CLUE WG
Internet Draft
Intended status: Standards Track
Expires: August 10, 2014

M. Duckworth, Ed.
Polycom
A. Pepperell
Acano
S. Wenger
Vidyo

February 10, 2014

Framework for Telepresence Multi-Streams
draft-ietf-clue-framework-14.txt

Abstract

This document defines a framework for a protocol to enable devices in a telepresence conference to interoperate. The protocol enables communication of information about multiple media streams so a sending system and receiving system can make reasonable decisions about transmitting, selecting and rendering the media streams. This protocol is used in addition to SIP signaling for setting up a telepresence session.

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 August 10, 2014.

Table of Contents

1. Introduction

Current telepresence systems, though based on open standards such as RTP [RFC3550] and SIP [RFC3261], cannot easily interoperate with each other. A major factor limiting the interoperability of telepresence systems is the lack of a standardized way to describe and negotiate the use of the multiple streams of audio and video comprising the media flows. This document provides a framework for protocols to enable interoperability by handling multiple streams in a standardized way. The framework is intended to support the use cases described in draft-ietf-clue-telepresence-use-cases and to meet the requirements in draft-ietf-clue-telepresence-requirements.

The basic session setup for the use cases is based on SIP [RFC3261] and SDP offer/answer [RFC3264]. In addition to basic SIP & SDP offer/answer, CLUE specific signaling is required to exchange the information describing the multiple media streams. The motivation for this framework, an overview of the signaling, and information required to be exchanged is described in subsequent sections of this document. The signaling details and data model are provided in subsequent documents.

2. Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [RFC2119].

3. Definitions

The terms defined below are used throughout this document and companion documents and they are normative. In order to easily identify the use of a defined term, those terms are capitalized.

Advertisement: a CLUE message a Media Provider sends to a Media Consumer describing specific aspects of the content of the media, the formatting of the media streams it can send, and any restrictions it has in terms of being able to provide certain Streams simultaneously.

Audio Capture: Media Capture for audio. Denoted as ACn in the example cases in this document.

Camera-Left and Right: For Media Captures, camera-left and camera-right are from the point of view of a person observing the rendered media. They are the opposite of Stage-Left and Stage-Right.

Capture: Same as Media Capture.

Capture Device: A device that converts audio and video input into an electrical signal, in most cases to be fed into a media encoder.

Capture Encoding: A specific encoding of a Media Capture, to be sent by a Media Provider to a Media Consumer via RTP.

Capture Scene: a structure representing a spatial region containing one or more Capture Devices, each capturing media representing a portion of the region. The spatial region represented by a Capture Scene MAY or may not correspond to a real region in physical space, such as a room. A Capture Scene includes attributes and one or more Capture Scene Entries, with each entry including one or more Media Captures.

Capture Scene Entry: a list of Media Captures of the same media type that together form one way to represent the entire Capture Scene.

Conference: used as defined in [RFC4353], A Framework for Conferencing within the Session Initiation Protocol (SIP).

Configure Message: A CLUE message a Media Consumer sends to a Media Provider specifying which content and media streams it wants to receive, based on the information in a corresponding Advertisement message.

Consumer: short for Media Consumer.

Encoding or Individual Encoding: a set of parameters representing a way to encode a Media Capture to become a Capture Encoding.

Encoding Group: A set of encoding parameters representing a total media encoding capability to be sub-divided across potentially multiple Individual Encodings.

Endpoint: The logical point of final termination through receiving, decoding and rendering, and/or initiation through capturing, encoding, and sending of media streams. An endpoint consists of one or more physical devices which source and sink media streams, and exactly one [RFC4353] Participant (which, in turn, includes exactly one SIP User Agent). Endpoints can be anything from multiscreen/multicamera rooms to handheld devices.

Front: the portion of the room closest to the cameras. In going towards back you move away from the cameras.

MCU: Multipoint Control Unit (MCU) - a device that connects two or more endpoints together into one single multimedia conference [RFC5117]. An MCU includes an [RFC4353] like Mixer, without the [RFC4353] requirement to send media to each participant.

Media: Any data that, after suitable encoding, can be conveyed over RTP, including audio, video or timed text.

Media Capture: a source of Media, such as from one or more Capture Devices or constructed from other Media streams.

Media Consumer: an Endpoint or middle box that receives Media streams

Media Provider: an Endpoint or middle box that sends Media streams

Model: a set of assumptions a telepresence system of a given vendor adheres to and expects the remote telepresence system(s) also to adhere to.

Multiple Content Capture: A Capture for audio or video that indicates that the Capture contains multiple audio or video Captures. Single Media Captures may or may not be present in the resultant Capture Encoding depending on time or space. Denoted as MCCn in the example cases in this document.

Plane of Interest: The spatial plane containing the most relevant subject matter.

Provider: Same as Media Provider.

Render: the process of generating a representation from a media, such as displayed motion video or sound emitted from loudspeakers.

Simultaneous Transmission Set: a set of Media Captures that can be transmitted simultaneously from a Media Provider.

Single Media Capture: A capture which contains media from a single source capture device, i.e. audio capture, video capture.

Spatial Relation: The arrangement in space of two objects, in contrast to relation in time or other relationships. See also Camera-Left and Right.

Stage-Left and Right: For Media Captures, Stage-left and Stage-right are the opposite of Camera-left and Camera-right. For the case of a person facing (and captured by) a camera, Stage-left and Stage-right are from the point of view of that person.

Stream: a Capture Encoding sent from a Media Provider to a Media Consumer via RTP [RFC3550].

Stream Characteristics: the media stream attributes commonly used in non-CLUE SIP/SDP environments (such as: media codec, bit rate, resolution, profile/level etc.) as well as CLUE specific attributes, such as the Capture ID or a spatial location.

Video Capture: Media Capture for video. Denoted as VCn in the example cases in this document.

Video Composite: A single image that is formed, normally by an RTP mixer inside an MCU, by combining visual elements from separate sources.

4. Overview & Motivation

This section provides an overview of the functional elements defined in this document to represent a telepresence system. The motivations for the framework described in this document are also provided.

Two key concepts introduced in this document are the terms "Media Provider" and "Media Consumer". A Media Provider represents the entity that is sending the media and a Media Consumer represents the entity that is receiving the media. A Media Provider provides Media in the form of RTP packets, a Media Consumer consumes those RTP packets. Media Providers and Media Consumers can reside in Endpoints or in middleboxes such as Multipoint Control Units (MCUs). A Media Provider in an Endpoint is usually associated with the generation of media for Media Captures; these Media Captures are typically sourced from cameras, microphones, and the like. Similarly, the Media Consumer in an Endpoint is usually associated with renderers, such as screens and loudspeakers. In middleboxes, Media Providers and Consumers can have the form of outputs and inputs, respectively, of RTP mixers, RTP translators, and similar devices. Typically, telepresence devices such as Endpoints and middleboxes would perform as both Media Providers and Media Consumers, the former being concerned with those devices' transmitted media and the latter with those devices' received media. In a few circumstances, a CLUE Endpoint middlebox includes only Consumer or Provider functionality, such as recorder-type Consumers or webcam-type Providers.

The motivations for the framework outlined in this document include the following:

(1) Endpoints in telepresence systems typically have multiple Media Capture and Media Render devices, e.g., multiple cameras and screens. While previous system designs were able to set up calls that would capture media using all cameras and display media on all screens, for example, there is no mechanism that can associate these Media Captures with each other in space and time.

(2) The mere fact that there are multiple capture and rendering devices, each of which may be configurable in aspects such as zoom,

leads to the difficulty that a variable number of such devices can be used to capture different aspects of a region. The Capture Scene concept allows for the description of multiple setups for those multiple capture devices that could represent sensible operation points of the physical capture devices in a room, chosen by the operator. A Consumer can pick and choose from those configurations based on its rendering abilities and inform the Provider about its choices. Details are provided in section 7.

(3) In some cases, physical limitations or other reasons disallow the concurrent use of a device in more than one setup. For example, the center camera in a typical three-camera conference room can set its zoom objective either to capture only the middle few seats, or all seats of a room, but not both concurrently. The Simultaneous Transmission Set concept allows a Provider to signal such limitations. Simultaneous Transmission Sets are part of the Capture Scene description, and discussed in section 8.

(4) Often, the devices in a room do not have the computational complexity or connectivity to deal with multiple encoding options simultaneously, even if each of these options is sensible in certain scenarios, and even if the simultaneous transmission is also sensible (i.e. in case of multicast media distribution to multiple endpoints). Such constraints can be expressed by the Provider using the Encoding Group concept, described in section 9.

(5) Due to the potentially large number of RTP flows required for a Multimedia Conference involving potentially many Endpoints, each of which can have many Media Captures and media renderers, it has become common to multiplex multiple RTP media flows onto the same transport address, so to avoid using the port number as a multiplexing point and the associated shortcomings such as NAT/firewall traversal. While the actual mapping of those RTP flows to the header fields of the RTP packets is not subject of this specification, the large number of possible permutations of sensible options a Media Provider can make available to a Media Consumer makes a mechanism desirable that allows to narrow down the number of possible options that a SIP offer-answer exchange has to consider. Such information is made available using protocol mechanisms specified in this document and companion documents, although it should be stressed that its use in an implementation is OPTIONAL. Also, there are aspects of the control of both Endpoints and middleboxes/MCUs that dynamically change during the progress of a call, such as audio-level based screen switching, layout changes, and so on, which need to be conveyed. Note that these control

aspects are complementary to those specified in traditional SIP based conference management such as BFCP. An exemplary call flow can be found in section 5.

Finally, all this information needs to be conveyed, and the notion of support for it needs to be established. This is done by the negotiation of a "CLUE channel", a data channel negotiated early during the initiation of a call. An Endpoint or MCU that rejects the establishment of this data channel, by definition, is not supporting CLUE based mechanisms, whereas an Endpoint or MCU that accepts it is REQUIRED to use it to the extent specified in this document and its companion documents.

5. Overview of the Framework/Model

The CLUE framework specifies how multiple media streams are to be handled in a telepresence conference.

A Media Provider (transmitting Endpoint or MCU) describes specific aspects of the content of the media and the formatting of the media streams it can send in an Advertisement; and the Media Consumer responds to the Media Provider by specifying which content and media streams it wants to receive in a Configure message. The Provider then transmits the asked-for content in the specified streams.

This Advertisement and Configure MUST occur during call initiation but MAY also happen at any time throughout the call, whenever there is a change in what the Consumer wants to receive or (perhaps less common) the Provider can send.

An Endpoint or MCU typically act as both Provider and Consumer at the same time, sending Advertisements and sending Configurations in response to receiving Advertisements. (It is possible to be just one or the other.)

The data model is based around two main concepts: a Capture and an Encoding. A Media Capture (MC), such as audio or video, describes the content a Provider can send. Media Captures are described in terms of CLUE-defined attributes, such as spatial relationships and purpose of the capture. Providers tell Consumers which Media Captures they can provide, described in terms of the Media Capture attributes.

A Provider organizes its Media Captures into one or more Capture Scenes, each representing a spatial region, such as a room. A Consumer chooses which Media Captures it wants to receive from each Capture Scene.

In addition, the Provider can send the Consumer a description of the Individual Encodings it can send in terms of the media attributes of the Encodings, in particular, audio and video parameters such as bandwidth, frame rate, macroblocks per second. Note that this is OPTIONAL, and intended to minimize the number of options a later SDP offer-answer would have to include in the SDP in case of complex setups, as should become clearer shortly when discussing an outline of the call flow.

The Provider can also specify constraints on its ability to provide Media, and a sensible design choice for a Consumer is to take these into account when choosing the content and Capture Encodings it requests in the later offer-answer exchange. Some constraints are due to the physical limitations of devices--for example, a camera may not be able to provide zoom and non-zoom views simultaneously. Other constraints are system based, such as maximum bandwidth and maximum video coding performance measured in macroblocks/second.

The following diagram illustrates the information contained in an Advertisement.

