CLUE WG
Internet-Draft
Intended status: Informational
Expires: April 5, 2012

A. Romanow
Cisco Systems
M. Duckworth
Polycom
A. Pepperell
B. Baldino
Cisco Systems

October 3, 2011

Framework for Telepresence Multi-Streams
draft-romanow-clue-framework-01.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 RTP 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 April 5, 2012.

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-00 and to meet the requirements in draft-romanow-clue-requirements-xx.

The solution described here is strongly focused on what is being done today, rather than on a vision of future conferencing. At the same time, the highest priority has been given to creating an extensible framework to make it easy to accommodate future conferencing functionality as it evolves.

The purpose of this effort is to make it possible to handle multiple streams of media in such a way that a satisfactory user experience is possible even when participants are on different vendor equipment and when they are using devices with different types of communication capabilities. Information about the relationship of media streams must be communicated so that audio/video rendering can be done in the best possible manner. In addition, it is necessary to choose which media streams are sent.

There is no attempt here to dictate to the renderer what it should do. What the renderer does is up to the renderer.

After the following Definitions, two short sections introduce key concepts. The body of the text comprises three sections that deal with in turn stream content, choosing streams and an implementation example. The media provider and media consumer behavior are described in separate sections as well. Several appendices describe further details for using the framework.

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 definitions marked with an "*" are new; all the others are from draft-wenger-clue-definitions-00-01.txt.

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

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 Scene: the scene that is captured by a collection of Capture Devices. A Capture Scene may be represented by more than one type of Media. A Capture Scene may include more than one Media Capture of the same type. An example of a Capture Scene is 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. A middle box may also express Capture Scenes that it constructs from Media streams it receives.

A Capture Set includes Media Captures that all represent some aspect of the same Capture Scene. The items (rows) in a Capture Set represent different alternatives for representing the same Capture Scene.

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

Individual Encode: 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, maximum width, maximum height, maximum frame rate. [Edt. These are based on H.264.]

*Encoding Group: Encoding group: A set of encoding parameters representing a device's complete encoding capabilities or a subdivision of them. Media stream providers formed of multiple physical units, in each of which resides some encoding capability, would typically advertise themselves to the remote media stream consumer as being formed multiple encoding groups. Within each encoding group, multiple potential actual encodings are possible, with the sum of those encodings' characteristics constrained to being less than or equal to the group-wide constraints.

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.

Endpoint Characteristics: include placement of Capture and Rendering Devices, capture/render angle, resolution of cameras and screens, spatial location and mixing parameters of microphones. Endpoint characteristics are not specific to individual media streams sent by the endpoint.

Left: For media captures, left and right is from the point of view of a person observing the rendered media.

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 may be the source of one or more Media streams. 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.

Right: For media captures, left and right is from the point of view of a person observing the rendered media.

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 Left and Right.

*Stream: RTP stream as in [RFC3550].

Stream Characteristics: include 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 (which could include for example and depending on the solution found: the I-D or spatial location of a capture device a stream originates from).

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. Framework Features

Two key functions must be accomplished so that multiple media streams can be handled in a telepresence conference. These are:

- How to choose which streams the provider should send to the consumer
- What information needs to be added to the streams to allow a rendering of the capture scene

The framework/model we present here can be understood as specifying these two functions.

Media stream providers and consumers are central to the framework. The provider's job is to advertise its capabilities (as described here) to the consumer, whose job it is to configure the provider's encoding capabilities as described below. Both providers and consumers can each send and receive information, that is, we do not have one party as the provider and one as the consumer exclusively,

but all parties have both sending and receiving parts to them. Most devices function as both a media provider and as a media consumer.

For two devices to communicate bidirectionally, with media flowing in both directions, both devices act as both a media provider and a media consumer. The protocol exchange shown later in the "Choosing Streams" section happens twice independently between the 2 bidirectional devices.

