Standard ECMA-218
December 1994

ECMA

Standardizing Information and Communication Systems

Protocol for Computer Supported Telecommunications Applications (CSTA) Phase II

Phone: +41 22 849.60.00 - **Fax:** +41 22 849.60.01 - **URL:** http://www.ecma.ch - **Internet:** helpdesk@ecma.ch

Standard ECMA-218
December 1994

ECMA

Standardizing Information and Communication Systems

Protocol for Computer Supported Telecommunications Applications (CSTA) Phase II

Phone: +41 22 849.60.00 - **Fax:** +41 22 849.60.01 - **URL:** http://www.ecma.ch - **Internet:** helpdesk@ecma.ch

Brief History

This Standard defines Phase II of the Protocol for Computer Supported Telecommunications Applications (CSTA) for OSI Layer 7 communication between a computing network and a telecommunications network. This Standard and its companion Standard ECMA-217 ***Services for Computer Supported Telecommunications Applications (CSTA) Phase II*** reflect agreements of ECMA member companies on Phase II of the standards for CSTA. Additional phases are anticipated. This Standard is based on the practical experience of ECMA member companies and represents a pragmatic and widely-based consensus.

This Standard takes direction from Technical Report ECMA TR/52 ***Computer Supported Telecommunications Applications***.

Phase II of CSTA extends the previous Phase I standard in major theme directions as well as numerous details. Major areas of advancement include:

- the addition of explicit application context negotiation mechanisms;
- the addition of I/O services;
- the addition of Special Resource Functions and, particularly, Voice Unit services;
- new and/or enhanced services and event reports for commonly used call control and monitoring applications; new services include Single Step Transfer, Single Step Conference, Call Park and Send DTMF Signals.

The Phase II CSTA standards are not fully backwards compatible with the Phase I standards. Although backwards compatibility is an important consideration and has been maintained whenever possible, the addition of new parameters in certain services and events, as well as the deletion of unused Phase I services and the addition of entirely new Phase II services and events, did not allow complete backwards compatibility.

This Standard is dedicated to the memory of Terry Wuerfel.

This ECMA Standard has been adopted by the ECMA General Assembly of December 1994.

Usage Note regarding removing Year 2000 (Y2K) risks in the CSTA Phase I and Phase II Protocol Standards

1 January 1999

In both ITU-T Recs. X.208 (ASN.1 1988) and X.680 (ASN.1 1994), the definition of UTCTime, which is used in the CSTA Phase I and Phase II Protocol Standards (ECMA-180 and ECMA-218 respectively), is not Y2K-safe. The Year field (the YY field) is represented by a two-digit string, with no accompanying text mandating a Y2K-safe interpretation.

For the CSTA Phase I and Phase II Protocol Standards, the following interpretation of the Year field is mandated :

- If the YY component is in the range [00-49], the century is determined to be "20" (i.e. the year is 2000 to 2049);
- If the YY component is in the range [50-99], the century is determined to be "19" (i.e. the year is 1950 to 1999).

The risk has been removed in the CSTA Phase III Protocol Standard (ECMA-285) by replacing the references to "UTCTime" with references to "GeneralizedTime", which is Y2K-safe.

List of corrected errata for ECMA-218

11 December 1996

Summary

Following is a summary of the errors detected and corrected in Standard ECMA-218, Protocol for Computer Supported Telecommunications Applications (CSTA) Phase II. These results were obtained by using the OSS and SNACC ASN.1 compilers.

All definitions

- Compilers complained about “icd-ecma(0012)” statements, therefore “00” has been removed from the definitions

Corrected:

CSTA-application-context-information
{ iso(1) identified-organization(3) icd-ecma(12)
standard(0) csta2(218) application-context-information(200) }

Original:

CSTA-application-context-information
{ iso(1) identified-organization(3) icd-ecma(0012)
standard(0) csta2(218) application-context-information(200) }

Multiple definitions

- Many of the definitions referred to “ECMA-180” instead of “ECMA-218”

Corrected:

CSTA-route-end-request
{ iso(1) identified-organization(3) icd-ecma(12)
standard(0) csta2(218) route-end-request(85) }
CSTACommonArguments, CSTAPrivateData FROM CSTA-extension-types
{ iso(1) identified-organization(3) icd-ecma(12)
standard(0) csta2(218) extension-types(129) };
CSTA-change-monitor-filter
{ iso(1) identified-organization(3) icd-ecma(12)
standard(0) csta2(218) change-monitor-filter(102) }
CSTAObject FROM CSTA-switching-function-objects
{ iso(1) identified-organization(3) icd-ecma(12)
standard(0) csta2(218) switching-function-objects(122) }
CSTA-switching-function-objects
{ iso(1) identified-organization(3) icd-ecma(12)
standard(0) csta2(218)switching-function-objects(122) }

Original:

CSTA-route-end-request
{ iso(1) identified-organization(3) icd-ecma(0012)
standard(0) csta(180) route-end-request(85) }
CSTACommonArguments, CSTAPrivateData FROM CSTA-extension-types
{ iso(1) identified-organization(3) icd-ecma(0012)
standard(0) csta(180) version1(1) extension-types(129) };
CSTA-change-monitor-filter
{ iso(1) identified-organization(3) icd-ecma(0012)
standard(0) csta(180) version1 (1) change-monitor-filter(102) }

```
CSTAObject FROM CSTA-switching-function-objects
    { iso( 1) identified-organization( 3) icd-ecma( 0012)
        standard( 0) csta( 180) switching-function-objects( 122) }
CSTA-switching-function-objects
    { iso( 1) identified-organization( 3) icd-ecma( 0012)
        standard( 0) csta( 180) switching-function-objects( 122) }
```

Multiple Sections

- "BITSTRING" should be "BIT STRING"

Corrected:

```
CSTAVersion ::= BIT STRING
SwitchingFunctionServices ::= BIT STRING
EventReportServices ::= BIT STRING
ComputingFunctionServices ::= BIT STRING
BidirectionalServices ::= BIT STRING
StatusReportingServices ::= BIT STRING
InputOutputServices ::= BIT STRING
VoiceUnitServices ::= BIT STRING
DeviceClass ::= BIT STRING
```

Original:

```
CSTAVersion ::= BITSTRING
SwitchingFunctionServices ::= BITSTRING
EventReportServices ::= BITSTRING
ComputingFunctionServices ::= BITSTRING
BidirectionalServices ::= BITSTRING
StatusReportingServices ::= BITSTRING
InputOutputServices ::= BITSTRING
VoiceUnitServices ::= BITSTRING
DeviceClass ::= BITSTRING
```

Section 8.2.1, Encoding of application context information

- Comma missing after reRouteRequest

Corrected:

```
ComputingFunctionServices ::= BIT STRING
    {routeRequest               (0),
     reRouteRequest             (1),
     routeSelectRequest         (2),
     routeUsedRequest           (3),
     routeEndRequest            (4) }
```

Original:

```
ComputingFunctionServices ::= BIT STRING
    {routeRequest               (0),
     reRouteRequest             (1)
     routeSelectRequest         (2),
     routeUsedRequest           (3),
     routeEndRequest            (4) }
```

Section 9.3, Associate data

- AssociateDataArgument/Correlator Data misspelled
- Extra brace at end of SEQUENCE statement for AssociateDataArgument

Corrected:

```
correlatorData    [2] IMPLICIT CorrelatorData    OPTIONAL,
extensions        CSTACommonArguments            OPTIONAL}
```

Original:

```
correlatorData    [2] IMPLICIT CorellatorData    OPTIONAL,
extensions         CSTACommonArguments           OPTIONAL} }
```

Section 9.8, Consultation call

- Missing comma after AuthCode

Corrected:

AccountInfo, AuthCode, CorrelatorData, DeviceProfile FROM CSTA-device-feature-types

Original:

AccountInfo, AuthCode CorrelatorData, DeviceProfile FROM CSTA-device-feature-types

Section 9.17, Send DTMF tones

- Removed specific characters from IA5String type (or need to define a new type of just these characters)

Corrected:

```
{connectionToSendTones  ConnectionID,
    charactersToSend    IA5String,
    toneDuration        [0] IMPLICIT INTEGER  OPTIONAL,
```

Original:

```
{connectionToSendTones  ConnectionID,
    charactersToSend    IA5String                {0,1,2,3,4,5,6,7,8,9,*,#,A,B,C,D},
    toneDuration        [0] IMPLICIT INTEGER  OPTIONAL,
```

Section 9.20, Single step transfer

- SingleStepTransResult should be SEQUENCE

Corrected:

```
SingleStepTransResult ::=
    SEQUENCE
    {transferedCall     ConnectionID            OPTIONAL,
```

Original:

```
SingleStepTransResult ::=
    {transferedCall     ConnectionID            OPTIONAL,
```

Section 10.1.10, Originated

- OriginatingDevice (SubjectDeviceID) missing

Corrected:

CalledDeviceID, CallingDeviceID, SubjectDeviceID FROM CSTA-device-identifiers

Original:

CalledDeviceID, CallingDeviceID FROM CSTA-device-identifiers

Section 10.3.1, Agent busy

- Missing agentGroup from AgentBusyEvent sequence

Corrected:

Add to IMPORTS:

AgentID, AgentGroup FROM CSTA-device-feature-types

```
AgentBusyEvent ::=
        SEQUENCE
        {agentDevice    SubjectDeviceID,
         agentID        [10] IMPLICIT   AgentID         OPTIONAL,
         agentGroup     AgentGroup                      OPTIONAL,
         cause          EventCause                      OPTIONAL}
```

Original:

```
AgentBusyEvent ::=
        SEQUENCE
        {agentDevice    SubjectDeviceID,
         agentID        AgentID                 OPTIONAL,
         cause          EventCause              OPTIONAL}
```

Sections 10.3.2 and 10.3.3, Agent logged on and Agent logged off

- Conflict between tag for agentID and DeviceID, dialingNumber

Corrected:

```
        SEQUENCE
        {agentDevice    SubjectDeviceID,
         agentID        [10] IMPLICIT   AgentID         OPTIONAL,
         agentGroup     AgentGroup                      OPTIONAL,
```

Original:

```
        SEQUENCE
        {agentDevice    SubjectDeviceID,
         agentID        [0] IMPLICIT    AgentID         OPTIONAL,
         agentGroup     AgentGroup                      OPTIONAL,
```

Section 10.3.6, Agent working after call

- Missing agentGroup from WorkingAfterCallEvent sequence

Corrected:

Add to IMPORTS:

AgentID, AgentGroup FROM CSTA-device-feature-types

```
WorkingAfterCallEvent ::=
        SEQUENCE
        {agentDevice    SubjectDeviceID,
         agentID        [10] IMPLICIT   AgentID         OPTIONAL,
         agentGroup     AgentGroup                      OPTIONAL,
         cause          EventCause                      OPTIONAL}
```

Original:

```
WorkingAfterCallEvent ::=
        SEQUENCE
        {agentDevice    SubjectDeviceID,
         agentID        AgentID                 OPTIONAL,
         cause          EventCause              OPTIONAL}
```

Section 11.1, Route request

- DeviceProfile missing from IMPORTS

Corrected:

CorrelatorData, DeviceClass, DeviceProfile, SelectValue, PriorityValue, RoutingCrossRefID

Original:

CorrelatorData, DeviceClass, SelectValue, PriorityValue, RoutingCrossRefID

Section 12.1, Escape

- "CSTA-escape-service" should be "escape-service" in header

Corrected:

```
{ iso( 1) identified-organization( 3) icd-ecma( 12)
        standard( 0) csta2( 218) escape-service( 91) }
```

Original:

```
{ iso( 1) identified-organization( 3) icd-ecma( 0012)
        standard( 0) csta2( 218) CSTA-escape-service( 91) }
```

Section 13.1, Monitor start

- CSTACommonArguments misspelled

Corrected:

```
extensions          CSTACommonArguments     OPTIONAL}
```

Original:

```
extensions          CSTACommomArguments     OPTIONAL}
```

Section 14.1, Start data path

- "IA5STRING" should be "IA5String", in argument and result

Corrected:

```
terminationCharacter        IA5String                   OPTIONAL,
```

Original:

```
terminationCharacter        IA5STRING                   OPTIONAL,
```

Section 14.2, Stop data path

- "RRORS" should be "ERRORS" in stopDataPath

Corrected:

```
ERRORS  {universalFailure}
```

Original:

```
RRORS   {universalFailure}
```

Section 15.1, Concatenate Message

- Tag conflicted between CHOICE elements, because MessageIDList should be marked as IMPLICIT

Corrected:

```
ConcatenateMessageArgument ::=
        CHOICE
        {messagesToConcatenate  [0]IMPLICIT     MessageIDList,
         SEQUENCE
                {messagesToConcatenate  MessageIDList,
                 extensions             CSTACommonArguments} }
```

Original:

```
ConcatenateMessageArgument ::=
        CHOICE
        {messagesToConcatenate          MessageIDList,
         SEQUENCE
                {messagesToConcatenate  MessageIDList,
                 extensions             CSTACommonArguments} }
```

Section 15.5, Record Message

- Missing IMPLICIT tags for SEQUENCE elements
- Extraneous brace at end of RecordMessageArgument

Corrected:

```
RecordMessageArgument ::=
        SEQUENCE
        {callToBeRecorded       ConnectionID,
         samplingRate           REAL                                    OPTIONAL,
         encodingAlg            [0]IMPLICIT ENUMERATED
                                        {aDPCM6K        (0),
                                         aDPCM8K        (1),
                                         muLawPCM6K     (2),
                                         aLawPCM6K      (3)}    OPTIONAL,
         maxDuration            [1]IMPLICIT INTEGER             OPTIONAL,
         termination            [2]IMPLICIT TerminatingConditions OPTIONAL,
         extensions             CSTACommonArguments             OPTIONAL}
```

Original:

```
RecordMessageArgument ::=
        SEQUENCE
        {callToBeRecorded       ConnectionID,
         samplingRate           REAL                                    OPTIONAL,
         encodingAlg            ENUMERATED
                                        {aDPCM6K        (0),
                                         aDPCM8K        (1),
                                         muLawPCM6K     (2),
                                         aLawPCM6K      (3)}    OPTIONAL,
         maxDuration            INTEGER                         OPTIONAL,
         termination            TerminatingConditions           OPTIONAL,
         extensions             CSTACommonArguments             OPTIONAL}}
```

Section 15.9, Set voice attribute

- OPERATION, ERROR imports missing

Corrected:

```
IMPORTS
OPERATION, ERROR FROM Remote-Operations-Notation
      { joint-iso-ccitt( 2) remote-operations( 4) notation( 0) }
-- Data Types --
```

Original:

```
IMPORTS
-- Data Types --
```

- universalFailure import missing

Corrected:

```
universalFailure FROM CSTA-error-definition
        { iso( 1) identified-organization( 3) icd-ecma( 12)
                standard( 0) csta2( 218) error-definition( 120) }
```

- CSTACommonArguments import mising

Corrected:

```
{ iso( 1) identified-organization( 3) icd-ecma( 12)
        standard( 0) csta2( 218) device-feature-types( 127) }
CSTACommonArguments FROM CSTA-extension-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
        standard( 0) csta2( 218) extension-types( 129) };
```

Original:

```
{ iso( 1) identified-organization( 3) icd-ecma( 0012)
 standard( 0) csta2( 218) device-feature-types( 127) };
```

- SetVoiceAttributeArgument misspelled, comma after brace missing

Corrected:

```
SetVoiceAttributeArgument ::=
        SEQUENCE
        {connection             ConnectionID,
         attributeToSet         CHOICE
                                {speed          [0] IMPLICIT Speed,
                                 speakerVolume  [1] IMPLICIT INTEGER (0 .. 100),
                                 recordingLevel [2] IMPLICIT INTEGER (0 .. 100) },
         message                MessageID                   OPTIONAL,
         extensions             CSTACommonArguments         OPTIONAL}
```

Original:

```
SetVoiveAttributeArgument ::=
        SEQUENCE
        {connection             ConnectionID,
         attributeToSet         CHOICE
                                {speed          [0] IMPLICIT Speed,
                                 speakerVolume  [1] IMPLICIT INTEGER (0 .. 100),
                                 recordingLevel [2] IMPLICIT INTEGER (0 .. 100) }
         message                MessageID                   OPTIONAL,
         extensions             CSTACommonArguments         OPTIONAL}
```

- setVoiceAttributeResult misspelled

Corrected:

```
SetVoiceAttributeResult ::=
    CHOICE
```

Original:

```
SetVoiveAttributeResult ::=
    CHOICE
```

Section 15.10, Stop

- Opening brace missing in StopArgument

Corrected:

```
StopArgument ::=
    SEQUENCE
    {connection     ConnectionID,
```

Original:

```
StopArgument ::=
    {SEQUENCE
     connection     ConnectionID,
```

Section 15.11, Suspend

- Opening brace missing in SuspendArgument

Corrected:

```
SuspendArgument ::=
    SEQUENCE
    {connection     ConnectionID,
```

Original:

```
SuspendArgument ::=
    {SEQUENCE
     connection     ConnectionID,
```

Section 15.12, Synthesize message

- Opening brace missing in SynthesizeMessageArgument

Corrected:

```
SynthesizeMessageArgument ::=
    SEQUENCE
    {textToBeSynthesized          OCTET STRING,
```

Original:

```
SynthesizeMessageArgument ::=
    {SEQUENCE
     textToBeSynthesized          OCTET STRING,
```

Section 18, CSTA data types

- Comment line not properly aligned

Corrected:

```
--The major parameters have been assigned distinct application tags to facilitate parsing. The
data is defined in logical
--groups in ascending order of application tag. Application tags used are:
```

Original:

```
--The major parameters have been assigned distinct application tags to facilitate parsing. The
data is defined in logical --gro
ps in ascending order of application tag. Application tags used are:
```

Section 18.4, Connection states

- ConnectionIDList should be removed from EXPORTS list

Corrected:

```
EXPORTS
ConnectionList, LocalConnectionState;
```

Original:

```
EXPORTS
ConnectionList, ConnectionIDList, LocalConnectionState;
```

Section 18.5, Status reporting

- Extraneous semicolon at end of CSTAObject import statement

Corrected:

```
{ iso( 1) identified-organization( 3) icd-ecma( 12)
        standard( 0) csta2( 218) switching-function-objects( 122) }
```

Original:

```
{ iso( 1) identified-organization( 3) icd-ecma( 0012)
        standard( 0) csta2( 218) switching-function-objects( 122) };
```

Section 18.6, Device and feature types and other parameters

- IOCrossRefIDList misspelled in EXPORTS statement

Corrected:

```
IOCrossRefIDList, IOData, Speed, DataPathDirec
```

Original:

```
IOCrassRefIDList, IOData, Speed, DataPathDirec
```

- SetupValues misspelled in EXPORTS statement

Corrected:

```
SetupValues
```

Original:

```
SetUpValues,
```

- CalledDeviceID misspelled in IMPORTS statement

Corrected:

```
DeviceID, NumberDigits, CalledDeviceID FROM CSTA-device-identifiers
```

Original:

```
DeviceID, NumberDigits, CallDeviceID FROM CSTA-device-identifiers
```

- DeviceInfo definition had extra space in “::=” symbol

Corrected:

```
DeviceInfo      ::= SEQUENCE
    {       deviceID        DeviceID        OPTIONAL,
```

Original:

```
DeviceInfo      :: = SEQUENCE
    {       deviceID        DeviceID        OPTIONAL,
```

- DivertInfo, groupRequestingDevice choice should not be marked IMPLICIT

Corrected:

```
DivertInfo ::= CHOICE          -- used by Divert Call service
    {       deflect                 [0] IMPLICIT SEQUENCE
                                    {callToBeDiverted       ConnectionID,
                                     newDestination         CalledDeviceID  },
            pickup                  [1] IMPLICIT SEQUENCE
                                    {callToBePickedUp       ConnectionID,
                                     requestingDevice       DeviceID        },
            groupRequestingDevice   [2] DeviceID    }
```

Original:

```
DivertInfo ::= CHOICE     -- used by Divert Call service
        {       deflect                 [0] IMPLICIT SEQUENCE
                                        {callToBeDiverted       ConnectionID,
                                         newDestination         CalledDeviceID  },
                pickup                  [1] IMPLICIT SEQUENCE
                                        {callToBePickedUp       ConnectionID,
                                         requestingDevice       DeviceID        },
                groupRequestingDevice   [2] IMPLICIT           DeviceID        }
```

- SetupValues needs space in OCTETSTRING

Corrected:

```
SetupValues ::= OCTET STRING
```

Original:

```
SetupValues ::= OCTETSTRING
```

- LoggedOffInfo missing “password” in sequence

Corrected:

```
LoggedOffInfo       ::= SEQUENCE
    {       agentID         [10] IMPLICIT   AgentID         OPTIONAL,
            password        [11] IMPLICIT   AgentPassword   OPTIONAL,
            group           AgentGroup                      OPTIONAL    }
```

Original:

```
LoggedOffInfo       ::= SEQUENCE
    {       agentID         [10] IMPLICIT   AgentID         OPTIONAL,
            group           AgentGroup                      OPTIONAL    }
```

- Comma missing in IOCrossRefID definition, and OCTETSTRING needs space

Corrected:

```
IOCrossRefID ::= CHOICE
    {   switchProvided      [0] OCTET STRING,
        computerProvided    [1] OCTET STRING }
```

Original:

```
IOCrossRefID ::= CHOICE
    {   switchProvided      [0] OCTETSTRING
        computerProvided    [1] OCTETSTRING }
```

- IOData needs space in OCTETSTRING

Corrected:

```
IOData ::= OCTET STRING
```

Original:

```
IOData ::= OCTETSTRING
```

- MessageID definition had extra space, and needs space in OCTETSTRING

Corrected:

```
MessageID ::= OCTET STRING
```

Original:

```
Message ID::= OCTETSTRING
```

- "SetUpValues" (with capital "u") should be removed, as "Setupvalues" is already defined

Original:

```
SetUpValues ::= OCTET STRING  -- Contains Q.931 Setup message
```

- Period description missing name for choices - selected "relativePosition" and "position" for names; also, choice designations ("[0]" and "[1]") missing

Corrected:

```
Period ::= CHOICE
{       relativePosition [0] ENUMERATED
                {       beginningOfMessage              (0),
                        endOfMessage                    (1)     },
        position [1] INTEGER    }
```

Original:

```
Period ::= CHOICE
{       ENUMERATED
            {   beginingOfMessage       (0),
                endOfMessage            (1)   },
        INTEGER }
```

- AttributeInfo/fileName needs space in OCTETSTRING

Corrected:

```
fileName        OCTET STRING
```

Original:

```
fileName        OCTETSTRING
```

- ControlData/languague needs space in OCTETSTRING

Corrected:

```
language        OCTET STRING
```

Original:

```
language        OCTETSTRING
```

List of corrected errata for ECMA-218

6 January 1999

Summary

Following is a summary of the errors detected and corrected in Standard ECMA-218, Protocol for Computer Supported Telecommunications Applications (CSTA) Phase II.

Section 8.2.1, Encoding of application Context information, definition of CSTAFunctionality

- The first field is misspelt (was ***sitchingFunctionServices***)
- The definition of ***CSTAFunctionality*** can't be accurately decoded because:

  a) Each of the fields (***eventReportServices, computingFunctionServices*** etc) can be ommitted during encoding as indicated by the ***DEFAULT {}***.

  b) Each of the underlying definitions of these fields is a ***BIT STRING*** and hence has the same tag (0x03).

  Because it is not possible to uniquely identify each field during decoding, it is not possible to determine if a given field has been omitted. Addition of tags removes any ambiguity.

Corrected:

```
CSTAFunctionality ::= SEQUENCE
        {switchingFunctionServices      [0] IMPLICIT SwitchingFunctionServices    DEFAULT {},
         eventReportServices            [1] IMPLICIT EventReportServices          DEFAULT {},
         computingFunctionServices      [2] IMPLICIT ComputingFunctionServices    DEFAULT {},
         bidirectionalServices          [3] IMPLICIT BidirectionalServices        DEFAULT {},
         statusReportingServices        [4] IMPLICIT StatusReportingServices      DEFAULT {},
         inputOutputServices            [5] IMPLICIT InputOutputServices          DEFAULT {},
         voiceUnitServices              [6] IMPLICIT VoiceUnitServices            DEFAULT {} }
```

Original:

```
CSTAFunctionality ::= SEQUENCE
        {sitchingFunctionServices       SwitchingFunctionServices       DEFAULT {},
         eventReportServices            EventReportServices             DEFAULT {},
         computingFunctionServices      ComputingFunctionServices       DEFAULT {},
         bidirectionalServices          BidirectionalServices           DEFAULT {},
         statusReportingServices        StatusReportingServices         DEFAULT {},
         inputOutputServices            InputOutputServices             DEFAULT {},
         voiceUnitServices              VoiceUnitServices               DEFAULT {} }
```

Section 10.1.11, Queued event, field misspelt

- The field name for the number of calls in queue is misspelt.

Corrected:

```
numberQueued            [0] IMPLICIT NoOfCallsInQueue   OPTIONAL,
```

Original:

numberedQueued [0] IMPLICIT NoOfCallsInQueue OPTIONAL,

Section 13.3, Monitor Stop service, IMPORTS statement

- The import of ***CSTAPrivateData*** is not needed.

Corrected:

CSTACommonArguments FROM CSTA-extension-types

Original:

CSTACommonArguments, CSTAPrivateData FROM CSTA-extension-types

Section 16, Switching Function Errors, definition of UniversalFailure

- The decoding of a ***UniversalFailure*** choice is ambigious:
  a) ***CSTAPrivateData*** is defined as NULL

```
CSTAPrivateData ::= NULL    -- The actual encoding of the private
                            -- event is added here, replacing NULL with
                            -- another valid ASN.1 type.
```

  b) The ***nonStandardErrors*** CHOICE of ***UniversalFailure*** is defined as ***CSTAPrivateData***

```
UniversalFailure ::= CHOICE
    {operationalErrors              [1] IMPLICIT Operations,
     stateErrors                    [2] IMPLICIT StateIncompatibility,
     systemResourceErrors           [3] IMPLICIT SystemResourceAvailability,
     subscribedResourceAvail...     [4] IMPLICIT SubscribedResourceAvail…,
     performanceErrors              [5] IMPLICIT PerformanceManagement,
     securityErrors                 [6] IMPLICIT SecurityError,
     unspecifiedErrors              [7] IMPLICIT NULL,
     nonStandardErrors              CSTAPrivateData}
```

The tag chosen for ***CSTAPrivateData*** by a particular implementation may clash with one of the existing ***UniversalFailure*** CHOICE tags making the decode process ambigious.

When decoding a ***UniversalFailure*** CHOICE an application has no way of knowing the range of tag values that may be encountered, without prior arrangement with the switch.

Therefore the ***nonStandardErrors*** field must have an implicit tag (as all other uses of ***CSTAPrivateData*** in ECMA-218 do).

Corrected:

nonStandardErrors [8] IMPLICIT CSTAPrivateData}

Original:

nonStandardErrors CSTAPrivateData}

Note: This problem does not exist in Phase I as ***CSTAPrivateData*** had a defined tag (**[Application 29]**) rather than NULL.

Section 16, Switching function errors, error value name

- The name of ***StateIncompatibility*** value 14 is misspelt.

Corrected:

messageSuspended (14) }

Original:

```
                massageSuspended        (14)    }
```

Section 18.5, Status Reporting, CallFilter definition

- The name of ***CallFilter*** value 13 is misspelt.

Corrected:

```
                transferred             (13)    }
```

Original:

```
                transferrd              (13)    }
```

Section 18.6, Device and feature types and other parameters, DomainValue description

- The word "destination" in the comment is misspelt.

Corrected:

```
        DomainValue ::= BOOLEAN     -- TRUE means CSTA subdomain destination
```

Original:

```
        DomainValue ::= BOOLEAN     -- TRUE means CSTA subdomain destinationp
```

Table of contents

SECTION I - GENERAL

1 Scope

This Standard specifies application protocol data units (APDUs) for the services described in Standard ECMA-217, Services for Computer Supported Telecommunications Applications (CSTA) Phase II. The field of application of this Standard is the interconnection of switches and computers in a private telecommunications environment.

Section II (clause 5 to clause 7 inclusive) describes the concepts underlying the Remote Operations model, notation and service.

Section III (clause 8 to clause 19 inclusive) contains CSTA-specific protocol details and forms the main part of this Standard.

The protocol in this Standard operates in the context of an application association.

2 Conformance

A manufacturer may select any part (one or more operations) of the CSTA protocol, as specified in this Standard, for implemention on a system.

A system is in conformance with this Standard if one or more of the CSTA operations are implemented according to the requirements of this Standard.

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 transfer syntax (derived from the encoding rules specified in CCITT Rec. X.209) named {joint-iso-ccitt(2) asn1(1) basic-encoding(1)}; for the purpose of generating and interpreting CSTA protocol information as defined by the abstract syntax "CSTA-ASN.1-Object-Descriptor" for the operations supported.

2.2 Dynamic requirements

To conform to this Standard, a system shall:

i) follow the procedures as specified in this Standard, and Standard ECMA-217, relevant to each CSTA operation that the system claims to implement; and

ii) satisfy the definitions, as specified in Standard ECMA-217, 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:

i) which CSTA operations, as defined in Standard ECMA-217, are supported by the system for the particular implementation; and

ii) which optional parameters are supported by the PDUs belonging to the supported operations; and

iii) the types and ranges of values for all the parameters supported.

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

3 References

ECMA-138	Security in Open Systems - Data Elements and Service Definitions (1989)
ECMA-217	Services for Computer Supported Telecommunications Applications (CSTA) Phase II (1994)

ECMA TR/46	Security in Open Systems - A Security Framework (1988)
ECMA TR/52	Computer Supported Telecommunications Applications (1990)
ECMA TR/68	Scenarios for Computer Supported Telecommunications Applications (CSTA) Phase II (1994)
CENELEC ENV 41007	Definitions of Terms in Private Telecommunications Networks
CCITT Rec. E.164	Numbering Plan for ISDN
ISO 8649:1988	Information processing systems - Open Systems Interconnection - Service definition for the Association Control Service Element (this corresponds to CCITT Rec. X.217)
ISO 8650:1988	Information processing systems - Open Systems Interconnection - Protocol specification for the Association Control Service Element (this corresponds to CCITT Rec. X.227)
ISO/IEC 8824:1990	Information technology - Open Systems Interconnection - Specification of Abstract Syntax Notation One (ASN.1) (this corresponds to CCITT Rec. X.208)
ISO/IEC 8825:1990	Information technology - Open Systems Interconnection - Specification of basic encoding rules for Abstract Syntax Notation One (ASN.1) (this corresponds to CCITT Rec. X.209)
ISO/IEC 9072-1:1989	Information processing systems - Text communication - Remote operations - Part 1: Model, notation and service definition (this corresponds to CCITT Rec. X.219)
ISO/IEC 9072-2:1989	Information processing systems - Text communication - Remote operations - Part 2: Protocol specification (this corresponds to CCITT Rec. X.229)
ISO/IEC 9545:1994	Information technology - Open Systems Interconnection - Application Layer structure

All the CCITT Recommendations that are referenced are the 1988 (Blue Book) version.

4 Definitions

CSTA-specific terminology is defined in Standard ECMA-217. For the purposes of this Standard, the following additional definitions, defined in other standards, shall apply:

- Remote Operations (as per CCITT Rec. X.219)
- Application Association (as per CCITT Rec. X.217)
- Application Context (as per CCITT Rec. X.217)
- Private Telecommunications Network (as per CENELEC ENV 41007)

SECTION II - PROTOCOL STRUCTURE FOR CSTA

5 CSTA service definition model

5.1 CSTA application layer structure

The CSTA Application Layer structure conforms to the model described in ISO/IEC 9545.

5.2 Remote operations

The services of CSTA are modelled as Remote Operations as described in CCITT Rec. X.219. Typically, one entity requests that a particular operation be performed; the other entity attempts to perform the operation and responds to the requestor. Consequently the operation of the protocol is an elementary request/reply interaction, supported within the OSI Application Layer, and carried out within the context of an application association.

For some of the CSTA services, the entity to which the request is sent generates a reply which can indicate success or failure. For these services, CSTA shall adopt the operations Class 2, defined in CCITT Rec. X.219 as:

- Asynchronous, reporting success or failure (result or error).

For some of the CSTA services, the entity to which the request is sent generates a reply which can only indicate failure. For these services, CSTA shall adopt the operations Class 3, defined in CCITT Rec. X.219 as:

- Asynchronous, reporting failure (error) only, if any.

For some of the CSTA services, particularly the ongoing reporting of events, no reply is generated. For these services, CSTA shall adopt the operations Class 5, defined in CCITT Rec. X.219 as:

- Asynchronous, outcome not reported.

The protocol description for the particular service defines the relevant class of the operation used for that service.

CSTA shall correlate the single response, denoting success or failure, with the originating request by using the mechanisms within the ROSE protocol.

5.3 The CSTA service response

CSTA employs a generic response mechanism which is, in principle, decoupled from the specifics of the switching activity. The following points describe the operation of the CSTA service response:

i) Specific services may have an unconfirmed mode where responses to correct requests are not returned.

ii) The server shall check the correctness of the request (e.g. syntactical checks) before issuing the response. Incorrect requests shall result in an error response, even in the unconfirmed mode.

iii) If a response is sent before the action requested by the service is completed (i.e. the response is a service request acknowledgement), event reporting may be used to keep track of the subsequent server activity.

iv) The precise moment at which the response is generated in relation to the switching activity is implementation- and service- dependent.

- Some implementations may generate the response after checking the correctness of the request and at the point they initiate the request.
- Other implementations may delay the response until the service has completed (or is guaranteed to complete). In this case, a failure of the switching request will be reflected in the response.

NOTE 1

Irrespective of implementation details, when an operation succeeds, the same event reports are generated if the necessary monitoring has been established. As an example, a Held-Event (if selected) is always reported in addition to the response to a successful request (even in those implementations that delay the response until the Hold operation was complete). In a given context, and with appropriate monitoring in place, an operation generates the same set of event reports, whether it was invoked manually or with CSTA service requests from the computer.

5.4 Cross referencing of event reports

A computer application process may need to cross reference a CSTAEventReport to one of the following:

a) a CSTA Object ID (Call ID or Device ID),

b) an earlier Monitor request; or

c) one of many Monitor requests (pertaining to the same CSTA Object).

For the above scenarios the necessary cross referencing function may be fulfilled by use of the parameter "MonitorCrossRefID". The content of MonitorCrossRefID depends upon the context and it may be one of the following: Call ID, StaticDevice ID or another independently switch managed static identifier. The independent identifier may have a unique correlation to either: one device, one call, or one monitor request.

The switching system limit on the number of monitor requests on one CSTA Object (Call or Device) is an implementation option. This Standard does not stipulate how many monitor requests on one object are to be supported by the switch. If using Static Device or Call identifiers the limit can only be one.

5.5 Handling of private data

If an entity receives the parameter CSTAPrivateData, and it can not recognize the information contained, the parameter 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 and uses the Remote Operations protocol defined in CCITT Rec. X.229. The Remote Operations protocol assumes certain services are provided by the underlying layers, and these services are also assumed by the protocol for CSTA.

