Text

Description automatically generated ISO/IEC JTC 1/SC 29/WG 3 N01647

**ISO/IEC JTC 1/SC 29/WG 3**

**MPEG Systems   
Convenorship: KATS (Korea, Republic of)**

**Document type:** Output Document

**Title:** Text of ISO/IEC 23090-24 DAM1 Conformance and reference software for scene description on haptics, augmented reality, avatars, interactivity, and lighting

**Status:** Approved

**Date of document:** 2025-10-23

**Source:** ISO/IEC JTC 1/SC 29/WG 3

**Expected action:** ACT

**Action due date:** 2025-10-23

**No. of pages:** 16(with cover page)

**Email of Convenor:** young.L@samsung.com

**Committee URL:** <https://isotc.iso.org/livelink/livelink/open/jtc1sc29wg3>

**INTERNATIONAL ORGANISATION FOR STANDARDISATION**

**ORGANISATION INTERNATIONALE DE NORMALISATION**

**ISO/IEC JTC 1/SC 29/WG 3**

**CODING OF MOVING PICTURES AND AUDIO**

**ISO/IEC JTC 1/SC 29/WG 3 N** **01647**

**Geneva, Switzerland – October 2025**

|  |  |
| --- | --- |
| **Title** | **Potential improvements of ISO/IEC 23090-24 DAM1 Conformance and reference software for scene description- Amendment 1: Support for Haptics, augmented reality, avatars, interactivity, audio and lighting** |
| **Source** | **WG 03, MPEG Systems** |
| **Status** | **Approved** |
| **Serial Number** | **25621** |

**ISO/IEC 23090-24 Amd1**

ISO/IEC TC JTC 1/SC 29/WG3

Secretariat: JISC

**Information technology — Coded representation of immersive media — Part 24: Conformance and reference software for scene description – Amendment 1: Support for Haptics, augmented reality, avatars, interactivity, and lighting**

Potential improvement of DAM 1

**Warning for WDs and CDs**

This document is not an ISO International Standard. It is distributed for review and comment. It is subject to change without notice and may not be referred to as an International Standard.

Recipients of this draft are invited to submit, with their comments, notification of any relevant patent rights of which they are aware and to provide supporting documentation.

*To help you, this guide on writing standards was produced by the ISO/TMB and is available at* [*https://www.iso.org/iso/how-to-write-standards.pdf*](https://www.iso.org/iso/how-to-write-standards.pdf)

*A model manuscript of a draft International Standard (known as “The Rice Model”) is available at* [*https://www.iso.org/iso/model\_document-rice\_model.pdf*](https://www.iso.org/iso/model_document-rice_model.pdf)

© ISO 2022

All rights reserved. Unless otherwise specified, or required in the context of its implementation, no part of this publication may be reproduced or utilized otherwise in any form or by any means, electronic or mechanical, including photocopying, or posting on the internet or an intranet, without prior written permission. Permission can be requested from either ISO at the address below or ISO’s member body in the country of the requester.

ISO copyright office

CP 401 • Ch. de Blandonnet 8

CH-1214 Vernier, Geneva

Phone: +41 22 749 01 11

Email: copyright@iso.org

Website: www.iso.org

Published in Switzerland

Information technology — Coded representation of immersive media — Part 24: Conformance and reference software for Scene Description – Amendment 1: Support for Haptics, augmented reality, avatars, interactivity, and lighting

========== First change ===========

*Add the following terms to Section 3.1.*

**3.2 Unit test**

Unit test is a software testing technique in which individual units or components of a software application are tested in isolation. This process helps in identifying issues in the development cycle and enhancing code quality thus promoting the reliability of the code.

**3.3 XR**

Extended Reality

========== Next change ===========

*Add the following paragraph to Section 4.1*

The reference software version 3.0 with advance features such as haptics, augmented reality, avatars, interactivity and lighting is available in https://standards.iso.org/iso-iec/23090/-24/ed-1/en/amd/1/. The reference software ‘pympegsd’ is released with a tag version 3.0. The version 3.0 of the reference software contains implementation for the following extensions.