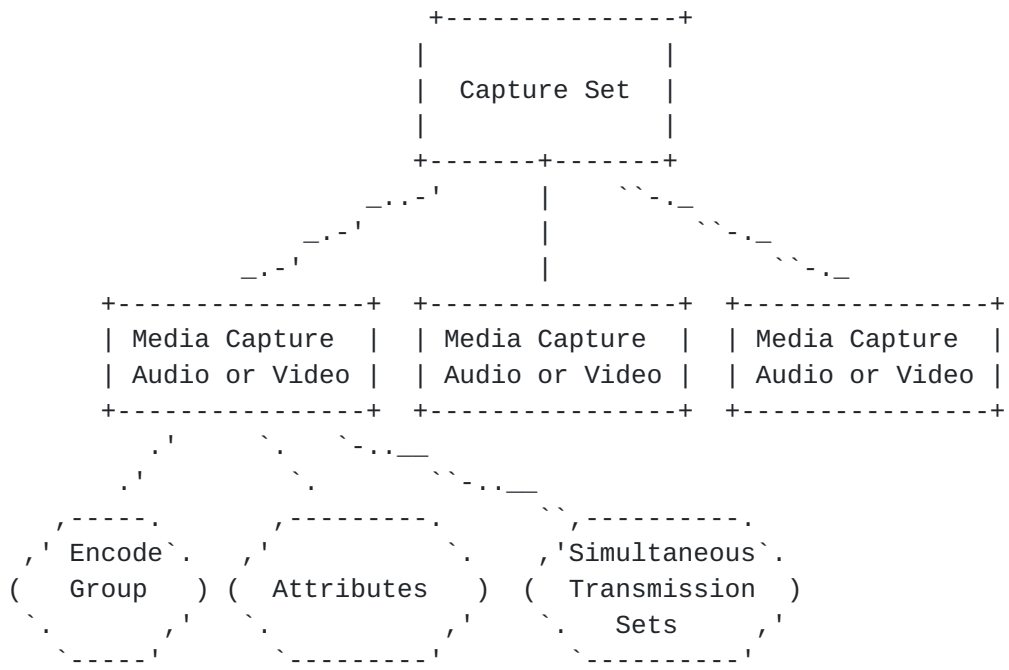
Both endpoints and MCUs, or more generally "middleboxes", can be media providers and consumers.

Generally, the provider is capable of sending alternate captures of a capture scene. These are described by the provider as capabilities and chosen by the consumer.

5. Stream Information

This section describes the structure for communicating information between providers and consumers. Figure illustrates how information to be communicated is organized. Each construct illustrated in the diagram is discussed in the sections below.

Diagram for Stream Content



5.1. Media capture -- Audio and Video

A media capture, as defined in definitions, is a fundamental concept of the model. Media can be captured in different ways, for example by various arrangements of cameras and microphones. The model uses the terms "video capture" (VC) and "audio capture" (AC) to refer to sources of media streams. To distinguish between multiple instances, they are numbered for example VC1, VC2, and VC3 could refer to three different video captures which can be used simultaneously.

Media captures are dynamic. They can come and go in a conference - and their parameters can change. A provider can advertise a new list of captures at any time. Both the media provider and media consumer can send "their messages" (i.e., capture set advertisements, stream configurations) any number of times during a call, and the other end is always required to act on any new information received (e.g., stopping streams it had previously configured that are no longer valid).

A media capture can be a media source such as video from a specific camera, or it can be more conceptual such as a composite image from several cameras, or an automatic dynamically switched capture choosing from several cameras depending on who is talking or other factors.

A media capture is described by Attributes and associated with an Encode Group, and Physical Simultaneity Set.

Audio and video captures are aggregated into Capture Sets as described below.

5.2. Attributes

Audio and video capture attributes describe information about streams and their relationships. [Edt: We do not mean to duplicate SDP, if an SDP description can be used, great.] The attributes of media captures refer to static aspects of those captures that can be used by the consumer for selecting the captures offered by the provider.

The mechanism of Attributes make the framework extensible. Although we are defining some attributes now based on the most common use cases, new attributes can be added for new use cases as they arise. In general, the way to extend the solution to handle new features is by adding attributes and/or values.

We describe attributes by variables and their values. The current attributes are listed below and then described. The variable is shown in parentheses, and the values follow after the colon:

- (Purpose): main, presentation
- (Audio mixed): true, false
- (Audio Channel Format): mono, stereo, tbd
- (Area of Capture): A set of 'Ranges' describing the relevant area being capture by a capture device
- (Point of Capture): A 'Point' describing the location of the capture device or pseudo-device
- (Area scale): true, false indicating if area numbers are in millimeters
- (Video composed): true, false
- (Auto-switched): true, false

5.2.1. Purpose

A variable with enumerated values describing the purpose or role of the Media Capture. It could be applied to any media type. Possible values: main, presentation, others TBD.

Main:

The audio or video capture is of one or more people participating in a conference (or where they would be if they were there). It is of part or all of the Capture Scene.

Presentation:

The stream provides a presentation, e. g., from a connected laptop or other input device.

5.2.2. Audio mixed

A Boolean variable to indicate whether the AC is a mix of other ACs or Streams.

5.2.3. Audio Channel Format

The "channel format" attribute of an Audio Capture indicates how the meaning of the channels is determined. It is an enumerated variable describing the type of audio channel or channels in the Audio Capture. The possible values of the "channel format" attribute are:

- mono
- stereo
- TBD - other possible future values (to potentially include other things like 3.0, 3.1, 5.1 surround sound and binaural)

All ACs in the same row of a Capture Set MUST have the same value of the "channel format" attribute.

There can be multiple ACs of a particular type, or even different types. These multiple ACs could each have an area of capture attribute to indicate they represent different areas of the capture scene.

