ETSI TR 126 962 V17.0.0 (2022-04)

TECHNICAL REPORT

5G;
Immersive Teleconferencing and Telepresence for Remote Terminals (ITT4RT) Operation and Usage Guidelines
(3GPP TR 26.962 version 17.0.0 Release 17)

Reference

DTR/TSGS-0426962vH00

Keywords

5G

ETSI

650 Route des Lucioles
F-06921 Sophia Antipolis Cedex - FRANCE

Tel.: +33 4 92 94 42 00 Fax: +33 4 93 65 47 16

Siret N° 348 623 562 00017 - APE 7112B
Association à but non lucratif enregistrée à la
Sous-Préfecture de Grasse (06) N° w061004871

Important notice

The present document can be downloaded from:
http://www.etsi.org/standards-search

The present document may be made available in electronic versions and/or in print. The content of any electronic and/or print versions of the present document shall not be modified without the prior written authorization of ETSI. In case of any existing or perceived difference in contents between such versions and/or in print, the prevailing version of an ETSI deliverable is the one made publicly available in PDF format at www.etsi.org/deliver.

Users of the present document should be aware that the document may be subject to revision or change of status. Information on the current status of this and other ETSI documents is available at https://portal.etsi.org/TB/ETSIDeliverableStatus.aspx

If you find errors in the present document, please send your comment to one of the following services:
https://portal.etsi.org/People/CommiteeSupportStaff.aspx

If you find a security vulnerability in the present document, please report it through our Coordinated Vulnerability Disclosure Program:
https://www.etsi.org/standards/coordinated-vulnerability-disclosure

Notice of disclaimer & limitation of liability

The information provided in the present deliverable is directed solely to professionals who have the appropriate degree of experience to understand and interpret its content in accordance with generally accepted engineering or other professional standard and applicable regulations.
No recommendation as to products and services or vendors is made or should be implied.
No representation or warranty is made that this deliverable is technically accurate or sufficient or conforms to any law and/or governmental rule and/or regulation and further, no representation or warranty is made of merchantability or fitness for any particular purpose or against infringement of intellectual property rights.
In no event shall ETSI be held liable for loss of profits or any other incidental or consequential damages.

Any software contained in this deliverable is provided "AS IS" with no warranties, express or implied, including but not limited to, the warranties of merchantability, fitness for a particular purpose and non-infringement of intellectual property rights and ETSI shall not be held liable in any event for any damages whatsoever (including, without limitation, damages for loss of profits, business interruption, loss of information, or any other pecuniary loss) arising out of or related to the use of or inability to use the software.

Copyright Notification

Intellectual Property Rights

Essential patents

IPRs essential or potentially essential to normative deliverables may have been declared to ETSI. The declarations pertaining to these essential IPRs, if any, are publicly available for **ETSI members and non-members**, and can be found in ETSI SR 000 314: *"Intellectual Property Rights (IPRs); Essential, or potentially Essential, IPRs notified to ETSI in respect of ETSI standards"*, which is available from the ETSI Secretariat. Latest updates are available on the ETSI Web server (https://ipr.etsi.org/).

Pursuant to the ETSI Directives including the ETSI IPR Policy, no investigation regarding the essentiality of IPRs, including IPR searches, has been carried out by ETSI. No guarantee can be given as to the existence of other IPRs not referenced in ETSI SR 000 314 (or the updates on the ETSI Web server) which are, or may be, or may become, essential to the present document.

Trademarks

The present document may include trademarks and/or tradenames which are asserted and/or registered by their owners. ETSI claims no ownership of these except for any which are indicated as being the property of ETSI, and conveys no right to use or reproduce any trademark and/or tradename. Mention of those trademarks in the present document does not constitute an endorsement by ETSI of products, services or organizations associated with those trademarks.

DECT™, **PLUGTESTS™**, **UMTS™** and the ETSI logo are trademarks of ETSI registered for the benefit of its Members. **3GPP™** and **LTE™** are trademarks of ETSI registered for the benefit of its Members and of the 3GPP Organizational Partners. **oneM2M™** logo is a trademark of ETSI registered for the benefit of its Members and of the oneM2M Partners. **GSM**® and the GSM logo are trademarks registered and owned by the GSM Association.

Legal Notice

This Technical Specification (TS) has been produced by ETSI 3rd Generation Partnership Project (3GPP).

The present document may refer to technical specifications or reports using their 3GPP identities. These shall be interpreted as being references to the corresponding ETSI deliverables.

The cross reference between 3GPP and ETSI identities can be found under http://webapp.etsi.org/key/queryform.asp.

Modal verbs terminology

In the present document "**shall**", "**shall not**", "**should**", "**should not**", "**may**", "**need not**", "**will**", "**will not**", "**can**" and "**cannot**" are to be interpreted as described in clause 3.2 of the ETSI Drafting Rules (Verbal forms for the expression of provisions).

"**must**" and "**must not**" are **NOT** allowed in ETSI deliverables except when used in direct citation.

Contents

Foreword

This Technical Report has been produced by the 3rd Generation Partnership Project (3GPP).

The contents of the present document are subject to continuing work within the TSG and may change following formal TSG approval. Should the TSG modify the contents of the present document, it will be re-released by the TSG with an identifying change of release date and an increase in version number as follows:

Version x.y.z

where:

x the first digit:

1 presented to TSG for information;

2 presented to TSG for approval;

3 or greater indicates TSG approved document under change control.

Y the second digit is incremented for all changes of substance, i.e. technical enhancements, corrections, updates, etc.

z the third digit is incremented when editorial only changes have been incorporated in the document.

In the present document, modal verbs have the following meanings:

shall indicates a mandatory requirement to do something

shall not indicates an interdiction (prohibition) to do something

The constructions “shall” and “shall not” are confined to the context of normative provisions, and do not appear in Technical Reports.

The constructions “must” and “must not” are not used as substitutes for “shall” and “shall not”. Their use is avoided insofar as possible, and they are not used in a normative context except in a direct citation from an external, referenced, non-3GPP document, or so as to maintain continuity of style when extending or modifying the provisions of such a referenced document.

Should indicates a recommendation to do something

should not indicates a recommendation not to do something

may indicates permission to do something

need not indicates permission not to do something

The construction “may not” is ambiguous and is not used in normative elements. The unambiguous constructions “might not” or “shall not” are used instead, depending upon the meaning intended.

Can indicates that something is possible

cannot indicates that something is impossible

The constructions “can” and “cannot” are not substitutes for “may” and “need not”.

Will indicates that something is certain or expected to happen as a result of action taken by an agency the behaviour of which is outside the scope of the present document

will not indicates that something is certain or expected not to happen as a result of action taken by an agency the behaviour of which is outside the scope of the present document

might indicates a likelihood that something will happen as a result of action taken by some agency the behaviour of which is outside the scope of the present document

might not indicates a likelihood that something will not happen as a result of action taken by some agency the behaviour of which is outside the scope of the present document

In addition:

is (or any other verb in the indicative mood) indicates a statement of fact

is not (or any other negative verb in the indicative mood) indicates a statement of fact

The constructions “is” and “is not” do not indicate requirements.

1 Scope

The present document describes operation and usage guidelines for Immersive Teleconferencing and Telepresence for Remote Terminals (ITT4RT). It is intended to help implementers of the ITT4RT functionality of TS 26.114 for realizing Virtual Reality conversational services with unidirectional omnidirectional video.

2 References

The following documents contain provisions which, through reference in this text, constitute provisions of the present document.

- References are either specific (identified by date of publication, edition number, version number, etc.) or non-specific.
- For a specific reference, subsequent revisions do not apply.
- For a non-specific reference, the latest version applies. In the case of a reference to a 3GPP document (including a GSM document), a non-specific reference implicitly refers to the latest version of that document *in the same Release as the present document*.

[1] 3GPP TR 21.905: “Vocabulary for 3GPP Specifications”.

[2] M. M. Hannuksela, Y.-K. Wang, and A. Hourunranta, “An overview of the OMAF standard for 360° video,” Data Compression Conference, Mar. 2019.

[3] N19274, Potential improvement of OMAF, MPEG 130, April 2020.

[4] Alireza Zare, Alireza Aminlou, and Miska M. Hannuksela. 2018. 6K Effective Resolution with 4K HEVC Decoding Capability for OMAF-compliant 360° Video Streaming. In Proceedings of the 23rd Packet Video Workshop (PV ’18). Association for Computing Machinery, New York, NY, USA, 72–77. DOI:https://doi.org/10.1145/3210424.3210425

[5] High Efficiency Video Coding (HEVC) Test Model 16 (HM 16) Improved Encoder Description Update 14.

[6] J. van der Hooft, M. Torres Vega, S. Petrangeli, T. Wauters and F. De Turck, “Quality Assessment for Adaptive Virtual Reality Video Streaming: A Probabilistic Approach on the User’s Gaze,” 2019 22nd Conference on Innovation in Clouds, Internet and Networks and Workshops (ICIN), 2019, pp. 19-24, doi: 10.1109/ICIN.2019.8685904.

[7] RFC 3986, Uniform Resource Identifier (URI): Generic Syntax, IETF, January 2005.

[8] ISO/IEC DIS 23090-14, Information technology — Coded representation of immersive media — Part 14: Scene Description for MPEG Media.

3 Definitions of terms, symbols and abbreviations

.

3.1 Terms

For the purposes of the present document, the terms given in 3GPP TR 21.905 [1] and the following apply. A term defined in the present document takes precedence over the definition of the same term, if any, in 3GPP TR 21.905 [1].

No new terms defined for this TR.

3.2 Abbreviations

For the purposes of the present document, the abbreviations given in 3GPP TR 21.905 [1] and the following apply. An abbreviation defined in the present document takes precedence over the definition of the same abbreviation, if any, in 3GPP TR 21.905 [1].

ML/AI	Machine Learning / Artificial Intelligence
CU	Coding Unit
ERP	EquiRectangular Projection
GOP	Group of Pictures
GSM	Global System for Mobile communications
HEVC	High Efficiency Video Coding
HMD	Head-Mounted Display
HQ	High Quality
INTRA	refers to intra-frame coding
LQ	Low Quality
LTE	Long-Term Evolution
MCU	Multipoint Control Unit
MPEG	Moving Picture Experts Group
MRF	Multimedia Resource Function
MRFC	Multimedia Resource Function Controller
OMAF	Omnidirectional MediA Format
PPM	Packed Picture Mapping
QP	Quantization Parameter
RAP	Random Access Point
RFC	Request for Comment
RTP	Real-time Transport Protocol
RTCP	Real-time Transport (Control) Protocol
RTT	Round Trip Time
RWP	Region-Wise Packing
SDP	Session Description Protocol
SEI	Supplemental Enhancement Informatio
UE	User Equipment
UMTS	Universal Mobile Telecommunications System
VDP	Viewport Dependent Processing
VL	Viewport-Locked

4 Viewport Dependent Processing (VDP)

4.1 Introduction

NOTE: Considerations for interoperability of ITT4RT clients using different VDP options need to be considered and normative text, if any.

An ITT4RT-Tx client may offer Viewport Dependent Processing for delivering bandwidth-efficient 360-degree video to ITT4RT-Rx clients when both ends have successfully negotiated the required capabilities. There may be several ways in which the ITT4RT-Tx client may support VDP. This section lists some ways as informational with appropriate required signalling. This is a non-exhaustive list for guidance. Implementations may use other methods not defined here or a combination as long as the appropriate capabilities and required signalling is supported by ITT4RT clients.

4.2 360-degree Video Optimized for the Viewport

An ITT4RT-Tx client may deliver the full 360-degree video as a single encoded stream with higher quality in the viewport region as shown in Figure 4.1. This may be implemented by encoding parameters which provide higher quality in the viewport region compared to the other regions. Compared to when only the viewport is sent, the bitrate savings are limited in this case because the full 360-degree video is delivered. However, a delay in viewport update does not cause disruption in the viewing experience in terms of content flow, but just a temporary lowered quality of the visible content during head motion.

The case described here is to provide an omnidirectional video to receiver UEs without the need for unpacking operation(s). It is noted that region-wise packing may still be used with such encoding, if desired. Tiling, which may lead to lowered encoding efficiency, is not required in this case but multi-quality tiles may still be used without need for packing.

