

ICS 35.040
CCS L 71

团 标 准

T/AI 128. 1-2025

信息技术 时空图形数据编码

第 1 部分：系统

Information Technology Space - Time Graphical Data Coding -

Part1: System

T/AI
128.1
Part1: System

2025-11-19 发布

2025-11-19 实施

中 关 村 视 听 产 业 技 术 创 新 联 盟 发 布

T/AI 128.1-2025

T/AI 128·1-2025



版权保护文件

版权所有归属于该标准的发布机构，除非有其他规定，否则未经许可，此发行物及其章节不得以其他形式或任何手段进行复制、再版或使用，包括电子版，影印件，或发布在互联网及内部网络等。使用许可可于发布机构获取。

T/AI 128.1-2025

目次

前言	II
引言	III
1 范围	1
2 规范性引用文件	1
3 术语和定义	1
4 缩略语	2
5 架构	2
6 容积媒体通用格式	3
7 点云数据的文件封装	4
8 基于 SMT 的点云数据传输	33
9 基于 DASH 的点云数据传输	42
附录 A (规范性) 文件标牌	48
附录 B (资料性) 可替换位流	49

前言

本文件按照GB/T 1.1—2020《标准化工作导则 第1部分：标准化文件的结构和起草规则》的规定起草。

本文件是T/AI 128《信息技术 时空图形数据编码》的第1部分。T/AI 128已经发布了以下部分：

——第1部分：系统；

——第2部分：点云。

本文件由数字音视频编解码技术标准（AVS）工作组提出。

本文件由中关村视听产业技术创新联盟归口。

本文件起草单位：中兴通讯股份有限公司、上海交通大学、腾讯科技（深圳）有限公司、北京工业大学、北京大学、广东博华超高清创新中心有限公司、马栏山音视频实验室、中国移动通信集团有限公司。

本文件主要起草人：黄成、徐异凌、白雅贤、胡颖、朱文婕、李秋婷、牟伦田、许晓中、刘杉、侯朴玥、王一帆、杨开发、李岳、侯礼志、赵海英、张伟民、黄铁军、高文、郭勐、杨蕾。

T/AI 128.1-2025

引言

T/AI 128《信息技术 时空图形数据编码》旨在面向自动驾驶、兼顾数字博物馆及虚拟现实等行业应用中的点云数据，支持点云数据的高效编码、封装和传输，拟由6个部分构成。

- 第1部分：系统。目的在于定义点云编码码流文件格式、传输格式与传输信令规范。
- 第2部分：点云。目的在于定义高效的点云解码过程和解码规范。
- 第3部分：点云主观质量评价方法。目的在于定义点云的主观质量评价方法。
- 第5部分：符合性测试。目的在于提供点云解码符合性测试描述。
- 第6部分：参考软件。目的在于提供符合规范的点云解码参考软件和点云编码参考软件。

本文件是T/AI 128的第1部分。本文件第6章和第7章规定了点云编码数据的基本文件封装格式，第8章和第9章规定了基于第6章和第7章的点云文件封装格式进行SMT传输和DASH传输的技术要求。

本文件的发布机构提请注意，声明符合本文件时，可能涉及如下13项与时空图形编解码技术相关的专利的使用。专利申请号及名称如下：

序号	专利申请号	专利名称
1	CN115039132B	Three-dimensional content processing method and device
2	CN202110015204.2	三维点云数据处理方法、装置、存储介质及电子装置
3	CN202410216859.X	点云数据的处理方法、装置及存储介质
4	CN202110441806.4	一种三维点云的封装方法、装置及介质
5	CN202010026089.4	点云数据封装方法及传输方法
6	CN202011407418.6	数据处理方法、装置、通信节点和存储介质
7	CN201810760047.6	基于视频内容消费的反馈信息标识方法、系统及存储介质
8	CN202010301699.0	多媒体数据收发方法、系统、处理器和播放器
9	CN201710557241.X	基于用户消费信息的视频处理方法及装置
10	CN201811108139.2	沉浸媒体内容随用户移动变化的方法及系统
11	CN202210238075.8	一种三维点云的收发方法及装置
12	CN202310642452.9	一种三维媒体的收发方法及装置

本文件的发布机构对上述专利的真实性、有效性和范围无任何立场。

上述专利持有人已向本文件的发布机构保证，愿意同任何申请人在合理且无歧视的条款和条件下，就专利授权许可进行谈判。上述专利持有人的声明已在本文件的发布机构备案，相关信息可以通过以下联系方式获得：

联系人：黄铁军（数字音视频编解码技术标准工作组秘书长）

通讯地址：北京大学理科2号楼2641室

邮政编码：100871

电子邮件：tjhuang@pku.edu.cn

电话：+8610-62756172

传真：+8610-62751638

网址：<http://www.avc.org.cn>

请注意除上述专利外，本文件的某些内容仍可能涉及专利。本文件的发布机构不承担识别专利的责任。

T/AI 128.1-2025

信息技术 时空图形数据编码 第 1 部分：系统

1 范围

本文件给出了基于时空图形数据编码的点云数据的系统架构，规定了基于时空图形数据编码的点云数据的媒体通用格式、文件封装格式、基于 SMT 的传输信令与基于 DASH 的传输信令。

本文件适用基于时空图形数据编码的点云数据的自主导航系统、实时巡检系统、地理信息系统、数字文化遗产、自由视点广播、三维沉浸通信、三维沉浸交互等应用。

2 规范性引用文件

下列文件中的内容通过文中的规范性引用而构成本文件必不可少的条款。其中，注日期的引用文件，仅该日期对应的版本适用于本文件；不注日期的引用文件，其最新版本（包括所有的修改单）适用于本文件。

GB/T 17969.1-2015 信息技术 开放系统互连 OSI 登记机构的操作规程 第 1 部分：一般规程和国际对象标识符树的顶级弧 (Information technology—Open systems interconnection—Procedures for the operation of OSI registration authorities—Part 1: General procedures and top arcs of the International Object Identifier tree)

T/AI 114-2024 信息技术 高效多媒体编码 第 6 部分：智能媒体传输

T/AI 128.2-2024 信息技术 时空图形数据编码 第2部分：点云

ISO/IEC 14496-12 信息技术 音视频对象的编码 第 12 部分：ISO 基本媒体文件格式 (Information technology – Coding of audio-visual objects – Part 12: ISO base media file format)

ISO/IEC 23008-12 信息技术 MPEG 系统技术 第 12 部分：图片文件格式 (Information technology – MPEG systems technologies – Part 12: Image File Format)

ISO/IEC 23009-1 信息技术 基于 HTTP 的动态自适应流媒体 第 1 部分：媒体呈现描述和片段格式 (Information technology – Dynamic adaptive streaming over HTTP (DASH) – Part 1: Media presentation description and segment formats)

IEEE 754 浮点数算术标准 (Standard for Floating-Point Arithmetic)

3 术语和定义

下列术语和定义适用于本文件。

3.1

AVS PCC 数据 AVS PCC data

符合T/AI 128.2-2024的编码点云数据所形成的二进制数据流。

3.2

点云 point cloud

一组无序的点的集合，表达三维物体或场景的空间结构及表面属性。

3.3

点云轨道 track

封装点云编码数据的轨道。

3.4

点云项 item

数据项，表示静态的点云数据。

3.5

点云样本 sample

点云轨道的一个样本，包含一个点云帧。

3.6

片 slice

一系列的点云编码语法元素，用于表示一部分或者整个点云。

3.7

数据项 item

不需要定时呈现的媒体数据，即时序媒体数据。

3.8

组件数据 component

构成点云片的部分点云编码数据，分为几何组件数据和属性组件数据。

4 缩略语

下列缩略语适用于本文件。

AVS	音视频编解码标准	(audio video coding standard)
DASH	基于 HTTP 的动态自适应流媒体	(dynamic adaptive streaming over HTTP)
HTTP	超文本传输协议	(hyper text transfer protocol)
ISOBMFF	ISO 基本媒体文件格式	(ISO base media file format)
MPD	媒体表示描述	(media presentation description)
OID	对象标识符	(object identifier)
SMT	智能媒体传输	(smart media transport)
SMTP	智能媒体传输协议	(SMT Protocol)
URI	统一资源标识符	(uniform resource identifier)
URL	统一资源定位器	(uniform resource locator)
URN	统一资源名称	(uniform resource name)
UTC	协调世界时间	(coordinated universal time)
XML	可扩展标记语言	(extensible mark-up language)
3DoF	三自由度	(three degree of freedom)
3DoF+	增强的三自由度	(three degree of freedom plus)

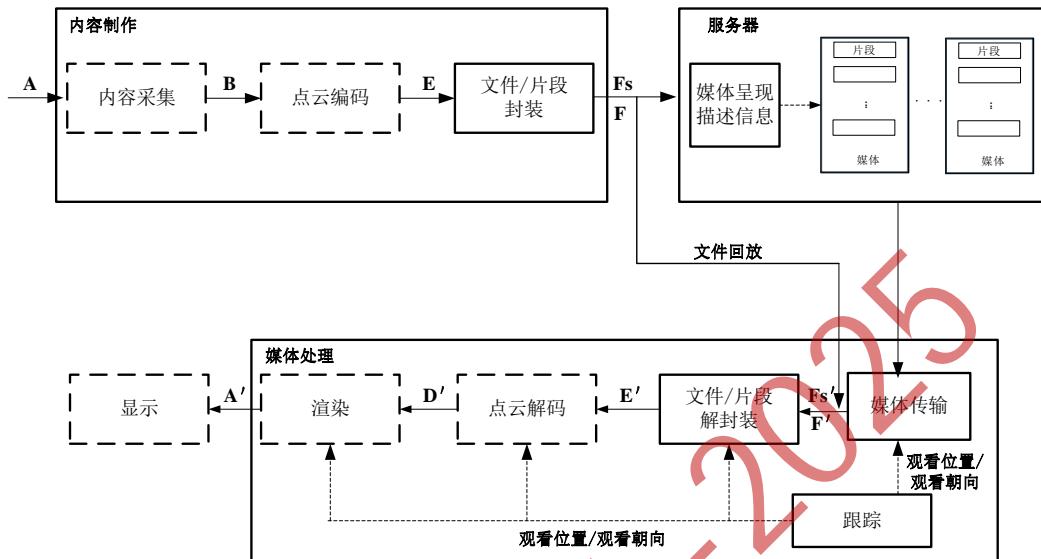
5 架构

5.1 概述

点云编码数据由连续的点云帧表示，每个点云帧包含大量的点，每个点包含它某一时间点中在三维空间中的位置，以及相关的属性。本文件对基于时空图形数据编码的点云数据的系统进行研究，其规范性范畴包括：AVS PCC 数据的文件封装格式、AVS PCC 数据的传输格式与传输信令。

5.2 AVS PCC 系统流程

AVS PCC 系统流程见图 1。



注：其中实线模块表示相关内容在本标准规定范围内，虚线模块表示不在本标准规定范围内。

图 1 AVS PCC 系统流程图

真实世界的视觉场景 A 由一组相机或一个具有多个镜头和传感器的相机设备捕获。采集结果为点云源数据 B，点云源数据 B 是由大量点云帧组成的帧序列。一个或多个点云帧被编码为编码的点云位流 E，包括编码的几何数据和属性数据。根据特定的媒体容器文件格式（如 ISOBMFF）进行封装处理，一个或多个编码的位流被组合成用于流化传输的初始化片段和媒体片段的序列 (Fs) 或用于文件回放的媒体文件 (F)。同时，文件封装还将元数据包含到文件 F 或媒体片段中 Fs，使用传输机制将片段 Fs 传送给播放器。

文件解封装处理接收到的文件 F'或片段 Fs'，提取编码的位流 E'并解析元数据，然后再解码生成点云数据 D'。媒体处理过程中，根据当前的观看位置、观看方向或由各种类型的传感器（例如头部、位置或眼动跟踪传感器）确定的视窗，将点云数据渲染并显示到头戴式显示器或任何其他显示设备的屏幕上。通过当前观看位置或观看方向来部分地访问、解码点云数据，可用于优化媒体处理过程。在基于视窗的传输过程中，当前的观看位置和观看方向也会被传递给策略模块，用于确定要接收的媒体轨道。

6 容积媒体通用格式

6.1 容积视频通用数据盒

AVS PCC 数据的封装应采用基于容积视频的通用数据盒，AVS PCC 数据的文件标牌应符合附录 A 的要求。

对于容积视频的媒体轨道，应使用 ISO/IEC 14496-12 标准中规定的 MediaBox 的 HandlerBox 中的处理程序类型 (handler_type) 来标识，容积视频的 handler_type 为 'volv'。同时，也应包含一个本文件规定的容积视频媒体头。

6.2 容积视频媒体头

6.2.1 定义

数据盒类型: 'vvhd'

数据盒容器: MediaInformationBox

强制性: 强制

数量: 有且仅有 1 个

容积视频媒体轨道的 MediaInformationBox 应符合 ISO/IEC 14496-12 标准规定, 其中应包含 VolumetricVisualMediaHeaderBox。

6.2.2 语法

```
aligned(8) class VolumetricVisualMediaHeaderBox extends FullBox('vvhd', version = 0, 1) {
    unsigned int(8) application_media_type;
}
```

6.2.3 语义

version 指示本数据盒的版本类型。

application_media_type 指示应用场景使用的媒体数据类型, 具体取值见表 1。

表 1 媒体数据类型

取值	含义
0	多视图视频
1	基于视频编码的点云数据
2	基于基于时空图形数据编码的 AVS 点云数据
3~255	保留

6.3 容积视频样本入口

6.3.1 定义

容积视频轨道的样本入口应使用 VolumetricVisualSampleEntry。

6.3.2 语法

```
class VolumetricVisualSampleEntry(codingname) extends SampleEntry (codingname){
    unsigned int(8)[32] compressorname;
}
```

7 点云数据的文件封装

7.1 动态点云数据的存储

7.1.1 点云数据通用数据盒和数据结构

7.1.1.1 点云数据解码器配置数据结构

7.1.1.1.1 定义

点云解码器配置指示了基于 T/AI 128.2-2024 的点云数据的解码器配置信息, 用于点云解码器的进一步配置和初始化。如果版本号无法识别, 文件阅读器不应尝试解码此配置或其所适用的位流。文件阅读器应忽略相关标准定义之外的无法识别的数据。

profile_id、level_id 的值应适用于在解码此配置所描述的位流时激活的所有参数集。具体而言, 适用以下限制:

`profile_id` 应指示与该配置相关的位流符合的配置文件。

`level_id` 应指示等于或大于所有参数集中的最高层指示的最高级别的能力级别。

7.1.1.1.2 语法

```
aligned(8) class AVSPCCDecoderConfigurationRecord() {
    unsigned int(8) configurationVersion = 1;
    unsigned int(4) profile_id;
        bit(4) reserved = 0;
    unsigned int(8) level_id;
    unsigned int(4) frame_rate_code;
        bit(4) reserved = 0;
    sequence_header PCC_header_configuration;
}
```

7.1.1.1.3 语义

`configurationVersion` 指示解码器配置记录版本号。

`profile_id` 指示点云数据的档次，应符合 T/AI 128.2-2024 中的规定。

`level_id` 指示点云数据的级别，应符合 T/AI 128.2-2024 中的规定。

`frame_rate_code` 指示点云数据的帧率，应符合 T/AI 128.2-2024 中的规定。

`PCC_header_configuration` 指示点云数据序列头配置信息，应符合 T/AI 128.2-2024 中的规定。

7.1.1.2 点云解码配置数据盒

7.1.1.2.1 定义

数据盒类型: '`pccC`'

包含于: 样本入口

强制性: 强制包含于'`apce`'、'`apeb`'或'`apcb`'类型的点云轨道

数量: 1 个

点云解码器配置数据盒包括 7.1.1.1 中规定的 `AVSPCCDecoderConfigurationRecord`。

在本文件中，数据盒的版本号为 0。

7.1.1.2.2 语法

```
class AVSPCCConfigurationBox extends FullBox('pccC', version = 0, 0) {
    AVSPCCDecoderConfigurationRecord() PCCConfig;
}
```

