CLUE Working Group
Internet-Draft
Intended status: Informational
Expires: September 9, 2013

R. Presta
S P. Romano
University of Napoli

March 8, 2013

An XML Schema for the CLUE data model
draft-presta-clue-data-model-schema-03

Abstract

This document provides an XML schema file for the definition of CLUE data model types.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at http://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on September 9, 2013.

Copyright Notice

Copyright (c) 2013 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (http://trustee.ietf.org/license-info) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction 4
2. Terminology 4
3. XML Schema 4
4. <mediaCaptures> 13
5. <encodings> 13
6. <encodingGroups> 13
7. <captureScenes> 14
8. <simultaneousSets> 14
9. <captureEncodings> 14
10. <mediaCapture> 14
10.1. <capturedMedia> 15
10.2. <captureSceneIDREF> 15
10.3. <encGroupIDREF> 16
10.4. <spatialInformation> 16
10.4.1. <capturePoint> 17
10.4.2. <captureArea> 18
10.5. <nonSpatiallyDefinible> 18
10.6. <description> 19
10.7. <priority> 19
10.8. <lang> 19
10.9. <content> 19
10.10. <switched> 20
10.11. <dynamic> 20
10.12. <composed> 20
10.13. <maxCaptureEncodings> 20
10.14. <relatedTo> 20
10.15. captureID attribute 21
11. Audio captures 21
11.1. <audioChannelFormat> 21
11.2. <micPattern> 22
12. Video captures 22
12.1. <nativeAspectRatio> 23
12.2. <embeddedText> 23
13. Text captures 24
14. <captureScene> 24
14.1. <sceneSpace> (was:<sceneArea>) 25
14.2. <sceneEntries> 26
14.3. sceneID attribute 26
14.4. scale attribute 26
15. <sceneEntry> 27
15.1. <switchingPolicies> 27
15.2. <mediaCaptureIDs> 28
15.3. sceneEntryID attribute 29
15.4. mediaType attribute 29
16. <encoding> 29
16.1. <encodingName> 29
16.2. <maxBandwidth> 29
16.3. encodingID attribute 30

17. Audio encodings 30
18. Video encodings 30
18.1. <maxWidth> 31
18.2. <maxHeight> 31
18.3. <maxFrameRate> 31
19. H26X encodings 31
20. <encodingGroup> 32
20.1. <maxGroupBandwidth> 32
20.2. <maxGroupPps> 33
20.3. <encodingIDList> 33
20.4. encodingGroupID attribute 33
21. <simultaneousSet> 33
21.1. <captureIDREF> 34
21.2. <sceneEntryIDREF> 34
22. <captureEncoding> 34
22.1. <mediaCaptureID> 34
22.2. <encodingID> 34
23. <clueInfo> 34
24. Sample XML file 35
25. Diff with unofficial -02 version 44
26. Diff with -02 version 46
27. Informative References 46

1. Introduction

This document provides an XML schema file for the definition of CLUE data model types.

The schema is based on information contained in [I-D.ietf-clue-framework] and also relates to the data model sketched in [I-D.romanow-clue-data-model]. It encodes information and constraints defined in the aforementioned documents in order to provide a formal representation of the concepts therein presented. The schema definition is intended to be modified according to changes applied to the above mentioned CLUE documents.

The document actually represents a strawman proposal aiming at the definition of a coherent structure for all the information associated with the description of a telepresence scenario.

2. Terminology

[TBD] Copy text from the framework document.

3. XML Schema

This section contains the proposed CLUE data model schema definition.

The element and attribute definitions are formal representation of the concepts needed to describe the capabilities of a media provider and the current streams it is transmitting within a telepresence session.

The main groups of information are:

<mediaCaptures>: the list of media captures available (Section 4)

<encodings>: the list of individual encodings (Section 5)

<encodingGroups>: the list of encodings groups (Section 6)

<captureScenes>: the list of capture scenes (Section 7)

<simultaneousSets>: the list of simultaneous capture sets(Section 8)

<captureEncodings>: the list of instantiated capture encodings (Section 9)

All of the above refers to concepts that have been introduced in [I-D.ietf-clue-framework] and [I-D.romanow-clue-data-model] and

