 ISO/IEC JTC 1/SC 29/WG 3 N01756

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

**MPEG Systems**

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

**Document type:** Output Document

**Title:** Exploration of carriage of static Gaussian Splat in HEIF

**Status:** Approved

**Date of document:** 2026-01-23

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

**Expected action:** None

**Action due date:** None

**No. of pages:** 20 (including cover page)

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

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

**INTERNATIONAL ORGANIZATION FOR STANDARDIZATION**

**ORGANISATION INTERNATIONALE DE NORMALISATION**

**ISO/IEC JTC1/SC29/WG3**

**CODING OF MOVING PICTURES AND AUDIO**

**ISO/IEC JTC1/SC29/WG3 N01756**

**January 2026, online**

|  |  |
| --- | --- |
| **Title** | **Exploration of** **carriage of static Gaussian Splat in HEIF** |
| **Source** | **WG 03, MPEG Systems** |
| **Editor** | **Thomas Stockhammer (Qualcomm)** |
| **Status** | **Approved** |
| **Serial Number** | **26916** |

# Summary

During MPEG#153 a first proposal was provided of carriage of static Gaussian Splat in HEIF. This document summarizes the proposal and provides also the discussion and open questions collected during the meeting.

# 1 Introduction

A High Efficiency Image File Format (HEIF)-based carriage for Gaussian Splat content as new profiles is considered. Gaussian Splats represent a scene as a set of anisotropic 3D Gaussians with per primitive appearance terms and are increasingly used to enable photorealistic rendering from different viewpoints.

The objective is to enable interoperable storage and delivery of Gaussian Splats in the same packaging ecosystem used for still images and image sequences. A conforming file shall remain usable by legacy HEIF viewers by providing a conventional cover image, while Gaussian splat aware decoders can access the additional representation and render a 3D experience.

The proposal is compression agnostic. Uncompressed stream items are defined for baseline exchange. Compressed stream items are supported through explicit per item signaling of the compressor used, and through track-based carriage for timed Gaussian Splats.

# 2 Overview: GS as a HEIF profile

## 2.1 General

A Gaussian Splat representation is carried as an auxiliary image item associated with a conventional cover/master image item.

## 2.2 Carriage model

The primary item is a coded image item that acts as a 2D cover image for legacy HEIF viewers.

A Gaussian Splat (GS) representation item is an auxiliary image item linked to the cover image through an item reference of type 'auxl'. The GS representation item groups one or more GS stream items in the MetaBox or in one or more tracks identified by the GS representation item.

When GS data units are carried as items, each GS stream item carries one or more attribute sets and signals the role(s) through one or more item properties, enabling selective loading and parallel decoding.

When GS data units are carried as tracks, one or more attributes may be packed into samples of an auxiliary track or into samples of a track using an image or video coding format. Each compressed GS stream item signals the compressor used through an item property using a 4CC code. The design is independent of any specific compression algorithm.

## 2.3Role Assignment

The following is an example of roles composing a GS representation (based on the INRIA format for GS [4]). The roles may be conveyed in separate stream items or may be grouped together. When multiple roles are present in a single item or track sample, one instance of the role signalling is associated for each role.

* One stream item with role 'posi' contains position data for all points.
* One stream item with role 'rotq' contains rotation quaternions for all points.
* One stream item with role 'scl3' contains per point scale values.
* One stream item with role 'opac' contains per point opacity values.
* One stream item with role 'shco' with sh\_band\_index equal to 0 contains spherical harmonics band 0 coefficients, also called the DC term.
* One stream item with role 'shco' with sh\_band\_index equal to 1 contains spherical harmonics band 1 coefficients.
* Additional stream items with role 'shco' with sh\_band\_index equal to 2, 3, and so on, contain higher order spherical harmonics bands when present.

## 2.4 Example File Structures

The following examples show different examples of HEIF files carrying GS representations:

The first example shows a single GS image with one 2D cover image.

|  |
| --- |
| FileTypeBox 'ftyp': major-brand='heic', compatible-brands='gsc1, mif1, ...'  MetaBox 'meta':  HandlerBox 'hdlr': 'pict'  PrimaryItemBox 'pitm': item\_ID=1  ItemInfoBox 'iinf': entry\_count=8  1) infe: item\_ID=1, item\_type='hvc1' // cover image  2) infe: item\_ID=10, item\_type='gspl' // GS master item (auxiliary)  3) infe: item\_ID=11, item\_type='gspc' // stream: position  4) infe: item\_ID=12, item\_type='gspc' // stream: rotation  5) infe: item\_ID=13, item\_type='gspc' // stream: opacity  6) infe: item\_ID=14, item\_type='gspc' // stream: SH band 0 (DC)  7) infe: item\_ID=15, item\_type='gspc' // stream: SH band 1  8) infe: item\_ID=16, item\_type='gspc' // stream: SH band 2  ItemLocationBox 'iloc': item\_count=8  item\_ID=1, extent\_offset=P0, extent\_length=Q0  item\_ID=10, extent\_offset=P10, extent\_length=Q10  item\_ID=11, extent\_offset=P11, extent\_length=Q11  ...  item\_ID=16, extent\_offset=P16, extent\_length=Q16  ItemReferenceBox 'iref':  referenceType='auxl', from\_item\_ID=10, to\_item\_ID=1 // GS is auxiliary to cover  referenceType='gsst', from\_item\_ID=10, to\_item\_ID=11..16 // GS master points at streams  referenceType='cdsc', from\_item\_ID=20, to\_item\_ID=1 // optional camera metadata item  ItemPropertiesBox 'iprp':  ItemPropertyContainerBox 'ipco':  'hvcC' 'ispe' // cover config and size  'auxC' // on item 10, identifies gaussian\_splats via URI  'gsgC' // GS global config (numPoints, shDegree, coord system)  'gscC' // coding method config; compressor\_code='gscu' indicates uncompressed  'gsrC' (x6) // per-stream role property, one per stream item  'rref' // optional, to require processing 'gsst' references  ItemPropertyAssociation 'ipma':  item\_ID=1: essential=1 hvcC, essential=0 ispe  item\_ID=10: essential=1 auxC, essential=1 gsgC, essential=1 rref  item\_ID=11: essential=1 gscC, essential=1 gsrC(role='posi')  item\_ID=12: essential=1 gscC, essential=1 gsrC(role='xfrm')  item\_ID=13: essential=1 gscC, essential=1 gsrC(role='opac')  item\_ID=14: essential=1 gscC, essential=1 gsrC(role='shb0')  item\_ID=15: essential=1 gscC, essential=1 gsrC(role='shb1')  item\_ID=16: essential=1 gscC, essential=1 gsrC(role='shb2')  MediaDataBox 'mdat' (or 'idat'): payloads at P0, P10..P16 |

