CLUE WG
Internet Draft
Intended status: Informational
Expires: June, 2013

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

December 24, 2012

Framework for Telepresence Multi-Streams
draft-ietf-clue-framework-08.txt

Abstract

This memo offers a framework for a protocol that enables devices in a telepresence conference to interoperate by specifying the relationships between multiple media streams.

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 June 24, 2013.

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 draft provides a framework for a protocol to enable interoperability by handling multiple streams in a standardized way. It is intended to support the use cases described in draft-ietf-clue-telepresence-use-cases-02 and to meet the requirements in draft-ietf-clue-telepresence-requirements-01.

Conceptually distinguished are Media Providers and Media Consumers. 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 may include only Consumer or Provider functionality, such as recorder-type Consumers or webcam-type Providers.

One initial motivation for this memo and its companion documents has been that Endpoints according to this memo can, and usually do, have multiple Media Captures and Media Renderers. While previous system designs can deal with such a situation, what was missing was a mechanism that can associate the Media Captures with each other in space and time. Further, 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, a sensible system design is 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 may 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 memo 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. 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 memo and its companion documents.

A very brief outline of the call flow used by a simple system in compliance with this memo can be described as follows.

An initial offer/answer exchange establishes a CLUE channel between two Endpoints. With the establishment of that channel, the endpoints have consented to use the CLUE protocol mechanisms and have to adhere to them.

Over this CLUE channel, the Provider in each Endpoint conveys its characteristics and capabilities as specified herein (which will typically not be 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 can, 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. Second, it takes note of the spatial relationship associated with the Media that are described.

It is often sensible to take that spatial relationship into account when tailoring the SDP.

This CLUE exchange is 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 cases can (but not necessarily must) 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.

During the lifetime of a call, further exchanges can occur over the CLUE channel. In some cases, those further exchanges can be dealt with by Provider or Consumer without any other protocol activity. 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 may become 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, that could require a new O/A exchange.

Numerous optimizations may be possible, and are the implementer's choice. For example, it may be sensible to establish one or more initial media channels during the initial offer/answer exchange, which would allow, for example, for a fast startup of audio. Depending on the system design, it may be possible to re-use this established channel using only CLUE mechanisms, thereby avoiding further offer/answer exchanges.

One aspect of the protocol outlined herein and specified in normative 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 Media 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 has to follow the CLUE protocol and, therefore, has to "accept" 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.

Legacy devices are defined here in as those Endpoints and MCUs that do not support the setup and use of the CLUE channel. The notion of a device being a legacy device is established during the initial offer/answer exchange, in which the legacy device will not understand the offer for the CLUE channel and, therefore, reject it. This is the indication for the CLUE-implementing Endpoint or MCU that the other side of the communication is not compliant with CLUE, and to fall back to whatever mechanism was used before the introduction of 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. However, it should be noted that forms of RTP multiplexing of multiple RTP flows onto the same transport address are developed concurrently with the CLUE suite of specifications, and it is widely expected that most, if not all, Endpoints or MCUs supporting CLUE will also support those mechanisms. Some design choices made in this memo reflect this coincidence in spec development timing.

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 memo and companion documents and they are normative. In order to easily identify the use of a defined term, those terms are capitalized.

Audio Capture: Media Capture for audio. Denoted as ACn.

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

Cameras and microphones are examples for capture devices.

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 the scene that is captured by a collection of capture devices. 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 capture scene.

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

Individual Encoding: A variable with a set of attributes that describes the maximum values of a single audio or video capture encoding. The attributes include: maximum bandwidth- and for video maximum macroblocks (for H.264), maximum width, maximum height, maximum frame rate.

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). In contrast to an endpoint, an MCU may also send and receive media streams, but it is not the initiator nor the final terminator in the sense that Media is Captured or Rendered. 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] Mixer. [Edt. RFC4353 is tardy in requiring that media from the mixer be sent to EACH participant. I think we have practical use cases where this is not the case. But the bug (if it is one) is in 4353 and not herein.]

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. A Media Capture (MC) may be the source of one or more capture encodings. A Media Capture may also be constructed from other Media streams. A middle box can express Media Captures that it constructs from Media streams it receives.