further detailed in threads on the mailing list as well as in the following of this document.

```
<?xml version="1.0" encoding="UTF-8" ?>
<xs:schema
   targetNamespace="urn:ietf:params:xml:ns:clue-info"
   xmlns:tns="urn:ietf:params:xml:ns:clue-info"
   xmlns:xs="http://www.w3.org/2001/XMLSchema"
   xmlns="urn:ietf:params:xml:ns:clue-info"
   elementFormDefault="qualified"
   attributeFormDefault="unqualified">

<!-- ELEMENT DEFINITIONS -->
<xs:element name="mediaCaptures" type="mediaCapturesType"/>
<xs:element name="encodings" type="encodingsType"/>
<xs:element name="encodingGroups" type="encodingGroupsType"/>
<xs:element name="captureScenes" type="captureScenesType"/>
<xs:element name="simultaneousSets" type="simultaneousSetsType"/>
<xs:element name="captureEncodings" type="captureEncodingsType"/>

<!-- MEDIA CAPTURES TYPE -->
<!-- envelope of media captures -->
<xs:complexType name="mediaCapturesType">
 <xs:sequence>
   <xs:element name="mediaCapture" type="mediaCaptureType"
   maxOccurs="unbounded"/>
 </xs:sequence>
</xs:complexType>

<!-- DESCRIPTION element -->
<xs:element name="description">
 <xs:complexType>
  <xs:simpleContent>
   <xs:extension base="xs:string">
     <xs:attribute name="lang" type="xs:language"/>
   </xs:extension>
  </xs:simpleContent>
 </xs:complexType>
</xs:element>

<!-- MEDIA CAPTURE TYPE -->
<xs:complexType name="mediaCaptureType" abstract="true">
  <xs:sequence>
    <!-- mandatory fields -->
    <xs:element name="capturedMedia" type="xs:string"/>
    <xs:element name="captureSceneIDREF" type="xs:IDREF"/>
    <xs:element name="encGroupIDREF" type="xs:IDREF"/>
```

```
    <xs:choice>
      <xs:sequence>
        <xs:element name="spatialInformation" type="tns:spatialInformationType"
        maxOccurs="unbounded"/>
      </xs:sequence>
      <xs:element name="nonSpatiallyDefinible" type="xs:boolean" fixed="true"/>
    </xs:choice>
    <!-- optional fields -->
    <xs:element ref="description" minOccurs="0" maxOccurs="unbounded"/>
    <xs:element name="priority" type="xs:integer" minOccurs="0"/>
    <xs:element name="lang" type="xs:language" minOccurs="0"/>
    <xs:element name="content" type="xs:string" minOccurs="0"/>
    <xs:element name="switched" type="xs:boolean" minOccurs="0"/>
    <xs:element name="dynamic" type="xs:boolean" minOccurs="0"/>
    <xs:element name="composed" type="xs:boolean" minOccurs="0"/>
    <xs:element name="maxCaptureEncodings" type="xs:unsignedInt"
    minOccurs="0"/>
    <!-- this is in place of "supplementary info": -->
    <xs:element name="relatedTo" type="xs:IDREF" minOccurs="0"/>
    <xs:any namespace="##other" processContents="lax" minOccurs="0"
    maxOccurs="unbounded"/>
  </xs:sequence>
  <xs:attribute name="captureID" type="xs:ID" use="required"/>
  <xs:anyAttribute namespace="##other" processContents="lax"/>
</xs:complexType>

<!-- SPATIAL INFORMATION TYPE -->
<xs:complexType name="spatialInformationType">
 <xs:sequence>
  <xs:element name="capturePoint" type="capturePointType"/>
  <xs:element name="captureArea" type="captureAreaType"
  minOccurs="0"/>
  <xs:any namespace="##other" processContents="lax" minOccurs="0"
  maxOccurs="unbounded"/>
 </xs:sequence>
 <xs:anyAttribute namespace="##other" processContents="lax"/>
</xs:complexType>

<!-- TEXT CAPTURE TYPE -->
<xs:complexType name="textCaptureType">
 <xs:complexContent>
  <xs:extension base="tns:mediaCaptureType">
  </xs:extension>
 </xs:complexContent>
</xs:complexType>


<!-- AUDIO CAPTURE TYPE -->
```

```
<xs:complexType name="audioCaptureType">
 <xs:complexContent>
  <xs:extension base="tns:mediaCaptureType">
   <xs:sequence>
    <xs:element name="audioChannelFormat" type="audioChannelFormatType"
    minOccurs="0"/>
    <xs:element name="micPattern" type="tns:micPatternType"
    minOccurs="0"/>
   </xs:sequence>
  </xs:extension>
 </xs:complexContent>
</xs:complexType>

<!-- MIC PATTERN TYPE -->
<xs:simpleType name="micPatternType">
 <xs:restriction base="xs:string">
  <xs:enumeration value="uni"/>
  <xs:enumeration value="shotgun"/>
  <xs:enumeration value="omni"/>
  <xs:enumeration value="figure8"/>
  <xs:enumeration value="cardioid"/>
  <xs:enumeration value="hyper-cardioid"/>
 </xs:restriction>
</xs:simpleType>

<!-- AUDIO CHANNEL FORMAT TYPE -->
<xs:simpleType name="audioChannelFormatType">
 <xs:restriction base="xs:string">
  <xs:enumeration value="mono"/>
  <xs:enumeration value="stereo"/>
 </xs:restriction>
</xs:simpleType>

<!-- VIDEO CAPTURE TYPE -->
<xs:complexType name="videoCaptureType">
 <xs:complexContent>
  <xs:extension base="tns:mediaCaptureType">
   <xs:sequence>
    <xs:element name="nativeAspectRatio" type="xs:string"
    minOccurs="0"/>
    <xs:element ref="embeddedText" minOccurs="0"/>
   </xs:sequence>
  </xs:extension>
 </xs:complexContent>
</xs:complexType>

<!-- EMBEDDED TEXT ELEMENT -->
<xs:element name="embeddedText">
```

```
 <xs:complexType>
  <xs:simpleContent>
   <xs:extension base="xs:boolean">
    <xs:attribute name="lang" type="xs:language"/>
   </xs:extension>
  </xs:simpleContent>
 </xs:complexType>
</xs:element>

<!-- CAPTURE SCENES TYPE -->
<!-- envelope of capture scenes -->
<xs:complexType name="captureScenesType">
 <xs:sequence>
  <xs:element name="captureScene" type="captureSceneType"
  maxOccurs="unbounded"/>
 </xs:sequence>
</xs:complexType>

<!-- CAPTURE SCENE TYPE -->
<xs:complexType name="captureSceneType">
 <xs:sequence>
  <xs:element ref="description" minOccurs="0" maxOccurs="unbounded"/>
  <xs:element name="sceneSpace" type="captureSpaceType" minOccurs="0"/>
  <xs:element name="sceneEntries" type="sceneEntriesType"/>
  <xs:any namespace="##other" processContents="lax" minOccurs="0"
  maxOccurs="unbounded"/>
 </xs:sequence>
 <xs:attribute name="sceneID" type="xs:ID" use="required"/>
 <xs:attribute name="scale" type="scaleType" use="required"/>
 <xs:anyAttribute namespace="##other" processContents="lax"/>
</xs:complexType>

<!-- SCALE TYPE -->
<xs:simpleType name="scaleType">
 <xs:restriction base="xs:string">
  <xs:enumeration value="millimeters"/>
  <xs:enumeration value="unknown"/>
  <xs:enumeration value="noscale"/>
 </xs:restriction>
</xs:simpleType>

<!-- CAPTURE AREA TYPE -->
<xs:complexType name="captureAreaType">
 <xs:sequence>
  <xs:element name="bottomLeft" type="pointType"/>
  <xs:element name="bottomRight" type="pointType"/>
  <xs:element name="topLeft" type="pointType"/>
  <xs:element name="topRight" type="pointType"/>
```

```
 </xs:sequence>
</xs:complexType>

<!-- CAPTURE SPACE TYPE -->
<xs:complexType name="captureSpaceType">
 <xs:sequence>
  <xs:element name="bottomLeftFront" type="pointType"/>
  <xs:element name="bottomRightFront" type="pointType"/>
  <xs:element name="topLeftFront" type="pointType"/>
  <xs:element name="topRightFront" type="pointType"/>
  <xs:element name="bottomLeftBack" type="pointType"/>
  <xs:element name="bottomRightBack" type="pointType"/>
  <xs:element name="topLeftBack" type="pointType"/>
  <xs:element name="topRightBack" type="pointType"/>
 </xs:sequence>
</xs:complexType>

<!-- POINT TYPE -->
<xs:complexType name="pointType">
 <xs:sequence>
  <xs:element name="x" type="xs:decimal"/>
  <xs:element name="y" type="xs:decimal"/>
  <xs:element name="z" type="xs:decimal"/>
 </xs:sequence>
</xs:complexType>

<!-- CAPTURE POINT TYPE -->
<xs:complexType name="capturePointType">
 <xs:complexContent>
  <xs:extension base="pointType">
   <xs:sequence>
    <xs:element name="lineOfCapturePoint" type="tns:pointType"
    minOccurs="0"/>
   </xs:sequence>
   <xs:attribute name="pointID" type="xs:ID"/>
  </xs:extension>
 </xs:complexContent>
</xs:complexType>


<!-- SCENE ENTRIES TYPE -->
<!-- envelope of scene entries of a capture scene -->
<xs:complexType name="sceneEntriesType">
 <xs:sequence>
  <xs:element name="sceneEntry" type="sceneEntryType"
  maxOccurs="unbounded"/>
 </xs:sequence>
</xs:complexType>
```

```
<!-- SCENE ENTRY TYPE -->
<xs:complexType name="sceneEntryType">
 <xs:sequence>
  <xs:element ref="description" minOccurs="0" maxOccurs="unbounded"/>
  <xs:element name="switchingPolicies" type="switchingPoliciesType"
  minOccurs="0"/>
  <xs:element name="mediaCaptureIDs" type="captureIDListType"/>
 </xs:sequence>
 <xs:attribute name="sceneEntryID" type="xs:ID" use="required"/>
 <xs:attribute name="mediaType" type="xs:string" use="required"/>
</xs:complexType>

<!-- SWITCHING POLICIES TYPE -->
<xs:complexType name="switchingPoliciesType">
 <xs:sequence>
  <xs:element name="siteSwitching" type="xs:boolean" minOccurs="0"/>
  <xs:element name="segmentSwitching" type="xs:boolean"
  minOccurs="0"/>
 </xs:sequence>
</xs:complexType>

<!-- CAPTURE ID LIST TYPE -->
<xs:complexType name="captureIDListType">
 <xs:sequence>
  <xs:element name="captureIDREF" type="xs:IDREF"
  maxOccurs="unbounded"/>
 </xs:sequence>
</xs:complexType>

<!-- ENCODINGS TYPE -->
<xs:complexType name="encodingsType">
 <xs:sequence>
  <xs:element name="encoding" type="encodingType"
  maxOccurs="unbounded"/>
 </xs:sequence>
</xs:complexType>

<!-- ENCODING TYPE -->
<xs:complexType name="encodingType" abstract="true">
 <xs:sequence>
  <xs:element name="encodingName" type="xs:string"/>
  <xs:element name="maxBandwidth" type="xs:integer"/>
  <xs:any namespace="##other" processContents="lax" minOccurs="0"
  maxOccurs="unbounded"/>
 </xs:sequence>
 <xs:attribute name="encodingID" type="xs:ID" use="required"/>
 <xs:anyAttribute namespace="##any" processContents="lax"/>
</xs:complexType>
```

```
<!-- AUDIO ENCODING TYPE -->
<xs:complexType name="audioEncodingType">
 <xs:complexContent>
  <xs:extension base="tns:encodingType">
   <xs:sequence>
    <xs:element name="encodedMedia" type="xs:string" fixed="audio"
    minOccurs="0"/>
   </xs:sequence>
  </xs:extension>
 </xs:complexContent>
</xs:complexType>

<!-- VIDEO ENCODING TYPE -->
<xs:complexType name="videoEncodingType">
 <xs:complexContent>
  <xs:extension base="tns:encodingType">
   <xs:sequence>
    <xs:element name="encodedMedia" type="xs:string" fixed="video"
    minOccurs="0"/>
    <xs:element name="maxWidth" type="xs:integer" minOccurs="0"/>
    <xs:element name="maxHeight" type="xs:integer" minOccurs="0"/>
    <xs:element name="maxFrameRate" type="xs:integer" minOccurs="0"/>
   </xs:sequence>
  </xs:extension>
 </xs:complexContent>
</xs:complexType>

<!-- H26X ENCODING TYPE -->
<xs:complexType name="h26XEncodingType">
 <xs:complexContent>
  <xs:extension base="tns:videoEncodingType">
   <xs:sequence>
    <!-- max number of pixels to be processed per second -->
    <xs:element name="maxH26Xpps" type="xs:integer"
    minOccurs="0"/>
   </xs:sequence>
  </xs:extension>
 </xs:complexContent>
</xs:complexType>

<!-- ENCODING GROUPS TYPE -->
<xs:complexType name="encodingGroupsType">
 <xs:sequence>
  <xs:element name="encodingGroup" type="tns:encodingGroupType"
  maxOccurs="unbounded"/>
 </xs:sequence>
</xs:complexType>
```

```
<!-- ENCODING GROUP TYPE -->
<xs:complexType name="encodingGroupType">
 <xs:sequence>
  <xs:element name="maxGroupBandwidth" type="xs:integer"/>
  <xs:element name="maxGroupPps" type="xs:integer"
  minOccurs="0"/>
  <xs:element name="encodingIDList" type="encodingIDListType"/>
  <xs:any namespace="##other" processContents="lax" minOccurs="0"
  maxOccurs="unbounded"/>
 </xs:sequence>
 <xs:attribute name="encodingGroupID" type="xs:ID" use="required"/>
 <xs:anyAttribute namespace="##any" processContents="lax"/>
</xs:complexType>

<!-- ENCODING ID LIST TYPE -->
<xs:complexType name="encodingIDListType">
 <xs:sequence>
  <xs:element name="encIDREF" type="xs:IDREF" maxOccurs="unbounded"/>
 </xs:sequence>
</xs:complexType>

<!-- SIMULTANEOUS SETS TYPE -->
<xs:complexType name="simultaneousSetsType">
 <xs:sequence>
  <xs:element name="simultaneousSet" type="simultaneousSetType"
  maxOccurs="unbounded"/>
 </xs:sequence>
</xs:complexType>

<!-- SIMULTANEOUS SET TYPE -->
<xs:complexType name="simultaneousSetType">
 <xs:sequence>
   <xs:element name="captureIDREF" type="xs:IDREF"
   minOccurs="0" maxOccurs="unbounded"/>
   <xs:element name="sceneEntryIDREF" type="xs:IDREF"
   minOccurs="0" maxOccurs="unbounded"/>
 </xs:sequence>
</xs:complexType>

<!-- CAPTURE ENCODING TYPE -->
<xs:complexType name="captureEncodingType">
 <xs:sequence>
  <xs:element name="mediaCaptureID" type="xs:string"/>
  <xs:element name="encodingID" type="xs:string"/>
 </xs:sequence>
</xs:complexType>

<!-- CAPTURE ENCODINGS TYPE -->
```

```
<xs:complexType name="captureEncodingsType">
 <xs:sequence>
  <xs:element name="captureEncoding" type="captureEncodingType"
  maxOccurs="unbounded"/>
 </xs:sequence>
</xs:complexType>

<!-- CLUE INFO ELEMENT -->
<!-- the <clueInfo> envelope can be seen
     as the ancestor of an <advertisement> envelope -->
<xs:element name="clueInfo" type="clueInfoType"/>

<!-- CLUE INFO TYPE -->
<xs:complexType name="clueInfoType">
  <xs:sequence>
   <xs:element ref="mediaCaptures"/>
   <xs:element ref="encodings"/>
   <xs:element ref="encodingGroups"/>
   <xs:element ref="captureScenes"/>
   <xs:element ref="simultaneousSets"/>
   <xs:any namespace="##other" processContents="lax" minOccurs="0"
   maxOccurs="unbounded"/>
  </xs:sequence>
  <xs:attribute name="clueInfoID" type="xs:ID" use="required"/>
  <xs:anyAttribute namespace="##other" processContents="lax"/>
</xs:complexType>

</xs:schema>
```