The following example depicts a single Gaussian Splat frame that is packed and encoded using a video codec in a track that is referenced from a Gaussian Splat item.

|  |
| --- |
| FileTypeBox 'ftyp': major-brand='heic' (or 'gsc1'), compatible-brands='gsc1, mif1, mp42, ...'   MetaBox 'meta':  HandlerBox 'hdlr': 'pict'    PrimaryItemBox 'pitm': item\_ID=1    ItemInfoBox 'iinf': entry\_count=2  1) infe: item\_ID=1, item\_type='hvc1' // cover image  2) infe: item\_ID=10, item\_type='gspl' // GS representation item (auxiliary pointer)    ItemLocationBox 'iloc':  item\_ID=1, extent\_offset=P0, extent\_length=Q0  item\_ID=10, extent\_offset=P10, extent\_length=Q10 // optional bootstrap data, may be zero length    ItemReferenceBox 'iref':  referenceType='auxl', from\_item\_ID=10, to\_item\_ID=1    ItemPropertiesBox 'iprp':  ItemPropertyContainerBox 'ipco':  'hvcC' 'ispe' // cover config and size  'auxC' // on item 10, identifies gaussian\_splats via aux\_type URN  'gsgC' // GS global config (numPoints, shDegree, coord system)  'gstR' // points to track\_ID=100 carrying packed GS data    ItemPropertyAssociation 'ipma':  item\_ID=1: essential=1 hvcC, essential=0 ispe  item\_ID=10: essential=1 auxC, essential=1 gsgC, essential=1 gstR   MovieBox 'moov':  trak (track\_ID=100) // Packed GS attribute track, encoded as video frames  mdia  hdlr: 'auxv'  minf.stbl  stsd:  sample\_entry: 'hvc1' (or any video coding sample entry)  auxi: aux\_track\_type = "urn:mpeg:gaussian\_splats:2026:auxid:1"  gsrm: access\_unit\_sample\_count=7  entry[0]: role\_code='posi', sample\_offset=0, sample\_count=1, role\_offset=0, role\_length=Lpos  entry[1]: role\_code='rotq', sample\_offset=1, sample\_count=1, role\_offset=0, role\_length=Lrot  entry[2]: role\_code='scl3', sample\_offset=2, sample\_count=1, role\_offset=0, role\_length=Lscl  entry[3]: role\_code='opac', sample\_offset=3, sample\_count=1, role\_offset=0, role\_length=Lopac  entry[4]: role\_code='shco', sh\_band\_index=0, sample\_offset=4, sample\_count=1, role\_offset=0, role\_length=Lsh0  entry[5]: role\_code='shco', sh\_band\_index=1, sample\_offset=5, sample\_count=1, role\_offset=0, role\_length=Lsh1  entry[6]: role\_code='shco', sh\_band\_index=2, sample\_offset=6, sample\_count=1, role\_offset=0, role\_length=Lsh2  samples:  F0, F1, F2, F3, F4, F5, F6 // one access unit representing a single GS frame   MediaDataBox 'mdat':  cover image item data and video-coded samples for track\_ID=100 |

The following example shows a HEIF file with multiple views of a single Gaussian Splat frame from different camera views and stored as an image sequence.

|  |
| --- |
| FileTypeBox 'ftyp':  major-brand='heic' (or 'gsc1')  compatible-brands='gsc1, mif1, mp42, ...' (and any codec brands used by 'pict' tracks)  MovieBox 'moov':  TrackBox 'trak' (track\_ID=1) // Viewpoint 0 2D image sequence  MediaBox 'mdia':  HandlerBox 'hdlr': 'pict'  MediaInformationBox 'minf':  SampleTableBox 'stbl':  SampleDescriptionBox 'stsd':  sample\_entry: 'hvc1' (or 'av01', 'vvc1', etc.)  samples:  V0\_0, V0\_1, V0\_2, ...  TrackBox 'trak' (track\_ID=2) // Viewpoint 1 2D image sequence  MediaBox 'mdia':  HandlerBox 'hdlr': 'pict'  MediaInformationBox 'minf':  SampleTableBox 'stbl':  SampleDescriptionBox 'stsd':  sample\_entry: 'hvc1' (or 'av01', 'vvc1', etc.)  samples:  V1\_0, V1\_1, V1\_2, ...  TrackBox 'trak' (track\_ID=3) // Viewpoint 2 2D image sequence  MediaBox 'mdia':  HandlerBox 'hdlr': 'pict'  MediaInformationBox 'minf':  SampleTableBox 'stbl':  SampleDescriptionBox 'stsd':  sample\_entry: 'hvc1' (or 'av01', 'vvc1', etc.)  samples:  V2\_0, V2\_1, V2\_2, ...  TrackBox 'trak' (track\_ID=30) // timed camera intrinsics/extrinsics for viewpoint 0  TrackReferenceBox 'tref':  track\_reference\_type 'cdsc' -> track\_ID=1  MediaBox 'mdia':  HandlerBox 'hdlr': 'meta'  MediaInformationBox 'minf':  SampleTableBox 'stbl':  SampleDescriptionBox 'stsd':  sample\_entry: (camera metadata sample entry)  samples:  C0\_0, C0\_1, C0\_2, ...  TrackBox 'trak' (track\_ID=31) // timed camera intrinsics/extrinsics for viewpoint 1  TrackReferenceBox 'tref':  track\_reference\_type 'cdsc' -> track\_ID=2  MediaBox 'mdia':  HandlerBox 'hdlr': 'meta'  MediaInformationBox 'minf':  SampleTableBox 'stbl':  SampleDescriptionBox 'stsd':  sample\_entry: (camera metadata sample entry)  samples:  C1\_0, C1\_1, C1\_2, ...  TrackBox 'trak' (track\_ID=32) // timed camera intrinsics/extrinsics for viewpoint 2  TrackReferenceBox 'tref':  track\_reference\_type 'cdsc' -> track\_ID=3  MediaBox 'mdia':  HandlerBox 'hdlr': 'meta'  MediaInformationBox 'minf':  SampleTableBox 'stbl':  SampleDescriptionBox 'stsd':  sample\_entry: (camera metadata sample entry)  samples:  C2\_0, C2\_1, C2\_2, ...  // Split-stream GS auxiliary tracks shared by all viewpoints  TrackBox 'trak' (track\_ID=10) // GS positions  TrackReferenceBox 'tref':  track\_reference\_type 'auxl' -> track\_ID=1, track\_ID=2, track\_ID=3  MediaBox 'mdia':  HandlerBox 'hdlr': 'auxv'  MediaInformationBox 'minf':  SampleTableBox 'stbl':  SampleDescriptionBox 'stsd':  sample\_entry: 'gspc'  AuxiliaryTypeInfoBox 'auxi': aux\_track\_type="urn:mpeg:gaussian\_splats:2026:auxid:1"  GaussianSplatCompressionConfigurationProperty 'gscC': compressor\_code='gscu'  (optional) GaussianSplatTrackRoleMapBox 'gsrm' (not needed when the track conveys only one role)  (optional) role signalling in sample entry, e.g. role\_code='posi' |

