:SetButtonInformationCaps"
minOccurs="0"/>
    <xsd:element name="setDisplay" type="csta:SetDisplayCaps" minOccurs="0"/>
    <xsd:element name="setHookSwitchStatus" type="csta:SetHookSwitchStatusCaps" minOccurs="0"/>
    <xsd:element name="setLampMode" type="csta:SetLampModeCaps" minOccurs="0"/>
    <xsd:element name="setMessageWaitingIndicator" type="csta:SetMessageWaitingIndicatorCaps"
minOccurs="0"/>
    <xsd:element name="setMicrophoneGain" type="csta:SetMicrophoneGainCaps" minOccurs="0"/>
    <xsd:element name="setMicrophoneMute" type="csta:SetMicrophoneMuteCaps" minOccurs="0"/>
    <xsd:element name="setRingerStatus" type="csta:SetRingerStatusCaps" minOccurs="0"/>
    <xsd:element name="setSpeakerMute" type="csta:SetSpeakerMuteCaps" minOccurs="0"/>
    <xsd:element name="setSpeakerVolume" type="csta:SetSpeakerVolumeCaps" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="ButtonPressCaps">
  <xsd:sequence>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="ackModelMultiStep" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="GetAuditoryApparatusInfoCaps">
  <xsd:sequence>
    <xsd:element name="auditoryApparatus" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="auditoryApparatusTypeSpeakerphone" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="auditoryApparatusTypeHandset" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="auditoryApparatusTypeHeadset" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="auditoryApparatusTypeSpeakerOnlyPhone" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="auditoryApparatusTypeother" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="speakerPresent" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="speakerVolumeSettable" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="speakerVolumeReadable" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="speakerMuteSettable" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="speakerMuteReadable" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="microphonePresent" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="microphoneGainSettable" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="microphoneGainReadable" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="microphoneMuteSettable" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="microphoneMuteReadable" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="hookswitchSettable" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

```
<xsd:element name="hookswitchOnHook" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
<xsd:complexType name="GetButtonInformationCaps">
<xsd:sequence>
<xsd:element name="button" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="buttonLabelInAck" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="buttonLabelSettableInAck" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="buttonFunctionInAck" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="buttonAssociatedNumberInAck" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="buttonAssociatedNumberSettableInAck" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="listOfLampsInAck" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
<xsd:complexType name="GetDisplayCaps">
<xsd:sequence>
<xsd:element name="displayID" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="characterSetASCII" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="characterSetUnicode" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="characterSetProprietary" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
<xsd:complexType name="GetHookSwitchStatusCaps">
<xsd:sequence>
<xsd:element name="hookSwitch" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
<xsd:complexType name="GetLampInfoCaps">
<xsd:sequence>
<xsd:element name="lamp" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="lampLabelInAck" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="buttonInAck" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="lampColorInAck" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
<xsd:complexType name="GetLampModeCaps">
<xsd:sequence>
<xsd:element name="lamp" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="lampModeInAck" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="lampBrightnessNormal" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="lampBrightnessDim" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="lampBrightnessBright" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="lampColorInAck" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="buttonInAck" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
<xsd:complexType name="GetMessageWaitingIndicatorCaps">
<xsd:sequence>
<xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="deviceForMsgInAck" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="lampIsPresentInAck" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
<xsd:complexType name="GetMicrophoneGainCaps">
<xsd:sequence>
<xsd:element name="auditoryApparatus" type="xsd:boolean" minOccurs="0"/>

```

```
<xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="micGainAbsInAck" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
<xsd:complexType name="GetMicrophoneMuteCaps">
  <xsd:sequence>
    <xsd:element name="auditoryApparatus" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="GetRingerStatusCaps">
  <xsd:sequence>
    <xsd:element name="ringer" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="ringCountInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="ringPatternInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="ringVolumeInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="ringVolumeAbsInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="GetSpeakerMuteCaps">
  <xsd:sequence>
    <xsd:element name="auditoryApparatus" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="GetSpeakerVolumeCaps">
  <xsd:sequence>
    <xsd:element name="auditoryApparatus" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="speakerVolAbsInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="SetButtonInformationCaps">
  <xsd:sequence>
    <xsd:element name="buttonLabel" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="buttonAssociatedNumber" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="ackModelMultiStep" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="SetDisplayCaps">
  <xsd:sequence>
    <xsd:element name="physBaseRowIndex" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="physColumnRowIndex" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="offset" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="supportsModifyingPosition" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="ackModelMultiStep" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="SetHookSwitchStatusCaps">
  <xsd:sequence>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="ackModelMultiStep" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="SetLampModeCaps">
  <xsd:sequence>
    <xsd:element name="lampModeBrokenFlutter" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="lampModeFlutter" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```



```
<xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="ackModelMultiStep" type="xsd:boolean" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
<xsd:complexType name="SetMessageWaitingIndicatorCaps">
  <xsd:sequence>
    <xsd:element name="deviceForMsg" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="ackModelMultiStep" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="SetMicrophoneGainCaps">
  <xsd:sequence>
    <xsd:element name="microphoneGainAbs" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="microphoneGainInc" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="ackModelMultiStep" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="SetMicrophoneMuteCaps">
  <xsd:sequence>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="ackModelMultiStep" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="SetRingerStatusCaps">
  <xsd:sequence>
    <xsd:element name="ringerModeRinging" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="ringerModeNotRinging" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="ringVolumeAbs" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="ringVolumeInc" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="ackModelMultiStep" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="SetSpeakerMuteCaps">
  <xsd:sequence>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="ackModelMultiStep" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="SetSpeakerVolumeCaps">
  <xsd:sequence>
    <xsd:element name="speakerVolumeAbs" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="speakerVolumeInc" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="ackModelMultiStep" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="resettedAfterCall" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="notSettableWhileActive" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="PhysDevEvtsList">
  <xsd:sequence>
    <xsd:element name="buttonInformation" type="csta:ButtonInformationCaps" minOccurs="0"/>
    <xsd:element name="buttonPress" type="csta:ButtonPressEventCaps" minOccurs="0"/>
    <xsd:element name="displayUpdated" type="csta:DisplayUpdatedCaps" minOccurs="0"/>
    <xsd:element name="hookswitch" type="csta:HookswitchCaps" minOccurs="0"/>
    <xsd:element name="lampMode" type="csta:LampModeCaps" minOccurs="0"/>
    <xsd:element name="messageWaiting" type="csta:MessageWaitingCaps" minOccurs="0"/>
    <xsd:element name="microphoneGain" type="csta:MicrophoneGainCaps" minOccurs="0"/>
    <xsd:element name="microphoneMute" type="csta:MicrophoneMuteCaps" minOccurs="0"/>
    <xsd:element name="ringerStatus" type="csta:RingerStatusCaps" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

```
<xsd:element name="speakerMute" type="csta:SpeakerMuteCaps" minOccurs="0"/>
  <xsd:element name="speakerVolume" type="csta:SpeakerVolumeCaps" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
<xsd:complexType name="ButtonInformationCaps">
  <xsd:sequence>
    <xsd:element name="buttonLabel" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="buttonAssociatedNumber" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="buttonPressIndicator" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="ButtonPressEventCaps">
  <xsd:sequence>
    <xsd:element name="buttonLabel" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="buttonAssociatedNumber" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="DisplayUpdatedCaps">
  <xsd:sequence>
    <xsd:element name="characterSetASCII" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="characterSetUnicode" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="characterSetProprietary" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="HookswitchCaps">
  <xsd:sequence>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="LampModeCaps">
  <xsd:sequence>
    <xsd:element name="lampModeBrokenFlutter" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="lampModeFlutter" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="lampModeOff" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="lampModeSteady" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="lampModeWink" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="lampModeReserved" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="lampModeSf0" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="lampModeSf1" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="lampModeSf2" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="lampModeSf3" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="lampModeSf4" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="lampModeSf5" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="lampModeSf6" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="lampModeSf7" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="lampModeSf8" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="lampModeSf9" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="lampModeSf10" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="lampModeSf11" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="lampModeSf12" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="lampModeSf13" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="lampModeSf14" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="lampModeSf15" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="lampModeSf16" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="lampModeSf17" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="lampModeSf18" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="lampModeSf19" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="lampModeSf20" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="lampModeSf21" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="lampModeSf22" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="lampModeSf23" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="lampModeSf24" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="lampModeSf25" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="lampModeSf26" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="lampModeSf27" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```



```
<xsd:sequence>
  <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
<xsd:complexType name="SpeakerVolumeCaps">
  <xsd:sequence>
    <xsd:element name="speakerVolumeAbs" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="speakerVolumeInc" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="LogicalServList">
  <xsd:sequence>
    <xsd:element name="callBackNonCallRel" type="csta:CallBackNonCallRelCaps" minOccurs="0"/>
    <xsd:element name="callBackMsgNonCallRel" type="csta:CallBackMsgNonCallRelCaps"
      minOccurs="0"/>
    <xsd:element name="cancelCallBack" type="csta:CancelCallBackCaps" minOccurs="0"/>
    <xsd:element name="cancelCallBackMsg" type="csta:CancelCallBackMsgCaps" minOccurs="0"/>
    <xsd:element name="getAgentState" type="csta:GetAgentStateCaps" minOccurs="0"/>
    <xsd:element name="getAutoAnswer" type="csta:GetAutoAnswerCaps" minOccurs="0"/>
    <xsd:element name="getAutoWorkMode" type="csta:GetAutoWorkModeCaps" minOccurs="0"/>
    <xsd:element name="getCallerIDStatus" type="csta:GetCallerIDStatusCaps" minOccurs="0"/>
    <xsd:element name="getDoNotDisturb" type="csta:GetDoNotDisturbCaps" minOccurs="0"/>
    <xsd:element name="getForwarding" type="csta:GetForwardingCaps" minOccurs="0"/>
    <xsd:element name="getLastNumberDialed" type="csta:GetLastNumberDialedCaps" minOccurs="0"/>
    <xsd:element name="getRouteingMode" type="csta:GetRouteingModeCaps" minOccurs="0"/>
    <xsd:element name="setAgentState" type="csta:SetAgentStateCaps" minOccurs="0"/>
    <xsd:element name="setAutoAnswer" type="csta:SetAutoAnswerCaps" minOccurs="0"/>
    <xsd:element name="setAutoWorkMode" type="csta:SetAutoWorkModeCaps" minOccurs="0"/>
    <xsd:element name="setCallerIDStatus" type="csta:SetCallerIDStatusCaps" minOccurs="0"/>
    <xsd:element name="setDoNotDisturb" type="csta:SetDoNotDisturbCaps" minOccurs="0"/>
    <xsd:element name="setForwarding" type="csta:SetForwardingCaps" minOccurs="0"/>
    <xsd:element name="setRouteingMode" type="csta:SetRouteingModeCaps" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="CallBackNonCallRelCaps">
  <xsd:sequence>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="additionalReqForbidden" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="ackModelMultiStep" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="CallBackMsgNonCallRelCaps">
  <xsd:sequence>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="additionalReqForbidden" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="ackModelMultiStep" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="CancelCallBackCaps">
  <xsd:sequence>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="ackModelMultiStep" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="supportsClearing" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="CancelCallBackMsgCaps">
  <xsd:sequence>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="ackModelMultiStep" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="supportsClearing" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="GetAgentStateCaps">
```

```
<xsd:sequence>
  <xsd:element name="acdGroup" type="xsd:boolean" minOccurs="0"/>
  <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
  <xsd:element name="agentStateListAgentIDInAck" type="xsd:boolean" minOccurs="0"/>
  <xsd:element name="agentGroupInAck" type="xsd:boolean" minOccurs="0"/>
  <xsd:element name="pendingAgentStateInAck" type="xsd:boolean" minOccurs="0"/>
  <xsd:element name="agentStateConditionForcedPauseInAck" type="xsd:boolean" minOccurs="0"/>
  <xsd:element name="agentStateConditionPauseInAck" type="xsd:boolean" minOccurs="0"/>
  <xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
<xsd:complexType name="GetAutoAnswerCaps">
  <xsd:sequence>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="numberOfRingsInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="GetAutoWorkModeCaps">
  <xsd:sequence>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="autoWorkIntervalInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="GetCallerIDStatusCaps">
  <xsd:sequence>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="GetDoNotDisturbCaps">
  <xsd:sequence>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="callOriginParameterInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="callOriginInternInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="callOriginExternInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="callingDeviceListInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="GetForwardingCaps">
  <xsd:sequence>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="forwardListInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="forwardListImmediateInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="forwardListBusyInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="forwardListDNDInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="forwardListNoAnsInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="forwardListBusyIntInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="forwardListBusyExtInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="forwardListDNDIntInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="forwardListDNDExtInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="forwardListNoAnsIntInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="forwardListNoAnsExtInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="forwardListImmIntInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="forwardListImmExtInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="forwardDNInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="forwardDefaultInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="forwardDefaultTypeAndDNInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="forwardDefaultTypeInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="forwardDefaultDNInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="ringCountInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="GetLastNumberDialedCaps">
  <xsd:sequence>
```

```
<xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
<xsd:complexType name="GetRouteingModeCaps">
  <xsd:sequence>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="SetAgentStateCaps">
  <xsd:sequence>
    <xsd:element name="requestedAgentStateLoggedOn" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="requestedAgentStateLoggedOff" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="requestedAgentStateNotReady" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="requestedAgentStateReady" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="requestedAgentStateWorkingAfterCall" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="agentID" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="password" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="group" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="pendingAgentStateWorkingAfterCallInAck" type="xsd:boolean"
minOccurs="0"/>
    <xsd:element name="pendingAgentStateNotReadyInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="pendingAgentStateNullInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="ackModelMultiStep" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="groupDeviceAllowedInReq" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="aCDDDeviceAllowedInReq" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="delayTransitionIfBusy" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="delayTransitionIfWorkingAfterCall" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="SetAutoAnswerCaps">
  <xsd:sequence>
    <xsd:element name="numberOfRings" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="ackModelMultiStep" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="SetAutoWorkModeCaps">
  <xsd:sequence>
    <xsd:element name="autoWorkInterval" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="ackModelMultiStep" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="groupDeviceAllowedInReq" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="aCDDDeviceAllowedInReq" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="SetCallerIDStatusCaps">
  <xsd:sequence>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="ackModelMultiStep" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="SetDoNotDisturbCaps">
  <xsd:sequence>
    <xsd:element name="callOriginationInternal" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="callOriginationExternal" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="callingDeviceList" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="ackModelMultiStep" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

```
<xsd:complexType name="SetForwardingCaps">
  <xsd:sequence>
    <xsd:element name="forwardingTypeBusy" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="forwardingTypeBusyInt" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="forwardingTypeBusyExt" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="forwardingTypeDND" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="forwardingTypeDNDInt" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="forwardingTypeDNDEExt" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="forwardingTypeNoAns" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="forwardingTypeNoAnsInt" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="forwardingTypeNoAnsExt" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="forwardingTypeImmediate" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="forwardingTypeImmInt" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="forwardingTypeImmExt" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="forwardDN" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="ringCount" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="ackModelMultiStep" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="SetRouteingModeCaps">
  <xsd:sequence>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="ackModelMultiStep" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="LogicalEvtsList">
  <xsd:sequence>
    <xsd:element name="agentBusy" type="csta:AgentBusyCaps" minOccurs="0"/>
    <xsd:element name="agentLoggedOff" type="csta:AgentLoggedOffCaps" minOccurs="0"/>
    <xsd:element name="agentLoggedOn" type="csta:AgentLoggedOnCaps" minOccurs="0"/>
    <xsd:element name="agentNotReady" type="csta:AgentNotReadyCaps" minOccurs="0"/>
    <xsd:element name="agentReady" type="csta:AgentReadyCaps" minOccurs="0"/>
    <xsd:element name="agentWorkingAfterCall" type="csta:AgentWorkingAfterCallCaps"
      minOccurs="0"/>
    <xsd:element name="autoAnswer" type="csta:AutoAnswerCaps" minOccurs="0"/>
    <xsd:element name="autoWorkMode" type="csta:AutoWorkModeCaps" minOccurs="0"/>
    <xsd:element name="callBack" type="csta:CallBackEventCaps" minOccurs="0"/>
    <xsd:element name="callBackMessage" type="csta:CallBackMessageEventCaps" minOccurs="0"/>
    <xsd:element name="callerIDStatus" type="csta:CallerIDStatusCaps" minOccurs="0"/>
    <xsd:element name="doNotDisturb" type="csta:DoNotDisturbCaps" minOccurs="0"/>
    <xsd:element name="forwarding" type="csta:ForwardingCaps" minOccurs="0"/>
    <xsd:element name="routeingMode" type="csta:RouteingModeCaps" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="AgentBusyCaps">
  <xsd:sequence>
    <xsd:element name="agentID" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="acdGroup" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="pendingAgentStateWorkingAfterCall" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="pendingAgentStateNotReady" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="pendingAgentStateReady" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="pendingAgentStateNull" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="cause" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="AgentLoggedOffCaps">
  <xsd:sequence>
    <xsd:element name="agentID" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="acdGroup" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="agentPassword" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="cause" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

```
<xsd:complexType name="AgentLoggedOnCaps">
  <xsd:sequence>
    <xsd:element name="agentID" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="acdGroup" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="agentPassword" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="cause" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="AgentNotReadyCaps">
  <xsd:sequence>
    <xsd:element name="agentID" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="acdGroup" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="cause" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="AgentReadyCaps">
  <xsd:sequence>
    <xsd:element name="agentID" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="acdGroup" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="cause" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="AgentWorkingAfterCallCaps">
  <xsd:sequence>
    <xsd:element name="agentID" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="acdGroup" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="pendingAgentStateNotReady" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="pendingAgentStateReady" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="pendingAgentStateNull" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="cause" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="AutoAnswerCaps">
  <xsd:sequence>
    <xsd:element name="numberOfRings" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="AutoWorkModeCaps">
  <xsd:sequence>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="CallBackEventCaps">
  <xsd:sequence>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="CallBackMessageEventCaps">
  <xsd:sequence>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="CallerIDStatusCaps">
  <xsd:sequence>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="DoNotDisturbCaps">
  <xsd:sequence>
    <xsd:element name="callOriginationParameter" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="callOriginationInternal" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="callOriginationExternal" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="callingDeviceList" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```



```
<xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
<xsd:complexType name="ForwardingCaps">
  <xsd:sequence>
    <xsd:element name="forwardingTypeBusy" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="forwardingTypeBusyInt" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="forwardingTypeBusyExt" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="forwardingTypeDND" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="forwardingTypeDNDInt" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="forwardingTypeDNDExt" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="forwardingTypeNoAns" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="forwardingTypeNoAnsInt" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="forwardingTypeNoAnsExt" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="forwardingTypeImmediate" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="forwardingTypeImmInt" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="forwardingTypeImmExt" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="forwardTo" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="forwardDefaultTypeAndDN" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="forwardDefaultType" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="forwardDefaultDN" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="ringCount" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="RouteingModeCaps">
  <xsd:sequence>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="DeviceMaintEvtList">
  <xsd:sequence>
    <xsd:element name="backInService" type="csta:BackInServiceCaps" minOccurs="0"/>
    <xsd:element name="deviceCapsChanged" type="csta:DeviceCapsChangedCaps" minOccurs="0"/>
    <xsd:element name="outOfService" type="csta:OutOfServiceCaps" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="BackInServiceCaps">
  <xsd:sequence>
    <xsd:element name="cause" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="DeviceCapsChangedCaps">
  <xsd:sequence>
    <xsd:element name="cause" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="OutOfServiceCaps">
  <xsd:sequence>
    <xsd:element name="cause" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="IOServicesServList">
  <xsd:sequence>
    <xsd:element name="ioRegister" type="csta:IoRegisterCaps" minOccurs="0"/>
    <xsd:element name="ioRegisterAbort" type="csta:IoRegisterAbortCaps" minOccurs="0"/>
    <xsd:element name="ioRegisterCancel" type="csta:IoRegisterCancelCaps" minOccurs="0"/>
    <xsd:element name="dataPathResumed" type="csta:DataPathResumedCaps" minOccurs="0"/>
    <xsd:element name="dataPathSuspended" type="csta:DataPathSuspendedCaps" minOccurs="0"/>
    <xsd:element name="fastData" type="csta:FastDataCaps" minOccurs="0"/>
    <xsd:element name="resumeDataPath" type="csta:ResumeDataPathCaps" minOccurs="0"/>
    <xsd:element name="sendBroadcastData" type="csta:SendBroadcastDataCaps" minOccurs="0"/>
    <xsd:element name="sendData" type="csta:SendDataCaps" minOccurs="0"/>
    <xsd:element name="sendMulticastData" type="csta:SendMulticastDataCaps" minOccurs="0"/>
    <xsd:element name="startDataPath" type="csta:StartDataPathCaps" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

```
<xsd:element name="stopDataPath" type="csta:StopDataPathCaps" minOccurs="0"/>
<xsd:element name="suspendDataPath" type="csta:SuspendDataPathCaps" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
<xsd:complexType name="IoRegisterCaps">
<xsd:sequence>
<xsd:element name="ioDevice" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="allIODevices" type="xsd:boolean" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
<xsd:complexType name="IoRegisterAbortCaps">
<xsd:sequence>
<xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
<xsd:complexType name="IoRegisterCancelCaps">
<xsd:sequence>
<xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
<xsd:complexType name="DataPathResumedCaps">
<xsd:sequence>
<xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
<xsd:complexType name="DataPathSuspendedCaps">
<xsd:sequence>
<xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
<xsd:complexType name="FastDataCaps">
<xsd:sequence>
<xsd:element name="objectDevice" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="objectCall" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="dataPathTypeText" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="dataPathTypeVoice" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="displayAttribPhyBaseRowIndex" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="displayAttribPhyBaseColumnNumber" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="displayAttribOffset" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="supportsModifyingPosition" type="xsd:boolean" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
<xsd:complexType name="ResumeDataPathCaps">
<xsd:sequence>
<xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="sendsDataPathResumed" type="xsd:boolean" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
<xsd:complexType name="SendBroadcastDataCaps">
<xsd:sequence>
<xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="dataPathTypeText" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="dataPathTypeVoice" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="displayAttribPhyBaseRowIndex" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="displayAttribPhyBaseColumnNumber" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="displayAttribOffset" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="supportsModifyingPosition" type="xsd:boolean" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
```

```
<xsd:complexType name="SendDataCaps">
  <xsd:sequence>
    <xsd:element name="displayAttribPhyBaseRowNumber" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="displayAttribPhyBaseColumnNumber" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="displayAttribOffset" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="ioCauseTerminationCharReceived" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="ioCauseCharCountReached" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="ioCauseTimeout" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="ioCauseSfTerminated" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="supportsModifyingPosition" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="SendMulticastDataCaps">
  <xsd:sequence>
    <xsd:element name="ioData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="displayAttribPhyBaseRowNumber" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="displayAttribPhyBaseColumnNumber" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="displayAttribOffset" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="supportsModifyingPosition" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="StartDataPathCaps">
  <xsd:sequence>
    <xsd:element name="objectDevice" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="objectCall" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="dataPathDirectionCfToObject" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="dataPathDirectionObjectToCf" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="dataPathDirectionBidirectional" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="dataPathTypeText" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="dataPathTypeVoice" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="numberOfCharsToCollect" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="terminationChar" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="timeout" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="numberOfCharsToCollectInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="terminationCharInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="timeoutInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="StopDataPathCaps">
  <xsd:sequence>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="SuspendDataPathCaps">
  <xsd:sequence>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="DataCollectionServList">
  <xsd:sequence>
    <xsd:element name="dataCollected" type="csta:DataCollectedCaps" minOccurs="0"/>
    <xsd:element name="dataCollectionResumed" type="csta:DataCollectionResumedCaps"
      minOccurs="0"/>
    <xsd:element name="dataCollectionSuspended" type="csta:DataCollectionSuspendedCaps"
      minOccurs="0"/>
    <xsd:element name="resumeDataCollection" type="csta:ResumeDataCollectionCaps"
      minOccurs="0"/>
    <xsd:element name="startDataCollection" type="csta:StartDataCollectionCaps" minOccurs="0"/>
    <xsd:element name="stopDataCollection" type="csta:StopDataCollectionCaps" minOccurs="0"/>
    <xsd:element name="suspendDataCollection" type="csta:SuspendDataCollectionCaps"
      minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

```
minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
<xsd:complexType name="DataCollectedCaps">
  <xsd:sequence>
    <xsd:element name="digitsDuration" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="digitsPauseDuration" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="toneDetectedBeep" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="toneDetectedBilling" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="toneDetectedBusy" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="toneDetectedCarrier" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="toneDetectedConfirmation" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="toneDetectedDial" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="toneDetectedFaxCNG" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="toneDetectedHold" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="toneDetectedHowler" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="toneDetectedIntrusion" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="toneDetectedModemCNG" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="toneDetectedPark" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="toneDetectedRecordWarning" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="toneDetectedReorder" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="toneDetectedRingback" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="toneDetectedSilence" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="toneDetectedSitVC" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="toneDetectedSitIC" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="toneDetectedSitRO" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="toneDetectedSitNC" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="toneDetectedSf0" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="toneDetectedSf1" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="toneDetectedSf2" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="toneDetectedSf3" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="toneDetectedSf4" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="toneDetectedSf5" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="toneDetectedSf6" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="toneDetectedSf7" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="toneDetectedSf8" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="toneDetectedSf9" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="toneDetectedSf10" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="toneDetectedOther" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="toneFrequency" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="toneDuration" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="tonePauseDuration" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="connectionInfo" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="dcollCauseFushCharReceived" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="dcollCauseCharCountReached" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="dcollCauseTimeout" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="dcollCauseSFTerminated" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="DataCollectionResumedCaps">
  <xsd:sequence>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="DataCollectionSuspendedCaps">
  <xsd:sequence>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="ResumeDataCollectionCaps">
  <xsd:sequence>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

```
</xsd:complexType>
<xsd:complexType name="StartDataCollectionCaps">
  <xsd:sequence>
    <xsd:element name="objectDevice" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="objectCall" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="dataCollectionTypeDigits" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="dataCollectionTypeTones" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="digitsReportingCriteriaNumChars" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="digitsReportingCriteriaFlushChar" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="digitsReportingCriteriaTimeout" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="StopDataCollectionCaps">
  <xsd:sequence>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="SuspendDataCollectionCaps">
  <xsd:sequence>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="VoiceUnitServList">
  <xsd:sequence>
    <xsd:element name="concatenateMsg" type="csta:ConcatenateMsgCaps" minOccurs="0"/>
    <xsd:element name="deleteMsg" type="csta>DeleteMsgCaps" minOccurs="0"/>
    <xsd:element name="playMsg" type="csta:PlayMsgCaps" minOccurs="0"/>
    <xsd:element name="queryVoiceAttrib" type="csta:QueryVoiceAttribCaps" minOccurs="0"/>
    <xsd:element name="recordMsg" type="csta:RecordMsgCaps" minOccurs="0"/>
    <xsd:element name="reposition" type="csta:RepositionCaps" minOccurs="0"/>
    <xsd:element name="resume" type="csta:ResumeCaps" minOccurs="0"/>
    <xsd:element name="review" type="csta:ReviewCaps" minOccurs="0"/>
    <xsd:element name="setVoiceAttrib" type="csta:SetVoiceAttribCaps" minOccurs="0"/>
    <xsd:element name="stop" type="csta:StopCaps" minOccurs="0"/>
    <xsd:element name="suspend" type="csta:SuspendCaps" minOccurs="0"/>
    <xsd:element name="synthesizeMsg" type="csta:SynthesizeMsgCaps" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="ConcatenateMsgCaps">
  <xsd:sequence>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="DeleteMsgCaps">
  <xsd:sequence>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="PlayMsgCaps">
  <xsd:sequence>
    <xsd:element name="duration" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="terminationParameter" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="terminationDurationExceeded" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="terminationDTMFDigitDetected" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="terminationEndOfSpeechDetected" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="terminationSpeech" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="multipleMsgsSimultaneously" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="QueryVoiceAttribCaps">
```

```
<xsd:sequence>
  <xsd:element name="attribToQueryEncodingAlgorithm" type="xsd:boolean" minOccurs="0"/>
  <xsd:element name="attribToQuerySamplingRate" type="xsd:boolean" minOccurs="0"/>
  <xsd:element name="attribToQueryDuration" type="xsd:boolean" minOccurs="0"/>
  <xsd:element name="attribToQueryFilename" type="xsd:boolean" minOccurs="0"/>
  <xsd:element name="attribToQueryCurrentPosition" type="xsd:boolean" minOccurs="0"/>
  <xsd:element name="attribToQueryCurrentSpeed" type="xsd:boolean" minOccurs="0"/>
  <xsd:element name="attribToQueryCurrentVolume" type="xsd:boolean" minOccurs="0"/>
  <xsd:element name="attribToQueryCurrentLevel" type="xsd:boolean" minOccurs="0"/>
  <xsd:element name="attribToQueryCurrentState" type="xsd:boolean" minOccurs="0"/>
  <xsd:element name="connection" type="xsd:boolean" minOccurs="0"/>
  <xsd:element name="duration" type="xsd:boolean" minOccurs="0"/>
  <xsd:element name="terminationParameter" type="xsd:boolean" minOccurs="0"/>
  <xsd:element name="terminationDurationExceeded" type="xsd:boolean" minOccurs="0"/>
  <xsd:element name="terminationDTMFDigitDetected" type="xsd:boolean" minOccurs="0"/>
  <xsd:element name="terminationEndOfSpeechDetected" type="xsd:boolean" minOccurs="0"/>
  <xsd:element name="terminationSpeech" type="xsd:boolean" minOccurs="0"/>
  <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
  <xsd:element name="attribInAckEncodingAlgorithmADPCM6K" type="xsd:boolean" minOccurs="0"/>
  <xsd:element name="attribInAckEncodingAlgorithmADPCM8K" type="xsd:boolean" minOccurs="0"/>
  <xsd:element name="attribInAckEncodingAlgorithmMuLawPCM6K" type="xsd:boolean"
minOccurs="0"/>
  <xsd:element name="attribInAckEncodingAlgorithmALawPCM6K" type="xsd:boolean" minOccurs="0"/>
  <xsd:element name="attribInAckSamplingRate" type="xsd:boolean" minOccurs="0"/>
  <xsd:element name="attribInAckDuration" type="xsd:boolean" minOccurs="0"/>
  <xsd:element name="attribInAckFilename" type="xsd:boolean" minOccurs="0"/>
  <xsd:element name="attribInAckCurrentPosition" type="xsd:boolean" minOccurs="0"/>
  <xsd:element name="attribInAckCurrentSpeed" type="xsd:boolean" minOccurs="0"/>
  <xsd:element name="attribInAckCurrentVolumeAbs" type="xsd:boolean" minOccurs="0"/>
  <xsd:element name="attribInAckCurrentGain" type="xsd:boolean" minOccurs="0"/>
  <xsd:element name="attribInAckCurrentState" type="xsd:boolean" minOccurs="0"/>
  <xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
<xsd:complexType name="RecordMsgCaps">
  <xsd:sequence>
    <xsd:element name="samplingRate" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="encodingAlgorithmADPCM6K" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="encodingAlgorithmADPCM8K" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="encodingAlgorithmMuLawPCM6K" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="encodingAlgorithmALawPCM6K" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="maxDuration" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="terminationParameter" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="terminationDurationExceeded" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="terminationDTMFDigitDetected" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="terminationEndOfDataDetected" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="terminationSpeechDetected" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="RepositionCaps">
  <xsd:sequence>
    <xsd:element name="periodOfRepositionStartOfMsg" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="periodOfRepositionEndOfMsg" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="periodOfRepositionRelativePointer" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="msgToReposition" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="ResumeCaps">
  <xsd:sequence>
    <xsd:element name="msgToResume" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="duration" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

```
</xsd:complexType>
<xsd:complexType name="ReviewCaps">
  <xsd:sequence>
    <xsd:element name="periodToResumeStartOfMsg" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="periodToResumeLengthOfReview" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="SetVoiceAttribCaps">
  <xsd:sequence>
    <xsd:element name="currentSpeed" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="currentVolumeAbs" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="currentVolumeInc" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="periodToResumeStartOfMsg" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="periodToResumeLengthOfReview" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="currentGain" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="message" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="StopCaps">
  <xsd:sequence>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="SuspendCaps">
  <xsd:sequence>
    <xsd:element name="message" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="SynthesizeMsgCaps">
  <xsd:sequence>
    <xsd:element name="genderMale" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="genderFemale" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="VoiceUnitEvtsList">
  <xsd:sequence>
    <xsd:element name="play" type="csta:PlayCaps" minOccurs="0"/>
    <xsd:element name="record" type="csta:RecordCaps" minOccurs="0"/>
    <xsd:element name="review" type="csta:ReviewEventCaps" minOccurs="0"/>
    <xsd:element name="stop" type="csta:StopEventCaps" minOccurs="0"/>
    <xsd:element name="suspendPlay" type="csta:SuspendPlayCaps" minOccurs="0"/>
    <xsd:element name="suspendRecord" type="csta:SuspendRecordCaps" minOccurs="0"/>
    <xsd:element name="voiceAttribChanged" type="csta:VoiceAttribChangedCaps" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="PlayCaps">
  <xsd:sequence>
    <xsd:element name="length" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="currentPosition" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="speed" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="cause" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="servicesPermitted" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="RecordCaps">
  <xsd:sequence>
    <xsd:element name="length" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="currentPosition" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

```
<xsd:element name="speed" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="cause" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="servicesPermitted" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
<xsd:complexType name="ReviewEventCaps">
  <xsd:sequence>
    <xsd:element name="length" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="currentPosition" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="cause" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="servicesPermitted" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="StopEventCaps">
  <xsd:sequence>
    <xsd:element name="length" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="currentPosition" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="speed" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="cause" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="servicesPermitted" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="SuspendPlayCaps">
  <xsd:sequence>
    <xsd:element name="length" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="currentPosition" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="cause" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="servicesPermitted" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="SuspendRecordCaps">
  <xsd:sequence>
    <xsd:element name="length" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="currentPosition" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="cause" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="servicesPermitted" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="VoiceAttribChangedCaps">
  <xsd:sequence>
    <xsd:element name="playVolumeAbs" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="playVolumeInc" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="recordGain" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="speed" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="currentPosition" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="cause" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="CDRServList">
  <xsd:sequence>
    <xsd:element name="cdrNotification" type="csta:CdrNotificationCaps" minOccurs="0"/>
    <xsd:element name="cdrReport" type="csta:CdrReportCaps" minOccurs="0"/>
    <xsd:element name="sendStoredCDRs" type="csta:SendStoredCDRsCaps" minOccurs="0"/>
    <xsd:element name="startCDRTransmission" type="csta:StartCDRTransmissionCaps"
minOccurs="0"/>
    <xsd:element name="stopCDRTransmission" type="csta:StopCDRTransmissionCaps" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="CdrNotificationCaps">
  <xsd:sequence>
    <xsd:element name="cdrReasonTimeout" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="cdrReasonThresholdReached" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```



```
<xsd:element name="cdrReasonOther" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
<xsd:complexType name="CdrReportCaps">
  <xsd:sequence>
    <xsd:element name="cdrReasonTimeout" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="cdrReasonThresholdReached" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="cdrReasonOther" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="recordNumber" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="recordCreationTime" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="callingDevice" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="calledDevice" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="assocCallingDevice" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="assocCalledDevice" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="netwCallingDevice" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="netwCalledDevice" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="callCharacteristics" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="mediaCallCharacteristics" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="chargedDeviceOperator" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="chargedDeviceNonOperator" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="recordedCall" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="nodeNumberArea0" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="nodeNumberArea1" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="nodeNumberArea2" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="tarifTable" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="connectionStart" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="connectionEnd" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="connectionDuration" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="accessCode" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="carrier" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="selectedRoute" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="billingIndicatorNormalCharging" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="billingIndicatorReverseCharging" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="billingIndicatorCreditCardCharging" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="billingIndicatorCallForwarding" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="billingIndicatorCallDeflection" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="billingIndicatorCallTransfer" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="billingIndicatorOther" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="chargingInfo" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="suppServiceInfoNormalCall" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="suppServiceInfoConsultationCall" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="suppServiceInfoTransferCall" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="suppServiceInfoCallCompletion" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="suppServiceInfoCallForwarding" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="suppServiceInfoCallDiversion" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="suppServiceInfoConferencing" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="suppServiceInfoIntrusion" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="suppServiceInfoUserUserInfo" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="suppServiceInfoOther" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="reasonForTermNormalClearing" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="reasonForTermUnsuccessfulCallAttempt" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="reasonForTermAbnormalTermination" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="reasonForTermCallTransferred" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="reasonForTermOther" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="authCode" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="accountInfo" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="deviceCategory" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="namedDeviceTypes" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="operatorDevice" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="lastStoredCDRReportSent" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="SendStoredCDRCaps">
  <xsd:sequence>
```

```
<xsd:element name="timePeriod" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
<xsd:complexType name="StartCDRTransmissionCaps">
  <xsd:sequence>
    <xsd:element name="transferModeTransferAtEndOfCall" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="transferModeTransferOnRequest" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="transferModeTransferOnThresholdReached" type="xsd:boolean"
      minOccurs="0"/>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="StopCDRTransmissionCaps">
  <xsd:sequence>
    <xsd:element name="cdrTermReasonEndOfData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="cdrTermReasonError" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="cdrTermReasonThresholdReached" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="cdrTermReasonOther" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="swFunctionSupportsSending" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="swFunctionSupportsReceiving" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="VendorSpecificServList">
  <xsd:sequence>
    <xsd:element name="escapeRegister" type="csta:EscapeRegisterCaps" minOccurs="0"/>
    <xsd:element name="escapeRegisterCancel" type="csta:EscapeRegisterCancelCaps"
      minOccurs="0"/>
    <xsd:element name="escapeRegisterAbort" type="csta:EscapeRegisterAbortCaps" minOccurs="0"/>
    <xsd:element name="escape" type="csta:EscapeCaps" minOccurs="0"/>
    <xsd:element name="privateDataVersionSelection" type="csta:PrivateDataVersionSelectionCaps"
      minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="EscapeRegisterCaps">
  <xsd:sequence>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="EscapeRegisterCancelCaps">
  <xsd:sequence>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="EscapeRegisterAbortCaps">
  <xsd:sequence>
    <xsd:element name="privateData" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="EscapeCaps">
  <xsd:sequence>
    <xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="swFunctionSupportsSending" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="swFunctionSupportsReceiving" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="PrivateDataVersionSelectionCaps">
  <xsd:sequence>
    <xsd:element name="privateDataInAck" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

```
<xsd:complexType name="VendorSpecificEvtsList">
  <xsd:sequence>
    <xsd:element name="privateEvent" type="csta:PrivateEventCaps" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="PrivateEventCaps">
  <xsd:complexContent>
    <xsd:extension base="csta:Empty"/>
  </xsd:complexContent>
</xsd:complexType>

<xsd:annotation>
  <xsd:documentation>other types</xsd:documentation>
</xsd:annotation>