Following sections describe the XML schema in more detail.

4. <mediaCaptures>

<mediaCaptures> represents the list of one ore more media captures available on the media provider's side. Each media capture is represented by a <mediaCapture> element (Section 10).

5. <encodings>

<encodings> represents the list of individual encodings available on the media provider's side. Each individual encoding is represented by an <encoding> element (Section 16).

6. <encodingGroups>

<encodingGroups> represents the list of the encoding groups organized on the media provider's side. Each encoding group is represented by

a <encodingGroup> element (Section 20).

7. <captureScenes>

<captureScenes> represents the list of the capture scenes organized on the media provider's side. Each capture scene is represented by a <captureScene> element. (Section 14).

8. <simultaneousSets>

<simultaneousSets> contains the simultaneous sets indicated by the media provider. Each simultaneous set is represented by a <simultaneousSet> element. (Section 21).

9. <captureEncodings>

<captureEncodings> is a list of capture encodings. It can represents the list of the desired capture encodings indicated by the media consumer or the list of instantiated captures on the provider's side. Each capture encoding is represented by a <captureEncoding> element. (Section 22).

10. <mediaCapture>

According to the CLUE framework, a media capture is the fundamental representation of a media flow that is available on the provider's side. Media captures are characterized with a set of features that are independent from the specific type of medium, and with a set of feature that are media-specific. We design the media capture type as an abstract type, providing all the features that can be common to all media types. Media-specific captures, such as video captures, audio captures and others, are specialization of that media capture type, as in a typical generalization-specialization hierarchy.

The following is the XML Schema definition of the media capture type:

```
<!-- MEDIA CAPTURE TYPE -->
<xs:complexType name="mediaCaptureType" abstract="true">
  <xs:sequence>
    <!-- mandatory fields -->
    <xs:element name="capturedMedia" type="xs:string"/>
    <xs:element name="captureSceneIDREF" type="xs:IDREF"/>
    <xs:element name="encGroupIDREF" type="xs:IDREF"/>
    <xs:choice>
      <xs:sequence>
        <xs:element name="spatialInformation" type="tns:spatialInformationType"
        maxOccurs="unbounded"/>
      </xs:sequence>
      <xs:element name="nonSpatiallyDefinible" type="xs:boolean" fixed="true"/>
    </xs:choice>
    <!-- optional fields -->
    <xs:element ref="description" minOccurs="0" maxOccurs="unbounded"/>
    <xs:element name="priority" type="xs:integer" minOccurs="0"/>
    <xs:element name="lang" type="xs:language" minOccurs="0"/>
    <xs:element name="content" type="xs:string" minOccurs="0"/>
    <xs:element name="switched" type="xs:boolean" minOccurs="0"/>
    <xs:element name="dynamic" type="xs:boolean" minOccurs="0"/>
    <xs:element name="composed" type="xs:boolean" minOccurs="0"/>
    <xs:element name="maxCaptureEncodings" type="xs:unsignedInt"
    minOccurs="0"/>
    <!-- this is in place of "supplementary info": -->
    <xs:element name="relatedTo" type="xs:IDREF" minOccurs="0"/>
    <xs:any namespace="##other" processContents="lax" minOccurs="0"
    maxOccurs="unbounded"/>
  </xs:sequence>
  <xs:attribute name="captureID" type="xs:ID" use="required"/>
  <xs:anyAttribute namespace="##other" processContents="lax"/>
</xs:complexType>
```

10.1. <capturedMedia>

<capturedMedia> is a mandatory field specifying the media type of the capture ("audio", "video", "text",...).

10.2. <captureSceneIDREF>

<captureSceneIDREF> is a mandatory field containing the identifier of the capture scene the media capture belongs to. Indeed, each media capture must be associated with one and only capture scene. When a media capture is spatially definible, some spatial information is provided along with it in the form of point coordinates (see Section 10.4). Such coordinates refers to the space of coordinates defined for the capture scene containing the capture.

10.3. <encGroupIDREF>

<encGroupIDREF> is a mandatory field containing the identifier of the encoding group the media capture is associated with.

10.4. <spatialInformation>

Media captures are divided into two categories: non spatially definible captures and spatially definible captures.

Non spatially definible captures are those that do not capture parts of the telepresence room. Capture of this case are for example those related to registrations, text captures, DVDs, registered presentation, or external streams, that are played in the telepresence room and transmitted to remote sites.

Spatially definible captures are those that capture part of the telepresence room. The captured part of the telepresence room is described by means of the <spatialInformation> element.

This is the definition of the spatial information type:

```
<!-- SPATIAL INFORMATION TYPE -->
<xs:complexType name="spatialInformationType">
 <xs:sequence>
  <xs:element name="capturePoint" type="capturePointType"/>
  <xs:element name="captureArea" type="captureAreaType"
  minOccurs="0"/>
  <xs:any namespace="##other" processContents="lax"
  minOccurs="0" maxOccurs="unbounded"/>
 </xs:sequence>
 <xs:anyAttribute namespace="##other"
 processContents="lax"/>
</xs:complexType>
```

The <capturePoint> contains the coordinates of the capture device that is taking the capture, as well as, optionally, the pointing direction (see Section 10.4.1). It is a mandatory field when the media capture is spatially definible, independently from the media type.

The <captureArea> is an optional field containing four points defining the captured area represented by the capture (see Section 10.4.2).

10.4.1. <capturePoint>

The <capturePoint> element is used to represent the position and the line of capture of a capture device. The XML Schema definition of the <capturePoint> element type is the following:

```
<!-- CAPTURE POINT TYPE -->
<xs:complexType name="capturePointType">
 <xs:complexContent>
  <xs:extension base="pointType">
   <xs:sequence>
    <xs:element name="lineOfCapturePoint"
    type="tns:pointType"
    minOccurs="0"/>
   </xs:sequence>
   <xs:attribute name="pointID" type="xs:ID"/>
  </xs:extension>
 </xs:complexContent>
</xs:complexType>

<!-- POINT TYPE -->
<xs:complexType name="pointType">
 <xs:sequence>
  <xs:element name="x" type="xs:decimal"/>
  <xs:element name="y" type="xs:decimal"/>
  <xs:element name="z" type="xs:decimal"/>
 </xs:sequence>
</xs:complexType>
```

The point type contains three spatial coordinates ("x","y","z") representing a point in the space associated with a certain capture scene.