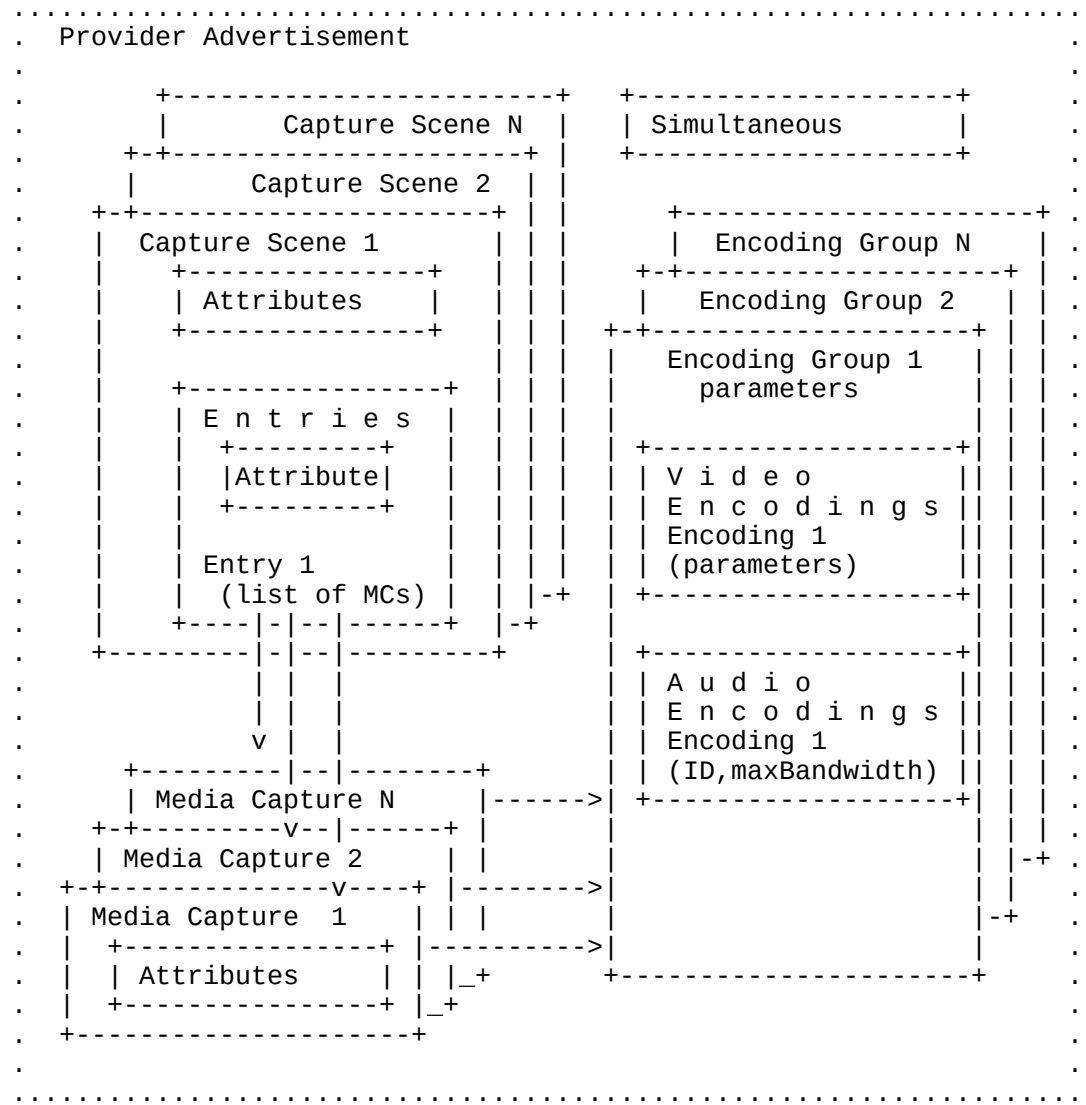


Figure 1: Advertisement Structure

A very brief outline of the call flow used by a simple system (two Endpoints) in compliance with this document can be described as follows, and as shown in the following figure.

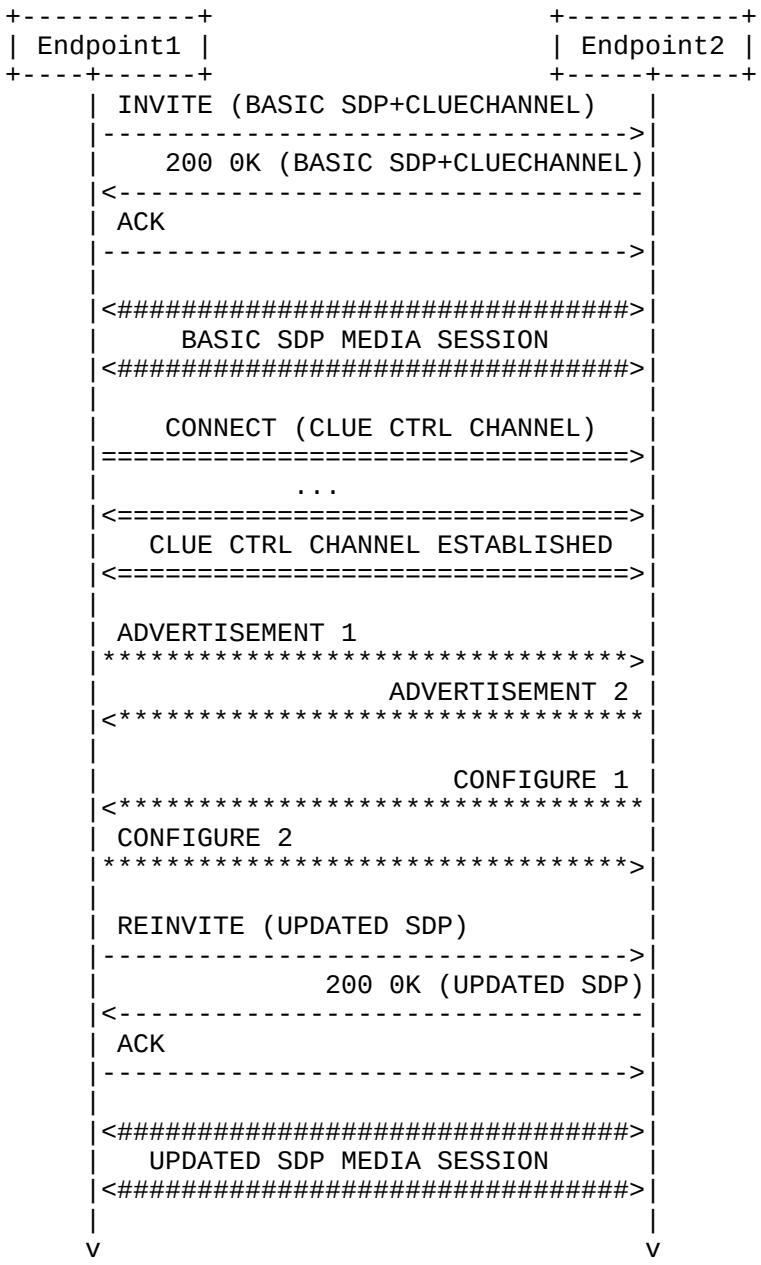


Figure 2: Basic Information Flow

An initial offer/answer exchange establishes a basic media session, for example audio-only, and a CLUE channel between two Endpoints. With the establishment of that channel, the endpoints have consented to use the CLUE protocol mechanisms and, therefore, MUST adhere to the CLUE protocol suite as outlined herein.

Over this CLUE channel, the Provider in each Endpoint conveys its characteristics and capabilities by sending an Advertisement as specified herein. The Advertisement is typically not sufficient to set up all media. The Consumer in the Endpoint receives the information provided by the Provider, and can use it for two purposes. First, it MUST construct and send a CLUE Configure message to tell the Provider what the Consumer wishes to receive. Second, it MAY, but is not necessarily REQUIRED to, use the information provided to tailor the SDP it is going to send during the following SIP offer/answer exchange, and its reaction to SDP it receives in that step. It is often a sensible implementation choice to do so, as the representation of the media information conveyed over the CLUE channel can dramatically cut down on the size of SDP messages used in the O/A exchange that follows. Spatial relationships associated with the Media can be included in the Advertisement, and it is often sensible for the Media Consumer to take those spatial relationships into account when tailoring the SDP.

This CLUE exchange MUST be followed by an SDP offer answer exchange that not only establishes those aspects of the media that have not been "negotiated" over CLUE, but has also the side effect of setting up the media transmission itself, involving potentially security exchanges, ICE, and whatnot. This step is plain vanilla SIP, with the exception that the SDP used herein, in most (but not necessarily all) cases can be considerably smaller than the SDP a system would typically need to exchange if there were no pre-established knowledge about the Provider and Consumer characteristics. (The need for cutting down SDP size is not quite obvious for a point-to-point call involving simple endpoints; however, when considering a large multipoint conference involving many multi-screen/multi-camera endpoints, each of which can operate using multiple codecs for each camera and microphone, it becomes perhaps somewhat more intuitive.)

During the lifetime of a call, further exchanges MAY occur over the CLUE channel. In some cases, those further exchanges lead to a

modified system behavior of Provider or Consumer (or both) without any other protocol activity such as further offer/answer exchanges. For example, voice-activated screen switching, signaled over the CLUE channel, ought not to lead to heavy-handed mechanisms like SIP re-invites. However, in other cases, after the CLUE negotiation an additional offer/answer exchange becomes necessary. For example, if both sides decide to upgrade the call from a single screen to a multi-screen call and more bandwidth is required for the additional video channels compared to what was previously negotiated using offer/answer, a new O/A exchange is REQUIRED.

One aspect of the protocol outlined herein and specified in more detail in companion documents is that it makes available information regarding the Provider's capabilities to deliver Media, and attributes related to that Media such as their spatial relationship, to the Consumer. The operation of the renderer inside the Consumer is unspecified in that it can choose to ignore some information provided by the Provider, and/or not render media streams available from the Provider (although it MUST follow the CLUE protocol and, therefore, MUST gracefully receive and respond (through a Configure) to the Provider's information). All CLUE protocol mechanisms are OPTIONAL in the Consumer in the sense that, while the Consumer MUST be able to receive (and, potentially, gracefully acknowledge) CLUE messages, it is free to ignore the information provided therein. Obviously, this is not a particularly sensible design choice in almost all conceivable cases.

A CLUE-implementing device interoperates with a device that does not support CLUE, because the non-CLUE device does, by definition, not understand the offer of a CLUE channel in the initial offer/answer exchange and, therefore, will reject it. This rejection MUST be used as the indication to the CLUE-implementing device that the other side of the communication is not compliant with CLUE, and to fall back to behavior that does not require CLUE.

As for the media, Provider and Consumer have an end-to-end communication relationship with respect to (RTP transported) media; and the mechanisms described herein and in companion documents do not change the aspects of setting up those RTP flows and sessions. In other words, the RTP media sessions conform to the negotiated SDP whether or not CLUE is used.

6. Spatial Relationships

In order for a Consumer to perform a proper rendering, it is often necessary or at least helpful for the Consumer to have received spatial information about the streams it is receiving. CLUE defines a coordinate system that allows Media Providers to describe the spatial relationships of their Media Captures to enable proper scaling and spatially sensible rendering of their streams. The coordinate system is based on a few principles:

- Simple systems which do not have multiple Media Captures to associate spatially need not use the coordinate model.

- Coordinates can be either in real, physical units (millimeters), have an unknown scale or have no physical scale. Systems which know their physical dimensions (for example professionally installed Telepresence room systems) MUST always provide those real-world measurements. Systems which don't know specific physical dimensions but still know relative distances MUST use 'unknown scale'. 'No scale' is intended to be used where Media Captures from different devices (with potentially different scales) will be forwarded alongside one another (e.g. in the case of a middle box).

  * "Millimeters" means the scale is in millimeters.

  * "Unknown" means the scale is not necessarily millimeters, but the scale is the same for every Capture in the Capture Scene.

  * "No Scale" means the scale could be different for each capture- an MCU provider that advertises two adjacent captures and picks sources (which can change quickly) from different endpoints might use this value; the scale could be different and changing for each capture. But the areas of capture still represent a spatial relation between captures.

- The coordinate system is Cartesian X, Y, Z with the origin at a spatial location of the provider's choosing. The Provider MUST use the same coordinate system with the same scale and origin for all coordinates within the same Capture Scene.

The direction of increasing coordinate values is:
X increases from Camera-Left to Camera-Right
Y increases from front to back
Z increases from low to high (i.e. floor to ceiling)

7. Media Captures and Capture Scenes

This section describes how Providers can describe the content of media to Consumers.

7.1. Media Captures

Media Captures are the fundamental representations of streams that a device can transmit. What a Media Capture actually represents is flexible:

- It can represent the immediate output of a physical source (e.g. camera, microphone) or 'synthetic' source (e.g. laptop computer, DVD player).
- It can represent the output of an audio mixer or video composer
- It can represent a concept such as 'the loudest speaker'
- It can represent a conceptual position such as 'the leftmost stream'

To identify and distinguish between multiple Capture instances Captures have a unique identity. For instance: VC1, VC2 and AC1, AC2, where VC1 and VC2 refer to two different video captures and AC1 and AC2 refer to two different audio captures.

Some key points about Media Captures:

. A Media Capture is of a single media type (e.g. audio or video)
. A Media Capture is defined in a Capture Scene and is given an advertisement unique identity. The identity may be referenced outside the Capture Scene that defines it through a Multiple Content Capture (MCC)
. A Media Capture is associated with one or more Capture Scene Entries
. A Media Capture has exactly one set of spatial information
. A Media Capture can be the source of one or more Capture Encodings

Each Media Capture can be associated with attributes to describe what it represents.

7.1.1. Media Capture Attributes

Media Capture Attributes describe information about the Captures. A Provider can use the Media Capture Attributes to describe the Captures for the benefit of the Consumer in the Advertisement message. Media Capture Attributes include:

- Spatial information, such as point of capture, point on line of capture, and area of capture, all of which, in combination define the capture field of, for example, a camera;
- Capture multiplexing information (composed/switched video, mono/stereo audio, maximum number of simultaneous encodings per Capture and so on); and
- Other descriptive information to help the Consumer choose between captures (description, presentation, view, priority, language, participant information and type).
- Control information for use inside the CLUE protocol suite.

The sub-sections below define the Capture attributes.

7.1.1.1. Point of Capture

The Point of Capture attribute is a field with a single Cartesian (X, Y, Z) point value which describes the spatial location of the capturing device (such as camera).

7.1.1.2. Point on Line of Capture

The Point on Line of Capture attribute is a field with a single Cartesian (X, Y, Z) point value which describes a position in space of a second point on the axis of the capturing device; the first point being the Point of Capture (see above).