If there are multiple audio streams, they might be correlated (that is, someone talking might be heard in multiple captures from the same room). Echo cancellation and stream synchronization in consumers should take this into account.

Mono:

An AC with channel format="mono" has one audio channel.

Stereo:

An AC with channel format = "stereo" has exactly two audio channels, left and right, as part of the same AC. [Edt: should we mention RFC 3551 here? The channel format may be related to how Audio Captures are mapped to RTP streams. This stereo is not the same as the effect produced from two mono ACs one from the left and one from the right.]

5.2.4. Area of capture

The area_of_capture attribute is used to describe the relevant area of which a media capture is "capturing". By comparing the area of capture for different media captures, a consumer can determine the spatial relationships of the captures on the provider so that they can be rendered correctly. The attribute consists of a set of 'Ranges', one range for each spatial dimension, where each range has a Begin and End coordinate. It is not necessary to fill out all of the dimensions if they are not relevant (i.e. if an endpoint's captures only span a single dimension, only the 'x' coordinate can be used). There is no need to pre-define a possible range for this coordinate system; a device may choose what is most appropriate for describing its captures. However, it is specified that as numbers move from lower to higher, the location is going from: left to right

(in the case of the 'x' dimension), front to back (in the case of the 'y' dimension or low to high (in the case of the 'z' dimension).

5.2.5. Point of capture

The point_of_capture attribute can be used to describe the location of a capture device or pseudo-device. If there are multiple captures which share the same 'area_of_capture' value, then it is useful to know the location from which they are capturing that area (e.g. a device which has multiview). Point of capture is expressed as a single {x, y, z} coordinate where, as with area_of_capture, only the necessary dimensions need be expressed.

5.2.6. Area Scale Millimeters

An optional Boolean variable indicating if the numbers used for area of capture and point of capture are in terms of millimeters. If this attribute is true, then the x,y,z numbers represent millimeters. If this attribute is false, then there is no physical scale. The default value is false.

5.2.7. Video composed

An optional Boolean variable indicating if the VC is constructed by composing multiple other video captures together. (This could indicate for example a continuous presence view of multiple images in a grid, or a large image with smaller picture-in-picture images in it.)

Note: this attribute is not intended to differentiate between different ways of composing images. For possible extension of the framework, additional attributes could be defined to distinguish between different ways of composing images, with different video layout arrangements of composing multiple images into one.

5.2.8. Auto-switched

A Boolean variable that may be used or audio and/or video streams. In this case the offered AC or VC varies depending on some rule; it is auto-switched between possible VCs, or between possible ACs. The most common example of this is sending the video capture associated with the "loudest" speaker according to an audio detection algorithm.

5.3. Capture Set

A capture set describes the alternative media streams that the provider offers to send to the consumer. As shown in the content diagram above, the capture set is an aggregation of all audio and

video captures for a particular scene that a provider is willing to send.

A provider describes its ability to send alternative media streams in the capture set, which lists the media captures in rows, as shown below. Each row of the capture set consists of either a single capture or a group of captures. A group means the individual captures in the group are spatially related with the specific ordering of the captures described through the use of attributes.

Here is an example of a simple capture set with three video captures and three audio channels:

(VC0, VC1, VC2)

(AC0, AC1, AC2)

The three VCs together in a row indicate those captures are spatially related to each other. Similarly for the 3 ACs in the second row. The ACs and VCs in the same capture set are spatially related to each other.

Multiple Media Captures of the same media type are often spatially related to each other. Typically multiple Video Captures should be rendered next to each other in a particular order, or multiple audio channels should be rendered to match different speakers in a particular way. Also, media of different types are often associated with each other, for example a group of Video Captures can be associated with a group of Audio Captures meaning they should be rendered together.

Media Captures of the same media type are associated with each other by grouping them together in a single row of a Capture Set. Media Captures of different media types are associated with each other by putting them in different rows of the same Capture Set.

Since all captures have an area_of_capture associated with them, a consumer can determine the spatial relationships of captures by comparing the locations of their areas of capture with one another.

Association between audio and video can be made by finding audio and video captures which share overlapping areas of capture.

The items (rows) in a capture set represent different alternatives for representing the same Capture Scene. For example the following are alternative ways of capturing the same Capture Scene - two cameras each viewing half of a room, or one camera viewing the whole room, or one stream that automatically captures the person in the

room who is currently speaking. Each row of the Capture Set contains either a single media capture or one group of media captures.

The following example shows a capture set for an endpoint media provider where:

- (VC0, VC1, VC2) - left camera capture, center camera capture, right camera capture
- (VC3) - capture associated with loudest
- (VC4) - zoomed out view of all people in the room
- (AC0) - main audio

The first item in this capture set example is a group of video captures with a spatial relationship to each other. These are VC0, VC1, and VC2. VC3 and VC4 are additional alternatives of how to capture the same room in different ways. The audio capture is included in the same capture set to indicate AC0 is associated with those video captures, meaning the audio should be rendered along with the video in the same set.