7.1.1.2.3 语义

`AVSPCCDecoderConfigurationRecord` 在 7.1.1.1 中规定。

7.1.1.3 组件信息数据盒

7.1.1.3.1 定义

数据盒类型: '`acif`'

包含于: 样本入口

强制性: 非强制

数量: 0 个或 1 个

组件信息数据盒指示 AVS PCC 的媒体组件的数据类型，即几何、属性等。当该数据盒包含于轨道的样本入口时，该数据盒指示对应轨道中携带的 AVS PCC 的媒体组件的数据类型。该数据盒同时提供

属性组件轨道中属性数据相关的信息。

当 AVS PCC 位流以单轨形式存储时，其样本入口中不应包含组件信息数据盒。

7.1.1.3.2 语法

```
aligned(8) class AVSPCCComponentInfoBox extends FullBox('acif', version = 0, 0) {
    unsigned int(4) avs_pcc_type;
    bit(4) reserved = 0;
    if(avs_pcc_type == 4) {
        unsigned int(4) attr_type_num;
        unsigned int(1) attr_group_info_flag;
        bit(3) reserved = 0;
        for(i=0; i<attr_type_num; i++){
            unsigned int(4) attr_type;
            bit(4) reserved = 0;
        }
        if(attr_group_info_flag == 1){
            unsigned int(8) attr_group_id;
        }
    }
}
```

7.1.1.3.3 语义

`avs_pcc_type` 指示轨道中 AVS PCC 的媒体组件的数据类型，取值见表 2。

表 2 组件类型

取值	含义
0, 1	保留
2	几何数据
3	保留
4	属性数据
5..31	保留

`attr_type_num` 指示轨道中包含的属性组件对应的属性类型个数。

`attr_type` 指示轨道中包含的属性组件的类型。取值为 0 表示包含颜色属性；取值为 1 表示包含反射率属性。

`attr_group_info_flag` 取值为 0 时表示不指示属性组信息；取值为 1 时表示指示属性组信息，且当前组件轨道包含的属性组件属于对应属性组。

`attr_group_id` 指示属性组的标识符。同属于一个属性组的不同属性数据存在编解码或者呈现上的依赖关系。

7.1.1.4 跨属性依赖信息样本群组

7.1.1.4.1 定义

样本群组类型：'casg'

包含于： Sample Group Description Box ('sgpd')

强制性：非强制

数量：0个，1个或多个

跨属性依赖信息样本群组用于指示轨道内样本包含或对应的属性数据的跨属性依赖信息。

当跨属性依赖信息样本群组位于'apcc'类型的属性组件轨道中时，CrossAttrDependencyInfoEntry 对应的样本应符合相应的跨属性依赖关系。

当跨属性依赖信息样本群组位于'apeb'或'apcb'类型的点云片基础轨道中时，CrossAttrDependencyInfoEntry 对应的样本所关联索引到的点云片轨道中的样本应符合相应的跨属性依赖关系。

若轨道中某个样本包含或对应的属性数据不存在跨属性依赖，则该样本的 group_description_index 应取值为 0。

7.1.1.4.2 语法

```
class CrossAttrDependencyInfoEntry extends VolumetricVisualSampleGroupEntry('casg') {
    unsigned int(8) dependent_attr_type;
}
```

7.1.1.4.3 语义

dependent_attr_type 指示被依赖的属性数据类型。取值为 0 表示颜色属性被反射率属性依赖；取值为 1 表示反射率属性被颜色属性依赖。

7.1.1.5 参数头信息样本群组

7.1.1.5.1 定义

数据盒类型：'appi'

包含于：Sample Group Description Box ('sgpd')

强制性：非强制

数量：0个，1个或多个

AVS PCC 参数头信息样本群组对包含点云序列头、几何头、属性头的点云样本进行标识，该样本群组可以包含于样本入口类型为'apce'，'apcc'，'apeb'或'apcb'的点云轨道中，不应包含于样本入口类型为'apsl'的点云轨道中。

当点云轨道的样本中不包含任何参数头时，该样本在群组类型为'appi'的 SampleToGroupBox 中的 group_description_index 应取值为 0。

7.1.1.5.2 语法

```
aligned(8) class AVSPCCHHeaderInfoEntry() extends VolumetricVisualSampleGroupEntry ('appi') {
```

7.1.2 点云数据的单轨封装

7.1.2.1 概述

单轨封装模式下，使用 ISOBMFF 数据盒将点云位流封装在单个点云轨道，此时无需对点云位流进行额外处理。

点云的几何数据和属性数据封装在单个点云轨道中时的结构示例见图 2。

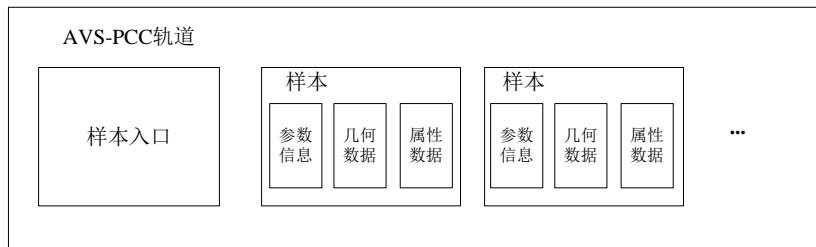


图 2 点云数据单轨封装结构

7.1.2.2 样本入口

样本入口类型: 'apce'

包含于: SampleDescriptionBox

强制性: 单轨封装模式下, 强制包含'apce'类型的样本入口

数量: 1 个或多个

单轨封装模式下, 点云轨道应使用类型为'apce'的样本入口。'apce'类型的样本入口应包含 7.1.1.2 中规定的 AVSPCCConfigurationBox。

7.1.2.2.1 语法

```
aligned(8) class AVSPCCSampleEntry() extends VolumetricVisualSampleEntry ('apce') {
    AVSPCCConfigurationBox;
}
```

7.1.2.3 样本格式

7.1.2.3.1 定义

每一个点云样本包含一个完整的点云帧, 可能包含一个序列头、一个几何头、一个属性头 (存在属性数据时) 和一组序列级别的用户数据。一个完整的点云帧包含一个点云帧头和对应同一呈现时间的一个或多个点云片数据, 可能包含一组帧级别的用户数据。

7.1.2.3.2 语法

```
aligned(8) class AVSPCCSample () {
    AVSPCC_sequence_header; //可选
    AVSPCC_geometry_header; //可选
    AVSPCC_attribute_header; //可选
    AVSPCC_userdata; //可选
    AVSPCC_frame;
}
```

7.1.2.3.3 语义

AVSPCC_sequence_header 指示序列头数据, 具体数据结构应符合 T/AI 128.2-2024 中相关规定。点云样本中可能包含一个序列头。

AVSPCC_geometry_header 指示几何头数据, 具体数据结构应符合 T/AI 128.2-2024 中相关规定。点云样本中可能包含一个几何头。

AVSPCC_attribute_header 指示属性头数据, 具体数据结构应符合 T/AI 128.2-2024 中相关规定。当点云位流中包含属性数据时, 点云样本中可能包含一个属性头。

AVSPCC_userdata 指示用户数据，具体数据结构应符合 T/AI 128.2-2024 中相关规定。点云样本中可能包含一组序列级别的用户数据。

AVSPCC_frame 指示点云帧数据，具体点云帧数据结构应符合 T/AI 128.2-2024 中相关规定。

7.1.2.4 子样本

每个点云样本可被划分为一个或多个点云子样本，子样本结构应符合 ISO/IEC 14496-12 标准的规定，并使用 SubSampleInformationBox 的标志字段（flags）指示子样本信息的类型，包括：

——0：基于点云数据类型的子样本。一个子样本只包含一种类型，该类型由 AVSPCCPayloadType 表示。

——1：基于点云片的子样本。一个子样本仅包含一个点云片的相关信息。当相应轨道包含组件信息数据盒时，相应轨道的子样本中仅包含相应组件信息数据盒对应的组件数据。当相应轨道不包含组件信息数据盒时，相应轨道的子样本包含所有的组件数据。

——保留其他标志字段取值。

SubsampleInformationBox 的 codec_specific_parameters 字段语法如下：

```
if(flags == 0){
    unsigned int(4) AVSPCCPayloadType;
    if(AVSPCCPayloadType == 6){
        unsigned int(4) attr_type;
        unsigned int(1) attr_group_flag;
        if(attr_group_flag == 1){
            unsigned int(8) attr_group_id;
            bit(15) reserved = 0;
        }
        else{
            bit(23) reserved = 0;
        }
    }
    else{
        bit(28) reserved = 0;
    }
}
else if(flags == 1){
    unsigned int(1) slice_data_flag;
    if(slice_data == 1){
        unsigned int(24) slice_id;
        bit(7) reserved = 0;
    }
    else{
        bit(31) reserved = 0;
    }
}
```

AVSPCCPayloadType 指示子样本中包含点云的数据类型，取值含义见表 3。

表 3 点云数据类型

取值	含义
0	序列头
1	几何头
2	用户数据
3	点云帧头
4	几何组件数据
5	属性头
6	属性组件数据
7..31	保留

attr_type 指示子样本中包含的属性数据的类型。取值为 0 表示颜色属性；取值为 1 表示反射率属性。

attr_group_flag 取值为 0 时表示子样本不属于属性组；取值为 1 时表示子样本属于属性组。

attr_group_id 指示属性组的标识符。同属于一个属性组的不同属性数据存在编解码或者呈现上的依赖关系。

slice_data_flag 指示子样本中是否包含点云片的数据，取值为 1 时表示包含点云片几何和/或属性类型数据；值为 0 时表示包含点云片的参数信息。

slice_id 指示子样本中包含的数据对应的点云片的标识。

7.1.3 基于组件的多轨封装

7.1.3.1 概述

当 AVS PCC 位流通过多轨方式进行文件封装时，位流中的不同组件被封装到多个点云轨道，即组件轨道，封装结构见图 3。组件轨道分为几何轨道和属性轨道，组件轨道中的每个样本包含一个或多个同类型的点云组件。

多轨封装模式下，组件轨道需符合如下约束：

- a) 应包含 1 个几何组件轨道且该几何组件轨道作为接入点。
- b) 可能包含 0 个或多个属性组件轨道，组件轨道中的 track_in_movie 字段应置为 0。
- c) 每个组件轨道的样本入口中，需包含一个组件信息数据盒 AVSPCCComponentInfoBox，用以指示组件轨道中包含的点云组件数据类型。
- d) 几何组件轨道通过轨道索引关联索引至相应的属性组件轨道。
- e) 存在编解码依赖关系的多个属性组件应包含于同一个属性组件轨道。

同一 AVS PCC 序列的不同组件轨道之间时序对齐。对应同一点云帧的不同组件轨道样本应具备相同的呈现时间。当轨道样本中存在参数集时，参数集的解码时间应等于或者早于相应的 AVS PCC 组件数据的解码时间。当所有组件轨道的所有参数集均包含于轨道样本中时，包含序列头参数集的样本应等于或者早于包含几何头参数集或属性头参数集的样本。此外，同一 AVS PCC 序列的所有组件轨道应具备相同的隐含或明确指示的编辑列表（编辑列表的使用，应符合 ISO/IEC 14496-12 标准中规定的 EditListBox）。

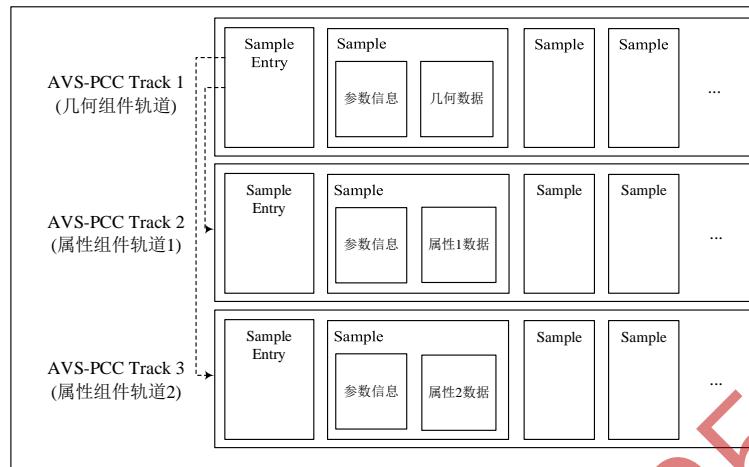


图3 基于组件的点云数据多轨道封装结构

7.1.3.2 样本入口

7.1.3.2.1 定义

样本入口类型: 'apcc'

包含于: SampleDescriptionBox

强制性: 基于组件的多轨封装模式下, 强制包含'apcc'类型的样本入口

数量: 1个或多个

基于组件的多轨封装模式下, 点云轨道应使用类型为'apcc'的样本入口, 此时点云轨道也被称为组件轨道。

组件轨道的样本入口中应包含一个 7.1.1.3 规定的组件信息数据盒 AVSPCCComponentInfoBox。根据组件信息数据盒中指示的媒体组件的数据类型, 分为几何组件轨道和属性组件轨道。其中, 几何组件轨道中还应包含一个 7.1.1.2 规定的 AVSPCCConfigurationBox, 属性组件轨道中不应包含 AVSPCCConfigurationBox。

7.1.3.2.2 语法

```
aligned(8) class AVSPCCSampleEntry() extends VolumetricVisualSampleEntry ('apcc') {
    AVSPCCConfigurationBox; //可选
    AVSPCCComponentInfoBox;
}
```

7.1.3.3 样本格式

基于组件的多轨封装模式下, 所有组件轨道中每个样本都对应一个点云帧, 且对应同一个点云帧的样本应具备相同的呈现时间。

几何组件轨道的样本中包含一个点云帧头和对应同一呈现时间的一个或多个点云片几何数据, 可能包含一个序列头、一个几何头和一组帧级别的用户数据。

属性组件轨道的样本中包含对应同一呈现时间的一个或多个点云片属性数据, 可能包含一个属性头。当几何组件轨道通过轨道索引关联至多个属性组件轨道时, 若属性组件轨道样本中包含属性头, 则该属性头应重复包含于每个属性组件轨道的相应样本中。

7.1.3.4 子样本

点云组件轨道的子样本应符合 7.1.2.4 的规定。其中，标志字段 (flags) 的值为 0 时，几何组件轨道子样本的 AVSPCCPayloadType 取值应为 0~4；属性组件轨道子样本的 AVSPCCPayloadType 取值应为 5~6。

7.1.3.5 轨道索引

在多轨封装模式下，几何组件轨道和属性组件轨道之间需使用 ISO/IEC 14496-12 规定的 TrackReferenceTypeBox 进行关联索引。

几何组件轨道的 TrackBox 中应包含一个 reference_type 为 'apca' 的 TrackReferenceTypeBox，用于关联索引至一个或多个属性组件轨道。

当不同属性组件轨道之间存在单向的跨属性依赖关系时，依赖其他属性的属性组件轨道的 TrackBox 中应包含一个 reference_type 为 'cadr' 的 TrackReferenceTypeBox，用于关联索引至其解码时依赖的属性组件轨道。

7.1.4 基于点云片的多轨封装

7.1.4.1 概述

当 AVS PCC 位流中包含多个点云片时，AVS PCC 位流可以基于点云片封装为多个轨道，包括点云片基础轨道和点云片轨道。

在基于点云片的多轨封装模式下，点云片基础轨道和点云片轨道需符合如下约束：

- a) 应包含一个点云片基础轨道，该点云片基础轨道包含点云位流中的所有序列头、几何头、属性头（存在属性数据时）、点云帧头参数集以及所有的用户数据（存在序列级别或帧级别用户数据时），且该点云片基础轨道作为接入点。
- b) 包含一个或多个点云片轨道。
- c) 点云片基础轨道通过轨道索引关联索引至相应的点云片轨道。

当点云片的不同组件通过不同点云片轨道进行封装时，必然存在一个或多个包含几何组件数据的点云片轨道，可能存在 0 个，1 个或多个包含属性组件数据的点云片轨道。且包含属性组件数据的点云片轨道的 track_in_movie 字段应取值为 0。同时，存在编解码依赖关系的多个属性组件应包含于同一个包含属性组件数据的点云片轨道。