Together, the Point of Capture and Point on Line of Capture define an axis of the capturing device, for example the optical axis of a camera. The Media Consumer can use this information to adjust how it renders the received media if it so chooses.

7.1.1.3. Area of Capture

The Area of Capture is a field with a set of four (X, Y, Z) points as a value which describes the spatial location of what is being "captured". By comparing the Area of Capture for different Media Captures within the same Capture Scene a consumer can determine the spatial relationships between them and render them correctly.

The four points MUST be co-planar, forming a quadrilateral, which defines the Plane of Interest for the particular media capture.

If the Area of Capture is not specified, it means the Media Capture is not spatially related to any other Media Capture.

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

7.1.1.4. Mobility of Capture

The Mobility of Capture attribute indicates whether or not the point of capture, line on point of capture, and area of capture values stay the same over time, or are expected to change (potentially frequently). Possible values are static, dynamic, and highly dynamic.

An example for "dynamic" is a camera mounted on a stand which is occasionally hand-carried and placed at different positions in order to provide the best angle to capture a work task. A camera worn by a participant who moves around the room is an example for "highly dynamic". In either case, the effect is that the capture point, capture axis and area of capture change with time.

The capture point of a static capture MUST NOT move for the life of the conference. The capture point of dynamic captures is categorized by a change in position followed by a reasonable period of stability--in the order of magnitude of minutes. High dynamic captures are categorized by a capture point that is constantly moving. If the "area of capture", "capture point" and "line of capture" attributes are included with dynamic or highly dynamic captures they indicate spatial information at the time of the Advertisement.

7.1.1.5. Audio Channel Format

The Audio Channel Format attribute is a field with enumerated values which describes the method of encoding used for audio. A value of ’mono’ means the Audio Capture has one channel. ’stereo’ means the Audio Capture has two audio channels, left and right.

This attribute applies only to Audio Captures. A single stereo capture is different from two mono captures that have a left-right spatial relationship. A stereo capture maps to a single Capture

Encoding, while each mono audio capture maps to a separate Capture Encoding.

7.1.1.6. Max Capture Encodings

The Max Capture Encodings attribute is an optional attribute indicating the maximum number of Capture Encodings that can be simultaneously active for the Media Capture. The number of simultaneous Capture Encodings is also limited by the restrictions of the Encoding Group for the Media Capture.

7.1.1.7. Description

The Description attribute is a human-readable description of the Capture, which could be in multiple languages.

7.1.1.8. Presentation

The Presentation attribute indicates that the capture originates from a presentation device, that is one that provides supplementary information to a conference through slides, video, still images, data etc. Where more information is known about the capture it MAY be expanded hierarchically to indicate the different types of presentation media, e.g. presentation.slides, presentation.image etc.

Note: It is expected that a number of keywords will be defined that provide more detail on the type of presentation.

7.1.1.9. View

The View attribute is a field with enumerated values, indicating what type of view the Capture relates to. The Consumer can use this information to help choose which Media Captures it wishes to receive. The value MUST be one of:

Room - Captures the entire scene

Table - Captures the conference table with seated participants

Individual - Captures an individual participant

Lectern - Captures the region of the lectern including the presenter, for example in a classroom style conference room

Audience - Captures a region showing the audience in a classroom style conference room

7.1.1.10. Language

The language attribute indicates one or more languages used in the content of the Media Capture. Captures MAY be offered in different languages in case of multilingual and/or accessible conferences. A Consumer can use this attribute to differentiate between them and pick the appropriate one.

Note that the Language attribute is defined and meaningful both for audio and video captures. In case of audio captures, the meaning is obvious. For a video capture, "Language" could, for example, be sign interpretation or text.

7.1.1.11. Participant Information

The participant information attribute allows a Provider to provide specific information regarding the conference participants in a Capture. The Provider may gather the information automatically or manually from a variety of sources however the xCard [RFC6351] format is used to convey the information. This allows various information such as Identification information (section 6.2/[RFC6350]), Communication Information (section 6.4/[RFC6350]) and Organizational information (section 6.6/[RFC6350]) to be communicated. A Consumer may then automatically (i.e. via a policy) or manually select Captures based on information about who is in a Capture. It also allows a Consumer to render information regarding the participants or to use it for further processing.

The Provider may supply a minimal set of information or a larger set of information. However it MUST be compliant to [RFC6350] and supply a "VERSION" and "FN" property. A Provider may supply multiple xCards per Capture of any KIND (section 6.1.4/[RFC6350]).

In order to keep CLUE messages compact the Provider SHOULD use a URI to point to any LOGO, PHOTO or SOUND contained in the xCARD rather than transmitting the LOGO, PHOTO or SOUND data in a CLUE message.

7.1.1.12. Participant Type

The participant type attribute indicates the type of participant/s contained in the capture in the conference with respect to the

meeting agenda. As a capture may include multiple participants the attribute may contain multiple value. However values shall not be repeated within the attribute.

An Advertiser associates the participant type with an individual capture when it knows that a particular type is in the capture. If an Advertiser cannot link a particular type with some certainty to a capture then it is not included. A Consumer on reception of a capture with a participant type attribute knows with some certainly that the capture contains that participant type. The capture may contain other participant types but the Advertiser has not been able to determine that this is the case.

The types of Captured participants include:

- Chairman - the participant responsible for running the conference according to the agenda.
- Vice-Chairman - the participant responsible for assisting the chairman in running the meeting.
- Minute Taker - the participant responsible for recording the minutes of the conference 4. Member - the participant has no particular responsibilities with respect to running the meeting.
- Presenter - the participant is scheduled on the agenda to make a presentation in the meeting. Note: This is not related to any "active speaker" functionality.
- Translator - the participant is providing some form of translation or commentary in the meeting.
- Timekeeper - the participant is responsible for maintaining the meeting schedule.

Furthermore the participant type attribute may contain one or more strings allowing the Provider to indicate custom meeting specific roles.

7.1.1.13. Priority

The priority attribute indicates a relative priority between different Media Captures. The Provider sets this priority, and the Consumer MAY use the priority to help decide which captures it wishes to receive.

The "priority" attribute is an integer which indicates a relative priority between Captures. For example it is possible to assign a priority between two presentation Captures that would allow a

remote endpoint to determine which presentation is more important. Priority is assigned at the individual capture level. It represents the Provider's view of the relative priority between Captures with a priority. The same priority number MAY be used across multiple Captures. It indicates they are equally important. If no priority is assigned no assumptions regarding relative important of the Capture can be assumed.

7.1.1.14. Embedded Text

The Embedded Text attribute indicates that a Capture provides embedded textual information. For example the video Capture MAY contain speech to text information composed with the video image. This attribute is only applicable to video Captures and presentation streams with visual information.

7.1.1.15. Related To

The Related To attribute indicates the Capture contains additional complementary information related to another Capture. The value indicates the identity of the other Capture to which this Capture is providing additional information.

For example, a conference can utilize translators or facilitators that provide an additional audio stream (i.e. a translation or description or commentary of the conference). Where multiple captures are available, it may be advantageous for a Consumer to select a complementary Capture instead of or in addition to a Capture it relates to.

7.2. Multiple Content Capture

The MCC indicates that one or more Single Media Captures are contained in one Media Capture. Only one Capture type (i.e. audio, video, etc.) is allowed in each MCC instance. The MCC may contain a reference to the Single Media Captures (which may have their own attributes) as well as attributes associated with the MCC itself. A MCC may also contain other MCCs. The MCC MAY reference Captures from within the Capture Scene that defines it or from other Capture Scenes. No ordering is implied by the order that Captures appear within a MCC. A MCC MAY contain no references to other Captures to indicate that the MCC contains content from multiple sources but no information regarding those sources is given.

One or more MCCs may also be specified in a CSE. This allows an Advertiser to indicate that several MCC captures are used to represent a capture scene. Table 14 provides an example of this case.

As outlined in section 7.1. each instance of the MCC has its own Capture identity i.e. MCC1. It allows all the individual captures contained in the MCC to be referenced by a single MCC identity.

The example below shows the use of a Multiple Content Capture:

Capture Scene #1	
VC1	{attributes}
VC2	{attributes}
VCn	{attributes}
MCC1(VC1,VC2,...VCn)	{attributes}
CSE(MCC1)	

Table 1: Multiple Content Capture concept

This indicates that MCC1 is a single capture that contains the Captures VC1, VC2 and VC3 according to any MCC1 attributes.

7.2.1. MCC Attributes

Attributes may be associated with the MCC instance and the Single Media Captures that the MCC references. A provider should avoid providing conflicting attribute values between the MCC and Single Media Captures. Where there is conflict the attributes of the MCC override any that may be present in the individual captures.

A Provider MAY include as much or as little of the original source Capture information as it requires.

There are MCC specific attributes that MUST only be used with Multiple Content Captures. These are described in the sections below. The attributes described in section 7.1.1. MAY also be used with MCCs.

The spatial related attributes of an MCC indicate its area of capture and point of capture within the scene, just like any other

media capture. The spatial information does not imply anything about how other captures are composed within an MCC.

For example: A virtual scene could be constructed for the MCC capture with two Video Captures with a "MaxCaptures" attribute set to 2 and an "Area of Capture" attribute provided with an overall area. Each of the individual Captures could then also include an "Area of Capture" attribute with a sub-set of the overall area. The Consumer would then know how each capture is related to others within the scene, but not the relative position of the individual captures within the composed capture.

Capture Scene #1	
VC1	AreaofCapture=(0,0,0)(9,0,0) (0,0,9)(9,0,9)
VC2	AreaofCapture=(10,0,0)(19,0,0) (10,0,9)(19,0,9)
MCC1(VC1,VC2)	MaxCaptures=2
	AreaofCapture=(0,0,0)(19,0,0) (0,0,9)(19,0,9)
CSE(MCC1)	

Table 2: Example of MCC and Single Media Capture attributes

The sections below describe the MCC only attributes.

7.2.1.1. Maximum Number of Captures within a MCC

The Maximum Number of Captures MCC attribute indicates the maximum number of individual captures that may appear in a Capture Encoding at a time. The actual number at any given time can be less than this maximum. It may be used to derive how the Single Media Captures within the MCC are composed / switched with regards to space and time.

Max Captures MAY be set to one so that only content related to one of the sources are shown in the MCC Capture Encoding at a time or it may be set to any value up to the total number of Source Media Captures in the MCC.

If this attribute is not set then as default it is assumed that all source content can appear concurrently in the Capture Encoding associated with the MCC.

For example: The use of MaxCaptures equal to 1 on a MCC with three Video Captures VC1, VC2 and VC3 would indicate that the Advertiser in the capture encoding would switch between VC1, VC2 or VC3 as there may be only a maximum of one capture at a time.

7.2.1.2. Policy

The Policy MCC Attribute indicates the criteria that the Provider uses to determine when and/or where media content appears in the Capture Encoding related to the MCC.

The attribute is in the form of a token that indicates the policy and index representing an instance of the policy.

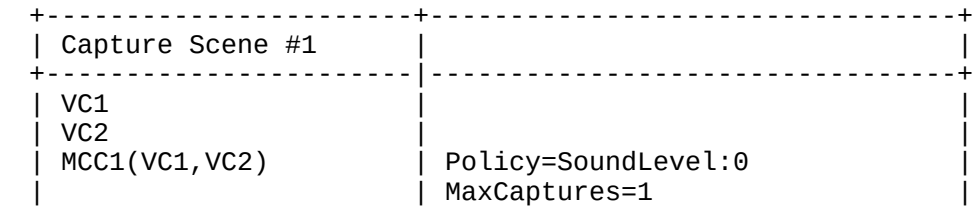
The tokens are:

SoundLevel - This indicates that the content of the MCC is determined by a sound level detection algorithm. For example: the loudest (active) speaker is contained in the MCC.

RoundRobin - This indicates that the content of the MCC is determined by a time based algorithm. For example: the Provider provides content from a particular source for a period of time and then provides content from another source and so on.

An index is used to represent an instance in the policy setting. A index of 0 represents the most current instance of the policy, i.e. the active speaker, 1 represents the previous instance, i.e. the previous active speaker and so on.

The following example shows a case where the Provider provides two media streams, one showing the active speaker and a second stream showing the previous speaker.



MCC2(VC1,VC2) CSE(MCC1,MCC2)	Policy=SoundLevel:1 MaxCaptures=1

Table 3: Example Policy MCC attribute usage

7.2.1.3. Synchronisation Identity

The Synchronisation Identity MCC attribute indicates how the individual captures in multiple MCC captures are synchronised. To indicate that the Capture Encodings associated with MCCs contain captures from the source at the same time a Provider should set the same Synchronisation Identity on each of the concerned MCCs. It is the provider that determines what the source for the Captures is, so a provider can choose how to group together Single Media Captures for the purpose of keeping them synchronized according to the SynchronisationID attribute. For example when the provider is in an MCU it may determine that each separate CLUE endpoint is a remote source of media. The Synchronisation Identity may be used across media types, i.e. to synchronize audio and video related MCCs.

Without this attribute it is assumed that multiple MCCs may provide content from different sources at any particular point in time.

