

国 际 电 信 联 盟

ITU-R

国际电联无线电通信部门

ITU-R BS.2125-1 建议书
(05/2022)

音频定义模型的系列呈现

BS系列
广播业务（声音）



国际电信联盟

前言

无线电通信部门的职责是确保卫星业务等所有无线电通信业务合理、平等、有效、经济地使用无线电频谱，不受频率范围限制地开展研究并在此基础上通过建议书。

无线电通信部门的规则和政策职能由世界或区域无线电通信大会以及无线电通信全会在研究组的支持下履行。

知识产权政策（IPR）

ITU-R的IPR政策述于ITU-R第1号决议中所参引的《ITU-T/ITU-R/ISO/IEC的通用专利政策》。专利持有人用于提交专利声明和许可声明的表格可从<https://www.itu.int/ITU-R/go/patents/zh>获得，在此处也可获取《ITU-T/ITU-R/ISO/IEC的通用专利政策实施指南》和ITU-R专利信息数据库。

ITU-R 系列建议书	
(也可在线查询 https://www.itu.int/publ/R-REC/zh)	
系列	标题
BO	卫星传送
BR	用于制作、存档和播出的录制；电视电影
BS	广播业务（声音）
BT	广播业务（电视）
F	固定业务
M	移动、无线电定位、业余和相关卫星业务
P	无线电波传播
RA	射电天文
RS	遥感系统
S	卫星固定业务
SA	空间应用和气象
SF	卫星固定业务和固定业务系统间的频率共用和协调
SM	频谱管理
SNG	卫星新闻采集
TF	时间信号和频率标准发射
V	词汇和相关问题

说明：该ITU-R建议书的英文版本根据ITU-R第1号决议详述的程序予以批准。

电子出版
2023年，日内瓦

ITU-R BS.2125-1建议书*

音频定义模型的系列呈现

(2019-2022年)

范围

本建议书描述了基于ITU-R BS.2076建议书规定的音频定义模型（ADM）的元数据格式，该格式被分割为帧的时间序列。与原始ADM一样，XML格式用于ADM的系列呈现。ADM的系列呈现旨在用于线性工作流，例如广播和流媒体应用的直播或实时制作。本建议书不包含元数据的传送方法或比特打包，或者元数据有关的音频样本的格式。

关键词

ADM、音频定义模型、序列化、分割、高级音响系统、多声道音频、基于声道的、基于对象的、基于场景的、元数据、沉浸式音频

国际电联无线电通信全会，

考虑到

- a) 基于信息技术的所有类型存储介质已经渗透到有关无线广播的音频制作的所有领域，即非线性编辑、空中播放和存档；
- b) 当节目材料现场制作或直播分发时，需要通过通信和广播网络进行实时流式传送；
- c) 直播交换采用单一的流元数据格式将极大简化个别装备和远端演播室之间的互操作；
- d) 与当前ITU-R BS.2076建议书中规定的音频相关的元数据、音频定义模型（ADM）兼容将最小化格式转换所需的努力；
- e) 对于线性工作流，如广播和流媒体的直播和实时制作，需要基于帧或序列化的音频和元数据；
- f) 未来的音频系统将要求在线性流中携带与音频相关联的元数据；
- g) 高级音响系统将使用各种各样的配置，包括基于声道的、基于对象的和基于场景的音频，如ITU-R BS.2051建议书所规定；
- h) 高级音响系统将使用ITU-R BS.2076建议书中规定的音频定义模型（ADM）来描述所传送和交换的音频的技术格式；
- i) 高级音响系统将使用ITU-R BS.2088建议书中规定的BW64音频文件格式与ADM交换音频节目，

建议

- 1 对于需要基于音频定义模型（ADM）的序列化元数据的工作流，应使用附件1中所述的ADM的序列呈现；

* 无线电通信第6研究组于2023年9月根据ITU-R第1号决议对本建议书进行了编辑性修正。

2 注1被视为本建议书的一部分。

注1 – 可在自愿基础上遵循本建议书。但是，本建议书可能包括某些强制性的规定（以确保互操作性或适用性等），只有当所有这些强制性的规定都得到执行时才能使本建议书得到遵守。“须”或其他一些强制性语言（如“必须”）及其相应否定形式用以表示要求，使用这些词语绝不意味着要求部分或全部遵守本建议书。

附件1

音频定义模型（ADM）的系列呈现

目录

	页码
知识产权政策（IPR）	ii
附件1 音频定义模型（ADM）的系列呈现	2
A1.1 引言	3
A1.2 S-ADM概述	4
A1.3 S-ADM元数据帧的结构	9
A1.4 S-ADM元素和属性	11
A1.5 ID的使用	24
A1.6 与XML广播元数据的兼容性	25

A1.1 引言

ITU-R BS.2051建议书中规定的高级音响系统要求音频相关的元数据来处理基于声道的、基于对象的和基于场景的音频元素。ITU-R BS.2076建议书提供了在可扩展标记语言（XML）中所述的音频定义模型（ADM），作为高级音响系统音频相关的元数据。

ITU-R BS.2088建议书提供了64位音频文件格式“BW64”，它可以将ADM元数据的XML代码存储在“axml”数据块中，并用于交换高级音响系统的音频节目。

不过，ADM不适合用于实时制作和流式音频应用。实时制作和流式音频应用涉及将现有音频文件切片成帧或生成帧，并通过传送接口实时传送这些帧（例如AES3（ITU-R BS.647建议书）、MADI（ITU-R BS.1873建议书）、SDI（ITU-R BT.1120建议书、BT.1365建议书和BT.2077建议书）和IP网络）。因此，需要一个串行的ADM格式，以允许切片音频及其相关的元数据。

本建议书描述了ADM如何以串行元数据的格式来表示，以便用于高级音响系统的现场制作和流式音频应用。它涵盖了元数据的分段和元数据的串行格式。该串行格式具有以下特征：

- 它与ADM的结构、属性和元素相兼容。
- 它对可以描述的音轨数量没有任何限制。
- 它独立于传送或接口方法。
- 它可以处理ITU-R BS.2076建议书中所述的、基于声道的、基于对象的和基于场景的音频节目的任何组合。
- 它对帧大小没有任何限制。
- 它允许随机访问支持。

本建议书未描述如何在特定接口中承载、约束或传送ADM的串行表示（S-ADM），也未描述如何将S-ADM元数据帧与音频实质内容相关联。

A1.1.1 定义

音频实质内容 – 音频信号的数据，以样本或代表这些样本的编码数据来表示。

数据块 – 是来自S-ADM帧的元数据元素的一个子集。

流 – S-ADM帧的一个序列称为一个流。一个流即等同于普通ADM中一个文件的S-ADM。因此，一个流可以包含一个或多个**audioProgramme**。使用UUID形式的**flowID**来标识流。

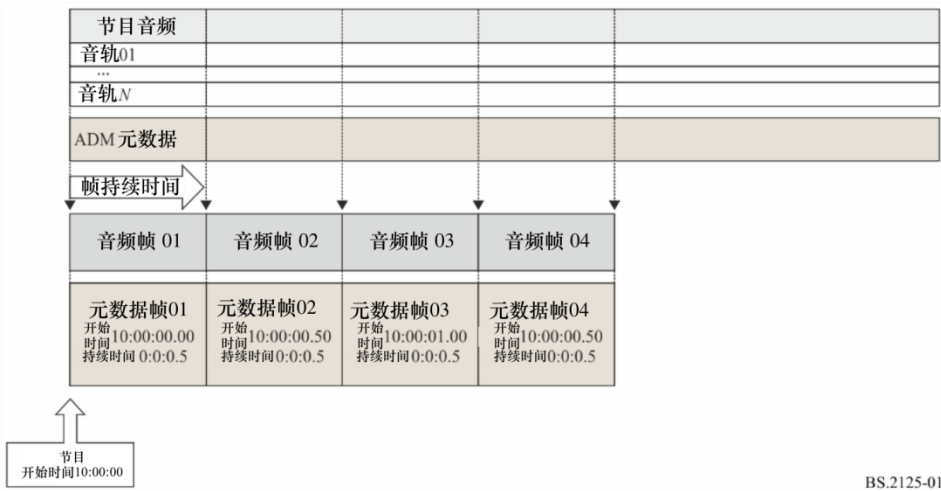
随机访问 – 访问一个流中任何帧并可完全解码之的能力。在本建议书的上下文中，它指的是访问一个流中任意帧并提取对应音频帧所需之所有元数据的能力。在某些情况下，可能需要多个帧来提取该元数据（具有一定延迟的随机访问）。当不支持随机访问时，可能需要流中的所有先前帧（从第一帧开始）来提取某个给定音频帧所需的元数据。

A1.2 S-ADM概述

一个S-ADM元数据的帧包含一组元数据，在与该帧相关联的时间段内至少用于描述音频帧。S-ADM具有与ADM相同的结构、属性和元素，以及用于规定帧格式的附加属性和元素（见第A1.4节）。S-ADM帧不重叠，并且与指定的持续时间和开始时间相连。不过，包含在一个S-ADM帧中的元数据可以用于描述超过该帧持续时间的音频。本建议书未涉及音频帧的分段和传送问题。

ADM元数据由内容部分（例如， **audioProgramme**）和格式部分（例如， **audioChannelFormat**）组成。只有**audioProgramme**、**audioObject**和**audioBlockFormat**三个元素有与时间相关的参数。在内容部分中， **audioProgramme**或**audioObject**的**start**、**end**和**duration**用于确定元素的开始时间、结束时间或持续时间。这些参数通常是固定的。在格式部分中， **audioBlockFormat**中的所有参数都是时变参数。

图1
S-ADM概述



BS.2125-01

ADM元数据可以分成两组，即时变动态元数据（如 **audioChannelFormat** 中的 **audioBlockFormat**）和时不变静态元数据（如**audioProgramme**和**audioContent**）。

S-ADM元数据帧须由一个或多个元数据块组成。

S-ADM元数据帧分为五种类型：

- “报头”：表示流中第一帧，包含与音频信号相关联的所有描述符。
- “完整”：与音频信号相关联的所有描述符。
- “分割”：元数据分为数据块，最后一个数据块包含动态元数据，其他数据块包含部分静态元数据。
- “中间”：只包含变自前一帧的描述符。
- “全部”：整个**audioProgramme**的所有描述符（初始ADM的整个XML代码）。

S-ADM流应为以下之一：

- 完整帧（FF）流：一系列“完整”帧，第一帧为“完整”、“报头”或“全部”。
- 中间帧（IF）流：一系列“中间”帧，第一帧为“完整”、“报头”或“全部”。

- 混合帧（MF）流：一系列“中间”和“完整”帧，第一帧为“完整”、“报头”或“全部”。
- 分割帧（DF）流：一系列“分割”帧，第一帧为“完整”、“分割”、“报头”或“全部”。

“分割”和“中间”帧的目的是通过不在每个帧中重复时不变的元数据，使S-ADM数据能够有效表示。S-ADM流类型旨在支持这种有效的表示，同时在需要时提供随机访问功能。设想的S-ADM流应用如下所示：

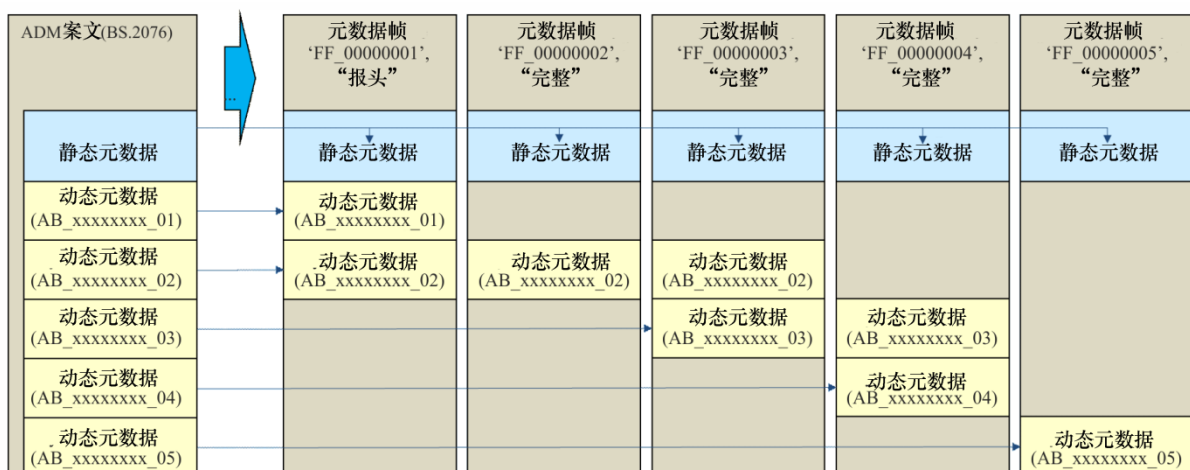
用例	建议的流
– 元数据主要是动态的，并且会改变每一帧，或者 – 每帧都需要随机访问	完整帧（FF）
– 流包括静态或缓慢变化的动态元数据，以及 – 不需要随机访问	中间帧（IF）
– 流包括静态或缓慢变化的动态元数据，以及 – 需要随机访问（但不是针对每一帧）	混合帧（MF）
– 流包括静态或缓慢变化的动态元数据，以及 – 需要随机访问（但不是针对每一帧），以及 – 希望更均匀地在所有帧之间分发数据	分割帧（DF）