The idea is to have sets of captures that represent the same information ("information" in this context might be a set of people and their associated audio / video streams, or might be a presentation supplied by a laptop, perhaps with an accompanying audio commentary). Spatial ordering of media captures is described through the use of attributes.

A media consumer could choose one row of each media type (e.g., audio and video) from a capture set. For example a three stream consumer could choose the first video row plus the audio row, while a single stream consumer could choose the second or third video row plus the audio row. An MCU consumer might choose to receive multiple rows.

The groupsSimultaneous Transmission Set and Encoding Groups as discussed in the next section apply to media captures listed in capture sets. The groupsSimultaneous Transmission Sets and Encoding Groups MUST allow all the Media Captures in a particular row of the capture set to be used simultaneously. But media captures in different rows of the capture set might not be able to be used simultaneously.

6. Choosing Streams

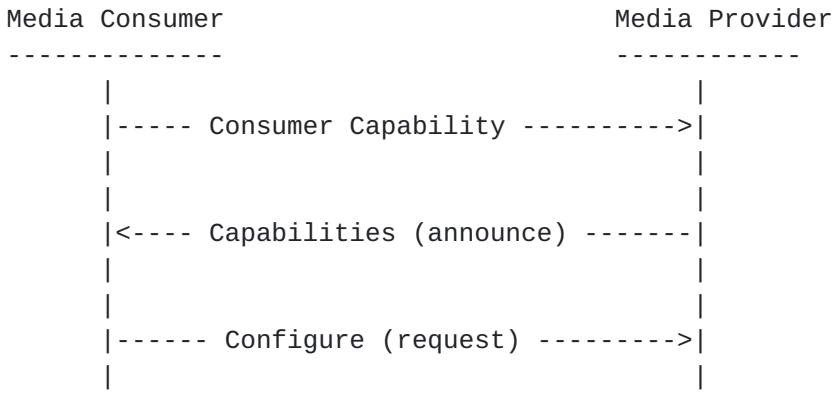
This section describes the process of choosing which streams the

provider sends to the consumer. In order for appropriate streams to be sent from providers to consumers, certain characteristics of the multiple streams must be understood by both providers and consumers. Two separate aspects of streams suffice to describe the necessary information to be shared by providers and consumers. The first aspect we call "physical simultaneity" and the other aspect we refer to as "encoding group". These are described in the following sections, after the message flow is discussed.

6.1. Message Flow

The following diagram shows the flow of messages between a media provider and a media consumer. The provider sends information about its capabilities (as specified in this section), then the consumer chooses which streams it wants, which we refer to as "configure". The consumer sends its own capability message to the provider which may contain information about its own capabilities or restrictions, in which case the provider might tailor its announcements to the consumer.

Diagram for Message Flow



6.1.1. Provider Capabilities Announcement

The provider capabilities announce message includes:

- the list of captures and their attributes
- the list of capture sets
- the list of Simultaneous Transmission Sets
- the list of the encoding groups

6.1.2. Consumer Capability Message

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 being able, 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 set it advertises to be tailored to the consumer.

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

6.1.3. Consumer Configure Request

After receiving a set of video capture information from a provider and making its choice of what media streams to receive based on the consumer's own capabilities and any provider-side simultaneity restrictions, the consumer needs to essentially configure the provider to transmit the chosen set.

The expectation is that this message will enumerate each of the encoding groups and potential encoders within those groups that the consumer wishes to be active (this may well be a subset of the complete set available). For each such encoder within an encoding group, the consumer would specify the video capture (i.e., VC<n> as described above) along with the specifics of the video encoding required, i.e. width, height, frame rate and bit rate. At this stage, the consumer would also provide RTP demultiplexing information as required to distinguish each stream from the others being configured by the same mechanism.

6.2. Physical Simultaneity

An endpoint or MCU can send multiple captures simultaneously. However, there may be constraints that limit which captures can be sent simultaneously with other captures.

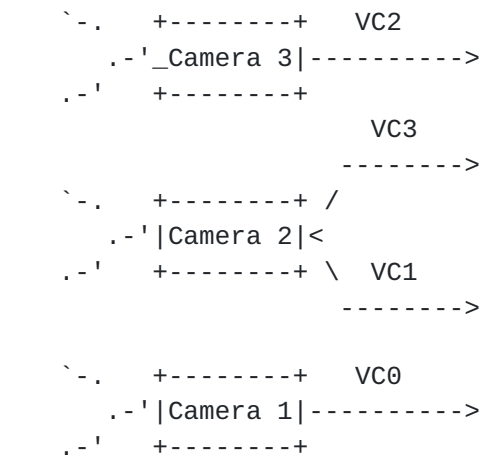
Physical or device simultaneity refers to fact that a device may not be able to be used in different ways at the same time. This shapes the way that offers are made from the provider. The offers are made so that the consumer will choose one of several possible usages of the device. This type of constraint is expressed in Simultaneous Transmission Sets. 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. We refer to this as a physical device simultaneity constraint.

