INTERNATIONAL STANDARD

ISO/IEC 22534

First edition
2005-05-01

Information technology — Telecommunications and information exchange between systems — Application session services

Technologies de l'information — Télécommunications et échange d'information entre systèmes — Services de session d'application

Reference number
ISO/IEC 22534:2005(E)

PDF disclaimer

This PDF file may contain embedded typefaces. In accordance with Adobe's licensing policy, this file may be printed or viewed but shall not be edited unless the typefaces which are embedded are licensed to and installed on the computer performing the editing. In downloading this file, parties accept therein the responsibility of not infringing Adobe's licensing policy. The ISO Central Secretariat accepts no liability in this area.

Adobe is a trademark of Adobe Systems Incorporated.

Details of the software products used to create this PDF file can be found in the General Info relative to the file; the PDF-creation parameters were optimized for printing. Every care has been taken to ensure that the file is suitable for use by ISO member bodies. In the unlikely event that a problem relating to it is found, please inform the Central Secretariat at the address given below.

ISO copyright office
Case postale 56 • CH-1211 Geneva 20
Tel. + 41 22 749 01 11
Fax + 41 22 749 09 47
E-mail copyright@iso.org
Web www.iso.org

Published in Switzerland

Contents

Page

Foreword iv

Introduction v

1 Scope 1
1.1 Overview 1
1.2 Template Description 1

2 Conformance 2
2.1 Application Conformance 2
2.2 Server Conformance 2

3 Normative references 2

4 Service Definitions 3
4.1 Start Application Session 3
4.1.1 Service Request 3
4.1.2 Service Response 4
4.1.3 Functional Requirements 4
4.2 Stop Application Session 5
4.2.1 Service Request 5
4.2.2 Service Response 5
4.2.3 Functional Requirements 6
4.3 Reset Application Session Timer 6
4.3.1 Service Request 6
4.3.2 Service Response 6
4.3.3 Functional Requirements 7
4.4 Application Session Terminated 7
4.4.1 Service Request 7
4.4.2 Service Response 7
4.4.3 Functional Requirements 8

5 Schema Definitions 8
5.1 Start Application Session 8
5.2 Stop Application Session 10
5.3 Reset Application Session Timer 11
5.4 Application Session Terminated 12

Annex A (informative) **Examples** 14

Annex B (informative) **CSTA Usage Notes** 18

Bibliography 19

Foreword

ISO (the International Organization for Standardization) is a worldwide federation of national standards bodies (ISO member bodies). The work of preparing International Standards is normally carried out through ISO technical committees. Each member body interested in a subject for which a technical committee has been established has the right to be represented on that committee. International organizations, governmental and non-governmental, in liaison with ISO, also take part in the work. ISO collaborates closely with the International Electrotechnical Commission (IEC) on all matters of electrotechnical standardization.

International Standards are drafted in accordance with the rules given in the ISO/IEC Directives, Part 2.

The main task of technical committees is to prepare International Standards. Draft International Standards adopted by the technical committees are circulated to the member bodies for voting. Publication as an International Standard requires approval by at least 75 % of the member bodies casting a vote.

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO shall not be held responsible for identifying any or all such patent rights.

ISO/IEC 22534 was prepared by Ecma International (as ECMA-354) and was adopted, under a special “fast-track procedure”, by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 6, *Telecommunications and information exhange between systems*, in parallel with its approval by national bodies of ISO and IEC.

Introduction

This International Standard defines Ecma Application Session Services – a set of XML-based services used to establish and maintain an application session for exchanging application messages.

The application session established by this International Standard is independent of the underlying transport protocol (TCP, HTTP, etc.) for exchanging application messages.

Sessions for application protocols such as CSTA-XML (ECMA-323) may be established using this International Standard.

Information technology — Telecommunications and information exchange between systems — Application session services

1 Scope

The services defined in this International Standard are used to establish and maintain a relationship between an application and a server for the purpose of exchanging application messages. For the purpose of this International Standard this relationship is called an application session.