7 Security

This protocol also provides a mechanism for secure transmission of CSTA PDUs as defined in this Standard. The parameters that constitute this secure PDU are imported from Standard ECMA-138.

SECTION III - CSTA PROTOCOL

8 Association management

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

- an implicit association achieved via off-line agreement; or
- a dynamically negotiated association realsied through the use of ACSE.

8.1 Implicit association

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

8.2 Dynamic association management using ACSE

A common application context Name is used, and CSTA protocol version information is carried within the User Information field of the A-ASSOCIATE request and response PDUs.

An application context is established (using ACSE) as follows:

- the system generating the A-ASSOCIATE request includes a list of all protocol versions that it is prepared to offer in the User Information field;
- on receipt of the A-ASSOCIATE request, the receiving system selects the protocol version to be used by identifying the highest version that is common to both systems;
- the protocol version selected is conveyed to the requestor in the User Information field of the A-ASSOCIATE response.

In addition to negotiating the protocol version, it is necessary for the requesting and responding systems to specify the CSTA services that they support. As with the protocol version information, this is also achieved by carrying additional information in the User Information field of the A-ASSOCIATE request and response PDUs. The application association requestor shall:

- list the services required from the serving application;
- list the services it can supply.

The responder shall include similar information for the responding application. At this point the association requestor will either accept or reject the association.

8.2.1 Encoding of application context information

```
CSTA-application-context-information
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) application-context-information( 200) }
DEFINITIONS ::=
BEGIN

ACSEUserInformationForCSTA ::= SEQUENCE
    {cSTAVersion                                CSTAVersion,
     cSTAFunctionsRequiredByApplication         CSTAFunctionality,
     cSTAFunctionsThatCanBeSupplied             CSTAFunctionality       }

CSTAVersion ::= BIT STRING
    {versionOne             (0),
     versionTwo             (1)
```

```
      -- others to be added in future

     }

--   When requesting an application association, the requestor will set the corresponding bits for each
--   version that it supports. When responding, the responder shall specify only one bit corresponding
--   to the version selected (the highest common to both systems).

CSTAFunctionality ::= SEQUENCE
     {switchingFunctionServices     [0] IMPLICIT SwitchingFunctionServices     DEFAULT {},
      eventReportServices           [1] IMPLICIT EventReportServices           DEFAULT {},
      computingFunctionServices     [2] IMPLICIT ComputingFunctionServices     DEFAULT {},
      bidirectionalServices         [3] IMPLICIT BidirectionalServices         DEFAULT {},
      statusReportingServices       [4] IMPLICIT StatusReportingServices       DEFAULT {},
      inputOutputServices           [5] IMPLICIT InputOutputServices           DEFAULT {},
      voiceUnitServices             [6] IMPLICIT VoiceUnitServices             DEFAULT {} }

SwitchingFunctionServices ::= BIT STRING
     {alternateCall                 (0),
      answerCall                    (1),
      associateData                 (2),
      callCompletion                (3),
      clearCall                     (4),
      clearConnection               (5),
      conferenceCall                (6),
      consultationCall              (7),
      divertCall                    (8),
      holdCall                      (9),
      makeCall                      (10),
      makePredictiveCall            (11),
      parkCall                      (12),
      queryDevice                   (13),
      reconnectCall                 (14),
      retrieveCall                  (15),
      sendDTMFTones                 (16),
      setFeature                    (17),
      singleStepConference          (18),
      singleStepTransfer            (19),
      transferCall                  (20)     }

EventReportServices ::= BIT STRING
     {

     -- Call events

      callCleared                   (0),
      conferenced                   (1),
      connectionCleared             (2),
      delivered                     (3),
      diverted                      (4),
```

```
    established                 (5),
    failed                      (6),
    held                        (7),
    networkReached              (8),
    originated                  (9),
    queued                      (10),
    retrieved                   (11),
    serviceInitiated            (12),
    transferred                 (13),

   -- Feature events

    autoAnswer                  (14),
    callInformation             (15),
    doNotDisturb                (16),
    forwarding                  (17),
    messageWaiting              (18),
    microphoneMute              (19),
    speakerMute                 (20),
    speakerVolume               (21),

   -- Agent state events

    agentBusy                   (22),
    loggedOn                    (23),
    loggedOff                   (24),
    notReady                    (25),
    ready                       (26),
    workingAfterCall            (27),

   -- Maintenance events

    backInService               (28),
    outOfServiceEvent           (29),

   -- Private events

    privateEvent                (30),

   -- Voice unit events

    playEvent                   (31),
    recordEvent                 (32),
    reviewEvent                 (33),
    stopEvent                   (34),
    suspendPlayEvent            (35),
    suspendRecordEvent          (36),
    voiceAttributesChangeEvent  (37)    }
```

```
ComputingFunctionServices ::= BIT STRING
     {routeRequest                (0),
      reRouteRequest              (1),
      routeSelectRequest          (2),
      routeUsedRequest            (3),
      routeEndRequest             (4)      }

BidirectionalServices ::= BIT STRING
     {escapeService               (0),
      systemStatus                (1)      }

StatusReportingServices ::= BIT STRING
     {monitorStart                (0),
      changeMonitorFilter         (1),
      monitorStop                 (2),
      snapshotDevice              (3),
      snapshotCall                (4)      }

InputOutputServices ::= BIT STRING
     {startDataPathService        (0),
      stopDataPathService         (1),
      sendDataService             (2),
      sendMulticastDataService    (3),
      sendBroadcastDataService    (4),
      suspendDataPathService      (5),
      dataPathSuspendedService    (6),
      resumeDataPath              (7),
      dataPathResumedService      (8),
      fastData                    (9)      }

VoiceUnitServices ::= BIT STRING
     {concatenateMessage          (0),
      deleteMessage               (1),
      playMessage                 (2),
      queryVoiceAttribute         (3),
      recordMessage               (4),
      reposition                  (5),
      resume                      (6),
      review                      (7),
      setVoiceAttribute           (8),
      stop                        (9),
      suspend                     (10),
      synthesizeMessage           (11)     }

END  -- of CSTA-application-context-information
```

9 Switching function services

This clause defines the protocol for the switching function services of CSTA, using ASN.1. Text descriptions of the CSTA services are provided in Standard ECMA-217.

The encoding of the protocol is defined in CCITT Rec. X.209.

NOTE 2

The range of services supported on a particular association is specified in the application context at association time.

9.1 Alternate call

```
CSTA-alternate-call
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) alternate-call( 1) }
DEFINITIONS ::=
BEGIN
IMPORTS
OPERATION, ERROR FROM Remote-Operations-Notation
    { joint-iso-ccitt( 2) remote-operations( 4) notation( 0) }
-- Data Types --
universalFailure FROM CSTA-error-definition
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) error-definition( 120) }
ConnectionDetails FROM CSTA-device-feature-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) device-feature-types( 127) }
CSTACommonArguments FROM CSTA-extension-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) extension-types( 129) };

alternateCall       OPERATION
    ARGUMENT        AlternateCallArgument
    RESULT          AlternateCallResult
    ERRORS          {universalFailure}
::= 1

AlternateCallArgument ::=
    CHOICE
    {callsInvolved          ConnectionDetails,
     SEQUENCE
            {callsInvolved  ConnectionDetails,
             extensions     CSTACommonArguments} }

AlternateCallResult ::=
    CHOICE
    {extensions     CSTACommonArguments,
     noData         NULL}

END  -- of CSTA-alternate-call
```

9.2 Answer call

```
CSTA-answer-call
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) answer-call( 2) }
DEFINITIONS ::=
BEGIN
IMPORTS
OPERATION, ERROR FROM Remote-Operations-Notation
    { joint-iso-ccitt( 2) remote-operations( 4) notation( 0) }
-- Data Types --
universalFailure FROM CSTA-error-definition
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) error-definition( 120) }
ConnectionID FROM CSTA-call-connection-identifiers
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) call-connection-identifiers( 124) }
CSTACommonArguments FROM CSTA-extension-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) extension-types( 129) };

answerCall          OPERATION
    ARGUMENT    AnswerCallArgument
    RESULT      AnswerCallResult
    ERRORS      {universalFailure}
::= 2

AnswerCallArgument ::=
    CHOICE
    {callToBeAnswered           ConnectionID,
     SEQUENCE
            {callToBeAnswered   ConnectionID,
             extensions         CSTACommonArguments} }

AnswerCallResult ::=
    CHOICE
    {extensions     CSTACommonArguments,
     noData         NULL}

END  -- of CSTA-answer-call
```

9.3 Associate data

```
CSTA-associate-data
    { iso( 1) identified-organization( 3) icd-ecma( 12)
        standard( 0) csta2( 218) associate-data( 17) }
DEFINITIONS ::=
BEGIN
IMPORTS
OPERATION, ERROR FROM Remote-Operations-Notation
    { joint-iso-ccitt( 2) remote-operations( 4) notation( 0) }
-- Data Types --
universalFailure FROM CSTA-error-definition
    { iso( 1) identified-organization( 3) icd-ecma( 12)
        standard( 0) csta2( 218) error-definition( 120) }
ConnectionID FROM CSTA-call-connection-identifiers
    { iso( 1) identified-organization( 3) icd-ecma( 12)
        standard( 0) csta2( 218) call-connection-identifiers( 124) }
AccountInfo, AuthCode, CorrelatorData FROM CSTA-device-feature-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
        standard( 0) csta2( 218) device-feature-types( 127) }
CSTACommonArguments FROM CSTA-extension-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
        standard( 0) csta2( 218) extension-types( 129) };

associateData       OPERATION
    ARGUMENT        AssociateDataArgument
    RESULT          AssociateDataResult
    ERRORS          {universalFailure}
::= 17

AssociateDataArgument ::=
    SEQUENCE
    {existingCall           ConnectionID,
     accountCode            [0] IMPLICIT AccountInfo        OPTIONAL,
     authCode               [1] IMPLICIT AuthCode           OPTIONAL,
     correlatorData         [2] IMPLICIT CorrelatorData     OPTIONAL,
     extensions             CSTACommonArguments             OPTIONAL}

AssociateDataResult ::=
    CHOICE
    {extensions     CSTACommonArguments,
     noData         NULL}

END  -- of CSTA-associate-data
```

9.4 Call completion

```
CSTA-call-completion
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) call-completion( 3) }
DEFINITIONS ::=
BEGIN
IMPORTS
OPERATION, ERROR FROM Remote-Operations-Notation
    { joint-iso-ccitt( 2) remote-operations( 4) notation( 0) }
-- Data Types --
universalFailure FROM CSTA-error-definition
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) error-definition( 120) }
FeatureInfo FROM CSTA-device-feature-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) device-feature-types( 127) }
CSTACommonArguments FROM CSTA-extension-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) extension-types( 129) };

callCompletion      OPERATION
    ARGUMENT    CallCompletionArgument
    RESULT      CallCompletionResult
    ERRORS      {universalFailure}
::= 3

CallCompletionArgument ::=
    CHOICE
    {featureInfo            FeatureInfo,
     SEQUENCE
            {featureInfo    FeatureInfo,
             extensions     CSTACommonArguments} }

CallCompletionResult ::=
            CHOICE
            {extensions     CSTACommonArguments,
             noData         NULL}

END  -- of CSTA-call-completion
```

9.5 Clear call

```
CSTA-clear-call
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) clear-call( 4) }
DEFINITIONS ::=
BEGIN
IMPORTS
OPERATION, ERROR FROM Remote-Operations-Notation
    { joint-iso-ccitt( 2) remote-operations( 4) notation( 0) }
-- Data Types --
universalFailure FROM CSTA-error-definition
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) error-definition( 120) }
ConnectionID FROM CSTA-call-connection-identifiers
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) call-connection-identifiers( 124) }
CSTACommonArguments FROM CSTA-extension-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) extension-types( 129) };

clearCall           OPERATION
    ARGUMENT        ClearCallArgument
    RESULT          ClearCallResult
    ERRORS          {universalFailure}
::= 4

ClearCallArgument ::=
    CHOICE
    {callToBeCleared                ConnectionID,
     SEQUENCE
            {callToBeCleared        ConnectionID,
             extensions             CSTACommonArguments} }

ClearCallResult ::=
    CHOICE
    {extensions     CSTACommonArguments,
     noData         NULL}

END  -- of CSTA-clear-call
```

9.6 Clear connection

```
CSTA-clear-connection
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) clear-connection( 5) }
DEFINITIONS ::=
BEGIN
IMPORTS
OPERATION, ERROR FROM Remote-Operations-Notation
    { joint-iso-ccitt( 2) remote-operations( 4) notation( 0) }
-- Data Types --
universalFailure FROM CSTA-error-definition
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) error-definition( 120) }
ConnectionID FROM CSTA-call-connection-identifiers
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) call-connection-identifiers( 124) }
CSTACommonArguments FROM CSTA-extension-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) extension-types( 129) };

clearConnection     OPERATION
    ARGUMENT    ClearConnectionArgument
    RESULT      ClearConnectionResult
    ERRORS      {universalFailure}
::= 5

ClearConnectionArgument ::=
    CHOICE
    {connectionToBeCleared              ConnectionID,
     SEQUENCE
            {connectionToBeCleared      ConnectionID,
             extensions                 CSTACommonArguments} }

ClearConnectionResult ::=
    CHOICE
    {extensions     CSTACommonArguments,
     noData         NULL}

END  -- of CSTA-clear-connection
```

9.7 Conference call

```
CSTA-conference-call
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) conference-call( 6) }
DEFINITIONS ::=
BEGIN
IMPORTS
OPERATION, ERROR FROM Remote-Operations-Notation
    { joint-iso-ccitt( 2) remote-operations( 4) notation( 0) }
-- Data Types --
universalFailure FROM CSTA-error-definition
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) error-definition( 120) }
ConnectionID FROM CSTA-call-connection-identifiers
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) call-connection-identifiers( 124) }
ConnectionList FROM CSTA-connection-states
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) connection-states( 125) }
ConnectionDetails FROM CSTA-device-feature-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) device-feature-types( 127) }
CSTACommonArguments FROM CSTA-extension-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) extension-types( 129) };

conferenceCall      OPERATION
    ARGUMENT    ConferenceCallArgument
    RESULT      ConferenceCallResult
    ERRORS      {universalFailure}
::= 6

ConferenceCallArgument ::=
    CHOICE
    {callsInvolved          ConnectionDetails,
     SEQUENCE
            {callsInvolved  ConnectionDetails,
             extensions     CSTACommonArguments} }

ConferenceCallResult ::=
    SEQUENCE
    {conferenceCall         ConnectionID,
     connections            ConnectionList          OPTIONAL,
     extensions             CSTACommonArguments     OPTIONAL}

END  -- of CSTA-conference-call
```

9.8 Consultation call

```
CSTA-consultation-call
    { iso( 1) identified-organization( 3) icd-ecma( 12)
        standard( 0) csta2( 218) consultation-call( 7) }
DEFINITIONS ::=
BEGIN
IMPORTS
OPERATION, ERROR FROM Remote-Operations-Notation
    { joint-iso-ccitt( 2) remote-operations( 4) notation( 0) }
-- Data Types --
universalFailure FROM CSTA-error-definition
    { iso( 1) identified-organization( 3) icd-ecma( 12)
        standard( 0) csta2( 218) error-definition( 120) }
CalledDeviceID FROM CSTA-device-identifiers
    { iso( 1) identified-organization( 3) icd-ecma( 12)
        standard( 0) csta2( 218) device-identifiers( 123) }
ConnectionID FROM CSTA-call-connection-identifiers
    { iso( 1) identified-organization( 3) icd-ecma( 12)
        standard( 0) csta2( 218) call-connection-identifiers( 124) }
AccountInfo, AuthCode, CorrelatorData, DeviceProfile FROM CSTA-device-feature-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
        standard( 0) csta2( 218) device-feature-types( 127) }
CSTACommonArguments FROM CSTA-extension-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
        standard( 0) csta2( 218) extension-types( 129) };

consultationCall    OPERATION
    ARGUMENT    ConsultationCallArgument
    RESULT      ConsultationCallResult
    ERRORS      {universalFailure}
::= 7

ConsultationCallArgument ::=
    SEQUENCE
    {existingCall           ConnectionID,
     consultedDevice        CalledDeviceID,
     consultedDeviceProfile DeviceProfile                OPTIONAL,
     accountCode            [0] IMPLICIT AccountInfo     OPTIONAL,
     authCode               [1] IMPLICIT AuthCode        OPTIONAL,
     correlatorData         [2] IMPLICIT CorrelatorData  OPTIONAL,
     extensions             CSTACommonArguments          OPTIONAL}

ConsultationCallResult ::=
    CHOICE
    {initiatedCall          ConnectionID,
     SEQUENCE
        {initiatedCall      ConnectionID,
         extensions         CSTACommonArguments} }
```

END -- of CSTA-consultation-call

9.9 Divert call

```
CSTA-divert-call
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) divert-call( 8) }
DEFINITIONS ::=
BEGIN
IMPORTS
OPERATION, ERROR FROM Remote-Operations-Notation
    { joint-iso-ccitt( 2) remote-operations( 4) notation( 0) }
-- Data Types --
universalFailure FROM CSTA-error-definition
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) error-definition( 120) }
CorrelatorData, DivertInfo, DeviceProfile FROM CSTA-device-feature-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) device-feature-types( 127) }
CSTACommonArguments FROM CSTA-extension-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) extension-types( 129) };

divertCall          OPERATION
    ARGUMENT        DivertCallArgument
    RESULT          DivertCallResult
    ERRORS          {universalFailure}
::= 8

DivertCallArgument ::=
    SEQUENCE
    {divertInfo     DivertInfo,
     deviceProfile  DeviceProfile              OPTIONAL,
     correlatorData CorrelatorData             OPTIONAL,
     extensions     CSTACommonArguments        OPTIONAL}

DivertCallResult ::=
    CHOICE
    {extensions     CSTACommonArguments,
     noData         NULL}

END  -- of CSTA-divert-call
```

9.10 Hold call

```
CSTA-hold-call
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) hold-call( 9) }
DEFINITIONS ::=
BEGIN
IMPORTS
OPERATION, ERROR FROM Remote-Operations-Notation
    { joint-iso-ccitt( 2) remote-operations( 4) notation( 0) }
-- Data Types --
universalFailure FROM CSTA-error-definition
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) error-definition( 120) }
ConnectionID FROM CSTA-call-connection-identifiers
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) call-connection-identifiers( 124) }
ReserveConnection FROM CSTA-device-feature-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) device-feature-types( 127) }
CSTACommonArguments FROM CSTA-extension-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) extension-types( 129) };

holdCall    OPERATION
    ARGUMENT    HoldCallArgument
    RESULT      HoldCallResult
    ERRORS      {universalFailure}
::= 9

HoldCallArgument ::=
    SEQUENCE
    {callToBeHeld           ConnectionID,
     connectionReservation  ReserveConnection          OPTIONAL,
     extensions             CSTACommonArguments        OPTIONAL}

HoldCallResult ::=
    CHOICE
    {extensions     CSTACommonArguments,
     noData         NULL}

END  -- of CSTA-hold-call
```

9.11 Make call

```
CSTA-make-call
    { iso( 1) identified-organization( 3) icd-ecma( 12)
        standard( 0) csta2( 218) make-call( 10) }
DEFINITIONS ::=
BEGIN
IMPORTS
OPERATION, ERROR FROM Remote-Operations-Notation
    { joint-iso-ccitt( 2) remote-operations( 4) notation( 0) }
-- Data Types --
universalFailure FROM CSTA-error-definition
    { iso( 1) identified-organization( 3) icd-ecma( 12)
        standard( 0) csta2( 218) error-definition( 120) }
DeviceID, CalledDeviceID FROM CSTA-device-identifiers
    { iso( 1) identified-organization( 3) icd-ecma( 12)
        standard( 0) csta2( 218) device-identifiers( 123) }
ConnectionID FROM CSTA-call-connection-identifiers
    { iso( 1) identified-organization( 3) icd-ecma( 12)
        standard( 0) csta2( 218) call-connection-identifiers( 124) }
AccountInfo, AuthCode, CorrelatorData, DeviceProfile FROM CSTA-device-feature-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
        standard( 0) csta2( 218) device-feature-types( 127) }
CSTACommonArguments FROM CSTA-extension-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
        standard( 0) csta2( 218) extension-types( 129) };

makeCall            OPERATION
    ARGUMENT        MakeCallArgument
    RESULT          MakeCallResult
    ERRORS          {universalFailure}
::= 10

MakeCallArgument ::=
    SEQUENCE
    {callingDevice          DeviceID,
     calledDirectoryNumber  CalledDeviceID,
     deviceProfile          DeviceProfile                   OPTIONAL,
     accountCode            [0] IMPLICIT AccountInfo        OPTIONAL,
     authCode               [1] IMPLICIT AuthCode           OPTIONAL,
     correlatorData         [2] IMPLICIT CorrelatorData     OPTIONAL,
     extensions             CSTACommonArguments             OPTIONAL}

MakeCallResult ::=
    CHOICE
    {initiatedCall          ConnectionID,
     SEQUENCE
        {initiatedCall      ConnectionID,
         extensions         CSTACommonArguments} }
```

```
END  -- of CSTA-make-call
```

9.12 Make predictive call

```
CSTA-make-predictive-call
    { iso( 1) identified-organization( 3) icd-ecma( 12)
        standard( 0) csta2( 218) make-predictive-call( 11) }
DEFINITIONS ::=
BEGIN
IMPORTS
OPERATION, ERROR FROM Remote-Operations-Notation
    { joint-iso-ccitt( 2) remote-operations( 4) notation( 0) }
-- Data Types --
universalFailure FROM CSTA-error-definition
    { iso( 1) identified-organization( 3) icd-ecma( 12)
        standard( 0) csta2( 218) error-definition( 120) }
DeviceID, CalledDeviceID FROM CSTA-device-identifiers
    { iso( 1) identified-organization( 3) icd-ecma( 12)
        standard( 0) csta2( 218) device-identifiers( 123) }
ConnectionID FROM CSTA-call-connection-identifiers
    { iso( 1) identified-organization( 3) icd-ecma( 12)
        standard( 0) csta2( 218) call-connection-identifiers( 124) }
AccountInfo, AuthCode, AllocationState, CorrelatorData, DeviceProfile
    FROM CSTA-device-feature-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
        standard( 0) csta2( 218) device-feature-types( 127) }
CSTACommonArguments FROM CSTA-extension-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
        standard( 0) csta2( 218) extension-types( 129) };

makePredictiveCall  OPERATION
    ARGUMENT    MakePredictiveCallArgument
    RESULT      MakePredictiveCallResult
    ERRORS      {universalFailure}
::= 11

MakePredictiveCallArgument ::=
    SEQUENCE
    {callingDevice          DeviceID,
     calledDirectoryNumber  CalledDeviceID,
     allocation             AllocationState                 OPTIONAL,
     deviceProfile          DeviceProfile                   OPTIONAL,
     accountCode            [0] IMPLICIT AccountInfo        OPTIONAL,
     authCode               [1] IMPLICIT AuthCode           OPTIONAL,
     correlatorData         [2] IMPLICIT CorrelatorData     OPTIONAL,
     extensions             CSTACommonArguments             OPTIONAL}

MakePredictiveCallResult ::=
    CHOICE
    {initiatedCall          ConnectionID,
     SEQUENCE
```

```
                {initiatedCall      ConnectionID,
                 extensions         CSTACommonArguments} }

END  -- of CSTA-make-predictive-call
```

9.13 Park call

```
CSTA-park-call
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) park-call( 18) }
DEFINITIONS ::=
BEGIN
IMPORTS
OPERATION, ERROR FROM Remote-Operations-Notation
    { joint-iso-ccitt( 2) remote-operations( 4) notation( 0) }
-- Data Types --
universalFailure FROM CSTA-error-definition
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) error-definition( 120) }
SubjectDeviceID FROM CSTA-device-identifiers
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) device-identifiers( 123) }
ConnectionID FROM CSTA-call-connection-identifiers
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) call-connection-identifiers( 124) }
CorrelatorData, DeviceProfile FROM CSTA-device-feature-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) device-feature-types( 127) }
CSTACommonArguments FROM CSTA-extension-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) extension-types( 129) };

parkCall            OPERATION
    ARGUMENT        ParkCallArgument
    RESULT          ParkCallResult
    ERRORS          {universalFailure}
::= 18

ParkCallArgument ::=
     SEQUENCE
    {callToPark             ConnectionID,
     parkTo                 SubjectDeviceID,
     parkToDeviceProfile    DeviceProfile           OPTIONAL,
     correlatorData         CorrelatorData          OPTIONAL,
     extensions             CSTACommonArguments     OPTIONAL}

ParkCallResult ::=
    CHOICE
    {extensions     CSTACommonArguments,
     noData         NULL}

END  -- of CSTA-park-call
```

9.14 Query device

```
CSTA-query-device
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) query-device( 12) }
DEFINITIONS ::=
BEGIN
IMPORTS
OPERATION, ERROR FROM Remote-Operations-Notation
            { joint-iso-ccitt( 2) remote-operations( 4) notation( 0) }
-- Data Types --
universalFailure FROM CSTA-error-definition
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) error-definition( 120) }
DeviceID FROM CSTA-device-identifiers
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) device-identifiers( 123) }
QueryDeviceFeature, QueryDeviceInformation FROM CSTA-device-feature-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) device-feature-types( 127) }
CSTACommonArguments FROM CSTA-extension-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) extension-types( 129) };

queryDevice         OPERATION
    ARGUMENT    QueryDeviceArgument
    RESULT      QueryDeviceResult
    ERRORS      {universalFailure}
::= 12

QueryDeviceArgument ::=
    SEQUENCE
    {device         DeviceID,
     feature        QueryDeviceFeature,
     extensions     CSTACommonArguments     OPTIONAL}

QueryDeviceResult ::=
    CHOICE
    {deviceInformation              QueryDeviceInformation,
     SEQUENCE
            {deviceInformation      QueryDeviceInformation,
             extensions             CSTACommonArguments} }

END  -- of CSTA-query-device
```

9.15 Reconnect call

```
CSTA-reconnect-call
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) reconnect-call( 13) }
DEFINITIONS ::=
BEGIN
IMPORTS
OPERATION, ERROR FROM Remote-Operations-Notation
    { joint-iso-ccitt( 2) remote-operations( 4) notation( 0) }
-- Data Types --
universalFailure FROM CSTA-error-definition
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) error-definition( 120) }
ConnectionDetails FROM CSTA-device-feature-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) device-feature-types( 127) }
CSTACommonArguments FROM CSTA-extension-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) extension-types( 129) };

reconnectCall       OPERATION
    ARGUMENT    ReconnectCallArgument
    RESULT      ReconnectCallResult
    ERRORS      {universalFailure}
::= 13

ReconnectCallArgument ::=
    CHOICE
    {reconnectInfo          ConnectionDetails,
     SEQUENCE
            {reconnectInfo  ConnectionDetails,
             extensions     CSTACommonArguments} }

ReconnectCallResult ::=
    CHOICE
    {extensions     CSTACommonArguments,
     noData         NULL}

END  -- of CSTA-reconnect-call
```

9.16 Retrieve call

```
CSTA-retrieve-call
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) retrieve-call( 14) }
DEFINITIONS ::=
BEGIN
IMPORTS
OPERATION, ERROR FROM Remote-Operations-Notation
    { joint-iso-ccitt( 2) remote-operations( 4) notation( 0) }
-- Data Types --
universalFailure FROM CSTA-error-definition
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) error-definition( 120) }
ConnectionID FROM CSTA-call-connection-identifiers
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) call-connection-identifiers( 124) }
CSTACommonArguments FROM CSTA-extension-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) extension-types( 129) };

retrieveCall        OPERATION
    ARGUMENT    RetrieveCallArgument
    RESULT      RetrieveCallResult
    ERRORS      {universalFailure}
::= 14

RetrieveCallArgument ::=
    CHOICE
    {callToBeRetrieved              ConnectionID,
     SEQUENCE
            {callToBeRetrieved      ConnectionID,
             extensions             CSTACommonArguments} }

RetrieveCallResult ::=
    CHOICE
    {extensions     CSTACommonArguments,
     noData         NULL}

END  -- of CSTA-retrieve-call
```

9.17 Send DTMF tones

```
CSTA-send-DTMF-tones
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) send-dtmf-tones( 19) }
DEFINITIONS ::=
BEGIN
IMPORTS
OPERATION, ERROR FROM Remote-Operations-Notation
    { joint-iso-ccitt( 2) remote-operations( 4) notation( 0) }
-- Data Types --
universalFailure FROM CSTA-error-definition
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) error-definition( 120) }
ConnectionID FROM CSTA-call-connection-identifiers
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) call-connection-identifiers( 124) }
CSTACommonArguments FROM CSTA-extension-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) extension-types( 129) };

sendDTMFTones   OPERATION
    ARGUMENT    SendDTMFTonesArgument
    RESULT      SendDTMFTonesResult
    ERRORS      {universalFailure}
::= 19

SendDTMFTonesArgument ::=
    SEQUENCE
    {connectionToSendTones      ConnectionID,
     charactersToSend           IA5String,
     toneDuration               [0] IMPLICIT INTEGER        OPTIONAL,
     pauseDuration              [1] IMPLICIT INTEGER        OPTIONAL,
     extensions                 CSTACommonArguments         OPTIONAL}

SendDTMFTonesResult ::=
    CHOICE
    {extensions     CSTACommonArguments,
     noData         NULL}

END  -- of CSTA-send-DTMF-tones
```

9.18 Set feature

```
CSTA-set-feature
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) set-feature( 15) }
DEFINITIONS ::=
BEGIN
IMPORTS
OPERATION, ERROR FROM Remote-Operations-Notation
    { joint-iso-ccitt( 2) remote-operations( 4) notation( 0) }
-- Data Types --
universalFailure FROM CSTA-error-definition
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) error-definition( 120) }
DeviceID FROM CSTA-device-identifiers
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) device-identifiers( 123) }
DeviceProfile, SetDeviceFeature FROM CSTA-device-feature-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) device-feature-types( 127) }
CSTACommonArguments FROM CSTA-extension-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) extension-types( 129) };

setFeature          OPERATION
    ARGUMENT        SetFeatureArgument
    RESULT          SetFeatureResult
    ERRORS          {universalFailure}
::= 15

SetFeatureArgument ::=
    SEQUENCE
    {device         DeviceID,
     feature        SetDeviceFeature,
     deviceProfile  DeviceProfile              OPTIONAL,
     extensions     CSTACommonArguments        OPTIONAL}

SetFeatureResult ::=
    CHOICE
    {extensions     CSTACommonArguments,
     noData         NULL}

END  -- of CSTA-set-feature
```

9.19 Single step conference

```
CSTA-single-step-conference
    { iso( 1) identified-organization( 3) icd-ecma( 12)
        standard( 0) csta2( 218) single-step-conference( 20) }
DEFINITIONS ::=
BEGIN
IMPORTS
OPERATION, ERROR FROM Remote-Operations-Notation
    { joint-iso-ccitt( 2) remote-operations( 4) notation( 0) }
-- Data Types --
universalFailure FROM CSTA-error-definition
    { iso( 1) identified-organization( 3) icd-ecma( 12)
        standard( 0) csta2( 218) error-definition( 120) }
DeviceID FROM CSTA-device-identifiers
    { iso( 1) identified-organization( 3) icd-ecma( 12)
        standard( 0) csta2( 218) device-identifiers( 123) }
ConnectionID FROM CSTA-call-connection-identifiers
    { iso( 1) identified-organization( 3) icd-ecma( 12)
        standard( 0) csta2( 218) call-connection-identifiers( 124) }
ConnectionList FROM CSTA-connection-states
    { iso( 1) identified-organization( 3) icd-ecma( 12)
        standard( 0) csta2( 218) connection-states( 125) }
AccountInfo, AuthCode, CorrelatorData, DeviceProfile, ParticipationType
    FROM CSTA-device-feature-types
        { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) device-feature-types( 127) }
CSTACommonArguments FROM CSTA-extension-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
        standard( 0) csta2( 218) extension-types( 129) };

singleStepConf      OPERATION
    ARGUMENT        SingleStepConfArgument
    RESULT          SingleStepConfResult
    ERRORS          {universalFailure}
::= 20

SingleStepConfArgument ::=
    SEQUENCE
    {activeCall             ConnectionID,
     deviceToJoin           DeviceID,
     participationType      ParticipationType           OPTIONAL,
     joiningDeviceProfile   DeviceProfile               OPTIONAL,
     accountCode            [0] IMPLICIT AccountInfo    OPTIONAL,
     authCode               [1] IMPLICIT AuthCode       OPTIONAL,
     correlatorData         [2] IMPLICIT CorrelatorData OPTIONAL,
     extensions             CSTACommonArguments         OPTIONAL}
```

```
SingleStepConfResult ::=
     CHOICE
     {conferencedCall                ConnectionID,
      SEQUENCE
               {conferencedCall      ConnectionID,
                extensions           CSTACommonArguments} }

END  -- of CSTA-single-step-conference
```

9.20 Single step transfer

```
CSTA-single-step-transfer
    { iso( 1) identified-organization( 3) icd-ecma( 12)
        standard( 0) csta2( 218) single-step-conference( 50) }
DEFINITIONS ::=
BEGIN
IMPORTS
OPERATION, ERROR FROM Remote-Operations-Notation
    { joint-iso-ccitt( 2) remote-operations( 4) notation( 0) }
-- Data Types --
universalFailure FROM CSTA-error-definition
    { iso( 1) identified-organization( 3) icd-ecma( 12)
        standard( 0) csta2( 218) error-definition( 120) }
DeviceID FROM CSTA-device-identifiers
    { iso( 1) identified-organization( 3) icd-ecma( 12)
        standard( 0) csta2( 218) device-identifiers( 123) }
ConnectionID FROM CSTA-call-connection-identifiers
    { iso( 1) identified-organization( 3) icd-ecma( 12)
        standard( 0) csta2( 218) call-connection-identifiers( 124) }
ConnectionList FROM CSTA-connection-states
    { iso( 1) identified-organization( 3) icd-ecma( 12)
        standard( 0) csta2( 218) connection-states( 125) }
AccountInfo, AuthCode, CorrelatorData, DeviceProfile FROM CSTA-device-feature-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
        standard( 0) csta2( 218) device-feature-types( 127) }
CSTACommonArguments FROM CSTA-extension-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
        standard( 0) csta2( 218) extension-types( 129) };

singleStepTrans     OPERATION
    ARGUMENT    SingleStepTransArgument
    RESULT      SingleStepTransResult
    ERRORS      {universalFailure}
::= 50

SingleStepTransArgument ::=
    SEQUENCE
    {activeCall             ConnectionID,
     deviceToTransferTo     DeviceID,
     transferToDeviceProfile DeviceProfile              OPTIONAL,
     accountCode            [0] IMPLICIT AccountInfo    OPTIONAL,
     authCode               [1] IMPLICIT AuthCode       OPTIONAL,
     correlatorData         [2] IMPLICIT CorrelatorData OPTIONAL,
     extensions             CSTACommonArguments         OPTIONAL}

SingleStepTransResult ::=
    SEQUENCE
    {transferedCall         ConnectionID                OPTIONAL,
```

```
    connections             ConnectionList          OPTIONAL,
    extensions              CSTACommonArguments     OPTIONAL}

END  -- of CSTA-single-step-transfer
```

9.21 Transfer call

```
CSTA-transfer-call
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) transfer-call( 16) }
DEFINITIONS ::=
BEGIN
IMPORTS
OPERATION, ERROR FROM Remote-Operations-Notation
    { joint-iso-ccitt( 2) remote-operations( 4) notation( 0) }
-- Data Types --
universalFailure FROM CSTA-error-definition
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) error-definition( 120) }
ConnectionID  FROM CSTA-call-connection-identifiers
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) call-connection-identifiers( 124) }
ConnectionList FROM CSTA-connection-states
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) connection-states( 125) }
ConnectionDetails FROM CSTA-device-feature-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) device-feature-types( 127) }
CSTACommonArguments FROM CSTA-extension-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) extension-types( 129) };

transferCall        OPERATION
    ARGUMENT    TransferCallArgument
    RESULT      TransferCallResult
    ERRORS      {universalFailure}
::= 16

TransferCallArgument ::=
    CHOICE
    {transferInfo   ConnectionDetails,
     SEQUENCE
            {transferInfo   ConnectionDetails,
             extensions     CSTACommonArguments} }

TransferCallResult ::=
    SEQUENCE
    {transferredCall ConnectionID                   OPTIONAL,
     connections    ConnectionList                  OPTIONAL,
     extensions     CSTACommonArguments             OPTIONAL}

END  -- of CSTA-transfer-call
```

10 Event report service

This clause defines the protocol for the event services of CSTA, using ASN.1. Text descriptions of the CSTA services are provided in Standard ECMA-217.