The following illustration shows 3 cameras with 4 video streams. The middle camera can be used as main video zoomed in on 2 people or it could be used in zoomed out mode and capture the whole endpoint. The idea here is that the middle camera cannot be used for both zoomed in and zoomed out captures simultaneously. This is a constraint imposed by the physical limitations of the devices.

Diagram for Simultaneity



VC0- video zoomed in on 2 people VC2- video zoomed in on 2 people
VC1- video zoomed in on 2 people VC3- video zoomed out on 6 people

Simultaneous transmission sets can be expressed as sets of the VCs 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:

{VC0, VC1, VC2}

{VC0, VC3, VC2}

In this example VC0, VC1 and VC2 can be sent OR VC0, VC3 and VC2. Only one set can be transmitted at a time. These are physical capabilities describing what can physically be sent at the same time, not what might make sense to send. For example, in the second set both VC0 and VC2 are redundant if VC3 is included.

In describing its capabilities, the provider must take physical simultaneity into account and send a list of its Simultaneous Transmission Sets to the consumer, along with the Capture Sets and Encoding Groups.

6.3. Encoding Groups

The second aspect of multiple streams that must be understood by providers and consumers in order to create the best experience possible, i. e., for the "right" or "best" streams to be sent, is the encoding characteristics of the possible audio and video streams which can be sent. Just as in the way that a constraint is imposed on the multiple streams due to the physical limitations, there are also constraints due to encoding limitations. These are described by four variables that make up an Encoding Group, as shown in the following table:

Table: Encoding Group

Name	Description
maxBandwidth	Maximum number of bits per second relating to all encodes combined
maxVideoMbps	Maximum number of macroblocks per second relating to a all video encodes combined ((width + 15) / 16) * ((height + 15) / 16) * framesPerSecond
videoEncodes[]	Set of potential video encodes can be generated
audioEncodes[]	Set of potential encodes that can be generated

An encoding group is the basic concept for describing encoding capability. As shown in the Table, it has an overall maxMbps and bandwidth limits, as well as being comprised of sets of individual encodes, which will be described in more detail below.

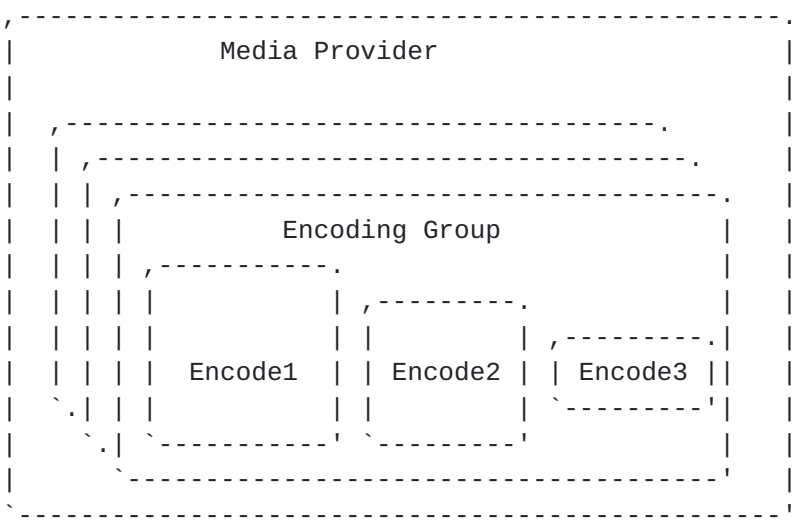
Each media stream provider includes one or more encoding groups. There may be multiple encoding groups per endpoint. For example, each video capture device might have an associated encoding group that describes the video streams that can result from that capture.

A remote receiver (i. e., stream consumer)configures some or all of the specific encodings within one or more groups in order to provide it with media streams to decode.

6.3.1. Encoding Group Structure

This section shows more detail on the media stream provider's encoding group structure. The encoding group includes several individual encodes, each has different encoding values. For example one may be high definition video 1080p60, and another 720p30, with a third being CIF. While a typical 3 codec/display system would have one encoding group per "box", there are many possibilities for the number of encoding groups a provider may be able offer and for what encoding values there are in each encoding group.

Diagram for Encoding Group Structure



As shown in the diagram, each encoding group has multiple potential individual encodes within it. Not all encodes are equally capable, the stream consumer chooses the encodes it wants by configuring the provider to send it what it wants to receive.

Some encoding endpoints are fixed, others are flexible, e. g., a single box with multiple DSPs where the resources are shared.