In this case, the full 360-degree video which has been optimized for the current viewport is delivered to the receiver, so that the viewport is at a higher quality as shown in Figure 1. For the sake of scalability, an MRF/MCU delivering content to multiple end users may receive viewport independent 360-video from the content source, and encode multiple versions of the content optimized for different viewing orientations, and deliver the version most suitable to the current viewport of each user.

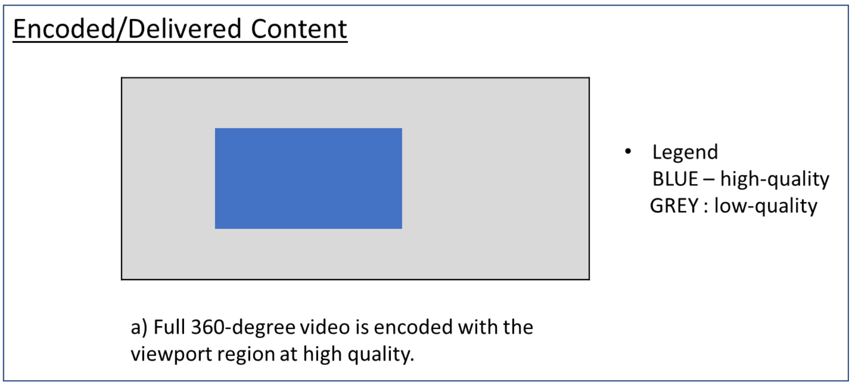

Encoded/Delivered Content
• Legend
BLUE – high-quality
GREY : low-quality
a) Full 360-degree video is encoded with the viewport region at high quality.

Figure 1: Full 360-degree video is encoded, optimized and delivered for the current viewport region.

Signalling and Delivery

The ITT4RT-Rx client does not require any additional signalling from the ITT4RT-Tx client to decode and render the stream apart from indicating in the SDP offer/answer that VDP will be used. The resolution in the attribute “imageattr” corresponds to resolution of the full 360-degree video.

Full Processing Pipeline

Figure 2 shows the processing pipeline for this case. The ITT4RT-Tx client receives an image from a capture device and stitches it, if not pre-stitched. It then determines the viewport of the receiver based on RTCP viewport feedback. Once the viewport has been determined, the video is encoded with higher quality in the viewport region, packetized and delivered to the ITT4RT-Rx client. The ITT4RT-Rx client decodes and renders the received video. No additional signalling is needed.

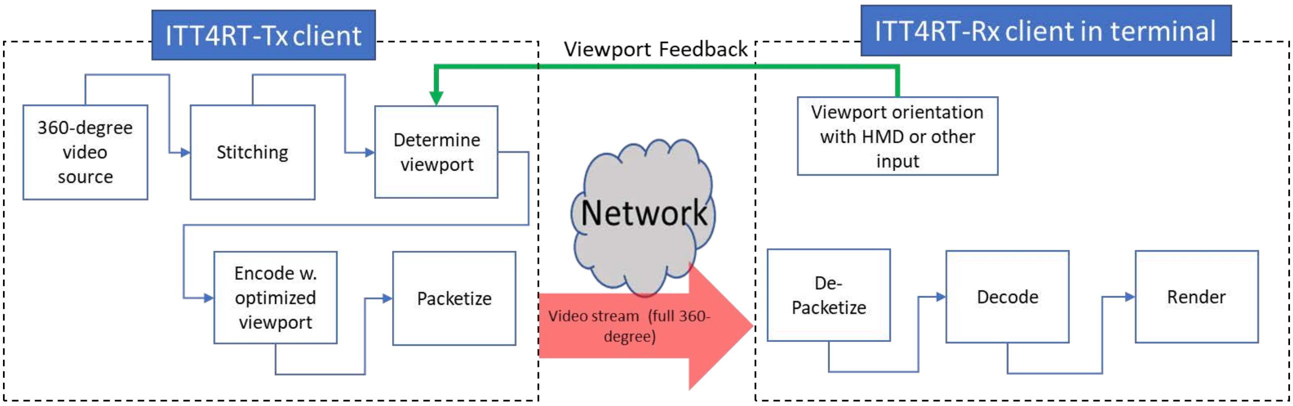
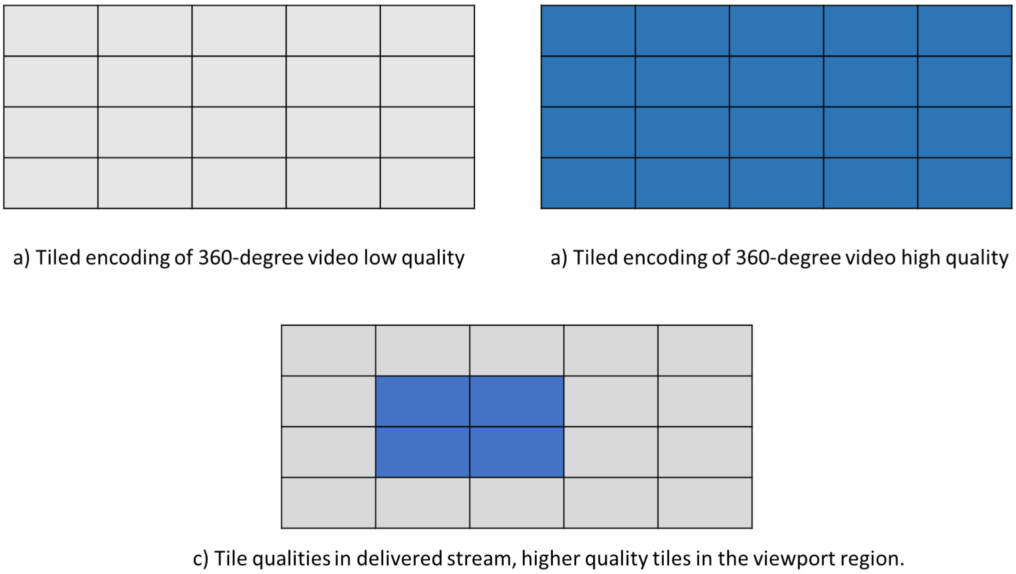

ITT4RT-Tx client
360-degree video source
Stitching
Determine viewport
Encode w. optimized viewport
Packetize
Viewport Feedback
Network
Video stream (full 360-degree)
ITT4RT-Rx client in terminal
Viewport orientation with HMD or other input
De-Packetize
Decode
Render

Figure 2: Processing pipeline for 360-degree video with optimized viewport solution.

4.3 Tiled Video with Multiple Quality Levels

An ITT4RT-Tx client may use tiled encoding, e.g., HEVC motion-constrained tiling, to create a tiled low-quality version of the 360-degree video and a tiled high-quality version of the 360-degree video (note that more than two quality levels for the encodings may be created). The delivered stream consists of high-quality tiles for the viewport region and low-quality tiles for the non-viewport region. When the low-quality and high-quality tiles are encoded using low fidelity and high fidelity, we refer to this as the mixed-quality tiled encoding approach. Whereas if the low-quality and high-quality tiles are created using lower resolution and higher resolution respectively, we refer to this as the mixed-resolution tiled encoding approach. Figure 3 illustrates the concept.



a) Tiled encoding of 360-degree video low quality

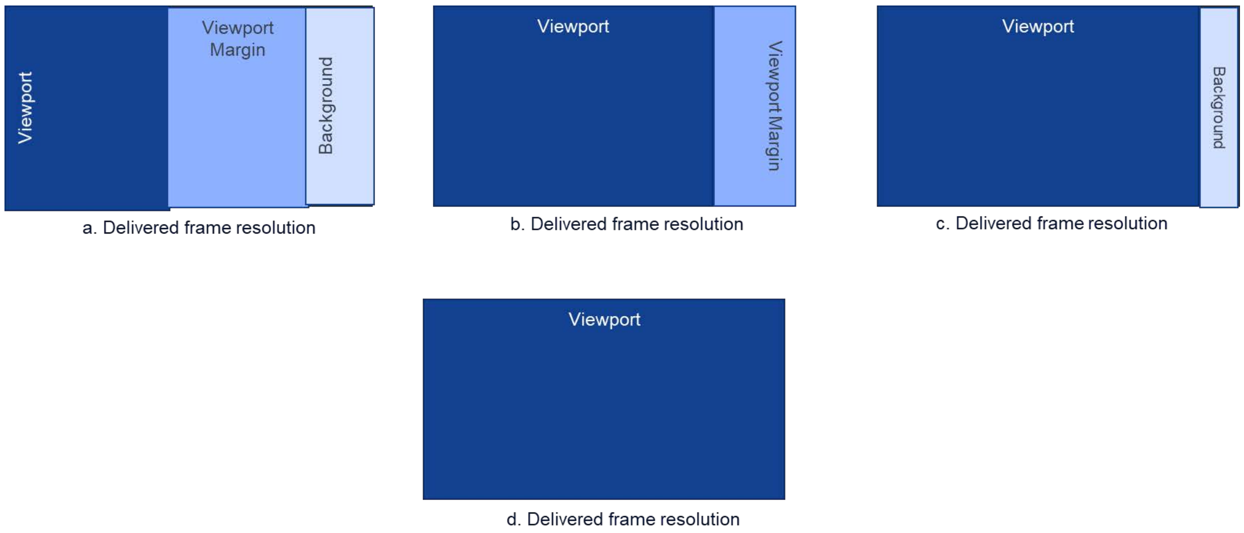
a) Tiled encoding of 360-degree video high quality

c) Tile qualities in delivered stream, higher quality tiles in the viewport region.

Figure 3: Tiled encoding is used to generate two versions of the full 360-degree video. The delivered stream consists of high-quality tiles in the viewport region and low-quality tiles in the non-viewport regions.

Mixed-resolution may be used to increase the effective viewport resolution of viewport for devices with limited decoding capability [4] (e.g., effective 6K viewport resolution with 4K viewport resolution). The mixed-resolution and mixed-quality client operation implementation is described in [2]. The slice header and other bitstream rewriting required to create a conformant HEVC bitstream which can be decoded by a single decoder is described in OMAF 2nd edition draft (MPEG N19274), clause 4.6.4.2 and 10.1.5.4 [3]. An ITT4RT-Tx client that uses mixed-resolution requires region-wise packing. The RWP SEI message is used to carry the information about the regions in the packed picture. Figure 4 illustrates different packing schemes that may be used by an ITT4RT-Tx client. An ITT4RT-Tx client may change the packing scheme based on the amount, frequency or probability of head motion. For example, an ITT4RT-Tx client may use margins as shown in a and b in the figure when head motion or probability of head motion is high and c and d in the figure when it is low.

The main advantage of tiled encoding as described in this section is scalability, making it more suitable for Implementation in an MRF. For example, a high-quality viewport-independent 360-degree video is delivered to an MRF, which re-encodes and delivers viewport-dependent 360-degree video to multiple ITT4RT-Rx clients The MRF does not have to produce content for each receiver individually, but the operations are limited to picking the right high quality tiles and low-quality tiles from the two versions of the video, and “assemble it” for each ITT4RT-Rx client based on its current viewport. However, tiled video has lower coding efficiency as inter-picture prediction is limited to each tile. Furthermore, the storage and processing requirements for encoding tiled video at multiple qualities may be unsuitable for smaller conference sizes and ITT4RT-Tx clients in terminal.



a. Delivered frame resolution

b. Delivered frame resolution

c. Delivered frame resolution

d. Delivered frame resolution

Figure 4: Region-wise packing for mixed-resolution tiles is shown with different packing schemes for viewport, viewport margins and background tiles. Each region comprises one or more tiles. The tiles in different regions may be encoded at different resolutions. The regions within the packed picture may be static or variant for a video stream.

Signalling and Delivery

All tiles can be packaged and delivered in a single RTP stream and can be decoded using a single decoder when all tiles are decoded together. In case of mixed-resolution, packing information needs to be signaled to the receiver, which uses it for rendering after decoding. The SEI messages for region-wise packing (RWP) can be used to carry the information about packed picture mapping to assist the receiver in the understanding of the high-quality and low-quality areas.

In the SDP, in addition to indicating VDP, the ITT4RT-Tx client will indicate it is using tiled encoding with mixed-quality or mixed-resolution using the PPM parameter.

The resolution in imageattr corresponds to the resolution of the full picture.