NOTE 3

The range of services supported on a particular association is specified in the application context at association time.

The structure of an event is as described by the remote operation 'cSTAEventReport'. This is a ROSE class 5 operation which has an argument containing some common information (that is provided for all events), an indication of the event type, and other information associated with that event.

```
CSTA-event-report-definitions
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) event-report-definitions( 21) }
DEFINITIONS ::=
BEGIN
IMPORTS
OPERATION FROM Remote-Operations-Notation
    { joint-iso-ccitt( 2) remote-operations( 4) notation( 0) }
-- Data Types --
CallClearedEvent FROM CSTA-call-cleared-event
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) call-cleared-event( 22) }
ConferencedEvent FROM CSTA-conferenced-event
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) conferenced-event( 23) }
ConnectionClearedEvent FROM CSTA-connection-cleared-event
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) connection-cleared-event( 24) }
DeliveredEvent FROM CSTA-delivered-event
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) delivered-event( 25) }
DivertedEvent FROM CSTA-diverted-event
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) diverted-event( 26) }
EstablishedEvent FROM CSTA-established-event
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) established-event( 27) }
FailedEvent FROM CSTA-failed-event
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) failed-event( 28) }
HeldEvent FROM CSTA-held-event
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) held-event( 29) }
NetworkReachedEvent FROM CSTA-network-reached-event
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) network-reached-event( 30) }
OriginatedEvent FROM CSTA-originated-event
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) originated-event( 31) }
```

```
QueuedEvent FROM CSTA-queued-event
    { iso( 1) identified-organization( 3) icd-ecma( 12)
        standard( 0) csta2( 218) queued-event( 32) }
RetrievedEvent FROM CSTA-retrieved-event
    { iso( 1) identified-organization( 3) icd-ecma( 12)
        standard( 0) csta2( 218) retrieved-event( 33) }
ServiceInitiatedEvent FROM CSTA-service-initiated-event
    { iso( 1) identified-organization( 3) icd-ecma( 12)
        standard( 0) csta2( 218) service-initiated-event( 34) }
TransferedEvent FROM CSTA-transferred-event
    { iso( 1) identified-organization( 3) icd-ecma( 12)
        standard( 0) csta2( 218) transferred-event( 35) }
AutoAnswerEvent FROM CSTA-auto-answer-event
    { iso( 1) identified-organization( 3) icd-ecma( 12)
        standard( 0) csta2( 218) auto-answer-event( 40) }
CallInformationEvent FROM CSTA-call-information-event
    { iso( 1) identified-organization( 3) icd-ecma( 12)
        standard( 0) csta2( 218) call-information-event( 41) }
DoNotDisturbEvent FROM CSTA-do-not-disturb-event
    { iso( 1) identified-organization( 3) icd-ecma( 12)
        standard( 0) csta2( 218) do-not-disturb-event( 42) }
ForwardingEvent FROM CSTA-forwarding-event
    { iso( 1) identified-organization( 3) icd-ecma( 12)
        standard( 0) csta2( 218) forwarding-event( 43) }
MessageWaitingEvent FROM CSTA-message-waiting-event
    { iso( 1) identified-organization( 3) icd-ecma( 12)
        standard( 0) csta2( 218) message-waiting-event( 44) }
MicrophoneMuteEvent FROM CSTA-microphone-mute-event
    { iso( 1) identified-organization( 3) icd-ecma( 12)
        standard( 0) csta2( 218) microphone-mute-event( 45) }
SpeakerMuteEvent FROM CSTA-speaker-mute-event
    { iso( 1) identified-organization( 3) icd-ecma( 12)
        standard( 0) csta2( 218) speaker-mute-event( 46) }
SpeakerVolumeEvent FROM CSTA-speaker-volume-event
    { iso( 1) identified-organization( 3) icd-ecma( 12)
        standard( 0) csta2( 218) speaker-volume-event( 47) }
AgentBusyEvent FROM CSTA-agent-busy-event
    { iso( 1) identified-organization( 3) icd-ecma( 12)
        standard( 0) csta2( 218) agent-busy-event( 56) }
LoggedOnEvent FROM CSTA-logged-on-event
    { iso( 1) identified-organization( 3) icd-ecma( 12)
        standard( 0) csta2( 218) logged-on-event( 51) }
LoggedOffEvent FROM CSTA-logged-off-event
    { iso( 1) identified-organization( 3) icd-ecma( 12)
        standard( 0) csta2( 218) logged-off-event( 52) }
NotReadyEvent FROM CSTA-not-ready-event
    { iso( 1) identified-organization( 3) icd-ecma( 12)
        standard( 0) csta2( 218) not-ready-event( 53) }
```

```
ReadyEvent FROM CSTA-ready-event
    { iso( 1) identified-organization( 3) icd-ecma( 12)
        standard( 0) csta2( 218) ready-event( 54) }
WorkingAfterCallEvent FROM CSTA-working-after-call-event
    { iso( 1) identified-organization( 3) icd-ecma( 12)
        standard( 0) csta2( 218) working-after-call-event( 55) }
BackInServiceEvent FROM CSTA-back-in-service-event
    { iso( 1) identified-organization( 3) icd-ecma( 12)
        standard( 0) csta2( 218) back-in-service-event( 61) }
OutOfServiceEvent FROM CSTA-out-of-service-event
    { iso( 1) identified-organization( 3) icd-ecma( 12)
        standard( 0) csta2( 218) out-of-service-event( 62) }
PrivateEvent FROM CSTA-private-event
    { iso( 1) identified-organization( 3) icd-ecma( 12)
        standard( 0) csta2( 218) private-event( 71) }
PlayEvent FROM CSTA-play-event
    { iso( 1) identified-organization( 3) icd-ecma( 12)
        standard( 0) csta2( 218) play( 75) }
RecordEvent FROM CSTA-record-event
    { iso( 1) identified-organization( 3) icd-ecma( 12)
        standard( 0) csta2( 218) record( 76) }
ReviewEvent FROM CSTA-review-event
    { iso( 1) identified-organization( 3) icd-ecma( 12)
        standard( 0) csta2( 218) review( 77) }
StopEvent FROM CSTA-stop-event
    { iso( 1) identified-organization( 3) icd-ecma( 12)
        standard( 0) csta2( 218) stop( 78) }
SuspendPlayEvent FROM CSTA-suspend-play-event
    { iso( 1) identified-organization( 3) icd-ecma( 12)
        standard( 0) csta2( 218) suspend-play( 79) }
SuspendRecordEvent FROM CSTA-suspend-record-event
    { iso( 1) identified-organization( 3) icd-ecma( 12)
        standard( 0) csta2( 218) suspend-record( 80) }
VoiceAttributesChangeEvent FROM CSTA-voice-attributes-change-event
    { iso( 1) identified-organization( 3) icd-ecma( 12)
        standard( 0) csta2( 218) voice-attributes-change-event( 74) }
MonitorCrossRefID FROM CSTA-status-reporting
    { iso( 1) identified-organization( 3) icd-ecma( 12)
        standard( 0) csta2( 218) status-reporting( 126) }
CSTACommonArguments FROM CSTA-extension-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
        standard( 0) csta2( 218) extension-types( 129) };

cSTAEventReport   OPERATION
    ARGUMENT    CSTAEventReportArgument
::= 21
```

```
CSTAEventReportArgument ::=
    SEQUENCE
    {crossRefIdentifier      MonitorCrossRefID,
     eventSpecificInfo       EventSpecificInfo,
     extensions              CSTACommonArguments      OPTIONAL}

EventSpecificInfo ::=
    -- The event type (Call Cleared, Conferenced etc.) is encoded within this parameter
    CHOICE
    {callEvent                    [0] CallEvent,
     featureEvent                 [1] FeatureEvent,
     agentStateEvent              [2] AgentStateEvent,
     maintenanceEvent             [3] MaintenanceEvent,
     privateEvent                 [4] PrivateEvent,
     voiceUnitEvent               [5] VoiceUnitEvent              }


CallEvent ::=
    CHOICE
    {callClearedEvent             [0] CallClearedEvent,
     conferencedEvent             [1] ConferencedEvent,
     connectionClearedEvent       [2] ConnectionClearedEvent,
     deliveredEvent               [3] DeliveredEvent,
     divertedEvent                [4] DivertedEvent,
     establishedEvent             [5] EstablishedEvent,
     failedEvent                  [6] FailedEvent,
     heldEvent                    [7] HeldEvent,
     networkReachedEvent          [8] NetworkReachedEvent,
     originatedEvent              [9] OriginatedEvent,
     queuedEvent                  [10] QueuedEvent,
     retrievedEvent               [11] RetrievedEvent,
     serviceInitiatedEvent        [12] ServiceInitiatedEvent,
     transferedEvent              [13] TransferedEvent            }

FeatureEvent ::=
    CHOICE
    {autoAnswerEvent              [0] AutoAnswerEvent,
     callInformationEvent         [1] CallInformationEvent,
     doNotDisturbEvent            [2] DoNotDisturbEvent,
     forwardingEvent              [3] ForwardingEvent,
     messageWaitingEvent          [4] MessageWaitingEvent,
     microphoneMuteEvent          [5] MicrophoneMuteEvent,
     speakerMuteEvent             [6] SpeakerMuteEvent,
     speakerVolumeEvent           [7] SpeakerVolumeEvent                }

AgentStateEvent ::=
    CHOICE
```

```
      {agentBusyEvent                [0] AgentBusyEvent,
       loggedOnEvent                 [1] LoggedOnEvent,
       loggedOffEvent                [2] LoggedOffEvent,
       notReadyEvent                 [3] NotReadyEvent,
       readyEvent                    [4] ReadyEvent,
       workingAfterCallEvent         [5] WorkingAfterCallEvent       }

MaintenanceEvent ::=
      CHOICE
      {backInServiceEvent            [0] BackInServiceEvent,
       outOfServiceEvent             [1] OutOfServiceEvent           }

VoiceUnitEvent ::=
      CHOICE
      {playEvent                     [0] PlayEvent,
       recordEvent                   [1] RecordEvent,
       reviewEvent                   [2] ReviewEvent,
       stopEvent                     [3] StopEvent,
       suspendPlayEvent              [4] SuspendPlayEvent,
       suspendRecordEvent            [5] SuspendRecordEvent,
       voiceAttributesChangeEvent    [6] VoiceAttributesChangeEvent }

END  -- of CSTA-event-report-definitions
```

10.1 Call events

Each event contains a ConnectionID which identifies the object of interest. Other parameters may also be used to identify the relevant devices.

Events may result from a call interacting with switch features that had been previously set or invoked. These features and their settings may be reflected in an Event as a value of the EventCause parameter.

10.1.1 Call cleared

```
CSTA-call-cleared-event
     { iso( 1) identified-organization( 3) icd-ecma( 12)
               standard( 0) csta2( 218) call-cleared-event( 22) }
DEFINITIONS ::=
BEGIN
IMPORTS
-- Data Types --
EventCause FROM CSTA-event-causes
     { iso( 1) identified-organization( 3) icd-ecma( 12)
               standard( 0) csta2( 218) event-causes( 121) }
ConnectionID FROM CSTA-call-connection-identifiers
     { iso( 1) identified-organization( 3) icd-ecma( 12)
               standard( 0) csta2( 218) call-connection-identifiers( 124) }
LocalConnectionState  FROM  CSTA-connection-states
     { iso( 1) identified-organization( 3) icd-ecma( 12)
               standard( 0) csta2( 218) connection-states( 125) }
CorrelatorData FROM CSTA-device-feature-types
     { iso( 1) identified-organization( 3) icd-ecma( 12)
               standard( 0) csta2( 218) device-feature-types( 127) };

CallClearedEvent ::=
     SEQUENCE
     {clearedCall             ConnectionID,
      localConnectionInfo     LocalConnectionState        OPTIONAL,
      correlatorData          CorrelatorData              OPTIONAL,
      cause                   EventCause                  OPTIONAL}

END  -- of CSTA-call-cleared-event
```

10.1.2 Conferenced

```
CSTA-conferenced-event
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) conferenced-event( 23) }
DEFINITIONS ::=
BEGIN
IMPORTS
-- Data Types --
EventCause FROM CSTA-event-causes
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) event-causes( 121) }
SubjectDeviceID FROM CSTA-device-identifiers
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) device-identifiers( 123) }
ConnectionID FROM CSTA-call-connection-identifiers
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) call-connection-identifiers( 124) }
LocalConnectionState, ConnectionList FROM CSTA-connection-states
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) connection-states( 125) }
CorrelatorData FROM CSTA-device-feature-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) device-feature-types( 127) };

ConferencedEvent ::=
    SEQUENCE
    {primaryOldCall          ConnectionID,
     secondaryOldCall        ConnectionID                OPTIONAL,
     confController          SubjectDeviceID,
     addedParty              SubjectDeviceID,
     conferenceConnections   ConnectionList              OPTIONAL,
     localConnectionInfo     LocalConnectionState        OPTIONAL,
     correlatorData          CorrelatorData              OPTIONAL,
     cause                   EventCause                  OPTIONAL}

END  -- of CSTA-conferenced-event
```

10.1.3 Connection cleared

```
CSTA-connection-cleared-event
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) connection-cleared-event( 24) }
DEFINITIONS ::=
BEGIN
IMPORTS
-- Data Types --
EventCause FROM CSTA-event-causes
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) event-causes( 121) }
SubjectDeviceID FROM CSTA-device-identifiers
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) device-identifiers( 123) }
ConnectionID FROM CSTA-call-connection-identifiers
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) call-connection-identifiers( 124) }
LocalConnectionState FROM CSTA-connection-states
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) connection-states( 125) }
CorrelatorData FROM CSTA-device-feature-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) device-feature-types( 127) };

ConnectionClearedEvent ::=
    SEQUENCE
    {droppedConnection      ConnectionID,
     releasingDevice        SubjectDeviceID,
     localConnectionInfo    LocalConnectionState    OPTIONAL,
     correlatorData         CorrelatorData          OPTIONAL,
     cause                  EventCause              OPTIONAL}

END  -- of CSTA-connection-cleared-event
```

10.1.4 Delivered

```
CSTA-delivered-event
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) delivered-event( 25) }
DEFINITIONS ::=
BEGIN
IMPORTS
-- Data Types --
EventCause FROM CSTA-event-causes
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) event-causes( 121) }
SubjectDeviceID, CallingDeviceID, CalledDeviceID, RedirectionDeviceID
    FROM CSTA-device-identifiers
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) device-identifiers( 123) }
ConnectionID FROM CSTA-call-connection-identifiers
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) call-connection-identifiers( 124) }
LocalConnectionState FROM CSTA-connection-states
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) connection-states( 125) }
CorrelatorData FROM CSTA-device-feature-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) device-feature-types( 127) };

DeliveredEvent ::=
    SEQUENCE
    {connection             ConnectionID,
     alertingDevice         SubjectDeviceID,
     callingDevice          CallingDeviceID,
     calledDevice           CalledDeviceID,
     lastRedirectionDevice  RedirectionDeviceID,
     originatingConnection  ConnectionID            OPTIONAL,
     localConnectionInfo    LocalConnectionState    OPTIONAL,
     correlatorData         CorrelatorData          OPTIONAL,
     cause                  EventCause              OPTIONAL}

END  -- of CSTA-delivered-event
```

10.1.5 Diverted

```
CSTA-diverted-event
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) diverted-event( 26) }
DEFINITIONS ::=
BEGIN
IMPORTS
-- Data Types --
EventCause FROM CSTA-event-causes
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) event-causes( 121) }
SubjectDeviceID FROM CSTA-device-identifiers
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) device-identifiers( 123) }
ConnectionID FROM CSTA-call-connection-identifiers
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) call-connection-identifiers( 124) }
LocalConnectionState FROM CSTA-connection-states
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) connection-states( 125) }
CorrelatorData FROM CSTA-device-feature-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) device-feature-types( 127) };

DivertedEvent ::= SEQUENCE
    {connection             ConnectionID            OPTIONAL,
     divertingDevice        SubjectDeviceID,
     newDestination         SubjectDeviceID,
     localConnectionInfo    LocalConnectionState    OPTIONAL,
     correlatorData         CorrelatorData          OPTIONAL,
     cause                  EventCause              OPTIONAL}

END  -- of diverted-event
```

10.1.6 Established

```
CSTA-established-event
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) established-event( 27) }
DEFINITIONS ::=
BEGIN
IMPORTS
-- Data Types --
LocalConnectionState FROM CSTA-connection-states
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) connection-states( 125) }
SubjectDeviceID, CalledDeviceID, CallingDeviceID, RedirectionDeviceID
    FROM CSTA-device-identifiers
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) device-identifiers( 123) }
ConnectionID FROM CSTA-call-connection-identifiers
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) call-connection-identifiers( 124) }
CorrelatorData FROM CSTA-device-feature-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) device-feature-types( 127) }
EventCause FROM CSTA-event-causes
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) event-causes( 121) };

EstablishedEvent ::=
    SEQUENCE
    {establishedConnection  ConnectionID,
     answeringDevice        SubjectDeviceID,
     callingDevice          CallingDeviceID,
     calledDevice           CalledDeviceID,
     lastRedirectionDevice  RedirectionDeviceID,
     originatingConnection  ConnectionID          OPTIONAL,
     localConnectionInfo    LocalConnectionState  OPTIONAL,
     correlatorData         CorrelatorData        OPTIONAL,
     cause                  EventCause            OPTIONAL}

END  -- of CSTA-established-event
```

10.1.7 Failed

```
CSTA-failed-event
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) failed-event( 28) }
DEFINITIONS ::=
BEGIN
IMPORTS
-- Data Types --
EventCause FROM CSTA-event-causes
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) event-causes( 121) }
SubjectDeviceID, CalledDeviceID FROM CSTA-device-identifiers
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) device-identifiers( 123) }
ConnectionID FROM CSTA-call-connection-identifiers
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) call-connection-identifiers( 124) }
LocalConnectionState FROM CSTA-connection-states
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) connection-states( 125) }
CorrelatorData FROM CSTA-device-feature-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) device-feature-types( 127) };

FailedEvent ::=
    SEQUENCE
    {failedConnection       ConnectionID,
     failingDevice          SubjectDeviceID,
     calledDevice           CalledDeviceID,
     localConnectionInfo    LocalConnectionState   OPTIONAL,
     correlatorData         CorrelatorData         OPTIONAL,
     cause                  EventCause             OPTIONAL}

END  -- of CSTA-failed-event
```

10.1.8 Held

```
CSTA-held-event
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) held-event( 29) }
DEFINITIONS ::=
BEGIN
IMPORTS
-- Data Types --
EventCause FROM CSTA-event-causes
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) event-causes( 121) }
SubjectDeviceID FROM CSTA-device-identifiers
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) device-identifiers( 123) }
ConnectionID FROM CSTA-call-connection-identifiers
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) call-connection-identifiers( 124) }
LocalConnectionState FROM CSTA-connection-states
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) connection-states( 125) }
CorrelatorData FROM CSTA-device-feature-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) device-feature-types( 127) };

HeldEvent ::=
    SEQUENCE
    {heldConnection         ConnectionID,
     holdingDevice          SubjectDeviceID,
     localConnectionInfo    LocalConnectionState   OPTIONAL,
     correlatorData         CorrelatorData         OPTIONAL,
     cause                  EventCause             OPTIONAL}

END  -- of CSTA-held-event
```

10.1.9 Network reached

```
CSTA-network-reached-event
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) network-reached-event( 30) }
DEFINITIONS ::=
BEGIN
IMPORTS
-- Data Types --
EventCause FROM CSTA-event-causes
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) event-causes( 121) }
SubjectDeviceID, CalledDeviceID FROM CSTA-device-identifiers
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) device-identifiers( 123) }
ConnectionID FROM CSTA-call-connection-identifiers
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) call-connection-identifiers( 124) }
LocalConnectionState FROM CSTA-connection-states
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) connection-states( 125) }
CorrelatorData FROM CSTA-device-feature-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) device-feature-types( 127) };

NetworkReachedEvent ::=
    SEQUENCE
    {outboundConnection   ConnectionID,
     trunkUsed            SubjectDeviceID,
     calledDevice         CalledDeviceID,
     localConnectionInfo  LocalConnectionState  OPTIONAL,
     correlatorData       CorrelatorData        OPTIONAL,
     cause                EventCause            OPTIONAL}

END  -- of CSTA-network-reached-event
```

10.1.10 Originated

```
CSTA-originated-event
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) originated-event( 31) }
DEFINITIONS ::=
BEGIN
IMPORTS
-- Data Types --
EventCause FROM CSTA-event-causes
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) event-causes( 121) }
CalledDeviceID, CallingDeviceID, SubjectDeviceID FROM CSTA-device-identifiers
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) device-identifiers( 123) }
ConnectionID FROM CSTA-call-connection-identifiers
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) call-connection-identifiers( 124) }
LocalConnectionState FROM CSTA-connection-states
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) connection-states( 125) }
CorrelatorData FROM CSTA-device-feature-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) device-feature-types( 127) };

OriginatedEvent ::=
    SEQUENCE
    {originatedConnection   ConnectionID,
     callingDevice          CallingDeviceID,
     calledDevice           CalledDeviceID,
     originatingDevice      SubjectDeviceID       OPTIONAL,
     localConnectionInfo    LocalConnectionState  OPTIONAL,
     correlatorData         CorrelatorData        OPTIONAL,
     cause                  EventCause            OPTIONAL}

END  -- of CSTA-originated-event
```

10.1.11 Queued

```
CSTA-queued-event
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) queued-event( 32) }
DEFINITIONS    ::=
BEGIN
IMPORTS
-- Data Types --
EventCause FROM CSTA-event-causes
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) event-causes( 121) }
CalledDeviceID, CallingDeviceID, RedirectionDeviceID, SubjectDeviceID
    FROM CSTA-device-identifiers
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) device-identifiers( 123) }
ConnectionID FROM CSTA-call-connection-identifiers
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) call-connection-identifiers( 124) }
LocalConnectionState FROM CSTA-connection-states
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) connection-states( 125) }
CorrelatorData, NoOfCallsInQueue, NoOfCallsInFront FROM CSTA-device-feature-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) device-feature-types( 127) };

QueuedEvent ::=
    SEQUENCE
    {queuedConnection      ConnectionID,
     queue                 SubjectDeviceID,
     callingDevice         CallingDeviceID,
     calledDevice          CalledDeviceID,
     lastRedirectionDevice RedirectionDeviceID,
     numberQueued          [0] IMPLICIT   NoOfCallsInQueue    OPTIONAL,
     callsInFront          [1] IMPLICIT   NoOfCallsInFront    OPTIONAL,
     localConnectionInfo   LocalConnectionState               OPTIONAL,
     correlatorData        CorrelatorData                     OPTIONAL,
     cause                 EventCause                         OPTIONAL}

END  -- of CSTA-queued-event
```

10.1.12 Retrieved

```
CSTA-retrieved-event
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) retrieved-event( 33) }
DEFINITIONS ::=
BEGIN
IMPORTS
-- Data Types --
EventCause FROM CSTA-event-causes
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) event-causes( 121) }
SubjectDeviceID FROM CSTA-device-identifiers
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) device-identifiers( 123) }
ConnectionID FROM CSTA-call-connection-identifiers
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) call-connection-identifiers( 124) }
LocalConnectionState FROM CSTA-connection-states
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) connection-states( 125) }
CorrelatorData FROM CSTA-device-feature-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) device-feature-types( 127) };

RetrievedEvent ::=
    SEQUENCE
    {retrievedConnection    ConnectionID,
     retrievingDevice       SubjectDeviceID,
     localConnectionInfo    LocalConnectionState   OPTIONAL,
     correlatorData         CorrelatorData         OPTIONAL,
     cause                  EventCause             OPTIONAL}

END  -- of CSTA-retrieved-event
```

10.1.13 Service initiated

```
CSTA-service-initiated-event
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) service-initiated-event( 34) }
DEFINITIONS ::=
BEGIN
IMPORTS
-- Data Types --
EventCause FROM CSTA-event-causes
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) event-causes( 121) }
ConnectionID FROM CSTA-call-connection-identifiers
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) call-connection-identifiers( 124) }
LocalConnectionState FROM CSTA-connection-states
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) connection-states( 125) };

ServiceInitiatedEvent ::=
    SEQUENCE
    {initiatedConnection      ConnectionID,
     localConnectionInfo      LocalConnectionState   OPTIONAL,
     cause                    EventCause             OPTIONAL}

END  -- of CSTA-service-initiated-event
```

10.1.14 Transferred

```
CSTA-transferred-event
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) transferred-event( 35) }
DEFINITIONS ::=
BEGIN
IMPORTS
-- Data Types --
EventCause FROM CSTA-event-causes
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) event-causes( 121) }
SubjectDeviceID FROM CSTA-device-identifiers
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) device-identifiers( 123) }
ConnectionID FROM CSTA-call-connection-identifiers
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) call-connection-identifiers( 124) }
LocalConnectionState, ConnectionList FROM CSTA-connection-states
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) connection-states( 125) }
CorrelatorData FROM CSTA-device-feature-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) device-feature-types( 127) };

TransferedEvent ::=
    SEQUENCE
    {primaryOldCall          ConnectionID,
     secondaryOldCall        ConnectionID          OPTIONAL,
     transferringDevice      SubjectDeviceID,
     transferredDevice       SubjectDeviceID,
     transferredConnections  ConnectionList        OPTIONAL,
     localConnectionInfo     LocalConnectionState  OPTIONAL,
     correlatorData          CorrelatorData        OPTIONAL,
     cause                   EventCause            OPTIONAL}

END  -- of CSTA-transferred-event
```

10.2 Feature events

10.2.1 Auto answer

```
CSTA-auto-answer-event
     { iso( 1) identified-organization( 3) icd-ecma( 12)
               standard( 0) csta2( 218) auto-answer-event( 40) }
DEFINITIONS ::=
BEGIN
IMPORTS
-- Data Types --
SubjectDeviceID FROM CSTA-device-identifiers
     { iso( 1) identified-organization( 3) icd-ecma( 12)
               standard( 0) csta2( 218) device-identifiers( 123) };

AutoAnswerEvent ::=
     SEQUENCE
     {device                   SubjectDeviceID,
      autoAnswerOn             BOOLEAN}

END  -- of CSTA-auto-answer-event
```

10.2.2 Call information

```
CSTA-call-information-event
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) call-information-event( 41) }
DEFINITIONS ::=
BEGIN
IMPORTS
-- Data Types --
SubjectDeviceID FROM CSTA-device-identifiers
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) device-identifiers( 123) }
ConnectionID FROM CSTA-call-connection-identifiers
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) call-connection-identifiers( 124) }
AccountInfo, AuthCode, CorrelatorData FROM CSTA-device-feature-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) device-feature-types( 127) };

CallInformationEvent ::=
    SEQUENCE
    {connection          ConnectionID,
     device              SubjectDeviceID,
     accountInfo         [0] IMPLICIT   AccountInfo     OPTIONAL,
     authorisationCode   [1] IMPLICIT   AuthCode        OPTIONAL,
     correlatorData      [2] IMPLICIT   CorrelatorData  OPTIONAL}

END  -- of CSTA-call-information-event
```

10.2.3 Do not disturb

```
CSTA-do-not-disturb-event
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) do-not-disturb-event( 42) }
DEFINITIONS ::=
BEGIN
IMPORTS
-- Data Types --
SubjectDeviceID FROM CSTA-device-identifiers
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) device-identifiers( 123) };

DoNotDisturbEvent ::=
    SEQUENCE
    {device                 SubjectDeviceID,
     doNotDisturbOn         BOOLEAN}

END  -- of CSTA-do-not-disturb-event
```

10.2.4 Forwarding

```
CSTA-forwarding-event
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) forwarding-event( 43) }
DEFINITIONS ::=
BEGIN
IMPORTS
-- Data Types --
SubjectDeviceID FROM CSTA-device-identifiers
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) device-identifiers( 123) }
ForwardParameter FROM CSTA-device-feature-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) device-feature-types( 127) };

ForwardingEvent ::=
    SEQUENCE
    {device                 SubjectDeviceID,
     forwardingInformation  ForwardParameter,
     forwardedTo            SubjectDeviceID          OPTIONAL}

END  -- of CSTA-forwarding-event
```

10.2.5 Message waiting

```
CSTA-message-waiting-event
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) message-waiting-event( 44) }
DEFINITIONS ::=
BEGIN
IMPORTS
-- Data Types --
SubjectDeviceID FROM CSTA-device-identifiers
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) device-identifiers( 123) };

MessageWaitingEvent ::=
    SEQUENCE
    {device                 SubjectDeviceID,
     invokingDevice         SubjectDeviceID,
     messageWaitingOn       BOOLEAN}

END  -- of CSTA-message-waiting-event
```

10.2.6 Microphone mute

```
CSTA-microphone-mute-event
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) microphone-mute-event( 45) }
DEFINITIONS ::=
BEGIN
IMPORTS
-- Data Types --
SubjectDeviceID FROM CSTA-device-identifiers
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) device-identifiers( 123) };

MicrophoneMuteEvent ::=
    SEQUENCE
    {device                 SubjectDeviceID,
     microphoneMuteOn       BOOLEAN}

END  -- of CSTA-microphone-mute-event
```

10.2.7 Speaker mute

```
CSTA-speaker-mute-event
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) speaker-mute-event( 46) }
DEFINITIONS ::=
BEGIN
IMPORTS
-- Data Types --
SubjectDeviceID FROM CSTA-device-identifiers
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) device-identifiers( 123) };

SpeakerMuteEvent ::=
    SEQUENCE
    {device                 SubjectDeviceID,
     speakerMuteOn          BOOLEAN}

END  -- of CSTA-speaker-mute-event
```

10.2.8 Speaker volume

```
CSTA-speaker-volume-event
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) speaker-volume-event( 47) }
DEFINITIONS ::=
BEGIN
IMPORTS
-- Data Types --
SubjectDeviceID FROM CSTA-device-identifiers
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) device-identifiers( 123) };

SpeakerVolumeEvent ::=
    SEQUENCE
    {device                 SubjectDeviceID,
     speakerVolume          INTEGER (0 .. 100)}

END  -- of CSTA-speaker-volume-event
```

10.3 Agent state events

10.3.1 Agent busy

```
CSTA-agent-busy-event
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) agent-busy-event( 56) }
DEFINITIONS ::=
BEGIN
IMPORTS
-- Data Types --
EventCause FROM CSTA-event-causes
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) event-causes( 121) }
SubjectDeviceID FROM CSTA-device-identifiers
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) device-identifiers( 123) }
AgentID, AgentGroup FROM CSTA-device-feature-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) device-feature-types( 127) };

AgentBusyEvent ::=
    SEQUENCE
    {agentDevice    SubjectDeviceID,
     agentID        [10] IMPLICIT   AgentID         OPTIONAL,
     agentGroup     AgentGroup                      OPTIONAL,
     cause          EventCause                      OPTIONAL}

END  -- of CSTA-agent-busy-event
```

10.3.2 Agent logged on

```
CSTA-logged-on-event
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) logged-on-event( 51) }
DEFINITIONS ::=
BEGIN
IMPORTS
-- Data Types --
EventCause FROM CSTA-event-causes
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) event-causes( 121) }
SubjectDeviceID FROM CSTA-device-identifiers
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) device-identifiers( 123) }
AgentID, AgentGroup, AgentPassword FROM CSTA-device-feature-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) device-feature-types( 127) };

LoggedOnEvent ::=
    SEQUENCE
    {agentDevice    SubjectDeviceID,
     agentID        [10] IMPLICIT   AgentID         OPTIONAL,
     agentGroup     AgentGroup                      OPTIONAL,
     password       AgentPassword                   OPTIONAL,
     cause          EventCause                      OPTIONAL}

END  -- of CSTA-logged-on-event
```

10.3.3 Agent logged off

```
CSTA-logged-off-event
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) logged-off-event( 52) }
DEFINITIONS ::=
BEGIN
IMPORTS
-- Data Types --
EventCause FROM CSTA-event-causes
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) event-causes( 121) }
SubjectDeviceID FROM CSTA-device-identifiers
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) device-identifiers( 123) }
AgentID, AgentGroup, AgentPassword FROM CSTA-device-feature-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) device-feature-types( 127) };

LoggedOffEvent ::=
    SEQUENCE
    {agentDevice    SubjectDeviceID,
     agentID        [10] IMPLICIT   AgentID         OPTIONAL,
     agentGroup     AgentGroup                      OPTIONAL,
     password       AgentPassword                   OPTIONAL,
     cause          EventCause                      OPTIONAL}

END  -- of CSTA-logged-off-event
```

10.3.4 Agent not ready

```
CSTA-not-ready-event
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) not-ready-event( 53) }
DEFINITIONS ::=
BEGIN
IMPORTS
-- Data Types --
EventCause FROM CSTA-event-causes
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) event-causes( 121) }
SubjectDeviceID FROM CSTA-device-identifiers
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) device-identifiers( 123)}
AgentID FROM CSTA-device-feature-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) device-feature-types( 127) };

NotReadyEvent ::=
    SEQUENCE
    {agentDevice        SubjectDeviceID,
     agentID            AgentID          OPTIONAL,
     cause              EventCause       OPTIONAL}

END  -- of CSTA-not-ready-event
```