<xsd:complexType name="DeviceIDFormat">
  <xsd:sequence>
    <xsd:element name="dialableDigitsAsterix" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="dialableDigitsHash" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="dialableDigitsABCD" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="dialableDigitsExclamation" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="dialableDigitsP" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="dialableDigitsT" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="dialableDigitsComma" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="dialableDigitsW" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="dialableDigitsAt" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="dialableDigitsDollar" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="dialableDigitsSemicolon" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="sFReprExclamation" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="sFReprEt" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="sFReprSlash" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="sFReprPercent" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="sFReprNM" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="sFReprGeneric" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="sFReprImplicitTON" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="sFReprPubTONUnkown" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="sFReprPubTONInternal" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="sFReprPubTONNational" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="sFReprPubTONSubscriber" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="sFReprPubTONAbbreviated" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="sFReprPriTONUnknown" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="sFReprPriTONLevel3" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="sFReprPriTONLevel2" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="sFReprPriTONLevel1" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="sFReprPriTONLocal" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="sFReprPriTONAbbreviated" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="sFReprOther" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="deviceNumber" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="SwDomainFeatures">
  <xsd:sequence>
    <xsd:element name="isForwardingBefore" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="isForwardingAfter" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="swFunctionDefaultSettings" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="userSpecific" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="userSpecificDefaultForwardingType" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="userSpecificDefaultForwardDestination" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="negativeAcknowledgment" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="supportFailedWithAssConn" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="supportFailedWithoutAssConn" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="supportFailedWithAssConnNotReported" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="recall" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="callBack" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="extCallsIncoming" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="extCallsOutgoing" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="prompting" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

```
</xsd:sequence>
</xsd:complexType>

<xsd:complexType name="SwAppearanceAddressability">
  <xsd:sequence>
    <xsd:element name="nonAddressable" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="addressable" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="SwAppearanceTypes">
  <xsd:sequence>
    <xsd:element name="selectedStandard" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="basicStandard" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="basicBridged" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="exclusiveBridged" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="independentSharedBridged" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="interDependentSharedBridged" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>

<xsd:simpleType name="IgnoreUnsupportedParameters">
  <xsd:restriction base="xsd:string">
    <xsd:enumeration value="ignoreParameters"/>
    <xsd:enumeration value="rejectMessage"/>
  </xsd:restriction>
</xsd:simpleType>

<xsd:simpleType name="PauseTime">
  <xsd:restriction base="xsd:long">
    <xsd:minInclusive value="0"/>
    <xsd:maxInclusive value="2000"/>
  </xsd:restriction>
</xsd:simpleType>

<xsd:complexType name="TimeStampMode">
  <xsd:sequence>
    <xsd:element name="allEvents" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="allAcks" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="allServReqs" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="MiscMonitorCaps">
  <xsd:sequence>
    <xsd:element name="groupInclusivModel" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="groupExclusiveModel" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="monitorPhysicalElement" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="acdDeviceInclusiv" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="acdDeviceExclusiv" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="MaxLengthParameters">
  <xsd:sequence>
    <xsd:element name="accountInfo">
      <xsd:simpleType>
        <xsd:restriction base="xsd:short">
          <xsd:minInclusive value="0"/>
          <xsd:maxInclusive value="32"/>
        </xsd:restriction>
      </xsd:simpleType>
    </xsd:element>
    <xsd:element name="authCode">
      <xsd:simpleType>
        <xsd:restriction base="xsd:short">
          <xsd:minInclusive value="0"/>
          <xsd:maxInclusive value="32"/>
        </xsd:restriction>
      </xsd:simpleType>
    </xsd:element>
  </xsd:sequence>
</xsd:complexType>
```

```
        </xsd:restriction>
    </xsd:simpleType>
</xsd:element>
<xsd:element name="agentID">
    <xsd:simpleType>
        <xsd:restriction base="xsd:short">
            <xsd:minInclusive value="0"/>
            <xsd:maxInclusive value="32"/>
        </xsd:restriction>
    </xsd:simpleType>
</xsd:element>
<xsd:element name="agentPassword">
    <xsd:simpleType>
        <xsd:restriction base="xsd:short">
            <xsd:minInclusive value="0"/>
            <xsd:maxInclusive value="32"/>
        </xsd:restriction>
    </xsd:simpleType>
</xsd:element>
<xsd:element name="callIDInConnectionID">
    <xsd:simpleType>
        <xsd:restriction base="xsd:short">
            <xsd:minInclusive value="0"/>
            <xsd:maxInclusive value="8"/>
        </xsd:restriction>
    </xsd:simpleType>
</xsd:element>
<xsd:element name="correlatorData">
    <xsd:simpleType>
        <xsd:restriction base="xsd:short">
            <xsd:minInclusive value="0"/>
            <xsd:maxInclusive value="32"/>
        </xsd:restriction>
    </xsd:simpleType>
</xsd:element>
<xsd:element name="privateData" type="xsd:long"/>
<xsd:element name="deviceIdentifiers">
    <xsd:simpleType>
        <xsd:restriction base="xsd:short">
            <xsd:minInclusive value="0"/>
            <xsd:maxInclusive value="128"/>
        </xsd:restriction>
    </xsd:simpleType>
</xsd:element>
<xsd:element name="userData">
    <xsd:simpleType>
        <xsd:restriction base="xsd:short">
            <xsd:minInclusive value="0"/>
            <xsd:maxInclusive value="256"/>
        </xsd:restriction>
    </xsd:simpleType>
</xsd:element>
<xsd:element name="buttonLabel">
    <xsd:simpleType>
        <xsd:restriction base="xsd:short">
            <xsd:minInclusive value="0"/>
            <xsd:maxInclusive value="64"/>
        </xsd:restriction>
    </xsd:simpleType>
</xsd:element>
<xsd:element name="lampLabel">
    <xsd:simpleType>
        <xsd:restriction base="xsd:short">
            <xsd:minInclusive value="0"/>
            <xsd:maxInclusive value="64"/>
        </xsd:restriction>
    </xsd:simpleType>
</xsd:element>
```

```
<xsd:element name="charactersToSend">
  <xsd:simpleType>
    <xsd:restriction base="xsd:short">
      <xsd:minInclusive value="0"/>
      <xsd:maxInclusive value="64"/>
    </xsd:restriction>
  </xsd:simpleType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>

<xsd:complexType name="FilterThreshold">
  <xsd:sequence>
    <xsd:element name="getLogicalDeviceInformation" type="xsd:long"/>
    <xsd:element name="getPhysicalDeviceInformation" type="xsd:long"/>
    <xsd:element name="getSwitchingFunctionCaps" type="xsd:long"/>
    <xsd:element name="getSwitchingFunctionDevices" type="xsd:long"/>
    <xsd:element name="switchingFunctionDevices" type="xsd:long"/>
    <xsd:element name="changeSystemStatusFilter" type="xsd:long"/>
    <xsd:element name="systemStatusRegister" type="xsd:long"/>
    <xsd:element name="systemStatusRegisterAbort" type="xsd:long"/>
    <xsd:element name="systemStatusRegisterCancel" type="xsd:long"/>
    <xsd:element name="requestSystemStatus" type="xsd:long"/>
    <xsd:element name="systemStatus" type="xsd:long"/>
    <xsd:element name="switchingFunctionCapsChanged" type="xsd:long"/>
    <xsd:element name="switchingFunctionDevsChanged" type="xsd:long"/>
    <xsd:element name="changeMonitorFilter" type="xsd:long"/>
    <xsd:element name="monitorStart" type="xsd:long"/>
    <xsd:element name="monitorStop" type="xsd:long"/>
    <xsd:element name="snapshotCall" type="xsd:long"/>
    <xsd:element name="snapshotDevice" type="xsd:long"/>
    <xsd:element name="snapshotCallData" type="xsd:long"/>
    <xsd:element name="snapshotDeviceData" type="xsd:long"/>
    <xsd:element name="acceptCall" type="xsd:long"/>
    <xsd:element name="alternateCall" type="xsd:long"/>
    <xsd:element name="answerCall" type="xsd:long"/>
    <xsd:element name="callBackCallRelated" type="xsd:long"/>
    <xsd:element name="callBackMessageCallRelated" type="xsd:long"/>
    <xsd:element name="campOnCall" type="xsd:long"/>
    <xsd:element name="clearCall" type="xsd:long"/>
    <xsd:element name="clearConnection" type="xsd:long"/>
    <xsd:element name="conferenceCall" type="xsd:long"/>
    <xsd:element name="consultationCall" type="xsd:long"/>
    <xsd:element name="deflectCall" type="xsd:long"/>
    <xsd:element name="dialDigits" type="xsd:long"/>
    <xsd:element name="directedPickupCall" type="xsd:long"/>
    <xsd:element name="groupPickupCall" type="xsd:long"/>
    <xsd:element name="holdCall" type="xsd:long"/>
    <xsd:element name="intrudeCall" type="xsd:long"/>
    <xsd:element name="joinCall" type="xsd:long"/>
    <xsd:element name="makeCall" type="xsd:long"/>
    <xsd:element name="makePredictiveCall" type="xsd:long"/>
    <xsd:element name="parkCall" type="xsd:long"/>
    <xsd:element name="reconnectCall" type="xsd:long"/>
    <xsd:element name="retrieveCall" type="xsd:long"/>
    <xsd:element name="singleStepConferenceCall" type="xsd:long"/>
    <xsd:element name="singleStepTransferCall" type="xsd:long"/>
    <xsd:element name="transferCall" type="xsd:long"/>
    <xsd:element name="associatedData" type="xsd:long"/>
    <xsd:element name="cancelTelephonyTone" type="xsd:long"/>
    <xsd:element name="generateDigits" type="xsd:long"/>
    <xsd:element name="generateTelephonyTone" type="xsd:long"/>
    <xsd:element name="sendUserInformation" type="xsd:long"/>
    <xsd:element name="startDTMFDigitsCollection" type="xsd:long"/>
    <xsd:element name="startTelephonyTonesCollection" type="xsd:long"/>
    <xsd:element name="stopDTMFDigitsCollection" type="xsd:long"/>
    <xsd:element name="stopTelephonyTonesCollection" type="xsd:long"/>
    <xsd:element name="attachMediaService" type="xsd:long"/>
  </xsd:sequence>
</xsd:complexType>
```

```
<xsd:element name="detachMediaService" type="xsd:long"/>
<xsd:element name="routeRegister" type="xsd:long"/>
<xsd:element name="routeRegisterAbort" type="xsd:long"/>
<xsd:element name="routeRegisterCancel" type="xsd:long"/>
<xsd:element name="reRoute" type="xsd:long"/>
<xsd:element name="routeEnd" type="xsd:long"/>
<xsd:element name="routeReject" type="xsd:long"/>
<xsd:element name="routeRequest" type="xsd:long"/>
<xsd:element name="routeSelect" type="xsd:long"/>
<xsd:element name="routeUsed" type="xsd:long"/>
<xsd:element name="buttonPress" type="xsd:long"/>
<xsd:element name="getAuditoryApparatusInfo" type="xsd:long"/>
<xsd:element name="getButtonInformation" type="xsd:long"/>
<xsd:element name="getDisplay" type="xsd:long"/>
<xsd:element name="getHookswitchStatus" type="xsd:long"/>
<xsd:element name="getLampInformation" type="xsd:long"/>
<xsd:element name="getLampMode" type="xsd:long"/>
<xsd:element name="getMicrophoneGain" type="xsd:long"/>
<xsd:element name="getMicrophoneMute" type="xsd:long"/>
<xsd:element name="getMessageWaitingIndicator" type="xsd:long"/>
<xsd:element name="getRingerStatus" type="xsd:long"/>
<xsd:element name="getSpeakerMute" type="xsd:long"/>
<xsd:element name="getSpeakerVolume" type="xsd:long"/>
<xsd:element name="setButtonInformation" type="xsd:long"/>
<xsd:element name="setDisplay" type="xsd:long"/>
<xsd:element name="setHookswitchStatus" type="xsd:long"/>
<xsd:element name="setLampMode" type="xsd:long"/>
<xsd:element name="setMsgWaitingIndicator" type="xsd:long"/>
<xsd:element name="setMicrophoneGain" type="xsd:long"/>
<xsd:element name="setMicrophoneMute" type="xsd:long"/>
<xsd:element name="setRingerStatus" type="xsd:long"/>
<xsd:element name="setSpeakerMute" type="xsd:long"/>
<xsd:element name="setSpeakerVolume" type="xsd:long"/>
<xsd:element name="callBackMessageNonCallRelated" type="xsd:long"/>
<xsd:element name="callBackNonCallRelated" type="xsd:long"/>
<xsd:element name="cancelCallBack" type="xsd:long"/>
<xsd:element name="cancelCallBackMessage" type="xsd:long"/>
<xsd:element name="getAgentState" type="xsd:long"/>
<xsd:element name="getAutoAnswer" type="xsd:long"/>
<xsd:element name="getAutoWorkMode" type="xsd:long"/>
<xsd:element name="getCallerIDStatus" type="xsd:long"/>
<xsd:element name="getDoNotDisturb" type="xsd:long"/>
<xsd:element name="getForwarding" type="xsd:long"/>
<xsd:element name="getLastNumberDialed" type="xsd:long"/>
<xsd:element name="getRouteingMode" type="xsd:long"/>
<xsd:element name="setAgentState" type="xsd:long"/>
<xsd:element name="setAutoAnswer" type="xsd:long"/>
<xsd:element name="setAutoWorkMode" type="xsd:long"/>
<xsd:element name="setCallerIDStatus" type="xsd:long"/>
<xsd:element name="setDoNotDisturb" type="xsd:long"/>
<xsd:element name="setForwarding" type="xsd:long"/>
<xsd:element name="setRouteingMode" type="xsd:long"/>
<xsd:element name="backInService" type="xsd:long"/>
<xsd:element name="deviceCaosChanged" type="xsd:long"/>
<xsd:element name="outOfService" type="xsd:long"/>
<xsd:element name="ioRegister" type="xsd:long"/>
<xsd:element name="ioRegisterAbort" type="xsd:long"/>
<xsd:element name="ioRegisterCancel" type="xsd:long"/>
<xsd:element name="dataPathResumed" type="xsd:long"/>
<xsd:element name="dataPathSuspended" type="xsd:long"/>
<xsd:element name="fastData" type="xsd:long"/>
<xsd:element name="resumeDataPath" type="xsd:long"/>
<xsd:element name="sendBroadcastData" type="xsd:long"/>
<xsd:element name="sendData" type="xsd:long"/>
<xsd:element name="sendMulticastData" type="xsd:long"/>
<xsd:element name="startDataPath" type="xsd:long"/>
<xsd:element name="stopDataPath" type="xsd:long"/>
<xsd:element name="suspendDataPath" type="xsd:long"/>
```

```
<xsd:element name="concatenateMsg" type="xsd:long"/>
<xsd:element name="deleteMsg" type="xsd:long"/>
<xsd:element name="playMsg" type="xsd:long"/>
<xsd:element name="queryVoiceAttribute" type="xsd:long"/>
<xsd:element name="recordMsg" type="xsd:long"/>
<xsd:element name="reposition" type="xsd:long"/>
<xsd:element name="resume" type="xsd:long"/>
<xsd:element name="review" type="xsd:long"/>
<xsd:element name="setVoiceAttribute" type="xsd:long"/>
<xsd:element name="stop" type="xsd:long"/>
<xsd:element name="suspend" type="xsd:long"/>
<xsd:element name="synthesizeMsg" type="xsd:long"/>
<xsd:element name="cDRNotification" type="xsd:long"/>
<xsd:element name="cDRReport" type="xsd:long"/>
<xsd:element name="sendStoredCDRs" type="xsd:long"/>
<xsd:element name="startCDRTransmission" type="xsd:long"/>
<xsd:element name="stopCDRTransmission" type="xsd:long"/>
<xsd:element name="escapeRegister" type="xsd:long"/>
<xsd:element name="escapeRegisterAbort" type="xsd:long"/>
<xsd:element name="escapeRegisterCancel" type="xsd:long"/>
<xsd:element name="escape" type="xsd:long"/>
<xsd:element name="privateDataVersion" type="xsd:long"/>
</xsd:sequence>
</xsd:complexType>

<xsd:complexType name="MediaServiceCapsList">
  <xsd:sequence>
    <xsd:element name="mediaServiceCapsItem" minOccurs="0" maxOccurs="unbounded">
      <xsd:complexType>
        <xsd:sequence>
          <xsd:element name="mediaServiceType" type="csta:MediaServiceType"/>
          <xsd:element name="mediaServiceVersion" type="xsd:long" minOccurs="0"/>
          <xsd:element name="mediaServiceInstance" type="csta:MediaServiceInstanceID"
            minOccurs="0"/>
          <xsd:element name="connectionMode" type="csta:ConnectionModeBMap" minOccurs="0"/>
          <xsd:element name="mediaStreamIDsSupported" type="xsd:boolean"/>
        </xsd:sequence>
      </xsd:complexType>
    </xsd:element>
  </xsd:sequence>
</xsd:complexType>

<xsd:simpleType name="DeviceCategory">
  <xsd:restriction base="xsd:string">
    <xsd:enumeration value="acd"/>
    <xsd:enumeration value="group"/>
    <xsd:enumeration value="networkInterface"/>
    <xsd:enumeration value="park"/>
    <xsd:enumeration value="routeingDevice"/>
    <xsd:enumeration value="station"/>
    <xsd:enumeration value="voiceUnit"/>
    <xsd:enumeration value="other"/>
  </xsd:restriction>
</xsd:simpleType>

<xsd:complexType name="GroupDeviceAttributes">
  <xsd:sequence>
    <xsd:element name="acd" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="hunt" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="pick" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="other" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>

<xsd:simpleType name="NamedDeviceTypes">
  <xsd:restriction base="xsd:string">
    <xsd:enumeration value="acd"/>
    <xsd:enumeration value="acdGroup"/>
  </xsd:restriction>
</xsd:simpleType>
```



```
<xsd:enumeration value="button"/>
<xsd:enumeration value="buttonGroup"/>
<xsd:enumeration value="conferenceBridge"/>
<xsd:enumeration value="line"/>
<xsd:enumeration value="lineGroup"/>
<xsd:enumeration value="operator"/>
<xsd:enumeration value="operatorGroup"/>
<xsd:enumeration value="parkingDevice"/>
<xsd:enumeration value="station"/>
<xsd:enumeration value="stationGroup"/>
<xsd:enumeration value="trunk"/>
<xsd:enumeration value="trunkGroup"/>
<xsd:enumeration value="other"/>
<xsd:enumeration value="otherGroup"/>
</xsd:restriction>
</xsd:simpleType>

<xsd:complexType name="ACDModels">
  <xsd:sequence>
    <xsd:element name="visibleACDRelatedDevices" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="nonVisibleACDRelatedDevices" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="AgentLogOnModels">
  <xsd:sequence>
    <xsd:element name="logOnToACDDevice" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="logOnToACDGroupExplOneStep" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="logOnToACDGroupExplTwoSteps" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="logOnToACDImplOneStep" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>

<xsd:simpleType name="AppearanceType">
  <xsd:restriction base="xsd:string">
    <xsd:enumeration value="selectedStandard"/>
    <xsd:enumeration value="basicStandard"/>
    <xsd:enumeration value="basicBridged"/>
    <xsd:enumeration value="exclusiveBridged"/>
    <xsd:enumeration value="independentSharedBridged"/>
    <xsd:enumeration value="interdependentSharedBridged"/>
  </xsd:restriction>
</xsd:simpleType>

<xsd:complexType name="TransAndConfSetup">
  <xsd:sequence>
    <xsd:element name="consultationCall" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="holdCallMakeCall" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="alternateCall" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="twoCallsInHold" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="twoCallsInConnected" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>

</xsd:schema>
```

9.11 Call Detail Record

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-call-detail-record</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-identifiers.xsd"/>
  <xsd:include schemaLocation="call-connection-identifiers.xsd"/>
  <xsd:include schemaLocation="device-feature-types.xsd"/>
  <xsd:include schemaLocation="security.xsd"/>
  <xsd:include schemaLocation="call-control.xsd"/>
  <xsd:include schemaLocation="capability-exchange.xsd"/>
  <xsd:include schemaLocation="charge-info.xsd"/>
  <xsd:include schemaLocation="media-services.xsd"/>

  <xsd:simpleType name="CDRCrossRefID">
    <xsd:restriction base="xsd:string">
      <xsd:minLength value="0"/>
      <xsd:maxLength value="4"/>
      <xsd:pattern value="\d*" />
    </xsd:restriction>
  </xsd:simpleType>

  <xsd:element name="cdrCrossRefID" type="csta:CDRCrossRefID"/>

  <xsd:simpleType name="CDRReason">
    <xsd:restriction base="xsd:string">
      <xsd:enumeration value="timeout"/>
      <xsd:enumeration value="thresholdReached"/>
      <xsd:enumeration value="other"/>
    </xsd:restriction>
  </xsd:simpleType>

  <xsd:complexType name="CDRInfo">
    <xsd:sequence>
      <xsd:element name="cdrInformationItem" type="csta:CDRInformationItem" minOccurs="0"
maxOccurs="unbounded" />
    </xsd:sequence>
  </xsd:complexType>

  <xsd:complexType name="CDRInformationItem">
    <xsd:sequence>
      <xsd:element name="recordNumber" minOccurs="0">
        <xsd:simpleType>
          <xsd:restriction base="xsd:integer">
            <xsd:minInclusive value="1"/>
            <xsd:maxInclusive value="128"/>
          </xsd:restriction>
        </xsd:simpleType>
      </xsd:element>
      <xsd:element name="recordCreationTime" type="csta:TimeInfo"/>
      <xsd:element ref="csta:callingDevice" minOccurs="0"/>
      <xsd:element ref="csta:calledDevice" minOccurs="0"/>
      <xsd:element ref="csta:associatedCallingDevice" minOccurs="0"/>
      <xsd:element ref="csta:associatedCalledDevice" minOccurs="0"/>
      <xsd:element ref="csta:networkCallingDevice" minOccurs="0"/>
      <xsd:element ref="csta:networkCalledDevice" minOccurs="0"/>
      <xsd:element ref="csta:callCharacteristics" minOccurs="0"/>
      <xsd:element ref="csta:mediaCallCharacteristics" minOccurs="0"/>
      <xsd:element name="chargedDevice" type="csta:ChargedDevice" minOccurs="0"/>
      <xsd:element name="recordedCall" type="csta:ConnectionID" minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:schema>
```

```
<xsd:element name="nodeNumber" type="csta:NodeNumber" minOccurs="0"/>
<xsd:element name="tariffTable" type="xsd:long" minOccurs="0"/>
<xsd:element name="connectionStart" type="csta:TimeInfo" minOccurs="0"/>
<xsd:element name="connectionEnd" type="csta:TimeInfo" minOccurs="0"/>
<xsd:element name="connectionDuration" type="xsd:long" minOccurs="0"/>
<xsd:element name="accessCode" type="xsd:string" minOccurs="0"/>
<xsd:element name="carrier" type="xsd:long" minOccurs="0"/>
<xsd:element name="selectedRoute" type="xsd:long" minOccurs="0"/>
<xsd:element name="billingID" type="csta:BillingID" minOccurs="0"/>
<xsd:element name="chargingInfo" type="csta:ChargingInfo" minOccurs="0"/>
<xsd:element name="supplServiceInfo" type="csta:SupplServiceInfo" minOccurs="0"/>
<xsd:element name="reasonForTerm" type="csta:ReasonForTerm" minOccurs="0"/>
<xsd:element name="authCode" type="csta:AuthCode" minOccurs="0"/>
<xsd:element name="accountInfo" type="csta:AccountInfo" minOccurs="0"/>
<xsd:element name="deviceCategory" type="csta:DeviceCategory" minOccurs="0"/>
<xsd:element name="namedDeviceTypes" type="csta:NamedDeviceTypes" minOccurs="0"/>
<xsd:element name="operatorDevice" type="csta:DeviceID" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>

<xsd:complexType name="ChargedDevice">
  <xsd:choice>
    <xsd:element name="operator" type="csta:DeviceID"/>
    <xsd:element name="nonOperator" type="csta:DeviceID"/>
  </xsd:choice>
</xsd:complexType>

<xsd:complexType name="NodeNumber">
  <xsd:sequence>
    <xsd:element name="area0" type="xsd:long" minOccurs="0"/>
    <xsd:element name="areal" type="xsd:long" minOccurs="0"/>
    <xsd:element name="area2" type="xsd:long" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>

<xsd:simpleType name="BillingID">
  <xsd:restriction base="xsd:string">
    <xsd:enumeration value="normalCharging"/>
    <xsd:enumeration value="reverseCharging"/>
    <xsd:enumeration value="creditCardCharging"/>
    <xsd:enumeration value="callForwarding"/>
    <xsd:enumeration value="callDeflection"/>
    <xsd:enumeration value="callTransfer"/>
    <xsd:enumeration value="other"/>
  </xsd:restriction>
</xsd:simpleType>

<xsd:complexType name="SupplServiceInfo">
  <xsd:sequence>
    <xsd:element name="normalCall" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="consultationCall" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="transferCall" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="callCompletion" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="callForwarding" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="callDiversion" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="conferencing" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="intrusion" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="userUserInfo" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="other" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>

<xsd:simpleType name="ReasonForTerm">
  <xsd:restriction base="xsd:string">
    <xsd:enumeration value="normalClearing"/>
    <xsd:enumeration value="unsuccessfulCallAttempt"/>
    <xsd:enumeration value="abnormalTermination"/>
    <xsd:enumeration value="callTransferred"/>
  </xsd:restriction>
</xsd:simpleType>
```

```
        <xsd:enumeration value="other" />
    </xsd:restriction>
</xsd:simpleType>

<xsd:complexType name="CDRTimePeriod">
    <xsd:sequence>
        <xsd:element name="beginningOfCDR" type="csta:TimeInfo" />
        <xsd:element name="endOfCDR" type="csta:TimeInfo" />
    </xsd:sequence>
</xsd:complexType>

<xsd:simpleType name="CDRTransferMode">
    <xsd:restriction base="xsd:string">
        <xsd:enumeration value="transferAtEndOfCall" />
        <xsd:enumeration value="transferOnRequest" />
        <xsd:enumeration value="transferOnThresholdReached" />
    </xsd:restriction>
</xsd:simpleType>

<xsd:simpleType name="CDRTermReason">
    <xsd:restriction base="xsd:string">
        <xsd:enumeration value="endOfDataDetected" />
        <xsd:enumeration value="errorDetected" />
        <xsd:enumeration value="thresholdReached" />
        <xsd:enumeration value="other" />
    </xsd:restriction>
</xsd:simpleType>

</xsd:schema>
```

9.12 Charge information

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-charge-info</xsd:documentation>
  </xsd:annotation>

  <xsd:complexType name="ChargingInfo">
    <xsd:sequence>
      <xsd:element name="numberUnits" type="csta:NumberUnits"/>
      <xsd:element name="typeOfChargingInfo">
        <xsd:simpleType>
          <xsd:restriction base="xsd:string">
            <xsd:enumeration value="subTotal"/>
            <xsd:enumeration value="total"/>
          </xsd:restriction>
        </xsd:simpleType>
      </xsd:element>
    </xsd:sequence>
  </xsd:complexType>

  <xsd:complexType name="NumberUnits">
    <xsd:choice>
      <xsd:element name="numberOfChargeUnits" type="csta:NumberOfChargingUnits"/>
      <xsd:element name="numberOfCurrencyUnits" type="csta:NumberOfCurrencyUnits"/>
    </xsd:choice>
  </xsd:complexType>

  <xsd:complexType name="NumberOfChargingUnits">
    <xsd:sequence>
      <xsd:element name="chargingUnitsRecord" type="csta:ChargingUnitsRecord" minOccurs="0"
maxOccurs="unbounded"/>
    </xsd:sequence>
  </xsd:complexType>

  <xsd:complexType name="ChargingUnitsRecord">
    <xsd:sequence>
      <xsd:element name="chargingUnits" type="xsd:long"/>
      <xsd:element name="typeOfUnits" type="xsd:string" minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>

  <xsd:complexType name="NumberOfCurrencyUnits">
    <xsd:sequence>
      <xsd:element name="currencyType">
        <xsd:simpleType>
          <xsd:restriction base="xsd:string">
            <xsd:enumeration value="oneThousandth"/>
            <xsd:enumeration value="oneHundredth"/>
            <xsd:enumeration value="oneTenth"/>
            <xsd:enumeration value="one"/>
            <xsd:enumeration value="ten"/>
            <xsd:enumeration value="hundred"/>
            <xsd:enumeration value="thousand"/>
          </xsd:restriction>
        </xsd:simpleType>
      </xsd:element>
      <xsd:element name="currencyAmount" type="xsd:long"/>
      <xsd:element name="currencyMultiplier">
        <xsd:simpleType>
          <xsd:restriction base="xsd:string">
            <xsd:enumeration value="oneThousandth"/>
            <xsd:enumeration value="oneHundredth"/>
            <xsd:enumeration value="oneTenth"/>
            <xsd:enumeration value="one"/>
            <xsd:enumeration value="ten"/>
            <xsd:enumeration value="hundred"/>
            <xsd:enumeration value="thousand"/>
          </xsd:restriction>
        </xsd:simpleType>
      </xsd:element>
    </xsd:sequence>
  </xsd:complexType>

```

```
    </xsd:element>
  </xsd:sequence>
</xsd:complexType>

</xsd:schema>
```

9.13 Data call types

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-data-call-types</xsd:documentation>
  </xsd:annotation>

  <xsd:simpleType name="NumberOfChannels">
    <xsd:restriction base="xsd:long"/>
  </xsd:simpleType>

  <xsd:simpleType name="MaxChannelBind">
    <xsd:restriction base="xsd:long"/>
  </xsd:simpleType>

  <xsd:complexType name="ConnectionRateList">
    <xsd:sequence>
      <xsd:element name="connectionRate" type="xsd:long" minOccurs="0" maxOccurs="unbounded"/>
    </xsd:sequence>
  </xsd:complexType>

  <xsd:complexType name="DelayToleranceList">
    <xsd:sequence>
      <xsd:element name="delayTolerance" type="xsd:long" minOccurs="0" maxOccurs="unbounded"/>
    </xsd:sequence>
  </xsd:complexType>

</xsd:schema>
```

9.14 Escape types

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-escape-types</xsd:documentation>
  </xsd:annotation>

  <xsd:simpleType name="EscapeRegisterID">
    <xsd:restriction base="xsd:string">
      <xsd:minLength value="0"/>
      <xsd:maxLength value="4"/>
      <xsd:pattern value="\d*" />
    </xsd:restriction>
  </xsd:simpleType>

  <xsd:element name="escapeRegisterID" type="csta:EscapeRegisterID" />

</xsd:schema>
```


9.15 Media services

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:include schemaLocation="empty.xsd"/>

  <xsd:annotation>
    <xsd:documentation>CSTA-media-services</xsd:documentation>
  </xsd:annotation>

  <xsd:simpleType name="MediaServiceType">
    <xsd:restriction base="xsd:string">
      <xsd:enumeration value="cstaVoiceUnit"/>
      <xsd:enumeration value="dataModem"/>
      <xsd:enumeration value="digitalDataIsochronousIeee1394"/>
      <xsd:enumeration value="digitalDataIsochronousGeoport"/>
      <xsd:enumeration value="digitalDataIsochronousIeeeAtm"/>
      <xsd:enumeration value="digitalDataIsochronousIeeeIsdn"/>
      <xsd:enumeration value="digitalDataApi"/>
      <xsd:enumeration value="ectfS100MediaServicesDefault"/>
      <xsd:enumeration value="ectfS100MediaServicesAppServices"/>
      <xsd:enumeration value="cstaIVRScript1"/>
      <xsd:enumeration value="cstaIVRScript2"/>
      <xsd:enumeration value="cstaIVRScript3"/>
      <xsd:enumeration value="cstaIVRScript4"/>
      <xsd:enumeration value="cstaIVRScript5"/>
      <xsd:enumeration value="cstaIVRScript6"/>
      <xsd:enumeration value="cstaIVRScript7"/>
      <xsd:enumeration value="cstaIVRScript8"/>
      <xsd:enumeration value="cstaIVRScript9"/>
      <xsd:enumeration value="cstaIVRScript10"/>
      <xsd:enumeration value="liveSoundCaptureAnalog"/>
      <xsd:enumeration value="liveSoundTransmitAnalog"/>
      <xsd:enumeration value="liveSoundCaptureIeee1394"/>
      <xsd:enumeration value="liveSoundTransmitIeee1394"/>
      <xsd:enumeration value="liveSoundCaptureTransmitGeoport"/>
      <xsd:enumeration value="liveSoundCaptureTransmitAtm"/>
      <xsd:enumeration value="liveSoundCaptureTransmitISDN"/>
      <xsd:enumeration value="soundCaptureTransmitADPCM"/>
      <xsd:enumeration value="soundCaptureTransmitApi"/>
      <xsd:enumeration value="usb"/>
      <xsd:enumeration value="sfSpecific1"/>
      <xsd:enumeration value="sfSpecific2"/>
      <xsd:enumeration value="sfSpecific3"/>
      <xsd:enumeration value="sfSpecific4"/>
      <xsd:enumeration value="sfSpecific5"/>
      <xsd:enumeration value="sfSpecific6"/>
      <xsd:enumeration value="sfSpecific7"/>
      <xsd:enumeration value="sfSpecific8"/>
      <xsd:enumeration value="sfSpecific9"/>
      <xsd:enumeration value="sfSpecific10"/>
    </xsd:restriction>
  </xsd:simpleType>

  <xsd:simpleType name="MediaStreamID">
    <xsd:restriction base="xsd:string">
      <xsd:pattern value="\d*" />
    </xsd:restriction>
  </xsd:simpleType>

  <xsd:simpleType name="MediaServiceInstanceID">
    <xsd:restriction base="xsd:string">
      <xsd:maxLength value="64" />
    </xsd:restriction>
  </xsd:simpleType>
</xsd:schema>
```

```
<xsd:pattern value="\d*" />
</xsd:restriction>
</xsd:simpleType>

<xsd:complexType name="ConnectionInformation">
  <xsd:sequence>
    <xsd:element name="flowDirection" minOccurs="0">
      <xsd:simpleType>
        <xsd:restriction base="xsd:string">
          <xsd:enumeration value="transmit" />
          <xsd:enumeration value="receive" />
          <xsd:enumeration value="transmitAndReceive" />
        </xsd:restriction>
      </xsd:simpleType>
    </xsd:element>
    <xsd:element name="numberOfChannels" type="xsd:long" default="1" minOccurs="0" />
  </xsd:sequence>
</xsd:complexType>

<xsd:simpleType name="ConnectionMode">
  <xsd:restriction base="xsd:string">
    <xsd:enumeration value="consultationConference" />
    <xsd:enumeration value="consultationConferenceHold" />
    <xsd:enumeration value="directedPickup" />
    <xsd:enumeration value="join" />
    <xsd:enumeration value="singleStepConference" />
    <xsd:enumeration value="singleStepConferenceHold" />
    <xsd:enumeration value="singleStepTransfer" />
    <xsd:enumeration value="transfer" />
    <xsd:enumeration value="direct" />
  </xsd:restriction>
</xsd:simpleType>

<xsd:complexType name="ConnectionModeBMap">
  <xsd:sequence>
    <xsd:element name="consultationConference" type="xsd:boolean" minOccurs="0" />
    <xsd:element name="consultationConferenceHold" type="xsd:boolean" minOccurs="0" />
    <xsd:element name="directedPickup" type="xsd:boolean" minOccurs="0" />
    <xsd:element name="join" type="xsd:boolean" minOccurs="0" />
    <xsd:element name="singleStepConference" type="xsd:boolean" minOccurs="0" />
    <xsd:element name="singleStepConferenceHold" type="xsd:boolean" minOccurs="0" />
    <xsd:element name="singleStepTransfer" type="xsd:boolean" minOccurs="0" />
    <xsd:element name="transfer" type="xsd:boolean" minOccurs="0" />
    <xsd:element name="direct" type="xsd:boolean" minOccurs="0" />
  </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="MediaCallCharacteristics">
  <xsd:sequence>
    <xsd:element name="mediaClass" type="csta:MediaClass" />
    <xsd:element name="connectionRate" minOccurs="0">
      <xsd:simpleType>
        <xsd:restriction base="xsd:long">
          <xsd:minInclusive value="0" />
        </xsd:restriction>
      </xsd:simpleType>
    </xsd:element>
    <xsd:element name="bitRate" default="constant" minOccurs="0">
      <xsd:simpleType>
        <xsd:restriction base="xsd:string">
          <xsd:enumeration value="constant" />
          <xsd:enumeration value="variable" />
        </xsd:restriction>
      </xsd:simpleType>
    </xsd:element>
    <xsd:element name="delayTolerance" type="xsd:long" minOccurs="0" />
    <xsd:element name="switchingSubDomainCCIEType" minOccurs="0">
      <xsd:simpleType>
```

```
<xsd:restriction base="xsd:string">
  <xsd:enumeration value="isdn" />
  <xsd:enumeration value="atm" />
  <xsd:enumeration value="isoEthernet" />
  <xsd:enumeration value="rsvp" />
  <xsd:enumeration value="other" />
</xsd:restriction>
</xsd:simpleType>
</xsd:element>
<xsd:element name="switchingSubDomainInformationElements" minOccurs="0">
  <xsd:simpleType>
    <xsd:restriction base="xsd:hexBinary" />
  </xsd:simpleType>
</xsd:element>
</xsd:sequence>
</xsd:complexType>

<xsd:element name="mediaCallCharacteristics" type="csta:MediaCallCharacteristics" />

<xsd:complexType name="MediaClass">
  <xsd:sequence>
    <xsd:element name="voice" type="xsd:boolean" minOccurs="0" />
    <xsd:element name="data" type="xsd:boolean" minOccurs="0" />
    <xsd:element name="image" type="xsd:boolean" minOccurs="0" />
    <xsd:element name="audio" type="xsd:boolean" minOccurs="0" />
    <xsd:element name="other" type="xsd:boolean" minOccurs="0" />
    <xsd:element name="notKnown" type="xsd:boolean" minOccurs="0" />
  </xsd:sequence>
</xsd:complexType>

</xsd:schema>
```

9.16 Physical device features

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-physical-device-features</xsd:documentation>
  </xsd:annotation>

  <xsd:complexType name="AuditoryApparatusList">
    <xsd:sequence>
      <xsd:element name="auditoryApparatusItem" type="csta:AuditoryApparatusItem" minOccurs="0"
maxOccurs="unbounded"/>
    </xsd:sequence>
  </xsd:complexType>

  <xsd:complexType name="AuditoryApparatusItem">
    <xsd:sequence>
      <xsd:element name="auditoryApparatus" type="csta:AuditoryApparatusID"/>
      <xsd:element name="auditoryApparatusType">
        <xsd:simpleType>
          <xsd:restriction base="xsd:string">
            <xsd:enumeration value="speakerphone"/>
            <xsd:enumeration value="handset"/>
            <xsd:enumeration value="headset"/>
            <xsd:enumeration value="speakerOnlyPhone"/>
            <xsd:enumeration value="other"/>
          </xsd:restriction>
        </xsd:simpleType>
      </xsd:element>
      <xsd:element name="speaker">
        <xsd:complexType>
          <xsd:sequence>
            <xsd:element name="present" type="xsd:boolean" minOccurs="0"/>
            <xsd:element name="volumeSettable" type="xsd:boolean" minOccurs="0"/>
            <xsd:element name="volumeReadable" type="xsd:boolean" minOccurs="0"/>
            <xsd:element name="muteSettable" type="xsd:boolean" minOccurs="0"/>
            <xsd:element name="muteReadable" type="xsd:boolean" minOccurs="0"/>
          </xsd:sequence>
        </xsd:complexType>
      </xsd:element>
      <xsd:element name="microphone">
        <xsd:complexType>
          <xsd:sequence>
            <xsd:element name="present" type="xsd:boolean" minOccurs="0"/>
            <xsd:element name="gainSettable" type="xsd:boolean" minOccurs="0"/>
            <xsd:element name="gainReadable" type="xsd:boolean" minOccurs="0"/>
            <xsd:element name="muteSettable" type="xsd:boolean" minOccurs="0"/>
            <xsd:element name="muteReadable" type="xsd:boolean" minOccurs="0"/>
          </xsd:sequence>
        </xsd:complexType>
      </xsd:element>
      <xsd:element name="hookswitch">
        <xsd:complexType>
          <xsd:sequence>
            <xsd:element name="hookswitchSettable" type="xsd:boolean" minOccurs="0"/>
            <xsd:element name="hookswitchOnHook" type="xsd:boolean" minOccurs="0"/>
          </xsd:sequence>
        </xsd:complexType>
      </xsd:element>
      <xsd:element name="hookswitchID" type="csta:HookswitchID"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:schema>
```

```
<xsd:simpleType name="AuditoryApparatusID">
  <xsd:restriction base="xsd:string">
    <xsd:minLength value="0" />
    <xsd:maxLength value="4" />
    <xsd:pattern value="\d*" />
  </xsd:restriction>
</xsd:simpleType>

<xsd:simpleType name="ButtonID">
  <xsd:restriction base="xsd:string">
    <xsd:minLength value="0" />
    <xsd:maxLength value="4" />
    <xsd:pattern value="\d*" />
  </xsd:restriction>
</xsd:simpleType>

<xsd:simpleType name="CharacterSet">
  <xsd:restriction base="xsd:string">
    <xsd:enumeration value="ascii" />
    <xsd:enumeration value="unicode" />
    <xsd:enumeration value="proprietary" />
  </xsd:restriction>
</xsd:simpleType>

<xsd:simpleType name="DisplayID">
  <xsd:restriction base="xsd:string">
    <xsd:minLength value="0" />
    <xsd:maxLength value="4" />
    <xsd:pattern value="\d*" />
  </xsd:restriction>
</xsd:simpleType>

<xsd:simpleType name="HookswitchID">
  <xsd:restriction base="xsd:string">
    <xsd:minLength value="0" />
    <xsd:maxLength value="4" />
    <xsd:pattern value="\d*" />
  </xsd:restriction>
</xsd:simpleType>

<xsd:simpleType name="LampBrightness">
  <xsd:restriction base="xsd:string">
    <xsd:enumeration value="unspecified" />
    <xsd:enumeration value="dim" />
    <xsd:enumeration value="bright" />
  </xsd:restriction>
</xsd:simpleType>

<xsd:simpleType name="LampColor">
  <xsd:annotation>
    <xsd:documentation>named values are represented as integer</xsd:documentation>
  </xsd:annotation>
  <xsd:restriction base="xsd:short">
    <xsd:minInclusive value="0" />
    <xsd:maxInclusive value="100" />
  </xsd:restriction>
</xsd:simpleType>

<xsd:simpleType name="LampID">
  <xsd:restriction base="xsd:string">
    <xsd:minLength value="0" />
    <xsd:maxLength value="4" />
    <xsd:pattern value="\d*" />
  </xsd:restriction>
</xsd:simpleType>

<xsd:simpleType name="LampMode">
  <xsd:annotation>
```

```

    <xsd:documentation>named values are represented as integer</xsd:documentation>
  </xsd:annotation>
  <xsd:restriction base="xsd:short">
    <xsd:minInclusive value="0"/>
    <xsd:maxInclusive value="100"/>
  </xsd:restriction>
</xsd:simpleType>

<xsd:complexType name="MicrophoneGain">
  <xsd:choice>
    <xsd:element name="micGainAbs" type="csta:MicGainAbs"/>
    <xsd:element name="micGainInc" type="csta:MicGainInc"/>
  </xsd:choice>
</xsd:complexType>

<xsd:simpleType name="MicGainInc">
  <xsd:restriction base="xsd:string">
    <xsd:enumeration value="increment"/>
    <xsd:enumeration value="decrement"/>
  </xsd:restriction>
</xsd:simpleType>

<xsd:simpleType name="MicGainAbs">
  <xsd:restriction base="xsd:short">
    <xsd:minInclusive value="0"/>
    <xsd:maxInclusive value="100"/>
  </xsd:restriction>
</xsd:simpleType>

<xsd:simpleType name="RingerID">
  <xsd:restriction base="xsd:string">
    <xsd:minLength value="0"/>
    <xsd:maxLength value="4"/>
    <xsd:pattern value="\d*" />
  </xsd:restriction>
</xsd:simpleType>

<xsd:simpleType name="RingMode">
  <xsd:restriction base="xsd:string">
    <xsd:enumeration value="ringing"/>
    <xsd:enumeration value="notRinging"/>
  </xsd:restriction>
</xsd:simpleType>

<xsd:complexType name="Volume">
  <xsd:choice>
    <xsd:element name="volAbs" type="csta:VolAbs"/>
    <xsd:element name="volInc" type="csta:VolInc"/>
  </xsd:choice>
</xsd:complexType>

<xsd:simpleType name="VolInc">
  <xsd:restriction base="xsd:string">
    <xsd:enumeration value="increment"/>
    <xsd:enumeration value="decrement"/>
  </xsd:restriction>
</xsd:simpleType>

<xsd:simpleType name="VolAbs">
  <xsd:restriction base="xsd:short">
    <xsd:minInclusive value="0"/>
    <xsd:maxInclusive value="100"/>
  </xsd:restriction>
</xsd:simpleType>

</xsd:schema>
```

9.17 Data Collection

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-data-collection</xsd:documentation>
  </xsd:annotation>

  <xsd:simpleType name="DcollCrossRefID">
    <xsd:restriction base="xsd:string">
      <xsd:minLength value="0"/>
      <xsd:maxLength value="4"/>
      <xsd:pattern value="\d*" />
    </xsd:restriction>
  </xsd:simpleType>

  <xsd:element name="dcollCrossRefID" type="csta:DcollCrossRefID"/>

</xsd:schema>
```

9.18 Event Cause

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-event-cause</xsd:documentation>
  </xsd:annotation>

  <xsd:simpleType name="EventCause">
    <xsd:restriction base="xsd:string">
      <xsd:enumeration value="aCDBusy" />
      <xsd:enumeration value="aCDForward" />
      <xsd:enumeration value="aCDSaturated" />
      <xsd:enumeration value="activeParticipation" />
      <xsd:enumeration value="alertTimeExpired" />
      <xsd:enumeration value="alternate" />
      <xsd:enumeration value="autoWork" />
      <xsd:enumeration value="blocked" />
      <xsd:enumeration value="busy" />
      <xsd:enumeration value="callBack" />
      <xsd:enumeration value="callCancelled" />
      <xsd:enumeration value="callForward" />
      <xsd:enumeration value="callForwardImmediate" />
      <xsd:enumeration value="callForwardBusy" />
      <xsd:enumeration value="callForwardNoAnswer" />
      <xsd:enumeration value="callNotAnswered" />
      <xsd:enumeration value="callPickup" />
      <xsd:enumeration value="campOn" />
      <xsd:enumeration value="campOnTrunks" />
      <xsd:enumeration value="characterCountReached" />
      <xsd:enumeration value="conference" />
      <xsd:enumeration value="consultation" />
      <xsd:enumeration value="destDetected" />
      <xsd:enumeration value="destNotObtainable" />
      <xsd:enumeration value="destOutOfOrder" />
      <xsd:enumeration value="distributed" />
      <xsd:enumeration value="distributionDelay" />
      <xsd:enumeration value="doNotDisturb" />
      <xsd:enumeration value="dTMFDigitDetected" />
      <xsd:enumeration value="durationExceeded" />
      <xsd:enumeration value="endOfMessageDetected" />
      <xsd:enumeration value="enteringDistribution" />
      <xsd:enumeration value="forcedPause" />
      <xsd:enumeration value="forcedTransition" />
      <xsd:enumeration value="incompatibleDestination" />
      <xsd:enumeration value="intrude" />
      <xsd:enumeration value="invalidAccountCode" />
      <xsd:enumeration value="invalidNumberFormat" />
      <xsd:enumeration value="joinCall" />
      <xsd:enumeration value="keyOperation" />
      <xsd:enumeration value="keyOperationInUse" />
      <xsd:enumeration value="lockout" />
      <xsd:enumeration value="maintenance" />
      <xsd:enumeration value="makeCall" />
      <xsd:enumeration value="makePredictiveCall" />
      <xsd:enumeration value="messageDurationExceeded" />
      <xsd:enumeration value="messageSizeExceeded" />
      <xsd:enumeration value="multipleAlerting" />
      <xsd:enumeration value="multipleQueuing" />
      <xsd:enumeration value="networkCongestion" />
      <xsd:enumeration value="networkDialling" />
      <xsd:enumeration value="networkNotObtainable" />
      <xsd:enumeration value="networkOutOfOrder" />
    </xsd:restriction>
  </xsd:simpleType>
</xsd:schema>
```



```
<xsd:enumeration value="networkSignal"/>
<xsd:enumeration value="newCall"/>
<xsd:enumeration value="nextMessage"/>
<xsd:enumeration value="noAvailableAgents"/>
<xsd:enumeration value="normal"/>
<xsd:enumeration value="normalClearing"/>
<xsd:enumeration value="noSpeechDetected"/>
<xsd:enumeration value="notAvaliableBearerService"/>
<xsd:enumeration value="notSupportedBearerService"/>
<xsd:enumeration value="numberChanged"/>
<xsd:enumeration value="numberUnallocated"/>
<xsd:enumeration value="overflow"/>
<xsd:enumeration value="override"/>
<xsd:enumeration value="park"/>
<xsd:enumeration value="queueCleared"/>
<xsd:enumeration value="recall"/>
<xsd:enumeration value="redirected"/>
<xsd:enumeration value="remainsInQueue"/>
<xsd:enumeration value="reorderTone"/>
<xsd:enumeration value="reserved"/>
<xsd:enumeration value="resourcesNotAvailable"/>
<xsd:enumeration value="selectedTrunkBusy"/>
<xsd:enumeration value="silentParticipation"/>
<xsd:enumeration value="singleStepConference"/>
<xsd:enumeration value="singleStepTransfer"/>
<xsd:enumeration value="speechDetected"/>
<xsd:enumeration value="suspend"/>
<xsd:enumeration value="switchingFunctionTerminated"/>
<xsd:enumeration value="terminationCharacterReceived"/>
<xsd:enumeration value="timeout"/>
<xsd:enumeration value="transfer"/>
<xsd:enumeration value="trunksBusy"/>
<xsd:enumeration value="unauthorisedBearerService"/>
</xsd:restriction>
</xsd:simpleType>

<xsd:element name="cause" type="csta:EventCause"/>
</xsd:schema>
```

9.19 Error Value

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-error-definition</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="empty.xsd"/>

  <xsd:simpleType name="OperationErrors">
    <xsd:restriction base="xsd:string">
      <xsd:enumeration value="generic"/>
      <xsd:enumeration value="atLeastOneConditionalParameterNotProvided"/>
      <xsd:enumeration value="featureAlreadySet"/>
      <xsd:enumeration value="invalidMsgID"/>
      <xsd:enumeration value="invalidParameterValue"/>
      <xsd:enumeration value="invalidAccountCode"/>
      <xsd:enumeration value="invalidAgentGroup"/>
      <xsd:enumeration value="invalidAgentIdentifier"/>
      <xsd:enumeration value="invalidAgentPassword"/>
      <xsd:enumeration value="invalidAgentState"/>
      <xsd:enumeration value="invalidAlertTime"/>
      <xsd:enumeration value="invalidAllocationState"/>
      <xsd:enumeration value="invalidAuthCode"/>
      <xsd:enumeration value="invalidAutoAnswer"/>
      <xsd:enumeration value="invalidBitRate"/>
      <xsd:enumeration value="invalidButtonIdentifier"/>
      <xsd:enumeration value="invalidCallType"/>
      <xsd:enumeration value="invalidConnectionRate"/>
      <xsd:enumeration value="invalidConsultPurpose"/>
      <xsd:enumeration value="invalidCorrelatorData"/>
      <xsd:enumeration value="invalidCrossRefID"/>
      <xsd:enumeration value="invalidDelayTolerance"/>
      <xsd:enumeration value="invalidDestination"/>
      <xsd:enumeration value="invalidDestinationDetect"/>
      <xsd:enumeration value="invalidDoNotDisturb"/>
      <xsd:enumeration value="invalidEscapeCrossRefID"/>
      <xsd:enumeration value="invalidFeature"/>
      <xsd:enumeration value="invalidFile"/>
      <xsd:enumeration value="invalidFlowDirection"/>
      <xsd:enumeration value="invalidForwardingDestination"/>
      <xsd:enumeration value="invalidForwardingFlag"/>
      <xsd:enumeration value="invalidForwardingType"/>
      <xsd:enumeration value="invalidHookswitchType"/>
      <xsd:enumeration value="invalidHookswitchComponent"/>
      <xsd:enumeration value="invalidLampMode"/>
      <xsd:enumeration value="invalidLampID"/>
      <xsd:enumeration value="invalidMessageWaitingSetting"/>
      <xsd:enumeration value="invalidMicrophoneGain"/>
      <xsd:enumeration value="invalidMicrophoneMute"/>
      <xsd:enumeration value="invalidMonitorCrossRefID"/>
      <xsd:enumeration value="invalidMonitorFilter"/>
      <xsd:enumeration value="invalidMonitorObject"/>
      <xsd:enumeration value="invalidMonitorType"/>
      <xsd:enumeration value="invalidNumberOfChannels"/>
      <xsd:enumeration value="invalidParticipationType"/>
      <xsd:enumeration value="invalidRemainRetry"/>
      <xsd:enumeration value="invalidRingCount"/>
      <xsd:enumeration value="invalidRingPattern"/>
      <xsd:enumeration value="invalidRingVolume"/>
      <xsd:enumeration value="invalidRouteingAlgorithm"/>
      <xsd:enumeration value="invalidRouteingCrossRefID"/>
    </xsd:restriction>
  </xsd:simpleType>
</xsd:schema>
```