除非另有规定，否则本建议书中表示的数值应假定用十进制符号表示，例如ID用第A1.5段所示的十六进制格式表示。

A1.2.1 完整帧（FF）流说明

在这种情况下，S-ADM的基本结构被构建成“完整”帧（如图1所示）。FF流提供对任何音频帧的访问，以支持随机访问（见图2）。

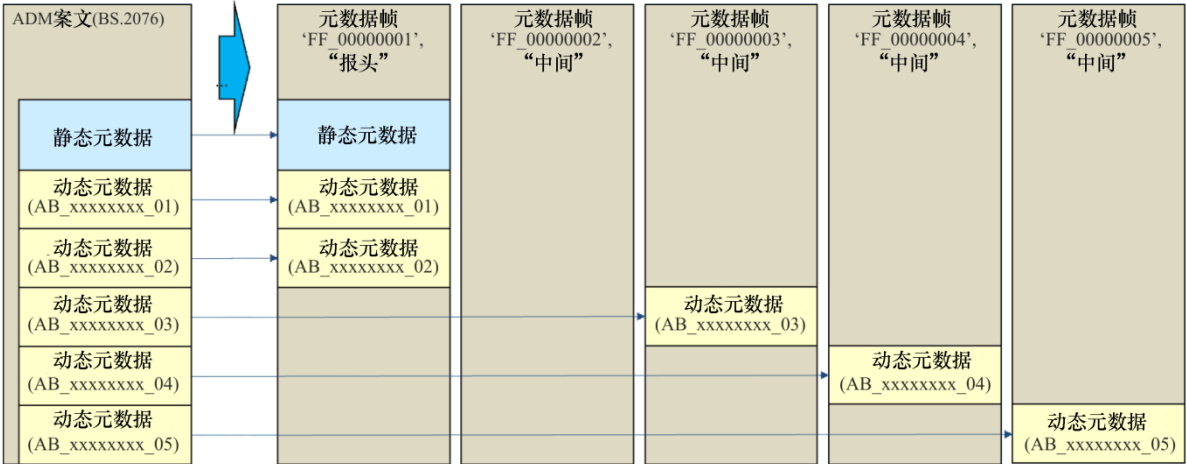
图2
完整帧（FF）流中S-ADM的基本结构



A1.2.2 中间帧（IF）流说明

接收机可能只需要接收一次静态ADM元数据，因此即使重复传送完整的元数据，也可以忽略任何重复的静态ADM元数据。因此，当广播公司不需要随机访问时，可以省略已经传送的ADM元数据。“中间”帧可以省略其值与前面帧中的值相比未发生变化的所有元素，即使该元素归类为动态元数据。IF流不支持随机访问（见图3）。

图3
中间帧（IF）流中S-ADM的结构

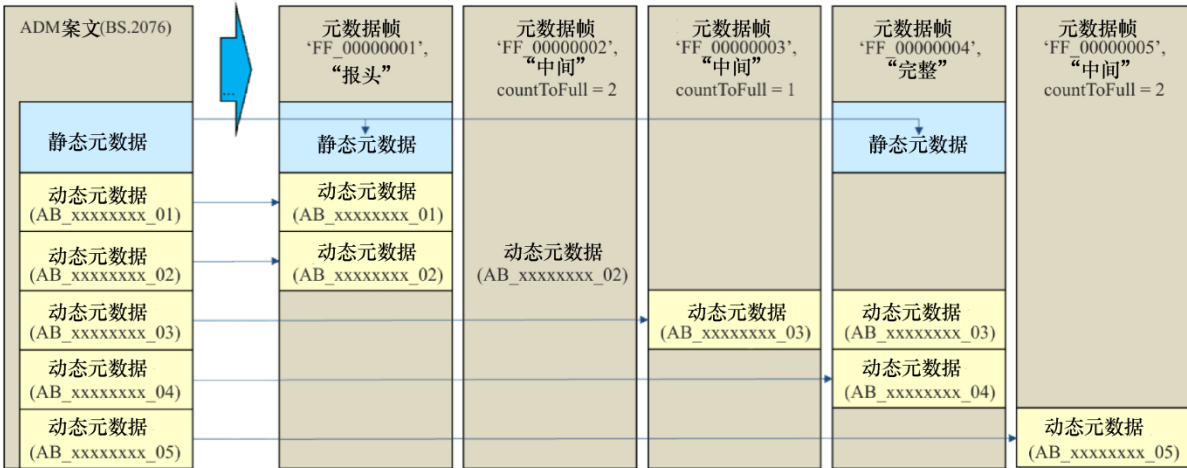


BS.2125-03

A1.2.3 混合帧（MF）流说明

“完整”和“中间”帧都可以用在单个流中（见图4）。在这种情况下，广播公司可以自由决定传送“完整”帧的时间间隔。MF流支持有一定延迟的随机访问：即接收机必须等待下一个“完整”帧。

图4
混合帧（MF）流中S-ADM的结构



BS.2125-04

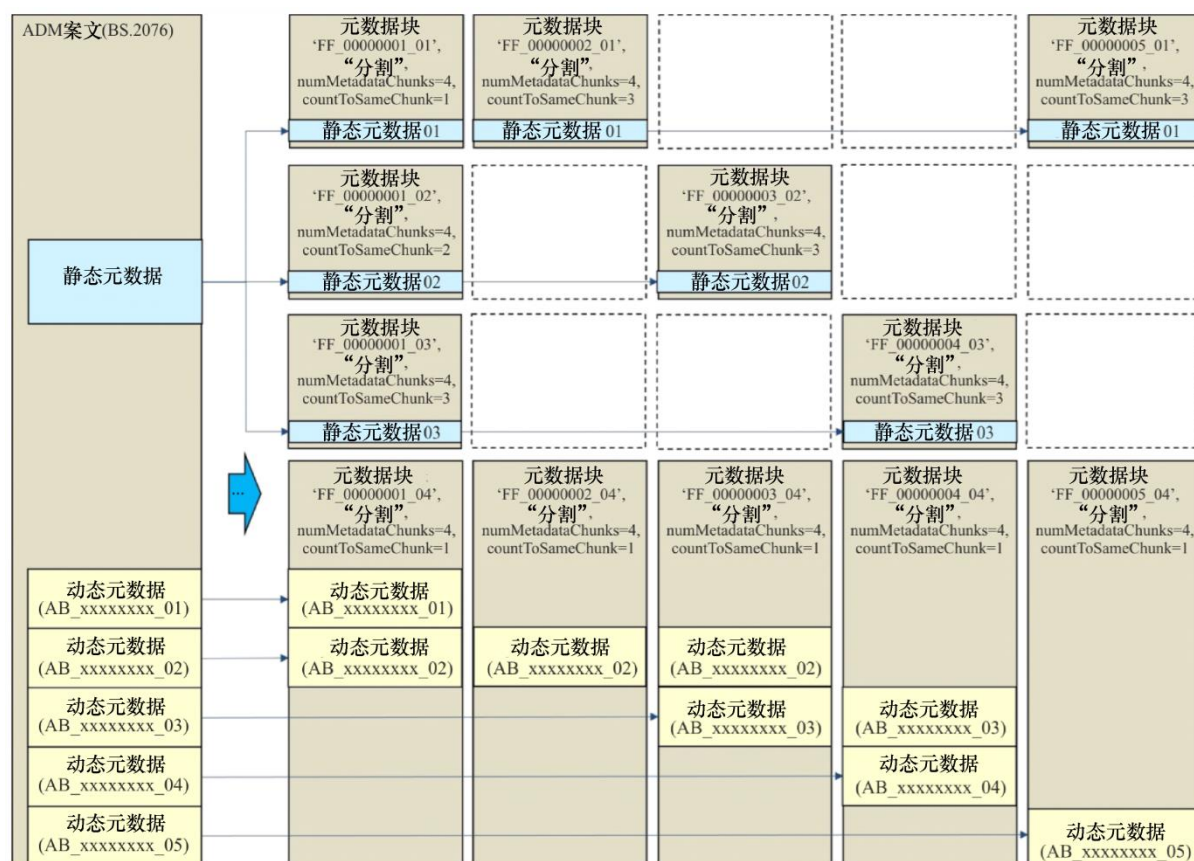
A1.2.4 分割帧（DF）流说明

根据是“完整”帧还是“中间”帧，MF流的流量可有很大变化。随着静态元数据被分成数据块，DF流被设计得可在所有的帧间更均匀地分发数据（见图5）。

图5中的示例：帧元数据，如“FF_00000001”，被分割成元数据块，如“FF_00000001_01”“FF_00000001_02”和“FF_00000001_03”。这些元数据块在同一时间实例中进行传送。元数据块“FF_0000000X_04”包括动态元数据，元数据块“FF_0000000X_01”到“FF_0000000X_03”包括经分割的静态元数据。由于元数据块“FF_00000002_01”具有与其他帧相同的静态元数据（如“FF_00000003_01”和“FF_00000004_01”），因此可以省略“FF_00000003_01”和“FF_00000004_01”。

在DF流中，最后一个数据块总是包含动态元数据，而所有其他数据块都包含静态元数据。DF流支持有一定延迟的随机访问：即接收机必须等待，直至接收到重构完整静态元数据集所需的所有元数据块。

图5
分割帧（DF）流中S-ADM的结构



BS.2125-05

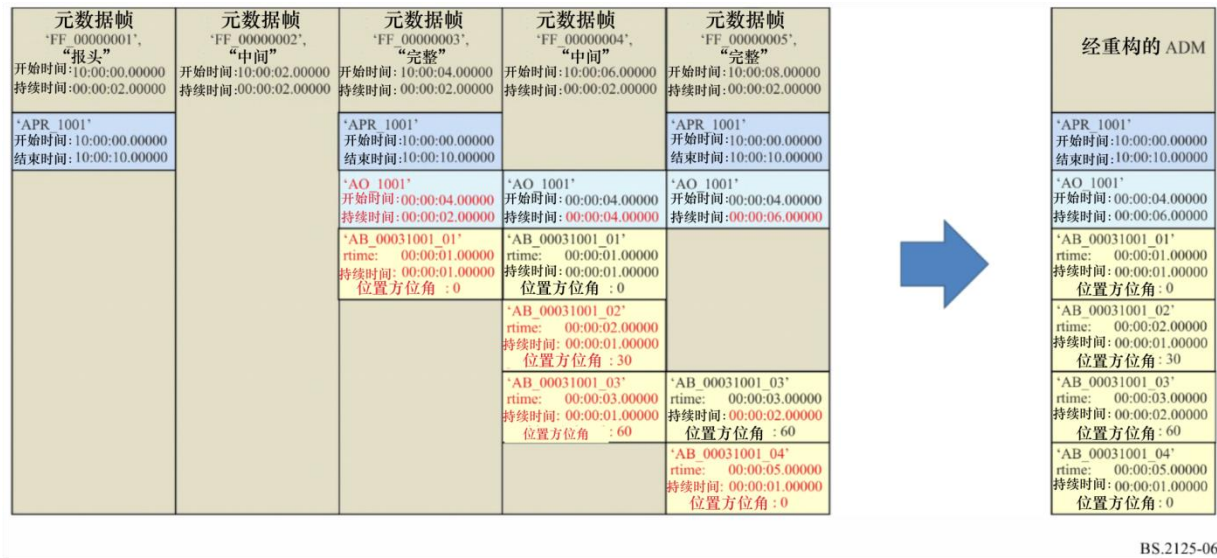
A1.2.5 实时S-ADM生成

图6至图8中举例说明了如何在实时环境中生成S-ADM。示例显示了MF流和FF流，但类似的过程也可以应用于其他类型的流。

图6显示了如何在实时场景中初始化 **audioObject**（“AO_1001”）和一些 **audioBlockFormats**（“AB_00030001_NN”）。当“AO_1001”首次出现在“FF_00000003”中时，“AO_1001”的持续时间从2秒开始（以匹配帧的长度）；然后 **duration** 更新为4秒，然后在接下来的帧中更新为6秒。新的 **audioBlockFormat** 出现在“FF_00000003”、“FF_00000004”和“FF_00000005”中，当块 **audioBlockFormat** 在第一次出现的帧之后的帧中使用，它们的一些 **duration** 值会进行调整。

图右侧经重构的ADM显示了在接收到“FF_00000005”后元素将如何出现，因此“AO_1001”的 **duration** 为6秒。

图6
实时场景中S-ADM的结构，说明如何引入和修改新元素



BS.2125-06

图7显示了如何引入一个新的 **audioObject**，但在最初的两个帧中不包含任何子元素。因此，其 **start** 时间在连续的帧上予以修改，直至它被分配一些子元素。在这种情况下，当它到达“FF_00000003”时，会出现一个新的 **audioBlockFormat**（“AB_00030001_01”），因此4秒的“AO_1001” **start** 时间是固定的，并且它的 **duration** 在随后的帧中会增加。