10.3.5 Agent ready

```
CSTA-ready-event
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) ready-event( 54) }
DEFINITIONS ::=
BEGIN
IMPORTS
-- Data Types --
EventCause FROM CSTA-event-causes
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) event-causes( 121) }
SubjectDeviceID FROM CSTA-device-identifiers
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) device-identifiers( 123) }
AgentID FROM CSTA-device-feature-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) device-feature-types( 127) };

ReadyEvent ::=
    SEQUENCE
    {agentDevice    SubjectDeviceID,
     agentID        AgentID             OPTIONAL,
     cause          EventCause          OPTIONAL}

END  -- of CSTA-ready-event
```

10.3.6 Agent working after call

```
CSTA-working-after-call-event
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) working-after-call-event( 55) }
DEFINITIONS ::=
BEGIN
IMPORTS
-- Data Types --
EventCause FROM CSTA-event-causes
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) event-causes( 121) }
SubjectDeviceID FROM CSTA-device-identifiers
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) device-identifiers( 123) }
AgentID, AgentGroup FROM CSTA-device-feature-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) device-feature-types( 127) };

WorkingAfterCallEvent ::=
    SEQUENCE
    {agentDevice    SubjectDeviceID,
     agentID        [10] IMPLICIT   AgentID         OPTIONAL,
     agentGroup     AgentGroup                      OPTIONAL,
     cause          EventCause                      OPTIONAL}

END  -- of CSTA-working-after-call-event
```

10.4 Maintenance events

10.4.1 Back in service

```
CSTA-back-in-service-event
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) back-in-service-event( 61) }
DEFINITIONS ::=
BEGIN
IMPORTS
-- Data Types --
EventCause FROM CSTA-event-causes
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) event-causes( 121) }
DeviceID FROM CSTA-device-identifiers
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) device-identifiers( 123) };

BackInServiceEvent ::=
    SEQUENCE
    {device          DeviceID,
     cause           EventCause     OPTIONAL}

END  -- of CSTA-back-in-service-event
```

10.4.2 Out of service

```
CSTA-out-of-service-event
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) out-of-service-event( 62) }
DEFINITIONS ::=
BEGIN
IMPORTS
-- Data Types --
EventCause FROM CSTA-event-causes
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) event-causes( 121) }
DeviceID FROM CSTA-device-identifiers
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) device-identifiers( 123) };

OutOfServiceEvent ::=
    SEQUENCE
    {device          DeviceID,
     cause           EventCause     OPTIONAL}

END  -- of CSTA-out-of-service-event
```

10.5 Private events

```
CSTA-private-event
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) private-event( 71) }
DEFINITIONS ::=
BEGIN

PrivateEvent ::= NULL       --      The actual encoding of the private event is added here,
                            --      replacing NULL with another valid ASN.1 type.

END  -- of CSTA-private-event
```

10.6. Voice unit events

10.6.1 Play

```
CSTA-play-event
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) play( 75) }
DEFINITIONS ::=
BEGIN
IMPORTS
-- Data Types --
EventCause FROM CSTA-event-causes
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) event-causes( 121) }
ConnectionID FROM CSTA-call-connection-identifiers
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) call-connection-identifiers( 124) }
MessageID, Speed FROM CSTA-device-feature-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) device-feature-types( 127) };

PlayEvent ::=
    SEQUENCE
    {connection      ConnectionID,
     message         MessageID,
     length          [0] IMPLICIT   INTEGER      OPTIONAL,
     currentPosition [1] IMPLICIT   INTEGER      OPTIONAL,
     speed           [2] IMPLICIT   Speed        OPTIONAL,
     cause           EventCause                  OPTIONAL}

END  -- of CSTA-play-event
```

10.6.2 Record

```
CSTA-record-event
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) record( 76) }
DEFINITIONS ::=
BEGIN
IMPORTS
-- Data Types --
EventCause FROM CSTA-event-causes
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) event-causes( 121) }
ConnectionID FROM CSTA-call-connection-identifiers
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) call-connection-identifiers( 124) }
MessageID FROM CSTA-device-feature-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) device-feature-types( 127) };

RecordEvent ::=
    SEQUENCE
    {connection      ConnectionID,
     message         MessageID,
     length          [0] IMPLICIT   INTEGER      OPTIONAL,
     currentPosition [1] IMPLICIT   INTEGER      OPTIONAL,
     cause           EventCause                  OPTIONAL}

END  -- of CSTA-record-event
```

10.6.3 Review

```
CSTA-review-event
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) review( 77) }
DEFINITIONS ::=
BEGIN
IMPORTS
-- Data Types --
EventCause FROM CSTA-event-causes
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) event-causes( 121) }
ConnectionID FROM CSTA-call-connection-identifiers
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) call-connection-identifiers( 124) }
MessageID FROM CSTA-device-feature-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) device-feature-types( 127) };

ReviewEvent ::=
    SEQUENCE
    {connection      ConnectionID,
     message         MessageID,
     length          [0] IMPLICIT   INTEGER      OPTIONAL,
     currentPosition [1] IMPLICIT   INTEGER      OPTIONAL,
     cause           EventCause                  OPTIONAL}

END  -- of CSTA-review-event
```

10.6.4 Stop

```
CSTA-stop-event
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) stop( 78) }
DEFINITIONS ::=
BEGIN
IMPORTS
-- Data Types --
EventCause FROM CSTA-event-causes
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) event-causes( 121) }
ConnectionID FROM CSTA-call-connection-identifiers
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) call-connection-identifiers( 124) }
MessageID FROM CSTA-device-feature-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) device-feature-types( 127) };

StopEvent ::=
    SEQUENCE
    {connection      ConnectionID,
     message         MessageID,
     length          [0] IMPLICIT    INTEGER       OPTIONAL,
     currentPosition [1] IMPLICIT    INTEGER       OPTIONAL,
     cause           EventCause                    OPTIONAL}

END  -- of CSTA-stop-event
```

10.6.5 Suspend play

```
CSTA-suspend-play-event
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) suspend-play( 79) }
DEFINITIONS ::=
BEGIN
IMPORTS
-- Data Types --
EventCause FROM CSTA-event-causes
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) event-causes( 121) }
ConnectionID FROM CSTA-call-connection-identifiers
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) call-connection-identifiers( 124) }
MessageID FROM CSTA-device-feature-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) device-feature-types( 127) };

SuspendPlayEvent ::=
    SEQUENCE
    {connection      ConnectionID,
     message         MessageID,
     length          [0] IMPLICIT    INTEGER      OPTIONAL,
     currentPosition [1] IMPLICIT    INTEGER      OPTIONAL,
     cause           EventCause                   OPTIONAL}

END  -- of CSTA-suspend-play-event
```

10.6.6 Suspend record

```
CSTA-suspend-record-event
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) suspend-record( 80) }
DEFINITIONS ::=
BEGIN
IMPORTS
-- Data Types --
EventCause FROM CSTA-event-causes
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) event-causes( 121) }
ConnectionID FROM CSTA-call-connection-identifiers
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) call-connection-identifiers( 124) }
MessageID FROM CSTA-device-feature-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) device-feature-types( 127) };

SuspendRecordEvent ::=
    SEQUENCE
    {connection      ConnectionID,
     message         MessageID,
     length          [0] IMPLICIT   INTEGER      OPTIONAL,
     currentPosition [1] IMPLICIT   INTEGER      OPTIONAL,
     cause           EventCause                  OPTIONAL}

END  -- of CSTA-suspend-record-event
```

10.6.7 Voice attributes change

```
CSTA-voice-attributes-change-event
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) voice-attributes-change-event( 74) }
DEFINITIONS ::=
BEGIN
IMPORTS
-- Data Types --
EventCause FROM CSTA-event-causes
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) event-causes( 121) }
ConnectionID FROM CSTA-call-connection-identifiers
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) call-connection-identifiers( 124) }
MessageID, Speed FROM CSTA-device-feature-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) device-feature-types( 127) };

VoiceAttributesChangeEvent ::=
    SEQUENCE
    {connection          ConnectionID,
     message             MessageID,
     speakerVolume       [0] IMPLICIT INTEGER (0 .. 100)      OPTIONAL,
     recordingLevel      [1] IMPLICIT INTEGER (0 .. 100)      OPTIONAL,
     speed               [2] IMPLICIT Speed                   OPTIONAL,
     currentPosition     [3] IMPLICIT INTEGER                 OPTIONAL,
     cause               EventCause                           OPTIONAL}

END  -- of CSTA-voice-attributes-change-event
```

11 Computing function services

This clause defines the protocol for the Computing function services of CSTA, using ASN.1. Text descriptions of the CSTA services are provided in Standard ECMA-217.

NOTE 4

The range of services supported on a particular association is specified in the application context at association time.

11.1 Route request

```
CSTA-route-request
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) route-request( 81) }
DEFINITIONS ::=
BEGIN
IMPORTS
OPERATION, ERROR FROM Remote-Operations-Notation
    { joint-iso-ccitt( 2) remote-operations( 4) notation( 0) }
-- Data Types --
universalFailure FROM CSTA-error-definition
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) error-definition( 120) }
CalledDeviceID, CallingDeviceID, SubjectDeviceID FROM CSTA-device-identifiers
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) device-identifiers( 123) }
ConnectionID FROM CSTA-call-connection-identifiers
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) call-connection-identifiers( 124) }
CorrelatorData, DeviceClass, DeviceProfile, SelectValue, PriorityValue, RoutingCrossRefID
    FROM CSTA-device-feature-types
            { iso( 1) identified-organization( 3) icd-ecma( 12)
                    standard( 0) csta2( 218) device-feature-types( 127) }
CSTACommonArguments FROM CSTA-extension-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) extension-types( 129) };

routeRequest OPERATION
    ARGUMENT    RouteRequestArgument
    ERRORS      {universalFailure}
::= 31

RouteRequestArgument ::=
    SEQUENCE
    {crossRefIdentifier     RoutingCrossRefID,
     currentRoute           CalledDeviceID,
     callingDevice          CallingDeviceID          OPTIONAL,
     routingDevice          SubjectDeviceID          OPTIONAL,
     routedCall             ConnectionID             OPTIONAL,
     routeSelAlgorithm      SelectValue              OPTIONAL,
     priority               PriorityValue            OPTIONAL,
     deviceProfile          DeviceProfile            OPTIONAL,
```

```
    correlatorData          CorrelatorData          OPTIONAL,
    extensions              CSTACommonArguments     OPTIONAL}

END  -- of CSTA-route-request
```

11.2 Reroute request

```
CSTA-re-route-request
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) re-route-request( 82) }
DEFINITIONS ::=
BEGIN
IMPORTS
OPERATION, ERROR FROM Remote-Operations-Notation
    { joint-iso-ccitt( 2) remote-operations( 4) notation( 0) }
-- Data Types --
RoutingCrossRefID FROM CSTA-device-feature-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) device-feature-types( 127) }
CSTACommonArguments FROM CSTA-extension-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) extension-types( 129) }
universalFailure FROM CSTA-error-definition
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) error-definition( 120) };

reRouteRequest OPERATION
    ARGUMENT    ReRouteRequestArgument
    ERRORS      {universalFailure}
::= 32

ReRouteRequestArgument ::=
    SEQUENCE
    {crossRefIdentifier     RoutingCrossRefID,
     extensions             CSTACommonArguments     OPTIONAL}

END  -- of CSTA-re-route-request
```

11.3 Route select request

```
CSTA-Route-select-request
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) route-select-request( 83) }
DEFINITIONS ::=
BEGIN
IMPORTS
OPERATION, ERROR FROM Remote-Operations-Notation
    { joint-iso-ccitt( 2) remote-operations( 4) notation( 0) }
-- Data Types --
CalledDeviceID FROM CSTA-device-identifiers
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) device-identifiers( 123) }
RouteUsedFlag, RetryValue, RoutingCrossRefID FROM CSTA-device-feature-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) device-feature-types( 127) }
CSTACommonArguments FROM CSTA-extension-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) extension-types( 129) }
CorrelatorData, DeviceProfile FROM CSTA-device-feature-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) device-feature-types( 127) }
universalFailure FROM CSTA-error-definition
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) error-definition( 120) };

routeSelectRequest OPERATION
    ARGUMENT    RouteSelectRequestArgument
    ERRORS      {universalFailure}
::= 33

RouteSelectRequestArgument ::=
    SEQUENCE
    {crossRefIdentifier     RoutingCrossRefID,
     routeSelected          CalledDeviceID,
     remainRetry            RetryValue                  OPTIONAL,
     deviceProfile          DeviceProfile               OPTIONAL,
     routeUsedReq           RouteUsedFlag               OPTIONAL,
     correlatorData         CorrelatorData              OPTIONAL,
     extensions             CSTACommonArguments         OPTIONAL}

END  -- of CSTA-route-select-request
```

11.4 Route used request

```
CSTA-route-used-request
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) route-used-request( 84) }
DEFINITIONS    ::=
BEGIN
IMPORTS
OPERATION, ERROR FROM Remote-Operations-Notation
            { joint-iso-ccitt( 2) remote-operations( 4) notation( 0) }
-- Data Types --
CallingDeviceID, CalledDeviceID FROM CSTA-device-identifiers
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) device-identifiers( 123) }
CorrelatorData, DomainValue, RoutingCrossRefID FROM CSTA-device-feature-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) device-feature-types( 127) }
CSTACommonArguments FROM CSTA-extension-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) extension-types( 129) }
universalFailure FROM CSTA-error-definition
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) error-definition( 120) };

routeUsedRequest  OPERATION
    ARGUMENT    RouteUsedRequestArgument
    ERRORS      {universalFailure}
::= 34

RouteUsedRequestArgument ::=
    SEQUENCE
    {crossRefIdentifier     RoutingCrossRefID,
     routeUsed              CalledDeviceID,
     callingDevice          CallingDeviceID          OPTIONAL,
     domain                 DomainValue              OPTIONAL,
     correlatorData         CorrelatorData           OPTIONAL,
     extensions             CSTACommonArguments      OPTIONAL}

END  -- of CSTA-route-used-request
```

11.5 Route end request

```
CSTA-route-end-request
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) route-end-request( 85) }
DEFINITIONS ::=
BEGIN
IMPORTS
OPERATION, ERROR FROM Remote-Operations-Notation
    { joint-iso-ccitt( 2) remote-operations( 4) notation( 0) }
-- Data Types --
RoutingCrossRefID  FROM CSTA-device-feature-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) device-feature-types( 127) }
CSTACommonArguments FROM CSTA-extension-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) extension-types( 129) }
universalFailure, UniversalFailure FROM CSTA-error-definition
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) error-definition( 120) };

routeEndRequest    OPERATION
    ARGUMENT    RouteEndRequestArgument
    ERRORS      {universalFailure}
::= 35

RouteEndRequestArgument ::=
    SEQUENCE
    {crossRefIdentifier     RoutingCrossRefID,
     errorValue             UniversalFailure            OPTIONAL,
     extensions             CSTACommonArguments         OPTIONAL}

END  -- of CSTA-route-end-request
```

12 Bidirectional services

This clause defines the protocol for the Bidirectional services of CSTA, using ASN.1. Text descriptions of the CSTA services are provided in Standard ECMA-217.

NOTE 5

The range of services supported on a particular association is specified in the application context at association time.

12.1 Escape

```
CSTA-escape-service
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) escape-service( 91) }
DEFINITIONS ::=
BEGIN
IMPORTS
OPERATION, ERROR FROM Remote-Operations-Notation
    { joint-iso-ccitt( 2) remote-operations( 4) notation( 0) }
-- Data Types --
universalFailure FROM CSTA-error-definition
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) error-definition( 120) }
CSTASecurityData FROM CSTA-security
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) security( 128) }
CSTACommonArguments, CSTAPrivateData FROM CSTA-extension-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) extension-types( 129) };

escapeService       OPERATION
    ARGUMENT        EscapeServiceArgument
    RESULT          EscapeServiceResult
    ERRORS          {universalFailure}
::=  51

EscapeServiceArgument ::=
    SEQUENCE
    {security       [0] IMPLICIT    CSTASecurityData                OPTIONAL,
     privateData    [1] IMPLICIT    SEQUENCE OF CSTAPrivateData}

EscapeServiceResult ::=
    CHOICE
    {extensions     CSTACommonArguments,
     noData         NULL}

END  -- of CSTA-escape-service
```

12.2 System status

```
CSTA-system-status
    { iso( 1) identified-organization( 3) icd-ecma( 12)
        standard( 0) csta2( 218) system-status( 92) }
DEFINITIONS ::=
BEGIN
IMPORTS
OPERATION, ERROR FROM Remote-Operations-Notation
    { joint-iso-ccitt( 2) remote-operations( 4) notation( 0) }
-- Data Types --
universalFailure FROM CSTA-error-definition
    { iso( 1) identified-organization( 3) icd-ecma( 12)
        standard( 0) csta2( 218) error-definition( 120) }
SystemStatus FROM CSTA-device-feature-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
        standard( 0) csta2( 218) device-feature-types( 127) }
CSTACommonArguments FROM CSTA-extension-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
        standard( 0) csta2( 218) extension-types( 129) };

systemStatus        OPERATION
    ARGUMENT        SystemStatusArgument
    RESULT          SystemStatusResult
    ERRORS          {universalFailure}
::= 52

SystemStatusArgument ::=
    CHOICE
    {systemStatus           SystemStatus,
     SEQUENCE
        {systemStatus       SystemStatus,
         extensions         CSTACommonArguments} }

SystemStatusResult ::=
    CHOICE
    {extensions     CSTACommonArguments,
     noData         NULL}

END  -- of CSTA-system-status
```

13 Status reporting services

This clause defines the protocol for the Status reporting services of CSTA, using ASN.1. Text descriptions of the CSTA services are provided in Standard ECMA-217.

NOTE 6

The range of services supported on a particular association is specified in the application context at association time.

13.1 Monitor start

```
CSTA-monitor-start
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) monitor-start( 101) }
DEFINITIONS ::=
BEGIN
IMPORTS
OPERATION, ERROR FROM Remote-Operations-Notation
    { joint-iso-ccitt( 2) remote-operations( 4) notation( 0) }
-- Data Types --
universalFailure FROM CSTA-error-definition
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) error-definition( 120) }
MonitorObject, MonitorFilter, MonitorType, MonitorCrossRefID FROM CSTA-status-reporting
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) status-reporting( 126) }
CSTACommonArguments FROM CSTA-extension-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) extension-types( 129) };

monitorStart        OPERATION
    ARGUMENT    MonitorStartArgument
    RESULT      MonitorStartResult
    ERRORS      {universalFailure}
::= 71

MonitorStartArgument ::=
    SEQUENCE
    {monitorObject          MonitorObject,
     monitorFilter          MonitorFilter               OPTIONAL,
     monitorType            MonitorType                 OPTIONAL,
     extensions             CSTACommonArguments         OPTIONAL}

MonitorStartResult ::=
    SEQUENCE
    {crossRefIdentifier     MonitorCrossRefID,
     monitorFilter          MonitorFilter               OPTIONAL,
     extensions             CSTACommonArguments         OPTIONAL}

END  -- of CSTA-monitor-start
```

13.2 Change monitor filter

```
CSTA-change-monitor-filter
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) change-monitor-filter( 102) }
DEFINITIONS ::=
BEGIN
IMPORTS
OPERATION, ERROR FROM Remote-Operations-Notation
    { joint-iso-ccitt( 2) remote-operations( 4) notation( 0) }
-- Data Types --
universalFailure FROM CSTA-error-definition
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) error-definition( 120) }
MonitorFilter, MonitorCrossRefID FROM CSTA-status-reporting
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) status-reporting( 126) }
CSTACommonArguments FROM CSTA-extension-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) extension-types( 129) };

changeMonitorFilter OPERATION
    ARGUMENT    ChangeMonitorFilterArgument
    RESULT      ChangeMonitorFilterResult
    ERRORS      {universalFailure}
::= 72

ChangeMonitorFilterArgument ::=
    SEQUENCE
    {monitorCrossRefID      MonitorCrossRefID,
     filterlist             MonitorFilter,
     extensions             CSTACommonArguments      OPTIONAL}

ChangeMonitorFilterResult ::=
    CHOICE
    {filterList             [0] IMPLICIT    MonitorFilter,
     SEQUENCE
            {filterList     MonitorFilter,
             extensions     CSTACommonArguments} }

END  -- of CSTA-change-monitor-filter
```

13.3 Monitor stop

```
CSTA-monitor-stop
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) monitor-stop( 103) }
DEFINITIONS ::=
BEGIN
IMPORTS
OPERATION, ERROR FROM Remote-Operations-Notation
    { joint-iso-ccitt( 2) remote-operations( 4) notation( 0) }
-- Data Types --
universalFailure FROM CSTA-error-definition
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) error-definition( 120) }
MonitorCrossRefID FROM CSTA-status-reporting
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) status-reporting( 126) }
CSTACommonArguments FROM CSTA-extension-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) extension-types( 129) };

monitorStopOPERATION
    ARGUMENT    MonitorStopArgument
    RESULT      MonitorStopResult
    ERRORS      {universalFailure}
::= 73

MonitorStopArgument ::=
    CHOICE
    {crossRefIdentifier     MonitorCrossRefID,
     SEQUENCE
            {crossRefIdentifier     MonitorCrossRefID,
             extensions             CSTACommonArguments} }

MonitorStopResult ::=
    CHOICE
    {extensions     CSTACommonArguments,
     noData         NULL}

END  -- of CSTA-monitor-stop
```

13.4 Snapshot device

```
CSTA-snapshot-device
    { iso( 1) identified-organization( 3) icd-ecma( 12)
        standard( 0) csta2( 218) snapshot-device( 104) }
DEFINITIONS ::=
BEGIN
IMPORTS
OPERATION, ERROR FROM Remote-Operations-Notation
    { joint-iso-ccitt( 2) remote-operations( 4) notation( 0) }
-- Data Types --
universalFailure FROM CSTA-error-definition
    { iso( 1) identified-organization( 3) icd-ecma( 12)
        standard( 0) csta2( 218) error-definition( 120) }
DeviceID FROM Device-identifiers
    { iso( 1) identified-organization( 3) icd-ecma( 12)
        standard( 0) csta2( 218) device-identifiers( 123) }
SnapshotDeviceData FROM CSTA-status-reporting
    { iso( 1) identified-organization( 3) icd-ecma( 12)
        standard( 0) csta2( 218) status-reporting( 126) }
CSTACommonArguments FROM CSTA-extension-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
        standard( 0) csta2( 218) extension-types( 129) };

snapshotDevice      OPERATION
    ARGUMENT        SnapshotDeviceArgument
    RESULT          SnapshotDeviceResult
    ERRORS          {universalFailure}
::= 74

SnapshotDeviceArgument ::=
    CHOICE
    {snapshotObject             DeviceID,
     SEQUENCE
        {snapshotObject         DeviceID,
         extensions             CSTACommonArguments} }

SnapshotDeviceResult ::=
    CHOICE
    {snapshotData               SnapshotDeviceData,
     SEQUENCE
        {snapshotData           SnapshotDeviceData,
         extensions             CSTACommonArguments} }

END  -- of CSTA-snapshot-device
```

13.5 Snapshot call

```
CSTA-snapshot-call
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) snapshot-call( 105) }
DEFINITIONS ::=
BEGIN
IMPORTS
OPERATION, ERROR FROM Remote-Operations-Notation
    { joint-iso-ccitt( 2) remote-operations( 4) notation( 0) }
-- Data Types --
universalFailure FROM CSTA-error-definition
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) error-definition( 120) }
ConnectionID FROM CSTA-call-connection-identifiers
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) call-connection-identifiers( 124) }
SnapshotCallData FROM CSTA-status-reporting
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) status-reporting( 126) }
CSTACommonArguments FROM CSTA-extension-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) extension-types( 129) };

snapshotCall        OPERATION
    ARGUMENT    SnapshotCallArgument
    RESULT      SnapshotCallResult
    ERRORS      {universalFailure}
::= 75

SnapshotCallArgument ::=
    CHOICE
    {snapshotObject             ConnectionID,
     SEQUENCE
            {snapshotObject     ConnectionID,
             extensions         CSTACommonArguments} }

SnapshotCallResult ::=
    CHOICE
    {snapshotData               SnapshotCallData,
     SEQUENCE
            {snapshotData       SnapshotCallData,
             extensions         CSTACommonArguments} }

END  -- of CSTA-snapshot-call
```

14 Input/output services

This clause defines the protocol for the input/output (I/O) services of CSTA, using ASN.1. Text descriptions of the CSTA services are provided in Standard ECMA-217.

14.1 Start data path

```
CSTA-start-data-path
    { iso( 1) identified-organization( 3) icd-ecma( 12)
        standard( 0) csta2( 218) start-data-path( 110) }
DEFINITIONS ::=
BEGIN
IMPORTS
OPERATION, ERROR FROM Remote-Operations-Notation
    { joint-iso-ccitt( 2) remote-operations( 4) notation( 0) }
-- Data Types --
universalFailure FROM CSTA-error-definition
    { iso( 1) identified-organization( 3) icd-ecma( 12)
        standard( 0) csta2( 218) error-definition( 120) }
CSTAObject FROM CSTA-switching-function-objects
    { iso( 1) identified-organization( 3) icd-ecma( 12)
        standard( 0) csta2( 218) switching-function-objects( 122) }
DataPathDirection, DataPathType, IOCrossRefID FROM CSTA-device-feature-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
        standard( 0) csta2( 218) device-feature-types( 127) }
CSTACommonArguments FROM CSTA-extension-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
        standard( 0) csta2( 218) extension-types( 129) };

startDataPath       OPERATION
    ARGUMENT        StartDataPathArgument
    RESULT          StartDataPathResult
    ERRORS          {universalFailure}
::= 110

StartDataPathArgument ::=
    SEQUENCE
    {device                         CSTAObject,
     dataPathDirection              [0] IMPLICIT DataPathDirection      OPTIONAL,
     dataPathType                   [1] IMPLICIT DataPathType           OPTIONAL,
     numberOfCharactersToCollect    [2] IMPLICIT INTEGER                OPTIONAL,
     terminationCharacter           IA5String                           OPTIONAL,
     timeout                        [3] IMPLICIT INTEGER                OPTIONAL,
     extensions                     CSTACommonArguments                 OPTIONAL}

StartDataPathResult ::=
    SEQUENCE
    {ioCrossRefID                   IOCrossRefID,
     numberOfCharactersToCollect    [0] IMPLICIT INTEGER                OPTIONAL,
     terminationCharacter           IA5String                           OPTIONAL,
     timeout                        [1] IMPLICIT INTEGER                OPTIONAL,
```

```
    extensions                  CSTACommonArguments          OPTIONAL}

END  -- of CSTA-start-data-path
```

14.2 Stop data path

```
CSTA-stop-data-path
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) stop-data-path( 111) }
DEFINITIONS ::=
BEGIN
IMPORTS
OPERATION, ERROR FROM Remote-Operations-Notation
    { joint-iso-ccitt( 2) remote-operations( 4) notation( 0) }
-- Data Types --
universalFailure FROM CSTA-error-definition
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) error-definition( 120) }
IOCrossRefID FROM CSTA-device-feature-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) device-feature-types( 127) }
CSTACommonArguments FROM CSTA-extension-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) extension-types( 129) };

stopDataPath        OPERATION
    ARGUMENT        StopDataPathArgument
    RESULT          StopDataPathResult
    ERRORS          {universalFailure}
::= 111

StopDataPathArgument ::=
    CHOICE
    {ioCrossRefID           IOCrossRefID,
     SEQUENCE
            {ioCrossRefID   IOCrossRefID,
             extensions     CSTACommonArguments} }

StopDataPathResult ::=
    CHOICE
    {extensions     CSTACommonArguments,
     noData         NULL}

END  -- of CSTA-stop-data-path
```

14.3 Send data

```
CSTA-send-data
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) send-data( 112) }
DEFINITIONS ::=
BEGIN
IMPORTS
OPERATION, ERROR FROM Remote-Operations-Notation
    { joint-iso-ccitt( 2) remote-operations( 4) notation( 0) }
-- Data Types --
universalFailure FROM CSTA-error-definition
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) error-definition( 120) }
EventCause FROM CSTA-event-causes
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) event-causes( 121) }
IOCrossRefID, IOData FROM CSTA-device-feature-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) device-feature-types( 127) }
CSTACommonArguments FROM CSTA-extension-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) extension-types( 129) };

sendData            OPERATION
    ARGUMENT        SendDataArgument
    RESULT          SendDataResult
    ERRORS          {universalFailure}
::= 112

SendDataArgument ::=
    SEQUENCE
    {ioCrossRefID           IOCrossRefID,
     data                   IOData,
     cause                  EventCause              OPTIONAL,
     extensions             CSTACommonArguments     OPTIONAL}

SendDataResult ::=
    CHOICE
    {extensions     CSTACommonArguments,
     noData         NULL}

END  -- of CSTA-send-data
```

14.4 Send multicast data

```
CSTA-send-multicast-data
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) send-multicast-data( 113) }
DEFINITIONS ::=
BEGIN
IMPORTS
OPERATION, ERROR FROM Remote-Operations-Notation
    { joint-iso-ccitt( 2) remote-operations( 4) notation( 0) }
-- Data Types --
universalFailure FROM CSTA-error-definition
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) error-definition( 120) }
IOCrossRefIDList, IOData FROM CSTA-device-feature-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) device-feature-types( 127) }
CSTACommonArguments FROM CSTA-extension-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) extension-types( 129) };

sendMulticastData   OPERATION
    ARGUMENT    SendMulticastDataArgument
    RESULT      SendMulticastDataResult
    ERRORS      {universalFailure}
::= 113

SendMulticastDataArgument ::=
    SEQUENCE
    {ioCrossRefIDList       IOCrossRefIDList,
     data                   IOData,
     extensions             CSTACommonArguments}

SendMulticastDataResult ::=
    CHOICE
    {extensions     CSTACommonArguments,
     noData         NULL}

END  -- of CSTA-send-multicast-data
```

14.5 Send broadcast data

```
CSTA-send-broadcast-data
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) send-broadcast-data( 114) }
DEFINITIONS ::=
BEGIN
IMPORTS
OPERATION, ERROR FROM Remote-Operations-Notation
    { joint-iso-ccitt( 2) remote-operations( 4) notation( 0) }
-- Data Types --
universalFailure FROM CSTA-error-definition
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) error-definition( 120) }
DataPathType, IOData FROM CSTA-device-feature-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) device-feature-types( 127) }
CSTACommonArguments FROM CSTA-extension-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) extension-types( 129) };

sendBroadcastData OPERATION
    ARGUMENT    SendBroadcastDataArgument
    RESULT      SendBroadcastDataResult
    ERRORS      {universalFailure}
::= 114

SendBroadcastDataArgument ::=
     SEQUENCE
    {data                   IOData,
     dataPathType           DataPathType                OPTIONAL,
     extensions             CSTACommonArguments         OPTIONAL}

SendBroadcastDataResult ::=
    CHOICE
    {extensions     CSTACommonArguments,
     noData         NULL}

END  -- of CSTA-send-broadcast-data
```

14.6 Suspend data path

```
CSTA-suspend-data-path
    { iso( 1) identified-organization( 3) icd-ecma( 12)
        standard( 0) csta2( 218) suspend-data-path( 115) }
DEFINITIONS ::=
BEGIN
IMPORTS
OPERATION, ERROR FROM Remote-Operations-Notation
    { joint-iso-ccitt( 2) remote-operations( 4) notation( 0) }
-- Data Types --
universalFailure FROM CSTA-error-definition
    { iso( 1) identified-organization( 3) icd-ecma( 12)
        standard( 0) csta2( 218) error-definition( 120) }
IOCrossRefID FROM CSTA-device-feature-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
        standard( 0) csta2( 218) device-feature-types( 127) }
CSTACommonArguments FROM CSTA-extension-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
        standard( 0) csta2( 218) extension-types( 129) };

suspendDataPath   OPERATION
    ARGUMENT      SuspendDataPathArgument
    RESULT        SuspendDataPathResult
    ERRORS        {universalFailure}
::= 115

SuspendDataPathArgument ::=
    CHOICE
    {ioCrossRefID                 IOCrossRefID,
    SEQUENCE
        {ioCrossRefID             IOCrossRefID,
         extensions               CSTACommonArguments} }

SuspendDataPathResult ::=
    CHOICE
    {extensions     CSTACommonArguments,
     noData         NULL }

END  -- of CSTA-suspend-data-path
```

14.7 Data path suspended

```
CSTA-data-path-suspended
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) data-path-suspended( 116) }
DEFINITIONS ::=
BEGIN
IMPORTS
OPERATION, ERROR FROM Remote-Operations-Notation
    { joint-iso-ccitt( 2) remote-operations( 4) notation( 0) }
-- Data Types --
universalFailure FROM CSTA-error-definition
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) error-definition( 120) }
IOCrossRefID FROM CSTA-device-feature-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) device-feature-types( 127) }
CSTACommonArguments FROM CSTA-extension-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) extension-types( 129) };

dataPathSuspended       OPERATION
    ARGUMENT            DataPathSuspendedArgument
    RESULT              DataPathSuspendedResult
    ERRORS              {universalFailure}
::= 116

DataPathSuspendedArgument ::=
    CHOICE
    {ioCrossRefID                   IOCrossRefID,
    SEQUENCE
            {ioCrossRefID           IOCrossRefID,
             extensions             CSTACommonArguments} }

DataPathSuspendedResult ::=
    CHOICE
    {extensions     CSTACommonArguments,
     noData         NULL }

END  -- of CSTA-data-path-suspended
```

14.8 Resume data path

```
CSTA-resume-data-path
    { iso( 1) identified-organization( 3) icd-ecma( 12)
        standard( 0) csta2( 218) resume-data-path( 117) }
DEFINITIONS ::=
BEGIN
IMPORTS
OPERATION, ERROR FROM Remote-Operations-Notation
    { joint-iso-ccitt( 2) remote-operations( 4) notation( 0) }
-- Data Types --
universalFailure FROM CSTA-error-definition
    { iso( 1) identified-organization( 3) icd-ecma( 12)
        standard( 0) csta2( 218) error-definition( 120) }
IOCrossRefID FROM CSTA-device-feature-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
        standard( 0) csta2( 218) device-feature-types( 127) }
CSTACommonArguments FROM CSTA-extension-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
        standard( 0) csta2( 218) extension-types( 129) };

resumeDataPath    OPERATION
    ARGUMENT      ResumeDataPathArgument
    RESULT        ResumeDataPathResult
    ERRORS        {universalFailure}
::= 117

ResumeDataPathArgument ::=
    CHOICE
    {ioCrossRefID                IOCrossRefID,
    SEQUENCE
        {ioCrossRefID            IOCrossRefID,
         extensions              CSTACommonArguments} }

ResumeDataPathResult ::=
    CHOICE
    {extensions    CSTACommonArguments,
     noData        NULL }

END  -- of CSTA-resume-data-path
```