6.3.2. Individual Encodes

An encoding group is associated with a media capture through the individual encodes, that is, an audio or video capture is encoded in one or more individual encodes, as described by the videoEncodes[] and audioEncodes[]variables.

The following table shows the variables for a Video Encode. (There is a similar table for audio.)

Table: Individual Video Encode

Name	Description
maxBandwidth	Maximum number of bits per second relating to a single video encoding
maxMbps	Maximum number of macroblocks per second relating to a single video encoding: ((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

A remote receiver configures (i. e., instantiates) some or all of the specific encodes such that:

- The configuration of each active ENC<n> does not exceed that individual encode's maxWidth, maxHeight, maxFrameRate.

- The total bandwidth of the configured ENC<n&gtt; does not exceed the maxBandwidth of the encoding group.

- The sum of the macroblocks per second of each configured encode does not exceed the maxMbps attribute of the encoding group.

An equivalent set of attributes holds for audio encodes within an audio encoding group.

6.3.3. More on Encoding Groups

An encoding group EG<n> comprises one or more potential encodings ENC<n>. For example,

```
EG0:  maxMbps=489600, maxBandwidth=6000000
     VIDEO_ENC0: maxWidth=1920, maxHeight=1088, maxFrameRate=60,
                 maxMbps=244800, maxBandwidth=4000000
     VIDEO_ENC1: maxWidth=1920, maxHeight=1088, maxFrameRate=60,
                 maxMbps=244800, maxBandwidth=4000000
     AUDIO_ENC0: maxBandwidth=96000
     AUDIO_ENC1: maxBandwidth=96000
     AUDIO_ENC2: maxBandwidth=96000
```

Here, the encoding group is EG0. It can transmit up to two 1080p30 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, AUDIO_ENC<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 encode groups, as in this illustration.

```
VIDEO_EG0:  maxMbps=489600, maxBandwidth=6000000
     VIDEO_ENC0: maxWidth=1920, maxHeight=1088, maxFrameRate=60,
                 maxMbps=244800, maxBandwidth=4000000
     VIDEO_ENC1: maxWidth=1920, maxHeight=1088, maxFrameRate=60,
                 maxMbps=244800, maxBandwidth=4000000
AUDIO_EG0: maxBandwidth=500000
     AUDIO_ENC0: maxBandwidth=96000
     AUDIO_ENC1: maxBandwidth=96000
     AUDIO_ENC2: maxBandwidth=96000
```

6.3.4. Examples of Encoding Groups

This section illustrates further examples of encoding groups. In the first example, the capability parameters are the same across ENCs. In the second example, they vary.

An endpoint that has 3 similar video capture devices would advertise 3 encoding groups that can each transmit up to 2 1080p30 encodings, as follows:

```
EG0:  maxMbps = 489600, maxBandwidth=6000000
    ENC0: maxWidth=1920, maxHeight=1088, maxFrameRate=60,
          maxMbps=244800, maxBandwidth=4000000
    ENC1: maxWidth=1920, maxHeight=1088, maxFrameRate=60,
          maxMbps=244800, maxBandwidth=4000000
EG1:  maxMbps = 489600, maxBandwidth=6000000
    ENC0: maxWidth=1920, maxHeight=1088, maxFrameRate=60,
          maxMbps=244800, maxBandwidth=4000000
    ENC1: maxWidth=1920, maxHeight=1088, maxFrameRate=60,
          maxMbps=244800, maxBandwidth=4000000
EG2:  maxMbps = 489600, maxBandwidth=6000000
    ENC0: maxWidth=1920, maxHeight=1088, maxFrameRate=60,
          maxMbps=244800, maxBandwidth=4000000
    ENC1: maxWidth=1920, maxHeight=1088, maxFrameRate=60,
          maxMbps=244800, maxBandwidth=4000000
```

A remote consumer configures some or all of the specific encodings such that:

- The configuration of each active ENC<n> parameter values does not cause that encoding's maxWidth, maxHeight, maxFrameRate to be exceeded

- The total bandwidth of the configured ENC <n> encodings does not exceed the maxBandwidth of the encoding group

- The sum of the "macroblocks per second" values of each configured encoding does not exceed the maxMbps of the encoding group

There is no requirement for all encodings within an encoding group to be activated when configured by the consumer.

Depending on the provider's encoding methods, the consumer may be able to request fixed encode values or choose encode values in the range less than the maximum offered. We will discuss consumer behavior in more detail in a section below.