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.

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

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.

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 ID of a capture or a spatial location.

Telepresence: an environment that gives non co-located users or user groups a feeling of (co-located) presence - the feeling that a Local user is in the same room with other Local users and the Remote parties. The inclusion of Remote parties is achieved through multimedia communication including at least audio and video signals of high fidelity.

Video Capture: Media Capture for video. Denoted as VCn.

Video composite: A single image that is formed from combining visual elements from separate sources.

4. Overview of the Framework/Model

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

The main goals include:

- Interoperability
- Extensibility
- Flexibility

Interoperability is achieved by the media provider describing the relationships between media streams in constructs that are understood by the consumer, who can then render the media. Extensibility is achieved through abstractions and the generality of the model, making it easy to add new parameters. Flexibility is achieved largely by having the consumer choose what content and

format it wants to receive from what the provider is capable of sending.

A transmitting endpoint or MCU describes specific aspects of the content of the media and the formatting of the media streams it can send (advertisement); and the receiving end responds to the provider by specifying which content and media streams it wants to receive (configuration). The provider then transmits the asked for content in the specified streams.

This advertisement and configuration occurs at call initiation but may also happen at any time throughout the conference, whenever there is a change in what the consumer wants or the provider can send.

An endpoint or MCU typically acts 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 that represent the same scene into capture scenes. A consumer chooses which media captures it wants to receive according to the capture scenes sent by the provider.

In addition, the provider sends the consumer a description of the individual encodings it can send in terms of the media attributes of the encodings, in particular, well-known audio and video parameters such as bandwidth, frame rate, macroblocks per second.

The provider also specifies constraints on its ability to provide media, and the consumer must take these into account in choosing the content and capture encodings it wants. 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 constraints, such as maximum bandwidth and maximum macroblocks/second.

The following sections discuss these constructs and processes in detail, followed by use cases showing how the framework specification can be used.

5. Spatial Relationships

In order for a consumer to perform a proper rendering, it is often necessary to provide 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 spatial 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 either be in real, physical units (millimeters), have an unknown scale or have no physical scale. Systems which know their physical dimensions should always provide those real-world measurements. Systems which don't know specific physical dimensions but still know relative distances should 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 spot of the provider's choosing. The provider must use the same coordinate system with 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

6. Media Captures and Capture Scenes

This section describes how media providers can describe the content of media to consumers.

6.1. 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 distinguish between multiple instances, video and audio captures are numbered such as: VC1, VC2 and AC1, AC2. VC1 and VC2 refer to two different video captures and AC1 and AC2 refer to two different audio captures.

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

6.1.1. Media Capture Attributes

Media Capture Attributes describe static information about the captures. A provider uses the media capture attributes to describe the media captures to the consumer. The consumer will select the captures it wants to receive. Attributes are defined by a variable and its value. The currently defined attributes and their values are:

Content: {slides, speaker, sl, main, alt}

A field with enumerated values which describes the role of the media capture and can be applied to any media type. The enumerated values are defined by [RFC4796]. The values for this attribute are the same as the mediacnt values for the content attribute in [RFC4796]. This attribute can have multiple values, for example content={main, speaker}.

Composed: {true, false}

A field with a Boolean value which indicates whether or not the Media Capture is a mix (audio) or composition (video) of streams.

This attribute is useful for a media consumer to avoid nesting a composed video capture into another composed capture or rendering. This attribute is not intended to describe the layout a media provider uses when composing video streams.

Audio Channel Format: {mono, stereo} 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.

A value of '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 RTP stream, while each mono audio capture maps to a separate RTP stream.

Switched: {true, false}

A field with a Boolean value which indicates whether or not the Media Capture represents the (dynamic) most appropriate subset of a 'whole'. What is 'most appropriate' is up to the provider and could be the active speaker, a lecturer or a VIP.

Point of Capture: {(X, Y, Z)}

A field with a single Cartesian (X, Y, Z) point value which describes the spatial location, virtual or physical, of the capturing device (such as camera).

When the Point of Capture attribute is specified, it must include X, Y and Z coordinates. If the point of capture is not specified, it means the consumer should not assume anything about the spatial location of the capturing device. Even if the provider specifies an area of capture attribute, it does not need to specify the point of capture.

