Text

Description automatically generatedISO/IEC JTC 1/SC 29/WG 03 N01399

**ISO/IEC JTC 1/SC 29/WG 03  
MPEG Systems   
Convenorship: KATS (Korea, Republic of)**

**Document type:** Output document

**Title:** Working draft of ISO/IEC 23090-14 2nd edition AMD 1: Support of MPEG-I immersive audio, scene understanding and other extensions

**Date of document:** 2024-11-15

**Source:** WG03

**No. of pages:** 10 (with cover page)

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

*Modify clause 8 with the following content*

*In clause 8.2.1 change*

The MPEG\_node\_interactivity extension is used to complement the interactivity defined at the scene level. One particular case is the definition of the parameters for the physics engine. That is, when an MPEG\_node\_interactivity extension contains a trigger of type TRIGGER\_COLLISION without being referenced by a trigger of type TRIGGER\_COLLISION at the MPEG\_scene\_interactivity extension, this node shall not be considered for collision detection and instead only be used by the physics engine.

*To*

The MPEG\_node\_interactivity extension is used to complement the interactivity extension defined at the scene level. One particular case is the definition of the parameters for a physics engine. That is, when an MPEG\_node\_interactivity extension contains a trigger of type TRIGGER\_COLLISION without being referenced by a trigger of type TRIGGER\_COLLISION at the MPEG\_scene\_interactivity extension, this node shall not be considered for collision detection and instead only be used by the physics engine.

When specified, at node and/or scene level, a physics object specifies a set of parameters for the physic engine, used by the application.

*In clause 8.2.2.1, change the table 31 to*

**Table 31: Semantic of the MPEG\_scene\_interactivity extension**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Type** | **Usage** | **Default** | **Description** |
| physics | object | O | N/A | Provides a set of parameters at scene level to be used for the physics simulation. The semantics of this object is given in table xx3. |
| triggers | array | M | [] | Contains the definition of all the triggers used in that scene |
| actions | array | M | [] | Contains the definition of all the actions used in that scene |
| behaviors | array | M | [] | Contains the definition of all the behaviors used in that scene. A behavior is composed of a pair of (triggers, actions), control parameters of triggers and actions, a priority weight and an optional interrupt action |

*And add the following text*

The semantics of a *physics* object at scene level, are defined in Table xx3.

**Table xx3: Semantic of a scene level *physics* object**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Type** | **Usage** | **Default** | **Description** |
| recommendedPhysicsHighPrecision | Boolean | O | false | Determines whether the application should enable a more deterministic and precise physic simulation |
| gravity | Number | O | -9.81 | Determine the gravity for the whole scene. In meter per secondsquare (m.s-2), as defined in the international unit system. |
| recommendedPhysicsFrameRate | Number | O | 50 | Provides the recommended frame rate at which the Physics Engine should operate. In frame per second, as defined in the international unit system. |
| bounceThreshold | number | O | 1 | A contact with a relative velocity below this threshold will not result in a bounce. In meter per second (m.s-1), as defined in the international unit system. |

*In clause 8.2.2.2, change the table 51 to*

**Table 51: Semantic of the MPEG\_node\_interactivity.trigger extension**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Type** | **Usage** | **Default** | **Description** |
| type | enumeration | M |  | One element of Table 8 that defines the type of the trigger. |
| if (type == TRIGGER\_COLLISION){ |  |  |  |  |
| collider | integer | M |  | the index of the mesh element that provides the collider geometry for the current node.  The collider mesh may reference a material. |
| isStatic | boolean | M |  | If True, the collider is defined as a static collider. |
| physics | object | O | N/A | Provides a set of parameters at node level to be used for the physics simulation. The semantics of this object is given in table xx4. |
| primitives | array(Primitive) | O | N/A | List of primitives used to activate the proximity or collision trigger.  Semantics are presented in Table 34. |
| } |  |  |  |  |
| … |  |  |  |  |

*And add the following text*

The semantics of a *physics* object at node level, are defined in Table xx4.