For example:

Capture Scene #1	
VC1	Description=Left
VC2	Description=Centre
VC3	Description=Right
AC1	Description=room
CSE(VC1,VC2,VC3)	
CSE(AC1)	
Capture Scene #2	
VC4	Description=Left
VC5	Description=Centre
VC6	Description=Right
AC2	Description=room
CSE(VC4,VC5,VC6)	

CSE(AC2)	
Capture Scene #3	
VC7	
AC3	
Capture Scene #4	
VC8	
AC4	
Capture Scene #3	
MCC1(VC1,VC4,VC7)	SynchronisationID=1
	MaxCaptures=1
MCC2(VC2,VC5,VC8)	SynchronisationID=1
	MaxCaptures=1
MCC3(VC3,VC6)	MaxCaptures=1
MCC4(AC1,AC2,AC3,AC4)	SynchronisationID=1
	MaxCaptures=1
CSE(MCC1,MCC2,MCC3)	
CSE(MCC4)	

Table 4: Example Synchronisation Identity MCC attribute usage

The above Advertisement would indicate that MCC1, MCC2, MCC3 and MCC4 make up a Capture Scene. There would be four capture encodings (one for each MCC). Because MCC1 and MCC2 have the same SynchronisationID, each encoding from MCC1 and MCC2 respectively would together have content from only Capture Scene 1 or only Capture Scene 2 or the combination of VC7 and VC8 at a particular point in time. In this case the provider has decided the sources to be synchronized are Scene #1, Scene #2, and Scene #3 and #4 together. The encoding from MCC3 would not be synchronised with MCC1 or MCC2. As MCC4 also has the same Synchronisation Identity as MCC1 and MCC2 the content of the audio encoding will be synchronised with the video content.

7.3. Capture Scene

In order for a Provider’s individual Captures to be used effectively by a Consumer, the provider organizes the Captures into one or more Capture Scenes, with the structure and contents of

these Capture Scenes being sent from the Provider to the Consumer in the Advertisement.

A Capture Scene is a structure representing a spatial region containing one or more Capture Devices, each capturing media representing a portion of the region. A Capture Scene includes one or more Capture Scene entries, with each entry including one or more Media Captures. A Capture Scene represents, for example, the video image of a group of people seated next to each other, along with the sound of their voices, which could be represented by some number of VCs and ACs in the Capture Scene Entries. A middle box can also describe in Capture Scenes what it constructs from media Streams it receives.

A Provider MAY advertise one or more Capture Scenes. What constitutes an entire Capture Scene is up to the Provider. A simple Provider might typically use one Capture Scene for participant media (live video from the room cameras) and another Capture Scene for a computer generated presentation. In more complex systems, the use of additional Capture Scenes is also sensible. For example, a classroom may advertise two Capture Scenes involving live video, one including only the camera capturing the instructor (and associated audio), the other including camera(s) capturing students (and associated audio).

A Capture Scene MAY (and typically will) include more than one type of media. For example, a Capture Scene can include several Capture Scene Entries for Video Captures, and several Capture Scene Entries for Audio Captures. A particular Capture MAY be included in more than one Capture Scene Entry.

A provider MAY express spatial relationships between Captures that are included in the same Capture Scene. However, there is not necessarily the same spatial relationship between Media Captures that are in different Capture Scenes. In other words, Capture Scenes can use their own spatial measurement system as outlined above in section 6.

A Provider arranges Captures in a Capture Scene to help the Consumer choose which captures it wants to render. The Capture Scene Entries in a Capture Scene are different alternatives the Provider is suggesting for representing the Capture Scene. The order of Capture Scene Entries within a Capture Scene has no significance. The Media Consumer can choose to receive all Media Captures from one Capture Scene Entry for each media type (e.g.

audio and video), or it can pick and choose Media Captures regardless of how the Provider arranges them in Capture Scene Entries. Different Capture Scene Entries of the same media type are not necessarily mutually exclusive alternatives. Also note that the presence of multiple Capture Scene Entries (with potentially multiple encoding options in each entry) in a given Capture Scene does not necessarily imply that a Provider is able to serve all the associated media simultaneously (although the construction of such an over-rich Capture Scene is probably not sensible in many cases). What a Provider can send simultaneously is determined through the Simultaneous Transmission Set mechanism, described in section 8.

Captures within the same Capture Scene entry MUST be of the same media type - it is not possible to mix audio and video captures in the same Capture Scene Entry, for instance. The Provider MUST be capable of encoding and sending all Captures in a single Capture Scene Entry simultaneously. The order of Captures within a Capture Scene Entry has no significance. A Consumer can decide to receive all the Captures in a single Capture Scene Entry, but a Consumer could also decide to receive just a subset of those captures. A Consumer can also decide to receive Captures from different Capture Scene Entries, all subject to the constraints set by Simultaneous Transmission Sets, as discussed in section 8.

When a Provider advertises a Capture Scene with multiple entries, it is essentially signaling that there are multiple representations of the same Capture Scene available. In some cases, these multiple representations would typically be used simultaneously (for instance a "video entry" and an "audio entry"). In some cases the entries would conceptually be alternatives (for instance an entry consisting of three Video Captures covering the whole room versus an entry consisting of just a single Video Capture covering only the center of a room). In this latter example, one sensible choice for a Consumer would be to indicate (through its Configure and possibly through an additional offer/answer exchange) the Captures of that Capture Scene Entry that most closely matched the Consumer’s number of display devices or screen layout.

The following is an example of 4 potential Capture Scene Entries for an endpoint-style Provider:

1. (VC0, VC1, VC2) - left, center and right camera Video Captures

2. (VC3) - Video Capture associated with loudest room segment

3. (VC4) - Video Capture zoomed out view of all people in the room

4. (AC0) - main audio

The first entry in this Capture Scene example is a list of Video Captures which have a spatial relationship to each other. Determination of the order of these captures (VC0, VC1 and VC2) for rendering purposes is accomplished through use of their Area of Capture attributes. The second entry (VC3) and the third entry (VC4) are alternative representations of the same room's video, which might be better suited to some Consumers' rendering capabilities. The inclusion of the Audio Capture in the same Capture Scene indicates that AC0 is associated with all of those Video Captures, meaning it comes from the same spatial region. Therefore, if audio were to be rendered at all, this audio would be the correct choice irrespective of which Video Captures were chosen.

7.3.1. Capture Scene attributes

Capture Scene Attributes can be applied to Capture Scenes as well as to individual media captures. Attributes specified at this level apply to all constituent Captures. Capture Scene attributes include

- Human-readable description of the Capture Scene, which could be in multiple languages;
- xCard scene information
- Scale information (millimeters, unknown, no scale), as described in Section 6.

7.3.1.1. Scene Information

The Scene information attribute provides information regarding the Capture Scene rather than individual participants. The Provider may gather the information automatically or manually from a variety of sources. The scene information attribute allows a Provider to indicate information such as: organizational or geographic information allowing a Consumer to determine which Capture Scenes are of interest in order to then perform Capture selection. It also allows a Consumer to render information regarding the Scene or to use it for further processing.

As per 7.1.1.11. the xCard format is used to convey this information and the Provider may supply a minimal set of information or a larger set of information.

In order to keep CLUE messages compact the Provider SHOULD use a URI to point to any LOGO, PHOTO or SOUND contained in the xCARD rather than transmitting the LOGO, PHOTO or SOUND data in a CLUE message.

7.3.2. Capture Scene Entry attributes

A Capture Scene can include one or more Capture Scene Entries in addition to the Capture Scene wide attributes described above. Capture Scene Entry attributes apply to the Capture Scene Entry as a whole, i.e. to all Captures that are part of the Capture Scene Entry.

Capture Scene Entry attributes include:

- Human-readable description of the Capture Scene Entry, which could be in multiple languages;

8. Simultaneous Transmission Set Constraints

In many practical cases, a Provider has constraints or limitations on its ability to send Captures simultaneously. One type of limitation is caused by the physical limitations of capture mechanisms; these constraints are represented by a simultaneous transmission set. The second type of limitation reflects the encoding resources available, such as bandwidth or video encoding throughput (macroblocks/second). This type of constraint is captured by encoding groups, discussed below.

Some Endpoints or MCUs can send multiple Captures simultaneously; however sometimes there are constraints that limit which Captures can be sent simultaneously with other Captures. A device may not be able to be used in different ways at the same time. Provider Advertisements are made so that the Consumer can choose one of several possible mutually exclusive usages of the device. This type of constraint is expressed in a Simultaneous Transmission Set, which lists all the Captures of a particular media type (e.g. audio, video, text) that can be sent at the same time. There are different Simultaneous Transmission Sets for each media type in the Advertisement. This is easier to show in an example.

Consider the example of a room system where there are three cameras each of which can send a separate capture covering two persons each- VC0, VC1, VC2. The middle camera can also zoom out (using an optical zoom lens) and show all six persons, VC3. But the middle camera cannot be used in both modes at the same time - it has to either show the space where two participants sit or the whole six seats, but not both at the same time. As a result, VC1 and VC3 cannot be sent simultaneously.

Simultaneous Transmission Sets are expressed as sets of the Media Captures that the Provider could transmit at the same time (though, in some cases, it is not intuitive to do so). If a Multiple Content Capture is included in a Simultaneous Transmission Set it indicates that the Capture Encoding associated with it could be transmitted as the same time as the other Captures within the Simultaneous Transmission Set. It does not imply that the Single Media Captures contained in the Multiple Content Capture could all be transmitted at the same time.

In this example the two simultaneous sets are shown in Table 1. If a Provider advertises one or more mutually exclusive Simultaneous Transmission Sets, then for each media type the Consumer MUST ensure that it chooses Media Captures that lie wholly within one of those Simultaneous Transmission Sets.

Simultaneous Sets
{VC0, VC1, VC2}
{VC0, VC3, VC2}

Table 5: Two Simultaneous Transmission Sets

A Provider OPTIONALLY can include the simultaneous sets in its provider Advertisement. These simultaneous set constraints apply across all the Capture Scenes in the Advertisement. It is a syntax conformance requirement that the simultaneous transmission sets MUST allow all the media captures in any particular Capture Scene Entry to be used simultaneously.

For shorthand convenience, a Provider MAY describe a Simultaneous Transmission Set in terms of Capture Scene Entries and Capture Scenes. If a Capture Scene Entry is included in a Simultaneous Transmission Set, then all Media Captures in the Capture Scene

Entry are included in the Simultaneous Transmission Set. If a Capture Scene is included in a Simultaneous Transmission Set, then all its Capture Scene Entries (of the corresponding media type) are included in the Simultaneous Transmission Set. The end result reduces to a set of Media Captures in either case.

If an Advertisement does not include Simultaneous Transmission Sets, then the Provider MUST be able to provide all Capture Scenes simultaneously. If multiple capture Scene Entries are in a Capture Scene then the Consumer chooses at most one Capture Scene Entry per Capture Scene for each media type.

If an Advertisement includes multiple Capture Scene Entries in a Capture Scene then the Consumer MAY choose one Capture Scene Entry for each media type, or MAY choose individual Captures based on the Simultaneous Transmission Sets.

9. Encodings

Individual encodings and encoding groups are CLUE's mechanisms allowing a Provider to signal its limitations for sending Captures, or combinations of Captures, to a Consumer. Consumers can map the Captures they want to receive onto the Encodings, with encoding parameters they want. As for the relationship between the CLUE-specified mechanisms based on Encodings and the SIP Offer-Answer exchange, please refer to section 4.

9.1. Individual Encodings

An Individual Encoding represents a way to encode a Media Capture to become a Capture Encoding, to be sent as an encoded media stream from the Provider to the Consumer. An Individual Encoding has a set of parameters characterizing how the media is encoded.

Different media types have different parameters, and different encoding algorithms may have different parameters. An Individual Encoding can be assigned to at most one Capture Encoding at any given time.

The parameters of an Individual Encoding represent the maximum values for certain aspects of the encoding. A particular instantiation into a Capture Encoding MAY use lower values than these maximums if that is applicable for the media in question. For example, most video codec specifications require a conformant decoder to decode resolutions and frame rates smaller than what has

been negotiated as a maximum, so downgrading the CLUE maximum values for macroblocks/second is appropriate. On the other hand, downgrading the sample rate of G.711 audio below 8kHz is not specified in G.711 and therefore not applicable in the sense described here.

Individual Encoding parameters are represented in SDP [RFC4566], not in CLUE messages. For example, for a video encoding using H.26x compression technologies, this can include parameters such as:

- Maximum bandwidth;
- Maximum picture size in pixels;
- Maxmimum number of pixels to be processed per second;

The bandwidth parameter is the only one that specifically relates to a CLUE Advertisement, as it can be further constrained by the maximum group bandwidth in an Encoding Group.

9.2. Encoding Group

An Encoding Group includes a set of one or more Individual Encodings, and parameters that apply to the group as a whole. By grouping multiple individual Encodings together, an Encoding Group describes additional constraints on bandwidth for the group.

The Encoding Group data structure contains:

- Maximum bitrate for all encodings in the group combined;
- A list of identifiers for audio and video encodings, respectively, belonging to the group.

When the Individual Encodings in a group are instantiated into Capture Encodings, each Capture Encoding has a bitrate that MUST be less than or equal to the max bitrate for the particular individual encoding. The "maximum bitrate for all encodings in the group" parameter gives the additional restriction that the sum of all the individual capture encoding bitrates MUST be less than or equal to the this group value.

The following diagram illustrates one example of the structure of a media provider’s Encoding Groups and their contents.