# 3 Proposed Specification text

# Definitions

* **Gaussian Splat:** a parametric primitive representing scene radiance as an oriented anisotropic Gaussian with associated appearance parameters.
* **Gaussian Splat image:** an image representation that comprises a set of Gaussian Splats intended for rendering one or more 2D views of a scene.
* **Gaussian Splat representation:** the complete set of data needed to decode and render a Gaussian Splat image, including geometry, appearance, and any required metadata.
* **Gaussian Splat auxiliary image:** an auxiliary image item whose auxiliary type identifies it as a Gaussian Splat representation, and which is associated with a master image item.
* **Gaussian Splat representation item:** an item of type 'gspl' that identifies a Gaussian Splat representation and provides references to one or more Gaussian Splat stream items and optionally to one or more Gaussian Splat data tracks.
* **Gaussian Splat stream item:** an item of type 'gspc' that carries a coded byte sequence corresponding to one or more Gaussian Splat streams, where the coding method is identified by an associated GaussianSplatCompressionConfigurationProperty ('gscC').
* **compressor\_code:** a four character code that identifies the coding method used for a Gaussian Splat stream item or Gaussian Splat data track. The value 'gscu' indicates uncompressed stream data.
* **Gaussian Splat data track:** a track that carries coded Gaussian Splat data units, optionally using an image or video coding format, and signals the coding method using compressor\_code.
* **Spherical harmonics degree:** the maximum spherical harmonics order represented by the spherical harmonics coefficients of a Gaussian Splat representation.
* **Spherical harmonics band:** a subset of spherical harmonics coefficients that share the same order and are signaled as a distinct component of the representation.

# Annex X (normative) Gaussian Splat Image File Format

## X.1 General

This annex derives a format to encapsulate Gaussian Splat representations in files conforming to the Image File Format specified in this document. The Gaussian Splat representation enables view synthesis and photorealistic rendering from novel viewpoints by storing a set of 3D Gaussian primitives and associated appearance parameters.

One profile is specified in this annex. The coding method used for each Gaussian Splat stream item is signalled by a 4CC compressor\_code in a GaussianSplatCompressionConfigurationProperty ('gscC'), including a value that indicates uncompressed stream data.

## X.2 Gaussian Splat images and image collections

### X.2.1 General

Clause X.2 specifies requirements for files containing Gaussian Splat representation items. When a brand specified in X.4.1 is among the compatible brands of a file, the requirements specified in Clause X.2 shall be applied.

The specifications of Clause 6 apply.

### X.2.2 Gaussian Splat representation items

#### X.2.2.1 Representation item of type 'gspl'

An item of type 'gspl' is a Gaussian Splat representation item.

Each Gaussian Splat representation item of type 'gspl' shall be identified as an auxiliary image item by an associated AuxiliaryTypeProperty ('auxC') as specified in 6.9 and as further constrained in X.2.4.

Each Gaussian Splat representation item of type 'gspl' shall have an item reference of type 'auxl' to its master image item as specified in 6.9.

Each Gaussian Splat representation item of type 'gspl' shall group one or more item Gaussian Splat data units. The data units shall be carried either as one or more Gaussian Splat stream items referenced using an item reference of type 'gsst' as specified in X.2.2.2, or in one or more tracks identified by associated GaussianSplatTrackReferenceProperty ('gstR ') as specified in X.2.3.3, or both.

#### X.2.2.2 Gaussian Splat stream items

A Gaussian Splat stream item contains the data for one or more Gaussian Splat streams. Each role present in the stream item shall be indicated by an associated GaussianSplatRoleProperty ('gsrC') as specified in X.2.3.1. When more than one role is present, multiple 'gsrC' item properties shall be associated with the stream item, one for each role.

Gaussian Splat stream items shall be referenced by exactly one Gaussian Splat representation item of type 'gspl' using an item reference of type 'gsst'.

##### X.2.2.2.1 Stream item of type 'gspc'

An item of type 'gspc' consists of a coded representation of one or more Gaussian Splat streams. The coding method used for the stream item shall be signalled by an associated GaussianSplatCompressionConfigurationProperty ('gscC') as specified in X.2.3.2.

The decoded stream data for each role present in the stream item shall conform to the data layouts specified in X.2.5. Each role present in the stream item shall be indicated by one or more associated GaussianSplatRoleProperty ('gsrC') item properties as specified in X.2.3.1.

### X.2.3 Gaussian Splat item properties

#### X.2.3.1 Gaussian Splat global configuration property

##### X.2.3.1.1 Definition

Box type: gsgC  
Property type: Descriptive item property  
Container: ItemPropertyContainerBox  
Mandatory (per item): Yes, for a Gaussian Splat representation item of type gspl  
Quantity (per item): One for a Gaussian Splat representation item of type gspl

The GaussianSplatGlobalConfigurationProperty specifies configuration parameters that apply to the associated Gaussian Splat representation as a whole, including the number of points, spherical harmonics degree, component format for uncompressed streams, and coordinate system flags.

essential shall be equal to 1 for a gsgC item property associated with a Gaussian Splat representation item of type gspl.