**Table xx4: Semantic of a node level *physics* object**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Type** | **Usage** | **Default** | **Description** |
| needPreciseCollisionDetection | Boolean | O | false | If true, the physics engine should handle the collision detection more accurately by increasing the detection rate for this node. |
| linearDamping | Number | O | 0 | A non-negative value, in second-1 (s-1), as defined in the international unit system. It defines the linear drag coefficient which corresponds to the rate of decrease of the linear velocity over time.  It is used to compute a new velocity value V(t) at each simulation step (dt):  V(t+dt) = V(t)\*(1-linearDamping\*dt), the velocity being clamped to 0. |
| angularDamping | number | O | 0 | A non-negative value, in second-1 (s-1), as defined in the international unit system. It defines the angular drag coefficient which corresponds to the rate of decrease of the angular velocity over time.  It is used to compute a new velocity value V(t) at each simulation step (dt):  V(t+dt) = V(t)\*(1-angularDamping\*dt), the velocity being clamped to 0. |
| useGravity | boolean | M |  | Indicates if the gravity affects the object |
| mass | number | M |  | Mass of the object in kilogram, as defined in the international unit system. |
| restitution | number | M |  | Provides the ratio of the final to initial relative velocity between two objects after they collide |
| staticFriction | number | M |  | Unitless static friction coefficient as defined in the Coulomb friction model. Friction is the quantity which prevents surfaces from sliding off each other. Static friction is used when the object is lying still. It will prevent the object from starting to move. |
| dynamicFriction | Number | M |  | Unitless static friction coefficient as defined in the Coulomb friction model. When a large enough force is applied to the object, a dynamic friction is used, and will attempt to slow down the object while in contact with another. |
| } |  |  |  |  |

*In clause 8.2.3 change*

If the scene description document contains a description of physics properties based on another physics model, then that physics model shall take precedence in the processing of the scene.

Otherwise, the application shall handle a physics simulation if the usePhysics Boolean is TRUE on any of the collision trigger extensions defined at the node level. When a collision occurs between two nodes, the application should calculate the combination of the restitution, static friction and dynamic friction values based on the values provided by the collision trigger extension of the two nodes.

*To*

The application shall handle a physics simulation if a physics object is defined at a scene level or/and at any of the nodes. When a collision occurs between two nodes, the application should calculate the combination of the restitution, static friction and dynamic friction values based on the values provided by the physics objects of the two nodes.

*In clause 8, add the following sub-section*

## Support of MPEG-I immersive audio

## General

MPEG-I Immersive Audio has been specified in ISO/IEC 23090-4. The specification assumes the presence of an MPEG-I immersive audio renderer that will receive the MPEG-I audio bitstream, a set of MPEG-H audio streams, as well as information about some scene metadata, such as listener’s pose. It will then use the audio scene metadata in the MPEG-I audio bitstream, the decoded MPEG-H bitstreams, and the pose information to render the spatial audio.

The support of MPEG-I Immersive Audio is achieved by referencing an MPEG-I audio stream in in a MPEG-I scene description document.

The MPEG-I Audio bitstream contains a description of the audio scene that is independent of the main scene description consumed by the Presentation Engine. An alignment between the Presentation Engine and the Audio Renderer is needed, that goes beyond the traditional time alignment but includes also spatial alignment. For that, a mapping need to be established between the node in a MPEG-I scene description document and the node of the audio scene.

The following figure depicts an example of a mapping between a node that contains a car and an external audio node in an MPEG-I Audio bitstream, with a simplified geometry of that car and the attached audio sources. This mapping is described in a The MPEG node mapping extension.

A black and white image of a camera

Description automatically generated

This MPEG node mapping extension, identified by MPEG\_node\_mapping, can be used in a broader scope: It establishes a mapping between the node in a MPEG-I scene description document and an external entity, e.g. an MPEG-I audio renderer, that handles a dedicated scene graph, separate from the main scene description. When present, the MPEG\_node\_mapping extension shall be included in a node object.

The architecture for the support of such an external renderer is depicted in the following figure:

A diagram of a flowchart

Description automatically generated

The Generic Render Control API is an abstract API that is offered by external renderers to enable applications, such as Presentation Engines, to control the rendering process by aligning and synchronizing their rendering state to that of the Presentation Engine. This API is used by the Presentation Engine to configure and update the status of the external renderer.