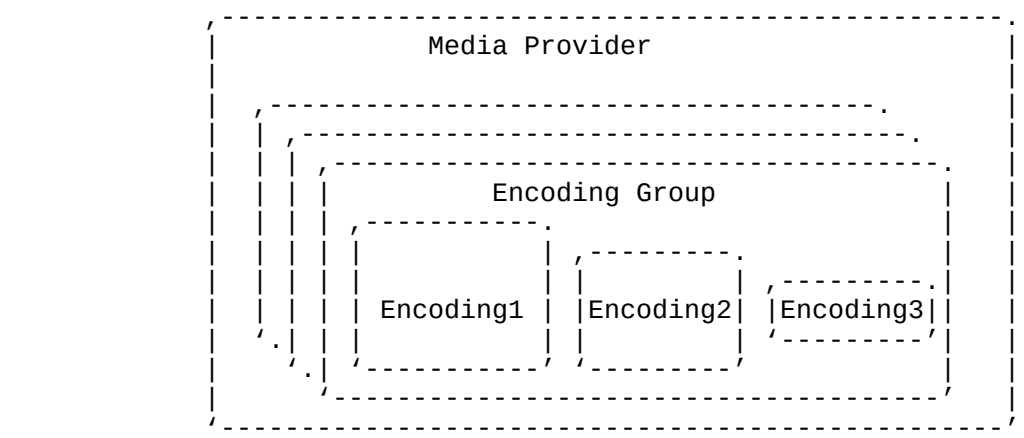


Figure 3: Encoding Group Structure

A Provider advertises one or more Encoding Groups. Each Encoding Group includes one or more Individual Encodings. Each Individual Encoding can represent a different way of encoding media. For example one Individual Encoding may be 1080p60 video, another could be 720p30, with a third being CIF, all in, for example, H.264 format.
While a typical three codec/display system might have one Encoding Group per "codec box" (physical codec, connected to one camera and one screen), there are many possibilities for the number of Encoding Groups a Provider may be able to offer and for the encoding values in each Encoding Group.

There is no requirement for all Encodings within an Encoding Group to be instantiated at the same time.

9.3. Associating Captures with Encoding Groups

Each Media Capture MAY be associated with at least one Encoding Group, which is used to instantiate that Capture into one or more Capture Encodings. Typically MCCs are assigned an Encoding Group and thus become a Capture Encoding. The Captures (including other MCCs) referenced by the MCC do not need to be assigned to an Encoding Group. This means that all the Media Captures referenced by the MCC will appear in the Capture Encoding according to any MCC attributes. This allows an Advertiser to specify Capture attributes associated with the Media Captures without the need to provide an individual Capture Encoding for each of the inputs.

If an Encoding Group is assigned to a Media Capture referenced by the MCC it indicates that this Capture may also have an individual Capture Encoding.

For example:

Capture Scene #1	
VC1	EncodeGroupID=1
VC2	
MCC1(VC1,VC2)	EncodeGroupID=2
CSE(VC1)	
CSE(MCC1)	

Table 6: Example usage of Encoding with MCC and source Captures

This would indicate that VC1 may be sent as its own Capture Encoding from EncodeGroupID=1 or that it may be sent as part of a Capture Encoding from EncodeGroupID=2 along with VC2.

More than one Capture MAY use the same Encoding Group.

The maximum number of streams that can result from a particular Encoding Group constraint is equal to the number of individual Encodings in the group. The actual number of Capture Encodings used at any time MAY be less than this maximum. Any of the Captures that use a particular Encoding Group can be encoded according to any of the Individual Encodings in the group. If there are multiple Individual Encodings in the group, then the Consumer can configure the Provider, via a Configure message, to encode a single Media Capture into multiple different Capture Encodings at the same time, subject to the Max Capture Encodings constraint, with each capture encoding following the constraints of a different Individual Encoding.

It is a protocol conformance requirement that the Encoding Groups MUST allow all the Captures in a particular Capture Scene Entry to be used simultaneously.

10. Consumer’s Choice of Streams to Receive from the Provider

After receiving the Provider’s Advertisement message (that includes media captures and associated constraints), the Consumer composes

its reply to the Provider in the form of a Configure message. The Consumer is free to use the information in the Advertisement as it chooses, but there are a few obviously sensible design choices, which are outlined below.

If multiple Providers connect to the same Consumer (i.e. in a n MCU-less multiparty call), it is the responsibility of the Consumer to compose Configures for each Provider that both fulfill each Provider's constraints as expressed in the Advertisement, as well as its own capabilities.

In an MCU-based multiparty call, the MCU can logically terminate the Advertisement/Configure negotiation in that it can hide the characteristics of the receiving endpoint and rely on its own capabilities (transcoding/transrating/...) to create Media Streams that can be decoded at the Endpoint Consumers. The timing of an MCU's sending of Advertisements (for its outgoing ports) and Configures (for its incoming ports, in response to Advertisements received there) is up to the MCU and implementation dependent.

As a general outline, A Consumer can choose, based on the Advertisement it has received, which Captures it wishes to receive, and which Individual Encodings it wants the Provider to use to encode the Captures.

On receipt of an Advertisement with an MCC the Consumer treats the MCC as per other non-MCC Captures with the following differences:

- The Consumer would understand that the MCC is a Capture that includes the referenced individual Captures and that these individual Captures are delivered as part of the MCC's Capture Encoding.

- The Consumer may utilise any of the attributes associated with the referenced individual Captures and any Capture Scene attributes from where the individual Captures were defined to choose Captures and for rendering decisions.

- The Consumer may or may not choose to receive all the indicated captures. Therefore it can choose to receive a sub-set ofCaptures indicated by the MCC.

For example if the Consumer receives:

MCC1(VC1,VC2,VC3){attributes}

A Consumer could choose all the Captures within a MCCs however if the Consumer determines that it doesn't want VC3 it can return MCC1(VC1,VC2). If it wants all the individual Captures then it returns only the MCC identity (i.e. MCC1). If the MCC in the advertisement does not reference any individual captures, then the Consumer cannot choose what is included in the MCC, it is up to the Provider to decide.

A Configure Message includes a list of Capture Encodings. These are the Capture Encodings the Consumer wishes to receive from the Provider. Each Capture Encoding refers to one Media Capture, one Individual Encoding, and includes the encoding parameter values. A Configure Message does not include references to Capture Scenes or Capture Scene Entries.

For each Capture the Consumer wants to receive, it configures one or more of the encodings in that capture's encoding group. The Consumer does this by telling the Provider, in its Configure Message, parameters such as the resolution, frame rate, bandwidth, etc. for each Capture Encodings for its chosen Captures. Upon receipt of this Configure from the Consumer, common knowledge is established between Provider and Consumer regarding sensible choices for the media streams and their parameters. The setup of the actual media channels, at least in the simplest case, is left to a following offer-answer exchange. Optimized implementations MAY speed up the reaction to the offer-answer exchange by reserving the resources at the time of finalization of the CLUE handshake.

CLUE advertisements and configure messages don't necessarily require a new SDP offer-answer for every CLUE message exchange. But the resulting encodings sent via RTP must conform to the most recent SDP offer-answer result.

In order to meaningfully create and send an initial Configure, the Consumer needs to have received at least one Advertisement from the Provider.

In addition, the Consumer can send a Configure at any time during the call. The Configure MUST be valid according to the most recently received Advertisement. The Consumer can send a Configure either in response to a new Advertisement from the Provider or on its own, for example because of a local change in conditions (people leaving the room, connectivity changes, multipoint related considerations).

When choosing which Media Streams to receive from the Provider, and the encoding characteristics of those Media Streams, the Consumer advantageously takes several things into account: its local preference, simultaneity restrictions, and encoding limits.

10.1. Local preference

A variety of local factors influence the Consumer's choice of Media Streams to be received from the Provider:

- if the Consumer is an Endpoint, it is likely that it would choose, where possible, to receive video and audio Captures that match the number of display devices and audio system it has
- if the Consumer is a middle box such as an MCU, it MAY choose to receive loudest speaker streams (in order to perform its own media composition) and avoid pre-composed video Captures
- user choice (for instance, selection of a new layout) MAY result in a different set of Captures, or different encoding characteristics, being required by the Consumer

10.2. Physical simultaneity restrictions

Often there are physical simultaneity constraints of the Provider that affect the Provider's ability to simultaneously send all of the captures the Consumer would wish to receive. For instance, a middle box such as an MCU, when connected to a multi-camera room system, might prefer to receive both individual video streams of the people present in the room and an overall view of the room from a single camera. Some Endpoint systems might be able to provide both of these sets of streams simultaneously, whereas others might not (if the overall room view were produced by changing the optical zoom level on the center camera, for instance).

10.3. Encoding and encoding group limits

Each of the Provider's encoding groups has limits on bandwidth and computational complexity, and the constituent potential encodings have limits on the bandwidth, computational complexity, video frame rate, and resolution that can be provided. When choosing the Captures to be received from a Provider, a Consumer device MUST ensure that the encoding characteristics requested for each individual Capture fits within the capability of the encoding it

is being configured to use, as well as ensuring that the combined encoding characteristics for Captures fit within the capabilities of their associated encoding groups. In some cases, this could cause an otherwise "preferred" choice of capture encodings to be passed over in favor of different Capture Encodings--for instance, if a set of three Captures could only be provided at a low resolution then a three screen device could switch to favoring a single, higher quality, Capture Encoding.

11. Extensibility

One important characteristics of the Framework is its extensibility. Telepresence is a relatively new industry and while we can foresee certain directions, we also do not know everything about how it will develop. The standard for interoperability and handling multiple streams must be future-proof. The framework itself is inherently extensible through expanding the data model types. For example:

- Adding more types of media, such as telemetry, can done by defining additional types of Captures in addition to audio and video.
- Adding new functionalities, such as 3-D, say, may require additional attributes describing the Captures.
- Adding a new codecs, such as H.265, can be accomplished by defining new encoding variables.

The infrastructure is designed to be extended rather than requiring new infrastructure elements. Extension comes through adding to defined types.

12. Examples - Using the Framework (Informative)

This section gives some examples, first from the point of view of the Provider, then the Consumer.

12.1. Provider Behavior

This section shows some examples in more detail of how a Provider can use the framework to represent a typical case for telepresence rooms. First an endpoint is illustrated, then an MCU case is shown.

12.1.1. Three screen Endpoint Provider

Consider an Endpoint with the following description:

3 cameras, 3 displays, a 6 person table

- Each camera can provide one Capture for each 1/3 section of the table
- A single Capture representing the active speaker can be provided (voice activity based camera selection to a given encoder input port implemented locally in the Endpoint)
- A single Capture representing the active speaker with the other 2 Captures shown picture in picture within the stream can be provided (again, implemented inside the endpoint)
- A Capture showing a zoomed out view of all 6 seats in the room can be provided

The audio and video Captures for this Endpoint can be described as follows.

Video Captures:

- VC0- (the camera-left camera stream), encoding group=EG0, switched=false, view=table
- VC1- (the center camera stream), encoding group=EG1, switched=false, view=table
- VC2- (the camera-right camera stream), encoding group=EG2, switched=false, view=table
- VC3- (the loudest panel stream), encoding group=EG1, switched=true, view=table
- VC4- (the loudest panel stream with PiPs), encoding group=EG1, composed=true, switched=true, view=room
- VC5- (the zoomed out view of all people in the room), encoding group=EG1, composed=false, switched=false, view=room
- VC6- (presentation stream), encoding group=EG1, presentation, switched=false

The following diagram is a top view of the room with 3 cameras, 3 displays, and 6 seats. Each camera is capturing 2 people. The six seats are not all in a straight line.

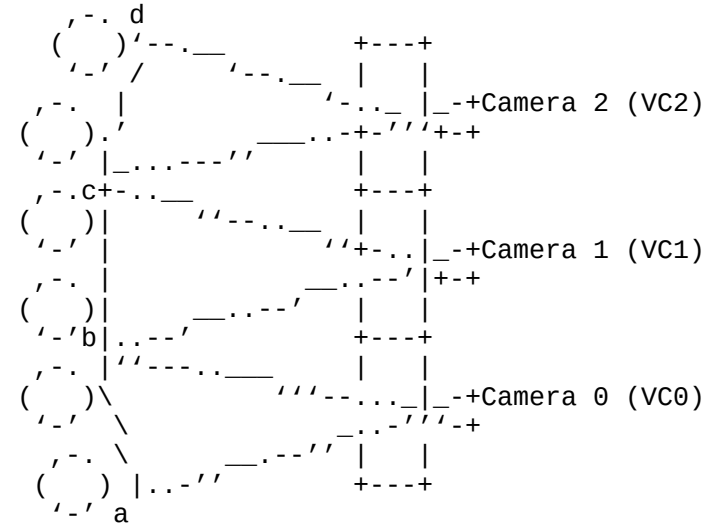


Figure 4: Room Layout

The two points labeled b and c are intended to be at the midpoint between the seating positions, and where the fields of view of the cameras intersect.

The plane of interest for VC0 is a vertical plane that intersects points ’a’ and ’b’.

The plane of interest for VC1 intersects points ’b’ and ’c’. The plane of interest for VC2 intersects points ’c’ and ’d’.

This example uses an area scale of millimeters.