Point on Line of Capture: {(X,Y,Z)}

A field with a single Cartesian (X, Y, Z) point value (virtual or physical) 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). This point MUST lie between the Point of Capture and the Area of Capture.

The Point on Line of Capture MUST be ignored if the Point of Capture is not present for this capture device. When the Point on Line of Capture attribute is specified, it must include X, Y and Z coordinates. These coordinates MUST NOT be identical to the Point of Capture coordinates. If the Point on Line of Capture is not specified, no assumptions are made about the axis of the capturing device.

Area of Capture:

{bottom left(X1, Y1, Z1), bottom right(X2, Y2, Z2), top left(X3, Y3, Z3), top right(X4, Y4, Z4)}

A field with a set of four (X, Y, Z) points as a value which describe 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 should be co-planar. The four points form a quadrilateral, not necessarily a rectangle.

The quadrilateral described by the four (X, Y, Z) points defines the plane of interest for the particular media capture.

If the area of capture attribute is specified, it must include X, Y and Z coordinates for all four points. If the area of capture is not specified, it means the media capture is not spatially related to any other media capture (but this can change in a subsequent provider advertisement).

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

EncodingGroup: {<encodeGroupID value>}

A field with a value equal to the encodeGroupID of the encoding group associated with the media capture.

Max Capture Encodings: {unsigned integer}

An optional attribute indicating the maximum number of capture encodings that can be simultaneously active for the media capture. If absent, this parameter defaults to 1. The minimum value for this attribute is 1. The number of simultaneous capture encodings is also limited by the restrictions of the encoding group for the media capture.

6.2. Capture Scene

In order for a provider's individual media captures to be used effectively by a consumer, the provider organizes the media captures into capture scenes, with the structure and contents of these capture scenes being sent from the provider to the consumer.

A capture scene is a structure representing the scene that is captured by a collection of capture devices. 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 may also express capture scenes that it constructs from media streams it receives.

A provider may advertise multiple capture scenes or just a single capture scene. A media provider might typically use one capture scene for main participant media and another capture scene for a computer generated presentation. A capture scene may 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 provider can express spatial relationships between media captures that are included in the same capture scene. But there

is no spatial relationship between media captures that are in different capture scenes.

A media provider arranges media captures in a capture scene to help the media consumer choose which captures it wants. The capture scene entries in a capture scene are different alternatives the provider is suggesting for representing the capture scene. 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.

Media 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 media captures in a single entry simultaneously. A consumer may decide to receive all the media 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 media captures from different capture scene entries.

When a provider advertises a capture scene with multiple entries, it is essentially signaling that there are multiple representations of the same 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 3 video captures versus an entry consisting of just a single video capture). In this latter example, the provider would in the simple case end up providing to the consumer the entry containing the number of video captures that most closely matched the media consumer's number of display devices.

The following is an example of 4 potential capture scene entries for an endpoint-style media 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 with 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 additional alternatives of how to capture the same room in different ways. The inclusion of the audio capture in the same capture scene indicates that AC0 is associated with those video captures, meaning it comes from the same scene. The audio should be rendered in conjunction with any rendered video captures from the same capture scene.

6.2.1. Capture scene attributes

Attributes can be applied to capture scenes as well as to individual media captures. Attributes specified at this level apply to all constituent media captures.

Description attribute - list of {<description text>, <language tag>}

The optional description attribute is a list of human readable text strings which describe the capture scene. If there is more than one string in the list, then each string in the list should contain the same description, but in a different language. A provider that advertises multiple capture scenes can provide descriptions for each of them. This attribute can contain text in any number of languages.

The language tag identifies the language of the corresponding description text. The possible values for a language tag are the values of the 'Subtag' column for the "Type: language" entries in the "Language Subtag Registry" at [IANA-Lan] originally defined in [RFC5646]. A particular language tag value MUST NOT be used more than once in the description attribute list.

Area of Scene attribute

The area of scene attribute for a capture scene has the same format as the area of capture attribute for a media capture. The area of scene is for the entire scene, which is captured by the one or more media captures in the capture scene entries. If the provider does not specify the area of scene, but does specify

areas of capture, then the consumer may assume the area of scene is greater than or equal to the outer extents of the individual areas of capture.

