|  |  |
| --- | --- |
| 3GPP TR 26.854 V1.0.2 (2025-02) | |
| Technical Report | |
| 3rd Generation Partnership Project;  Technical Specification Group Services and System Aspects  Study on Haptics in 5G Media Services  (Release 19) | |
|  | |
|  |  |
|  | |
| The present document has been developed within the 3rd Generation Partnership Project (3GPP TM) and may be further elaborated for the purposes of 3GPP. The present document has not been subject to any approval process by the 3GPPOrganizational Partners and shall not be implemented. This Specification is provided for future development work within 3GPPonly. The Organizational Partners accept no liability for any use of this Specification. Specifications and Reports for implementation of the 3GPP TM system should be obtained via the 3GPP Organizational Partners' Publications Offices. | |

|  |
| --- |
|  |
| ***3GPP***  Postal address  3GPP support office address  650 Route des Lucioles - Sophia Antipolis  Valbonne - FRANCE  Tel.: +33 4 92 94 42 00 Fax: +33 4 93 65 47 16  Internet  http://www.3gpp.org |
| ***Copyright Notification***  No part may be reproduced except as authorized by written permission. The copyright and the foregoing restriction extend to reproduction in all media.  © 2024, 3GPP Organizational Partners (ARIB, ATIS, CCSA, ETSI, TSDSI, TTA, TTC).  All rights reserved.  UMTS™ is a Trade Mark of ETSI registered for the benefit of its members  3GPP™ is a Trade Mark of ETSI registered for the benefit of its Members and of the 3GPP Organizational Partners LTE™ is a Trade Mark of ETSI registered for the benefit of its Members and of the 3GPP Organizational Partners  GSM® and the GSM logo are registered and owned by the GSM Association |

Contents

Foreword 7

1 Scope 9

2 References 9

3 Definitions of terms, symbols and abbreviations 12

3.1 Terms 12

3.2 Abbreviations 12

4 Background 13

4.1 Introduction 13

4.2 Definition of Haptics Media 14

4.3 Haptics media pipeline 14

4.3.1 Haptics media creation 15

4.3.2 Haptic media rendering 16

4.4 Example of Haptics media devices 16

5 Use cases and requirements 16

5.1 Introduction 16

5.2 Haptics enhanced media distribution 16

5.3 Haptic-enhanced Communication 18

5.4 Immersive Entertainment 19

5.5 Immersive multi-modal XR and metaverse 21

6 Haptics signals, media formats and device types 23

6.1 Introduction 23

6.2 Haptics signals 23

6.2.1 Introduction 23

6.2.2 Time-sampled signals 23

6.2.3 Synthetic signals 24

6.2.4 Comparison and applications 25

6.3 Haptic media formats 25

6.3.1 Introduction 25

6.3.2 PCM format 25

6.3.4 Proprietary parametric formats 25

6.3.5 Standardized and Interchangeable parametric formats 26

6.4 Haptics device types 26

6.4.1 Introduction 26

6.4.2 Haptics device type 1: Basic sensory feedback 26

6.4.3 Haptics device type 2: Sensorial texture feedback or Spatial sensory feedback 26

6.4.4 Haptics device type 3: Full-body and complex motion feedback 27

6.4.5 Haptics device type 4: Interactive and spatialised feedback 27

7 Candidate technologies 27

7.1 Codecs 27

7.1.1 MPEG Haptics Coding 27

7.1.1.1 Overview 27

7.1.1.2 MPEG Haptics Codec Architecture 28

7.1.1.3 Integration in MPEG Scene Description 29

7.1.2 IEEE 29

7.1.2.1 Overview 29

7.2 Storage format 30

7.3 Transport protocols 30

7.3.1 Haptics media delivery over DASH 30

7.3.2 Haptics top-level media type and subtypes 30

7.3.3 Haptics media RTP payload 30

7.3.4 OpenXR APIs 31

8 3GPP services 31

8.1 Haptics media integration in the Generalized Media Delivery architecture 31

8.1.1 Overview 31

8.1.2 Network functions and UE entities 32

8.1.3 Haptics media integration in XR Split rendering architecture 32

8.1.3.1 Overview 32

8.1.3.2 Network functions and UE entities 34

8.1.3.3 Split haptics media operations 34

8.2 Haptics media integration in IMS architecture. 34

8.2.1 Overview. 34

8.2.2 Network functions and UE entities 35

8.3 Haptics media integration in 3GPP services 35

8.3.1 5G Media Streaming services 35

8.3.2 Broadcast services 36

8.3.3 Real time media communication 37

8.3.3.1 WebRTC services 37

8.3.3.2 IMS based RTC services 37

8.4 Messaging Media services 38

8.3.4.1 System description 38

8.3.4.2 Media Capabilities and Profiles 39

8.3.4.2.1 Player and Decoding capabilities 39

8.3.4.2.2 MMBP content generator capabilities: 39

9 Typical haptics media traffic characteristics 40

9.1 Typical traffic characteristics summarized per use cases and formats. 40

10 QoS and QoE 41

10.1 Introduction 41

10.2 Asynchronicity between haptics and other media 41

10.3 QoS requirements for haptic media enhanced services 42

10.4 PDU Set Guidelines for haptics media 43

10.4.1 PDU Set Importance (PSI) 43

10.4.2 IP packet filtering 44

11 Identified characteristics, gaps and potential requirements 45

11.1 Identified characteristics and gaps in scenarios and services 45

11.2 Potential requirements for a haptics media codec and its support 46

12 Conclusions and proposed next steps 46

Annex A (informative): Example of a JSON HJIF file 48

Annex B (informative): Change history 62

# Foreword

This Technical Report has been produced by the 3rd Generation Partnership Project (3GPP).

The contents of the present document are subject to continuing work within the TSG and may change following formal TSG approval. Should the TSG modify the contents of the present document, it will be re-released by the TSG with an identifying change of release date and an increase in version number as follows:

Version x.y.z

where:

x the first digit:

1 presented to TSG for information;

2 presented to TSG for approval;

3 or greater indicates TSG approved document under change control.

y the second digit is incremented for all changes of substance, i.e. technical enhancements, corrections, updates, etc.

z the third digit is incremented when editorial only changes have been incorporated in the document.

In the present document, modal verbs have the following meanings:

**shall** indicates a mandatory requirement to do something

**shall not** indicates an interdiction (prohibition) to do something

The constructions "shall" and "shall not" are confined to the context of normative provisions, and do not appear in Technical Reports.

The constructions "must" and "must not" are not used as substitutes for "shall" and "shall not". Their use is avoided insofar as possible, and they are not used in a normative context except in a direct citation from an external, referenced, non-3GPP document, or so as to maintain continuity of style when extending or modifying the provisions of such a referenced document.

**should** indicates a recommendation to do something

**should not** indicates a recommendation not to do something

**may** indicates permission to do something

**need not** indicates permission not to do something

The construction "may not" is ambiguous and is not used in normative elements. The unambiguous constructions "might not" or "shall not" are used instead, depending upon the meaning intended.

**can** indicates that something is possible

**cannot** indicates that something is impossible

The constructions "can" and "cannot" are not substitutes for "may" and "need not".

**will** indicates that something is certain or expected to happen as a result of action taken by an agency the behaviour of which is outside the scope of the present document

**will not** indicates that something is certain or expected not to happen as a result of action taken by an agency the behaviour of which is outside the scope of the present document

**might** indicates a likelihood that something will happen as a result of action taken by some agency the behaviour of which is outside the scope of the present document

**might not** indicates a likelihood that something will not happen as a result of action taken by some agency the behaviour of which is outside the scope of the present document

In addition:

**is** (or any other verb in the indicative mood) indicates a statement of fact

**is not** (or any other negative verb in the indicative mood) indicates a statement of fact

The constructions "is" and "is not" do not indicate requirements.

# 1 Scope

The present technical report documents relevant use-cases, interoperability requirements, performance characteristics of haptics formats and codecs to enable haptics enhanced media streaming and communication services. This report documents the integration of haptics media in 3GPP media streaming and communication services to support the addition of a Haptics media type at the same level than audio and video in 3GPP media services.

- Clause 4 provides some background information on haptics systems as wells as a definition of haptics media and media pipeline.

- Clause 5 lists use cases illustrating the added value of haptics in 3GPP scenarios, identifies haptics characteristics and introduces interoperability considerations.

- Clause 6 presents the different types of haptics signals, the existing media formats and categorizes the different types of devices and their haptics specificities.

- Clause 7 documents the candidate technologies, in particular codecs, storage formats and transport protocols.

- Clause 8 focuses on the integration of haptics into 3GPP services and architectures including IMS, RTC, Messaging and XR device.

- Clause 9 summarizes the typical traffic characteristics of haptics media depending on the use cases and formats under consideration.

- Clause 10 addresses the QoS and QoE aspects.

- Clause 11 summarizes the identified characteristics, gaps and potential requirements

- Finally, clause 12 provides a conclusion and addresses the potential next steps.

# 2 References

The following documents contain provisions which, through reference in this text, constitute provisions of the present document.

- References are either specific (identified by date of publication, edition number, version number, etc.) or non‑specific.

- For a specific reference, subsequent revisions do not apply.

- For a non-specific reference, the latest version applies. In the case of a reference to a 3GPP document (including a GSM document), a non-specific reference implicitly refers to the latest version of that document *in the same Release as the present document*.

[1] 3GPP TR 21.905: "Vocabulary for 3GPP Specifications".

[2] 3GPP TR 22.847: "Study on supporting tactile and multi-modality communication services"

[3] 3GPP TR 26.813: "Study of Avatars in Real-Time Communication Services"

[4] 3GPP TR 22.856: "Study on Localized Mobile Metaverse Services"

[5] Haptic editor for full-body immersive experiences, *In Proceedings of the 24th ACM Symposium on Virtual Reality Software and Technology* 2018

[6] Khronos: "OpenXR 1.1.40 Specification (with all registered extensions) "

[7] ISO/IEC 23090-31: "Information technology — Coded representation of immersive media Part 31: Haptics coding"

[8] bHaptics Designer: [*https://docs.bhaptics.com/designer/*](https://docs.bhaptics.com/designer/) retrieved Sept 15th 2024.

[9] Meta Haptics Studio : [*https://www.meta.com/experiences/meta-haptics-studio/6759764157450104*/](https://www.meta.com/experiences/meta-haptics-studio/6759764157450104/) retrieved Sept 15th 2024.

[10] Interhaptics Haptic Composer: [*https://www.interhaptics.com/tech/haptic-composer*](https://www.interhaptics.com/tech/haptic-composer) retrieved Sept 15th 2024.

[11] Meta Haptics SDK for Unity: *https:*[*//developers.meta.com/horizon/documentation/unity/unity-haptic*](//interdigital.sharepoint.com///interdigital.sharepoint.com///interdigital.sharepoint.com///developers.meta.com/horizon/documentation/unity/unity-haptic)*s-sdk* retrieved Sept 15th 2024.

[12] Meta Haptics SDK for Unreal: [*https://developers.meta.com/horizon/documentation/unreal/unreal-haptics-sdk*](https://developers.meta.com/horizon/documentation/unreal/unreal-haptics-sdk)*,* retrieved Sept 15th 2024.

[13] bHaptics SDK: *<https://docs.bhaptics.com/sdk/>* *,* retrieved Sept 15th 2024.

[14] Apple Core Haptics SDK: [*https://developer.apple.com/documentation/corehaptics*](https://developer.apple.com/documentation/corehaptics)*,* retrieved Sept 15th 2024.

[15] Interhaptics Platform: <https://www.interhaptics.com/platform-partner/> *,* retrieved Sept 15th 2024.

[16] Qualcomm Snapdragon G3X gen2: [*https://docs.qualcomm.com/bundle/publicresource/87-69021-1\_REV\_A\_Snapdragon\_G3x\_Gen\_2\_Gaming\_Platform\_Product\_Brief.pdf*](https://docs.qualcomm.com/bundle/publicresource/87-69021-1_REV_A_Snapdragon_G3x_Gen_2_Gaming_Platform_Product_Brief.pdf), retrieved Sept 15th 2024.

[17] SMPTE ST 2100-1:2017 “Definition and Representation of Haptic-Tactile Essence for Broadcast Production Applications”.

[18] Apple Haptic and Audio Pattern JSON format for Haptics <https://developer.apple.com/documentation/corehaptics/representing_haptic_patterns_in_ahap_files>, retrieved Sept. 2024.

[19] Immersion: https://www.immersion.com/technology/ retrieved Sept. 15th. 2024.

[20] I. M. Vogels, " Detection of temporal delays in visual-haptic interfaces " Human Factors, vol. 46, no. 1, pp. 118–134, 2004

[21] https://dl.acm.org/doi/pdf/10.1145/3625468.3647626 ; "How do Users Experience Asynchrony between Visual and Haptic Information? " retrieved nov.11 2024.

[22] 3GPP TS 26.114: "IP Multimedia Subsystem (IMS); Multimedia telephony; Media handling and interaction".

[23] 3GPP TS 26.264: "IMS-based AR Real-Time Communication"

[24] 3GPP TS 26.565: "Split Rendering Media Service Enabler".

[25] ISO/IEC 23090-32: "Information technology — Coded representation of immersive media — Part 32: Carriage of Haptics Data"

[26] ISO/IEC 23090-37: "Information technology — Coded representation of immersive media — Part 37: Conformance and Reference Software for Carriage of Haptics Data"

[27] IANA: <https://www.iana.org/assignments/media-types/media-types.xhtml#haptics>, "Media Types" retrieved Oct 24th.2024

[28] IETF: <https://datatracker.ietf.org/doc/draft-ietf-mediaman-haptics/>, "The 'haptics' Top-level Media Type", retrieved Oct 24th, 2024.

[29] IETF: <https://datatracker.ietf.org/doc/html/draft-ietf-avtcore-rtp-haptics>, "RTP Payload for Haptics" retrieved Oct 24th, 2024.

[30] IEEE 1918.1.1-2024: "IEEE Standard for Haptic Codecs for the Tactile Internet"

[31] ISO/IEC 23090-14: "Information technology — Coded representation of immersive media Part 14: Scene description"

[32] 3GPP TS 26.119: "Media Capabilities for Augmented Reality"

[33] 3GPP TS 26.501: "5G Media Streaming (5GMS); General description and architecture"

[34] 3GPP TS 26.506: "5G Real-time Media Communication Architecture"

[35] 3GPP TS 26.113: "Real-Time Media Communication; Protocols and APIs"

[36] 3GPP TS 26.143: "Messaging Media profiles"

[37] 3GPP TS 26.244: "Transparent end-to-end packet switched streaming service (PSS); 3GPP file format (3GP) "

[38] 3GPP TR 26.841: "Study on Media Messaging"

[39] IETF More Instant Messaging Interoperability: <https://datatracker.ietf.org/doc/draft-ietf-mimi-content/> "(MIMI) message content", retrieved Feb. 03 2025.