##### X.2.3.1.2 Syntax

aligned(8) class GaussianSplatGlobalConfigurationProperty

extends ItemFullProperty('gsgC', version == 0, flags == 0) {

unsigned int(8) gs\_version;

unsigned int(8) gs\_flags;

reserved bit(16) reserved;

unsigned int(32) num\_points;

unsigned int(8) sh\_degree;

unsigned int(8) component\_format;

unsigned int(16) reserved2;

}

##### X.2.3.1.3 Semantics

***gs\_version*** specifies the version of the Gaussian Splat global configuration. For this version of this annex, gs\_version shall be equal to 0.

***gs\_flags*** provides flags that define the coordinate system and interpretation of values in the data streams as defined in the following table.

**Table X.1 gs\_flags values for GaussianSplatGlobalConfigurationProperty**

|  |  |
| --- | --- |
| **Bits** | **Description** |
| 0 | when equal to 1, the coordinate system is right handed, when equal to 0, it is left handed. |
| 1 | when equal to 1, positions and scales are expressed in meters, when equal to 0, units are unspecified. |
| 2 | when equal to 1, quaternion components are ordered (qx, qy, qz, qw), when equal to 0, ordering is (qw, qx, qy, qz). |
| 3..15 | reserved, set to 0. |

***num\_points*** specifies the number of points in the Gaussian Splat representation.

***sh\_degree*** specifies the maximum spherical harmonics degree present in the Gaussian Splat representation.

***component\_format*** specifies the component representation used for uncompressed role byte sequences, i.e. when compressor\_code is equal to gscu. For this version of this annex, component\_format equal to 0 indicates IEEE 754 binary32. Other values are reserved for future use.

***reserved*** and ***reserved2*** shall be set to 0.

#### X.2.3.2 Gaussian Splat role property

##### X.2.3.2.1 Definition

Box type: 'gsrC'

Property type: Descriptive item property

Container: ItemPropertyContainerBox

Mandatory (per item): Yes, for a Gaussian Splat stream item

Quantity (per item): One or more for a Gaussian Splat stream item

The GaussianSplatRoleProperty identifies a role present in a Gaussian Splat stream item and, when applicable, parameters such as spherical harmonics band index, stream fragmentation information, and the byte range within the decoded byte sequence corresponding to the role.

*essential* shall be equal to 1 for a 'gsrC' item property associated with a Gaussian Splat stream item.

##### X.2.3.2.2 Syntax

aligned(8) class GaussianSplatRoleProperty

extends ItemFullProperty('gsrC', version == 0, flags == 0){

unsigned int(32) role\_code;

unsigned int(16) role\_instance;

unsigned int(16) role\_count;

unsigned int(16) sh\_band\_index;

reserved bit(16) reserved;

unsigned int(32) role\_offset;

unsigned int(32) role\_length;

}

##### X.2.3.2.3 Semantics

***role\_code*** specifies the role identified by this property. The role\_code value shall be one of the values specified in Table X.1, unless otherwise specified by an extension of this annex.

***role\_instance*** and ***role\_count*** specify fragmentation of a role across multiple stream items. When role\_count is greater than 1, the complete role stream is formed by concatenating the role byte sequences of the fragments in increasing role\_instance order. When role\_count is equal to 1, role\_instance shall be equal to 0.

***sh\_band\_index*** specifies the spherical harmonics band index when role\_code indicates spherical harmonics coefficients. For roles other than spherical harmonics coefficients, sh\_band\_index shall be set to 0xFFFF.

***role\_offset*** specifies the zero-based byte offset from the start of the decoded byte sequence of the associated stream item to the first byte of the role byte sequence identified by this property.

***role\_length*** specifies the length in bytes of the role byte sequence identified by this property.

For a given stream item, the byte ranges indicated by 'gsrC' item properties associated with the stream item shall not overlap, unless otherwise specified by an extension of this annex.

**Table X.1 – role\_code values for GaussianSplatRoleProperty**

|  |  |
| --- | --- |
| **role\_code** | **Semantics** |
| posi | Position of each Gaussian, expressed as three components per point. |
| rotq | Orientation of each Gaussian, expressed as four components per point (quaternion). |
| scl3 | Scale of each Gaussian, expressed as three components per point. |
| opac | Opacity of each Gaussian, expressed as one component per point. |
| shco | Spherical harmonics coefficients. The sh\_band\_index field indicates which band is present in the stream item. |

#### X.2.3.3 Gaussian Splat compression configuration property

##### X.2.3.3.1 Definition

Box type: 'gscC'

Property type: Descriptive item property

Container: ItemPropertyContainerBox

Mandatory (per item): Yes, for a Gaussian Splat stream item of type 'gspc'

Quantity (per item): One for a Gaussian Splat stream item of type 'gspc'

The GaussianSplatCompressionConfigurationProperty identifies the coding method used for an item of type 'gspc' and provides parameters required to initialize a decompressor.

essential shall be equal to 1 for a 'gscC' item property associated with a Gaussian Splat stream item of type 'gspc'.

##### X.2.3.3.3 Syntax

aligned(8) class GaussianSplatCompressionConfigurationProperty

extends ItemFullProperty('gscC', version == 0, flags == 0) {

unsigned int(32) compressor\_code;

unsigned int(32) compressor\_parameter\_length;

unsigned int(8) compressor\_parameter[compressor\_parameter\_length];

}

##### X.2.3.3.3 Semantics

***compressor\_code*** is a four-character code identifying the coding method used for the item of type 'gspc'. The interpretation of compressor\_parameter is defined by the identified coding method. When compressor\_code is equal to 'gscu', the stream item is uncompressed, compressor\_parameter\_length shall be equal to 0, and the decoded byte sequence shall be identical to the stored byte sequence.

#### X.2.3.4 Gaussian Splat track reference property

##### X.2.3.4.1 Definition

Box type: 'gstR'

Property type: Descriptive item property

Container: ItemPropertyContainerBox

Mandatory (per item): Conditional, for a Gaussian Splat representation item of type 'gspl' that identifies one or more tracks carrying Gaussian Splat data

Quantity (per item): One for a Gaussian Splat representation item of type 'gspl' that identifies tracks

The GaussianSplatTrackReferenceProperty identifies one or more tracks in the file that carry Gaussian Splat data units for the associated Gaussian Splat representation item.

essential shall be equal to 1 for a 'gstR' item property associated with a Gaussian Splat representation item of type 'gspl'.