Application protocols, such as ECMA-323, require that an application session is established before application messages are exchanged. ECMA-269 specifies several mechanisms for establishing an application context. One possible mechanism is ACSE (ISO/IEC 8649) – but since ASCE uses ASN.1 encoding for its services, it is not desirable for use with XML based protocols such as ECMA-323.

This International Standard provides an XML-based alternative for establishing application sessions.

1.1 Overview

The Ecma Application Session Services consist of the following services:

- Start Application Session. This service is used by an application to establish an application session with a server. A globally unique sessionID is returned by the server in the response message. The sessionID is used in other services to address the session as long as the session exists.

- Stop Application Session. This service is used by an application to stop an application session.

- Reset Application Session Timer. This service is used by an application to reset the timer associated with an application session. If the application timer is not reset before it expires, the application session will be terminated.

- Application Session Terminated – This service is used by the server to indicate that the application session has been abnormally terminated (i.e. not via the Stop Application Session service).

1.2 Template Description

The services specified in Clause 4 include tables that contain the following columns:

- Parameter Name – the name that is used to reference the parameter. This corresponds to the XML element name specified in Clause 5.

- Type – the format of the parameter. This corresponds to the XML element type used in the XML schema definition in Clause 5.

- M/O – the Mandatory/Optional aspect of the parameter. Mandatory parameters must be included in the corresponding XML instance document while optional parameters may be omitted.

- Description – Describes the purpose of the parameter.

2 Conformance

This Clause specifies the conformance requirements for an application and a server to conform to this International Standard.

2.1 Application Conformance

In order to conform to this International Standard an application shall:

1) Support the following services as follows:

   b) Sending the Start Application Session service and all of the mandatory aspects of this service as specified in Clause 4.

   c) Sending the Stop Application Session service and all of the mandatory aspects of this service as specified in Clause 4.

   d) Sending the Reset Application Session service and all of the mandatory aspects of this service as specified in Clause 4.

   e) Receiving the Application Session Terminated service and all of the mandatory aspects of this service as specified in Clause 4.

2) Send and receive XML Instance documents according to the XML Schema specified in Clause 5 in this International Standard for the services listed in item 1.

2.2 Server Conformance

In order to conform to this International Standard a server shall:

1) Support the following services as follows:

   a) Receiving the Start Application Session service and all of the mandatory aspects of this service as specified in Clause 4.

   b) Receiving the Stop Application Session service and all of the mandatory aspects of this service as specified in Clause 4.

   c) Receiving the Reset Application Session service and all of the mandatory aspects of this service as specified in Clause 4.

   d) Sending the Application Session Terminated service and all of the mandatory aspects of this service as specified in Clause 4.

2) Send and receive XML Instance documents according to the XML Schema specified in Clause 5 in this International Standard for the services listed in item 1.

3 Normative references

This Clause has no content in this document.

4 Service Definitions

4.1 Start Application Session

The Start Application Session service is used to initiate an application session between an application and a server.

A globally unique identifier, called a sessionID, is returned in the positive service response that identifies the application session.

Once an application session is established, the server must maintain information associated with the application session. When the application session is terminated, the application context information is cleared.

The application session exists until:

- it is stopped by using a Stop Application Session service
- the session is abnormally terminated by the server as indicated by the Application Session Terminated service (e.g., due to the session timer expiry)
- etc.

4.1.1 Service Request

Table 4-1 — Start Application Session – Service Request

Parameter Name	Type	M/O	Description
applicationInfo	Sequence	M	Specifies information associated with the application requesting the application session. This information consists of: • applicationID (M) – a character string that identifies the application requesting the application association. • applicationSpecificInfo (O) – this consists of a list of one or more implementation specific attributes associated with the application requesting the application association.
requestedProtocolVersions	List of character strings	M	Specifies one or more application protocol versions that the application wishes to use for the application association. The list is ordered by highest priority protocol version first. The server shall choose the highest priority protocol version that it supports and return the chosen protocol version in the positive response. The character string specifying a specific protocol must be a globally unique string (e.g., a standardized namespace (URI) that indicates a given protocol version).
requestedSessionDuration	Value	O	Specifies the length of time (in seconds) that the application session should be maintained. The sessionDuration timer can be periodically refreshed via the Reset Application Session Timer service. If the requestedSessionDuration is not provided the server will choose a default sessionDuration value.
extensions	Sequence	O	Specifies non-standardized information.