Scale attribute

An optional attribute indicating if the numbers used for area of scene, area of capture and point of capture are in terms of millimeters, unknown scale factor, or not any scale, as described in Section 5. If any media captures have an area of capture attribute or point of capture attribute, then this scale attribute must also be defined. The possible values for this attribute are:

"millimeters"

"unknown"

"no scale"

6.2.2. Capture scene entry attributes

Attributes can be applied to capture scene entries. Attributes specified at this level apply to the capture scene entry as a whole.

Scene-switch-policy: {site-switch, segment-switch}

A media provider uses this scene-switch-policy attribute to indicate its support for different switching policies. In the provider's advertisement, this attribute can have multiple values, which means the provider supports each of the indicated policies. The consumer, when it requests media captures from this capture scene entry, should also include this attribute but with only the single value (from among the values indicated by the provider) indicating the consumer's choice for which policy it wants the provider to use. If the provider does not support any of these policies, it should omit this attribute.

The "site-switch" policy means all captures are switched at the same time to keep captures from the same endpoint site together. Let's say the speaker is at site A and everyone else is at a "remote" site.

When the room at site A shown, all the camera images from site A are forwarded to the remote sites. Therefore at each receiving

remote site, all the screens display camera images from site A. This can be used to preserve full size image display, and also provide full visual context of the displayed far end, site A. In site switching, there is a fixed relation between the cameras in each room and the displays in remote rooms. The room or participants being shown is switched from time to time based on who is speaking or by manual control.

The "segment-switch" policy means different captures can switch at different times, and can be coming from different endpoints. Still using site A as where the speaker is, and "remote" to refer to all the other sites, in segment switching, rather than sending all the images from site A, only the image containing the speaker at site A is shown. The camera images of the current speaker and previous speakers (if any) are forwarded to the other sites in the conference.

Therefore the screens in each site are usually displaying images from different remote sites - the current speaker at site A and the previous ones. This strategy can be used to preserve full size image display, and also capture the non-verbal communication between the speakers. In segment switching, the display depends on the activity in the remote rooms - generally, but not necessarily based on audio / speech detection.

6.3. Simultaneous Transmission Set Constraints

The provider may have constraints or limitations on its ability to send media captures. One type 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 - bandwidth and macroblocks/second. This type of constraint is captured by encoding groups, discussed below.

An endpoint or MCU 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 will 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 media captures that can be sent at the same time. This is easier to show in an example.

Consider the example of a room system where there are 3 cameras each of which can send a separate capture covering 2 persons each- VC0, VC1, VC2. The middle camera can also zoom out and show all 6 persons, VC3. But the middle camera cannot be used in both modes at the same time - it has to either show the space where 2 participants sit or the whole 6 seats, but not both at the same time.

Simultaneous transmission sets are expressed as sets of the MCs that could physically be transmitted at the same time, (though it may not make sense to do so). In this example the two simultaneous sets are shown in Table 1. The consumer must make sure that it chooses one and not more of the mutually exclusive sets. A consumer may choose any subset of the media captures in a simultaneous set, it does not have to choose all the captures in a simultaneous set if it does not want to receive all of them.

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

Table 1: Two Simultaneous Transmission Sets

A media provider includes the simultaneous sets in its provider advertisement. These simultaneous set constraints apply across all the captures scenes in the advertisement. The simultaneous transmission sets MUST allow all the media captures in a particular capture scene entry to be used simultaneously.

7. Encodings

We have considered how providers can describe the content of media to consumers. We will now consider how the providers communicate information about their abilities to send streams. We introduce two constructs - individual encodings and encoding groups. Consumers will then map the media captures they want onto the encodings with encoding parameters they want. This process is then described.

7.1. 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 media provider to the media 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 only one capture encoding at a time.

The parameters of an individual encoding represent the maximum values for certain aspects of the encoding. A particular instantiation into a capture encoding might use lower values than these maximums.

The following tables show the variables for audio and video encoding.