##### X.2.3.4.2 Syntax

aligned(8) class GaussianSplatTrackReferenceProperty

extends ItemFullProperty('gstR', version == 0, flags == 0) {

unsigned int(16) track\_count;

reserved bit(16) reserved;

unsigned int(32) track\_ID[track\_count];

}

##### X.2.3.4.3 Semantics

***track\_count*** specifies the number of track\_ID values that follow.

***track\_ID*** identifies a track by the track\_ID in the TrackHeaderBox of that track. Each referenced track shall be present in the same file and shall carry Gaussian Splat data units as specified in X.3.

When a 'gstR' item property is associated with an item of type 'gspl', the item may omit any item references of type 'gsst'. When both 'gstR' and 'gsst' are present, the identified data units shall be treated as components of the same Gaussian Splat representation.

### X.2.4 Gaussian Splat auxiliary images

#### X.2.4.1 General

The following URNs are specified for aux\_type of AuxiliaryTypeProperty in this annex:

urn:mpeg:gaussian\_splats:2026:auxid:xxx. The xxx in the URN string is the decimal string representation of an integer identifying the auxiliary image type that is equal to the AuxId value specified in Table X.2.

NOTE: The URN scheme specified in this annex follows the same pattern as the URN scheme used for HEVC auxiliary images in Annex B.

The label urn:mpeg:gaussian\_splats:2026:auxid:1 indicates a Gaussian Splat representation.

A Gaussian Splat representation auxiliary image uses the item\_type value 'gspl'.

GaussianSplatAuxConfigSubType structure specified below replaces the aux\_subtype byte array in AuxiliaryTypeProperty.

**Table X.2 –** AuxId **values for Gaussian Splat auxiliary images**

|  |  |
| --- | --- |
| **AuxId** | **Semantics** |
| 1 | Gaussian Splat representation. |

#### X.2.4.2 Syntax

aligned(8) class GaussianSplatAuxConfigSubType {

unsigned int(8) gs\_version;

unsigned int(8) gs\_flags;

reserved bit(16) reserved;

unsigned int(32) num\_points;

unsigned int(8) sh\_degree;

unsigned int(8) component\_format;

unsigned int(16) reserved2;

}

#### X.2.4.3 Semantics

***gs\_version*** specifies the version of the Gaussian Splat auxiliary configuration.

***gs\_flags*** provides flags for coordinate system and component representation. The semantics of gs\_flags are specified in X.2.5.

***num\_points*** specifies the number of points in the Gaussian Splat representation.

***sh\_degree*** specifies the maximum spherical harmonics degree present in the representation.

***component\_format*** specifies the component representation used by uncompressed stream data when compressor\_code is equal to 'gscu'. component\_format equal to 0 indicates IEEE 754 binary32. Other values are reserved for future use.

### X.2.5 Uncompressed stream data layout

The uncompressed stream data layout is role specific and is defined by the role\_code in the associated GaussianSplatRoleProperty. When a stream item contains more than one role, role\_offset and role\_length indicate the location of each role byte sequence within the decoded byte sequence of the stream item.

Unless otherwise stated in this annex, each component in the uncompressed stream data shall be represented as IEEE 754 binary32 in little endian byte order.

For role\_code values specified in Table X.1, the following layouts apply for each point, in the order of points from 0 to num\_points − 1:

* 'posi': three components (x, y, z).
* 'rotq': four components (qx, qy, qz, qw).
* 'scl3': three components (sx, sy, sz).
* 'opac': one component (opacity).
* 'shco': an array of coefficient triplets for the signalled sh\_band\_index. The number of coefficient triplets for a given band index shall be as specified by the spherical harmonics degree and the coding method.

### X.2.6 Item reference type '**gsst'**

The item reference type 'gsst' identifies Gaussian Splat stream items that belong to a Gaussian Splat representation item of type 'gspl'.

In an item reference of type 'gsst', the from\_item\_ID shall identify the Gaussian Splat representation item and the to\_item\_ID values shall identify Gaussian Splat stream items.

## X.3 Gaussian Splat image sequences

### X.3.1 General

Gaussian Splat representations may be carried as timed sequences. The specifications of Clause 7 apply.

A timed Gaussian Splat sequence may be carried in one or more tracks. When a timed Gaussian Splat sequence is present, it is recommended to provide a viewable image sequence track and to associate the Gaussian Splat sequence to the viewable track using the mechanisms specified in Clause 7 for auxiliary tracks.

NOTE: This annex is compression agnostic. Image or video coding formats may be used to carry Gaussian Splat data units in tracks, provided that the roles and the mapping to the uncompressed stream data layouts specified in X.2.5 are signaled.

### X.3.2 Gaussian Splat data tracks

#### X.3.2.1 General

Gaussian Splat data units may be carried in a Gaussian Splat data track. A Gaussian Splat data track shall be an auxiliary image sequence track with handler type 'auxv' and shall signal its auxiliary track type using an AuxiliaryTypeInfoBox ('auxi') contained in the sample entry, as specified in Clause 7.

A Gaussian Splat data track may be associated with one or more master image sequence tracks using a track reference of type 'auxl' as specified in Clause 7. A Gaussian Splat data track may also be identified by a Gaussian Splat representation item of type 'gspl' using the GaussianSplatTrackReferenceProperty ('gstR') as specified in X.2.3.4.

A Gaussian Splat data track may convey one or more roles. When a Gaussian Splat data track conveys more than one role, or when the order of roles cannot be derived from other information, the mapping of roles to samples within a Gaussian Splat access unit shall be provided by a GaussianSplatTrackRoleMapBox ('gsrm') as specified in X.3.2.1.

#### X.3.2.1 Gaussian Splat track role map box

##### X.3.2.1.1 Definition

Box type: 'gsrm'

Container: Sample entry box

Mandatory: Conditional, when a Gaussian Splat data track conveys more than one role or when an explicit mapping of roles to samples is needed

Quantity: One

The GaussianSplatTrackRoleMapBox identifies the roles conveyed by a Gaussian Splat data track and provides the mapping of each role to one or more samples within a Gaussian Splat access unit.