The capture point type extends the point type, i.e., it is represented by three coordinates identifying the position of the capture device, but can add further information. Such further information is conveyed by the <lineOfCapturePoint>, which is another point-type element representing the "point on line of capture", that gives the pointing direction of the capture device.

If the point of capture is not specified, it means the consumer should not assume anything about the spatial location of the capturing device.

The coordinates of the point on line of capture MUST NOT be identical

to the capture point coordinates. If the point on line of capture is not specified, no assumptions are made about the axis of the capturing device.

10.4.2. <captureArea>

<captureArea> is an optional element that can be contained within the spatial information associated with a media capture. It represents the spatial area captured by the media capture.

The XML representation of that area is provided through a set of four point-type element, <bottomLeft>, <bottomRight>, <topLeft>, and <topRight>, as it can be seen from the following definition:

```
<!-- CAPTURE AREA TYPE -->
<xs:complexType name="captureAreaType">
 <xs:sequence>
  <xs:element name="bottomLeft" type="pointType"/>
  <xs:element name="bottomRight" type="pointType"/>
  <xs:element name="topLeft" type="pointType"/>
  <xs:element name="topRight" type="pointType"/>
 </xs:sequence>
</xs:complexType>
```

<bottomLeft>, <bottomRight>, <topLeft>, and <topRight> should be co-planar.

For a switched capture that switches between different sections within a larger area, the area of capture should use coordinates for the larger potential area.

By comparing the capture area of different media captures within the same capture scene, a consumer can determine the spatial relationships between them and render them correctly. If the area of capture is not specified, it means the Media Capture is not spatially related to any other media capture.

10.5. <nonSpatiallyDefinible>

When media captures are non spatially definible, they are marked with the boolean <nonSpatiallyDefinible> element set to "true".

10.6. <description>

<description> is used to provide optionally human-readable textual information. It is used to describe media captures, capture scenes and capture scene entries. A media capture can be described by using multiple <description> elements, each one providing information in a different language. Indeed, the <description> element definition is the following:

```
<!-- DESCRIPTION element -->
<xs:element name="description">
 <xs:complexType>
  <xs:simpleContent>
   <xs:extension base="xs:string">
     <xs:attribute name="lang" type="xs:language"/>
   </xs:extension>
  </xs:simpleContent>
 </xs:complexType>
</xs:element>
```

As it can be seen, <description> is a string element with an attribute ("lang") indicating the language used in the textual description.

10.7. <priority>

<priority> ([I-D.groves-clue-capture-attr]) is an optional integer field indicating the importance of a media capture according to the media provider's perspective. It can be used on the receiver's side to automatically identify the most "important" contribution available from the media provider.

[edt note: no final consensus has been reached on the adoption of such media capture attribute.]

10.8. <lang>

<lang> is an optional element containing the language used in the capture, if any. The purpose of the element could match the one of the "language" attribute proposed in [I-D.groves-clue-capture-attr].

10.9. <content>

<content> is an optional string element. It contains enumerated values describing the "role" of the media capture according to what

is envisionend in [RFC4796] ("slides", "speaker", "sl", "main", "alt"). The values for this attribute are the same as the mediacnt values for the content attribute in [RFC4796]. This attribute can list multiple values, for example "main, speaker".

[edt note: a better XML Schema definition for that element will soon be defined.]

10.10. <switched>

<switched> is a boolean element which indicates whether or not the media capture represents the most appropriate subset of a "whole". What is "most appropriate" is up to the provider and could be the active speaker, a lecturer or a VIP.

[edt note: :(]

10.11. <dynamic>

<dynamic> is an optional boolean element indicating wheter or not the capture device originating the capture moves during the telepresence session. That optional boolean element has the same purpose of the dynamic attribute proposed in [I-D.groves-clue-capture-attr].

[edt note: There isn't yet final consensus about that element.]

10.12. <composed>

<composed> is an optional boolean element indicating wheter or not the media capture is a mix (audio) or composition (video) of streams. This attribute is useful for a media consumer for example to avoid nesting a composed video capture into another composed capture or rendering.

10.13. <maxCaptureEncodings>

The optional <maxCaptureEncodings> contains an unsigned integer indicating the maximum number of capture encodings that can be simultaneously active for the media capture. If absent, this parameter defaults to 1. The minimum value for this attribute is 1. The number of simultaneous capture encodings is also limited by the restrictions of the encoding group the media capture refers to my means of the <encGroupIDREF> element.

10.14. <relatedTo>

The optional <relatedTo> element contains the value of the ID attribute of the media capture it refers to. The media capture

marked with a <relatedTo> element can be for example the translation of a main media capture in a different language. The <relatedTo> element could be interpreted the same manner of the supplementary information attribute proposed in [I-D.groves-clue-capture-attr] and further discussed in http://www.ietf.org/mail-archive/web/clue/current/msg02238.html.

[edt note: There isn't yet final consensus about that element.]

10.15. captureID attribute

The "captureID" attribute is a mandatory field containing the identifier of the media capture.

11. Audio captures

Audio captures inherit all the features of a generic media capture and present further audio-specific characteristics. The XML Schema definition of the audio capture type is reported below:

```
<!-- AUDIO CAPTURE TYPE -->
<xs:complexType name="audioCaptureType">
 <xs:complexContent>
  <xs:extension base="tns:mediaCaptureType">
   <xs:sequence>
    <xs:element name="audioChannelFormat" type="audioChannelFormatType"
    minOccurs="0"/>
    <xs:element name="micPattern" type="tns:micPatternType"
    minOccurs="0"/>
   </xs:sequence>
  </xs:extension>
 </xs:complexContent>
</xs:complexType>
```

Audio-specific information about the audio capture is contained in <audioChannelFormat> (Section 11.1) and in <micPattern> (Section 11.2).

11.1. <audioChannelFormat>

The optional <audioChannelFormat> element is a field with enumerated values ("mono" and "stereo") which describes the method of encoding used for audio. A value of "mono" means the audio capture has one channel. A value of "stereo" means the audio capture has two audio channels, left and right. A single stereo capture is different from

two mono captures that have a left-right spatial relationship. A stereo capture maps to a single RTP stream, while each mono audio capture maps to a separate RTP stream.

The XML Schema definition of the <audioChannelFormat> element type is provided below:

```
<!-- AUDIO CHANNEL FORMAT TYPE -->
<xs:simpleType name="audioChannelFormatType">
 <xs:restriction base="xs:string">
  <xs:enumeration value="mono"/>
  <xs:enumeration value="stereo"/>
 </xs:restriction>
</xs:simpleType>
```

11.2. <micPattern>

The <micPattern> element is an optional field describing the characteristic of the mic capturing the audio signal. It can contains the enumerated values listed below:

```
<!-- MIC PATTERN TYPE -->
<xs:simpleType name="micPatternType">
 <xs:restriction base="xs:string">
  <xs:enumeration value="uni"/>
  <xs:enumeration value="shotgun"/>
  <xs:enumeration value="omni"/>
  <xs:enumeration value="figure8"/>
  <xs:enumeration value="cardioid"/>
  <xs:enumeration value="hyper-cardioid"/>
 </xs:restriction>
</xs:simpleType>
```

12. Video captures

Video captures, similarly to audio captures, extend the information of a generic media capture with video-specific features, such as <nativeAspectRatio> (Section 12.1) and <embeddedText> (Section 12.2).

The XML Schema representation of the video capture type is provided in the following:

```
<!-- VIDEO CAPTURE TYPE -->
<xs:complexType name="videoCaptureType">
 <xs:complexContent>
  <xs:extension base="tns:mediaCaptureType">
   <xs:sequence>
    <xs:element name="nativeAspectRatio" type="xs:string"
    minOccurs="0"/>
    <xs:element ref="embeddedText" minOccurs="0"/>
   </xs:sequence>
  </xs:extension>
 </xs:complexContent>
</xs:complexType>
```

12.1. <nativeAspectRatio>

If a video capture has a native aspect ratio (for instance, it corresponds to a camera that generates 4:3 video), then it can be supplied as a value of the <nativeAspectRatio> element, in order to help rendering.

12.2. <embeddedText>

The <embeddedText> element is a boolean element indicating that there is text embedded in the video capture. The language used in such embedded textual description is reported in <embeddedText> "lang" attribute.