* Anchoring extensions: MPEG\_node\_anchor and MPEG\_anchor
* Interactivity extensions: MPEG\_scene\_interactivity and MPEG\_node\_interactivity
* Sampler extension: MPEG\_sampler\_YCbCr
* Lighting extension: MPEG\_lights\_texture\_based
* V3C extension: MPEG\_primitive\_V3C
* Avatar extension: MPEG\_node\_avatar

[Editors Note: The reference software version 3.0 is scheduled to be released at FDAM stage. Currently, the reference software is at version 2.0.3.]

========== Next change ===========

*Change the implementation status for MPEG\_scene\_dynamic extension to* ***Yes*** *in Table 1*

========== Next change ===========

*Add the following items to Table 1 in sub-clause 4.2.*

|  |  |  |
| --- | --- | --- |
| **Extension name** | **Brief description** | **Implementation status** |
| MPEG\_anchor | The extension allows to ancho the glTF scene onto the user’s physical environment. | Yes |
| MPEG\_node\_anchor | The extension allows to anchor of some of the root nodes in the scene description format onto the user’s physical environment. | Yes |
| MPEG\_lights\_texture\_based | The extension expands the functionalities of the EXT\_lights\_image\_based extension by enabling the lighting information to vary over time | Yes |
| MPEG\_node\_interactivity | The extension is used to complement the interactivity defined at the scene level. | Yes |
| MPEG\_scene\_interactivity | The extension describes an interactivity model based on trigger, action and behaviours. | Yes |
| MPEG\_sampler\_YCbCr | The extension describes properties to sample a video texture i.e. YCbCr natively in parallel processing devices. | Yes |
| MPEG\_primitive\_V3C | The extension describes the syntax of the V3C object which is provided as an extension to mesh.primitive in a scene description format. The extension refers to the decoded data of a V3C object. | Yes |
| MPEG\_avatar | The extension describes an avatar entity at a node level in the scene description format. | Yes |

========== Next change ===========

*Add the following paragraphs to sub-clause 4.2.*

The version 3.0 of the reference software supports parsing of “targetSampleRate” property on audio sources with MPEG\_audio\_spatial extension as specified in ISO/IEC 23090-14:2025 2nd edition.

The version 3.0 of the reference software supports parsing of “media.alternative.track” property in MPEG\_media extension. When parsing fails, a default track id with value 0 is passed to the MAF pipeline. This ensures compatibility with test assets that refer to media files with extension such as “mp3”.

========== Next change ===========

*Update Table 2 with the information as below.*

Table 2- Library packages

|  |  |  |
| --- | --- | --- |
| **Library package** | **Description** | **Version** |
| av | Python binding to FFmpeg library | 10.0.0 |
| networkx | Creating and manipulating graphs and network | 2.8.4 |
| pillow | Imaging processing library | 9.2.0 |
| pyglet | Windowing and multimedia library | 1.5.27 |
| scipy | Fast N-dimensional array manipulation | 1.8.1 |
| bitstream | Manage binary data as bitstreams | 3.1.9 |
| trimesh | Loading and using triangular meshes | 3.12.8 |
| pyopenxr | Python binding for OpenXR SDK | Any version greater than 1.0.2301 |
| pyrender | Rendering engine | 0.1.45 |
| aiortc | Library for real-time communication | 1.4.0 |
| aiohttp | Asynchronous HTTP client/server | 3.8.1 |
| pyopengl | Python binding to OpenGL | 3.1.6 |
| strenum | Specialized type of enumeration that allows enumeration members to behave like strings. | 0.4.10 |
| opencv-contrib-python | Extended version of the OpenCV library for Python | Any version greater than 4.0 |

========== Next change ===========

*Add the following sentence after Table 2*

Note: The reference software is tested with python version 3.10. The reference software may or may not work with other python versions.

========== Next change ===========

*Replace sub-clause 4.8 with following paragraph.*

## 4.8 MPEG\_anchor extension testing

The assets with the anchoring extensions are provided as described in Section 7.1.