14.9 Data path resumed

```
CSTA-data-path-resumed
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) data-path-resumed( 118) }
DEFINITIONS ::=
BEGIN
IMPORTS
OPERATION, ERROR FROM Remote-Operations-Notation
    { joint-iso-ccitt( 2) remote-operations( 4) notation( 0) }
-- Data Types --
universalFailure FROM CSTA-error-definition
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) error-definition( 120) }
IOCrossRefID FROM CSTA-device-feature-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) device-feature-types( 127) }
CSTACommonArguments FROM CSTA-extension-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) extension-types( 129) };

dataPathResumed  OPERATION
    ARGUMENT     DataPathResumedArgument
    RESULT       DataPathResumedResult
    ERRORS       {universalFailure}
::= 118

DataPathResumedArgument ::=
    CHOICE
    {ioCrossRefID                IOCrossRefID,
    SEQUENCE
            {ioCrossRefID        IOCrossRefID,
             extensions          CSTACommonArguments} }

DataPathResumedResult ::=
    CHOICE
    {extensions      CSTACommonArguments,
     noData          NULL }

END  -- of CSTA-data-path-resumed
```

14.10 Fast data

```
CSTA-fast-data
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) fast-data( 119) }
DEFINITIONS ::=
BEGIN
IMPORTS
OPERATION, ERROR FROM Remote-Operations-Notation
    { joint-iso-ccitt( 2) remote-operations( 4) notation( 0) }
-- Data Types --
universalFailure FROM CSTA-error-definition
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) error-definition( 120) }
CSTAObject FROM CSTA-switching-function-objects
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) switching-function-objects( 122) }
DataPathDirection, DataPathType, IOCrossRefID, IOData FROM CSTA-device-feature-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) device-feature-types( 127) }
CSTACommonArguments FROM CSTA-extension-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) extension-types( 129) };

fastData            OPERATION
    ARGUMENT        FastDataArgument
    RESULT          FastDataResult
    ERRORS          {universalFailure}
::= 119

FastDataArgument ::=
    SEQUENCE
    {device                 CSTAObject,
     data                   IOData,
     dataPathDirection      [0] IMPLICIT DataPathDirection      OPTIONAL,
     dataPathType           [1] IMPLICIT DataPathType           OPTIONAL,
     extensions             CSTACommonArguments                 OPTIONAL}

FastDataResult ::=
    CHOICE
    {extensions     CSTACommonArguments,
     noData         NULL}

END  -- of CSTA-fast-data
```

15 Voice unit services

15.1 Concatenate message

```
CSTA-concatenate-message
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) concatenate-message( 500) }
DEFINITIONS ::=
BEGIN
IMPORTS
OPERATION, ERROR FROM Remote-Operations-Notation
    { joint-iso-ccitt( 2) remote-operations( 4) notation( 0) }
-- Data Types --
universalFailure FROM CSTA-error-definition
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) error-definition( 120) }
MessageID, MessageIDList FROM CSTA-device-feature-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) device-feature-types( 127) }
CSTACommonArguments FROM CSTA-extension-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) extension-types( 129) };

concatenateMessage      OPERATION
    ARGUMENT            ConcatenateMessageArgument
    RESULT              ConcatenateMessageResult
    ERRORS              {universalFailure}
::= 500

ConcatenateMessageArgument ::=
    CHOICE
    {messagesToConcatenate      [0]IMPLICIT     MessageIDList,
     SEQUENCE
            {messagesToConcatenate      MessageIDList,
             extensions                 CSTACommonArguments} }

ConcatenateMessageResult ::=
    CHOICE
    {concatenatedMessage                MessageID,
     SEQUENCE
            {concatenatedMessage        MessageID,
             extensions                 CSTACommonArguments} }

END  -- of CSTA-concatenate-message
```

15.2 Delete message

```
CSTA-delete-message
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) delete-message( 501) }
DEFINITIONS ::=
BEGIN
IMPORTS
OPERATION, ERROR FROM Remote-Operations-Notation
    { joint-iso-ccitt( 2) remote-operations( 4) notation( 0) }
-- Data Types --
universalFailure FROM CSTA-error-definition
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) error-definition( 120) }
MessageID FROM CSTA-device-feature-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) device-feature-types( 127) }
CSTACommonArguments FROM CSTA-extension-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) extension-types( 129) };

deleteMessage       OPERATION
    ARGUMENT        DeleteMessageArgument
    RESULT          DeleteMessageResult
    ERRORS          {universalFailure}
::= 501

DeleteMessageArgument ::=
    CHOICE
    {messageToBeDeleted         MessageID,
     SEQUENCE
            {messageToBeDeleted  MessageID,
             extensions          CSTACommonArguments} }

DeleteMessageResult ::=
    CHOICE
    {extensions     CSTACommonArguments,
     noData         NULL}

END  -- of CSTA-delete-message
```

15.3 Play message

```
CSTA-play-message
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) play-message( 502) }
DEFINITIONS ::=
BEGIN
IMPORTS
OPERATION, ERROR FROM Remote-Operations-Notation
    { joint-iso-ccitt( 2) remote-operations( 4) notation( 0) }
-- Data Types --
universalFailure FROM CSTA-error-definition
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) error-definition( 120) }
ConnectionID FROM CSTA-call-connection-identifiers
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) call-connection-identifiers( 124) }
MessageID, TerminatingConditions FROM CSTA-device-feature-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) device-feature-types( 127) }
CSTACommonArguments FROM CSTA-extension-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) extension-types( 129) };

playMessage         OPERATION
    ARGUMENT        PlayMessageArgument
    RESULT          PlayMessageResult
    ERRORS          {universalFailure}
::= 502

PlayMessageArgument ::=
    SEQUENCE
    {messageToBePlayed   MessageID,
     overConnection      ConnectionID,
     duration            INTEGER                 OPTIONAL,
     termination         TerminatingConditions   OPTIONAL,
     extensions          CSTACommonArguments     OPTIONAL}

PlayMessageResult ::=
    CHOICE
    {extensions     CSTACommonArguments,
     noData         NULL}

END  -- of CSTA-play-message
```

15.4 Query voice attribute

```
CSTA-query-voice-attribute
    { iso( 1) identified-organization( 3) icd-ecma( 12)
        standard( 0) csta2( 218) query-voice-attribute( 503) }
DEFINITIONS ::=
BEGIN
IMPORTS
OPERATION, ERROR FROM Remote-Operations-Notation
    { joint-iso-ccitt( 2) remote-operations( 4) notation( 0) }
-- Data Types --
universalFailure FROM CSTA-error-definition
    { iso( 1) identified-organization( 3) icd-ecma( 12)
        standard( 0) csta2( 218) error-definition( 120) }
ConnectionID FROM CSTA-call-connection-identifiers
    { iso( 1) identified-organization( 3) icd-ecma( 12)
        standard( 0) csta2( 218) call-connection-identifiers( 124) }
AttributeInfo, MessageID FROM CSTA-device-feature-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
        standard( 0) csta2( 218) device-feature-types( 127) }
CSTACommonArguments FROM CSTA-extension-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
        standard( 0) csta2( 218) extension-types( 129) };

queryVoiceAttribute OPERATION
    ARGUMENT    QueryVoiceAttributeArgument
    RESULT      QueryVoiceAttributeResult
    ERRORS      {universalFailure}
::= 503

QueryVoiceAttributeArgument ::=
    SEQUENCE
    {messageToQuery     MessageID,
     attributeToQuery   ENUMERATED
                            {encodingAlgorithm  (0),
                             samplingRate       (1),
                             duration           (2),
                             fileName           (3),
                             currentPosition    (4),
                             currentSpeed       (5),
                             currentVolume      (6),
                             currentLevel       (7),
                             currentState       (8)     },
     connection         ConnectionID                    OPTIONAL,
     extensions         CSTACommonArguments             OPTIONAL}

QueryVoiceAttributeResult ::=
     SEQUENCE
    {attribute          AttributeInfo,
```

```
    extensions              CSTACommonArguments     OPTIONAL}

END  -- of CSTA-query-voice-attribute
```

15.5 Record message

```
CSTA-record-message
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) record-message( 511) }
DEFINITIONS ::=
BEGIN
IMPORTS
OPERATION, ERROR FROM Remote-Operations-Notation
    { joint-iso-ccitt( 2) remote-operations( 4) notation( 0) }
-- Data Types --
universalFailure FROM CSTA-error-definition
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) error-definition( 120) }
ConnectionID FROM CSTA-call-connection-identifiers
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) call-connection-identifiers( 124) }
MessageID, TerminatingConditions FROM CSTA-device-feature-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) device-feature-types( 127) }
CSTACommonArguments FROM CSTA-extension-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) extension-types( 129) };

recordMessage     OPERATION
    ARGUMENT      RecordMessageArgument
    RESULT        RecordMessageResult
    ERRORS        {universalFailure}
::= 511

RecordMessageArgument ::=
    SEQUENCE
    {callToBeRecorded     ConnectionID,
     samplingRate         REAL                                  OPTIONAL,
     encodingAlg          [0]IMPLICIT ENUMERATED
                                {aDPCM6K          (0),
                                 aDPCM8K          (1),
                                 muLawPCM6K       (2),
                                 aLawPCM6K        (3)    }     OPTIONAL,
     maxDuration          [1]IMPLICIT INTEGER                   OPTIONAL,
     termination          [2]IMPLICIT TerminatingConditions     OPTIONAL,
     extensions           CSTACommonArguments                   OPTIONAL}

RecordMessageResult ::=
    CHOICE
    {identityOfMessageRecorded          MessageID,
     SEQUENCE
            {identityOfMessageRecorded  MessageID,
             extensions                 CSTACommonArguments} }
```

END -- of CSTA-record-message

15.6 Reposition

```
CSTA-reposition
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) reposition( 504) }
DEFINITIONS ::=
BEGIN
IMPORTS
OPERATION, ERROR FROM Remote-Operations-Notation
    { joint-iso-ccitt( 2) remote-operations( 4) notation( 0) }
-- Data Types --
universalFailure FROM CSTA-error-definition
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) error-definition( 120) }
ConnectionID FROM CSTA-call-connection-identifiers
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) call-connection-identifiers( 124) }
MessageID, Period FROM CSTA-device-feature-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) device-feature-types( 127) }
CSTACommonArguments FROM CSTA-extension-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) extension-types( 129) };

reposition          OPERATION
    ARGUMENT        RepositionArgument
    RESULT          RepositionResult
    ERRORS          {universalFailure}
::= 504

RepositionArgument ::=
    SEQUENCE
    {connection             ConnectionID,
     periodOfReposition     Period,
     mesageToReposition     MessageID                OPTIONAL,
     extensions             CSTACommonArguments      OPTIONAL}

RepositionResult ::=
    CHOICE
    {extensions     CSTACommonArguments,
     noData         NULL}

END  -- of CSTA-reposition
```

15.7 Resume

```
CSTA-resume
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) resume( 505) }
DEFINITIONS ::=
BEGIN
IMPORTS
OPERATION, ERROR FROM Remote-Operations-Notation
    { joint-iso-ccitt( 2) remote-operations( 4) notation( 0) }
-- Data Types --
universalFailure FROM CSTA-error-definition
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) error-definition( 120) }
ConnectionID FROM CSTA-call-connection-identifiers
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) call-connection-identifiers( 124) }
MessageID FROM CSTA-device-feature-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) device-feature-types( 127) }
CSTACommonArguments FROM CSTA-extension-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) extension-types( 129) };

resume              OPERATION
    ARGUMENT        ResumeArgument
    RESULT          ResumeResult
    ERRORS          {universalFailure}
::= 505

ResumeArgument ::=
    SEQUENCE
    {connection             ConnectionID,
     mesageToResume         MessageID                   OPTIONAL,
     duration               INTEGER                     OPTIONAL,
     extensions             CSTACommonArguments         OPTIONAL}

ResumeResult ::=
    CHOICE
    {extensions     CSTACommonArguments,
     noData         NULL}

END  -- of CSTA-resume
```

15.8 Review

```
CSTA-review
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) review( 506) }
DEFINITIONS ::=
BEGIN
IMPORTS
OPERATION, ERROR FROM Remote-Operations-Notation
    { joint-iso-ccitt( 2) remote-operations( 4) notation( 0) }
-- Data Types --
universalFailure FROM CSTA-error-definition
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) error-definition( 120) }
ConnectionID FROM CSTA-call-connection-identifiers
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) call-connection-identifiers( 124) }
MessageID, Period FROM CSTA-device-feature-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) device-feature-types( 127) }
CSTACommonArguments FROM CSTA-extension-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) extension-types( 129) };

review              OPERATION
    ARGUMENT        ReviewArgument
    RESULT          ReviewResult
    ERRORS          {universalFailure}
::= 506

ReviewArgument ::=
    SEQUENCE
    {connection             ConnectionID,
     periodToReview         Period,
     mesageToReview         MessageID                OPTIONAL,
     extensions             CSTACommonArguments      OPTIONAL}

ReviewResult ::=
    CHOICE
    {extensions     CSTACommonArguments,
     noData         NULL}

END  -- of CSTA-review
```

15.9 Set voice attribute

```
CSTA-set-voice-attribute
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) set-voice-attribute( 507) }
DEFINITIONS ::=
BEGIN
IMPORTS
OPERATION, ERROR FROM Remote-Operations-Notation
    { joint-iso-ccitt( 2) remote-operations( 4) notation( 0) }
-- Data Types --
universalFailure FROM CSTA-error-definition
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) error-definition( 120) }
EventCause FROM CSTA-event-causes
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) event-causes( 121) }
ConnectionID FROM CSTA-call-connection-identifiers
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) call-connection-identifiers( 124) }
MessageID, Speed FROM CSTA-device-feature-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) device-feature-types( 127) }
CSTACommonArguments FROM CSTA-extension-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) extension-types( 129) };

setVoiceAttribute    OPERATION
    ARGUMENT    SetVoiveAttributeArgument
    RESULT      SetVoiveAttributeResult
    ERRORS      {universalFailure}
::= 507

SetVoiceAttributeArgument ::=
    SEQUENCE
    {connection             ConnectionID,
     attributeToSet         CHOICE
                                {speed          [0] IMPLICIT Speed,
                                 speakerVolume  [1] IMPLICIT INTEGER (0 .. 100),
                                 recordingLevel [2] IMPLICIT INTEGER (0 .. 100)      },
     message                MessageID                   OPTIONAL,
     extensions             CSTACommonArguments         OPTIONAL}

SetVoiceAttributeResult ::=
    CHOICE
    {extensions     CSTACommonArguments,
     noData         NULL}

END  -- of CSTA-set-voice-attribute
```

15.10 Stop

```
CSTA-stop
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) stop( 508) }
DEFINITIONS ::=
BEGIN
IMPORTS
OPERATION, ERROR FROM Remote-Operations-Notation
    { joint-iso-ccitt( 2) remote-operations( 4) notation( 0) }
-- Data Types --
universalFailure FROM CSTA-error-definition
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) error-definition( 120) }
ConnectionID FROM CSTA-call-connection-identifiers
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) call-connection-identifiers( 124) }
MessageID FROM CSTA-device-feature-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) device-feature-types( 127) }
CSTACommonArguments FROM CSTA-extension-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) extension-types( 129) };

stop                OPERATION
    ARGUMENT        StopArgument
    RESULT          StopResult
    ERRORS          {universalFailure}
::= 508

StopArgument ::=
    SEQUENCE
    {connection     ConnectionID,
     mesage         MessageID                   OPTIONAL,
     extensions     CSTACommonArguments         OPTIONAL}

StopResult ::=
    CHOICE
    {extensions     CSTACommonArguments,
     noData         NULL}

END  -- of CSTA-stop
```

15.11 Suspend

```
CSTA-suspend
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) suspend( 509) }
DEFINITIONS ::=
BEGIN
IMPORTS
OPERATION, ERROR FROM Remote-Operations-Notation
    { joint-iso-ccitt( 2) remote-operations( 4) notation( 0) }
-- Data Types --
universalFailure FROM CSTA-error-definition
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) error-definition( 120) }
ConnectionID FROM CSTA-call-connection-identifiers
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) call-connection-identifiers( 124) }
MessageID FROM CSTA-device-feature-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) device-feature-types( 127) }
CSTACommonArguments FROM CSTA-extension-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) extension-types( 129) };

suspend             OPERATION
    ARGUMENT        SuspendArgument
    RESULT          SuspendResult
    ERRORS          {universalFailure}
::= 509

SuspendArgument ::=
    SEQUENCE
    {connection     ConnectionID,
     mesage         MessageID                   OPTIONAL,
     extensions     CSTACommonArguments         OPTIONAL}

SuspendResult ::=
    CHOICE
    {extensions     CSTACommonArguments,
     noData         NULL}

END  -- of CSTA-suspend
```

15.12 Synthesize message

```
CSTA-synthesize-message
    { iso( 1) identified-organization( 3) icd-ecma( 12)
        standard( 0) csta2( 218) synthesize-message( 510) }
DEFINITIONS ::=
BEGIN
IMPORTS
OPERATION, ERROR FROM Remote-Operations-Notation
    { joint-iso-ccitt( 2) remote-operations( 4) notation( 0) }
-- Data Types --
universalFailure FROM CSTA-error-definition
    { iso( 1) identified-organization( 3) icd-ecma( 12)
        standard( 0) csta2( 218) error-definition( 120) }
ControlData, MessageID FROM CSTA-device-feature-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
        standard( 0) csta2( 218) device-feature-types( 127) }
CSTACommonArguments FROM CSTA-extension-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
        standard( 0) csta2( 218) extension-types( 129) };

synthesizeMessage OPERATION
    ARGUMENT    SynthesizeMessageArgument
    RESULT      SynthesizeMessageResult
    ERRORS      {universalFailure}
::= 510

SynthesizeMessageArgument ::=
    SEQUENCE
    {textToBeSynthesized        OCTET STRING,
     control                    ControlData             OPTIONAL,
     extensions                 CSTACommonArguments     OPTIONAL}

SynthesizeMessageResult ::=
    CHOICE
    {synthesizedMessage         MessageID,
     SEQUENCE
        {synthesizedMessage     MessageID,
         extensions             CSTACommonArguments} }

END  -- of CSTA-synthesize-message
```

16 Switching function errors

```
CSTA-error-definition
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) error-definition( 120) }
DEFINITIONS ::=
BEGIN

EXPORTS
UniversalFailure, universalFailure;

IMPORTS
ERROR FROM Remote-Operations-Notation
    { joint-iso-ccitt( 2) remote-operations( 4) notation( 0) }
CSTAPrivateData FROM CSTA-extension-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) extension-types( 129) };

universalFailure    ERROR
PARAMETER           UniversalFailure
::= 1

UniversalFailure ::= CHOICE
    {operationalErrors                  [1] IMPLICIT    Operations,
     stateErrors                        [2] IMPLICIT    StateIncompatibility,
     systemResourceErrors               [3] IMPLICIT    SystemResourceAvailability,
     subscribedResourceAvailabilityErrors [4] IMPLICIT  SubscribedResourceAvailability,
     performanceErrors                  [5] IMPLICIT    PerformanceManagement,
     securityErrors                     [6] IMPLICIT    SecurityError,
     unspecifiedErrors                  [7] IMPLICIT    NULL,
     nonStandardErrors                  [8] IMPLICIT    CSTAPrivateData}

Operations ::= ENUMERATED
    {       generic                             (1),
            requestIncompatibleWithObject       (2),
            valueOutOfRange                     (3),
            objectNotKnown                      (4),
            invalidCallingDevice                (5),
            invalidCalledDevice                 (6),
            invalidForwardingDestination        (7),
            privilegeViolationOnSpecifiedDevice (8),
            privilegeViolationOnCalledDevice    (9),
            privilegeViolationOnCallingDevice   (10),
            invalidCSTACallIdentifier           (11),
            invalidCSTADeviceIdentifier         (12),
            invalidCSTAConnectionIdentifier     (13),
            invalidDestination                  (14),
            invalidFeature                      (15),
            invalidAllocationState              (16),
```

```
                invalidCrossRefID                       (17),
                invalidObjectType                       (18),
                securityViolation                       (19),
                invalidCSTAApplicationCorrelator        (20),
                invalidAccountCode                      (21),
                invalidAuthorisationCode                (22),
                requestIncompatibleWithCallingDevice    (23),
                requestIncompatibleWithCalledDevice     (24),
                invalidMessageIdentifier                (25),
                messageIdentifierRequired               (26),
                mediaIncompatible                       (27)    }

StateIncompatibility ::= ENUMERATED
    {           generic                                 (1),
                invalidObjectState                      (2),
                invalidConnectionID                     (3),
                noActiveCall                            (4),
                noHeldCall                              (5),
                noCallToClear                           (6),
                noConnectionToClear                     (7),
                noCallToAnswer                          (8),
                noCallToComplete                        (9),
                notAbleToPlay                           (10),
                notAbleToResume                         (11),
                endOfMessage                            (12),
                beginingOfMessage                       (13),
                messageSuspended                        (14)    }

SystemResourceAvailability ::= ENUMERATED
    {           generic                                 (1),
                serviceBusy                             (2),
                resourceBusy                            (3),
                resourceOutOfService                    (4),
                networkBusy                             (5),
                networkOutOfService                     (6),
                overallMonitorLimitExceeded             (7),
                conferenceMemberLimitExceeded           (8)     }

SubscribedResourceAvailability ::= ENUMERATED
    {           generic                                 (1),
                objectMonitorLimitExceeded              (2),
                trunkLimitExceeded                      (3),
                outstandingRequestLimitExceeded         (4)     }

PerformanceManagement ::= ENUMERATED
    {           generic                                 (1),
                performanceLimitExceeded                (2)     }
```

```
SecurityError ::= ENUMERATED
    {       unspecified                 (0),
            sequenceNumberViolated      (1),
            timeStampViolated           (2),
            pACViolated                 (3),
            sealViolated                (4)     }

END  -- of CSTA-error-definition
```

17 Switching event cause values

```
CSTA-event-causes
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) event-causes( 121) }
DEFINITIONS ::=
BEGIN

EXPORTS
EventCause;

EventCause ::= ENUMERATED
-- a general list of cause codes
--                              Present in     Added in
--                              Version 1      Version2
{   activeMonitor               (1),
    alternate                   (2),
    blocked                                    (35),
    busy                        (3),
    callBack                    (4),
    callCancelled               (5),
    callForwardImmediate        (6),
    callForwardBusy             (7),
    callForwardNoAnswer         (8),
    callForward                 (9),
    callNotAnswered             (10),
    callPickup                  (11),
    campOn                      (12),
    characterCountReached                      (36),
    consultation                               (37),
    destNotObtainable           (13),
    distributed                                (38),
    doNotDisturb                (14),
    dTMFToneDetected                           (39),
    durationExceeded                           (40),
    endOfDataDetected                          (41),
    enteringDistribution                       (42),
    forcedPause                                (43),
    incompatibleDestination     (15),
    invalidAccountCode          (16),
    keyOperation                (17),
    lockout                     (18),
    maintenance                 (19),
    makeCall                                   (44),
    messageSizeExceeded                        (45),
    networkCongestion           (20),
    networkNotObtainable        (21),
    networkSignal                              (46),
    newCall                     (22),
```

```
    nextMessage                                 (47),
    noAvailableAgents           (23),
    normalClearing                              (48),
    noSpeechDetected                            (49),
    numberChanged                               (50),
    override                    (24),
    park                        (25),
    overflow                    (26),
    recall                      (27),
    redirected                  (28),
    reorderTone                 (29),
    resourcesNotAvailable       (30),
    silentMonitor               (31),
    singleStepConference                        (51),
    singleStepTransfer                          (52),
    speechDetected                              (53),
    switchTerminated                            (54),
    terminationCharacterReceived                (55),
    timeout                                     (56),
    transfer                    (32),
    trunksBusy                  (33),
    voiceUnitInitiator          (34) }

END  -- of event-cause-definitions
```

18 CSTA data types

The major parameters have been assigned distinct application tags to facilitate parsing. The data is defined in logical groups in ascending order of application tag. Application tags used are:

- APPLICATION 1 - 5 : Device identifiers
- APPLICATION 11 - 14: Connection identifiers and local connection states
- APPLICATION 21 - 24 : Status reporting
- APPLICATION 30 : CSTACommonArguments

18.1 Switching function objects

```
CSTA-switching-function-objects
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) switching-function-objects( 122) }
DEFINITIONS ::=
BEGIN

EXPORTS
CSTAObject;

IMPORTS
DeviceID FROM CSTA-device-identifiers
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) device-identifiers( 123) }
ConnectionID FROM  CSTA-call-connection-identifiers
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) call-connection-identifiers( 124) };

CSTAObject ::= CHOICE
    {device          DeviceID,
     call            ConnectionID}

END  -- of CSTA-switching-function-objects
```

18.2 Device identifiers

```
CSTA-device-identifiers
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) device-identifiers( 123) }
DEFINITIONS ::=
BEGIN

EXPORTS
DeviceID, NumberDigits, ExtendedDeviceID, CallingDeviceID, CalledDeviceID, SubjectDeviceID,
RedirectionDeviceID;

DeviceID ::= CHOICE
    {dialingNumber          [0] IMPLICIT    NumberDigits,
     deviceNumber           [1] IMPLICIT    DeviceNumber}

-- NumberDigits is a string of digits that represents a number (address) that
-- the switch can route a call on. It can be dialled by a user  (i.e from a telephone
-- keypad) to make a call.  DeviceNumber is not a routing address and cannot be dialed
-- from a keypad. It is allocated by the switch to reference a device.

NumberDigits        ::= IA5String

DeviceNumber        ::= INTEGER

ExtendedDeviceID ::= CHOICE
    {deviceIdentifier       DeviceID,
     implicitPublic         [2] IMPLICIT    NumberDigits,
     explicitPublic         [3]             PublicTON,
     implicitPrivate        [4] IMPLICIT    NumberDigits,
     explicitPrivate        [5]             PrivateTON,
     other                  [6] IMPLICIT    OtherPlan}

CallingDeviceID     ::= [APPLICATION 1] CHOICE
    {deviceIdentifier       ExtendedDeviceID,
     notKnown               [7] IMPLICIT    NULL,
     notRequired            [8] IMPLICIT    NULL}

CalledDeviceID ::= [APPLICATION 2] CHOICE
    {deviceIdentifier       ExtendedDeviceID,
     notKnown               [7] IMPLICIT    NULL,
     notRequired            [8] IMPLICIT    NULL}

SubjectDeviceID ::= [APPLICATION 3] CHOICE
    {deviceIdentifier       ExtendedDeviceID,
     notKnown               [7] IMPLICIT    NULL,
     notRequired            [8] IMPLICIT    NULL}
```

```
RedirectionDeviceID ::= [APPLICATION 4] CHOICE
    {numberdialed          ExtendedDeviceID,
     notKnown              [7] IMPLICIT    NULL,
     notRequired           [8] IMPLICIT    NULL}

-- SubjectDeviceID is used in some event reports to specify which device the report
-- refers to. If the SubjectDeviceID has had a monitor invoked then this data is not
-- required and so the implicit NULL encoding for notRequired is returned.
-- RedirectionDeviceID is used in Events as the lastRedirectionDevice.
-- CalledDeviceID is used in Events to specify the number dialled.

PublicTON ::= CHOICE
    {       unknown               [0] IMPLICIT    IA5String,
            international         [1] IMPLICIT    IA5String,
            national              [2] IMPLICIT    IA5String,
            networkspecific       [3] IMPLICIT    IA5String,
            subscriber            [4] IMPLICIT    IA5String,
            abbreviated           [5] IMPLICIT    IA5String       }

-- The public type of numbers  are derived from CCITT E.164

PrivateTON ::= CHOICE
    {       unknown               [0] IMPLICIT    IA5String,
            level3RegionalNumber  [1] IMPLICIT    IA5String,
            level2RegionalNumber  [2] IMPLICIT    IA5String,
            level1RegionalNumber  [3] IMPLICIT    IA5String,
            pTNSpecificNumber     [4] IMPLICIT    IA5String,
            localNumber           [5] IMPLICIT    IA5String,
            abbreviated           [6] IMPLICIT    IA5String       }

-- The private type of numbers are derived from ECMA-155

OtherPlan ::= OCTET STRING      -- Allows future expansion to cover other numbering
                                -- plans (such as X.121 etc.)

END  -- of CSTA-device-identifiers
```

18.3 Call and connection identifiers

```
CSTA-call-connection-identifiers
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) call-connection-identifiers( 124) }
DEFINITIONS ::=
BEGIN

EXPORTS
ConnectionID;

IMPORTS
DeviceID FROM CSTA-device-identifiers
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) device-identifiers( 123) };

ConnectionID ::= [APPLICATION 11] IMPLICIT SEQUENCE
    {call           [2] IMPLICIT    OCTET STRING        OPTIONAL,
     device         CHOICE
                    {staticID       DeviceID,
                     dynamicID      [3] IMPLICIT    OCTET STRING}       OPTIONAL}

END  -- of CSTA-call-connection-identifiers
```

18.4 Connection states

```
CSTA-connection-states
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) connection-states( 125) }
DEFINITIONS ::=
BEGIN

EXPORTS
ConnectionList, LocalConnectionState;

IMPORTS
ConnectionID FROM CSTA-call-connection-identifiers
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) call-connection-identifiers( 124) }
DeviceID FROM CSTA-device-identifiers
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) device-identifiers( 123) };

ConnectionList ::= SEQUENCE OF
    CHOICE
    {       newConnectionOnly           [0] ConnectionID,
            deviceIDOnly                [1] DeviceID,
            oldConnectionOnly           [2] ConnectionID,
            newConnectionAndDeviceID    [3] SEQUENCE
                                        {newConnectionConnectionID,
                                         deviceID           DeviceID    },
            newAndOldConnections        [4] SEQUENCE
                                        {newConnectionConnectionID,
                                         oldConnection      ConnectionID },
            deviceIDAndOldConnection    [5] SEQUENCE
                                        {deviceID           DeviceID,
                                         oldConnection      ConnectionID },
            allInformation              [6] SEQUENCE
                                        {newConnectionConnectionID,
                                         deviceID           DeviceID,
                                         oldConnection      ConnectionID }
    }

LocalConnectionState ::= [APPLICATION 14] IMPLICIT ENUMERATED
    {       null      (0),
            initiate  (1),
            alerting  (2),
            connect   (3),
            hold      (4),
            queued    (5),
            fail      (6)       }

END  -- CSTA-connection-states
```

18.5 Status reporting

```
CSTA-status-reporting
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) status-reporting( 126) }
DEFINITIONS ::=
BEGIN

EXPORTS
MonitorObject, MonitorCrossRefID, MonitorFilter, MonitorType, SnapshotCallData,
SnapshotDeviceData;

IMPORTS
CSTAObject FROM CSTA-switching-function-objects
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) switching-function-objects( 122) }
DeviceID, SubjectDeviceID FROM CSTA-device-identifiers
    { iso( 1) identified-organization( 3) icd-ecma( 012)
            standard( 0) csta2( 218) device-identifiers( 123) }
ConnectionID FROM CSTA-call-connection-identifiers
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) call-connection-identifiers( 124) }
LocalConnectionState FROM CSTA-connection-states
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) connection-states( 125) }
DeviceClass, DeviceProfile FROM CSTA-device-feature-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) device-feature-types( 127) };

MonitorObject ::= CSTAObject

MonitorCrossRefID ::= [APPLICATION 21] IMPLICIT OCTET STRING

MonitorFilter ::= SEQUENCE          -- default is no filter (i.e. all events)
    {       call            [0] IMPLICIT CallFilter             DEFAULT {},
            feature         [1] IMPLICIT FeatureFilter          DEFAULT {},
            agent           [2] IMPLICIT AgentFilter            DEFAULT {},
            maintenance     [3] IMPLICIT MaintenanceFilter      DEFAULT {},
            voiceUnit       [5] IMPLICIT VoiceUnitFilter        DEFAULT {},
            private         [4] IMPLICIT BOOLEAN DEFAULT FALSE          }

-- setting the relevant bit requests the filter for the appropriate events

CallFilter ::= BIT STRING
    {       callCleared             (0),
            conferenced             (1),
            connectionCleared       (2),
            delivered               (3),
            diverted                (4),
```

```
                established             (5),
                failed                  (6),
                held                    (7),
                networkReached          (8),
                originated              (9),
                queued                  (10),
                retrieved               (11),
                serviceInitiated        (12),
                transferred             (13)    }

FeatureFilter ::= BIT STRING
    {           callInformation         (0),
                doNotDisturb            (1),
                forwarding              (2),
                messageWaiting          (3),
                autoAnswer              (4),
                microphoneMute          (5),
                speakerMute             (6),
                speakerVolume           (7)     }

AgentFilter ::= BIT STRING
    {           loggedOn                (0),
                loggedOff               (1),
                notReady                (2),
                ready                   (3),
                workingAfterCall        (4),
                busy                    (5)     }

MaintenanceFilter ::= BIT STRING
    {           backInService           (0),
                outOfService            (1)     }

VoiceUnitFilter ::= BIT STRING
    {           stop                    (0),
                play                    (1),
                suspendPlay             (2),
                record                  (3),
                suspendRecord           (4),
                review                  (5),
                voiceAttributesChange   (6)     }

MonitorType ::= ENUMERATED
    {           call                    (0),
                device                  (1)     }

SnapshotDeviceData ::= [APPLICATION 22] IMPLICIT SEQUENCE OF
SnapshotDeviceResponseInfo
```

```
SnapshotDeviceResponseInfo ::= SEQUENCE
    {       callIdentifier          ConnectionID,
            localCallState          CallState,
            deviceProfile           DeviceProfile           OPTIONAL    }

SnapshotCallData ::= [APPLICATION 23] IMPLICIT SEQUENCE OF SnapshotCallResponseInfo

SnapshotCallResponseInfo ::= SEQUENCE
    {       deviceOnCall            SubjectDeviceID,
            callIdentifier          ConnectionID,
            callClass               DeviceProfile           OPTIONAL,
            localConnectionState    LocalConnectionState    OPTIONAL    }

CallState ::= CHOICE
    {       compound        [0] IMPLICIT CompoundCallState,
            simple          [1] IMPLICIT SimpleCallState,
            unknown         [2] IMPLICIT NULL                       }

-- unknown is returned by server if no other CallState can be supplied

CompoundCallState ::= SEQUENCE OF LocalConnectionState

SimpleCallState ::= ENUMERATED
    {       callNull                (0),    -- '00'H - null-null
            callPending             (1),    -- '01'H - null-initiate
            callOriginated          (3),    -- '03'H - null-connect
            callDelivered           (35),   -- '23'H - alerting-connect
            callDeliveredHeld       (36),   -- '24'H - alerting-held
            callReceived            (50),   -- '32'H - connect-alerting
            callEstablished         (51),   -- '33'H - connect-connect
            callEstablishedHeld     (52),   -- '34'H - connected-held
            callReceivedOnHold      (66),   -- '42'H - held-alerting
            callEstablishedOnHold   (67),   -- '43'H - held-connect
            callQueued              (83),   -- '53'H - queued-connect
            callQueuedHeld          (84),   -- '54'H - queued-held
            callFailed              (99),   -- '63'H - failed-connect
            callFailedHeld          (100)}  -- '64'H - failed-held

-- This represents the main call states in a simplified encoding.  The semantics
-- are identical to the sequence of connection states but they are represented by
-- an item from an enumerated list.

END  -- of CSTA-status-reporting
```

