Text

Description automatically generated ISO/IEC JTC 1/SC 29/WG 7 N868

**ISO/IEC JTC 1/SC 29/WG 7  
MPEG 3D Graphics and Haptics Coding   
Convenorship: AFNOR (France)**

**Document type:** Output Document

**Title:** Text of ISO/IEC CD 23090-32 Carriage of haptics data

**Status:** Approved

**Date of document:** 2023-05

**Source:** ISO/IEC JTC 1/SC 29/WG 7

**Expected action:** None

**Action due date:** None

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

**Email of Convenor:** marius.preda @ imt . fr

**Committee URL:** [https://isotc.iso.org/livelink/livelink/open/jtc1sc29wg7](https://isotc.iso.org/livelink/livelink/open/jtc1sc29wg3)

**INTERNATIONAL ORGANIZATION FOR STANDARDIZATION**

**ORGANISATION INTERNATIONALE DE NORMALISATION**

**ISO/IEC JTC 1/SC 29/WG 7 MPEG Coding for 3D Graphics and haptics**

**ISO/IEC JTC 1/SC 29/WG 7 N868**

**May 2023, Antalya**

|  |  |
| --- | --- |
| **Title** | **Text of ISO/IEC CD 23090-32 Carriage of haptics data** |
| **Source** | **WG 7, MPEG Coding for 3D Graphics and Haptics** |
| **Status** | **Approved** |
| **Serial Number** | **22610** |

**Abstract**

This document contains a Committee Draft for the carriage of haptics data. It is based on text and decisions made at MPEG-142 (April 2023, Antalya) regarding the following input contributions:

* m62896, Update of the ISOBMFF format for haptics according to Haptics CD changes;
* m62987, Use of data packet type in ISOBMFF haptics binding;
* m63178, On haptics media presentation;
* m63235, Summary of Referenceable Code Points in ISO/IEC 23090-32;
* m63285, Proposed text for the CD of ISO/IEC 23090-32 Carriage of haptics data.

The changes in this document are with respect to m63285, version 3.

Contents

[1 Scope 5](#_Toc134530095)

[2 Normative references 5](#_Toc134530096)

[3 Terms, definitions and abbreviated terms 5](#_Toc134530097)

[3.1 Terms and definitions 5](#_Toc134530098)

[3.2 Abbreviated terms 6](#_Toc134530099)

[4 Overview 6](#_Toc134530100)

[4.1 Organization of this document 6](#_Toc134530101)

[4.2 Haptic media 7](#_Toc134530102)

[4.2.1 Structure of haptic media 7](#_Toc134530103)

[4.2.2 MIHS streams 7](#_Toc134530104)

[4.2.3 Temporal units and MIHS samples 8](#_Toc134530105)

[4.3 Synchronization with other media 9](#_Toc134530106)

[4.4 Summary of referenceable code points 9](#_Toc134530107)

[4.4.1 Brands 9](#_Toc134530108)

[4.4.2 Sample entry types 9](#_Toc134530109)

[4.4.3 Box types 10](#_Toc134530110)

[4.4.4 Sample grouping types 10](#_Toc134530111)

[5 Carriage of haptic coding data 10](#_Toc134530112)

[5.1 General 10](#_Toc134530113)

[5.2 MIHS streams and tracks 10](#_Toc134530114)

[5.2.1 MIHS sample entry 11](#_Toc134530115)

[5.2.2 MIHS configuration box 11](#_Toc134530116)

[5.2.3 Haptic experience description box 12](#_Toc134530117)

[5.2.4 Haptic avatar description box 13](#_Toc134530118)

[5.2.5 Haptic perception description box 14](#_Toc134530119)

[5.2.6 Haptic reference device description box 15](#_Toc134530120)

[5.2.7 Haptic channel description box 17](#_Toc134530121)

[5.2.8 Haptic band description box 19](#_Toc134530122)

[5.2.9 Sample content format 20](#_Toc134530123)

[5.2.10 Haptics presentation dependency sample group 21](#_Toc134530124)

[Annex A (Normative) File format toolsets and brands 22](#_Toc134530125)

[A.1 General 22](#_Toc134530126)

[A.2 MIHS brand 22](#_Toc134530127)

[Annex B (Normative) MIME types and sub-parameters 23](#_Toc134530128)

[B.1 MIME types and sub-types 23](#_Toc134530129)

[B.2 Sub-parameters for ‘codecs’ parameter 23](#_Toc134530130)

[B.2.1 General 23](#_Toc134530131)

[B.2.2 Haptic codec family 23](#_Toc134530132)

[Annex C (Informative) Multiple MIHS tracks and alternate groups 24](#_Toc134530133)

[C.1 General 24](#_Toc134530134)

[C.2 Criteria for alternate groups of MIHS tracks 24](#_Toc134530135)

[Annex D (informative) Player handling of MIHS tracks 25](#_Toc134530136)

[D.1 General 25](#_Toc134530137)

Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work.

The procedures used to develop this document and those intended for its further maintenance are described in the ISO/IEC Directives, Part 1. In particular, the different approval criteria needed for the different types of document should be noted. This document was drafted in accordance with the editorial rules of the ISO/IEC Directives, Part 2 (see [www.iso.org/directives](http://www.iso.org/directives) or [www.iec.ch/members\_experts/refdocs](http://www.iec.ch/members_experts/refdocs)).