```
<xsd:enumeration value="invalidRouteRegistrationCrossRefID"/>
<xsd:enumeration value="invalidSpeakerVolume"/>
<xsd:enumeration value="invalidSpeakerMute"/>
<xsd:enumeration value="invalidSwitchingSubdomainCharsType"/>
<xsd:enumeration value="invalidObjectType"/>
<xsd:enumeration value="invalidActiveCallObject"/>
<xsd:enumeration value="invalidCalledDeviceObjectType"/>
<xsd:enumeration value="invalidCallingDeviceObjectType"/>
<xsd:enumeration value="invalidCallToBePickedUpObjectType"/>
<xsd:enumeration value="invalidCallToDivertObjectType"/>
<xsd:enumeration value="invalidCallToParkObjectType"/>
<xsd:enumeration value="invalidDestinationDeviceObject"/>
<xsd:enumeration value="invalidHeldCallObject"/>
<xsd:enumeration value="invalidMonitorObjectType"/>
<xsd:enumeration value="invalidParkToObjectType"/>
<xsd:enumeration value="messageIDRequired"/>
<xsd:enumeration value="notDifferentDevices"/>
<xsd:enumeration value="notSameDevice"/>
<xsd:enumeration value="objectNotKnown"/>
<xsd:enumeration value="invalidCallID"/>
<xsd:enumeration value="invalidActiveCallID"/>
<xsd:enumeration value="invalidHeldCallID"/>
<xsd:enumeration value="invalidConnectionID"/>
<xsd:enumeration value="invalidActiveConnectionID"/>
<xsd:enumeration value="invalidHeldConnectionID"/>
<xsd:enumeration value="invalidDeviceID"/>
<xsd:enumeration value="invalidActiveDeviceID"/>
<xsd:enumeration value="invalidCalledDeviceID"/>
<xsd:enumeration value="invalidCallingDeviceID"/>
<xsd:enumeration value="invalidCallToParkDeviceID"/>
<xsd:enumeration value="invalidDestinationDeviceID"/>
<xsd:enumeration value="invalidDivertingDeviceID"/>
<xsd:enumeration value="invalidHeldDeviceID"/>
<xsd:enumeration value="invalidParkToDeviceID"/>
<xsd:enumeration value="invalidPickUpDeviceID"/>
<xsd:enumeration value="parameterNotSupported"/>
<xsd:enumeration value="accountCodeNotSupported"/>
<xsd:enumeration value="agentGroupNotSupported"/>
<xsd:enumeration value="agentPasswordNotSupported"/>
<xsd:enumeration value="agentStateNotSupported"/>
<xsd:enumeration value="alertTimeNotSupported"/>
<xsd:enumeration value="allocationNotSupported"/>
<xsd:enumeration value="authorisationCodeNotSupported"/>
<xsd:enumeration value="autoAnswerNotSupported"/>
<xsd:enumeration value="bitRateNotSupported"/>
<xsd:enumeration value="buttonNotSupported"/>
<xsd:enumeration value="callTypeNotSupported"/>
<xsd:enumeration value="charactersToSendNotSupported"/>
<xsd:enumeration value="connectionRateNotSupported"/>
<xsd:enumeration value="connectionReservationNotSupported"/>
<xsd:enumeration value="consultPurposeNotSupported"/>
<xsd:enumeration value="correlatorDataNotSupported"/>
<xsd:enumeration value="delayToleranceNotSupported"/>
<xsd:enumeration value="destinationDetectNotSupported"/>
<xsd:enumeration value="digitModeNotSupported"/>
<xsd:enumeration value="errorValueNotSupported"/>
<xsd:enumeration value="flowDirectionNotSupported"/>
<xsd:enumeration value="forwardingDestinationNotSupported"/>
<xsd:enumeration value="lampNotSupported"/>
<xsd:enumeration value="monitorTypeNotSupported"/>
<xsd:enumeration value="numberOfChannelsNotSupported"/>
<xsd:enumeration value="parameterTypeNotSupported"/>
<xsd:enumeration value="priorityNotSupported"/>
<xsd:enumeration value="privateDataNotSupported"/>
<xsd:enumeration value="pulseDurationNotSupported"/>
<xsd:enumeration value="pulseRateNotSupported"/>
<xsd:enumeration value="remainRetryNotSupported"/>
<xsd:enumeration value="ringCountNotSupported"/>
```

```
<xsd:enumeration value="routeUsedNotSupported" />
<xsd:enumeration value="securityNotSupported" />
<xsd:enumeration value="swSubdomainCCIETypeNotSupported" />
<xsd:enumeration value="toneDurationNotSupported" />
<xsd:enumeration value="sysStatRegIDNotSupported" />
<xsd:enumeration value="userDataNotSupported" />
<xsd:enumeration value="privilegeViolationSpecifiedDevice" />
<xsd:enumeration value="privilegeViolationActiveDevice" />
<xsd:enumeration value="privilegeViolationCalledDevice" />
<xsd:enumeration value="privilegeViolationCallingDevice" />
<xsd:enumeration value="privilegeViolationCallToParkDevice" />
<xsd:enumeration value="privilegeViolationDestinationDevice" />
<xsd:enumeration value="privilegeViolationOnDivertingDevice" />
<xsd:enumeration value="privilegeViolationHeldDevice" />
<xsd:enumeration value="privilegeViolationOnParkToDevice" />
<xsd:enumeration value="privilegeViolationPickUpDevice" />
<xsd:enumeration value="routeingTimerExpired" />
<xsd:enumeration value="requestIncompatibleWithObject" />
<xsd:enumeration value="requestIncompatibleWithConnection" />
<xsd:enumeration value="requestIncompatibleWithActiveConnection" />
<xsd:enumeration value="requestIncompatibleWithHeldConnection" />
<xsd:enumeration value="requestIncompatibleWithDevice" />
<xsd:enumeration value="requestIncompatibleWithCalledDevice" />
<xsd:enumeration value="requestIncompatibleWithCallingDevice" />
<xsd:enumeration value="requestIncompatibleWithSubjectDevice" />
<xsd:enumeration value="requestIncompatibleWithActiveDevice" />
<xsd:enumeration value="requestIncompatibleWithCallToParkDevice" />
<xsd:enumeration value="requestIncompatibleWithDestinationDevice" />
<xsd:enumeration value="requestIncompatibleWithDivertingDevice" />
<xsd:enumeration value="requestIncompatibleWithHeldDevice" />
<xsd:enumeration value="requestIncompatibleWithMedia" />
<xsd:enumeration value="requestIncompatibleWithParkToDevice" />
<xsd:enumeration value="requestIncompatibleWithPickupDevice" />
<xsd:enumeration value="serviceNotSupported" />
<xsd:enumeration value="securityViolation" />
<xsd:enumeration value="valueOutOfRange" />
<xsd:enumeration value="agentStateOutOfRange" />
<xsd:enumeration value="alertTimeOutOfRange" />
<xsd:enumeration value="allocationOutOfRange" />
<xsd:enumeration value="autoAnswerOutOfRange" />
<xsd:enumeration value="bitRateOutOfRange" />
<xsd:enumeration value="callTypeOutOfRange" />
<xsd:enumeration value="connectionRateOutOfRange" />
<xsd:enumeration value="connectionReservoationOutOfRange" />
<xsd:enumeration value="consultPurposeOutOfRange" />
<xsd:enumeration value="correlatorDataOutOfRange" />
<xsd:enumeration value="delayToleranceOutOfRange" />
<xsd:enumeration value="destinationDetectOutOfRange" />
<xsd:enumeration value="digitModeOutOfRange" />
<xsd:enumeration value="doNotDisturbOutOfRange" />
<xsd:enumeration value="flowDirectionOutOfRange" />
<xsd:enumeration value="forwardingFlagOutOfRange" />
<xsd:enumeration value="forwardingTypeOutOfRange" />
<xsd:enumeration value="hookswitchComponentOutOfRange" />
<xsd:enumeration value="hookswitchTypeOutOfRange" />
<xsd:enumeration value="lampModeOutOfRange" />
<xsd:enumeration value="messageWaitingSettingOutOfRange" />
<xsd:enumeration value="micGainOutOfRange" />
<xsd:enumeration value="micMuteOutOfRange" />
<xsd:enumeration value="monitorTypeOutOfRange" />
<xsd:enumeration value="numberOfChannelsOutOfRange" />
<xsd:enumeration value="participationTypeOutOfRange" />
<xsd:enumeration value="pulseDurationOutOfRange" />
<xsd:enumeration value="pulseRateOutOfRange" />
<xsd:enumeration value="ringCountOutOfRange" />
<xsd:enumeration value="ringPatternOutOfRange" />
<xsd:enumeration value="ringVolumnOutOfRange" />
<xsd:enumeration value="routeingAlgorithmOutOfRange" />
```

```
<xsd:enumeration value="speakerMuteOutOfRange" />
<xsd:enumeration value="speakerVolumeOutOfRange" />
<xsd:enumeration value="switchingCcittTypeOutOfRange" />
<xsd:enumeration value="systemStatusOutOfRange" />
<xsd:enumeration value="toneCharacterOutOfRange" />
<xsd:enumeration value="toneDurationOutOfRange" />
</xsd:restriction>
</xsd:simpleType>

<xsd:simpleType name="SecurityErrors">
  <xsd:restriction base="xsd:string">
    <xsd:enumeration value="generic" />
    <xsd:enumeration value="sequenceNumberViolated" />
    <xsd:enumeration value="timeStampViolated" />
    <xsd:enumeration value="securityInfoViolated" />
  </xsd:restriction>
</xsd:simpleType>

<xsd:simpleType name="StateIncompatibilityErrors">
  <xsd:restriction base="xsd:string">
    <xsd:enumeration value="generic" />
    <xsd:enumeration value="invalidObjectState" />
    <xsd:enumeration value="invalidDeviceState" />
    <xsd:enumeration value="connectedCallExists" />
    <xsd:enumeration value="invalidActiveDeviceState" />
    <xsd:enumeration value="invalidCalledDeviceState" />
    <xsd:enumeration value="invalidCallingDeviceState" />
    <xsd:enumeration value="invalidCallToParkDeviceState" />
    <xsd:enumeration value="invalidDestinationDeviceState" />
    <xsd:enumeration value="invalidDivertingDeviceState" />
    <xsd:enumeration value="invalidHeldDeviceState" />
    <xsd:enumeration value="invalidParkToDeviceState" />
    <xsd:enumeration value="invalidConnectionState" />
    <xsd:enumeration value="invalidActiveConnectionState" />
    <xsd:enumeration value="invalidConnectionIDForActiveCall" />
    <xsd:enumeration value="invalidHeldConnectionState" />
    <xsd:enumeration value="noActiveCall" />
    <xsd:enumeration value="noCallToAnswer" />
    <xsd:enumeration value="noCallToClear" />
    <xsd:enumeration value="noCallToComplete" />
    <xsd:enumeration value="noConnectionToClear" />
    <xsd:enumeration value="noHeldCall" />
    <xsd:enumeration value="incorrectMsgState" />
    <xsd:enumeration value="beginningOfMsg" />
    <xsd:enumeration value="endOfMsg" />
    <xsd:enumeration value="msgSuspended" />
    <xsd:enumeration value="notAbleToPlay" />
    <xsd:enumeration value="notAbleToResume" />
  </xsd:restriction>
</xsd:simpleType>

<xsd:simpleType name="SystemResourceAvailabilityErrors">
  <xsd:restriction base="xsd:string">
    <xsd:enumeration value="generic" />
    <xsd:enumeration value="resourceBusy" />
    <xsd:enumeration value="internalResourceBusy" />
    <xsd:enumeration value="classifierBusy" />
    <xsd:enumeration value="noMediaChannelsAvailable" />
    <xsd:enumeration value="channelsInUseForBridgedDevices" />
    <xsd:enumeration value="channelsInUseForData" />
    <xsd:enumeration value="toneDetectorBusy" />
    <xsd:enumeration value="toneGeneratorBusy" />
    <xsd:enumeration value="networkBusy" />
    <xsd:enumeration value="resourceOutOfService" />
    <xsd:enumeration value="deviceOutOfService" />
    <xsd:enumeration value="activeDeviceOutOfService" />
    <xsd:enumeration value="calledDeviceOutOfService" />
    <xsd:enumeration value="callingDeviceOutOfService" />
  </xsd:restriction>
</xsd:simpleType>
```

```
<xsd:enumeration value="callToParkDeviceOutOfService"/>
<xsd:enumeration value="destinationDeviceOutOfService"/>
<xsd:enumeration value="divertingDeviceOutOfService"/>
<xsd:enumeration value="heldDeviceOutOfService"/>
<xsd:enumeration value="parkToDeviceOutOfService"/>
<xsd:enumeration value="pickupDeviceOutOfService"/>
<xsd:enumeration value="networkOutOfService"/>
<xsd:enumeration value="otherResourceOutOfService"/>
<xsd:enumeration value="resourceLimitExceeded"/>
<xsd:enumeration value="overallMonitorLimitExceeded"/>
<xsd:enumeration value="conferenceMemberLimitExceeded"/>
<xsd:enumeration value="registrationLimitExceeded"/>
</xsd:restriction>
</xsd:simpleType>

<xsd:simpleType name="SubscribedResourceAvailabilityErrors">
  <xsd:restriction base="xsd:string">
    <xsd:enumeration value="generic"/>
    <xsd:enumeration value="objectMonitorLimitExceeded"/>
    <xsd:enumeration value="trunkLimitExceeded"/>
    <xsd:enumeration value="outstandingReqLimitExceeded"/>
    <xsd:enumeration value="objectRegistrationLimitExceeded"/>
  </xsd:restriction>
</xsd:simpleType>

<xsd:simpleType name="PerformanceManagementErrors">
  <xsd:restriction base="xsd:string">
    <xsd:enumeration value="generic"/>
    <xsd:enumeration value="performanceLimitExceeded"/>
  </xsd:restriction>
</xsd:simpleType>

<xsd:simpleType name="PrivateDataInfoErrors">
  <xsd:restriction base="xsd:string">
    <xsd:enumeration value="cSTAPrivateDataInfoError"/>
  </xsd:restriction>
</xsd:simpleType>

<xsd:element name="CSTACode" type="csta:UniversalFailure"/>
<xsd:complexType name="UniversalFailure">
  <xsd:choice>
    <xsd:element name="operation" type="csta:OperationErrors"/>
    <xsd:element name="security" type="csta:SecurityErrors"/>
    <xsd:element name="stateIncompatibility" type="csta:StateIncompatibilityErrors"/>
    <xsd:element name="systemResourceAvailability"
      type="csta:SystemResourceAvailabilityErrors"/>
    <xsd:element name="subscribedResourceAvailability"
      type="csta:SubscribedResourceAvailabilityErrors"/>
    <xsd:element name="performanceManagement" type="csta:PerformanceManagementErrors"/>
    <xsd:element name="privateData" type="csta:PrivateDataInfoErrors"/>
    <xsd:element name="unspecified" type="csta:Empty"/>
  </xsd:choice>
</xsd:complexType>

<xsd:complexType name="ErrorValue">
  <xsd:complexContent>
    <xsd:extension base="csta:UniversalFailure"/>
  </xsd:complexContent>
</xsd:complexType>

</xsd:schema>
```

9.20 Empty (null element)

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>replacement for ASN.1 NULL type and helper type for representation of the
ASN.1 BIT STRING type</xsd:documentation>
  </xsd:annotation>

  <xsd:complexType name="Empty"/>

</xsd:schema>
```

10 Top level schema

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:import namespace="http://schemas.xmlsoap.org/soap/envelope/" />

  <xsd:annotation>
    <xsd:documentation>csta</xsd:documentation>
  </xsd:annotation>

  <xsd:annotation>
    <xsd:documentation>Capability Exchange Services</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="get-logical-device-information.xsd" />
  <xsd:include schemaLocation="get-physical-device-information.xsd" />
  <xsd:include schemaLocation="get-switching-function-capabilities.xsd" />
  <xsd:include schemaLocation="get-switching-function-devices.xsd" />
  <xsd:include schemaLocation="switching-function-devices.xsd" />

  <xsd:annotation>
    <xsd:documentation>System Services</xsd:documentation>
  </xsd:annotation>
  <xsd:include schemaLocation="change-system-status-filter.xsd" />
  <xsd:include schemaLocation="system-register.xsd" />
  <xsd:include schemaLocation="system-register-abort.xsd" />
  <xsd:include schemaLocation="system-register-cancel.xsd" />
  <xsd:include schemaLocation="request-system-status.xsd" />
  <xsd:include schemaLocation="switching-functions-capabilities-changed.xsd" />
  <xsd:include schemaLocation="switching-functions-devices-changed.xsd" />
  <xsd:annotation>
    <xsd:documentation>Monitoring Services</xsd:documentation>
  </xsd:annotation>
  <xsd:include schemaLocation="change-monitor-filter.xsd" />
  <xsd:include schemaLocation="monitor-start.xsd" />
  <xsd:include schemaLocation="monitor-stop.xsd" />
  <xsd:annotation>
    <xsd:documentation>Snapshot Services</xsd:documentation>
  </xsd:annotation>
  <xsd:include schemaLocation="snapshot-call.xsd" />
  <xsd:include schemaLocation="snapshot-device.xsd" />
  <xsd:include schemaLocation="snapshot-call-data.xsd" />
  <xsd:include schemaLocation="snapshot-device-data.xsd" />
  <xsd:annotation>
    <xsd:documentation>Call Control Services</xsd:documentation>
  </xsd:annotation>
  <xsd:include schemaLocation="accept-call.xsd" />
  <xsd:include schemaLocation="alternate-call.xsd" />
  <xsd:include schemaLocation="answer-call.xsd" />
  <xsd:include schemaLocation="call-back-call-related.xsd" />
  <xsd:include schemaLocation="call-back-message-call-related.xsd" />
  <xsd:include schemaLocation="camp-on-call.xsd" />
  <xsd:include schemaLocation="clear-call.xsd" />
  <xsd:include schemaLocation="clear-connection.xsd" />
  <xsd:include schemaLocation="conference-call.xsd" />
  <xsd:include schemaLocation="consultation-call.xsd" />
  <xsd:include schemaLocation="deflect-call.xsd" />
  <xsd:include schemaLocation="dial-digits.xsd" />
  <xsd:include schemaLocation="directed-pickup-call.xsd" />
  <xsd:include schemaLocation="group-pickup-call.xsd" />
  <xsd:include schemaLocation="hold-call.xsd" />
  <xsd:include schemaLocation="intrude-call.xsd" />
  <xsd:include schemaLocation="join-call.xsd" />
```



```
<xsd:include schemaLocation="make-call.xsd"/>
<xsd:include schemaLocation="make-predictive-call.xsd"/>
<xsd:include schemaLocation="park-call.xsd"/>
<xsd:include schemaLocation="reconnect-call.xsd"/>
<xsd:include schemaLocation="retrieve-call.xsd"/>
<xsd:include schemaLocation="single-step-conference-call.xsd"/>
<xsd:include schemaLocation="single-step-transfer.xsd"/>
<xsd:include schemaLocation="transfer-call.xsd"/>
<xsd:annotation>
  <xsd:documentation>Call Control Events</xsd:documentation>
</xsd:annotation>
<xsd:include schemaLocation="bridged-event.xsd"/>
<xsd:include schemaLocation="call-cleared-event.xsd"/>
<xsd:include schemaLocation="conferenced-event.xsd"/>
<xsd:include schemaLocation="connection-cleared-event.xsd"/>
<xsd:include schemaLocation="delivered-event.xsd"/>
<xsd:include schemaLocation="digits-dialed-event.xsd"/>
<xsd:include schemaLocation="diverted-event.xsd"/>
<xsd:include schemaLocation="failed-event.xsd"/>
<xsd:include schemaLocation="established-event.xsd"/>
<xsd:include schemaLocation="held-event.xsd"/>
<xsd:include schemaLocation="network-capabilities-changed-event.xsd"/>
<xsd:include schemaLocation="network-reached-event.xsd"/>
<xsd:include schemaLocation="offered-event.xsd"/>
<xsd:include schemaLocation="originated-event.xsd"/>
<xsd:include schemaLocation="queued-event.xsd"/>
<xsd:include schemaLocation="retrieved-event.xsd"/>
<xsd:include schemaLocation="service-initiated-event.xsd"/>
<xsd:include schemaLocation="transferred-event.xsd"/>
<xsd:annotation>
  <xsd:documentation>Call Associated Services</xsd:documentation>
</xsd:annotation>
<xsd:include schemaLocation="associate-data.xsd"/>
<xsd:include schemaLocation="cancel-telephony-tones.xsd"/>
<xsd:include schemaLocation="generate-digits.xsd"/>
<xsd:include schemaLocation="generate-telephony-tones.xsd"/>
<xsd:include schemaLocation="send-user-information.xsd"/>
<xsd:annotation>
  <xsd:documentation>Call Associated Events</xsd:documentation>
</xsd:annotation>
<xsd:include schemaLocation="call-information-event.xsd"/>
<xsd:include schemaLocation="charging-event.xsd"/>
<xsd:include schemaLocation="digits-generated-event.xsd"/>
<xsd:include schemaLocation="telephony-tones-generated-event.xsd"/>
<xsd:include schemaLocation="service-completion-failure-event.xsd"/>
<xsd:annotation>
  <xsd:documentation>Media Attachment Services</xsd:documentation>
</xsd:annotation>
<xsd:include schemaLocation="attach-media-service.xsd"/>
<xsd:include schemaLocation="detach-media-service.xsd"/>
<xsd:annotation>
  <xsd:documentation>Media Attachment Events</xsd:documentation>
</xsd:annotation>
<xsd:include schemaLocation="media-attached-event.xsd"/>
<xsd:include schemaLocation="media-detached-event.xsd"/>
<xsd:annotation>
  <xsd:documentation>Routeing Services</xsd:documentation>
</xsd:annotation>
<xsd:include schemaLocation="route-register.xsd"/>
<xsd:include schemaLocation="route-register-abort.xsd"/>
<xsd:include schemaLocation="route-register-cancel.xsd"/>
<xsd:include schemaLocation="re-route.xsd"/>
<xsd:include schemaLocation="route-end.xsd"/>
<xsd:include schemaLocation="route-reject.xsd"/>
<xsd:include schemaLocation="route-request.xsd"/>
<xsd:include schemaLocation="route-select.xsd"/>
<xsd:include schemaLocation="route-used.xsd"/>
<xsd:annotation>
```

```
<xsd:documentation>Physical Device Services</xsd:documentation>
</xsd:annotation>
<xsd:include schemaLocation="button-press.xsd" />
<xsd:include schemaLocation="get-auditory-apparatus-information.xsd" />
<xsd:include schemaLocation="get-button-information.xsd" />
<xsd:include schemaLocation="get-display.xsd" />
<xsd:include schemaLocation="get-hookswitch-status.xsd" />
<xsd:include schemaLocation="get-lamp-information.xsd" />
<xsd:include schemaLocation="get-lamp-mode.xsd" />
<xsd:include schemaLocation="get-message-waiting-indicator.xsd" />
<xsd:include schemaLocation="get-microphone-gain.xsd" />
<xsd:include schemaLocation="get-microphone-mute.xsd" />
<xsd:include schemaLocation="get-ringer-status.xsd" />
<xsd:include schemaLocation="get-speaker-mute.xsd" />
<xsd:include schemaLocation="get-speaker-volume.xsd" />
<xsd:include schemaLocation="set-button-information.xsd" />
<xsd:include schemaLocation="set-display.xsd" />
<xsd:include schemaLocation="set-hookswitch-status.xsd" />
<xsd:include schemaLocation="set-lamp-mode.xsd" />
<xsd:include schemaLocation="set-message-waiting-indicator.xsd" />
<xsd:include schemaLocation="set-microphone-gain.xsd" />
<xsd:include schemaLocation="set-microphone-mute.xsd" />
<xsd:include schemaLocation="set-ringer-status.xsd" />
<xsd:include schemaLocation="set-speaker-mute.xsd" />
<xsd:include schemaLocation="set-speaker-volume.xsd" />
<xsd:annotation>
  <xsd:documentation>Physical Device Events</xsd:documentation>
</xsd:annotation>
<xsd:include schemaLocation="button-information-event.xsd" />
<xsd:include schemaLocation="button-press-event.xsd" />
<xsd:include schemaLocation="display-updated-event.xsd" />
<xsd:include schemaLocation="hookswitch-event.xsd" />
<xsd:include schemaLocation="lamp-mode-event.xsd" />
<xsd:include schemaLocation="message-waiting-event.xsd" />
<xsd:include schemaLocation="microphone-gain-event.xsd" />
<xsd:include schemaLocation="microphone-mute-event.xsd" />
<xsd:include schemaLocation="ringer-status-event.xsd" />
<xsd:include schemaLocation="speaker-mute-event.xsd" />
<xsd:include schemaLocation="speaker-volume-event.xsd" />
<xsd:annotation>
  <xsd:documentation>Logical Device Services</xsd:documentation>
</xsd:annotation>
<xsd:include schemaLocation="call-back-non-call-related.xsd" />
<xsd:include schemaLocation="call-back-message-non-call-related.xsd" />
<xsd:include schemaLocation="cancel-call-back.xsd" />
<xsd:include schemaLocation="cancel-call-back-message.xsd" />
<xsd:include schemaLocation="get-agent-state.xsd" />
<xsd:include schemaLocation="get-auto-answer.xsd" />
<xsd:include schemaLocation="get-auto-work-mode.xsd" />
<xsd:include schemaLocation="get-caller-id-status.xsd" />
<xsd:include schemaLocation="get-do-not-disturb.xsd" />
<xsd:include schemaLocation="get-forwarding.xsd" />
<xsd:include schemaLocation="get-last-number-dialed.xsd" />
<xsd:include schemaLocation="get-routeing-mode.xsd" />
<xsd:include schemaLocation="set-agent-state.xsd" />
<xsd:include schemaLocation="set-auto-answer.xsd" />
<xsd:include schemaLocation="set-auto-work-mode.xsd" />
<xsd:include schemaLocation="set-caller-id-status.xsd" />
<xsd:include schemaLocation="set-do-not-disturb.xsd" />
<xsd:include schemaLocation="set-forwarding.xsd" />
<xsd:include schemaLocation="set-routeing-mode.xsd" />
<xsd:annotation>
  <xsd:documentation>Logical Device Events</xsd:documentation>
</xsd:annotation>
<xsd:include schemaLocation="agent-busy-event.xsd" />
<xsd:include schemaLocation="agent-logged-off-event.xsd" />
<xsd:include schemaLocation="agent-logged-on-event.xsd" />
<xsd:include schemaLocation="agent-not-ready-event.xsd" />
```

```
<xsd:include schemaLocation="agent-ready-event.xsd"/>
<xsd:include schemaLocation="agent-working-after-call-event.xsd"/>
<xsd:include schemaLocation="auto-answer-event.xsd"/>
<xsd:include schemaLocation="auto-work-mode-event.xsd"/>
<xsd:include schemaLocation="call-back-event.xsd"/>
<xsd:include schemaLocation="call-back-message-event.xsd"/>
<xsd:include schemaLocation="caller-id-status-event.xsd"/>
<xsd:include schemaLocation="do-not-disturb-event.xsd"/>
<xsd:include schemaLocation="forwarding-event.xsd"/>
<xsd:include schemaLocation="routeing-event.xsd"/>
<xsd:annotation>
  <xsd:documentation>Device Maintenance Events</xsd:documentation>
</xsd:annotation>
<xsd:include schemaLocation="back-in-service-event.xsd"/>
<xsd:include schemaLocation="device-capabilities-changed-event.xsd"/>
<xsd:include schemaLocation="out-of-service-event.xsd"/>
<xsd:annotation>
  <xsd:documentation>I/O Services</xsd:documentation>
</xsd:annotation>
<xsd:include schemaLocation="io-register.xsd"/>
<xsd:include schemaLocation="io-register-abort.xsd"/>
<xsd:include schemaLocation="io-register-cancel.xsd"/>
<xsd:include schemaLocation="data-path-resumed.xsd"/>
<xsd:include schemaLocation="data-path-suspended.xsd"/>
<xsd:include schemaLocation="fast-data.xsd"/>
<xsd:include schemaLocation="resume-data-path.xsd"/>
<xsd:include schemaLocation="send-broadcast-data.xsd"/>
<xsd:include schemaLocation="send-data.xsd"/>
<xsd:include schemaLocation="send-multicast-data.xsd"/>
<xsd:include schemaLocation="start-data-path.xsd"/>
<xsd:include schemaLocation="stop-data-path.xsd"/>
<xsd:include schemaLocation="suspend-data-path.xsd"/>
<xsd:annotation>
  <xsd:documentation>Data Collection Services</xsd:documentation>
</xsd:annotation>
<xsd:include schemaLocation="data-collected.xsd"/>
<xsd:include schemaLocation="data-collection-resumed.xsd"/>
<xsd:include schemaLocation="data-collection-suspended.xsd"/>
<xsd:include schemaLocation="resume-data-collection.xsd"/>
<xsd:include schemaLocation="start-data-collection.xsd"/>
<xsd:include schemaLocation="stop-data-collection.xsd"/>
<xsd:include schemaLocation="suspend-data-collection.xsd"/>
<xsd:annotation>
  <xsd:documentation>Voice Unit Services</xsd:documentation>
</xsd:annotation>
<xsd:include schemaLocation="concatenate-message.xsd"/>
<xsd:include schemaLocation="delete-message.xsd"/>
<xsd:include schemaLocation="play-message.xsd"/>
<xsd:include schemaLocation="query-voice-attribute.xsd"/>
<xsd:include schemaLocation="record-message.xsd"/>
<xsd:include schemaLocation="reposition.xsd"/>
<xsd:include schemaLocation="resume.xsd"/>
<xsd:include schemaLocation="review.xsd"/>
<xsd:include schemaLocation="set-voice-attribute.xsd"/>
<xsd:include schemaLocation="stop.xsd"/>
<xsd:include schemaLocation="synthesize-message.xsd"/>
<xsd:annotation>
  <xsd:documentation>Voice Unit Events</xsd:documentation>
</xsd:annotation>
<xsd:include schemaLocation="play-event.xsd"/>
<xsd:include schemaLocation="record-event.xsd"/>
<xsd:include schemaLocation="review-event.xsd"/>
<xsd:include schemaLocation="stop-event.xsd"/>
<xsd:include schemaLocation="suspend-play-event.xsd"/>
<xsd:include schemaLocation="suspend-record-event.xsd"/>
<xsd:include schemaLocation="voice-attribute-changed-event.xsd"/>
<xsd:annotation>
  <xsd:documentation>Call Detail Record Services</xsd:documentation>
```

```
</xsd:annotation>
<xsd:include schemaLocation="call-detail-records-notification.xsd"/>
<xsd:include schemaLocation="call-detail-records-report.xsd"/>
<xsd:include schemaLocation="send-stored-call-detail-records.xsd"/>
<xsd:include schemaLocation="start-call-detail-records-transmission.xsd"/>
<xsd:include schemaLocation="stop-call-detail-records-transmission.xsd"/>
<xsd:annotation>
  <xsd:documentation>Vendor Specific Extensions Services</xsd:documentation>
</xsd:annotation>
<xsd:include schemaLocation="escape-register.xsd"/>
<xsd:include schemaLocation="escape-register-abort.xsd"/>
<xsd:include schemaLocation="escape-register-cancel.xsd"/>
<xsd:include schemaLocation="escape.xsd"/>
<xsd:include schemaLocation="private-data-version-selection.xsd"/>
<xsd:annotation>
  <xsd:documentation>Vendor Specific Extensions Events</xsd:documentation>
</xsd:annotation>
<xsd:include schemaLocation="private-event.xsd"/>

</xsd:schema>
```

11 Capability exchange services

11.1 Services

11.1.1 Get logical device information

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

```
<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"  
xmlns:xsd="http://www.w3.org/2001/XMLSchema"  
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"  
elementFormDefault="qualified" attributeFormDefault="unqualified">
```

```
  <xsd:annotation>  
    <xsd:documentation>CSTA-get-logical-device-information</xsd:documentation>  
  </xsd:annotation>
```

```
  <xsd:include schemaLocation="device-identifiers.xsd"/>  
  <xsd:include schemaLocation="status-reporting.xsd"/>  
  <xsd:include schemaLocation="extension-types.xsd"/>  
  <xsd:include schemaLocation="capability-exchange.xsd"/>  
  <xsd:include schemaLocation="data-call-types.xsd"/>
```

```
  <xsd:element name="GetLogicalDeviceInformation">  
    <xsd:complexType>  
      <xsd:sequence>  
        <xsd:element name="device" type="csta:DeviceID"/>  
        <xsd:element ref="csta:extensions" minOccurs="0"/>  
      </xsd:sequence>  
    </xsd:complexType>  
  </xsd:element>
```

```
  <xsd:element name="GetLogicalDeviceInformationResponse">  
    <xsd:complexType>  
      <xsd:sequence>  
        <xsd:element name="deviceCategory" type="csta:DeviceCategory" default="station"  
minOccurs="0"/>  
        <xsd:element name="groupDeviceAttributes" type="csta:GroupDeviceAttributes" minOccurs="0"/>  
        <xsd:element name="namedDeviceTypes" type="csta:NamedDeviceTypes" minOccurs="0"/>  
        <xsd:element name="shortFormDeviceID" type="csta:DeviceID" minOccurs="0"/>  
        <xsd:element name="hasPhysicalElement" type="xsd:boolean"/>  
        <xsd:element name="acdModels" type="csta:ACDModels"/>  
        <xsd:element name="agentLogOnModels" type="csta:AgentLogOnModels" minOccurs="0"/>  
        <xsd:element name="appearanceAddressable" type="xsd:boolean"/>  
        <xsd:element name="appearanceType" type="csta:AppearanceType"/>  
        <xsd:element name="appearanceList" minOccurs="0">  
          <xsd:complexType>  
            <xsd:sequence>  
              <xsd:element name="appearance" type="xsd:string" minOccurs="0" maxOccurs="unbounded"/>  
            </xsd:sequence>  
          </xsd:complexType>  
        </xsd:element>  
        <xsd:element name="otherPhysicalDeviceList" minOccurs="0">  
          <xsd:complexType>  
            <xsd:sequence>  
              <xsd:element name="deviceID" type="csta:DeviceID" minOccurs="0"  
maxOccurs="unbounded"/>  
            </xsd:sequence>  
          </xsd:complexType>  
        </xsd:element>  
        <xsd:element name="miscMonitorCaps" type="csta:MiscMonitorCaps" minOccurs="0"/>  
        <xsd:element name="associatedGroupList" minOccurs="0">  
          <xsd:complexType>  
            <xsd:sequence>  
              <xsd:element name="deviceID" type="csta:DeviceID" minOccurs="0"  
maxOccurs="unbounded"/>  
            </xsd:sequence>  
          </xsd:complexType>  
        </xsd:element>
```

```
<xsd:element name="maxCallbacks" type="xsd:long" minOccurs="0"/>
<xsd:element name="maxAutoAnswerRings" type="xsd:long" minOccurs="0"/>
<xsd:element name="maxActiveCalls" type="xsd:long" minOccurs="0"/>
<xsd:element name="maxHeldCalls" type="xsd:long" minOccurs="0"/>
<xsd:element name="maxFwdSettings" type="xsd:long" minOccurs="0"/>
<xsd:element name="maxDevicesInConf" type="xsd:long" minOccurs="0"/>
<xsd:element name="transAndConfSetup" type="csta:TransAndConfSetup" minOccurs="0"/>
<xsd:element name="deviceOnDeviceMonitorFilter" type="csta:MonitorFilter" minOccurs="0"/>
<xsd:element name="deviceOnConnectionMonitorFilter" type="csta:MonitorFilter"
minOccurs="0"/>
<xsd:element name="callOnDeviceMonitorFilter" type="csta:MonitorFilter" minOccurs="0"/>
<xsd:element name="callOnConnectionMonitorFilter" type="csta:MonitorFilter" minOccurs="0"/>
<xsd:element name="mediaClassSupport" type="csta:MonitorMediaClass" minOccurs="0"/>
<xsd:element name="mediaServiceCapsList" type="csta:MediaServiceCapsList" minOccurs="0"/>
<xsd:element name="connectionRateList" type="csta:ConnectionRateList" minOccurs="0"/>
<xsd:element name="delayToleranceList" type="csta:DelayToleranceList" minOccurs="0"/>
<xsd:element name="numberOfChannels" type="csta:NumberOfChannels" minOccurs="0"/>
<xsd:element name="maxChannelBind" type="csta:MaxChannelBind" minOccurs="0"/>
<xsd:element name="routeingServList" type="csta:RouteingServList" minOccurs="0"/>
<xsd:element name="logDevServList" type="csta:LogDevServList" minOccurs="0"/>
<xsd:element name="logDevEvtsList" type="csta:LogDevEvtsList" minOccurs="0"/>
<xsd:element name="deviceMaintEvtsList" type="csta:DeviceMaintEvtsList" minOccurs="0"/>
<xsd:element ref="csta:extensions" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>

</xsd:schema>
```

11.1.2 Get physical device information

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-get-physical-device-information</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-identifiers.xsd"/>
  <xsd:include schemaLocation="status-reporting.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>
  <xsd:include schemaLocation="capability-exchange.xsd"/>

  <xsd:element name="GetPhysicalDeviceInformation">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="device" type="csta:DeviceID"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="GetPhysicalDeviceInformationResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="deviceCategory" type="csta:DeviceCategory" default="station"/>
        <xsd:element name="groupDeviceAttributes" type="csta:GroupDeviceAttributes" minOccurs="0"/>
        <xsd:element name="namedDeviceTypes" type="csta:NamedDeviceTypes" minOccurs="0"/>
        <xsd:element name="hasLogicalElement" type="xsd:boolean"/>
        <xsd:element name="otherLogicalDeviceList" minOccurs="0">
          <xsd:complexType>
            <xsd:sequence>
              <xsd:element name="deviceID" type="csta:DeviceID" minOccurs="0"
maxOccurs="unbounded"/>
            </xsd:sequence>
          </xsd:complexType>
        </xsd:element>
        <xsd:element name="deviceModelName" minOccurs="0">
          <xsd:simpleType>
            <xsd:restriction base="xsd:string">
              <xsd:minLength value="0"/>
              <xsd:maxLength value="64"/>
            </xsd:restriction>
          </xsd:simpleType>
        </xsd:element>
        <xsd:element name="deviceOnDeviceMonitorFilter" type="csta:MonitorFilter" minOccurs="0"/>
        <xsd:element name="deviceOnConnectionMonitorFilter" type="csta:MonitorFilter"
minOccurs="0"/>
        <xsd:element name="callOnDeviceMonitorFilter" type="csta:MonitorFilter" minOccurs="0"/>
        <xsd:element name="callOnConnectionMonitorFilter" type="csta:MonitorFilter" minOccurs="0"/>
        <xsd:element name="maxDisplays" type="xsd:long" minOccurs="0"/>
        <xsd:element name="maxButtons" type="xsd:long" minOccurs="0"/>
        <xsd:element name="maxLamps" type="xsd:long" minOccurs="0"/>
        <xsd:element name="maxRingPatterns" type="xsd:long" minOccurs="0"/>
        <xsd:element name="physDevServList" type="csta:PhysDevServList" minOccurs="0"/>
        <xsd:element name="physDevEvtsList" type="csta:PhysDevEvtsList" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

11.1.3 Get switching function capabilities

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-get-switching-function-capabilities</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="status-reporting.xsd"/>
  <xsd:include schemaLocation="security.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>
  <xsd:include schemaLocation="call-control.xsd"/>
  <xsd:include schemaLocation="capability-exchange.xsd"/>
  <xsd:include schemaLocation="data-call-types.xsd"/>

  <xsd:element name="GetSwitchingFunctionCaps">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="GetSwitchingFunctionCapsResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="switchingSubDomainName">
          <xsd:simpleType>
            <xsd:restriction base="xsd:string">
              <xsd:minLength value="0"/>
              <xsd:maxLength value="64"/>
            </xsd:restriction>
          </xsd:simpleType>
        </xsd:element>
        <xsd:element name="manufacturerName">
          <xsd:simpleType>
            <xsd:restriction base="xsd:string">
              <xsd:minLength value="0"/>
              <xsd:maxLength value="64"/>
            </xsd:restriction>
          </xsd:simpleType>
        </xsd:element>
        <xsd:element name="profiles" type="csta:Profiles"/>
        <xsd:element name="deviceIDFormat" type="csta:DeviceIDFormat"/>
        <xsd:element name="swDomainFeatures" type="csta:SwDomainFeatures"/>
        <xsd:element name="swAppearanceAddressability" type="csta:SwAppearanceAddressability"/>
        <xsd:element name="swAppearanceTypes" type="csta:SwAppearanceTypes"/>
        <xsd:element name="ignoreUnsupportedParameters" type="csta:IgnoreUnsupportedParameters"/>
        <xsd:element name="callCharacteristicsSupported" type="csta:CallCharacteristics"
minOccurs="0"/>
        <xsd:element name="mediaClassSupport" type="csta:MonitorMediaClass" minOccurs="0"/>
        <xsd:element name="numberOfChannels" type="csta:NumberOfChannels" minOccurs="0"/>
        <xsd:element name="maxChannelBind" type="csta:MaxChannelBind" minOccurs="0"/>
        <xsd:element name="miscMediaCallCharacteristics" type="csta:MiscMediaCallCharacteristics"
minOccurs="0"/>
        <xsd:element name="connectionRateList" type="csta:ConnectionRateList" minOccurs="0"/>
        <xsd:element name="delayToleranceList" type="csta:DelayToleranceList" minOccurs="0"/>
        <xsd:element name="pauseTime" type="csta:PauseTime" minOccurs="0"/>
        <xsd:element name="currentTime" type="csta:TimeInfo" minOccurs="0"/>
        <xsd:element name="messageSeqNumbers" type="csta:MessageSeqNumbers" minOccurs="0"/>
        <xsd:element name="timeStampMode" type="csta:TimeStampMode" minOccurs="0"/>
        <xsd:element name="securityMode" type="csta:SecurityMode" minOccurs="0"/>
        <xsd:element name="securityFormat" type="csta:SecurityFormat" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>
</xsd:schema>
```



```
<xsd:element name="privateDataFormat" type="csta:SecurityFormat" minOccurs="0"/>
<xsd:element name="transAndConfSetup" type="csta:TransAndConfSetup" minOccurs="0"/>
<xsd:element name="monitorFilterItems" type="csta:MonitorFilterItems" minOccurs="0"/>
<xsd:element name="miscMonitorCaps" type="csta:MiscMonitorCaps" minOccurs="0"/>
<xsd:element name="correlatorDataSupported" type="xsd:boolean" minOccurs="0"/>
<xsd:element name="dynamicFeatureSupported" type="csta:DynamicFeatureSupported"
minOccurs="0"/>
<xsd:element name="callLinkageOptions" type="csta:CallLinkageOptions" minOccurs="0"/>
<xsd:element name="aCDModels" type="csta:ACDModels" minOccurs="0"/>
<xsd:element name="agentLogOnModels" type="csta:AgentLogOnModels" minOccurs="0"/>
<xsd:element name="agentStateModels" type="csta:AgentStateModels" minOccurs="0"/>
<xsd:element name="connectionView" type="csta:ConnectionView"/>
<xsd:element name="maxLengthParameters" type="csta:MaxLengthParameters"/>
<xsd:element name="servEvsList" type="csta:ServEvsList" minOccurs="0"/>
<xsd:element name="privateDataVersionList" type="csta:PrivateDataVersionList"
minOccurs="0"/>
<xsd:element name="systemStatusTimer" minOccurs="0">
  <xsd:simpleType>
    <xsd:restriction base="xsd:long">
      <xsd:minInclusive value="0"/>
      <xsd:maxInclusive value="180"/>
    </xsd:restriction>
  </xsd:simpleType>
</xsd:element>
<xsd:element name="simpleThreshold" type="xsd:long" minOccurs="0"/>
<xsd:element name="filterThreshold" type="csta:FilterThreshold" minOccurs="0"/>
<xsd:element name="mediaServiceCapsList" type="csta:MediaServiceCapsList" minOccurs="0"/>
<xsd:element name="extensions" type="csta:CSTACCommonArguments" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:complexType name="MonitorFilterItems">
  <xsd:sequence>
    <xsd:element name="deviceOnDeviceMonitorFilter" type="csta:MonitorFilter" minOccurs="0"/>
    <xsd:element name="deviceOnConnectionMonitorFilter" type="csta:MonitorFilter"
minOccurs="0"/>
    <xsd:element name="callOnDeviceMonitorFilter" type="csta:MonitorFilter" minOccurs="0"/>
    <xsd:element name="callOnConnectionMonitorFilter" type="csta:MonitorFilter" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="MessageSeqNumbers">
  <xsd:sequence>
    <xsd:element name="allEvents" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="allAcks" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="allServReqs" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:simpleType name="SecurityMode">
  <xsd:restriction base="xsd:string">
    <xsd:enumeration value="allEvents"/>
    <xsd:enumeration value="allAcks"/>
    <xsd:enumeration value="allServReqs"/>
  </xsd:restriction>
</xsd:simpleType>
<xsd:complexType name="SecurityFormat">
  <xsd:sequence>
    <xsd:element name="octetStringFromSF" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="otherTypeFromSF" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="octetStringToSF" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="otherTypeToSF" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="Profiles">
  <xsd:sequence>
    <xsd:element name="basicTelephonyProfile" type="xsd:boolean" minOccurs="0"/>
    <xsd:element name="routingProfile" type="xsd:boolean" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

