**3GPP TSG-SA4 Meeting #114-e *S4-210856***

**19th – 28th May 2021**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| *CR-Form-v12.1* | | | | | | | | |
| **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 |  |

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | | | | | | | | | |
| ***Title:*** | [FS\_5GSTAR] Background on MPEG Scene Description | | | | | | | | | |
|  |  | | | | | | | | | |
| ***Source to WG:*** | Qualcomm Incorporated | | | | | | | | | |
| ***Source to TSG:*** | S4 | | | | | | | | | |
|  |  | | | | | | | | | |
| ***Work item code:*** | FS\_5GSTAR | | | | |  | ***Date:*** | | |  |
|  |  | | | |  | |  | | |  |
| ***Category:*** |  |  | | | | | ***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-15 (Release 15) Rel-16 (Release 16) Rel-17 (Release 17) Rel-18 (Release 18)* | |
|  |  | | | | | | | | | |
| ***Reason for change:*** | |  | | | | | | | | |
|  | |  | | | | | | | | |
| ***Summary of change:*** | |  | | | | | | | | |
|  | |  | | | | | | | | |
| ***Consequences if not approved:*** | |  | | | | | | | | |
|  | |  | | | | | | | | |
| ***Clauses affected:*** | |  | | | | | | | | |
|  | |  | | | | | | | | |
|  | | **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:*** | |  | | | | | | | | |

**===== CHANGE =====**

[4.3.j] ISO/IEC 23090-14:2021 DIS: “Information technology — Coded representation of immersive media — Part 14: Scene Description for MPEG-I Media”

[4.3.k] *Khronos Group, The GL Transmission Format (glTF) 2.0 Specification*, Available at <https://github.com/KhronosGroup/glTF/tree/master/specification/2.0/>

**===== CHANGE =====**

### 4.3.5 MPEG Scene Description

A key technology in enabling immersive 3D user experiences is scene description. Scene description is used to describe the composition of a 3D scene, referencing and positioning the different 2D and 3D assets in the scene. The information provided in the scene description is then used by an application to render the 3D scene properly, using techniques such as Physically-Based Rendering (PBR) that produce realistic views.

A scene description is typically organized as a directed acyclic graph, typically a plain tree-structure, that represents an object-based hierarchy of the geometry of a scene and its attributes/properties. Nodes are organized in a parent-child hierarchy known informally as the node hierarchy. A node is called a root node when it doesn't have a parent. Any node can define a local space transformation.

Spatial transformations are represented by transformation matrices or separate transform operations such as translation, rotation, and scaling. The transformations are applied hierarchically and iteratively from the root node down to the child nodes. Scene description also support animation nodes that allow to animate properties of the corresponding objects over time.

This structure of scene description has the advantage of reduced processing complexity, e.g. while traversing the graph for rendering. An example operation that is simplified by the graph representation is the culling operation, where branches of the graph are omitted, if deemed that the parent node’s space is not visible or relevant (level of detail culling) to the rendering of the current view frustum.

While there are many proprietary solutions for scene description (typically at the heat of game engines, VFX design tools or AR/VR authoring tools), several solutions have also been standardized. In particular, in 2001, Virtual Reality Modeling Language (VRML), which uses and XML syntax, was the first scene description solution to be standardized for WEB usages. Later on, OpenSceneGraph, an open source project using OpenGL that has been released in 2005, has been used as a component in several computer games and rendering platforms such as Delta3D or FlightGear. In 2010, X3D, standardized by ISO/IEC, became the successor of VRML, introduced binary formats and JSON format for scene graph description and featured new capabilities such as multi-texture rendering, shading, real-time environment lightning, and culling. More recently in 2015, the Khronos Group, released the Graphics Library Tramsmission Format (glTF) standard. glTF is an API-neutral, runtime asset delivery format that is based on the JSON format. It is intended to be efficient and interoperable and can be used as a common scene description format for 3D content tools and services.