6.3.4.1. Sample video encoding group specification #2

This example specification expresses a system whose encoding groups can each transmit up to 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 (as maxMbps for the entire encoding group is also 489600). Significantly, as up to 3 encodings are available per group, some sets of captures which weren't able to be transmitted simultaneously in example #1 above now become possible, for instance VC1, VC3 and VC6 together. In common

with example #1, all encoding groups have an identical specification.

```
EG0:  maxMbps = 489600, maxBandwidth=6000000
    ENC0: maxWidth=1920, maxHeight=1088, maxFrameRate=60,
          maxMbps=489600, maxBandwidth=4000000
    ENC1: maxWidth=1280, maxHeight=720, maxFrameRate=30,
          maxMbps=108000, maxBandwidth=4000000
    ENC2: maxWidth=960, maxHeight=544, maxFrameRate=30,
          maxMbps=61200, maxBandwidth=4000000
EG1:  maxMbps = 489600, maxBandwidth=6000000
    ENC0: maxWidth=1920, maxHeight=1088, maxFrameRate=60,
          maxMbps=489600, maxBandwidth=4000000
    ENC1: maxWidth=1280, maxHeight=720, maxFrameRate=30,
          maxMbps=108000, maxBandwidth=4000000
    ENC2: maxWidth=960, maxHeight=544, maxFrameRate=30,
          maxMbps=61200, maxBandwidth=4000000
EG2:  maxMbps = 489600, maxBandwidth=6000000
    ENC0: maxWidth=1920, maxHeight=1088, maxFrameRate=60,
          maxMbps=489600, maxBandwidth=4000000
    ENC1: maxWidth=1280, maxHeight=720, maxFrameRate=30,
          maxMbps=108000, maxBandwidth=4000000
    ENC2: maxWidth=960, maxHeight=544, maxFrameRate=30,
          maxMbps=61200, maxBandwidth=4000000
```

7. Using the Framework

This section shows 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.

Consider an endpoint with the following characteristics:

- 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. The Encode Group specifications can be found above in Section 6.3.4.1, Sample video encoding group specification #2.

Video Captures:

- VC0- (the left camera stream), encoding group:EG0, attributes: purpose=main;auto-switched:no; area_of_capture={xBegin=0, xEnd=33}

- VC1- (the center camera stream), encoding group:EG1, attributes: purpose=main; auto-switched:no; area_of_capture={xBegin=33, xEnd=66}

- VC2- (the right camera stream), encoding group:EG2, attributes: purpose=main;auto-switched:no; area_of_capture={xBegin=66, xEnd=99}

- VC3- (the loudest panel stream), encoding group:EG1, attributes: purpose=main;auto-switched:yes; area_of_capture={xBegin=0, xEnd=99}

- VC4- (the loudest panel stream with PiPs), encoding group:EG1, attributes: purpose=main; composed=true; auto-switched:yes; area_of_capture={xBegin=0, xEnd=99}

- VC5- (the zoomed out view of all people in the room), encoding group:EG1, attributes: purpose=main;auto-switched:no; area_of_capture={xBegin=0, xEnd=99}

- VC6- (presentation stream), encoding group:EG1, attributes: purpose=presentation;auto-switched:no; area_of_capture={xBegin=0, xEnd=99}

Summary of video captures - 3 codecs, center one is used for center camera stream, presentation stream, auto-switched, and zoomed views.

Note the text in parentheses (e.g. "the 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.

[edt. It is arbitrary that for this example the alternative views are on EG1 - they could have been spread out- it was not a necessary choice.]

Audio Captures:

- AC0 (left), attributes: purpose=main;channel format=mono; area_of_capture={xBegin=0, xEnd=33}

- AC1 (right), attributes: purpose=main;channel format=mono; area_of_capture={xBegin=66, xEnd=99}

- AC2 (center) attributes: purpose=main;channel format=mono; area_of_capture={xBegin=33, xEnd=66}

- AC3 being a simple pre-mixed audio stream from the room (mono), attributes: purpose=main;channel format=mono; mixed=true; area_of_capture={xBegin=0, xEnd=99}

- AC4 audio stream associated with the presentation video (mono) attributes: purpose=presentation;channel format=mono; area_of_capture={xBegin=0, xEnd=99}

The physical simultaneity information is:

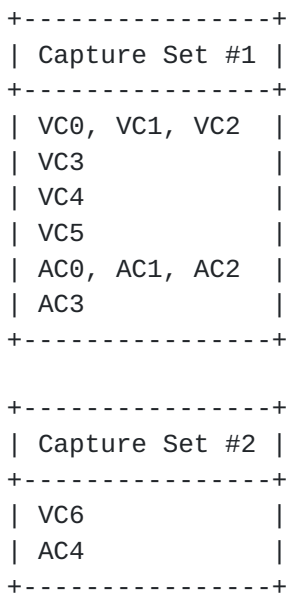
{VC0, VC1, VC2, VC3, VC4, VC6}

{VC0, VC2, VC5, VC6}

It is possible to select any or all of the rows in a capture set. This is strictly what is possible from the devices. However, 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 Capture list and encoding groups, not in the simultaneous transmission sets.)

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

The following table represents the capture sets for this provider. Recall that a capture set is composed of alternative captures covering the same scene. Capture Set #1 is for the main people captures, and Capture Set #2 is for presentation.