[40] "Timestamp-independent Haptic-Visual Synchronization", <https://arxiv.org/abs/2205.03684>, retrieved Jan 14th 2025

[41] 3GPP TS 26.522: "5G Real-time Media Transport Protocol Configurations"

[42] 3GPP TS 29.514: " G System; Policy Authorization Service; Stage 3"

[43] ITU-R BS.1534-3: "Method for the subjective assessment of intermediate quality level of audio systems"

[44] MPEG: S/N 24923 "Report on the MPEG Haptics Phase 1 Performances, Jan’25, Geneva"

[45] ISO/IEC 23090-33: "Information technology — Coded representation of immersive media — Part 33: Conformance and reference software for haptics coding"

[46] 3GPP TS 23.501: "System architecture for the 5G System (5GS)"

[47] 3GPP TS 26.511: "5G Media Streaming (5GMS); Profiles, codecs and formats"

[48] 3GPP TS 26.567: "Split rendering over IMS"

# 3 Definitions of terms, symbols and abbreviations

## 3.1 Terms

For the purposes of the present document, the terms given in 3GPP TR 21.905 [1] and the following apply. A term defined in the present document takes precedence over the definition of the same term, if any, in 3GPP TR 21.905 [1].

## 3.2 Abbreviations

For the purposes of the present document, the abbreviations given in 3GPP TR 21.905 [1] and the following apply. An abbreviation defined in the present document takes precedence over the definition of the same abbreviation, if any, in 3GPP TR 21.905 [1].

AI Artificial Intelligence

AHAP Apple Haptic and Audio Pattern

AVH Audio Video Haptics

DASH Dynamic Adaptive Streaming over HTTP

ERM Eccentric Rotating Mass

HJIF Haptic JSON Interchanged Format

HMD Head Mounted Display

JSON JavaScript Object Notation

LRA Linear Resonant Actuators

MAF Media Access Function

MIHS MPEG-I Haptic Stream

MIMI More Instant Messaging Interoperability

MMBP multimedia messaging body part

MPD Media Presentation Description

MRFP Multimedia Resource Function Processor

MTSI Multimedia Telephony Service for IMSPCM Pulse-Code Modulation

PDU Protocol Data Unit

PSI PDU Set Importance

SMPTE Society of Motion Picture Television Experts

SRC Split Rendering Client

SRS Split rendering Server

RIFF Resource Interchange File Format

WAV Waveform Audio File format

XR Extended Reality

# 4 Background

## 4.1 Introduction

For the purpose of this document, two different haptic control-loops systems are defined: closed-loop-haptics and open-loop-haptics.

Closed-loop-haptics systems refer to architectures where the user received feedback is based on (local or remote) information (sensory information or simulated information) received from the system following user interactions. Closed-loop-haptics systems are illustrated in figure 4-1.

The haptic feedback is an integral part of the interaction mechanism, it provides necessary information to the user to adjust his input to the system. User inputs are continuously monitored by the system in order to constantly adjust the haptic effects sent to the haptic device. Such closed-loop-haptics system is typically (but not exclusively) used for human–machine interactions, i.e. a human controls a remote machine and the machine responds according to the input control with measures, allowing the human to adapt his behaviour. Rapid response time is therefore mandatory along with ultra reliability which may require new network architectures. Some examples of closed-loop haptic systems are surgical simulators and teleoperation robots.

Closed-loop-haptic systems are out of scope for this feasibility study.



Figure 4-1: closed-loop: remote involvement (top), and local involvement only (bottom).

Open-loop haptics systems refer to architectures where the user received haptic feedback is provided by the system (local or remote) without using sensory or simulated information, i.e., they operate based on pre-defined settings and do not adjust based on the user's interactions. They include unidirectional or bi-directional haptics media transmission with similar requirements/constraints on QoE as those applying to audio and video streaming or real-time communication in the current 3GPP architecture, taking into account human touch perceptual thresholds. Some examples of open-loop haptic systems are vibration feedback in gaming controllers and tactile feedback in wearable devices like smartwatches. The different haptic modalities (including both tactile and kinaesthetic) relevant in open loop haptic systems are part of the study. Open-loop haptics systems are illustrated in the figure 4-2 below.





Figure 4-2: open-loop: unidirectional (top ), and bidirectional (bottom).

NOTE: When a local open-loop system is in place, the Network and Remote system elements in figure 4-2 are not used.

## 4.2 Definition of Haptics Media

Haptics relates to the sense of touch. As a media, Haptics represents information describing physical feedback rendered for a specific user device and body location. This information is defined as a new media type and is described by a time-based signal (a haptic effect) or a spatial signal (a physical property of an object). The rendering is triggered by a timing information or an interaction. Different Haptic modalities are considered, targeting different human mechanoreceptors (tactile, kinesthetic, proprioception) and thermoreceptors. [7]

## 4.3 Haptics media pipeline

Haptics may be used as a media type at the same level as audio and video. Figure 4.3-1 depicts an end-to-end streaming pipeline associating audio, video and haptics in the most complete scenario of interactive 3D scenes. A simpler version of the pipeline consists in 2D AVH (Audio-Video-Haptics media) with only 2D assets and no interaction.

A computer screen shot of a blockchain

Description automatically generated

Figure 4.3-1 Audio-visual Haptic end-to-end streaming pipeline.

This figure also illustrates the different formats and some existing APIs supporting haptics media.

Figure 4.3-2 depicts an end-to-end real-time communication pipeline associating audio, video and haptics media. In a real-time communication use case, the haptic media is managed similarly to audio and video. The input is captured from input sensors, a camera for the video, a microphone for audio and dedicated haptic sensors (motion, pressure…) for haptics media. Alternatively, the signal can be loaded from the storage, typically a library of encoded effects can be stored and retrieved on the fly and sent to the receiver. The signals are then encoded and distributed using existing coded formats and protocols.

On the receiver side, the various media are decoded and rendered with the appropriate devices, loudspeaker, screen or haptics devices (e.g. vibrotactile).

A close-up of a network

Description automatically generated

Figure 4.3-2- Audio-visual Haptic real-time communication pipeline.

Figure 4.3-2 also illustrates the different formats and some existing APIs supporting haptics media.

### 4.3.1 Haptics media creation

For the creation of haptics effects and their association with A/V content, several tools have been developed and can be used. In Figure 4.3.1-1 an example of such tool, the HFX studio [5] is illustrated. It shows how to create timeline for haptics, with several channels and different haptic effects. It also shows how to design effects for particular body parts on the user.

The main principle of authoring tools is to create effects for the user and the targeted experience, and let the application manage this experience regarding the available rendering devices. For instance, in this picture a haptic effect is generated on the user torso assuming several actuators. If the receiving application does not support a haptic suit as a rendering device but may just be run on a smartphone with a single actuator, the application will select only the first vibrotactile signal from the distributed file or stream. On the opposite more complex receivers might use the full file or stream with more complex setups.

A computer screen shot of a video game

Description automatically generated

Figure 4.3.1-1- Haptic studio editing tool.

The HFX studio supports the HJIF interchanged format (or mezzanine format) defined in clause 6 of [7] for the creation, editing and distribution of haptic effects.

Other commercial tools exist, and they are generally dedicated to their associated proprietary format or platform:

* bHaptics Designer [8]
* Meta Haptics Studio [9]
* Haptic Composer - Design, Test, & Play Haptics [10]

### 4.3.2 Haptic media rendering

Several sdks have been provided to integrate haptics into standalone applications or for developing dedicated applications such as Unity and Unreal. Often provided with their proprietary format such as the meta sdk for unity [11] or unreal [12], bHaptics sdk [13] or Apple Core Haptics sdk [14].

The Interhaptics platform [15] is providing both authoring tools and Software Development Kit (SDK) for software providers and OEMs manufacturers.

APIs with the devices are often based on OpenXR [6]. However the current haptic support in OpenXR is limited, but some work is ongoing to extend capabilities.

## 4.4 Example of Haptics media devices

A large number of haptic devices exist. The simpler devices are the smartphone and wearables integrating vibrotactile devices, usually one. Chipsets support mono or stereo haptics, for instance the Snapdragon G3X gen2 [16]. When several wearables are supported by an application some form of spatialization is then possible.

# 5 Use cases and requirements

## 5.1 Introduction

The following use cases have been extracted from 3GPP TR22.847 [2], TR26.813 [3] and TR22.856 [4]. Only the use cases and requirements where Haptics can be used are considered and detailed. Similar use cases have been grouped into one.

NOTE: The representative use-cases described in this section aimed at identifying the use, requirements and gaps when considering the support of haptics media signals in the 5GMS streaming and Communication architecture. Some of the functions of the described use-cases may not be fully studied.

## 5.2 Haptics enhanced media distribution

Table 5.2-1: use case 3GPP media

|  |
| --- |
| **Use Case Name:** |
| The Haptics-enhanced media distribution is the basic scenario extending the delivery of audio-visual media with additional channel(s) for haptics.  NOTE: this use-case is not listed in TR 22.847[2], TR 22.856[4] and TR 26.813[3] |
| **Description:** |
| A service provider receives from various content providers, media contents such as movies, live sport feeds, 2D video or audio content (audio books, music,) which are enhanced with haptics. The service provider distributes the content to various devices that can render and augment the Audio-Visual experience with haptics data. The haptic effects are encapsulated in additional channel(s) and delivered synchronously with the AV content to add physical feedback in the form of haptic effects.  This haptic enhanced experience is created by the content provider or creator to maximize the user engagement. The user is enticed to this new type of media and consumes more content from his service provider. The UE does the rendering and content adaptation based on the UE haptic device(s) capabilities. In some cases, adaptation and selection into the available haptics data is necessary (either at the UE or by the service provider) ). This adaptation may include, modifying the sampling rate, limiting the number of channels, converting a force feedback signal to a vibration signal, etc.  Haptic modalities included here: motion, thermal and vibrotactile.  Haptic characteristics: 1-way, passive, single-user, low density and data-rates per channel but several channels. A channel may contain one or more haptic modalities. |
| **Categorization** |
| **Type:** distributed Audio Visual and haptics content.  **Delivery:**broadcast, http streaming  **Device:** smartphones, wearables, seats, suits. |
| **Preconditions** |
| - End-user devices that support the haptic modalities  - haptic effects associated to the media content to be delivered |
| **Characteristics** |
| Bit-rates = p to 8kbit/s per channel for compressed parametric signals, up to 64 kbit/s per channel for time sampled signals, depending on the density of the signal.  Message size = For parametric compressed signals, the burst metadata message size is, in average, up to 550 bits per channel, while data packets are up to 2500 bits. with many silent units per channel.  For time sample signal, the signal is continuous (no silent unit, or burst) with metadata message size, in average, up to 400 and the message size is, in average, up to 11600bits per channel  NOTE: these numbers are based on analysis of MPEG Haptics sequence and traces.  Number of Channels = low to high, typically 1 to 32 with non-continuous packets or continuous signal (depending on the format)  Delay = The haptic media streams is synchronized with the AV media. The perceived delay or asynchronicity between Haptic media and Video or Audio needs to be within the tolerable delay for passive experience.  Format: both signal (time samples) and parametric |
| **Feasibility** |
| Enabling technologies have reached a sufficient maturity in terms of:  - Haptics experiences creation, coding, transmission and rendering  - Availability of sensors to capture live information (accelerometer, pressure, temperature…)  - Editing tools to create Haptic effects  - Integration into Media applications  - Integration into hardware devices  Connected devices in various form factors exist that are capable of decoding and rendering haptics (smartphones, headphones, suits, cushions…). |
| **Requirements and interoperability considerations** |
| Haptic data should be able to be transmitted and rendered with various applications, environments and devices.  Support for a haptics media type is considered with one to several data channels.  RTP, ISOBMFF or DASH support for haptics media is needed.  Adaptation to the rendering capabilities should be supported.  User adaptation could be provided. |
| **Potential Standardization Status and Needs** |
| - Haptic data representation and coding (section 6 and 7)  - Haptic data carriage, transport and streaming (section 7)  - Integration in 5G architecture and services. (Section 8) |

## 5.3 Haptic-enhanced Communication

Table 5.3-1: use case Haptics-enhanced Communication

|  |
| --- |
| **Use Case Name:** |
| Haptic-enhanced Communication  Including:  TR 22.856 clause 5.9 “Synchronized predictive avatars”  TR 22.856 clause 5.11 “IMS-based 3D Avatar Communication”  TR 26.813 UC1 “Avatar Communication” |
| **Description:** |
| This use case addresses communication between people, directly or through an avatar representation.  In the simple case, haptics is a new channel added to the traditional communication means, such as i) on text messaging services to add vibrations to emojis, ii) on top of an audio call to enhance the context or just for adding effects, iii) in addition to A/V in a video call to increase the emotional impact.  In a more complex scenario, this use case is about one-to-one and multi-party communication (which may be an IMS multimedia telephony call using AR/MR/VR) with spatial audio and haptics rendering, where avatars, audio and haptics of each participant in avatar call are transmitted and spatially rendered in the direction of their geolocation. An avatar call is similar to a video call in that both are visual, interactive, provide live feedback to participants regarding their emotions, attentiveness and other social information. An avatar may interact with the environment, another avatar or object, and, through direct or indirect communication, relay haptics feedback to one or more avatars in the shared space. Similarly, avatar interaction could include throwing a ball towards another participant or using a stick to interact with environmental objects or through another object interact with thermal sinks (cold / hot sources) within the environment or scene through another object. Each participant is equipped with display devices (phones, AR glasses, etc.) with external or built-in headphones and haptics renderer.  NOTE 1: This can include help to people with disabilities.  NOTE 2: Emotional haptic is relevant in this use case.  Haptic modalities included here: mostly vibrotactile.  Haptic characteristics: Includes 2-way communication (talks, social media, phone calls…) and 1-way (alerts messages, information). Multi-users, low to high data-rates per channel, limited number of channels. |
|  |
| **Type:** 2D/3D audio-video sessions  **Delivery:** Conversational, Split  **Device:** headphones, smartphones, smartwatches |
| **Preconditions** |
| - Video and audio communication devices with associated haptic actuators for physical feedback.  - Library of created effects to be sent or/and  - Accessories to capture information and generate haptic effects. |
| **Characteristics** |
| Bit-rates = up to 5kbit/s per channel depending on density. Uplink and downlink.  Message size = For parametric compressed signals, the burst metadata message size is, on average, up to 550 bits per channel, while data packets are, in average, up to 2500 bits. with many silent units per channel.  For time sample signal, the signal is continuous (no silent unit, or burst) with metadata message size, in average, up to 400 and the message size is, in average, up to 1600 bits per channel  NOTE: these numbers are based on analysis of MPEG sequences and traces  Number of Channels = limited, between 1 and 4 with sparse packets.  Delay = The haptic media streams is synchronized with the AV media. The perceived delay or asynchronicity between Haptic media and Audio needs to be lower than 1 audio frame  Format: parametric |
| **Feasibility** |
| Enabling technologies have reached a sufficient maturity in terms of:  - Haptics experiences creation, coding, transmission and rendering. Haptic effects library can be used.  - Integration into communication applications  - Integration into hardware devices  Connected devices in various form factors exist that are capable of decoding and rendering haptics (smartwatches, mobile phones, rings…). |
| **Requirements and interoperability considerations** |
| Haptic data should be distributed with low latency and rendered with various applications, environments and devices.  RTP or ISOBMFF and DASH support for haptics media is needed.  Adaptation to the rendering capabilities should be supported.  QoS parameters needs to be developed and address asynchronicity threshold. |
| **Potential Standardization Status and Needs** |
| - Haptics data representation and coding (section 6 and 7)  - Haptics support in Scene description and avatar representation  - Haptics data carriage, transport and streaming (section 7)  - Integration in 5G architecture and services. (Section 8)  - QoS support for haptic media along with AV stream. (Section 10) |