To address the needs of immersive applications, MPEG is finalizing the development of a scene description solution that adds extensions to glTF to support scene description. glTF 2.0 [4.3.k] provides a solid and efficient baseline for exchangeable and interoperable scene descriptions. However, glTF 2.0 has traditionally been focused on static scenes and assets, which makes it unfit to address the requirements and needs of dynamic and rich 3D scenes in immersive environments.

As part of its effort to define solutions for immersive multimedia, MPEG has identified the following gaps in glTF 2.0:

* No support for timed media
* No support for audio
* Limited support for interactions with the scene and the assets in the scene
* No support for local and real-time media, which are crucial for example for AR experiences

Based on this analysis, MPEG has an ongoing project to extend glTF2.0 with the ability to add timed media to gltf2.0-based scenes standardized in ISO/IEC 23090-14 [4.3.j]. Table 4.3.5-1 summarizes the currently defined extensions to glTF 2.0 in ISO/IEC 23090-14 [4.3.j].

Table 4.3.5-1: MPEG-I defined extensions to gltf2.0 in ISO/IEC 23090-14 [4.3.j].

|  |  |
| --- | --- |
| Extension Name | Brief Description |
| MPEG\_media | Extension for referencing external media sources. |
| MPEG\_accessor\_timed | An accessor extension to support timed media. |
| MPEG\_buffer\_circular | A buffer extension to support circular buffers. |
| MPEG\_scene\_dynamic | An extension to support scene updates. |
| MPEG\_texture\_video | A texture extension to support video textures. |
| MPEG\_mesh\_linking | An extension to link two meshes and provide mapping information |
| MPEG\_audio\_spatial | Adds support for spatial audio. |
| MPEG\_viewport\_recommended | An extension to describe a recommended viewport. |
| MPEG\_animation\_timing | An extension to control animation timelines. |

Additional extensions for the support of interactivity and AR are currently being developed and will be part of the MPEG Scene Description in the next phase.

MPEG also developed an architecture to guide the work on immersive media and scene description. Figure 4.3.5-1 depicts the MPEG-I architecture and defines the key interfaces.

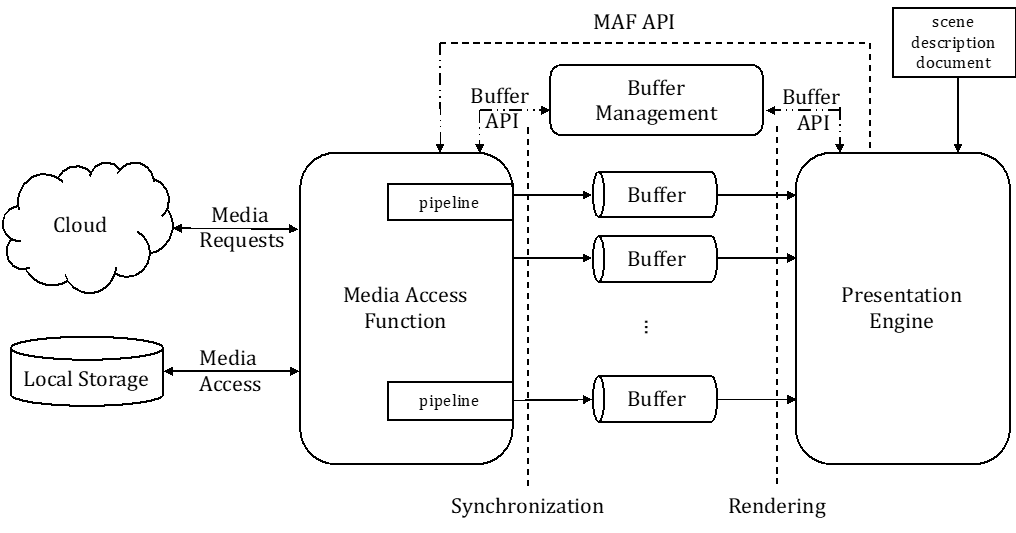


Figure 4.3.5-1: MPEG-I architecture and defines the key interfaces

The design focuses mainly on buffers as means for data exchange throughout the media access and rendering pipeline. It also defines a Media Access Function API to request media that is referenced by the scene description, which will be made accessible through buffers.