```
<xsd:simpleType name="DynamicFeatureSupported">
  <xsd:restriction base="xsd:string">
    <xsd:enumeration value="none" />
    <xsd:enumeration value="all" />
    <xsd:enumeration value="some" />
  </xsd:restriction>
</xsd:simpleType>
<xsd:complexType name="CallLinkageOptions">
  <xsd:sequence>
    <xsd:element name="callLinkageFeatureSupported" type="xsd:boolean" minOccurs="0" />
    <xsd:element name="threadLinkageFeatureSupported" type="xsd:boolean" minOccurs="0" />
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="AgentStateModels">
  <xsd:sequence>
    <xsd:element name="multiState" type="xsd:boolean" minOccurs="0" />
    <xsd:element name="multiStateSemiIndependentLinked" type="xsd:boolean" minOccurs="0" />
    <xsd:element name="agentOriented" type="xsd:boolean" minOccurs="0" />
  </xsd:sequence>
</xsd:complexType>
<xsd:simpleType name="ConnectionView">
  <xsd:restriction base="xsd:string">
    <xsd:enumeration value="fixed" />
    <xsd:enumeration value="local" />
  </xsd:restriction>
</xsd:simpleType>
<xsd:complexType name="ServEvtsList">
  <xsd:sequence>
    <xsd:element name="capExchangeServList" type="csta:CapExchangeServList" minOccurs="0" />
    <xsd:element name="systemServList" type="csta:SystemStatusServList" minOccurs="0" />
    <xsd:element name="monitoringServList" type="csta:MonitoringServList" minOccurs="0" />
    <xsd:element name="snapshotServList" type="csta:SnapshotServList" minOccurs="0" />
    <xsd:element name="callControlServList" type="csta:CallControlServList" minOccurs="0" />
    <xsd:element name="callControlEvtsList" type="csta:CallControlEvtsList" minOccurs="0" />
    <xsd:element name="callAssociatedServList" type="csta:CallAssociatedServList"
minOccurs="0" />
    <xsd:element name="callAssociatedEvtsList" type="csta:CallAssociatedEvtsList"
minOccurs="0" />
    <xsd:element name="mediaServList" type="csta:MediaServList" minOccurs="0" />
    <xsd:element name="mediaEvtsList" type="csta:MediaEvtsList" minOccurs="0" />
    <xsd:element name="routeingServList" type="csta:RouteingServList" minOccurs="0" />
    <xsd:element name="physDevServList" type="csta:PhysDevServList" minOccurs="0" />
    <xsd:element name="physDevEvtsList" type="csta:PhysDevEvtsList" minOccurs="0" />
    <xsd:element name="logicalServList" type="csta:LogicalServList" minOccurs="0" />
    <xsd:element name="logicalEvtsList" type="csta:LogicalEvtsList" minOccurs="0" />
    <xsd:element name="deviceMaintEvtsList" type="csta:DeviceMaintEvtsList" minOccurs="0" />
    <xsd:element name="iOServicesServList" type="csta:IOServicesServList" minOccurs="0" />
    <xsd:element name="dataCollectionServList" type="csta:DataCollectionServList"
minOccurs="0" />
    <xsd:element name="voiceUnitServList" type="csta:VoiceUnitServList" minOccurs="0" />
    <xsd:element name="voiceUnitEvtsList" type="csta:VoiceUnitEvtsList" minOccurs="0" />
    <xsd:element name="cdrServList" type="csta:CDRServList" minOccurs="0" />
    <xsd:element name="vendorSpecificServList" type="csta:VendorSpecificServList"
minOccurs="0" />
    <xsd:element name="vendorSpecificEvtsList" type="csta:VendorSpecificEvtsList"
minOccurs="0" />
    <xsd:element name="statusReportingServList" type="csta:StatusReportingServList"
minOccurs="0" />
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="StatusReportingServList">
  <xsd:sequence>
    <xsd:element name="monitoringServices" type="csta:MonitoringServList" minOccurs="0" />
    <xsd:element name="snapshotServices" type="csta:SnapshotServList" minOccurs="0" />
    <xsd:element name="systemServices" type="csta:SystemStatusServList" minOccurs="0" />
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="MiscMediaCallCharacteristics">
```

```
<xsd:sequence>
  <xsd:element name="supportAdjustment" type="xsd:boolean" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
<xsd:complexType name="PrivateDataVersionList">
  <xsd:sequence>
    <xsd:element name="privateDataVersion" type="xsd:long" minOccurs="0" maxOccurs="unbounded"/>
  </xsd:sequence>
</xsd:complexType>
</xsd:schema>
```

11.1.4 Get switching function devices

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-get-switching-function-devices</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-identifiers.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>
  <xsd:include schemaLocation="capability-exchange.xsd"/>

  <xsd:element name="GetSwitchingFunctionDevices">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="requestedDeviceID" type="csta:DeviceID" minOccurs="0"/>
        <xsd:element name="requestedDeviceCategory" type="csta:ReqDeviceCategory" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="GetSwitchingFunctionDevicesResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="serviceCrossRefID" type="csta:ServiceCrossRefID"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:simpleType name="ReqDeviceCategory">
    <xsd:restriction base="xsd:string">
      <xsd:enumeration value="acd"/>
      <xsd:enumeration value="groupACD"/>
      <xsd:enumeration value="groupHunt"/>
      <xsd:enumeration value="groupPick"/>
      <xsd:enumeration value="groupOther"/>
      <xsd:enumeration value="networkInterface"/>
      <xsd:enumeration value="park"/>
      <xsd:enumeration value="routeingDevice"/>
      <xsd:enumeration value="station"/>
      <xsd:enumeration value="voiceUnit"/>
      <xsd:enumeration value="other"/>
    </xsd:restriction>
  </xsd:simpleType>

</xsd:schema>
```

11.1.5 Switching function devices

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-switching-function-devices</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-identifiers.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>
  <xsd:include schemaLocation="capability-exchange.xsd"/>

  <xsd:element name="SwitchingFunctionDevices">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="serviceCrossRefID" type="csta:ServiceCrossRefID"/>
        <xsd:element name="segmentID" type="xsd:long" minOccurs="0"/>
        <xsd:element name="lastSegment" type="xsd:boolean"/>
        <xsd:element name="deviceList" type="csta:DeviceList"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:complexType name="DeviceList">
    <xsd:sequence>
      <xsd:element name="device" minOccurs="0" maxOccurs="unbounded">
        <xsd:complexType>
          <xsd:sequence>
            <xsd:element name="deviceID" type="csta:DeviceID"/>
            <xsd:element name="deviceCategory" type="csta:DeviceCategory" default="station"
minOccurs="0"/>
            <xsd:element name="namedDeviceTypes" type="csta:NamedDeviceTypes" minOccurs="0"/>
            <xsd:element name="deviceAttributes" type="csta:DeviceAttributes" minOccurs="0"/>
            <xsd:element name="deviceModelName" minOccurs="0">
              <xsd:simpleType>
                <xsd:restriction base="xsd:string">
                  <xsd:minLength value="0"/>
                  <xsd:maxLength value="64"/>
                </xsd:restriction>
              </xsd:simpleType>
            </xsd:element>
          </xsd:sequence>
        </xsd:complexType>
      </xsd:element>
    </xsd:sequence>
  </xsd:complexType>

  <xsd:complexType name="DeviceAttributes">
    <xsd:sequence>
      <xsd:element name="mediaAccessDevice" type="xsd:boolean" minOccurs="0"/>
      <xsd:element name="routingDevice" type="xsd:boolean" minOccurs="0"/>
      <xsd:element name="groupACD" type="xsd:boolean" minOccurs="0"/>
      <xsd:element name="groupHunt" type="xsd:boolean" minOccurs="0"/>
      <xsd:element name="groupPick" type="xsd:boolean" minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>

</xsd:schema>
```

12 System services

12.1 Registration services

12.1.1 Change system status filter

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-change-system-status-filter</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-feature-types.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>

  <xsd:element name="ChangeSysStatFilter">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:sysStatRegisterID"/>
        <xsd:element name="requestedStatusFilter" type="csta:SystemStatus"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="ChangeSysStatFilterResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="actualStatusFilter" type="csta:SystemStatus"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

12.1.2 System register

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta" x
mlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-system-register</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-feature-types.xsd" />
  <xsd:include schemaLocation="extension-types.xsd" />

  <xsd:element name="SystemRegister">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="requestTypes" type="csta:RequestTypes" />
        <xsd:element name="requestedStatusFilter" type="csta:StatusFilter" minOccurs="0" />
        <xsd:element ref="csta:extensions" minOccurs="0" />
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="SystemRegisterResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:sysStatRegisterID" />
        <xsd:element name="actualStatusFilter" type="csta:StatusFilter" minOccurs="0" />
        <xsd:element ref="csta:extensions" minOccurs="0" />
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:complexType name="StatusFilter">
    <xsd:sequence>
      <xsd:element name="initializing" type="xsd:boolean" minOccurs="0" />
      <xsd:element name="enabled" type="xsd:boolean" minOccurs="0" />
      <xsd:element name="normal" type="xsd:boolean" minOccurs="0" />
      <xsd:element name="messageLost" type="xsd:boolean" minOccurs="0" />
      <xsd:element name="disabled" type="xsd:boolean" minOccurs="0" />
      <xsd:element name="partiallyDisabled" type="xsd:boolean" minOccurs="0" />
      <xsd:element name="overloadImminent" type="xsd:boolean" minOccurs="0" />
      <xsd:element name="overloadReached" type="xsd:boolean" minOccurs="0" />
      <xsd:element name="overloadRelieved" type="xsd:boolean" minOccurs="0" />
    </xsd:sequence>
  </xsd:complexType>

  <xsd:complexType name="RequestTypes">
    <xsd:sequence>
      <xsd:element name="systemStatus" type="xsd:boolean" minOccurs="0" />
      <xsd:element name="requestSystemStatus" type="xsd:boolean" minOccurs="0" />
      <xsd:element name="switchingFunctionCapsChanged" type="xsd:boolean" minOccurs="0" />
      <xsd:element name="switchingFunctionDevicesChanged" type="xsd:boolean" minOccurs="0" />
    </xsd:sequence>
  </xsd:complexType>

</xsd:schema>
```

12.1.3 System register abort

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-system-register-abort</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-feature-types.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>

  <xsd:element name="SystemRegisterAbort">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:sysStatRegisterID"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:annotation>
    <xsd:documentation>There is no positive response to this request</xsd:documentation>
  </xsd:annotation>

</xsd:schema>
```


12.1.4 System register cancel

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-system-register-cancel</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-feature-types.xsd" />
  <xsd:include schemaLocation="extension-types.xsd" />

  <xsd:element name="SystemRegisterCancel">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:sysStatRegisterID" />
        <xsd:element ref="csta:extensions" minOccurs="0" />
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="SystemRegisterCancelResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:extensions" minOccurs="0" />
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

12.2 Services

12.2.1 Request system status

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-request-system-status</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-feature-types.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>

  <xsd:element name="RequestSystemStatus">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:sysStatRegisterID" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="RequestSystemStatusResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="systemStatus" type="csta:SystemStatus"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

12.2.2 System status

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-system-status1</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-feature-types.xsd" />
  <xsd:include schemaLocation="extension-types.xsd" />

  <xsd:element name="SystemStatus">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:sysStatRegisterID" minOccurs="0"/>
        <xsd:element name="systemStatus" type="csta:SystemStatus"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="SystemStatusResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

12.2.3 Switching function capabilities changed

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-switching-functions-capabilities-changed</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-feature-types.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>

  <xsd:element name="SwFunctionCapsChanged">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:sysStatRegisterID" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="SwFunctionCapsChangedResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

12.2.4 Switching function devices changed

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-switching-functions-devices-changed</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-feature-types.xsd" />
  <xsd:include schemaLocation="extension-types.xsd" />

  <xsd:element name="SwFunctionDevicesChanged">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:sysStatRegisterID" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="SwFunctionDevicesChangedResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

13 Monitoring services

13.1 Services

13.1.1 Change monitor filter

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-change-monitor-filter</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="status-reporting.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>

  <xsd:element name="ChangeMonitorFilter">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:monitorCrossRefID"/>
        <xsd:element name="requestedFilterList" type="csta:MonitorFilter"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="ChangeMonitorFilterResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="actualFilterList" type="csta:MonitorFilter" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

13.1.2 Monitor start

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-monitor-start</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="status-reporting.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>

  <xsd:element name="MonitorStart">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="monitorObject" type="csta:MonitorObject"/>
        <xsd:element name="requestedMonitorFilter" type="csta:MonitorFilter" minOccurs="0"/>
        <xsd:element name="monitorType" type="csta:MonitorType" minOccurs="0"/>
        <xsd:element name="requestedMonitorMediaClass" type="csta:MonitorMediaClass"
minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="MonitorStartResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:monitorCrossRefID"/>
        <xsd:element name="actualMonitorFilter" type="csta:MonitorFilter" minOccurs="0"/>
        <xsd:element name="actualMonitorMediaClass" type="csta:MonitorMediaClass" minOccurs="0"/>
        <xsd:element name="monitorExistingCalls" type="xsd:boolean" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

13.1.3 Monitor stop

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-monitor-stop</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="status-reporting.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>

  <xsd:element name="MonitorStop">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:monitorCrossRefID"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="MonitorStopResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```


14 Snapshot services

14.1 Services

14.1.1 Snapshot call

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-snapshot-call</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-identifiers.xsd"/>
  <xsd:include schemaLocation="call-connection-identifiers.xsd"/>
  <xsd:include schemaLocation="status-reporting.xsd"/>
  <xsd:include schemaLocation="device-feature-types.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>
  <xsd:include schemaLocation="call-control.xsd"/>
  <xsd:include schemaLocation="capability-exchange.xsd"/>
  <xsd:include schemaLocation="media-services.xsd"/>

  <xsd:element name="SnapshotCall">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="snapshotObject" type="csta:ConnectionID"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="SnapshotCallResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="crossRefIDorSnapshotData">
          <xsd:complexType>
            <xsd:choice>
              <xsd:element name="serviceCrossRefID" type="csta:ServiceCrossRefID"/>
              <xsd:element name="snapshotData" type="csta:SnapshotCallData"/>
            </xsd:choice>
          </xsd:complexType>
        </xsd:element>
        <xsd:element ref="csta:mediaCallCharacteristics" minOccurs="0"/>
        <xsd:element ref="csta:callCharacteristics" minOccurs="0"/>
        <xsd:element ref="csta:callingDevice" minOccurs="0"/>
        <xsd:element ref="csta:calledDevice" minOccurs="0"/>
        <xsd:element ref="csta:associatedCallingDevice" minOccurs="0"/>
        <xsd:element ref="csta:associatedCalledDevice" minOccurs="0"/>
        <xsd:element ref="csta:correlatorData" minOccurs="0"/>
        <xsd:element ref="csta:callLinkageData" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

14.1.2 Snapshot device

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-snapshot-device</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-identifiers.xsd"/>
  <xsd:include schemaLocation="status-reporting.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>
  <xsd:include schemaLocation="capability-exchange.xsd"/>

  <xsd:element name="SnapshotDevice">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="snapshotObject" type="csta:DeviceID"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="SnapshotDeviceResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="crossRefIDorSnapshotData">
          <xsd:complexType>
            <xsd:choice>
              <xsd:element name="serviceCrossRefID" type="csta:ServiceCrossRefID"/>
              <xsd:element name="snapshotData" type="csta:SnapshotDeviceData"/>
            </xsd:choice>
          </xsd:complexType>
        </xsd:element>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

14.1.3 Snapshot calldata

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-snapshot-call-data</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="status-reporting.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>
  <xsd:include schemaLocation="capability-exchange.xsd"/>

  <xsd:element name="SnapshotCallData">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="serviceCrossRefID" type="csta:ServiceCrossRefID"/>
        <xsd:element name="segmentID" type="xsd:long" minOccurs="0"/>
        <xsd:element name="lastSegment" type="xsd:boolean"/>
        <xsd:element name="snapshotData" type="csta:SnapshotCallData"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:annotation>
    <xsd:documentation>There is no positive response to this request.</xsd:documentation>
  </xsd:annotation>

</xsd:schema>
```

14.1.4 Snapshot devicedata

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-snapshot-device-data</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="status-reporting.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>
  <xsd:include schemaLocation="capability-exchange.xsd"/>

  <xsd:element name="SnapshotDeviceData">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="serviceCrossRefID" type="csta:ServiceCrossRefID"/>
        <xsd:element name="segmentID" type="xsd:long" minOccurs="0"/>
        <xsd:element name="lastSegment" type="xsd:boolean"/>
        <xsd:element name="snapshotData" type="csta:SnapshotDeviceData"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:annotation>
    <xsd:documentation>There is no positive response to this request.</xsd:documentation>
  </xsd:annotation>

</xsd:schema>
```

15 Call control services and events

15.1 Services

15.1.1 Accept call

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-accept-call</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="call-connection-identifiers.xsd"/>
  <xsd:include schemaLocation="device-feature-types.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>

  <xsd:element name="AcceptCall">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="callToBeAccepted" type="csta:ConnectionID"/>
        <xsd:element ref="csta:correlatorData" minOccurs="0"/>
        <xsd:element ref="csta:userData" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="AcceptCallResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

15.1.2 Alternate call

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-alternate-call</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="call-connection-identifiers.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>
  <xsd:include schemaLocation="call-control.xsd"/>
  <xsd:element name="AlternateCall">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="heldCall" type="csta:ConnectionID"/>
        <xsd:element name="activeCall" type="csta:ConnectionID"/>
        <xsd:element name="connectionReservation" type="xsd:boolean" minOccurs="0"/>
        <xsd:element ref="csta:consultOptions" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="AlternateCallResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

15.1.3 Answer call

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-answer-call</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="call-connection-identifiers.xsd"/>
  <xsd:include schemaLocation="device-feature-types.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>

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

  <xsd:element name="AnswerCallResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

15.1.4 Call back call-related

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-call-back-call-related</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-identifiers.xsd"/>
  <xsd:include schemaLocation="call-connection-identifiers.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>
  <xsd:include schemaLocation="call-control.xsd"/>

  <xsd:element name="CallBack">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="callbackConnection" type="csta:ConnectionID"/>
        <xsd:element ref="csta:callCharacteristics" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="CallBackResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="targetDevice" type="csta:DeviceID" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```


15.1.5 Call back message call-related

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-call-back-message-call-related</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-identifiers.xsd"/>
  <xsd:include schemaLocation="call-connection-identifiers.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>

  <xsd:element name="CallBackMessage">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="callbackMessageConnection" type="csta:ConnectionID"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="CallBackMessageResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="targetDevice" type="csta:DeviceID" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

15.1.6 Camp on call

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-camp-on-call</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="call-connection-identifiers.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>

  <xsd:element name="CamponCall">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="camponConnection" type="csta:ConnectionID"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="CamponCallResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

15.1.7 Clear call

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-clear-call</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="call-connection-identifiers.xsd"/>
  <xsd:include schemaLocation="device-feature-types.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>
  <xsd:element name="ClearCall">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="callToBeCleared" type="csta:ConnectionID"/>
        <xsd:element ref="csta:userData" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="ClearCallResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

15.1.8 Clear Connection

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-clear-connection</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="call-connection-identifiers.xsd"/>
  <xsd:include schemaLocation="device-feature-types.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>

  <xsd:element name="ClearConnection">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="connectionToBeCleared" type="csta:ConnectionID"/>
        <xsd:element ref="csta:correlatorData" minOccurs="0"/>
        <xsd:element ref="csta:userData" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="ClearConnectionResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

15.1.9 Conference call

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-conference-call</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="call-connection-identifiers.xsd"/>
  <xsd:include schemaLocation="connection-states.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>
  <xsd:include schemaLocation="media-services.xsd"/>

  <xsd:element name="ConferenceCall">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="heldCall" type="csta:ConnectionID"/>
        <xsd:element name="activeCall" type="csta:ConnectionID"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="ConferenceCallResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="conferenceCall" type="csta:ConnectionID"/>
        <xsd:element name="connections" type="csta:ConnectionList" minOccurs="0"/>
        <xsd:element name="conferenceCallInfo" type="csta:ConnectionInformation" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

15.1.10 Consultation call

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-consultation-call</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-identifiers.xsd"/>
  <xsd:include schemaLocation="call-connection-identifiers.xsd"/>
  <xsd:include schemaLocation="device-feature-types.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>
  <xsd:include schemaLocation="call-control.xsd"/>
  <xsd:include schemaLocation="media-services.xsd"/>

  <xsd:element name="ConsultationCall">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="existingCall" type="csta:ConnectionID"/>
        <xsd:element name="consultedDevice" type="csta:DeviceID"/>
        <xsd:element name="connectionReservation" type="xsd:boolean" minOccurs="0"/>
        <xsd:element ref="csta:accountCode" minOccurs="0"/>
        <xsd:element ref="csta:authCode" minOccurs="0"/>
        <xsd:element ref="csta:correlatorData" minOccurs="0"/>
        <xsd:element ref="csta:userData" minOccurs="0"/>
        <xsd:element ref="csta:callCharacteristics" minOccurs="0"/>
        <xsd:element ref="csta:mediaCallCharacteristics" minOccurs="0"/>
        <xsd:element name="callingConnectionInfo" type="csta:ConnectionInformation" minOccurs="0"/>
        <xsd:element ref="csta:consultOptions" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="ConsultationCallResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="initiatedCall" type="csta:ConnectionID"/>
        <xsd:element ref="csta:mediaCallCharacteristics" minOccurs="0"/>
        <xsd:element name="initiatedCallInfo" type="csta:ConnectionInformation" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

15.1.11 Deflect call

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-deflect-call</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-identifiers.xsd"/>
  <xsd:include schemaLocation="call-connection-identifiers.xsd"/>
  <xsd:include schemaLocation="device-feature-types.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>

  <xsd:element name="DeflectCall">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="callToBeDiverted" type="csta:ConnectionID"/>
        <xsd:element name="newDestination" type="csta:DeviceID"/>
        <xsd:element ref="csta:correlatorData" minOccurs="0"/>
        <xsd:element ref="csta:userData" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="DeflectCallResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

15.1.12 Dial digits

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-dial-digits</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-identifiers.xsd"/>
  <xsd:include schemaLocation="call-connection-identifiers.xsd"/>
  <xsd:include schemaLocation="device-feature-types.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>

  <xsd:element name="DialDigits">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="diallingConnection" type="csta:ConnectionID"/>
        <xsd:element name="diallingSequence" type="csta:DeviceID"/>
        <xsd:element ref="csta:correlatorData" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="DialDigitsResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```


15.1.13 Directed pickup call

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-directed-pickup-call</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-identifiers.xsd"/>
  <xsd:include schemaLocation="call-connection-identifiers.xsd"/>
  <xsd:include schemaLocation="device-feature-types.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>
  <xsd:include schemaLocation="media-services.xsd"/>

  <xsd:element name="DirectedPickupCall">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="callToBePickedUp" type="csta:ConnectionID"/>
        <xsd:element name="requestingDevice" type="csta:DeviceID"/>
        <xsd:element ref="csta:correlatorData" minOccurs="0"/>
        <xsd:element ref="csta:userData" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="DirectedPickupCallResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="pickedCall" type="csta:ConnectionID" minOccurs="0"/>
        <xsd:element name="pickedCallInfo" type="csta:ConnectionInformation" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

15.1.14 Group pickup call

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-group-pickup-call</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-identifiers.xsd"/>
  <xsd:include schemaLocation="call-connection-identifiers.xsd"/>
  <xsd:include schemaLocation="device-feature-types.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>
  <xsd:include schemaLocation="media-services.xsd"/>

  <xsd:element name="GroupPickupCall">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="newDestination" type="csta:DeviceID"/>
        <xsd:element name="pickGroup" type="csta:DeviceID"/>
        <xsd:element ref="csta:correlatorData" minOccurs="0"/>
        <xsd:element ref="csta:userData" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="GroupPickupCallResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="pickedCall" type="csta:ConnectionID" minOccurs="0"/>
        <xsd:element name="pickedCallInfo" type="csta:ConnectionInformation" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

15.1.15 Hold call

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-hold-call</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="call-connection-identifiers.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>

  <xsd:element name="HoldCall">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="callToBeHeld" type="csta:ConnectionID"/>
        <xsd:element name="connectionReservation" type="xsd:boolean"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="HoldCallResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

15.1.16 Intrude call

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-intrude-call</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-identifiers.xsd"/>
  <xsd:include schemaLocation="call-connection-identifiers.xsd"/>
  <xsd:include schemaLocation="device-feature-types.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>
  <xsd:include schemaLocation="media-services.xsd"/>

  <xsd:element name="IntrudeCall">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="intrude" type="csta:ConnectionID"/>
        <xsd:element name="participationType" type="csta:ParticipationType" default="active"/>
        <xsd:element ref="csta:userData" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="IntrudeCallResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="conferencedCall" type="csta:ConnectionID" minOccurs="0"/>
        <xsd:element name="conferencedCallInfo" type="csta:ConnectionInformation" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

15.1.17 Join call

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-join-call</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-identifiers.xsd"/>
  <xsd:include schemaLocation="call-connection-identifiers.xsd"/>
  <xsd:include schemaLocation="device-feature-types.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>
  <xsd:include schemaLocation="media-services.xsd"/>
  <xsd:include schemaLocation="call-control.xsd"/>

  <xsd:element name="JoinCall">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="activeCall" type="csta:ConnectionID"/>
        <xsd:element name="joiningDevice" type="csta:DeviceID"/>
        <xsd:element name="autoOriginate" type="csta:AutoOriginate" default="prompt"/>
        <xsd:element name="participationType" type="csta:ParticipationType" default="active"/>
        <xsd:element ref="csta:accountCode" minOccurs="0"/>
        <xsd:element ref="csta:authCode" minOccurs="0"/>
        <xsd:element ref="csta:correlatorData" minOccurs="0"/>
        <xsd:element ref="csta:userData" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="JoinCallResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="conferencedCall" type="csta:ConnectionID"/>
        <xsd:element name="conferencedCallInfo" type="csta:ConnectionInformation" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

15.1.18 Make call

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-make-call</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-identifiers.xsd"/>
  <xsd:include schemaLocation="call-connection-identifiers.xsd"/>
  <xsd:include schemaLocation="device-feature-types.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>
  <xsd:include schemaLocation="call-control.xsd"/>
  <xsd:include schemaLocation="media-services.xsd"/>

  <xsd:element name="MakeCall">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="callingDevice" type="csta:DeviceID"/>
        <xsd:element name="calledDirectoryNumber" type="csta:DeviceID"/>
        <xsd:element ref="csta:accountCode" minOccurs="0"/>
        <xsd:element ref="csta:authCode" minOccurs="0"/>
        <xsd:element name="autoOriginate" type="csta:AutoOriginate" default="prompt"
minOccurs="0"/>
        <xsd:element ref="csta:correlatorData" minOccurs="0"/>
        <xsd:element ref="csta:userData" minOccurs="0"/>
        <xsd:element ref="csta:callCharacteristics" minOccurs="0"/>
        <xsd:element ref="csta:mediaCallCharacteristics" minOccurs="0"/>
        <xsd:element name="callingConnectionInfo" type="csta:ConnectionInformation" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="MakeCallResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="callingDevice" type="csta:ConnectionID"/>
        <xsd:element ref="csta:mediaCallCharacteristics" minOccurs="0"/>
        <xsd:element name="initiatedCallInfo" type="csta:ConnectionInformation" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

15.1.19 Make predictive call

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-make-predictive-call</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-identifiers.xsd"/>
  <xsd:include schemaLocation="call-connection-identifiers.xsd"/>
  <xsd:include schemaLocation="device-feature-types.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>
  <xsd:include schemaLocation="call-control.xsd"/>
  <xsd:include schemaLocation="media-services.xsd"/>

  <xsd:element name="MakePredictiveCall">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="callingDevice" type="csta:DeviceID"/>
        <xsd:element name="calledDirectoryNumber" type="csta:DeviceID"/>
        <xsd:element name="signallingDetection" type="csta:SignallingDetection" minOccurs="0"/>
        <xsd:element name="destinationDetection" type="csta:DestinationDetection" minOccurs="0"/>
        <xsd:element name="defaultAction" type="csta:DetectionAction" minOccurs="0"/>
        <xsd:element ref="csta:accountCode" minOccurs="0"/>
        <xsd:element ref="csta:authCode" minOccurs="0"/>
        <xsd:element name="autoOriginate" type="csta:AutoOriginate" default="prompt"
minOccurs="0"/>
        <xsd:element name="alertTime" type="xsd:long" minOccurs="0"/>
        <xsd:element ref="csta:correlatorData" minOccurs="0"/>
        <xsd:element ref="csta:callCharacteristics" minOccurs="0"/>
        <xsd:element ref="csta:userData" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="MakePredictiveCallResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="initiatedCall" type="csta:ConnectionID"/>
        <xsd:element name="initiatedCallInfo" type="csta:ConnectionInformation" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:complexType name="SignallingDetection">
    <xsd:sequence>
      <xsd:element name="signallingCondition" type="csta:SignallingCondition"/>
      <xsd:element name="signallingConditionsAction" type="csta:SignallingConditionsAction"/>
    </xsd:sequence>
  </xsd:complexType>

  <xsd:simpleType name="SignallingCondition">
    <xsd:restriction base="xsd:string">
      <xsd:enumeration value="callDelivered"/>
      <xsd:enumeration value="callEstablished"/>
    </xsd:restriction>
  </xsd:simpleType>

  <xsd:simpleType name="SignallingConditionsAction">
    <xsd:restriction base="xsd:string">
      <xsd:enumeration value="destinationDetection"/>
    </xsd:restriction>
  </xsd:simpleType>
</xsd:schema>
```

```
        <xsd:enumeration value="remainConnected" />
    </xsd:restriction>
</xsd:simpleType>

<xsd:complexType name="DestinationDetection">
    <xsd:sequence>
        <xsd:element name="detectionItem" type="csta:DetectionItem" maxOccurs="unbounded" />
    </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="DetectionItem">
    <xsd:sequence>
        <xsd:element name="destinationCondition" type="csta:DestinationCondition" />
        <xsd:element name="detectionAction" type="csta:DetectionAction" />
    </xsd:sequence>
</xsd:complexType>

<xsd:simpleType name="DestinationCondition">
    <xsd:restriction base="xsd:string">
        <xsd:enumeration value="humanVoice" />
        <xsd:enumeration value="answeringMachine" />
        <xsd:enumeration value="facsimileMachine" />
    </xsd:restriction>
</xsd:simpleType>

<xsd:simpleType name="DetectionAction">
    <xsd:restriction base="xsd:string">
        <xsd:enumeration value="clearCalledConnection" />
        <xsd:enumeration value="remainConnected" />
    </xsd:restriction>
</xsd:simpleType>

</xsd:schema>
```


15.1.20 Park call

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-park-call</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-identifiers.xsd"/>
  <xsd:include schemaLocation="call-connection-identifiers.xsd"/>
  <xsd:include schemaLocation="device-feature-types.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>

  <xsd:element name="ParkCall">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="parking" type="csta:ConnectionID"/>
        <xsd:element name="parkTo" type="csta:DeviceID"/>
        <xsd:element ref="csta:correlatorData" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="ParkCallResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="parkedTo" type="csta:ConnectionID" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

15.1.21 Reconnect call

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-reconnect-call</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="call-connection-identifiers.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>

  <xsd:element name="ReconnectCall">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="activeCall" type="csta:ConnectionID"/>
        <xsd:element name="heldCall" type="csta:ConnectionID"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="ReconnectCallResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

15.1.22 Retrieve call

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-retrieve-call</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="call-connection-identifiers.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>

  <xsd:element name="RetrieveCall">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="callToBeRetrieved" type="csta:ConnectionID"/>
        <xsd:element ref="csta:extensions"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="RetrieveCallResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

15.1.23 Single step conference call

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-single-step-conference-call</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-identifiers.xsd"/>
  <xsd:include schemaLocation="call-connection-identifiers.xsd"/>
  <xsd:include schemaLocation="device-feature-types.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>
  <xsd:include schemaLocation="media-services.xsd"/>

  <xsd:element name="SingleStepConferenceCall">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="activeCall" type="csta:ConnectionID"/>
        <xsd:element name="deviceToJoin" type="csta:DeviceID"/>
        <xsd:element name="participationType" type="csta:ParticipationType" default="active"/>
        <xsd:element ref="csta:accountCode" minOccurs="0"/>
        <xsd:element ref="csta:authCode" minOccurs="0"/>
        <xsd:element ref="csta:correlatorData" minOccurs="0"/>
        <xsd:element ref="csta:userData" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="SingleStepConferenceCallResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="conferencedCall" type="csta:ConnectionID"/>
        <xsd:element name="conferencedCallInfo" type="csta:ConnectionInformation" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

15.1.24 Single step transfer call

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-single-step-transfer-call</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-identifiers.xsd"/>
  <xsd:include schemaLocation="call-connection-identifiers.xsd"/>
  <xsd:include schemaLocation="connection-states.xsd"/>
  <xsd:include schemaLocation="device-feature-types.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>
  <xsd:include schemaLocation="media-services.xsd"/>

  <xsd:element name="SingleStepTransferCall">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="activeCall" type="csta:ConnectionID"/>
        <xsd:element name="transferredTo" type="csta:DeviceID"/>
        <xsd:element ref="csta:accountCode" minOccurs="0"/>
        <xsd:element ref="csta:authCode" minOccurs="0"/>
        <xsd:element ref="csta:correlatorData" minOccurs="0"/>
        <xsd:element ref="csta:userData" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="SingleStepTransferCallResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="transferredCall" type="csta:ConnectionID"/>
        <xsd:element name="connections" type="csta:ConnectionList" minOccurs="0"/>
        <xsd:element name="transferredCallInfo" type="csta:ConnectionInformation" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

15.1.25 Transfer call

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-transfer-call</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="call-connection-identifiers.xsd"/>
  <xsd:include schemaLocation="connection-states.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>
  <xsd:include schemaLocation="media-services.xsd"/>

  <xsd:element name="TransferCall">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="heldCall" type="csta:ConnectionID"/>
        <xsd:element name="activeCall" type="csta:ConnectionID"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="TransferCallResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="transferredCall" type="csta:ConnectionID"/>
        <xsd:element name="connections" type="csta:ConnectionList" minOccurs="0"/>
        <xsd:element name="transferredCallInfo" type="csta:ConnectionInformation" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

15.2 Events

15.2.1 Bridged

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-bridged-event</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="event-cause.xsd"/>
  <xsd:include schemaLocation="device-identifiers.xsd"/>
  <xsd:include schemaLocation="call-connection-identifiers.xsd"/>
  <xsd:include schemaLocation="connection-states.xsd"/>
  <xsd:include schemaLocation="device-feature-types.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>
  <xsd:include schemaLocation="call-control.xsd"/>
  <xsd:include schemaLocation="media-services.xsd"/>
  <xsd:include schemaLocation="status-reporting.xsd"/>

  <xsd:element name="BridgedEvent">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:monitorCrossRefID"/>
        <xsd:element name="bridgedConnection" type="csta:ConnectionID"/>
        <xsd:element name="bridgedAppearance" type="csta:SubjectDeviceID"/>
        <xsd:element ref="csta:localConnectionInfo" minOccurs="0"/>
        <xsd:element ref="csta:correlatorData" minOccurs="0"/>
        <xsd:element ref="csta:userData" minOccurs="0"/>
        <xsd:element ref="csta:cause"/>
        <xsd:element ref="csta:servicesPermitted" minOccurs="0"/>
        <xsd:element ref="csta:mediaCallCharacteristics" minOccurs="0"/>
        <xsd:element ref="csta:callCharacteristics" minOccurs="0"/>
        <xsd:element name="bridgedConnectionInfo" type="csta:ConnectionInformation" minOccurs="0"/>
        <xsd:element ref="csta:callLinkageData" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>
</xsd:schema>
```

15.2.2 Call cleared

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-call-cleared-event</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="event-cause.xsd" />
  <xsd:include schemaLocation="call-connection-identifiers.xsd" />
  <xsd:include schemaLocation="device-feature-types.xsd" />
  <xsd:include schemaLocation="extension-types.xsd" />
  <xsd:include schemaLocation="call-control.xsd" />
  <xsd:include schemaLocation="media-services.xsd" />
  <xsd:include schemaLocation="status-reporting.xsd" />

  <xsd:element name="CallClearedEvent">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:monitorCrossRefID" />
        <xsd:element name="clearedCall" type="csta:ConnectionID" />
        <xsd:element ref="csta:correlatorData" minOccurs="0" />
        <xsd:element ref="csta:userData" minOccurs="0" />
        <xsd:element ref="csta:cause" />
        <xsd:element ref="csta:mediaCallCharacteristics" minOccurs="0" />
        <xsd:element ref="csta:callCharacteristics" minOccurs="0" />
        <xsd:element ref="csta:callLinkageData" minOccurs="0" />
        <xsd:element ref="csta:extensions" minOccurs="0" />
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```


15.2.3 Conferenced

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-conferenced-event</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="event-cause.xsd"/>
  <xsd:include schemaLocation="device-identifiers.xsd"/>
  <xsd:include schemaLocation="call-connection-identifiers.xsd"/>
  <xsd:include schemaLocation="connection-states.xsd"/>
  <xsd:include schemaLocation="device-feature-types.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>
  <xsd:include schemaLocation="call-control.xsd"/>
  <xsd:include schemaLocation="media-services.xsd"/>
  <xsd:include schemaLocation="status-reporting.xsd"/>

  <xsd:element name="ConferencedEvent">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:monitorCrossRefID"/>
        <xsd:element name="primaryOldCall" type="csta:ConnectionID"/>
        <xsd:element name="secondaryOldCall" type="csta:ConnectionID" minOccurs="0"/>
        <xsd:element name="conferencingDevice" type="csta:SubjectDeviceID"/>
        <xsd:element name="addedParty" type="csta:SubjectDeviceID"/>
        <xsd:element name="conferenceConnections" type="csta:ConnectionList"/>
        <xsd:element ref="csta:localConnectionInfo" minOccurs="0"/>
        <xsd:element ref="csta:correlatorData" minOccurs="0"/>
        <xsd:element ref="csta:userData" minOccurs="0"/>
        <xsd:element ref="csta:cause"/>
        <xsd:element ref="csta:servicesPermitted" minOccurs="0"/>
        <xsd:element ref="csta:mediaCallCharacteristics" minOccurs="0"/>
        <xsd:element ref="csta:callCharacteristics" minOccurs="0"/>
        <xsd:element name="callLinkageDataList" type="csta:CallLinkageDataList" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>
</xsd:schema>
```

15.2.4 Connection cleared

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-connection-cleared-event</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="event-cause.xsd" />
  <xsd:include schemaLocation="device-identifiers.xsd" />
  <xsd:include schemaLocation="call-connection-identifiers.xsd" />
  <xsd:include schemaLocation="connection-states.xsd" />
  <xsd:include schemaLocation="device-feature-types.xsd" />
  <xsd:include schemaLocation="extension-types.xsd" />
  <xsd:include schemaLocation="call-control.xsd" />
  <xsd:include schemaLocation="charge-info.xsd" />
  <xsd:include schemaLocation="media-services.xsd" />
  <xsd:include schemaLocation="status-reporting.xsd" />

  <xsd:element name="ConnectionClearedEvent">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:monitorCrossRefID" />
        <xsd:element name="droppedConnection" type="csta:ConnectionID" />
        <xsd:element name="releasingDevice" type="csta:SubjectDeviceID" />
        <xsd:element ref="csta:localConnectionInfo" minOccurs="0" />
        <xsd:element ref="csta:correlatorData" minOccurs="0" />
        <xsd:element ref="csta:userData" minOccurs="0" />
        <xsd:element name="chargingInfo" type="csta:ChargingInfo" minOccurs="0" />
        <xsd:element ref="csta:cause" />
        <xsd:element ref="csta:servicesPermitted" minOccurs="0" />
        <xsd:element ref="csta:mediaCallCharacteristics" minOccurs="0" />
        <xsd:element ref="csta:callCharacteristics" minOccurs="0" />
        <xsd:element name="droppedConnectionInfo" type="csta:ConnectionInformation" minOccurs="0" />
        <xsd:element ref="csta:callLinkageData" minOccurs="0" />
        <xsd:element ref="csta:extensions" minOccurs="0" />
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

15.2.5 Delivered

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-delivered-event</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="event-cause.xsd"/>
  <xsd:include schemaLocation="device-identifiers.xsd"/>
  <xsd:include schemaLocation="call-connection-identifiers.xsd"/>
  <xsd:include schemaLocation="connection-states.xsd"/>
  <xsd:include schemaLocation="device-feature-types.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>
  <xsd:include schemaLocation="call-control.xsd"/>
  <xsd:include schemaLocation="media-services.xsd"/>
  <xsd:include schemaLocation="status-reporting.xsd"/>

  <xsd:element name="DeliveredEvent">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:monitorCrossRefID"/>
        <xsd:element name="connection" type="csta:ConnectionID"/>
        <xsd:element name="alertingDevice" type="csta:SubjectDeviceID"/>
        <xsd:element ref="csta:callingDevice"/>
        <xsd:element ref="csta:calledDevice"/>
        <xsd:element ref="csta:lastRedirectionDevice"/>
        <xsd:element name="originatingNIDConnection" type="csta:ConnectionID" minOccurs="0"/>
        <xsd:element ref="csta:localConnectionInfo" minOccurs="0"/>
        <xsd:element ref="csta:correlatorData" minOccurs="0"/>
        <xsd:element ref="csta:userData" minOccurs="0"/>
        <xsd:element ref="csta:cause"/>
        <xsd:element ref="csta:servicesPermitted" minOccurs="0"/>
        <xsd:element ref="csta:networkCallingDevice" minOccurs="0"/>
        <xsd:element ref="csta:networkCalledDevice" minOccurs="0"/>
        <xsd:element ref="csta:associatedCallingDevice" minOccurs="0"/>
        <xsd:element ref="csta:associatedCalledDevice" minOccurs="0"/>
        <xsd:element ref="csta:mediaCallCharacteristics" minOccurs="0"/>
        <xsd:element ref="csta:callCharacteristics" minOccurs="0"/>
        <xsd:element name="connectionInfo" type="csta:ConnectionInformation" minOccurs="0"/>
        <xsd:element ref="csta:callLinkageData" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

15.2.6 Digits dialed

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-digits-dialed-event</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="event-cause.xsd" />
  <xsd:include schemaLocation="device-identifiers.xsd" />
  <xsd:include schemaLocation="call-connection-identifiers.xsd" />
  <xsd:include schemaLocation="connection-states.xsd" />
  <xsd:include schemaLocation="device-feature-types.xsd" />
  <xsd:include schemaLocation="extension-types.xsd" />
  <xsd:include schemaLocation="call-control.xsd" />
  <xsd:include schemaLocation="media-services.xsd" />
  <xsd:include schemaLocation="status-reporting.xsd" />

  <xsd:element name="DigitsDialedEvent">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:monitorCrossRefID" />
        <xsd:element name="dialingConnection" type="csta:ConnectionID" />
        <xsd:element name="dialingDevice" type="csta:SubjectDeviceID" />
        <xsd:element name="dialingSequence" type="csta:DeviceID" />
        <xsd:element ref="csta:localConnectionInfo" minOccurs="0" />
        <xsd:element ref="csta:correlatorData" minOccurs="0" />
        <xsd:element ref="csta:cause" />
        <xsd:element ref="csta:servicesPermitted" minOccurs="0" />
        <xsd:element ref="csta:networkCallingDevice" minOccurs="0" />
        <xsd:element ref="csta:networkCalledDevice" minOccurs="0" />
        <xsd:element ref="csta:associatedCallingDevice" minOccurs="0" />
        <xsd:element ref="csta:associatedCalledDevice" minOccurs="0" />
        <xsd:element name="dialingConnectionInfo" type="csta:ConnectionInformation" minOccurs="0" />
        <xsd:element ref="csta:callCharacteristics" minOccurs="0" />
        <xsd:element ref="csta:callLinkageData" minOccurs="0" />
        <xsd:element ref="csta:extensions" minOccurs="0" />
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

15.2.7 Diverted

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-diverted-event</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="event-cause.xsd"/>
  <xsd:include schemaLocation="device-identifiers.xsd"/>
  <xsd:include schemaLocation="call-connection-identifiers.xsd"/>
  <xsd:include schemaLocation="connection-states.xsd"/>
  <xsd:include schemaLocation="device-feature-types.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>
  <xsd:include schemaLocation="call-control.xsd"/>
  <xsd:include schemaLocation="media-services.xsd"/>
  <xsd:include schemaLocation="status-reporting.xsd"/>

  <xsd:element name="DivertedEvent">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:monitorCrossRefID"/>
        <xsd:element name="connection" type="csta:ConnectionID"/>
        <xsd:element name="divertingDevice" type="csta:SubjectDeviceID"/>
        <xsd:element name="newDestination" type="csta:SubjectDeviceID"/>
        <xsd:element ref="csta:callingDevice" minOccurs="0"/>
        <xsd:element ref="csta:calledDevice" minOccurs="0"/>
        <xsd:element ref="csta:lastRedirectionDevice"/>
        <xsd:element ref="csta:localConnectionInfo" minOccurs="0"/>
        <xsd:element ref="csta:correlatorData" minOccurs="0"/>
        <xsd:element ref="csta:userData" minOccurs="0"/>
        <xsd:element ref="csta:cause"/>
        <xsd:element ref="csta:servicesPermitted" minOccurs="0"/>
        <xsd:element ref="csta:mediaCallCharacteristics" minOccurs="0"/>
        <xsd:element ref="csta:callCharacteristics" minOccurs="0"/>
        <xsd:element name="connectionInfo" type="csta:ConnectionInformation" minOccurs="0"/>
        <xsd:element ref="csta:networkCallingDevice" minOccurs="0"/>
        <xsd:element ref="csta:networkCalledDevice" minOccurs="0"/>
        <xsd:element ref="csta:associatedCallingDevice" minOccurs="0"/>
        <xsd:element ref="csta:associatedCalledDevice" minOccurs="0"/>
        <xsd:element ref="csta:callLinkageData" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

15.2.8 Established

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-established-event</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="event-cause.xsd" />
  <xsd:include schemaLocation="device-identifiers.xsd" />
  <xsd:include schemaLocation="call-connection-identifiers.xsd" />
  <xsd:include schemaLocation="connection-states.xsd" />
  <xsd:include schemaLocation="device-feature-types.xsd" />
  <xsd:include schemaLocation="extension-types.xsd" />
  <xsd:include schemaLocation="call-control.xsd" />
  <xsd:include schemaLocation="media-services.xsd" />
  <xsd:include schemaLocation="status-reporting.xsd" />

  <xsd:element name="EstablishedEvent">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:monitorCrossRefID" />
        <xsd:element name="establishedConnection" type="csta:ConnectionID" />
        <xsd:element name="answeringDevice" type="csta:SubjectDeviceID" />
        <xsd:element ref="csta:callingDevice" />
        <xsd:element ref="csta:calledDevice" />
        <xsd:element ref="csta:lastRedirectionDevice" />
        <xsd:element name="originatinNIDConnection" type="csta:ConnectionID" minOccurs="0" />
        <xsd:element ref="csta:localConnectionInfo" minOccurs="0" />
        <xsd:element ref="csta:correlatorData" minOccurs="0" />
        <xsd:element ref="csta:userData" minOccurs="0" />
        <xsd:element ref="csta:cause" />
        <xsd:element ref="csta:servicesPermitted" minOccurs="0" />
        <xsd:element ref="csta:networkCallingDevice" minOccurs="0" />
        <xsd:element ref="csta:networkCalledDevice" minOccurs="0" />
        <xsd:element ref="csta:associatedCallingDevice" minOccurs="0" />
        <xsd:element ref="csta:associatedCalledDevice" minOccurs="0" />
        <xsd:element ref="csta:mediaCallCharacteristics" minOccurs="0" />
        <xsd:element ref="csta:callCharacteristics" minOccurs="0" />
        <xsd:element name="establishedConnectionInfo" type="csta:ConnectionInformation"
minOccurs="0" />
        <xsd:element ref="csta:callLinkageData" minOccurs="0" />
        <xsd:element ref="csta:extensions" minOccurs="0" />
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