For all test cases, start the render without the animations i.e. no “-a” argument when running the pympegsd renderer and pressing the key "c" to start the anchoring.

There are in total 16 test files to validate the function of MPEG\_anchor extension. The test files are available as mentioned in section 7.1. A base of 5 assets is used to build the files as shown in Table 4. The different configurations allow testing of the 8 different trackables and MPEG\_anchor extension in glTF at node or at scene level.

Only trackable of type “TRACKABLE\_FLOOR” and “TRACKABLE\_MARKER\_2D” are simulated in the example files.

*Table 3: glTF files with Anchoring extension*

|  |  |  |
| --- | --- | --- |
| **glTF file** | **Trackable type** | **Extension level** |
| anchorTest\_v0 | TRACKABLE\_FLOOR | glTF and scene |
| anchorTest\_v1 | TRACKABLE\_FLOOR | glTF and node |
| anchorTest\_v2 | TRACKABLE\_VIEWER | glTF and scene |
| anchorTest\_v3 | TRACKABLE\_VIEWER | glTF and node |
| anchorTest\_v4 | TRACKABLE\_CONTROLLER | glTF and scene |
| anchorTest\_v5 | TRACKABLE\_CONTROLLER | glTF and node |
| anchorTest\_v6 | TRACKABLE\_GEOMETRIC | glTF and scene |
| anchorTest\_v7 | TRACKABLE\_GEOMETRIC | glTF and node |
| anchorTest\_v8 | TRACKABLE\_MARKER\_2D | glTF and scene |
| anchorTest\_v9 | TRACKABLE\_MARKER\_2D | glTF and node |
| anchorTest\_v10 | TRACKABLE\_MARKER\_3D | glTF and scene |
| anchorTest\_v11 | TRACKABLE\_MARKER\_3D | glTF and node |
| anchorTest\_v12 | TRACKABLE\_MARKER\_GEO | glTF and scene |
| anchorTest\_v13 | TRACKABLE\_MARKER\_GEO | glTF and node |
| anchorTest\_v14 | TRACKABLE\_APPLICATION | glTF and scene |
| anchorTest\_v15 | TRACKABLE\_APPLICATION | glTF and node |

Each glTF file includes 5 assets represented by different nodes in a scene. A node in a scene is mapped to an asset. The list of assets in the scene are shown in in Table 4.

*Table 4: glTF file with nodes describing different elements in the scene*

|  |  |  |
| --- | --- | --- |
| Node 0 | Floor | Real Element |
| Node 1 | Image | Real Element |
| Node 2 | Camera | Real Element |
| Node 3 | Blue Cube | Children |
| Node 4 | Red Cone | Root |

The floor and image nodes are used as real elements by simulation software. The camera property is included in the glTF file as the user camera. The virtual scene is composed of two elements, a red cone i.e. Node4 as shown in Table 4 and a blue cube i.e. Node3 as shown in Table 4.

When the MPEG\_anchor extension is present at scene level, all the root nodes of the scene are anchored. As some nodes are used as real element for the simulation, these nodes must be excluded from the anchoring process e.g. nodes corresponding to Floor, Image and Camera assets.

The reference software has a folder named “*simulation*”. In this folder, a simulation.ini file allows to identify which root nodes to be anchored in the scene array.

The *simulation* folder contains source code and data to simulate the trackable. The *simulation.ini* file in the *simulation/data* folder contains parameters for the simulation.

For **TRACKABLE\_MARKER\_2D**, the size of the maker must be given as follows:

[TRACKABLE\_MARKER\_2D]

marker\_width = 6.181522

marker\_height = 4.71872

For **TRACKABLE\_MARKER\_2D**, the index of the node used as floor plane must be given as follows:

[TRACKABLE\_FLOOR]

floor\_node = 0

For **MPEG\_scene\_anchor** extension, the index of the root node must be given as follows:

[root\_node\_in\_scene]

root\_node = 4

For TRACKABLE\_FLOOR (anchorTest\_v0, anchorTest\_v1), the detection of the floor plane is not implemented. Instead, a mesh is referenced by the Floor node, i.e. Node 0 as shown in Table 4, of the glTF file as a result of the floor plane detection. Tracking is not implemented.