##### X.3.2.1.2 Syntax

aligned(8) class GaussianSplatTrackRoleMapBox

extends FullBox('gsrm', version == 0, flags == 0){

unsigned int(16) access\_unit\_sample\_count;

unsigned int(16) entry\_count;

for(i=0; i<entry\_count; i++){

unsigned int(32) role\_code;

unsigned int(16) sh\_band\_index;

reserved bit(16) reserved;

unsigned int(16) sample\_offset;

unsigned int(16) sample\_count;

unsigned int(32) role\_offset;

unsigned int(32) role\_length;

}

}

##### X.3.2.1.3 Semantics

***access\_unit\_sample\_count*** specifies the number of consecutive samples that form one Gaussian Splat access unit. When access\_unit\_sample\_count is equal to 1, each sample corresponds to one access unit.

***entry\_count*** specifies the number of role mapping entries that follow.

***role\_code*** and ***sh\_band\_index*** specify the role and, when applicable, the spherical harmonics band index, using the same semantics as specified for GaussianSplatRoleProperty in X.2.3.2.

***sample\_offset*** specifies the zero-based index of the first sample within the access unit that contributes to the role. sample\_count specifies the number of consecutive samples within the access unit that contribute to the role.

***role\_offset*** specifies the zero-based byte offset within the decoded byte sequence of each contributing sample to the first byte of the role byte sequence. role\_length specifies the number of bytes contributed by each contributing sample. When sample\_count is greater than 1, the complete role byte sequence is formed by concatenating role\_length bytes from each contributing sample in increasing sample order, starting at role\_offset.

## X.4 Gaussian Splat specific brands

### X.4.1 Gaussian Splat image and image collection brands

#### X.4.1.1 General

The brand 'gsc1' is specified in the following subclauses.

#### X.4.1.2 Requirements on files

Files shall include 'mif1' among the compatible brands and hence conform to the specifications in 10.2.1.1.

Additionally, there shall be at least one Gaussian Splat representation item of type 'gspl' that is present in the file and has an item reference of type 'auxl' to a master image item that is present in the file. The Gaussian Splat representation item shall identify Gaussian Splat data units either by one or more item references of type 'gsst', or by an associated 'gstR' item property, or both.

When the 'gsc1' brand is among the compatible brands, Gaussian Splat stream items referenced by the Gaussian Splat representation item shall be of type 'gspc'. Each stream item shall have an associated 'gscC' item property as specified in X.2.3.3. The value 'gscu' of compressor\_code indicates uncompressed stream data. If the Gaussian Splat representation item identifies one or more tracks using an associated 'gstR' item property, each referenced track may carry Gaussian Splat data units using an image or video coding format. When a referenced track conveys more than one role, a 'gsrm' box shall be present in the sample entry as specified in X.3.2.1.

#### X.4.1.3 Requirements on readers

The requirements on readers specified in 10.2.1.2 shall be supported.

Readers for the 'gsc1' brand shall be able to parse a Gaussian Splat representation item of type 'gspl' and the identified data units and reconstruct the uncompressed stream data for each referenced role. Readers shall be able to parse the 'gscC' property and obtain the compressor\_code and decoding parameters for each stream item. When compressor\_code is equal to 'gscu', no decoding is required. When compressor\_code identifies a supported coding method, a reader shall be able to decode the stream item and reconstruct the uncompressed stream data. When compressor\_code identifies an unsupported coding method, a reader may ignore the corresponding stream item. If a Gaussian Splat representation item identifies one or more tracks using an associated 'gstR' item property, a reader shall be able to locate the referenced track(s) and, when present, parse the 'gsrm' box in the sample entry to determine the mapping of roles to samples.

NOTE: A file including a Gaussian Splat representation is required to include a viewable master image item conforming to 'mif1'. A reader that does not support Gaussian Splat rendering can still display the master image.

# 4 Extracted 4CCs

|  |  |  |
| --- | --- | --- |
| 4CC | Clause | First Occurrence Paragraph |
| **auxl** | 23008-12 | A Gaussian Splat (GS) representation item is an auxiliary image item linked to the cover image through an item reference of type 'auxl'... |
| **scl3** | 23008-12 | One stream item with role 'scl3' contains per point scale values. |
| **gsrC** | X.2.2.2 Gaussian Splat stream items | A Gaussian Splat stream item ... ('gsrC') ... |
| **gspc** | X.2.3.3 Semantics | compressor\_code is a four character code identifying the compression method used for the item of type 'gspc'... |
| **gspl** | X.2.3.4 Gaussian Splat track reference property | Mandatory (per item): Conditional, for a Gaussian Splat representation item of type 'gspl'... |
| **gstR** | X.2.3.4 Syntax | extends ItemFullProperty('gstR', version = 0, flags = 0)... |
| **gsst** | X.2.3.4 Semantics | When a 'gstR' item property ... 'gsst' ... |
| **auxi** | X.3.2 Gaussian Splat data tracks | Gaussian Splat data units ... AuxiliaryTypeInfoBox ('auxi') ... |
| **gsrm** | X.3.2 Gaussian Splat data tracks | A Gaussian Splat data track ... GaussianSplatTrackRoleMapBox ('gsrm') ... |
| **gsc1** | X.4.1.1 General | The brands 'gsu1' and 'gsc1' are specified... |
| **gsu1** | X.4.1.1 General | The brands 'gsu1' and 'gsc1' are specified... |
| mif1 | X.4.1.2 Requirements on files | Files shall include 'mif1' among the compatible brands... |
| **gspu** | X.4.1.2 Requirements on files | When the 'gsu1' brand ... type 'gspu'... |
| **gscC** | X.4.1.2 Requirements on files | Each stream item of type 'gspc' ... 'gscC' ... |
| meta | 2.4 Examples | MetaBox 'meta': |
| hvc1 | 2.4 Examples | 1) infe: item\_ID=1, item\_type='hvc1'... |
| ipco | 2.4 Examples | ItemPropertyContainerBox 'ipco': |
| shb0 | 2.4 Examples | item\_ID=14: essential=1 gsrC(role='shb0') |
| shb1 | 2.4 Examples | item\_ID=15: essential=1 gsrC(role='shb1') |
| shb2 | 2.4 Examples | item\_ID=16: essential=1 gsrC(role='shb2') |
| ftyp | 2.4 Examples | FileTypeBox 'ftyp': major-brand='heic' (or 'gsc1')... |
| heic | 2.4 Examples | FileTypeBox 'ftyp': major-brand='heic'... |
| hdlr | 2.4 Examples | HandlerBox 'hdlr': 'pict' |
| pict | 2.4 Examples | HandlerBox 'hdlr': 'pict' |
| pitm | 2.4 Examples | PrimaryItemBox 'pitm': item\_ID=1 |
| iinf | 2.4 Examples | ItemInfoBox 'iinf': entry\_count=2 |
| iloc | 2.4 Examples | ItemLocationBox 'iloc': |
| iref | 2.4 Examples | ItemReferenceBox 'iref': |
| iprp | 2.4 Examples | ItemPropertiesBox 'iprp': |
| hvcC | 2.4 Examples | 'hvcC' 'ispe' // cover config and size |
| ispe | 2.4 Examples | 'hvcC' 'ispe' // cover config and size |
| auxC | 2.4 Examples | 'auxC' // identifies gaussian\_splats via aux\_type URN |
| gsgC | 2.4 Examples | 'gsgC' // GS global config |
| ipma | 2.4 Examples | ItemPropertyAssociation 'ipma': |
| moov | 2.4 Examples | MovieBox 'moov': |
| auxv | 2.4 Examples | hdlr: 'auxv' |
| posi | 2.4 Examples | entry[0]: role\_code='posi'... |
| rotq | 2.4 Examples | entry[1]: role\_code='rotq'... |
| opac | 2.4 Examples | entry[3]: role\_code='opac'... |
| shco | 2.4 Examples | entry[4]: role\_code='shco'... |
| mdat | 2.4 Examples | MediaDataBox 'mdat': |
| mp42 | 2.4 Examples | compatible\_brands: 'mif1', 'mp42'... |
| av01 | 2.4 Examples | sample\_entry: 'hvc1' (or 'av01',... |
| vvc1 | 2.4 Examples | sample\_entry: 'hvc1' (or 'av01', 'vvc1',...) |