图7

实时场景中S-ADM的结构，说明如何处理没有子元素的新元素

元数据帧 'FF_00000001', “报头” 10:00:00.00000 00:00:02.00000	元数据帧 'FF_00000002', “中间” 10:00:02.00000 00:00:02.00000	元数据帧 'FF_00000003', “完整” 10:00:04.00000 00:00:02.00000	元数据帧 'FF_00000004', “中间” 10:00:06.00000 00:00:02.00000	元数据帧 'FF_00000005', “完整” 10:00:08.00000 00:00:02.00000	经重构的 ADM
'APR_1001' 开始时间:10:00:00.00000 结束时间:10:00:10.00000		'APR_1001' 开始时间:10:00:00.00000 结束时间:10:00:10.00000		'APR_1001' 开始时间:10:00:00.00000 结束时间:10:00:10.00000	'APR_1001' 开始时间:10:00:00.00000 结束时间:10:00:10.00000
'AO_1001' 开始时间:00:00:02.00000 持续时间:00:00:02.00000	'AO_1001' 开始时间:00:00:04.00000 持续时间:00:00:02.00000	'AO_1001' 开始时间:00:00:04.00000 持续时间:00:00:02.00000	'AO_1001' 开始时间:00:00:04.00000 持续时间:00:00:04.00000	'AO_1001' 开始时间:00:00:04.00000 持续时间:00:00:06.00000	'AO_1001' 开始时间:00:00:04.00000 持续时间:00:00:06.00000
		'AB_00031001_01' rttime: 00:00:01.00000 持续时间:00:00:01.00000 位置方位角 : 0	'AB_00031001_01' rttime: 00:00:01.00000 持续时间:00:00:01.00000 位置方位角 : 0		'AB_00031001_01' rttime: 00:00:01.00000 持续时间:00:00:01.00000 位置方位角 : 0
			'AB_00031001_02' rttime: 00:00:02.00000 持续时间:00:00:01.00000 位置方位角 : 30		'AB_00031001_02' rttime: 00:00:02.00000 持续时间:00:00:01.00000 位置方位角 : 30
			'AB_00031001_03' rttime: 00:00:03.00000 持续时间:00:00:01.00000 位置方位角 : 60	'AB_00031001_03' rttime: 00:00:03.00000 持续时间:00:00:02.00000 位置方位角 : 60	'AB_00031001_03' rttime: 00:00:03.00000 持续时间:00:00:02.00000 位置方位角 : 60
				'AB_00031001_04' rttime: 00:00:05.00000 持续时间:00:00:01.00000 位置方位角 : 0	'AB_00031001_04' rttime: 00:00:05.00000 持续时间:00:00:01.00000 位置方位角 : 0

BS.2125-07

图8显示了在“APR_1001”初始end时间结束后出现新的帧（“FF_00000006”）时，是如何修改 audioProgramme（“APR_1001”）的 end 时间的。“AO_1001”和“AB_00031001_04”的duration也在该新的帧中进行修改。结果是，经重构的ADM的“APR_1001” end时间也得到了更新。

图8

实时场景中S-ADM的结构，说明如何修改现有元素

元数据帧 'FF_00000001', “报头”	元数据帧 'FF_00000002', “完整”	元数据帧 'FF_00000003', “完整”	元数据帧 'FF_00000004', “完整”	元数据帧 'FF_00000005', “完整”	元数据帧 'FF_00000006', “完整”	经重构的 ADM
开始时间:10:00:00.00000 持续时间:00:00:02.00000	开始时间:10:00:02.00000 持续时间:00:00:02.00000	开始时间:10:00:04.00000 持续时间:00:00:02.00000	开始时间:10:00:06.00000 持续时间:00:00:02.00000	开始时间:10:00:08.00000 持续时间:00:00:02.00000	开始时间:10:00:10.00000 持续时间:00:00:02.00000	
'APR_1001' 开始时间:10:00:00.00000 结束时间:10:00:10.00000	'APR_1001' 开始时间:10:00:00.00000 结束时间:10:00:10.00000	'APR_1001' 开始时间:10:00:00.00000 结束时间:10:00:10.00000	'APR_1001' 开始时间:10:00:00.00000 结束时间:10:00:10.00000	'APR_1001' 开始时间:10:00:00.00000 结束时间:10:00:10.00000	'APR_1001' 开始时间:10:00:00.00000 结束时间:10:00:12.00000	
		'AO_1001' 开始时间:00:00:04.00000 持续时间:00:00:02.00000	'AO_1001' 开始时间:00:00:04.00000 持续时间:00:00:04.00000	'AO_1001' 开始时间:00:00:04.00000 持续时间:00:00:06.00000	'AO_1001' 开始时间:00:00:04.00000 持续时间:00:00:08.00000	
		'AB_00031001_01' rtime: 00:00:01.00000 持续时间:00:00:01.00000 位置方位角 : 0	'AB_00031001_01' rtime: 00:00:01.00000 持续时间:00:00:01.00000 位置方位角 : 0		'AB_00031001_01' rtime: 00:00:01.00000 持续时间:00:00:01.00000 位置方位角 : 0	
			'AB_00031001_02' rtime: 00:00:02.00000 持续时间:00:00:01.00000 位置方位角 : 30		'AB_00031001_02' rtime: 00:00:02.00000 持续时间:00:00:01.00000 位置方位角 : 30	
			'AB_00031001_03' rtime: 00:00:03.00000 持续时间:00:00:01.00000 位置方位角 : 60	'AB_00031001_03' rtime: 00:00:03.00000 持续时间:00:00:02.00000 位置方位角 : 60	'AB_00031001_03' rtime: 00:00:03.00000 持续时间:00:00:02.00000 位置方位角 : 60	
				'AB_00031001_04' rtime: 00:00:05.00000 持续时间:00:00:01.00000 位置方位角 : 0	'AB_00031001_04' rtime: 00:00:05.00000 持续时间:00:00:03.00000 位置方位角 : 0	

BS.2125-08

当读取一个S-ADM帧（其中某个特定元数据元素的属性相对前面帧的属性已发生变化）时，须使用最近帧的元数据元素。

A1.3 S-ADM元数据帧的结构

S-ADM元数据帧应由两部分组成。第一部分是frameHeader，它包含用于描述ADM元数据帧规范的、S-ADM的附加元素，第二部分是audioFormatExtended，它包含ITU-R BS.2076建议书中规定的ADM元数据。

A1.3.1 “完整”帧的结构

“完整”帧应包含audioFormatExtended部分中的所有元素。

A1.3.2 “中间”帧的结构

当与前一ADM元数据帧相比时，“中间”帧应仅包括值已更改的元素。**audioProgramme**、**audioObject**和**audioBlockFormat**的ADM元素用于规定时间信息。用于“DirectSpeakers”的**typeDefinition**的**audioBlockFormat**通常具有时不变的元数据，而用于“对象”的**typeDefinition**的**audioBlockFormat**通常具有时变的元数据。“中间”帧通常由**audioChannelFormat**中的**audioBlockFormat**组成，用于“对象”的**typeDefinition**。

A1.3.3 “分割”帧的结构

“分割”帧包含分割成至少两个数据块的元数据。每个帧应至少承载一个数据块。每个数据块应包含一个将在一个完整的帧中承载的、所有元素的子集。由于静态元数据元素不会在连续帧上发生变化，因此不必将之放在所有帧中。可以在每个帧中改变的动态元数据元素应承载于帧所承载的最后一个数据块中。

A1.3.4 “报头”帧的结构

“报头”帧是一个“完整”帧，具有特殊功能，用于指示一个新的**audioProgramme**的开始或一个新的流的开始。

A1.3.5 “全部”帧的结构

“全部”帧应包含整个**audioProgramme**的所有元数据。因此，这可以包括用于描述过去和未来帧中的音频的元数据以及当前帧中的元数据。

只有在流式传送S-ADM帧之前已知整个**audioProgramme**的元数据时，方可使用“全部”类型的帧。因此，只应考虑预先录制的节目或者带有完全静态元数据的直播节目。

A1.3.6 每个帧的通用属性

A1.3.6.1 PCM音频

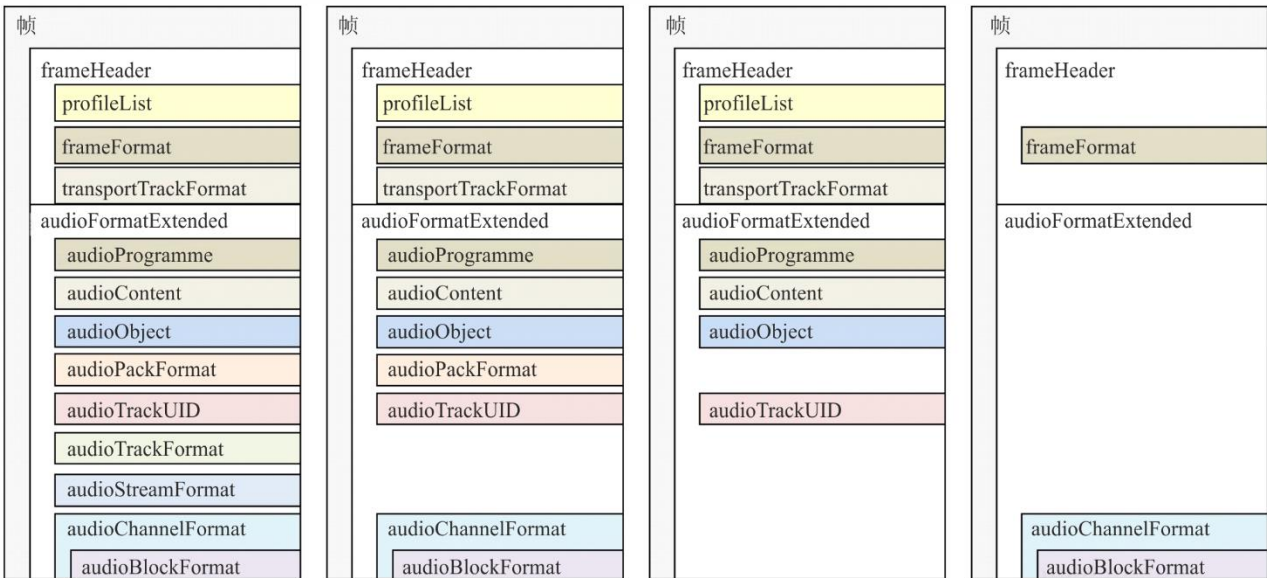
对于PCM音频，**audioTrackFormat**应引用单个**audioStreamFormat**，而**audioStreamFormat**应引用单个**audioChannelFormat**。如果**audioTrackUID**直接引用**audioChannelFormat**，则可以省略**audioTrackFormat**和**audioStreamFormat**。

A1.3.6.2 通用定义

ITU-R BS.2094建议书将ADM元素的一些典型用例定义为通用定义，尤其是对基于声道的音频。当使用通用定义时，ADM的格式部分（包括**audioTrackFormat**、**audioStreamFormat**、**audioChannelFormat**和**audioPackFormat**）应被省略。

图9

S-ADM帧中ADM元数据的结构



BS.2125-09

A1.4 S-ADM元素和属性

A1.4.1 帧

S-ADM父元素是**frame**。它有两个子元素，**frameHeader**和**audioFormatExtended**或**coreMetadata**。**frameHeader**应该放在每个帧的第一个位置。

表1所示的XML说明了在**frame**元素中有两种构造元素的方法，其中方法1用于只需要携带ADM元数据的情况；方法2用于亦需携带广播元数据的情况。

表1

用于**frame**子元素的两种方法

方法1	方法2
<pre><frame> <frameHeader> ... </frameHeader> <audioFormatExtended> ... ADM metadata here ... </audioFormatExtended> </frame></pre>	<pre><frame> <frameHeader> ... </frameHeader> <coreMetadata> ... broadcast metadata here ... <format> ... broadcast metadata here ... <audioFormatExtended> ... ADM metadata here ... </audioFormatExtended> </format> </coreMetadata> </frame></pre>

S-ADM XML示例见第A2.1节。

A1.4.1.1 属性

表2

Frame的属性

属性	说明	示例	是否必要
版本 ⁽¹⁾	S-ADM建议书的名称和修订号	“ITU-R_BS.2125-1”	是

⁽¹⁾ 由于本建议书的早期版本（ITU-R BS.2125-0建议书）不包含版此属性，那么如果该属性不存在，应解释为符合ITU-R BS.2125-0建议书。

A1.4.1.2 子元素

表3

frame子元素

子元素	说明	数量 ⁽²⁾
frameHeader	见第A1.4.5段	1
audioFormatExtended ⁽¹⁾	包含ADM元数据，如ITU-R BS.2076建议书所述，见第A1.4.4段	0或1
coreMetadata ⁽¹⁾	如果需要携带广播元数据，则使用coreMetadata而不是audioFormatExtended。参见第A1.6段。核心元数据子元素参见第A1.4.2段。	0或1
-	其他建议书可能为元数据的传送定义了附加的子元素。如果某实现操作不知道这些附加的子元素，则应该忽略这些子元素。	0 ... *

⁽¹⁾ 此处须使用audioFormatExtended或coreMetadata（而不是两者）作为frame子元素。如果使用了coreMetadata子元素，该子元素将把audioFormatExtended元素带到其结构的更下方。

⁽²⁾ “n...m”表示介于n和m项之间的数量，而“n...*”表示n个或更多项，其中n是零或正整数，m是正整数。

A1.4.2 coreMetadata

代替audioFormatExtended子元素使用的coreMetadata子元素，携带format子元素和Broadcast Metadata（见第A1.6段）。则format子元素须携带audioFormatExtended子元素。