## Semantics

The semantics of the MPEG\_node\_mapping extension is provided in the following table. When present, the MPEG\_node\_mapping extension shall be included in a node object.

**Table x1: Semantic of the MPEG\_node\_mapping extension**

| **Name** | **Type** | **Default** | **Usage** | **Description** |
| --- | --- | --- | --- | --- |
| mappings | array(object) |  | M | An array of mappings associated with the containing node. |
| component | string | “urn:mpeg:sd:component:default” | O | An identifier of the component associated with this mapping. The component may for instance be  “urn:mpeg:sd:component:audio-renderer” to indicate that the component is an audio renderer. |
| source | number | N/A | M | The index in the MPEG\_media that provides the media resource that contains the mapped element. |
| referenceId | number | N/A | M | An identifier of the element in the referenced resource. |
| transform | array(number) | Identity | O | A 4x4 TRS matrix which transforms the 3D coordinates of the node having this glTF extension expressed in the glTF2.0 scene coordinate system to the 3D coordinates of the node of the external renderer graph referenced by the referenceId identifier expressed in the external renderer scene coordinate system.  If the mapped node is a child of an AR Anchor/Trackable, the 4x4 TRS matrix transforms the 3D coordinates of the node having this glTF extension expressed in the AR Anchor/Trackable coordinate system to the 3D coordinates of the node of the external renderer graph referenced by the referenceId identifier expressed in the AR Anchor/Trackable coordinate system. |
| updateRecommendation | object | N/A | O | Indicate update recommendations for the node. The semantics is given in table x2. |
| supportsInteractivity | boolean | false | O | Indicates if interactivity actions applied to the node should be exposed if an API is made available to the Presentation Engine by the renderer of the resource. |

Table x2: Semantics of an updateRecommendation object

| **Name** | **Type** | **Default** | **Usage** | **Description** |
| --- | --- | --- | --- | --- |
| synchronizationOccurrences | array(enum) | [EVENT] | O | An array of synchronization occurrences. Each element of this array is an Enumerator with the following possible values:   * ONCE: the synchronization is done once at the configuration step, * EVENT: the synchronization is done based on the activation of one or more triggers (e.g., visibility, proximity, collision, user input) as those defined in the MPEG\_interactivity extension. The indices of the triggers, from the triggers array of the MPEG\_scene\_interactivity extension, are given in the *events* parameter, * N\_FRAME: the synchronization is periodic every N (1, 2, …) rendering frames. The N value is provided in the *frameNumber* parameter. |
| events | Array(number) | N/A | 0 | Array of indices of triggers from the triggers array of the MPEG\_scene\_interactivity extension. Required when EVENT is mentioned in the *synchronizationOccurrences* array. |
| frameNumber | Number | N/A | 0 | Indicate the periodicity, in number of frames, of the synchronisation, when N\_FRAME is mentioned in the *synchronizationOccurrences* array. |
| synchronizationOccurrenceCombination | string | “|” | O | A set of logical operations to apply to the synchronization occurrences. A ‘#’ indicates the occurrence index, ‘&’ indicates a logical AND operation, ‘|’ a logical OR operation and ‘~’ a NOT operation. Parenthesis are used to group some operations. Such a syntax may give the following string: “#1&~#2|(#3&#4)”. |

## Processing model

When processing the MPEG\_node\_mapping extension, the Presentation Engine shall identify nodes in the scene description that have a node mapping. The Presentation Engine shall determine if the component identified by the indicated component parameter supports the Generic Render Control API. If it does, the Presentation Engine shall pass the mapping information to the identified component.

The Presentation Engine shall then use the API to align the rendering with the component as configured over the API.

*Add the following to Annex A*

A.10 JSON schema for MPEG\_node\_mapping

MPEG\_node\_mapping schema is downloadable from <https://standards.iso.org/iso-iec/23090/-14/amd1/MPEG_node_mapping.schema.json>.