18.6 Device and feature types and other parameters

```
CSTA-device-feature-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) device-feature-types( 127) }
DEFINITIONS ::=
BEGIN

EXPORTS
AccountInfo, AgentID, AgentGroup, AgentPassword, AgentParameter, AgentState, AllocationState,
AuthCode, ConnectionDetails, CorrelatorData, DeviceClass, DeviceInfo, DeviceProfile , DeviceType,
DivertInfo, FeatureInfo, ListForwardParameters, LoggedOnInfo, LoggedOffInfo, ForwardParameter,
ForwardingType, MessageID, MessageIDList, NoOfCallsInQueue, NoOfCallsInFront, ParticipationType,
QueryDeviceFeature, QueryDeviceInformation, ReserveConnection, SetDeviceFeature, SystemStatus,
SelectValue, PriorityValue, SetupValues, RetryValue, RouteUsedFlag, DomainValue,
RoutingCrossRefID, IOCrossRefID, IOCrossRefIDList, IOData, Speed, DataPathDirection,
DataPathType, Period, TerminatingConditions, AttributeInfo, ControlData;

IMPORTS
ConnectionID FROM CSTA-call-connection-identifiers
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) call-connection-identifiers( 124) }
DeviceID, NumberDigits, CalledDeviceID FROM CSTA-device-identifiers
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) device-identifiers( 123) };

AccountInfo ::= OCTET STRING      -- used in CallInformation event

AgentID ::= OCTET STRING

AgentGroup ::= DeviceID

AgentPassword ::= OCTET STRING

AgentParameter ::= CHOICE
    {       agentLoggedIn           [0] IMPLICIT LoggedOnInfo,
            agentLoggedOut          [1] IMPLICIT LoggedOffInfo,
            agentNotReady           [2] IMPLICIT NULL,
            agentReady              [3] IMPLICIT NULL,
            agentBusy               [4] IMPLICIT NULL,
            agentWorkingAfterCall   [5] IMPLICIT NULL               }

AgentState ::= ENUMERATED
    {       agentNotReady               (0),
            agentNull                   (1),
            agentReady                  (2),
            agentBusy                   (3),
            agentWorkingAfterCall       (4)     }
```

```
AllocationState        ::= ENUMERATED
    {       callDelivered     (0),
            callEstablished   (1)       }

-- used in MakePredictiveCall to indicate when call should be allocated to the device

AuthCode ::= OCTET STRING        -- used in Information Events

ConnectionDetails ::= CHOICE
    {       heldCall                [0] IMPLICIT ConnectionID,
            activeCall                      [1] IMPLICIT ConnectionID,
            bothCalls                       [2] IMPLICIT SEQUENCE
                                                    {heldCall       ConnectionID,
                                                     activeCall     ConnectionID}   }

CorrelatorData ::= OCTET STRING

DeviceClass ::= BIT STRING
    {       voice   (0),
            data    (1),
            image   (2),
            audio   (4),
            other   (3)       }

DeviceInfo ::= SEQUENCE
    {       deviceID        DeviceID        OPTIONAL,
            deviceType      DeviceType,
            deviceClass     DeviceClass                     }

DeviceProfile ::= SEQUENCE
    {       deviceClass     DeviceClass,
            iSDNSetup       SetupValues     OPTIONAL        }

SetupValues ::= OCTET STRING

--  When SetupValues is required, it may include:
--          Bearer Capability
--          Calling Party Subaddress (to enable callers specific TE to be selected)
--          Called Party Subaddress
--          High Layer Compatibility
--          Low Layer Compatibility
--  Where all the above information elements are defined in ISO/IEC 11572: 1993

DeviceType ::= ENUMERATED
    {       station                 (0),
            line                    (1),
            button                  (2),
            aCD                     (3),
```

```
            trunk                   (4),
            operator                (5),
            other                   (6),
            conferenceBridge        (7),
            station-group           (16),
            line-group              (17),
            button-group            (18),
            aCD-group               (19),
            trunk-group             (20),
            operator-group          (21),
            parkingDevice           (22),
            other-group             (255)   }

DivertInfo ::= CHOICE       -- used by Divert Call service
    {       deflect                 [0] IMPLICIT SEQUENCE
                                    {callToBeDiverted       ConnectionID,
                                     newDestination         CalledDeviceID },
            pickup                  [1] IMPLICIT SEQUENCE
                                    {callToBePickedUp       ConnectionID,
                                     requestingDevice       DeviceID        },
            groupRequestingDevice   [2] DeviceID    }

FeatureInfo ::= CHOICE      -- used by Call Completion service
    {       campon                  [0] IMPLICIT    ConnectionID,
            callback                [1] IMPLICIT    ConnectionID,
            intrude                 [2] IMPLICIT    ConnectionID,
            callbackMessage         [3] IMPLICIT    ConnectionID    }

ListForwardParameters  ::= SEQUENCE OF SEQUENCE
    {       forwardingType          ForwardingType,
            forwardDN               NumberDigits    }

LoggedOnInfo ::= SEQUENCE
    {       agentID         [10] IMPLICIT   AgentID         OPTIONAL,
            password        [11] IMPLICIT   AgentPassword   OPTIONAL,
            group           AgentGroup                      OPTIONAL    }

LoggedOffInfo       ::= SEQUENCE
    {       agentID         [10] IMPLICIT   AgentID         OPTIONAL,
            password        [11] IMPLICIT   AgentPassword   OPTIONAL,
            group           AgentGroup                      OPTIONAL    }

ForwardParameter ::= SEQUENCE
    {       forwardingType          ForwardingType,
            forwardDN               NumberDigits            OPTIONAL    }

ForwardingType      ::= ENUMERATED
    {       forwardImmediateOn      (0),
            forwardImmediateOff     (1),
```

```
            forwardBusyOn           (2),
            forwardBusyOff          (3),
            forwardNoAnsOn          (4),
            forwardNoAnsOff         (5),
            forwardBusyIntOn        (6),
            forwardBusyIntOff       (7),
            forwardBusyExtOn        (8),
            forwardBusyExtOff       (9),
            forwardNoAnsIntOn       (10),
            forwardNoAnsIntOff      (11),
            forwardNoAnsExtOn       (12),
            forwardNoAnsExtOff      (13),
            forwardImmIntOn         (14),
            forwardImmIntOff        (15),
            forwardImmExtOn         (16),
            forwardImmExtOff        (17)    }

IOCrossRefID ::= CHOICE
    {       switchProvided          [0] OCTET STRING,
            computerProvided        [1] OCTET STRING    }

IOCrossRefIDList ::= SEQUENCE OF IOCrossRefID

IOData ::= OCTET STRING

MessageID ::= OCTET STRING

MessageIDList ::= SEQUENCE OF MessageID

NoOfCallsInQueue  ::= INTEGER           -- used in Call Queued Event

NoOfCallsInFront    ::= INTEGER         -- used in Call Queued Event

ParticipationType ::= ENUMERATED
    {       silent                  (0),
            active                  (1)     }

QueryDeviceFeature ::= ENUMERATED       -- used by Query Feature service request
    {       msgWaitingOn            (0),
            doNotDisturbOn          (1),
            forward                 (2),
            deviceInfo              (4),
            agentState              (5),
            routingEnabled          (6),
            autoAnswer              (7),
            microphoneMuteOn        (8),
            speakerMuteOn           (9),
            speakerVolume           (10)    }
```

```
QueryDeviceInformation ::= CHOICE
    {       msgWaitingOn            [0] IMPLICIT BOOLEAN,
            doNotDisturbOn          [1] IMPLICIT BOOLEAN,
            forward                 [2] IMPLICIT ListForwardParameters,
            deviceInfo              [4] IMPLICIT DeviceInfo,
            agentState              [5] IMPLICIT AgentState,
            routingEnabled          [6] IMPLICIT BOOLEAN,
            autoAnswerOn            [7] IMPLICIT BOOLEAN,
            microphoneMuteOn        [8] IMPLICIT BOOLEAN,
            speakerMuteOn           [9] IMPLICIT BOOLEAN,
            speakerVolume           [10] IMPLICIT INTEGER (0 .. 100)        }

ReserveConnection ::= BOOLEAN   -- used with Hold service to reserve ISDN connection

SetDeviceFeature ::= CHOICE     -- used by SetFeature service request
    {       msgWaitingOn            [0] IMPLICIT BOOLEAN,
            doNotDisturbOn          [1] IMPLICIT BOOLEAN,
            forward                 [2] IMPLICIT ForwardParameter,
            requestedAgentState     [3] AgentParameter,
            enableRouting           [4] IMPLICIT BOOLEAN,
            autoAnswerOn            [5] IMPLICIT BOOLEAN,
            microphoneMuteOn        [6] IMPLICIT BOOLEAN,
            speakerMuteOn           [7] IMPLICIT BOOLEAN,
            speakerVolume           [8] IMPLICIT INTEGER (0 .. 100)                 }

SystemStatus ::= ENUMERATED
    {       initializing            (0),
            enabled                 (1),
            normal                  (2),
            messagesLost            (3),
            disabled                (4),
            overloadImminent        (5),
            overloadReached         (6),
            overloadRelieved        (7)     }

SelectValue ::= ENUMERATED
    {       normal          (0),
            leastCost       (1),
            emergency       (2),
            aCD             (3),
            userDefined     (4)     }

PriorityValue ::= BOOLEAN           -- TRUE means priority call

RetryValue ::= CHOICE       -- used in RouteSelect Request service
    {       noListAvailable         [0] IMPLICIT BOOLEAN,
            noCountAvailable        [1] IMPLICIT BOOLEAN,
            retryCount              [2] IMPLICIT INTEGER            }
```

```
RouteUsedFlag ::= BOOLEAN       -- TRUE means RouteUsed Request service
                                -- requested

DomainValue ::= BOOLEAN -- TRUE means CSTA subdomain destination

RoutingCrossRefID ::= [APPLICATION 24] IMPLICIT OCTET STRING

Speed ::= INTEGER (1 .. 100)

DataPathDirection ::= ENUMERATED
    {       fromRequestor       (0),
            toRequestor         (1),
            biDirectional       (2)     }

DataPathType ::= ENUMERATED
    {       text                (0),
            digitalVoice        (1)     }

Period ::= CHOICE
    {       relativePosition [0] ENUMERATED
                    {       beginningOfMessage          (0),
                            endOfMessage                (1)     },
            position [1] INTEGER    }

TerminatingConditions ::= ENUMERATED
    {       durationExceeded    (0),
            dTMFDetected        (1),
            endOfDataDetected   (2),
            speechDetected      (3)     }

AttributeInfo ::= CHOICE
    {       encodingAlgorithm   [0] ENUMERATED
                                        {aDPCM6K            (0),
                                         aDPCM8K            (1),
                                         muLawPCM6K         (2),
                                         aLawPCM6K          (3)     },
            samplingRate        REAL,
            durations           [1] INTEGER,
            fileName            OCTET STRING,
            position            [2] INTEGER,
            speed               [3] Speed,
            volume              [4] INTEGER (1 .. 100),
            level               [5] INTEGER (1 .. 100),
```

```
            state                   [6] ENUMERATED
                                        {stop               (0),
                                         play               (1),
                                         record             (2),
                                         suspendPlay        (3),
                                         suspendRecord      (4),
                                         review             (5)     }
    }

ControlData ::= SEQUENCE
    {       sex             ENUMERATED
                                {male           (0),
                                 female         (1)     },
            language        OCTET STRING
    }

END  -- of CSTA-device-feature-types
```

18.7 Security service

```
CSTA-security
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) security( 128) }

DEFINITIONS ::=
BEGIN

EXPORTS
CSTASecurityData;

IMPORTS
PrivilegeAttributeCertificate, UTCTime FROM Security-Information
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) desd( 138) securityData( 1) };

Seal ::= SEQUENCE
    {       algorithmIdentifier     OBJECT IDENTIFIER,
            keyIdentifier           OCTET STRING,
            seal                    OCTET STRING        }

CSTASecurityData ::= SEQUENCE
    {       messageSequenceNumber       INTEGER                         OPTIONAL,
            timeStamp                   UTCTime                         OPTIONAL,
            privilegeAttributeCertificate   PrivilegeAttributeCertificate   OPTIONAL,
            seal                        Seal                            OPTIONAL    }

END  -- of CSTA-security
```

18.8 Common extensions

```
CSTA-extension-types
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) extension-types( 129) }
DEFINITIONS ::=
BEGIN

EXPORTS
CSTACommonArguments, CSTAPrivateData;

IMPORTS
CSTASecurityData FROM CSTA-security
    { iso( 1) identified-organization( 3) icd-ecma( 12)
            standard( 0) csta2( 218) security( 128) };

CSTACommonArguments ::= [APPLICATION 30] IMPLICIT SEQUENCE
    {       security        [0] IMPLICIT    CSTASecurityData                OPTIONAL,
            privateData     [1] IMPLICIT    SEQUENCE OF CSTAPrivateData     OPTIONAL }

CSTAPrivateData ::= NULL        -- The actual encoding of the private event is added here,
                                -- replacing NULL with another valid ASN.1 type.

END  -- of CSTA-extension-types
```

19 CSTA parameter size constraints

Parameter	Maximum length (in bytes)
call (part of ConnectionID)	8
MonitorCrossRefID	4
ApplicationCorrelator	32
PublicTON	32
PrivateTON	32
NumberDigits	32
DeviceNumber	4
dynamicID (part of ConnectionID)	32
AgentID	16
AgentPassword	16
AccountInfo	32
AuthCode	16
SetupValues	260
RoutingCrossRefID	4
Review Variable	4
Period	4
Speed	4
ROSE Invoke ID	4
Number Ahead In Queue	2
Number In Queue	2
IOCrossRefID	4
IOData	implementation-specific
MessageID	256

Annex A

(normative)

Protocol Implementation Conformance Statement (PICS) Proforma

Contents

.

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 Definitions and abbreviations

This Standard uses the following terms defined in ISO 9646-1:

- Protocol Implementation Conformance Statement (PICS);
- PICS Proforma.

In the "Reference" columns of the body of the PICS proforma, the letter S refers to the CSTA Phase II Services Standard ECMA-217 and the letter P refers to the CSTA Phase II Protocol Standard ECMA-218.

The following terms are used in the "Status" columns of the body of the PICS proforma:

m	mandatory; the capability is required for conformance to the protocol.
o	optional; the capability is not required for conformance to the protocol, or is required only within constraints described in dependencies ("if" statements). If the capability is implemented, it is required to conform to the protocol specifications.
o.<n>	optional, but support of at least one of the group of options labelled by the same numeral <n> is required.
C.<cid>	conditional; the requirement is conditional according to the condition identified by <cid>.
<item>	simple-predicate condition, dependent on the support marked for <item>.

A.3 Conformance

The supplier of a protocol implementation which is claimed to conform to ECMA-218 shall complete a copy of the Protocol Implementation Conformance Statement (PICS) proforma in clauses A.5 to A.13.

A.4 Instructions for completing the PICS proforma

The first part of the PICS proforma, the Implementation Identification (clause A.5), is to be completed as indicated with the information necessary to identify fully both the supplier and the implementation.

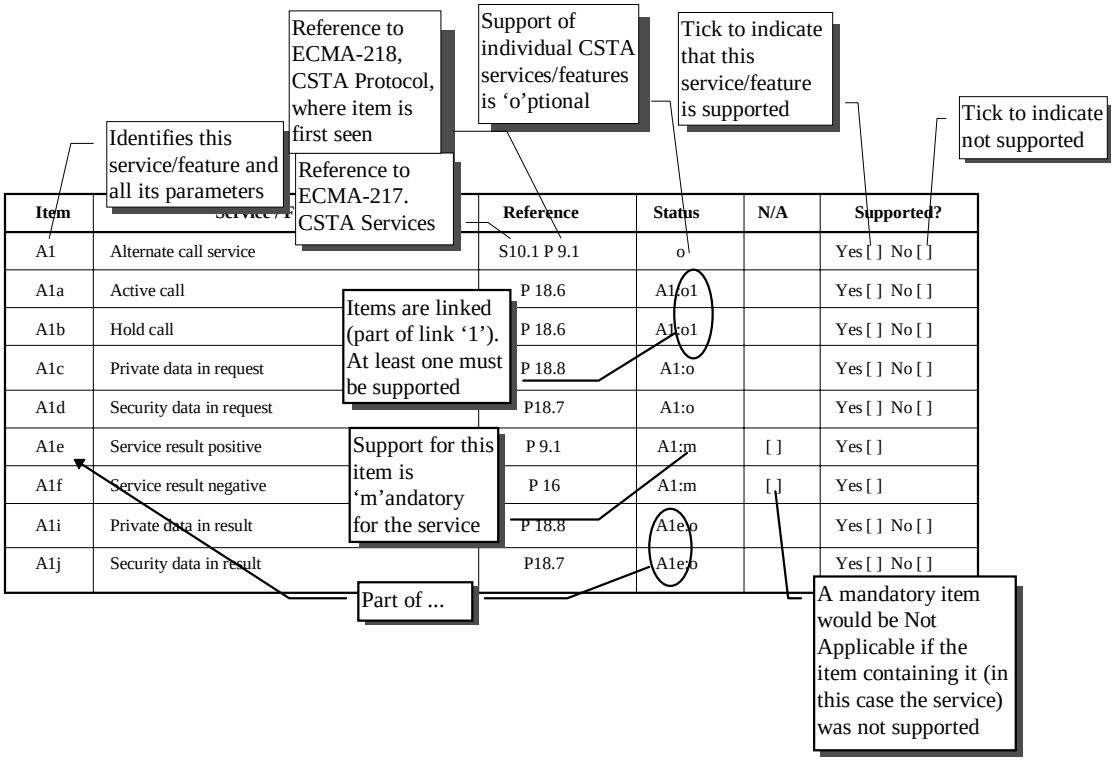
The main part of the PICS proforma (clauses A.6 to A.13) is a fixed format questionnaire divided into subclauses each containing a group of individual items. Answers to the questionnaire items are to be provided in the rightmost column, either by marking an answer to indicate a restricted choice (usually Yes or No), or by checking off all supported values (for parameters with a default).

Each item is identified by an item reference in the first column; the second column title indicates the nature of the table items which follow. The third column contains the references to material that specifies the item in the main body of ECMA-217 and ECMA-218. The remaining columns record the status of the item - whether support is mandatory, optional, or not applicable - and provide space for the answers.

Where a service is not supported, any parameters or dependent service components are not applicable. These dependencies are indicated in the status column using the item identifier as a key. It is not necessary to complete items in any subsidiary sections if a "No" response is given to the primary service component.

For supported services, a negative response to a mandatory subsidiary item indicates that the service does not conform to ECMA-218, and conformance cannot be claimed for that service.

Figure A.1 below is offered as an informative clarifying example of the first of the PICS proforma tables, that given in A.6.1 for the Alternate call service.



Item	Service / Parameter	Reference	Status	N/A	Supported?
A1	Alternate call service	S10.1 P 9.1	o		Yes [ ] No [ ]
A1a	Active call	P 18.6	A1:o1		Yes [ ] No [ ]
A1b	Hold call	P 18.6	A1:o1		Yes [ ] No [ ]
A1c	Private data in request	P 18.8	A1:o		Yes [ ] No [ ]
A1d	Security data in request	P18.7	A1:o		Yes [ ] No [ ]
A1e	Service result positive	P 9.1	A1:m	[ ]	Yes [ ]
A1f	Service result negative	P 16	A1:m	[ ]	Yes [ ]
A1i	Private data in result	P 18.8	A1e:o		Yes [ ] No [ ]
A1j	Security data in result	P18.7	A1e:o		Yes [ ] No [ ]

Figure A.1 - Clarifying diagram (informative)

A.5 Implementation identification

Supplier	
Protocol Version	First Edition
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.6 Switching function services

A.6.1 Alternate call

Item	Service / Feature	Reference	Status	N/A	Supported?
A1	Alternate call service	S10.1 P9.1	o		Yes [ ] No [ ]
A1a	Active call	P18.6	A1:o1		Yes [ ] No [ ]
A1b	Held call	P18.6	A1:o1		Yes [ ] No [ ]
A1c	Private data in request	P18.8	A1:o		Yes [ ] No [ ]
A1d	Security data in request	P18.7	A1:o		Yes [ ] No [ ]
A1e	Service result positive	P9.1	A1:m	[ ]	Yes [ ]
A1f	Service result negative	P16	A1:m	[ ]	Yes [ ]
A1g	Private data in result	P18.8	A1e:o		Yes [ ] No [ ]
A1h	Security data in result	P18.7	A1e:o		Yes [ ] No [ ]

A.6.2 Answer call

Item	Service / Feature	Reference	Status	N/A	Supported?
A2	Answer call service	S10.2 P9.2	o		Yes [ ] No [ ]
A2a	Call to be answered	P9.2	A2:m	[ ]	Yes [ ]
A2b	Private data in request	P18.8	A2:o		Yes [ ] No [ ]
A2c	Security data in request	P18.7	A2:o		Yes [ ] No [ ]
A2d	Service result positive	P9.2	A2:m	[ ]	Yes [ ]
A2e	Service result negative	P16	A2:m	[ ]	Yes [ ]
A2f	Private data in result	P18.8	A2d:o		Yes [ ] No [ ]
A2g	Security data in result	P18.7	A2d:o		Yes [ ] No [ ]

A.6.3 Associate data

Item	Service / Feature	Reference	Status	N/A	Supported?
A3	Associate data service	S10.3 P9.3	o		Yes [ ] No [ ]
A3a	Existing call	P9.3	A3:m	[ ]	Yes [ ]
A3b	Account code	P9.3	A3:o1		Yes [ ] No [ ]
A3c	Authorization code	P9.3	A3:o1		Yes [ ] No [ ]
A3d	Correlator data	P9.3	A3:o1		Yes [ ] No [ ]
A3e	Private data in request	P18.8	A3:o		Yes [ ] No [ ]
A3f	Security data in request	P18.7	A3:o		Yes [ ] No [ ]
A3g	Service result positive	P9.3	A3:m	[ ]	Yes [ ]
A3h	Service result negative	P16	A3:m	[ ]	Yes [ ]
A3i	Private data in result	P18.8	A3g:o		Yes [ ] No [ ]
A3j	Security data in result	P18.7	A3g:o		Yes [ ] No [ ]

A.6.4 Call completion

Item	Service / Feature	Reference	Status	N/A	Supported?
A4	Call completion service	S10.4 P9.4	o		Yes [ ] No [ ]
A4a	Camp on caller	P18.6	A4:o1		Yes [ ] No [ ]
A4b	Call back caller	P18.6	A4:o1		Yes [ ] No [ ]
A4c	Intrude caller	P18.6	A4:o1		Yes [ ] No [ ]
A4d	Call back message caller	P18.6	A4:o1		Yes [ ] No [ ]
A4e	Private data in request	P18.8	A4:o		Yes [ ] No [ ]
A4f	Security data in request	P18.7	A4:o		Yes [ ] No [ ]
A4g	Service result positive	P9.4	A4:m	[ ]	Yes [ ]
A4h	Service result negative	P16	A4:m	[ ]	Yes [ ]
A4i	Private data in result	P18.8	A4g:o		Yes [ ] No [ ]
A4j	Security data in result	P18.7	A4g:o		Yes [ ] No [ ]

A.6.5 Clear call

Item	Service / Feature	Reference	Status	N/A	Supported?
A5	Clear call service	S10.5 P9.5	o		Yes [ ] No [ ]
A5a	Call to be cleared	P9.5	A5:m	[ ]	Yes [ ]
A5b	Private data in request	P18.8	A5:o		Yes [ ] No [ ]
A5c	Security data in request	P18.7	A5:o		Yes [ ] No [ ]
A5d	Service result positive	P9.5	A5:m	[ ]	Yes [ ]
A5e	Service result negative	P16	A5:m	[ ]	Yes [ ]
A5f	Private data in result	P18.8	A5d:o		Yes [ ] No [ ]
A5g	Security data in result	P18.7	A5d:o		Yes [ ] No [ ]

A.6.6 Clear connection

Item	Service / Feature	Reference	Status	N/A	Supported?
A6	Clear connection service	S10.6 P9.6	o		Yes [ ] No [ ]
A6a	Connection to be cleared	P9.6	A6:m	[ ]	Yes [ ]
A6b	Private data in request	P18.8	A6:o		Yes [ ] No [ ]
A6c	Security data in request	P18.7	A6:o		Yes [ ] No [ ]
A6d	Service result positive	P9.6	A6:m	[ ]	Yes [ ]
A6e	Service result negative	P16	A6:m	[ ]	Yes [ ]
A6f	Private data in result	P18.8	A6d:o		Yes [ ] No [ ]
A6g	Security data in result	P18.7	A6d:o		Yes [ ] No [ ]

A.6.7 Conference call

Item	Service / Feature	Reference	Status	N/A	Supported?
A7	Conference call service	S10.7 P9.7	o		Yes [ ] No [ ]
A7a	Held call	P18.6	A7:o1		Yes [ ] No [ ]
A7b	Active call	P18.6	A7:o1		Yes [ ] No [ ]
A7c	Private data in request	P18.8	A7:o		Yes [ ] No [ ]
A7d	Security data in request	P18.7	A7:o		Yes [ ] No [ ]
A7e	Service result positive	P9.7	A7:m	[ ]	Yes [ ]
A7f	Service result negative	P16	A7:m	[ ]	Yes [ ]
A7g	Conference call	P9.7	A7e:m	[ ]	Yes [ ]
A7h	Connections	P18.4	A7e:o		Yes [ ] No [ ]
A7i	New connection	P18.4	A7h:o2		Yes [ ] No [ ]
A7j	Device ID	P18.4	A7h:o2		Yes [ ] No [ ]
A7k	Old connection	P18.4	A7h:o2		Yes [ ] No [ ]
A7l	Private data in result	P18.8	A7e:o		Yes [ ] No [ ]
A7m	Security data in result	P18.7	A7e:o		Yes [ ] No [ ]

A.6.8 Consultation call

Item	Service / Feature	Reference	Status	N/A	Supported?
A8	Consultation call service	S10.8 P9.8	o		Yes [ ] No [ ]
A8a	Existing call	P9.8	A8:m	[ ]	Yes [ ]
A8b	Consulted device	P9.8	A8:m	[ ]	Yes [ ]
A8c	Consulted device profile	P9.8	A8:o		Yes [ ] No [ ]
A8d	Device class	P18.6	A8c:m	[ ]	Yes [ ]
A8e	ISDN set up	P18.6	A8c:o		Yes [ ] No [ ]
A8f	Account code	P9.8	A8:o		Yes [ ] No [ ]
A8g	Authorization code	P9.8	A8:o		Yes [ ] No [ ]
A8h	Correlator data	P9.8	A8:o		Yes [ ] No [ ]
A8i	Private data in request	P18.8	A8:o		Yes [ ] No [ ]
A8j	Security data in request	P18.7	A8:o		Yes [ ] No [ ]
A8k	Service result positive	P9.8	A8:m	[ ]	Yes [ ]
A8l	Service result negative	P16	A8:m	[ ]	Yes [ ]
A8m	Initiated call	P9.8	A8k:m	[ ]	Yes [ ]
A8n	Private data in result	P18.8	A8k:o		Yes [ ] No [ ]
A8o	Security data in result	P18.7	A8k:o		Yes [ ] No [ ]

A.6.9 Divert call

Item	Service / Feature	Reference	Status	N/A	Supported?
A9	Divert call service	S10.9 P9.9	o		Yes [ ] No [ ]
A9a	Deflect	P18.6	A9:o1		Yes [ ] No [ ]
A9b	Pickup	P18.6	A9:o1		Yes [ ] No [ ]
A9c	Group pickup	P18.6	A9:o1		Yes [ ] No [ ]
A9d	Correlator data	P9.9	A9:o		Yes [ ] No [ ]
A9e	Private data in request	P18.8	A9:o		Yes [ ] No [ ]
A9f	Security data in request	P18.7	A9:o		Yes [ ] No [ ]
A9g	Service result positive	P9.9	A9:m	[ ]	Yes [ ]
A9h	Service result negative	P16	A9:m	[ ]	Yes [ ]
A9i	Private data in result	P18.8	A9g:o		Yes [ ] No [ ]
A9j	Security data in result	P18.7	A9g:o		Yes [ ] No [ ]

A.6.10 Hold call

Item	Service / Feature	Reference	Status	N/A	Supported?
A10	Hold call service	S10.10 P9.10	o		Yes [ ] No [ ]
A10a	Call to be held	P9.10	A10:m	[ ]	Yes [ ]
A10b	Connection reservation	P9.10	A10:o		Yes [ ] No [ ]
A10c	Private data in request	P18.8	A10:o		Yes [ ] No [ ]
A10d	Security data in request	P18.7	A10:o		Yes [ ] No [ ]
A10e	Service result positive	P9.10	A10:m	[ ]	Yes [ ]
A10f	Service result negative	P16	A10:m	[ ]	Yes [ ]
A10g	Private data in result	P18.8	A10e:o		Yes [ ] No [ ]
A10h	Security data in result	P18.7	A10e:o		Yes [ ] No [ ]

A.6.11 Make call

Item	Service / Feature	Reference	Status	N/A	Supported?
A11	Make call service	S10.11 P9.11	o		Yes [ ] No [ ]
A11a	Calling device	P9.11	A11:m	[ ]	Yes [ ]
A11b	Called device	P9.11	A11:m	[ ]	Yes [ ]
A11c	Device profile	P9.11	A11:o		Yes [ ] No [ ]
A11d	Device class	P18.6	A11c:m	[ ]	Yes [ ]
A11e	ISDN set up	P18.6	A11c:o		Yes [ ] No [ ]
A11f	Account code	P9.11	A11:o		Yes [ ] No [ ]
A11g	Authorization code	P9.11	A11:o		Yes [ ] No [ ]
A11h	Correlator data	P9.11	A11:o		Yes [ ] No [ ]
A11i	Private data in request	P18.8	A11:o		Yes [ ] No [ ]
A11j	Security data in request	P18.7	A11:o		Yes [ ] No [ ]
A11k	Service result positive	P9.11	A11:m	[ ]	Yes [ ]
A11l	Service result negative	P16	A11:m	[ ]	Yes [ ]
A11m	Initiated call	P9.11	A11k:m	[ ]	Yes [ ]
A11n	Private data in result	P18.8	A11k:o		Yes [ ] No [ ]
A11o	Security data in result	P18.7	A11k:o		Yes [ ] No [ ]

A.6.12 Make predictive call

Item	Service / Feature	Reference	Status	N/A	Supported?
A12	Make predictive call service	S10.12 P9.12	o		Yes [ ] No [ ]
A12a	Calling device	P9.12	A12:m	[ ]	Yes [ ]
A12b	Called device	P9.12	A12:m	[ ]	Yes [ ]
A12c	Allocation	P9.12	A12:o		Yes [ ] No [ ]
A12d	Device profile	P9.12	A12:o		Yes [ ] No [ ]
A12e	Device class	P18.6	A12d:m	[ ]	Yes [ ]
A12f	ISDN set up	P18.6	A12d:o		Yes [ ] No [ ]
A12g	Account code	P9.12	A12:o		Yes [ ] No [ ]
A12h	Authorization code	P9.12	A12:o		Yes [ ] No [ ]
A12i	Correlator data	P9.12	A12:o		Yes [ ] No [ ]
A12j	Private data in request	P18.8	A12:o		Yes [ ] No [ ]
A12k	Security data in request	P18.7	A12:o		Yes [ ] No [ ]
A12l	Service result positive	P9.12	A12:m	[ ]	Yes [ ]
A12m	Service result negative	P16	A12:m	[ ]	Yes [ ]
A12n	Initiated call	P9.12	A12l:m	[ ]	Yes [ ]
A12o	Private data in result	P18.8	A12l:o		Yes [ ] No [ ]
A12p	Security data in result	P18.7	A12l:o		Yes [ ] No [ ]

A.6.13 Park call

Item	Service / Feature	Reference	Status	N/A	Supported?
A13	Park call service	S10.13 P9.13	o		Yes [ ] No [ ]
A13a	Call to park	P9.13	A13:m	[ ]	Yes [ ]
A13b	Park to	P9.13	A13:m	[ ]	Yes [ ]
A13c	Park device profile	P9.13	A13:o		Yes [ ] No [ ]
A13d	Device class	P18.6	A13c:m	[ ]	Yes [ ]
A13e	ISDN set up	P18.6	A13c:o		Yes [ ] No [ ]
A13f	Correlator data	P9.13	A13:o		Yes [ ] No [ ]
A13g	Private data in request	P18.8	A13:o		Yes [ ] No [ ]
A13h	Security data in request	P18.7	A13:o		Yes [ ] No [ ]
A13i	Service result positive	P9.13	A13:m	[ ]	Yes [ ]
A13j	Service result negative	P16	A13:m	[ ]	Yes [ ]
A13k	Private data in result	P18.8	A13i:o		Yes [ ] No [ ]
A13l	Security data in result	P18.7	A13i:o		Yes [ ] No [ ]

A.6.14 Query device

Item	Service / Feature	Reference	Status	N/A	Supported?
A14	Query device service	S10.14 P9.14	o		Yes [ ] No [ ]
A14a	Device	P9.14	A14:m	[ ]	Yes [ ]
A14b	Feature	P9.14	A14:m	[ ]	Yes [ ]
A14c	Private data in request	P18.8	A14:o		Yes [ ] No [ ]
A14d	Security data in request	P18.7	A14:o		Yes [ ] No [ ]
A14e	Service result positive	P9.14	A14:m	[ ]	Yes [ ]
A14f	Service result negative	P16	A14:m	[ ]	Yes [ ]
A14g	Device information in service result	P9.14	A14e:m	[ ]	Yes [ ]
A14h	Message waiting on	P18.6	A14g:o1		Yes [ ] No [ ]
A14i	Do not disturb on	P18.6	A14g:o1		Yes [ ] No [ ]
A14j	Forward	P18.6	A14g:o1		Yes [ ] No [ ]
A14k	Forwarding type	P18.6	A14j:m	[ ]	Yes [ ]
A14l	Forward to device	P18.6	A14j:m	[ ]	Yes [ ]
A14m	Device information	P18.6	A14g:o1		Yes [ ] No [ ]
A14n	Device ID	P18.6	A14m:o		Yes [ ] No [ ]
A14o	Device type	P18.6	A14m:m	[ ]	Yes [ ]
A14p	Device class	P18.6	A14m:m	[ ]	Yes [ ]
A14q	Agent state	P18.6	A14g:o1		Yes [ ] No [ ]
A14r	Routing enabled	P18.6	A14g:o1		Yes [ ] No [ ]
A14s	Auto answer on	P18.6	A14g:o1		Yes [ ] No [ ]
A14t	Microphone mute on	P18.6	A14g:o1		Yes [ ] No [ ]
A14u	Speaker mute on	P18.6	A14g:o1		Yes [ ] No [ ]
A14v	Speaker volume	P18.6	A14g:o1		Yes [ ] No [ ]
A14w	Private data in result	P18.8	A14e:o		Yes [ ] No [ ]
A14x	Security data in result	P18.7	A14e:o		Yes [ ] No [ ]