15.2.9 Failed

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-failed-event</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="event-cause.xsd"/>
  <xsd:include schemaLocation="device-identifiers.xsd"/>
  <xsd:include schemaLocation="call-connection-identifiers.xsd"/>
  <xsd:include schemaLocation="connection-states.xsd"/>
  <xsd:include schemaLocation="device-feature-types.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>
  <xsd:include schemaLocation="call-control.xsd"/>
  <xsd:include schemaLocation="media-services.xsd"/>
  <xsd:include schemaLocation="status-reporting.xsd"/>

  <xsd:element name="FailedEvent">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:monitorCrossRefID"/>
        <xsd:element name="failedConnection" type="csta:ConnectionID"/>
        <xsd:element name="failingDevice" type="csta:SubjectDeviceID"/>
        <xsd:element ref="csta:callingDevice"/>
        <xsd:element ref="csta:calledDevice"/>
        <xsd:element ref="csta:lastRedirectionDevice"/>
        <xsd:element name="originatinNIDConnection" type="csta:ConnectionID" minOccurs="0"/>
        <xsd:element ref="csta:localConnectionInfo" minOccurs="0"/>
        <xsd:element ref="csta:correlatorData" minOccurs="0"/>
        <xsd:element ref="csta:userData" minOccurs="0"/>
        <xsd:element ref="csta:cause"/>
        <xsd:element ref="csta:servicesPermitted" minOccurs="0"/>
        <xsd:element ref="csta:networkCallingDevice" minOccurs="0"/>
        <xsd:element ref="csta:networkCalledDevice" minOccurs="0"/>
        <xsd:element ref="csta:associatedCallingDevice" minOccurs="0"/>
        <xsd:element ref="csta:associatedCalledDevice" minOccurs="0"/>
        <xsd:element ref="csta:mediaCallCharacteristics" minOccurs="0"/>
        <xsd:element ref="csta:callCharacteristics" minOccurs="0"/>
        <xsd:element name="failedConnectionInfo" type="csta:ConnectionInformation" minOccurs="0"/>
        <xsd:element ref="csta:callLinkageData" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

15.2.10 Held

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-held-event</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="event-cause.xsd" />
  <xsd:include schemaLocation="device-identifiers.xsd" />
  <xsd:include schemaLocation="call-connection-identifiers.xsd" />
  <xsd:include schemaLocation="connection-states.xsd" />
  <xsd:include schemaLocation="device-feature-types.xsd" />
  <xsd:include schemaLocation="extension-types.xsd" />
  <xsd:include schemaLocation="call-control.xsd" />
  <xsd:include schemaLocation="media-services.xsd" />
  <xsd:include schemaLocation="status-reporting.xsd" />

  <xsd:element name="HeldEvent">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:monitorCrossRefID" />
        <xsd:element name="heldConnection" type="csta:ConnectionID" />
        <xsd:element name="holdingDevice" type="csta:SubjectDeviceID" />
        <xsd:element ref="csta:localConnectionInfo" minOccurs="0" />
        <xsd:element ref="csta:correlatorData" minOccurs="0" />
        <xsd:element ref="csta:cause" />
        <xsd:element ref="csta:servicesPermitted" minOccurs="0" />
        <xsd:element ref="csta:mediaCallCharacteristics" minOccurs="0" />
        <xsd:element ref="csta:callCharacteristics" minOccurs="0" />
        <xsd:element name="heldConnectionInfo" type="csta:ConnectionInformation" minOccurs="0" />
        <xsd:element ref="csta:callLinkageData" minOccurs="0" />
        <xsd:element ref="csta:extensions" minOccurs="0" />
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>
</xsd:schema>
```


15.2.11 Network capabilities changed

```
<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-network-capabilities-changed-event</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="event-cause.xsd"/>
  <xsd:include schemaLocation="device-identifiers.xsd"/>
  <xsd:include schemaLocation="call-connection-identifiers.xsd"/>
  <xsd:include schemaLocation="connection-states.xsd"/>
  <xsd:include schemaLocation="device-feature-types.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>
  <xsd:include schemaLocation="call-control.xsd"/>
  <xsd:include schemaLocation="media-services.xsd"/>
  <xsd:include schemaLocation="status-reporting.xsd"/>

  <xsd:element name="NetworkCapabilitiesChangedEvent">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:monitorCrossRefID"/>
        <xsd:element name="outboundConnection" type="csta:ConnectionID"/>
        <xsd:element name="networkInterfaceUsed" type="csta:SubjectDeviceID"/>
        <xsd:element ref="csta:calledDevice"/>
        <xsd:element name="progressIndicator" type="csta:ProgressIndicator"/>
        <xsd:element ref="csta:localConnectionInfo" minOccurs="0"/>
        <xsd:element ref="csta:correlatorData" minOccurs="0"/>
        <xsd:element ref="csta:userData" minOccurs="0"/>
        <xsd:element name="networkCapability" type="csta:NetworkCapability" minOccurs="0"/>
        <xsd:element ref="csta:cause"/>
        <xsd:element ref="csta:servicesPermitted" minOccurs="0"/>
        <xsd:element ref="csta:mediaCallCharacteristics" minOccurs="0"/>
        <xsd:element ref="csta:callCharacteristics" minOccurs="0"/>
        <xsd:element name="outboundConnectionInfo" type="csta:ConnectionInformation"
minOccurs="0"/>
        <xsd:element ref="csta:callLinkageData" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>
</xsd:schema>
```

15.2.12 Network reached

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-network-reached-event</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="event-cause.xsd"/>
  <xsd:include schemaLocation="device-identifiers.xsd"/>
  <xsd:include schemaLocation="call-connection-identifiers.xsd"/>
  <xsd:include schemaLocation="connection-states.xsd"/>
  <xsd:include schemaLocation="device-feature-types.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>
  <xsd:include schemaLocation="call-control.xsd"/>
  <xsd:include schemaLocation="media-services.xsd"/>
  <xsd:include schemaLocation="status-reporting.xsd"/>

  <xsd:element name="NetworkReachedEvent">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:monitorCrossRefID"/>
        <xsd:element name="outboundConnection" type="csta:ConnectionID"/>
        <xsd:element name="networkInterfaceUsed" type="csta:SubjectDeviceID"/>
        <xsd:element ref="csta:callingDevice"/>
        <xsd:element ref="csta:calledDevice"/>
        <xsd:element ref="csta:lastRedirectionDevice"/>
        <xsd:element name="originatinNIDConnection" type="csta:ConnectionID" minOccurs="0"/>
        <xsd:element ref="csta:localConnectionInfo" minOccurs="0"/>
        <xsd:element ref="csta:correlatorData" minOccurs="0"/>
        <xsd:element ref="csta:userData" minOccurs="0"/>
        <xsd:element name="networkCapability" type="csta:NetworkCapability" minOccurs="0"/>
        <xsd:element ref="csta:cause"/>
        <xsd:element ref="csta:servicesPermitted" minOccurs="0"/>
        <xsd:element ref="csta:mediaCallCharacteristics" minOccurs="0"/>
        <xsd:element ref="csta:callCharacteristics" minOccurs="0"/>
        <xsd:element name="outboundConnectionInfo" type="csta:ConnectionInformation"
minOccurs="0"/>
        <xsd:element ref="csta:networkCallingDevice" minOccurs="0"/>
        <xsd:element ref="csta:networkCalledDevice" minOccurs="0"/>
        <xsd:element ref="csta:associatedCallingDevice" minOccurs="0"/>
        <xsd:element ref="csta:callLinkageData" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

15.2.13 Offered

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-offered-event</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="event-cause.xsd"/>
  <xsd:include schemaLocation="device-identifiers.xsd"/>
  <xsd:include schemaLocation="call-connection-identifiers.xsd"/>
  <xsd:include schemaLocation="connection-states.xsd"/>
  <xsd:include schemaLocation="device-feature-types.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>
  <xsd:include schemaLocation="call-control.xsd"/>
  <xsd:include schemaLocation="media-services.xsd"/>
  <xsd:include schemaLocation="status-reporting.xsd"/>

  <xsd:element name="OfferedEvent">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:monitorCrossRefID"/>
        <xsd:element name="offeredConnection" type="csta:ConnectionID"/>
        <xsd:element name="offeredDevice" type="csta:SubjectDeviceID"/>
        <xsd:element ref="csta:callingDevice"/>
        <xsd:element ref="csta:calledDevice"/>
        <xsd:element ref="csta:lastRedirectionDevice"/>
        <xsd:element name="originatingNIDConnection" type="csta:ConnectionID" minOccurs="0"/>
        <xsd:element ref="csta:localConnectionInfo" minOccurs="0"/>
        <xsd:element ref="csta:correlatorData" minOccurs="0"/>
        <xsd:element ref="csta:userData" minOccurs="0"/>
        <xsd:element ref="csta:cause"/>
        <xsd:element ref="csta:servicesPermitted" minOccurs="0"/>
        <xsd:element ref="csta:networkCallingDevice" minOccurs="0"/>
        <xsd:element ref="csta:networkCalledDevice" minOccurs="0"/>
        <xsd:element ref="csta:associatedCallingDevice" minOccurs="0"/>
        <xsd:element ref="csta:associatedCalledDevice" minOccurs="0"/>
        <xsd:element ref="csta:mediaCallCharacteristics" minOccurs="0"/>
        <xsd:element ref="csta:callCharacteristics" minOccurs="0"/>
        <xsd:element name="offeredConnectionInfo" type="csta:ConnectionInformation" minOccurs="0"/>
        <xsd:element ref="csta:callLinkageData" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

15.2.14 Originated

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-originated-event</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="event-cause.xsd" />
  <xsd:include schemaLocation="device-identifiers.xsd" />
  <xsd:include schemaLocation="call-connection-identifiers.xsd" />
  <xsd:include schemaLocation="connection-states.xsd" />
  <xsd:include schemaLocation="device-feature-types.xsd" />
  <xsd:include schemaLocation="extension-types.xsd" />
  <xsd:include schemaLocation="call-control.xsd" />
  <xsd:include schemaLocation="media-services.xsd" />
  <xsd:include schemaLocation="status-reporting.xsd" />

  <xsd:element name="OriginatedEvent">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:monitorCrossRefID" />
        <xsd:element name="originatedConnection" type="csta:ConnectionID" />
        <xsd:element name="callingDevice" type="csta:SubjectDeviceID" />
        <xsd:element ref="csta:calledDevice" />
        <xsd:element name="originatingDevice" type="csta:DeviceID" minOccurs="0" />
        <xsd:element ref="csta:localConnectionInfo" minOccurs="0" />
        <xsd:element ref="csta:correlatorData" minOccurs="0" />
        <xsd:element ref="csta:cause" />
        <xsd:element ref="csta:servicesPermitted" minOccurs="0" />
        <xsd:element ref="csta:networkCallingDevice" minOccurs="0" />
        <xsd:element ref="csta:networkCalledDevice" minOccurs="0" />
        <xsd:element ref="csta:associatedCallingDevice" minOccurs="0" />
        <xsd:element ref="csta:associatedCalledDevice" minOccurs="0" />
        <xsd:element ref="csta:mediaCallCharacteristics" minOccurs="0" />
        <xsd:element ref="csta:callCharacteristics" minOccurs="0" />
        <xsd:element name="originatedConnectionInfo" type="csta:ConnectionInformation"
minOccurs="0" />
        <xsd:element ref="csta:callLinkageData" minOccurs="0" />
        <xsd:element ref="csta:extensions" minOccurs="0" />
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

15.2.15 Queued

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-queued-event</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="event-cause.xsd"/>
  <xsd:include schemaLocation="device-identifiers.xsd"/>
  <xsd:include schemaLocation="call-connection-identifiers.xsd"/>
  <xsd:include schemaLocation="connection-states.xsd"/>
  <xsd:include schemaLocation="device-feature-types.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>
  <xsd:include schemaLocation="call-control.xsd"/>
  <xsd:include schemaLocation="media-services.xsd"/>
  <xsd:include schemaLocation="status-reporting.xsd"/>

  <xsd:element name="QueuedEvent">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:monitorCrossRefID"/>
        <xsd:element name="queuedConnection" type="csta:ConnectionID"/>
        <xsd:element name="queue" type="csta:SubjectDeviceID"/>
        <xsd:element ref="csta:callingDevice"/>
        <xsd:element ref="csta:calledDevice"/>
        <xsd:element ref="csta:lastRedirectionDevice"/>
        <xsd:element name="numberQueued" type="xsd:long" minOccurs="0"/>
        <xsd:element name="callsInFront" type="xsd:long" minOccurs="0"/>
        <xsd:element ref="csta:localConnectionInfo" minOccurs="0"/>
        <xsd:element ref="csta:correlatorData" minOccurs="0"/>
        <xsd:element ref="csta:userData" minOccurs="0"/>
        <xsd:element ref="csta:cause"/>
        <xsd:element ref="csta:servicesPermitted" minOccurs="0"/>
        <xsd:element ref="csta:networkCallingDevice" minOccurs="0"/>
        <xsd:element ref="csta:networkCalledDevice" minOccurs="0"/>
        <xsd:element ref="csta:associatedCallingDevice" minOccurs="0"/>
        <xsd:element ref="csta:associatedCalledDevice" minOccurs="0"/>
        <xsd:element ref="csta:mediaCallCharacteristics" minOccurs="0"/>
        <xsd:element ref="csta:callCharacteristics" minOccurs="0"/>
        <xsd:element name="queuedConnectionInfo" type="csta:ConnectionInformation" minOccurs="0"/>
        <xsd:element ref="csta:callLinkageData" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

15.2.16 Retrieved

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-retrieved-event</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="event-cause.xsd" />
  <xsd:include schemaLocation="device-identifiers.xsd" />
  <xsd:include schemaLocation="call-connection-identifiers.xsd" />
  <xsd:include schemaLocation="connection-states.xsd" />
  <xsd:include schemaLocation="device-feature-types.xsd" />
  <xsd:include schemaLocation="extension-types.xsd" />
  <xsd:include schemaLocation="call-control.xsd" />
  <xsd:include schemaLocation="media-services.xsd" />
  <xsd:include schemaLocation="status-reporting.xsd" />

  <xsd:element name="RetrievedEvent">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:monitorCrossRefID" />
        <xsd:element name="retrievedConnection" type="csta:ConnectionID" />
        <xsd:element name="retrievingDevice" type="csta:SubjectDeviceID" />
        <xsd:element ref="csta:localConnectionInfo" minOccurs="0" />
        <xsd:element ref="csta:correlatorData" minOccurs="0" />
        <xsd:element ref="csta:cause" />
        <xsd:element ref="csta:servicesPermitted" minOccurs="0" />
        <xsd:element ref="csta:mediaCallCharacteristics" minOccurs="0" />
        <xsd:element ref="csta:callCharacteristics" minOccurs="0" />
        <xsd:element name="retrievedConnectionInfo" type="csta:ConnectionInformation"
minOccurs="0" />
        <xsd:element ref="csta:callLinkageData" minOccurs="0" />
        <xsd:element ref="csta:extensions" minOccurs="0" />
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

15.2.17 Service initiated

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-service-initiated-event</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="event-cause.xsd"/>
  <xsd:include schemaLocation="device-identifiers.xsd"/>
  <xsd:include schemaLocation="call-connection-identifiers.xsd"/>
  <xsd:include schemaLocation="connection-states.xsd"/>
  <xsd:include schemaLocation="device-feature-types.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>
  <xsd:include schemaLocation="call-control.xsd"/>
  <xsd:include schemaLocation="media-services.xsd"/>
  <xsd:include schemaLocation="status-reporting.xsd"/>

  <xsd:element name="ServiceInitiatedEvent">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:monitorCrossRefID"/>
        <xsd:element name="initiatedConnection" type="csta:ConnectionID"/>
        <xsd:element name="initiatingDevice" type="csta:SubjectDeviceID"/>
        <xsd:element ref="csta:localConnectionInfo" minOccurs="0"/>
        <xsd:element ref="csta:correlatorData" minOccurs="0"/>
        <xsd:element ref="csta:cause"/>
        <xsd:element ref="csta:servicesPermitted" minOccurs="0"/>
        <xsd:element ref="csta:mediaCallCharacteristics" minOccurs="0"/>
        <xsd:element ref="csta:callCharacteristics" minOccurs="0"/>
        <xsd:element name="initiatedConnectionInfo" type="csta:ConnectionInformation"
minOccurs="0"/>
        <xsd:element ref="csta:networkCallingDevice" minOccurs="0"/>
        <xsd:element ref="csta:networkCalledDevice" minOccurs="0"/>
        <xsd:element ref="csta:associatedCallingDevice" minOccurs="0"/>
        <xsd:element ref="csta:callLinkageData" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

15.2.18 Transferred

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-transferred-event</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="event-cause.xsd" />
  <xsd:include schemaLocation="device-identifiers.xsd" />
  <xsd:include schemaLocation="call-connection-identifiers.xsd" />
  <xsd:include schemaLocation="connection-states.xsd" />
  <xsd:include schemaLocation="device-feature-types.xsd" />
  <xsd:include schemaLocation="extension-types.xsd" />
  <xsd:include schemaLocation="call-control.xsd" />
  <xsd:include schemaLocation="charge-info.xsd" />
  <xsd:include schemaLocation="media-services.xsd" />
  <xsd:include schemaLocation="status-reporting.xsd" />

  <xsd:element name="TransferredEvent">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:monitorCrossRefID" />
        <xsd:element name="primaryOldCall" type="csta:ConnectionID" />
        <xsd:element name="secondaryOldCall" type="csta:ConnectionID" minOccurs="0" />
        <xsd:element name="transferringDevice" type="csta:SubjectDeviceID" />
        <xsd:element name="transferredToDevice" type="csta:SubjectDeviceID" />
        <xsd:element name="transferredConnections" type="csta:ConnectionList" />
        <xsd:element ref="csta:localConnectionInfo" minOccurs="0" />
        <xsd:element ref="csta:correlatorData" minOccurs="0" />
        <xsd:element ref="csta:userData" minOccurs="0" />
        <xsd:element name="chargingInfo" type="csta:ChargingInfo" minOccurs="0" />
        <xsd:element ref="csta:cause" />
        <xsd:element ref="csta:servicesPermitted" minOccurs="0" />
        <xsd:element ref="csta:mediaCallCharacteristics" minOccurs="0" />
        <xsd:element ref="csta:callCharacteristics" minOccurs="0" />
        <xsd:element name="callLinkageDataList" type="csta:CallLinkageDataList" minOccurs="0" />
        <xsd:element ref="csta:extensions" minOccurs="0" />
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```


16 Call associated features

16.1 Services

16.1.1 Associate data

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-associate-data</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="call-connection-identifiers.xsd"/>
  <xsd:include schemaLocation="device-feature-types.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>
  <xsd:include schemaLocation="call-control.xsd"/>

  <xsd:element name="AssociateData">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="existingCall" type="csta:ConnectionID"/>
        <xsd:element ref="csta:accountCode" minOccurs="0"/>
        <xsd:element ref="csta:authCode" minOccurs="0"/>
        <xsd:element ref="csta:correlatorData" minOccurs="0"/>
        <xsd:element name="callQualifyingData" type="csta:CallQualifyingData" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="AssociateDataResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

16.1.2 Cancel telephony tones

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-cancel-telephony-tones</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="call-connection-identifiers.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>

  <xsd:element name="CancelTelephonyTones">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="connectionToStopTone" type="csta:ConnectionID"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="CancelTelephonyTonesResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

16.1.3 Generate digits

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-generate-digits</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="call-connection-identifiers.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>

  <xsd:element name="GenerateDigits">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="connectionToSendDigits" type="csta:ConnectionID"/>
        <xsd:element name="digitMode" type="csta:DigitMode" minOccurs="0"/>
        <xsd:element name="charactersToSend" type="xsd:string"/>
        <xsd:element name="toneDuration" type="xsd:long" minOccurs="0"/>
        <xsd:element name="pulseRate" type="xsd:long" minOccurs="0"/>
        <xsd:element name="pauseDuration" type="xsd:long" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="GenerateDigitsResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:simpleType name="DigitMode">
    <xsd:restriction base="xsd:string">
      <xsd:enumeration value="rotaryPulse"/>
      <xsd:enumeration value="DTMF"/>
    </xsd:restriction>
  </xsd:simpleType>

</xsd:schema>
```

16.1.4 Generate telephony tones

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-generate-telephony-tones</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="call-connection-identifiers.xsd"/>
  <xsd:include schemaLocation="device-feature-types.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>

  <xsd:element name="GenerateTelephonyTones">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="connectionToSendTone" type="csta:ConnectionID"/>
        <xsd:element name="toneToSend" type="csta:TelephonyTone"/>
        <xsd:element name="toneDuration" type="xsd:long" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="GenerateTelephonyTonesResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

16.1.5 Send user information

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-send-user-information</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="call-connection-identifiers.xsd"/>
  <xsd:include schemaLocation="device-feature-types.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>

  <xsd:element name="SendUserInformation">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="existingCall" type="csta:ConnectionID"/>
        <xsd:element ref="csta:userData"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="SendUserInformationResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

16.2 Events

16.2.1 Call information

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

```
<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"  
xmlns:xsd="http://www.w3.org/2001/XMLSchema"  
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"  
elementFormDefault="qualified" attributeFormDefault="unqualified">
```

```
  <xsd:annotation>  
    <xsd:documentation>CSTA-call-information-event</xsd:documentation>  
  </xsd:annotation>
```

```
  <xsd:include schemaLocation="device-identifiers.xsd"/>  
  <xsd:include schemaLocation="call-connection-identifiers.xsd"/>  
  <xsd:include schemaLocation="device-feature-types.xsd"/>  
  <xsd:include schemaLocation="extension-types.xsd"/>  
  <xsd:include schemaLocation="call-control.xsd"/>  
  <xsd:include schemaLocation="media-services.xsd"/>  
  <xsd:include schemaLocation="status-reporting.xsd"/>
```

```
  <xsd:element name="CallInformationEvent">  
    <xsd:complexType>  
      <xsd:sequence>  
        <xsd:element ref="csta:monitorCrossRefID"/>  
        <xsd:element name="connection" type="csta:ConnectionID"/>  
        <xsd:element name="device" type="csta:SubjectDeviceID"/>  
        <xsd:element ref="csta:callingDevice" minOccurs="0"/>  
        <xsd:element name="accountInfo" type="csta:AccountInfo" minOccurs="0"/>  
        <xsd:element ref="csta:authCode" minOccurs="0"/>  
        <xsd:element ref="csta:correlatorData" minOccurs="0"/>  
        <xsd:element ref="csta:servicesPermitted" minOccurs="0"/>  
        <xsd:element ref="csta:userData" minOccurs="0"/>  
        <xsd:element name="callQualifyingData" type="csta:CallQualifyingData" minOccurs="0"/>  
        <xsd:element name="connectionInfo" type="csta:ConnectionInformation" minOccurs="0"/>  
        <xsd:element name="callLinkageDataList" type="csta:CallLinkageDataList" minOccurs="0"/>  
        <xsd:element ref="csta:extensions" minOccurs="0"/>  
      </xsd:sequence>  
    </xsd:complexType>  
  </xsd:element>
```

```
</xsd:schema>
```

16.2.2 Charging

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-charging-event</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="event-cause.xsd"/>
  <xsd:include schemaLocation="device-identifiers.xsd"/>
  <xsd:include schemaLocation="call-connection-identifiers.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>
  <xsd:include schemaLocation="charge-info.xsd"/>
  <xsd:include schemaLocation="status-reporting.xsd"/>

  <xsd:element name="ChargingEvent">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:monitorCrossRefID"/>
        <xsd:element name="connection" type="csta:ConnectionID"/>
        <xsd:element name="chargedDevice" type="csta:DeviceID"/>
        <xsd:element name="chargingInfo" type="csta:ChargingInfo"/>
        <xsd:element ref="csta:cause" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

16.2.3 Digits generated

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-digits-generated-event</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="call-connection-identifiers.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>
  <xsd:include schemaLocation="media-services.xsd"/>
  <xsd:include schemaLocation="status-reporting.xsd"/>

  <xsd:element name="DigitsGeneratedEvent">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:monitorCrossRefID"/>
        <xsd:element name="connection" type="csta:ConnectionID"/>
        <xsd:element name="digitGeneratedList" type="xsd:string"/>
        <xsd:element name="digitDurationList" minOccurs="0">
          <xsd:complexType>
            <xsd:sequence>
              <xsd:element name="digitDuration" type="xsd:long" minOccurs="0"
maxOccurs="unbounded"/>
            </xsd:sequence>
          </xsd:complexType>
        </xsd:element>
        <xsd:element name="pauseDurationList" minOccurs="0">
          <xsd:complexType>
            <xsd:sequence>
              <xsd:element name="pauseDuration" type="xsd:long" minOccurs="0"
maxOccurs="unbounded"/>
            </xsd:sequence>
          </xsd:complexType>
        </xsd:element>
        <xsd:element name="connectionInfo" type="csta:ConnectionInformation" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```


16.2.4 Telephony tones generated

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-telephony-tones-generated-event</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="call-connection-identifiers.xsd"/>
  <xsd:include schemaLocation="device-feature-types.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>
  <xsd:include schemaLocation="media-services.xsd"/>
  <xsd:include schemaLocation="status-reporting.xsd"/>

  <xsd:element name="TelephonyTonesGeneratedEvent">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:monitorCrossRefID"/>
        <xsd:element name="connection" type="csta:ConnectionID"/>
        <xsd:element name="toneGenerated" type="csta:TelephonyTone" minOccurs="0"/>
        <xsd:element name="toneFrequency" type="xsd:long" minOccurs="0"/>
        <xsd:element name="toneDuration" type="xsd:long" minOccurs="0"/>
        <xsd:element name="pauseDuration" type="xsd:long" minOccurs="0"/>
        <xsd:element name="connectionInfo" type="csta:ConnectionInformation" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

16.2.5 Service completion failure

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-service-completion-failure-event</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="event-cause.xsd" />
  <xsd:include schemaLocation="device-identifiers.xsd" />
  <xsd:include schemaLocation="call-connection-identifiers.xsd" />
  <xsd:include schemaLocation="connection-states.xsd" />
  <xsd:include schemaLocation="extension-types.xsd" />
  <xsd:include schemaLocation="media-services.xsd" />
  <xsd:include schemaLocation="status-reporting.xsd" />

  <xsd:element name="ServiceCompletionFailureEvent">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:monitorCrossRefID" />
        <xsd:element name="primaryCall" type="csta:PrimaryOrSecondaryCall" />
        <xsd:element name="secondaryCall" type="csta:PrimaryOrSecondaryCall" minOccurs="0" />
        <xsd:element name="otherDevsPrimaryCallList" minOccurs="0">
          <xsd:complexType>
            <xsd:sequence>
              <xsd:element name="otherDevsPrimaryCall" type="csta:OtherCall" minOccurs="0"
maxOccurs="unbounded" />
            </xsd:sequence>
          </xsd:complexType>
        </xsd:element>
        <xsd:element name="otherDevsSecondaryCallList" minOccurs="0">
          <xsd:complexType>
            <xsd:sequence>
              <xsd:element name="otherDevsSecondaryCall" type="csta:OtherCall" minOccurs="0"
maxOccurs="unbounded" />
            </xsd:sequence>
          </xsd:complexType>
        </xsd:element>
        <xsd:element ref="csta:mediaCallCharacteristics" minOccurs="0" />
        <xsd:element ref="csta:cause" minOccurs="0" />
        <xsd:element ref="csta:extensions" minOccurs="0" />
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:complexType name="PrimaryOrSecondaryCall">
    <xsd:sequence>
      <xsd:element name="deviceID" type="csta:DeviceID" />
      <xsd:element name="connectionID" type="csta:ConnectionID" />
      <xsd:element name="localConnectionState" type="csta:LocalConnectionState" />
      <xsd:element name="connectionInfo" type="csta:ConnectionInformation" minOccurs="0" />
    </xsd:sequence>
  </xsd:complexType>

  <xsd:complexType name="OtherCall">
    <xsd:sequence>
      <xsd:element name="deviceID" type="csta:DeviceID" />
      <xsd:element name="connectionID" type="csta:ConnectionID" />
      <xsd:element name="localConnectionState" type="csta:LocalConnectionState" minOccurs="0" />
      <xsd:element name="connectionInfo" type="csta:ConnectionInformation" minOccurs="0" />
    </xsd:sequence>
  </xsd:complexType>
</xsd:schema>
```

17 Media attachment services and events

17.1 Services

17.1.1 Attach media service

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-attach-media-service</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-identifiers.xsd"/>
  <xsd:include schemaLocation="call-connection-identifiers.xsd"/>
  <xsd:include schemaLocation="connection-states.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>
  <xsd:include schemaLocation="media-services.xsd"/>

  <xsd:element name="AttachMediaService">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="connection" type="csta:ConnectionID"/>
        <xsd:element name="mediaServiceType" type="csta:MediaServiceType"/>
        <xsd:element name="mediaServiceVersion" type="xsd:long" minOccurs="0"/>
        <xsd:element name="mediaServiceInstanceID" type="csta:MediaServiceInstanceID"
minOccurs="0"/>
        <xsd:element name="connectionMode" type="csta:ConnectionMode"/>
        <xsd:element name="requestedConnectionState" type="csta:LocalConnectionState"
minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="AttachMediaServiceResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="mediaConnection" type="csta:ConnectionID" minOccurs="0"/>
        <xsd:element name="mediaDevice" type="csta:DeviceID" minOccurs="0"/>
        <xsd:element name="mediaServiceInstanceID" type="csta:MediaServiceInstanceID"
minOccurs="0"/>
        <xsd:element name="mediaConnectionInfo" type="csta:ConnectionInformation" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

17.1.2 Detach media service

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-detach-media-service</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="call-connection-identifiers.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>
  <xsd:include schemaLocation="media-services.xsd"/>

  <xsd:element name="DetachMediaService">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="connection" type="csta:ConnectionID"/>
        <xsd:element name="mediaServiceType" type="csta:MediaServiceType"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="DetachMediaServiceResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

17.2 Events

17.2.1 Media attached

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-media-attached-event</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-identifiers.xsd"/>
  <xsd:include schemaLocation="call-connection-identifiers.xsd"/>
  <xsd:include schemaLocation="connection-states.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>
  <xsd:include schemaLocation="call-control.xsd"/>
  <xsd:include schemaLocation="media-services.xsd"/>
  <xsd:include schemaLocation="status-reporting.xsd"/>

  <xsd:element name="MediaAttachedEvent">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:monitorCrossRefID"/>
        <xsd:element name="mediaConnection" type="csta:ConnectionID"/>
        <xsd:element name="mediaDevice" type="csta:SubjectDeviceID"/>
        <xsd:element name="mediaServiceType" type="csta:MediaServiceType"/>
        <xsd:element name="mediaServiceVersion" type="xsd:long" minOccurs="0"/>
        <xsd:element name="mediaServiceInstanceID" type="csta:MediaServiceInstanceID"
minOccurs="0"/>
        <xsd:element name="mediaStreamID" type="csta:MediaStreamID" minOccurs="0"/>
        <xsd:element ref="csta:mediaCallCharacteristics" minOccurs="0"/>
        <xsd:element ref="csta:callCharacteristics" minOccurs="0"/>
        <xsd:element ref="csta:localConnectionInfo" minOccurs="0"/>
        <xsd:element name="mediaConnectionInfo" type="csta:ConnectionInformation" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

17.2.2 Media detached

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-media-detached-event</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-identifiers.xsd"/>
  <xsd:include schemaLocation="call-connection-identifiers.xsd"/>
  <xsd:include schemaLocation="connection-states.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>
  <xsd:include schemaLocation="call-control.xsd"/>
  <xsd:include schemaLocation="media-services.xsd"/>
  <xsd:include schemaLocation="status-reporting.xsd"/>

  <xsd:element name="MediaDetachedEvent">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:monitorCrossRefID"/>
        <xsd:element name="mediaConnection" type="csta:ConnectionID"/>
        <xsd:element name="mediaDevice" type="csta:SubjectDeviceID"/>
        <xsd:element name="mediaServiceType" type="csta:MediaServiceType"/>
        <xsd:element name="mediaServiceVersion" type="xsd:long" minOccurs="0"/>
        <xsd:element name="mediaServiceInstanceID" type="csta:MediaServiceInstanceID"
minOccurs="0"/>
        <xsd:element name="mediaStreamID" type="csta:MediaStreamID" minOccurs="0"/>
        <xsd:element ref="csta:mediaCallCharacteristics" minOccurs="0"/>
        <xsd:element ref="csta:callCharacteristics" minOccurs="0"/>
        <xsd:element ref="csta:localConnectionInfo" minOccurs="0"/>
        <xsd:element name="mediaConnectionInfo" type="csta:ConnectionInformation" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

18 Routing services

18.1 Registration services

18.1.1 Route register

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-route-register</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-identifiers.xsd"/>
  <xsd:include schemaLocation="device-feature-types.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>
  <xsd:include schemaLocation="media-services.xsd"/>

  <xsd:element name="RouteRegister">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="routingDevice" type="csta:DeviceID" minOccurs="0"/>
        <xsd:element name="requestedRoutingMediaClass" type="csta:MediaClass" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="RouteRegisterResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:routeRegisterReqID"/>
        <xsd:element name="actualRoutingMediaClass" type="csta:MediaClass" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

18.1.2 Route register abort

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-route-register-abort</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-feature-types.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>

  <xsd:element name="RouteRegisterAbort">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:routeRegisterReqID"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:annotation>
    <xsd:documentation>There is no positive response to this request.</xsd:documentation>
  </xsd:annotation>

</xsd:schema>
```


18.1.3 Route register cancel

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-route-register-cancel</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-feature-types.xsd" />
  <xsd:include schemaLocation="extension-types.xsd" />

  <xsd:element name="RouteRegisterCancel">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:routeRegisterReqID" />
        <xsd:element ref="csta:extensions" minOccurs="0" />
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="RouteRegisterCancelResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:extensions" minOccurs="0" />
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

18.2 Services

18.2.1 Re-Route

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-re-route</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-feature-types.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>
  <xsd:include schemaLocation="call-control.xsd"/>

  <xsd:element name="ReRoute">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:crossRefIdentifier"/>
        <xsd:element ref="csta:routeRegisterReqID" minOccurs="0"/>
        <xsd:element name="replyTimeout" type="xsd:long" minOccurs="0"/>
        <xsd:element ref="csta:correlatorData" minOccurs="0"/>
        <xsd:element ref="csta:callLinkageData" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:annotation>
    <xsd:documentation>There is no positive response to this request.</xsd:documentation>
  </xsd:annotation>

</xsd:schema>
```

18.2.2 Route end

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-route-end</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="error.xsd"/>
  <xsd:include schemaLocation="device-feature-types.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>

  <xsd:element name="RouteEnd">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:crossRefIdentifier"/>
        <xsd:element ref="csta:routeRegisterReqID" minOccurs="0"/>
        <xsd:element name="errorValue" type="csta:ErrorValue" minOccurs="0"/>
        <xsd:element ref="csta:correlatorData" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:annotation>
    <xsd:documentation>There is no positive response to this request.</xsd:documentation>
  </xsd:annotation>

</xsd:schema>
```

18.2.3 Route reject

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-route-reject</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-feature-types.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>

  <xsd:element name="RouteReject">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:crossRefIdentifier"/>
        <xsd:element ref="csta:routeRegisterReqID" minOccurs="0"/>
        <xsd:element name="rejectCause" type="csta:RejectCause" minOccurs="0"/>
        <xsd:element ref="csta:correlatorData" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:simpleType name="RejectCause">
    <xsd:restriction base="xsd:string">
      <xsd:enumeration value="busyOverflow"/>
      <xsd:enumeration value="queueTimeOverflow"/>
      <xsd:enumeration value="capacityOverflow"/>
      <xsd:enumeration value="calendarOverflow"/>
      <xsd:enumeration value="unknownOverflow"/>
    </xsd:restriction>
  </xsd:simpleType>

  <xsd:annotation>
    <xsd:documentation>There is no positive response to this request.</xsd:documentation>
  </xsd:annotation>

</xsd:schema>
```

18.2.4 Route request

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-route-request</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-identifiers.xsd"/>
  <xsd:include schemaLocation="call-connection-identifiers.xsd"/>
  <xsd:include schemaLocation="device-feature-types.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>
  <xsd:include schemaLocation="call-control.xsd"/>
  <xsd:include schemaLocation="media-services.xsd"/>

  <xsd:element name="RouteRequest">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:crossRefIdentifier"/>
        <xsd:element ref="csta:routeRegisterReqID" minOccurs="0"/>
        <xsd:element name="currentRoute" type="csta:CalledDeviceID"/>
        <xsd:element ref="csta:callingDevice" minOccurs="0"/>
        <xsd:element name="routingDevice" type="csta:SubjectDeviceID" minOccurs="0"/>
        <xsd:element name="routedCall" type="csta:ConnectionID" minOccurs="0"/>
        <xsd:element name="routeSelAlgorithm" type="csta:SelectValue" minOccurs="0"/>
        <xsd:element ref="csta:associatedCallingDevice" minOccurs="0"/>
        <xsd:element ref="csta:associatedCalledDevice" minOccurs="0"/>
        <xsd:element name="priority" type="xsd:boolean" minOccurs="0"/>
        <xsd:element name="replyTimeout" type="xsd:long" minOccurs="0"/>
        <xsd:element ref="csta:correlatorData" minOccurs="0"/>
        <xsd:element ref="csta:mediaCallCharacteristics" minOccurs="0"/>
        <xsd:element ref="csta:callCharacteristics" minOccurs="0"/>
        <xsd:element name="routeCallInfo" type="csta:ConnectionInformation" minOccurs="0"/>
        <xsd:element ref="csta:callLinkageData" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:annotation>
    <xsd:documentation>There is no positive response to this request.</xsd:documentation>
  </xsd:annotation>

</xsd:schema>
```

18.2.5 Route select

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-route-select</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-identifiers.xsd"/>
  <xsd:include schemaLocation="device-feature-types.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>

  <xsd:element name="RouteSelect">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:crossRefIdentifier"/>
        <xsd:element ref="csta:routeRegisterReqID" minOccurs="0"/>
        <xsd:element name="routeSelected" type="csta:DeviceID"/>
        <xsd:element name="alternateRoutes" type="csta:alternateRoutesRecord" minOccurs="0"/>
        <xsd:element name="remainRetries" type="csta:RetryValue" minOccurs="0"/>
        <xsd:element name="routeUsedReq" type="xsd:boolean" minOccurs="0"/>
        <xsd:element ref="csta:correlatorData" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:complexType name="alternateRoutesRecord">
    <xsd:sequence>
      <xsd:element name="deviceID" type="csta:DeviceID" minOccurs="0" maxOccurs="unbounded"/>
    </xsd:sequence>
  </xsd:complexType>

  <xsd:annotation>
    <xsd:documentation>There is no positive response to this request.</xsd:documentation>
  </xsd:annotation>

</xsd:schema>
```

18.2.6 Route used

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-route-used</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-identifiers.xsd"/>
  <xsd:include schemaLocation="device-feature-types.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>
  <xsd:include schemaLocation="call-control.xsd"/>

  <xsd:element name="RouteUsed">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:crossRefIdentifier"/>
        <xsd:element ref="csta:routeRegisterReqID" minOccurs="0"/>
        <xsd:element name="routeUsed" type="csta:CalledDeviceID"/>
        <xsd:element ref="csta:callingDevice" minOccurs="0"/>
        <xsd:element name="domain" type="xsd:boolean" minOccurs="0"/>
        <xsd:element ref="csta:correlatorData" minOccurs="0"/>
        <xsd:element ref="csta:callLinkageData" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:annotation>
    <xsd:documentation>There is no positive response to this request.</xsd:documentation>
  </xsd:annotation>

</xsd:schema>
```

19 Physical device features

19.1 Services

19.1.1 Button press

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-button-press</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-identifiers.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>
  <xsd:include schemaLocation="physical-device-features.xsd"/>

  <xsd:element name="ButtonPress">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="device" type="csta:DeviceID"/>
        <xsd:element name="button" type="csta:ButtonID"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="ButtonPressResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```


19.1.2 Get auditory apparatus information

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-get-auditory-apparatus-information</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-identifiers.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>
  <xsd:include schemaLocation="physical-device-features.xsd"/>

  <xsd:element name="GetAuditoryApparatusInformation">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="device" type="csta:DeviceID"/>
        <xsd:element name="auditoryApparatus" type="csta:AuditoryApparatusID" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="GetAuditoryApparatusInformationResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="auditoryApparatusList" type="csta:AuditoryApparatusList"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

19.1.3 Get button information

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-get-button-information</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-identifiers.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>
  <xsd:include schemaLocation="physical-device-features.xsd"/>

  <xsd:element name="GetButtonInformation">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="device" type="csta:DeviceID"/>
        <xsd:element name="button" type="csta:ButtonID" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="GetButtonInformationResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="buttonList" type="csta:ButtonList"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:complexType name="ButtonList">
    <xsd:sequence>
      <xsd:element name="buttonItem" type="csta:ButtonItem" minOccurs="0" maxOccurs="unbounded"/>
    </xsd:sequence>
  </xsd:complexType>

  <xsd:complexType name="ButtonItem">
    <xsd:sequence>
      <xsd:element name="button" type="csta:ButtonID"/>
      <xsd:element name="buttonLabel" type="xsd:string" minOccurs="0"/>
      <xsd:element name="buttonLabelSettable" type="xsd:boolean" minOccurs="0"/>
      <xsd:element name="buttonFunction" type="xsd:string" minOccurs="0"/>
      <xsd:element name="buttonAssociatedNumber" type="csta:DeviceID" minOccurs="0"/>
      <xsd:element name="buttonAssociatedNumberSettable" type="xsd:boolean" minOccurs="0"/>
      <xsd:element name="buttonPressIndicator" type="xsd:boolean" minOccurs="0"/>
      <xsd:element name="lampList" type="csta:LampIDList" minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>

  <xsd:complexType name="LampIDList">
    <xsd:sequence>
      <xsd:element name="lampID" type="csta:LampID" minOccurs="0" maxOccurs="unbounded"/>
    </xsd:sequence>
  </xsd:complexType>

</xsd:schema>
```

19.1.4 Get display

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-get-display</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-identifiers.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>
  <xsd:include schemaLocation="physical-device-features.xsd"/>

  <xsd:element name="GetDisplay">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="device" type="csta:DeviceID"/>
        <xsd:element name="display" type="csta:DisplayID" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="GetDisplayResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="displayList" type="csta:DisplayList"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:complexType name="DisplayList">
    <xsd:sequence>
      <xsd:element name="displayListItem" type="csta:DisplayListItem" minOccurs="0"
maxOccurs="unbounded"/>
    </xsd:sequence>
  </xsd:complexType>

  <xsd:complexType name="DisplayListItem">
    <xsd:sequence>
      <xsd:element name="displayID" type="csta:DisplayID"/>
      <xsd:element name="logicalRows" type="xsd:long"/>
      <xsd:element name="logicalColumns" type="xsd:long"/>
      <xsd:element name="physicalRows" type="xsd:long" minOccurs="0"/>
      <xsd:element name="physicalColumns" type="xsd:long" minOccurs="0"/>
      <xsd:element name="physicalBaseRowNumber" type="xsd:long" minOccurs="0"/>
      <xsd:element name="physicalBaseColumnNumber" type="xsd:long" minOccurs="0"/>
      <xsd:element name="characterSet" type="csta:CharacterSet" minOccurs="0"/>
      <xsd:element name="contentsOfDisplay" type="xsd:string"/>
    </xsd:sequence>
  </xsd:complexType>

</xsd:schema>
```

19.1.5 Get hookswitch status

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-get-hookswitch-status</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-identifiers.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>
  <xsd:include schemaLocation="physical-device-features.xsd"/>

  <xsd:element name="GetHookswitchStatus">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="device" type="csta:DeviceID"/>
        <xsd:element name="hookswitch" type="csta:HookswitchID" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="GetHookswitchStatusResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="hookswitchStatusList" type="csta:HookswitchStatusList"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:complexType name="HookswitchStatusList">
    <xsd:sequence>
      <xsd:element name="hookswitchStatusItem" type="csta:HookswitchStatusItem" minOccurs="0"
maxOccurs="unbounded"/>
    </xsd:sequence>
  </xsd:complexType>

  <xsd:complexType name="HookswitchStatusItem">
    <xsd:sequence>
      <xsd:element name="hookswitch" type="csta:HookswitchID"/>
      <xsd:element name="hookswitchOnHook" type="xsd:boolean"/>
    </xsd:sequence>
  </xsd:complexType>

</xsd:schema>
```

19.1.6 Get lamp information

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-get-lamp-information</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-identifiers.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>
  <xsd:include schemaLocation="physical-device-features.xsd"/>

  <xsd:element name="GetLampInformation">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="device" type="csta:DeviceID"/>
        <xsd:element name="lamp" type="csta:LampID" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="GetLampInformationResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="lampList" type="csta:LampList"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:complexType name="LampList">
    <xsd:sequence>
      <xsd:element name="lampItem" type="csta:LampItem" minOccurs="0" maxOccurs="unbounded"/>
    </xsd:sequence>
  </xsd:complexType>

  <xsd:complexType name="LampItem">
    <xsd:sequence>
      <xsd:element name="lamp" type="csta:LampID"/>
      <xsd:element name="lampLabel" type="xsd:string" minOccurs="0"/>
      <xsd:element name="button" type="csta:ButtonID" minOccurs="0"/>
      <xsd:element name="lampColor" type="csta:LampColor" minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>

</xsd:schema>
```

19.1.7 Get lamp mode

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-get-lamp-mode</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-identifiers.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>
  <xsd:include schemaLocation="physical-device-features.xsd"/>

  <xsd:element name="GetLampMode">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="device" type="csta:DeviceID"/>
        <xsd:element name="lamp" type="csta:LampID" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="GetLampModeResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="lampModeList" type="csta:LampModeList"/>
        <xsd:element name="lamp" type="csta:LampID" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:complexType name="LampModeList">
    <xsd:sequence>
      <xsd:element name="lampModeItem" type="csta:LampModeItem" minOccurs="0"
maxOccurs="unbounded"/>
    </xsd:sequence>
  </xsd:complexType>

  <xsd:complexType name="LampModeItem">
    <xsd:sequence>
      <xsd:element name="lamp" type="csta:LampID"/>
      <xsd:element name="lampMode" type="csta:LampMode"/>
      <xsd:element name="lampBrightness" type="csta:LampBrightness" minOccurs="0"/>
      <xsd:element name="lampColor" type="csta:LampColor" minOccurs="0"/>
      <xsd:element name="button" type="csta:ButtonID" minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>

