**3GPP TSG-SA4 Meeting #114-eS4-210782**

**19th-28th May 2021**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| *CR-Form-v12.0* | | | | | | | | |
| **PSEUDO CHANGE REQUEST** | | | | | | | | |
|  | | | | | | | | |
|  | **26**.**998** | **CR** |  | **rev** |  | **Current version:** |  |  |
|  | | | | | | | | |
| *For* [***HE******LP***](http://www.3gpp.org/3G_Specs/CRs.htm#_blank)*on using this form: comprehensive instructions can be found at* [*http://www.3gpp.org/Change-Requests*](http://www.3gpp.org/Change-Requests)*.* | | | | | | | | |
|  | | | | | | | | |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ***Proposed change affects:*** | UICC apps |  | ME |  | Radio Access Network |  | Core Network | **X** |

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | | | | | | | | | |
| ***Title:*** | **[5GSTAR] Content formats and codecs** | | | | | | | | | |
|  |  | | | | | | | | | |
| ***Source to WG:*** | Samsung Electronics Co., Ltd. | | | | | | | | | |
| ***Source to TSG:*** | SA4 | | | | | | | | | |
|  |  | | | | | | | | | |
| ***Work item code:*** | FS\_5GSTAR | | | | |  | ***Date:*** | | | 2021-05-12 |
|  |  | | | |  | |  | | |  |
| ***Category:*** | **B** |  | | | | | ***Release:*** | | |  |
|  | *Use one of the following categories:* ***F*** *(correction)* ***A*** *(mirror corresponding to a change in an earlier release)* ***B*** *(addition of feature),* ***C*** *(functional modification of feature)* ***D*** *(editorial modification)*  Detailed explanations of the above categories can be found in 3GPP [TR 21.900](http://www.3gpp.org/ftp/Specs/html-info/21900.htm). | | | | | | | | *Use one of the following releases: Rel-8 (Release 8) Rel-9 (Release 9) Rel-10 (Release 10) Rel-11 (Release 11) Rel-12 (Release 12)* *Rel-13 (Release 13) Rel-14 (Release 14) Rel-15 (Release 15) Rel-16 (Release 16)* | |
|  |  | | | | | | | | | |
| ***Reason for change:*** | | Move content formats section from Permanent Document, and add relavant technologies such as encapsulation and manifest to improve TR 26.998. | | | | | | | | |
|  | |  | | | | | | | | |
| ***Summary of change:*** | | Add study on content formats for AR/MR service.  Add encapsulation format and manifest in accordance with the use of media formats. | | | | | | | | |
|  | |  | | | | | | | | |
| ***Consequences if not approved:*** | | Incomplete study of AR/MR services. | | | | | | | | |
|  | |  | | | | | | | | |
| ***Clauses affected:*** | | 6 | | | | | | | | |
|  | |  | | | | | | | | |
|  | | **Y** | **N** |  | | | |  | | |
| ***Other specs*** | |  |  | Other core specifications | | | | TS/TR ... CR ... | | |
| ***affected:*** | |  |  | Test specifications | | | | TS/TR ... CR ... | | |
| ***(show related CRs)*** | |  |  | O&M Specifications | | | | TS/TR ... CR ... | | |
|  | |  | | | | | | | | |
| ***Other comments:*** | |  | | | | | | | | |
|  | |  | | | | | | | | |
| ***This CR's revision history:*** | |  | | | | | | | | |

## \*\*\* Start change 1 \*\*\*

that is supported by media content that is partially or fully accessed through 5GS

a scene with typically and is service

component of an AR scene

## 1

## \*\*\* Start change 2 \*\*\*

## 4.X AR Content formats and codecs

### 4.X.1 Overview

5G AR/MR application provider provides a 5G AR/MR service to 5G AR/MR application. A 5G AR/MR service consists of AR/MR content and description of supported processings by the 5G immersive service architecture. An AR/MR content is agnostic to a service architecture and consists of one or more AR/MR objects, which of each usually corresponds to an immersive media in clause 4.X.4. Delivery of an immersive media adaptive to device capability and network bandwidth can be described by a delivery manifest in clause 4.X.5. An AR/MR content consists of one or more AR objects, and may include their spatial and temporal compositions. A spatial and temporal composition of AR/MR objects can be described by a scene description in clause 4.X.2. Processing of AR/MR functions in 5GMS AS may require additional metadata in clause 4.X.3 to properly recognize user’s pose and surroundings.

Editor’s note) A further study is required on cascading of AR service entities.

* For example, whether an AR object can have another AR object as a component, whether an AR content can have another AR content as a component, and whether an AR service can have another AR service as a component.

AR/MR functions include encoding, decoding, rendering and compositing of AR/MR object, after which localization and correction is performed based on the user’s pose information.

STAR-based architecture has both basic AR functions and AR/MR functions on the device. EDGAR-based architecture has only basic AR functions on the device.