## 5.4 Immersive Entertainment

Table 5.4-1: Use-case - Immersive Entertainment

|  |
| --- |
| **Use Case Name:** |
| Immersive entertainment (live events, sport, gaming, movies, music)  Including:  TR 22.847 5.6 “Live Event Selective Immersion”  TR 22.856 5.6 “Mobile Metaverse for Immersive Gaming and Live Shows”  TR 22.856 5.22 “Mobile Metaverse Live Concert” |
| **Description:** |
| Use cases from SA2 include complex scenario for the mobile live metaverse entertainment. This use case built upon the SA2 use cases, and considers simpler experiences covering traditional 2D and future mobile immersive applications  The considered media including here are: immersive movies, immersive TV series, social networks, gaming... The later could be multi-party or single user. In this scenario users are playing content (local or streamed) with video (2D or 3D immersive), audio (potentially spatialized) and haptics. The haptic signal is synchronized with the content to add physical feedback in the form of some haptic effects and is also generated from triggers and actions in a gaming environment. The user is using at least a smartphone with vibrotactile actuators, or more advanced devices such as gaming controllers, XR glasses and potentially multi wearable devices to spatialize haptic effects. The gaming use case is the most challenging, as low delay interactive experiences are required.  Haptic modalities included here: thermal, kinaesthetic and vibrotactile, but acceleration can be used for some events, especially gaming.  Haptic characteristics: 2-way, interactive, spatialized, multi-users, potentially significant data-rates per channel and several channels. |
| **Categorization** |
| **Type:** VR, AR, mobile  **Delivery:** broadcast,streaming, real time interactive communication  **Device:** HMD, headphone, cushion, chair, suits, gaming controller, mobile phone |
| **Preconditions** |
| - Capture sensors for feedback from the live events  - End-user devices with haptics actuators, displays and headphones.  - Interactive content creation |
| **Characteristics** |
| Bit-rates = up to 8kbit/s for high density compressed parametric signal per channel. Up to 64 kbps for high density time sample signal per channel  Message size = For parametric compressed signals, the burst metadata message size is, in average, up to 550 bits per channel, while data packets are in average up to 2500 bits. with many silent units per channel.  For time sample signal, the signal is continuous (no silent unit, or burst) with metadata message size, in average up to 400 and the message size is in average up to 1600 bits per channel.  NOTE: this numbers are based on MPEG sequences traces analysis.  Number of Channels = low to high, typically 1 to 32 with non-continuous packets or continuous signal (depending on the format). Live applications could be continuous if direct sensors information is recorded and distributed.  Delay = The haptic media is synchronized with the AV stream. The perceived delay should be lower than 1 video frame (< 25 ms), except for gaming (< 15ms)  Format: both signal (time samples) and parametric. |
| **Feasibility** |
| Enabling technologies have reached a sufficient maturity in terms of:  - Haptics experiences creation, coding, transmission and rendering  - Editing tools to create Haptic effects  - Integration into Media applications  - Integration into hardware devices  - Availability of sensors to capture live information (accelerometer, pressure, temperature…)  Connected devices in various form factors exist that are capable of decoding and rendering haptics (cushion, controllers, suits…).  On site production units can ingest sensors signals and integrate synthesizers to generate haptic effects directly in the production vans for live haptics. |
| **Requirements and interoperability considerations** |
| Haptic data should be delivered with low delay and low latency and rendered with various applications, environments and devices.  RTP, ISOBMFF or DASH support for haptics media is needed.  Adaptation to the rendering capabilities should be supported.  User adaptation could be provided. |
| **Potential Standardization Status and Needs** |
| - Haptics data representation and coding: (Section 6 and 7)  - Haptics data carriage, transport and streaming (Section 7)  - Integration in 5G architecture and services. (Section 8)  - QoS support for haptics media along with AV stream. (Section 10) |

## 5.5 Immersive multi-modal XR and metaverse

Table 5.5-1: use case Immersive multi-modal XR and metaverse

|  |
| --- |
| **Use Case Name:** |
| Immersive multi-modal XR and metaverse (VR, AR, multi-users gaming)  From:  TR 22.847 5.1 “Immersive multi-modal Virtual Reality (VR) application”  TR 26.813 UC3 “Multi-user Gaming”  TR 22.856 5.7 “AR Enabled Immersive Experience”  TR 22.856 5.12 “Virtual humans in metaverse” |
| **Description:** |
| Immersive multi-modal VR application describes the case of a human interacting with virtual entities in a remote environment such that the perception of interaction with a real physical world is achieved. Users are supposed to perceive multiple senses (vision, sound, touch) for full immersion in the virtual environment.  Virtual humans (or digital representations of humans, also referred to as 'avatars' in this use case) are simulations of human beings on computers. There is a wide range of applications using avatars, such as games, film and TV productions, financial industry (smart adviser), telecommunications (avatars), etc.  In the coming era, the technology of virtual humans is one of foundations of mobile metaverse service. A virtual human can be a digital representation of a natural person in a mobile metaverse service, which is driven by the natural person, or a virtual human also can be a digital representation of a digital assistant driven by AI model.  Mobile metaverse services offer an important opportunity for socialization and entertainment, where user experience of the virtual world and the real world combine. This use case focuses on the scenario of a natural person's digital embodiment in a metaverse as a location agnostic service experience. A virtual human is customized according to a user's personal characteristics and shape preferences. Users wear motion capture devices, vibrating backpacks, haptic gloves, VR glasses to drive the virtual human in a meta-universe space for semi-open exploration. The devices mentioned above are 5G UEs, which need to collaborate with each other to complete the actions of user and get real-time feedback.  NOTE: This scenario is the most complex considered in this study. While a metaverse platform might require new architectures, this scenario is considered here for the purpose of evaluating the coding and representation formats for haptics and their capability for extensions in current architecture to support future complex scenario.  Haptic modalities included here: force, motion, thermal and vibrotactile.  Haptic characteristics: Bi-directional, interactive, spatialized, multi-users, low latency, high data-rates. |
| **Categorization** |
| **Type:** VR /AR/XR  **Delivery:** streaming, Split, conversational  **Device:** HMD, glove, suit, AR/VR controller, motion platform |
| **Preconditions** |
| - Capture sensors for feedback from the metaverse scene and other users actions  - End-user devices with haptic actuators, sensors, force, HMDs and headphones.  - Interactive content creation  - Bi-directional streaming  - VR/AR/metaverse and supporting architecture. |
| **Characteristics** |
| Bit-rates = up to 8 kbit/s per channel for compressed parametric signals.  Message size = For parametric compressed signals, the burst metadata message size is, in average, up to 550 bits per channel, while data packets are in average, up to 2500 bits. with many silent units per channel.  For time sample signal, the signal is continuous (no silent unit, or burst) with metadata message size, in average, up to 400 and the message size is, in average, up to 1600 bits per channel  NOTE: this numbers are based on MPEG sequences traces analysis  Number of Channels = high, typically 6 to 32 with non-continuous packets.  Delay = The haptic media streams is synchronized with the AV media for rendering. The perceived delay or asynchronicity between Haptic media and Video or Audio needs to be lower than 1 audio frame?  Format: mostly parametric (esp. virtual scenes) |
| **Feasibility** |
| Enabling technologies have reached a sufficient maturity in terms of:  - Haptics experiences creation, coding, transmission and rendering  - Editing tools to create Haptic effects  - Integration into CR/Metaverse environment is considered, but still in development.  - Integration into hardware devices  Connected devices in various form factors exist that are capable of decoding and rendering haptics (HMD, gloves, controllers, suits…). |
| **Requirements and interoperability considerations** |
| Haptic data should be able to be transmitted with low delay and low latency and rendered with various applications, environments and devices.  RTP, ISOBMFF or DASH support for haptics media is needed  Adaptation to the rendering capabilities should be supported.  User adaptation could be provided. |
| **Potential Standardization Status and Needs** |
| - Haptics data representation and coding (section 6 and 7)  - Haptic data integration into scenes and with avatar representations  - Haptics data carriage, transport and streaming (section 7)  - Integration in 3GPP architecture and services. (Section 8)  - QoS support for haptics media along with AV stream. (Section 10) |

# 6 Haptics signals, media formats and device types

## 6.1 Introduction

Haptics refers to the sense of touch, and encompasses the generation, manipulation, and perception of tactile sensations, forces, and motions. In the media domain it relates to the capture and rendering of physical information about objects and environments due to interactions with the user. The capture is generally performed with sensors and the rendering with hardware actuators allowing to render different modalities.

In general, three types of modalities are considered, tactile (vibration, temperature, pression), kinaesthetic (force) and proprioception (motion, acceleration). Those modalities are rendered in the human body through different mechanoreceptors. The density and properties of those receptors are different depending on the modality and the body parts. In addition, each individual has different sensibility, which can be managed through users’ profiles and adaptation.

Finally, the rendering is performed with specific hardware actuators embedded into user devices. The different Haptic modalities are rendered with different actuators. Typically, tactile sensations can be rendered with vibrotactile (e.g. linear resonant actuators (LRA), eccentric rotating mass (ERM), temperature, wind (fan), force devices.

Haptics signals correspond to the raw haptics data at the output of the sensors or authoring tools, described in clause 6.2.

Haptics media formats are the formats representing the haptics signals once processed for use by a production framework, as described in section 6.3. For example, the raw synthetic signal described in 6.2.3 can be transformed in any of the haptics media parametric representation formats described in 6.3.4.

## 6.2 Haptics signals

### 6.2.1 Introduction

Haptic signals are essential in capturing and transmitting tactile sensations in a variety of applications, ranging from simple vibrations in mobile devices to complex force feedback in virtual reality systems. Two types of signals exist, time-sampled signals typically captured by sensors, and synthetic signals of haptic effects, typically created from authoring tools.

### 6.2.2 Time-sampled signals

In Figure 6.2-1, two examples of time-sampled signals are provided. The left side shows the samples of 3D angular orientations and linear accelerations of a horse captured with an accelerometer positioned on the horseman. The right side illustrates the haptic signal of a texture (here a carpet) captured from the sound of a pen moving on the surface of the carpet in one direction.

Those haptics time sampled signals are generally used for signals captured from sensors.

A graph of red lines

Description automatically generatedA red line graph with text

Description automatically generated

Figure 6.2-1 - Example of time sampled signals (left: Horse riding 3D angular orientations and linear accelerations captured with an accelerometer, right: texture captured from sound on one direction).

Generally, the sampling frequency is set between 2kHz and 16kHz (8kHz in Figure 6.2-1) with the haptic perceptual bandwidth being around [5Hz-1kHz], with some variations depending on the haptic modality, rendering device and human haptic receptor. 5Hz and below is particularly significant for force feedback. This leads to a raw data-rate average of 128 kbps per channel.

### 6.2.3 Synthetic signals

Synthetic signals of haptic effects are mostly generated with design tools and thus correspond to synthetic data. In Figure 6.2-2, two examples of synthetic signals are provided, on the left a heartbeat synthetic signal (thus the regularity of the signal), and on the right a synthetic rain effect (quasi random design) with 4 channels corresponding to 4 different target locations on the user.

A graph with red lines

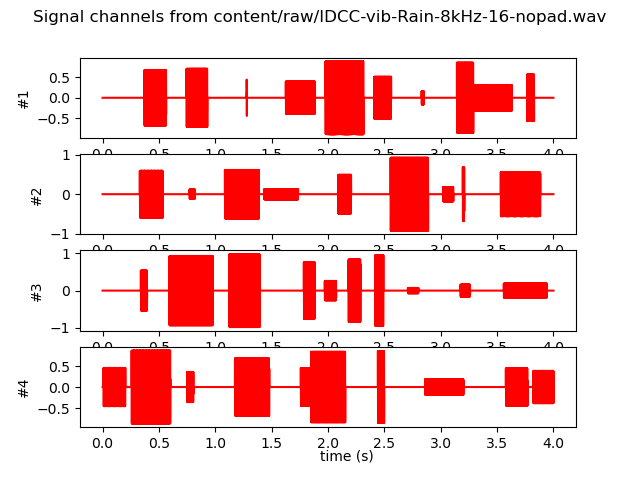
Description automatically generated

Figure 6.2-2 - Example of synthetic rendered haptic signal (left: heart beat synthetic signal, right: synthetic rain effect with 4 channels for 4 different locations on the user).

The synthetic aspect of those signals compared to time-sampled signals captured from sensor is obvious. In general, a synthetic signal is used to synthesize a short duration effect and is used only at some events or at dedicated point in time during a gaming or movie experience, by opposition to time-sampled signals that represent a continuous signal sampled at a constant frequency

Synthetic signals need a synthesizer to be converted to a timed signal compatible with the actuators used. The raw bit-rate dependents on the number of effects and their distribution in time.