4.1.2 Service Response

4.1.2.1 Positive Acknowledgement

Table 4-2 — Start Application Session – Positive Service Response

Parameter Name	Type	M/O	Description
sessionID	Character String	M	Specifies the globally unique identifier associated with the application session that has been created.
actualProtocolVersion	Character String	M	Specifies the protocol version that is being used for the application session. This protocol version shall be one of the protocol versions specified in the service request.
actualSessionDuration	Value	M	Specifies the value for the length of time (in seconds) that the application session will be maintained by the server. This value may be less than or equal to the requestedSessionDuration in the service request. If the requestedSessionDuration is not provided in the service request, the server shall provide a default value which shall be used for the application session.
extensions	Sequence	O	Specifies non-standardized information.

4.1.2.2 Negative Acknowledgement

Table 4-3 — Start Application Session – Negative Service Response

Parameter Name	Type	M/O	Description
errorCode	Character String	M	Specifies the type of error. Either a standardized error or an application specific error may be used. The standardized set of errors are: • invalidApplicationInfo – the server is unable to establish an application session due to invalid or unrecognized information in the applicationInfo parameter. • requestedProtocolVersionNotSupported – none of the requested protocol versions specified in the service request are supported. • serverResourcesBusy – the server cannot establish an application session due to internal resource constraints. • maxNumberSessions – the server cannot create an application session because it has reached the maximum number of allowed application sessions (e.g. license/provisioning limitations).
extensions	Sequence	O	Specifies non-standardized information.

4.1.3 Functional Requirements

1. The sessionID returned in the response is used in other Ecma Application Session services to manage the application session.
2. The positive response includes the actual value of the sessionDuration used for the specified application session. This value may be less than what the application requested in the service request.

3. For an existing session, an application can restart the sessionDuration timer by using the Reset Application Session Timer service.
4. Once the sessionDuration timer expires the server shall terminate the application session and clear any application context information associated with the application session.
5. Mechanisms defined outside of this International Standard may be used to authenticate a user before an application session is started.

4.2 Stop Application Session

The Stop Application Session service is used by the application to terminate an existing application session.

4.2.1 Service Request

Table 4-4 — Stop Application Session – Service Request

Parameter Name	Type	M/O	Description
sessionID	Character String	M	Specifies the globally unique identifier associated with the application session that is being stopped.
sessionEndReason	Character String	O	Specifies the reason that the application session is being stopped. Either a standardized reason or an application specific reason may be used. The standardized set of reasons are: • normal – the application is no longer interested in maintaining this application association.
extensions	Sequence	O	Specifies non-standardized information.

4.2.2 Service Response

4.2.2.1 Positive Acknowledgement

Table 4-5 — Stop Application Session – Positive Service Response

Parameter Name	Type	M/O	Description
extensions	Sequence	O	Specifies non-standardized information.

4.2.2.2 Negative Acknowledgement

Table 4-6 — Stop Application Session – Negative Service Response

Parameter Name	Type	M/O	Description
errorCode	Character String	M	Specifies the type of error. Either a standardized error or an application specific error may be used. The standardized set of errors are: • invalidSessionID – the sessionID is not valid or known by the server.
extensions	Sequence	O	Specifies non-standardized information.

4.2.3 Functional Requirements

1) If the server does not recognize the sessionID in the service request it shall provide a negative service response with the error code “invalidSessionID”.

4.3 Reset Application Session Timer

The Reset Application Session Timer service is used by the application to reset the duration that an existing application session should be maintained.

4.3.1 Service Request

Table 4-7 — Reset Application Session Timer – Service Request

Parameter Name	Type	M/O	Description
sessionID	Character String	M	Specifies the globally unique identifier associated with the application session whose timer is being reset.
requestedSessionDuration	Value	O	Specifies the new value for the length of time (in seconds) that the application requests that the server maintain the application session.
extensions	Sequence	O	Specifies non-standardized information.