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO and IEC shall not be held responsible for identifying any or all such patent rights. Details of any patent rights identified during the development of the document will be in the Introduction and/or on the ISO list of patent declarations received (see [www.iso.org/patents](https://www.iso.org/iso-standards-and-patents.html)) or the IEC list of patent declarations received (see [patents.iec.ch](https://patents.iec.ch)).

Any trade name used in this document is information given for the convenience of users and does not constitute an endorsement.

For an explanation of the voluntary nature of standards, the meaning of ISO specific terms and expressions related to conformity assessment, as well as information about ISO's adherence to the World Trade Organization (WTO) principles in the Technical Barriers to Trade (TBT) see [www.iso.org/iso/foreword.html](http://www.iso.org/iso/foreword.html). In the IEC, see [www.iec.ch/understanding-standards](https://www.iec.ch/understanding-standards).

This document was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 29, *Coding of audio, picture, multimedia and hypermedia information*.

A list of all parts in the ISO/IEC 23090 series can be found on the ISO and IEC websites.

Any feedback or questions on this document should be directed to the user’s national standards body. A complete listing of these bodies can be found at [www.iso.org/members.html](http://www.iso.org/members.html) and [www.iec.ch/national-committees](http://www.iec.ch/national-committees).

Introduction

This document addresses the carriage of haptic media in an ISO base media file. A separate document, ISO/IEC 23090-31, specifies the MIHS stream format for haptic media. This document specifies the process for incorporating MIHS formatted bitstreams into an ISO base media file.

Information technology — Coded representation of immersive media —

Part 32:  
Carriage of haptics in ISO base media file formatINTERNATIONAL STANDARD© ISO/IEC 2015 – All rights reservedISO/IEC 15444-12:2015(E) 63Part 12: ISO base media file formatInformation technology — JPEG 2000 image coding systemTechnologies de l'information — Codage des objets audiovisuels — Partie 12: Format ISO de base pour les fichiers médiasInformation technology — JPEG 2000 image coding system — Part 12: ISO base media file formatE2015-02-20(60) PublicationISO/IECISO/IEC J   International Standard 2015ISO/IEC 15444‑ISO/IEC 15444‑12ISO/IEC 15444-12  Coding of audio, picture, multimedia and hypermedia informationInformation technology 291 2見出し 2見出し 1    02 STD Version 2.1c260   4C:\Users\shinji\_w\AppData\Roaming\Microsoft\Templates\STD\w15177\_14496\_5th.-restyle-R1.doc

# Scope

This document specifies carriage of haptic media in ISO base media files.

# Normative references

The following documents are referred to in the text in such a way that some or all of their content constitutes requirements of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

ISO/IEC 14496-12:2020, *Information technology — Coding of audio-visual objects — Part 12: ISO base media file format*

ISO/IEC 23090-31:xxxx, *Information technology — Coded representation of immersive media — Part 31: Haptics Coding*

# Terms, definitions and abbreviated terms

## Terms and definitions

For the purposes of this document, the terms and definitions given in ISO/IEC 23090-5 and the following apply.

ISO and IEC maintain terminology databases for use in standardization at the following addresses:

* ISO Online browsing platform: available at <https://www.iso.org/obp>
* IEC Electropedia: available at <https://www.electropedia.org/>
  + 1. band

haptic band described in ISO/IEC 23090-31 for containing a haptic signal

* + 1. channel

haptic channel described in ISO/IEC 23090-31 for containing bands to be combined, with the resulting haptics rendered at a specific body location

* + 1. experience

haptic experience described in ISO/IEC 23090-31 containing perceptions and global information

* + 1. ISOBMFF track

track defined in ISO/IEC 14496-12

* + 1. MIHS sample

sample belonging to an MIHS track.

* + 1. MIHS stream

MIHS formatted bitstream described in ISO/IEC 23090-31

* + 1. MIHS track

ISOBMFF track having the 'hapt' handler type and an MIHSSampleEntry

* + 1. metadata

information about a haptic experience, perception, channel, or band described in ISO/IEC 23090-31

Note 1 to entry ISO/IEC 14496-12 also uses the term “metadata” but with a different meaning from ISO/IEC 23090-31, hence this clarification.

* + 1. perception

haptic perception described in ISO/IEC 23090-31 for containing channels of a specific modality such as vibration, force, pressure, etc.

* + 1. sample

sample defined in ISO/IEC 14496-12

## Abbreviated terms

|  |  |
| --- | --- |
| IEC | International Electrotechnical Commission |
| ISO | International Organization for Standardization |
| ISOBMFF | ISO base media file format (specified in ISO/IEC 14496-12) |
| MIHS | MPEG-I haptic stream |
| MPEG | Moving Pictures Expert Group |
| MPEG-I | MPEG immersive media |

# Overview

## Organization of this document

Subclause 4.2 provides the overall architecture for storage of MIHS streams in ISOBMFF.

Subclause 4.3 provides information for synchronizing MIHS streams with other media.