A1.4.2.1 子元素

表4

coreMetadata子元素

子元素	说明	数量
format	携带与格式相关的元数据。 format 子元素见表5	1
-	ITU-R BS.2088建议书描述了这里可以使用的其他子元素。更多详情见第A1.6段	0 ... *
-	其他建议书可能为元数据的携带定义了附加子元素。如果某实现操作不知道这些附加的子元素，则应该忽略这些子元素。	0 ... *

A1.4.3 format

format子元素须携带**audioFormatExtended**子元素。

A1.4.3.1 子元素

表5

format子元素

子元素	说明	数量
audioFormatExtended	包含ADM元数据，如ITU-R BS.2076建议书所述	1
-	ITU-R BS.2088建议书描述了这里可以使用的其他子元素。更多详情见第A1.6段	0 ... *
-	其他建议书可能为元数据的携带定义了附加子元素。如果某实现操作不知道这些附加的子元素，则应该忽略这些子元素。	0 ... *

A1.4.4 audioFormatExtended

audioFormatExtended子元素用于代替**frame**元素中的**coreMetadata**子元素，或者用于**format**子元素中。**audioFormatExtended**子元素须承载ADM元数据，如ITU-R BS.2076建议书所规定的那样。

audioBlockFormat是一个现有的ADM元素，本建议书描述了在现有ADM **audioBlockFormat**属性（见第A1.4.9段）之上的、S-ADM的附加属性。

A1.4.5 frameHeader

frameHeader包括**frameFormat**（见第A1.4.7段）和**transportTrackFormat**（见第A1.4.8段），添加这两个格式是为了规定S-ADM元数据帧的结构以及描述S-ADM的传送音频接口。此外，**frameHeader**元素可以包括**profileList**子元素（见第A1.4.6段）。

A1.4.5.1 子元素

表6
frameHeader子元素

子元素	说明	数量
profileList	配置文件合规性的描述	0或1
frameFormat	ADM元数据帧的格式描述	1
transportTrackFormat	传送音频接口格式的描述	1...*

A1.4.6 profileList

profileList须指出流遵守哪些规范的约束。

A1.4.6.1 子元素

表7
profileList子元素

子元素	说明	示例	数量
profile ⁽¹⁾	各配置文件规范须规定如何为各自的配置文件设置配置文件字符串。此文件应指出S-ADM元数据遵从的相应配置文件的参引（如文档引用）。	“ITU-R BS.XXXX附件1”	1 ... *

⁽¹⁾ 如果存在多个profile元素，那么流将受到每个profile中约束最多部分的约束。

表8
profile属性

属性	说明	示例	是否必要
profileName	profileName属性字符串应须针对相应的配置文件加以设置。它应指出S-ADM元数据所遵从的各个配置文件的名称。	“生产Profile”	是
profileVersion	profileVersion属性字符串应须针对相应的配置文件加以设置。它应指出S-ADM元数据所遵从的相应配置文件的版本。	“1.0.0”	是
profileLevel	profileLevel属性字符串应针对相应的配置文件加以设置。它应指出S-ADM元数据所遵从的相应配置文件的级别。	“1”	是

A1.4.7 frameFormat

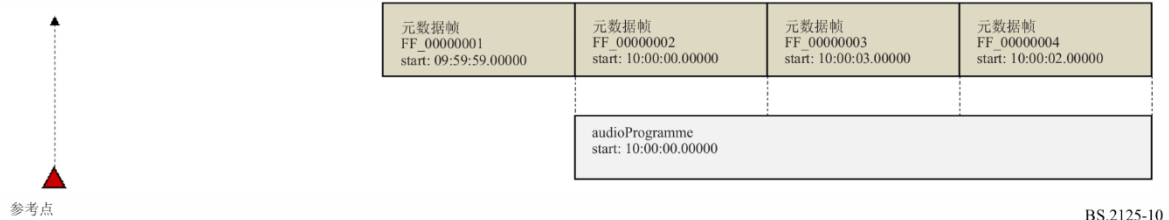
frameFormat须描述包含ADM元素的帧的规格和相关音频帧的规格。

S-ADM帧与相关音频实质内容的同步和/或偏移由传送/接口协议决定。

当一个 **audioProgramme** 包含在某 **frame** 的 **audioFormatExtended** 中时，**audioProgramme**的**start**和**frameFormat**的开始须相对于同一参考时间点。参见图10。

图10

将**audioProgramme**与**frame**对齐到相同的参考时间点



BS.2125-10

参数**countToSameChunk**须指明当前帧和同一数据块重复出现的帧之间的帧数。

参数**numMetadataChunks**须指明随机访问所需的元数据块的数量。图11中的示例其**numMetadataChunks**值为4。注意：一个流中每个帧的元数据块数量应该是相同的。

图11

countToSameChunk用例



BS.2125-11

A1.4.7.1 属性

表9

frameFormat属性

属性	描述	示例	是否必要
frameFormatID (1)	帧的ID。见第A1.5段。	– FF_00000001 – FF_00000001_01	是
start ⁽²⁾	帧的开始时间。可用的格式有： – “hh:mm:ss.zzzzz”表示时、分、秒 – “zzzzsffff”，其中“zzzzz”表示经过的音频样本数，“ffff”表示采样速率。 – “hh:mm:ss . zzzzzsffff”表示经过时间的小时、分钟和秒，其中“zzzzz”并非是时间而是音频样本，“ffff”表示采样率。	– 00:00:00.00000 – 0S48000 – 09:59:59.47999S48000	是
duration ⁽²⁾	帧的持续时间。可用的格式有： – “hh:mm:ss.zzzzz” – “hh:mm:ss.zzzzzSffff” – “zzzzzSffff” – “ss.zzzzz” 注：“z”和“f”的数可以是5到9位数字，这取决于所需的精度	– 00:00:00.25000 – 00:00:00.12000S48000 – 12000S48000 – 00.25000	是
type	帧类型的描述符。 见第A1.3.1至A1.3.5段	– 报头 – 完整 – 分割 – 中间 – 全部	是
timeReference	audioBlockFormat 定时参数的时间模式描述符。 “总计”表示使用从 audioProgramme 开始时间开始的、已经经过的时间。 “本地”表示使用从帧开始处开始的、已经经过的时间。 对于整个流，该参数应是固定的。 有关 audioBlockFormat 的更详细信息，请参见第A1.4.9段	– 总计 – 本地	可选（默认为总计）
flowID	由RFC 4122或ISO/IEC 11578:1196中规定的UUID描述的S-ADM帧序列的唯一ID	12345678-abcd-4000-a000-112233445566	可选

^① 早期版本（ITU-R BS.2125-0建议书）规定了包含11位十六进制索引/计数器的frameFormatID的数字格式，但这是一个错误并非初衷。任何读取S-ADM文件的软件都应支持frameFormatID的8位和11位变体。

^② 早期版本（ITU-R BS.2125-0建议书）允许以包含日期的格式指定开始时间“yyyy-mm-ddThh:mm:ss.zzzzzZ”。任何读取S-ADM文件的软件都应该允许这种扩展时间格式的存在。

表10

“报头”“完整”“中间”和“全部”帧类型的frameFormat属性

属性	描述	示例	要求
countToFull	在FF流中：设置为“1”。 在MF流中：下一个“完整”帧的帧数。 在IF流中：设置为“0”。	– 0 – 1 – 2 – 3 ...	可选的 （缺省：在FF中为1，在IF中为0）

表11

“分割”帧类型的frameFormat属性

属性	描述	示例	要求
numMetadataChunks	随机访问所需的元数据块数量。流中每个帧的元数据块的数量应该相同	– 2 – 3 ...	是
countToSameChunk	给定静态元数据块下次重复出现的帧数。如果设置为“1”，则每个帧都包含静态元数据块	– 1 – 2 – 3 ...	可选的 （缺省：未知）

关于时间格式和小数位数的说明

本文件中基于时间的时间格式显示秒的五位小数（“ss.zzz”或“hh:mm:ss.zzzzz”），但这是最小小数位数。可以使用更多的小数位，并建议在使用48 kHz以上的采样率时使用更多的小数位。九位小数（即hh:mm:ss.zzzzzzzzz）给出了纳秒级的精度。

对于较长的、基于样本的时间格式（hh:mm:ss.zzzzzSffff），“z”数字表示多个样本，“z”的数量必须与“f”的数量相匹配（即“hh:mm:ss.zzzzzS48000”“hh:mm:ss.zzzzzS192000”）。“zzzzz”的值应小于“ffff”的值。

对于较短的、基于样本的格式“zzzzzSffff”，数字位数可以是可变的（即“0S48000”或“500000S48000”）。如果表示的时间大于一秒，则“zzzzz”的值可以大于“ffff”的值。

可选的属性**countToFull**和**countToSameChunk**对于接收机了解何时能够开始随机访问回放是有用的。然而，不要求这些属性支持随机访问功能：即使不使用这些属性，接收机也可以在接收到随机访问所需的所有元数据时从所接收的数据中识别。

A1.4.7.2 子元素

frameFormat的**changedIDs**元素可以显示其值与之前的帧相比不同的ADM元素。

表12

frameFormat子元素

子元素	描述	示例	数量
changedIDs	与前一帧相比已做更改的ADM元素ID的引用列表。包含表13中所述的子元素	XML示例见第A2.2节	0...1
chunkAdmElement	如果 frameFormatID 有多个元数据块，则引用在每个元数据块中分类的ADM元素	audioChannelFormat	0...*

changedIDs子元素只能清楚地显示其值变自前面帧的ADM元素。**changedIDs**的子元素如表13所示。

表13

changedIDs子元素

子元素	属性	描述	示例	数量
audioChannelFormatIDRef	status	对 audioChannelFormatID 的引用与前一帧不同。属性“ status ”表示添加或修改了已做更改的描述符	AC_00031001	0...*
audioPackFormatIDRef	status	对 audioPackFormat ID的引用与前一帧不同	AP_00031001	0...*
audioTrackUIDRef	status	对 audioTrackUID ID的引用与前一帧不同	ATU_00000001	0...*
audioTrackFormatIDRef	status	对 audioTrackFormat ID的引用与前一帧不同	AT_00031001_01	0...*
audioStreamFormatIDRef	status	对 audioStreamFormat ID的引用与前一帧不同	AS_00031001	0...*
audioObjectIDRef	status	对 audioObject ID的引用与前一帧不同	AO_1001	0...*
audioContentIDRef	status	对 audioContent ID的引用与前一帧不同	ACO_1001	0...*
audioProgrammeIDRef	status	对 audioProgramme ID的引用与前一帧不同	APR_1001	0...*

表14

changedID属性

属性	描述	示例	要求
status	状态属性用于指示生成一个新的元素、一个经过更改的元素、一个元素的扩展或一个元素的过期。	– “新的” – “经更改的” – “经扩展的” – “过期的”	是

status属性应为四个值中的一个：

- “新的” – 当新的元素首次出现时，应使用该值；
- “经更改的” – 当自上一帧以来元素内的任何参数或值发生变化时，应使用该值；
- “经扩展的” – 当定时参数自前一帧以来发生变化但所有其他参数均保持不变时，应使用该值；
- “过期的” – 当元素不再存在于当前帧中但出现于前一帧中时，应使用该值。

第A2.2节显示了一些XML代码示例，以说明**changedIDs**的用法。

A1.4.8 transportTrackFormat

transportTrackFormat表示物理音轨（如AES3接口的声道1）和ADM中音轨的**UIDs**（如“ATU_00000001”）之间的关系。对于ADM，该信息在BW64文件的“chna”块中予以描述。**transportTrackFormat**是相当于BW64“chna”数据块的S-ADM。

A1.4.8.1 属性

transportName是用于传送相关音频实质内容的接口的名称。本建议书中未定义接口的具体名称。用户可以自由使用接口的任何名称。当使用多个接口时，这些接口可以标记为设备-A、设备-B和设备-C。**numTracks**是每个接口中相关联的音轨的数量。**numIDs**是每个接口中相关联的**audioTrackUIDs**的数量。

表15

transportTrackFormat属性

属性	描述	示例	要求
transportID	传送音频信号的接口索引（见第A1.5段）	TP_0001	是
transportName	传送音频信号的接口描述符	AES3-A、AES3-B等	可选
numTracks	每个接口中相关传送音轨的数量	16	可选
numIDs	每个接口中相关 audioTrackUID 的数量	32	可选

A1.4.8.2 子元素

audioTrack的**trackID**是每个接口中传送音频轨道的索引。该索引相当于BW64文件中的音轨号。**formatLabel**和**formatDefinition**指明音频信号的格式类型。这些值在ITU-R BS.2076建议书中予以定义。

表16
transportTrackFormat子元素

子元素	属性	描述	示例	要求
audioTrack	trackID	每个接口中传送音轨的索引。 例如，对于AES3接口（ITU-R BS.647建议书），它被设置为1或2；对于MADI接口（ITU-R BS.1873建议书），它被设置为1到64	1	是
	formatLabel	音频样本格式的描述符。当省略 formatLabel 和 formatDefinition 时，指定 formatLabel 为“0001”	0001	可选
	formatDefinition	音频样本格式的描述。当省略 formatLabel 和 formatDefinition 时，指定 formatLabel 为“PCM”	PCM	可选

audioTrackFormatIDRef和**audioPackFormatIDRef**都不包含在**transportTrackFormat**中，因此它们应该通过**audioTrackUID**来引用。**audioTrackFormat**和**audioStreamFormat**对PCM音频而言均可省略，**audioTrackUID**可直接引用**audioChannelFormat**，而不是**audioTrackFormat**。然后，对**audioTrackFormat**和**audioChannelFormat**的ID使用相同的数字。