The XML Schema definition of the <embeddedText> element is:

```
<!-- EMBEDDED TEXT ELEMENT -->
<xs:element name="embeddedText">
 <xs:complexType>
  <xs:simpleContent>
   <xs:extension base="xs:boolean">
    <xs:attribute name="lang" type="xs:language"/>
   </xs:extension>
  </xs:simpleContent>
 </xs:complexType>

</xs:element>
```

The <embeddedText> element could correspond to the embedded-text attribute introduced in [I-D.groves-clue-capture-attr]

[edt note: no final consensus has been reached yet about the adoption of such element]

13. Text captures

Also text captures can be described by extending the generic media capture information, similarly to audio captures and video captures.

The XML Schema representation of the text capture type is currently lacking text-specific information, as it can be seen by looking at the definition below:

```
<!-- TEXT CAPTURE TYPE -->
<xs:complexType name="textCaptureType">
 <xs:complexContent>
  <xs:extension base="tns:mediaCaptureType">
  </xs:extension>
 </xs:complexContent>
</xs:complexType>
```

14. <captureScene>

A media provider organizes the available capture in capture scenes in order to help the receiver both in the rendering and in the selection of the group of captures. Capture scenes are made of capture scene entries, that are set of media captures of the same media type. Each capture scene entry represents an alternative to represent completely a capture scene for a fixed media type.

The XML Schema representation of a <captureScene> element is the following:

```
<!-- CAPTURE SCENE TYPE -->
<xs:complexType name="captureSceneType">
 <xs:sequence>
  <xs:element ref="description" minOccurs="0" maxOccurs="unbounded"/>
  <xs:element name="sceneSpace" type="captureSpaceType" minOccurs="0"/>
  <xs:element name="sceneEntries" type="sceneEntriesType"/>
  <xs:any namespace="##other" processContents="lax" minOccurs="0"
  maxOccurs="unbounded"/>
 </xs:sequence>
 <xs:attribute name="sceneID" type="xs:ID" use="required"/>
 <xs:attribute name="scale" type="scaleType" use="required"/>
```

```
  <xs:anyAttribute namespace="##other" processContents="lax"/>
 </xs:complexType>
```

The <captureScene> element can contain zero or more textual <description> elements, defined as in Section 10.6. Besides <description>, there are two other fields: <sceneSpace> (Section 14.1), describing the coordinate space which the media captures of the capture scene refer to, and <sceneEntries> (Section 14.2), the list of the capture scene entries.

14.1. <sceneSpace> (was:<sceneArea>)

The <sceneSpace> describes a bounding volume for the spatial information provided alongside spatially-definible media capture associated with the considered capture scene. Such volume is described as an arbitrary hexahedrons with eight points (<bottomLeftFront>, <bottomRightFront>, <topLeftFront>, <topRightFront>, <bottomLeftBack>, <bottomRightBack>, <topLeftBack>, and <topRightBack>). The coordinate system is Cartesian X, Y, Z with the origin at a spatial location of the media provider's choosing. The media provider must use the same coordinate system with same scale and origin for all media capture coordinates within the same capture scene.

```
<!-- CAPTURE SPACE TYPE -->
<xs:complexType name="captureSpaceType">
 <xs:sequence>
  <xs:element name="bottomLeftFront" type="pointType"/>
  <xs:element name="bottomRightFront" type="pointType"/>
  <xs:element name="topLeftFront" type="pointType"/>
  <xs:element name="topRightFront" type="pointType"/>
  <xs:element name="bottomLeftBack" type="pointType"/>
  <xs:element name="bottomRightBack" type="pointType"/>
  <xs:element name="topLeftBack" type="pointType"/>
  <xs:element name="topRightBack" type="pointType"/>
 </xs:sequence>
</xs:complexType>
```

[edt note: this is just a place holder, the definition of the bounding volume has to be discussed]

14.2. <sceneEntries>

The <sceneEntries> element is a mandatory field of a capture scene containing the list of scene entries. Each scene entry is represented by a <sceneEntry> element (Section 15).

```
<!-- SCENE ENTRIES TYPE -->
<!-- envelope of scene entries of a capture scene -->
<xs:complexType name="sceneEntriesType">
 <xs:sequence>
  <xs:element name="sceneEntry" type="sceneEntryType"
  maxOccurs="unbounded"/>
 </xs:sequence>
</xs:complexType>
```

14.3. sceneID attribute

The sceneID attribute is a mandatory attribute containing the identifier of the capture scene.

14.4. scale attribute

The scale attribute is a mandatory attribute that specifies the scale of the coordinates provided in the capture space and in the spatial information of the media capture belonging to the considered capture scene. The scale attribute can assume three different values:

"millimeters" - the scale is in millimeters. Systems which know their physical dimensions (for example professionally installed telepresence room systems) should always provide those real-world measurements.

"unknown" - the scale is not necessarily millimeters, but the scale is the same for every media capture in the capture scene. Systems which don't know specific physical dimensions but still know relative distances should select "unknown" in the scale attribute of the capture scene to be described.

"noscale" - there is no a common physical scale among the media captures of the capture scene. That means the scale could be different for each media capture.

```
<!-- SCALE TYPE -->
<xs:simpleType name="scaleType">
 <xs:restriction base="xs:string">
  <xs:enumeration value="millimeters"/>
  <xs:enumeration value="unknown"/>
  <xs:enumeration value="noscale"/>
 </xs:restriction>
</xs:simpleType>
```

15. <sceneEntry>

A <sceneEntry> element represents a capture scene entry, which contains a set of media capture of the same media type describing a capture scene.

A <sceneEntry> element is characterized as follows.

```
<!-- SCENE ENTRY TYPE -->
<xs:complexType name="sceneEntryType">
 <xs:sequence>
  <xs:element ref="description" minOccurs="0" maxOccurs="unbounded"/>
  <xs:element name="switchingPolicies" type="switchingPoliciesType"
  minOccurs="0"/>
  <xs:element name="mediaCaptureIDs" type="captureIDListType"/>
 </xs:sequence>
 <xs:attribute name="sceneEntryID" type="xs:ID" use="required"/>
 <xs:attribute name="mediaType" type="xs:string" use="required"/>
</xs:complexType>
```

One or more optional <description> elements provide human-readable information about what the scene entry contains. <description> is defined as already seen in Section 10.6.

The remaining child elements are described in the following subsections.

15.1. <switchingPolicies>

<switchingPolicies> represents the switching policies the media provider support for the media captures contained inside a scene entry. The <switchingPolicies> element contains two boolean elements:

<siteSwitching>: if set to "true", it means that the media provider supports the site switching policy for the included media captures;

<segmentSwitching>: if set to "true", it means that the media provider supports the segment switching policy for the included media captures.

The "site-switch" policy means all captures are switched at the same time to keep captures from the same endpoint site together.

The "segment-switch" policy means different captures can switch at different times, and can be coming from different endpoints.

```
<!-- SWITCHING POLICIES TYPE -->
<xs:complexType name="switchingPoliciesType">
 <xs:sequence>
  <xs:element name="siteSwitching" type="xs:boolean" minOccurs="0"/>
  <xs:element name="segmentSwitching" type="xs:boolean"
  minOccurs="0"/>
 </xs:sequence>
</xs:complexType>
```

15.2. <mediaCaptureIDs>

The <mediaCaptureIDs> is the list of the identifiers of the media captures included in the scene entry. It is an element of the captureIDListType type, which is defined as a sequence of <captureIDREF> each one containing the identifier of a media capture listed within the <mediaCaptures> element:

```
<!-- CAPTURE ID LIST TYPE -->
<xs:complexType name="captureIDListType">
 <xs:sequence>
  <xs:element name="captureIDREF" type="xs:IDREF"
  maxOccurs="unbounded"/>
 </xs:sequence>
</xs:complexType>
```

15.3. sceneEntryID attribute

The sceneEntryID attribute is a mandatory attribute containing the identifier of the capture scene entry represented by the <sceneEntry> element.

15.4. mediaType attribute

The mediaType attribute contains the media type of the media captures included in the scene entry.

16. <encoding>

The <encoding> element represents an individual encoding, i.e., a way to encode a media capture. Individual encodings can be characterized with features that are independent from the specific type of medium, and with features that are media-specific. We design the individual encoding type as an abstract type, providing all the features that can be common to all media types. Media-specific individual encodings, such as video encodings, audio encodings and others, are specialization of that type, as in a typical generalization-specialization hierarchy.

```
<!-- ENCODING TYPE -->
<xs:complexType name="encodingType" abstract="true">
 <xs:sequence>
  <xs:element name="encodingName" type="xs:string"/>
  <xs:element name="maxBandwidth" type="xs:integer"/>
  <xs:any namespace="##other" processContents="lax" minOccurs="0"
  maxOccurs="unbounded"/>
 </xs:sequence>
 <xs:attribute name="encodingID" type="xs:ID" use="required"/>
 <xs:anyAttribute namespace="##any" processContents="lax"/>
</xs:complexType>
```

16.1. <encodingName>

<encodingName> is a mandatory field containing the name of the encoding (e.g., G711, H264, ...).