Name	Description
encodeID	A unique identifier for the individual encoding
maxBandwidth	Maximum number of bits per second
maxH264Mbps	Maximum number of macroblocks per second: ((width + 15) / 16) * ((height + 15) / 16) * framesPerSecond
maxWidth	Video resolution's maximum supported width, expressed in pixels
maxHeight	Video resolution's maximum supported height, expressed in pixels
maxFrameRate	Maximum supported frame rate

```
   |
   +--------------+-------------------------------------------------
   --+
```

Table 2: Individual Video Encoding Parameters

Name	Description
maxBandwidth	Maximum number of bits per second

Table 3: Individual Audio Encoding Parameters

7.2. Encoding Group

An encoding group includes a set of one or more individual encodings, plus some parameters that apply to the group as a whole. By grouping multiple individual encodings together, an encoding group describes additional constraints on bandwidth and other parameters for the group. Table 4 shows the parameters and individual encoding sets that are part of an encoding group.

Name	Description
encodeGroupID	A unique identifier for the encoding group
maxGroupBandwidth	Maximum number of bits per second relating to all encodings combined
maxGroupH264Mbps	Maximum number of macroblocks per second relating to all video encodings combined
videoEncodings[]	Set of potential encodings (list of encodeIDs)
audioEncodings[]	Set of potential encodings (list of encodeIDs)

Table 4: Encoding Group

When the individual encodings in a group are instantiated into capture encodings, each capture encoding has a bandwidth that must be less than or equal to the maxBandwidth for the particular individual encoding. The maxGroupBandwidth parameter gives the additional restriction that the sum of all the individual capture encoding bandwidths must be less than or equal to the maxGroupBandwidth value.

Likewise, the sum of the macroblocks per second of each instantiated encoding in the group must not exceed the maxGroupH264Mbps value.

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

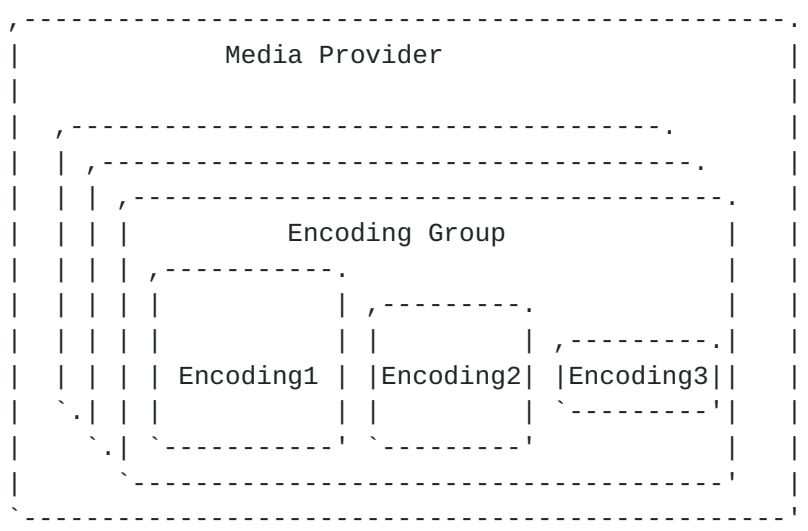


Figure 1: Encoding Group Structure

A media 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.

While a typical 3 codec/display system might have one encoding group per "codec box", 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 once.

8. Associating Media Captures with Encoding Groups

Every media capture is associated with an encoding group, which is used to instantiate that media capture into one or more capture encodings. Each media capture has an encoding group attribute. The value of this attribute is the encodeGroupID for the encoding group with which it is associated. More than one media 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 media 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 media consumer can configure the media provider 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.

The Encoding Groups MUST allow all the media captures in a particular capture scene entry to be used simultaneously.

9. Consumer's Choice of Streams to Receive from the Provider

After receiving the provider's advertised media captures and associated constraints, the consumer must choose which media captures it wishes to receive, and which individual encodings from the provider it wants to use to encode the captures. Each media capture has an encoding group ID attribute which specifies which individual encodings are available to be used for that media capture.

For each media 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 the resolution, frame rate, bandwidth, etc. when asking for capture encodings for its chosen captures. Upon receipt of this configuration command from the consumer, the provider generates a stream for each such configured capture encoding and sends those streams to the consumer.