当一个点云片轨道的点云片中包含所有组件数据时，基于点云片的多轨道封装结构见图 4。

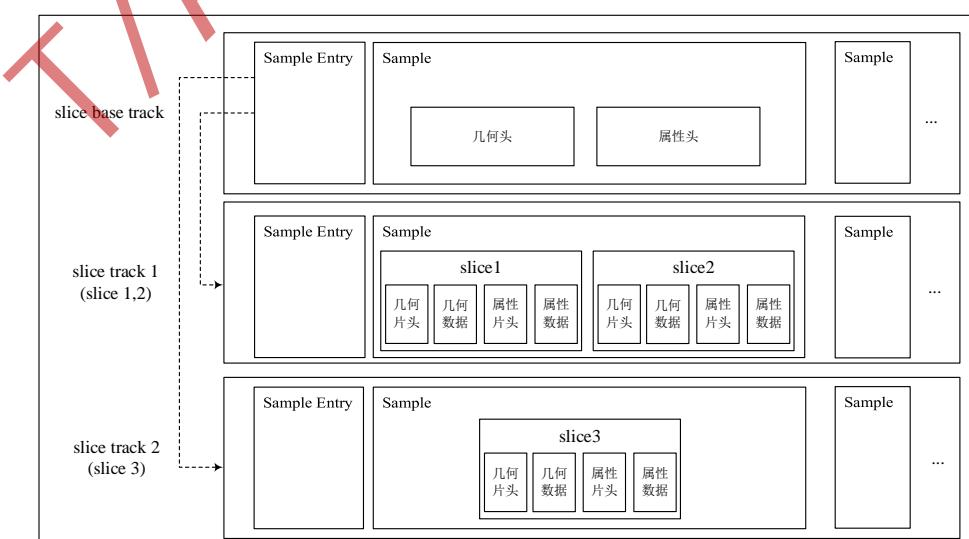


图 4 包含全部组件数据的基于点云片的多轨道封装结构

当一个点云片轨道的点云片中包含部分组件数据时，基于点云片的多轨道封装结构见图 5。

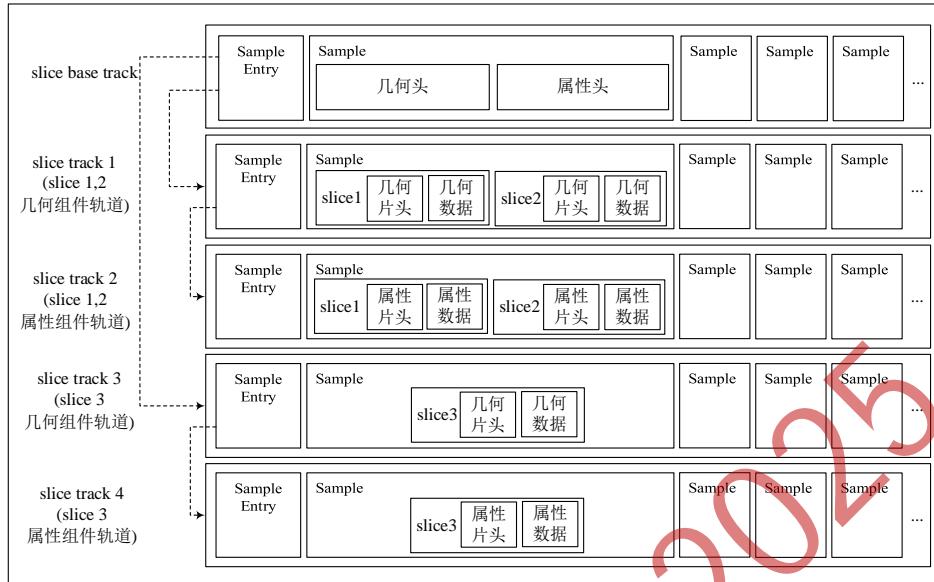


图 5 包含部分组件数据的基于点云片的多轨道封装结构

7.1.4.2 点云片基础轨道

7.1.4.2.1 样本入口定义

样本入口类型：'apeb'或'apcb'

包含于：SampleDescriptionBox

强制性：基于点云片的多轨封装模式下，强制包含'apeb'或'apcb'类型的样本入口

数量：1 个

基于点云片的多轨封装模式下，点云轨道应使用类型为'apeb'或'apcb'的样本入口，此时点云轨道也被称为点云片基础轨道。

点云片基础轨道样本入口中应包含一个 7.1.1.2 规定的 AVSPCCConfigurationBox，且不应包含 AVSPCCComponentInfoBox。

当点云片轨道包含点云片对应的所有组件数据时，点云片基础轨道的样本入口类型为'apeb'。当点云片轨道仅包含点云片对应的部分组件数据时，点云片基础轨道的样本入口类型为'apcb'。

点云片基础轨道的样本入口中可以包含一个静态空间区域数据盒 PCC3DSpatialRegionBox。当点云片基础轨道的样本入口中不包含 PCC3DSpatialRegionBox 时，点云片基础轨道应关联一个动态空间区域信息元数据轨道。

7.1.4.2.2 样本入口语法

```
aligned(8) class AVSPCCSampleEntry() extends VolumetricVisualSampleEntry ('apeb'或'apcb') {
    AVSPCCConfigurationBox;
}
```

7.1.4.2.3 样本格式

每一个点云片基础轨道的样本包含一个点云帧头，可能包含一个序列头、一个几何头、一个属性头（存在属性数据时）、一个点云帧头以及用户数据（存在序列级别或帧级别用户数据时），且不应包含任何点云几何或属性组件数据。

7.1.4.3 点云片轨道

7.1.4.3.1 样本入口定义

样本入口类型: 'apsl'

包含于: SampleDescriptionBox

强制性: 基于点云片的多轨封装模式下, 强制包含'apsl'类型的样本入口

数量: 1 个或多个

AVS PCC 点云片轨道应使用扩展自 VolumetricVisualSampleEntry, 类型为'apsl'的样本入口。

当点云片轨道仅包含点云片对应的部分组件数据时, 点云片轨道样本入口中应包含一个 AVSPCCComponentInfoBox。当点云片轨道包含点云片对应的所有组件数据时, 点云片轨道样本入口中不应包含 AVSPCCComponentInfoBox。

点云片轨道样本入口中包含该轨道对应的点云片的片标号信息, 点云片基础轨道关联索引的不同点云片轨道对应的片标号信息不应存在重叠。

7.1.4.3.2 样本入口语法

```
aligned(8) class AVSPCCSliceSampleEntry() extends VolumetricVisualSampleEntry ('apsl') {
    unsigned int(1) dynamic_num_slices_flag;
    unsigned int(1) region_id_present;
    bit(6) reserved = 0;
    if(region_id_present == 1){
        unsigned int(16) num_regions;
        for(i=0; i < num_regions; i++){
            unsigned int(16) region_id;
        }
    }
    unsigned int(16) max_num_slices;
    for(i=0; i < max_num_slices; i++){
        unsigned int(16) slice_id;
    }
    AVSPCCComponentInfoBox componentInfo; //可选
}
```

7.1.4.3.3 样本入口语义

`dynamic_num_slices_flag` 取值为 1 时表示当前点云片轨道内的点云片信息会动态变化。取值为 0 时表示当前点云片轨道内的点云片信息不会动态变化。

`region_id_present` 取值为 1 时表示点云片轨道样本入口存在空间区域标识符, 该点云片轨道对应的空间区域的标识符是不变的。取值为 0 时, 表示点云片轨道样本入口不存在空间区域标识符。

`num_regions` 指示点云片轨道对应的三维空间区域的数量。

`region_id` 指示点云片轨道对应的三维空间区域的标识符, 该标识符等于基础轨道中的静态空间区域信息数据盒的空间区域的标识符或动态空间区域信息元数据轨道中规定的空间区域的标识符。

`max_num_slices` 指示当前点云片轨道内对应的点云片最大数量。当 `dynamic_num_slices_flag` 取值为 0 时, 点云片轨道的每个样本内均存在 `max_num_slices` 个点云片。当 `dynamic_num_slices_flag` 取值为 1 时, 点云片轨道的每个样本内存在最多 `max_num_slices` 个点云片。

`slice_id` 指示点云片的片标号。

7.1.4.3.4 样本格式

点云片轨道的样本可包含 0 个、1 个或多个点云片，且包含点云片相应的全部或者部分组件数据。点云片轨道的样本不应包含任何序列头、几何头、属性头、点云帧头参数集或用户数据。

不同点云片轨道中对应同一个点云帧的样本应具备相同的呈现时间。

样本语法如下：

```
aligned(8) class AVSPCCSample()
{
    unsigned int(8) num_slices;
    for(i=0;i < num_slices; i++){
        AVS_PCC_Slice[i];
    }
}
```

样本语义如下：

`num_slices` 指示当前样本中包含的点云片数量。

`AVS_PCC_Slice` 为符合 T/AI 128.2-2024 规定的点云片数据。

7.1.4.3.5 子样本

点云片轨道中，`flags` 字段取值为 1 的子样本数据盒 `SubSampleInformationBox` 应包含于 `SampleTableBox` 中或者包含于每个相应的 `MovieFragmentBoxes` 的 `TrackFragmentBox` 中。此时子样本应符合以下规定：

- a) 当点云片轨道包含相应点云片的全部组件数据时，每个子样本对应一个点云片的全部组件数据。
- b) 当点云片轨道包含相应点云片的部分组件数据时，每个子样本对应一个点云片的部分组件数据，组件数据的类型由点云片轨道样本入口中的 `AVSPCCComponentInfoBox` 指示。
- c) 点云片轨道中，`flags` 字段取值为 0 的子样本数据盒 `SubSampleInformationBox` 可能包含于 `SampleTableBox` 中或者包含于每个相应的 `MovieFragmentBoxes` 的 `TrackFragmentBox` 中。此时子样本的 `AVSPCCPayloadType` 只能取值为 4 或 6。

7.1.4.4 点云轨道关联

7.1.4.4.1 轨道索引

在基于点云片的多轨封装模式下，AVS PCC 点云片基础轨道和点云片轨道之间需使用 ISO/IEC 14496-12 规定的 `TrackReferenceTypeBox` 进行关联索引。

点云片基础轨道的 `TrackBox` 中应包含一个 `reference_type` 为'apbs'的 `TrackReferenceTypeBox`，用于关联索引至一个或多个点云片轨道。

轨道样本入口类型为'apeb'的点云片基础轨道，通过'apbs'类型的轨道索引关联索引至所有点云片轨道。

轨道样本入口类型为'apcb'的点云片基础轨道，通过'apbs'类型的轨道索引关联索引至所有包含几何组件数据的点云片轨道。同时，包含几何组件数据的点云片轨道应通过'apca'类型的轨道索引关联索引至一个或多个包含属性组件数据的点云片轨道。当包含不同属性组件数据的点云片轨道之间存在单向的跨属性依赖关系时，依赖其他属性的点云片轨道的 `TrackBox` 中应包含一个 `reference_type` 为'cadr'的 `TrackReferenceTypeBox`，用于关联索引至其解码时依赖的点云片轨道。

7.1.4.4.2 样本关联

每个点云片基础轨道的样本都对应至一个或多个点云片轨道的样本，且点云片基础轨道样本的呈

现时间应等于或小于点云片轨道中相应样本的呈现时间。

7.2 静态点云数据的存储

7.2.1 概述

非时序 AVS PCC 点云数据在媒体文件中封装为数据项的格式。其中，静态点云数据处理程序类型为'volv'，通过 MetaBox 中的 handler_type 来指示。

7.2.2 数据项格式

7.2.2.1 点云项

7.2.2.1.1 概述

点云项是一个数据项，表示静态的点云数据。

当非时序点云数据存储在一个点云项中时，该点云项包含一个序列头、一个几何头和一个完整的点云帧，可能包含一个属性头和用户数据，且该点云项的数据项类型应为'apce'，该类型的点云项应与一个 AVSPCCConfigurationProperty 相关联。

当非时序点云数据存储在多个点云项中，每个数据项对应一个点云组件，应使用类型为'apcc'的点云项。当点云数据存在于多个数据项类型为'apcc'的点云项中时，包含几何组件的数据项应作为入口点，包含属性组件的数据项应被标记为隐藏数据项。包含同一点云数据的几何组件的数据项与属性组件的数据项，使用 4CC 代码为'apca'的数据项参考类型，表示几何组件的数据项关联索引至属性组件的数据项。

当非时序点云数据包含多个点云片并且点云片存储在多个点云片项中时，应使用类型为'apeb'的点云项，并且该类型的点云项应与 AVSPCCConfigurationProperty 相关联。该类型的点云项不应包含任何的几何或属性数据。该点云项存在时，应同时存在一个或多个点云片项。'apeb'的点云项与点云片项之间的关联关系，使用 4CC 代码为'apbs'的数据项参考类型，表示点云项目关联索引至点云片项。

数据项类型为'apce'或'apcc'的点云项可以关联索引到一个'subs'类型的点云数据项属性。

数据项类型为'apcc'的点云项应关联一个 AVSPCCComponentInfoProperty，以指示不同的组件类型。当不同属性组件类型的点云项之间存在跨属性依赖关系时，依赖其他属性的点云项使用 4CC 代码'cadr'的数据项参考类型，用于指示该属性解码时依赖的属性组件的点云项。

AVSPCCItemData 在结构上与 7.1.2.3 规定的点云样本的语法相同。

7.2.2.1.2 语法

```
aligned(8) class AVSPCCSample {
    AVSPCC_sequence_header;
    AVSPCC_geometry_header;
    AVSPCC_attribute_header; //可选
    AVSPCC_userdata; //可选
    AVSPCC_frame;
}
```

7.2.2.1.3 语义

AVSPCC_sequence_header 指示序列头数据，具体数据结构应符合 T/AI 128.2-2024 中相关规定。点云项应包含一个序列头。

AVSPCC_geometry_header 指示几何头数据，具体数据结构应符合 T/AI 128.2-2024 中相关规定。点云项应包含一个几何头。

AVSPCC_attribute_header 指示属性头数据，具体数据结构应符合 T/AI 128.2-2024 中相关规定。当点云位流中包含属性数据时，点云项应包含一个属性头。

AVSPCC_userdata 指示用户数据，具体数据结构应符合 T/AI 128.2-2024 中相关规定。点云项可能包含用户数据。

AVSPCC_frame 指示点云帧数据，具体点云帧数据结构应符合 T/AI 128.2-2024 中相关规定。

7.2.2.2 点云片项

点云片项应包含一个或多个点云片对应的所有组件数据，并存储为'apsl'类型的数据项。此数据项不应包括任何参数集。

类型为'apsl'的每个点云片项目应与一个'apsi'类型的数据项属性 AVSPCCSliceInfoProperty 相关联。AVSPCCSliceInfoProperty 指示相关联的点云片项中的点云片的数量，以及点云片的片标号。

一个点云片项可以与一个'subs'类型的子样本项属性关联，用于确定点云片及其对应的点云片项的数据。当点云片项与'subs'类型的数据项属性关联时，'subs'类型的数据项属性中的点云片标号应与关联到相同点云片项的 AVSPCCSliceInfoProperty 中的点云片标号一致。

注：点云片项可以包含在文件中，以允许快速获取数据，而无需分析具体的点云数据。对于更细粒度和/或更通用的点云片指示，可以使用 7.1.2.4 中指定的子样本信息。例如，子样本信息适用于指示点云片项中包含的点云片标号。

7.2.3 数据项属性

7.2.3.1 点云配置项属性

数据盒类型：'pccC'

包含于：ItemPropertyContainerBox

强制性：强制关联至'apce'或'apeb'类型的点云项

数量：1 个

AVSPCCConfigurationProperty 指示点云项的解码配置信息。每个'apce'或'apeb'类型的 AVS PCC 点云项均应包含一个 AVSPCCConfigurationProperty。