Areas of capture:

	bottom left	bottom right	top left	top right
VC0	(-2011,2850,0)	(-673,3000,0)	(-2011,2850,757)	(-673,3000,757)
VC1	(-673,3000,0)	(673,3000,0)	(-673,3000,757)	(673,3000,757)
VC2	(673,3000,0)	(2011,2850,0)	(673,3000,757)	(2011,3000,757)
VC3	(-2011,2850,0)	(2011,2850,0)	(-2011,2850,757)	(2011,3000,757)
VC4	(-2011,2850,0)	(2011,2850,0)	(-2011,2850,757)	(2011,3000,757)
VC5	(-2011,2850,0)	(2011,2850,0)	(-2011,2850,757)	(2011,3000,757)
VC6	none			

Points of capture:
VC0 (-1678,0,800)
VC1 (0,0,800)
VC2 (1678,0,800)
VC3 none
VC4 none
VC5 (0,0,800)
VC6 none

In this example, the right edge of the VC0 area lines up with the left edge of the VC1 area. It doesn’t have to be this way. There could be a gap or an overlap. One additional thing to note for this example is the distance from a to b is equal to the distance from b to c and the distance from c to d. All these distances are 1346 mm. This is the planar width of each area of capture for VC0, VC1, and VC2.

Note the text in parentheses (e.g. "the camera-left camera stream") is not explicitly part of the model, it is just explanatory text for this example, and is not included in the

model with the media captures and attributes. Also, the "composed" boolean attribute doesn't say anything about how a capture is composed, so the media consumer can't tell based on this attribute that VC4 is composed of a "loudest panel with PiPs".

Audio Captures:

- AC0 (camera-left), encoding group=EG3, content=main, channel format=mono
- AC1 (camera-right), encoding group=EG3, content=main, channel format=mono
- AC2 (center) encoding group=EG3, content=main, channel format=mono
- AC3 being a simple pre-mixed audio stream from the room (mono), encoding group=EG3, content=main, channel format=mono
- AC4 audio stream associated with the presentation video (mono) encoding group=EG3, content=slides, channel format=mono

Areas of capture:

	bottom left	bottom right	top left	top right
AC0	(-2011,2850,0)	(-673,3000,0)	(-2011,2850,757)	(-673,3000,757)
AC1	(673,3000,0)	(2011,2850,0)	(673,3000,757)	(2011,3000,757)
AC2	(-673,3000,0)	(673,3000,0)	(-673,3000,757)	(673,3000,757)
AC3	(-2011,2850,0)	(2011,2850,0)	(-2011,2850,757)	(2011,3000,757)
AC4	none			

The physical simultaneity information is:

Simultaneous transmission set #1 {VC0, VC1, VC2, VC3, VC4, VC6}

Simultaneous transmission set #2 {VC0, VC2, VC5, VC6}

This constraint indicates it is not possible to use all the VCs at the same time. VC5 cannot be used at the same time as VC1 or VC3 or VC4. Also, using every member in the set simultaneously may not make sense - for example VC3(loudest) and VC4 (loudest with PIP). (In addition, there are encoding constraints that make choosing all of the VCs in a set impossible. VC1, VC3, VC4, VC5,

VC6 all use EG1 and EG1 has only 3 ENCs. This constraint shows up in the encoding groups, not in the simultaneous transmission sets.)

In this example there are no restrictions on which audio captures can be sent simultaneously.

Encoding Groups:

This example has three encoding groups associated with the video captures. Each group can have 3 encodings, but with each potential encoding having a progressively lower specification. In this example, 1080p60 transmission is possible (as ENC0 has a maxPps value compatible with that). Significantly, as up to 3 encodings are available per group, it is possible to transmit some video captures simultaneously that are not in the same entry in the capture scene. For example VC1 and VC3 at the same time.

It is also possible to transmit multiple capture encodings of a single video capture. For example VC0 can be encoded using ENC0 and ENC1 at the same time, as long as the encoding parameters satisfy the constraints of ENC0, ENC1, and EG0, such as one at 4000000 bps and one at 2000000 bps.

```
encodeGroupID=EG0, maxGroupBandwidth=6000000
    encodeID=ENC0, maxWidth=1920, maxHeight=1088, maxFrameRate=60,
                   maxPps=124416000, maxBandwidth=4000000
    encodeID=ENC1, maxWidth=1280, maxHeight=720, maxFrameRate=30,
                   maxPps=27648000, maxBandwidth=4000000
    encodeID=ENC2, maxWidth=960, maxHeight=544, maxFrameRate=30,
                   maxPps=15552000, maxBandwidth=4000000
encodeGroupID=EG1  maxGroupBandwidth=6000000
    encodeID=ENC3, maxWidth=1920, maxHeight=1088, maxFrameRate=60,
                   maxPps=124416000, maxBandwidth=4000000
    encodeID=ENC4, maxWidth=1280, maxHeight=720, maxFrameRate=30,
                   maxPps=27648000, maxBandwidth=4000000
    encodeID=ENC5, maxWidth=960, maxHeight=544, maxFrameRate=30,
                   maxPps=15552000, maxBandwidth=4000000
encodeGroupID=EG2  maxGroupBandwidth=6000000
    encodeID=ENC6, maxWidth=1920, maxHeight=1088, maxFrameRate=60,
                   maxPps=124416000, maxBandwidth=4000000
    encodeID=ENC7, maxWidth=1280, maxHeight=720, maxFrameRate=30,
                   maxPps=27648000, maxBandwidth=4000000
    encodeID=ENC8, maxWidth=960, maxHeight=544, maxFrameRate=30,
                   maxPps=15552000, maxBandwidth=4000000
```

Figure 5: Example Encoding Groups for Video

For audio, there are five potential encodings available, so all five audio captures can be encoded at the same time.

```
encodeGroupID=EG3, maxGroupBandwidth=320000
    encodeID=ENC9, maxBandwidth=64000
    encodeID=ENC10, maxBandwidth=64000
    encodeID=ENC11, maxBandwidth=64000
    encodeID=ENC12, maxBandwidth=64000
    encodeID=ENC13, maxBandwidth=64000
```

Figure 6: Example Encoding Group for Audio

Capture Scenes:

The following table represents the capture scenes for this provider. Recall that a capture scene is composed of alternative capture scene entries covering the same spatial region. Capture Scene #1 is for the main people captures, and Capture Scene #2 is for presentation.

Each row in the table is a separate Capture Scene Entry

Capture Scene #1
VC0, VC1, VC2
VC3
VC4
VC5
AC0, AC1, AC2
AC3

Capture Scene #2
VC6
AC4

Table 7: Example Capture Scene Entries

Different capture scenes are unique to each other, non-overlapping. A consumer can choose an entry from each capture scene. In this case the three captures VC0, VC1, and VC2 are one way of representing the video from the endpoint. These three captures should appear adjacent next to each other. Alternatively, another way of representing the Capture Scene is with the capture VC3, which automatically shows the person who is talking. Similarly for the VC4 and VC5 alternatives.

As in the video case, the different entries of audio in Capture Scene #1 represent the "same thing", in that one way to receive the audio is with the 3 audio captures (AC0, AC1, AC2), and another way is with the mixed AC3. The Media Consumer can choose an audio capture entry it is capable of receiving.

The spatial ordering is understood by the media capture attributes Area of Capture and Point of Capture.

A Media Consumer would likely want to choose a capture scene entry to receive based in part on how many streams it can simultaneously receive. A consumer that can receive three people streams would probably prefer to receive the first entry of Capture Scene #1 (VC0, VC1, VC2) and not receive the other entries. A consumer that can receive only one people stream would probably choose one of the other entries.

If the consumer can receive a presentation stream too, it would also choose to receive the only entry from Capture Scene #2 (VC6).

12.1.2. Encoding Group Example

This is an example of an encoding group to illustrate how it can express dependencies between encodings.

```
encodeGroupID=EG0 maxGroupBandwidth=6000000
    encodeID=VIDENC0, maxWidth=1920, maxHeight=1088,
      maxFrameRate=60, maxPps=62208000, maxBandwidth=4000000
    encodeID=VIDENC1, maxWidth=1920, maxHeight=1088,
      maxFrameRate=60, maxPps=62208000, maxBandwidth=4000000
    encodeID=AUDENC0, maxBandwidth=96000
    encodeID=AUDENC1, maxBandwidth=96000
    encodeID=AUDENC2, maxBandwidth=96000
```

Here, the encoding group is EG0. Although the encoding group is capable of transmitting up to 6Mbit/s, no individual video encoding can exceed 4Mbit/s.

This encoding group also allows up to 3 audio encodings, AUDENC<0-2>. It is not required that audio and video encodings reside within the same encoding group, but if so then the group's overall maxBandwidth value is a limit on the sum of all audio and video encodings configured by the consumer. A system that does not wish or need to combine bandwidth limitations in this way should instead use separate encoding groups for audio and video in order for the bandwidth limitations on audio and video to not interact.

Audio and video can be expressed in separate encoding groups, as in this illustration.

```
encodeGroupID=EG0 maxGroupBandwidth=6000000
    encodeID=VIDENC0, maxWidth=1920, maxHeight=1088,
      maxFrameRate=60, maxPps=62208000, maxBandwidth=4000000
    encodeID=VIDENC1, maxWidth=1920, maxHeight=1088,
      maxFrameRate=60, maxPps=62208000, maxBandwidth=4000000
encodeGroupID=EG1 maxGroupBandwidth=500000
    encodeID=AUDENC0, maxBandwidth=96000
    encodeID=AUDENC1, maxBandwidth=96000
    encodeID=AUDENC2, maxBandwidth=96000
```

12.1.3. The MCU Case

This section shows how an MCU might express its Capture Scenes, intending to offer different choices for consumers that can handle different numbers of streams. A single audio capture stream is provided for all single and multi-screen configurations that can be associated (e.g. lip-synced) with any combination of video captures at the consumer.

Capture Scene #1	
VC0	VC for a single screen consumer
VC1, VC2	VCs for a two screen consumer
VC3, VC4, VC5	VCs for a three screen consumer
VC6, VC7, VC8, VC9	VCs for a four screen consumer
AC0	AC representing all participants
CSE(VC0)	
CSE(VC1,VC2)	

CSE(VC3,VC4,VC5)	
CSE(VC6,VC7,VC8,VC9)	
CSE(AC0)	

Table 8: MCU main Capture Scenes

If / when a presentation stream becomes active within the conference the MCU might re-advertise the available media as:

Capture Scene #2	note
VC10	video capture for presentation
AC1	presentation audio to accompany VC10
CSE(VC10)	
CSE(AC1)	

Table 9: MCU presentation Capture Scene

12.2. Media Consumer Behavior

This section gives an example of how a Media Consumer might behave when deciding how to request streams from the three screen endpoint described in the previous section.

The receive side of a call needs to balance its requirements, based on number of screens and speakers, its decoding capabilities and available bandwidth, and the provider's capabilities in order to optimally configure the provider's streams. Typically it would want to receive and decode media from each Capture Scene advertised by the Provider.

A sane, basic, algorithm might be for the consumer to go through each Capture Scene in turn and find the collection of Video Captures that best matches the number of screens it has (this might include consideration of screens dedicated to presentation video display rather than "people" video) and then decide between alternative entries in the video Capture Scenes based either on hard-coded preferences or user choice. Once this choice has been made, the consumer would then decide how to configure the provider's encoding groups in order to make best use of the available network bandwidth and its own decoding capabilities.

12.2.1. One screen Media Consumer

VC3, VC4 and VC5 are all different entries by themselves, not grouped together in a single entry, so the receiving device should choose between one of those. The choice would come down to whether to see the greatest number of participants simultaneously at roughly equal precedence (VC5), a switched view of just the loudest region (VC3) or a switched view with PiPs (VC4). An endpoint device with a small amount of knowledge of these differences could offer a dynamic choice of these options, in-call, to the user.

12.2.2. Two screen Media Consumer configuring the example

Mixing systems with an even number of screens, "2n", and those with "2n+1" cameras (and vice versa) is always likely to be the problematic case. In this instance, the behavior is likely to be determined by whether a "2 screen" system is really a "2 decoder" system, i.e., whether only one received stream can be displayed per screen or whether more than 2 streams can be received and spread across the available screen area. To enumerate 3 possible behaviors here for the 2 screen system when it learns that the far end is "ideally" expressed via 3 capture streams:

1. Fall back to receiving just a single stream (VC3, VC4 or VC5 as per the 1 screen consumer case above) and either leave one screen blank or use it for presentation if / when a presentation becomes active.

2. Receive 3 streams (VC0, VC1 and VC2) and display across 2 screens (either with each capture being scaled to 2/3 of a screen and the center capture being split across 2 screens) or, as would be necessary if there were large bezels on the screens, with each stream being scaled to 1/2 the screen width and height and there being a 4th "blank" panel. This 4th panel could potentially be used for any presentation that became active during the call.

3. Receive 3 streams, decode all 3, and use control information indicating which was the most active to switch between showing the left and center streams (one per screen) and the center and right streams.

For an endpoint capable of all 3 methods of working described above, again it might be appropriate to offer the user the choice of display mode.

12.2.3. Three screen Media Consumer configuring the example

This is the most straightforward case - the Media Consumer would look to identify a set of streams to receive that best matched its available screens and so the VC0 plus VC1 plus VC2 should match optimally. The spatial ordering would give sufficient information for the correct video capture to be shown on the correct screen, and the consumer would either need to divide a single encoding group's capability by 3 to determine what resolution and frame rate to configure the provider with or to configure the individual video captures' encoding groups with what makes most sense (taking into account the receive side decode capabilities, overall call bandwidth, the resolution of the screens plus any user preferences such as motion vs sharpness).