### 6.2.4 Comparison and applications

The choice between time-sampled signals and synthetic signals depends on the application and the desired level of control. Time-sampled signals are more suitable for applications requiring high fidelity and real-time capture of physical interactions, such as in medical simulations, playback of recorded signals (motion) or fine-grained tactile feedback systems. Synthetic signals, on the other hand, are ideal for applications where flexibility and ease of use are paramount, such as in mobile notifications, communication, movies or gaming, where the haptic feedback needs to be easily adjustable and consistent across a variety of devices. Synthetic signals are also well suited for XR applications, where haptics effect may be attached to events or objects.

## 6.3 Haptic media formats

### 6.3.1 Introduction

In this clause, haptics media formats are the formats representing the raw data (or haptics signals described in 6.2) and used at the production side (e.g. for encoding, packaging and delivery). Most common media formats for haptics are proprietary. However standardised interchangeable media formats (or mezzanine formats) have now been defined.

### 6.3.2 PCM format

PCM or Pulse Code Modulation is a digital format that represents analog signals in digital format as a series of samples at a regular sampling frequency. It has been defined for audio but can be used to store haptic signals from sensors.

Various file formats support PCM such as the audio WAV format. ISO/IEC MPEG Haptic [7] is using WAV as one of the input file format for time-sampled haptic signals. This type of file format uses different bit depth and data types to store the signal usually interpreted on a [-1 ; 1] range. For this reason, additional headers and metadata is required to specify the actual value ranges, gain or other characteristics of the signal essential for rendering. There are currently no standard defining this type of metadata and therefore the interpretation of the signal is often application dependent.

This PCM format is suitable for storing or distributing sensors data and its implementability has been demonstrated for audio signals. The drawback of the PCM format is its overhead as it is sampled continuously at constant frequency in time as opposed to parametric formats that are much sparser.

### 6.3.4 Proprietary parametric formats

Descriptive haptic is represented through parametric primitives. It can be described by an amplitude, a frequency, a duration, and a waveform shape such as sinusoid, or square. Two types of parametric effects are usually defined: transient and continuous. A transient effect is a short pulse with an amplitude, a position in time or space, and a reference frequency. A continuous effect is represented by a frequency and an amplitude information modulated on the timeline.

Existing proprietary descriptive formats use JSON or XML to describe haptic effects. Two of those proprietary formats are commercially deployed: AHAP from Apple [18] and IVS [19] from immersion corp.

AHAP has been mainly developed for vibrotactile signals and implementation into iOS Core Haptic framework.

IVS was targeting media applications (movies, gaming) and communication (smartphone) using mainly vibrotactile signals described in XML format. A binary version is available for networks transmission.

HAPT is a vendor-specific haptic coding format based on the RIFF format. It is designed to enable efficient coding of a device-specific haptic effects. HAPT is integrated in Immersion’s SDK [15].

These formats may further be encoded and compressed using haptic codecs. The ISO/IEC MPEG Haptic [7] codec supports both the AHAP and IVS formats, while the IEEE codec [30] only supports PCM input signals.

### 6.3.5 Standardized and Interchangeable parametric formats

SMPTE defined the Haptic-Tactile Essence for Broadcast Production Applications [17]. However, it was mainly dedicated to position, orientation, velocity and acceleration.

MPEG HJIF: MPEG defined in section 6 of specification [7] a standardized and interchanged parametric JSON based format (or mezzanine format) for describing haptics effects called HJIF. An example of an HJIF file is provided in Annex A.

This format is compatible with the proprietary AHAP format. Transcoding from and to the other is possible. Similarly, the IVS format can be transcoded from XML to JSON, for most of its functionalities, since similar parametric patterns/functions are used. Transcoding of the primitives is lossless.

In addition, MPEG HJIF can integrate a binary encoded version of a PCM signal.

The main advantage of MPEG HJIF is its flexibility to support various haptic modalities, various parametric formats and its flexibility for heterogeneous end-user devices and experiences. Thanks to its JSON formatting, it is also easy to edit and modify the file.

MPEG HJIF can further be encoded and compressed using MPEG haptic codecs, in a binary format for networks transmission.

## 6.4 Haptics device types

### 6.4.1 Introduction

Haptics devices may be categorized into types defining their primary capability. Each category reflects the device's intended application and the type of haptic experience it provides. Distinction is done between input (sensors) and playback (actuators or rendering devices) functionalities of devices. These two functions can be collocated or be on different devices.

### 6.4.2 Haptics device type 1: Basic sensory feedback

Devices in this category are designed to provide passive, non-spatialised sensory feedback, in applications such as simple mobile game and entertainment, alerting, or communication.

They include actuators integrated into smartphone, smartwatch, wearables, finger UI, headphone, HMD. They usually have a low number of actuators and usually vibrotactile actuators.

In this category it is anticipated that the haptic signals are synthetic haptic effects created by a designer.

In addition, some simple sensors to capture sensory information mostly related to the environment, such as temperature, pressure, sound, humidity can be available into smartphones and wearables. Generally, it consists of a single sensor with time sampled signals recorded or sent.

### 6.4.3 Haptics device type 2: Sensorial texture feedback or Spatial sensory feedback

Devices in this category are designed to deliver highly detailed and precise haptics (e.g., tactile, kinaesthetic, proprioception) sensations. These devices simulate the feel of textures, pressure, and other subtle interactions, enabling users to experience intricate touch-based feedback. They are commonly used in applications where detailed touch interaction is crucial, such as virtual reality try-on, objects manipulation, training/education.

Fine-tuned sensory feedback devices include Haptic gloves Haptic touchpads, touch-sensitive surfaces and screens. Haptic modalities here are not restricted to vibrotactile, kinaesthetic and mid-air feedback is also considered.

For the capture, sensors in this category capture sensory information related to objects surfaces physical properties. It consists of more complex systems combining tracking and measurement to get the localization and physics. The tracking measures the location and speed of the sensors, the measurement unit captures the pressure, force feedback and motion to infer smoothness, roughness, relief and other physical properties of a 2D surface.

### 6.4.4 Haptics device type 3: Full-body and complex motion feedback

Devices in this category provide immersive, whole-body sensations, but mostly as passive feedback to the user. They are ideal for enhancing immersion in virtual environments or simulation scenarios by delivering comprehensive, multi-sensory (e.g., vibrotactile, thermal, pneumatic, electrotactile) feedback.

These rendering devices use a network of actuators to simulate a wide range of physical experiences, such as vibrations, impacts, and movements across the body. Haptics suits and furniture such as racing game seats, motion platforms and simulators are example of Full-Body and complex feedback devices. Multiple modalities are also provided ranging from vibrations, wind, water spray, heat or motion.

For the capture, sensors in this category capture information related to position, speed and forces. Typical GPS units and accelerometers can be used (such as those integrated into smartphones), as well as pressure sensors. The information is related to position in 3D space with global tensor fields and targeted force vectors.

### 6.4.5 Haptics device type 4: Interactive and spatialised feedback

Devices in this category are focused on providing targeted, spatialised interactive haptics (e.g., tactile, kinaesthetic, proprioception, thermal) feedback. These devices simulate specific actions or interactions, such as button presses or in-game effects, through vibrations or adaptive triggers. This category includes haptic controllers such as game console, phones, HMD or VR controllers offering localized feedback through vibration motors and adaptive triggers to simulate various in-game sensations and interactions. Usually, bi-directional interaction is considered.

Sensors for capturing haptics signals in this category are mostly related to pressure sensors to get a position and force related to an action. Here typically input devices such as phone, HMD or VR controllers are used for feedback but also their sensors are used to capture interactions, motion and forces.

# 7 Candidate technologies

## 7.1 Codecs

### 7.1.1 MPEG Haptics Coding

#### 7.1.1.1 Overview

JTC 1/SC 29/WG 7 (MPEG-3DGH) has completed the development of the MPEG Haptics Coding standard at the April 2023 meeting and is pending publication as ISO/IEC 23090-31 [7].

In addition to a codec and bitstream format, this MPEG Haptic coding standard supports multiple haptic modalities and specifies a human readable JSON coded representation format (HJIF) for descriptive and quantized media formats. Haptic primitive from AHAP and IVS parametric signals can be losslessly converted to and from HJIF. The HJIF data model is a flexible hierarchical structure that can describe one or more channels, allowing for mono, stereo or multi-channel haptics media experiences, targeting one or more actuators or devices. Application areas especially targeted for the use of MPEG Haptics coding include media streaming and broadcast, immersive applications and XR services, real time communications, as well as tactile communication, the latter being out of scope of this study.

During the standardization process, MPEG also worked on a reference and conformance software available in ISO/IEC 23090-33 [45] that was publicly released.

An in-depth evaluation of the performances of this reference codec was conducted with both objective metrics (PSNR) and subjective tests MUlti Stimulus test with Hidden Reference and Anchor (MUSHRA) as defined in ITU-R recommendation BS.1534-3 [43] for different target bitrates and different input test streams.

Three representative sets of test streams were provided by different companies and corresponded to market needs: two sets for vibrotactile signals (short effects and long effects) and one set for kinaesthetic signals (including force signals, acceleration, or movement). For a total of 43 test streams.

The objectives performances reported by MPEG in [44] are given in Figure 7.1.1.1-1. On the left the PSNR is given for three configurations of the MPEG encoder CRM3.2 considering PCM input signals (C2V: vectorial encoding, C2W: wavelet encoding, C2VWR: hybrid encoding). On the right the histogram of the bit-rate is depicted for parametric transcoding of the .ivs and .ahap parametric input streams.

A graph of different colored lines

Description automatically generated A graph of numbers and a number of data

Description automatically generated with medium confidence

Encoded PCM signals

Figure 7.1.1.1-1Objective performances (left: PCM signals, right: parametric signals).

The transcoding of parametric input content is usually around 1-4 kbps (It consists in transcoding without loss the input parametric file to the MPEG parametric format which is then binarized).

The results on PCM input data show that signals lossy encoded with the 2 kbps target bitrate present perceptible distortions with an average PSNR of 24db. Signals encoded with a 8 kbps target bitrates show some distortions, but not annoying, with an average PSNR of 39.17db. Finally signals encoded with a 16kpbs bitrate have no perceptible distortion with an average PSNR of 45.62db

Thus, 8kbps is considered as a reasonable average bitrate for MPEG-Haptic encoding. Considering an average input bitrate being 128kbp (8kHz sampling for 16bits samples), it leads to a compression ratio of 16 per channel.

NOTE: For PCM, both lossy and near lossless compression can be achieved; however the use case of section 5 do not require lossless or near lossless compression.

Subjective performances, performed by three independent laboratories and two reference haptic devices, confirmed the above results. The MUSHRA score was higher than 94 (maximum is 100) for bitrates of 8kbps and higher.

#### 7.1.1.2 MPEG Haptics Codec Architecture

Figure 7.1.1.2-1 represents the MPEG haptics codec architecture. Media formats supported by the MPEG Haptics representation and coding includes AHAP, HJIF, IVS parametric media formats, as well as the WAV time sampled media format.

The synthesizer is not defined by the standard and is illustrated in this figure to highlight how the MPEG Haptics codec can be integrated with a renderer.

Diagram

Description automatically generated

Fig 7.1.1.2-1 – MPEG Haptics codec architecture.

#### 7.1.1.3 Integration in MPEG Scene Description

MPEG Scene Description [31] is used in IBACS [23] and in [SR-MSE] via MeCAR [32]. To address use-cases in section 5.4 and 5.5 and these 3GPP services, Haptics media may be integrated with a scene description.

MPEG Scene Description provides extensions for the support of haptics media defined in [7]. Two extensions have been defined *MPEG\_haptic* and *MPEG\_haptic\_material*. The first one associates a haptic media stream on a set of nodes and defines every haptic media object. The second describes the texture-based haptic media data when used (as a 2D map associated to an object).

The trigger/action mechanism of MPEG Scene Description can be used to associate and play haptic media effects on objects and/or avatars with specified location and playback type. In particular, the relation between a node in the scene and haptic media data is established through haptic actions. Interactive haptic media feedback is produced by defining behaviors with triggers (e.g collisions, proximity, etc.) and haptic actions. For each node in a haptic action, the associated Haptic media data is defined either through a reference to an element or through a *MPEG\_haptic\_material* attached to a mesh of the node. When a haptic action is triggered, the associated haptic media data is rendered according to the properties specified in the action.

### 7.1.2 IEEE

#### 7.1.2.1 Overview

The IEEE Standard for Haptic Codecs for the Tactile Internet was developed in 2016 and approved in 2024 [30]. This standard specifies 3 different codecs (no-delay kinaesthetic codec, delay-robust kinaesthetic codec, and a tactile codec) supporting only a time-sampled media format (PCM) and no parametric media formats.

Some performances of the IEEE Haptic Codecs were documented in [2] in relation to closed loop haptic systems (with a reliability of 99,999%), which are out of scope of this study. Performance of the IEEE Haptic Codecs for open loop haptic system are for further study.

Further, with no support for parametric media formats, these codecs could not be used as mezzanine format nor interoperable formats with parametric haptics media source content widely used in use cases of section 5.

## 7.2 Storage format

A specification for the storage and delivery signaling for haptics media as defined in [7] is currently under development by MPEG and has reached FDIS stage. ISO/IEC 23090-32 [25] defines how a haptics MIHS bitstream can be encapsulated in ISOBMFF media containers. The specification supports single and multi-track encapsulation, where different channels or bands of the haptics media can be stored in separate tracks to enable selective access.

MPEG is also working on a reference software implementation for the storage and delivery aspects defined in ISO/IEC 23090-32 as part of a new specification (ISO/IEC 23090-37 [26]) that has recently been initiated and is expected to be completed by the end of 2025.

## 7.3 Transport protocols

### 7.3.1 Haptics media delivery over DASH

ISO/IEC 23090-32 also specifies how haptics media is signaled in an MPEG-DASH manifest (MPD) for adaptive media delivery and defines descriptors to signal information pertaining to the haptics experience to allow a streaming client to select the parts of the haptics media to stream based on playback timeline, network conditions, and/or user interaction.

### 7.3.2 Haptics top-level media type and subtypes

The 'haptics' media type has been documented and registered by IANA as a top-level media type, along with 'audio', 'video', 'application', and others. [27].

Under this top-level media type, the following haptics subtypes are currently registered:

* ivs, haptics/ivs
* hjif, haptics/hjif
* hmpg, haptics/hmpg

The justification for the 'haptics' top-level media type is found in [28].