</xsd:schema>
```

19.1.8 Get message waiting indicator

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-get-message-waiting-indicator</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-identifiers.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>

  <xsd:element name="GetMessageWaitingIndicator">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="device" type="csta:DeviceID"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="GetMessageWaitingIndicatorResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="messageWaitingOn" type="xsd:boolean"/>
        <xsd:element name="deviceForMessage" type="csta:DeviceID" minOccurs="0"/>
        <xsd:element name="lampIsPresent" type="xsd:boolean" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

19.1.9 Get microphone gain

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-get-microphone-gain</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-identifiers.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>
  <xsd:include schemaLocation="physical-device-features.xsd"/>

  <xsd:element name="GetMicrophoneGain">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="device" type="csta:DeviceID"/>
        <xsd:element name="auditoryApparatus" type="csta:AuditoryApparatusID" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="GetMicrophoneGainResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="microphoneGainList" type="csta:MicrophoneGainList"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:complexType name="MicrophoneGainList">
    <xsd:sequence>
      <xsd:element name="microphoneGainItem" type="csta:MicrophoneGainItem" minOccurs="0"
maxOccurs="unbounded"/>
    </xsd:sequence>
  </xsd:complexType>

  <xsd:complexType name="MicrophoneGainItem">
    <xsd:sequence>
      <xsd:element name="auditoryApparatus" type="csta:AuditoryApparatusID"/>
      <xsd:element name="micGainAbs" type="csta:MicGainAbs" minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>

</xsd:schema>
```


19.1.10 Get microphone mute

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-get-microphone-mute</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-identifiers.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>
  <xsd:include schemaLocation="physical-device-features.xsd"/>

  <xsd:element name="GetMicrophoneMute">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="device" type="csta:DeviceID"/>
        <xsd:element name="auditoryApparatus" type="csta:AuditoryApparatusID" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="GetMicrophoneMuteResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="microphoneMuteList" type="csta:MicrophoneMuteList"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:complexType name="MicrophoneMuteList">
    <xsd:sequence>
      <xsd:element name="microphoneMuteItem" type="csta:MicrophoneMuteItem" minOccurs="0"
maxOccurs="unbounded"/>
    </xsd:sequence>
  </xsd:complexType>

  <xsd:complexType name="MicrophoneMuteItem">
    <xsd:sequence>
      <xsd:element name="auditoryApparatus" type="csta:AuditoryApparatusID"/>
      <xsd:element name="microphoneMuteOn" type="xsd:boolean"/>
    </xsd:sequence>
  </xsd:complexType>

</xsd:schema>
```

19.1.11 Get ringer status

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-get-ringer-status</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-identifiers.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>
  <xsd:include schemaLocation="physical-device-features.xsd"/>

  <xsd:element name="GetRingerStatus">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="device" type="csta:DeviceID"/>
        <xsd:element name="ringer" type="csta:RingerID" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="GetRingerStatusResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="ringerStatusList" type="csta:RingerStatusList"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:complexType name="RingerStatusList">
    <xsd:sequence>
      <xsd:element name="ringerStatusItem" type="csta:RingerStatusItem" minOccurs="0"
maxOccurs="unbounded"/>
    </xsd:sequence>
  </xsd:complexType>

  <xsd:complexType name="RingerStatusItem">
    <xsd:sequence>
      <xsd:element name="ringer" type="csta:RingerID"/>
      <xsd:element name="ringMode" type="csta:RingMode"/>
      <xsd:element name="ringCount" minOccurs="0">
        <xsd:simpleType>
          <xsd:restriction base="xsd:integer">
            <xsd:minInclusive value="0"/>
            <xsd:maxInclusive value="1000"/>
          </xsd:restriction>
        </xsd:simpleType>
      </xsd:element>
      <xsd:element name="ringPattern" type="xsd:long" minOccurs="0"/>
      <xsd:element name="ringVolAbs" minOccurs="0">
        <xsd:simpleType>
          <xsd:restriction base="xsd:integer">
            <xsd:minInclusive value="0"/>
            <xsd:maxInclusive value="100"/>
          </xsd:restriction>
        </xsd:simpleType>
      </xsd:element>
    </xsd:sequence>
  </xsd:complexType>

</xsd:schema>
```

19.1.12 Get speaker mute

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-get-speaker-mute</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-identifiers.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>
  <xsd:include schemaLocation="physical-device-features.xsd"/>

  <xsd:element name="GetSpeakerMute">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="device" type="csta:DeviceID"/>
        <xsd:element name="auditoryApparatus" type="csta:AuditoryApparatusID" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="GetSpeakerMuteResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="speakerMuteList" type="csta:SpeakerMuteList"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:complexType name="SpeakerMuteList">
    <xsd:sequence>
      <xsd:element name="speakerMuteItem" type="csta:SpeakerMuteItem" minOccurs="0"
maxOccurs="unbounded"/>
    </xsd:sequence>
  </xsd:complexType>

  <xsd:complexType name="SpeakerMuteItem">
    <xsd:sequence>
      <xsd:element name="auditoryApparatus" type="csta:AuditoryApparatusID"/>
      <xsd:element name="speakerMuteOn" type="xsd:boolean"/>
    </xsd:sequence>
  </xsd:complexType>

</xsd:schema>
```

19.1.13 Get speaker volume

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-get-speaker-volume</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-identifiers.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>
  <xsd:include schemaLocation="physical-device-features.xsd"/>

  <xsd:element name="GetSpeakerVolume">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="device" type="csta:DeviceID"/>
        <xsd:element name="auditoryApparatus" type="csta:AuditoryApparatusID" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="GetSpeakerVolumeResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="speakerVolumeList" type="csta:SpeakerVolumeList"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:complexType name="SpeakerVolumeList">
    <xsd:sequence>
      <xsd:element name="speakerVolumeItem" type="csta:SpeakerVolumeItem" minOccurs="0"
maxOccurs="unbounded"/>
    </xsd:sequence>
  </xsd:complexType>

  <xsd:complexType name="SpeakerVolumeItem">
    <xsd:sequence>
      <xsd:element name="auditoryApparatus" type="csta:AuditoryApparatusID"/>
      <xsd:element name="speakerVolAbs" type="csta:VolAbs" minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>

</xsd:schema>
```

19.1.14 Set button information

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-set-button-information</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-identifiers.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>
  <xsd:include schemaLocation="physical-device-features.xsd"/>

  <xsd:element name="SetButtonInformation">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="device" type="csta:DeviceID"/>
        <xsd:element name="button" type="csta:ButtonID"/>
        <xsd:element name="buttonLabel" minOccurs="0">
          <xsd:simpleType>
            <xsd:restriction base="xsd:string">
              <xsd:minLength value="0"/>
              <xsd:maxLength value="64"/>
            </xsd:restriction>
          </xsd:simpleType>
        </xsd:element>
        <xsd:element name="buttonAssociatedNumber" type="csta:DeviceID" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="SetButtonInformationResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

19.1.15 Set display

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-set-display</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-identifiers.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>
  <xsd:include schemaLocation="physical-device-features.xsd"/>

  <xsd:element name="SetDisplay">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="device" type="csta:DeviceID"/>
        <xsd:element name="display" type="csta:DisplayID" minOccurs="0"/>
        <xsd:element name="physicalBaseRowNumber" type="xsd:long" minOccurs="0"/>
        <xsd:element name="physicalBaseColumnNumber" type="xsd:long" minOccurs="0"/>
        <xsd:element name="contentsOfDisplay">
          <xsd:simpleType>
            <xsd:restriction base="xsd:string">
              <xsd:minLength value="0"/>
              <xsd:maxLength value="240"/>
            </xsd:restriction>
          </xsd:simpleType>
        </xsd:element>
        <xsd:element name="offset" type="xsd:long" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="SetDisplayResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

19.1.16 Set hookswitch status

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-set-hookswitch-status</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-identifiers.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>
  <xsd:include schemaLocation="physical-device-features.xsd"/>

  <xsd:element name="SetHookswitchStatus">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="device" type="csta:DeviceID"/>
        <xsd:element name="hookswitch" type="csta:HookswitchID"/>
        <xsd:element name="hookswitchOnhook" type="xsd:boolean"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="SetHookswitchStatusResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

19.1.17 Set lamp mode

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-set-lamp-mode</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-identifiers.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>
  <xsd:include schemaLocation="physical-device-features.xsd"/>

  <xsd:element name="SetLampMode">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="device" type="csta:DeviceID"/>
        <xsd:element name="lamp" type="csta:LampID"/>
        <xsd:element name="LampMode" type="csta:LampMode"/>
        <xsd:element name="lampBrightness" type="csta:LampBrightness" minOccurs="0"/>
        <xsd:element name="lampColor" type="csta:LampColor" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="SetLampModeResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```


19.1.18 Set message waiting indicator

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-set-message-waiting-indicator</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-identifiers.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>

  <xsd:element name="SetMessageWaitingIndicator">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="device" type="csta:DeviceID"/>
        <xsd:element name="messageWaitingOn" type="xsd:boolean"/>
        <xsd:element name="deviceForMessage" type="csta:DeviceID" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="SetMessageWaitingIndicatorResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

19.1.19 Set microphone gain

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-set-microphone-gain</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-identifiers.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>
  <xsd:include schemaLocation="physical-device-features.xsd"/>

  <xsd:element name="SetMicrophoneGain">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="device" type="csta:DeviceID"/>
        <xsd:element name="auditoryApparatus" type="csta:AuditoryApparatusID"/>
        <xsd:element name="microphoneGain" type="csta:MicrophoneGain"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="SetMicrophoneGainResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

19.1.20 Set microphone mute

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-set-microphone-mute</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-identifiers.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>
  <xsd:include schemaLocation="physical-device-features.xsd"/>

  <xsd:element name="SetMicrophoneMute">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="device" type="csta:DeviceID"/>
        <xsd:element name="auditoryApparatus" type="csta:AuditoryApparatusID"/>
        <xsd:element name="microphoneMuteOn" type="xsd:boolean"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="SetMicrophoneMuteResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

19.1.21 Set ringer status

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-set-ringer-status</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-identifiers.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>
  <xsd:include schemaLocation="physical-device-features.xsd"/>

  <xsd:element name="SetRingerStatus">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="device" type="csta:DeviceID"/>
        <xsd:element name="ringer" type="csta:RingerID"/>
        <xsd:element name="ringMode" type="csta:RingMode" minOccurs="0"/>
        <xsd:element name="ringPattern" type="xsd:long" minOccurs="0"/>
        <xsd:element name="ringVolume" type="csta:Volume" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="SetRingerStatusResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

19.1.22 Set speaker mute

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-set-speaker-mute</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-identifiers.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>
  <xsd:include schemaLocation="physical-device-features.xsd"/>

  <xsd:element name="SetSpeakerMute">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="device" type="csta:DeviceID"/>
        <xsd:element name="auditoryApparatus" type="csta:AuditoryApparatusID"/>
        <xsd:element name="speakerMuteOn" type="xsd:boolean"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="SetSpeakerMuteResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

19.1.23 Set speaker volume

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-set-speaker-volume</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-identifiers.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>
  <xsd:include schemaLocation="physical-device-features.xsd"/>

  <xsd:element name="SetSpeakerVolume">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="device" type="csta:DeviceID"/>
        <xsd:element name="auditoryApparatus" type="csta:AuditoryApparatusID"/>
        <xsd:element name="speakerVolume" type="csta:Volume"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="SetSpeakerVolumeResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

19.2 Events

19.2.1 Button information

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-button-information-event</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-identifiers.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>
  <xsd:include schemaLocation="physical-device-features.xsd"/>
  <xsd:include schemaLocation="status-reporting.xsd"/>

  <xsd:element name="ButtonInformationEvent">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:monitorCrossRefID"/>
        <xsd:element name="device" type="csta:SubjectDeviceID"/>
        <xsd:element name="button" type="csta:ButtonID"/>
        <xsd:element name="buttonLabel" minOccurs="0">
          <xsd:simpleType>
            <xsd:restriction base="xsd:string">
              <xsd:length value="0"/>
              <xsd:minLength value="64"/>
            </xsd:restriction>
          </xsd:simpleType>
        </xsd:element>
        <xsd:element name="buttonAssociatedNumber" type="csta:DeviceID" minOccurs="0"/>
        <xsd:element name="buttonPressIndicator" type="xsd:boolean" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

19.2.2 Button press

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-button-press-event</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-identifiers.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>
  <xsd:include schemaLocation="physical-device-features.xsd"/>
  <xsd:include schemaLocation="status-reporting.xsd"/>

  <xsd:element name="ButtonPressEvent">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:monitorCrossRefID"/>
        <xsd:element name="device" type="csta:SubjectDeviceID"/>
        <xsd:element name="button" type="csta:ButtonID"/>
        <xsd:element name="buttonLabel" minOccurs="0">
          <xsd:simpleType>
            <xsd:restriction base="xsd:string">
              <xsd:minLength value="0"/>
              <xsd:maxLength value="64"/>
            </xsd:restriction>
          </xsd:simpleType>
        </xsd:element>
        <xsd:element name="buttonAssociatedNumber" type="csta:DeviceID" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```


19.2.3 Display updated

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-display-updated-event</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-identifiers.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>
  <xsd:include schemaLocation="physical-device-features.xsd"/>
  <xsd:include schemaLocation="status-reporting.xsd"/>

  <xsd:element name="DisplayUpdatedEvent">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:monitorCrossRefID"/>
        <xsd:element name="device" type="csta:SubjectDeviceID"/>
        <xsd:element name="displayID" type="csta:DisplayID" minOccurs="0"/>
        <xsd:element name="logicalRows" type="xsd:long"/>
        <xsd:element name="logicalColumns" type="xsd:long"/>
        <xsd:element name="physicalRows" type="xsd:long" minOccurs="0"/>
        <xsd:element name="physicalColumns" type="xsd:long" minOccurs="0"/>
        <xsd:element name="physicalBaseRowNumber" type="xsd:long" minOccurs="0"/>
        <xsd:element name="physicalBaseColumnNumber" type="xsd:long" minOccurs="0"/>
        <xsd:element name="characterSet" type="csta:CharacterSet" minOccurs="0"/>
        <xsd:element name="contentsOfDisplay" type="xsd:string"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

19.2.4 Hookswitch

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-hookswitch-event</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-identifiers.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>
  <xsd:include schemaLocation="physical-device-features.xsd"/>
  <xsd:include schemaLocation="status-reporting.xsd"/>

  <xsd:element name="HookswitchEvent">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:monitorCrossRefID"/>
        <xsd:element name="device" type="csta:SubjectDeviceID"/>
        <xsd:element name="hookswitch" type="csta:HookswitchID"/>
        <xsd:element name="hookswitchOnHook" type="xsd:boolean"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

19.2.5 Lamp mode

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-lamp-mode-event</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-identifiers.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>
  <xsd:include schemaLocation="physical-device-features.xsd"/>
  <xsd:include schemaLocation="status-reporting.xsd"/>

  <xsd:element name="LampModeEvent">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:monitorCrossRefID"/>
        <xsd:element name="device" type="csta:SubjectDeviceID"/>
        <xsd:element name="lamp" type="csta:LampID"/>
        <xsd:element name="lampLabel" minOccurs="0">
          <xsd:simpleType>
            <xsd:restriction base="xsd:string">
              <xsd:minLength value="0"/>
              <xsd:maxLength value="64"/>
            </xsd:restriction>
          </xsd:simpleType>
        </xsd:element>
        <xsd:element name="lampMode" type="csta:LampMode"/>
        <xsd:element name="lampBrightness" type="csta:LampBrightness" minOccurs="0"/>
        <xsd:element name="lampColor" type="csta:LampColor" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

19.2.6 Message waiting

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-message-waiting-event</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-identifiers.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>
  <xsd:include schemaLocation="status-reporting.xsd"/>

  <xsd:element name="MessageWaitingEvent">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:monitorCrossRefID"/>
        <xsd:element name="targetDevice" type="csta:SubjectDeviceID"/>
        <xsd:element name="deviceForMessage" type="csta:DeviceID" minOccurs="0"/>
        <xsd:element name="messageWaitingOn" type="xsd:boolean"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

19.2.7 Microphone gain

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-microphone-gain-event</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-identifiers.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>
  <xsd:include schemaLocation="physical-device-features.xsd"/>
  <xsd:include schemaLocation="status-reporting.xsd"/>

  <xsd:element name="MicrophoneGainEvent">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:monitorCrossRefID"/>
        <xsd:element name="invokingDevice" type="csta:SubjectDeviceID"/>
        <xsd:element name="auditoryApparatus" type="csta:AuditoryApparatusID"/>
        <xsd:element name="microphoneGain" type="csta:MicrophoneGain"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

19.2.8 Microphone mute

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-microphone-mute-event</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-identifiers.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>
  <xsd:include schemaLocation="physical-device-features.xsd"/>
  <xsd:include schemaLocation="status-reporting.xsd"/>

  <xsd:element name="MicrophoneMuteEvent">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:monitorCrossRefID"/>
        <xsd:element name="invokingDevice" type="csta:SubjectDeviceID"/>
        <xsd:element name="auditoryApparatus" type="csta:AuditoryApparatusID"/>
        <xsd:element name="microphoneMuteOn" type="xsd:boolean"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

19.2.9 Ringer status

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-ringer-status-event</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-identifiers.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>
  <xsd:include schemaLocation="physical-device-features.xsd"/>
  <xsd:include schemaLocation="status-reporting.xsd"/>

  <xsd:element name="RingerStatusEvent">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:monitorCrossRefID"/>
        <xsd:element name="device" type="csta:SubjectDeviceID"/>
        <xsd:element name="ringer" type="csta:RingerID"/>
        <xsd:element name="ringMode" type="csta:RingMode"/>
        <xsd:element name="ringCount" minOccurs="0">
          <xsd:simpleType>
            <xsd:restriction base="xsd:integer">
              <xsd:minInclusive value="0"/>
              <xsd:maxInclusive value="1000"/>
            </xsd:restriction>
          </xsd:simpleType>
        </xsd:element>
        <xsd:element name="ringPattern" type="xsd:long" minOccurs="0"/>
        <xsd:element name="ringVolume" type="csta:Volume" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

19.2.10 Speaker mute

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-speaker-mute-event</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-identifiers.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>
  <xsd:include schemaLocation="physical-device-features.xsd"/>
  <xsd:include schemaLocation="status-reporting.xsd"/>

  <xsd:element name="SpeakerMuteEvent">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:monitorCrossRefID"/>
        <xsd:element name="invokingDevice" type="csta:SubjectDeviceID"/>
        <xsd:element name="auditoryApparatus" type="csta:AuditoryApparatusID"/>
        <xsd:element name="speakerMuteOn" type="xsd:boolean"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```


19.2.11 Speaker volume

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-speaker-volume-event</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-identifiers.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>
  <xsd:include schemaLocation="physical-device-features.xsd"/>
  <xsd:include schemaLocation="status-reporting.xsd"/>

  <xsd:element name="SpeakerVolumeEvent">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:monitorCrossRefID"/>
        <xsd:element name="invokingDevice" type="csta:SubjectDeviceID"/>
        <xsd:element name="auditoryApparatus" type="csta:AuditoryApparatusID"/>
        <xsd:element name="speakerVolume" type="csta:Volume"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

20 Logical device features

20.1 Services

20.1.1 Call back non-call-related

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-call-back-message-non-call-related</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-identifiers.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>

  <xsd:element name="CallBackMessageNonCallRelated">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="originatingDevice" type="csta:DeviceID"/>
        <xsd:element name="targetDevice" type="csta:DeviceID"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="CallBackMessageNonCallRelatedResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

20.1.2 Call back message non-call-related

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-call-back-non-call-related</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-identifiers.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>

  <xsd:element name="CallBackNonCallRelated">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="originatingDevice" type="csta:DeviceID"/>
        <xsd:element name="targetDevice" type="csta:DeviceID"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="CallBackNonCallRelatedResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

20.1.3 Cancel call back

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-cancel-call-back</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-identifiers.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>

  <xsd:element name="CancelCallBack">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="originatingDevice" type="csta:DeviceID"/>
        <xsd:element name="targetDevice" type="csta:DeviceID"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="CancelCallBackResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

20.1.4 Cancel call back message

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-cancel-call-back-message</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-identifiers.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>

  <xsd:element name="CancelCallBackMessage">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="originatingDevice" type="csta:DeviceID"/>
        <xsd:element name="targetDevice" type="csta:DeviceID"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="CancelCallBackMessageResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

20.1.5 Get agent state

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-get-agent-state</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-identifiers.xsd"/>
  <xsd:include schemaLocation="device-feature-types.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>

  <xsd:element name="GetAgentState">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="device" type="csta:DeviceID"/>
        <xsd:element name="acdGroup" type="csta:DeviceID" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="GetAgentStateResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="agentStateList" type="csta:AgentStateList"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:complexType name="AgentStateList">
    <xsd:sequence>
      <xsd:element name="agentStateEntry" type="csta:AgentStateEntry" maxOccurs="32"/>
    </xsd:sequence>
  </xsd:complexType>

  <xsd:complexType name="AgentStateEntry">
    <xsd:sequence>
      <xsd:element name="agentID" type="csta:AgentID" minOccurs="0"/>
      <xsd:element name="loggedOnState" type="xsd:boolean"/>
      <xsd:element name="agentInfo" minOccurs="0">
        <xsd:complexType>
          <xsd:sequence>
            <xsd:element name="agentInfoItem" type="csta:AgentInfo" minOccurs="0"
maxOccurs="unbounded"/>
          </xsd:sequence>
        </xsd:complexType>
      </xsd:element>
    </xsd:sequence>
  </xsd:complexType>

  <xsd:complexType name="AgentInfo">
    <xsd:sequence>
      <xsd:element name="acdGroup" type="csta:DeviceID" minOccurs="0"/>
      <xsd:element name="agentState" type="csta:AgentState"/>
      <xsd:element name="pendingAgentState" type="csta:PendingAgentState" minOccurs="0"/>
      <xsd:element name="agentStateCondition" type="csta:AgentStateCondition" minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>

  <xsd:simpleType name="AgentStateCondition">
    <xsd:restriction base="xsd:string">
```

```
        <xsd:enumeration value="forcedPause"/>
        <xsd:enumeration value="other"/>
    </xsd:restriction>
</xsd:simpleType>

</xsd:schema>
```

20.1.6 Get auto answer

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-get-auto-answer</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-identifiers.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>

  <xsd:element name="GetAutoAnswer">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="device" type="csta:DeviceID"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="GetAutoAnswerResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="autoAnswerOn" type="xsd:boolean"/>
        <xsd:element name="numberOfRings" type="xsd:long" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```


20.1.7 Get auto work mode

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-get-auto-work-mode</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-identifiers.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>

  <xsd:element name="GetAutoWorkMode">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="device" type="csta:DeviceID"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="GetAutoWorkModeResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="autoWorkOn" type="xsd:boolean"/>
        <xsd:element name="autoWorkInterval" type="xsd:long" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

20.1.8 Get caller id status

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-get-caller-id-status</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-identifiers.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>

  <xsd:element name="GetCallerIDStatus">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="device" type="csta:DeviceID"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="GetCallerIDStatusResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="callerIDProvided" type="xsd:boolean"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

20.1.9 Get do not disturb

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-get-do-not-disturb</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-identifiers.xsd"/>
  <xsd:include schemaLocation="device-feature-types.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>

  <xsd:element name="GetDoNotDisturb">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="device" type="csta:DeviceID"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="GetDoNotDisturbResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="doNotDisturbOn" type="xsd:boolean"/>
        <xsd:element name="callOrigination" type="csta:CallOrigination" minOccurs="0"/>
        <xsd:element name="callingDeviceList" minOccurs="0">
          <xsd:complexType>
            <xsd:sequence>
              <xsd:element name="callingDevice" type="csta:DeviceID" minOccurs="0"
maxOccurs="unbounded"/>
            </xsd:sequence>
          </xsd:complexType>
        </xsd:element>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

20.1.10 Get forwarding

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-get-forwarding</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-identifiers.xsd"/>
  <xsd:include schemaLocation="device-feature-types.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>

  <xsd:element name="GetForwarding">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="device" type="csta:DeviceID"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="GetForwardingResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="forwardingList" type="csta:ForwardList"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

20.1.11 Get last number dialed

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-get-last-number-dialed</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-identifiers.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>

  <xsd:element name="GetLastNumberDialed">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="device" type="csta:DeviceID"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="GetLastNumberDialedResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="numberDialed" type="csta:DeviceID"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

20.1.12 Get routing mode

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-get-routing-mode</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-identifiers.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>

  <xsd:element name="GetRoutingMode">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="device" type="csta:DeviceID"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="GetRoutingModeResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="routingMode" type="xsd:boolean"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

20.1.13 Set agent state

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-set-agent-state</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-identifiers.xsd"/>
  <xsd:include schemaLocation="device-feature-types.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>

  <xsd:element name="SetAgentState">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="device" type="csta:DeviceID"/>
        <xsd:element name="requestedAgentState" type="csta:ReqAgentState"/>
        <xsd:element name="agentID" type="csta:AgentID" minOccurs="0"/>
        <xsd:element name="password" type="csta:AgentPassword" minOccurs="0"/>
        <xsd:element name="group" type="csta:DeviceID" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="SetAgentStateResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="pendingAgentState" type="csta:PendingAgentState" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:simpleType name="ReqAgentState">
    <xsd:restriction base="xsd:string">
      <xsd:enumeration value="loggedOn"/>
      <xsd:enumeration value="loggedOff"/>
      <xsd:enumeration value="notReady"/>
      <xsd:enumeration value="ready"/>
      <xsd:enumeration value="workingAfterCall"/>
    </xsd:restriction>
  </xsd:simpleType>

</xsd:schema>
```

20.1.14 Set auto answer

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-set-auto-answer</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-identifiers.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>

  <xsd:element name="SetAutoAnswer">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="device" type="csta:DeviceID"/>
        <xsd:element name="autoAnswerOn" type="xsd:boolean"/>
        <xsd:element name="numberOfRings" type="xsd:long" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="SetAutoAnswerResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```


20.1.15 Set auto work mode

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-set-auto-work-mode</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-identifiers.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>

  <xsd:element name="SetAutoWorkMode">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="device" type="csta:DeviceID"/>
        <xsd:element name="autoWorkOn" type="xsd:boolean"/>
        <xsd:element name="autoWorkInterval" type="xsd:long" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="SetAutoWorkModeResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

20.1.16 Set caller id status

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-set-caller-id-status</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-identifiers.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>

  <xsd:element name="SetCallerIDStatus">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="device" type="csta:DeviceID"/>
        <xsd:element name="callerIDProvided" type="xsd:boolean"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="SetCallerIDStatusResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

20.1.17 Set do not disturb

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-set-do-not-disturb</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-identifiers.xsd"/>
  <xsd:include schemaLocation="device-feature-types.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>

  <xsd:element name="SetDoNotDisturb">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="device" type="csta:DeviceID"/>
        <xsd:element name="doNotDisturbOn" type="xsd:boolean"/>
        <xsd:element name="callOrigination" type="csta:CallOrigination" minOccurs="0"/>
        <xsd:element name="callingDeviceList" minOccurs="0">
          <xsd:complexType>
            <xsd:sequence>
              <xsd:element name="callingDevice" type="csta:DeviceID" minOccurs="0"
maxOccurs="unbounded"/>
            </xsd:sequence>
          </xsd:complexType>
        </xsd:element>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>
  <xsd:element name="SetDoNotDisturbResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>
</xsd:schema>
```

20.1.18 Set forwarding

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-set-forwarding</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-identifiers.xsd"/>
  <xsd:include schemaLocation="device-feature-types.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>

  <xsd:element name="SetForwarding">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="device" type="csta:DeviceID"/>
        <xsd:element name="forwardingType" type="csta:ForwardingType" minOccurs="0"/>
        <xsd:element name="activateForward" type="xsd:boolean"/>
        <xsd:element name="forwardDN" type="csta:DeviceID" minOccurs="0"/>
        <xsd:element name="ringCount" minOccurs="0">
          <xsd:simpleType>
            <xsd:restriction base="xsd:long">
              <xsd:minInclusive value="1"/>
              <xsd:maxInclusive value="100"/>
            </xsd:restriction>
          </xsd:simpleType>
        </xsd:element>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="SetForwardingResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

20.1.19 Set routing mode

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-set-forwarding</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-identifiers.xsd"/>
  <xsd:include schemaLocation="device-feature-types.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>

  <xsd:element name="SetForwarding">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="device" type="csta:DeviceID"/>
        <xsd:element name="forwardingType" type="csta:ForwardingType" minOccurs="0"/>
        <xsd:element name="activateForward" type="xsd:boolean"/>
        <xsd:element name="forwardDN" type="csta:DeviceID" minOccurs="0"/>
        <xsd:element name="ringCount" minOccurs="0">
          <xsd:simpleType>
            <xsd:restriction base="xsd:long">
              <xsd:minInclusive value="1"/>
              <xsd:maxInclusive value="100"/>
            </xsd:restriction>
          </xsd:simpleType>
        </xsd:element>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="SetForwardingResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

20.2 Events

20.2.1 Agent busy

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-agent-busy-event</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="event-cause.xsd"/>
  <xsd:include schemaLocation="device-identifiers.xsd"/>
  <xsd:include schemaLocation="status-reporting.xsd"/>
  <xsd:include schemaLocation="device-feature-types.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>

  <xsd:element name="AgentBusyEvent">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:monitorCrossRefID"/>
        <xsd:element name="agentDevice" type="csta:SubjectDeviceID"/>
        <xsd:element name="agentID" type="csta:AgentID" minOccurs="0"/>
        <xsd:element name="acdGroup" type="csta:DeviceID" minOccurs="0"/>
        <xsd:element name="pendingAgentState" type="csta:PendingAgentState" minOccurs="0"/>
        <xsd:element ref="csta:cause" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

20.2.2 Agent logged off

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-agent-logged-off-event</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="event-cause.xsd"/>
  <xsd:include schemaLocation="device-identifiers.xsd"/>
  <xsd:include schemaLocation="status-reporting.xsd"/>
  <xsd:include schemaLocation="device-feature-types.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>

  <xsd:element name="AgentLoggedOffEvent">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:monitorCrossRefID"/>
        <xsd:element name="agentDevice" type="csta:SubjectDeviceID"/>
        <xsd:element name="agentID" type="csta:AgentID" minOccurs="0"/>
        <xsd:element name="acdGroup" type="csta:DeviceID" minOccurs="0"/>
        <xsd:element name="agentPassword" type="csta:AgentPassword" minOccurs="0"/>
        <xsd:element ref="csta:cause" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

20.2.3 Agent logged on

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-agent-logged-on-event</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="event-cause.xsd" />
  <xsd:include schemaLocation="device-identifiers.xsd" />
  <xsd:include schemaLocation="status-reporting.xsd" />
  <xsd:include schemaLocation="device-feature-types.xsd" />
  <xsd:include schemaLocation="extension-types.xsd" />

  <xsd:element name="AgentLoggedOnEvent">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:monitorCrossRefID" />
        <xsd:element name="agentDevice" type="csta:SubjectDeviceID" />
        <xsd:element name="agentID" type="csta:AgentID" minOccurs="0" />
        <xsd:element name="acdGroup" type="csta:DeviceID" minOccurs="0" />
        <xsd:element name="agentPassword" type="csta:AgentPassword" minOccurs="0" />
        <xsd:element ref="csta:cause" minOccurs="0" />
        <xsd:element ref="csta:extensions" minOccurs="0" />
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```


20.2.4 Agent not ready

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-agent-not-ready-event</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="event-cause.xsd"/>
  <xsd:include schemaLocation="device-identifiers.xsd"/>
  <xsd:include schemaLocation="status-reporting.xsd"/>
  <xsd:include schemaLocation="device-feature-types.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>

  <xsd:element name="AgentNotReadyEvent">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:monitorCrossRefID"/>
        <xsd:element name="agentDevice" type="csta:SubjectDeviceID"/>
        <xsd:element name="agentID" type="csta:AgentID" minOccurs="0"/>
        <xsd:element name="acdGroup" type="csta:DeviceID" minOccurs="0"/>
        <xsd:element ref="csta:cause" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

20.2.5 Agent ready

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-agent-ready-event</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="event-cause.xsd" />
  <xsd:include schemaLocation="device-identifiers.xsd" />
  <xsd:include schemaLocation="status-reporting.xsd" />
  <xsd:include schemaLocation="device-feature-types.xsd" />
  <xsd:include schemaLocation="extension-types.xsd" />

  <xsd:element name="AgentReadyEvent">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:monitorCrossRefID" />
        <xsd:element name="agentDevice" type="csta:SubjectDeviceID" />
        <xsd:element name="agentID" type="csta:AgentID" minOccurs="0" />
        <xsd:element name="acdGroup" type="csta:DeviceID" minOccurs="0" />
        <xsd:element ref="csta:cause" minOccurs="0" />
        <xsd:element ref="csta:extensions" minOccurs="0" />
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

20.2.6 Agent working after call

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-agent-working-after-call-event</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="event-cause.xsd"/>
  <xsd:include schemaLocation="device-identifiers.xsd"/>
  <xsd:include schemaLocation="status-reporting.xsd"/>
  <xsd:include schemaLocation="device-feature-types.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>

  <xsd:element name="AgentWorkingAfterCallEvent">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:monitorCrossRefID"/>
        <xsd:element name="agentDevice" type="csta:SubjectDeviceID"/>
        <xsd:element name="agentID" type="csta:AgentID" minOccurs="0"/>
        <xsd:element name="acdGroup" type="csta:DeviceID" minOccurs="0"/>
        <xsd:element name="pendingAgentState" minOccurs="0">
          <xsd:simpleType>
            <xsd:restriction base="xsd:string">
              <xsd:enumeration value="notReady"/>
              <xsd:enumeration value="ready"/>
              <xsd:enumeration value="null"/>
            </xsd:restriction>
          </xsd:simpleType>
        </xsd:element>
        <xsd:element ref="csta:cause" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

20.2.7 Auto answer

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-auto-answer-event</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-identifiers.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>
  <xsd:include schemaLocation="status-reporting.xsd"/>

  <xsd:element name="AutoAnswerEvent">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:monitorCrossRefID"/>
        <xsd:element name="invokingDevice" type="csta:SubjectDeviceID"/>
        <xsd:element name="autoAnswerOn" type="xsd:boolean"/>
        <xsd:element name="numberOfRings" type="xsd:long" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

20.2.8 Auto work mode

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-auto-work-mode-event</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-identifiers.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>
  <xsd:include schemaLocation="status-reporting.xsd"/>

  <xsd:element name="AutoWorkModeEvent">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:monitorCrossRefID"/>
        <xsd:element name="invokingDevice" type="csta:SubjectDeviceID"/>
        <xsd:element name="autoWorkOn" type="xsd:boolean"/>
        <xsd:element name="autoWorkInterval" type="xsd:long"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

20.2.9 Call back

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-call-back-event</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-identifiers.xsd"/>
  <xsd:include schemaLocation="status-reporting.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>

  <xsd:element name="CallBackEvent">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:monitorCrossRefID"/>
        <xsd:element name="originatingDevice" type="csta:SubjectDeviceID"/>
        <xsd:element name="targetDevice" type="csta:SubjectDeviceID"/>
        <xsd:element name="callBackSetCanceled" type="xsd:boolean"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

20.2.10 Call back message

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-call-back-message-event</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-identifiers.xsd"/>
  <xsd:include schemaLocation="status-reporting.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>

  <xsd:element name="CallBackMessageEvent">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:monitorCrossRefID"/>
        <xsd:element name="originatingDevice" type="csta:SubjectDeviceID"/>
        <xsd:element name="targetDevice" type="csta:SubjectDeviceID"/>
        <xsd:element name="callBackMsgSetCanceled" type="xsd:boolean"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

20.2.11 Caller id status

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-caller-id-status-event</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-identifiers.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>
  <xsd:include schemaLocation="status-reporting.xsd"/>

  <xsd:element name="CallerIDStatusEvent">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:monitorCrossRefID"/>
        <xsd:element name="device" type="csta:DeviceID"/>
        <xsd:element name="callerIDProvided" type="xsd:boolean"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```


20.2.12 Do not disturb

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-do-not-disturb-event</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-identifiers.xsd"/>
  <xsd:include schemaLocation="status-reporting.xsd"/>
  <xsd:include schemaLocation="device-feature-types.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>

  <xsd:element name="DoNotDisturbEvent">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:monitorCrossRefID"/>
        <xsd:element name="device" type="csta:SubjectDeviceID"/>
        <xsd:element name="doNotDisturbOn" type="xsd:boolean"/>
        <xsd:element name="callOrigination" type="csta:CallOrigination" minOccurs="0"/>
        <xsd:element name="callingDeviceList" minOccurs="0">
          <xsd:complexType>
            <xsd:sequence>
              <xsd:element name="callingDevice" type="csta:DeviceID" minOccurs="0"
                maxOccurs="unbounded"/>
            </xsd:sequence>
          </xsd:complexType>
        </xsd:element>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>
</xsd:schema>
```

20.2.13 Forwarding

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-forwarding-event</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-identifiers.xsd"/>
  <xsd:include schemaLocation="status-reporting.xsd"/>
  <xsd:include schemaLocation="device-feature-types.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>

  <xsd:element name="ForwardingEvent">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:monitorCrossRefID"/>
        <xsd:element name="device" type="csta:SubjectDeviceID"/>
        <xsd:element name="forwardingType" type="csta:ForwardingType" minOccurs="0"/>
        <xsd:element name="forwardStatus" type="xsd:boolean"/>
        <xsd:element name="forwardTo" type="csta:DeviceID" minOccurs="0"/>
        <xsd:element name="forwardDefault" type="csta:ForwardDefault" minOccurs="0"/>
        <xsd:element name="ringCount" minOccurs="0">
          <xsd:simpleType>
            <xsd:restriction base="xsd:short">
              <xsd:minInclusive value="1"/>
              <xsd:maxInclusive value="100"/>
            </xsd:restriction>
          </xsd:simpleType>
        </xsd:element>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

20.2.14 Routing mode

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-routing-event</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-identifiers.xsd"/>
  <xsd:include schemaLocation="status-reporting.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>

  <xsd:element name="RoutingEvent">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:monitorCrossRefID"/>
        <xsd:element name="device" type="csta:SubjectDeviceID"/>
        <xsd:element name="routingMode" type="xsd:boolean"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

21 Device maintenance events

21.1 Events

21.1.1 Back in service

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-back-in-service-event</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="event-cause.xsd"/>
  <xsd:include schemaLocation="device-identifiers.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>
  <xsd:include schemaLocation="status-reporting.xsd"/>

  <xsd:element name="BackInServiceEvent">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:monitorCrossRefID"/>
        <xsd:element name="device" type="csta:SubjectDeviceID"/>
        <xsd:element ref="csta:cause" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

21.1.2 Device capabilities changed

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-device-capabilities-changed-event</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="event-cause.xsd"/>
  <xsd:include schemaLocation="device-identifiers.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>
  <xsd:include schemaLocation="status-reporting.xsd"/>

  <xsd:element name="DeviceCapsChangedEvent">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:monitorCrossRefID"/>
        <xsd:element name="device" type="csta:SubjectDeviceID"/>
        <xsd:element ref="csta:cause" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

21.1.3 Out of service

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-out-of-service-event</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="event-cause.xsd" />
  <xsd:include schemaLocation="device-identifiers.xsd" />
  <xsd:include schemaLocation="extension-types.xsd" />
  <xsd:include schemaLocation="status-reporting.xsd" />

  <xsd:element name="OutOfServiceEvent">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:monitorCrossRefID" />
        <xsd:element name="device" type="csta:SubjectDeviceID" />
        <xsd:element ref="csta:cause" minOccurs="0" />
        <xsd:element ref="csta:extensions" minOccurs="0" />
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

22 I/O services

22.1 Registration services

22.1.1 I/O register

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-io-register</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-identifiers.xsd"/>
  <xsd:include schemaLocation="device-feature-types.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>

  <xsd:element name="IORegister">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="ioDevice" type="csta:DeviceID"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="IORegisterResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="ioRegisterRequestID" type="csta:IORegisterReqID"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

22.1.2 I/O register abort

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-io-register-abort</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-feature-types.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>

  <xsd:element name="IORegisterAbort">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="ioRegisterRequestID" type="csta:IORegisterReqID"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:annotation>
    <xsd:documentation>There is no positive response to this request.</xsd:documentation>
  </xsd:annotation>

</xsd:schema>
```


22.1.3 I/O register cancel

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-io-register-cancel</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-feature-types.xsd" />
  <xsd:include schemaLocation="extension-types.xsd" />

  <xsd:element name="IORegisterCancel">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="ioRegisterRequestID" type="csta:IORegisterReqID" />
        <xsd:element ref="csta:extensions" minOccurs="0" />
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="IORegisterCancelResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:extensions" minOccurs="0" />
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

22.2 Services

22.2.1 Data path resumed

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-data-path-resumed</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-feature-types.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>

  <xsd:element name="DataPathResumed">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="ioCrossRefID" type="csta:IOCrossRefID"/>
        <xsd:element name="ioRegisterRequestID" type="csta:IORegisterReqID" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="DataPathResumedResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

22.2.2 Data path suspended

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-data-path-suspended</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-feature-types.xsd" />
  <xsd:include schemaLocation="extension-types.xsd" />

  <xsd:element name="DataPathSuspended">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="ioCrossRefID" type="csta:IOCrossRefID" />
        <xsd:element name="ioRegisterRequestID" type="csta:IORegisterReqID" minOccurs="0" />
        <xsd:element ref="csta:extensions" minOccurs="0" />
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="DataPathSuspendedResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:extensions" minOccurs="0" />
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

22.2.3 Fast data

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-fast-data</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="switching-function-objects.xsd" />
  <xsd:include schemaLocation="device-feature-types.xsd" />
  <xsd:include schemaLocation="extension-types.xsd" />
  <xsd:include schemaLocation="physical-device-features.xsd" />

  <xsd:element name="FastData">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="ioRegisterReqID" type="csta:IORegisterReqID" minOccurs="0"/>
        <xsd:element name="object" type="csta:CSTAObject" />
        <xsd:element name="dataPathType" type="csta:DataPathType" minOccurs="0"/>
        <xsd:element name="displayAttribute" type="csta:DisplayAttribute" minOccurs="0"/>
        <xsd:element name="ioData">
          <xsd:simpleType>
            <xsd:restriction base="xsd:hexBinary">
              <xsd:minLength value="0"/>
              <xsd:maxLength value="480"/>
            </xsd:restriction>
          </xsd:simpleType>
        </xsd:element>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="FastDataResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:complexType name="DisplayAttribute">
    <xsd:sequence>
      <xsd:element name="displayID" type="csta:DisplayID" minOccurs="0"/>
      <xsd:element name="physicalBaseRowNumber" type="xsd:long" minOccurs="0"/>
      <xsd:element name="physicalBaseColumnNumber" type="xsd:long" minOccurs="0"/>
      <xsd:element name="offset" type="xsd:long" minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>

</xsd:schema>
```

22.2.4 Resume data path

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-resume-data-path</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-feature-types.xsd" />
  <xsd:include schemaLocation="extension-types.xsd" />

  <xsd:element name="ResumeDataPath">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="ioCrossRefID" type="csta:IOCrossRefID" />
        <xsd:element name="ioRegisterRequestID" type="csta:IORegisterReqID" minOccurs="0" />
        <xsd:element ref="csta:extensions" minOccurs="0" />
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="ResumeDataPathResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:extensions" minOccurs="0" />
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

22.2.5 Send broadcast data

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-send-broadcast-data</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-feature-types.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>

  <xsd:element name="SendBroadcastData">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="ioData">
          <xsd:simpleType>
            <xsd:restriction base="xsd:hexBinary">
              <xsd:minLength value="0"/>
              <xsd:maxLength value="480"/>
            </xsd:restriction>
          </xsd:simpleType>
        </xsd:element>
        <xsd:element name="dataPathType" type="csta:DataPathType" minOccurs="0"/>
        <xsd:element name="displayAttributes" type="csta:DisplayAttributeList" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="SendBroadcastDataResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

22.2.6 Send data

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-send-data</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="event-cause.xsd"/>
  <xsd:include schemaLocation="device-feature-types.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>

  <xsd:element name="SendData">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="ioCrossRefID" type="csta:IOCrossRefID"/>
        <xsd:element name="ioRegisterReqID" type="csta:IORegisterReqID" minOccurs="0"/>
        <xsd:element name="displayAttributes" type="csta:DisplayAttributeList" minOccurs="0"/>
        <xsd:element name="ioData">
          <xsd:simpleType>
            <xsd:restriction base="xsd:hexBinary">
              <xsd:minLength value="0"/>
              <xsd:maxLength value="480"/>
            </xsd:restriction>
          </xsd:simpleType>
        </xsd:element>
        <xsd:element name="ioCause" type="csta:EventCause" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="SendDataResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

22.2.7 Send multicast data

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-send-multicast-data</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-feature-types.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>

  <xsd:element name="SendMulticastData">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="ioCrossRefIDList">
          <xsd:complexType>
            <xsd:sequence>
              <xsd:element name="ioCrossRefID" type="csta:IOCrossRefID" minOccurs="0"
maxOccurs="unbounded"/>
            </xsd:sequence>
          </xsd:complexType>
        </xsd:element>
        <xsd:element name="ioData" minOccurs="0">
          <xsd:simpleType>
            <xsd:restriction base="xsd:hexBinary">
              <xsd:minLength value="0"/>
              <xsd:maxLength value="480"/>
            </xsd:restriction>
          </xsd:simpleType>
        </xsd:element>
        <xsd:element name="displayAttributes" type="csta:DisplayAttributeList" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="SendMulticastDataResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```


22.2.8 Start data path

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-start-data-path</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="switching-function-objects.xsd"/>
  <xsd:include schemaLocation="device-feature-types.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>
  <xsd:include schemaLocation="physical-device-features.xsd"/>