We also checked MP4RA, and none of the proposed new 4CCs highlighted in **red bold** above are registered in MP4RA.

We also ran the SDL through the <https://mpeggroup.github.io/mpeg-sdl-editor/> validator for anz bugs.

# 5 Example

Here is a detailed example with SPZ:

FileTypeBox 'ftyp':

major\_brand = 'heic'

minor\_version = 0

compatible\_brands[] = { 'gsc1', 'mif1', 'heic', 'mp42' }

MetaBox 'meta':

FullBox header:

version = 0

flags = 0

HandlerBox 'hdlr':

version = 0

flags = 0

pre\_defined = 0

handler\_type = 'pict'

reserved[3] = 0

name = "Picture handler\0"

PrimaryItemBox 'pitm':

version = 0

flags = 0

item\_ID = 1

ItemInfoBox 'iinf':

version = 0

flags = 0

entry\_count = 3

ItemInfoEntry 'infe' (for item\_ID=1):

version = 2

flags = 0

item\_ID = 1

item\_protection\_index = 0

item\_type = 'hvc1'

item\_name = "CoverImage\0"

ItemInfoEntry 'infe' (for item\_ID=10):

version = 2

flags = 0

item\_ID = 10

item\_protection\_index = 0

item\_type = 'gspl'

item\_name = "GaussianSplatRepresentation\0"

ItemInfoEntry 'infe' (for item\_ID=11):

version = 2

flags = 0

item\_ID = 11

item\_protection\_index = 0

item\_type = 'gspc'

item\_name = "GaussianSplatStream\_SPZ\0"

ItemLocationBox 'iloc':

version = 0

flags = 0

offset\_size = 4

length\_size = 4

base\_offset\_size = 8

index\_size = 0

item\_count = 3

item\_ID = 1:

construction\_method = 0

data\_reference\_index = 0

base\_offset = 0

extent\_count = 1

extent[0]:

extent\_offset = 0x00001000 // P1

extent\_length = 0x0001F000 // Q1

item\_ID = 10:

construction\_method = 0

data\_reference\_index = 0

base\_offset = 0

extent\_count = 1

extent[0]:

extent\_offset = 0x00000000 // P10

extent\_length = 0x00000000 // Q10

item\_ID = 11:

construction\_method = 0

data\_reference\_index = 0

base\_offset = 0

extent\_count = 1

extent[0]:

extent\_offset = 0x00020000 // P11

extent\_length = 0x00009400 // Q11

ItemReferenceBox 'iref':

version = 0

flags = 0

SingleItemTypeReferenceBox 'auxl':

from\_item\_ID = 10

reference\_count = 1

to\_item\_ID[0] = 1

SingleItemTypeReferenceBox 'gsst':

from\_item\_ID = 10

reference\_count = 1

to\_item\_ID[0] = 11

ItemPropertiesBox 'iprp':

ItemPropertyContainerBox 'ipco':

property[1] = HEVCConfigurationBox 'hvcC':

configurationVersion = 1

general\_profile\_space = 0

general\_tier\_flag = 0

general\_profile\_idc = 1

general\_level\_idc = 120

nalArray[] = (HEVC config payload)

property[2] = ImageSpatialExtentsProperty 'ispe':

version = 0

flags = 0

image\_width = 1920

image\_height = 1080

property[3] = AuxiliaryTypeProperty 'auxC':

version = 0

flags = 0

aux\_type = "urn:mpeg:gaussian\_splats:2026:auxid:1\0"

aux\_subtype = GaussianSplatAuxConfigSubType:

gs\_version = 0

gs\_flags = 0x0001 // bit0 = right-handed coordinate system

reserved = 0

num\_points = 1000 // 0x000003E8

sh\_degree = 2

component\_format = 0 // float32

reserved2 = 0

property[4] = GaussianSplatGlobalConfigurationProperty 'gsgC':

version = 0

flags = 0

num\_points = 1000

sh\_degree = 2

component\_format = 0

coord\_system = 1 // 1 = right-handed

reserved = 0

property[5] = GaussianSplatCompressionConfigurationProperty 'gscC':