表17
audioTrack子元素

子元素	描述	示例	数量
audioTrackUIDRef	在ADM代码中引用 audioTrackUID	ATU_00000001	0...*

用于说明**transportTrackFormat**用法的示例代码见第A2.4节。

A1.4.9 **audioBlockFormat**

audioBlockFormat是一个现有的ADM元素，本节描述了在现有ADM **audioBlockFormat**属性之上的、S-ADM的附加属性。

如果**timeReference**（**frameFormat**中的属性）设置为“本地”，则使用**lstart**和**lduration**元素，而不是**audioBlockFormat**中的**rtime**和**duration**元素。**lstart**元素和**lduration**元素表示相对于S-ADM帧开始时间的、音频块的开始时间和持续时间。

audioBlockFormat中与当前帧重叠的时变参数（如**position**）可在S-ADM帧之外的时间进行定义。**lstart**和**lduration**允许在不重新计算的情况下包含此信息。为此，**lstart**可以是负的（即在帧开始之前），和/或**lstart+lduration**可以超出帧的结尾。如果需要在帧边界上放置时变参数，则可能需要重新计算参数。

audioBlockFormat中的时变参数定义块结束处的值。块起始处的值由前一个块来定义。如果前一个块不存在（因为它在前一帧中，所以可能没有被接收到），则需要定义帧中第一个块开始处的值。这是通过在第一个块之前插入初始化器**audioBlockFormat**来完成的，其ID设置为“AB_xxxxxyyy_00000000”，**initializeBlock**属性设置为“1”。该初始化器**audioBlockFormat**没有持续时间，因此不得包含**lduration**属性。

当从非串行**audioBlockFormat**转换时，总计时间和本地时间之间的比较如图12和图13所示。这两种情况都表明，通过规定帧之外的时间点，可以避免重新计算**position**的值。这允许渲染器（或元数据的任何其他处理器）决定如何重新计算位置。

图12

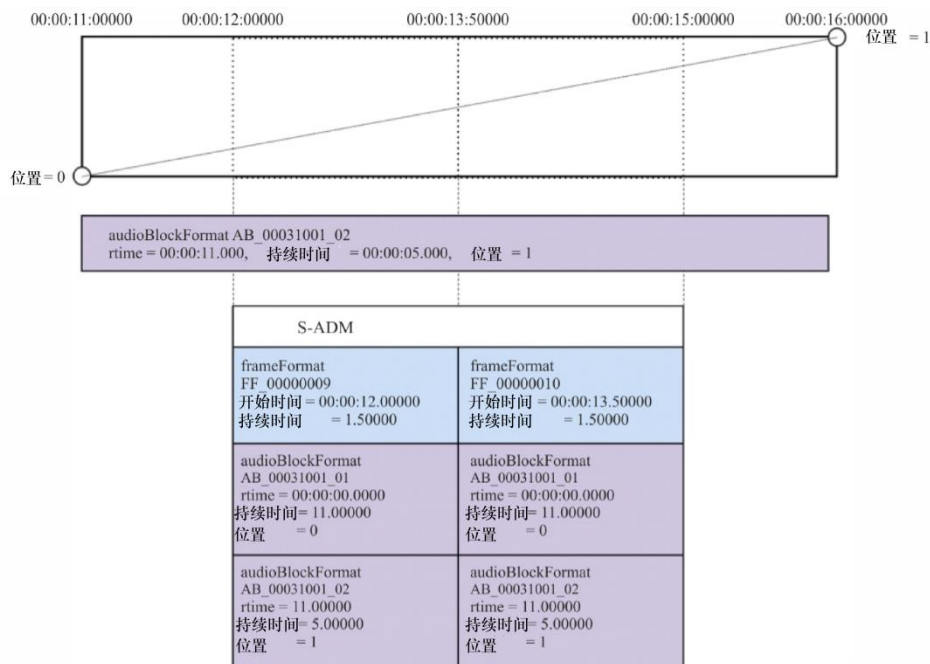
使用**rtime**和**duration**的**audioBlockFormat**用例

图13

使用Istart和Iduration的audioBlockFormat用例

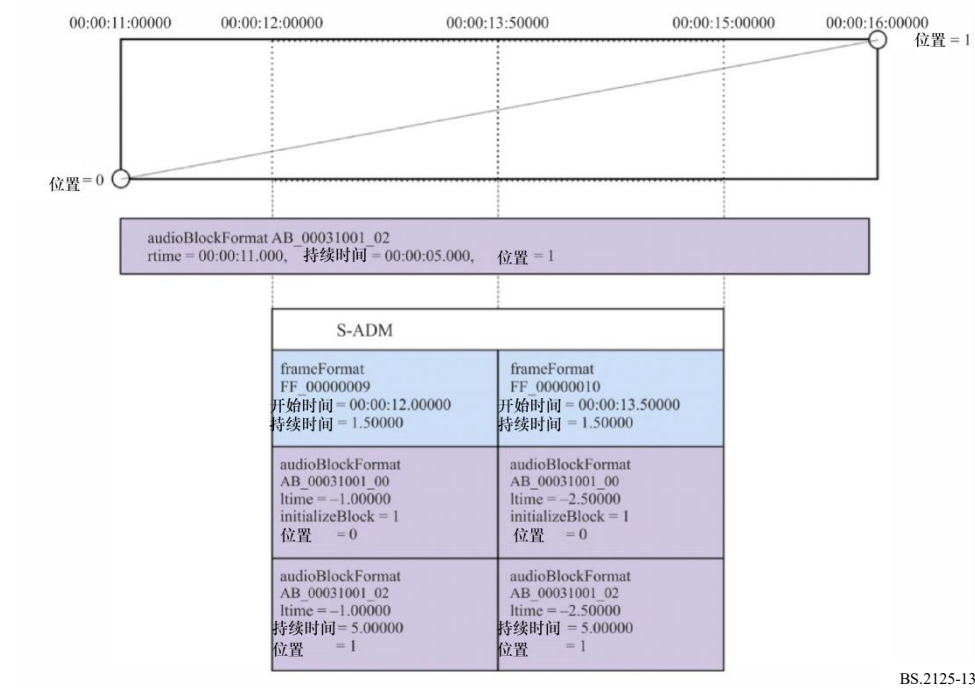


图14和图15显示了当从头开始生成S-ADM帧时如何使用总计时间和本地时间。在这种情况下，中间位置值是已知的，并且已经出现在帧边界上，因此Istart和Iduration值出现在帧内。

图14

当从头开始生成时，使用Istart和duration的audioBlockFormat用例

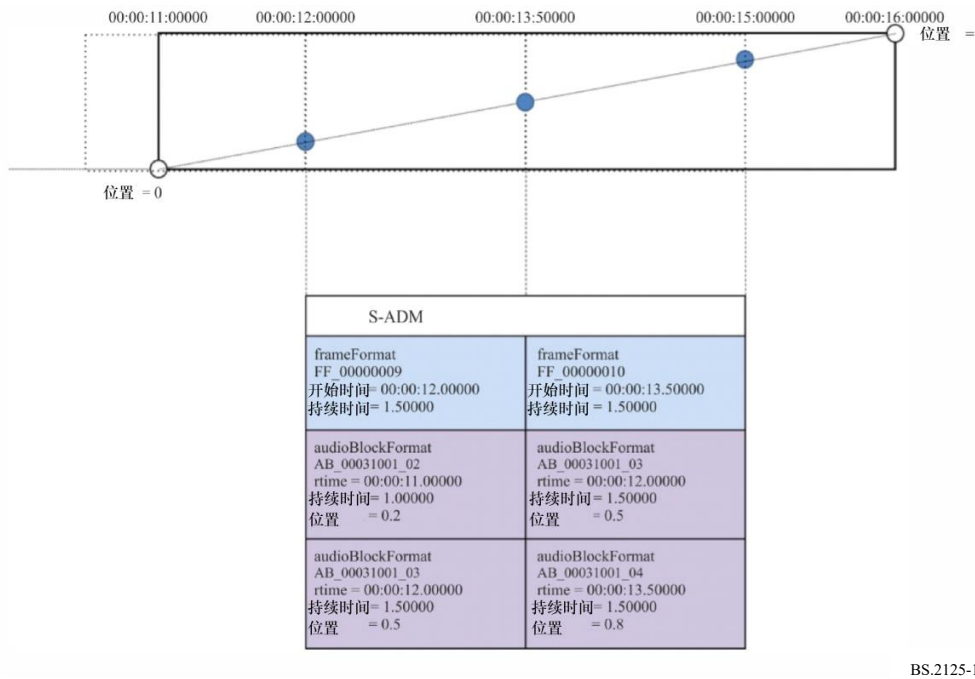
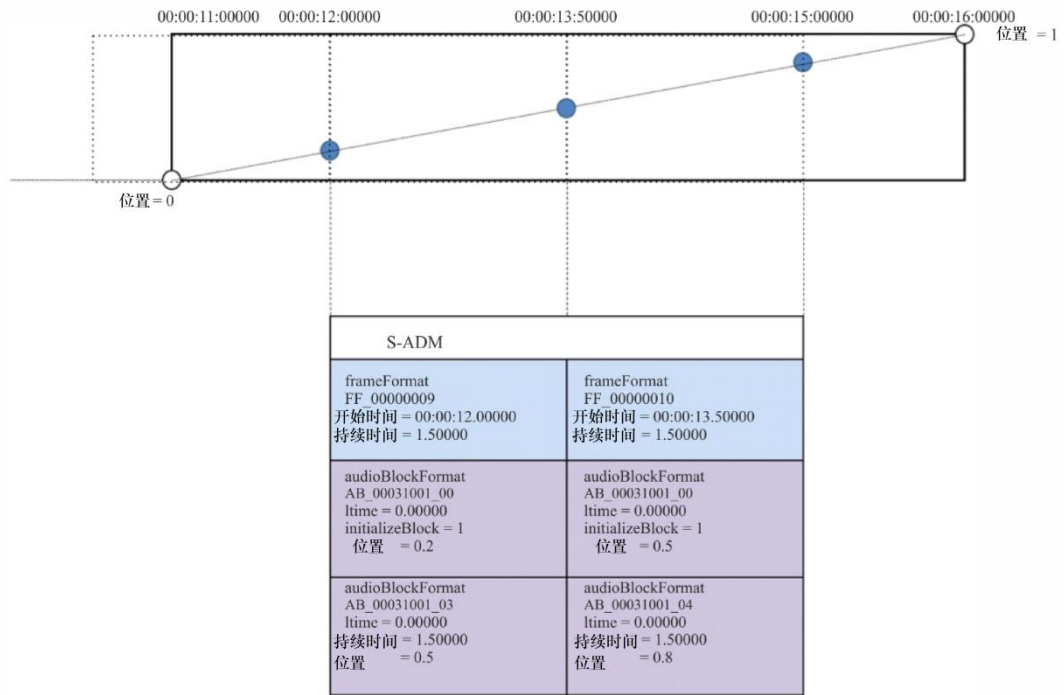


图15

当从头开始生成时，使用lstart和lduration的audioBlockFormat用例



BS.2125-15

A1.4.9.1 附加属性

表18

audioBlockFormat属性

属性	描述	示例	要求
lstart⁽¹⁾	块的开始时间相对于S-ADM元数据帧的开始时间。块的开始时间由以下格式之一来表示： – “hh:mm:ss.zzzzz”，表示小时、分钟、秒和小数秒。 – “ss.zzzzz”，如果不需要小时和分钟的话。 – 如果分数部分附有“Sffff”，则分数部分和“ffff”分别显示音频样本数和采样速率。 注：“z”和“f”的数可以是5到9位数字，这取决于所需的精度	– 00:00:00.00000 – 00.00000 – 00:00:00.00000S48000 – 0S48000	可选的

表18（结束）

属性	描述	示例	要求
lduration	S-ADM元数据帧中块的持续时间。块的持续时间由以下格式之一表示： – “hh:mm:ss.zzzzz” ，表示小时、分钟、秒和小数秒。 – “ss.zzzzz” ， 如果不需要小时和分钟的话。 – 如果分数部分附有 “Sffffff” ， 则分数部分和 “fffff” 分别显示音频样本数和采样速率。 注：“z” 和 “f” 的数可以是5到9位数字，这取决于所需的精度	– 00:00:00.50000 – 00.50000 – 00:00:00.24000S48000 – 24000S48000	可选的
initializeBlock	如果 initializeBlock 设置为“1”，则表示 “ AB_xxxxxyyy_00000000 ” 的 audioBlockFormat 用于规定帧中第一个音频块的所有元素的初始值	1（开）、0（关）	可选的

⁽¹⁾ 早期版本（ITU-R BS.2125-0建议书）包含的编辑文本和示例XML代码错误地将**lstart**属性标识为**ltime**。任何读取S-ADM文件的软件都应容忍**ltime**属性的存在，并将其解释为**lstart**属性。