12.3. Multipoint Conference utilizing Multiple Content Captures

The use of MCCs allows the MCU to construct outgoing Advertisements describing complex and media switching and composition scenarios. The following sections provide several examples.

Note: In the examples the identities of the CLUE elements (e.g. Captures, Capture Scene) in the incoming Advertisements overlap. This is because there is no co-ordination between the endpoints. The MCU is responsible for making these unique in the outgoing advertisement.

12.3.1. Single Media Captures and MCC in the same Advertisement

Four endpoints are involved in a Conference where CLUE is used. An MCU acts as a middlebox between the endpoints with a CLUE channel between each endpoint and the MCU. The MCU receives the following Advertisements.

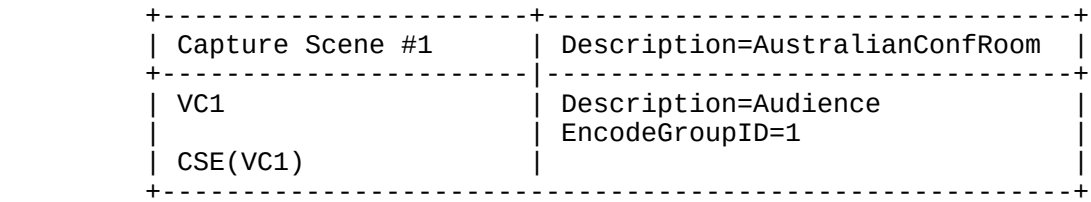


Table 10: Advertisement received from Endpoint A

Capture Scene #1	Description=ChinaConfRoom
VC1	Description=Speaker
	EncodeGroupID=1
VC2	Description=Audience
	EncodeGroupID=1
CSE(VC1, VC2)	

Table 11: Advertisement received from Endpoint B

Capture Scene #1	Description=USAConfRoom
VC1	Description=Audience
	EncodeGroupID=1
CSE(VC1)	

Table 12: Advertisement received from Endpoint C

Note: Endpoint B above indicates that it sends two streams.

If the MCU wanted to provide a Multiple Content Capture containing a round robin switched view of the audience from the 3 endpoints and the speaker it could construct the following advertisement:

Advertisement sent to Endpoint F

Capture Scene #1	Description=AustralianConfRoom
VC1	Description=Audience
CSE(VC1)	
Capture Scene #2	Description=ChinaConfRoom
VC2	Description=Speaker
VC3	Description=Audience
CSE(VC2, VC3)	
Capture Scene #3	Description=USAConfRoom

VC4 CSE(VC4)	Description=Audience
Capture Scene #4	
MCC1(VC1,VC2,VC3,VC4) CSE(MCC1)	Policy=RoundRobin:1 MaxCaptures=1 EncodingGroup=1

Table 13: Advertisement sent to Endpoint F - One Encoding

Alternatively if the MCU wanted to provide the speaker as one media stream and the audiences as another it could assign an encoding group to VC2 in Capture Scene 2 and provide a CSE in Capture Scene #4 as per the example below.

Advertisement sent to Endpoint F

Capture Scene #1	Description=AustralianConfRoom
VC1 CSE(VC1)	Description=Audience
Capture Scene #2	Description=ChinaConfRoom
VC2 VC3 CSE(VC2, VC3)	Description=Speaker EncodingGroup=1 Description=Audience
Capture Scene #3	Description=USAConfRoom
VC4 CSE(VC4)	Description=Audience
Capture Scene #4	
MCC1(VC1,VC3,VC4) MCC2(VC2)	Policy=RoundRobin:1 MaxCaptures=1 EncodingGroup=1 MaxCaptures=1

	EncodingGroup=1	
CSE2(MCC1,MCC2)		

Table 14: Advertisement sent to Endpoint F - Two Encodings

Therefore a Consumer could choose whether or not to have a separate speaker related stream and could choose which endpoints to see. If it wanted the second stream but not the Australian conference room it could indicate the following captures in the Configure message:

MCC1(VC3,VC4)	Encoding
VC2	Encoding

Table 15: MCU case: Consumer Response

12.3.2. Several MCCs in the same Advertisement

Multiple MCCs can be used where multiple streams are used to carry media from multiple endpoints. For example:

A conference has three endpoints D, E and F. Each end point has three video captures covering the left, middle and right regions of each conference room. The MCU receives the following advertisements from D and E.

Capture Scene #1	Description=AustralianConfRoom
VC1	CaptureArea=Left
	EncodingGroup=1
VC2	CaptureArea=Centre
	EncodingGroup=1
VC3	CaptureArea=Right
	EncodingGroup=1
CSE(VC1,VC2,VC3)	

Table 16: Advertisement received from Endpoint D

Capture Scene #1	Description=ChinaConfRoom
VC1	CaptureArea=Left

		EncodingGroup=1
VC2		CaptureArea=Centre
		EncodingGroup=1
VC3		CaptureArea=Right
		EncodingGroup=1
CSE(VC1,VC2,VC3)		

Table 17: Advertisement received from Endpoint E

The MCU wants to offer Endpoint F three Capture Encodings. Each Capture Encoding would contain all the Captures from either Endpoint D or Endpoint E depending based on the active speaker. The MCU sends the following Advertisement:

Capture Scene #1	Description=AustralianConfRoom
VC1	
VC2	
VC3	
CSE(VC1,VC2,VC3)	
Capture Scene #2	Description=ChinaConfRoom
VC4	
VC5	
VC6	
CSE(VC4,VC5,VC6)	
Capture Scene #3	
MCC1(VC1,VC4)	CaptureArea=Left
	MaxCaptures=1
	SynchronisationID=1
	EncodingGroup=1
MCC2(VC2,VC5)	CaptureArea=Centre
	MaxCaptures=1
	SynchronisationID=1
	EncodingGroup=1
MCC3(VC3,VC6)	CaptureArea=Right
	MaxCaptures=1
	SynchronisationID=1
	EncodingGroup=1

| CSE(MCC1,MCC2,MCC3) | |
+=====================+=================================+

Table 17: Advertisement received from Endpoint E

12.3.3. Heterogeneous conference with switching and composition

Consider a conference between endpoints with the following characteristics:

Endpoint A - 4 screens, 3 cameras

Endpoint B - 3 screens, 3 cameras

Endpoint C - 3 screens, 3 cameras

Endpoint D - 3 screens, 3 cameras

Endpoint E - 1 screen, 1 camera

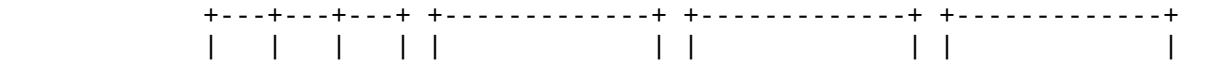
Endpoint F - 2 screens, 1 cameras

Endpoint G - 1 screen, 1 camera

This example focuses on what the user in one of the 3-camera multi-screen endpoints sees. Call this person User A, at Endpoint A. There are 4 large display screens at Endpoint A. Whenever somebody at another site is speaking, all the video captures from that endpoint are shown on the large screens. If the talker is at a 3-camera site, then the video from those 3 cameras fills 3 of the screens. If the talker is at a single-camera site, then video from that camera fills one of the screens, while the other screens show video from other single-camera endpoints.

User A hears audio from the 4 loudest talkers.

User A can also see video from other endpoints, in addition to the current talker, although much smaller in size. Endpoint A has 4 screens, so one of those screens shows up to 9 other Media Captures in a tiled fashion. When video from a 3 camera endpoint appears in the tiled area, video from all 3 cameras appears together across the screen with correct spatial relationship among those 3 images.



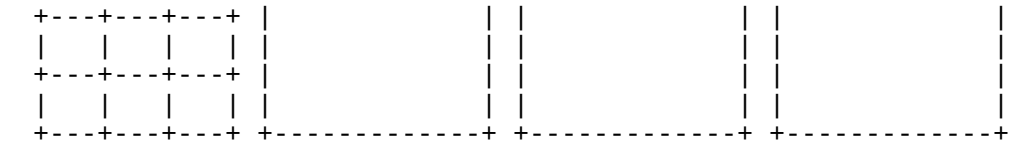


Figure 7: Endpoint A - 4 Screen Display

User B at Endpoint B sees a similar arrangement, except there are only 3 screens, so the 9 other Media Captures are spread out across the bottom of the 3 displays, in a picture-in-picture (PIP) format. When video from a 3 camera endpoint appears in the PIP area, video from all 3 cameras appears together across a single screen with correct spatial relationship.

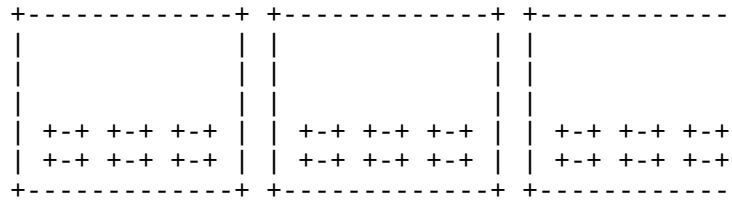


Figure 8: Endpoint B - 3 Screen Display with PiPs

When somebody at a different endpoint becomes the current talker, then User A and User B both see the video from the new talker appear on their large screen area, while the previous talker takes one of the smaller tiled or PIP areas. The person who is the current talker doesn't see themselves; they see the previous talker in their large screen area.

One of the points of this example is that endpoints A and B each want to receive 3 capture encodings for their large display areas, and 9 encodings for their smaller areas. A and B are be able to each send the same Configure message to the MCU, and each receive the same conceptual Media Captures from the MCU. The differences are in how they are rendered and are purely a local matter at A and B.

The Advertisements for such a scenario are described below.

Capture Scene #1	Description=Endpoint x
VC1	EncodingGroup=1
VC2	EncodingGroup=1

VC3	EncodingGroup=1
AC1	EncodingGroup=2
CSE1(VC1, VC2, VC3)	
CSE2(AC1)	

Table 19: Advertisement received at the MCU from Endpoints A to D

Capture Scene #1	Description=Endpoint y
VC1	EncodingGroup=1
AC1	EncodingGroup=2
CSE1(VC1)	
CSE2(AC1)	

Table 20: Advertisement received at the MCU from Endpoints E to F

Rather than considering what is displayed the CLUE concentrates more on what the MCU sends. The MCU doesn't know anything about the number of screens an endpoint has.

As Endpoints A to D each advertise that three Captures make up a Capture Scene, the MCU offers these in a "site" switching mode. That is that there are three Multiple Content Captures (and Capture Encodings) each switching between Endpoints. The MCU switches in the applicable media into the stream based on voice activity. Endpoint A will not see a capture from itself.

Using the MCC concept the MCU would send the following Advertisement to endpoint A:

Capture Scene #1	Description=Endpoint B
VC4	Left
VC5	Center
VC6	Right
AC1	
CSE(VC4,VC5,VC6)	
CSE(AC1)	
Capture Scene #2	Description=Endpoint C

VC7	Left
VC8	Center
VC9	Right
AC2	
CSE(VC7,VC8,VC9)	
CSE(AC2)	
Capture Scene #3	**Description=Endpoint D**
VC10	Left
VC11	Center
VC12	Right
AC3	
CSE(VC10,VC11,VC12)	
CSE(AC3)	
Capture Scene #4	**Description=Endpoint E**
VC13	
AC4	
CSE(VC13)	
CSE(AC4)	
Capture Scene #5	**Description=Endpoint F**
VC14	
AC5	
CSE(VC14)	
CSE(AC5)	
Capture Scene #6	**Description=Endpoint G**
VC15	
AC6	
CSE(VC15)	
CSE(AC6)	

Table 21: Advertisement sent to endpoint A - Source Part

The above part of the Advertisement presents information about the sources to the MCC. The information is effectively the same as the received Advertisements except that there are no Capture Encodings associated with them and the identities have been re-numbered.

In addition to the source Capture information the MCU advertises "site" switching of Endpoints B to G in three streams.

Capture Scene #7	Description=Output3streammix
MCC1(VC4,VC7,VC10, VC13)	CaptureArea=Left MaxCaptures=1 SynchronisationID=1 Policy=SoundLevel:0 EncodingGroup=1
MCC2(VC5,VC8,VC11, VC14)	CaptureArea=Center MaxCaptures=1 SynchronisationID=1 Policy=SoundLevel:0 EncodingGroup=1
MCC3(VC6,VC9,VC12, VC15)	CaptureArea=Right MaxCaptures=1 SynchronisationID=1 Policy=SoundLevel:0 EncodingGroup=1
MCC4() (for audio)	CaptureArea=whole scene MaxCaptures=1 Policy=SoundLevel:0 EncodingGroup=2
MCC5() (for audio)	CaptureArea=whole scene MaxCaptures=1 Policy=SoundLevel:1 EncodingGroup=2
MCC6() (for audio)	CaptureArea=whole scene MaxCaptures=1 Policy=SoundLevel:2 EncodingGroup=2
MCC7() (for audio)	CaptureArea=whole scene MaxCaptures=1 Policy=SoundLevel:3 EncodingGroup=2
CSE(MCC1,MCC2,MCC3)	

CSE(MCC4,MCC5,MCC6, MCC7)	

Table 22: Advertisement send to endpoint A - switching part

The above part describes the switched 3 main streams that relate to site switching. MaxCaptures=1 indicates that only one Capture from the MCC is sent at a particular time. SynchronisationID=1 indicates that the source sending is synchronised. The provider can choose to group together VC13, VC14, and VC15 for the purpose of switching according to the SynchronisationID. Therefore when the provider switches one of them into an MCC, it can also switch the others even though they are not part of the same Capture Scene.