Full Processing Pipeline

The full processing pipeline is shown in Figure 5. In this case, the encoding process is the same regardless of the viewport of the ITT4RT-Rx client. Once encoding is done, the ITT4RT-Tx client can determine the viewport and select the tile qualities based on the viewport region. The tiles are packed and delivered to the ITT4RT-Rx client, which decodes and renders. The stream would carry information about RWP (using SEI) to assist the receiver in unpacking. The unpacking is part of the rendering process and may require upsampling tiles and arranging the upsampled tiles into the projected format in the mixed-resolution case. The packing/unpacking step and signalling packed picture information in the stream is not needed for the mixed-quality case.

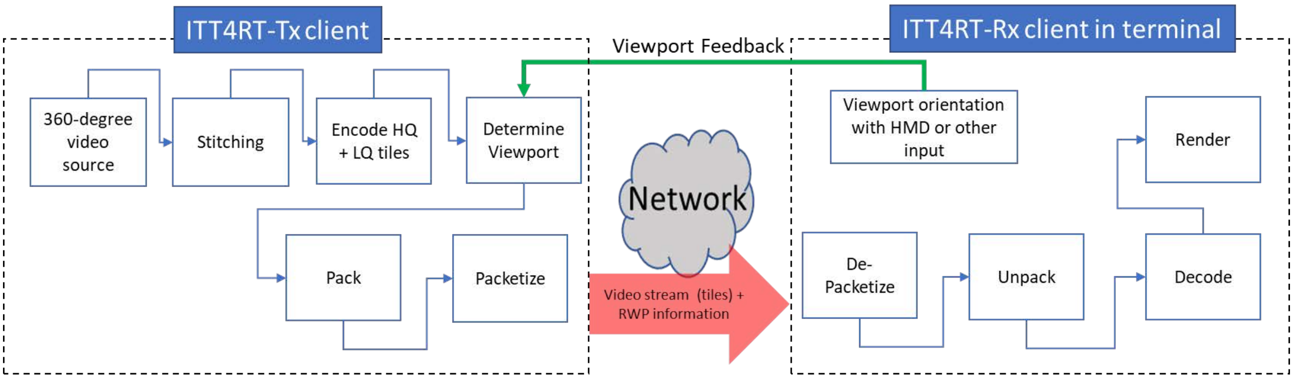


Figure 5: Processing pipeline for 360-degree video with tiled encoding. Packing and RWP information may not be needed in cased of mixed-quality.

4.4 Viewport-only Delivery

For maximim bandwidth savings, the ITT4RT-Tx client may deliver only an extracted high-quality region corresponding to the viewport of the ITT4RT-Rx client. If the viewport region (with or without a viewport margin) is extracted from a projected picture (e.g., ERP), the resolution would change depending on where the viewport is located on the picture. To prevent changing resolution, the ITT4RT-Tx client may rotate the sphere to re-orient the selected viewport to the center of the projected picture. Figure 6 illustrates the rotated and centered viewport extracted from the 360-degree video. The delivered video may be rendered in one of two ways, which affect the signalling requirements i) sphere-locked, where the ITT4RT-Rx client reverses rotation of the received image before rendering, and ii) viewport-locked, where the ITT4RT-Rx client always renders the received image to the center of the viewport. Note that sphere-locked operation implies that it is necessary to map the delivered content to the global coordinate (real capture position) of the stitched 360-degree video. This is the default behaviour when VDP is used without the VL parameter regardless of the VDP method.

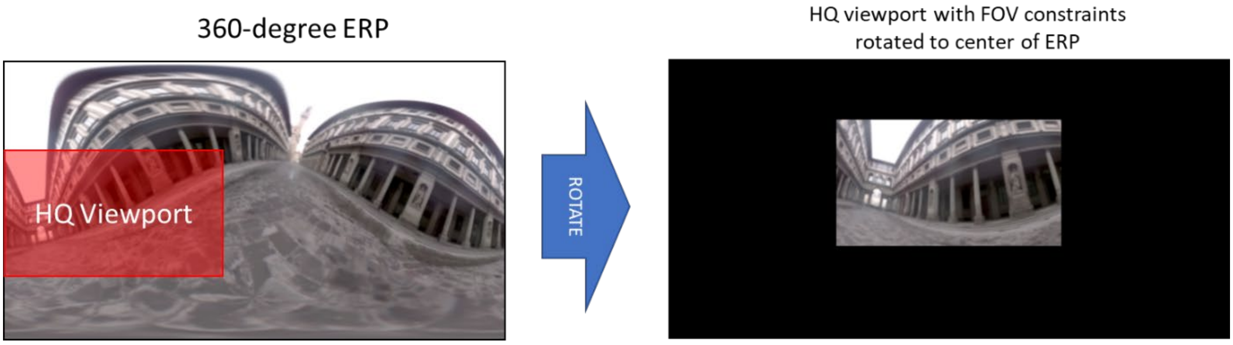


Figure 6: The 360-degree video is rotated with respect to the original capture orientation to bring the viewport to the center of the ERP in order to ensure the same resolution for the viewport region.

4.4.1 Sphere-locked

For the ITT4RT-Rx client to be able to display the received viewport correctly in reference to the capture orientation (sphere-locked), the ITT4RT-Tx client can signal the rotation information to the ITT4RT-Rx client using the rotation SEI message.. The ITT4RT-Rx client can then reverse the rotation before rendering. The solution also requires the region-wise packing SEI message to indicate the size of the projected region, which may change if different sizes of margins are used during the session.

Signalling and Delivery

The video is delivered as a single stream. The SDP signalling needs to indicate VDP and rotation and region-wise packing SEI message are used.

The resolution in the imageattr corresponds to the resolution of the delivered viewport region.

Full Processing Pipeline

The full process is illustrated in Figure 7. The ITT4RT-Tx client receives an image from a capture device and stitches it, if not pre-stitched. It then determines the viewport region (with or without a viewport margin) of the ITT4RT-Rx client based on RTCP viewport feedback. Once the viewport region has been determined, the center of the viewport region is rotated to the center of the projected picture and encoded. ITT4RT-Rx client always receives a constant sized, constant resolution image for the viewport region. The sender is expected to include information about the selected viewport with the stream so that the receiver is able to reverse the rotation applied by the sender.

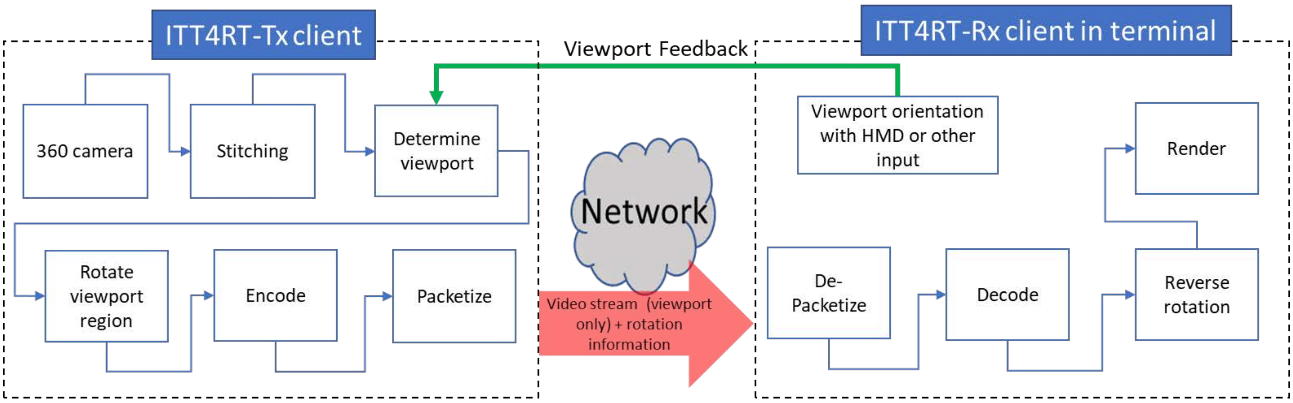


Figure 7: A high-quality viewport-only solution with sphere-locked rendering

4.4.2 Viewport-Locked

The viewport only solution can be rendered in the viewport-locked mode. In this case, the received viewport does not need to be mapped to the original capture orientation, i.e., there is no need to reverse the rotation. Instead, the ITT4RT-Rx client can render the received video as viewport-locked, i.e., centered at the center of the viewport/display.

The solution is suitable for viewport followers that are following the viewport orientation of another device and any 2D display devices, as a delayed viewport update (at least 1 RTT) does not significantly lower the user experience. In case of ultra low-latency networks, the solution can be used for HMDs, but may cause motion sickness if latency increases and, hence, is expected to be used carefully.

Signalling and Delivery

The viewport locked mode does not require any bit stream signalling making it lightweight as well as faster to render due to the absence of the reverse rotation step. ITT4RT clients using viewport-locked rendering are required to know when viewport-locked rendering is used so that appropriate signalling for rotation SEI messages are not included in the bitstream and are not processed. To enable interoperability, viewport-locked needs to be defined as either the default mechanism for a type of device or it requires session level signalling to indicate that this mode is in use.

Full Processing Pipeline

The process as shown in Figure 8 is the same as sphere-locked viewport only solution, except the rotation information is not signalled and the ITT4RT-Rx client renders the received video such that the center of the image is aligned to the center of the current viewport.

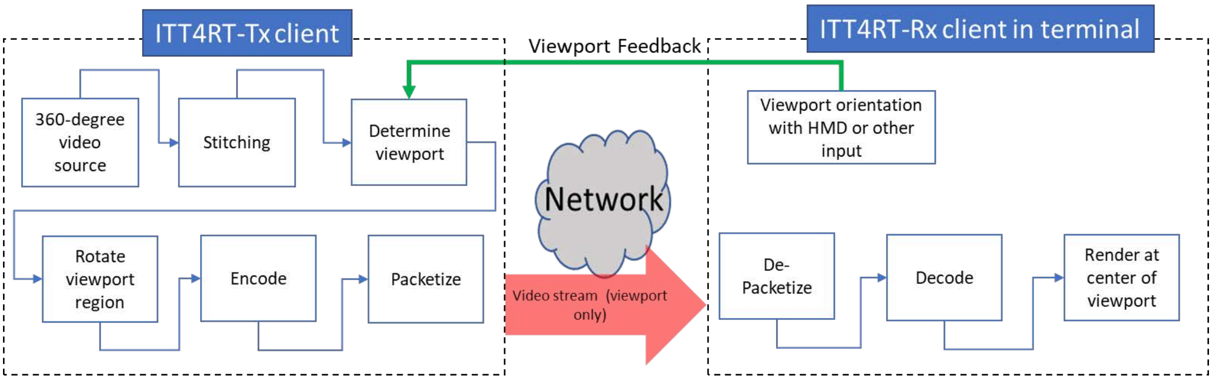


Figure 8: A high-quality viewport only, viewport-locked (VL) solution. The viewport feedback may be received from a different UE in case of follower uEs.

4.5 HQ Viewport with LQ background

A viewport-only solution, like the one described in 4.3, can be combined with a low-quality (LQ) viewport-independent 360-degree video as background to deliver a more continuous viewing experience in case of head motion. The LQ viewport-independent 360-degree video is frame packed with the viewport region (with or without margin) at a higher quality (HQ) and delivered as a single stream as shown in Figure 9. Since the size and shape of the pictures remain same, the packing information is signaled in SDP, using the Packed Picture Mapping (PPM) parameter as defined in TS 26.114 Section Y.6.2.4. The ITT4RT-Rx client would render the HQ viewport region where it overlaps with the LQ background. Further details on the signalling and pipeline of a single stream case follow.