4.3.2 Service Response

4.3.2.1 Positive Acknowledgement

Table 4-8 — Reset Application Session Timer – Positive Service Response

Parameter Name	Type	M/O	Description
actualSessionDuration	Value	M	Specifies the new value for the length of time (in seconds) that the application session will be maintained by the server. This value may be less than or equal to the requestedSessionDuration in the service request. If the requestedSessionDuration is not provided in the service request, the server shall provide a default value which shall be used for the application session.
extensions	Sequence	O	Specifies non-standardized information.

4.3.2.2 Negative Acknowledgement

Table 4-9 — Reset Application Session Timer – Negative Service Response

Parameter Name	Type	M/O	Description
errorCode	Character String	M	Specifies the type of error. Either a standardized error or an application specific error may be used. The standardized set of errors are: • invalidSessionID – the sessionID is not valid or known by the server. • serverCannotResetSessionDuration – the server cannot reset the session timer associated with the application session.
extensions	Sequence	O	Specifies non-standardized information.

4.3.3 Functional Requirements

1) A positive response to the service request indicates that the server has reset the session timer associated with the specified application session.

2) The positive response includes the actual value of the sessionTimer used for the specified application session. This value may be less than what the application requested in the service request.

4.4 Application Session Terminated

The Application Session Terminated service is sent by a server when the server has terminated an application session.

4.4.1 Service Request

Table 4-10 — Application Session Terminated – Service Request

Parameter Name	Type	M/O	Description
sessionID	Character String	M	Specifies globally unique identifier associated with the application session that is being terminated.
sessionEndReason	Character String	M	Specifies the reason that the application session has been terminated. Either a standardized reason or an application specific reason may be specified. The standardized set of reasons are: • resourceLimitation – application session terminated due to resource constraints. • sessionTimerExpired – application session terminated due to the sessionTimer expiry.
extensions	Sequence	O	Specifies non-standardized information.

4.4.2 Service Response

There is no acknowledgement to this service request.

4.4.3 Functional Requirements

1) When the server terminates an application session, it clears all of the application context information associated with the application session.

5 Schema Definitions

5.1 Start Application Session

```
<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema targetNamespace="http://www.ecma-international.org/standards/ecma-354/appl_session"
xmlns:xsd="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">

    <xsd:annotation>
        <xsd:documentation>Ecma-Start-Application-Session</xsd:documentation>
    </xsd:annotation>

    <xsd:element name="StartApplicationSession">
        <xsd:complexType>
            <xsd:sequence>
                <xsd:element name="applicationInfo">
                    <xsd:complexType>
                        <xsd:sequence>
                            <xsd:element name="applicationID" type="xsd:string"/>
                            <xsd:element name="applicationSpecificInfo" minOccurs="0">
                                <xsd:complexType>
                                    <xsd:sequence>
                                        <xsd:any namespace="##any" maxOccurs="unbounded"/>
                                    </xsd:sequence>
                                </xsd:complexType>
                            </xsd:element>
                        </xsd:sequence>
                    </xsd:complexType>
                </xsd:element>
                <xsd:element name="requestedProtocolVersions">
                    <xsd:complexType>
                        <xsd:sequence>
                            <xsd:element name="protocolVersion" type="xsd:string" maxOccurs="unbounded"/>
                        </xsd:sequence>
                    </xsd:complexType>
                </xsd:element>
                <xsd:element name="requestedSessionDuration" type="xsd:integer" minOccurs="0"/>
                <xsd:element name="extensions" minOccurs="0">
                    <xsd:complexType>
                        <xsd:sequence>
                            <xsd:any namespace="##any" maxOccurs="unbounded"/>
                        </xsd:sequence>
                    </xsd:complexType>
                </xsd:element>
            </xsd:sequence>
        </xsd:complexType>
    </xsd:element>

    <xsd:element name="StartApplicationSessionPosResponse">
        <xsd:complexType>
            <xsd:sequence>
                <xsd:element name="sessionID" type="xsd:string"/>
```

```
            <xsd:element name="actualProtocolVersion" type="xsd:string"/>
            <xsd:element name="actualSessionDuration" type="xsd:integer"/>
            <xsd:element name="extensions" minOccurs="0">
                <xsd:complexType>
                    <xsd:sequence>
                        <xsd:any namespace="##any" maxOccurs="unbounded"/>
                    </xsd:sequence>
                </xsd:complexType>
            </xsd:element>
        </xsd:sequence>
    </xsd:complexType>
</xsd:element>

<xsd:element name="StartApplicationSessionNegResponse">
    <xsd:complexType>
        <xsd:sequence>
            <xsd:element name="errorCode">
                <xsd:complexType>
                    <xsd:choice>
                        <xsd:element name="definedError">
                            <xsd:simpleType>
                                <xsd:restriction base="xsd:string">
                                    <xsd:enumeration value="invalidApplicationInfo"/>
                                    <xsd:enumeration value="requestedProtocolVersionNotSupported"/>
                                    <xsd:enumeration value="serverResourcesBusy"/>
                                    <xsd:enumeration value="maxNumberSessions"/>
                                </xsd:restriction>
                            </xsd:simpleType>
                        </xsd:element>
                        <xsd:element name="applError" type="xsd:string"/>
                    </xsd:choice>
                </xsd:complexType>
            </xsd:element>
            <xsd:element name="extensions" minOccurs="0">
                <xsd:complexType>
                    <xsd:sequence>
                        <xsd:any namespace="##any" maxOccurs="unbounded"/>
                    </xsd:sequence>
                </xsd:complexType>
            </xsd:element>
        </xsd:sequence>
    </xsd:complexType>
</xsd:element>

</xsd:schema>
```