Since AR/MR functions are on-device for the STAR-based architecture, immersive media including 2D media can be considered as the input media for the architecture.

Examples of immersive media are 2D/3D media such as overlay graphics and drawing of instructions (UC#16 in A.2), 3D media such as furniture, a house and an animated representation of 3D modeled person (UC#17 in A.3), a photorealistic volumetric video of a person (UC#18 in A.4), a 3D volumetric representation of conference participants (UC#19 in A.), 2D video, and volumetric information and simple textual overlays (UC#20 in A.6).

For the EDGAR-based architecture, basic AR functions are on-device therefore 2D media and additional information (such as depth map) generated from immersive media renderer can be considered as the input media for basic AR functions. A rasterized and physically-based rendering (PBR) image is an example of 2D media.

A study into the existing technologies to be considered as inputs to each function and device type are identified and presented as a non-exclusive list below.

* several visual media representation formats were documented in clause 4.X.4
* several delivery manifests were documented in clause 4.X.5
* several scene description formats were documented in clause 4.X.2
* metadata such as user pose information and camera information were documented in clause 4.X.3, respectively



Figure 4.X.1-1 Media Functions and formats

In order to integrate real-time media into AR scenes, a Media Access Function (MAF) provides the ability to access media and adds it to the AR scene. The MAF instantiates and manages Media Pipelines. A media pipeline typically handles content of an attribute/component of an object/mesh that is part of the scene graph. The media pipeline produces content in the format indicated by the scene description file. For real-time media, the formatted frame is then pushed into the circular buffer. Media Pipelines are typically highly optimized and customized for the type and format of media that is being fetched. Media Pipelines also maintain sync information (time and space) and passes that information as buffer metadata to the scene manager.

### 4.X.2 Scene Graph and Scene Description

A scene description may correspond to an AR/MR content. A volumetric media containing the primitives ranging from one vertex to a complex object can be described by a scene description. For the use cases of annex A, scene description is useful to locate AR/MR objects in user’s world. A scene description typically has a tree or a graph structure which of each leaf represents a component of a scene. A primitive or a group of primitives are referenced as a leaf node of the scene tree. A skeleton to allow for motion rigging or an animation of motion of the skeleton in time can present an animation of volumetric presentation.

* Formats for scene description

Khronos glTF2.0 and MPEG Scene description (ISO/IEC 23090-14) are examples of scene description technologies. They have a tree structure and internal/external resource references. There are many types of leaf of the tree. For example, a Node is one type of leaf under a Scene. A node can have a Camera as a subsidiary leaf. The node with camera represents one of the rendering frustum/viewport to be used by a scene renderer (i.e., immersive media renderer). Any translation/rotation/scaling of the node affects position and direction of its subsidiary, in this example, a camera. A node with mesh can be used as an anchor that represents AR object with its location and direction in geometric space.

MPEG Scene description is an extension of glTF2.0. It is extended to support MPEG immersive media. MPEG\_media and MPEG\_scene\_description are the major changes to provide support of media access link including manifest, and temporal update of the scene description itself.

Editor’s note) A further study is required of whether scene description technology is sufficient in defining a 5G AR/MR service and/or AR/MR content.

### 4.X.3 Metadata

#### 4.X.3.1 User pose information

User’s position can be represented as a geolocation with longitude and latitude. The position can also be represented as a point in a scene. The scene can be represented as a bounding box on a geometry which represents user’s real environment. When an AR/MR device reports the user position to obtain a split render of the immersive media from a server, the device calculating the user pose should be either a geolocation, a point in a scene or a point in a user’s geometry. Depending on the representation, the server should be aware of the underlying scene or the geometry. A device should update whenever there is any change in the scene or the geometry through user interaction (e.g., rotating a scene by hand gesture) and/or SLAM (e.g., finer modelling of surrounding environment).

A direction can be represented with a rotation matrix, or roll, pitch, and yaw. The direction is relative to a scene/geometry and the scene/geometry has an origin and default direction of the three axes.

The devices representing a user’s pose moves continuously, and if the device is worn on the user’s head, it can be assumed that he or she frequently turns their head around. A set of position and direction information is only meaningful at a certain moment in time. Since the device can report the user pose at around a frequency of 1KHz, any pose information should include a timestamp to specify when it was measured or created. A pose corrector (e.g., ATW and LSR) in a server may estimate the user’s future pose, whilst a pose corrector in a device may correct the received rendered image to fit the latest user pose.

* Formats for user pose

A position in Cartesian coordinate system can be represented by either X, Y and Z or by a translation matrix. A direction can be represented by a rotation matrix or by quaternions.

[OpenXR](https://www.khronos.org/registry/OpenXR/specs/1.0/html/xrspec.html) describes a possible format for user pose. It consists of 4 quaternions for orientation and 3 vectors for position. Timestamp is represented by a 64 bit monotonically increasing nano-second-based integer.