A.6.15 Reconnect call

Item	Service / Feature	Reference	Status	N/A	Supported?
A15	Reconnect call service	S10.15 P9.15	o		Yes [ ] No [ ]
A15a	Held call	P18.6	A15:o1		Yes [ ] No [ ]
A15b	Active call	P18.6	A15:o1		Yes [ ] No [ ]
A15c	Private data in request	P18.8	A15:o		Yes [ ] No [ ]
A15d	Security data in request	P18.7	A15:o		Yes [ ] No [ ]
A15e	Service result positive	P9.15	A15:m	[ ]	Yes [ ]
A15f	Service result negative	P16	A15:m	[ ]	Yes [ ]
A15g	Private data in result	P18.8	A15e:o		Yes [ ] No [ ]
A15h	Security data in result	P18.7	A15e:o		Yes [ ] No [ ]

A.6.16 Retrieve call

Item	Service / Feature	Reference	Status	N/A	Supported?
A16	Retrieve call service	S10.16 P9.16	o		Yes [ ] No [ ]
A16a	Call to be retrieved	P9.16	A16:m	[ ]	Yes [ ]
A16b	Private data in request	P18.8	A16:o		Yes [ ] No [ ]
A16c	Security data in request	P18.7	A16:o		Yes [ ] No [ ]
A16d	Service result positive	P9.16	A16:m	[ ]	Yes [ ]
A16e	Service result negative	P16	A16:m	[ ]	Yes [ ]
A16f	Private data in result	P18.8	A16d:o		Yes [ ] No [ ]
A16g	Security data in result	P18.7	A16d:o		Yes [ ] No [ ]

A.6.17 Send DTMF tones

Item	Service / Feature	Reference	Status	N/A	Supported?
A17	Send DTMF tones service	S10.17 P9.17	o		Yes [ ] No [ ]
A17a	Connection to send	P9.17	A17:m	[ ]	Yes [ ]
A17b	Characters to send	P9.17	A17:m	[ ]	Yes [ ]
A17c	Tone duration	P9.17	A17:o		Yes [ ] No [ ]
A17d	Pause duration	P9.17	A17:o		Yes [ ] No [ ]
A17e	Private data in request	P18.8	A17:o		Yes [ ] No [ ]
A17f	Security data in request	P18.7	A17:o		Yes [ ] No [ ]
A17g	Service result positive	P9.17	A17:m	[ ]	Yes [ ]
A17h	Service result negative	P16	A17:m	[ ]	Yes [ ]
A17i	Private data in result	P18.8	A17g:o		Yes [ ] No [ ]
A17j	Security data in result	P18.7	A17g:o		Yes [ ] No [ ]

A.6.18 Set feature

Item	Service / Feature	Reference	Status	N/A	Supported?
A18	Set feature service	S10.18 P9.18	o		Yes [ ] No [ ]
A18a	Device	P9.18	A18:m	[ ]	Yes [ ]
A18b	Feature	P9.18	A18:m	[ ]	Yes [ ]
A18c	Message waiting on	P18.6	A18b:o1		Yes [ ] No [ ]
A18d	Do not disturb on	P18.6	A18b:o1		Yes [ ] No [ ]
A18e	Forward	P18.6	A18b:o1		Yes [ ] No [ ]
A18f	Forward immediate	P18.6	A18e:o2	[ ]	Yes [ ] No [ ]
A18g	Forward busy	P18.6	A18e:o2	[ ]	Yes [ ] No [ ]
A18h	Forward no answer	P18.6	A18e:o2	[ ]	Yes [ ] No [ ]
A18i	Forward busy internal	P18.6	A18e:o2	[ ]	Yes [ ] No [ ]
A18j	Forward busy external	P18.6	A18e:o2	[ ]	Yes [ ] No [ ]
A18k	Forward no answer internal	P18.6	A18e:o2	[ ]	Yes [ ] No [ ]
A18l	Forward no answer external	P18.6	A18e:o2	[ ]	Yes [ ] No [ ]
A18m	Forward immediate internal	P18.6	A18e:o2	[ ]	Yes [ ] No [ ]
A18n	Forward immediate external	P18.6	A18e:o2	[ ]	Yes [ ] No [ ]
A18o	Forward to device	P18.6	A18e:m	[ ]	Yes [ ]
A18p	Agent state	P18.6	A18b:o1		Yes [ ] No [ ]
A18q	Agent working after call	P18.6	A18p:o3	[ ]	Yes [ ] No [ ]
A18r	Agent ready	P18.6	A18p:o3	[ ]	Yes [ ] No [ ]
A18s	Agent logged on info	P18.6	A18p:o3	[ ]	Yes [ ] No [ ]
A18t	Agent logged off info	P18.6	A18p:o3	[ ]	Yes [ ] No [ ]
A18u	Agent not ready	P18.6	A18p:o3	[ ]	Yes [ ] No [ ]
A18v	Enable routing	P18.6	A18b:o1		Yes [ ] No [ ]
A18w	Auto answer on	P18.6	A18b:o1		Yes [ ] No [ ]
A18x	Microphone mute on	P18.6	A18b:o1		Yes [ ] No [ ]
A18y	Speaker mute on	P18.6	A18b:o1		Yes [ ] No [ ]
A18z	Speaker volume	P18.6	A18b:o1		Yes [ ] No [ ]
A18aa	Device profile	P9.18	A18:o		Yes [ ] No [ ]
A18ab	Device class	P18.6	A18aa:m	[ ]	Yes [ ]
A18ac	ISDN set up	P18.6	A18aa:o		Yes [ ] No [ ]
A18ad	Private data in request	P18.8	A18:o		Yes [ ] No [ ]
A18ae	Security data in request	P18.7	A18:o		Yes [ ] No [ ]
A18af	Service result positive	P9.18	A18:m	[ ]	Yes [ ]

A18ag	Service result negative	P16	A18:m	[ ]	Yes [ ]
A18ah	Private data in result	P18.8	A18af:o		Yes [ ] No [ ]
A18ai	Security data in result	P18.7	A18af:o		Yes [ ] No [ ]

A.6.19 Single step conference

Item	Service / Feature	Reference	Status	N/A	Supported?
A19	Single step conference service	S10.19 P9.19	o		Yes [ ] No [ ]
A19a	Active call	P9.19	A19:m	[ ]	Yes [ ]
A19b	Device to join	P9.19	A19:m	[ ]	Yes [ ]
A19c	Participation type	P9.19	A19:o		Yes [ ] No [ ]
A19d	Device profile	P9.19	A19:o		Yes [ ] No [ ]
A19e	Device class	P18.6	A19d:m	[ ]	Yes [ ]
A19f	ISDN set up	P18.6	A19d:o		Yes [ ] No [ ]
A19g	Account information	P9.19	A19:o		Yes [ ] No [ ]
A19h	Authorization code	P9.19	A19:o		Yes [ ] No [ ]
A19i	Correlator data	P9.19	A19:o		Yes [ ] No [ ]
A19j	Private data in request	P18.8	A19:o		Yes [ ] No [ ]
A19k	Security data in request	P18.7	A19:o		Yes [ ] No [ ]
A19l	Service result positive	P9.19	A19:m	[ ]	Yes [ ]
A19m	Service result negative	P16	A19:m	[ ]	Yes [ ]
A19n	Conferenced call	P9.19	A19l:m	[ ]	Yes [ ]
A19o	Private data in result	P18.8	A19l:o		Yes [ ] No [ ]
A19p	Security data in result	P18.7	A19l:o		Yes [ ] No [ ]

A.6.20 Single step transfer

Item	Service / Feature	Reference	Status	N/A	Supported?
A20	Single step transfer service	S10.20 P9.20	o		Yes [ ] No [ ]
A20a	Active call	P9.20	A20:m	[ ]	Yes [ ]
A20b	Device to transfer	P9.20	A20:m	[ ]	Yes [ ]
A20c	Device profile	P9.20	A20:o		Yes [ ] No [ ]
A20d	Device class	P18.6	A20c:m	[ ]	Yes [ ]
A20e	ISDN set up	P18.6	A20c:o		Yes [ ] No [ ]
A20f	Account information	P9.20	A20:o		Yes [ ] No [ ]
A20g	Authorization code	P9.20	A20:o		Yes [ ] No [ ]
A20h	Correlator data	P9.20	A20:o		Yes [ ] No [ ]
A20i	Private data in request	P18.8	A20:o		Yes [ ] No [ ]
A20j	Security data in request	P18.7	A20:o		Yes [ ] No [ ]
A20k	Service result positive	P9.20	A20:m	[ ]	Yes [ ]
A20l	Service result negative	P16	A20:m	[ ]	Yes [ ]
A20m	Transferred call	P9.20	A20k:o		Yes [ ] No [ ]
A20n	Connections	P18.4	A20k:o		Yes [ ] No [ ]
A20o	New connection	P18.4	A20n:o1		Yes [ ] No [ ]
A20p	Device ID	P18.4	A20n:o1		Yes [ ] No [ ]
A20q	Old connection	P18.4	A20n:o1		Yes [ ] No [ ]
A20r	Private data in result	P18.8	A20k:o		Yes [ ] No [ ]
A20s	Security data in result	P18.7	A20k:o		Yes [ ] No [ ]

A.6.21 Transfer call

Item	Service / Feature	Reference	Status	N/A	Supported?
A21	Transfer call service	S10.21 P9.21	o		Yes [ ] No [ ]
A21a	Transfer information	P9.21	A21:m	[ ]	Yes [ ]
A21b	Held call	P18.6	A21a:o1		Yes [ ] No [ ]
A21c	Active call	P18.6	A21a:o1		Yes [ ] No [ ]
A21d	Private data in request	P18.8	A21:o		Yes [ ] No [ ]
A21e	Security data in request	P18.7	A21:o		Yes [ ] No [ ]
A21f	Service result positive	P9.21	A21:m	[ ]	Yes [ ]
A21g	Service result negative	P16	A21:m	[ ]	Yes [ ]
A21h	Transferred call	P9.21	A21f:o		Yes [ ] No [ ]
A21i	Connections	P18.4	A21f:o		Yes [ ] No [ ]
A21j	New connection	P18.4	A21i:o2		Yes [ ] No [ ]
A21k	Device ID	P18.4	A21i:o2		Yes [ ] No [ ]
A21l	Old connection	P18.4	A21i:o2		Yes [ ] No [ ]
A21m	Private data in result	P18.8	A21f:o		Yes [ ] No [ ]
A21n	Security data in result	P18.7	A21f:o		Yes [ ] No [ ]

A.7 Switching function events

A.7.1 Event report service

Item	Service / Feature	Reference	Status	N/A	Supported?
B1	Event report service	S11.2 P10	o		Yes [ ] No [ ]
B1a	Monitor cross reference	P5.4 P10	B1:m	[ ]	Yes [ ]
B1b	Event specific information	P10	B1:m	[ ]	Yes [ ]
B1c	Private data	P18.8	B1:o		Yes [ ] No [ ]
B1d	Security data	P18.7	B1:o		Yes [ ] No [ ]

A.7.2 Call events

A.7.2.1 Call cleared

Item	Service / Feature	Reference	Status	N/A	Supported?
B2	Call cleared event	S11.2.3.1 P10.1.1	B1b:o		Yes [ ] No [ ]
B2a	Cleared call	P10.1.1	B2:m	[ ]	Yes [ ]
B2b	Local connection state	P10.1.1	B2:o		Yes [ ] No [ ]
B2c	Correlator data	P10.1.1	B2:o		Yes [ ] No [ ]
B2d	Cause	P10.1.1	B2:o		Yes [ ] No [ ]

A.7.2.2 Conferenced

Item	Service / Feature	Reference	Status	N/A	Supported?
B3	Conferenced event	S11.2.3.2 P10.1.2	B1b:o		Yes [ ] No [ ]
B3a	Primary old call	P10.1.2	B3:m	[ ]	Yes [ ]
B3b	Conference controller	P10.1.2	B3:m	[ ]	Yes [ ]
B3c	Added party	P10.1.2	B3:m	[ ]	Yes [ ]
B3d	Secondary old call	P10.1.2	B3:C	[ ]	Yes [ ] No [ ]
B3e	Connections	P10.1.2	B3:o		Yes [ ] No [ ]
B3f	New connection	P18.4	B3e:o1		Yes [ ] No [ ]
B3g	Device ID	P18.4	B3e:o1		Yes [ ] No [ ]
B3h	Old connection	P18.4	B3e:o1		Yes [ ] No [ ]
B3i	Local connection information	P10.1.2	B3:o		Yes [ ] No [ ]
B3j	Correlator data	P10.1.2	B3:o		Yes [ ] No [ ]
B3k	Cause	P10.1.2	B3:o		Yes [ ] No [ ]

C: If provided in previous events then mandatory

else optional

A.7.2.3 Connection cleared

Item	Service / Feature	Reference	Status	N/A	Supported?
B4	Connection cleared event	S11.2.3.3 P10.1.3	B1b:o		Yes [ ] No [ ]
B4a	Dropped connection	P10.1.3	B4:m	[ ]	Yes [ ]
B4b	Releasing device	P10.1.3	B4:m	[ ]	Yes [ ]
B4c	Local connection information	P10.1.3	B4:o		Yes [ ] No [ ]
B4d	Correlator data	P10.1.3	B4:o		Yes [ ] No [ ]
B4e	Cause	P10.1.3	B4:o		Yes [ ] No [ ]

A.7.2.4 Delivered

Item	Service / Feature	Reference	Status	N/A	Supported?
B5	Delivered event	S11.2.3.4 P10.1.4	B1b:o		Yes [ ] No [ ]
B5a	Connection	P10.1.4	B5:m	[ ]	Yes [ ]
B5b	Alerting device	P10.1.4	B5:m	[ ]	Yes [ ]
B5c	Calling device	P10.1.4	B5:m	[ ]	Yes [ ]
B5d	Called device	P10.1.4	B5:m	[ ]	Yes [ ]
B5e	Last redirection device	P10.1.4	B5:m	[ ]	Yes [ ]
B5f	Originating connection	P10.1.4	B5:o		Yes [ ] No [ ]
B5g	Local connection information	P10.1.4	B5:o		Yes [ ] No [ ]
B5h	Correlator data	P10.1.4	B5:o		Yes [ ] No [ ]
B5i	Cause	P10.1.4	B5:o		Yes [ ] No [ ]

A.7.2.5 Diverted

Item	Service / Feature	Reference	Status	N/A	Supported?
B6	Diverted event	S11.2.3.5 P10.1.5	B1b:o		Yes [ ] No [ ]
B6a	Diverted connection	P10.1.5	B6:C	[ ]	Yes [ ] No [ ]
B6b	Diverting device	P10.1.5	B6:m	[ ]	Yes [ ]
B6c	New destination	P10.1.5	B6:m	[ ]	Yes [ ]
B6d	Local connection information	P10.1.5	B6:o		Yes [ ] No [ ]
B6e	Correlator data	P10.1.5	B6:o		Yes [ ] No [ ]
B6f	Cause	P10.1.5	B6:o		Yes [ ] No [ ]

C: If the call alerted the device then mandatory

else optional

A.7.2.6 Established

Item	Service / Feature	Reference	Status	N/A	Supported?
B7	Established event	S11.2.3.6 P10.1.6	B1b:o		Yes [ ] No [ ]
B7a	Established connection	P10.1.6	B7:m	[ ]	Yes [ ]
B7b	Answering device	P10.1.6	B7:m	[ ]	Yes [ ]
B7c	Calling device	P10.1.6	B7:m	[ ]	Yes [ ]
B7d	Called device	P10.1.6	B7:m	[ ]	Yes [ ]
B7e	Last redirection device	P10.1.6	B7:m	[ ]	Yes [ ]
B7f	Originating connection	P10.1.6	B7:o		Yes [ ] No [ ]
B7g	Local connection information	P10.1.6	B7:o		Yes [ ] No [ ]
B7h	Correlator data	P10.1.6	B7:o		Yes [ ] No [ ]
B7i	Cause	P10.1.6	B7:o		Yes [ ] No [ ]

A.7.2.7 Failed

Item	Service / Feature	Reference	Status	N/A	Supported?
B8	Failed event	S11.2.3.7 P10.1.7	B1b:o		Yes [ ] No [ ]
B8a	Failed connection	P10.1.7	B8:m	[ ]	Yes [ ]
B8b	Failed device	P10.1.7	B8:m	[ ]	Yes [ ]
B8c	Called device	P10.1.7	B8:m	[ ]	Yes [ ]
B8d	Local connection information	P10.1.7	B8:o		Yes [ ] No [ ]
B8e	Correlator data	P10.1.7	B8:o		Yes [ ] No [ ]
B8f	Cause	P10.1.7	B8:o		Yes [ ] No [ ]

A.7.2.8 Held

Item	Service / Feature	Reference	Status	N/A	Supported?
B9	Held event	S11.2.3.8 P10.1.8	B1b:o		Yes [ ] No [ ]
B9a	Held connection	P10.1.8	B9:m	[ ]	Yes [ ]
B9b	Holding device	P10.1.8	B9:m	[ ]	Yes [ ]
B9c	Local connection information	P10.1.8	B9:o		Yes [ ] No [ ]
B9d	Correlator data	P10.1.8	B9:o		Yes [ ] No [ ]
B9e	Cause	P10.1.8	B9:o		Yes [ ] No [ ]

A.7.2.9 Network reached

Item	Service / Feature	Reference	Status	N/A	Supported?
B10	Network reached event	S11.2.3.9 P10.1.9	B1b:o		Yes [ ] No [ ]
B10a	Connection	P10.1.9	B10:m	[ ]	Yes [ ]
B10b	Trunk used	P10.1.9	B10:m	[ ]	Yes [ ]
B10c	Called device	P10.1.9	B10:m	[ ]	Yes [ ]
B10d	Local connection information	P10.1.9	B10:o		Yes [ ] No [ ]
B10e	Correlator data	P10.1.9	B10:o		Yes [ ] No [ ]
B10f	Cause	P10.1.9	B10:o		Yes [ ] No [ ]

A.7.2.10 Originated

Item	Service / Feature	Reference	Status	N/A	Supported?
B11	Originated event	S11.2.3.10 P10.1.10	B1b:o		Yes [ ] No [ ]
B11a	Originated connection	P10.1.10	B11:m	[ ]	Yes [ ]
B11b	Calling device	P10.1.10	B11:m	[ ]	Yes [ ]
B11c	Called device	P10.1.10	B11:m	[ ]	Yes [ ]
B11d	Originating device	P10.1.10	B11:o		Yes [ ] No [ ]
B11e	Local connection information	P10.1.10	B11:o		Yes [ ] No [ ]
B11f	Correlator data	P10.1.10	B11:o		Yes [ ] No [ ]
B11g	Cause	P10.1.10	B11:o		Yes [ ] No [ ]

A.7.2.11 Queued

Item	Service / Feature	Reference	Status	N/A	Supported?
B12	Queued event	S11.2.3.11 P10.1.11	B1b:o		Yes [ ] No [ ]
B12a	Queued connection	P10.1.11	B12:m	[ ]	Yes [ ]
B12b	Queue	P10.1.11	B12:m	[ ]	Yes [ ]
B12c	Calling device	P10.1.11	B12:m	[ ]	Yes [ ]
B12d	Called device	P10.1.11	B12:m	[ ]	Yes [ ]
B12e	Last redirection device	P10.1.11	B12:m	[ ]	Yes [ ]
B12f	Number of calls in queue	P10.1.11	B12:o		Yes [ ] No [ ]
B12g	Calls in front	P10.1.11	B12:o		Yes [ ] No [ ]
B12h	Local connection information	P10.1.11	B12:o		Yes [ ] No [ ]
B12i	Correlator data	P10.1.11	B12:o		Yes [ ] No [ ]
B12j	Cause	P10.1.11	B12:o		Yes [ ] No [ ]

A.7.2.12 Retrieved

Item	Service / Feature	Reference	Status	N/A	Supported?
B13	Retrieved event	S11.2.3.12 P10.1.12	B1b:o		Yes [ ] No [ ]
B13a	Retrieved connection	P10.1.12	B13:m	[ ]	Yes [ ]
B13b	Retrieving device	P10.1.12	B13:m	[ ]	Yes [ ]
B13c	Local connection information	P10.1.12	B13:o		Yes [ ] No [ ]
B13d	Correlator data	P10.1.12	B13:o		Yes [ ] No [ ]
B13e	Cause	P10.1.12	B13:o		Yes [ ] No [ ]

A.7.2.13 Service initiated

Item	Service / Feature	Reference	Status	N/A	Supported?
B14	Service initiated event	S11.2.3.13 P10.1.13	B1b:o		Yes [ ] No [ ]
B14a	Initiated connection	P10.1.13	B14:m	[ ]	Yes [ ]
B14b	Local connection information	P10.1.13	B14:o		Yes [ ] No [ ]
B14c	Cause	P10.1.13	B14:o		Yes [ ] No [ ]

A.7.2.14 Transferred

Item	Service / Feature	Reference	Status	N/A	Supported?
B15	Transferred event	S11.2.3.14 P10.1.14	B1b:o		Yes [ ] No [ ]
B15a	Primary old call	P10.1.14	B15:m	[ ]	Yes [ ]
B15b	Secondary old call	P10.1.14	B15:C	[ ]	Yes [ ] No [ ]
B15c	Transferring device	P10.1.14	B15:m	[ ]	Yes [ ]
B15d	Transferred-to device	P10.1.14	B15:m	[ ]	Yes [ ]
B15e	Connections	P10.1.14	B15:o		Yes [ ] No [ ]
B15f	New connection	P18.4	B15e:o1		Yes [ ] No [ ]
B15g	Device ID	P18.4	B15e:o1		Yes [ ] No [ ]
B15h	Old connection	P18.4	B15e:o1		Yes [ ] No [ ]
B15i	Local connection information	P10.1.14	B15:o		Yes [ ] No [ ]
B15j	Correlator data	P10.1.14	B15:o		Yes [ ] No [ ]
B15k	Cause	P10.1.14	B15:o		Yes [ ] No [ ]

C: If parameter in previous events then mandatory

else optional

A.7.3 Agent events

A.7.3.1 Agent logged on

Item	Service / Feature	Reference	Status	N/A	Supported?
B16	Agent logged on event	S11.2.2.2 P10.3.2	B1b:o		Yes [ ] No [ ]
B16a	Agent device	P10.3.2	B16:m	[ ]	Yes [ ]
B16b	Agent ID	P10.3.2	B16:o		Yes [ ] No [ ]
B16c	Agent group	P10.3.2	B16:o		Yes [ ] No [ ]
B16d	Password	P10.3.2	B16:o		Yes [ ] No [ ]
B16e	Cause	P10.3.2	B16:o		Yes [ ] No [ ]

A.7.3.2 Agent logged off

Item	Service / Feature	Reference	Status	N/A	Supported?
B17	Agent logged off event	S11.2.2.1 P10.3.3	B1b:o		Yes [ ] No [ ]
B17a	Agent device	P10.3.3	B17:m	[ ]	Yes [ ]
B17b	Agent ID	P10.3.3	B17:o		Yes [ ] No [ ]
B17c	Password	P10.3.3	B17:o		Yes [ ] No [ ]
B17d	Agent group	P10.3.3	B17:o		Yes [ ] No [ ]
B17e	Cause	P10.3.3	B17:o		Yes [ ] No [ ]

A.7.3.3 Agent ready

Item	Service / Feature	Reference	Status	N/A	Supported?
B18	Agent ready event	S11.2.2.4 P10.3.5	B1b:o		Yes [ ] No [ ]
B18a	Agent device	P10.3.5	B18:m	[ ]	Yes [ ]
B18b	Agent ID	P10.3.5	B18:o		Yes [ ] No [ ]
B18c	Cause	P10.3.5	B18:o		Yes [ ] No [ ]

A.7.3.4 Agent not ready

Item	Service / Feature	Reference	Status	N/A	Supported?
B19	Agent not ready event	S11.2.2.3 P10.3.4	B1b:o		Yes [ ] No [ ]
B19a	Agent device	P10.3.4	B19:m	[ ]	Yes [ ]
B19b	Agent ID	P10.3.4	B19:o		Yes [ ] No [ ]
B19c	Cause	P10.3.4	B19:o		Yes [ ] No [ ]

A.7.3.5 Agent working after call

Item	Service / Feature	Reference	Status	N/A	Supported?
B20	Agent working after call event	S11.2.2.5 P10.3.6	B1b:o		Yes [ ] No [ ]
B20a	Agent device	P10.3.6	B20:m	[ ]	Yes [ ]
B20b	Agent ID	P10.3.6	B20:o		Yes [ ] No [ ]
B20c	Agent group	P10.3.6	B20:o		Yes [ ] No [ ]
B20d	Cause	P10.3.6	B20:o		Yes [ ] No [ ]

A.7.3.6 Agent busy

Item	Service / Feature	Reference	Status	N/A	Supported?
B21	Agent busy event	S11.2.2.6 P10.3.1	B1b:o		Yes [ ] No [ ]
B21a	Agent device	P10.3.1	B21:m	[ ]	Yes [ ]
B21b	Agent ID	P10.3.1	B21:o		Yes [ ] No [ ]
B21c	Agent group	P10.3.1	B21:o		Yes [ ] No [ ]
B21d	Cause	P10.3.1	B21:o		Yes [ ] No [ ]

A.7.4 Other feature events

A.7.4.1 Call information

Item	Service / Feature	Reference	Status	N/A	Supported?
B22	Call information event	S11.2.4.1 P10.2.2	B1b:o		Yes [ ] No [ ]
B22a	Connection ID	P10.2.2	B22:m	[ ]	Yes [ ]
B22b	Device	P10.2.2	B22:m	[ ]	Yes [ ]
B22c	Account information	P10.2.2	B22:o		Yes [ ] No [ ]
B22d	Authorization code	P10.2.2	B22:o		Yes [ ] No [ ]
B22e	Correlator data	P10.2.2	B22:o		Yes [ ] No [ ]

A.7.4.2 Do not disturb

Item	Service / Feature	Reference	Status	N/A	Supported?
B23	Do not disturb event	S11.2.4.2 P10.2.3	B1b:o		Yes [ ] No [ ]
B23a	Device	P10.2.3	B23:m	[ ]	Yes [ ]
B23b	Do not disturb on	P10.2.3	B23:m	[ ]	Yes [ ]

A.7.4.3 Forwarding

Item	Service / Feature	Reference	Status	N/A	Supported?
B24	Forwarding event	S11.2.4.3 P10.2.4	B1b:o		Yes [ ] No [ ]
B24a	Device	P10.2.4	B24:m	[ ]	Yes [ ]
B24b	Forwarding type	P18.6	B24:m	[ ]	Yes [ ]
B24c	Forward DN	P18.6	B24:o		Yes [ ] No [ ]

A.7.4.4 Message waiting

Item	Service / Feature	Reference	Status	N/A	Supported?
B25	Message waiting event	S11.2.4.4 P10.2.5	B1b:o		Yes [ ] No [ ]
B25a	Device for message	P10.2.5	B25:m	[ ]	Yes [ ]
B25b	Invoking device	P10.2.5	B25:m	[ ]	Yes [ ]
B25c	Message waiting on	P10.2.5	B25:m	[ ]	Yes [ ]

A.7.4.5 Auto answer

Item	Service / Feature	Reference	Status	N/A	Supported?
B26	Auto answer event	S11.2.4.5 P10.2.1	B1b:o		Yes [ ] No [ ]
B26a	Device	P10.2.1	B26:m	[ ]	Yes [ ]
B26b	Auto answer on	P10.2.1	B26:m	[ ]	Yes [ ]

A.7.4.6 Microphone mute

Item	Service / Feature	Reference	Status	N/A	Supported?
B27	Microphone mute event	S11.2.4.6 P10.2.6	B1b:o		Yes [ ] No [ ]
B27a	Device	P10.2.6	B27:m	[ ]	Yes [ ]
B27b	Microphone mute on	P10.2.6	B27:m	[ ]	Yes [ ]

A.7.4.7 Speaker mute

Item	Service / Feature	Reference	Status	N/A	Supported?
B28	Speaker mute event	S11.2.4.7 P10.2.7	B1b:o		Yes [ ] No [ ]
B28a	Device	P10.2.7	B28:m	[ ]	Yes [ ]
B28b	Speaker mute on	P10.2.7	B28:m	[ ]	Yes [ ]

A.7.4.8 Speaker volume

Item	Service / Feature	Reference	Status	N/A	Supported?
B29	Speaker volume event	S11.2.4.8 P10.2.8	B1b:o		Yes [ ] No [ ]
B29a	Device	P10.2.8	B29:m	[ ]	Yes [ ]
B29b	Speaker volume	P10.2.8	B29:m	[ ]	Yes [ ]

A.7.5 Maintenance events

A.7.5.1 Back in service

Item	Service / Feature	Reference	Status	N/A	Supported?
B30	Back in service event	S11.2.5.1 P10.4.1	B1b:o		Yes [ ] No [ ]
B30a	Device	P10.4.1	B30:m	[ ]	Yes [ ]
B30b	Cause	P10.4.1	B30:o		Yes [ ] No [ ]

A.7.5.2 Out of service

Item	Service / Feature	Reference	Status	N/A	Supported?
B31	Out of service event	S10.2.5.2 P10.4.2	B1b:o		Yes [ ] No [ ]
B31a	Device	P10.4.2	B31:m	[ ]	Yes [ ]
B31b	Cause	P10.4.2	B31:o		Yes [ ] No [ ]

A.7.6 Private events

A.7.6.1 Private

Item	Service / Feature	Reference	Status	N/A	Supported?
B32	Is/are private event(s)	S11.2.6 P10.5	B1b:o		Yes [ ] No [ ]

A.7.7 Voice unit events

A.7.7.1 Play

Item	Service / Feature	Reference	Status	N/A	Supported?
B33	Play event	S11.2.7.1 P10.6.1	B1b:o		Yes [ ] No [ ]
B33a	Connection	P10.6.1	B33:m	[ ]	Yes [ ]
B33b	Message	P10.6.1	B33:m	[ ]	Yes [ ]
B33c	Message length	P10.6.1	B33:o		Yes [ ] No [ ]
B33d	Current position in message	P10.6.1	B33:o		Yes [ ] No [ ]
B33e	Speed	P10.6.1	B33:o		Yes [ ] No [ ]
B33f	Cause	P10.6.1	B33:o		Yes [ ] No [ ]

A.7.7.2 Record

Item	Service / Feature	Reference	Status	N/A	Supported?
B34	Record event	S11.2.7.2 P10.6.2	B1b:o		Yes [ ] No [ ]
B34a	Connection	P10.6.2	B34:m	[ ]	Yes [ ]
B34b	Message	P10.6.2	B34:m	[ ]	Yes [ ]
B34c	Message length	P10.6.2	B34:o		Yes [ ] No [ ]
B34d	Current position in message	P10.6.2	B34:o		Yes [ ] No [ ]
B34e	Cause	P10.6.2	B34:o		Yes [ ] No [ ]

A.7.7.3 Review

Item	Service / Feature	Reference	Status	N/A	Supported?
B35	Review event	S11.2.7.3 P10.6.3	B1b:o		Yes [ ] No [ ]
B35a	Connection	P10.6.3	B35:m	[ ]	Yes [ ]
B35b	Message	P10.6.3	B35:m	[ ]	Yes [ ]
B35c	Message length	P10.6.3	B35:o		Yes [ ] No [ ]
B35d	Current position in message	P10.6.3	B35:o		Yes [ ] No [ ]
B35e	Cause	P10.6.3	B35:o		Yes [ ] No [ ]

A.7.7.4 Stop

Item	Service / Feature	Reference	Status	N/A	Supported?
B36	Stop event	S11.2.7.4 P10.6.4	B1b:o		Yes [ ] No [ ]
B36a	Connection	P10.6.4	B36:m	[ ]	Yes [ ]
B36b	Message	P10.6.4	B36:m	[ ]	Yes [ ]
B36c	Message length	P10.6.4	B36:o		Yes [ ] No [ ]
B36d	Current position in message	P10.6.4	B36:o		Yes [ ] No [ ]
B36e	Cause	P10.6.4	B36:o		Yes [ ] No [ ]

A.7.7.5 Suspend play

Item	Service / Feature	Reference	Status	N/A	Supported?
B37	Suspend play event	S11.2.7.5 P10.6.5	B1b:o		Yes [ ] No [ ]
B37a	Connection	P10.6.5	B37:m	[ ]	Yes [ ]
B37b	Message	P10.6.5	B37:m	[ ]	Yes [ ]
B37c	Message length	P10.6.5	B37:o		Yes [ ] No [ ]
B37d	Current position in message	P10.6.5	B37:o		Yes [ ] No [ ]
B37e	Cause	P10.6.5	B37:o		Yes [ ] No [ ]

A.7.7.6 Suspend record

Item	Service / Feature	Reference	Status	N/A	Supported?
B38	Suspend record event	S11.2.7.6 P10.6.6	B1b:o		Yes [ ] No [ ]
B38a	Connection	P10.6.6	B38:m	[ ]	Yes [ ]
B38b	Message	P10.6.6	B38:m	[ ]	Yes [ ]
B38c	Message length	P10.6.6	B38:o		Yes [ ] No [ ]
B38d	Current position in message	P10.6.6	B38:o		Yes [ ] No [ ]
B38e	Cause	P10.6.6	B38:o		Yes [ ] No [ ]

A.7.7.7 Voice attribute change

Item	Service / Feature	Reference	Status	N/A	Supported?
B39	Voice attribute change event	S11.2.7.7 P10.6.7	B1b:o		Yes [ ] No [ ]
B39a	Connection	P10.6.7	B39:m	[ ]	Yes [ ]
B39b	Message	P10.6.7	B39:m	[ ]	Yes [ ]
B39c	Speaker volume	P10.6.7	B39:o		Yes [ ] No [ ]
B39d	Recording level	P10.6.7	B39:o		Yes [ ] No [ ]
B39e	Speed	P10.6.7	B39:o		Yes [ ] No [ ]
B39f	Current position in message	P10.6.7	B39:o		Yes [ ] No [ ]
B39g	Cause	P10.6.7	B39:o		Yes [ ] No [ ]

A.8 Computing function services

A.8.1 Route request

Item	Service / Feature	Reference	Status	N/A	Supported?
C1	Route request service	S12.3 P11.1	o		Yes [ ] No [ ]
C1a	Cross reference	P11.1	C1:m	[ ]	Yes [ ]
C1b	Current route	P11.1	C1:m	[ ]	Yes [ ]
C1c	Calling device	P11.1	C1:o		Yes [ ] No [ ]
C1d	Routing device	P11.1	C1:o		Yes [ ] No [ ]
C1e	Routed call	P11.1	C1:o		Yes [ ] No [ ]
C1f	Route select algorithm	P11.1	C1:o		Yes [ ] No [ ]
C1g	Priority	P11.1	C1:o		Yes [ ] No [ ]
C1h	Device profile	P11.1	C1:o		Yes [ ] No [ ]
C1i	Device class	P18.6	C1h:m	[ ]	Yes [ ]
C1j	ISDN set up	P18.6	C1h:o		Yes [ ] No [ ]
C1k	Correlator data	P11.1	C1:o		Yes [ ] No [ ]
C1l	Private data in request	P18.8	C1:o		Yes [ ] No [ ]
C1m	Security data in request	P18.7	C1:o		Yes [ ] No [ ]
C1n	Service result negative	P16	C1:m	[ ]	Yes [ ]