该数据项属性数据盒应与 7.1.1.2 规定的 AVSPCCConfigurationBox 完全一致。

点云项的'pccC'数据项属性，其 essential 应等于 1。

7.2.3.2 点云组件信息项属性

数据盒类型：'acif'

包含于：ItemPropertyContainerBox

强制性：强制关联至'apcc'类型的点云项

数量：1 个

AVSPCCComponentInfoProperty 指示各个点云项所包含的组件的类型。该数据项属性同时提供属性组件点云项中属性数据相关的信息。该数据项属性与 AVSPCCComponentTypeBox 完全相同。

对于与'apcc'类型的点云项相关联的'acif'类型的数据项属性，其 essential 应等于 1。该数据项属性不应与'apce'类型的点云项相关联。

7.2.3.3 点云子样本项属性

该数据项属性类型为'subs'，应符合 ISO/IEC 23008-12 的约束。

该数据项属性应与 7.1.2.4 规定的 SubSampleInformationBox 完全相同。

7.2.3.4 点云片信息的数据项属性

7.2.3.4.1 定义

数据盒类型：'apsi'

包含于：ItemPropertyContainerBox

强制性：强制关联至'apsl'类型的点云片项

数量：1 个

AVSPCCSliceInfoProperty 用于指示其关联的点云片项中存在的点云片的片标号。

7.2.3.4.2 语法

```
aligned(8) class AVSPCCSliceInfoProperty extends ItemFullProperty('apsi', 0, 0)
{
    unsigned int(16) num_slices;
    for(int i=0; i< num_slices; i++){
        unsigned int(16) slice_id;
    }
}
```

7.2.3.4.3 语义

num_slices 指示关联的点云片项中存在的点云片的数量；

slice_id 指示关联的点云片项中存在的点云片的片标号。

7.2.3.5 点云空间区域的数据项属性

数据盒类型: 'apsr'

包含于: ItemPropertyContainerBox

强制性: 强制关联至'apce'或'apeb'类型的点云项

数量: 1 个

AVSPCCSpatialRegionInfoProperty 用于指示点云项的空间区域信息和关联的点云片信息，通过该信息可确定对应该空间区域的点云片项。

该数据项属性应与 7.3.1.2 规定的 PCC3DSpatialRegionBox 完全相同。

7.2.3.6 点云对象信息数据项属性

数据盒类型: 'objt'

包含于: ItemPropertyContainerBox

强制性: 非强制

数量: 0 个或 1 个

对象信息数据项属性用于描述静态点云项中包含的对象信息。该数据项属性可以包含于类型为 'apce'、'apcc'或'apeb'类型的点云项中。该数据项属性应与 7.3.5.2 规定的 ObjectBox 完全相同。

7.3 点云数据的元数据扩展

7.3.1 点云数据的空间访问

7.3.1.1 空间区域信息结构

7.3.1.1.1 定义

空间区域结构 (PCC3DSpatialRegionStruct) 提供了三维点云数据的空间区域相关信息，包括三维点云中的特定点和空间区域的坐标信息、空间区域与相应一个或多个点云片之间的映射信息、空间区域的显著度等级信息。

7.3.1.1.2 语法

```
aligned(8)     class     PCC3DSpatialRegionStruct     (coordinate_flag,     saliency_flag,     slice_flag,
dimensions_included_flag) {
    unsigned int(16) 3d_region_id;
    if(coordinate_flag == 1){
```

```

3DPoint anchor;
if(dimensions_included_flag == 1) {
    CuboidRegionStruct();
}
}
if(saliency_flag == 1){
unsigned int(8) saliency_level;
}
if(slice_flag == 1) {
    SliceMapping slice_map();
}
}

```

```
aligned(8) class 3DPoint() {
```

```
    unsigned int(16) x;
```

```
    unsigned int(16) y;
```

```
    unsigned int(16) z;
}
```

```
aligned(8) class CuboidRegionStruct() {
```

```
    unsigned int(16) cuboid_dx;
```

```
    unsigned int(16) cuboid_dy;
```

```
    unsigned int(16) cuboid_dz;
}
```

```
aligned(8) class SliceMapping () {
```

```
    unsigned int(16) num_slices;
```

```
    for (j=0; j < num_slices; j++) {
```

```
        unsigned int(16) slice_id;
    }
}

```

7.3.1.1.3 语义

coordinate_flag 取值为 1 时表示存在当前空间区域的具体坐标信息。取值为 0 时表示不存在当前空间区域的具体坐标信息。

slice_flag 取值为 1 时表示当前空间区域映射到特定的点云片。取值为 0 时表示当前空间区域没有映射到特定的点云片。

saliency_flag 取值为 1 时表示当前空间区域具备特定的显著度等级信息。取值为 0 时表示当前空间区域不具备特定的显著度等级信息。

dimensions_included_flag 指示空间区域维度是否已经被标识的标识符。值为 0 时 PCC3DSpatialRegionStruct 指示空间中的一个点；值为 1 时 PCC3DSpatialRegionStruct 指示空间中的一个长方体区域；

3d_region_id 指示空间区域的标识信息，即空间区域的标识符；

anchor 指示笛卡尔坐标系下作为三维空间区域的锚点；

128.1-2025

x, y, z 指示笛卡尔坐标系下一个 3D 点的 x, z, y 坐标值;
 cuboid_dx、cuboid_dy、cuboid_dz 指示笛卡尔坐标系下一个三维空间区域相对于锚点在 x, y, z 轴的延伸。

saliency_level 指示当前空间区域点云数据的显著度等级信息。

num_slices 指示与三维空间区域关联的分片数量。

slice_id 指示与空间区域关联的 AVS PCC 点云片的片标号。

7.3.1.2 静态空间区域信息数据盒

7.3.1.2.1 定义

数据盒类型: 'apsr'

包含于: 样本入口

强制性: 非强制

数量: 0 个或 1 个

静态空间区域信息数据盒指示点云数据场景中不随时间变化的空间区域相关信息。

当 PCC3DSpatialRegionBox 包含于 'apce' 类型的单轨封装的点云轨道的样本入口中时，PCC3DSpatialRegionBox 指示该点云轨道对应的静态空间区域信息。

当 PCC3DSpatialRegionBox 包含于 'apcc' 类型的几何组件轨道的样本入口中时，PCC3DSpatialRegionBox 指示几何组件轨道和该轨道关联的属性组件轨道对应的静态空间区域。PCC3DSpatialRegionBox 不能包含于属性组件轨道中。

7.3.1.2.2 语法

```
aligned(8) class PCC3DSpatialRegionBox extends FullBox('apsr',0,0){
    unsigned int(16) num_region;
    for (int i=0; i < num_region; i++) {
        PCC3DSpatialRegionStruct();
    }
}
```

7.3.1.2.3 语义

num_region 指示 3D 空间区域的数量。

PCC3DSpatialRegionStruct 指示 3D 空间区域信息及其关联点云片的信息。

7.3.1.3 动态空间区域信息元数据轨道

动态空间区域信息元数据轨道用于指示点云中动态变化的空间区域相关信息。该元数据轨道应包含一个引用类型为 'cdsc' 轨道引用，用于关联索引相应的单轨封装的点云轨道或几何组件的轨道。

当动态空间区域信息元数据轨道关联索引至 'apce' 类型的单轨封装点云轨道时，其指示该点云轨道对应的动态空间区域信息。

当动态空间区域信息元数据轨道关联索引至 'apcc' 类型的几何组件轨道时，其指示该几何组件轨道和几何组件轨道关联的属性组件轨道对应的动态空间区域信息。

7.3.1.3.1 样本入口

动态空间区域信息元数据轨道样本入口语法如下：

```
aligned(8) class DynamicAVSPCCSSpatialInfoSampleEntry extends MetaDataSampleEntry('apdy') {
    unsigned int(1) saliency_info_flag;
    bit(7) reserved = 0;
    if(saliency_info_flag == 1){
```

```

        unsigned int(8) default_saliency_level;
    }
    PCC3DSpatialRegionBox ();
}

```

样本入口语义如下：

saliency_info_flag 取值为 1 时表示动态空间区域信息元数据轨道的样本中包含动态变化的显著度等级信息。取值为 0 时表示动态空间区域信息元数据轨道的样本中不包含显著度等级信息，此时该元数据轨道所有样本中的 **saliency_present_flag** 取值应为 0。

default_saliency_level 指示缺省的显著度等级信息。

PCC3DSpatialRegionBox 指示一个或多个初始三维空间区域信息。

7.3.1.3.2 样本格式

样本语法如下：

```

aligned(8) DynamicPCCSpatialRegionSample () {
    unsigned int(16) num_region;
    for (int i=0; i < num_region; i++) {
        bit(1) region_updates_flag;
        bit(1) saliency_flag;
        bit(1) slice_flag;
        bit(1) coordinate_flag;
        bit(1) dimensions_included_flag;
        bit(3) reserved = 0;
        if (region_updates_flag) {
            PCC3DSpatialRegionStruct (coordinate_present_flag, saliency_present_flag,slice_present_flag,
dimensions_included_flag );
        }
    }
}

```

语义如下：

region_updates_flag 指示样本中相应三维空间区域是否更新，包括区域尺寸位置、显著度等级更新、空间区域和关联的分片之间的映射的更新。

saliency_flag 取值为 1 时表示样本中相应三维空间区域存在更新的显著度等级信息。取值为 0 时表示样本中相应三维空间区域不存在更新的显著度等级信息。

slice_flag 取值为 1 时表示样本中相应三维空间区域存在更新的和其关联的分片之间的映射。取值为 0 时表示样本中相应三维空间区域不存在更新的和其关联的分片之间的映射。

coordinate_flag 取值为 1 时表示样本中相应三维空间区域存在更新的尺寸位置信息。取值为 0 时表示样本中相应三维空间区域不存在更新的尺寸位置信息。

num_region 指示样本中的三维空间区域的数量。只有更新的空间区域才会出现在样本中。

7.3.2 点云数据的视窗信息

7.3.2.1 相机信息结构

相机外参信息结构 **ExtCameraInfoStruct** 用于指示相机的外参信息，包含视窗的位置和旋转信息。

相机外参信息结构语法如下：

```
aligned(8) class ExtCameraInfoStruct() {
    unsigned_int(1) camera_pos_present;
    unsigned int(1) camera_ori_present;
    bit(6) reserved = 0;
    if (camera_pos_present == 1){
        signed int(32) camera_pos_x;
        signed int(32) camera_pos_y;
        signed int(32) camera_pos_z;
    }
    if (camera_ori_present == 1){
        signed int(32) camera_quat_x;
        signed int(32) camera_quat_y;
        signed int(32) camera_quat_z;
    }
}
```

语义如下：

`camera_pos_present` 取值为 1 时表示存在相机位置参数；取值为 0 时表示不存在相机位置参数。

`camera_ori_present` 取值为 1 时表示存在相机方向参数；取值为 0 时表示不存在相机方向参数。

`camera_pos_x, camera_pos_y, camera_pos_z` 分别表示全局参考坐标系中相机位置的 x, y 和 z 坐标。

该值应以 2^{-16} 米为单位。

`cam_quat_x, cam_quat_y` 和 `cam_quat_z` 用于计算相机旋转四元数中的 `qX, qY, qZ` 分量，取值范围为 $[-2^{30}, 2^{30}]$ 。各分量的计算规则如下：

$$qX = \text{cam_quat_x} \div 2^{30}, qY = \text{cam_quat_y} \div 2^{30}, qZ = \text{cam_quat_z} \div 2^{30}$$

第 4 个分量 `qW` 可按照四元数的数学特性推导得到：

$$qW = \text{Sqrt}(1 - (qX^2 + qY^2 + qZ^2))$$

相机旋转四元数各分量缺省值为 0。

四元数 (w, x, y, z) 代表围绕向量 (x, y, z) 以角度 $2 \cos^{-1} w = 2 * \sin^{-1} \sqrt{x^2 + y^2 + z^2}$ 进行旋转。

7.3.2.1.2 相机内参信息结构

相机内参信息结构 `IntCameraInfoStruct` 用于指示相机的内参信息，包含相机类型和相关参数信息。

相机内参信息结构语法如下：

```
aligned(8) class IntCameraInfoStruct() {
    unsigned int(10) camera_id;
    unsigned int(1) camera_depth_present;
    unsigned int(3) camera_type;
    bit(2) reserved = 0;
    if (camera_type == 0) {
        signed int(32) erp_horizontal_fov;
        signed int(32) erp_vertical_fov;
    }
    if (camera_type == 1) {
        signed int(32) perspective_horizontal_fov;
        unsigned int(8)[4] perspective_aspect_ratio;
```

```

    }
    if (camera_type == 2) {
        unsigned int(8)[4] ortho_aspect_ratio;
        unsigned int(8)[4] ortho_horizontal_size;
    }
    if (camera_depth_present == 1){
        unsigned int(32) camera_near_depth;
        unsigned int(32) camera_far_depth;
    }
}

```

语义如下：

`camera_id` 指示对应视窗的相机标识符。

`camera_depth_present` 取值为 1 时表示存在相机深度参数；取值为 0 时表示不存在相机深度参数。

`camera_type` 指示视窗相机的投影算法类型。取值为 0 表示 ERP 投影；取值为 1 表示透视投影；取值为 2 表示正交投影；其余取值保留。

`erp_horizontal_fov` 指示 ERP 投影时，视窗区域水平方向的经度范围，以弧度为单位。该字段取值范围为 $(0, 2\pi)$ 。

`erp_vertical_fov` 指示 ERP 投影时，视窗区域垂直方向的维度范围，以弧度为单位。该字段取值范围为 $(0, \pi)$ 。

`perspective_horizontal_fov` 指示透视投影时，视窗区域的水平范围，以弧度为单位。该字段取值范围为 $(0, \pi)$ 。

`perspective_aspect_ratio` 指示透视投影时，视窗的纵横比例(水平/垂直)。该字段取值为 32 位浮点数，解析过程应符合 IEEE 754 的规定。

`ortho_aspect_ratio` 指示正交投影时，视窗的纵横比例(水平/垂直)。该字段取值为 32 位浮点数，解析过程应符合 IEEE 754 的规定。

`ortho_horizontal_size` 指示正交投影时，视窗水平方向的尺寸，以米为单位。该字段取值为 32 位浮点数，解析过程应符合 IEEE 754 的规定。

`camera_near_depth`、`camera_far_depth` 分别指示与相机相关联的视锥体的近和远平面指示近和远深度（或距离），值应以 2^{-16} 米为单位。

7.3.2.1.3 视窗信息结构

视窗信息结构 `ViewportInfoStruct` 用于指示视窗信息，该结构包含相机的内参和外参信息。

视窗信息结构语法如下：

```

aligned(8) class ViewportInfoStruct(ext_camera_flag, int_camera_flag) {
    if (ext_camera_flag == 1) {
        unsigned int(1) center_view_flag;
        bit(6) reserved = 0;
        if (center_view_flag == 0) {
            unsigned int(1) left_view_flag;
        } else {
            bit(1) reserved = 0;
        }
        ExtCameraInfoStruct extCamInfo();
    }
}

```

```

if (int_camera_flag == 1) {
    IntCameraInfoStruct intCamInfo();
}
}

```

语义如下：

`center_view_flag` 取值为 1 时表示指示的视窗位置对应视窗中心；取值为 0 时表示指示的视窗位置对应视窗双目位置中的一个。

`left_view_flag` 取值为 1 时表示指示的视窗位置信息对应视窗双目位置中的左目；取值为 0 时表示指示的视窗位置信息对应视窗双目位置中的右目。

`extCamInfo()` 指示视窗的外参信息。

`intCamInfo()` 指示视窗的内参信息。

7.3.2.2 视窗信息定时元数据

7.3.2.2.1 概述

视窗信息定时元数据用于指示点云中的视窗信息，以及与视窗相关的相机内参和外参信息，其中相机外参信息包含了视窗的位置和旋转信息。

视窗信息定时元数据仅关联索引至相应的点云轨道或点云轨道组，不直接关联索引组件轨道。视窗信息定时元数据轨道使用'cdtg'轨道索引关联索引至相应的轨道或轨道组时，该定时元数据轨道描述被关联索引的所有点云轨道的视窗信息。当视窗信息定时元数据轨道使用'cdsc'轨道索引关联索引至一个或多个点云轨道时，该定时元数据轨道分别描述每个点云轨道的视窗信息。