The common axes for anchoring examples are defined as shown in Figure 3.

Chart

Description automatically generated with medium confidence

*Figure 3: Common axes for anchoring examples*

The position of the Red Cone is in the center of the plane i.e. there is no transformation information for the Red Cone provided in the glTF File. Figure 4 shows a screenshot of the result of the anchoring Red Cone on a Floor in the grey square. The Red Cone is positioned at the center of the Floor.

A close up of a sign

AI-generated content may be incorrect.

Figure 4 Screenshot of trackable floor anchoring

For TRACKABLE\_MARKER\_2D in anchorTest\_v8 and anchorTest\_v9 as shown in Table 3, the Image i.e. Node 1 as shown in Table 4 is used as the 2D marker.

Note: The Image has the word “Metro” written on it.

The result of the anchoring process is shown in Figure 4. The position of the Red Cone is the center of the image as there is not transformation information provided for the Red Cone Node in the glTF File.

Note: The Image marker has been rotated of 180°.

Figure 5 shows a screen shot of the result of anchoring Red Cone on a 2D marker. The cone is positioned at the center of the image.

A close-up of a logo

AI-generated content may be incorrect.

Figure 5 Screenshot of the trackable marker anchoring

========== Next change ===========

*Replace sub-clause 4.9 with the following clause*

## MPEG\_interactivity testing

The testing procedure for MPEG\_interactivity extension is described in this section. The testing procedure for the assets is described below:

* For all test cases, start the render without the animations i.e. no “-a” argument when running the pympegsd renderer.
* For “UseCase\_01xxxxx.gltf” and “UseCase\_0Xxxxxx.gltf” test cases, once the renderer window is open, start the animation i.e. by pressing the key “a” and let the balls collide.
* For “UseCase\_02xxxxx.gltf” and “UseCase\_03xxxxx.gltf” test cases, enter full screen by pressing the key “f” and use the mouse to make the scene visible and/or get it closer or further to the camera.
* For “UseCase\_03-variant1-Interactivity”, by pressing the key “l” to force animation loop.

Table 5 describes 4 test cases in which 3 balls are rolling on a surface and collide.

Table 5 Different variants of Use case 1 of Interactivity

|  |  |
| --- | --- |
| glTF file | Description |
| UseCase\_01-variant1-Interactivity | When the red and yellow balls collide, the material of the gray ball changes to blue color. |
| UseCase\_01-variant2-Interactivity | When the red and yellow balls collide, the material of the gray ball changes to blue color and simultaneously play a sound. When the red and gray/blue balls collide, the sound stops playing. |
| UseCase\_01-variant3-Interactivity | When the red and yellow balls collide, the material of the gray ball changes to blue color and play a sound 3s later. |
| UseCase\_01-variant4-Interactivity | Additional glTF file with the description of many triggers and actions that may be combined to create new behaviors and use cases. |

Table 6 describes 4 test cases with a virtual scene composed of 3 balls, 3 cones and 1 cylinder.

Table 6 Different variants of Use case 2 of Interactivity

|  |  |
| --- | --- |
| glTF file | Description |
| UseCase\_02-variant1-Interactivity | When the scene is visible OR the scene is close to the camera, the 7 objects bump, and the 3 cones play a sound. |
| UseCase\_02-variant2-Interactivity | When the scene is visible AND the scene is close to the camera, the 7 objects bump and the 3 cones play a sound after 2, 4 and 6s. |
| UseCase\_03-variant1-Interactivity | When the scene is close to the camera, the 7 objects bump, and the 3 cones play a sound. When the scene is far, the objects stop bumping. |
| UseCase\_03-variant2-Interactivity | When the scene is visible AND the scene is close to the camera, the 7 objects bump, and the 3 cones play a sound. |

Table 7 describes an avatar file contains a virtual scene with several cubes. An MPEG\_node\_avatar extension is defined but not processed in this branch.