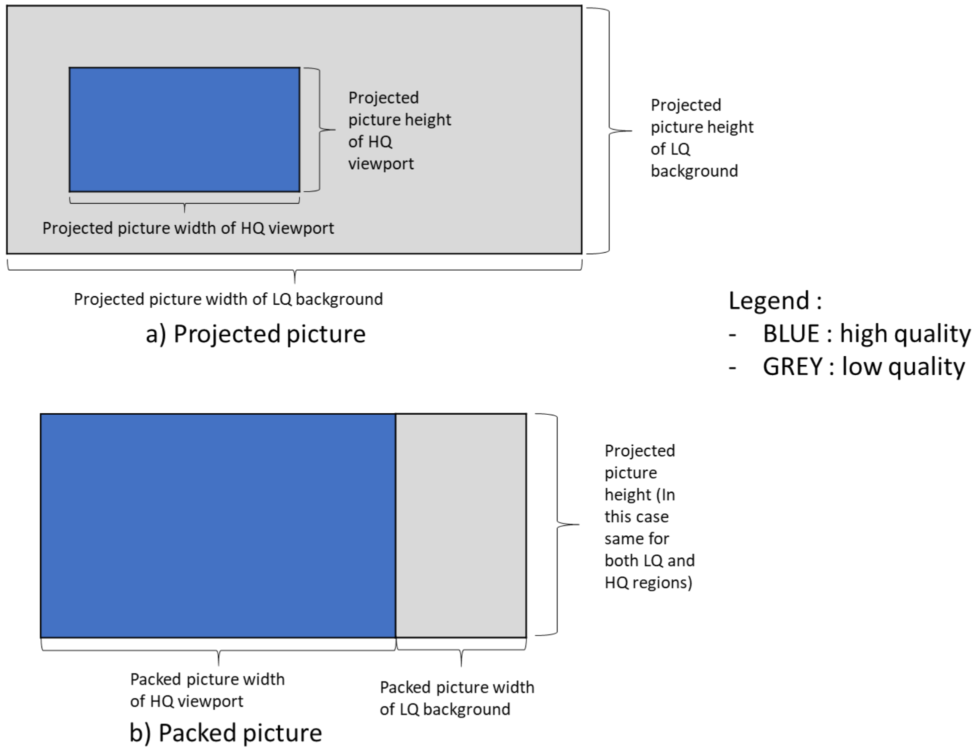

Projected picture height of HQ viewport
Projected picture width of HQ viewport
Projected picture height of LQ background
Projected picture width of LQ background
a) Projected picture
Legend :
- BLUE : high quality
- GREY : low quality
Projected picture height (In this case same for both LQ and HQ regions)
Packed picture width of HQ viewport
Packed picture width of LQ background
b) Packed picture

Figure 9: A high-quality viewport and low-quality background packed in a single stream.

Signalling and Delivery:

When the video is delivered as a single stream, the SDP indicates VDP with the appropriate PPM parameter. Additionally, rotation SEI is required for reverse rotation.

The resolution in the imageattr corresponds to the resolution of the encoded picture. The resolution of the viewport may be higher than the background.

Full Processing Pipeline

The full processing pipeline is shown in Figure 10. A high-quality viewport is encoded in the same way as described in 3 for HQ viewport-only delivery. In addition, a LQ 360-degree viewport-independent version of the video is encoded. Both streams are packed together, packetized and delivered to the ITT4RT-Rx client. The ITT4RT-Rx client unpacks, decodes and reverse rotates the HQ viewport to the right coordinates. The viewport region is rendered on top of the low-quality background.

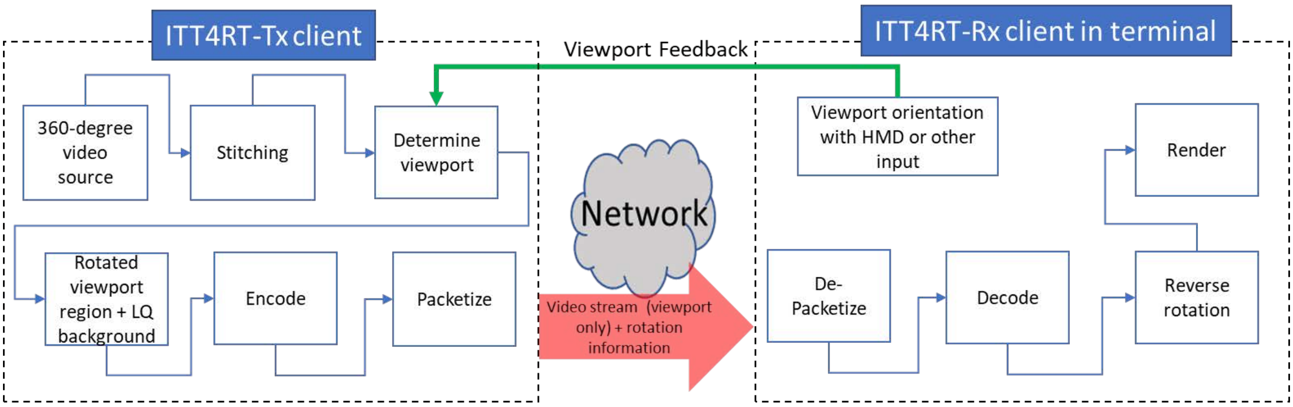

ITT4RT-Tx client
360-degree video source
Stitching
Determine viewport
Rotated viewport region + LQ background
Encode
Packetize
Viewport Feedback
Network
Video stream (viewport only) + rotation information
ITT4RT-Rx client in terminal
Viewport orientation with HMD or other input
Render
De-Packetize
Decode
Reverse rotation

Figure 10: A high-quality viewport and low-quality background using a single stream.

4.6 Comparisons of the Proposed Solutions

We present here a short discussion on the advantages and disadvantages of each of the proposed solutions. The purpose is to formulate a clear working assumption for ITT4RT applications on what type of VDP solution to use.

- 360-degree video with optimized viewport as described in 4.1 is delivered entirely as a single stream and does not require any bitstream signalling, packing/unpacking or rotation. It can be made scalable if multiple versions with different viewport orientations are encoded. However, the bandwidth savings when the full 360-degree video is delivered are low.
- Tiled video as described in 4.2 is scalable to a large number of receivers with different viewport sizes. The tiles are independently decodeable, and hence only the required tiles can be delivered (e.g., only the viewport with or without a viewport margin) but decoding the tiles independently requires either multiple decoders or serializing the decoding process (introducing latency). Decoding all tiles within the frame is more suited to a single decoder solution. This way the receiver also maintains media availability in case of head motion. However, this comes at the expense of bit rate savings. Finally, since tiled encoding at multiple qualities has higher storage and processing requirements it is, therefore, less suited to smaller conference sizes and may work better in the presence of an MRF/MCU.
- A HQ viewport region only solution described in 4.3 is able to maximize bandwidth savings and also does not require packing but limits the availability of media in case of head motion. Viewport margins as described in 4.6 may be used to extend the viewport for better experience. The frames maintain shape and resolution despite projection. The content is delivered as a single stream and a single encoder/decoder is required making it suitable for simpler ITT4RT-Tx and ITT4RT-Rx clients. However, additional bitstream signalling about the rotation and region-wise packing is required from the ITT4RT-Tx to ITT4RT-Rx client to reverse the rotation for sphere-locked mode. The special viewport-locked case in 4.3.2 omits the reverse rotation and provides a simplistic solution for ITT4RT clients with 2D screens and also ultra-low latency operation.

- A HQ viewport region can be paired with a background LQ 360-degree video for a fuller viewing experience as described in 4.4. When packed in a single stream, the LQ background + HQ viewport solution does not require updating the packing information with a changing viewport. The packing information (PPM) can be signaled once in the beginning using SDP. This method delivers redundant content in the viewport region.

4.7 Viewport Margins

Editor's Note: Signalling the extent of margins is FFS. The section below is informative about the use of viewport margins in viewport-dependent delivery.

When VDP is used, a change in viewport, e.g., due to head motion at the ITT4RT-Rx client in terminal, may require an update in the viewport region. This change is triggered by an RTCP feedback with the new viewport information. It may take up to at least one Round Trip Time (RTT) or more for the viewport to update, resulting in a motion-to-high-quality delay. Motion-to-high-quality delay is the amount of time it takes for the new viewport to reach comparable quality to the early viewport after head motion. An ITT4RT-Tx client that support VDP may use viewport margins to minimize this delay and also to reduce the need for frequent viewport updates.

Viewport margins can be extended on all or some sides of the viewport and may be at the same quality (or resolution) as the viewport or at a quality (or resolution) lower than the viewport but higher than the background. Figure 11 shows an ERP with the viewport, viewport margin and background regions in different colours.

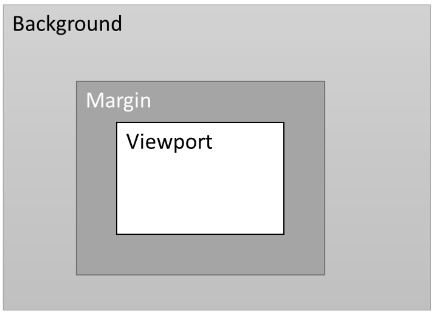
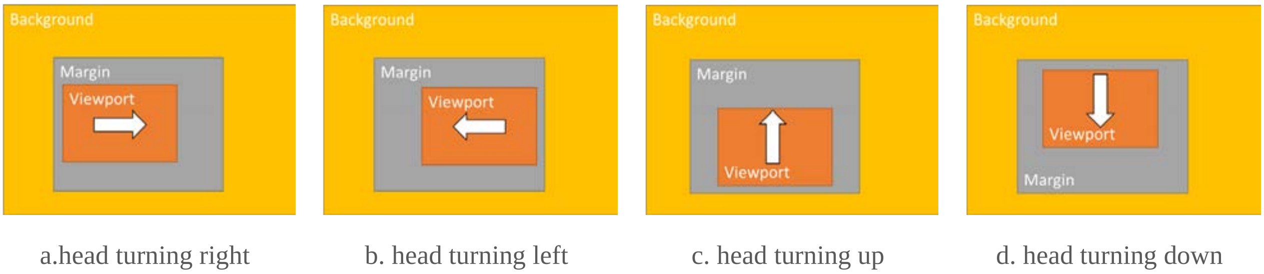


Figure 11: An equirectangular projected picture with viewport, margin and background areas.

Viewport margins may be extended around the viewport evenly or unevenly. Some example scenarios where viewport margins may be used to improve playout are listed below:

- Equally extended margins (symmetric) around the viewport in all directions may be used to decrease the motion to high-quality delay. The margins may be gradually extended farther by probing the network and reduced when the network is congested. In this scenario, the use of margins is akin to sending a larger viewport.
- Margins may be unevenly extended around the viewport (directional) with larger margins in the direction the user's head is turning. In the absence of head motion, the margins may return to being equally extended all around the viewport. In this case, RTCP viewport feedback is used to decide the distribution of margins. See Figure 12.
- Margins may be unevenly extended around the viewport with larger margins in the direction of the predicted head motion, e.g., based on audio input, motion tracking or other application level functions.



a.head turning right b. head turning left c. head turning up d. head turning down

Figure 12: Uneven extension of margins based on the user head motion. Similar uneven extensions may be used based on other application-level parameters.

Negative margins may also be utilized, which extend inward into the viewport region instead of extending outward into the background region. Negative margins are transmitted at a quality lower than the viewport. Introducing lower quality in the viewport can lower user experience. It may still be useful to do so in some cases, for example:

- Negative margins can be used when bandwidth is insufficient. This may lead to lower quality in parts of the viewport. However, this may be acceptable since the gaze is likely to be centered towards the centre of the viewport[X1].
- Negative margin can be used in the direction opposite to the direction of motion in a bandwidth neutral way, where any bandwidth savings from using them can be utilized for extending the positive viewport margin farther in the direction of motion. In this case the negative margins are likelyto be not visible in the updated viewport.

4.8 Encoding assumptions

Several encoding techniques may be used to optimize the real-time transmission and consumption of 360 video in conjunction with VDP.

Some of these techniques are briefly described in this section.

Adaptive RAP frequency

The inter-RAP distance may be varied according to the user's head movement. When the user's head motion increases beyond a certain threshold (H_{TH}), the inter-RAP distance is reduced. Therefore, an ITT4RT-Tx client may encode the RTP bitstream with a shorter duration between successive RAPs in real-time. When the viewport of the end-user is slowly changed, or the network bandwidth is not enough, the ITT4RT-Tx client may encode and transmit RTP bitstream with a longer distance between successive RAPs since this scheme allows the videos to be compressed optimally, thereby reducing the required bandwidth. The ITT4RT-Tx client can define a lower bound for the interval between two successive RAPs due to the bandwidth constraints. A maximum threshold (H_{MAX}) for the user's head motion may be defined. Therefore, even if the user's head motion increases beyond the corresponding maximum threshold (H_{MAX}), the ITT4RT-Tx client does not reduce the duration between two successive RAPs from that lower bound.

If the user's head motion is within the viewport margin, the new tiles or sub-pictures sent by the ITT4RT-Tx client (to update the margins) may have RTP bitstream encoded with a longer duration between successive RAPs. This is applicable when the resolution of the margin is the same as that of the viewport, i.e. high-resolution tiles or sub-pictures. If the resolution of the tiles or sub-pictures in the margin is lower compared to the tiles in the viewport, the ITT4RT-Tx client may send high-resolution RTP bitstream encoded with a shorter duration between successive RAPs as the user's head moves even within the margin to reduce the M2HQ delay.