视窗信息定时元数据轨道中的每个样本均可被标记为同步样本。若视窗信息定时元数据轨道中某个样本关联索引的点云轨道中与之相同解码时间的样本中存在至少一个同步样本，则该定时元数据样本需被标记为同步样本。

7.3.2.2.2 样本入口

样本入口类型：'avpt'

包含于：Sample Description Box

强制性：非强制

数量：0 个或 1 个

视窗信息样本入口为 `ViewportInfoSampleEntry`，类型为'avpt'。

视窗信息样本入口中需包含 `ViewportInfoConfigurationBox`，该数据盒中指示了视窗类型和可能的相机内参或相机外参信息，该数据盒中的元数据对轨道中的所有样本生效。

样本入口语法如下：

```

aligned(8) class ViewportInfoConfigurationBox extends FullBox('avpC', version=0, 0) {
    unsigned int(8) viewport_type;
    string viewport_description;
    unsigned int(1) dynamic_int_camera_flag;
    unsigned int(1) dynamic_ext_camera_flag;
    bit(6) reserved = 0;
    if (dynamic_int_camera_flag == 0) {
        IntCameraInfoStruct();
    }
    if (dynamic_ext_camera_flag == 0) {
        ExtCameraInfoStruct();
    }
}

```

```
}
```

```
aligned(8) class ViewportInfoSampleEntry() extends MetadataSampleEntry ('avpt') {
    ViewportInfoConfigurationBox();
}
```

语义如下：

`viewport_type` 指示当前样本入口对应的所有样本的视窗类型，其取值含义见表 4。

表 4 视窗类型

取值	描述
0	根据导演或者内容提供者意图生成的推荐视窗
1	根据用户观看数据统计的推荐视窗
2	根据其余用户选择的视窗生成的推荐视窗
3	推荐的初始视窗
4	与特定空间区域关联的推荐视窗
5	包含对象信息指示的推荐视窗
6..239	保留
240..255	自定义

`viewport_description` 以空字符结尾的字符串，提供推荐视窗的文本描述信息。

`dynamic_int_camera_flag` 取值为 0 时表示当前样本入口对应的所有样本的相机内参固定不变。取值为 1 时表示当前样本入口对应的所有样本的相机内参动态变化。

`dynamic_ext_camera_flag` 取值为 0 时表示当前样本入口对应的所有样本的相机外参固定不变。取值为 1 时表示当前样本入口对应的所有样本的相机外参动态变化。

当 `viewport_type` 字段取值为 3 时，视窗定时元数据指示推荐的初始视窗信息，包括初始视窗位置和旋转信息。当点云数据需要以特定初始视窗（非缺省初始视窗）开始播放时，视窗定时元数据轨道应关联索引至该点云轨道。当该类型的元数据缺失时，则初始视窗的 `cam_pos_x`, `cam_pos_y`, `cam_pos_z`, `cam_quat_x`, `cam_quat_y`, 和 `cam_quat_z` 缺省值均为 0。

7.3.2.2.3 样本格式

每个视窗样本携带一个或多个在样本入口中指定视窗类型的视窗。每个视窗包含相机外参信息和相机内参信息，由 `IntCameraInfoStruct` 和 `ExtCameraInfoStruct` 表示。相机外参信息应存在于每个样本中，但相机内参信息仅存在于相机内参发生变化的样本中。前一个样本中指示的某个视窗的相机外参和相机内参信息将持续至该视窗的对应信息发生变化时。

样本格式语法如下：

```
aligned(8) class ViewportInfoSample() {
    unsigned int(8) num_viewports;
    for (i=1; i <= num_viewports; i++) {
        unsigned int(7) viewport_id[i];
        unsigned int(1) viewport_cancel_flag[i];
        if (viewport_cancel_flag[i] == 0) {
            unsigned int(1) camera_extrinsic_flag[i];
            unsigned int(1) camera_intrinsic_flag[i];
        }
    }
}
```

```
unsigned int(1) slice_info_flag[i];
unsigned int(1) object_flag[i];
bit(4) reserved = 0;
ViewportInfoStruct(camera_extrinsic_flag[i], camera_intrinsic_flag[i]);
if(slice_info_flag[i] == 1){
    unsigned int(8) num_slices;
    for (j=0; j < num_slices; j++) {
        unsigned int(16) slice_id;
    }
}
if(object_flag[i]==1){
    unsigned int(16) num_objects;
    for(i=0; i<num_objects; i++){
        unsigned int(16) ref_object_info_id;
    }
}
}
```

语义如下：

视窗信息元数据轨道应在相机外参信息 ExtCameraInfoStruct() 中保存，可包含于样本入口中或者样本中。当 dynamic_ext_camera_flag 取值为 1 时，所有样本中 camera_extrinsic_flag[i] 的取值至少有一个为 1。

`num_viewports` 指示样本中指示的视窗数目。

viewport id[i]指示对应视窗的标识符。

`viewport_cancel_flag[i]`取值为 1 时表示视窗标识符取值为 `viewport_id[i]` 的视窗被取消了。取值为 0 时表示视窗标识符取值为 `viewport_id[i]` 的视窗相关信息在样本中指示。

`camera_intrinsic_flag[i]` 取值为 1 时表示当前样本中第 i 个视窗存在相机内参。如果 `dynamic_int_camera_flag` 取值为 0，则该字段取值为 0。取值为 0 时表示当前样本中第 i 个视窗不指示相机内参。

`camera_extrinsic_flag[i]` 取值为 1 时表示当前样本中的第 i 个视窗存在相机外参。如果 `dynamic_ext_camera_flag` 取值为 0，则该字段取值为 0。取值为 0 时表示当前样本中第 i 个视窗不指示相机外参。

`slice_info_flag[i]` 取值为 1 时表示当前样本中的第 i 个视窗关联了相应的 slice 信息；取值为 0 时表示当前样本中第 i 个视窗没有关联相应的 slice 信息。

`num_slices` 指示当前视窗关联的 slice 数量。

slice_id 指示点云片标号。

`object_flag[i]`取值为 1 时表示当前样本中的第 i 个视窗关联了对象指示信息；取值为 0 时表示当前样本中的第 i 个视窗没有关联对象指示信息。

`num_object` 指示当前视窗关联的对象数量。

`ref_object_info_id` 指示当前视窗中关联的对象信息的标识符，该标识符参考对象信息定时元数据轨道中对应样本所指示的部分或全部对象信息的标识符。

7.3.3 可替换内容指示

7.3.3.1 概述

点云数据内容可以被编码为多个可替换的点云码流，且每个可替换的点云码流可以按照不同的模式进行封装，并根据可替换信息元数据对不同点云码流进行区分。

7.3.3.2 可替换信息结构

7.3.3.2.1 定义

点云可替换信息结构 AVSPCCAlternativeInfoStruct 用于指示可替换点云码流的元数据信息。

7.3.3.2.2 语法

```
aligned(8) class AVSPCCAlternativeInfoStruct() {
    unsigned int(1) quality_ranking_flag;
    bit(7) reserved=0;
    if(quality_ranking_flag == 1){
        unsigned int(8) quality_ranking;
    }
}
```

7.3.3.2.3 语义

`quality_ranking_flag` 取值为 0 时表示当前数据结构中不指示相应点云码流的质量信息；取值为 1 时表示当前数据结构中指示相应码流的质量信息。

`quality_ranking` 指示质量等级信息，该字段取值越小，表明相应点云码流的质量越高

7.3.3.3 动态点云内容的可替换信息指示

7.3.3.3.1 概述

动态点云内容的每个可替换点云码流可以按照单轨模式、基于组件的多轨模式、基于点云片的多轨模式进行封装，并采用可替换组标识符（`TrackHeaderBox` 中的 `alternate_group` 字段）和可替换信息数据盒指示相应码流的可替换信息：

- a) 当可替换点云码流采用单轨模式进行封装时，可替换信息数据盒包含于相应单轨封装点云轨道中。
- b) 当可替换点云码流采用基于组件的多轨模式进行封装时，可替换信息数据盒包含于几何组件轨道中。
- c) 当可替换点云码流采用基于点云片的多轨模式进行封装时，可替换信息数据盒包含于点云片基础轨道中。
- d) 包含可替换信息数据盒的点云轨道的可替换组标识符取值不应为 0，具备相同可替换组标识符的码流互为可替换关系。

7.3.3.3.2 可替换信息数据盒

数据盒类型：'apai'

包含于：样本入口

强制性：非强制

数量：0 个或 1 个

可替换信息数据盒中包含一个或多个可替换信息数据结构，指示相应码流的可替换信息。

可替换信息数据盒语法如下：

```
aligned(8) class AVSPCCAlternativeInfoBox extends Fullbox('apai', 0, 0) {
```

```
unsigned int(1) multi_alternative_bitstream_flag;
bit(7) reserved;
if(multi_alternative_bitstream_flag == 0){
    AVSPCCAlternativeInfoStruct();
}
}
```

语义如下：

multi_alternative_bitstream_flag 取值为 0 时表示当前轨道仅属于一个可替换码流；取值为 1 时表示当前轨道同时属于多个可替换码流的组成部分。

AVSPCCAlternativeInfoStruct 指示相应码流的可替换信息。

7.3.3.3.3 可替换轨道组

数据盒类型:'apag'

包含于:TrackGroupBox

强制性:非强制

数量:0 个, 1 个或多个

当点云轨道的可替换信息数据盒中 multi_alternative_bitstream_flag 字段取值为 1 时，可替换轨道组用于关联同属于一个可替换码流的所有点云轨道。同属于一个可替换码流的点云轨道应具备相同的 track_group_id。

可替换轨道组语法如下：

```
aligned(8) class AvsPCCAlternativeTrackGroupBox extends TrackGroupTypeBox('apag') {
    AVSPCCAlternativeInfoStruct();
}
```

7.3.3.4 静态点云数据的可替换内容指示

7.3.3.4.1 概述

静态点云数据的每个可替换点云位流可以按照单数据项模式、基于组件的多数据项模式、基于点云片的多数据项模式进行封装，并采用可替换信息项目属性指示相应码流的可替换信息：

- a) 当可替换点云位流采用单数据项模式进行封装时，相应点云项应包含一个可替换信息数据项属性。
- b) 当可替换点云位流采用基于组件的多数据项模式进行封装时，相应几何组件项应包含一个可替换信息数据项属性。
- c) 当可替换点云位流采用基于点云片的多数据项模式进行封装时，相应点云片基础项应包含一个可替换信息数据项属性。

7.3.3.4.2 可替换信息数据项属性

数据盒类型：'apap'

包含于: ItemPropertyContainerBox

强制性: 非强制

数量: 0 个, 1 个或多个

可替换信息数据项属性中包含一个或多个可替换信息数据结构，指示相应位流的可替换信息。

可替换信息数据项属性语法如下：

```
aligned(8) class AvsPCCAlternativeInfoProperty extends ItemFullProperty('apap', 0, 0)
{
    unsigned int(1) multi_alternative_bitstream_flag;
```

bit(7) reserved;

```
    unsigned int(16) alternate_group;
    if(multi_alternative_bitstream_flag == 0){
        AVSPCCAlternativeInfoStruct();
    }
}
```

语义如下：

`multi_alternative_bitstream_flag` 取值为 0 时表示当前数据项仅属于一个可替换位流；取值为 1 时表示当前数据项同时属于多个可替换位流的组成成分。

`alternative_group` 指示可替换组标识符，具备相同可替换组标识符的位流互为可替换关系。该字段取值应大于 0。

`AVSPCCAlternativeInfoStruct` 指示相应码流的可替换信息。

7.3.3.4.3 可替换实体组

数据盒类型：'apae'

包含于：GroupsListBox

强制性：非强制

数量：0 个，1 个或多个

当点云项的可替换信息数据项属性中 `multi_alternative_bitstream_flag` 字段取值为 1 时，可替换实体组用于关联同属于一个可替换码流的所有点云项。同属于一个可替换码流的点云项应具备相同的 `entity_group_id`。

可替换实体组语法如下：

```
aligned(8) class AVSPCCAlternativeEntityToGroupBox extends EntityToGroupBox('apae',0,0) {
    unsigned int(32) group_id;
    unsigned int(32) num_entities_in_group;
    for(i=0; i<num_entities_in_group; i++){
        unsigned int(32) entity_id;
    }
    AVSPCCAlternativeInfoStruct();
}
```

语义如下：

`group_id` 指示当前实体组的标识符。

`num_entities_in_group` 指示当前实体组内的实体（轨道或者项目）数目。

`entity_id` 指示实体的标识符。

`AVSPCCAlternativeInfoStruct` 指示当前实体组对应的可替换码流的可替换信息。

7.3.4 优先级信息定时元数据

7.3.4.1 概述

优先级信息定时元数据用于指示相应轨道中各样本的优先级信息，例如内容制作者对不同样本的推荐优先级。

该元数据轨道应包含一个引用类型为'cdsc'轨道引用，用于关联索引至相应的包含序列头的点云轨道，该元数据轨道用于描述被关联索引至轨道动态变化的样本优先级信息。

7.3.4.2 样本入口

7.3.4.2.1 定义

样本入口类型: 'avsp'

包含于: Sample Description Box

强制性: 否

数量: 0 个或 1 个

样本优先级定时元数据轨道的样本入口为 PriorityInfoSampleEntry，类型为'avsp'。

7.3.4.2.2 语法

```
aligned(8) class PriorityInfoSampleEntry () extends MetadataSampleEntry ('avsp') {  
    string description;  
}
```

7.3.4.2.3 语义

description 指示样本优先级的文本描述，如内容制作者的推荐内容等，以空值结尾的 UTF-8 字符串。

7.3.4.3 样本格式

7.3.4.3.1 概述

样本优先级元数据轨道中的每个样本包含所关联轨道的样本优先级。

7.3.4.3.2 语法

```
aligned(8) class PriorityInfoSample() {  
    unsigned int(8) priority; //可选  
}
```

7.3.4.3.3 语义

priority 指示该元数据轨道所关联轨道的样本优先级，即从该元数据轨道样本开始到下一元数据轨道样本前所关联轨道的样本优先级。

7.3.5 对象信息元数据

7.3.5.1 对象信息结构

7.3.5.1.1 定义

对象信息结构 ObjectInfoStruct 用于指示对象的标识、空间区域及对应分片等信息。

7.3.5.1.2 语法

```
aligned(8) class ObjectInfoStruct() {  
    unsigned int(16) object_info_id;  
    unsigned int(1) object_oid_flag;  
    unsigned int(1) object_spatial_info_flag;  
    unsigned int(1) object_slice_info_flag;  
    bit(5) reserved;  
    if(object_oid_flag == 1){  
        string object_oid;  
    }  
    else{  
        string object_label;  
    }
```

```

    }
    if(object_spatial_info_flag==1){
        3DPoint() anchor;
        CuboidRegionStruct() cuboidRegion;
    }
    if(object_slice_info_flag){
        SliceMapping() slice_info;
    }
}

```

7.3.5.1.3 语义

`object_info_id` 指示对应对象信息条目的标识符。

`object_oid_flag` 取值为 0 时表示对象描述信息以可读文本字符串形式指示。取值为 1 时表示对象描述信息以 OID 的形式指示，OID 应符合 GB/T 17969.1-2015 的规定。

`object_spatial_info_flag` 取值为 0 时表示不指示对象对应的具体空间信息；取值为 1 时表示指示对象对应的具体空间信息。

`object_slice_info_flag` 取值为 0 时表示不指示对象对应的点云片信息；取值为 1 时表示指示对象对应的点云片信息。

`object_oid` 指示相对对象信息条目对应的对象描述信息，以 OID 的形式指示。

`object_label` 指示相对对象信息条目对应的对象描述信息，以人眼可读的字符串形式指示。

`anchor` 指示对象所在空间区域的锚点坐标。

`cuboidRegion` 指示对象所在空间区域的尺寸信息。笛卡尔坐标系下一个三维空间区域相对于锚点在 x, y, z 轴的延伸。

`slice_info` 指示点云片信息。指示与三维空间区域关联的分片数量，标识与空间区域关联的点云片。

7.3.5.2 对象信息数据盒

7.3.5.2.1 定义

数据盒类型：'objt'