A1.5 ID的使用

各元素中的ID属性有三个主要目的：允许元素相互参考；为每个已定义的元素提供一个唯一的标识；为元素内容提供一种逻辑数值表示。每个元素的ID遵循以下格式。

表19
元素ID格式

元素	ID格式
frameFormat	FF_XXXXXXXX FF_XXXXXXXX_zz
transportTrackFormat	TP_wwww

A1.5.1 frameFormat

在**frameFormatID**中，XXXXXXXX部分是一个8位十六进制数，用作自流启动以来已出现的累计帧数的索引/计数器。对于第一帧，该索引应从1开始且对于随后的每个连续帧，应增加1。

zz部分是一个2位数的十六进制数，仅用于分割帧模式，表示当前元数据块的索引。该索引应从1开始，且对于随后的每个连续的元数据块应递增1。

A1.5.2 transportTrackFormat

在transportID中，www部分是一个4位的十六进制数，用作传送音频信号的接口索引。

A1.6 与XML广播元数据的兼容性

A1.6.1 广播元数据的起源

BWF文件格式（ITU-R BS.1352建议书）包含用于承载广播元数据的<bext>和<ubxt>数据块。该广播元数据可以以BW64文件格式（ITU-R BS.2088建议书）承载于ADM元数据旁的<axml>数据块中。当广播元数据包含在<axml>数据块的XML中时，广播元数据参数位于表20所示的元素下。

表20

用于包括广播元数据的元素结构

层级	元素	广播元数据参数
1（顶层）	coreMetadata	bextOriginator bextOriginatorReference bextDescription bextOriginationDate bextOriginationTime bextUMID
2	format	bextCodingHistory
3	audioFormatExtended (contains ADM metadata)	bextTimeReference（在audioProgramme属性内）

A1.6.2 S-ADM中的广播元数据

如果要将广播元数据包括在ADM元数据中，则应该应用表20中的元素结构，其中**coreMetadata**是顶层元素（在**frame**元素下面）。第A.2.1节中的示例XML说明了包括广播元数据的此结构。

如果在S-ADM流中使用广播元数据，则它应仅用于“全部”或“报头”帧中，而“完整”“分割”和“中间”帧保留表20中的三个元素，但没有广播元数据参数。这确保了流中的每个帧都具有相同的元素结构，但只有第一帧包含广播元数据（第A2.6节中的第二个示例XML说明了没有广播元数据的此结构）。

当在S-ADM流中没有使用广播元数据时，ADM元数据的顶层元素（在**frame**元素之下）可以是**audioFormatExtended**，或者可以使用表20中的三个元素（因此**coreMetadata**包含**format**，然后是**audioFormatExtended**）。

附件2

S-ADM的XML代码示例

A2.1 frame、frameHeader和audioFormatExtended代码示例

以下S-ADM XML代码说明了顶层 **frame** 元素的结构，以及其中的 **frameHeader** 和 **audioFormatExtended** 元素。显示了两个帧。

目录

	页 码
附件2 S-ADM的XML代码示例	26
A2.1 frame、frameHeader和audioFormatExtended代码示例.....	26
A2.2 changedIDs用法示例	28
A2.3 源自ADM XML的S-ADM XML示例	29
A2.4 transportTrackFormat用法示例	42
A2.5 timeReference标志用法示例	44
A2.6 广播元数据用法示例	46

```
<frame>
  <frameHeader>
    <frameFormat frameFormatID = "FF_00000001" start = "00:00:00.00000" duration = "00:00:00.50000"
    flowID = "12345678-abcd-4000-a000-112233445566" type = "header"/>
    <transportTrackFormat/>
  </frameHeader>

  <!-- the associated XML code of the ADM is written below -->
  <audioFormatExtended>
    <audioProgramme/>
    ...
    <audioChannelFormat/>
  </audioFormatExtended>
</frame>

<frame>
  <frameHeader>
    <frameFormat frameFormatID="FF_00000002" start="00:00:00.50000" duration="00:00:00.50000"
    flowID="12345678-abcd-4000-a000-112233445566" type="full"/>
    <transportTrackFormat/>
  </frameHeader>

  <!-- the associated XML code of the ADM is written below -->
```

```

<audioFormatExtended>
  <audioProgramme/>
  ...
  <audioChannelFormat/>
</audioFormatExtended>
</frame>

```

以下S-ADM XML代码显示了如何使用父元素内的**audioFormatExtended**表示上述代码中的帧。父元素用于S-ADM帧的任何帧中。

```

<frame>
  <frameHeader>
    <frameFormat frameFormatID = "FF_00000001" start = "00:00:00.00000" duration = "00:00:00.50000"
    flowID = "12345678-abcd-4000-a000-112233445566" type = "header"/>
    <transportTrackFormat/>
  </frameHeader>

  <!-- the associated XML code of the ADM is written below -->
  <coreMetadata>
    <format>
      <audioFormatExtended>
        <audioProgramme/>
        ...
        <audioChannelFormat/>
      </audioFormatExtended>
    </format>
  </coreMetadata>
</frame>

<frame>
  <frameHeader>
    <frameFormat frameFormatID="FF_00000002" start="00:00:00.50000" duration="00:00:00.50000"
    flowID="12345678-abcd-4000-a000-112233445566" type="full"/>
    <transportTrackFormat/>
  </frameHeader>

  <!-- the associated XML code of the ADM is written below -->
  <coreMetadata>
    <format>
      <audioFormatExtended>
        <audioProgramme/>
        ...
        <audioChannelFormat/>
      </audioFormatExtended>
    </format>
  </coreMetadata>
</frame>

```

A2.2 changedIDs用法示例

下面的示例代码显示了三个 **audioChannelFormat** 如何在两个帧（'FF_00000003'和'FF_00000004'）上进行更改，以及如何设置**changedIDs**列表ID引用中的**status**属性：

- “AC_00031001” 首先出现在“FF_00000003”中（因此**status**=“新的”），并在“FF_00000004”中添加另一个**audioBlockFormat**（因此**status**=“经更改的”）；
- “AC_000310002” 已由“FF_00000003”建立（因此未列出ID引用），并在“FF_00000004”中消失（因此**status**=“过期的”）；
- “AC_000310003” 已由“FF_00000003”建立（因此未列出ID引用），“FF_00000004”中**audioBlockFormat**的持续时间已增加（因此**status**=“经扩展的”）。

```
<frame>
  <frameHeader>
    <frameFormat    frameFormatID="FF_00000003"    start="00:00:02.00000"    duration="00:00:01.00000"
    type="full">
      <changedIDs>
        <audioChannelFormatIDRef status="new">AC_00031001</audioChannelFormatIDRef>
      </changedIDs>
    </frameFormat>
    ...
  </frameHeader>

  <audioFormatExtended>
    ...
    <audioChannelFormat audioChannelFormatID="AC_00031001">
      <audioBlockFormat    audioBlockFormatID="AB_00031001_00000001"    rtime="00:00:00.00000"
      duration="00:00:01.00000">
        <position coordinate= "azimuth">30.0</position>
        <position coordinate= "elevation">0.0</position>
      </audioBlockFormat>
    </audioChannelFormat>
    <audioChannelFormat audioChannelFormatID="AC_00031002">
      <audioBlockFormat    audioBlockFormatID="AB_00031002_00000002"    rtime="00:00:01.00000"
      duration="00:00:01.00000">
        <position coordinate= "azimuth">45.0</position>
        <position coordinate= "elevation">0.0</position>
      </audioBlockFormat>
    </audioChannelFormat>
    <audioChannelFormat audioChannelFormatID="AC_00031003">
      <audioBlockFormat    audioBlockFormatID="AB_00031003_00000002"    rtime="00:00:01.00000"
      duration="00:00:01.00000">
        <position coordinate= "azimuth">90.0</position>
        <position coordinate= "elevation">0.0</position>
      </audioBlockFormat>
    </audioChannelFormat>
    ...
  </audioFormatExtended>
```

```

</frame>
<frame>
  <frameHeader>
    <frameFormat frameFormatID="FF_00000004" start="00:00:03.00000" duration="00:00:01.00000" type=
"full">
      <changedIDs>
        <audioChannelFormatIDRef status="changed">AC_00031001</audioChannelFormatIDRef>
        <audioChannelFormatIDRef status="expired">AC_00031002</audioChannelFormatIDRef>
        <audioChannelFormatIDRef status="extended">AC_00031003</audioChannelFormatIDRef>
      </changedIDs>
    </frameFormat>
    ...
  </frameHeader>

  <audioFormatExtended>
    ...
    <audioChannelFormat audioChannelFormatID="AC_00031001">
      <audioBlockFormat      audioBlockFormatID="AB_00031001_00000001"      rtime="00:00:00.00000"
duration="00:00:01.00000">
        <position coordinate= "azimuth">30.0</position>
        <position coordinate= "elevation">0.0</position>
      </audioBlockFormat>
      <audioBlockFormat      audioBlockFormatID="AB_00031001_00000002"      rtime="00:00:01.00000"
duration="00:00:01.00000">
        <position coordinate= "azimuth">20.0</position>
        <position coordinate= "elevation">0.0</position>
      </audioBlockFormat>
    </audioChannelFormat>
    <audioChannelFormat audioChannelFormatID="AC_00031003">
      <audioBlockFormat      audioBlockFormatID="AB_00031003_00000002"      rtime="00:00:01.00000"
duration="00:00:02.00000">
        <position coordinate= "azimuth">90.0</position>
        <position coordinate= "elevation">0.0</position>
      </audioBlockFormat>
    </audioChannelFormat>
    ...
  </audioFormatExtended>
</frame>

```

A2.3 源自ADM XML的S-ADM XML示例

本示例显示了如何从一个ADM XML文件派生出一组S-ADM帧。

初始ADM的XML代码样本如下所示：

```

<audioFormatExtended>
  <audioProgramme  audioProgrammeID="APR_1001"  audioProgrammeName="Main"  start="10:00:00.00000"
end="10:00:10.00000">
    <audioContentIDRef>ACO_1001</audioContentIDRef>
  </audioProgramme>
  <audioContent  audioContentID="ACO_1001">

```

```

    <audioObjectIDRef>AO_1001</audioObjectIDRef>
</audioContent>
<audioObject audioObjectID="AO_1001" start="00:00:00.00000" duration="00:00:10.00000">
    <audioPackFormatIDRef>AP_00031001</audioPackFormatIDRef>
    <audioTrackUIDRef>ATU_00000001</audioTrackUIDRef>
</audioObject>
<audioPackFormat audioPackFormatID="AP_00031001">
    <audioChannelFormatIDRef>AC_00031001</audioChannelFormatIDRef>
</audioPackFormat>
<audioTrackUID UID="ATU_00000001" sampleRate="48000" bitDepth="24">
    <audioPackFormatIDRef>AP_00031001</audioPackFormatIDRef>
    <audioTrackFormatIDRef>AT_00031001_01</audioTrackFormatIDRef>
</audioTrackUID>
<audioTrackFormat audioTrackFormatID="AT_00031001_01">
    <audioStreamFormatIDRef>AS_00031001</audioStreamFormatIDRef>
</audioTrackFormat>
<audioStreamFormat audioStreamFormatID="AS_00031001">
    <audioTrackFormatIDRef>AT_00031001_01</audioTrackFormatIDRef>
    <audioChannelFormatIDRef>AC_00031001</audioChannelFormatIDRef>
</audioStreamFormat>
<audioChannelFormat audioChannelFormatID="AC_00031001">
    <audioBlockFormat          audioBlockFormatID="AB_00031001_00000001"          rtime="00:00:00.00000"
duration="00:00:03.00000">
        <position coordinate= "azimuth">30.0</position>
        <position coordinate= "elevation">0.0</position>
        <jumpPosition>1</jumpPosition>
    </audioBlockFormat>
    <audioBlockFormat          audioBlockFormatID="AB_00031001_00000002"          rtime="00:00:03.00000"
duration="00:00:03.00000">
        <position coordinate= "azimuth">-30.0</position>
        <position coordinate= "elevation">0.0</position>
        <jumpPosition>1</jumpPosition>
    </audioBlockFormat>
    <audioBlockFormat          audioBlockFormatID="AB_00031001_00000003"          rtime="00:00:06.00000"
duration="00:00:03.00000">
        <position coordinate= "azimuth">0.0</position>
        <position coordinate= "elevation">0.0</position>
        <jumpPosition>0</jumpPosition>
    </audioBlockFormat>
    <audioBlockFormat          audioBlockFormatID="AB_00031001_00000004"          rtime="00:00:09.00000"
duration="00:00:01.00000">
        <position coordinate= "azimuth">30.0</position>
        <position coordinate= "elevation">0.0</position>
        <jumpPosition>0</jumpPosition>
    </audioBlockFormat>
</audioChannelFormat>
</audioFormatExtended>