version = 0

flags = 0

compressor\_code = 'spz ' // 0x73707A20

compressor\_parameter\_length = 8

compressor\_parameter[8] = {

0x00, // cfg\_version = 0

0x03, // spz\_version\_policy = require v3

0x02, // input\_coord\_system = RDF

0x01, // output\_coord\_system = RUB

0x01, // flags: bit0 require\_magic = 1

0x00, 0x00, 0x00 // reserved

}

property[6] = GaussianSplatRoleProperty 'gsrC' for role posi:

version = 0

flags = 0

role\_code = 'posi'

role\_instance = 0

role\_count = 1

sh\_band\_index = 0xFFFF

reserved = 0

role\_offset = 0

role\_length = 12000 // 1000 points \* 3 float32

property[7] = GaussianSplatRoleProperty 'gsrC' for role rotq:

version = 0

flags = 0

role\_code = 'rotq'

role\_instance = 0

role\_count = 1

sh\_band\_index = 0xFFFF

reserved = 0

role\_offset = 12000

role\_length = 16000 // 1000 \* 4 float32

property[8] = GaussianSplatRoleProperty 'gsrC' for role scl3:

version = 0

flags = 0

role\_code = 'scl3'

role\_instance = 0

role\_count = 1

sh\_band\_index = 0xFFFF

reserved = 0

role\_offset = 28000

role\_length = 12000 // 1000 \* 3 float32

property[9] = GaussianSplatRoleProperty 'gsrC' for role opac:

version = 0

flags = 0

role\_code = 'opac'

role\_instance = 0

role\_count = 1

sh\_band\_index = 0xFFFF

reserved = 0

role\_offset = 40000

role\_length = 4000 // 1000 \* 1 float32

property[10] = GaussianSplatRoleProperty 'gsrC' for role shco band 0:

version = 0

flags = 0

role\_code = 'shco'

role\_instance = 0

role\_count = 1

sh\_band\_index = 0

reserved = 0

role\_offset = 44000

role\_length = 12000 // band0 has 1 coeff triplet per point, 3 float32 per triplet

property[11] = GaussianSplatRoleProperty 'gsrC' for role shco band 1:

version = 0

flags = 0

role\_code = 'shco'

role\_instance = 0

role\_count = 1

sh\_band\_index = 1

reserved = 0

role\_offset = 56000

role\_length = 36000 // band1 has 3 coeff triplets per point

property[12] = GaussianSplatRoleProperty 'gsrC' for role shco band 2:

version = 0

flags = 0

role\_code = 'shco'

role\_instance = 0

role\_count = 1

sh\_band\_index = 2

reserved = 0

role\_offset = 92000

role\_length = 60000 // band2 has 5 coeff triplets per point

ItemPropertyAssociation 'ipma':

version = 0

flags = 0

entry\_count = 3

entry for item\_ID=1:

association\_count = 2

assoc[0]:

essential = 1

property\_index = 1 // hvcC

assoc[1]:

essential = 0

property\_index = 2 // ispe

entry for item\_ID=10:

association\_count = 2

assoc[0]:

essential = 1

property\_index = 3 // auxC

assoc[1]:

essential = 1

property\_index = 4 // gspC

entry for item\_ID=11:

association\_count = 8

assoc[0]:

essential = 1

property\_index = 5 // gscC compressor\_code='spz '

assoc[1]:

essential = 1

property\_index = 6 // gsrC posi

assoc[2]:

essential = 1

property\_index = 7 // gsrC rotq

assoc[3]:

essential = 1

property\_index = 8 // gsrC scl3

assoc[4]:

essential = 1

property\_index = 9 // gsrC opac

assoc[5]:

essential = 1

property\_index = 10 // gsrC shco band0

assoc[6]:

essential = 1

property\_index = 11 // gsrC shco band1

assoc[7]:

essential = 1

property\_index = 12 // gsrC shco band2

MediaDataBox 'mdat':

at file offset 0x00001000:

item\_ID=1 bytes, length 0x0001F000

at file offset 0x00020000:

item\_ID=11 bytes, length 0x00009400, contents:

gzip stream that is an SPZ bitstream

SPZ header and payload are opaque to HEIF

decompression produces a decoded byte sequence of length 152000 bytes

the role\_offset and role\_length in gsrC point into that decoded sequence

# 6 Discussion during MPEG#153

During MPEG#153, online [and](https://git.mpeg.expert/MPEG/Systems/FileFormat/HEIF/-/issues/215) gitlab based discussions happened. The gitlab discussions are here.

A summary of the questions are as follows

## A. Conceptual Scope & Placement

1. Should GS be part of HEIF or be restricted to volumetric media standards (23090‑10 / 18)?
2. Does treating GS as an “image” stretch HEIF beyond its intended scope?
3. If both domains support GS, how do we avoid fragmentation and ensure interoperability?

## B. Technical Model & Semantics

1. Should GS representations really be modeled as auxiliary image items?
   * If so, what are the semantics of image dimensions (ispe) for 3D content?
2. Is the use of role\_offset and role\_length appropriate or overly constraining?
   * Does it assume too much about the decoded GS structure?
3. Should track referencing (gstR) be permitted within image items, or is this mixing incompatible mechanisms?

## C. Compatibility with Existing Codecs

1. How to support GS codecs that internally use video/image coding formats (e.g., SOG) without duplicating existing signaling?
2. Should new 4CCs be defined for codecs that already have registered sample entries?

## D. Real‑World Deployment Concerns

1. Are current GS codecs mature enough to define a generic long‑term container?
2. Do GS formats share enough structural commonality to justify a unified HEIF‑based framework?

## E. Spec Evolution & Clarity

1. Should the definitions of gspl, gspc, gscC, gsrC, etc., be simplified or aligned more closely with ISOBMFF semantics?
2. Is more alignment needed with existing HEIF “image item” and “coded image” concepts?

# 7 References

[1] ISO/IEC 23008-12:2024, Information technology - High efficiency coding and media delivery in heterogeneous environments - Part 12: Image File Format.

[2] ISO/IEC 14496-12:2022, Information technology - Coding of audio-visual objects - Part 12: ISO base media file format.

[3] RFC 3986, Uniform Resource Identifier (URI): Generic Syntax.

[4] Kerbl et al., “3D Gaussian Splatting for Real-Time Radiance Field Rendering”