The consumer must have received at least one capture advertisement from the provider to be able to configure the provider's generation of media streams.

The consumer is able to change its configuration of the provider's encodings any number of times during the call, either in response to a new capture advertisement from the provider or autonomously. The consumer need not send a new configure message to the provider when it receives a new capture advertisement from the provider unless the contents of the new capture advertisement cause the consumer's current configure message to become invalid.

When choosing which streams to receive from the provider, and the encoding characteristics of those streams, the consumer needs to take several things into account: its local preference, simultaneity restrictions, and encoding limits.

9.1. Local preference

A variety of local factors will influence the consumer's choice of 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 media captures, or different encoding characteristics, being required by the consumer

9.2. Physical simultaneity restrictions

There may be physical simultaneity constraints imposed by 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 camera 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 may not (if the overall room view were produced by changing the zoom level on the center camera, for instance).

9.3. Encoding and encoding group limits

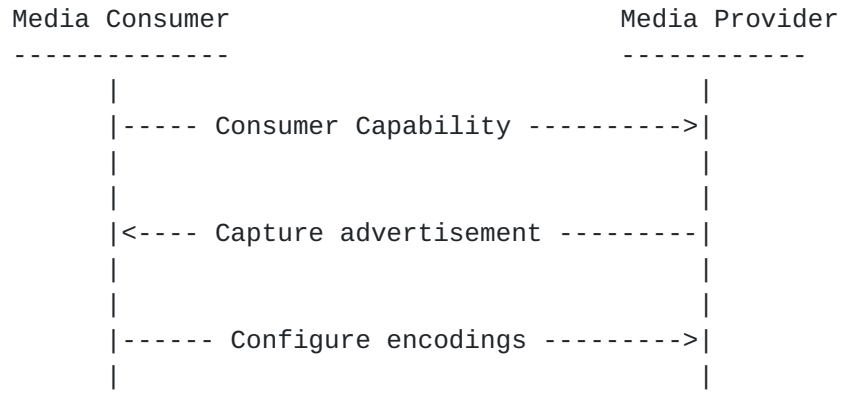
Each of the provider's encoding groups has limits on bandwidth and macroblocks per second, and the constituent potential encodings have limits on the bandwidth, macroblocks per second, video frame rate, and resolution that can be provided. When choosing the media captures to be received from a provider, a consumer device must ensure that the encoding characteristics requested for each individual media capture fits within the capability of the

encoding it is being configured to use, as well as ensuring that the combined encoding characteristics for media 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 favour of different capture encodings - for instance, if a set of 3 media captures could only be provided at a low resolution then a 3 screen device could switch to favoring a single, higher quality, capture encoding.

9.4. Message Flow

The following diagram shows the basic flow of messages between a media provider and a media consumer. The usage of the "capture advertisement" and "configure encodings" message is described above. The consumer also sends its own capability message to the provider which may contain information about its own capabilities or restrictions.

Diagram for Message Flow



In order for a maximally-capable provider to be able to advertise a manageable number of video captures to a consumer, there is a potential use for the consumer, at the start of CLUE, to be able to inform the provider of its capabilities. One example here would be the video capture attribute set - a consumer could tell the provider the complete set of video capture attributes it is able to understand and so the provider would be able to reduce the capture scene it advertises to be tailored to the consumer.

TBD - the content of the consumer capability message needs to be better defined. The authors believe there is a need for this message, but have not worked out the details yet.

10. Extensibility

One of the most 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, will 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.

Assuming the implementation is in something like XML, adding data elements and attributes makes extensibility easy.