Subclause 4.4 provides a summary of the referenceable code points, in tabular format.

Clause 5 specifies extensions to the ISOBMFF for storage of MIHS streams.

## Haptic media

### Structure of haptic media

ISO/IEC 23090-31 describes haptic experiences composed of perceptions containing channels, which in turn contain bands.



Figure 1 – Structure of haptic media

Experiences, perceptions, channels, and bands also contain metadata, which do not depend on time.

A perception may additionally contain an effect library comprising haptic effect definitions. Bands may reference haptic effects in their containing perception’s effect library.

Bands additionally contain haptic effect definitions or references to effects in the containing perception’s effect library to be presented at specific times, possibly following external events.

### MIHS streams

ISO/IEC 23090-31 describes an MIHS format for streaming haptic media. An MIHS stream is composed of MIHS units. Each MIHS unit is composed of MIHS packets.

There are several types of MIHS units: *initialization*, *temporal*, *spatial*, and *silent*; and several types of MIHS packets: *timing*, *experience metadata*, *perception metadata*, *channel metadata*, *band metadata*, *effect library*, and *data*.

Figure 2 shows the structure of the different types of MIHS units. Boxes with dashed lines denote optional packets.



Figure – MIHS stream units and packets

An MIHS stream begins with an initialization unit. The initialization unit contains a timing packet providing a timestamp and a timescale. The initialization unit may also contain packets with metadata for the haptic experience, perceptions, channels, and bands, and an effect library packet.

One or more temporal or spatial units follow the initialization unit. Each temporal or spatial unit contains one or more data packets. The data packets contain haptic effect data.

Temporal units contain a *sync* flag indicating whether the temporal unit is a sync unit. The data packets in sync units do not depend on data in previous data packets.

Silent units may appear at any time in the MIHS stream to establish periods of haptic silence.

Initialization units may appear from time to time in the MIHS stream to update timing information or provide additional metadata or effect library information.

The metadata and effect library packets in the first initialization unit of an MIHS stream, before any temporal unit, are used as decoder configuration information for the MIHS track.

The initialization, spatial and silent units are sync units by definition.

### Temporal units and MIHS samples

The haptic data for the bands of a channel are stored in one or more temporal units. The data packets in each temporal unit form an MIHS sample.



Figure 3 – Temporal units and MIHS samples

MIHS samples created from temporal units that are sync units are sync samples. An MIHS sample may also include one or more spatial units.

A silent unit results in an MIHS sample containing a single data packet with a payload size of zero.

## Synchronization with other media

An MIHS track may use a 'sync' track reference to indicate that the MIHS track contains haptic data that are synchronized and should be presented with the media in the referenced tracks. When an MIHS track uses a 'sync' track reference, the MIHS samples in the MIHS track may be grouped by the 'mhpg' sample grouping type as specified in subclause 5.2.10 to indicate the MIHS samples that should be presented with the referenced tracks. If there is no SampleGroupDescriptionBox with grouping\_type 'mhpg', all samples in the MIHS track shall be deemed to be dependent on the referenced tracks.

## Summary of referenceable code points

### Brands

ISO/IEC 14496-12 defines the concept of brands, which may be indicated in the FileTypeBox. Brands are used in this document to indicate conformance to an encapsulation mode and a specific set of tools, as well as requirements on other specifications (e.g., ISO/IEC 14496-12).

The brands specified in this document are listed in Table 1.

Table 1 – Brands specified in this document

|  |  |  |
| --- | --- | --- |
| Brand | Clause | Informative description |
| mih1 | A.2 | Single track encapsulation of an MIHS bitstream |

### Sample entry types

The sample entry types specified in this document are listed in Table 2.

Table 2 – Sample entry types specified in this document

|  |  |  |
| --- | --- | --- |
| Sample entry type | Clause | Informative description |
| mih1 | 5.2.1 | Sample entry for an MIHS track carrying haptics data |

### Box types

In the table, the box types specified in ISO/IEC 23090-32 are in bold text with links to the corresponding clauses in the specification. Related container boxes specified in ISOBMFF are not bolded. Non-related ISOBMFF boxes are not included in the table. Mandatory boxes are, as in ISOBMFF, marked with an asterisk. Box types without a four-character code are marked with ‘‑‘ in the structure.