### 7.3.3 Haptics media RTP payload

An RTP payload format for haptics media is under development in IETF [29] and has reached the working group last call. The draft describes how the haptic data, in MIHS units defined in [7] can be transmitted using the RTP protocol. This RTP payload format enables the transport of the “hmpg” media subtype defined in [28].

Some of the characteristics of the payload formats include:

- The 4 types of MIHS units are indicated in the payload header: initialization, time dependent (temporal), time independent (spatial), and silent units.

- A MIHS unit can be marked as independent or dependent. An independent unit resets the previous haptics effect and correspond to a “sync” MIHS unit as defined in [7].

- 3 payload structures are defined and can be used within a same stream: a single unit payload structure for a single MIHS unit per packet, a fragmented unit payload structure for MIHS units which are too large to be transmitted in a single packet, and an aggregation packet payload structure to transport multiple MIHS units in a single RTP packet.

- The aggregation packet payload structure can be used to transport multiple MISH units that correspond to the same timestamp, single-time aggregation packet (STAP), or to transport multiple MISH units that correspond to different timestamps, multi-time aggregation packet (MTAP).

- In case of congestion control, the draft recommends prioritizing initialization units, to treat silent units as less important, and to use the MIHS unit layer information present in the RTP payload header to prioritize packets.

- The draft also describes SDP considerations, to define additional optional parameters that can be used in the SDP exchange.

### 7.3.4 OpenXR APIs

The OpenXR API [6] standardizes the use of cross-platform XR device capabilities, including haptic devices. Current support mainly addresses XR and game controllers handled by the user hands and most commercial controllers are supported. Functions such as trigger, click, touch, squeeze, grip are input actions with a potential haptic output specified by an *hapticsAction* and *hapticPath*. The *hapticsAction* specifies the type of haptic feedback (e.g. vibrations) and *hapticPath* specifies where the effect is applied (e.g. left hand). Haptic feedback is sent to a device using the *ApplyHapticFeedback* and *StopHapticFeedback* functions.

The only haptics type supported by the OpenXR API v 1.1.43 is *XrHapticVibration,* for haptic vibrations (vibration amplitude, duration, frequency). Some proprietary extensions also provide support for vibrations described by a haptic amplitude envelope or PCM signal (*XrHapticAmplitudeEnvelopeVibrationFB*, *XrHapticPcmVibrationFB*).

In 2022, the HIF (Haptic Industry Forum) submitted a high-level proposal for advanced haptics APIs to the OpenXR consortium. The advanced haptics APIs are designed to extend the OpenXR API to other haptics modalities and process several haptics standards among which the MPEG HJIF, IEEE P2861.3 standards as well as other proprietary formats (AHAP, IVS).

# 8 3GPP services

## 8.1 Haptics media integration in the Generalized Media Delivery architecture

### 8.1.1 Overview

The Haptics media functionalities in the generalized 5G Media Delivery architecture defined in [33], and [34] are shown in figure 8.1.1-1. Existing interfaces and functions are reused to support Haptic media type and Haptic media transport in 5G services (e.g. 5GMS, RTC) to address the use-cases of section 5.

The architecture is not modified, the haptics media functions are added in the below figures for illustrative purposes.



Figure 8.1.1.-1: Haptics media in the Generalized Media Delivery architecture

### 8.1.2 Network functions and UE entities

As Haptics media can be integrated in 3GPP services as a companion media type to Video and Audio, the media related definitions described in [34] and [33] are extended to describe the haptics media related functions highlighted in the figure 8.1.1-1 as follow:

- **Media AF**: An Application Function as defined in clause 6.2.10 of [46] dedicated to Media Delivery, including haptics media delivery.

- **Media AS:** An Application Server dedicated to Media Delivery including haptics media delivery. The depicted Haptics media engine support access, coding and delivery function, of user plane haptic media data.

- **Media Client:** A UE internal function dedicated to Media Delivery comprising:

- **Media Session Handler:** An entity on the UE that communicates with the Media AF in order to establish, control and support the delivery of a media session. The media session includes delivery of Haptics media.

- **Media Access Function:** An entity on the UE that communicates with the Media AS in order to access and deliver media content, including haptics media. The media access function for example may be further sub-divided into content delivery protocols, codecs, media types and metadata representation, including those related to Haptic media data.

- **Media-aware Application:** An application entity on the UE that makes use of 3GPP-defined APIs to invoke the Media Session Handler and/or the Media Access Function in order to support Media Delivery, including haptics media delivery.

### 8.1.3 Haptics media integration in XR Split rendering architecture

#### 8.1.3.1 Overview

As expressed in [24], typical use cases for split rendering includes immersive gaming and immersive communication. These use cases, haptic enabled, are also described in section 5 of this document.

Figure 8.1.3.1-1 is based on figure 6.1.4-1 of [24] and illustrates the haptics media functions as part of the SRS, providing support for haptic media coding and delivery, similarly to audio and video media.

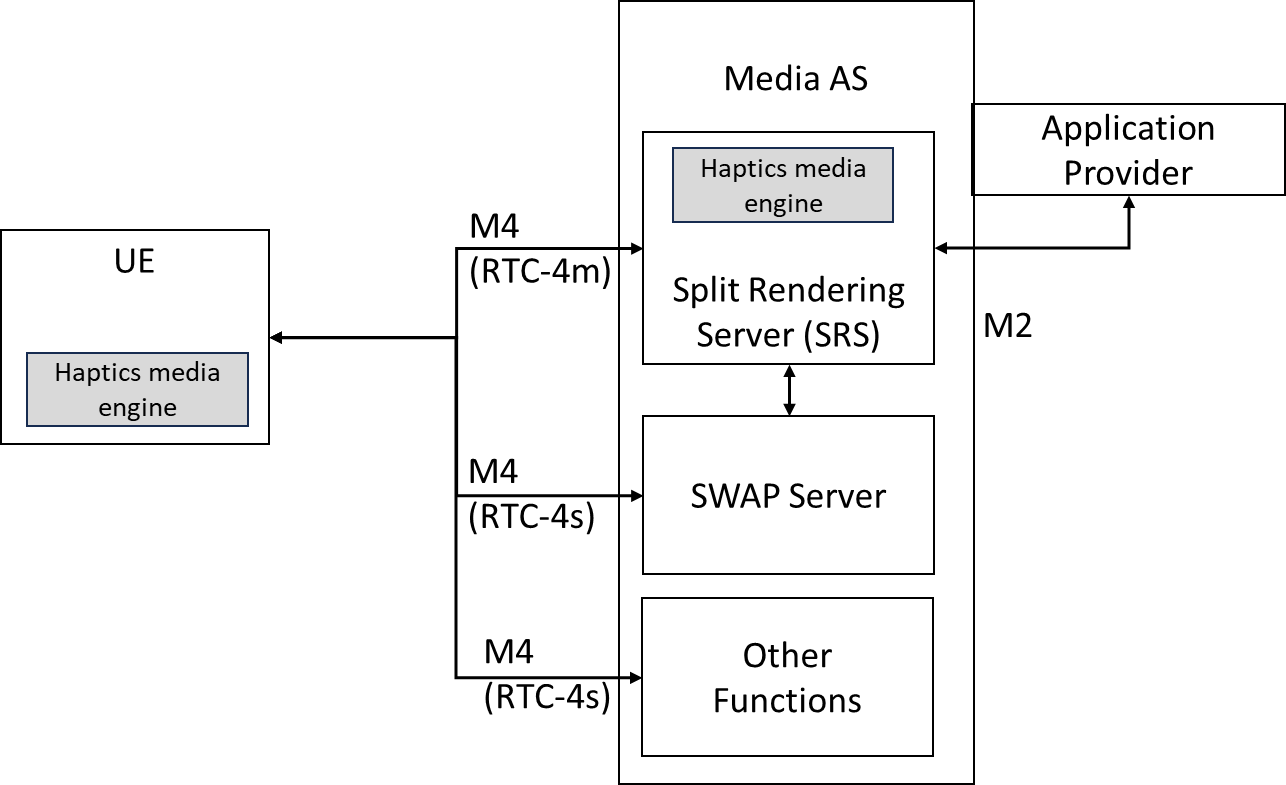


Figure 8.1.3.1-1: Haptic media functions in the User Plane Architecture for Split management architecture

On the UE, Figure 8.1.3.1-2, based on Figure 5.1.2-1[24] illustrates the haptics media entities in the XR baseline client.

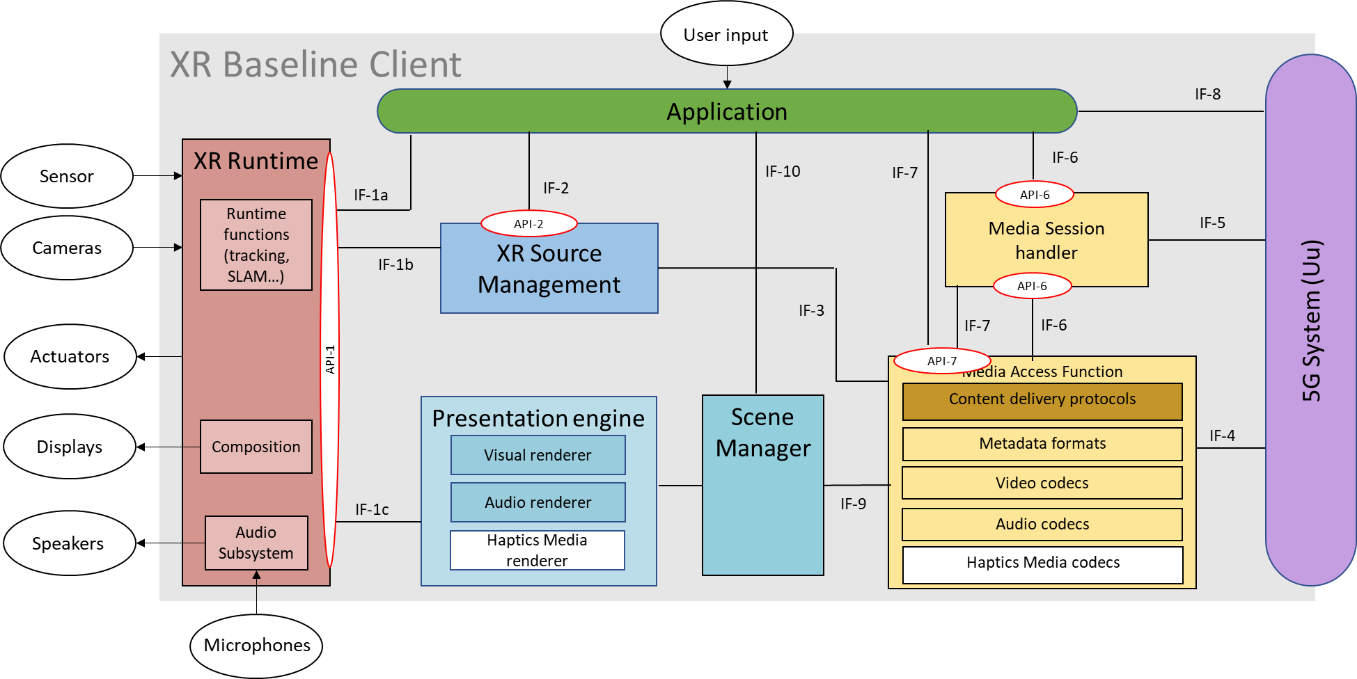


Figure 8.1.3.1-2 Haptic media entities in the XR Baseline Client architecture

#### 8.1.3.2 Network functions and UE entities

The haptics media related network functions in the XR end to end split rendering architecture are similar to the one described in section 8.1.2. The haptics media engines in the Media AS and in the UE need to make sure that the format of the haptics media sent by the Media AS is understandable to the UE. Negotiating the split of haptic media capabilities is for further studies. Use of Non-3GPP codecs are out of scope of this document.

The haptics media entities on the XR baseline Client consist of:

- the haptic media codec, handling and decompressing a compressed haptics media bitstream is illustrated in the MAF function of the SRC along with AV codecs.

- the haptic renderer, handling the rendering of haptics effects using the targeted actuators is illustrated in the presentation engine function of the SRC.

#### 8.1.3.3 Split haptics media operations

When a UE intends to offload part of its haptics media processing to the SRS:

* The SRC and the SRS negotiate the desired haptics media capabilities (or profile) on the M4 interface using a SWAP (Simple WebRTC Application Protocol) message or a data channel message.
* The SRS processes and renders the haptics media content and may use pose or interaction information to spatialised the rendered haptics media content in correlation with other rendered media stream (scene, video, objects, audio).
* The SRS transmits to the UE the resulting haptics media streams.
* The UE decodes and renders the haptics media stream.

## 8.2 Haptics media integration in IMS architecture.

### 8.2.1 Overview.

To support the use-cases in section 5 for IMS based services (MTSI, and IBACS), the existing IMS architecture is not modified. Existing functions and API are reused to enable haptic enhanced IMS based real time communication.

Figure 8.2.1-1, illustrates the haptic media functionalities in the MTSI client defined in figure 4.2 of [22] which is similar to figure 4.2.1 of the AR-MTSI client of [23].



Figure 8.2.1-1: Functional components of an MTSI client with haptic media support.

### 8.2.2 Network functions and UE entities

For MTSI, [22] the network functions such as the MRFP and MGW may handle haptics media similarly to other media for supplemental services.

Referring to Figure 4.3.1 of [23], Generalized IMS DC Architecture to support AR communication, the AR AS, the IMS AD and the MF/MRD handles haptics media similarly to video or speech media.

## 8.3 Haptics media integration in 3GPP services

### 8.3.1 5G Media Streaming services

The support of Haptic media streaming over 5G would enable the use-cases defined in section 5, particularly those for on demand and live streaming.

The 5GMS architecture [33] defines in table 4.0.1-1 the list of main features of 5GMS, highlighting whether they are specified for the uplink and/or downlink. These features are equally applicable to Haptics media as illustrated in Table 8.3.1-1.

**Table 8.3.1-1applicability of the 5GMS features to haptics enhanced media streaming services.**

|  |  |  |
| --- | --- | --- |
| **Feature** | **Procedures applicable for haptics media** | |
| **Downlink media streaming** | **Uplink media streaming** |
| Content hosting | Yes | Not applicable |
| Content publishing | Not applicable | Yes |
| Content preparation | Not defined | Not defined |
| Network assistance | Yes | Yes |
| Dynamic policies | Yes | Yes |
| Remote control | Not applicable | Yes |
| Consumption reporting | Yes | Not applicable |
| QoE metrics reporting | Yes | Not applicable |
| Edge processing | Yes | |
| eMBMS delivery | Yes | Not applicable |
| Data collection, reporting and exposure | Yes | Yes |
| Service URL handling | Yes | Yes |

In addition to the generalized architecture described in section 8.2 of this document, the 5GMS specification [33] details the UE functions of the 5GMSd client for downlink, in section 4.2.2 and uplink in section 4.2.3.