16.2. <maxBandwidth>

<maxBandwidth> represent the maximum bitrate the media provider can instantiate for that encoding.

16.3. encodingID attribute

The encodingID attribute is a mandatory attribute containing the identifier of the individual encoding.

17. Audio encodings

Audio encodings inherit all the features of a generic individual encoding and can present further audio-specific encoding characteristics. The XML Schema definition of the audio encoding type is reported below:

```
<!-- AUDIO ENCODING TYPE -->
<xs:complexType name="audioEncodingType">
 <xs:complexContent>
  <xs:extension base="tns:encodingType">
   <xs:sequence>
    <xs:element name="encodedMedia" type="xs:string" fixed="audio"
    minOccurs="0"/>
   </xs:sequence>
  </xs:extension>
 </xs:complexContent>
</xs:complexType>
```

Up to now the only audio-specific information is the <encodedMedia> element containing the media type of the media captures that can be encoded with the considered individual encoding. In the case of audio encoding, that element is forced to "audio".

18. Video encodings

Similarly to audio encodings, video encodings can extend the information of a generic individual encoding with video-specific encoding features, such as <maxWidth>, <maxHeight> and <maxFrameRate>.

The <encodedMedia> element contains the media type of the media captures that can be encoded with the considered individual encoding. In the case of video encoding, that element is forced to "video".

```
<!-- VIDEO ENCODING TYPE -->
<xs:complexType name="videoEncodingType">
 <xs:complexContent>
  <xs:extension base="tns:encodingType">
   <xs:sequence>
    <xs:element name="encodedMedia" type="xs:string" fixed="video"
    minOccurs="0"/>
    <xs:element name="maxWidth" type="xs:integer" minOccurs="0"/>
    <xs:element name="maxHeight" type="xs:integer" minOccurs="0"/>
    <xs:element name="maxFrameRate" type="xs:integer" minOccurs="0"/>
   </xs:sequence>
  </xs:extension>
 </xs:complexContent>
</xs:complexType>
```

18.1. <maxWidth>

<maxWidth> represents the video resolution's maximum width supported by the video encoding, expressed in pixels.

[edt note: not present in -09 version of the framework doc]

18.2. <maxHeight>

<maxHeight> representd the video resolution's maximum heith supported by the video encoding, expressed in pixels.

[edt note: not present in -09 version of the framework doc]

18.3. <maxFrameRate>

<maxFrameRate> provides the maximum frame rate supported by the video encoding for the video capture to be encoded.

[edt note: not present in -09 version of the framework doc]

19. H26X encodings

This is an example of how it is possible to further specialize the definition of a video individual encoding in order to cover encoding specific information. A H26X video encoding can be represented through an element inheriting the video encoding characteristics described above (Section 18) and by adding other information such as <maxH26Xpps>, which represent the maximum number of pixels to be processed per second;.

```
<!-- H26X ENCODING TYPE -->
<xs:complexType name="h26XEncodingType">
 <xs:complexContent>
  <xs:extension base="tns:videoEncodingType">
   <xs:sequence>
    <!-- max number of pixels to be processed per second -->
    <xs:element name="maxH26Xpps" type="xs:integer"
    minOccurs="0"/>
   </xs:sequence>
  </xs:extension>
 </xs:complexContent>
</xs:complexType>
```

[edt note: Need to be checked]

20. <encodingGroup>

The <encodingGroup> element represents an encoding group, which is a set of one or more individual encodings, and parameters that apply to the group as a whole. The definition of the <encodingGroup> element is the following:

```
<!-- ENCODING GROUP TYPE -->
<xs:complexType name="encodingGroupType">
 <xs:sequence>
  <xs:element name="maxGroupBandwidth" type="xs:integer"/>
  <xs:element name="maxGroupPps" type="xs:integer"
  minOccurs="0"/>
  <xs:element name="encodingIDList" type="encodingIDListType"/>
  <xs:any namespace="##other" processContents="lax" minOccurs="0"
  maxOccurs="unbounded"/>
 </xs:sequence>
 <xs:attribute name="encodingGroupID" type="xs:ID" use="required"/>
 <xs:anyAttribute namespace="##any" processContents="lax"/>
</xs:complexType>
```

In the following, the contained elements are further described.

20.1. <maxGroupBandwidth>

<maxGroupBandwidth> is an optional field containing the maximum bitrate supported for all the individual encodings included in the encoding group.

20.2. <maxGroupPps>

<maxGroupPps> is an optional field containing the maximum number of pixel per second for all the individual encodings included in the encoding group.

[edt note: Need to be checked]

20.3. <encodingIDList>

<maxGroupBandwidth> is the list of the individual encoding grouped together. Each individual encoding is represented through its identifier contained within an <encIDREF> element.

```
<!-- ENCODING ID LIST TYPE -->
<xs:complexType name="encodingIDListType">
 <xs:sequence>
  <xs:element name="encIDREF" type="xs:IDREF" maxOccurs="unbounded"/>
 </xs:sequence>
</xs:complexType>
```

20.4. encodingGroupID attribute

The encodingGroupID attribute contains the identifier of the encoding group.

21. <simultaneousSet>

<simultaneousSet> represents a simultaneous set, i.e. a list of capture of the same type that cab be transmitted at the same time by a media provider. There are different simultaneous transmission sets for each media type.

```
<!-- SIMULTANEOUS SET TYPE -->
<xs:complexType name="simultaneousSetType">
 <xs:sequence>
   <xs:element name="captureIDREF" type="xs:IDREF"
   minOccurs="0" maxOccurs="unbounded"/>
   <xs:element name="sceneEntryIDREF" type="xs:IDREF"
   minOccurs="0" maxOccurs="unbounded"/>
 </xs:sequence>
</xs:complexType>
```

[edt note: need to be checked]

21.1. <captureIDREF>

<captureIDREF> contains the identifier of the media capture that belongs to the simultanous set.

21.2. <sceneEntryIDREF>

<captureIDREF> contains the identifier of the scene entry containing a group of capture that are able to be sent simultaneously with the other capture of the simultaneous set.

22. <captureEncoding>

A <captureEncoding> is given from the association of a media capture and an individual encoding, to form a capture stream. It is defined as en element of the following type:

```
<!-- CAPTURE ENCODING TYPE -->
<xs:complexType name="captureEncodingType">
 <xs:sequence>
  <xs:element name="mediaCaptureID" type="xs:string"/>
  <xs:element name="encodingID" type="xs:string"/>
 </xs:sequence>
</xs:complexType>
```

22.1. <mediaCaptureID>

<mediaCaptureID> contains the identifier of the media capture that has been encoded to form the capture encoding.

22.2. <encodingID>

<encodingID> contains the identifier of the applied individual encoding.

23. <clueInfo>

The <clueInfo> element has been left within the XML Schema for the sake of convenience when representing a prototype of ADVERTISEMENT message (see the example section).

```
<!-- CLUE INFO ELEMENT -->
<!-- the <clueInfo> envelope can be seen
     as the ancestor of an <advertisement> envelope -->
<xs:element name="clueInfo" type="clueInfoType"/>

<!-- CLUE INFO TYPE -->
<xs:complexType name="clueInfoType">
  <xs:sequence>
   <xs:element ref="mediaCaptures"/>
   <xs:element ref="encodings"/>
   <xs:element ref="encodingGroups"/>
   <xs:element ref="captureScenes"/>
   <xs:element ref="simultaneousSets"/>
   <xs:any namespace="##other" processContents="lax" minOccurs="0"
   maxOccurs="unbounded"/>
  </xs:sequence>
  <xs:attribute name="clueInfoID" type="xs:ID" use="required"/>
  <xs:anyAttribute namespace="##other" processContents="lax"/>
</xs:complexType>
```

24. Sample XML file

The following XML document represents a schema compliant example of a CLUE telepresence scenario.

There are 5 video captures:

VC0: the video from the left camera

VC1: the video from the central camera

VC2: the video from the right camera

VC3: the overall view of the telepresence room taken from the central camera

VC4: the video associated with the slide stream

There are 2 audio captures:

AC0: the overall room audio taken from the central camera

AC1: the audio associated with the slide stream presentation

The captures are organized into two capture scenes:

CS1: this scene contains captures associated with the participants that are in the telepresence room.

CS2: this scene contains captures associated with the slide presentation, which is a pre-registered presentation played within the context of the telepresence session.

Within the capture scene CS1, there are three scene entries available:

CS1_SE1: this entry contains the partipants' video captures taken from the three cameras (VC0, VC1, VC2).