The box types specified in this document are listed in Table 3

Table 3 – Box types specified in this document

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Box types, structure and cross-reference (Informative) | | | | | | | | | | | | | |
| moov |  |  |  |  |  |  |  |  |  |  | \* | ISOBMFF | *container for all the metadata* |
|  | trak |  |  |  |  |  |  |  |  |  | \* | ISOBMFF | *container for an individual track or stream* |
|  |  | mdia |  |  |  |  |  |  |  |  | \* | ISOBMFF | *container for the media information in a track* |
|  |  |  | minf |  |  |  |  |  |  |  | \* | ISOBMFF | *sample information container* |
|  |  |  |  | stbl |  |  |  |  |  |  | \* | ISOBMFF | *sample table box, container for the time/space map* |
|  |  |  |  |  | stsd |  |  |  |  |  | \* | ISOBMFF | *sample descriptions (codec types, initialization, etc.)* |
|  |  |  |  |  |  | **-** |  |  |  |  |  | **5.2.1** | *haptic sample entry* |
|  |  |  |  |  |  |  | **mh1C** |  |  |  |  | **5.2.2** | *MIHS configuration box* |
|  |  |  |  |  |  |  | **hexd** |  |  |  |  | **5.2.3** | *haptic experience description box* |
|  |  |  |  |  |  |  |  | **hexh** |  |  |  | **5.2.3** | *haptic experience header box* |
|  |  |  |  |  |  |  |  | **havd** |  |  |  | **5.2.4** | *haptic avatar description box* |
|  |  |  |  |  |  |  |  | **hprd** |  |  |  | **5.2.5** | *haptic perception description box* |
|  |  |  |  |  |  |  |  |  | **hprh** |  |  | **5.2.5** | *haptic perception header box* |
|  |  |  |  |  |  |  |  |  | **hrdd** |  |  | **5.2.6** | *Haptic reference device description box* |
|  |  |  |  |  |  |  |  |  | **hchd** |  |  | **5.2.7** | *haptic channel description box* |
|  |  |  |  |  |  |  |  |  |  | **hchh** |  | **5.2.7** | *haptic channel header box* |
|  |  |  |  |  |  |  |  |  |  | **hbnd** |  | **5.2.8** | *haptic band description box* |

### Sample grouping types

The sample grouping types specified in this document are listed in Table 4.

Table 4 – Sample grouping types specified in this document

|  |  |  |
| --- | --- | --- |
| Sample entry type | Clause | Informative description |
| mhpg | 5.2.10 | Haptics presentation dependency sample group |

# Carriage of haptic coding data

## General

This clause defines the storage of haptic media utilizing the existing capabilities of the ISOBMFF and defining extensions, when necessary.

## MIHS streams and tracks

This subclause defines the boxes and data formats for incorporating all of the data from an MIHS stream into an MIHS track. From the data in an MIHS track, it shall be possible to construct a complete MIHS stream and vice-versa.

An ISO base media file may contain more than one MIHS track.

### MIHS sample entry

* + - 1. Definition

Sample Entry Type: 'mih1'  
Container: Sample description box('stsd')  
Mandatory: Yes  
Quantity: One

An MIHS sample entry shall contain an MIHSConfigurationBox and an optional HapticExperienceDescriptionBox.

* + - 1. Syntax

aligned(8) class MIHSSampleEntry() extends HapticSampleEntry('mih1') {  
 MIHSConfigurationBox configuration;  
 HapticExperienceDescriptionBox experience; // optional  
}

### MIHS configuration box

* + - 1. Definition

Box Type: 'mh1C'  
Container: MIHS sample entry ('mih1')  
Mandatory: Yes  
Quantity: One

An MIHS configuration box contains the metadata and effect library packets necessary to decode the MIHS samples that are in the MIHS track.

The packet type shall have one of the values documented in Table 4.

Table 4 – Configuration packet types

|  |  |
| --- | --- |
| Value | Type |
| 1 | Experience |
| 2 | Perception |
| 3 | Channel |
| 4 | Band |
| 6 | Effect library |

* + - 1. Syntax

aligned(8) class MIHSConfigurationBox()  
 extends FullBox('mh1C', version = 0, flags= 0) {  
 unsigned int(32) configuration\_packet\_count;  
 for (int i=0; i<configuration\_packet\_count; i++) {  
 unsigned int(6) configuration\_packet\_type;  
 unsigned int(4) configuration\_packet\_layer;  
 unsigned int(5) reserved = 0;  
 unsigned int(17) configuration\_packet\_payload\_size;  
 bit(configuration\_packet\_payload\_size\*8)  
 configuration\_packet\_payload;  
 }  
}

* + - 1. Semantics

configuration\_packet\_count indicates the number of packets included in the MIHSConfigurationRecord.

configuration\_packet\_type indicates the packet type as documented in Table 4.

configuration\_packet\_layer indicates the scalability layer of the packet. This value shall be same as the MIHSLayer in the MIHS packet.. Zero means the packet shall not be skipped. Larger values than zero mean the packet may be skipped.

configuration\_packet\_payload\_size indicates the length in bytes of the packet payload that follows.

configuration\_packet\_payload contains the packet payload formatted according to ISO/IEC 23090-31 for the packet type.

### Haptic experience description box

* + - 1. Definition

Box Type: 'hexd'  
Container: MIHS sample entry ('mih1')  
Mandatory: No  
Quantity: Zero or one

Box Type: 'hexh'  
Container: Haptic experience description box ('hexd')  
Mandatory: Yes  
Quantity: One

A HapticExperienceDescriptionBox contains descriptive information about the haptic experience associated with the MIHS track and may contain HapticAvatarDescriptionBoxes and HapticPerceptionDescriptionBoxes describing the haptic avatars and perceptions, respectively, that are part of the haptic experience.