包含于：AVSPCCSSampleEntry ('apce'、'apcc'、'apeb'或'apcb')

强制性：非强制

数量：0 个或 1 个

对象信息信息数据盒指示点云场景中不随时间变化的对象相关信息。

当 `ObjectBox` 包含于'apce'类型的单轨封装点云轨道的样本入口中时，该数据盒指示该点云轨道对应的对象信息。

当 `ObjectBox` 包含于'apcc'类型的几何组件轨道的样本入口中时，该数据盒指示该几何组件轨道和该轨道关联的属性组件轨道对应的对象信息。

该数据盒不能包含于属性组件轨道中。

7.3.5.2.2 语法

```

aligned(8) class ObjectBox extends FullBox('objt', 0, 0){
    unsigned int(16) num_objects;
    for (int i=0; i < num_objects; i++) {
        ObjectInfoStruct();
    }
}

```

7.3.5.2.3 语义

`num_objects` 指示对象的数量。

`ObjectInfoStruct` 指示对象的具体信息。

7.3.5.3 对象信息定时元数据轨道

7.3.5.3.1 概述

对象信息定时元数据轨道用于指示元数据轨道描述的点云轨道对应的点云位流中的对象信息。

对象信息定时元数据轨道应包含一个引用类型为'cdsc'轨道引用，用于关联索引相应的轨道，具体约束如下：

- a) 对象信息定时元数据轨道可以通过类型为'cdsc'轨道引用关联索引至样本入口类型为'apce'的单轨封装点云轨道，该元数据轨道描述的点云位流包含于相应的单轨封装点云轨道；
- b) 对象信息定时元数据轨道可以通过类型为'cdsc'轨道引用关联索引至样本入口类型为'apcc'的几何组件轨道，该元数据轨道描述的点云位流包含于相应的几何组件轨道及其关联的属性组件轨道；
- c) 对象信息定时元数据轨道可以通过类型为'cdsc'轨道引用关联索引至样本入口类型为'apeb'或'apcb'的点云片基础轨道，该元数据轨道描述的点云位流包含于相应的点云片基础轨道及其关联的点云片轨道。

7.3.5.3.2 样本入口

对象信息定时元数据轨道样本入口为 `ObjectInfoSampleEntry`，类型为'obdi'。对象信息定时元数据轨道样本入口中包含元数据轨道描述的点云轨道对应的点云位流中的对象信息。

对象信息定时元数据轨道样本入口语法如下：

```
aligned(8) class ObjectInfoSampleEntry extends MetaDataSampleEntry ('obdi')
{
    ObjectBox objectInfo;
}
```

7.3.5.3.3 样本格式

对象信息定时元数据轨道语法如下：

```
aligned(8) ObjectInfoSample() {
    unsigned int(16) num_object_update;
    for(i=0;i<num_object_update;i++){
        unsigned int(1) object_canceled_flag;
        bit(7) reserved;
        if(object_canceled_flag == 0){
            ObjectInfoStruct();
        }
        else
            unsigned int(16) ref_object_info_id;
    }
}
```

语义如下：

`num_object_update` 指示当前样本包含的对象信息相对于前一个 `num_object_update` 取值不为 0 的样本变化的条目数量，包括新增的数量。当该字段取值为 0 时表示当前样本包含的对象信息与前一个样本

相同。

`object_canceled_flag` 取值为 1 时表示相应用对象不再包含于当前样本中；取值为 0 时表示相应用对象包含于当前样本中，但发生了更新，或表示相应用对象在当前样本中是新增的。

`ObjectInfoStruct()` 指示当前样本中变化的对象及其信息。

`ref_object_info_id` 指示当前样本中不再包含的对象标识。

7.3.6 采集时间信息数据盒

7.3.6.1 定义

数据盒类型：'atts'

包含于：SampleTableBox

强制性：非强制

数量：0 个或 1 个

采集时间数据盒用于指示轨道中样本和采集时间的对应关系。

7.3.6.2 语法

```
aligned(8) class AcquisitionTimestampBox extends FullBox('atts', 0, 0){
    unsigned int(1) initial_timestamp_flag;
    if(initial_timestamp_flag == 1){
        unsigned int(64) initial_acquisition_time;
    }
    unsigned int(32) entry_count;
    for (i=0; i < entry_count; i++) {
        unsigned int(32) sample_count;
        signed int(32) sample_offset;
    }
}
```

7.3.6.3 语义

`initial_timestamp_flag` 取值为 0 表示初始采集时间信息等同于媒体文件的创建时间。取值为 1 表示初始采集时间由 `initial_acquisition_time` 指示。

`initial_acquisition_time` 指示初始采集时刻的 UTC 时间。

`entry_count` 指示当前采集时间表的时间信息条目数量。

`sample_count` 指示具备相应 `sample_offset` 的连续样本数量。

`sample_offset` 指示当前样本采集时间相对于上一样本 ATS 的偏移量，即 $ATS[n+1] = ATS[n] + sample_offset[n]$ ，以文件中的时间刻度为单位。

8 基于 SMT 的点云数据传输

8.1 概述

SMT 信令功能区中包含了一整套用于传输和消费 SMT 数据包的必要信令消息，SMT 流传输点云数据应符合点云数据的 SMT 扩展信令表、描述符或者传输相关信息的消息格式，包括：

- 点云数据视角依赖数据包消息；
- 点云数据资源索引描述符；
- 媒体资源组消息；
- Asset 关系信息描述符；
- 交互反馈信息及信令表；

f) 用户实时位置以及观看方向。

8.2 点云视角依赖数据包消息

8.2.1 概述

点云视角依赖数据包消息用于服务器向客户端指示与用户视角相关的媒体数据包索引信息，以及对应的点云空间区域信息。服务器根据客户端发送的当前视角相关状态信息确定视角依赖点云空间区域信息，并选取对应的空间区域点云数据。以数据包消息方式将确定的点云数据发送给客户端。客户端可以根据用户需求处理（如渲染）与用户视角相关的点云数据空间区域信息及对应的数据。

8.2.2 语法

点云视角依赖数据包消息的语法见表 5。

表 5 点云视角依赖数据包消息语法

语法	值	位数	类型
Viewdependent_packet_message()			
message_id		16	uimsbf
version		8	uimsbf
length		32	uimsbf
message_payload {			
packet_id		16	uimsbf
num_regions;			
for (i=0; i<num_regions; i++) {			
3DSpatialRegionStruct(1);			
}			
}			
}			

8.2.3 语义

message_id 指示点云视角依赖媒体数据包消息的标识符。

version 指示点云视角依赖媒体数据包消息的版本。新的版本所携带的信息将覆盖任何之前的旧版本。

length 指示以字节计算的点云视角依赖媒体数据包消息的长度，即从下一字段起直到该数据包消息最后一个字节的长度。该字段取值不能为 0。

packet_id 指示与点云 asset 关联的媒体数据包索引信息。此字段将被传递到渲染器，用以指示当前正在接收和播放的点云数据 asset 的索引，基于该 asset 的索引，渲染器能够确定与当前 asset 对应的空间覆盖信息。

num_regions 指示与用户视窗产生重叠的点云数据三维空间区域数量。

3DSpatialRegionsStruct() 指示三维空间区域结构，在不同场景中可使用不同的空间区域结构，在进行点云数据传输时，采用 PCC3DSpatialRegionStruct() 结构。

8.3 点云资源索引描述符

8.3.1 概述

点云资源索引描述符采用 Volumetric_media_descriptor 结构，对于点云，该媒体资源索引描述符应当位于点云数据流对应的包含媒体资源描述符的信令消息或信令表中。

当存在点云片时，点云资源索引描述符指示其和点云片资源之间的索引关系。

8.3.2 语法

点云资源索引描述符的语法见表 6。

表 6 点云资源索引描述符语法

语法	值	位数	类型
Volumetric_media_descriptor () {			
descriptor_tag		16	uimsbf
descriptor_length		16	uimsbf
data_type		8	uimsbf
all_tiles_slices_present_flag	'1111111'	1	bslbf
reserved	N1	7	bslbf
if (!all_tiles_slices_present_flag) {			
num_slices		16	uimsbf
for (i=0; i<N1; i++) {			
tile_slice_id		16	bslbf
}			
}			
}			

8.3.3 语义

descriptor_tag 指示标识符，用于表示描述符类型。

descriptor_length 指示标识符的长度，单位为字节。

data_type 指示容积媒体（容积媒体包含容积视频以及几何编码点云）的媒体资源类型，类型取值见表 7。在基于 SMT 的点云传输中，data_type 的取值限定为 0x00、0x03、0x04 或 0x05。

表 7 媒体资源类型取值

值	含义
0x00	所有容积媒体组件数据
0x01	容积视频图集组件数据
0x02	容积视频占用图组件数据
0x03	容积媒体几何组件数据
0x04	容积媒体属性组件数据
0x05	动态容积媒体时序元数据信息
0x06-0xFF	保留

all_tiles_slices_present_flag 指示当前媒体资源是否包含容积媒体的所有分片，取值为 1 时表示当前媒体资源包含了点云的所有点云片。取值为 0 时表示当前媒体资源仅包含一部分点云片。

num_tiles_slices 指示当前媒体资源包含的容积媒体分片数目，即点云片数量。

tile_slice_id 指示容积媒体分片的标识符，即点云片的标识符。

8.4 媒体资源组消息

8.4.1 概述

当通过 SMT 发送点云内容时，应使用媒体资源组消息向接收实体提供与点云内容相关联的资源信息。该消息还可用于指示正在传输至接收实体的媒体资源。

8.4.2 语法

媒体资源组消息的语法见表 8。

表 8 媒体资源组消息语法

语法	值	位数	类型
Asset_group_message () { message_id version length num_asset_groups start_time for (i=0; i<N1; i++) { asset_group_id pending_flag 3D_spatial_region_info_flag reserved if (3D_spatial_region_info_flag){ num_regions for (j=0; j<N2; j++) { 3DSpatialRegionStruct(1) } } } }	N1 ‘111111’ N2	16 8 16 8 16 8 1 1 6 16	uimsbf uimsbf uimsbf uimsbf uimsbf uimsbf bslbf bslbf bslbf uimsbf

8.4.3 语义

message_id 指示媒体资源组消息的标识符。

version 指示媒体资源组消息的版本。

length 指示媒体资源组消息的长度，以字节为单位，长度计算从下一个字段起直至消息结尾。该字段取值不能为 0。

num_asset_groups 指示媒体资源组的数目，每一个媒体资源组中的媒体资源都对应一个媒体组件。

start_time 指示点云内容的起始呈现时间，该起始呈现时间对本消息内所有的媒体资源均适用。

asset_group_id 指示点云资源组标识，与 Asset 关系信息描述符中的 asset_group_id 对应。

pending_flag 指示媒体资源组中所有的数据组件是否均可被渲染。取值为 1 时，所有数据均可被渲染。

3D_spatial_region_info_flag 指示是否携带三维空间区域信息。取值为 1 时，表示消息中携带三维空间信息。

num_regions 指示三维空间区域的数量。

3DSpatialRegionStruct(1) 指示该媒体资源组对应的空间区域的信息，在不同场景中可使用不同的空间区域结构，在进行点云数据传输时，采用 PCC3DSpatialRegionStruct() 结构。

8.5 Asset 关系信息描述符

8.5.1 概述

SMT 中规定的 Asset 关系信息描述符用于指示同一个 SMT 数据包中资源间的关联关系。在点云的多个组件轨道具备联合质量等级时，通过 combine_quality_flag 表示新的联合质量等级关系，对相应的联合质量等级进行指示。

8.5.2 语法

Asset关系信息描述符的语法见表9。

表 9 Asset 关系信息描述符语法

语法	值	位数	备注
Asset_relationship_information_descriptor()			
descriptor_tag	'11'	16	uimsbf
descriptor_length		16	uimsbf
library_flag		1	blsbf
reserved		2	blsbf
combine_quality_flag		1	blsbf
dependency_flag		1	blsbf
composition_flag		1	blsbf
equivalence_flag		1	blsbf
similarity_flag		1	blsbf
if(dependency_flag) {			
num_dependencies	N1	8	uimsbf
for(i = 0; i < N1; i++) {			
asset_id()			
}			
}			
if(composition_flag) {			
asset_group_id_flag	N2	1	blsbf
if(asset_group_id_flag){			
asset_group_id		8	uimsbf
}			
num_compositions		8	uimsbf
for(i = 0; i < N2; i++) {			
asset_id()			
}			
}			
if(equivalence_flag) {			
equivalence_selection_level	N3	8	uimsbf
num_equivalences		8	uimsbf
for(i = 0; i < N3; i++) {			
asset_id()			
equivalence_selection_level		8	uimsbf
}			
}			
if(similarity_flag) {			
similarity_selection_level	N4	8	uimsbf
num_similarities		8	uimsbf
for(i = 0; i < N4; i++) {			
asset_id()			
similarity_selection_level		8	uimsbf
}			
}			
if(combine_quality_flag) {			
combine_quality_ranking	N5	8	uimsbf
num_combine_assets		8	uimsbf
for(i=0; i<N5;i++){			
asset_id()			
}			
}			
if(library_flag){			
num_libraries	N6	8	unimsbf
for(i = 0; i < N6; i++) {			
asset_id()			
}			
}			

8.5.3 语义

- descriptor_tag**该字段为16位。用于指示此类型描述符的标签值。
- descriptor_length**该字段为16位。指示此描述符的字节长度，从下一个字段计算至最后一个字段。
- library_flag**表示是否需要添加非对齐时间段的知识位流依赖关系。值为‘0’表示不需要添加，值为‘1’表示需要添加。
- combine_quality_flag**该字段为1位。取值为1时表示具备组合关系的多个Asset整体具备一个联合质量等级；取值为0时表示具备组合关系的多个Asset不存在联合质量等级。
- dependency_flag**该字段为1位。指示在此描述符中是否需要添加依赖关系。值‘0’意味着不需要添加。
- composition_flag**该字段为1位。指示在此描述符中是否需要添加组合关系。值‘0’意味着不需要添加。
- equivalence_flag**该字段为1位。指示在此描述符中是否需要添加等同关系。值‘0’意味着不需要添加。
- similarity_flag**该字段为1位。指示在此描述符中是否需要添加相似关系。值‘0’意味着不需要添加。
- num_dependencies**该字段为8位。指示此描述符所描述的Asset所依赖的Asset的数目。
- asset_group_id_flag**指示组合关系中是否包含asset_group_id标识符。
- asset_group_id**指示该组合关系的标识符。
- num_compositions**该字段为8位。指示与此描述符所描述的Asset有组合关系的Asset的数目。
- equivalence_selection_level**该字段为8位。指示所对应的Asset在等同关系组中的呈现等级。‘0’值表示该Asset被默认呈现。当默认Asset无法被选择时，拥有呈现等级较小的Asset会作为替代被选择和呈现。
- num_equivalences**该字段为8位。指示与此描述符所描述的Asset有等同关系的Asset的数目。
- similarity_selection_level**该字段为8位。指示所对应的Asset在相似关系组中的呈现等级。‘0’值表示该Asset被默认呈现。当默认Asset无法被选择时，拥有呈现等级较小的Asset会作为替代被选择和呈现。
- num_similarities**该字段为8位。指示与此描述符所描述的Asset有相似关系的Asset的数目。
- combine_quality_ranking**指示多个Asset整体的联合质量等级。拥有联合质量等级较小的Asset整体具有更好的呈现质量。
- num_combine_assets**指示与此描述符所描述的Asset有联合质量等级关系的Asset的数目。
- num_libraries**指示此描述符所描述的Asset所依赖的非对齐时间段的知识位流Asset的数目。
- asset_id:** 该字段指示Asset的标识符， Asset关系信息描述符中的asset_id:
- a) 当 Asset 关系信息描述符用于指示依赖关系时，asset_id 字段指示此描述符所描述的 Asset 所依赖的 Asset 的标识符，同时，此描述符中提供的 Asset 标识符顺序与其内部编码依赖层次相对应；
 - b) 当 Asset 关系信息描述符用于指示组合关系时，asset_id 字段指示与此描述符所描述的 Asset 有组合关系的 Asset 的标识符；
 - c) 当 Asset 关系信息描述符用于指示等同关系时，asset_id 字段指示与此描述符所描述的 Asset 有等同关系的 Asset 的标识符；
 - d) 当 Asset 关系信息描述符用于指示相似关系时，asset_id 字段指示与此描述符所描述的 Asset 有相似关系的 Asset 的标识符；
 - e) 当 Asset 关系信息描述符用于指示联合质量关系时，asset_id 字段指示与此描述符所描述的 Asset 有联合质量关系关系的 Asset 的标识符；
 - f) 当 Asset 关系信息描述符用于指示非对齐时间段的知识位流依赖关系时，asset_id 字段指示与此描述符所描述的 Asset 有非对齐时间段的知识位流依赖关系的 Asset 的标识符；该字段位数及具体用法见 T/AI 114-2024。