CS1_SE2: this entry contains the zoomed-out view of the overall telepresence room (VC3)

CS1_SE3: this entry contains the overall telepresence room audio (AC0)

On the other hand, capture scene CS2 presents two scene entries:

CS2_SE1: this entry contains the presentation audio stream (AC1)

CS2_SE2: this entry contains the presentation video stream (VC4)

There are two encoding groups:

EG0 This encoding groups involves video encodings ENC0, ENC1, ENC2

EG1 This encoding groups involves audio encodings ENC3, ENC4

As to the simultaneous sets, only VC1 and VC3 cannot be transmitted simultaneously since they are captured by the same device. i.e. the central camera (VC3 is a zoomed-out view while VC1 is a focused view of the front participants). The simultaneous sets would then be the following:

SS1 made by VC0, VC1, VC2, VC4, AC0, AC1

SS2 made by VC0, VC3, VC2, VC4, AC0, AC1

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<clueInfo xmlns="urn:ietf:params:xml:ns:clue-info" clueInfoID="prova">
    <mediaCaptures>
        <mediaCapture
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
```

```
        xsi:type="audioCaptureType" captureID="AC1">
            <capturedMedia>audio</capturedMedia>
            <captureSceneIDREF>CS2</captureSceneIDREF>
            <encGroupIDREF>EG1</encGroupIDREF>
            <nonSpatiallyDefinible>true</nonSpatiallyDefinible>
            <description lang="en">presentation audio</description>
            <content>slide</content>
            <audioChannelFormat>mono</audioChannelFormat>
        </mediaCapture>
        <mediaCapture
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        xsi:type="videoCaptureType" captureID="VC4">
            <capturedMedia>video</capturedMedia>
            <captureSceneIDREF>CS2</captureSceneIDREF>
            <encGroupIDREF>EG0</encGroupIDREF>
            <nonSpatiallyDefinible>true</nonSpatiallyDefinible>
            <description lang="en">presentation video</description>
            <content>slides</content>
        </mediaCapture>
        <mediaCapture
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        xsi:type="audioCaptureType" captureID="AC0">
            <capturedMedia>audio</capturedMedia>
            <captureSceneIDREF>CS1</captureSceneIDREF>
            <encGroupIDREF>EG1</encGroupIDREF>
            <spatialInformation>
                <capturePoint>
                    <x>0.5</x>
                    <y>1.0</y>
                    <z>0.5</z>
                    <lineOfCapturePoint>
                        <x>0.5</x>
                        <y>0.0</y>
                        <z>0.5</z>
                    </lineOfCapturePoint>
                </capturePoint>
            </spatialInformation>
            <description lang="en">
            audio from the central camera mic</description>
            <audioChannelFormat>mono</audioChannelFormat>
            <micPattern>figure8</micPattern>
        </mediaCapture>
        <mediaCapture
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        xsi:type="videoCaptureType" captureID="VC3">
            <capturedMedia>video</capturedMedia>
            <captureSceneIDREF>CS1</captureSceneIDREF>
            <encGroupIDREF>EG0</encGroupIDREF>
```

```
            <spatialInformation>
                <capturePoint>
                    <x>1.5</x>
                    <y>1.0</y>
                    <z>0.5</z>
                    <lineOfCapturePoint>
                        <x>1.5</x>
                        <y>0.0</y>
                        <z>0.5</z>
                    </lineOfCapturePoint>
                </capturePoint>
                <captureArea>
                    <bottomLeft>
                        <x>0.0</x>
                        <y>3.0</y>
                        <z>0.0</z>
                    </bottomLeft>
                    <bottomRight>
                        <x>3.0</x>
                        <y>3.0</y>
                        <z>0.0</z>
                    </bottomRight>
                    <topLeft>
                        <x>0.0</x>
                        <y>3.0</y>
                        <z>3.0</z>
                    </topLeft>
                    <topRight>
                        <x>3.0</x>
                        <y>3.0</y>
                        <z>3.0</z>
                    </topRight>
                </captureArea>
            </spatialInformation>
            <description lang="en">
            zoomed out view of the room</description>
        </mediaCapture>
        <mediaCapture
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        xsi:type="videoCaptureType" captureID="VC2">
            <capturedMedia>video</capturedMedia>
            <captureSceneIDREF>CS1</captureSceneIDREF>
            <encGroupIDREF>EG0</encGroupIDREF>
            <spatialInformation>
                <capturePoint>
                    <x>2.5</x>
                    <y>1.0</y>
                    <z>0.5</z>
```

```
                <lineOfCapturePoint>
                    <x>2.5</x>
                    <y>0.0</y>
                    <z>0.5</z>
                </lineOfCapturePoint>
            </capturePoint>
            <captureArea>
                <bottomLeft>
                    <x>2.0</x>
                    <y>3.0</y>
                    <z>0.0</z>
                </bottomLeft>
                <bottomRight>
                    <x>3.0</x>
                    <y>3.0</y>
                    <z>0.0</z>
                </bottomRight>
                <topLeft>
                    <x>2.0</x>
                    <y>3.0</y>
                    <z>3.0</z>
                </topLeft>
                <topRight>
                    <x>3.0</x>
                    <y>3.0</y>
                    <z>3.0</z>
                </topRight>
            </captureArea>
        </spatialInformation>
        <description lang="en">right camera video</description>
    </mediaCapture>
    <mediaCapture
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:type="videoCaptureType" captureID="VC1">
        <capturedMedia>video</capturedMedia>
        <captureSceneIDREF>CS1</captureSceneIDREF>
        <encGroupIDREF>EG0</encGroupIDREF>
        <spatialInformation>
            <capturePoint>
                <x>1.5</x>
                <y>1.0</y>
                <z>0.5</z>
                <lineOfCapturePoint>
                    <x>1.5</x>
                    <y>0.0</y>
                    <z>0.5</z>
                </lineOfCapturePoint>
            </capturePoint>
```

```
                <captureArea>
                    <bottomLeft>
                        <x>1.0</x>
                        <y>3.0</y>
                        <z>0.0</z>
                    </bottomLeft>
                    <bottomRight>
                        <x>2.0</x>
                        <y>3.0</y>
                        <z>0.0</z>
                    </bottomRight>
                    <topLeft>
                        <x>1.0</x>
                        <y>3.0</y>
                        <z>3.0</z>
                    </topLeft>
                    <topRight>
                        <x>2.0</x>
                        <y>3.0</y>
                        <z>3.0</z>
                    </topRight>
                </captureArea>
            </spatialInformation>
            <description lang="en">central camera video</description>
        </mediaCapture>
        <mediaCapture
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        xsi:type="videoCaptureType" captureID="VC0">
            <capturedMedia>video</capturedMedia>
            <captureSceneIDREF>CS1</captureSceneIDREF>
            <encGroupIDREF>EG0</encGroupIDREF>
            <spatialInformation>
                <capturePoint>
                    <x>0.5</x>
                    <y>1.0</y>
                    <z>0.5</z>
                    <lineOfCapturePoint>
                        <x>0.5</x>
                        <y>0.0</y>
                        <z>0.5</z>
                    </lineOfCapturePoint>
                </capturePoint>
                <captureArea>
                    <bottomLeft>
                        <x>0.0</x>
                        <y>3.0</y>
                        <z>0.0</z>
                    </bottomLeft>
```

```
                    <bottomRight>
                        <x>1.0</x>
                        <y>3.0</y>
                        <z>0.0</z>
                    </bottomRight>
                    <topLeft>
                        <x>0.0</x>
                        <y>3.0</y>
                        <z>3.0</z>
                    </topLeft>
                    <topRight>
                        <x>1.0</x>
                        <y>3.0</y>
                        <z>3.0</z>
                    </topRight>
                </captureArea>
            </spatialInformation>
            <description lang="en">left camera video</description>
        </mediaCapture>
    </mediaCaptures>
    <encodings>
        <encoding xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        xsi:type="videoEncodingType" encodingID="ENC0">
            <encodingName>h263</encodingName>
            <maxBandwidth>4000000</maxBandwidth>
            <encodedMedia>video</encodedMedia>
            <maxWidth>1920</maxWidth>
            <maxHeight>1088</maxHeight>
        </encoding>
        <encoding xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        xsi:type="videoEncodingType" encodingID="ENC1">
            <encodingName>h263</encodingName>
            <maxBandwidth>4000000</maxBandwidth>
            <encodedMedia>video</encodedMedia>
            <maxWidth>1920</maxWidth>
            <maxHeight>1088</maxHeight>
        </encoding>
        <encoding xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        xsi:type="videoEncodingType" encodingID="ENC2">
            <encodingName>h263</encodingName>
            <maxBandwidth>4000000</maxBandwidth>
            <encodedMedia>video</encodedMedia>
            <maxWidth>1920</maxWidth>
            <maxHeight>1088</maxHeight>
        </encoding>
        <encoding xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        xsi:type="audioEncodingType" encodingID="ENC3">
            <encodingName>g711</encodingName>
```