* + - 1. Syntax

aligned(8) class HapticExperienceDescriptionHeaderBox()  
 extends FullBox('hexh', version = 0, flags= 0) {  
 utf8string edition;  
 utf8string profile;  
 unsiged int(8) level;  
 utf8string date;  
 utf8string description;  
}  
aligned(8) class HapticExperienceDescriptionBox()  
 extends Box('hexd')  
 HapticExperienceDescriptionHeaderBox header;  
 HapticAvatarDescriptionBox avatars[];  
 HapticPerceptionDescriptionBox perceptions[];  
}

* + - 1. Semantics

edition indicates the year of the edition and amendment of ISO/IEC 23090-31 that this haptic experience conforms to, in the following format: XXXX or XXXX-Y, where XXXX is the year of publication and Y is the amendment number, if any.

profile indicates the name of the profile used to generate the encoded stream according to the profile and level definition in ISO/IEC 23090-31.

level indicates the number of the level used to generate the encoded stream according to the profile and level definition in ISO/IEC 23090-31.

date indicates the human-readable creation date of the haptic experience. The date format shall follow the ISO 8601 standard.

description contains a brief description of the haptic experience.

### Haptic avatar description box

* + - 1. Definition

Box Type: 'havd'  
Container: Haptic experience description box ('hexd')  
Mandatory: No  
Quantity: Zero or more

A HapticAvatarDescriptionBox contains descriptive information about a haptic avatar that is part of the haptic experience.

The avatar type shall have one of the values documented in Table 5.

Table 5 – Avatar types

|  |  |
| --- | --- |
| Value | Type |
| 0 | Custom |
| 1 | Vibration |
| 2 | Pressure |
| 3 | Temperature |
| 4–255 | *Reserved* |

* + - 1. Syntax

aligned(8) class HapticAvatarDescriptionBox()  
 extends FullBox('havd', version = 0, flags= 0) {  
 unsigned int(8) avatar\_id;  
 unsigned int(8) level\_of\_detail;  
 unsigned int(8) avatar\_type;  
 if (type == 0) {  
 utf8string mesh\_uri;  
 }  
}

* + - 1. Semantics

avatar\_id indicates the unique ID of the avatar within the haptic experience.

level\_of\_detail indicates which level of detail should be used for the avatar if the avatar uses a mesh with several levels of detail.

avatar\_type indicates the type of haptic perception represented by the avatar as documented in Table 5.

mesh\_uri indicates the URI to access the associated 3D mesh file. The URI must follow the syntax defined in RFC3986.

### Haptic perception description box

* + - 1. Definition

Box Type: 'hprd'  
Container: Haptic experience description box ('hexd')  
Mandatory: No  
Quantity: Zero or more

Box Type: 'hprh'  
Container: Haptic perception description box ('hprd')  
Mandatory: Yes  
Quantity: One

A HapticPerceptionDescriptionBox contains descriptive information about a haptic perception that is part of the haptic experience and may contain HapticReferenceDeviceDescriptionBoxes and HapticChannelDescriptionBoxes describing the reference devices and haptic channels, respectively, that are part of the haptic perception.

The haptic modality type shall have one of the values documented in Table 6.

Table 6 – Haptic modalities

|  |  |
| --- | --- |
| Value | Modality |
| 0 | Other |
| 1 | Pressure |
| 2 | Acceleration |
| 3 | Velocity |
| 4 | Position |
| 5 | Temperature |
| 6 | Vibrotactile |
| 7 | Water |
| 8 | Wind |
| 9 | Force |
| 10 | Vibrotactile texture |
| 11 | Electrotactile |
| 12 | Stiffness |
| 13 | Friction |
| 14 | Humidity |
| 15 | User-define temporal |
| 16 | User-defined spatial |
| 17–255 | *Reserved* |

* + - 1. Syntax

aligned(8) class HapticPerceptionDescriptionHeaderBox()  
 extends FullBox('hprh', version = 0, flags= 0) {  
 unsigned int(8) perception\_id;

utf8string description;  
 unsigned int(8) perception\_modality;  
 unsigned int(8) avatar\_id;

unsigned int(16) library\_effect\_count;

unsigned int(8) unit\_exponent;

unsigned int(8) perception\_unit\_exponent;  
}

aligned(8) class HapticPerceptionDescriptionBox()  
 extends Box('hprd')  
 HapticPerceptionDescriptionHeaderBox header;  
 HapticReferenceDeviceDescriptionBox reference\_devices[];  
 HapticChannelDescriptionBox channels[];  
}

* + - 1. Semantics

perception\_id indicates the unique ID of the haptic perception.

description contains a brief description of the haptic perception.

perception\_modality indicates the type of perception as documented in Table 6.

avatar\_id indicates the unique identifier of the associated avatar body model.

libreary\_effect\_count indicates the number of effects in the haptic perception's effect library.

unit\_exponent refers to the power of 10 exponent for the SI unit of the independent variable.

perception\_unit\_exponent refers to the power of 10 exponent for the SI unit of the dependent variable.

### Haptic reference device description box

* + - 1. Definition

Box Type: 'hrdd'  
Container: Haptic perception description box ('hprd')  
Mandatory: No  
Quantity: Zero or more

A HapticReferenceDeviceDescriptionsBox contains descriptive information about a reference device that is part of the haptic perception.

The haptic actuator type shall have one of the values documented in Table 8.