For handling haptic media, no new UE subfunctions are needed. The following highlights some requirements needed within some of the subfunctions described in [33] to support haptic media:

- **Media Access Client:** Accesses media content, such as DASH-formatted media segments, for immediate or delayed consumption. *To access haptic media content, the haptic media content needs to be provided in suitable formatted media segments (e.g. Dash segments)*

- **Media encoder** and **decoder**: Decodes or compress the media, such as audio, video *or haptics media.*

- **Media Capturing:** Devices such as video cameras or microphones that transform an analogue media signal into digital media data. For haptic media, capturing devices, sensors are described in section 6.4 of this document

**- Metrics Collection and Reporting** and **Consumption Collection and Reporting:** Information specific to haptic media, if any are for further study.

### 8.3.2 Broadcast services

The analysis of haptic media formats for broadcast services remains an area for further study. The integration of haptics media into broadcast services presents challenges that require deeper investigation such as transmission methods and user experience considerations to ensure seamless and effective implementation.

### 8.3.3 Real time media communication

#### 8.3.3.1 WebRTC services

In addition to the generalized architecture described in section 8.2 of this document, the GA4RTAR specification in section 4.2 [34] details the UE functions of the RTC endpoint. For handling haptic media, no new UE subfunctions are needed. The RTC Access Function is extended to support accessing, coding, and transmitting haptic media.

The protocol stack for a basic RTC endpoint is specified in section 13.1 of [35].

Figure 8.3.3.1-1 illustrates the integration of haptics media support in this protocol stack.



Figure 8.3.3.1-1 Haptics media support in the protocol stack for a basic RTC endpoint

Haptics media is represented at the same level as the other Video and Audio media. While there is no change to the structure of the protocol stacks, RTP payload for Haptics media would be handled similarly to those necessary for Audio and Video.

#### 8.3.3.2 IMS based RTC services

IMS based RTC services include IBACS [23] and MTSI [22], in which Haptics media can be integrated to satisfy the use case of section 5. Figure 4.3 of [22] specifies the user plane protocol stack for a basic MTSI client. It also applies to IBACS.

Figurr 8.3.32-1 illustrates the integration of haptics media support in this protocol stack.



Figure 8.3.3.2-1 Haptics media support in the user plane protocol stack for a basic MTSI endpoint

Haptics media is represented at the same level as the other media types. While there is no change to the structure of the protocol stacks, Payload formats and RTP payload for Haptics media would be handled similarly to those necessary for Audio and Video.

## 8.4 Messaging Media services

#### 8.3.4.1 System description

To support the use-cases in section 5 for haptics media enhanced messaging services, the following figures from TS 26.143 [36] are modified to illustrate haptics media functionalities in the overall Messaging Media system, Figure 8.3.4-1, and in the Messaging Media player model, Figure 8.3.4.-2.



Haptic Media Engine

Haptic Media Engine

Figure 8.3.4-1 Example system for Haptics media enhanced Messaging multimedia message exchange



Haptic media Renderer r

Haptic media Decoder

Figure 8.3.4-2 Haptics media enhanced MMBP Player Model

#### 8.3.4.2 Media Capabilities and Profiles

The MMBP media capabilities defined in TS 26.143 [36] currently does not support haptics media types. The inclusion of a top top-level haptics media type would happen at the same level as other media formats (text, speech, audio, image, etc.) and would not require modification to the multipart MMBPs and container formats (single, mixed, alternative, parallel, related, and 3GP9). A new sample entry for haptics media needs to be added to the supported sample entries in TS 26.244 [37]. For Messaging services, it is expected that a single File Format track is used for haptic media.

In addition, when the haptic media effect is included in a scene description in association with a particular node, the scene description capabilities of the messaging service need to be extended to support *MPEG\_haptic* and *MPEG\_haptic\_material* as well as *SD-Rendering-glTF-Interactive* [32].

##### 8.3.4.2.1 Player and Decoding capabilities

For instance, the capability 26143\_HAPTICSMEDIA\_MIHS can be defined as the capability of playing back (decoding and rendering) a file that:

* is decodable by a decoder capable of the MIHS decoding and rendering capabilities as defined in [7],
* is encapsulated in an ISO BMFF track with the sample entry ‘mih1’ as defined in [25]
* is contained in a 3GP file that conforms to the 26143\_CONTAINER\_MP4\_3GP9 capability defined in clause 5.2 of [36] and extended with haptics media capabilities.

The media type for files with this capability 26143\_HAPTICSMEDIA\_MIHS is signalled with haptics/mp4, profile="3gp9" codecs="mih1" or an equivalently compatible media type.

Multiple media files may be handled by creating multiple instances of the player or combining the files to output a single haptic signal for rendering through the onboard or connected actuator. Multi-technology feedback (e.g., thermal+vibrotactile) can be played simultaneously through different onboard or connected devices [7].

##### 8.3.4.2.2 MMBP content generator capabilities:

For instance, the capability 26143\_HAPTICSMEDIA\_ENC\_MIHS for a content generator can be defined as the capability of:

* Generate a haptics media file from an actuator or from a local stored file, such that the file can be played back by a player with the capability 26143\_HAPTICSMEDIA\_MIHS
* Generate an ISO BMFF track that conforms with the requirements of the sample entry 'mih1' as defined in [7].
* Generate a 3GP file from the ISO BMFF track that conforms to the 26143\_CONTAINER\_MP4\_3GP9 capability extended with haptics media capabilities
* Signal the haptic media capability with the generated file haptics/mp4, profile="3gp9" codecs="mih1" or an equivalently compatible media type

NOTE: TR 26.841 [38] studies media messaging enhancement, evaluation of haptics media as part of MMBP media capabilities in IETF MIMI [39], and in the advance file format would need to be done.

# 9 Typical haptics media traffic characteristics

## 9.1 Typical traffic characteristics summarized per use cases and formats.

In general, haptics media PCM coded bitstreams require substantially more bandwidth than parametric coded bitstreams, this is due to the capability of having silent units in a parametric bitstream.

When coding repetitive haptics media effect, a key difference between a parametric coded bitstream and a PCM coded bitstream on the traffic characteristic is the following:

- In a parametric coded bitstreams, identical or similar effects can be coded and sent a single time and then referenced, rather than coding multiple time the same or similar haptics media effect.

- In a PCM coded bitstream, similarly to audio and video, coding performance improves for these repeated consecutive effects but provide a much higher overhead compared to the parametric coded bitstream.

The following table summarise the expected characteristics and average bitrates per channel applicable for the use cases of section 5. In this table the density represents the quantity of haptics effects over the duration of the sequence, it is not related to the intensity of the haptics effect. Three parameters impact the bandwidth requirements: the number of channels, the media format and the density.

Table 9.1-1: summary of typical haptics media traffic characteristic.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Use case | Nb of channels | Media Format | Density | Average bitrate per channel |
| Haptic enhanced media distribution (5.2) | 1 to 32 | Parametric | Light  Medium  High | 0.25 to 0.75 kbps  0. 5 to 1.5 kbps  1 to 5 kbps |
| 1 to 32 | Time sampled | Light  Medium  High | From 6 to 64kbps depending on the density and the quality of the desired signal.  8-16 kbps for good quality at medium and high density.  32-64 kbps for very high quality at medium and high density. |
| Haptic enhanced communication (5.3) | 1 to 4 | Parametric | Light  Medium | 0.25 to 0.75 kbps  0.5 to 2 kbps |
| Immersive Entertainment (5.4) | 1 to 32 | Parametric | Light  Medium  High | 0.25 to 0.75 kbps  0. 5 to 1.5 kbps  1 to 5 kbps |
| 1 to 32 | Time sampled | Light  Medium  High | From 6 to 64kbps depending on the density and the quality of the desired signal.  8-16 kbps for good quality at medium and high density  32-64 kbps for very high quality at medium and high density |
| Immersive multi-modal XR and metaverse (5.5) | 6 to 32 | Parametric | Light  Medium  High | 0.25 to 0.75 kbps  0. 5 to 1.5 kbps  1 to 5 kbps |
| 6 to 32 | Time sampled | Light  Medium  High | From 6 to 64kbps depending on the density and the quality of the desired signal.  8-16 kbps for good quality at medium and high density  32-64 kbps for very high quality at medium and high density\* |

\*NOTE 1: TR 22.847, mentioned a bitrate of up to 200kpbs, which may be related to use cases for close-loop tactile internet, robotics etc.

NOTE 2: Values in the table may be re-evaluated if more tests are done.

# 10 QoS and QoE

## 10.1 Introduction

The QoS factors influencing the QoE for haptics media enhanced services as described in the use-cases of section 5 are mostly similar to those used for traditional AV and immersive AV services (jitter, delay, packet loss, etc). The asynchronicity between haptics and other media is a prevalent parameter to be considered when using haptics along with other media. Another characteristic of parametric haptics media is the presence of silent units alongside temporal and spatial units; which can be taken in account when setting PDU Set Importance.

## 10.2 Asynchronicity between haptics and other media

In Haptics enhanced media services, the quality of experience may be impacted by the asynchronicity of the haptics media with either audio, or video. This asynchronicity may be due to various QoS factors which are FFS.

Studies demonstrate (such as [21], [20] and [40]) that haptics effects perceived prior to visual cues is less detrimental to the user experience than the reverse and that asynchronicity is more tolerable in scenarios involving passive user involvement than active user involvement.

The suitable threshold for asynchronicity [20], [21], [2], [4] is listed in table 10.2-1

Table 10.2-1: Tolerable asynchronicity thresholds per use-cases

|  |  |  |  |
| --- | --- | --- | --- |
| Use case | Media | Tolerable asynchronicity threshold (note 1) | |
| Haptic enhanced media distribution (5.2)[40] | audio-haptics | audio delay:  100ms | haptic delay:  50ms |
| visual-haptics | visual delay:  80ms | Haptic delay:  60ms |
| Haptic enhanced communication (5.3) [4] | audio-haptics | audio delay:  3 frames (25ms) | haptic delay:  1 frame (12ms) |
| Visual-haptics | Visual delay:  20ms | Haptic delay:  30ms |
| Immersive games and Immersive multimodal XR and metaverse (5.4 and 5.5) [2] | audio-haptics | audio delay:  50 ms | haptic delay:  25 ms 1 frame for gaming |
| visual-haptics | visual delay:  15 ms | Haptic delay:  50 ms |
| Immersive entertainment (5.4) [4] | audio-haptics | audio delay:  25 ms | haptic delay:  12 ms |
| visual-haptics | visual delay:  20 ms | Haptic delay:  30ms |
|  |  |  |  |
|  |  |  |
|  | NOTE: For each media component, “delay” refers to the case where that media component is delayed compared to the other. | | |

## 10.3 QoS requirements for haptic media enhanced services

The following from [2] is adjusted to the use cases in section 5.

1. Users in an enhanced haptics communication services are equipped with multiple devices to receive and/or capture and transmit audio, video and haptics media.

2. When a session starts, multiple streams are established over the wireless network between the corresponding UEs. Table 10.3-1 depicts the typical QoS requirements that need be fulfilled for the users’ QoE to be satisfactory.

Table 10.3-1 Typical QoS requirements for haptic media enhanced services

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Haptics** | **Video** | **Audio** |
| Jitter (ms) | ≤ 2 | ≤ 30 | ≤ 30 |
| Delay (ms) | ≤ 50 | ≤ 400 | ≤ 150 |
| Packet loss (%) | ≤ 10 | ≤ 1 | ≤ 1 |
| Update rate (Hz) | ≥ 2000 | ≥ 24 | ≥ 20k |
| Packet size (bytes) per channel | 60-350 compressed parametric  50-1500 time sampled | ≤ MTU | 160-320 |
| Throughput (kbit/s) per channel | n\*(fe\*16)1 bits/s   time sampled  16-32kbps for compressed parametric2 | 2500 - 40000 | 64-128 |

1NOTE: where n is the number of channels and fe the sampling frequency synthetized in 16 bits.

2NOTE: throughput for compressed parametric vary based on density and placement of keyframe.

3. Haptics media (MIHS), video and voice or audio are captured with different sampling intervals. A pacer may be configured to transmit the haptic media unit (MIHS) earlier than the media units (MUs) of audio and video, which may result in different arrival time of the MUs of different modalities. Conversely, if the receiver outputs the MUs synchronously, the output of the haptics media MU may need to be delayed until the voice, audio and or video MUs arrive at the destination. Synchronization at the rendering time may be performed by buffer and jitter management. Other mechanisms are for further study.

4. Synchronization between the media streams becomes increasingly challenging when considering asynchronicity between two or three media types, and with the increasing demand from the application itself, for example in an immersive XR experience, or in dense haptics enhanced media services. . Whether any additional information needs to be exchanged with the 5G network for the assistance of the synchronization at a suitable granularity needs to be further study. NOTE: Alignment with the work in SA2 and RAN needs to be considered.

Whether the 5QIs defined in TS 23.501 for media streaming and communication services are sufficient to enable haptics media enhanced services may need to be studied further.

## 10.4 PDU Set Guidelines for haptics media

### 10.4.1 PDU Set Importance (PSI)

In general, whenever the RAN needs to discard packets (e.g., under congestion situations), it is better to discard packets of lower importance rather than discarding packets randomly. If a discarded packet is critical for the media stream, the QoE may be severely degraded. For this reason, the PDU Set Importance (PSI) field [41] can be used to mark PDU Sets with their importance level. The PSI field can then be used by the RAN to discard PDU sets. In case of congestion, PDU Sets with higher PSI values are more likely to be discarded.

NOTE: Whether PDU Set handling for haptics data is beneficial for the network and the application is for further study.

The RTP payload format for the transport of MPEG-I haptic data is defined in [draft-ietf-avtcore-rtp-haptics](https://datatracker.ietf.org/doc/html/draft-ietf-avtcore-rtp-haptics-02) IETF draft [29] . The RTP payload header format allows for packetization of MIHS units in an RTP packet payload as well as fragmentation of an MIHS unit into multiple RTP packets.