11. Examples - Using the Framework

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

11.1. Three screen endpoint media provider

Consider an endpoint with the following description:

3 cameras, 3 displays, a 6 person table

- Each video device can provide one capture for each 1/3 section of the table

- A single capture representing the active speaker can be provided

- A single capture representing the active speaker with the other 2 captures shown picture in picture within the stream can be provided

- 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, content=main, switched=false

- VC1- (the center camera stream), encoding group=EG1, content=main, switched=false

- VC2- (the camera-right camera stream), encoding group=EG2, content=main, switched=false

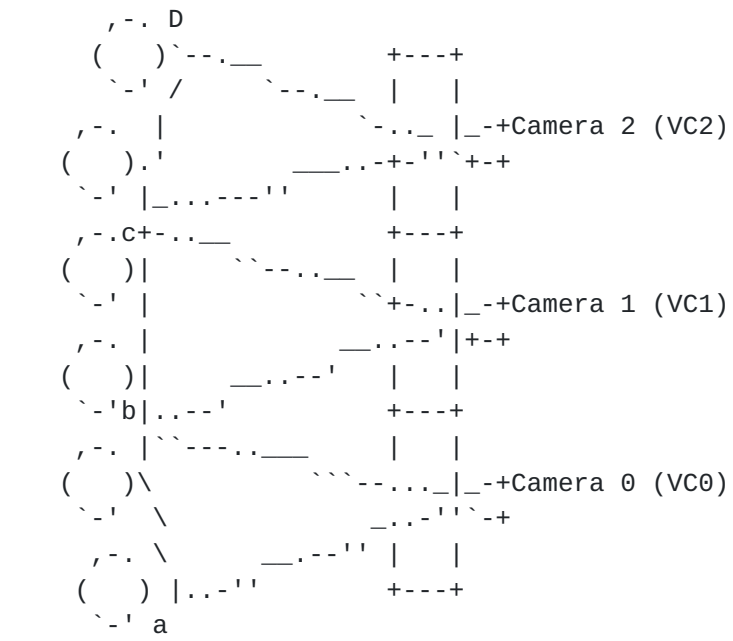
- VC3- (the loudest panel stream), encoding group=EG1, content=main, switched=true

- VC4- (the loudest panel stream with PiPs), encoding group=EG1, content=main, composed=true, switched=true

- VC5- (the zoomed out view of all people in the room), encoding group=EG1, content=main, composed=false, switched=false

- VC6- (presentation stream), encoding group=EG1, content=slides, 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.



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 can not 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 maxMbps value compatible with that) as long as it is the only active encoding in the group(as maxMbps for the entire encoding group is also 489600). 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 1080p30 and one at 720p30.

```
encodeGroupID=EG0, maxGroupH264Mbps=489600,
maxGroupBandwidth=6000000
    encodeID=ENC0, maxWidth=1920, maxHeight=1088, maxFrameRate=60,
                   maxH264Mbps=489600, maxBandwidth=4000000
    encodeID=ENC1, maxWidth=1280, maxHeight=720, maxFrameRate=30,
                   maxH264Mbps=108000, maxBandwidth=4000000
    encodeID=ENC2, maxWidth=960, maxHeight=544, maxFrameRate=30,
                   maxH264Mbps=61200, maxBandwidth=4000000
encodeGroupID=EG1 maxGroupH264Mbps=489600
maxGroupBandwidth=6000000
    encodeID=ENC3, maxWidth=1920, maxHeight=1088, maxFrameRate=60,
                   maxH264Mbps=489600, maxBandwidth=4000000
    encodeID=ENC4, maxWidth=1280, maxHeight=720, maxFrameRate=30,
                   maxH264Mbps=108000, maxBandwidth=4000000
    encodeID=ENC5, maxWidth=960, maxHeight=544, maxFrameRate=30,
                   maxH264Mbps=61200, maxBandwidth=4000000
encodeGroupID=EG2 maxGroupH264Mbps=489600
maxGroupBandwidth=6000000
    encodeID=ENC6, maxWidth=1920, maxHeight=1088, maxFrameRate=60,
                   maxH264Mbps=489600, maxBandwidth=4000000
    encodeID=ENC7, maxWidth=1280, maxHeight=720, maxFrameRate=30,
                   maxH264Mbps=108000, maxBandwidth=4000000
    encodeID=ENC8, maxWidth=960, maxHeight=544, maxFrameRate=30,
                   maxH264Mbps=61200, maxBandwidth=4000000
```