Table 7 – Haptic actuator types

|  |  |
| --- | --- |
| Value | Actuator type |
| 0 | Unknown |
| 1 | Linear Resonant Actuator (LRA) |
| 2 | Voice Coil Actuator (VCA) |
| 3 | Eccentric Rotating Mass (ERM) |
| 4 | Piezoelectric actuator |
| 5–15 | *Reserved* |

* + - 1. Syntax

aligned(8) class HapticReferenceDeviceDescriptionBox()  
 extends FullBox('hrdd', version = 0, flags= 0) {  
 unsigned int(8) device\_id;  
 utf8string name;  
 unsigned int(32) body\_part\_mask;  
 unsigned int(16) optional\_field\_mask;  
 if (optional\_field\_mask & 0x0001) {  
 unsigned int(32) maximum\_frequency;  
 }  
 if (optional\_field\_mask & 0x0002) {  
 unsigned int(32) minimum\_frequency;  
 }  
 if (optional\_field\_mask & 0x0004) {  
 unsigned int(32) resonance\_frequency;  
 }  
 if (optional\_field\_mask & 0x0008) {  
 unsigned int(32) maximum\_amplitude;  
 }  
 if (optional\_field\_mask & 0x0010) {  
 unsigned int(32) impedance;  
 }  
 if (optional\_field\_mask & 0x0020) {  
 unsigned int(32) maximum\_voltage;  
 }  
 if (optional\_field\_mask & 0x0040) {  
 unsigned int(32) maximum\_current;  
 }  
 if (optional\_field\_mask & 0x0080) {  
 unsigned int(32) maximum\_displacement;  
 }  
 if (optional\_field\_mask & 0x0100) {  
 unsigned int(32) weight;  
 }  
 if (optional\_field\_mask & 0x0200) {  
 unsigned int(32) size;  
 }  
 if (optional\_field\_mask & 0x0400) {  
 unsigned int(32) custom;  
 }  
 if (optional\_field\_mask & 0x0800) {  
 unsigned int(32) type;  
 }  
}

* + - 1. Semantics

device\_id indicates the unique ID of the device within the haptic perception.

name contains the user defined name of the device.

body\_part\_mask is a binary mask specifying the location of the device or actuator on the body as defined in ISO/IEC 23090-31.

optional\_field\_mask is a binary mask defining which of the device properties are stored.

maximum\_frequency indicates the maximum frequency of the actuator in Hertz, mapping the full unsigned int(32) range to [0,10000].

minimum\_frequency indicates the minimum frequency of the actuator in Hertz, mapping the full unsigned int(32) range to [0,10000].

resonance\_frequency indicates the resonance frequency of the actuator in Hertz, mapping the full unsigned int(32) range to [0,10000].

maximum\_amplitude indicates the maximum amplitude value of the targeted device according to the perception modality, mapping the full unsigned int(32) range to [0,10000].

impedance indicates the impedance of the actuator in Ohms, mapping the full unsigned int(32) range to [0,10000].

maximum\_voltage indicates the maximum voltage of the actuator in Volts, mapping the full unsigned int(32) range to [0,10000].

maximum\_current indicates the maximum current of the actuator in Amperes, mapping the full unsigned int(32) range to [0,10000].

maximum\_displacement indicates the maximum displacement of the actuator in millimetres, mapping the full unsigned int(32) range to [0,10000].

weight indicates the weight of the device in kilograms, mapping the full unsigned int(32) range to [0,10000].

size indicates the size of the device in millimetres, mapping the full unsigned int(32) range to [0,10000].

custom contains user defined data.

type indicates the type of actuator as documented in Table 8.

### Haptic channel description box

* + - 1. Definition

Box Type: 'hchd'  
Container: Haptic perception description box ('hprd')  
Mandatory: No  
Quantity: Zero or more

Box Type: 'hchh'  
Container: Haptic channel description box ('hchd')  
Mandatory: Yes  
Quantity: One

A HapticChannelDescriptionsBox contains descriptive information about a haptic channel that is part of the haptic perception and may contain HapticBandDescriptionBoxes describing haptic bands that are part of the haptic channel.

The HapticBandDescriptionBoxes are optional; there may be fewer boxes than the quantity indicated by num\_bands.

* + - 1. Syntax

aligned(8) class HapticChannelDescriptionHeaderBox()  
 extends FullBox('hchh', version = 0, flags= 0) {  
 unsigned int(8) channel\_id;  
 utf8string description;  
 unsigned int(8) device\_id;  
 unsigned int(32) gain;  
 unsigned int(32) mixing\_weight;  
 unsigned int(8) optional\_metadata\_mask;  
 if (optional\_metadata\_mask & 0x01) {  
 unsigned int(32) body\_part\_mask;  
 }  
 if (optional\_metadata\_mask & 0x02) {  
 signed int(8) track\_resolution\_x;  
 signed int(8) track\_resolution\_y;  
 signed int(8) track\_resolution\_z;  
 unsigned int(8) num\_body\_part\_targets;  
 for (int i=0; i<num\_body\_part\_targets; i++) {  
 unsigned int(8) body\_part\_target[i];  
 }  
 unsigned(8) num\_actuator\_targets;  
 for (i=0; i<num\_actuator\_targets; i++) {  
 actuator\_target\_x[i];  
 actuator\_target\_y[i];  
 actuator\_target\_z[i];  
 }  
 } unsigned int(32) sampling\_frequency;  
 if (sampling\_frequency > 0) {  
 unsigned int(32) sample\_count;  
 }  
 if (optional\_metadata\_mask & 0x04) {  
 unsigned int(8) direction\_x;  
 unsigned int(8) direction\_y;  
 unsigned int(8) direction\_z;  
 }  
 unsigned int(16) vertex\_count;  
 for (int i=0; i<vertex\_count; i++) {  
 unsigned int(32) vertex[i];  
 }