8.6 交互反馈消息

8.6.1 定义

交互反馈消息（Interaction Feedback Message）提供沉浸式媒体消费时，服务器与客户端之间的交互反馈。当沉浸式媒体消费中的服务器与客户端之间需要发送交互反馈信息时，使用此消息进行会话。一个交互反馈消息信令中可包含一个或多个交互反馈信令表。

8.6.2 语法

交互反馈消息的语法见表10。

表 10 交互反馈消息语法

语法	值	位数	类型
interaction_feedback_message()			
message_id		16	uimsbf
version		8	uimsbf
length	N1	32	uimsbf
extension{			
number_of_tables		8	uimsbf
for(i=0;i<N1;i++){			
table_id;		8	uimsbf
table_version;		8	uimsbf
table_length		16	uimsbf
}			
message_payload {			
for (i=0; i<N1; i++) {			
table()			
}			
}			
}			

8.6.3 语义

message_id 指示交互反馈消息的标识符。

version 指示交互反馈消息的版本。新的版本所携带的信息将覆盖任何之前的旧版本。

length 指示了以字节计算的交互反馈消息的长度，即从下一字段起直到交互反馈消息最后一个字节的长度。该字段取值不能为 0。。

number_of_tables 指示交互反馈消息中包含的信令表的数量。

table_id 指示交互反馈消息中包含的表的识别符。这是表中包含在交互反馈消息的有效负载中的 table_id 字段的一个副本。

table_version 指示交互反馈消息中所包含的表的版本。这是包含在交互反馈消息的有效负载中的表的版本字段的一个副本。

table_length 包含在交互反馈消息的有效负载中的表的长度字段的一个副本。

table() 指示一个交互反馈信令表实体。在有效负载中的该表与扩展域中 table_id 出现的顺序相同。一个交互反馈信令表可以作为一个 table() 的实例。

8.7 交互反馈信令表

8.7.1 定义

交互反馈信令表中包含了服务器和客户端之间交互反馈的信息，不同类型的交互反馈信令表用于指示不同类型的交互反馈信息。

8.7.2 语法

交互反馈信令的语法见表11。

表 11 交互反馈信令语法

语法	值	位数	类型
interaction_feedback_table()			
table_id		8	uimsbf
version		8	uimsbf
length		16	uimsbf
table_payload {			
table_type		8	uimsbf
timestamp		32	uimsbf
message_source		1	bool
asset_group_flag		1	bool
reserved		6	uimsbf
if(asset_group_flag){			
asset_group_id		8	uimsbf
}			
else{			
asset_id()			
}			
if (table_type == 0) {			
3DoF+_flag		1	bool
interaction_target		8	uimsbf
interaction_type		8	uimsbf
if(interaction_target==1){			
ClientRegion()			
}			
if(interaction_target==2){			
ClientRotation()			
}			
if(3DoF+_flag==1 && interaction target ==3){			
ClientPosition()			
behavior_coefficient			
}			
if(table_type == 1){			
ClientPosition()			
ClientOrientation()			
last_processed_media_timestamp			
}			
if(table_type == 2){			
ClientCam_pos_x;		32	float
ClientCam_pos_y;		32	float
ClientCam_pos_z;		32	float
}			
}			
}			

8.7.3 语义

table_type 指示交互反馈信令表携带的信息类型。其取值含义见表 12。在基于 SMT 的点云数据传输时，table_type 的值只能取 1。

表 12 交互反馈信令信息类型取值表

取值	含义
0	全景视频用户位置变动信息
1	容积媒体用户位置变动信息
2	自由视角视频用户位置变动信息
3..255	未定义

timestamp 指示当前交互产生的时间，使用 UTC 时间。

message_source 指示消息源，取值为 0 表示交互反馈消息是客户端发往服务器，取值为 1 表示交互反馈消息是服务器发往客户端。在本文件的规范中，该字段应取值为 0。

asset_group_flag 取值为 1 表示客户端当前消费内容属于一个资源组；取值为 0 表示客户端当前消费内容不属于资源组。

asset_group_id 指示客户端当前消费内容的资源组标识符，与 Asset 关系信息描述符中的 asset_group_id 对应。

asset_id 指示客户端当前消费内容的 asset_id。

3DoF+_flag 指示 3DoF+ 视频内容。

reserved 保留字节位。

interaction_num 指示当前信令中包含的交互的数目。

interaction_target 指示客户端当前交互的目标，包括头盔设备当前状态、用户关注目标、用户当前状态等；交互目标的取值见表 13。

表 13 交互目标取值表

类型	取值	含义
Null	0	交互目标为空，即没有特定交互目标
HMD_status	1	交互目标为头盔设备当前状态
Object of interests	2	交互目标为用户关注区域当前状态
User_status	3	交互目标为用户当前状态

interaction_type 指示交互类型。在本文件的规范中，该字段应取值为 0

interaction_content_length 指示客户端对于当前的交互行为，其具体交互内容的长度。该字段取值以字节为单位，为从该字段下一字节起到当前交互内容结束的长度。

ClientRegion 指示客户端视窗的尺寸和屏幕分辨率。

ClientRotation 指示视角方向，即用户实时的视角相对初始视角的变化。

behavior_coefficient 指示放大行为系数。

ClientPosition() 指示用户当前在全局坐标系中的位置，见 8.8。

ClientOrientation() 指示用户在以当前位置建立的笛卡尔坐标系中的观看方向，见 8.9。

last_processed_media_timestamp 指示已加入解码器缓冲区的最后一个媒体单元的时间戳。SMT 发送实体使用此字段从容积媒体播放器的新 asset 中确定下一个传输的媒体单元。下一个媒体单元是紧随该时间戳后的带有时间戳或序号的媒体单元。SMT 发送实体从随后的媒体时间戳开始，从传输先前的 asset（根据先前的视窗确定）切换到传输新的 asset（根据新的视窗确定），以减少接收对应于新视窗媒体内容的延迟。

ClientCam_pos_x、ClientCam_pos_y、ClientCam_pos_z 分别指示目标视角对应的虚拟相机的 x 轴、y 轴，z 轴分量。

8.8 用户实时位置

用户实时位置结构体表示用户在虚拟场景中相对起始位置的位移，在 3DoF 时该结构中所有值为 0，

在 3DoF+ 时为非 0 值且取值范围应在约束范围内。

```
aligned(8) class ClientPosition () {
    signed int(16) position_x;
    signed int(16) position_y;
    signed int(16) position_z;
}
```

position_x 指示用户实时位置相对起始位置沿着 x 轴位移，取值范围为 $(-2^{15}, 2^{15} - 1)$ ，以毫米为单位。

position_y 指示用户实时位置相对起始位置沿着 y 轴位移，取值范围为 $(-2^{15}, 2^{15} - 1)$ ，以毫米为单位。

position_z 指示用户实时位置相对起始位置沿着 z 轴位移，取值范围为 $(-2^{15}, 2^{15} - 1)$ ，以毫米为单位。

8.9 用户实时观看方向

用户实时观看方向结构体表示用户在以当前位置建立的笛卡尔坐标系中的观看方向。

```
aligned(8) class ClientOrientation () {
    signed int(32) dir_x;
    signed int(32) dir_y;
    signed int(32) dir_z;
}
```

dir_x 指示用户观看方向在 x 轴上的坐标，以用户所在位置为原点建立笛卡尔坐标系。

dir_y 指示用户观看方向在 y 轴上的坐标，以用户所在位置为原点建立笛卡尔坐标系。

dir_z 指示用户观看方向在 z 轴上的坐标，以用户所在位置为原点建立笛卡尔坐标系。

9 基于 DASH 的点云数据传输

9.1 概述

DASH 服务器基于 AVS PCC 的媒体文件进行切片封装并创建 MPD。DASH MPD 创建器生成点云特定的描述子，描述点云的基本信息、组件信息、空间区域等。这些信息可根据媒体文件中的相应信息产生。

DASH 客户端获得由服务器或其他方式传输/提供的 MPD，从 MPD 中获取并使用特定描述子信息以及预选择，选择相应的自适应集（Adaptation Set）和表示（Representation），访问服务器端并获取 Adaptation Set 和 Representation 对应的全部、部分点云片或某个三维空间区域的一个或多个点云数据片段，根据获取的一个或多个媒体文件的片段的生成呈现相应的点云数据内容。

基于对点云的不同访问需求，DASH 可支持单轨道、基于组件的多轨道和基于点云片的多轨道的封装传输。

9.2 基于单轨道的传输

DASH 中的单轨模式应由一个包含一个或多个 Representation 的 Adaptation Set 来表示。如果一个 Representation 由多个媒体片段组成，则应存在一个初始化媒体片段。初始化媒体片段应包含一个带有 AVS PCC 参数集的 AVSPCCDecoderConfigurationRecord。

部分属性的使用限制如下：

a) @codecs 参数应出现在 Adaptation Set 层级，并应指出解码 Adaptation Set 中任何 Representation 的最大解码能力。如果@codecs 参数与 Adaptation Set 层级的参数不同，则应在 Representation 层级中

指出解码能力。

- b) Representation 层级中@codecs 参数应表明解码 Representation 中任何组件所需的能力。
- c) @maxWidth 和@maxHeight 参数不应在任何 Adaptation Set 指出。
- d) @width 和@height 不应在任何 Representation 指出。

9.3 基于组件的多轨道的传输

9.3.1 概述

每个点云组件应在 DASH MPD 文件中表示为单独的 Adaptation Set，称为组件适配集（Component Adaptation Set）。包含几何信息的 Adaptation Set 也是作为点云的内容接入点的主适配集（Main Adaptation Set）。Main Adaptation Set 应包含 Adaptation Set 层级的单个初始化片段或 Representation 层级的多个初始化片段（每个 Representation 包含一个）。初始化片段应包含 T/AI 128.2-2024 中规定的 sequence_header 参数集，该参数集是初始化点云解码器所必需的。

9.3.2 预选择

点云的预选择应符合 ISO/IEC 23009-1 中的规定，在 MPD 中通过预选择（Preselection）元素和 @preselectionComponents 属性的标识列表（包括 Main Adaptation Set 的标识，以及后面跟随的 Component Adaptation Set 的标识）表示。Preselection 的@codecs 属性应设置为'apcc'，表示预选择的媒体是点云。点云的预选择可以使用周期（Period）元素中的 Preselection 元素或 Main Adaptation Set 中 Adaptation Set 层级的 Preselection 描述子来指示。

9.3.3 支持多版本的点云

同一点云数据的多个版本应使用单独的 Preselection 指出。如果预选择是相互替代的，表示相同点云数据的替代版本 Preselection 应包含具有相同@apcId 值的点云描述子。Main Adaptation Set 的标识是 Preselection 中 Adaptation Set 标识列表中的第一个标识符，然后是 Component Adaptation Sets 的标识符。

9.4 基于点云片的多轨道的传输

9.4.1 概述

当 AVS 文件采用基于点云片的文件封装时，点云片基础轨道应在 DASH MPD 文件中表示为 Main Adaptation Set，且 Main Adaptation Set 中的@codecs 属性应取值为'apeb'或'apcb'。每个点云片轨道应在 DASH MPD 文件中表示为一个 Adaptation Set，即 Slice Component Adaptation Set，且其 Slice Component Adaptation Set 中的@codecs 属性应取值为'apsl'。当 slice Component Adaptation Set 中仅包含相应点云片的几何或属性数据时，Main Adaptation Set 的@codecs 属性应取值为'apcb'，并且该 Slice Component Adaptation Set 应包含一个点云组件描述子；当 slice Component Adaptation Set 中包含相应点云片全部的几何和属性数据时，Main Adaptation Set 的@codecs 属性应取值为'apeb'。

当 Slice Component Adaptation Set 中仅包含相应点云片的几何或属性数据时，该 Slice Component Adaptation Set 应包含一个点云组件描述子。

9.4.2 点云片预选择

当 AVS 文件采用基于点云片的文件封装时，应使用预选择（Preselection）元素对点云片基础轨道对应的 Main Adaptation Set 以及点云片轨道对应的一个或多个 Slice Component Adaptation Set 的点云片预选择进行指示，Preselection 的@codecs 属性应设置为。该点云片预选择可以位于周期（Period）层级或 Adaptation Set 层级。

一个点云片预选择包含几何组件的 Slice Component Adaptation Set 和其对应的属性组件的 Slice Component Adaptation Set，其中，几何组件的 Slice Component Adaptation Set 可看作是点云片预选择中的 Main Adaptation Set。Preselection 的@preselectionComponents 属性的标识列表应设置为几何组件的

Slice Component Adaptation Set 的标识以及后面跟随相对应的属性组件的 Slice Component Adaptation Set 的标识。

当存在多个分片轨道时，几何组件 Slice Component Adaptation Set 中的每个 Representation 使用 ISO/IEC 23009-1 中规定的@dependencyId 属性关联到 Main Adaptation Set 中相应的 Representation，即 @dependencyId 属性应设置为 Main Adaptation Set 中 Representation 的标识符。

当每个分片轨道表示一种类型的组件数据时，属性组件 Slice Component Adaptation Set 中的每个 Representation 使用@dependencyId 属性关联到几何组件的 Slice Component Adaptation Set 中相应的 Representation。

每个基于点云片的 Preselection 可以包含一个或多个点云片信息描述子，用于识别每个 Preselection 中引用的点云片的标识以及空间区域的标识。

9.5 基于点云的 MPD 描述子

9.5.1 XML 命名空间和规则

本文件规定的 XML 元素和属性在单独的命名空间 "urn:avs:pccs:2022" 中定义。命名空间指示符 'pccs' 用于指代这个命名空间。

9.5.2 点云组件描述子

点云组件描述子(AVS PCC component descriptor)可识别 Component Adaptation Set 中点云组件的类型。点云组件描述子是一个 EssentialProperty 元素，其 @schemeIdUri 属性设置为 "urn:avs:pccs:2022:component"。

在 Adaptation Set 层级，存在于 Component Adaptation Set 的 Representation 中的每个点云组件应由一个点云组件描述子来表述。

点云组件描述子的元素和属性见表 14。

表 14 点云组件描述子元素和属性

元素和属性	使用	数据类型	描述
component	0..1	avspcc:avspccComponentType	一个元素，其属性指定适应集的表示中存在的 AVS 点云组件之一的信息。
component@type	M	xs:string	指示点云组件的类型。取值'geom'表示几何组件，取值'attr'表示属性组件。
component@attr_type	CM	xs:uintVectorType	指示属性组件的类型列表，不同类型值之间以空格分隔。只允许使用 0 到 255 之间的值(包括 0 和 255)。 仅当组件是点云属性数据时才应存在(即 @component_type 的值为'attr')。
component@attr_group_id	O	xs:uint	指示点云属性组件所属的属性组标识符。同属于一个属性组的不同属性数据存在编解码或者呈现上的依赖关系。