Figure 2: 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, maxGroupH264Mbps=0, maxGroupBandwidth=320000
    encodeID=ENC9, maxBandwidth=64000
    encodeID=ENC10, maxBandwidth=64000
    encodeID=ENC11, maxBandwidth=64000
    encodeID=ENC12, maxBandwidth=64000
    encodeID=ENC13, maxBandwidth=64000
```

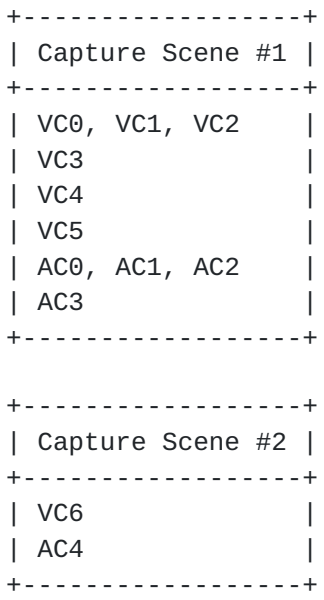
Figure 3: Example Encoding Group for Audio

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 scene. Capture Scene #1

is for the main people captures, and Capture Scene #2 is for presentation.

Each row in the table is a separate entry in the capture scene



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 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).

11.2. Encoding Group Example

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

```
encodeGroupID=EG0, maxGroupH264Mbps=489600,
maxGroupBandwidth=6000000
    encodeID=VIDENC0, maxWidth=1920, maxHeight=1088,
maxFrameRate=60,
                      maxH264Mbps=244800, maxBandwidth=4000000
    encodeID=VIDENC1, maxWidth=1920, maxHeight=1088,
maxFrameRate=60,
                      maxH264Mbps=244800, maxBandwidth=4000000
    encodeID=AUDENC0, maxBandwidth=96000
    encodeID=AUDENC1, maxBandwidth=96000
    encodeID=AUDENC2, maxBandwidth=96000
```

Here, the encoding group is EG0. It can transmit up to two 1080p30 capture encodings (Mbps for 1080p = 244800), but it is capable of transmitting a maxFrameRate of 60 frames per second (fps). To achieve the maximum resolution (1920 x 1088) the frame rate is limited to 30 fps. However 60 fps can be achieved at a lower resolution if required by the consumer. 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, maxGroupH264Mbps=489600,
maxGroupBandwidth=6000000
    encodeID=VIDENC0, maxWidth=1920, maxHeight=1088,
maxFrameRate=60,
                      maxH264Mbps=244800, maxBandwidth=4000000
    encodeID=VIDENC1, maxWidth=1920, maxHeight=1088,
maxFrameRate=60,
                      maxH264Mbps=244800, maxBandwidth=4000000
encodeGroupID=EG1, maxGroupH264Mbps=0, maxGroupBandwidth=500000
    encodeID=AUDENC0, maxBandwidth=96000
    encodeID=AUDENC1, maxBandwidth=96000
    encodeID=AUDENC2, maxBandwidth=96000
```

11.3. 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	note
VC0	video capture for single screen consumer
VC1, VC2	video capture for 2 screen consumer
VC3, VC4, VC5	video capture for 3 screen consumer
VC6, VC7, VC8, VC9	video capture for 4 screen consumer
AC0	audio capture representing all participants

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

11.4. 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 th provider.

A sane, basic, algorithm might be for the consumer to go through eac 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.

11.4.1. One screen 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.

11.4.2. Two screen 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 centre 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 centre streams (one per screen) and the centre 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.

11.4.3. Three screen consumer configuring the example

This is the most straightforward case - the 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. Acknowledgements

Mark Gorzyinski contributed much to the approach. We want to thank Stephen Botzko for helpful discussions on audio.

13. IANA Considerations

TBD

14. Security Considerations

TBD

15. Changes Since Last Version

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

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

[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.

[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.

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

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

[RFC5646] Phillips, A. and M. Davis, "Tags for Identifying Languages", BCP 47, RFC 5646, September 2009.

[IANA-Lan] IANA, "Language Subtag Registry", <http://www.iana.org/assignments/language-subtag-registry>.

16. Authors' Addresses

Mark Duckworth (editor)
Polycom
Andover, MA 01810
USA

Email: mark.duckworth@polycom.com

Andrew Pepperell
Silverflare
Uxbridge, England
UK

Email: apeppere@gmail.com

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

Email: stewe@stewe.org