5.2 Stop Application Session

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

 <xsd:annotation>
  <xsd:documentation>Ecma-Stop-Application-Session</xsd:documentation>
 </xsd:annotation>

 <xsd:element name="StopApplicationSession">
  <xsd:complexType>
   <xsd:sequence>
    <xsd:element name="sessionID" type="xsd:string"/>
    <xsd:element name="sessionEndReason">
     <xsd:complexType>
      <xsd:choice>
       <xsd:element name="definedEndReason">
        <xsd:simpleType>
         <xsd:restriction base="xsd:string">
          <xsd:enumeration value="normal"/>
         </xsd:restriction>
        </xsd:simpleType>
       </xsd:element>
       <xsd:element name="appEndReason" type="xsd:string"/>
      </xsd:choice>
     </xsd:complexType>
    </xsd:element>
    <xsd:element name="extensions" minOccurs="0">
     <xsd:complexType>
      <xsd:sequence>
       <xsd:any namespace="##any" maxOccurs="unbounded"/>
      </xsd:sequence>
     </xsd:complexType>
    </xsd:element>
   </xsd:sequence>
  </xsd:complexType>
 </xsd:element>

 <xsd:element name="StopApplicationSessionPosResponse">
  <xsd:complexType>
   <xsd:sequence>
    <xsd:element name="extensions" minOccurs="0">
     <xsd:complexType>
      <xsd:sequence>
       <xsd:any namespace="##any" maxOccurs="unbounded"/>
      </xsd:sequence>
     </xsd:complexType>
    </xsd:element>
   </xsd:sequence>
  </xsd:complexType>
 </xsd:element>

 <xsd:element name="StopApplicationSessionNegResponse">
  <xsd:complexType>
   <xsd:sequence>
    <xsd:element name="errorCode">
     <xsd:complexType>
      <xsd:choice>
```

```
            <xsd:element name="definedError">
              <xsd:simpleType>
                <xsd:restriction base="xsd:string">
                  <xsd:enumeration value="invalidSessionID"/>
                </xsd:restriction>
              </xsd:simpleType>
            </xsd:element>
            <xsd:element name="applError" type="xsd:string"/>
          </xsd:choice>
        </xsd:complexType>
      </xsd:element>
      <xsd:element name="extensions" minOccurs="0">
        <xsd:complexType>
          <xsd:sequence>
            <xsd:any namespace="##any" maxOccurs="unbounded"/>
          </xsd:sequence>
        </xsd:complexType>
      </xsd:element>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>