Adaptive Code Mode

The encoder may be instructed to favor INTRA mode decisions for cUs of newly covered areas during the user's head motion. This may be performed by adjusting the thresholds and lambda values for INTRA mode selection. This will produce INTRA slices for the newly covered picture tiles. Refer to [X1] on HEVC encoder description section 6.3.

Adaptive QP

This tool varies the quantization parameter for each CU to adjust the quality. Adaptive QP may prove useful to adjust the quality of the slices and tiles that fall within the current viewport. Higher QPs may be used for slices and tiles that cover the margins. Refer to [X1] on HEVC encoder description section 6.5.4.

Note that the use of higher QP helps only if the receiver decodes all the tiles of the viewport and margin area (and if needed the tile beyond the margin) for the entire GOP since due to the motion of HMD, in order to reduce the M2HQ delay, the corresponding tiles is expected to be decoded starting from the last RAP.

4.9 Scalability

Tiled encoding can be used for providing VDP when there are a large number of receivers as already discussed earlier.

The HQ viewport region-only solution in clause 4.3 can require an encoder for each receiver if the encoded viewport is matched exactly to the receiver viewport. The solution can be made scalable by encoding multiple predefinedpre-defined overlapping or non-overlapping viewport regions. The appropriate HQ viewport region is then delivered based on the current viewport of the receiver.

Figure 13 illustrates a sample architecture for an MRF that provides VDP for a large number of receivers. It is assumed here that the 360-degree video is encoded into four regions (HQ viewport + margin). It is assumed that that all possible viewports are covered by at least one of the regions. The viewport selector then provides the appropriate region to the ITT4RT-Rx client based on viewport feedback. The MRF may save on resources by only encoding the regions that have active viewers.

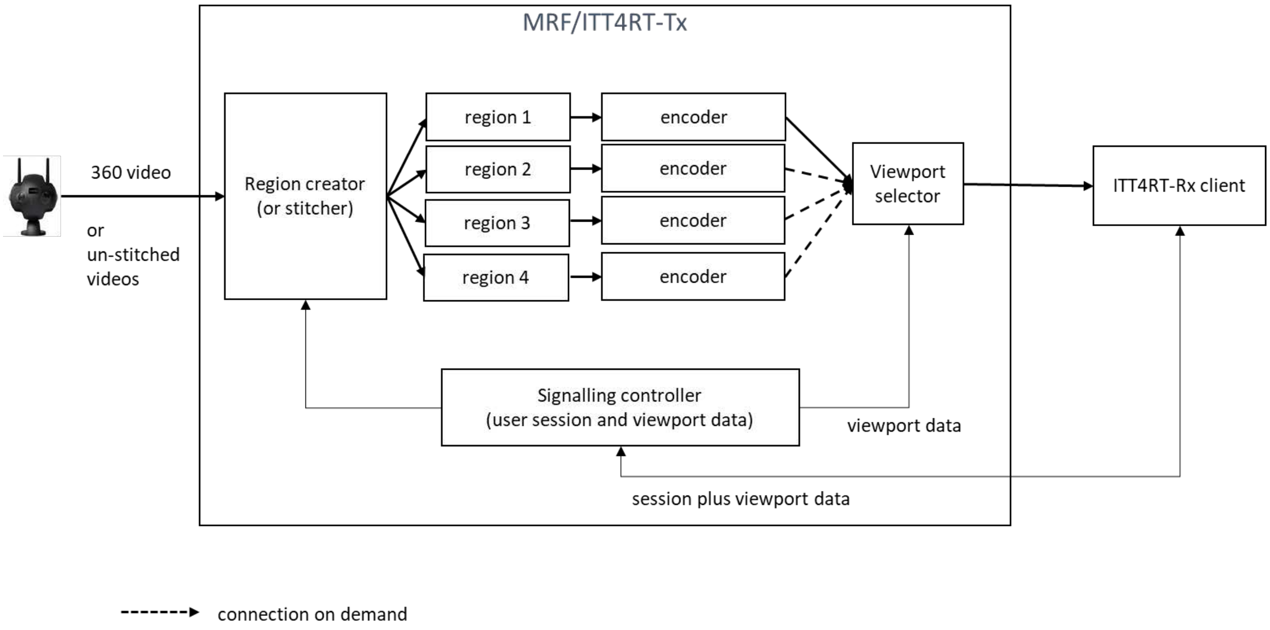


Figure 13: An MRF/ITT4RT-Tx client with four regions to cover all possible viewports.

One aspect to consider with creating such encoded region is the switching frequency from one region to another since a switch would require an I-frame update. Switching may result in visual artefacts in the absence of an I-frame. An I-frame may be inserted when a new ITT4RT-Rx client joins or when a region is switched. However, excessive I-frames can increase bandwidth requirements and impact encoding quality. Thus implementations are expected to weigh the benefits before inserting one. Advanced implementations may use ML/AI solutions to define regions dynamically based on user head movement to minimize the need for switching between regions. In Figure 4.13, the signalling controller provides a feedback to the region creator for this purpose.

5 RTCP Feedback for VDP

5.1 Introduction

This section provides guidance for the implementation of the RTCP feedback message type ‘Viewport’ described in clause Y.7.2 of TS 26.114.

Figure 14 illustrates the impact of the viewport feedback message on motion-to-high-quality delay. The figure shows a sender (ITT4RT-Tx client) delivering viewport-dependent content to a receiver (ITT4RT-Rx client). After a change in viewport, an RTCP viewport feedback is sent by the receiver to the sender, which responds with updating the viewport in the viewport-dependent content. The motion-to-high quality delay includes the RTT, processing delays and also any delay in generating the RTCP viewport feedback. Therefore, the viewport feedback mechanism requires careful consideration.

An ITT4RT-Rx client may include a predicted viewport (using application-level viewport prediction techniques) in the viewport feedback if the viewport control is set to *device_controlled* during SDP negotiation. An ITT4RT-Rx client may use fixed interval RTCP feedback or fixed interval with early feedback.

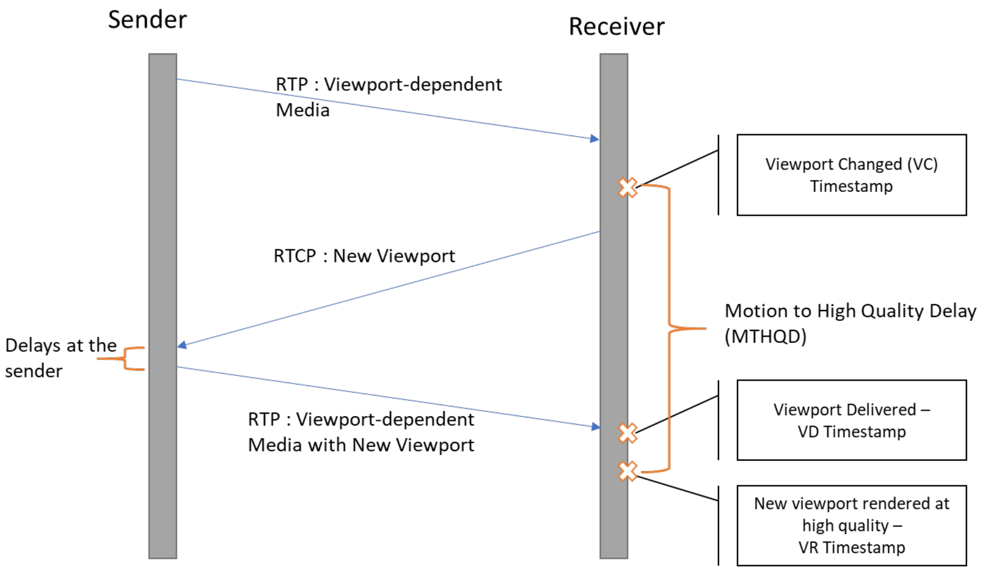

Sender
Receiver
RTP : Viewport-dependent Media
Viewport Changed (VC) Timestamp
RTCP : New Viewport
Motion to High Quality Delay (MTHQD)
Delays at the sender
RTP : Viewport-dependent Media with New Viewport
Viewport Delivered – VD Timestamp
New viewport rendered at high quality – VR Timestamp

Figure 14: Motion to High Quality Delay

5.2 Viewport Feedback Triggering Threshold

An ITT4RT-Tx may define a viewport feedback trigger value and signal this value to the ITT4RT-Rx client in the SDP. An ITT4RT-Tx client that uses viewport margins is expected to define the trigger based on viewport margin region. It is assumed that the ITT4RT-Tx client selects a threshold value that is suitable for the margin configuration that it intends to use for that stream. The threshold value is expected to be defined within the viewport margin region such that the ITT4RT-Tx client would update the high-quality region (viewport and viewport margin) if the viewport breaches this threshold.

If an ITT4RT-Rx client does not have the capability to provide an RTCP viewport feedback at the viewport feedback threshold value provided by the ITT4RT-Tx client (i.e., the provided threshold value is too low) in an SDP offer, it may respond with the the minimum threshold value it can support. The ITT4RT-Tx client may adjust its viewport margin configuration based on thethreshold value in the answer. If an ITT4RT-Rx client only supports periodic feedback, it is expected to remove the viewportfb_trigger parameter from the response.

It is assumed that an ITT4RT-Rx client that supports a viewport feedback trigger includes the parameter viewportfb_trigger with the minimum threshold value it can support in an SDP offer. The ITT4RT-Tx client may remove the parameter if it does not support this value or respond with an acceptable value that is equal or higher than the one in the ITT4RT-Rx's offer.

If both sides acknowledge the support of viewportfb_trigger, the ITT4RT-Rx client is expected to use event-driven viewport feedback. If viewportfb_trigger is not defined by the ITT4RT-Tx client, the ITT4RT-Rx client may still use event-driven feedback. All ITT4RT-Rx client sending RTCP viewport feedback is expected to at least support sending periodic viewport feedback.

5.3 Event-based Viewport Feedback

As per RFC 4585, the client can send event-based immediate feedback as long as:

Events per interval <= RTCP allocated bandwidth / Avg RTCP packet size

where,

Events per interval= Avg no of events reported/ Time interval.

For an RTCP viewport feedback, the event-based feedback interval can be calculated using the trigger angle by:

Events per interval = V / Trigger Angle

where V is the speed of the head mounted display and denotes the rate at which the viewport changes, and Trigger Angle is an angle representing the minimum fixed distance of the head movement that triggers sending a feedback. Therefore, to respect the 5% bandwidth allocated for the RTCP traffic allowed by the RTP standard, it is expected that the receiver does not trigger event-based immediate feedbacks for trigger angle smaller than TA_{min}, and

TA_{min} >= Average RTCP packet size / RTCP allocated bandwidth * V

With a changing trigger angle, receivers continue to operate within immediate feedback mode, where a feedback is sent as soon as an event is detected. However, the receiver redefines the event, which is the minimum trigger angle, based on current head speed. If the viewportfb_trigger is defined, then TA_{min} is expected to be greater than or equal to the defined trigger values.

5.4 Hybrid Fixed Interval and Event-driven Feedback

Alternatively, receivers may evaluate an average time interval, T, between RTCP feedback messages based on the allocated RTCP bandwidth. The value T can be used by the receiver to limit RTCP viewport feedbacks from exceeding the allocated bandwidth in case of high motion, as done for the early RTCP feedback mode in RFC 4585. Using the interval, it is possible to use a combination of regular and event-based feedback messages.

An ITT4RT-Rx client that uses this hybrid approach (a constant rate of RTCP feedback with event-driven feedback) may define a trigger that initiates an early RTCP feedback. An early RTCP feedback message is defined as a feedback message that is sent earlier than its scheduled transmission time. When viewport margins are used, the early viewport feedback trigger is assumed to take them into account. An ITT4RT-Rx client may use the viewportfb_trigger values to define the expected margin region. Alternatively, the extent of the viewport margin may be estimated from the bitstream. Figure 15 shows an image of an ERP of a 360-degree video with the current viewport and margins. As the viewport changes, the distance of the viewport, in degrees, from the boundary of the margins (shown as D_left, D_top, D_right, D_bottom and lastly D_direct, which is the distance in the direction of motion) decreases. An implementation may use the velocity of the head (causing the change in viewport), V, or the distance, D, to trigger the early viewport feedback i.e, an early viewport feedback is triggered when the V is above a certain threshold V', and D is below a certain threshold D'.