  <xsd:element name="StartDataPath">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="ioRegisterReqID" type="csta:IORegisterReqID" minOccurs="0"/>
        <xsd:element name="object" type="csta:CSTAObject"/>
        <xsd:element name="dataPathDirection" type="csta:DataPathDirection" minOccurs="0"/>
        <xsd:element name="dataPathType" type="csta:DataPathType" minOccurs="0"/>
        <xsd:element name="displayID" type="csta:DisplayID" minOccurs="0"/>
        <xsd:element name="numberOfCharactersToCollect" type="xsd:long" minOccurs="0"/>
        <xsd:element name="terminationCharacter" minOccurs="0">
          <xsd:simpleType>
            <xsd:restriction base="xsd:string">
              <xsd:length value="1"/>
            </xsd:restriction>
          </xsd:simpleType>
        </xsd:element>
        <xsd:element name="timeout" type="xsd:long" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="StartDataPathResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="ioCrossRefID" type="csta:IOCrossRefID"/>
        <xsd:element name="numberOfCharactersToCollect" type="xsd:long" minOccurs="0"/>
        <xsd:element name="terminationCharacter" minOccurs="0">
          <xsd:simpleType>
            <xsd:restriction base="xsd:string">
              <xsd:length value="1"/>
            </xsd:restriction>
          </xsd:simpleType>
        </xsd:element>
        <xsd:element name="timeout" type="xsd:long" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:simpleType name="DataPathDirection">
    <xsd:restriction base="xsd:string">
      <xsd:enumeration value="computeFunctionToObject"/>
      <xsd:enumeration value="objectToComputeFunction"/>
      <xsd:enumeration value="bidirectional"/>
    </xsd:restriction>
  </xsd:simpleType>

</xsd:schema>
```

22.2.9 Stop data path

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-stop-data-path</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-feature-types.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>

  <xsd:element name="StopDataPath">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="ioCrossRefID" type="csta:IOCrossRefID"/>
        <xsd:element name="ioRegisterRequestID" type="csta:IORegisterReqID" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="StopDataPathResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

22.2.10 Suspend data path

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-suspend-data-path</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-feature-types.xsd" />
  <xsd:include schemaLocation="extension-types.xsd" />

  <xsd:element name="SuspendDataPath">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="ioCrossRefID" type="csta:IOCrossRefID" />
        <xsd:element name="ioRegisterRequestID" type="csta:IORegisterReqID" minOccurs="0" />
        <xsd:element ref="csta:extensions" minOccurs="0" />
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="SuspendDataPathResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:extensions" minOccurs="0" />
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

23 Data Collection Services

23.1 Services

23.1.1 Data Collected

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

```
<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"  
xmlns:xsd="http://www.w3.org/2001/XMLSchema"  
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"  
elementFormDefault="qualified" attributeFormDefault="unqualified">
```

```
<xsd:annotation>  
  <xsd:documentation>CSTA-data-collected</xsd:documentation>  
</xsd:annotation>
```

```
<xsd:include schemaLocation="extension-types.xsd"/>  
<xsd:include schemaLocation="media-services.xsd"/>  
<xsd:include schemaLocation="data-collection.xsd"/>
```

```
<xsd:element name="DataCollected">  
  <xsd:complexType>  
    <xsd:sequence>  
      <xsd:element ref="csta:dcollCrossRefID"/>  
      <xsd:element name="digitsData" type="csta:DigitsData" minOccurs="0"/>  
      <xsd:element name="telTonesData" type="csta:TelTonesData" minOccurs="0"/>  
      <xsd:element name="connectionInformation" type="csta:ConnectionInformation" minOccurs="0"/>  
      <xsd:element name="dcollcause" type="csta:DcollCause" minOccurs="0"/>  
      <xsd:element ref="csta:extensions" minOccurs="0"/>  
    </xsd:sequence>  
  </xsd:complexType>  
</xsd:element>
```

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

```
<xsd:complexType name="DigitsData">  
  <xsd:sequence>  
    <xsd:element name="digitsDetected">  
      <xsd:simpleType>  
        <xsd:restriction base="xsd:string">  
          <xsd:minLength value="0"/>  
          <xsd:maxLength value="64"/>  
        </xsd:restriction>  
      </xsd:simpleType>  
    </xsd:element>  
    <xsd:element name="digitsDuration" type="csta:DigitsDurationList" minOccurs="0"/>  
    <xsd:element name="digitsPauseDuration" type="csta:DigitsPauseDurationList" minOccurs="0"/>  
  </xsd:sequence>  
</xsd:complexType>
```

```
<xsd:complexType name="TelTonesData">  
  <xsd:sequence>  
    <xsd:element name="toneDetected" type="csta:ToneDetected"/>  
    <xsd:element name="toneFrequency" type="xsd:long" minOccurs="0"/>  
    <xsd:element name="toneDuration" type="xsd:long" minOccurs="0"/>  
    <xsd:element name="tonePauseDuration" type="xsd:long" minOccurs="0"/>  
  </xsd:sequence>  
</xsd:complexType>
```

```
<xsd:simpleType name="ToneDetected">  
  <xsd:restriction base="xsd:string">  
    <xsd:enumeration value="beep"/>
```

```
<xsd:enumeration value="billing"/>
<xsd:enumeration value="busy"/>
<xsd:enumeration value="carrier"/>
<xsd:enumeration value="confirmation"/>
<xsd:enumeration value="dial"/>
<xsd:enumeration value="faxCNG"/>
<xsd:enumeration value="hold"/>
<xsd:enumeration value="howler"/>
<xsd:enumeration value="intrusion"/>
<xsd:enumeration value="modemCNG"/>
<xsd:enumeration value="park"/>
<xsd:enumeration value="recordWarning"/>
<xsd:enumeration value="reorder"/>
<xsd:enumeration value="ringback"/>
<xsd:enumeration value="silence"/>
<xsd:enumeration value="sitVC"/>
<xsd:enumeration value="sitIC"/>
<xsd:enumeration value="sitRO"/>
<xsd:enumeration value="sitNC"/>
<xsd:enumeration value="other"/>
</xsd:restriction>
</xsd:simpleType>

<xsd:simpleType name="DcollCause">
  <xsd:restriction base="xsd:string">
    <xsd:enumeration value="flushCharReceived"/>
    <xsd:enumeration value="charCountReceived"/>
    <xsd:enumeration value="timeout"/>
    <xsd:enumeration value="sfTerminated"/>
  </xsd:restriction>
</xsd:simpleType>

<xsd:complexType name="DigitsDurationList">
  <xsd:sequence>
    <xsd:element name="digitsDurationItem" type="xsd:long" minOccurs="0" maxOccurs="unbounded"/>
  </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="DigitsPauseDurationList">
  <xsd:sequence>
    <xsd:element name="digitsPauseDurationItem" type="xsd:long" minOccurs="0"
      maxOccurs="unbounded"/>
  </xsd:sequence>
</xsd:complexType>

</xsd:schema>
```

23.1.2 Data Collection Resumed

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-data-collection-resumed</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="extension-types.xsd"/>
  <xsd:include schemaLocation="data-collection.xsd"/>

  <xsd:element name="DataCollectionResumed">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:dcollCrossRefID"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="DataCollectionResumedResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

23.1.3 Data Collection Suspended

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-data-collection-suspended</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="extension-types.xsd"/>
  <xsd:include schemaLocation="data-collection.xsd"/>

  <xsd:element name="DataCollectionSuspended">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:dcollCrossRefID"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="DataCollectionSuspendedResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

23.1.4 Resume Data Collection

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-resume-data-collection</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="extension-types.xsd"/>
  <xsd:include schemaLocation="data-collection.xsd"/>

  <xsd:element name="ResumeDataCollection">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:dcollCrossRefID"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="ResumeDataCollectionResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```


23.1.5 Start Data Collection

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-start-data-collection</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-identifiers.xsd"/>
  <xsd:include schemaLocation="call-connection-identifiers.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>
  <xsd:include schemaLocation="data-collection.xsd"/>

  <xsd:element name="StartDataCollection">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="object" type="csta:CallObject"/>
        <xsd:element name="dataCollType" type="csta:DataCollType" minOccurs="0"/>
        <xsd:element name="digitsReportingCriteria" type="csta:DigitsReportingCriteria"
minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="StartDataCollectionResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:dcollCrossRefID"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:complexType name="CallObject">
    <xsd:choice>
      <xsd:element name="device" type="csta:DeviceID"/>
      <xsd:element name="call" type="csta:ConnectionID"/>
    </xsd:choice>
  </xsd:complexType>

  <xsd:simpleType name="DataCollType">
    <xsd:restriction base="xsd:string">
      <xsd:enumeration value="digits"/>
      <xsd:enumeration value="telTones"/>
    </xsd:restriction>
  </xsd:simpleType>

  <xsd:complexType name="DigitsReportingCriteria">
    <xsd:sequence>
      <xsd:element name="numChars" type="xsd:long" minOccurs="0"/>
      <xsd:element name="flushChar" minOccurs="0">
        <xsd:simpleType>
          <xsd:restriction base="xsd:string">
            <xsd:length value="1"/>
          </xsd:restriction>
        </xsd:simpleType>
      </xsd:element>
      <xsd:element name="timeout" type="xsd:long" minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>

</xsd:schema>
```

23.1.6 Stop Data Collection

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-stop-data-collection</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="extension-types.xsd"/>
  <xsd:include schemaLocation="data-collection.xsd"/>

  <xsd:element name="StopDataCollection">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:dcollCrossRefID"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="StopDataCollectionResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

23.1.7 Suspend Data Collection

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-suspend-data-collection</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="extension-types.xsd"/>
  <xsd:include schemaLocation="data-collection.xsd"/>

  <xsd:element name="SuspendDataCollection">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:dcollCrossRefID"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="SuspendDataCollectionResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

24 Voice unit services and events

24.1 Services

24.1.1 Concatenate message

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-concatenate-message</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-feature-types.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>

  <xsd:element name="ConcatenateMessage">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="messagesToConcatenate" type="csta:MessageIDList"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="ConcatenateMessageResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="concatenatedMessage" type="csta:MessageID"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:complexType name="MessageIDList">
    <xsd:sequence>
      <xsd:element name="messageIDItem" type="csta:MessageID" minOccurs="0"
maxOccurs="unbounded"/>
    </xsd:sequence>
  </xsd:complexType>

</xsd:schema>
```

24.1.2 Delete message

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-delete-message</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-feature-types.xsd" />
  <xsd:include schemaLocation="extension-types.xsd" />

  <xsd:element name="DeleteMessage">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="messageToBeDeleted" type="csta:MessageID"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="DeleteMessageResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

24.1.3 Play message

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-play-message</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="call-connection-identifiers.xsd"/>
  <xsd:include schemaLocation="device-feature-types.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>

  <xsd:element name="PlayMessage">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="messageToBePlayed" type="csta:MessageID"/>
        <xsd:element name="overConnection" type="csta:ConnectionID"/>
        <xsd:element name="duration" type="xsd:long" minOccurs="0"/>
        <xsd:element name="termination" type="csta:TerminatingConditions" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="PlayMessageResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

24.1.4 Query voice attribute

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-query-voice-attribute</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="call-connection-identifiers.xsd"/>
  <xsd:include schemaLocation="device-feature-types.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>

  <xsd:element name="QueryVoiceAttribute">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="messageToQuery" type="csta:MessageID"/>
        <xsd:element name="attributeToQuery" type="csta:AttributeToQuery"/>
        <xsd:element name="connection" type="csta:ConnectionID" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="QueryVoiceAttributeResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="attribute" type="csta:AttributeInfo"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:simpleType name="AttributeToQuery">
    <xsd:restriction base="xsd:string">
      <xsd:enumeration value="encodingAlgorithm"/>
      <xsd:enumeration value="samplingRate"/>
      <xsd:enumeration value="duration"/>
      <xsd:enumeration value="filename"/>
      <xsd:enumeration value="currentPosition"/>
      <xsd:enumeration value="currentSpeed"/>
      <xsd:enumeration value="currentVolume"/>
      <xsd:enumeration value="currentLevel"/>
      <xsd:enumeration value="currentState"/>
    </xsd:restriction>
  </xsd:simpleType>

</xsd:schema>
```

24.1.5 Record message

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-record-message</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="call-connection-identifiers.xsd"/>
  <xsd:include schemaLocation="device-feature-types.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>

  <xsd:element name="RecordMessage">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="callToBeRecorded" type="csta:ConnectionID"/>
        <xsd:element name="samplingRate" type="xsd:long" minOccurs="0"/>
        <xsd:element name="encodingAlgorithm" type="csta:EncodingAlgorithm" minOccurs="0"/>
        <xsd:element name="maxDuration" type="xsd:long" minOccurs="0"/>
        <xsd:element name="termination" type="csta:TerminatingConditions" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="RecordMessageResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="resultingMessage" type="csta:MessageID"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```


24.1.6 Reposition

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-reposition</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="call-connection-identifiers.xsd"/>
  <xsd:include schemaLocation="device-feature-types.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>

  <xsd:element name="Reposition">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="connection" type="csta:ConnectionID"/>
        <xsd:element name="periodOfReposition" type="csta:Period"/>
        <xsd:element name="messageToReposition" type="csta:MessageID" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="RepositionResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:complexType name="Period">
    <xsd:choice>
      <xsd:element name="absolutePosition">
        <xsd:simpleType>
          <xsd:restriction base="xsd:string">
            <xsd:enumeration value="startOfMessage"/>
            <xsd:enumeration value="endOfMessage"/>
          </xsd:restriction>
        </xsd:simpleType>
      </xsd:element>
      <xsd:element name="relativePosition" type="xsd:long"/>
    </xsd:choice>
  </xsd:complexType>

</xsd:schema>
```

24.1.7 Resume

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-resume</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="call-connection-identifiers.xsd"/>
  <xsd:include schemaLocation="device-feature-types.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>

  <xsd:element name="Resume">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="connection" type="csta:ConnectionID"/>
        <xsd:element name="messageToResume" type="csta:MessageID" minOccurs="0"/>
        <xsd:element name="duration" type="xsd:long" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="ResumeResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

24.1.8 Review

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-review</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="call-connection-identifiers.xsd"/>
  <xsd:include schemaLocation="device-feature-types.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>

  <xsd:element name="Review">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="connection" type="csta:ConnectionID"/>
        <xsd:element name="periodToReview" type="csta:PeriodToReview"/>
        <xsd:element name="messageToReview" type="csta:MessageID" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="ReviewResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:complexType name="PeriodToReview">
    <xsd:choice>
      <xsd:element name="startOfMessage" type="csta:Empty"/>
      <xsd:element name="lengthOfReview" type="xsd:long"/>
    </xsd:choice>
  </xsd:complexType>

</xsd:schema>
```

24.1.9 Set voice attribute

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-set-voice-attribute</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="call-connection-identifiers.xsd"/>
  <xsd:include schemaLocation="device-feature-types.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>
  <xsd:include schemaLocation="physical-device-features.xsd"/>

  <xsd:element name="SetVoiceAttribute">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="connection" type="csta:ConnectionID"/>
        <xsd:element name="attributeToSet" type="csta:AttributeToSet"/>
        <xsd:element name="message" type="csta:MessageID" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="SetVoiceAttributeResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:complexType name="AttributeToSet">
    <xsd:choice>
      <xsd:element name="currentSpeed" type="xsd:long"/>
      <xsd:element name="currentVolume" type="csta:Volume"/>
      <xsd:element name="currentGain">
        <xsd:simpleType>
          <xsd:restriction base="xsd:integer">
            <xsd:minInclusive value="0"/>
            <xsd:maxInclusive value="100"/>
          </xsd:restriction>
        </xsd:simpleType>
      </xsd:element>
    </xsd:choice>
  </xsd:complexType>

</xsd:schema>
```

24.1.10 Stop

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-stop</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="call-connection-identifiers.xsd"/>
  <xsd:include schemaLocation="device-feature-types.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>

  <xsd:element name="Stop">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="connection" type="csta:ConnectionID"/>
        <xsd:element name="messageToBeStopped" type="csta:MessageID" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="StopResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

24.1.11 Suspend

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-suspend</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="call-connection-identifiers.xsd"/>
  <xsd:include schemaLocation="device-feature-types.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>

  <xsd:element name="Suspend">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="connection" type="csta:ConnectionID"/>
        <xsd:element name="message" type="csta:MessageID" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="SuspendResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

24.1.12 Synthesize message

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-synthesize-message</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="device-feature-types.xsd" />
  <xsd:include schemaLocation="extension-types.xsd" />

  <xsd:element name="SynthesizeMessage">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="textToBeSynthesized" type="xsd:string"/>
        <xsd:element name="control" type="csta:ControlData" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="SynthesizeMessageResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="synthesizedMessage" type="csta:MessageID"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

24.2 Events

24.2.1 Play

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-play-event</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="event-cause.xsd"/>
  <xsd:include schemaLocation="call-connection-identifiers.xsd"/>
  <xsd:include schemaLocation="device-feature-types.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>
  <xsd:include schemaLocation="call-control.xsd"/>
  <xsd:include schemaLocation="status-reporting.xsd"/>

  <xsd:element name="PlayEvent">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:monitorCrossRefID"/>
        <xsd:element name="connection" type="csta:ConnectionID"/>
        <xsd:element name="message" type="csta:MessageID"/>
        <xsd:element name="length" type="xsd:long" minOccurs="0"/>
        <xsd:element name="currentPosition" type="xsd:long" minOccurs="0"/>
        <xsd:element name="speed" type="xsd:long" minOccurs="0"/>
        <xsd:element ref="csta:cause" minOccurs="0"/>
        <xsd:element ref="csta:servicesPermitted" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```


24.2.2 Record

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-record-event</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="event-cause.xsd"/>
  <xsd:include schemaLocation="call-connection-identifiers.xsd"/>
  <xsd:include schemaLocation="device-feature-types.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>
  <xsd:include schemaLocation="call-control.xsd"/>
  <xsd:include schemaLocation="status-reporting.xsd"/>

  <xsd:element name="RecordEvent">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:monitorCrossRefID"/>
        <xsd:element name="connection" type="csta:ConnectionID"/>
        <xsd:element name="message" type="csta:MessageID"/>
        <xsd:element name="length" type="xsd:long" minOccurs="0"/>
        <xsd:element name="currentPosition" type="xsd:long" minOccurs="0"/>
        <xsd:element ref="csta:cause" minOccurs="0"/>
        <xsd:element ref="csta:servicesPermitted" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>
</xsd:schema>
```

24.2.3 Review

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-review-event</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="event-cause.xsd" />
  <xsd:include schemaLocation="call-connection-identifiers.xsd" />
  <xsd:include schemaLocation="device-feature-types.xsd" />
  <xsd:include schemaLocation="extension-types.xsd" />
  <xsd:include schemaLocation="call-control.xsd" />
  <xsd:include schemaLocation="status-reporting.xsd" />

  <xsd:element name="ReviewEvent">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:monitorCrossRefID" />
        <xsd:element name="connection" type="csta:ConnectionID" />
        <xsd:element name="message" type="csta:MessageID" />
        <xsd:element name="length" type="xsd:long" minOccurs="0" />
        <xsd:element name="currentPosition" type="xsd:long" minOccurs="0" />
        <xsd:element ref="csta:cause" minOccurs="0" />
        <xsd:element ref="csta:servicesPermitted" minOccurs="0" />
        <xsd:element ref="csta:extensions" minOccurs="0" />
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

24.2.4 Stop

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-stop-event</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="event-cause.xsd"/>
  <xsd:include schemaLocation="call-connection-identifiers.xsd"/>
  <xsd:include schemaLocation="device-feature-types.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>
  <xsd:include schemaLocation="call-control.xsd"/>
  <xsd:include schemaLocation="status-reporting.xsd"/>

  <xsd:element name="StopEvent">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:monitorCrossRefID"/>
        <xsd:element name="connection" type="csta:ConnectionID"/>
        <xsd:element name="message" type="csta:MessageID"/>
        <xsd:element name="length" type="xsd:long" minOccurs="0"/>
        <xsd:element name="currentPosition" type="xsd:long" minOccurs="0"/>
        <xsd:element ref="csta:cause" minOccurs="0"/>
        <xsd:element ref="csta:servicesPermitted" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

24.2.5 Suspend play

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-suspend-play-event</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="event-cause.xsd" />
  <xsd:include schemaLocation="call-connection-identifiers.xsd" />
  <xsd:include schemaLocation="device-feature-types.xsd" />
  <xsd:include schemaLocation="extension-types.xsd" />
  <xsd:include schemaLocation="call-control.xsd" />
  <xsd:include schemaLocation="status-reporting.xsd" />

  <xsd:element name="SuspendPlayEvent">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:monitorCrossRefID" />
        <xsd:element name="connection" type="csta:ConnectionID" />
        <xsd:element name="message" type="csta:MessageID" />
        <xsd:element name="length" type="xsd:long" minOccurs="0" />
        <xsd:element name="currentPosition" type="xsd:long" minOccurs="0" />
        <xsd:element ref="csta:cause" minOccurs="0" />
        <xsd:element ref="csta:servicesPermitted" minOccurs="0" />
        <xsd:element ref="csta:extensions" minOccurs="0" />
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

24.2.6 Suspend record

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-suspend-record-event</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="event-cause.xsd"/>
  <xsd:include schemaLocation="call-connection-identifiers.xsd"/>
  <xsd:include schemaLocation="device-feature-types.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>
  <xsd:include schemaLocation="call-control.xsd"/>
  <xsd:include schemaLocation="status-reporting.xsd"/>

  <xsd:element name="SuspendRecordEvent">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:monitorCrossRefID"/>
        <xsd:element name="connection" type="csta:ConnectionID"/>
        <xsd:element name="message" type="csta:MessageID"/>
        <xsd:element name="length" type="xsd:long" minOccurs="0"/>
        <xsd:element name="currentPosition" type="xsd:long" minOccurs="0"/>
        <xsd:element ref="csta:cause" minOccurs="0"/>
        <xsd:element ref="csta:servicesPermitted" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

24.2.7 Voice attribute changed

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-voice-attribute-changed-event</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="event-cause.xsd"/>
  <xsd:include schemaLocation="call-connection-identifiers.xsd"/>
  <xsd:include schemaLocation="device-feature-types.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>
  <xsd:include schemaLocation="call-control.xsd"/>
  <xsd:include schemaLocation="physical-device-features.xsd"/>
  <xsd:include schemaLocation="status-reporting.xsd"/>

  <xsd:element name="VoiceAttributeChangedEvent">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:monitorCrossRefID"/>
        <xsd:element name="connection" type="csta:ConnectionID"/>
        <xsd:element name="message" type="csta:MessageID"/>
        <xsd:element name="playVolume" type="csta:Volume" minOccurs="0"/>
        <xsd:element name="recordingGain" minOccurs="0">
          <xsd:simpleType>
            <xsd:restriction base="xsd:integer">
              <xsd:minInclusive value="0"/>
              <xsd:maxInclusive value="100"/>
            </xsd:restriction>
          </xsd:simpleType>
        </xsd:element>
        <xsd:element name="speed" type="xsd:long" minOccurs="0"/>
        <xsd:element name="currentPosition" type="xsd:long" minOccurs="0"/>
        <xsd:element ref="csta:cause" minOccurs="0"/>
        <xsd:element ref="csta:servicesPermitted" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

25 Call detail record services

25.1 Services

25.1.1 Call detail records notification

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-call-detail-records-notification</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="extension-types.xsd"/>
  <xsd:include schemaLocation="call-detail-record.xsd"/>

  <xsd:element name="CDRNotification">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:cdrCrossRefID"/>
        <xsd:element name="cdrReason" type="csta:CDRReason" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="CDRNotificationResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

25.1.2 Call detail records report

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-call-detail-records-report</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="extension-types.xsd"/>
  <xsd:include schemaLocation="call-detail-record.xsd"/>

  <xsd:element name="CDRReport">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:cdrCrossRefID"/>
        <xsd:element name="numberOfRecordsSent">
          <xsd:simpleType>
            <xsd:restriction base="xsd:integer">
              <xsd:minInclusive value="1"/>
              <xsd:maxInclusive value="128"/>
            </xsd:restriction>
          </xsd:simpleType>
        </xsd:element>
        <xsd:element name="cdrInfo" type="csta:CDRInfo"/>
        <xsd:element name="lastStoredCDRReportSent" type="xsd:boolean" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="CDRReportResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```


25.1.3 Send stored call detail records

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-send-stored-call-detail-records</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="extension-types.xsd"/>
  <xsd:include schemaLocation="call-detail-record.xsd"/>

  <xsd:element name="SendStoredCDR">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:cdrCrossRefID"/>
        <xsd:element name="timePeriod" type="csta:CDRTimePeriod" minOccurs="0"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="SendStoredCDRResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

25.1.4 Start call detail records transmission

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-start-call-detail-records-transmission</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="extension-types.xsd"/>
  <xsd:include schemaLocation="call-detail-record.xsd"/>

  <xsd:element name="StartCDRTransmission">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="transferMode" type="csta:CDRTransferMode"/>
        <xsd:element name="extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="StartCDRTransmissionResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:cdrCrossRefID"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

25.1.5 Stop call detail records transmission

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-stop-call-detail-records-transmission</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="extension-types.xsd"/>
  <xsd:include schemaLocation="call-detail-record.xsd"/>

  <xsd:element name="StopCDRTransmission">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:cdrCrossRefID"/>
        <xsd:element name="cdrTermReason" type="csta:CDRTermReason" minOccurs="0"/>
        <xsd:element name="extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="StopCDRTransmissionResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

26 Vendor specific extensions services and events

26.1 Registration services

26.1.1 Escape register

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-escape-register</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="extension-types.xsd"/>
  <xsd:include schemaLocation="escape-types.xsd"/>

  <xsd:element name="EscapeRegister">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="EscapeRegisterResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:escapeRegisterID"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

26.1.2 Escape register abort

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-escape-register-abort</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="extension-types.xsd"/>
  <xsd:include schemaLocation="escape-types.xsd"/>

  <xsd:element name="EscapeRegisterAbort">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:escapeRegisterID"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:annotation>
    <xsd:documentation>There is no positive response to this request.</xsd:documentation>
  </xsd:annotation>

</xsd:schema>
```

26.1.3 Escape register cancel

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-escape-register-cancel</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="extension-types.xsd"/>
  <xsd:include schemaLocation="escape-types.xsd"/>

  <xsd:element name="EscapeRegisterCancel">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:escapeRegisterID"/>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="EscapeRegisterCancelResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

26.2 Services

26.2.1 Escape

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-escape</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="security.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>
  <xsd:include schemaLocation="escape-types.xsd"/>

  <xsd:element name="Escape">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:escapeRegisterID" minOccurs="0"/>
        <xsd:element name="security" type="csta:CSTASecurityData" minOccurs="0"/>
        <xsd:element name="privateData" type="csta:CSTAPrivateData"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="EscapeResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```

26.2.2 Private data version selection

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"
elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xsd:annotation>
    <xsd:documentation>CSTA-private-data-version-selection</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="extension-types.xsd"/>

  <xsd:element name="PrivateDataVersionSelection">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="privateDataVersionSelectionArgument"
          type="csta:PrivateDataVersionSelectionArgument"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="PrivateDataVersionSelectionResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:extensions" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:simpleType name="PrivateDataVersionSelectionArgument">
    <xsd:restriction base="xsd:integer"/>
  </xsd:simpleType>

</xsd:schema>
```


26.3 Events

26.3.1 Private event

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

<xsd:schema targetNamespace="http://www.ecma.ch/standards/ecma-323/csta"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta"

elementFormDefault="qualified" attributeFormDefault="unqualified">
  <xsd:annotation>
    <xsd:documentation>CSTA-private-event</xsd:documentation>
  </xsd:annotation>

  <xsd:include schemaLocation="security.xsd"/>
  <xsd:include schemaLocation="extension-types.xsd"/>
  <xsd:include schemaLocation="status-reporting.xsd"/>

  <xsd:element name="PrivateEvent">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="csta:monitorCrossRefID"/>
        <xsd:element name="security" type="csta:CSTASecurityData" minOccurs="0"/>
        <xsd:element name="privateData" type="csta:CSTAPrivateData"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```


Annex A

(normative)

Protocol Implementation Conformance Statement (PICS) Proforma

A.1 Introduction

The Protocol Implementation Conformance Statement (PICS) is a statement of which capabilities and options of the protocol have been implemented. The PICS can have a number of uses, including use:

- by the protocol implementor, as a check-list to reduce the risk of failure to conform to the standard through oversight;
- by the supplier and acquirer (or potential acquirer) of the implementation, as a detailed indication of the capabilities of the implementation, stated relative to the common basis for understanding provided by the standard PICS proforma;
- by the user (or potential user) of the implementation, as a basis for initially checking the possibility of interworking with another implementation (note that, while interworking cannot be guaranteed, failure to interwork can often be predicted from incompatible PICS);
- by a protocol tester, as the basis for selecting appropriate tests against which to assess the claim for conformance of the implementation.

A.2 Conformance

The supplier of a protocol implementation which is claimed to conform to this Standard shall complete a copy of the Protocol Implementation Conformance Statement (PICS) proforma in A.5, "PICS proforma".

A.3 Instructions for completing the PICS proforma

The first part of the PICS proforma, the Implementation Identification (A.4, "Implementation identification", on page 330), is to be completed as indicated with the information necessary to identify fully both the supplier and the implementation.

The main part of the PICS proforma, A.5, "PICS proforma", is a fixed format questionnaire divided into subclauses each containing a group of individual items. These items represent options specified in ECMA-269 (mandatory items are not represented). Answers to the questionnaire items are to be provided in the appropriate columns by marking an answer to indicate a restricted choice (Yes or No), and optionally clarifying this response with a comment.

Where a service or event is not supported, as indicated by entering No for the first entry in a table, any parameters or dependent service components in the rest of that table are not applicable, it is then not necessary to complete items in any subsidiary entries in the table.

Note that ECMA-269 should be used as a reference when completing the PICS proforma.

A.4 Implementation identification

Supplier	
Protocol Version	XML Protocol for CSTA Phase III
Date of Statement	
Contact point for queries about the PICS	
Implementation Name(s) and Version(s)	
Other information necessary for full identification - e.g. Name(s) and Version(s) for machines and/or operating systems; system name(s)	

The first five items are required for all implementations; other information may be completed as appropriate in meeting the requirement for full identification.

The terms Name and Version should be interpreted appropriately to correspond with a supplier's terminology (e.g. Type, Series, Model).

A.5 PICS proforma

A.5.1 CSTA Profiles

CSTA Profiles group together services and events, where each profile specifies a set of services and events that are supported by the implementation. Switching Function implementors shall indicate support for one or more profiles in the table below. Specifying support for a profile shall be accompanied by specifying support for the services and events encompassed by the profile, as well as any other services and events that the implementation supports.

Refer to ECMA-269 for a description of the CSTA Services and Events that must be supported for a specific CSTA profile.

Description: Profile(s) Supported	Supported?		Comments
	Yes	No	
Basic Telephony Profile			
Routeing Profile			

A.5.2 Capability Exchange Services

A.5.2.1 Get Logical Device Information

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Get Logical Device Information			
Service Request optional parameters			
security			
privateData			
Service Response optional parameters			
groupDeviceAttributes			
namedDeviceTypes			
shortFormDeviceID			
agentLogOnModels			
appearanceList			
otherPhysicalDeviceList			
miscMonitorCaps			
associatedGroupList			
maxCallbacks			
maxAutoAnswerRings			
maxActiveCalls			
maxHeldCalls			
maxFwdSettings			
maxDevicesInConf			
transAndConfSetup			
deviceOnDeviceMonitorFilter			
deviceOnConnectionMonitorFilter			
callOnDeviceMonitorFilter			
callOnConnectionMonitorFilter			
mediaClassSupport			
mediaServiceCapsList			

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
connectionRateList			
delayToleranceList			
numberOfChannels			
maxChannelBind			
routeingServList			
logDevServList			
logDevEvtsList			
deviceMaintEvtsList			
security			
privateData			

A.5.2.2 Get Physical Device Information

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Get Physical Device Information			
Service Request optional parameters			
security			
privateData			
Service Response optional parameters			
groupDeviceAttributes			
namedDeviceTypes			
otherLogicalDeviceList			
deviceModelName			
deviceOnDeviceMonitorFilter			
deviceOnConnectionMonitorFilter			
callOnDeviceMonitorFilter			
callOnConnectionMonitorFilter			
maxDisplays			

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
maxButtons			
maxLamps			
maxRingPatterns			
physDevServList			
physDevEvtsList			
security			
privateData			

A.5.2.3 Get Switching Function Capabilities

This service shall be supported by a switching function.

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Get Switching Function Capabilities			
Service Request optional parameters			
security			
privateData			
Service Response optional parameters			
callCharacteristicsSupported			
mediaClassSupport			
numberOfChannels			
maxChannelBind			
miscMediaCallCharacteristics			
connectionRateList			
delayToleranceRateList			
pauseTime			
currentTime			
messageSeqNumbers			
timeStampMode			

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
securityMode			
securityFormat			
privateDataFormat			
transAndConfSetup			
deviceOnDeviceMonitorFilter			
deviceOnConnectionMonitorFilter			
callOnDeviceMonitorFilter			
callOnConnectionMonitorFilter			
miscMonitorCaps			
correlatorDataSupported			
dynamicFeatureSupported			
callLinkageOptionsSupported			
acdModels			
agentLogonModels			
agentStateModels			
routeingServList			
logDevServList			
logDevEvtsList			
physDevServList			
physDevEvtsList			
deviceMaintEvtsList			
statusReportingServList			
capExchangeServList			
cdrServList			
vendorSpecificServList			
vendorSpecificEvtsList			
privateDataVersionList			
systemStatusTimer			

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
simpleThreshold			
filterThreshold			
mediaServiceCapsList			
security			
privateData			

A.5.2.4 Get Switching Function Devices

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Get Switching Function Devices			
Service Request optional parameters			
requestedDeviceID			
requestedDeviceCategory			
security			
privateData			
Service Response optional parameters			
security			
privateData			

A.5.2.5 Switching Function Devices

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Switching Function Devices			
Service Request optional parameters			
segmentID			
security			
privateData			

A.5.3 System Services

A.5.3.1 Change System Status Filter

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Change System Status Filter			
Service Request optional parameters			
security			
privateData			
Service Response optional parameters			
security			
privateData			

A.5.3.2 System Register

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
System Register			
Service Request optional parameters			
requestedStatusFilter			
security			
privateData			
Service Response optional parameters			
actualStatusFilter			
security			
privateData			

A.5.3.3 System Register Abort

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
System Register Abort			
Service Request optional parameters			
security			
privateData			

A.5.3.4 System Register Cancel

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
System Register Cancel			
Service Request optional parameters			
security			
privateData			
Service Response optional parameters			
security			
privateData			

A.5.3.5 Request System Status

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Request System Status			
Service Request optional parameters			
sysStatRegisterID			
security			
privateData			
Service Response optional parameters			
security			
privateData			

A.5.3.6 System Status

The System Status service shall be supported as part of the Application Association Initialisation as specified in Clause 7 of ECMA-269.

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
System Status			
Service Request optional parameters			
sysStatRegisterID			
security			
privateData			
Service Response optional parameters			
security			
privateData			

A.5.3.7 Switching Function Capabilities Changed

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Switching Function Capabilities Changed			
Service Request optional parameters			
sysStatRegisterID			
security			
privateData			
Service Response optional parameters			
security			
privateData			

A.5.3.8 Switching Function Devices Changed

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Switching Function Devices Changed			
Service Request optional parameters			
sysStatRegisterID			
security			
privateData			
Service Response optional parameters			
security			
privateData			

A.5.4 Monitoring Services

A.5.4.1 Change Monitor Filter

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Change Monitor Filter			
Service Request optional parameters			
security			
privateData			
Service Response optional parameters			
actualFilterList			
security			
privateData			

A.5.4.2 Monitor Start

This service must be supported if the Basic Telephony profile is supported (A.5.1).

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Monitor Start			
Service Request optional parameters			
requestedMonitorFilter			
monitorType			
requestedMonitorMediaClass			
security			
privateData			
Service Response optional parameters			
actualMonitorFilter			
actualMonitorMediaClass			
monitorExistingCalls			
security			
privateData			

A.5.4.3 Monitor Stop

This service must be supported if the Basic Telephony profile is supported (A.5.1).

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Monitor Stop			
Service Request optional parameters			
security			
privateData			
Service Response optional parameters			
security			
privateData			

A.5.5 Snapshot Services

A.5.5.1 Snapshot Call

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Snapshot Call			
Service Request optional parameters			
security			
privateData			
Service Response optional parameters			
serviceCrossRefID			
snapshotData			
mediaCallCharacteristics			
callCharacteristics			
callingDevice			
calledDevice			
associatedCallingDevice			
associatedCalledDevice			
correlatorData			
security			
privateData			

A.5.5.2 Snapshot Device

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Snapshot Device			
Service Request optional parameters			
security			
privateData			
Service Response optional parameters			
serviceCrossRefID			

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
snapshotData			
security			
privateData			

A.5.5.3 Snapshot CallData

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Snapshot CallData			
Service Request optional parameters			
segmentID			
security			
privateData			

A.5.5.4 Snapshot DeviceData

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Snapshot DeviceData			
Service Request optional parameters			
segmentID			
security			
privateData			

A.5.6 Call Control Services

A.5.6.1 Accept Call

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Accept Call			
Service Request optional parameters			
correlatorData			
userData			
security			
privateData			
Service Response optional parameters			
security			
privateData			

A.5.6.2 Alternate Call

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Alternate Call			
Service Request optional parameters			
connectionReservation			
consultOptions			
security			
privateData			
Service Response optional parameters			
security			
privateData			

A.5.6.3 Answer Call

This service must be supported if the Basic Telephony profile is supported (A.5.1).

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Answer Call			
Service Request optional parameters			
correlatorData			
userData			
security			
privateData			
Service Response optional parameters			
security			
privateData			

A.5.6.4 Call Back Call-Related

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Call Back Call-Related			
Service Request optional parameters			
callCharateristics			
security			
privateData			
Service Response optional parameters			
targetDevice			
security			
privateData			

A.5.6.5 Call Back Message Call-Related

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Call Back Message Call-Related			
Service Request optional parameters			
security			
privateData			
Service Response optional parameters			
targetDevice			
security			
privateData			

A.5.6.6 Camp On Call

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Camp On Call			
Service Request optional parameters			
security			
privateData			
Service Response optional parameters			
security			
privateData			

A.5.6.7 Clear Call

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Clear Call			
Service Request optional parameters			
userData			

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
security			
privateData			
Service Response optional parameters			
security			
privateData			

A.5.6.8 Clear Connection

This service must be supported if the Basic Telephony profile is supported (A.5.1).

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Clear Connection			
Service Request optional parameters			
correlatorData			
userData			
security			
privateData			
Service Response optional parameters			
security			
privateData			

A.5.6.9 Conference Call

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Conference Call			
Service Request optional parameters			
security			
privateData			

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Service Response optional parameters			
connections			
conferenceCallInfo			
security			
privateData			

A.5.6.10 Consultation Call

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Consultation Call			
Service Request optional parameters			
connectionReservation			
accountCode			
authCode			
correlatorData			
userData			
callCharacteristics			
mediaCallCharacteristics			
callingConnectionInfo			
consultOptions			
security			
privateData			
Service Response optional parameters			
mediaCallCharacteristics			
initiatedCallInfo			
security			
privateData			

A.5.6.11 Deflect Call

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Deflect Call			
Service Request optional parameters			
correlatorData			
userData			
security			
privateData			
Service Response optional parameters			
security			
privateData			

A.5.6.12 Dial Digits

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Dial Digits			
Service Request optional parameters			
correlatorData			
security			
privateData			
Service Response optional parameters			
security			
privateData			

A.5.6.13 Directed Pickup Call

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Directed Pickup Call			
Service Request optional parameters			
correlatorData			
userData			
security			
privateData			
Service Response optional parameters			
pickedCall			
pickedCallInfo			
security			
privateData			

A.5.6.14 Group Pickup Call

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Group Pickup Call			
Service Request optional parameters			
pickGroup			
correlatorData			
userData			
security			
privateData			
Service Response optional parameters			
pickedCall			
pickedCallInfo			
security			
privateData			

A.5.6.15 Hold Call

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Hold Call			
Service Request optional parameters			
connectionReservation			
security			
privateData			
Service Response optional parameters			
security			
privateData			

A.5.6.16 Intrude Call

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Intrude Call			
Service Request optional parameters			
participationType			
userData			
security			
privateData			
Service Response optional parameters			
conferencedCall			
conferencedCallInfo			
security			
privateData			

A.5.6.17 Join Call

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Join Call			
Service Request optional parameters			
autoOriginate			
participationType			
accountCode			
authCode			
correlatorData			
userData			
security			
privateData			
Service Response optional parameters			
conferencedCallInfo			
security			
privateData			

A.5.6.18 Make Call

This service must be supported if the Basic Telephony profile is supported (A.5.1).

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Make Call			
Service Request optional parameters			
accountCode			
authCode			
autoOriginate			
correlatorData			
userData			
callCharacteristics			

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
medaCallCharacteristics			
callingConnectionInfo			
security			
privateData			
Service Response optional parameters			
mediaCallCharacteristics			
initiatedCallInfo			
security			
privateData			

A.5.6.19 Make Predictive Call

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Make Predictive Call			
Service Request optional parameters			
signallingDectection			
destinationDetection			
defaultAction			
accountCode			
authCode			
autoOriginate			
alertTime			
correlatorData			
callCharacteristics			
userData			
security			
privateData			

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Service Response optional parameters			
initiatedCallInfo			
security			
privateData			

A.5.6.20 Park Call

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Park Call			
Service Request optional parameters			
correlatorData			
security			
privateData			
Service Response optional parameters			
parkedTo			
security			
privateData			

A.5.6.21 Reconnect Call

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Reconnect Call			
Service Request optional parameters			
security			
privateData			
Service Response optional parameters			
security			
privateData			

A.5.6.22 Retrieve Call

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Retrieve Call			
Service Request optional parameters			
security			
privateData			
Service Response optional parameters			
security			
privateData			

A.5.6.23 Single Step Conference Call

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Single Step Conference Call			
Service Request optional parameters			
participationType			
accountCode			
authCode			
correlatorData			
userData			
security			
privateData			
Service Response optional parameters			
conferencedCallInfo			
security			
privateData			

A.5.6.24 Single Step Transfer Call

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Single Step Transfer Call			
Service Request optional parameters			
accountCode			
authCode			
correlatorData			
userData			
security			
privateData			
Service Response optional parameters			
connections			
transferredCallInfo			
security			
privateData			

A.5.6.25 Transfer Call

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Transfer Call			
Service Request optional parameters			
security			
privateData			
Service Response optional parameters			
connections			
transferredCallInfo			
security			
privateData			

A.5.7 Call Control Events

A.5.7.1 Bridged

Description: Event, optional parameters	Supported?		Comments
	Yes	No	
Bridged			
localConnectionInfo			
correlatorData			
userData			
servicesPermitted			
mediaCallCharacteristics			
callCharacteristics			
bridgedConnectionInfo			
callLinkageData			
security			
privateData			

A.5.7.2 Call Cleared

Description: Event, optional parameters	Supported?		Comments
	Yes	No	
Call Cleared			
correlatorData			
userData			
mediaCallCharacteristics			
callCharacteristics			
callLinkageData			
security			
privateData			

A.5.7.3 Conferenced

Description: Event, optional parameters	Supported?		Comments
	Yes	No	
Conferenced			
secondaryOldCall			
localConnectionInfo			
correlatorData			
userData			
servicesPermitted			
mediaCallCharacteristics			
callCharacteristics			
security			
privateData			

A.5.7.4 Connection Cleared

This event must be supported if the Basic Telephony profile is supported (A.5.1).

Description: Event, optional parameters	Supported?		Comments
	Yes	No	
Connection Cleared			
localConnectionInfo			
correlatorData			
userData			
chargingInfo			
servicesPermitted			
mediaCallCharateristics			
callCharacteristics			
droppedConnectionInfo			
callLinkageData			
security			
privateData			

A.5.7.5 Delivered

This event must be supported if the Basic Telephony profile is supported (A.5.1).

Description: Event, optional parameters	Supported?		Comments
	Yes	No	
Delivered			
originatingNIDConnection			
localConnectionInfo			
correlatorData			
userData			
servicesPermitted			
networkCallingDevice			
networkCalledDevice			
associatedCallingDevice			
associatedCalledDevice			
mediaCallCharacteristics			
callCharacteristics			
connectionInfo			
security			
privateData			

A.5.7.6 Digits Dialed

Description: Event, optional parameters	Supported?		Comments
	Yes	No	
Digits Dialed			
localConnectionInfo			
correlatorData			
servicesPermitted			
networkCallingDevice			
networkCalledDevice			
associatedCallingDevice			

Description: Event, optional parameters	Supported?		Comments
	Yes	No	
associatedCalledDevice			
diallingConnectionInfo			
callCharacteristics			
security			
privateData			

A.5.7.7 Diverted

Description: Event, optional parameters	Supported?		Comments
	Yes	No	
Diverted			
callingDevice			
calledDevice			
localConnectionInfo			
correlatorData			
userData			
servicesPermitted			
mediaCallCharacteristics			
callCharacteristics			
connectionInfo			
networkCallingDevice			
networkCalledDevice			
associatedCallingDevice			
associatedCalledDevice			
security			
privateData			

A.5.7.8 Established

This event must be supported if the Basic Telephony profile is supported (A.5.1).

Description: Event, optional parameters	Supported?		Comments
	Yes	No	
Established			
originatingNIDConnection			
localConnectionInfo			
correlatorData			
userData			
servicesPermitted			
networkCallingDevice			
networkCalledDevice			
associatedCallingDevice			
associatedCalledDevice			
mediaCallCharacteristics			
callCharacteristics			
establishConnectionInfo			
security			
privateData			

A.5.7.9 Failed

This event must be supported if the Basic Telephony profile is supported (A.5.1).

Description: Event, optional parameters	Supported?		Comments
	Yes	No	
Failed			
originatingNIDConnection			
localConnectionInfo			
correlatorData			
userData			
servicesPermitted			

Description: Event, optional parameters	Supported?		Comments
	Yes	No	
networkCallingDevice			
networkCalledDevice			
associatedCallingDevice			
associatedCalledDevice			
mediaCallCharacteristics			
callCharacteristics			
failedConnectionInfo			
security			
privateData			

A.5.7.10 Held

Description: Event, optional parameters	Supported?		Comments
	Yes	No	
Held			
localConnectionInfo			
correlatorData			
servicesPermitted			
mediaCallCharacteristics			
callCharacteristics			
heldConnectionInfo			
callLinkageData			
security			
privateData			

A.5.7.11 Network Capabilities Changed

Description: Event, optional parameters	Supported?		Comments
	Yes	No	
Network Capabilities Changed			
localConnectionInfo			
correlatorData			
userData			
networkCapability			
servicesPermitted			
mediaCallCharacteristics			
callCharacteristics			
outboundConnectionInfo			
security			
privateData			

A.5.7.12 Network Reached

This event must be supported if the Basic Telephony profile is supported (A.5.1).