The RTP Payload Header of a haptics media RTP packet follows the RTP header. Figure 10.4.1-1 describes the RTP Payload Header for Haptics media as described in RTP Payload format for Haptics specification.

+---------------+

|0|1|2|3|4|5|6|7|

+-+-+-+-+-+-+-+-+

|D| UT | L |

+-+-----+-------+

Figure 10.4.1-1: RTP payload header for Haptics media

The haptic media characteristics may include a MIHS unit type, a dependent MIHS unit and a MIHS unit Layer information. The PSI field in a PDU Set Marking RTP HE may be assigned based on the haptic media characteristics.

An initialization MIHS unit contains MIHS packets carrying metadata necessary to reset and initialize a haptic decoder, including a timestamp. So, initialization MIHS units are essential to configure the decoder at the beginning of the decoding process. In a haptic media RTP packet containing the initialization MIHS units, the unit type field of the RTP payload header is set with value 1. When the unit type field value in the RTP payload header of an RTP packet is 1, then the corresponding PDUs in that PDU Set should be set with higher importance relative to other haptic media data PDUs.

A spatial MIHS unit may contain one or more MIHS packets providing time-independent effects, such as vibrotactile texture, stiffness, and friction. The spatial MIHS packets present in a haptics media bitstream are independently decodable by a haptics media decoder. Hence the spatial MIHS units present in a PDU Set may be delivered with higher priority. In a haptic media RTP packet containing the spatial MIHS units, the unit type field of the RTP payload header is set with value 3. When the unit type field value in the RTP payload header of an RTP packet is 3, then the corresponding PDUs in that PDU Set should be set with higher importance relative to temporal and silent haptic media data PDUs and equal or slightly lesser importance relative to initialization haptic media data PDUs.

A temporal MIHS unit may contain one or more MIHS packets defining time-dependent effects and providing modalities such as pressure, velocity, and acceleration. Temporal MIHS units can be dependent or independent units. Independent temporal MIHS units can be decoded independently using a haptics media decoder. Whereas a dependent temporal MIHS unit is the continuation of previous MIHS units and cannot be independently decoded and rendered without having decoded previous MIHS unit(s). In a haptic media RTP packet containing the temporal MIHS units, the unit type field of the RTP payload header is set with value 2. When the temporal MIHS unit is independently decodable, then the dependency field value in the RTP payload header of such haptics media RTP packet is set to 0. When the temporal MIHS unit is not independently decodable, then the dependency field value in the RTP payload header of such haptics media RTP packet is set to 1. When the unit type field value in the RTP payload header of an RTP packet is 1 and the dependency field value is zero, then the corresponding PDUs in that PDU Set should be set with higher importance relative to dependent temporal haptic media data PDUs and equal importance relative to spatial haptic media data PDUs. When the unit type field value in the RTP payload header of an RTP packet is 1 and the dependency field value is 1, then the corresponding PDUs in that PDU Set should be set with higher importance relative to silent haptic media data PDUs and lesser importance relative to initialization, spatial and independent temporal haptic media data PDUs.

A silent MIHS unit indicates that there is no effect during a time interval. The silent packets are of less importance relative to other packets as they are transmitted to the receiver to indicate that there is no haptic effect during a period of time. In a haptic media RTP packet containing the silent MIHS units, the unit type field of the RTP payload header is set with value 4. When the unit type field value in the RTP payload header of an RTP packet is 4, then the corresponding PDUs in that PDU Set should be set with lower importance relative to other haptic media data PDUs.

The MIHS Layer field present in the RTP payload header of a haptic RTP packet indicates the priority order of the MIHS unit included in the RTP payload, based on application-specific needs. The value zero for the MIHS Layer field corresponds to the highest priority. When the MIHS Layer field value in the RTP payload header of an RTP packet is zero, then the corresponding PDUs in that PDU Set shall be set with higher importance or with a lower PSI value compared to the PDUs or PDU Sets with MIHS Layer field value greater than zero. When a list of PDUs or PDU Sets contain the same MIHS Layer field value in the RTP payload header, then the PSI value of such PDUs or PDU Sets are set based on the Unit Type field and the dependency field values present in the RTP payload header.

### 10.4.2 IP packet filtering

The 3GPP TS 29.514 [42] specification defines various media types that can be carried by an application flow. The ApplicationFlowDescription data type lists the types of the media content present in a specific application flow.

For haptics media, the definition of MediaType present in clause 5.6.3.3 of [42] can be extended as below.

Table 10.4.2-1: Enumeration MediaType

|  |  |  |
| --- | --- | --- |
| Enumeration value | Description | Applicability |
| AUDIO | The type of media is audio. |  |
| VIDEO | The type of media is video. |  |
| DATA | The type of media is data. |  |
| APPLICATION | The type of media is application data. |  |
| CONTROL | The type of media is control. |  |
| TEXT | The type of media is text. |  |
| MESSAGE | The type of media is message |  |
| HAPTICS | The type of media is haptics |  |
| OTHER | Other type of media. |  |

When haptics media content is transported using RTP packets, the media name in the "m=" line of the SDP message is set to haptics as defined in RTP Payload format for Haptics specification.

An example of media representation corresponding to the hmpg RTP payload in SDP is as follows:

m=haptics 43291 UDP/TLS/RTP/SAVPF 115

a=rtpmap:115 hmpg/8000

a=fmtp:115 hmpg-profile=1;hmpg-lvl=1;hmpg-ver=2023

The Payload Type of the haptics media is identified by parsing the m=haptics line in the SDP message. In the above SDP example, the Payload Type value for haptics media is 115.

When an application data flow contains haptics media and other media content, the Payload Type (PT) values in the RTP header of RTP packets may be parsed to detect the PDUs or PDU Sets belonging to haptics media.

# 11 Identified characteristics, gaps and potential requirements

## 11.1 Identified characteristics and gaps in scenarios and services

To enable haptics enhanced media experiences, the following gaps and characteristics, based on the use-cases and relevant 3GPP services collected in this document, have been identified:

- 3GPP media services currently do not support any interoperable haptics media formats or codecs.

- Haptic interoperable services are hindered by the current prevalence of proprietary formats.

- A codec supporting haptics media in both parametric and PCM formats, as well as providing a parametric mezzanine format to and from which other parametric haptics formats could be converted without loss, would be required to address the use-cases.

- Lossless compression was not identified as a need for open loop haptics enhanced media services.

- The bandwidth required to support haptics-enhanced media services varies based on the number of haptics channels (up to 32 or 64), the haptics media format (time sampled versus parametric), and on the density of the channel. A compressed format is deemed suitable.

- In addition, the traffic characteristics of a haptics media stream in its parametric form differs substantially from any timed sampled media (haptics, audio, video), with bursts of data being followed by silent unit periods which can be longer than few seconds.

- The QoE of haptics-enhanced media services may be substantially impacted by the asynchronicity of haptics with other media and may trigger the end-user to turn-off haptics media rendering. Handling of asynchronicity delays in 3GPP services will need to be carefully considered.

The following technical aspects would need to be addressed to ensure the interoperability of haptics-enhanced media in 3GPP services:

- Defining the support for a top-level haptics media type and sample entry (e.g., in TS 26.143 [36], TS 26.244 [37], TS 29.514 [42]).

- Defining the support for an interoperable haptics media formats and codecs in streaming, and real-time communication architecture specifications, along with suitable constraints (e.g. number of channels, number of tracks, parametrics and/or PCM formats) in the service specifications (e.g. TS 26.264 [23], TS 26.511 [47]) and in UE capability specifications (e.g. TS 26.119 [32], TS 26.114 [22]).

- For services based on split rendering architecture (e.g. TS [26.565](https://www.3gpp.org/DynaReport/26565.htm" \t "_blank) [24] and TS [26.567](https://www.3gpp.org/DynaReport/26567.htm" \t "_blank) [48]), in addition to the support for an interoperable haptics media formats and codecs, the support for negotiating split haptic media capabilities needs to be defined and are for further study. Use of non-3GPP haptics codecs are out of scope of this document.

- Defining the support in TS 26.119 of the scene description extensions for haptics, and the support of scene description extension for interactivity at the node level, to support haptics media integration within a Scene Description.

- Defining the support for haptics media packaging in the file format; in transport protocols such as DASH, RTP, and webRTC, as well as signaling and protocol configurations.

- Assessing more precisely if the 5QIs and QoS parameters currently used in media services are sufficient to address the desired QoE in active and passive haptics media enhanced experiences.

## 11.2 Potential requirements for a haptics media codec and its support

From the collected scenarios and analysis of the identified 3GPP services the following requirements for a new haptics media codec have been identified to ensure interoperability across devices and services:

- The support of a standardised compression format to provide haptics enhanced media experience.

- The support of both parametric and time sampled signals.

- The provision of a parametric mezzanine format to and from which other parametric formats may be converted without loss.

- The support for compression and decompression of one or more haptics channels.

From the identified candidate technologies studied in this report, MPEG Haptics defined in [7] fulfills these potential requirements.

The following requirements have also been identified to support a haptics media codec:

- File format, DASH and RTP support for the haptics media content and stream.

- Further evaluation on the potential need for enhanced handling of asynchronicity and multi-flow synchronisation for haptics enhanced media services containing one or more haptics channels or effects

- Definition of QoS requirements for haptics enhanced media services.

From the identified candidate technologies studied in this document, ISO/IEC 23090-32 [25], the IETF "RTP Payload for Haptics"[29] fulfill these requirements.

# 12 Conclusions and proposed next steps

Haptics media refers to the sense of touch, and encompasses tactile (vibration, temperature, pressure, electro tactile), kinesthetic (force) and proprioception (motion, acceleration). Those modalities are rendered to the human body through different mechanoreceptors and thermoreceptors. Haptics media is an additional dimension that complements the sense of sight (video) and hearing (audio) and increases the level of immersion by enabling new user interactions with objects and environments. This functionality is currently not supported in 3GPP media services.

The present document provides a detailed review of the relevant use-cases, interoperability requirements, performance characteristics of haptics media formats and codecs to enable haptics enhanced media streaming and communication services.

This document has identified needs and characteristics of haptics media for different scenarios as well as gaps in 3GPP specifications and associated requirements for a new haptic media codec. Candidate technologies have been identified, including those that meet the requirements.

The addition of any new haptics codec in 3GPP services and specifications would require diligent preparation. The information in this document is meant to serve as a baseline for potential addition of a haptic media codec in 3GPP services and specifications.

Annex A (informative):  
Example of a JSON HJIF file

{

"version":"2023",

"profile":"Main",

"level":1,

"date":"Mon Sep 16 15:17:39 2024\n",

"description":"Ringtone effect",

"timescale":1000,

"avatars":[

],

"perceptions":[

{

"id":0,

"avatar\_id":0,

"description":"Vibration effect",

"perception\_modality":"Vibrotactile",

"unit\_exponent":-3,

"perception\_unit\_exponent":0,

"reference\_devices":[

],

"effect\_library":[

],

"channels":[

{

"id":0,

"description":"Left Hand Palm",

"gain":1.0,

"mixing\_coefficient":0.9999983310699463,

"body\_part\_mask":131072,

"bands":[

{

"band\_type":"VectorialWave",

"lower\_frequency\_limit":0,

"upper\_frequency\_limit":1000,

"effects":[

{

"effect\_type":"Basis",

"position":74,

"phase":0.0,

"base\_signal":"Sine",

"keyframes":[

{

"relative\_position":0,

"amplitude\_modulation":0.12156867980957031,

"frequency\_modulation":142

},

{

"relative\_position":195,

"amplitude\_modulation":0.12156867980957031,

"frequency\_modulation":142

}

]

},

{

"effect\_type":"Basis",

"position":542,

"phase":0.0,

"base\_signal":"Sine",

"keyframes":[

{

"relative\_position":0,

"amplitude\_modulation":0.5921568870544434,

"frequency\_modulation":200

},

{

"relative\_position":20,

"amplitude\_modulation":0.5921568870544434

},

{

"relative\_position":77,

"amplitude\_modulation":-0.003921568393707275,

"frequency\_modulation":200

}

]

},

{

"effect\_type":"Basis",

"position":844,

"phase":0.0,

"base\_signal":"Sine",

"keyframes":[

{

"relative\_position":0,

"amplitude\_modulation":0.12156867980957031,

"frequency\_modulation":142

},

{

"relative\_position":118,

"amplitude\_modulation":0.12156867980957031,

"frequency\_modulation":142

}

]

},

{

"effect\_type":"Basis",

"position":1004,

"phase":0.0,

"base\_signal":"Sine",

"keyframes":[

{

"relative\_position":0,

"amplitude\_modulation":0.4431372880935669,

"frequency\_modulation":200

},

{

"relative\_position":91,

"amplitude\_modulation":0.4431372880935669,

"frequency\_modulation":200

}

]

},

{

"effect\_type":"Basis",

"position":1320,

"phase":0.0,

"base\_signal":"Sine",

"keyframes":[

{

"relative\_position":0,

"amplitude\_modulation":0.12156867980957031,

"frequency\_modulation":142

},

{

"relative\_position":118,

"amplitude\_modulation":0.12156867980957031,

"frequency\_modulation":142

}

]

},

{

"effect\_type":"Basis",

"position":1651,

"phase":0.0,

"base\_signal":"Sine",

"keyframes":[

{

"relative\_position":0,

"amplitude\_modulation":0.5921568870544434,

"frequency\_modulation":200

},

{

"relative\_position":20,

"amplitude\_modulation":0.5921568870544434

},

{

"relative\_position":77,

"amplitude\_modulation":-0.003921568393707275,

"frequency\_modulation":200

}

]

},

{

"effect\_type":"Basis",

"position":1937,

"phase":0.0,

"base\_signal":"Sine",

"keyframes":[

{

"relative\_position":0,

"amplitude\_modulation":0.4431372880935669,

"frequency\_modulation":200

},

{

"relative\_position":91,

"amplitude\_modulation":0.4431372880935669,

"frequency\_modulation":200

}

]

},

{

"effect\_type":"Basis",

"position":2242,

"phase":0.0,

"base\_signal":"Sine",

"keyframes":[

{

"relative\_position":0,

"amplitude\_modulation":0.5921568870544434,

"frequency\_modulation":200

},

{

"relative\_position":0,

"amplitude\_modulation":0.46666669845581055

},

{

"relative\_position":45,

"amplitude\_modulation":-0.003921568393707275,

"frequency\_modulation":200

}

]