Table 7 Description of MPEG\_node\_anchor testing

|  |  |
| --- | --- |
| glTF file | Description |
| cubes\_avatar\_actions | After a mouse click on the scene, a SET AVATAR action is launch (urn console log only). |

Table 8 describes as a haptic test case with 3 balls are rolling on a surface and collide. The collision effect triggers haptic feedback.

Table 8 Description of Haptics extension testing

|  |  |
| --- | --- |
| glTF file | Description |
| UseCase\_0X-variant1-Haptic | When the red and yellow balls collide, a haptic effect is played (console log only). |

========== Next change ===========

*Reindex sub-clause 4.8 as sub-clause 4.10*

========== Next change ===========

*Reindex sub-clause 4.9 as sub-clause 4.11*

========== Next change ===========

*Reindex the following figures*

* *Figure 3 to Figure 6*
* *Figure 4 to Figure 7*
* *Figure 5 to Figure 8*
* *Figure 6 to Figure 9*
* *Figure 7 to Figure 10*

========== Next change ===========

*Add the following paragraph in sub-clause 6.2.2.*

The conformance software is released with a tag version 2.0 covering advanced extensions as described in sub-clause 6.2.3.

The version 2 .0 of the conformance software is available in <https://standards.iso.org/iso-iec/23090/-24/ed-1/en/amd/1/>.

========== Next change ===========

## Support for advanced extensions

The version 2.0 of conformance software in addition to extensions implemented in version 1.0 provides support for the following advance extensions:

1. MPEG\_anchor extension
2. MPEG\_node\_anchor extension
3. MPEG\_lights\_texture\_based extension
4. MPEG\_lights\_punctual extension
5. MPEG\_material\_haptic extension
6. MPEG\_haptic extension
7. MPEG\_scene\_interactivity extension
8. MPEG\_node\_interactivity extension
9. MPEG\_avatar extension
10. MPEG\_sampler\_YCbCr extension
11. MPEG\_primitive\_V3C extension

========== Next change ===========

*Add new sub-clause 6.2.4 as following*

## Unit testing

The version 2.0 of conformance software incorporates the functions to run unit tests. The unit tests are run on valid glTF files with corresponding MPEG extensions. The unit tests check the validation report of a glTF file and checks the expected value(s) of the various properties described in a glTF file. The version 2.0 of the conformance software provides unit testing for the following extensions as specified in ISO/IEC 23090-14:2025 2nd edition.

* MPEG\_media extension
* MPEG\_sampler\_YCbCr extension
* MPEG\_primitive\_V3C extension

========== Next change ===========

*Add the following paragraph in sub-clause 7.1*

The test assets with advanced extensions as specified in ISO/IEC 23090-14:2nd edition are available in asset file available at https://standards.iso.org/iso-iec/23090/-24/ed-1/en/amd/1/. The test asset file includes files with MPEG extensions such as MPEG\_anchor, MPEG\_interactivity, etc.

========== Next change ===========

*Add a new Annex*

1. (informative)  
     
   Implementation of MPEG-I Scene description in 5G-Media Action Group3

## A.1. Overview

This annex outlines the status and development of XR (Extended Reality) Media integration into 5G networks, focusing on open-source tools, standards, and reference implementations developed by 5G-MAG3. 5G-MAG3 in their project have tested with some of the core extensions and features specified in ISO/IEC 23090-14.