</xsd:schema>
```

5.3 Reset Application Session Timer

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

  <xsd:annotation>
    <xsd:documentation>Ecma-Reset-Application-Session-Timer</xsd:documentation>
  </xsd:annotation>

  <xsd:element name="ResetApplicationSessionTimer">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="sessionID" type="xsd:string"/>
        <xsd:element name="requestedSessionDuration" type="xsd:integer" minOccurs="0"/>
        <xsd:element name="extensions" minOccurs="0">
          <xsd:complexType>
            <xsd:sequence>
              <xsd:any namespace="##any" maxOccurs="unbounded"/>
            </xsd:sequence>
          </xsd:complexType>
        </xsd:element>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="ResetApplicationSessionTimerPosResponse">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="actualSessionDuration" type="xsd:integer"/>
        <xsd:element name="extensions" minOccurs="0">
          <xsd:complexType>
            <xsd:sequence>
              <xsd:any namespace="##any" maxOccurs="unbounded"/>
```

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

<xsd:element name="ResetApplicationSessionTimerNegResponse">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="errorCode">
        <xsd:complexType>
          <xsd:choice>
            <xsd:element name="definedError">
              <xsd:simpleType>
                <xsd:restriction base="xsd:string">
                  <xsd:enumeration value="invalidSessionID"/>
                  <xsd:enumeration value="serverCannotResetSessionTimer"/>
                </xsd:restriction>
              </xsd:simpleType>
            </xsd:element>
            <xsd:element name="applError" type="xsd:string"/>
          </xsd:choice>
        </xsd:complexType>
      </xsd:element>
      <xsd:element name="extensions" minOccurs="0">
        <xsd:complexType>
          <xsd:sequence>
            <xsd:any namespace="##any" maxOccurs="unbounded"/>
          </xsd:sequence>
        </xsd:complexType>
      </xsd:element>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>

</xsd:schema>
```

5.4 Application Session Terminated

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

  <xsd:annotation>
    <xsd:documentation>Ecma-Application-Session-Terminated</xsd:documentation>
  </xsd:annotation>

  <xsd:element name="ApplicationSessionTerminated">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="sessionID" type="xsd:string"/>
        <xsd:element name="sessionTermReason">
          <xsd:complexType>
            <xsd:choice>
              <xsd:element name="definedTermReason">
                <xsd:simpleType>
                  <xsd:restriction base="xsd:string">
```

```
                <xsd:enumeration value="resourceLimitation"/>
                <xsd:enumeration value="sessionTimerExpired"/>
              </xsd:restriction>
            </xsd:simpleType>
          </xsd:element>
          <xsd:element name="serverTermReason" type="xsd:string"/>
        </xsd:choice>
      </xsd:complexType>
    </xsd:element>
    <xsd:element name="extensions" minOccurs="0">
      <xsd:complexType>
        <xsd:sequence>
          <xsd:any namespace="##any" maxOccurs="unbounded"/>
        </xsd:sequence>
      </xsd:complexType>
    </xsd:element>
  </xsd:sequence>
 </xsd:complexType>
</xsd:element>

</xsd:schema>
```

Annex A
(informative)

Examples

This Annex illustrates the use of the Ecma Application Session Services for ECMA-323.

A.1 Starting an Application Session

This section illustrates examples of how an application session can be started.

A.1.1 Basic Example

In this example, an application uses the Start Application Session service to start an application session.

In this example:

- The application name is “CSTA Example App”.
- The requested application protocol is: http://www.ecma-international.org/standards/ecma-323/csta/ed3.
- The requested session duration is specified at 300 seconds. If the application does not send a Reset Application Session Timer within 300 seconds, the application session will be terminated.

```
<?xml version="1.0" encoding="UTF-8"?>
<StartApplicationSession xmlns="http://www.ecma-international.org/standards/ecma-
354/appl_session" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <applicationInfo>
    <applicationID>CSTA Example App</applicationID>
  </applicationInfo>
  <requestedProtocolVersions>
    <protocolVersion>http://www.ecma-international.org/standards/ecma-
323/csta/ed3</protocolVersion>
  </requestedProtocolVersions>
  <requestedSessionDuration>300</requestedSessionDuration>
</StartApplicationSession>
```

The following example shows the positive response message that the application receives based upon the previous request message.