Description: Event, optional parameters	Supported?		Comments
	Yes	No	
Network Reached			
originatingNIDConnection			
localConnectionInfo			
correlatorData			
userData			
networkCapability			
servicesPermitted			
mediaCallCharacteristics			
callCharacteristics			
outboundConnectionInfo			
networkCallingDevice			

Description: Event, optional parameters	Supported?		Comments
	Yes	No	
networkCalledDevice			
associatedCallingDevice			
security			
privateData			

A.5.7.13 Offered

Description: Event, optional parameters	Supported?		Comments
	Yes	No	
Offered			
originatingNIDConnection			
localConnectionInfo			
correlatorData			
userData			
servicesPermitted			
networkCallingDevice			
networkCalledDevice			
associatedCallingDevice			
associatedCalledDevice			
mediaCallCharacteristics			
callCharacteristics			
offeredConnectionInfo			
security			
privateData			

A.5.7.14 Originated

This event must be supported if the Basic Telephony profile is supported (A.5.1).

Description: Event, optional parameters	Supported?		Comments
	Yes	No	
Originated			
originatingDevice			
localConnectionInfo			
correlatorData			
servicesPermitted			
networkCallingDevice			
networkCalledDevice			
associatedCallingDevice			
associatedCalledDevice			
mediaCallCharacteristics			
callCharacteristics			
originatedConnectionInfo			
security			
privateData			

A.5.7.15 Queued

Description: Event, optional parameters	Supported?		Comments
	Yes	No	
Queued			
numberQueued			
callsInFront			
localConnectionInfo			
correlatorData			
userData			
servicesPermitted			
networkCallingDevice			

Description: Event, optional parameters	Supported?		Comments
	Yes	No	
networkCalledDevice			
associatedCallingDevice			
associatedCalledDevice			
mediaCallCharacteristics			
callCharacteristics			
queuedConnectionInfo			
security			
privateData			

A.5.7.16 Retrieved

Description: Event, optional parameters	Supported?		Comments
	Yes	No	
Retrieved			
localConnectionInfo			
correlatorData			
servicePermitted			
mediaCallCharacteristics			
callCharacteristics			
retrievedConnectionInfo			
callLinkageData			
security			
privateData			

A.5.7.17 Service Initiated

This event must be supported if the Basic Telephony profile is supported (A.5.1).

Description: Event, optional parameters	Supported?		Comments
	Yes	No	
Service Initiated			
localConnectionInfo			
correlatorData			
servicesPermitted			
mediaCallCharacteristics			
callCharacteristics			
initiatedConnectionInfo			
networkCallingDevice			
networkCalledDevice			
associatedCallingDevice			
security			
privateData			

A.5.7.18 Transferred

Description: Event, optional parameters	Supported?		Comments
	Yes	No	
Transferred			
secondaryOldCall			
localConnectionInfo			
correlatorData			
userData			
chargingInfo			
servicesPermitted			
mediaCallCharacteristics			

Description: Event, optional parameters	Supported?		Comments
	Yes	No	
callCharacteristics			
security			
privateData			

A.5.8 Call Associated Feature Services

A.5.8.1 Associate Data

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Associate Data			
Service Request optional parameters			
accountCode			
authCode			
correlatorData			
callQualifyingData			
security			
privateData			
Service Response optional parameters			
security			
privateData			

A.5.8.2 Cancel Telephony Tones

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Cancel Telephony Tones			
Service Request optional parameters			
security			
privateData			

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Service Response optional parameters			
security			
privateData			

A.5.8.3 Generate Digits

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Generate Digits			
Service Request optional parameters			
digitMode			
toneDuration			
pulseRate			
pauseDuration			
security			
privateData			
Service Response optional parameters			
security			
privateData			

A.5.8.4 Generate Telephony Tones

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Generate Telephony Tones			
Service Request optional parameters			
toneDuration			
security			
privateData			

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Service Response optional parameters			
security			
privateData			

A.5.8.5 Send User Information

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Send User Information			
Service Request optional parameters			
security			
privateData			
Service Response optional parameters			
security			
privateData			

A.5.9 Call Associated Feature Events

A.5.9.1 Call Information

Description: Event, optional parameters	Supported?		Comments
	Yes	No	
Call Information			
callingDevice			
accountInfo			
authorisationCode			
correlatorData			
servicesPermitted			
userData			
callQualifyingData			

Description: Event, optional parameters	Supported?		Comments
	Yes	No	
connectionInfo			
security			
privateData			

A.5.9.2 Charging

Description: Event, optional parameters	Supported?		Comments
	Yes	No	
Charging			
cause			
security			
privateData			

A.5.9.3 Digits Generated

Description: Event, optional parameters	Supported?		Comments
	Yes	No	
Digits Generated			
digitDurationList			
pauseDurationList			
connectionInfo			
security			
privateData			

A.5.9.4 Telephony Tones Generated

Description: Event, optional parameters	Supported?		Comments
	Yes	No	
Telephony Tones Generated			
toneGenerated			
toneFrequency			

Description: Event, optional parameters	Supported?		Comments
	Yes	No	
toneDuration			
pauseDuration			
connectionInfo			
security			
privateData			

A.5.9.5 Service Completion Failure

Description: Event, optional parameters	Supported?		Comments
	Yes	No	
Service Completion Failure			
secondaryCall			
otherDevicesPrimaryCallList			
otherDevicesSecondaryCallList			
mediaCallCharacteristics			
security			
privateData			

A.5.10 Media Attachment Services

A.5.10.1 Attach Media Service

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Attach Media Service			
Service Request optional parameters			
mediaServiceVersion			
mediaServiceInstanceID			
requestedConnectionState			
security			
privateData			

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Service Response optional parameters			
mediaConnection			
mediaDevice			
mediaServiceInstanceID			
mediaConnectionInfo			
security			
privateData			

A.5.10.2 Detach Media Service

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Detach Media Service			
Service Request optional parameters			
security			
privateData			
Service Response optional parameters			
security			
privateData			

A.5.11 Media Attachment Events

A.5.11.1 Media Attached

Description: Event, optional parameters	Supported?		Comments
	Yes	No	
Media Attached			
mediaServiceVersion			
mediaServiceInstanceID			
mediaStreamID			
mediaCallCharacteristics			

Description: Event, optional parameters	Supported?		Comments
	Yes	No	
callCharacteristics			
localConnectionInfo			
mediaConnectionInfo			
security			
privateData			

A.5.11.2 Media Detached

Description: Event, optional parameters	Supported?		Comments
	Yes	No	
Media Detached			
mediaServiceVersion			
mediaServiceInstanceID			
mediaStreamID			
mediaCallCharacteristics			
callCharacteristics			
localConnectionInfo			
mediaConnectionInfo			
security			
privateData			

A.5.12 Routeing Services

A.5.12.1 Route Register

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Route Register			
Service Request optional parameters			
routeingDevice			
requestedRouteingMediaClass			

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
security			
privateData			
Service Response optional parameters			
actualRouteingMediaClass			
security			
privateData			

A.5.12.2 Route Register Abort

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Route Register Abort			
Service Request optional parameters			
security			
privateData			

A.5.12.3 Route Register Cancel

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Route Register Cancel			
Service Request optional parameters			
security			
privateData			
Service Response optional parameters			
security			
privateData			

A.5.12.4 Re-Route

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Re-Route			
Service Request optional parameters			
routeRegisterReqID			
replyTimeout			
correlatorData			
security			
privateData			

A.5.12.5 Route End

This service must be supported if the Routeing Profile is supported (A.5.1).

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Route End			
Service Request optional parameters			
routeRegisterReqID			
errorValue			
correlatorData			
security			
privateData			

A.5.12.6 Route Reject

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Route Reject			
Service Request optional parameters			
routeRegisterReqID			

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
rejectCause			
correlatorData			
security			
privateData			

A.5.12.7 Route Request

This service must be supported if the Routeing Profile is supported (A.5.1).

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Route Request			
Service Request optional parameters			
routeRegisterReqID			
callingDevice			
routeingDevice			
routedCall			
routeSelAlgorithm			
associatedCallingDevice			
associatedCalledDevice			
priority			
replyTimeout			
correlatorData			
mediaCallCharacteristics			
callCharacteristics			
routedCallInfo			
security			
privateData			

A.5.12.8 Route Select

This service must be supported if the Routeing Profile is supported (A.5.1).

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Route Select			
Service Request optional parameters			
routeRegisterReqID			
alternateRoutes			
remainRetries			
routeUsedReq			
correlatorData			
security			
privateData			

A.5.12.9 Route Used

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Route Used			
Service Request optional parameters			
routeRegisterReqID			
callingDevice			
domain			
correlatorData			
security			
privateData			

A.5.13 Physical Device Services

A.5.13.1 Button Press

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Button Press			
Service Request optional parameters			
security			
privateData			
Service Response optional parameters			
security			
privateData			

A.5.13.2 Get Auditory Apparatus Information

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Get Auditory Apparatus Information			
Service Request optional parameters			
auditoryApparatus			
security			
privateData			
Service Response optional parameters			
security			
privateData			

A.5.13.3 Get Button Information

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Get Button Information			
Service Request optional parameters			
button			

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
security			
privateData			
Service Response optional parameters			
security			
privateData			

A.5.13.4 Get Display

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Get Display			
Service Request optional parameters			
displayID			
security			
privateData			
Service Response optional parameters			
security			
privateData			

A.5.13.5 Get HookSwitch Status

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Get HookSwitch Status			
Service Request optional parameters			
hookswitch			
security			
privateData			

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Service Response optional parameters			
security			
privateData			

A.5.13.6 Get Lamp Information

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Get Lamp Information			
Service Request optional parameters			
lamp			
security			
privateData			
Service Response optional parameters			
security			
privateData			

A.5.13.7 Get Lamp Mode

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Get Lamp Mode			
Service Request optional parameters			
lamp			
security			
privateData			
Service Response optional parameters			
lamp			
security			
privateData			

A.5.13.8 Get Message Waiting Indicator

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Get Message Waiting Indicator			
Service Request optional parameters			
security			
privateData			
Service Response optional parameters			
deviceForMessage			
lampIsPresent			
security			
privateData			

A.5.13.9 Get Microphone Gain

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Get Microphone Gain			
Service Request optional parameters			
auditoryApparatus			
security			
privateData			
Service Response optional parameters			
security			
privateData			

A.5.13.10 Get Microphone Mute

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Get Microphone Mute			
Service Request optional parameters			
auditoryApparatus			
security			
privateData			
Service Response optional parameters			
security			
privateData			

A.5.13.11 Get Ringer Status

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Get Ringer Status			
Service Request optional parameters			
ringer			
security			
privateData			
Service Response optional parameters			
security			
privateData			

A.5.13.12 Get Speaker Mute

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Get Speaker Mute			
Service Request optional parameters			
auditoryApparatus			

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
security			
privateData			
Service Response optional parameters			
security			
privateData			

A.5.13.13 Get Speaker Volume

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Get Speaker Volume			
Service Request optional parameters			
auditoryApparatus			
security			
privateData			
Service Response optional parameters			
security			
privateData			

A.5.13.14 Set Button Information

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Set Button Information			
Service Request optional parameters			
buttonLabel			
buttonAssociatedNumber			
security			
privateData			

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Service Response optional parameters			
security			
privateData			

A.5.13.15 Set Display

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Set Display			
Service Request optional parameters			
displayID			
physicalBaseRowNumber			
physicalBaseColumnNumber			
offset			
security			
privateData			
Service Response optional parameters			
security			
privateData			

A.5.13.16 Set HookSwitch Status

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Set HookSwitch Status			
Service Request optional parameters			
security			
privateData			

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Service Response optional parameters			
security			
privateData			

A.5.13.17 Set Lamp Mode

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Set Lamp Mode			
Service Request optional parameters			
lampBrightness			
lampColor			
security			
privateData			
Service Response optional parameters			
security			
privateData			

A.5.13.18 Set Message Waiting Indicator

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Set Message Waiting Indicator			
Service Request optional parameters			
deviceForMessage			
security			
privateData			
Service Response optional parameters			
security			
privateData			

A.5.13.19 Set Microphone Gain

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Set Microphone Gain			
Service Request optional parameters			
security			
privateData			
Service Response optional parameters			
security			
privateData			

A.5.13.20 Set Microphone Mute

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Set Microphone Mute			
Service Request optional parameters			
security			
privateData			
Service Response optional parameters			
security			
privateData			

A.5.13.21 Set Ringer Status

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Set Ringer Status			
Service Request optional parameters			
ringMode			
ringPattern			
ringVolume			

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
security			
privateData			
Service Response optional parameters			
security			
privateData			

A.5.13.22 Set Speaker Mute

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Set Speaker Mute			
Service Request optional parameters			
security			
privateData			
Service Response optional parameters			
security			
privateData			

A.5.13.23 Set Speaker Volume

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Set Speaker Volume			
Service Request optional parameters			
security			
privateData			
Service Response optional parameters			
security			
privateData			

A.5.14 Physical Device Events

A.5.14.1 Button Information

Description: Event, optional parameters	Supported?		Comments
	Yes	No	
Button Information			
buttonLabel			
buttonAssociatedNumber			
buttonPressIndicator			
security			
privateData			

A.5.14.2 Button Press

Description: Event, optional parameters	Supported?		Comments
	Yes	No	
Button Press			
buttonLabel			
buttonAssociatedNumber			
security			
privateData			

A.5.14.3 Display Updated

Description: Event, optional parameters	Supported?		Comments
	Yes	No	
Display Updated			
displayID			
physicalRows			
physicalColumns			
physicalBaseRowNumber			
physicalBaseColumnNumber			

Description: Event, optional parameters	Supported?		Comments
	Yes	No	
characterSet			
security			
privateData			

A.5.14.4 Hookswitch

Description: Event, optional parameters	Supported?		Comments
	Yes	No	
Hookswitch			
security			
privateData			

A.5.14.5 Lamp Mode

Description: Event, optional parameters	Supported?		Comments
	Yes	No	
Lamp Mode			
lampLabel			
lampBrightness			
lampColor			
security			
privateData			

A.5.14.6 Message Waiting

Description: Event, optional parameters	Supported?		Comments
	Yes	No	
Message Waiting			
deviceForMessage			
security			
privateData			

A.5.14.7 Microphone Gain

Description: Event, optional parameters	Supported?		Comments
	Yes	No	
Microphone Gain			
security			
privateData			

A.5.14.8 Microphone Mute

Description: Event, optional parameters	Supported?		Comments
	Yes	No	
Microphone Mute			
security			
privateData			

A.5.14.9 Ringer Status

Description: Event, optional parameters	Supported?		Comments
	Yes	No	
Ringer Status			
ringMode			
ringCount			
ringPattern			
ringVolume			
security			
privateData			

A.5.14.10 Speaker Mute

Description: Event, optional parameters	Supported?		Comments
	Yes	No	
Speaker Mute			
security			
privateData			

A.5.14.11 Speaker Volume

Description: Event, optional parameters	Supported?		Comments
	Yes	No	
Speaker Volume			
timeStamp			
security			
privateData			

A.5.15 Logical Device Services

A.5.15.1 Call Back Non-Call-Related

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Call Back Non-Call-Related			
Service Request optional parameters			
security			
privateData			
Service Response optional parameters			
security			
privateData			

A.5.15.2 Call Back Message Non-Call-Related

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Call Back Message Non-Call-Related			
Service Request optional parameters			
security			
privateData			
Service Response optional parameters			
security			
privateData			

A.5.15.3 Cancel Call Back

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Cancel Call Back			
Service Request optional parameters			
security			
privateData			
Service Response optional parameters			
security			
privateData			

A.5.15.4 Cancel Call Back Message

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Cancel Call Back Message			
Service Request optional parameters			
security			
privateData			

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Service Response optional parameters			
security			
privateData			

A.5.15.5 Get Agent State

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Get Agent State			
Service Request optional parameters			
acdGroup			
security			
privateData			
Service Response optional parameters			
security			
privateData			

A.5.15.6 Get Auto Answer

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Get Auto Answer			
Service Request optional parameters			
security			
privateData			
Service Response optional parameters			
numberOfRings			
security			
privateData			

A.5.15.7 Get Auto Work Mode

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Get Auto Work Mode			
Service Request optional parameters			
security			
privateData			
Service Response optional parameters			
autoWorkInterval			
security			
privateData			

A.5.15.8 Get Caller ID Status

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Get Caller ID Status			
Service Request optional parameters			
security			
privateData			
Service Response optional parameters			
security			
privateData			

A.5.15.9 Get Do Not Disturb

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Get Do Not Disturb			
Service Request optional parameters			
security			
privateData			

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Service Response optional parameters			
callOrigination			
callingDeviceList			
security			
privateData			

A.5.15.10 Get Forwarding

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Get Forwarding			
Service Request optional parameters			
security			
privateData			
Service Response optional parameters			
security			
privateData			

A.5.15.11 Get Last Number Dialed

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Get Last Number Dialed			
Service Request optional parameters			
security			
privateData			
Service Response optional parameters			
security			
privateData			

A.5.15.12 Get Routeing Mode

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Get Routeing Mode			
Service Request optional parameters			
security			
privateData			
Service Response optional parameters			
security			
privateData			

A.5.15.13 Set Agent State

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Set Agent State			
Service Request optional parameters			
agentID			
password			
group			
security			
privateData			
Service Response optional parameters			
pendingAgentState			
security			
privateData			

A.5.15.14 Set Auto Answer

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Set Auto Answer			
Service Request optional parameters			
numberOfRings			
security			
privateData			
Service Response optional parameters			
security			
privateData			

A.5.15.15 Set Auto Work Mode

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Set Auto Work Mode			
Service Request optional parameters			
autoWorkInterval			
security			
privateData			
Service Response optional parameters			
security			
privateData			

A.5.15.16 Set Caller ID Status

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Set Caller ID Status			
Service Request optional parameters			
security			

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
privateData			
Service Response optional parameters			
security			
privateData			

A.5.15.17 Set Do Not Disturb

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Set Do Not Disturb			
Service Request optional parameters			
callOrigination			
callingDeviceList			
security			
privateData			
Service Response optional parameters			
security			
privateData			

A.5.15.18 Set Forwarding

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Set Forwarding			
Service Request optional parameters			
forwardingType			
forwardDN			
ringCount			
security			
privateData			

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Service Response optional parameters			
security			
privateData			

A.5.15.19 Set Routeing Mode

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Set Routeing Mode			
Service Request optional parameters			
security			
privateData			
Service Response optional parameters			
security			
privateData			

A.5.16 Logical Device Events

A.5.16.1 Agent Busy

Description: Event, optional parameters	Supported?		Comments
	Yes	No	
Agent Busy			
agentID			
acdGroup			
pendingAgentState			
cause			
security			
privateData			

A.5.16.2 Agent Logged Off

Description: Event, optional parameters	Supported?		Comments
	Yes	No	
Agent Logged Off			
agentID			
acdGroup			
agentPassword			
cause			
security			
privateData			

A.5.16.3 Agent Logged On

Description: Event, optional parameters	Supported?		Comments
	Yes	No	
Agent Logged On			
agentID			
acdGroup			
agentPassword			
cause			
security			
privateData			

A.5.16.4 Agent Not Ready

Description: Event, optional parameters	Supported?		Comments
	Yes	No	
Agent Not Ready			
agentID			
acdGroup			

Description: Event, optional parameters	Supported?		Comments
	Yes	No	
cause			
security			
privateData			

A.5.16.5 Agent Ready

Description: Event, optional parameters	Supported?		Comments
	Yes	No	
Agent Ready			
agentID			
acdGroup			
cause			
security			
privateData			

A.5.16.6 Agent Working After Call

Description: Event, optional parameters	Supported?		Comments
	Yes	No	
Agent Working After Call			
agentID			
acdGroup			
pendingAgentState			
cause			
security			
privateData			

A.5.16.7 Auto Answer

Description: Event, optional parameters	Supported?		Comments
	Yes	No	
Auto Answer			
numberOfRings			
security			
privateData			

A.5.16.8 Auto Work Mode

Description: Event, optional parameters	Supported?		Comments
	Yes	No	
Auto Work Mode			
security			
privateData			

A.5.16.9 Call Back

Description: Event, optional parameters	Supported?		Comments
	Yes	No	
Call Back			
security			
privateData			

A.5.16.10 Call Back Message

Description: Event, optional parameters	Supported?		Comments
	Yes	No	
Call Back Message			
security			
privateData			

A.5.16.11 Caller ID Status

Description: Event, optional parameters	Supported?		Comments
	Yes	No	
Caller ID Status			
security			
privateData			

A.5.16.12 Do Not Disturb

Description: Event, optional parameters	Supported?		Comments
	Yes	No	
Do Not Disturb			
callOrigination			
callingDeviceList			
security			
privateData			

A.5.16.13 Forwarding

Description: Event, optional parameters	Supported?		Comments
	Yes	No	
Forwarding			
forwardingType			
forwardTo			
forwardDefault			
ringCount			
security			
privateData			

A.5.16.14 Routeing Mode

Description: Event, optional parameters	Supported?		Comments
	Yes	No	
Routeing Mode			
security			
privateData			

A.5.17 Device Maintenance Events

A.5.17.1 Back In Service

Description: Event, optional parameters	Supported?		Comments
	Yes	No	
Back In Service			
cause			
security			
privateData			

A.5.17.2 Device Capabilities Changed

Description: Event, optional parameters	Supported?		Comments
	Yes	No	
Device Capabilities Changed			
cause			
security			
privateData			

A.5.17.3 Out of Service

Description: Event, optional parameters	Supported?		Comments
	Yes	No	
Out of Service			
cause			
security			
privateData			

A.5.18 I/O Services

A.5.18.1 I/O Register

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
I/O Register			
Service Request optional parameters			
ioDevice			
security			
privateData			
Service Response optional parameters			
security			
privateData			

A.5.18.2 I/O Register Abort

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
I/O Register Abort			
Service Request optional parameters			
security			
privateData			

A.5.18.3 I/O Register Cancel

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
I/O Register Cancel			
Service Request optional parameters			
security			
privateData			
Service Response optional parameters			
security			
privateData			

A.5.18.4 Data Path Resumed

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Data Path Resumed			
Service Request optional parameters			
ioRegisterReqID			
security			
privateData			
Service Response optional parameters			
security			
privateData			

A.5.18.5 Data Path Suspended

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Data Path Suspended			
Service Request optional parameters			
ioRegisterReqID			
security			

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
privateData			
Service Response optional parameters			
security			
privateData			

A.5.18.6 Fast Data

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Fast Data			
Service Request optional parameters			
ioRegisterReqID			
dataPathType			
displayAttributes			
security			
privateData			
Service Response optional parameters			
security			
privateData			

A.5.18.7 Resume Data Path

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Resume Data Path			
Service Request optional parameters			
ioRegisterReqID			
security			
privateData			

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Service Response optional parameters			
security			
privateData			

A.5.18.8 Send Broadcast Data

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Send Broadcast Data			
Service Request optional parameters			
dataPathType			
displayAttributes			
security			
privateData			
Service Response optional parameters			
security			
privateData			

A.5.18.9 Send Data

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Send Data			
Service Request optional parameters			
ioRegisterReqID			
displayAttributes			
ioCause			
security			
privateData			

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Service Response optional parameters			
security			
privateData			

A.5.18.10 Send Multicast Data

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Send Multicast Data			
Service Request optional parameters			
ioData			
displayAttributes			
security			
privateData			
Service Response optional parameters			
security			
privateData			

A.5.18.11 Start Data Path

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Start Data Path			
Service Request optional parameters			
ioRegisterReqID			
dataPathDirection			
dataPathType			
displayID			
numberOfCharsToCollect			
terminationCharacter			

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
timeout			
security			
privateData			
Service Response optional parameters			
numberOfCharsToCollect			
terminationCharacter			
timeout			
security			
privateData			

A.5.18.12 Stop Data Path

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Stop Data Path			
Service Request optional parameters			
ioRegisterReqID			
security			
privateData			
Service Response optional parameters			
security			
privateData			

A.5.18.13 Suspend Data Path

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Suspend Data Path			
Service Request optional parameters			
ioRegisterReqID			

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
security			
privateData			
Service Response optional parameters			
security			
privateData			

A.5.19 Data Collection Services

A.5.19.1 Data Collected

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Data Collected			
Service Request optional parameters			
digitsData			
telTonesData			
connectionInfo			
dcollCause			
security			
privateData			
Service Response optional parameters			
security			
privateData			

A.5.19.2 Data Collection Resumed

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Data Collection Resumed			
Service Request optional parameters			
security			
privateData			
Service Response optional parameters			
security			
privateData			

A.5.19.3 Data Collection Suspended

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Data Collection Suspended			
Service Request optional parameters			
security			
privateData			
Service Response optional parameters			
security			
privateData			

A.5.19.4 Resume Data Collection

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Resume Data Collection			
Service Request optional parameters			
security			
privateData			

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Service Response optional parameters			
security			
privateData			

A.5.19.5 Start Data Collection

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Start Data Collection			
Service Request optional parameters			
dataCollType			
digitsReportingCriteria			
security			
privateData			
Service Response optional parameters			
security			
privateData			

A.5.19.6 Stop Data Collection

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Stop Data Collection			
Service Request optional parameters			
security			
privateData			
Service Response optional parameters			
security			
privateData			

A.5.19.7 Suspend Data Collection

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Suspend Data Collection			
Service Request optional parameters			
security			
privateData			
Service Response optional parameters			
security			
privateData			

A.5.20 Voice Unit Services

A.5.20.1 Concatenate Message

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Concatenate Message			
Service Request optional parameters			
security			
privateData			
Service Response optional parameters			
security			
privateData			

A.5.20.2 Delete Message

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Delete Message			
Service Request optional parameters			
security			
privateData			

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Service Response optional parameters			
security			
privateData			

A.5.20.3 Play Message

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Play Message			
Service Request optional parameters			
duration			
termination			
security			
privateData			
Service Response optional parameters			
security			
privateData			

A.5.20.4 Query Voice Attribute

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Query Voice Attribute			
Service Request optional parameters			
connection			
security			
privateData			
Service Response optional parameters			
security			
privateData			

A.5.20.5 Record Message

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Record Message			
Service Request optional parameters			
samplingRate			
encodingAlgorithm			
maxDuration			
termination			
security			
privateData			
Service Response optional parameters			
security			
privateData			

A.5.20.6 Reposition

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Reposition			
Service Request optional parameters			
messageToReposition			
security			
privateData			
Service Response optional parameters			
security			
privateData			

A.5.20.7 Resume

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Resume			
Service Request optional parameters			
messageToResume			
duration			
security			
privateData			
Service Response optional parameters			
security			
privateData			

A.5.20.8 Review

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Review			
Service Request optional parameters			
messageToReview			
security			
privateData			
Service Response optional parameters			
security			
privateData			

A.5.20.9 Set Voice Attribute

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Set Voice Attribute			
Service Request optional parameters			
message			
security			
privateData			
Service Response optional parameters			
security			
privateData			

A.5.20.10 Stop

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Stop			
Service Request optional parameters			
messageToBeStopped			
security			
privateData			
Service Response optional parameters			
security			
privateData			

A.5.20.11 Suspend

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Suspend			
Service Request optional parameters			
message			

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
security			
privateData			
Service Response optional parameters			
security			
privateData			

A.5.20.12 Synthesize Message

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Synthesize Message			
Service Request optional parameters			
control			
security			
privateData			
Service Response optional parameters			
security			
privateData			

A.5.21 Voice Unit Events

A.5.21.1 Play

Description: Event, optional parameters	Supported?		Comments
	Yes	No	
Play			
length			
currentPosition			
speed			
cause			

Description: Event, optional parameters	Supported?		Comments
	Yes	No	
servicesPermitted			
security			
privateData			

A.5.21.2 Record

Description: Event, optional parameters	Supported?		Comments
	Yes	No	
Record			
length			
currentPosition			
cause			
servicesPermitted			
security			
privateData			

A.5.21.3 Review

Description: Event, optional parameters	Supported?		Comments
	Yes	No	
Review			
length			
currentPosition			
cause			
servicesPermitted			
security			
privateData			

A.5.21.4 Stop

Description: Event, optional parameters	Supported?		Comments
	Yes	No	
Stop			
length			
currentPosition			
cause			
servicesPermitted			
security			
privateData			

A.5.21.5 Suspend Play

Description: Event, optional parameters	Supported?		Comments
	Yes	No	
Suspend Play			
length			
currentPosition			
cause			
servicesPermitted			
security			
privateData			

A.5.21.6 Suspend Record

Description: Event, optional parameters	Supported?		Comments
	Yes	No	
Suspend Record			
length			
currentPosition			
cause			

Description: Event, optional parameters	Supported?		Comments
	Yes	No	
servicesPermitted			
security			
privateData			

A.5.21.7 Voice Attribute Changed

Description: Event, optional parameters	Supported?		Comments
	Yes	No	
Voice Attribute Changed			
playVolume			
recordingGain			
speed			
currentPosition			
cause			
security			
privateData			

A.5.22 Call Detail Record (CDR) Services

A.5.22.1 Call Detail Records Notification

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Call Detail Records Notification			
Service Request optional parameters			
cdrReason			
security			
privateData			
Service Response optional parameters			
security			
privateData			

A.5.22.2 Call Detail Records Report

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Call Detail Records Report			
Service Request optional parameters			
security			
privateData			
Service Response optional parameters			
security			
privateData			

A.5.22.3 Send Stored Call Detail Records

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Send Stored Call Detail Records			
Service Request optional parameters			
timePeriod			
security			
privateData			
Service Response optional parameters			
security			
privateData			

A.5.22.4 Start Call Detail Records Transmission

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Start Call Detail Records Transmission			
Service Request optional parameters			
security			

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
privateData			
Service Response optional parameters			
security			
privateData			

A.5.22.5 Stop Call Detail Records Transmission

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Stop Call Detail Records Transmission			
Service Request optional parameters			
cdrTermReason			
security			
privateData			
Service Response optional parameters			
security			
privateData			

A.5.23 Vendor Specific Extension Services

A.5.23.1 Escape Register

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Escape Register			
Service Request optional parameters			
security			
privateData			
Service Response optional parameters			
security			
privateData			

A.5.23.2 Escape Register Abort

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Escape Register Abort			
Service Request optional parameters			
security			
privateData			

A.5.23.3 Escape Register Cancel

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Escape Register Cancel			
Service Request optional parameters			
security			
privateData			
Service Response optional parameters			
security			
privateData			

A.5.23.4 Escape

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Escape			
Service Request optional parameters			
escapeRegisterID			
security			
Service Response optional parameters			
security			
privateData			

A.5.23.5 Private Data Version Selection

Description: Service, optional parameters	Supported?		Comments
	Yes	No	
Private Data Version Selection			
Service Response optional parameters			
security			
privateData			

A.5.24 Vendor Specific Extension Events

A.5.24.1 Private Event

Description: Event, optional parameters	Supported?		Comments
	Yes	No	
Private Event			
security			

Annex B

(informative)

Examples of Device Identifier Element Encodings

This annex illustrates how deviceIDs are encoded in the XML protocol.

The CSTA Device Identifiers (deviceIDs) element type is associated with many different parameters in CSTA services and events.

In order to enhance readability, XML attributes are used to qualify the different possible types of deviceIDs.

B.1 Simple DeviceID (no attributes)

```
<csta:deviceID>4711</csta:deviceID>
```

B.2 DeviceID typed as a dialing number

```
<csta:deviceID typeOfNumber="dialingNumber">4711</csta:deviceID>
```

B.3 DeviceID typed as a device number

```
<csta:deviceID typeOfNumber="deviceNumber">100</csta:deviceID>
```

B.4 DeviceID typed as a explicit national number

```
<csta:deviceID typeOfNumber="explicitPublic:national">18005551212</csta:deviceID>
```

B.5 DeviceID typed as other type of numbering plan

```
<csta:deviceID typeOfNumber="other">abc=</csta:deviceID>
```

B.6 DeviceID media characteristics

```
<csta:deviceID mediaClass="voice audio other"
switchingSubDomainCCIEType="isdn" bitRate="variable">4711</csta:deviceID>
```

B.7 DeviceID including all attributes

```
<csta:deviceID typeOfNumber="implicitPublic" bitRate="constant"
connectionRate="0" delayTolerance="10" mediaClass="voice"
switchingSubDomainCCIEType="isdn"
switchingSubDomainInformationElements="abcdef==">123456</csta:deviceID>
```


Annex C

(informative)

Examples of Connection ID Encodings

This annex illustrates how connectionIDs are encoded in the XML protocol.

C.1 Fully Specified Connection ID

A CSTA connectionID, unless otherwise specified in ECMA-269, contains a callID and a deviceID portion. This is called a fully specified connectionID.

```
<camponConnection>  
  <callID>1</callID>  
  <deviceID>22222</deviceID>  
</camponConnection>
```

C.2 CallID only ConnectionID

In some cases, as specified in ECMA-269, a callID-only connectionID can be used (e.g., Clear Call service).

```
<callToBeAnswered>  
  <callID>1</callID>  
</callToBeAnswered>
```

C.3 DeviceID only ConnectionID

In some cases, as specified in ECMA-269, a device-only connectionID can be used (e.g., Answer Call service).

```
<connectionToBeCleared>  
  <deviceID>22222</deviceID>  
</connectionToBeCleared>
```


Annex D

(informative)

Examples of CSTA-specific XML

This annex illustrates examples of CSTA-specific information in XML instance documents.

D.1 Make Call service request - with only mandatory elements

This example illustrates a CSTA Make Call service with just the mandatory elements provided.

```
<?xml version="1.0" encoding="UTF-8"?>
<MakeCall xmlns="http://www.ecma.ch/standards/ecma-323/csta">
  <callingDevice>22343</callingDevice>
  <calledDirectoryNumber>18005551212</calledDirectoryNumber>
</MakeCall>
```

D.2 Make Call service request - with optional elements

This example illustrates some of the optional elements specified in the Make Call service.

```
<?xml version="1.0" encoding="UTF-8"?>
<MakeCall xmlns="http://www.ecma.ch/standards/ecma-323/csta">
  <callingDevice>22343</callingDevice>
  <calledDirectoryNumber>18005551212</calledDirectoryNumber>
  <accountCode>0001</accountCode>
  <authCode>0001</authCode>
  <autoOriginate>prompt</autoOriginate>
  <correlatorData>0001</correlatorData>
  <userData>0001</userData>
  <callCharacteristics>
    <priorityCall>true</priorityCall>
  </callCharacteristics>
  <mediaCallCharacteristics>
    <mediaClass>
      <voice>true</voice>
    </mediaClass>
  </mediaCallCharacteristics>
  <callingConnectionInfo>
    <flowDirection>transmitAndReceive</flowDirection>
    <numberOfChannels>2</numberOfChannels>
  </callingConnectionInfo>
  <extensions>
    <security>
      <messageSequenceNumber>123</messageSequenceNumber>
      <securityInfo>
        <string>0001</string>
      </securityInfo>
    </security>
    <privateData>
      <string>0001</string>
    </privateData>
  </extensions>
</MakeCall>
```

D.3 Make Call - Positive Acknowledgement

This example illustrates a positive acknowledgement to the Make Call service.

```
<?xml version="1.0" encoding="UTF-8"?>
<MakeCallResponse xmlns="http://www.ecma.ch/standards/ecma-323/csta">
  <callingDevice>
    <callID>1</callID>
    <deviceID>22343</deviceID>
  </callingDevice>
</MakeCallResponse>
```

D.4 Make Call - Negative Acknowledgement

This example illustrates a negative acknowledgement to the Make Call service.

```
<?xml version="1.0" encoding="UTF-8"?>
<CSTAErroCode xmlns="http://www.ecma.ch/standards/ecma-323/csta">
  <operation>invalidCallingDeviceID</operation>
</CSTAErroCode>
```

D.5 Delivered Event

This example illustrates a Delivered event.

```
<?xml version="1.0" encoding="UTF-8"?>
<DeliveredEvent xmlns="http://www.ecma.ch/standards/ecma-323/csta">
  <monitorCrossRefID>1</monitorCrossRefID>
  <connection>
    <callID>1</callID>
    <deviceID>33333</deviceID>
  </connection>
  <alertingDevice>
    <deviceIdentifier>33333</deviceIdentifier>
  </alertingDevice>
  <callingDevice>
    <deviceIdentifier>22222</deviceIdentifier>
  </callingDevice>
  <calledDevice>
    <deviceIdentifier>33333</deviceIdentifier>
  </calledDevice>
  <lastRedirectionDevice>
    <notRequired/>
  </lastRedirectionDevice>
  <cause>normal</cause>
</DeliveredEvent>
```

Annex E

(informative)

CSTA XML with SOAP

This annex illustrates examples of CSTA XML instance documents using the Simple Object Access Protocol (SOAP). These examples do not assume that HTTP is used as the transport.

Since multiple requests can be outstanding for a particular CSTA service, information is added to the SOAP Header element to provide the necessary correlation information. (When SOAP is used with HTTP, the additional header information is not necessary because there can only be one outstanding HTTP request per connection as illustrated in the following annex).

E.1 CSTA service request

This example illustrates a CSTA Make Call service with SOAP. The invokeID is provided in the SOAP header.

```
<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta">
  <SOAP-ENV:Header>
    <invokeID>1</invokeID>
  </SOAP-ENV:Header>
  <SOAP-ENV:Body>
    <csta:MakeCall>
      <csta:callingDevice>4711</csta:callingDevice>
      <csta:calledDirectoryNumber>4712</csta:calledDirectoryNumber>
    </csta:MakeCall>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

E.2 CSTA service response (positive acknowledgement)

This example illustrates a positive acknowledgement to the Make Call service. The invokeID in the SOAP header can be used to correlate the response to the request.

```
<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta">
  <SOAP-ENV:Header>
    <invokeID>1</invokeID>
  </SOAP-ENV:Header>
  <SOAP-ENV:Body>
    <csta:MakeCallResponse>
      <csta:callingDevice>
        <csta:callID>1</callID>
        <csta:deviceID>22343</deviceID>
      </csta:callingDevice>
    </csta:MakeCallResponse>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

E.3 CSTA service response (negative acknowledgement)

This example illustrates a negative acknowledgement to the Make Call service. The SOAP FAULT element is used to provide the error information. The invokeID in the SOAP header can be used to correlate the response to the request.

The faultcode indicates that the failure is due to incorrect information provided by the client. The CSTAErrorCode indicates that the error category is "Operation" and the specific error is "Invalid Called DeviceID".

```
<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta">
  <SOAP-ENV:Header>
    <invokeID>1</invokeID>
  </SOAP-ENV:Header>
  <SOAP-ENV:Body>
    <SOAP-ENV:Fault>
      <SOAP-ENV:faultcode>Client</SOAP-ENV:faultcode>
      <SOAP-ENV:faultstring>CSTA Error</SOAP-ENV:faultstring>
      <SOAP-ENV:detail>
        <csta:CSTAErrorCode
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta">
          <csta:operation>invalidCalledDeviceID</csta:operation>
        </csta:CSTAErrorCode>
      </SOAP-ENV:detail>
    </SOAP-ENV:Fault>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

E.4 Delivered Event

This example illustrates a Delivered event.

```
<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta">
  <SOAP-ENV:Body>
    <csta:DeliveredEvent>
      <csta:monitorCrossRefID>1</csta:monitorCrossrefID>
      <csta:connection>
        <csta:callID>1</csta:callID>
        <csta:deviceID>33333</csta:deviceID>
      </csta:connection>
      <csta:alertingDevice>
        <csta:deviceIdentifier>33333</csta:deviceIdentifier>
      </csta:alertingDevice>
      <csta:callingDevice>
        <csta:deviceIdentifier>22222</csta:deviceIdentifier>
      </csta:callingDevice>
      <csta:calledDevice>
        <csta:deviceIdentifier>33333</csta:deviceIdentifier>
      </csta:calledDevice>
      <csta:lastRedirectionDevice>
        <csta:notRequired/>
      </csta:lastRedirectionDevice>
      <csta:cause>csta:normal</csta:cause>
    </csta:DeliveredEvent>
```

```
</SOAP-ENV:Body>  
</SOAP-ENV:Envelope>
```


Annex F

(informative)

CSTA XML with SOAP with HTTP

This annex illustrates examples of CSTA XML instance documents using the Simple Object Access Protocol (SOAP) with HTTP.

Since, with HTTP, only one request can be outstanding for a particular CSTA service, header information is not needed to correlate the service request to the service response. (The single response can be correlated to the request via the method name (e.g. MakeCall and MakeCallRequest)).

F.1 CSTA service request

This example illustrates a CSTA Make Call service with SOAP and HTTP. The first group of lines represent the HTTP header information.

```
POST /anyserver HTTP/1.1
Host: anyweb.com
Content-Type: text/xml
Content-Length: nnn
SOAPAction: Makecall
```

```
<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta">
  <SOAP-ENV:Body>
    <csta:MakeCall>
      <csta:callingDevice>4711</csta:callingDevice>
      <csta:calledDirectoryNumber>4712</csta:calledDirectoryNumber>
    </csta:MakeCall>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

F.2 CSTA service response (positive acknowledgement)

This example illustrates a positive acknowledgement to the Make Call service. The first group of lines represent the HTTP header information. The 200 response code indicates a successful response.

```
HTTP/1.1 200 OK
Content-Type: text/xml
Content-Length: nnnn
```

```
<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta">
  <SOAP-ENV:Body>
    <csta:MakeCallResponse>
      <csta:callingDevice>
        <csta:callID>1</callID>
        <csta:deviceID>22343</deviceID>
      </csta:callingDevice>
    </csta:MakeCallResponse>
  </SOAP-ENV:Body>
```

```
</SOAP-ENV:Envelope>
```

F.3 CSTA service response (negative acknowledgement)

This example illustrates a negative acknowledgement to the Make Call service. The SOAP FAULT element is used to provide the error information. The 500 response code indicates that the request failed.

```
HTTP/1.1 500 Internal Server Error
```

```
Content-Type: text/xml
```

```
Content-Length: nnnn
```

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

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
```

```
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta">
```

```
  <SOAP-ENV:Body>
```

```
    <SOAP-ENV:Fault>
```

```
      <SOAP-ENV:faultcode>Client</SOAP-ENV:faultcode>
```

```
      <SOAP-ENV:faultstring>CSTA Error</SOAP-ENV:faultstring>
```

```
      <SOAP-ENV:detail>
```

```
        <csta:CSTAErrorCode
```

```
          xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta">
```

```
            <csta:operation>invalidCalledDeviceID</csta:operation>
```

```
          </csta:CSTAErrorCode>
```

```
        </SOAP-ENV:detail>
```

```
      </SOAP-ENV:Fault>
```

```
    </SOAP-ENV:Body>
```

```
</SOAP-ENV:Envelope>
```

F.4 Delivered Event

This example illustrates a Delivered event. For CSTA events, the SOAP proxy (in the client machine) becomes a type of SOAP server and does not generate any response to the CSTA event (One-way).

```
HTTP/1.1 204 OK
```

```
Content-Type: text/xml
```

```
Content-Length: nnnn
```

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

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
```

```
xmlns:csta="http://www.ecma.ch/standards/ecma-323/csta">
```

```
  <SOAP-ENV:Body>
```

```
    <csta:DeliveredEvent>
```

```
      <csta:monitorCrossRefID>1</csta:monitorCrossrefID>
```

```
      <csta:connection>
```

```
        <csta:callID>1</csta:callID>
```

```
        <csta:deviceID>33333</csta:deviceID>
```

```
      </csta:connection>
```

```
      <csta:alertingDevice>
```

```
        <csta:deviceIdentifier>33333</csta:deviceIdentifier>
```

```
      </csta:alertingDevice>
```

```
      <csta:callingDevice>
```

```
        <csta:deviceIdentifier>22222</csta:deviceIdentifier>
```

```
      </csta:callingDevice>
```

```
      <csta:calledDevice>
```

```
        <csta:deviceIdentifier>33333</csta:deviceIdentifier>
```



```
</csta:calledDevice>  
<csta:lastRedirectionDevice>  
  <csta:notRequired/>  
</csta:lastRedirectionDevice>  
<csta:cause>csta:normal</csta:cause>  
</csta:DeliveredEvent  
</SOAP-ENV:Body>  
</SOAP-ENV:Envelope>
```


Annex G
(informative)

CSTA XML over TCP

This annex illustrates how CSTA XML instance documents are transported over TCP/IP.

G.1 Message Format

Each XML instance message that is sent across a TCP/IP stream is preceded by a message prefix. This prefix consists of a two byte header and a two byte length component as illustrated below..

0	1	2	3	4...
Header		Length		message body

- the header is two bytes. It indicates the format of the XML instance message. The following header formats are defined:
 - 00 - indicates that the message body consists of a CSTA XML instance message without SOAP. As illustrated in G.2, the message body consists of an Invoke ID component followed by an XML instance message.
 - 01 - indicates that the message body consists of a SOAP enveloped CSTA XML instance message as illustrated in G.3.
- the length is defined as the length of the full message, i.e., the length of the message body plus the length of the message prefix.

There is one XML instance message (CSTA service request, service response, or event) per TCP packet.

The XML instance message is encoded as ASCII text. White space and line feeds are allowed (and ignored) between XML statements.

G.2 TCP without SOAP

When TCP is used to transport a CSTA XML instance message without SOAP, a four byte Invoke ID is provided in order to correlate CSTA service request with response messages. This Invoke ID is provided in the message body that precedes the XML instance message as illustrated in the following figure.

0	1	2	3	4	5	6	7	8...
0	0	Length		Invoke ID			XML message body	

- Invoke ID is four bytes. It is used to correlate a CSTA service request with a service response. It is encoded as four ASCII numerical characters. A new Invoke ID is created in a CSTA service request message and is repeated in the corresponding service (positive or fault) response message. A value of "9999" is used for CSTA events.
- XML Message Body. Examples of the XML message body content are provided in annex D.

G.3 TCP with SOAP

When TCP is used to transport a CSTA XML instance message with SOAP, the message body consists of the SOAP enveloped CSTA XML instance message as illustrated in the following figure.

0	1	2	3	4...
0	1	Length	SOAP XML message body	

- SOAP XML Message - Examples of the SOAP XML message body content are provided in annex E. Note that the Invoke ID is contained in the SOAP header as described in annex E.

Free printed copies can be ordered from:

ECMA

114 Rue du Rhône

CH-1204 Geneva

Switzerland

Fax: +41 22 849.60.01

Internet: documents@ecma.ch

Files of this Standard can be freely downloaded from the ECMA web site (www.ecma.ch). This site gives full information on ECMA, ECMA activities, ECMA Standards and Technical Reports.

ECMA
114 Rue du Rhône
CH-1204 Geneva
Switzerland

See inside cover page for obtaining further soft or hard copies.