}  
aligned(8) class HapticChannelDescriptionBox()  
 extends Box('hchd')  
 HapticChannelDescriptionHeaderBox header;  
 HapticBandDescriptionBox bands[];  
}

* + - 1. Semantics

channel\_id indicates the unique ID of the haptic channel within the haptic perception.

description contains a brief description of the haptic channel.

device\_id indicates the unique ID of the associated reference device.

gain indicates the gain associated with the channel, mapping the full unsigned int(32) range to [−10000,10000].

mixing\_weight indicates the weight of the channel when mixing different channels together, mapping the full unsigned int(32) range to [0,10000].

optional\_metadata\_mask is a binary mask defining which of the channel's optional properties are stored.

body\_part\_mask is a binary mask specifying the location of the channel's effects on the user's body according to ISO/IEC 23090-31.

track\_resolution contains the reference actuator resolution used to design the haptic experience for each spatial coordinate X, Y and Z.

body\_part\_target identifies a body part or group of body parts on the human body according to ISO/IEC 23090-31.

actuator\_target identifies the actuators targeted by the channel according to their X, Y and Z coordinates.

sampling\_frequency indicates the sampling frequency of the original encoded signal in Hertz, mapping the full unsigned int(32) range to [0,10000].

sample\_count indicates the number of samples of the original encoded signal.

direction indicates the ‘right’, 'up' and 'forward' components of the encoded signal as X, Y and Z coordinates, respectively, in the targeted body part's local coordinate system, mapping the full unsigned int(8) range to [−1,1].

vertex\_count is the number of vertices.

vertex is the index of a vertex from the avatar impacted by the channel's effects.

### Haptic band description box

* + - 1. Definition

Box Type: 'hbnd'  
Container: Haptic channel description box ('hchd')  
Mandatory: No  
Quantity: Zero or more

A HapticBandDescriptionsBox contains descriptive information about a haptic band that is part of the haptic channel.

The band type shall have one of the values documented in Table 8.

Table 8 – Band types

|  |  |
| --- | --- |
| Value | Type |
| 0 | Transient |
| 1 | Curve |
| 2 | Vectorial wave |
| 3 | Wavelet wave |
| 4–7 | *Reserved* |

The curve type shall have one of the values documented in Table 9.

Table 9 – Curve types

|  |  |
| --- | --- |
| Value | Type |
| 0 | Unknown |
| 1 | Cubic |
| 2 | Linear |
| 3 | Akima |
| 4 | Bézier |
| 5 | B-spline |
| 6–15 | *Reserved* |

* + - 1. Syntax

aligned(8) class HapticBandDescriptionBox()  
 extends FullBox('hbnd', version = 0, flags= 0) {  
 unsigned int(16) band\_id;  
 unsigned int(3) band\_type;  
 if (band\_type == 1) {  
 unsigned int(4) curve\_type;  
 }  
 if (band\_type == 3) {  
 unsigned int(8) block\_length\_log;  
 }  
 unsigned int(16) lower\_frequency\_limit;  
 unsigned int(16) upper\_frequency\_limit;  
}

* + - 1. Semantics

band\_id indicates the unique ID of the haptic band.

band\_type indicates the type of data contained in the band as documented in Table 8.

curve\_type indicates the type of interpolation function that should be used by the synthesizer as documented in Table 9.

block\_length\_log indicates the wavelet wave block length as the samples coded using the logarithmic transformation , where *block\_length* is the number of wavelet wave samples.

lower\_frequency\_limit indicates the lower frequency limit of the band in Hertz, mapping the full unsigned int(32) range to [0,10000].

upper\_frequency\_limit indicates the upper frequency limit of the band in Hertz, mapping the full unsigned int(32) range to [0,10000].

### Sample content format

* + - 1. Definition

An MIHS sample contains data packets belonging to a temporal unit. See subclause 4.2.3, for further details.

MIHS samples are externally framed and have a size supplied by that external framing; for example, by sample size ('stsz') boxes.