All the audio for the conference is included in this Scene #7. There isn't necessarily a one to one relation between any audio capture and video capture in this scene. Typically a change in loudest talker will cause the MCU to switch the audio streams more quickly than switching video streams.

The MCU can also supply nine media streams showing the active and previous eight speakers. It includes the following in the Advertisement:

Capture Scene #8	Description=Output9stream
MCC4(VC4,VC5,VC6,VC7, VC8,VC9,VC10,VC11, VC12,VC13,VC14,VC15)	MaxCaptures=1 Policy=SoundLevel:0 EncodingGroup=1
MCC5(VC4,VC5,VC6,VC7, VC8,VC9,VC10,VC11, VC12,VC13,VC14,VC15)	MaxCaptures=1 Policy=SoundLevel:1 EncodingGroup=1
to	to
MCC12(VC4,VC5,VC6,VC7, VC8,VC9,VC10,VC11, VC12,VC13,VC14,VC15)	MaxCaptures=1 Policy=SoundLevel:8 EncodingGroup=1
CSE(MCC4,MCC5,MCC6, MCC7,MCC8,MCC9,	

MCC10,MCC11,MCC12)|	

Table 23: Advertisement sent to endpoint A - 9 switched part

The above part indicates that there are 9 capture encodings. Each of the Capture Encodings may contain any captures from any source site with a maximum of one Capture at a time. Which Capture is present is determined by the policy. The MCCs in this scene do not have any spatial attributes.

Note: The Provider alternatively could provide each of the MCCs above in its own Capture Scene.

If the MCU wanted to provide a composed Capture Encoding containing all of the 9 captures it could Advertise in addition:

Capture Scene #9	Description=NineTiles
MCC13(MCC4,MCC4,MCC6, MCC7,MCC8,MCC9, MCC10,MCC11,MCC12) CSE(MCC13)	MaxCaptures=9 EncodingGroup=1

Table 24: Advertisement sent to endpoint A - 9 composed part

As MaxCaptures is 9 it indicates that the capture encoding contains information from up to 9 sources at a time.

The Advertisement to Endpoint B is identical to the above other than the captures from Endpoint A would be added and the captures from Endpoint B would be removed. Whether the Captures are rendered on a four screen display or a three screen display is up to the Consumer to determine. The Consumer wants to place video captures from the same original source endpoint together, in the correct spatial order, but the MCCs do not have spatial attributes. So the Consumer needs to associate incoming media packets with the original individual captures in the advertisement (such as VC4, VC5, and VC6) in order to know the spatial information it needs for correct placement on the screens.

Editor's note: this is an open issue, about how to associate incoming packets with the original capture that is a constituent of an MCC. This document probably should mention it in an earlier section, after the solution is worked out in the other CLUE documents.

13. Acknowledgements

Allyn Romanow and Brian Baldino were authors of early versions. Mark Gorzyinski contributed much to the approach. We want to thank Stephen Botzko for helpful discussions on audio.

14. IANA Considerations

None.

15. Security Considerations

There are several potential attacks related to telepresence, and specifically the protocols used by CLUE, in the case of conferencing sessions, due to the natural involvement of multiple endpoints and the many, often user-invoked, capabilities provided by the systems.

A middle box involved in a CLUE session can experience many of the same attacks as that of a conferencing system such as that enabled by the XCON framework [RFC 6503]. Examples of attacks include the following: an endpoint attempting to listen to sessions in which it is not authorized to participate, an endpoint attempting to disconnect or mute other users, and theft of service by an endpoint in attempting to create telepresence sessions it is not allowed to create. Thus, it is RECOMMENDED that a middle box implementing the protocols necessary to support CLUE, follow the security recommendations specified in the conference control protocol documents. In the case of CLUE, SIP is the default conferencing protocol, thus the security considerations in RFC 4579 MUST be followed.

One primary security concern, surrounding the CLUE framework introduced in this document, involves securing the actual protocols and the associated authorization mechanisms. These concerns apply to endpoint to endpoint sessions, as well as sessions involving multiple endpoints and middle boxes. Figure 2 in section 5 provides a basic flow of information exchange for CLUE and the protocols involved.

As described in section 5, CLUE uses SIP/SDP to establish the session prior to exchanging any CLUE specific information. Thus the security mechanisms recommended for SIP [RFC 3261], including user authentication and authorization, SHOULD be followed. In addition, the media is based on RTP and thus existing RTP security mechanisms, such as DTLS/SRTP, MUST be supported.

A separate data channel is established to transport the CLUE protocol messages. The contents of the CLUE protocol messages are based on information introduced in this document, which is represented by an XML schema for this information defined in the CLUE data model [ref]. Some of the information which could possibly introduce privacy concerns is the xCard information as described in section x. In addition, the (text) description field in the Media Capture attribute (section 7.1.1.7) could possibly reveal sensitive information or specific identities. The same would be true for the descriptions in the Capture Scene (section 7.3.1) and Capture Scene Entry (7.3.2) attributes. One other important consideration for the information in the xCard as well as the description field in the Media Capture and Capture Scene Entry attributes is that while the endpoints involved in the session have been authenticated, there is no assurance that the information in the xCard or description fields is authentic. Thus, this information SHOULD not be used to make any authorization decisions and the participants in the sessions SHOULD be made aware of this.

While other information in the CLUE protocol messages does not reveal specific identities, it can reveal characteristics and capabilities of the endpoints. That information could possibly uniquely identify specific endpoints. It might also be possible for an attacker to manipulate the information and disrupt the CLUE sessions. It would also be possible to mount a DoS attack on the CLUE endpoints if a malicious agent has access to the data channel. Thus, It MUST be possible for the endpoints to establish a channel which is secure against both message recovery and message modification. Further details on this are provided in the CLUE data channel solution document.

There are also security issues associated with the authorization to perform actions at the CLUE endpoints to invoke specific capabilities (e.g., re-arranging screens, sharing content, etc.). However, the policies and security associated with these actions are outside the scope of this document and the overall CLUE solution.

16. Changes Since Last Version

NOTE TO THE RFC-Editor: Please remove this section prior to publication as an RFC.

Changes from 13 to 14:

1. Fill in section for Security Considerations.

2. Replace Role placeholder with Participant Information, Participant Type, and Scene Information attributes.

3. Spatial information implies nothing about how constituent media captures are combined into a composed MCC.

4. Clean up MCC example in Section 12.3.3. Clarify behavior of tiled and PIP display windows. Add audio. Add new open issue about associating incoming packets to original source capture.

5. Remove editor's note and associated statement about RTP multiplexing at end of section 5.

6. Remove editor's note and associated paragraph about overloading media channel with both CLUE and non-CLUE usage, in section 5.

7. In section 10, clarify intent of media encodings conforming to SDP, even with multiple CLUE message exchanges. Remove associated editor's note.

Changes from 12 to 13:

1. Added the MCC concept including updates to existing sections to incorporate the MCC concept. New MCC attributes: MaxCaptures, SynchronisationID and Policy.

2. Removed the "composed" and "switched" Capture attributes due to overlap with the MCC concept.

3. Removed the "Scene-switch-policy" CSE attribute, replaced by MCC and SynchronisationID.

4. Editorial enhancements including numbering of the Capture attribute sections, tables, figures etc.

Changes from 11 to 12:

1. Ticket #44. Remove note questioning about requiring a Consumer to send a Configure after receiving Advertisement.

2. Ticket #43. Remove ability for consumer to choose value of attribute for scene-switch-policy.

3. Ticket #36. Remove computational complexity parameter, MaxGroupPps, from Encoding Groups.

4. Reword the Abstract and parts of sections 1 and 4 (now 5) based on Mary's suggestions as discussed on the list. Move part of the Introduction into a new section Overview & Motivation.

5. Add diagram of an Advertisement, in the Overview of the Framework/Model section.

6. Change Intended Status to Standards Track.

7. Clean up RFC2119 keyword language.

Changes from 10 to 11:

1. Add description attribute to Media Capture and Capture Scene Entry.

2. Remove contradiction and change the note about open issue regarding always responding to Advertisement with a Configure message.

3. Update example section, to cleanup formatting and make the media capture attributes and encoding parameters consistent with the rest of the document.

Changes from 09 to 10:

1. Several minor clarifications such as about SDP usage, Media Captures, Configure message.

2. Simultaneous Set can be expressed in terms of Capture Scene and Capture Scene Entry.

3. Removed Area of Scene attribute.

4. Add attributes from draft-groves-clue-capture-attr-01.

5. Move some of the Media Capture attribute descriptions back into this document, but try to leave detailed syntax to the data model. Remove the OUTSOURCE sections, which are already incorporated into the data model document.

Changes from 08 to 09:

1. Use "document" instead of "memo".

2. Add basic call flow sequence diagram to introduction.

3. Add definitions for Advertisement and Configure messages.

4. Add definitions for Capture and Provider.

5. Update definition of Capture Scene.

6. Update definition of Individual Encoding.

7. Shorten definition of Media Capture and add key points in the Media Captures section.

8. Reword a bit about capture scenes in overview.

9. Reword about labeling Media Captures.

10. Remove the Consumer Capability message.

11. New example section heading for media provider behavior

12. Clarifications in the Capture Scene section.

13. Clarifications in the Simultaneous Transmission Set section.

14. Capitalize defined terms.

15. Move call flow example from introduction to overview section

16. General editorial cleanup

17. Add some editors’ notes requesting input on issues

18. Summarize some sections, and propose details be outsourced to other documents.

Changes from 06 to 07:

1. Ticket #9. Rename Axis of Capture Point attribute to Point on Line of Capture. Clarify the description of this attribute.

2. Ticket #17. Add "capture encoding" definition. Use this new term throughout document as appropriate, replacing some usage of the terms "stream" and "encoding".

3. Ticket #18. Add Max Capture Encodings media capture attribute.

4. Add clarification that different capture scene entries are not necessarily mutually exclusive.

Changes from 05 to 06:

1. Capture scene description attribute is a list of text strings, each in a different language, rather than just a single string.

2. Add new Axis of Capture Point attribute.

3. Remove appendices A.1 through A.6.

4. Clarify that the provider must use the same coordinate system with same scale and origin for all coordinates within the same capture scene.

Changes from 04 to 05:

1. Clarify limitations of "composed" attribute.

2. Add new section "capture scene entry attributes" and add the attribute "scene-switch-policy".

3. Add capture scene description attribute and description language attribute.

4. Editorial changes to examples section for consistency with the rest of the document.

Changes from 03 to 04:

1. Remove sentence from overview - "This constitutes a significant change ..."

2. Clarify a consumer can choose a subset of captures from a capture scene entry or a simultaneous set (in section "capture scene" and "consumer's choice...").

3. Reword first paragraph of Media Capture Attributes section.

4. Clarify a stereo audio capture is different from two mono audio captures (description of audio channel format attribute).

5. Clarify what it means when coordinate information is not specified for area of capture, point of capture, area of scene.

6. Change the term "producer" to "provider" to be consistent (it was just in two places).

7. Change name of "purpose" attribute to "content" and refer to RFC4796 for values.

8. Clarify simultaneous sets are part of a provider advertisement, and apply across all capture scenes in the advertisement.

9. Remove sentence about lip-sync between all media captures in a capture scene.

10. Combine the concepts of "capture scene" and "capture set" into a single concept, using the term "capture scene" to replace the previous term "capture set", and eliminating the original separate capture scene concept.

Informative References

Edt. Note: Decide which of these really are Normative References.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC3261] Rosenberg, J., Schulzrinne, H., Camarillo, G., Johnston, A., Peterson, J., Sparks, R., Handley, M., and E.

Schooler, "SIP: Session Initiation Protocol", RFC 3261, June 2002.

[RFC3264] Rosenberg, J., Schulzrinne, H., "An Offer/Answer Model with the Session Description Protocol (SDP)", RFC 3264, June 2002.

[RFC3550] Schulzrinne, H., Casner, S., Frederick, R., and V. Jacobson, "RTP: A Transport Protocol for Real-Time Applications", STD 64, RFC 3550, July 2003.

[RFC4353] Rosenberg, J., "A Framework for Conferencing with the Session Initiation Protocol (SIP)", RFC 4353, February 2006.

[RFC4579] Johnston, A., Levin, O., "SIP Call Control - Conferencing for User Agents", RFC 4579, August 2006

[RFC5117] Westerlund, M. and S. Wenger, "RTP Topologies", RFC 5117,
January 2008.

17. Authors' Addresses

Mark Duckworth (editor)
Polycom
Andover, MA 01810
USA

Email: mark.duckworth@polycom.com

Andrew Pepperell
Acano
Uxbridge, England
UK

Email: apeppere@gmail.com

Stephan Wenger
Vidyo, Inc.
433 Hackensack Ave.
Hackensack, N.J. 07601
USA

Email: stewe@stewe.org