```
            <maxBandwidth>64000</maxBandwidth>
            <encodedMedia>audio</encodedMedia>
        </encoding>
        <encoding xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        xsi:type="audioEncodingType" encodingID="ENC4">
            <encodingName>g711</encodingName>
            <maxBandwidth>64000</maxBandwidth>
            <encodedMedia>audio</encodedMedia>
        </encoding>
    </encodings>
    <encodingGroups>
        <encodingGroup encodingGroupID="EG0">
            <maxGroupBandwidth>12000000</maxGroupBandwidth>
            <encodingIDList>
                <encIDREF>ENC0</encIDREF>
                <encIDREF>ENC1</encIDREF>
                <encIDREF>ENC2</encIDREF>
            </encodingIDList>
        </encodingGroup>
        <encodingGroup encodingGroupID="EG1">
            <maxGroupBandwidth>12000000</maxGroupBandwidth>
            <encodingIDList>
                <encIDREF>ENC3</encIDREF>
                <encIDREF>ENC4</encIDREF>
            </encodingIDList>
        </encodingGroup>
    </encodingGroups>
    <captureScenes>
        <captureScene scale="unknown" sceneID="CS1">
            <description lang="en">main scene</description>
            <sceneSpace>
                <bottomLeftFront>
                    <x>0.0</x>
                    <y>3.0</y>
                    <z>0.0</z>
                </bottomLeftFront>
                <bottomRightFront>
                    <x>3.0</x>
                    <y>3.0</y>
                    <z>0.0</z>
                </bottomRightFront>
                <topLeftFront>
                    <x>0.0</x>
                    <y>3.0</y>
                    <z>2.0</z>
                </topLeftFront>
                <topRightFront>
                    <x>3.0</x>
```

```
                <y>3.0</y>
                <z>2.0</z>
            </topRightFront>
            <bottomLeftBack>
                <x>0.0</x>
                <y>3.0</y>
                <z>0.0</z>
            </bottomLeftBack>
            <bottomRightBack>
                <x>3.0</x>
                <y>3.0</y>
                <z>0.0</z>
            </bottomRightBack>
            <topLeftBack>
                <x>0.0</x>
                <y>3.0</y>
                <z>2.0</z>
            </topLeftBack>
            <topRightBack>
                <x>3.0</x>
                <y>3.0</y>
                <z>2.0</z>
            </topRightBack>
        </sceneSpace>
        <sceneEntries>
            <sceneEntry mediaType="video" sceneEntryID="SE1">
                <description lang="en">
                participants streams</description>
                <mediaCaptureIDs>
                    <captureIDREF>VC0</captureIDREF>
                    <captureIDREF>VC1</captureIDREF>
                    <captureIDREF>VC2</captureIDREF>
                </mediaCaptureIDs>
            </sceneEntry>
            <sceneEntry mediaType="video" sceneEntryID="SE2">
                <description lang="en">room stream</description>
                <mediaCaptureIDs>
                    <captureIDREF>VC3</captureIDREF>
                </mediaCaptureIDs>
            </sceneEntry>
            <sceneEntry mediaType="audio" sceneEntryID="SE3">
                <description lang="en">room audio</description>
                <mediaCaptureIDs>
                    <captureIDREF>AC0</captureIDREF>
                </mediaCaptureIDs>
            </sceneEntry>
        </sceneEntries>
    </captureScene>
```

```
        <captureScene scale="noscale" sceneID="CS2">
            <description lang="en">presentation</description>
            <sceneEntries>
                <sceneEntry mediaType="video" sceneEntryID="CS2_SE1">
                    <description lang="en">
                    presentation video</description>
                    <mediaCaptureIDs>
                        <captureIDREF>VC4</captureIDREF>
                    </mediaCaptureIDs>
                </sceneEntry>
                <sceneEntry mediaType="audio" sceneEntryID="CS2_SE2">
                    <description lang="en">
                    presentation audio</description>
                    <mediaCaptureIDs>
                        <captureIDREF>AC1</captureIDREF>
                    </mediaCaptureIDs>
                </sceneEntry>
            </sceneEntries>
        </captureScene>
    </captureScenes>
    <simultaneousSets>
        <simultaneousSet setID="SS1">
            <captureIDREF>VC0</captureIDREF>
            <captureIDREF>VC1</captureIDREF>
            <captureIDREF>VC2</captureIDREF>
            <captureIDREF>VC4</captureIDREF>
            <captureIDREF>AC0</captureIDREF>
            <captureIDREF>AC1</captureIDREF>
        </simultaneousSet>
        <simultaneousSet setID="SS2">
            <captureIDREF>VC0</captureIDREF>
            <captureIDREF>VC3</captureIDREF>
            <captureIDREF>VC2</captureIDREF>
            <captureIDREF>VC4</captureIDREF>
            <captureIDREF>AC0</captureIDREF>
            <captureIDREF>AC1</captureIDREF>
        </simultaneousSet>
    </simultaneousSets>
</clueInfo>
```

25. Diff with unofficial -02 version

Here the link to the unofficial -02 version:
http://www.grid.unina.it/Didattica/RetiDiCalcolatori/inf/draft-presta-clue-data-model-schema-02.html

<mediaCaptures> moved from <sceneEntry> to <clueInfo> elements. <mediaCaptures> have been moved out from the <captureScene> blob again. Media captures should have identifiers that are valid out of the local scope of capture scenes, since a consumer should be able to require also single captures in the CONFIGURE message. This design choice reflects a bottom up approach where captures are the basis of the data model. In each media capture a reference to the capture scene containing it is provided. It identifies the space the spatial information of the media capture refers to.

XML document example updated A new example, compliant with the updated schema, has been provided.

language attribute added to <mediaCapture> Such optional attribute reflects the language used in the capture, if any. The purpose of the element could match the one of the language attribute proposed in [I-D.groves-clue-capture-attr].

<priority> added to <mediaCapture> The priority element has an integer value helping in specifying a media capture relative importance with respect to the other captures. That element could correspond to the priority attribute introduced in [I-D.groves-clue-capture-attr].

<embeddedText> added to <videoCapture> The element, if present, indicates text embedded in the video capture. The language used in such embedded textual description is also envisioned within the <embeddedText> element itself. That element could correspond to the priority attribute introduced in [I-D.groves-clue-capture-attr]

<relatedTo> added to <mediaCapture> That optional element contains the ID of a capture the capture refers to. This is for supporting cases where there is the translation of a main capture in a different language. Such translation can be marked with a <relatedTo> tag to refer to the main capture. This could be interpreted the same manner of the supplementary information attribute proposed in [I-D.groves-clue-capture-attr] and further discussed in http://www.ietf.org/mail-archive/web/clue/current/msg02238.html.

<dynamic> added to <mediaCapture> That optional boolean element has the same purpose of the dynamic attribute proposed in [I-D.groves-clue-capture-attr]. It indicates if the capture device originating the capture moves during the telepresence session.

new element definition for <description> <description> has a new attribute, lang, indicating the language used for the text within <description>. <description> is used to provide human readable information about captures, scene, and scene entries. The definitions of the corresponding XML elements (i.e., <mediaCapture>, <captureScene>, <sceneEntry>) have been updated to make them able to contain more than one <description>. In that way, they can be described in different languages.

text capture added as new type of capture The element is just a place holder, since it is not characterized with any further information up to now.

26. Diff with -02 version

<sceneSpace> of capture space type <sceneSpace> (was:<sceneArea>) describes a bounding volume for the space of a capture scene as an arbitrary hexahedrons with eight points (placeholder solution).

H26X encoding to be checked.

Simultaneous sets The XML Schema definition of the simultaneous sets has changed. A simultaneous set is defined as a list of L media capture identifiers and M capture scene entrie identifiers, where L, M can be 0 or unbounded.

Capture encoding A new XML Schema type has been added to describe capture encodings as the result of the association of a media capture, represented by its identifier, with an individual encoding, represented by its identifier as well.

Clue info The <clueInfo> element has been left within the XML Schema for the sake of convenience when representing a prototype of ADVERTISEMENT message (see the example section).

Data model definitions added For each element of the datamodel a brief description has been reported to foster discussion.

27. Informative References

[I-D.groves-clue-capture-attr] Groves, C., Yang, W., and R. Even, "CLUE media capture description", draft-groves-clue-capture-attr-01 (work in progress), February 2013.

[I-D.ietf-clue-framework] Duckworth, M., Pepperell, A., and S. Wenger, "Framework for Telepresence Multi-Streams",

draft-ietf-clue-framework-09 (work in progress), February 2013.

[I-D.romanow-clue-data-model] Romanow, A. and A. Pepperell, "Data model for the CLUE Framework", draft-romanow-clue-data-model-01 (work in progress), June 2012.

[RFC4796] Hautakorpi, J. and G. Camarillo, "The Session Description Protocol (SDP) Content Attribute", RFC 4796, February 2007.

Authors' Addresses

Roberta Presta
University of Napoli
Via Claudio 21
Napoli 80125
Italy

EMail: roberta.presta@unina.it

Simon Pietro Romano
University of Napoli
Via Claudio 21
Napoli 80125
Italy

EMail: spromano@unina.it