The silent\_flag, temporal\_flag and spatial\_flag are related as follows:

* If silent\_flag is 1, temporal\_flag and spatial\_flag shall both be zero.
* If temporal\_flag or spatial\_flag is 1, silent\_flag shall be zero.
* The temporal\_flag and spatial\_flag may both be 1.
  + - 1. Syntax

aligned(8) class MIHSSample {  
 int i = 0;  
 while (i<sample\_size) { // to end of sample  
 unsigned int(1) silent\_flag;  
 unsigned int(1) temporal\_flag;  
 unsigned int(1) spatial\_flag;  
 unsigned int(3) reserved = 0;  
 unsigned int(4) data\_packet\_layer;  
 unsigned int(5) reserved = 0;  
 unsigned int(17) data\_packet\_payload\_size;  
 bit(data\_packet\_payload\_size\*8) data\_packet\_payload;  
 i += 4 + data\_packet\_payload\_size;  
 }  
}

* + - 1. Semantics

silent\_flag indicates that the data packet does not contain haptics data, in which case data\_packet\_payload\_size shall be zero.

temporal\_flag indicates that the data packet contains temporal haptics data, in which case data\_packet\_payload\_size shall be non-zero.

spacial\_flag indicates that the data packet contains spatial haptics data, in which case data\_packet\_payload\_size shall be non-zero.

data\_packet\_layer indicates whether the data packet can be skipped for low bitrate applications. Zero means the packet must not be skipped, higher values mean the packet may be skipped.

data\_packet\_payload\_size indicates the length in bytes of the data packet payload that follows.

data\_packet\_payload contains the data packet payload formatted according to ISO/IEC 23090-31.

### Haptics presentation dependency sample group

* + - 1. Definition

Group Type: 'mhpg'  
Container: Sample group description box ('sgpd')  
Mandatory: No  
Quantity: Zero or more

The haptics presentation dependency sample group indicates that a group of samples in the MIHS track are intended to be rendered in sync and only along with samples of the corresponding referenced tracks, indicated by a 'sync' track reference box.

NOTE This sample group indicates the content author's intent and is not a renderer requirement.

The grouping\_type\_parameter is not defined for the SampleToGroupBox with grouping type 'mhpg'.

* + - 1. Syntax

aligned(8) class HapticsPresentationDepGroupEntry

extends HapticSampleGroupEntry('mhpg') {  
}

1. (Normative)  
     
   File format toolsets and brands
   1. General

This annex defines what constitutes tools, for the purposes of branding files containing haptic content. A specific brand may require some or all of the tools indicated here. A brand should be chosen that indicates the full level of support required, including any requirements on other specifications (e.g., support for aspects of ISO/IEC 14496-12).

* 1. MIHS brand

The brand 'mih1' may be present among the compatible\_brands of the FileTypeBox. File readers conforming to the 'mih1' brand shall support MIHS tracks specified in subclause 5.2.

1. (Normative)  
     
   MIME types and sub-parameters
   1. MIME types and sub-types

When MIME type is associated with haptic content as described in this document, the MIME type depends on the other media types that may also be present in the content.

* For content with audio, video, and haptics, the MIME type shall be video/mp4, for backward compatibility with existing mp4 files.
* Similarly, for files with audio and haptics, the MIME type shall be audio/mp4.
* For files with haptics only, the MIME type shall be haptics/mp4[[1]](#footnote-2).
  1. Sub-parameters for ‘codecs’ parameter
     1. General

When the ‘codecs‘ parameter of a MIME type is used, as defined in IETF RFC 6381, the sub-parameters in this annex apply when the MIME type identifies a file format of this family and the ‘codecs‘ parameter starts with a sample-entry code from this document.

* + 1. Haptic codec family

When the first element of a value is a code indicating a codec from ISO/IEC 23090-31, as documented in subclause 5.2 ('mih1'), the ‘codecs‘ parameter has the form:

codecs=mih1.oo

where ‘oo‘ is the Object Type Indication value, as defined on the MP4 Registration Authority website’s [Object Types](https://mp4ra.org/#/object_types) page[[2]](#footnote-3).

1. (Informative)  
     
   Multiple MIHS tracks and alternate groups
   1. General

A media file may contain more than one MIHS track. Examples where more than one MIHS track is needed include:

* when MIHS tracks with non-zero values for the alternate\_group in the TrackHeaderBox are used;
* when the bands for different perceptions or channels are segmented differently in time (the data packets within each track must be time-aligned using temporal units as described in subclause 4.2.3.)
  1. Criteria for alternate groups of MIHS tracks

An MIHS track among tracks with the same non-zero value for the alternate\_group in the TrackHeaderBox may be selected based on criteria such as the following:

* the contents of the (optional) BitRateBox in the track‘s MIHSSampleEntry;
* the contents of the perceptions, channels, or other data contained in the track; for example, perception modality, channel device, or channel body mask.

1. (informative)  
     
   Player handling of MIHS tracks
   1. General

Media players that support MIHS tracks should render as much of the haptic content in the tracks as possible. Depending on the capabilities of the available haptic devices, a player may map or transform haptic content; for example:

* from one body part (specified in a channel) to another;
* from one device type or set of device characteristics (specified in a track) to another;
* from one haptic modality (specified in a perception) to another.

A player may be incapable of performing certain transformations, or may decide that certain mappings are inappropriate, and may render none or some of the MIHS tracks rather than all.

1. Once ‘haptics’ is approved as a top-level media type by IETF. There is a proposal to that effect pending with the IETF: <https://datatracker.ietf.org/doc/draft-ietf-mediaman-haptics/01/>. It is expected to be approved as a Standards Track RFC in the next few months. [↑](#footnote-ref-2)
2. The registration of the Object Type for haptics has not been initiated yet; expected shortly. [↑](#footnote-ref-3)