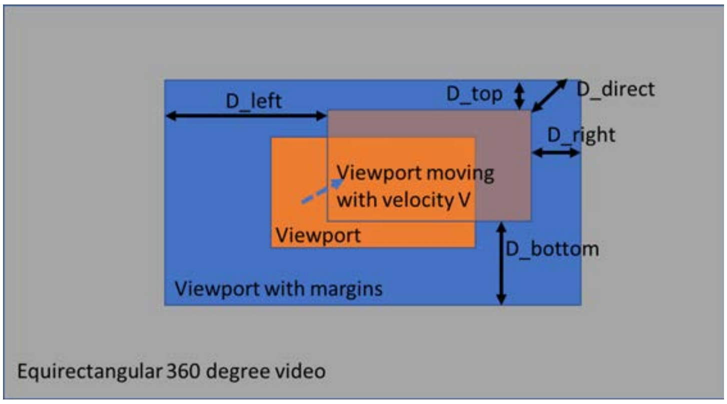


Figure 15. Moving viewport shown on an ERP of 360-degree video when viewport margins are used

Alternatively, the ITT4RT-Rx client may compute the time till the changing viewport will breach the boundary of the viewport margin, T_breach. An early viewport feedback is triggered when T_breach is below a certain threshold. Regardless of whether D, V or T_breach is used to trigger an early RTCP feedback, if the time to the next scheduled RTCP viewport feedback is small, the early feedback can be suppressed and the regular scheduled feedback is sent. As an example, Figure 16 shows a timeline with T, the interval between two scheduled viewport feedbacks and T_breach, the time at which the viewport is expected to breach the viewport margin i.e., the high-quality region of the 360-degree video. An ITT4RT-Rx client may trigger an early RTCP viewport feedback as long as T-T_breach < T', where T' is the suppression threshold for an early RTCP feedback.

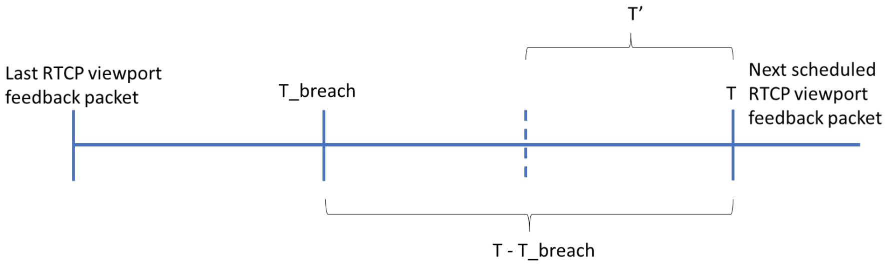

T'
Last RTCP viewport feedback packet
T_breach
T
Next scheduled RTCP viewport feedback packet
T - T_breach

Figure 16: A timeline showing T, T'and T_breach in reference to the scheduled RTCP viewport feedback. Time increases left to right.

When viewport margins are not used, an ITT4RT-Rx client may trigger an early RTCP viewport feedback if the current viewport has deviated more than a threshold from the last reported viewport. This solution is independent of the received 360-degree content and the current high-quality region. Figure 17 shows an illustration for the spherical distance D, in degrees, between the center of the viewport in the last RTCP viewport feedback and the center of the current viewport at the ITT4RT-Rx client. An ITT4RT-Rx client may trigger an early RTCP feedback message in this case when D > D', where D' is some threshold. However, if the time to next scheduled RTCP feedback is less than the suppression threshold T', the early RTCP feedback is not sent. The suppression threshold is used for controlling the rate of the viewport feedback. When an early RTCP feedback is sent, the ITT4RT-Rx client may send the next RTCP viewport feedback after T from the time of the event-driven viewport feedback. When the viewportfb_trigger is used, it is expected that D' is derived from it.

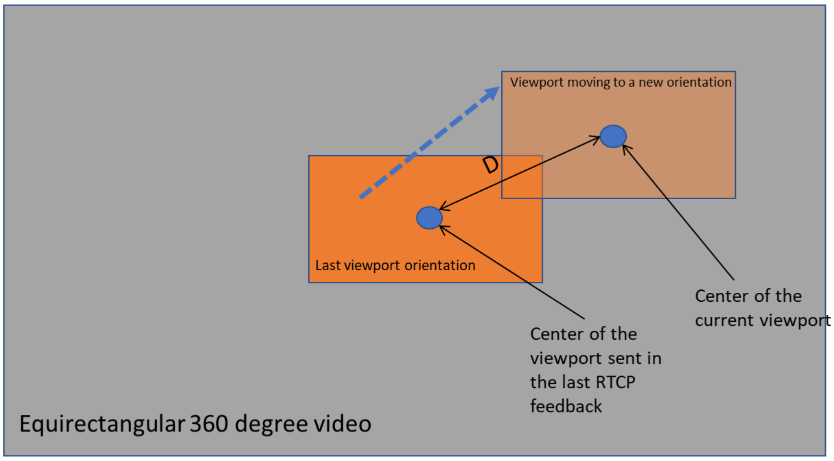

Viewport moving to a new orientation
D
Last viewport orientation
Center of the current viewport
Center of the viewport sent in the last RTCP feedback
Equirectangular 360 degree video

Figure 17: Changing viewport orientation (dotted blue arrow) shown on an ERP

NOTE: Whether there is a need for signalling any of the thresholds for triggering or suppressing an early RTCP feedback message, for triggering an event-based feedback or for defining periodic feedbacks is FFS.

6 Presentation replacement (screen share)

6.1 Introduction

One common situation in a meeting is to present additional material (e.g., slides, screen share video, notes, etc.) on a display (screen or projector). When capturing such a display with a 360-degree camera, this can lead to significant quality degradations, depending on the characteristics of the camera, display and lighting conditions. Simply most setups will not allow to capture both users and a display in high detail and ideal lighting; further, the display refresh rate and camera capture rate are often mis-aligned. To mitigate this problem the ITT4RT client allows to replace the captured content with the original presentation material. Further, the replacement might have benefits in terms of reduced bandwidth and processing load the to the receiving ITT4RT-Rx client (compared to transmitting the presentation content as overlay parallel to the 360-degree content). We can consider the replacement of image data in the 360-degree video as a special case of overlays that is expected either be handled in the sending client (ITT4RT-Tx) of the 360-video or in the network (MRF/MCU) in the following way:

1. Signal that content replacement capability is available
2. Obtain (presentation) material
3. Analyze and determine the position of content in the 360-recording
4. Replace content or signal overlay parameters.

Signal that content replacement capability is available

Currently, the 360-degree video is indicated with the attribute “a=3gpp_360video” in the SDP negotiation. To indicate that the presentation content replacement is available, the SDP negotiation includes an additional new attribute “a=3gpp_360video_ replacement”. This is signalled as part of the SDP negotiation between the 360-degree ITT4RT-Tx client and the MRF/MCU.

If the replacement is fully handled in the 360-degree sending client (that is, this client is responsible for both capturing the 360-degree content and displaying the presentation content), it will not signal the attribute “a=3gpp_360video_replacement”.

Note: The main importance of the “a=3gpp_360video_replacement” attribute is to distinguish who can perform the replacement in the event that both the 360-degree capture client and the MRF/MCU support replacement.

Obtain (presentation) material

The availability of the presentation content is expected to be signalled with the SDP parameter “a=content:slides”[29].

Note: this step can be skipped if the replacement is fully handled in the 360-degree sending client (i.e., this client is both responsible for capturing the 360-degree content and the display of the presentation content).

Analyze and determine the replacement configuration in the 360-recording

How the replacement configuration (i.e., configuration in terms of sphere-relative overlay coordinates) is determined is expected to be left as an implementation detail that does not need further specification. The output of this analysis includes the position of the content in the 360-degree video with the associated overlay characteristics to overlay/replace the image accordingly.

Note: Ideally, while receiving both 360-degree video and presentation content, the region is expected to be identified automatically (e.g., with image recognition tasks like pattern matching). However, a manual process may also be possible when handled directly by the sending UE.

Note: Assuming a static configuration of the 360-degree camera the content position only needs to be identified once for the lifetime of an ITT4RT communication session. Even if the presentation content changes, the positional parameters in the 360-degree video might be reused.

Replace content or signal overlay parameters

Replacement implies decoding, replacement of the captured presentation content at the (exact) display coordinates in the 360-degree video, and finally encoding the new 360-degree video (i.e., with the same encoding parameters as the original 360-degree video).

The solution is based on the definition of OMAF edition 1 that remote users "viewing position is the center of the unit sphere" [4] of the 360-degree image of the conference room. This means that all users view the 360-degree conference from the centre of the sphere, which is the capture position of the 360-degree camera.

Two options to replace content are possible, a) replace content directly in the 360-degree video (by injecting and re-encoding an adjusted version of the content given the identified overlay characteristics) and b) sending the video separately as overlay in the way specified in Chapter 6.3.

Replacing the content directly in the 360-degree video can be done either in the sending client of the 360-degree video or in the network (MRF/MCU).

6.2 Presentation replacement example message flow

Based on the four steps described in Section 6.1, we can end up in five likely situations between the ITT4RT-Tx and ITT4RT-MRF negotiations:

1. The 360-content from the ITT4RT-Tx is suitable for replacement and the ITT4RT-MRF does support replacement

2. The 360-content from the ITT4RT-Tx is suitable for replacement and the ITT4RT-MRF does NOT support replacement – Replacement is executed in the ITT4RT-Tx

3. The 360-content from the ITT4RT-Tx is suitable for replacement and the ITT4RT-MRF does NOT support replacement – no replacement is executed

4. The 360-content is not suitable for replacement but the ITT4RT-MRF does support replacement

5. The 360-content is not suitable for replacement (e.g., the presentation is not visible in the conference room) and the ITT4RT-MRF does not supported replacement

Note: In the following examples, we focus on the flows where the ITT4RT-Tx is initiating the SDP negotiation. However, the flow would remain very similar for the case in which the ITT4RT-MRF initiates the SDP negotiation, in terms of added SDP parameters.

6.2.1 Case 1–- The 360-content from the ITT4RT-Tx is suitable for replacement and the ITT4RT-MRF does support replacement

Both ITT4RT-Tx and ITT4RT-MRF signal "a=3gpp_360video_replacement" in the SDP negotiation and the replacement is executed in the ITT4RT-MRF.