Different capture sets are unique to each other, non-overlapping. A consumer chooses a capture row from each capture set. 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 rows of audio in Capture Set #1 represent the "same thing", in that one way to receive the audio is with the 3 linear position audio captures (AC0, AC1, AC2), and another way is with the single channel monaural format AC3. The Media Consumer would choose the one audio capture row it is capable of receiving.

The spatial ordering is understood by the media capture attributes area and point of capture.

The consumer finds a "row" in each capture set #x section of the table that it wants. It configures the streams according to the encoding group for the row.

A Media Consumer would likely want to choose a row 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 row of Capture Set #1 (VC0, VC1, VC2) and not receive the other rows. A consumer that can receive only one people stream would probably choose one of the other rows.

If the consumer can receive a presentation stream too, it would also

choose to receive the only row from Capture Set #2 (VC6).

7.1. The MCU Case

This section shows how an MCU might express its Capture Sets, 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 Set #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 Set #2	note
VC10	video capture for presentation
AC1	presentation audio to accompany VC10

7.2. Media Consumer Behavior

[Edt. Should this be moved to appendix?]

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 set advertised by the provider.

A sane, basic, algorithm might be for the consumer to go through each capture set 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 rows in the video capture sets 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 encode groups in order to make best use of the available network bandwidth and its own decoding capabilities.

7.2.1. One screen consumer

VC3, VC4 and VC5 are all on different rows by themselves, not in a group, 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.

7.2.2. 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:

v

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.

7.2.3. 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 encode group's capability by 3 to determine what resolution and frame rate to configure the provider with or to configure the individual video captures' encode 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).

8. Acknowledgements

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

9. IANA Considerations

TBD

10. Security Considerations

TBD

11. 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.

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

Appendix A. Open Issues

A.1. Video layout arrangements and centralized composition

In the context of a conference with a central MCU, there has been discussion about a consumer requesting the provider to provide a certain type of layout arrangement or perform a certain composition algorithm, such as combining some number of most recent talkers, or producing a video layout using a 2x2 grid or 1 large cell with 5 smaller cells around it. The current framework does not address this. It isn't clear if this topic should be included in this framework, or maybe a different part of CLUE, or maybe outside of CLUE altogether.

A.2. Source is selectable

A Boolean variable. True indicates the media consumer can request a particular media source be mapped to a media capture. Default is false.

TBD - how does the consumer make the request for a particular source? How does the consumer know what is available? Need to explain better how multiple media captures are different from a single media capture with choices for the source, and when each concept should be used.

A.3. Media Source Selection

The use cases include a case where the person at a receiving endpoint can request to receive media from a particular other endpoint, for example in a multipoint call to request to receive the video from a certain section of a certain room, whether or not people there are talking.

TBD - this framework should address this case. Maybe need a roster list of rooms or people in the conference, with a mechanism to select from the roster and associate it with media captures. This is different from selecting a particular media capture from a capture set. The mechanism to do this will probably need to be different than selecting media captures based on capture sets and attributes.

A.4. Endpoint requesting many streams from MCU

TBD - how to do VC selection for a system where the endpoint media consumers want to receive lots of streams and do their own composition, rather than MCU doing transcoding and composing. Example is 3 screen consumer that wants 3 large loudest speaker streams, and a bunch of small ones to render as PiP. How the small ones are chosen, which could potentially be chosen by either the endpoint or MCU. There are other more complicated examples also. Is the current framework adequate to support this?

A.5. VAD (voice activity detection) tagging of audio streams

TBD - do we want to have VAD be mandatory? All audio streams originating from a media provider must be tagged with VAD information. This tagging would include an overall energy value for the stream plus information on which sections of the capture scene are "active".

Each audio stream which forms a constituent of a row within a capture set should include this tagging, and the energy value within it calculated using a fixed, consistent algorithm.

When a system determines the most active area of a capture scene (either "loudest", or determined by other means such as a button press) it should convey that information to the corresponding media stream consumer via any audio streams being sent within that capture set. Specifically, there should be a list of active linear positions and their VAD characteristics within the audio stream in addition to the overall VAD information for the capture set. This is to ensure all media stream consumers receive the same, consistent, audio energy information whichever audio capture or captures they choose to receive for a capture set. Additionally, linear position information can be mapped to video captures by a media stream consumer in order that it can perform "panel switching" if required.

A.6. Private Information

Authors' Addresses

Allyn Romanow
Cisco Systems
San Jose, CA 95134
USA

Email: allyn@cisco.com

Mark Duckworth
Polycom
Andover, MA 01810
US

Email: mark.duckworth@polycom.com

Andrew Pepperell
Cisco Systems
Langley, England
UK

Email: apeppere@cisco.com

Brian Baldino
Cisco Systems
San Jose, CA 95134
US

Email: bbaldino@cisco.com