The response includes:

- A sessionID used to identify the session.
- The actual application protocol version used for this session. Since the application specified only one application protocol in the request, it is the only possible application protocol that the server could choose.
- The actual session duration used by the server. This is the same as the value requested by the application.

```
<?xml version="1.0" encoding="UTF-8"?>
<StartApplicationSessionPosResponse
xmlns="http://www.ecma-international.org/standards/ecma-354/appl_session"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
```

```
  <sessionID>AEF12111212</sessionID>
  <actualProtocolVersion>http://www.ecma-international.org/standards/ecma-323/csta/ed3
  </actualProtocolVersion>
  <actualSessionDuration>300</actualSessionDuration>
</StartApplicationSessionPosResponse>
```

A.1.2 Application Protocol Negotiation

This example illustrates how an application can specify several versions of CSTA protocols that it can support. The response message indicates the highest priority version that the server supports which is the application protocol used for the application session.

In this example:

- The application includes the namespace associated with the three ECMA-323 protocol versions. The first protocol version in the list is the highest priority from the application perspective. The server should choose the highest priority version from the list of requestedProtocolVersions that it can support.

```
<?xml version="1.0" encoding="UTF-8"?>
<StartApplicationSession xmlns="http://www.ecma-international.org/standards/ecma-
354/appl_session" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <applicationInfo>
    <applicationID>CSTA Example App</applicationID>
  </applicationInfo>
  <requestedProtocolVersions>
    <protocolVersion>http://www.ecma-international.org/standards/ecma-323/csta/ed3
</protocolVersion>
    <protocolVersion>http://www.ecma-international.org/standards/ecma-323/csta/ed2
</protocolVersion>
    <protocolVersion>http://www.ecma.ch/standards/ecma-323/csta
</protocolVersion>
  </requestedProtocolVersions>
  <requestedSessionDuration>300</requestedSessionDuration>
</StartApplicationSession>
```

The following example shows the response message that the application receives based upon the previous request message.

The response includes:

- The actual application protocol version used for this session. Of the three requested application protocols in the request, the highest priority application protocol that the server supports is the one associated with the ECMA-323 2nd Edition. This is the application protocol chosen for this application session and it is returned in the response as shown below.

```
<?xml version="1.0" encoding="UTF-8"?>
<StartApplicationSessionPosResponse
xmlns="http://www.ecma-international.org/standards/ecma-354/appl_session"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <sessionID>AEF12111212</sessionID>
  <actualProtocolVersion>http://www.ecma-international.org/standards/ecma-323/csta/ed2
</actualProtocolVersion>
  <actualSessionDuration>300</actualSessionDuration>
</StartApplicationSessionPosResponse>
```

A.1.3 Vendor Specific Application Information

This example illustrates how an application can provide additional application information in the service request.

In this example:

- The application provides two additional vendor specific elements needed to identify this particular application: a userID and a password. These two elements are defined in another schema namespace referenced as vendorData in the example below.

```
<?xml version="1.0" encoding="UTF-8"?>
<StartApplicationSession xmlns="http://www.ecma-international.org/standards/ecma-
354/appl_session" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:vendorData="http://www.vendorname/product/release">
  <applicationInfo>
    <applicationID>CSTA App 22343</applicationID>
    <applicationSpecificInfo>
    <vendorData:userID>userName</vendorData:userID>
    <vendorData:password>defaultPassword</vendorData:password>
    </applicationSpecificInfo>
  </applicationInfo>
  <requestedProtocolVersions>
    <protocolVersion>http://www.ecma-international.org/standards/ecma-
323/csta/ed3</protocolVersion>
  </requestedProtocolVersions>
  <requestedSessionDuration>300</requestedSessionDuration>
</StartApplicationSession>
```