#### 4.X.3.2 Camera information

Immersive media can be captured by camera(s). The camera parameters such as focal length, principal points, calibration parameters and the pose of the camera all contribute in understanding the relevance between points in the volumetric scene and pixels in the captured image. Photogrammetry is the technology used to construct immersive media from a continuous capturing of images. Depth sensor-based cameras can be used to capture immersive media from one capturing of the volumetric scene

* Formats for camera information

Camera intrinsic parameters can be represented by a camera matrix. Extrinsic parameters can be represented by a transform matrix.

### 4.X.4 Media Formats/Primitives in AR Scenes

Immersive media can be considered as an AR/MR object and can be used to provide an immersive experience to users. The immersive experience may include a volumetric presentation of such media. The volumetric presentation does not bind to a specific display technology. For example, a mobile phone can be used to present either the whole AR media, or a part of the AR media. Users can see a volumetric presentation of a part of the AR media augmented in real space. Therefore, immersive media includes not only volumetric media formats such as omnidirectional visual formatsERP image, 3D meshesPrimitives, point cloudsPrimitives, light fieldsPlenopotic image, scene description, and 3D audio formats, but also 2D video2D image as studied in TR 26.928.

* Primitives

3D meshes and point clouds consists of thousands and millions of primitives such as vertex, edge, face, attribute and texture. Primitives are the very basic elements in all volumetric presentation. A vertex is a point in volumetric space, and contains position information in terms of three axes in coordinate system. In a Cartesian coordinate system, X, Y, and Z make the position information for a vertex. A vertex may have one or more attributes. Color and reflectance are typical examples of attributes. An edge is a line between two vertices. A face is a triangle or a rectangle formed by three or four vertices. The area of a face is filled by interpolated color of vertex attributes or from textures.

### 4.X.5 Compression Formats

##### 4.X.5.1 Elementary stream

An elementary stream is an output of a media encoder. Immersive media and 2D media in clause 4.X.4 have relevant technologies to encode each media format as follows.

* 2D Video codecs

There are differences in terms of context of 2D media, such as RGB image versus depth map image, one planar perspective camera image versus ERP, or one camera image versus HDCA plenoptic image. Such differences can be considered in the proper encoder/decoder coding tools. In general, 2D video codecs can encode 2D media types listed in clause 4.X.4. AVC and HEVC are industry wide examples of 2D video codecs.

* MPEG OMAF

OMAF consists of two parts; the first part is a pre-processing which includes a packing and projection of spherical volumetric media onto a 2D image, and the second part is an encapsulation of the compressed 2D frame packed image with metadata signalling the projection.

For the compression of the 2D images, 2D video codecs can be considered and the pre-processing operations are agnostic to specific 2D codec technology.

* MPEG V3C and V-PCC

V3C and V-PCC consists of two parts; the first part is a pre-processing which includes the decomposition of a part of the volumetric media into the planar projection, a patch, of different characteristics such as texture, geometry and occupancy. The second part is an encoding of 2D patch packing images, with metadata which signalling the decomposition.

For the encoding of the 2D images, 2D video codecs can be considered and the pre-processing operations are agnostic to specific 2D codec technology.

* MPEG G-PCC

G-PCC divides volumetric media into multiple sub-blocks. Triangle (Trisoup) or leaf (Octree) are used as the units of the divisions. A volumetric media is subdivided recursively until no more sub-blocks are left. The dimension (or level) of the tree is relatively large, such as 2^24. Tools including arithmetic encoding are used to encode all the tree information into the bitstream.

##### 4.X.5.2 Storage and Delivery Formats

An encapsulation format encapsulates an elementary stream with its coding structure information and metadata information. ISOBMFF (ISO based Media File Format, ISO/IEC 14496-12) is one of encapsulation format technology. DASH initialization/media segment and CMAF track are the extensions of ISOBMFF for both adaptive streaming and storage purpose. They are extended to provide partial access of a media fragment on time axis.

A delivery manifest provides a description of media service consisting of multiple media components such as video and audio. Adaptation to device capability or network bandwidth is key features of a delivery manifest. In a delivery manifest, there is a group of multiple different encodings of the same media component context with the description of the encoding variations. An encapsulation format for an adaptive streaming is used to allow temporal access of media fragment to enable adaptive switching of a group of different encodings. MPD (Media Presentation Description) for DASH is one of delivery manifest for the purpose.

* File formats for Primitives

OBJ, PLY, and GPU command buffer in OpenGL-based languages (e.g., glTF Buffer) are methods of encapsulating the primitives. A sequence of primitive files – such as multiple OBJs, PLYs or a set of GPU command buffers in a time can present an animation of volumetric presentation.

Editor’s note) A further study is required of how to describe adaptation of delivery and presentation of immersive media.

## \*\*\* End change 2 \*\*\*

## 

## \*\*\* End of changes \*\*\*