当在主适配集中包含多个 Representation 时，每个 Representation 的初始化段应包含该 Representation 的点云参数集。组件适配集的表示应使用 ISO/IEC 23009-1 中规定的@dependencyId 属性指示主适配集的相应标识符。

9.5.3 点云描述子

点云描述子 (AVS PCC descriptor) 可识别 MPD 文件中的点云流，用户终端可以通过该描述子区分不同的点云流。

点云描述子是一个 SupplementalProperty 元素，其 @schemeIdUri 属性设置为 "urn:avs:pccs:2022:avspcc"。

在点云的 Main Adaptation Set 的 Adaptation Set 层级或 Preselection 层级最多可以存在一个点云描述子。点云描述子的元素和属性见表 15。

表 15 点云描述子元素和属性

元素和属性	使用	数据类型	描述
apcc:@apcId	CM	xs:string	AVS 点云数据标识符。如果在单独的 Adaptation set 中指出点云数据的多个版本，则该属性应存在。

9.5.4 点云空间描述子

9.5.4.1 静态空间区域

如果三维空间区域是静态的(即, 每个三维空间区域的锚点和尺寸信息不会随着呈现时间而改变), 则三维空间区域的特征信息以及该区域与点云片之间的映射应使用点云空间描述子 (AVSPCCSpatialRegions, APSR)指出。该描述子是一个 SupplementalProperty 元素, 其 @schemeIdUri 属性设置为 “urn:avs:pccs:2022:apsr”。

在 Main Adaptation Set 中的 Adaptation Set 层级或 Representation 层级, 或 Preselection 层级包含单个点云空间描述子。使用点云空间描述子的 @sliceIds 属性来指示三维空间区域和相应点云片组件流的映射。点云空间描述子的 @value 属性不应存在。

点云空间描述子规定的元素和属性见表 16。

表 16 点云空间描述子元素和属性

元素和属性	使用	数据类型	描述
apsr	0..1	avspcc:spatialRegionMapType	容器元素, 其属性和元素指示 3D 空间区域和点云片之间的映射。
apsr.spatialRegion	1..N	apsr:spatialRegionType	一个元素, 其属性指示一个 3D 空间区域, 并指示了该区域和多个点云片之间的映射。
apsr.spatialRegion@id	M	xs:unsignedShort	3D 空间区域的标识符。该属性的值应与文件封装中的相应三维区域的 3d_region_id 字段的值匹配。
apsr.spatialRegion@x	OD	xs:int	指示空间区域的边界框参考点的 x 坐标。 如果不存在, 则缺省值为 0。
apsr.spatialRegion@y	OD	xs:int	指示空间区域的边界框参考点的 y 坐标。 如果不存在, 则缺省值为 0。
apsr.spatialRegion@z	OD	xs:int	指示空间区域的边界框参考点的 z 坐标。 如果不存在, 则缺省值为 0。
apsr.spatialRegion@d_x	M	xs:int	指示边界框沿 x 轴的长度 (即宽度)。负值表示在轴的负方向上延伸的长度。
apsr.spatialRegion@d_y	M	xs:int	指示边界框沿 y 轴的长度 (即高度度)。负值表示在轴的负方向上延伸的长度。
apsr.spatialRegion@d_z	M	xs:int	指示边界框沿 z 轴的长度 (即深度)。负值表示在轴的负方向上延伸的长度。

表 16 点云空间描述子元素和属性（续）

元素和属性	使用	数据类型	描述
aspr.spatialRegion@sliceIds	O	xs:UIntVectorType	空格分隔的点云片标号列表。该标识符标识的点云片与 3D 空间区域相关联。
apsr.spatialRegion@significanceLevel	O	xs:unsignedShort	指示当前空间区域点云数据的显著度等级信息。

9.5.4.2 动态空间区域

当 3D 空间区域是随时间动态变化的动态空间区域时，由动态的三维空间区域定时元数据指出的呈现时间轴上每个三维空间区域的锚点和尺寸信息应承载于具有唯一 Representation 的单独的 Adaptation Set 中，并且使用@associationId 属性与 Main Adaptation Set 的 Representation 关联，Adaptation Set 或 Representation 中@associationType 值应设置为 4CC 代码'apdy'。

9.5.5 点云片信息描述子

点云片信息描述子(AVSPCCSlices, APSI)用于描述相应点云片轨道的相关信息。点云片信息描述子是一个 SupplementalProperty 元素，其@schemeIdUri 属性设置为"urn:avs:pccs:2022:apsi"。

至多一个点云片信息描述子存在于 slice Component Adaptation Set。一个或多个点云片信息描述子也可以存在于一个基于点云片的 Preselection 中。

点云片信息描述子规定的元素和属性见表 17。

表 17 点云片信息描述子的元素和属性

元素和属性	使用	数据类型	描述
apsi	0..N	avspcc:sliceInfoType	容器元素，其属性指定相应点云片轨道的相关信息。
apsi@sliceIds	M	xs:UIntVectorType	点云片标号列表，指示相应点云片轨道对应的点云片标号。
apsi@regionIds	O	xs:UIntVectorType	三维空间区域标识符列表，指示相应点云片轨道对应的三维空间区域的标识符。

9.5.6 点云对象描述子

点云对象描述子(AVSPCCObject, APOB)用于指示 MPD 文件中点云对象的信息。该描述子是一个 SupplementalProperty 元素，其 @schemeIdUri 属性设置为 "urn:avs:pccs:2022:apob"。

在 Main Adaptation Set 中的 Adaptation Set 层级或 Representation 层级，或 Preselection 层级包含单个点云对象描述子。点云对象描述子的@value 属性不应存在。

点云对象描述子规定的元素和属性见表 18。

表 18 点云对象描述子的元素和属性

元素和属性	使用	数据类型	描述
apob	0..1	avspcc:objectType	容器元素，其属性和元素指定点云对象信息。
apob.object	1..N	apob:objectInfoType	一个元素，其属性表示一个点云对象，提供了点云对象的信息描述。
apob.object@info_id	M	xs:unsignedShort	点云对象的标识符。 该属性的值应与 ISOBMFF 容器中的相对应对象的 object_info_id 字段的值匹配。

表 18 点云对象描述子的元素和属性（续）

元素和属性	使用	数据类型	描述
apob.object@oid	O	xs:string	以 OID 的形式指示的点云对象信息。 该属性的值应与为 ISOBMFF 容器中的相对对象的 object_oid 字段的值匹配。 Oid 形式的对象信息和人眼可读的字符串形式的对象信息不能同时存在。
apob.object@label	O	xs:string	以人眼可读的字符串形式指示的点云对象信息。 该属性的值应与为 ISOBMFF 容器中的相对对象的 object_label 字段的值匹配。
apob.object@x	OD	xs:int	指示包含点云对象的空间区域的边界框参考点的 x 坐标。如果不存在，则缺省值为 0。
apob.object@y	OD	xs:int	指示包含点云对象的空间区域的边界框参考点的 y 坐标。如果不存在，则缺省值为 0。
apob.object@z	OD	xs:int	指示包含点云对象的空间区域的边界框参考点的 z 坐标。如果不存在，则缺省值为 0。
apob.object@dx	M	xs:int	指示边界框沿 x 轴的长度（即宽度）。负值表示在轴的负方向上延伸的长度。
apob.object@dy	M	xs:int	指示边界框沿 y 轴的长度（即高度）。负值表示在轴的负方向上延伸的长度。
apob.object@dz	M	xs:int	指示边界框沿 z 轴的长度（即深度）。负值表示在轴的负方向上延伸的长度。
apob.object@slicIds	O	xs:UIntVectorType	空格分隔的点云片标号列表。该片标号对应的点云片与点云对象相关联。

附录 A
(规范性)
文件标牌

A. 1 概述

符合 T/AI 128.2-2024 规定的点云位流在按本标准文件的规范进行文件封装时，根据：

- a) 点云位流的动态或静态特性。
- b) 点云位流的封装方式。

可以进一步区分为不同的文件标牌。使用文件标牌时，应符合本标准文件的相关约束以及 ISO/IEC 14496-12 标准的相关约束。

A. 2 动态点云数据存储

A. 2. 1 点云数据的单轨道封装

当动态点云数据进行单轨道封装时，可在 FileTypeBox 的 compatible_brands 列表中使用'apef'类型的标牌。包含'apef'标牌的文件阅读器应支持 7.1.2 规定的单轨封装格式。

A. 2. 2 基于组件的多轨道封装

当动态点云数据进行基于组件的多轨道封装时，可在 FileTypeBox 的 compatible_brands 列表中使用'apcf'类型的标牌。包含'apcf'标牌的文件阅读器应支持 7.1.3 规定的基于组件的多轨封装格式。

A. 2. 3 基于点云片的多轨道封装

当动态点云数据进行基于点云片的多轨道封装时，可在 FileTypeBox 的 compatible_brands 列表中使用'apsf'类型的标牌。包含'apsf'标牌的文件阅读器应支持 7.1.4 规定的基于点云片的多轨封装格式。

A. 3 静态点云数据存储

封装静态点云数据时，可在 FileTypeBox 的 compatible_brands 列表中使用'apim'类型的标牌。包含'apim'标牌的文件阅读器应支持 7.2 规定的静态点云数据的存储。

附录 B
(资料性)
可替换位流

B. 1 动态点云内容的可替换位流

动态点云内容对应的可替换位流可以独立地封装为一个或多个点云轨道，见图 B.1。

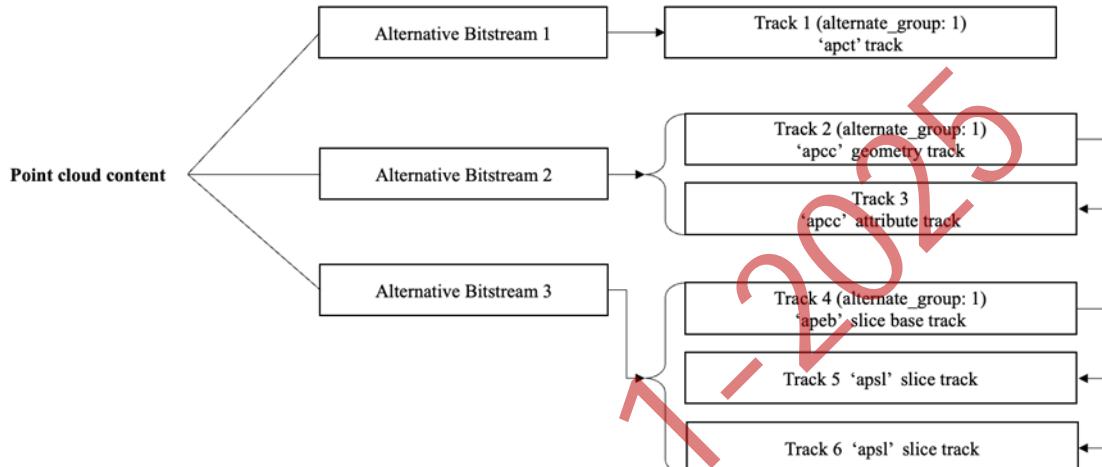


图 B. 1 可替换位流点云轨道封装结构

特别地，当可替换位流采用基于组件的多轨封装方式时，一个几何组件轨道可能同时属于多个可替换位流，见图 B. 2。

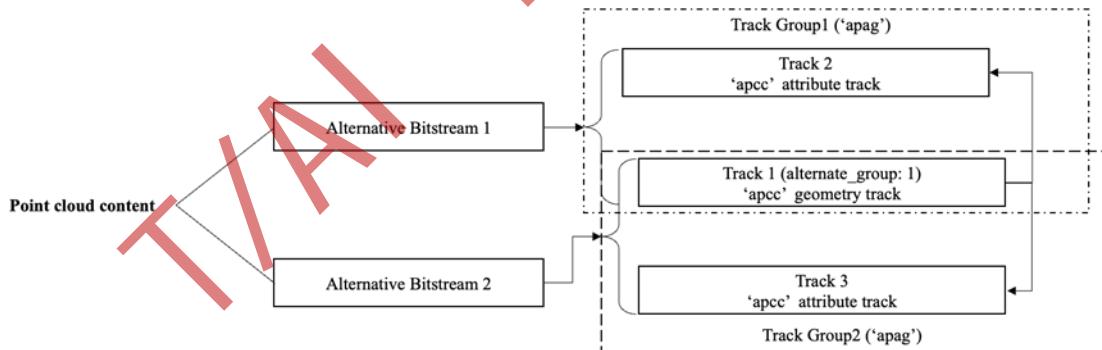


图 B. 2 可替换位流共用几何组件轨道结构

B. 2 静态点云内容的可替换位流

静态点云内容对应的可替换位流可以独立地封装为一个或多个点云项，见图 B. 3。

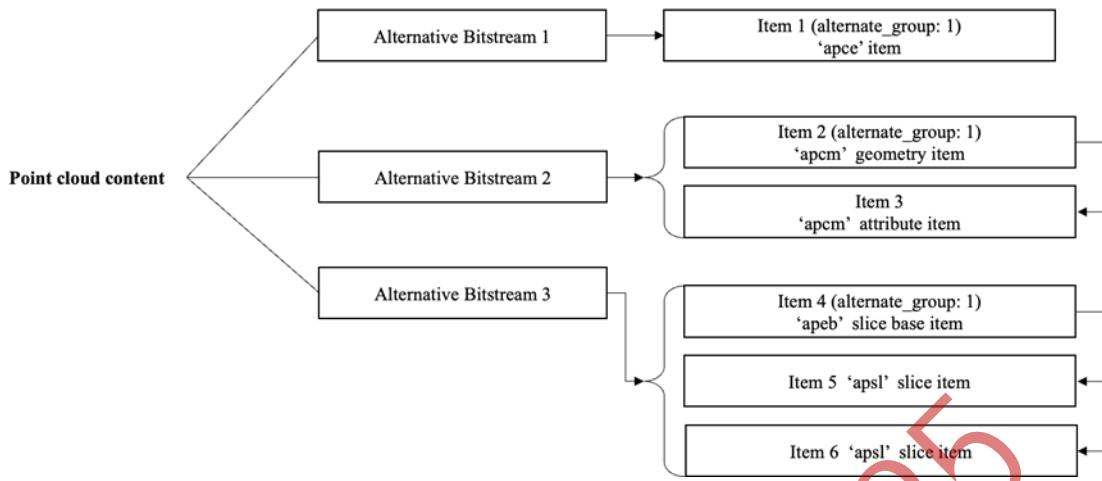


图 B.3 可替换位流点云项封装结构

特别地，当可替换位流采用基于组件的多点云项封装方式时，一个几何组件点云项可能同时属于多个可替换位流，见图 B.4。

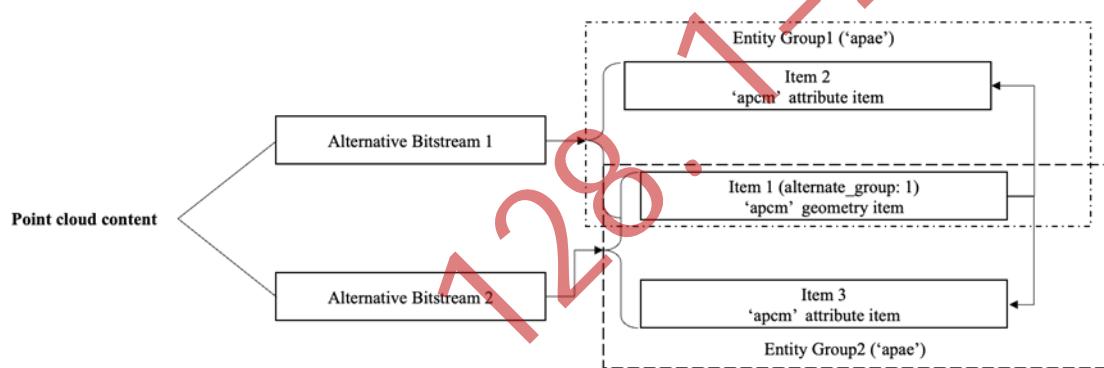


图 B.4 可替换位流共用几何组件点云项封装结构