},

{

"effect\_type":"Basis",

"position":2308,

"phase":0.0,

"base\_signal":"Sine",

"keyframes":[

{

"relative\_position":0,

"amplitude\_modulation":0.27843141555786133,

"frequency\_modulation":200

},

{

"relative\_position":68,

"amplitude\_modulation":0.27843141555786133,

"frequency\_modulation":200

}

]

},

{

"effect\_type":"Basis",

"position":2548,

"phase":0.0,

"base\_signal":"Sine",

"keyframes":[

{

"relative\_position":0,

"amplitude\_modulation":0.12156867980957031,

"frequency\_modulation":142

},

{

"relative\_position":195,

"amplitude\_modulation":0.12156867980957031,

"frequency\_modulation":142

}

]

},

{

"effect\_type":"Basis",

"position":3016,

"phase":0.0,

"base\_signal":"Sine",

"keyframes":[

{

"relative\_position":0,

"amplitude\_modulation":0.5921568870544434,

"frequency\_modulation":200

},

{

"relative\_position":20,

"amplitude\_modulation":0.5921568870544434

},

{

"relative\_position":77,

"amplitude\_modulation":-0.003921568393707275,

"frequency\_modulation":200

}

]

},

{

"effect\_type":"Basis",

"position":3318,

"phase":0.0,

"base\_signal":"Sine",

"keyframes":[

{

"relative\_position":0,

"amplitude\_modulation":0.12156867980957031,

"frequency\_modulation":142

},

{

"relative\_position":118,

"amplitude\_modulation":0.12156867980957031,

"frequency\_modulation":142

}

]

},

{

"effect\_type":"Basis",

"position":3478,

"phase":0.0,

"base\_signal":"Sine",

"keyframes":[

{

"relative\_position":0,

"amplitude\_modulation":0.4431372880935669,

"frequency\_modulation":200

},

{

"relative\_position":91,

"amplitude\_modulation":0.4431372880935669,

"frequency\_modulation":200

}

]

},

{

"effect\_type":"Basis",

"position":3794,

"phase":0.0,

"base\_signal":"Sine",

"keyframes":[

{

"relative\_position":0,

"amplitude\_modulation":0.12156867980957031,

"frequency\_modulation":142

},

{

"relative\_position":118,

"amplitude\_modulation":0.12156867980957031,

"frequency\_modulation":142

}

]

},

{

"effect\_type":"Basis",

"position":4099,

"phase":0.0,

"base\_signal":"Sine",

"keyframes":[

{

"relative\_position":0,

"amplitude\_modulation":0.5921568870544434,

"frequency\_modulation":200

},

{

"relative\_position":20,

"amplitude\_modulation":0.5921568870544434

},

{

"relative\_position":77,

"amplitude\_modulation":-0.003921568393707275,

"frequency\_modulation":200

}

]

},

{

"effect\_type":"Basis",

"position":4411,

"phase":0.0,

"base\_signal":"Sine",

"keyframes":[

{

"relative\_position":0,

"amplitude\_modulation":0.4431372880935669,

"frequency\_modulation":200

},

{

"relative\_position":91,

"amplitude\_modulation":0.4431372880935669,

"frequency\_modulation":200

}

]

},

{

"effect\_type":"Basis",

"position":4716,

"phase":0.0,

"base\_signal":"Sine",

"keyframes":[

{

"relative\_position":0,

"amplitude\_modulation":0.5921568870544434,

"frequency\_modulation":200

},

{

"relative\_position":0,

"amplitude\_modulation":0.46666669845581055

},

{

"relative\_position":45,

"amplitude\_modulation":-0.003921568393707275,

"frequency\_modulation":200

}

]

},

{

"effect\_type":"Basis",

"position":4782,

"phase":0.0,

"base\_signal":"Sine",

"keyframes":[

{

"relative\_position":0,

"amplitude\_modulation":0.27843141555786133,

"frequency\_modulation":200

},

{

"relative\_position":68,

"amplitude\_modulation":0.27843141555786133,

"frequency\_modulation":200

}

]

},

{

"effect\_type":"Basis",

"position":5023,

"phase":0.0,

"base\_signal":"Sine",

"keyframes":[

{

"relative\_position":0,

"amplitude\_modulation":0.12156867980957031,

"frequency\_modulation":142

},

{

"relative\_position":108,

"amplitude\_modulation":0.12156867980957031,

"frequency\_modulation":142

}

]

},

{

"effect\_type":"Basis",

"position":5487,

"phase":0.0,

"base\_signal":"Sine",

"keyframes":[

{

"relative\_position":0,

"amplitude\_modulation":0.5921568870544434,

"frequency\_modulation":200

},

{

"relative\_position":57,

"amplitude\_modulation":-0.003921568393707275,

"frequency\_modulation":200

}

]

},

{

"effect\_type":"Basis",

"position":5640,

"phase":0.0,

"base\_signal":"Sine",

"keyframes":[

{

"relative\_position":0,

"amplitude\_modulation":0.34117650985717773,

"frequency\_modulation":200

},

{

"relative\_position":86,

"amplitude\_modulation":0.34117650985717773,

"frequency\_modulation":200

}

]

},

{

"effect\_type":"Basis",

"position":5796,

"phase":0.0,

"base\_signal":"Sine",

"keyframes":[

{

"relative\_position":0,

"amplitude\_modulation":0.12156867980957031,

"frequency\_modulation":142

},

{

"relative\_position":118,

"amplitude\_modulation":0.12156867980957031,

"frequency\_modulation":142

}

]

},

{

"effect\_type":"Basis",

"position":5949,

"phase":0.0,

"base\_signal":"Sine",

"keyframes":[

{

"relative\_position":0,

"amplitude\_modulation":0.4431372880935669,

"frequency\_modulation":200

},

{

"relative\_position":91,

"amplitude\_modulation":0.4431372880935669,

"frequency\_modulation":200

}

]

},

{

"effect\_type":"Basis",

"position":6265,

"phase":0.0,

"base\_signal":"Sine",

"keyframes":[

{

"relative\_position":0,

"amplitude\_modulation":0.12156867980957031,

"frequency\_modulation":142

},

{

"relative\_position":118,

"amplitude\_modulation":0.12156867980957031,

"frequency\_modulation":142

}

]

},

{

"effect\_type":"Basis",

"position":6570,

"phase":0.0,

"base\_signal":"Sine",

"keyframes":[

{

"relative\_position":0,

"amplitude\_modulation":0.5921568870544434,

"frequency\_modulation":200

},

{

"relative\_position":20,

"amplitude\_modulation":0.5921568870544434

},

{

"relative\_position":77,

"amplitude\_modulation":-0.003921568393707275,

"frequency\_modulation":200

}

]

},

{

"effect\_type":"Basis",

"position":6882,

"phase":0.0,

"base\_signal":"Sine",

"keyframes":[

{

"relative\_position":0,

"amplitude\_modulation":0.4431372880935669,

"frequency\_modulation":200

},

{

"relative\_position":91,

"amplitude\_modulation":0.4431372880935669,

"frequency\_modulation":200

}

]

},

{

"effect\_type":"Basis",

"position":7187,

"phase":0.0,

"base\_signal":"Sine",

"keyframes":[

{

"relative\_position":0,

"amplitude\_modulation":0.5921568870544434,

"frequency\_modulation":200

},

{

"relative\_position":0,

"amplitude\_modulation":0.46666669845581055

},

{

"relative\_position":45,

"amplitude\_modulation":-0.003921568393707275,

"frequency\_modulation":200

}

]

},

{

"effect\_type":"Basis",

"position":7253,

"phase":0.0,

"base\_signal":"Sine",

"keyframes":[

{

"relative\_position":0,

"amplitude\_modulation":0.27843141555786133,

"frequency\_modulation":200

},

{

"relative\_position":68,

"amplitude\_modulation":0.27843141555786133,

"frequency\_modulation":200

}

]

},

{

"effect\_type":"Basis",

"position":7493,

"phase":0.0,

"base\_signal":"Sine",

"keyframes":[

{

"relative\_position":0,

"amplitude\_modulation":0.12156867980957031,

"frequency\_modulation":142

},

{

"relative\_position":107,

"amplitude\_modulation":0.12156867980957031,

"frequency\_modulation":142

}

]

},

{

"effect\_type":"Basis",

"position":7961,

"phase":0.0,

"base\_signal":"Sine",

"keyframes":[

{

"relative\_position":0,

"amplitude\_modulation":0.5921568870544434,

"frequency\_modulation":200

},

{

"relative\_position":20,

"amplitude\_modulation":0.5921568870544434

},

{

"relative\_position":77,

"amplitude\_modulation":-0.003921568393707275,

"frequency\_modulation":200

}

]

},

{

"effect\_type":"Basis",

"position":8271,

"phase":0.0,

"base\_signal":"Sine",

"keyframes":[

{

"relative\_position":0,

"amplitude\_modulation":0.12156867980957031,

"frequency\_modulation":142

},

{

"relative\_position":118,

"amplitude\_modulation":0.12156867980957031,

"frequency\_modulation":142

}

]

},

{

"effect\_type":"Basis",

"position":8423,

"phase":0.0,

"base\_signal":"Sine",

"keyframes":[

{

"relative\_position":0,

"amplitude\_modulation":0.4431372880935669,

"frequency\_modulation":200

},

{

"relative\_position":91,

"amplitude\_modulation":0.4431372880935669,

"frequency\_modulation":200

}

]

},

{

"effect\_type":"Basis",

"position":8739,

"phase":0.0,

"base\_signal":"Sine",

"keyframes":[

{

"relative\_position":0,

"amplitude\_modulation":0.12156867980957031,

"frequency\_modulation":142

},

{

"relative\_position":118,

"amplitude\_modulation":0.12156867980957031,

"frequency\_modulation":142

}

]

},

{

"effect\_type":"Basis",

"position":9044,

"phase":0.0,

"base\_signal":"Sine",

"keyframes":[

{

"relative\_position":0,

"amplitude\_modulation":0.5921568870544434,

"frequency\_modulation":200

},

{

"relative\_position":20,

"amplitude\_modulation":0.5921568870544434

},

{

"relative\_position":77,

"amplitude\_modulation":-0.003921568393707275,

"frequency\_modulation":200

}

]

},

{

"effect\_type":"Basis",

"position":9356,

"phase":0.0,

"base\_signal":"Sine",

"keyframes":[

{

"relative\_position":0,

"amplitude\_modulation":0.4431372880935669,

"frequency\_modulation":200

},

{

"relative\_position":91,

"amplitude\_modulation":0.4431372880935669,

"frequency\_modulation":200

}

]

},

{

"effect\_type":"Basis",

"position":9661,

"phase":0.0,

"base\_signal":"Sine",

"keyframes":[

{

"relative\_position":0,

"amplitude\_modulation":0.5921568870544434,

"frequency\_modulation":200

},

{

"relative\_position":0,

"amplitude\_modulation":0.46666669845581055

},

{

"relative\_position":45,

"amplitude\_modulation":-0.003921568393707275,

"frequency\_modulation":200

}

]

},

{

"effect\_type":"Basis",

"position":9727,

"phase":0.0,

"base\_signal":"Sine",

"keyframes":[

{

"relative\_position":0,

"amplitude\_modulation":0.27843141555786133,

"frequency\_modulation":200

},

{

"relative\_position":68,

"amplitude\_modulation":0.27843141555786133,

"frequency\_modulation":200

}

]

}

]

}

]

}

]

}

],

"syncs":[

{

"timestamp":0,

"timescale":1000

}

]

}

Annex B (informative):  
Change history

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Change history** | | | | | | | |
| **Date** | **Meeting** | **TDoc** | **CR** | **Rev** | **Cat** | **Subject/Comment** | **New version** |
| 2024-08 | S4-129e |  |  |  |  | Proposed skeleton TR | 0.0.1 |
| 2024-08 | S4-129e | S4-241506  S4-241771  S4-241770  S4-241761  S4-241769 |  |  |  | Includes: Use Cases for Immersive entertainment (S4-241506), Haptics-enhanced Communication (S4-241771) Immersive multi-modal XR and metaverse (S4-241770); Device types (S4-241761); input formats (S4-241769)  Editor’s clean-up | 0.1.0 |
| 2024-09 | RTC adhoc call sept 25th | S4aR240067  S4aR240048  S4aR240049  S4aR240050  S4aR240052 |  |  |  | Includes: new use-case for haptics enhanced media distribution (S4aR240067); revision to use-cases (S4aR240048, S4aR240049, S4aR240050); haptics media definition and pipeline (S4aR240052)  Editor’s clean-up | 0.1.1 |
| 2024-10 | RTC adhoc call Oct 23rd | S4aR240077 |  |  |  | Includes: S4aR240077  Editor’s clean-up | 0.2.0 |
| 2024-11 | S4#131 | S4-241852  S4-241854  S4-242099  S4-242100  S4-241856  S4-241857  S4-241858  S4-241859 |  |  |  | Includes: communication pipeline (S4-241852), control loops (S4-241854), update device types (S4-242099),define haptics signals and media formats (S4-242100), haptics media traffic characteristics (S4-241856), integration in 3GPP architecture (S4-241857), integration in 3GPP services (S4-241858) candidate technologies (S4-241859), Update to haptic communication use case (S4-241859)  Editor’s clean-up. | 0.3.0 |
| 2024-12 | SA#106 | SP-241753 |  |  |  | Version 1.0.0 created by MCC to present to TSG for information | 1.0.0 |
| 2024-12 | RTC adhoc call Dec 17th | S4aR250011 |  |  |  | Includes: details of Haptics media integration in SD (S4aR250011) | 1.0.1 |
| 2025-01 | RTC adhoc Call Jan 15th | S4aR250012  S4aR250016 |  |  |  | Includes: details for openXR API (S4aR250012) and remove editor’s note in section 6.4 device types (S4aR250013) | 1.0.1 |
| 2025-02 | RTC adhoc Call Feb 5th | S4-250050  S4-250053  S4-250054   S4-250056   S4-250055 |  |  |  | Includes: refinement of OpenXR API (S4-250050), use case clean-up (S4-250053), haptics in messaging services (S4-250053)  QoE and PSI (S4-250056), revised text for codec performances (S4-250055)  Editor’s clean-up. | 1.0.1 |
| 2025-02 | SA4#131 | S4-250153  S4-250181  S4-250241  S4-250334  S4-250339  S4-250337  S4-250340 |  |  |  | Includes: Editor’s review, scope (S4-250153), revision of use case 5.2 (S4-250181), split operation (S4-250241, S4-250334), revised QoE/QoS section (S4-250339), gap (S4-250337), conclusion (S4-250340) | 1.0.2 |