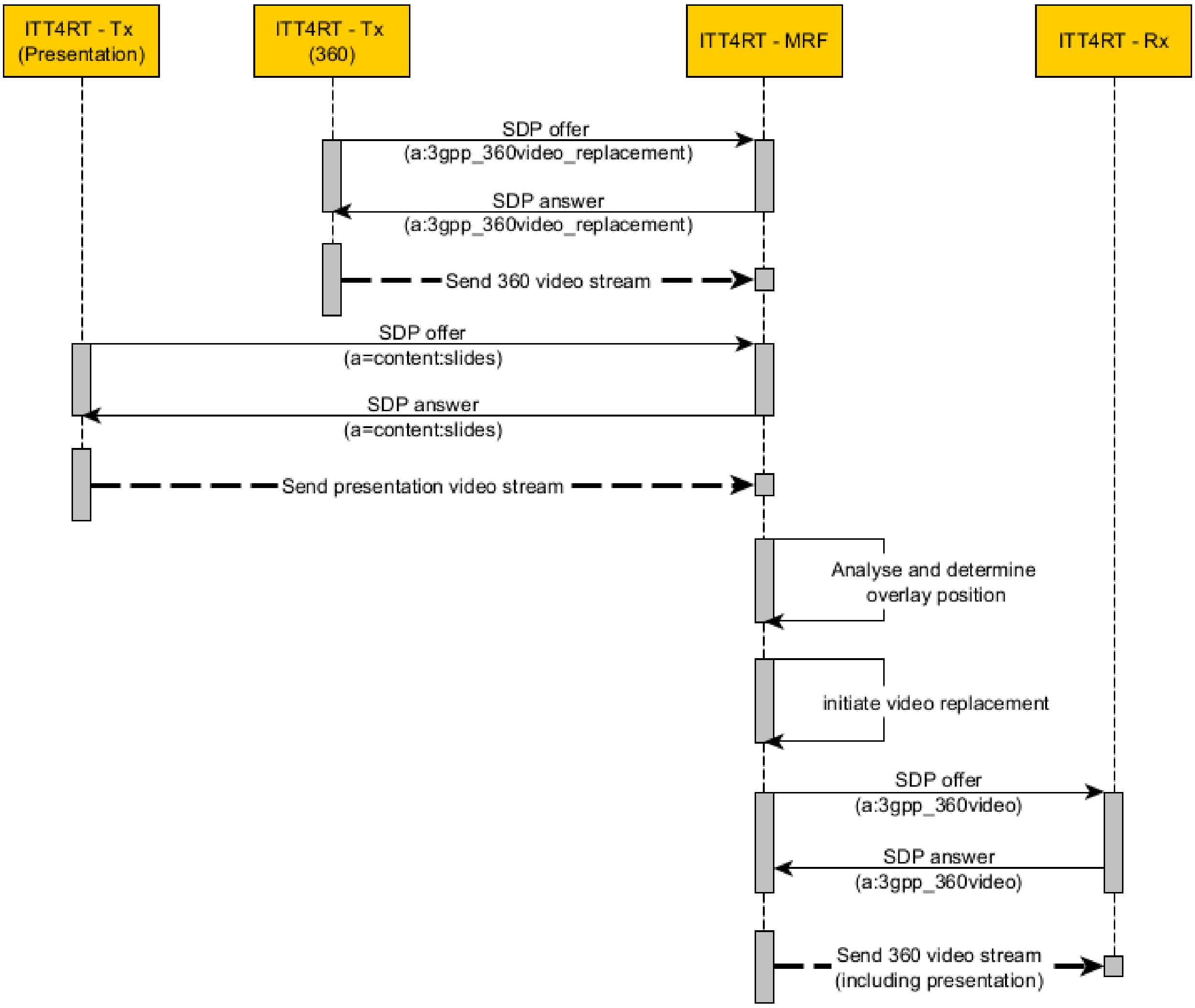

ITT4RT - Tx (Presentation)
ITT4RT - Tx (360)
ITT4RT - MRF
ITT4RT - Rx
SDP offer (a:3gpp_360video_replacement)
SDP answer (a:3gpp_360video_replacement)
Send 360 video stream
SDP offer (a=content:slides)
SDP answer (a=content:slides)
Send presentation video stream
Analyse and determine overlay position
initiate video replacement
SDP offer (a:3gpp_360video)
SDP answer (a:3gpp_360video)
Send 360 video stream (including presentation)

Figure 18: Replacement Flow – replacement in the ITT4RT-MRF

6.2.2 Case 2–- The 360-content from the ITT4RT-Tx is suitable for replacement and the ITT4RT-MRF does NOT support replacement – Replacement is executed in the ITT4RT-Tx

In this case, the ITT4RT-Tx includes the attribute “a=3gpp_360video_replacement” in the SDP negotiation, but any answer by the ITT4RT-MRF does NOT include the attribute “a=3gpp_360video_replacement”. The ITT4RT-Tx will execute the replacement.

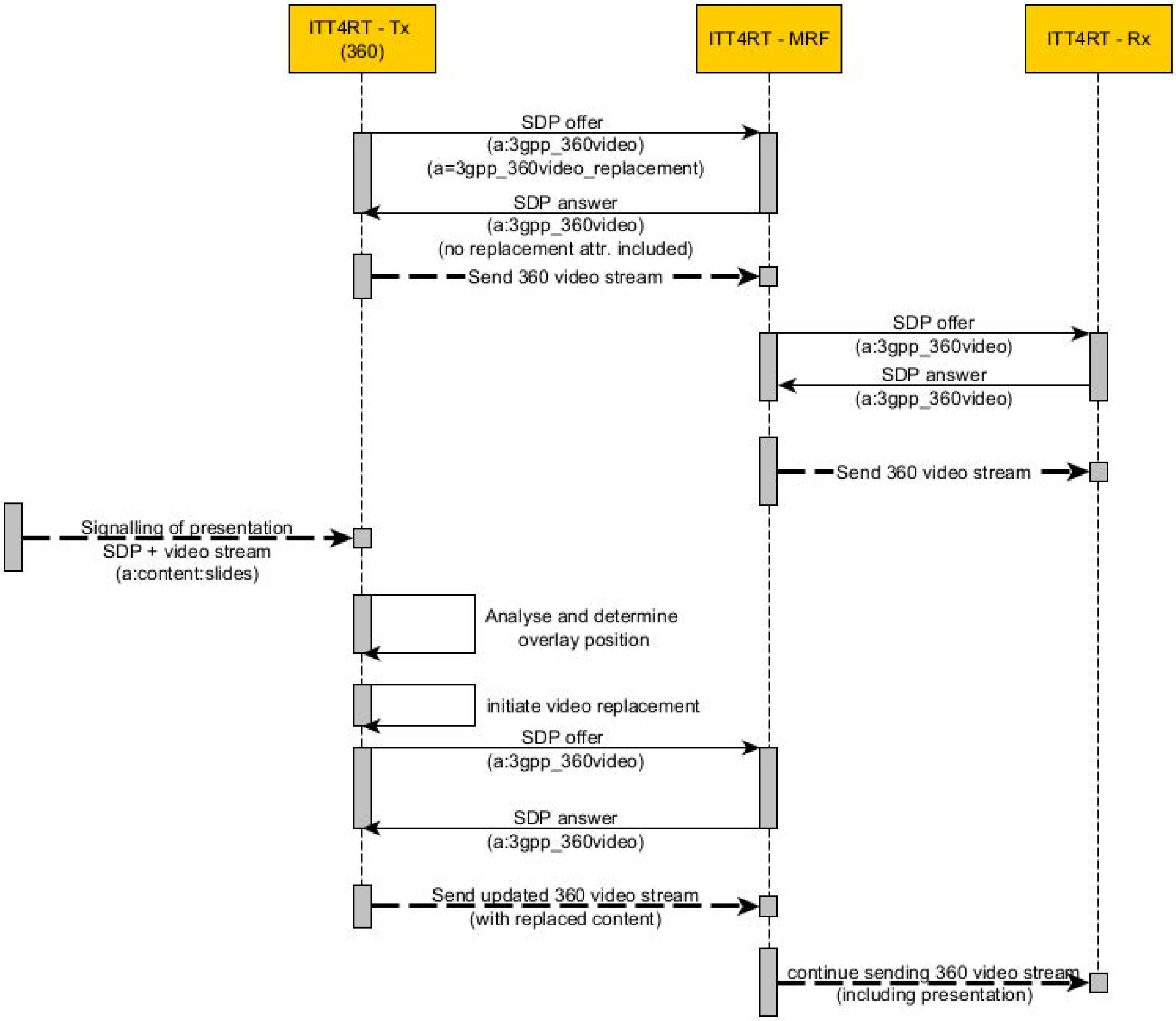

ITT4RT - Tx (360)
ITT4RT - MRF
ITT4RT - Rx
SDP offer (a:3gpp_360video) (a=3gpp_360video_replacement)
SDP answer (a:3gpp_360video) (no replacement attr. included)
Send 360 video stream
SDP offer (a:3gpp_360video)
SDP answer (a:3gpp_360video)
Send 360 video stream
Signalling of presentation SDP + video stream (a:content:slides)
Analyse and determine overlay position
initiate video replacement
SDP offer (a:3gpp_360video)
SDP answer (a:3gpp_360video)
Send updated 360 video stream (with replaced content)
continue sending 360 video stream (including presentation)

Figure 19: Replacement in ITT4RT-Tx Flow – the ITT4RT-MRF does not support replacement and ITT4RT-Tx is executing the replacement

Note: The details on how the ITT4RT-Tx (360) obtains the presentation material (a=content:slides) is not shown in this diagram and for further study. Important for the flow (if the ITT4RT-Tx is performing replacement) is that replacement can only be performed once the presentation material is signalled to the ITT4RT-Tx.

6.2.3 Case 3–- The 360-content from the ITT4RT-Tx is suitable for replacement and the ITT4RT-MRF does NOT support replacement – no replacement is executed

In this case, the ITT4RT-Tx includes the “a=3gpp_360video_replacement” in the SDP negotiation, but any answer by the ITT4RT-MRF does NOT include the “a=3gpp_360video_replacement” attribute. The presentation content is signalled by other means (send directly to the client).

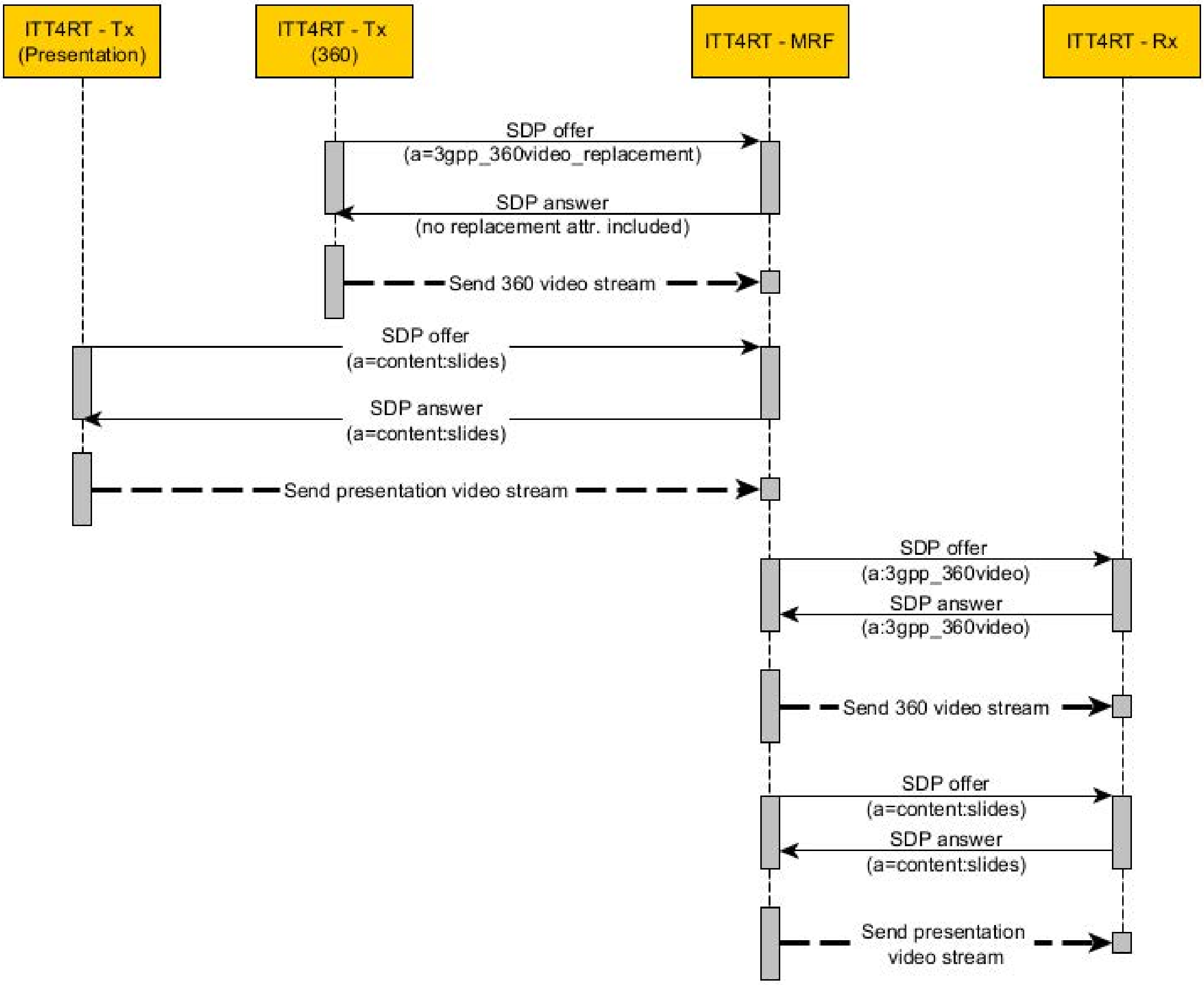

ITT4RT - Tx (Presentation)
ITT4RT - Tx (360)
ITT4RT - MRF
ITT4RT - Rx
SDP offer (a=3gpp_360video_replacement)
SDP answer (no replacement attr. included)
Send 360 video stream
SDP offer (a=content:slides)
SDP answer (a=content:slides)
Send presentation video stream
SDP offer (a:3gpp_360video)
SDP answer (a:3gpp_360video)
Send 360 video stream
SDP offer (a=content:slides)
SDP answer (a=content:slides)
Send presentation video stream

Figure 20: No Replacement Flow – the ITT4RT-MRF does not support replacement and additional content is signalled directly to the client as individual stream

Note: Figure 20 depicts one potential outcome of Case 3, but other might be possible based on the application logic. For example, signaling the presentation content as a sphere-relative overlay.