```

具有1.5秒帧大小的S-ADM XML代码样本和MF流如下所示：

```
<frame>
  <frameHeader>
    <frameFormat frameFormatID="FF_00000001" start="10:00:00.00000" duration="00:00:01.50000" type=
"header"/>
    <transportTrackFormat transportID="TP_0001" transportName="AES3-A" numIDs="1" numTracks="1">
      <audioTrack trackID="1">
        <audioTrackUIDRef>ATU_00000001</audioTrackUIDRef>
      </audioTrack>
    </transportTrackFormat>
  </frameHeader>

  <audioFormatExtended>
    <audioProgramme audioProgrammeID="APR_1001" audioProgrammeName="Main" start="10:00:00.00000"
end="10:00:10.00000">
      <audioContentIDRef>ACO_1001</audioContentIDRef>
    </audioProgramme>
    <audioContent audioContentID="ACO_1001">
      <audioObjectIDRef>AO_1001</audioObjectIDRef>
    </audioContent>
    <audioObject audioObjectID="AO_1001" start="00:00:00.00000" duration="00:00:10.00000">
      <audioPackFormatIDRef>AP_00031001</audioPackFormatIDRef>
      <audioTrackUIDRef>ATU_00000001</audioTrackUIDRef>
    </audioObject>
    <audioPackFormat audioPackFormatID="AP_00031001">
      <audioChannelFormatIDRef>AC_00031001</audioChannelFormatIDRef>
    </audioPackFormat>
    <audioChannelFormat audioChannelFormatID="AC_00031001">
      <audioBlockFormat audioBlockFormatID="AB_00031001_00000001" rtime="00:00:00.00000"
duration="00:00:03.00000">
        <position coordinate= "azimuth">30.0</position>
        <position coordinate= "elevation">0.0</position>
        <jumpPosition>1</jumpPosition>
      </audioBlockFormat>
    </audioChannelFormat>
    <audioStreamFormat audioStreamFormatID="AS_00031001">
      <audioTrackFormatIDRef>AT_00031001_01</audioTrackFormatIDRef>
      <audioChannelFormatIDRef>AC_00031001</audioChannelFormatIDRef>
    </audioStreamFormat>
    <audioTrackFormat audioTrackFormatID="AT_00031001_01">
      <audioStreamFormatIDRef>AS_00031001</audioStreamFormatIDRef>
    </audioTrackFormat>
    <audioTrackUID UID="ATU_00000001" sampleRate="48000" bitDepth="24">
      <audioPackFormatIDRef>AP_00031001</audioPackFormatIDRef>
      <audioTrackFormatIDRef>AT_00031001_01</audioTrackFormatIDRef>
    </audioTrackUID>
  </audioFormatExtended>
</frame>
```

```

<frame>
  <frameHeader>
    <frameFormat frameFormatID="FF_00000002" start="10:00:01.50000" duration="00:00:01.50000"
type="intermediate" countToFull="3"/>
  </frameHeader>

  <audioFormatExtended/>
</frame>
<frame>
  <frameHeader>
    <frameFormat frameFormatID="FF_00000003" start="10:00:03.00000" duration="00:00:01.50000"
type="intermediate" countToFull="2"/>
    <changedIDs>
      <audioChannelFormatIDRef status="changed">AC_00031001</audioChannelFormatIDRef>
    </changedIDs>
  </frameFormat>
</frameHeader>

  <audioFormatExtended>
    <audioChannelFormat audioChannelFormatID="AC_00031001">
      <audioBlockFormat audioBlockFormatID="AB_00031001_00000002" rtime="00:00:03.00000"
duration="00:00:03.00000">
        <position coordinate="azimuth">-30.0</position>
        <position coordinate="elevation">0.0</position>
        <jumpPosition>1</jumpPosition>
      </audioBlockFormat>
    </audioChannelFormat>
  </audioFormatExtended>
</frame>
<frame>
  <frameHeader>
    <frameFormat frameFormatID="FF_00000004" start="10:00:04.50000" duration="00:00:01.50000"
type="intermediate" countToFull="1"/>
  </frameHeader>

  <audioFormatExtended/>
</frame>
<frame>
  <frameHeader>
    <frameFormat frameFormatID="FF_00000005" start="10:00:06.00000" duration="00:00:01.50000"
type="full">
    <changedIDs>
      <audioChannelFormatIDRef status="changed"> AC_00031001</audioChannelFormatIDRef>
    </changedIDs>
  </frameFormat>
  <transportTrackFormat transportID="TP_0001" transportName="AES3-A" numIDs="1" numTracks="1">
    <audioTrack trackID="1">
      <audioTrackUIDRef>ATU_00000001</audioTrackUIDRef>
    </audioTrack>
  </transportTrackFormat>
</frame>

```

```

    </transportTrackFormat>
</frameHeader>

<audioFormatExtended>
  <audioProgramme audioProgrammeID="APR_1001" audioProgrammeName="Main" start="10:00:00.00000"
end="10:00:10.00000">
    <audioContentIDRef>ACO_1001</audioContentIDRef>
  </audioProgramme>
  <audioContent audioContentID="ACO_1001">
    <audioObjectIDRef>AO_1001</audioObjectIDRef>
  </audioContent>
  <audioObject audioObjectID="AO_1001" start="00:00:00.00000" duration="00:00:10.00000">
    <audioPackFormatIDRef>AP_00031001</audioPackFormatIDRef>
    <audioTrackUIDRef>ATU_00000001</audioTrackUIDRef>
  </audioObject>
  <audioPackFormat audioPackFormatID="AP_00031001">
    <audioChannelFormatIDRef>AC_00031001</audioChannelFormatIDRef>
  </audioPackFormat>
  <audioChannelFormat audioChannelFormatID="AC_00031001">
    <audioBlockFormat audioBlockFormatID="AB_00031001_00000002" rtime="00:00:03.00000"
duration="00:00:03.00000">
      <position coordinate= "azimuth">-30.0</position>
      <position coordinate= "elevation">0.0</position>
      <jumpPosition>1</jumpPosition>
    </audioBlockFormat>
    <audioBlockFormat audioBlockFormatID="AB_00031001_00000003" rtime="00:00:06.00000"
duration="00:00:03.00000">
      <position coordinate= "azimuth">0.0</position>
      <position coordinate= "elevation">0.0</position>
      <jumpPosition>0</jumpPosition>
    </audioBlockFormat>
  </audioChannelFormat>
  <audioStreamFormat audioStreamFormatID="AS_00031001">
    <audioTrackFormatIDRef>AT_00031001_01</audioTrackFormatIDRef>
    <audioChannelFormatIDRef>AC_00031001</audioChannelFormatIDRef>
  </audioStreamFormat>
  <audioTrackFormat audioTrackFormatID="AT_00031001_01">
    <audioStreamFormatIDRef>AS_00031001</audioStreamFormatIDRef>
  </audioTrackFormat>
  <audioTrackUID UID="ATU_00000001" sampleRate="48000" bitDepth="24">
    <audioPackFormatIDRef>AP_00031001</audioPackFormatIDRef>
    <audioTrackFormatIDRef>AT_00031001_01</audioTrackFormatIDRef>
  </audioTrackUID>
</audioFormatExtended>
</frame>
<frame>
  <frameHeader>
    <frameFormat frameFormatID="FF_00000006" start="10:00:07.50000" duration="00:00:01.50000"
type="intermediate" countToFull="3"/>

```

```

</frameHeader>

<audioFormatExtended/>
</frame>
<frame>
  <frameHeader>
    <frameFormat frameFormatID="FF_00000007" start="10:00:09.00000" duration="00:00:01.00000"
    type="intermediate" countToFull="2">
      <changedIDs>
        <audioChannelFormatIDRef status="changed">AC_00031001</audioChannelFormatIDRef>
      </changedIDs>
    </frameFormat>
  </frameHeader>

  <audioFormatExtended>
    <audioChannelFormat audioChannelFormatID="AC_00031001">
      <audioBlockFormat audioBlockFormatID="AB_00031001_00000004" rtime="00:00:09.00000"
      duration="00:00:01.00000">
        <position coordinate= "azimuth">30.0</position>
        <position coordinate= "elevation">0.0</position>
        <jumpPosition>0</jumpPosition>
      </audioBlockFormat>
    </audioChannelFormat>
  </audioFormatExtended>
</frame>

```

具有1.5秒帧大小的S-ADM XML代码样本和DF流如下所示：

```

<frame>
  <frameHeader>
    <frameFormat frameFormatID="FF_00000001_01" start="10:00:00.00000"
    duration="00:00:01.50000" type="divided" numMetadataChunks="4" countToSameChunk="1">
      <chunkAdmElement>audioProgramme</chunkAdmElement>
      <chunkAdmElement>audioContent</chunkAdmElement>
      <chunkAdmElement>audioObject</chunkAdmElement>
    </frameFormat>
    <transportTrackFormat transportID="TP_0001" transportName="AES3-A" numIDs="1" numTracks="1">
      <audioTrack trackID="1">
        <audioTrackUIDRef>ATU_00000001</audioTrackUIDRef>
      </audioTrack>
    </transportTrackFormat>
  </frameHeader>

  <audioFormatExtended>
    <audioProgramme audioProgrammeID="APR_1001" audioProgrammeName="Main" start="10:00:00.00000"
    end="10:00:10.00000">
      <audioContentIDRef>ACO_1001</audioContentIDRef>
    </audioProgramme>
  </audioFormatExtended>
</frame>

```

```

<audioContent audioContentID="ACO_1001">
  <audioObjectIDRef>AO_1001</audioObjectIDRef>
</audioContent>
<audioObject audioObjectID="AO_1001" start="00:00:00.00000" duration="00:00:10.00000">
  <audioPackFormatIDRef>AP_00031001</audioPackFormatIDRef>
  <audioTrackUIDRef>ATU_00000001</audioTrackUIDRef>
</audioObject>
</audioFormatExtended>
</frame>

<frame>
  <frameHeader>
    <frameFormat frameFormatID="FF_00000001_02" start="10:00:00.00000"
      duration="00:00:01.50000" type="divided" numMetadataChunks="4" countToSameChunk="2">
      <chunkAdmElement>audioPackFormat</chunkAdmElement>
      <chunkAdmElement>audioStreamFormat</chunkAdmElement>
    </frameFormat>
  </frameHeader>

  <audioFormatExtended>
    <audioPackFormat audioPackFormatID="AP_00031001">
      <audioChannelFormatIDRef>AC_00031001</audioChannelFormatIDRef>
    </audioPackFormat>
    <audioStreamFormat audioStreamFormatID="AS_00031001">
      <audioTrackFormatIDRef>AT_00031001_01</audioTrackFormatIDRef>
      <audioChannelFormatIDRef>AC_00031001</audioChannelFormatIDRef>
    </audioStreamFormat>
  </audioFormatExtended>
</frame>

<frame>
  <frameHeader>
    <frameFormat frameFormatID="FF_00000001_03" start="10:00:00.00000"
      duration="00:00:01.50000" type="divided" numMetadataChunks="4" countToSameChunk="3">
      <chunkAdmElement>audioTrackFormat</chunkAdmElement>
      <chunkAdmElement>audioTrackUID</chunkAdmElement>
    </frameFormat>
  </frameHeader>

  <audioFormatExtended>
    <audioTrackFormat audioTrackFormatID="AT_00031001_01">
      <audioStreamFormatIDRef>AS_00031001</audioStreamFormatIDRef>
    </audioTrackFormat>
    <audioTrackUID UID="ATU_00000001" sampleRate="48000" bitDepth="24">
      <audioPackFormatIDRef>AP_00031001</audioPackFormatIDRef>
      <audioTrackFormatIDRef>AT_00031001_01</audioTrackFormatIDRef>
    </audioTrackUID>
  </audioFormatExtended>
</frame>

```

```

    </audioFormatExtended>
</frame>

<frame>
  <frameHeader>
    <frameFormat frameFormatID="FF_00000001_04" start="10:00:00.00000"
      duration="00:00:01.50000" type="divided" numMetadataChunks="4" countToSameChunk="1">
      <chunkAdmElement>audioChannelFormat</chunkAdmElement>
    </frameFormat>
  </frameHeader>

  <audioFormatExtended>
    <audioChannelFormat audioChannelFormatID="AC_00031001">
      <audioBlockFormat audioBlockFormatID="AB_00031001_00000001" rtime="00:00:00.00000"
        duration="00:00:03.00000">
        <position coordinate="azimuth">30.0</position>
        <position coordinate="elevation">0.0</position>
        <jumpPosition>1</jumpPosition>
      </audioBlockFormat>
    </audioChannelFormat>
  </audioFormatExtended>
</frame>

<frame>
  <frameHeader>
    <frameFormat frameFormatID="FF_00000002_01" start="10:00:01.50000"
      duration="00:00:01.50000" type="divided" numMetadataChunks="4" countToSameChunk="3">
      <chunkAdmElement>audioProgramme</chunkAdmElement>
      <chunkAdmElement>audioContent</chunkAdmElement>
      <chunkAdmElement>audioObject</chunkAdmElement>
    </frameFormat>
    <transportTrackFormat transportID="TP_0001" transportName="AES3-A" numIDs="1" numTracks="1">
      <audioTrack trackID="1">
        <audioTrackUIDRef>ATU_00000001</audioTrackUIDRef>
      </audioTrack>
    </transportTrackFormat>
  </frameHeader>

  <audioFormatExtended>
    <audioProgramme audioProgrammeID="APR_1001" audioProgrammeName="Main" start="10:00:00.00000"
      end="10:00:10.00000">
      <audioContentIDRef>ACO_1001</audioContentIDRef>
    </audioProgramme>
    <audioContent audioContentID="ACO_1001">
      <audioObjectIDRef>AO_1001</audioObjectIDRef>
    </audioContent>
    <audioObject audioObjectID="AO_1001" start="00:00:00.00000" duration="00:00:10.00000">
      <audioPackFormatIDRef>AP_00031001</audioPackFormatIDRef>
    </audioObject>
  </audioFormatExtended>
</frame>