A.8.2 Reroute request

Item	Service / Feature	Reference	Status	N/A	Supported?
C2	Reroute request service	S12.1 P11.2	o		Yes [ ] No [ ]
C2a	Cross reference	P11.2	C2:m	[ ]	Yes [ ]
C2b	Private data in request	P18.8	C2:o		Yes [ ] No [ ]
C2c	Security data in request	P18.7	C2:o		Yes [ ] No [ ]
C2d	Service result negative	P16	C2:m	[ ]	Yes [ ]

A.8.3 Route select request

Item	Service / Feature	Reference	Status	N/A	Supported?
C3	Route select request service	S12.4 P11.3	o		Yes [ ] No [ ]
C3a	Cross reference	P11.3	C3:m	[ ]	Yes [ ]
C3b	Route selected	P11.3	C3:m	[ ]	Yes [ ]
C3c	Remaining retries	P11.3	C3:o		Yes [ ] No [ ]
C3d	Device profile	P11.3	C3:o		Yes [ ] No [ ]
C3e	Device class	P18.6	C3d:m	[ ]	Yes [ ]
C3f	ISDN set up	P18.6	C3d:o		Yes [ ] No [ ]
C3g	Route used request positive	P11.3	C3:o		Yes [ ] No [ ]
C3h	Correlator data	P11.3	C3:o		Yes [ ] No [ ]
C3i	Private data in request	P18.8	C3:o		Yes [ ] No [ ]
C3j	Security data in request	P18.7	C3:o		Yes [ ] No [ ]
C3k	Service result negative	P16	C3:m	[ ]	Yes [ ]

A.8.4 Route used request

Item	Service / Feature	Reference	Status	N/A	Supported?
C4	Route used request service	S12.5 P11.4	o		Yes [ ] No [ ]
C4a	Cross reference	P11.4	C4:m	[ ]	Yes [ ]
C4b	Route used	P11.4	C4:m	[ ]	Yes [ ]
C4c	Calling device	P11.4	C4:o		Yes [ ] No [ ]
C4d	Domain	P11.4	C4:o		Yes [ ] No [ ]
C4e	Correlator data	P11.4	C4:o		Yes [ ] No [ ]
C4f	Private data in request	P18.8	C4:o		Yes [ ] No [ ]
C4g	Security data in request	P18.7	C4:o		Yes [ ] No [ ]
C4h	Service result negative	P16	C4:m	[ ]	Yes [ ]

A.8.5 Route end request

Item	Service / Feature	Reference	Status	N/A	Supported?
C5	Route end request service	S12.2 P11.5	o		Yes [ ] No [ ]
C5a	Cross reference	P11.5	C5:m	[ ]	Yes [ ]
C5b	Error value	P11.5	C5:o		Yes [ ] No [ ]
C5c	Private data in request	P18.8	C5:o		Yes [ ] No [ ]
C5d	Security data in request	P18.7	C5:o		Yes [ ] No [ ]
C5e	Service result negative	P16	C5:o		Yes [ ] No [ ]

A.9 Bidirectional services

A.9.1 Escape

Item	Service / Feature	Reference	Status	N/A	Supported?
D1	Escape service	S13.1 P12.1	o		Yes [ ] No [ ]
D1a	Security data in request	P18.7	D1:o		Yes [ ] No [ ]
D1b	Private data in request	P18.8	D1:m	[ ]	Yes [ ]
D1c	Service result positive	P12.1	D1:m	[ ]	Yes [ ]
D1d	Service result negative	P16	D1:m	[ ]	Yes [ ]
D1e	Security data in result	P18.7	D1c:o		Yes [ ] No [ ]
D1f	Private data in result	P18.8	D1c:o		Yes [ ] No [ ]

A.9.2 System status

Item	Service / Feature	Reference	Status	N/A	Supported?
D2	System status service	S13.2 P12.2	o		Yes [ ] No [ ]
D2a	System status cause	P12.2	D2:m	[ ]	Yes [ ]
D2b	Initializing	P18.6	D2a:o		Yes [ ] No [ ]
D2c	Enabled	P18.6	D2a:o		Yes [ ] No [ ]
D2d	Normal	P18.6	D2a:o		Yes [ ] No [ ]
D2e	Messages lost	P18.6	D2a:o		Yes [ ] No [ ]
D2f	Disabled	P18.6	D2a:o		Yes [ ] No [ ]
D2g	Overload imminent	P18.6	D2a:o		Yes [ ] No [ ]
D2h	Overload reached	P18.6	D2a:o		Yes [ ] No [ ]
D2i	Overload relieved	P18.6	D2a:o		Yes [ ] No [ ]
D2j	Security data in request	P18.7	D2:o		Yes [ ] No [ ]
D2k	Private data in request	P18.8	D2:o		Yes [ ] No [ ]
D2l	Service result positive	P12.2	D2:m	[ ]	Yes [ ]
D2m	Service result negative	P16	D2:m	[ ]	Yes [ ]
D2n	Security data in result	P18.7	D2l:o		Yes [ ] No [ ]
D2o	Private data in result	P18.8	D2l:o		Yes [ ] No [ ]

A.10 Status reporting services

A.10.1 Change monitor filter

Item	Service / Feature	Reference	Status	N/A	Supported?
E1	Change monitor filter service	S11.1 P13.2	o		Yes [ ] No [ ]
E1a	Cross reference ID in request	P13.2	E1:m	[ ]	Yes [ ]
E1b	Filter list in request	P13.2	E1:m	[ ]	Yes [ ]
E1c	Security data in request	P18.7	E1:o		Yes [ ] No [ ]
E1d	Private data in request	P18.8	E1:o		Yes [ ] No [ ]
E1e	Service result positive	P13.2	E1:m	[ ]	Yes [ ]
E1f	Service result negative	P16	E1:m	[ ]	Yes [ ]
E1g	Filter list in result	P13.2	E1e:o		Yes [ ] No [ ]
E1h	Security data in result	P18.7	E1e:o		Yes [ ] No [ ]
E1i	Private data in result	P18.8	E1e:o		Yes [ ] No [ ]

A.10.2 Monitor start

Item	Service / Feature	Reference	Status	N/A	Supported?
E2	Monitor start service	S11.3 P13.1	o		Yes [ ] No [ ]
E2a	Monitor object	P13.1	E2:m	[ ]	Yes [ ]
E2b	Monitor object device	P18.5	E2a:o1		Yes [ ] No [ ]
E2c	Monitor object call	P18.5	E2a:o1		Yes [ ] No [ ]
E2d	Monitor type device	P18.5	E2a:o		Yes [ ] No [ ]
E2e	Monitor type call	P18.5	E2a:o		Yes [ ] No [ ]
E2f	Monitor filter for call processing events	P18.5	E2:o		Yes [ ] No [ ]
E2g	Monitor filter for feature events	P18.5	E2:o		Yes [ ] No [ ]
E2h	Monitor filter for agent events	P18.5	E2:o		Yes [ ] No [ ]
E2i	Monitor filter for maintenance events	P18.5	E2:o		Yes [ ] No [ ]
E2j	Monitor filter for voice unit events	P18.5	E2:o		Yes [ ] No [ ]
E2k	Monitor filter for private filter event	P18.5	E2:o		Yes [ ] No [ ]
E2l	Service result positive	P13.1	E2:m	[ ]	Yes [ ]
E2m	Service result negative	P16	E2:m	[ ]	Yes [ ]
E2n	Cross reference ID in result	P13.1	E2l:m	[ ]	Yes [ ]
E2o	Monitor filter in result	P13.1	E2l:o		Yes [ ] No [ ]
E2p	Security data in request	P18.7	E2:o		Yes [ ] No [ ]
E2q	Private data in request	P18.8	E2:o		Yes [ ] No [ ]
E2r	Security data in result	P18.7	E2l:o		Yes [ ] No [ ]
E2s	Private data in result	P18.8	E2l:o		Yes [ ] No [ ]

A.10.3 Monitor stop

Item	Service / Feature	Reference	Status	N/A	Supported?
E3	Monitor stop service	S11.4 P13.3	o		Yes [ ] No [ ]
E3a	Cross reference ID in request	P13.3	E3:m	[ ]	Yes [ ]
E3b	Security data in request	P18.7	E3:o		Yes [ ] No [ ]
E3c	Private data in request	P18.8	E3:o		Yes [ ] No [ ]
E3d	Service result positive	P13.3	E3:m	[ ]	Yes [ ]
E3e	Service result negative	P16	E3:m	[ ]	Yes [ ]
E3f	Security data in result	P18.7	E3d:o		Yes [ ] No [ ]
E3g	Private data in result	P18.8	E3d:o		Yes [ ] No [ ]

A.10.4 Snapshot call

Item	Service / Feature	Reference	Status	N/A	Supported?
E4	Snapshot call service	S11.5 P13.5	o		Yes [ ] No [ ]
E4a	Snapshot object in request	P13.5	E4:m	[ ]	Yes [ ]
E4b	Security data in request	P18.7	E4:o		Yes [ ] No [ ]
E4c	Private data in request	P18.8	E4:o		Yes [ ] No [ ]
E4d	Service result positive	P13.5	E4:m	[ ]	Yes [ ]
E4e	Service result negative	P16	E4:m	[ ]	Yes [ ]
E4f	Device ID in result	P13.5	E4d:m	[ ]	Yes [ ]
E4g	Connection ID in result	P13.5	E4d:m	[ ]	Yes [ ]
E4h	Device profile	P13.5	E4d:o		Yes [ ] No [ ]
E4i	Device class	P18.6	E4h:m	[ ]	Yes [ ]
E4j	ISDN set up	P18.6	E4h:o		Yes [ ] No [ ]
E4k	Local connection state in result	P13.5	E4d:o		Yes [ ] No [ ]
E4l	Security data in result	P18.7	E4d:o		Yes [ ] No [ ]
E4m	Private data in result	P18.8	E4d:o		Yes [ ] No [ ]

A.10.5 Snapshot device

Item	Service / Feature	Reference	Status	N/A	Supported?
E5	Snapshot device service	S11.6 P13.4	o		Yes [ ] No [ ]
E5a	Device ID in request	P13.4	E5:m	[ ]	Yes [ ]
E5b	Security data in request	P18.7	E5:o		Yes [ ] No [ ]
E5c	Private data in request	P18.8	E5:o		Yes [ ] No [ ]
E5d	Service result positive	P13.4	E5:m	[ ]	Yes [ ]
E5e	Service result negative	P16	E5:m	[ ]	Yes [ ]
E5f	Connection ID in result	P13.4	E5d:m	[ ]	Yes [ ]
E5g	Call state in result	P13.4	E5d:m	[ ]	Yes [ ]
E5h	Device profile	P13.4	E5d:o		Yes [ ] No [ ]
E5i	Device class	P18.6	E5h:m	[ ]	Yes [ ]
E5j	ISDN set up	P18.6	E5h:o		Yes [ ] No [ ]
E5k	Security data in result	P18.7	E5d:o		Yes [ ] No [ ]
E5l	Private data in result	P18.8	E5d:o		Yes [ ] No [ ]

A.11 Input/output services

A.11.1 Data path resumed

Item	Service / Feature	Reference	Status	N/A	Supported?
F1	Data path resumed service	S14.1 P14.9	o		Yes [ ] No [ ]
F1a	Cross reference ID	P14.9	F1:m	[ ]	Yes [ ]
F1b	Security data in request	P18.7	F1:o		Yes [ ] No [ ]
F1c	Private data in request	P18.8	F1:o		Yes [ ] No [ ]
F1d	Service result positive	P14.9	F1:m	[ ]	Yes [ ]
F1e	Service result negative	P16	F1:m	[ ]	Yes [ ]
F1f	Security data in result	P18.7	F1d:o		Yes [ ] No [ ]
F1g	Private data in result	P18.8	F1d:o		Yes [ ] No [ ]

A.11.2 Data path suspended

Item	Service / Feature	Reference	Status	N/A	Supported?
F2	Data path suspended service	S14.2 P14.7	o		Yes [ ] No [ ]
F2a	Cross reference ID	P14.7	F2:m	[ ]	Yes [ ]
F2b	Security data in request	P18.7	F2:o		Yes [ ] No [ ]
F2c	Private data in request	P18.8	F2:o		Yes [ ] No [ ]
F2d	Service result positive	P14.7	F2:m	[ ]	Yes [ ]
F2e	Service result negative	P16	F2:m	[ ]	Yes [ ]
F2f	Security data in result	P18.7	F2d:o		Yes [ ] No [ ]
F2g	Private data in result	P18.8	F2d:o		Yes [ ] No [ ]

A.11.3 Fast data

Item	Service / Feature	Reference	Status	N/A	Supported?
F3	Fast data service	S14.3 P14.10	o		Yes [ ] No [ ]
F3a	CSTA object	P14.10	F3:m	[ ]	Yes [ ]
F3b	Object device	P14.10	F3a:o1		Yes [ ] No [ ]
F3c	Object call	P14.10	F3a:o1		Yes [ ] No [ ]
F3d	I/O data	P14.10	F3:m	[ ]	Yes [ ]
F3e	Data path type text	P14.10	F3:o		Yes [ ] No [ ]
F3f	Data path type voice	P14.10	F3:o		Yes [ ] No [ ]
F3g	Security data in request	P18.7	F3:o		Yes [ ] No [ ]
F3h	Private data in request	P18.8	F3:o		Yes [ ] No [ ]
F3i	Service result positive	P14.10	F3:m	[ ]	Yes [ ]
F3j	Service result negative	P16	F3:m	[ ]	Yes [ ]
F3k	Security data in result	P18.7	F3i:o		Yes [ ] No [ ]
F3l	Private data in result	P18.8	F3i:o		Yes [ ] No [ ]

A.11.4 Resume data path

Item	Service / Feature	Reference	Status	N/A	Supported?
F4	Resume data path service	S14.4 P14.8	o		Yes [ ] No [ ]
F4a	Cross reference ID	P14.8	F4:m	[ ]	Yes [ ]
F4b	Security data in request	P18.7	F4:o		Yes [ ] No [ ]
F4c	Private data in request	P18.8	F4:o		Yes [ ] No [ ]
F4d	Service result positive	P14.8	F4:m	[ ]	Yes [ ]
F4e	Service result negative	P16	F4:m	[ ]	Yes [ ]
F4f	Security data in result	P18.7	F4d:o		Yes [ ] No [ ]
F4g	Private data in result	P18.8	F4d:o		Yes [ ] No [ ]

A.11.5 Send broadcast data

Item	Service / Feature	Reference	Status	N/A	Supported?
F5	Send broadcast data service	S14.5 P14.5	o		Yes [ ] No [ ]
F5a	I/O data	P14.5	F5:m	[ ]	Yes [ ]
F5b	Data path type text	P14.5	F5:o		Yes [ ] No [ ]
F5c	Data path type voice	P14.5	F5:o		Yes [ ] No [ ]
F5d	Security data in request	P18.7	F5:o		Yes [ ] No [ ]
F5e	Private data in request	P18.8	F5:o		Yes [ ] No [ ]
F5f	Service result positive	P14.5	F5:m	[ ]	Yes [ ]
F5g	Service result negative	P16	F5:m	[ ]	Yes [ ]
F5h	Security data in result	P18.7	F5f:o		Yes [ ] No [ ]
F5i	Private data in result	P18.8	F5f:o		Yes [ ] No [ ]

A.11.6 Send data

Item	Service / Feature	Reference	Status	N/A	Supported?
F6	Send data service	S14.6 P14.3	o		Yes [ ] No [ ]
F6a	Cross reference ID	P14.3	F6:m	[ ]	Yes [ ]
F6b	I/O data	P14.3	F6:m	[ ]	Yes [ ]
F6c	Cause code	S10.4 P16	F6:o		Yes [ ] No [ ]
F6d	Security data in request	P18.7	F6:o		Yes [ ] No [ ]
F6e	Private data in request	P18.8	F6:o		Yes [ ] No [ ]
F6f	Service result positive	P14.3	F6:m	[ ]	Yes [ ]
F6g	Service result negative	P16	F6:m	[ ]	Yes [ ]
F6h	Security data in result	P18.7	F6f:o		Yes [ ] No [ ]
F6i	Private data in result	P18.8	F6f:o		Yes [ ] No [ ]

A.11.7 Send multicast data

Item	Service / Feature	Reference	Status	N/A	Supported?
F7	Send multicast data service	S14.7 P14.4	o		Yes [ ] No [ ]
F7a	I/O cross reference ID list	P14.4	F7:m	[ ]	Yes [ ]
F7b	I/O data	P14.4	F7:m	[ ]	Yes [ ]
F7c	Security data in request	P18.7	F7:o		Yes [ ] No [ ]
F7d	Private data in request	P18.8	F7:o		Yes [ ] No [ ]
F7e	Service result positive	P14.4	F7:m	[ ]	Yes [ ]
F7f	Service result negative	P16	F7:m	[ ]	Yes [ ]
F7g	Security data in result	P18.7	F7e:o		Yes [ ] No [ ]
F7h	Private data in result	P18.8	F7e:o		Yes [ ] No [ ]

A.11.8 Start data path

Item	Service / Feature	Reference	Status	N/A	Supported?
F8	Start data path service	S14.8 P14.1	o		Yes [ ] No [ ]
F8a	CSTA object	P14.1	F8:m	[ ]	Yes [ ]
F8b	Object device	P14.1	F8:o1		Yes [ ] No [ ]
F8c	Object call	P14.1	F8:o1		Yes [ ] No [ ]
F8d	Data path from the computing function	P18.6	F8:o		Yes [ ] No [ ]
F8e	Data path to the computing function	P18.6	F8:o		Yes [ ] No [ ]
F8f	Data path bidirectional	P18.6	F8:o		Yes [ ] No [ ]
F8g	Data path type text	P14.1	F8:o		Yes [ ] No [ ]
F8h	Data path type voice	P14.1	F8:o		Yes [ ] No [ ]
F8i	Number of characters to collect	P14.1	F8:o		Yes [ ] No [ ]
F8j	Termination character	P14.1	F8:o		Yes [ ] No [ ]
F8k	Timeout	P14.1	F8:o		Yes [ ] No [ ]
F8l	Cross reference ID in result	P14.1	F8:m	[ ]	Yes [ ]
F8m	Number of characters to collect in result	P14.1	F8:o		Yes [ ] No [ ]
F8n	Termination character in result	P14.1	F8:o		Yes [ ] No [ ]
F8o	Timeout in result	P14.1	F8:o		Yes [ ] No [ ]
F8p	Security data in request	P18.7	F8:o		Yes [ ] No [ ]
F8q	Private data in request	P18.8	F8:o		Yes [ ] No [ ]
F8r	Service result positive	P14.1	F8:m	[ ]	Yes [ ]
F8s	Service result negative	P16	F8:m	[ ]	Yes [ ]
F8t	Security data in result	P18.7	F8r:o		Yes [ ] No [ ]
F8u	Private data in result	P18.8	F8r:o		Yes [ ] No [ ]

A.11.9 Stop data path

Item	Service / Feature	Reference	Status	N/A	Supported?
F9	Stop data path service	S14.9 P14.2	o		Yes [ ] No [ ]
F9a	CSTA cross reference ID	P14.2	F9:m	[ ]	Yes [ ]
F9b	Security data in request	P18.7	F9:o		Yes [ ] No [ ]
F9c	Private data in request	P18.8	F9:o		Yes [ ] No [ ]
F9d	Service result positive	P14.2	F9:m	[ ]	Yes [ ]
F9e	Service result negative	P16	F9:m	[ ]	Yes [ ]
F9f	Security data in result	P18.7	F9d:o		Yes [ ] No [ ]
F9g	Private data in result	P18.8	F9d:o		Yes [ ] No [ ]

A.11.10 Suspend data path

Item	Service / Feature	Reference	Status	N/A	Supported?
F10	Suspend data path service	S14.10 P14.6	o		Yes [ ] No [ ]
F10a	CSTA cross reference ID	P14.6	F10:m	[ ]	Yes [ ]
F10b	Security data in request	P18.7	F10:o		Yes [ ] No [ ]
F10c	Private data in request	P18.8	F10:o		Yes [ ] No [ ]
F10d	Service result positive	P14.6	F10:m	[ ]	Yes [ ]
F10e	Service result negative	P16	F10:m	[ ]	Yes [ ]
F10f	Security data in result	P18.7	F10d:o		Yes [ ] No [ ]
F10g	Private data in result	P18.8	F10d:o		Yes [ ] No [ ]

A.12 Voice unit services

A.12.1 Concatenate message

Item	Service / Feature	Reference	Status	N/A	Supported?
G1	Concatenate message service	S15.1 P15.1	o		Yes [ ] No [ ]
G1a	Message ID list	P15.1	G1:m	[ ]	Yes [ ]
G1b	Security data in request	P18.7	G1:o		Yes [ ] No [ ]
G1c	Private data in request	P18.8	G1:o		Yes [ ] No [ ]
G1d	Service result positive	P15.1	G1:m	[ ]	Yes [ ]
G1e	Service result negative	P16	G1:m	[ ]	Yes [ ]
G1f	Security data in result	P18.7	G1d:o		Yes [ ] No [ ]
G1g	Private data in result	P18.8	G1d:o		Yes [ ] No [ ]

A.12.2 Delete message

Item	Service / Feature	Reference	Status	N/A	Supported?
G2	Delete message service	S15.2 P15.2	o		Yes [ ] No [ ]
G2a	Message ID	P15.2	G2:m	[ ]	Yes [ ]
G2b	Security data in request	P18.7	G2:o		Yes [ ] No [ ]
G2c	Private data in request	P18.8	G2:o		Yes [ ] No [ ]
G2d	Service result positive	P15.2	G2:m	[ ]	Yes [ ]
G2e	Service result negative	P16	G2:m	[ ]	Yes [ ]
G2f	Security data in result	P18.7	G2d:o		Yes [ ] No [ ]
G2g	Private data in result	P18.8	G2d:o		Yes [ ] No [ ]

A.12.3 Play message

Item	Service / Feature	Reference	Status	N/A	Supported?
G3	Play message service	S15.3 P15.3	o		Yes [ ] No [ ]
G3a	Message ID	P15.3	G3:m	[ ]	Yes [ ]
G3b	Call associated with message	P15.3	G3:m	[ ]	Yes [ ]
G3c	Duration	P15.3	G3:o		Yes [ ] No [ ]
G3d	Duration exceeded	P15.3	G3:o		Yes [ ] No [ ]
G3e	DTMF detected	P15.3	G3:o		Yes [ ] No [ ]
G3f	End of data detected	P15.3	G3:o		Yes [ ] No [ ]
G3g	Speech detected	P15.3	G3:o		Yes [ ] No [ ]
G3h	Security data in request	P18.7	G3:o		Yes [ ] No [ ]
G3i	Private data in request	P18.8	G3:o		Yes [ ] No [ ]
G3j	Service result positive	P15.3	G3:m	[ ]	Yes [ ]
G3k	Service result negative	P16	G3:m	[ ]	Yes [ ]
G3l	Security data in result	P18.7	G3j:o		Yes [ ] No [ ]
G3m	Private data in result	P18.8	G3j:o		Yes [ ] No [ ]

A.12.4 Query voice attribute

Item	Service / Feature	Reference	Status	N/A	Supported?
G4	Query voice attribute service	S15.4 P15.4	o		Yes [ ] No [ ]
G4a	Message ID	P15.4	G4:m	[ ]	Yes [ ]
G4b	Connection associated with message	P15.4	G4:o		Yes [ ] No [ ]
G4c	Query attribute	P15.4	G4:m	[ ]	Yes [ ]
G4d	Attribute encode algorithm	P15.4	G4:o1		Yes [ ] No [ ]
G4e	Attribute sampling rate	P15.4	G4:o1		Yes [ ] No [ ]
G4f	Attribute duration	P15.4	G4:o1		Yes [ ] No [ ]
G4g	Attribute file name	P15.4	G4:o1		Yes [ ] No [ ]
G4h	Attribute current position	P15.4	G4:o1		Yes [ ] No [ ]
G4i	Attribute current speed	P15.4	G4:o1		Yes [ ] No [ ]
G4j	Attribute current volume	P15.4	G4:o1		Yes [ ] No [ ]
G4k	Attribute current level	P15.4	G4:o1		Yes [ ] No [ ]
G4l	Attribute current state	P15.4	G4:o1		Yes [ ] No [ ]
G4m	Attribute in result	P15.4	G4ag:m	[ ]	Yes [ ]
G4n	ADPCM-6k	P15.4	G4d:o2		Yes [ ] No [ ]
G4o	ADPCM-8k	P15.4	G4d:o2		Yes [ ] No [ ]
G4p	Mu-Law PCM 6k	P15.4	G4d:o2		Yes [ ] No [ ]
G4q	A-Law PCM 6k	P15.4	G4d:o2		Yes [ ] No [ ]
G4r	Sampling rate in result	P15.4	G4e:m	[ ]	Yes [ ]
G4s	Duration in result	P15.4	G4f:m	[ ]	Yes [ ]
G4t	File name in result	P15.4	G4g:m	[ ]	Yes [ ]
G4u	Current position in result	P15.4	G4h:m	[ ]	Yes [ ]
G4v	Current speed in result	P15.4	G4i:m	[ ]	Yes [ ]
G4w	Current volume in result	P15.4	G4j:m	[ ]	Yes [ ]
G4x	Current level in result	P15.4	G4k:m	[ ]	Yes [ ]
G4y	Stop	P15.4	G4l:o3		Yes [ ] No [ ]
G4z	Play	P15.4	G4l:o3		Yes [ ] No [ ]
G4aa	Record	P15.4	G4l:o3		Yes [ ] No [ ]
G4ab	Suspend play	P15.4	G4l:o3		Yes [ ] No [ ]
G4ac	Suspend record	P15.4	G4l:o3		Yes [ ] No [ ]
G4ad	Review	P15.4	G4l:o3		Yes [ ] No [ ]
G4ae	Security data in request	P18.7	G4:o		Yes [ ] No [ ]
G4af	Private data in request	P18.8	G4:o		Yes [ ] No [ ]

G4ag	Service result positive	P15.4	G4:m	[ ]	Yes [ ]
G4ah	Service result negative	P16	G4:m	[ ]	Yes [ ]
G4ai	Security data in result	P18.7	G4ag:o		Yes [ ] No [ ]
G4aj	Private data in result	P18.8	G4ag:o		Yes [ ] No [ ]

A.12.5 Record message

Item	Service / Feature	Reference	Status	N/A	Supported?
G5	Record message service	S15.5 P15.5	o		Yes [ ] No [ ]
G5a	Connection associated with record	P15.5	G5:m	[ ]	Yes [ ]
G5b	Sampling rate	P15.5	G5:o		Yes [ ] No [ ]
G5c	ADPCM-6k	P15.5	G5:o1		Yes [ ] No [ ]
G5d	ADPCM-8k	P15.5	G5:o1		Yes [ ] No [ ]
G5e	Mu-Law PCM 6k	P15.5	G5:o1		Yes [ ] No [ ]
G5f	A-Law PCM 6k	P15.5	G5:o1		Yes [ ] No [ ]
G5g	Maximum duration	P15.5	G5:o		Yes [ ] No [ ]
G5h	DTMF detected	P15.5	G5:o		Yes [ ] No [ ]
G5i	No speech detected	P15.5	G5:o		Yes [ ] No [ ]
G5j	Security data in request	P18.7	G5:o		Yes [ ] No [ ]
G5k	Private data in request	P18.8	G5:o		Yes [ ] No [ ]
G5l	Service result positive	P15.5	G5:m	[ ]	Yes [ ]
G5m	Service result negative	P16	G5:m	[ ]	Yes [ ]
G5n	Security data in result	P18.7	G5l:o		Yes [ ] No [ ]
G5o	Private data in result	P18.8	G5l:o		Yes [ ] No [ ]

A.12.6 Reposition

Item	Service / Feature	Reference	Status	N/A	Supported?
G6	Reposition service	S15.6 P15.6	o		Yes [ ] No [ ]
G6a	Connection associated with message	P15.6	G6:m	[ ]	Yes [ ]
G6b	Period	P15.6	G6:m	[ ]	Yes [ ]
G6c	Message ID	P15.6	G6:o		Yes [ ] No [ ]
G6d	Security data in request	P18.7	G6:o		Yes [ ] No [ ]
G6e	Private data in request	P18.8	G6:o		Yes [ ] No [ ]
G6f	Service result positive	P15.6	G6:m	[ ]	Yes [ ]
G6g	Service result negative	P16	G6:m	[ ]	Yes [ ]
G6h	Security data in result	P18.7	G6f:o		Yes [ ] No [ ]
G6i	Private data in result	P18.8	G6f:o		Yes [ ] No [ ]

A.12.7 Resume

Item	Service / Feature	Reference	Status	N/A	Supported?
G7	Resume service	S15.7 P15.7	o		Yes [ ] No [ ]
G7a	Connection associated with message	P15.7	G7:m	[ ]	Yes [ ]
G7b	Message ID	P15.7	G7:o		Yes [ ] No [ ]
G7c	Duration	P15.7	G7:o		Yes [ ] No [ ]
G7d	Security data in request	P18.7	G7:o		Yes [ ] No [ ]
G7e	Private data in request	P18.8	G7:o		Yes [ ] No [ ]
G7f	Service result positive	P15.7	G7:m	[ ]	Yes [ ]
G7g	Service result negative	P16	G7:m	[ ]	Yes [ ]
G7h	Security data in result	P18.7	G7f:o		Yes [ ] No [ ]
G7i	Private data in result	P18.8	G7f:o		Yes [ ] No [ ]

A.12.8 Review

Item	Service / Feature	Reference	Status	N/A	Supported?
G8	Review service	S15.8 P15.8	o		Yes [ ] No [ ]
G8a	Connection associated with message	P15.8	G8:m	[ ]	Yes [ ]
G8b	Period	P15.8	G8:m	[ ]	Yes [ ]
G8c	Message ID	P15.8	G8:o		Yes [ ] No [ ]
G8d	Security data in request	P18.7	G8:o		Yes [ ] No [ ]
G8e	Private data in request	P18.8	G8:o		Yes [ ] No [ ]
G8f	Service result positive	P15.8	G8:m	[ ]	Yes [ ]
G8g	Service result negative	P16	G8:m	[ ]	Yes [ ]
G8h	Security data in result	P18.7	G8f:o		Yes [ ] No [ ]
G8i	Private data in result	P18.8	G8f:o		Yes [ ] No [ ]

A.12.9 Set voice attribute

Item	Service / Feature	Reference	Status	N/A	Supported?
G9	Set voice attribute service	S15.9 P15.9	o		Yes [ ] No [ ]
G9a	Connection associated with message	P15.9	G9:m	[ ]	Yes [ ]
G9b	Speed	P15.9	G9:o1		Yes [ ] No [ ]
G9c	Volume	P15.9	G9:o1		Yes [ ] No [ ]
G9d	Level	P15.9	G9:o1		Yes [ ] No [ ]
G9e	Message ID	P15.9	G9:o		Yes [ ] No [ ]
G9f	Security data in request	P18.7	G9:o		Yes [ ] No [ ]
G9g	Private data in request	P18.8	G9:o		Yes [ ] No [ ]
G9h	Service result positive	P15.9	G9:m	[ ]	Yes [ ]
G9i	Service result negative	P16	G9:m	[ ]	Yes [ ]
G9j	Security data in result	P18.7	G9h:o		Yes [ ] No [ ]
G9k	Private data in result	P18.8	G9h:o		Yes [ ] No [ ]

A.12.10 Stop

Item	Service / Feature	Reference	Status	N/A	Supported?
G10	Stop service	S15.10 P15.10	o		Yes [ ] No [ ]
G10a	Connection associated with message	P15.10	G10:m	[ ]	Yes [ ]
G10b	Message ID	P15.10	G10:o		Yes [ ] No [ ]
G10c	Security data in request	P18.7	G10:o		Yes [ ] No [ ]
G10d	Private data in request	P18.8	G10:o		Yes [ ] No [ ]
G10e	Service result positive	P15.10	G10:m	[ ]	Yes [ ]
G10f	Service result negative	P16	G10:m	[ ]	Yes [ ]
G10g	Security data in result	P18.7	G10e:o		Yes [ ] No [ ]
G10h	Private data in result	P18.8	G10e:o		Yes [ ] No [ ]

A.12.11 Suspend

Item	Service / Feature	Reference	Status	N/A	Supported?
G11	Suspend service	S15.11 P15.11	o		Yes [ ] No [ ]
G11a	Connection associated with message	P15.11	G11:m	[ ]	Yes [ ]
G11b	Message ID	P15.11	G11:o		Yes [ ] No [ ]
G11c	Security data in request	P18.7	G11:o		Yes [ ] No [ ]
G11d	Private data in request	P18.8	G11:o		Yes [ ] No [ ]
G11e	Service result positive	P15.11	G11:m	[ ]	Yes [ ]
G11f	Service result negative	P16	G11:m	[ ]	Yes [ ]
G11g	Security data in result	P18.7	G11e:o		Yes [ ] No [ ]
G11h	Private data in result	P18.8	G11e:o		Yes [ ] No [ ]

A.12.12 Synthesize message

Item	Service / Feature	Reference	Status	N/A	Supported?
G12	Synthesize message service	S15.12 P15.12	o		Yes [ ] No [ ]
G12a	Text	P15.12	G12:m	[ ]	Yes [ ]
G12b	Male	P15.12	G12:o1		Yes [ ] No [ ]
G12c	Female	P15.12	G12:o1		Yes [ ] No [ ]
G12d	Language	P15.12	G12:o		Yes [ ] No [ ]
G12e	Message ID in result	P15.12	G12:m	[ ]	Yes [ ]
G12f	Security data in request	P18.7	G12:o		Yes [ ] No [ ]
G12g	Private data in request	P18.8	G12:o		Yes [ ] No [ ]
G12h	Service result positive	P15.12	G12:m	[ ]	Yes [ ]
G12i	Service result negative	P16	G12:m	[ ]	Yes [ ]
G12j	Security data in result	P18.7	G12h:o		Yes [ ] No [ ]
G12k	Private data in result	P18.8	G12h:o		Yes [ ] No [ ]

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 can be downloaded from our FTP site, **ftp.ecma.ch,** logging in as **anonymous** and giving your E-mail address as **password**. This Standard is available from library **ECMA-ST** as a compacted, self-expanding file in MSWord 6.0 format (file E218-DOC.EXE) and as an Acrobat PDF file (file E218-PDF.PDF). File E218-EXP.TXT gives a short presentation of the Standard.

Our web site, http://www.ecma.ch, gives full information on ECMA, ECMA activities, ECMA Standards and Technical Reports.

ECMA

114 Rue du Rhône
CH-1204 Geneva
Switzerland

This Standard ECMA-218 is available free of charge in printed form and as a file.

See inside cover page for instructions