## A.2. Resources

5G-MAG3 has implemented a set of tools based on 3GPP and ISO/IEC (notably ISO/IEC 23090-14 for MPEG-I Scene Description). The reference implementation includes:

* A Unity1-based XR player loads and render MPEG-I scenes in real time. The Unity-based XR player is available at <https://github.com/5G-MAG/rt-xr-unity-player>.
* MAF media pipeline plugin for the Unity-based XR Player's media player implementation of Media Access Functions API (MAF) as defined in ISO/IEC 23090-14:2025 2nd edition. The MAF plugin is available at <https://github.com/5G-MAG/rt-xr-maf-native>
* An exporter for 3D content based on Blender2 tool which exports 3D scenes with extensions and features defined in ISO/IEC 23090-14:2025 2nd edition such as audio/video textures and XR anchors. The exporter is available at <https://github.com/5G-MAG/rt-xr-blender-exporter>.
* A collection of scene description assets supporting functions such as timed media, interactivity, haptics, and anchoring is available at <https://github.com/5G-MAG/rt-xr-content>.

A tutorial on using 5G-MAG reference tools is available at https://5g-mag.github.io/Getting-Started/pages/xr-media-integration-in-5g/tutorials/blender\_exporter\_unity\_player.html. A comprehensive guide to create test asset is available at <https://5g-mag.github.io/Getting-Started/pages/xr-media-integration-in-5g/tutorials/creating_test_assets.html>.

|  |
| --- |
|  |

1 Unity Trademark is a trademark or registered trademark of Unity Technologies or its affiliates in the U.S. and elsewhere. More information is available at <https://unity.com/legal/branding-trademarks>.

2 Blender is a registered trademark (®) of the Blender Foundation in EU and USA.

3 5G-Media Action Group develops open-source software under the 5G-MAG Public License v1.0, which is a modified version of the Apache 2.0 license. The software is free for non-commercial use (e.g., research, testing, validation), but commercial use requires adherence to the FRAND terms.

## A.3. Test Assets

5G-MAG collects sample test assets in a repository which is available at https://github.com/5G-MAG/rt-xr-content. Following are the links to the sample test assets that make use of the MPEG\_media, MPEG\_buffer\_circular, MPEG\_accessor\_timed, MPEG\_texture\_videos and MPEG\_audio\_spatial extensions.

• https://github.com/5G-MAG/rt-xr-content/tree/main/studio\_apartment

• https://github.com/5G-MAG/rt-xr-content/tree/main/TV

• https://github.com/5G-MAG/rt-xr-content/tree/main/ar-video-plane

• <https://github.com/5G-MAG/rt-xr-content/tree/main/video>

The above-mentioned sample test assets were generated using the Blender add-on as described in sub-clause A.2. The copyright and license are described for each test sample asset. For example, the license for the “studio-apartment” test asset is available at https://github.com/5G-MAG/rt-xr-content/tree/main/studio\_apartment#legal).

Note: 5G-MAG is also working on additional sample test assets that makes use of MPEG-DASH streams. These should be available in the future.

## A.4. Features

Some of the key features from ISO/IEC 23090-14:2025 2nd edition implemented by 5G-MAG in their suite of toolsets:

1. **MPEG Extensions**:

* MPEG\_media, MPEG\_texture\_video and MPEG\_audio\_spatial extensions
* MPEG\_scene\_interactivity, MPEG\_anchor, and MPEG\_node\_avatar extensions

1. **Interactivity**: Trigger-action model using OpenXR inputs, collisions, proximity, etc.
2. **Anchoring**: Real-world feature anchoring including features to anchor assets based on planes, markers, geospatial coordinates, and etc.

In the upcoming release from 5G-MAG, some advanced features will be added to the suite of their toolsets. This includes the support for:

* Playback on Android Head Mounted Displays (e.g., Meta Quest 3)
* Real-time avatar communication
* Spatial audio codecs
* Integration with other 5G-MAG tools

## A.5. Learning & Demos

5G-MAG publishes tutorials on the usage of tools and other resources on their website at https://www.5g-mag.com/tutorials . A video library describes the use of 5G-MAG tools for ISO/IEC 23090-14:2025 2nd edition is available at https://www.youtube.com/watch?v=CVpTbiN-tmc&list=PLFqKJZ78\_IWVk0\_h1oeizy9IZ0DmOUXkA . 5G-MAG publishes information on the trial, demos and project which are being carried out to understand the production and distribution of media content and services. This information is available at <https://www.5g-mag.com/trials>.