A.2 Stopping an Application Session

In this example, an application uses the Stop Application Session service to request to stop an application session.

```
<?xml version="1.0" encoding="UTF-8"?>
<StopApplicationSession xmlns="http://www.ecma-international.org/standards/ecma-
354/appl_session" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <sessionID>AEF12111212</sessionID>
  <sessionEndReason>
    <definedEndReason>normal</definedEndReason>
  </sessionEndReason>
</StopApplicationSession>
```

The following example shows the response message that the application receives based upon the previous request message.

```
<?xml version="1.0" encoding="UTF-8"?>
<StopApplicationSessionPosResponse
xmlns="http://www.ecma-international.org/standards/ecma-354/appl_session"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"/>
```

A.3 Resetting an Application Session Timer

This section illustrates how an application session timer is reset by an application.

```
<?xml version="1.0" encoding="UTF-8"?>
<ResetApplicationSessionTimer
xmlns="http://www.ecma-international.org/standards/ecma-354/appl_session"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <sessionID>AEF12111212</sessionID>
  <requestedSessionDuration>500</requestedSessionDuration>
</ResetApplicationSessionTimer>
```

The following example shows the response message that the application receives based upon the previous request message.

```
<?xml version="1.0" encoding="UTF-8"?>
<ResetApplicationSessionTimerPosResponse
xmlns="http://www.ecma-international.org/standards/ecma-354/appl_session"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <actualSessionDuration>500</actualSessionDuration>
</ResetApplicationSessionTimerPosResponse>
```

A.4 Application Session Terminated

This section illustrates how a server indicates that application session has been terminated. The server indicates that the reason the application session was terminated was because the session timer expired.

```
<?xml version="1.0" encoding="UTF-8"?>
<ApplicationSessionTerminated
xmlns="http://www.ecma-international.org/standards/ecma-354/appl_session"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <sessionID>AEF12111212</sessionID>
  <sessionTermReason>
    <definedTermReason>sessionTimerExpired</definedTermReason>
  </sessionTermReason>
</ApplicationSessionTerminated>
```

Annex B
(informative)

CSTA Usage Notes

This Annex lists considerations when using this International Standard with CSTA protocols such as ECMA-323.

1) For CSTA protocols, the information associated with an application session (i.e. application context) consists of CSTA monitors and CSTA registrations established within the application session. When the application session is stopped, these CSTA monitors and registrations are released.

2) For the CSTA Protocols, the CSTA System Status services can be used to provide status related to an application session state. For example:

   a) The CSTA System Status (“enabled”) service clears monitors and registrations associated with an application session without terminating the application session.

   b) The CSTA System Status (“messagesLost”) service indicates that messages associated with an application session may have been lost.

   c) The CSTA Query System Status service can be used to obtain the system status related to the application session.

3) The sessionID may be transported as part of the header information associated with application messages. When SOAP is used, the sessionID should be included in the SOAP header for each application message exchanged over the application session. The recipient (application or server) of the application message should verify that the sessionID is valid otherwise it should ignore the message.

   a) For Web Services (e.g., CSTA XML with SOAP over HTTP), the sessionDuration expiry should be used to detect when an (e.g., CSTA Web Client) application is no longer present. The (e.g., CSTA Web) Server should tear down the application session after the time specified by sessionDuration (unless reset by the Reset Application Session Timer service).

Bibliography

[1] ECMA-269, *Information technology — Telecommunications and information exchange between systems — Services for Computer Supported Telecommunications Applications (CSTA) Phase III*, 6th edition, June 2004 (ISO/IEC 18051)

[2] ECMA-323, *Information technology — Telecommunications and information exchange between systems — XML Protocol for Computer Supported Telecommunications Applications (CSTA) Phase III*, 3rd edition, June 2004 (ISO/IEC 18056)

[3] ISO/IEC 8649:1996, *Information technology — Open Systems Interconnection — Service definition for the Association Control Service Element*

ICS 35.040

Price based on 19 pages