Note: The presentation content may also be signaled between the MRF and the Client with the 3gpp overlay properties (to instruct the client with suitable overlay characteristics).

Note: If the MRF initiates the SDP negotiation, the call flow would be very similar, but the MRF would send an offer that does not include the replacement attribute (while the ITT4RT-Tx client would answer without the replacement attribute as well).

6.2.4 Case 4–- The 360-content is not suitable for replacement but the ITT4RT-MRF does support replacement.

In this case, the ITT4RT-Tx does not include the “a=3gpp_360video_replacement” in the SDP negotiation. If the SDP negotiation is initiated by the ITT4RT-MRF, the ITT4RT-MRF includes the “a=3gpp_360video_replacement” attribute, followed by an SDP response of the ITT4RT-Tx that does NOT include the attribute “a=3gpp_360video_replacement”. Replacement will not be executed, and any additional presentation material can be signalled by other means (e.g., as sphere relative overlay).

6.2.5 Case 5–- The 360-content is not suitable for replacement (e.g., the presentation is not visible in the conference room) and the ITT4RT-MRF does not support replacement

In this case, no specific replacement related signalling will occur. Neither the ITT4RT-Tx nor the ITT4RT-MRF includes “a=3gpp_360video_replacement” in the SDP negotiation and no replacement will be made. Any additional presentation material can be signalled by other means (e.g., as a sphere relative overlay).

7 Scene Description-based Overlay

7.1 Scene Description

7.1.1 Overview

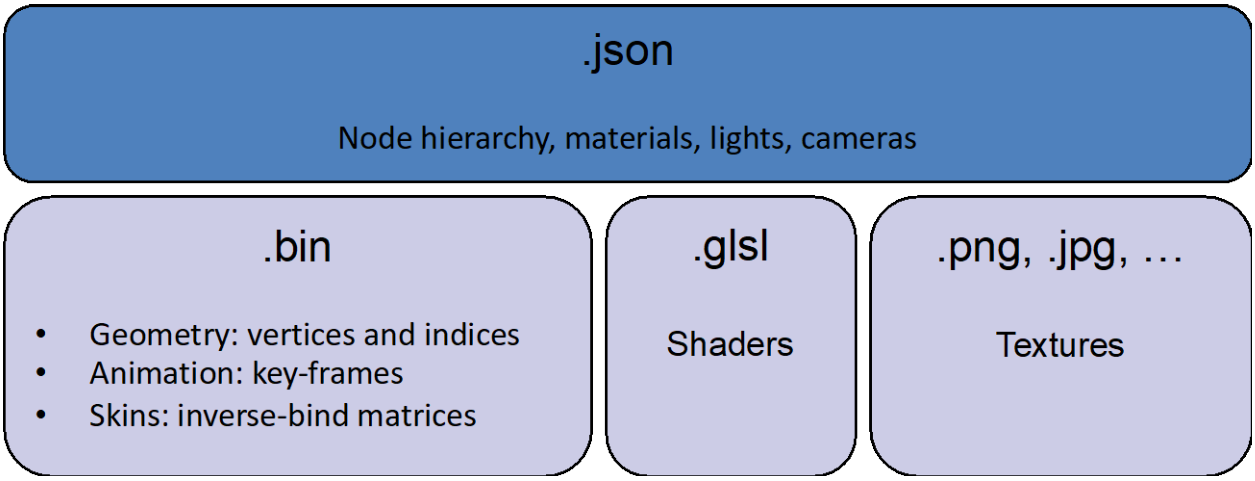
A scene graph is a directed acyclic graph, usually just a plain tree-structure, that represents an object-based hierarchy of the geometry of a scene. The leaf nodes of the graph represent geometric primitives such as polygons. Each node in the graph holds pointers to its children. The child nodes can among others be a group of other nodes, a geometry element, a transformation matrix, etc.

Spatial transformations are attached to nodes of the graph and represented by a transformation matrix.

7.1.2 glTF 2.0

glTF 2.0 is a new standard that was developed by Khronos to enable Physically Based Rendering. glTF 2.0 offers a compact and low-level representation of a scene graph. glTF 2.0 offers a flat hierarchy of the scene graph representation to simplify the processing. glTF 2.0 scene graphs are represented in JSON to ease the integration in web environments. The glTF 2.0 specification is designed to elimate redundancy in the representation and to offer efficient indexing of the different objects in the scene graph.

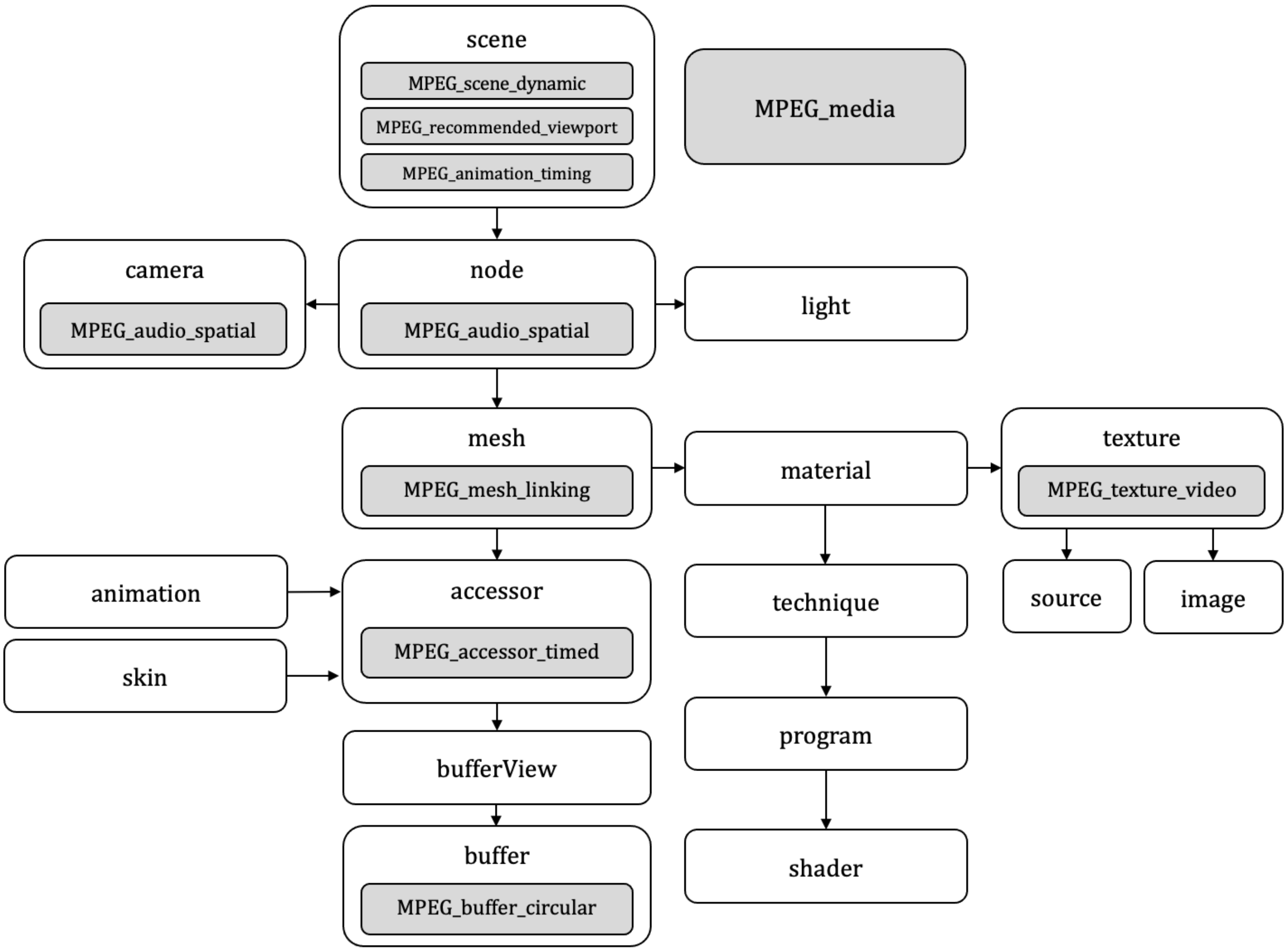
The structure of a glTF 2.0 scene graph document is arranged as follows:


.json
Node hierarchy, materials, lights, cameras
.bin
• Geometry: vertices and indices
• Animation: key-frames
• Skins: inverse-bind matrices
.glsl
Shaders
.png, .jpg, …
Textures

7.1.3 MPEG-I Scene Description

The MPEG solution specifies a key set of extensions to glTF 2.0 to support timed media such as dynamic objects, video textures, and audio. The MPEG_media, MPEG_accessor_timed, and MPEG_buffer_circular extensions make up the core of these extensions and enable integrating all timed media. The MPEG_texture_video allows the integration of video textures that may for example be the sources of an overlay in an ITT4RT conference.

The node structure of an MPEG-I scene description is depicted by the following diagram:



7.2 Scene Description for ITT4RT Sessions

The composition may be performed at an ITT4RT MRF that supports scene description-based overlays.

The scene description describes the whole scene, including all audible/visible participants and the main conference room. Each scene node is expected to contain at most one node with 360 degree content. Switching between different 360 conference rooms is then equivalent to switching between scenes in the scene description.

A node in the scene graph may describe the 360 degree content or overlay content. The node provides the geometry and the associated texture, which references a 360 or an overlay image/video source.

The 360 degree content is described through a sphere or cube-map geometry with associated video texture coming from the 360 video. The type of geometry depends on the selected projection for the ITT4RT session. The overlay nodes are typically rectangular plane regions with an associated video texture coming from the associated overlay video stream.

Participants are required to indicate if they support scene description by accepting an SDP offer that contains the data channel, which indicates “mpeg-sd” as the sub-protocol. A participant that does not support scene description will receive an overlay description in the SDP and may declare its overlay streams using the SDP 3gpp-overlay attribute.

In a scene, node names are expected to be unique to ensure there are no naming conflicts in nodes provided by different parties in a call. Nodes in the scene description may reference external media streams, such as other media streams that are declared in the SDP. A participant may mask nodes from certain parties in the rendering process, e.g. based on user input.

7.3 Referencing Media Streams

In order to reference the source video streams for the 360 content and the overlays in a scene description document, the URL format as specified in ISO/IEC DIS 23090-14 Annex C [8] is expected to be used.

The specified referencing scheme has the following advantages:

- doesn't require that the IP address, port number, and protocol scheme are known. All these fields can be substituted by a generic matching pattern.
- the stream identifier is flexible and allows usage of one of the "mid" attribute, the "label" attribute, or media stream position index as a stream identifier.
- the URL complies with RFC 3986 [7].

In an ITT4RT session, the session setup and SDP exchanges is done using the SIP protocol. Usually, the data channel for exchanging the scene description and scene description updates will also be described as part of the same SDP in that negotiation. In the absence of the source IP address and port number identification of the referenced media stream, the stream identifier would be sufficient and will refer to that same SDP.

Alternatively, a labelling scheme may be used to enforce an ITT4RT session-wide unique identifier of a media stream in the "label" attribute. The label may for example be prefixed by a unique participant name, e.g. "participant5_overlay1". Such a 28abelling scheme is maintained by the scene description author at the MRF and is out of scope of ITT4RT.

Annex <A> (informative): Change history

Change history

Date	Meeting	tDoc	CR	Rev	Cat	Subject/Comment	New version
2021-08	SA4#115	S4-211257				Addition of technical contributions	0.1.0
2021-11	SA4#116	S4-211617				Addition of technical contributions	0.2.0
2022-02	SÁ4#117	S4-220163				Addition of technical contribution S4aM220683. Terms definitions. Editorial clean-up.	0.3.0
2022-02	SA4#117	S4-220264				Addition of section on scene description-based overlay (S4-220256)	0.4.0
2022-02	SA4#117	S4-220269				Final version	0.5.0
2022-02	SA4#117	S4-220322				Language and editorial fixes	0.6.0
2022-03	SA#95-e	SP-220250				For approval in SA	2.0.0
2022-03	SA#95-e					Under change control	17.0.0

History

Document history		
V17.0.0	April 2022	Publication