```

```

    <audioTrackUIDRef>ATU_00000001</audioTrackUIDRef>
  </audioObject>
</audioFormatExtended>
</frame>

<frame>
  <frameHeader>
    <frameFormat frameFormatID="FF_00000002_04" start="10:00:01.50000"
      duration="00:00:01.50000" type="divided" numMetadataChunks="4" countToSameChunk="1">
      <chunkAdmElement>audioChannelFormat</chunkAdmElement>
    </frameFormat>
  </frameHeader>

  <audioFormatExtended>
    <audioChannelFormat audioChannelFormatID="AC_00031001">
      <audioBlockFormat audioBlockFormatID="AB_00031001_00000001" rtime="00:00:00.00000"
        duration="00:00:03.00000">
        <position coordinate="azimuth">30.0</position>
        <position coordinate="elevation">0.0</position>
        <jumpPosition>1</jumpPosition>
      </audioBlockFormat>
    </audioChannelFormat>
  </audioFormatExtended>
</frame>

<frame>
  <frameHeader>
    <frameFormat frameFormatID="FF_00000003_02" start="10:00:03.00000"
      duration="00:00:01.50000" type="divided" numMetadataChunks="4" countToSameChunk="3">
      <chunkAdmElement>audioPackFormat</chunkAdmElement>
      <chunkAdmElement>audioStreamFormat</chunkAdmElement>
    </frameFormat>

    <transportTrackFormat transportID="TP_0001" transportName="AES3-A" numIDs="1" numTracks="1">
      <audioTrack trackID="1">
        <audioTrackUIDRef>ATU_00000001</audioTrackUIDRef>
      </audioTrack>
    </transportTrackFormat>
  </frameHeader>

  <audioFormatExtended>
    <audioPackFormat audioPackFormatID="AP_00031001">
      <audioChannelFormatIDRef>AC_00031001</audioChannelFormatIDRef>
    </audioPackFormat>

    <audioStreamFormat audioStreamFormatID="AS_00031001">
      <audioTrackFormatIDRef>AT_00031001_01</audioTrackFormatIDRef>
      <audioChannelFormatIDRef>AC_00031001</audioChannelFormatIDRef>
    </audioStreamFormat>
  </audioFormatExtended>

```

```

</frame>

<frame>
  <frameHeader>
    <frameFormat frameFormatID="FF_00000003_04" start="10:00:03.00000"
      duration="00:00:01.50000" type="divided" numMetadataChunks="4" countToSameChunk="1">
      <chunkAdmElement>audioChannelFormat</chunkAdmElement>
    </frameFormat>
  </frameHeader>

  <audioFormatExtended>
    <audioChannelFormat audioChannelFormatID="AC_00031001">
      <audioBlockFormat audioBlockFormatID="AB_00031001_00000002" rtime="00:00:03.00000"
        duration="00:00:03.00000">
        <position coordinate="azimuth">-30.0</position>
        <position coordinate="elevation">0.0</position>
        <jumpPosition>1</jumpPosition>
      </audioBlockFormat>
    </audioChannelFormat>
  </audioFormatExtended>
</frame>

<frame>
  <frameHeader>
    <frameFormat frameFormatID="FF_00000004_03" start="10:00:04.50000"
      duration="00:00:01.50000" type="divided" numMetadataChunks="4" countToSameChunk="3">
      <chunkAdmElement>audioTrackFormat</chunkAdmElement>
      <chunkAdmElement>audioTrackUID</chunkAdmElement>
    </frameFormat>
    <transportTrackFormat transportID="TP_0001" transportName="AES3-A" numIDs="1" numTracks="1">
      <audioTrack trackID="1">
        <audioTrackUIDRef>ATU_00000001</audioTrackUIDRef>
      </audioTrack>
    </transportTrackFormat>
  </frameHeader>

  <audioFormatExtended>
    <audioTrackFormat audioTrackFormatID="AT_00031001_01">
      <audioStreamFormatIDRef>AS_00031001</audioStreamFormatIDRef>
    </audioTrackFormat>
    <audioTrackUID UID="ATU_00000001" sampleRate="48000" bitDepth="24">
      <audioPackFormatIDRef>AP_00031001</audioPackFormatIDRef>
      <audioTrackFormatIDRef>AT_00031001_01</audioTrackFormatIDRef>
    </audioTrackUID>
  </audioFormatExtended>
</frame>

<frame>

```

```

<frameHeader>
  <frameFormat frameFormatID="FF_00000004_04" start="10:00:04.50000"
    duration="00:00:01.50000" type="divided" numMetadataChunks="4" countToSameChunk="1">
    <chunkAdmElement>audioChannelFormat</chunkAdmElement>
    <chunkAdmElement>audioBlockFormat</chunkAdmElement>
  </frameFormat>
</frameHeader>

<audioFormatExtended>
  <audioChannelFormat audioChannelFormatID="AC_00031001">
    <audioBlockFormat audioBlockFormatID="AB_00031001_00000002" rtime="00:00:03.00000"
      duration="00:00:03.00000">
      <position coordinate="azimuth">-30.0</position>
      <position coordinate="elevation">0.0</position>
      <jumpPosition>1</jumpPosition>
    </audioBlockFormat>
  </audioChannelFormat>
</audioFormatExtended>
</frame>

<frame>
  <frameHeader>
    <frameFormat frameFormatID="FF_00000005_01" start="10:00:06.00000"
      duration="00:00:01.50000" type="divided" numMetadataChunks="4" countToSameChunk="3">
      <chunkAdmElement>audioProgramme</chunkAdmElement>
      <chunkAdmElement>audioContent</chunkAdmElement>
      <chunkAdmElement>audioObject</chunkAdmElement>
    </frameFormat>

    <transportTrackFormat transportID="TP_0001" transportName="AES3-A" numIDs="1" numTracks="1">
      <audioTrack trackID="1">
        <audioTrackUIDRef>ATU_00000001</audioTrackUIDRef>
      </audioTrack>
    </transportTrackFormat>
  </frameHeader>

  <audioFormatExtended>
    <audioProgramme audioProgrammeID="APR_1001" audioProgrammeName="Main" start="10:00:00.00000"
      end="10:00:10.00000">
      <audioContentIDRef>ACO_1001</audioContentIDRef>
    </audioProgramme>
    <audioContent audioContentID="ACO_1001">
      <audioObjectIDRef>AO_1001</audioObjectIDRef>
    </audioContent>
    <audioObject audioObjectID="AO_1001" start="00:00:00.00000" duration="00:00:10.00000">
      <audioPackFormatIDRef>AP_00031001</audioPackFormatIDRef>
      <audioTrackUIDRef>ATU_00000001</audioTrackUIDRef>
    </audioObject>
  </audioFormatExtended>

```

```

</frame>

<frame>
  <frameHeader>
    <frameFormat frameFormatID="FF_00000005_04" start="10:00:06.00000"
      duration="00:00:01.50000" type="divided" numMetadataChunks="4" countToSameChunk="1">
      <chunkAdmElement>audioChannelFormat</chunkAdmElement>
    </frameFormat>
  </frameHeader>

  <audioFormatExtended>
    <audioChannelFormat audioChannelFormatID="AC_00031001">
      <audioBlockFormat audioBlockFormatID="AB_00031001_00000002" rtime="00:00:03.00000"
        duration="00:00:03.00000">
        <position coordinate="azimuth">-30.0</position>
        <position coordinate="elevation">0.0</position>
        <jumpPosition>1</jumpPosition>
      </audioBlockFormat>
      <audioBlockFormat audioBlockFormatID="AB_00031001_00000003" rtime="00:00:06.00000"
        duration="00:00:03.00000">
        <position coordinate="azimuth">0.0</position>
        <position coordinate="elevation">0.0</position>
        <jumpPosition>0</jumpPosition>
      </audioBlockFormat>
    </audioChannelFormat>
  </audioFormatExtended>
</frame>

<frame>
  <frameHeader>
    <frameFormat frameFormatID="FF_00000006_02" start="10:00:07.50000"
      duration="00:00:01.50000" type="divided" numMetadataChunks="4" countToSameChunk="3">
      <chunkAdmElement>audioPackFormat</chunkAdmElement>
      <chunkAdmElement>audioStreamFormat</chunkAdmElement>
    </frameFormat>
    <transportTrackFormat transportID="TP_0001" transportName="AES3-A" numIDs="1" numTracks="1">
      <audioTrack trackID="1">
        <audioTrackUIDRef>ATU_00000001</audioTrackUIDRef>
      </audioTrack>
    </transportTrackFormat>
  </frameHeader>

  <audioFormatExtended>
    <audioPackFormat audioPackFormatID="AP_00031001">
      <audioChannelFormatIDRef>AC_00031001</audioChannelFormatIDRef>
    </audioPackFormat>
    <audioStreamFormat audioStreamFormatID="AS_00031001">
      <audioTrackFormatIDRef>AT_00031001_01</audioTrackFormatIDRef>
    </audioStreamFormat>
  </audioFormatExtended>
</frame>

```

```

    <audioChannelFormatIDRef>AC_00031001</audioChannelFormatIDRef>
  </audioStreamFormat>
</audioFormatExtended>
</frame>

<frame>
  <frameHeader>
    <frameFormat frameFormatID="FF_00000006_04" start="10:00:07.50000"
      duration="00:00:01.50000" type="divided" numMetadataChunks="4" countToSameChunk="1">
      <chunkAdmElement>audioChannelFormat</chunkAdmElement>
    </frameFormat>
  </frameHeader>

  <audioFormatExtended>
    <audioChannelFormat audioChannelFormatID="AC_00031001">
      <audioBlockFormat audioBlockFormatID="AB_00031001_00000002" rtime="00:00:03.00000"
        duration="00:00:03.00000">
        <position coordinate="azimuth">-30.0</position>
        <position coordinate="elevation">0.0</position>
        <jumpPosition>1</jumpPosition>
      </audioBlockFormat>
      <audioBlockFormat audioBlockFormatID="AB_00031001_00000003" rtime="00:00:06.00000"
        duration="00:00:03.00000">
        <position coordinate="azimuth">0.0</position>
        <position coordinate="elevation">0.0</position>
        <jumpPosition>0</jumpPosition>
      </audioBlockFormat>
    </audioChannelFormat>
  </audioFormatExtended>
</frame>

<frame>
  <frameHeader>
    <frameFormat frameFormatID="FF_00000007_03" start="10:00:09.00000"
      duration="00:00:01.50000" type="divided" numMetadataChunks="4" countToSameChunk="3">
      <chunkAdmElement>audioTrackFormat</chunkAdmElement>
      <chunkAdmElement>audioTrackUID</chunkAdmElement>
    </frameFormat>
    <transportTrackFormat transportID="TP_0001" transportName="AES3-A" numIDs="1" numTracks="1">
      <audioTrack trackID="1">
        <audioTrackUIDRef>ATU_00000001</audioTrackUIDRef>
      </audioTrack>
    </transportTrackFormat>
  </frameHeader>

  <audioFormatExtended>
    <audioTrackFormat audioTrackFormatID="AT_00031001_01">
      <audioStreamFormatIDRef>AS_00031001</audioStreamFormatIDRef>

```

```

</audioTrackFormat>
<audioTrackUID UID="ATU_00000001" sampleRate="48000" bitDepth="24">
  <audioPackFormatIDRef>AP_00031001</audioPackFormatIDRef>
  <audioTrackFormatIDRef>AT_00031001_01</audioTrackFormatIDRef>
</audioTrackUID>
</audioFormatExtended>
</frame>

<frame>
  <frameHeader>
    <frameFormat frameFormatID="FF_00000007_04" start="10:00:09.00000"
      duration="00:00:01.50000" type="divided" numMetadataChunks="4" countToSameChunk="1">
      <chunkAdmElement>audioChannelFormat</chunkAdmElement>
    </frameFormat>
  </frameHeader>

  <audioFormatExtended>
    <audioChannelFormat audioChannelFormatID="AC_00031001">
      <audioBlockFormat audioBlockFormatID="AB_00031001_00000003" rtime="00:00:06.00000"
        duration="00:00:03.00000">
        <position coordinate="azimuth">0.0</position>
        <position coordinate="elevation">0.0</position>
        <jumpPosition>0</jumpPosition>
      </audioBlockFormat>
      <audioBlockFormat audioBlockFormatID="AB_00031001_00000004" rtime="00:00:09.00000"
        duration="00:00:01.00000">
        <position coordinate="azimuth">30.0</position>
        <position coordinate="elevation">0.0</position>
        <jumpPosition>0</jumpPosition>
      </audioBlockFormat>
    </audioChannelFormat>
  </audioFormatExtended>
</frame>

```

A2.4 transportTrackFormat用法示例

本示例显示了**transportTrackFormat**如何等效于BW64文件中的“chna”数据块。

BW64 “chna”数据块中的元数据示例如下所示：

```

ckID = {'c', 'h', 'n', 'a'};
ckSize = 164;
numTracks = 3;
numUIDs = 4;
ID[0] = {trackIndex=1; UID="ATU_00000001"; trackRef="AT_00031001_01"; packRef="AP_00031001"};
ID[1] = {trackIndex=1; UID="ATU_00000002"; trackRef="AT_00031002_01"; packRef="AP_00031002"};
ID[2] = {trackIndex=2; UID="ATU_00000003"; trackRef="AT_00031003_01"; packRef="AP_00031001"};
ID[3] = {trackIndex=3; UID="ATU_00000004"; trackRef="AT_00031004_01"; packRef="AP_00031003"};

```

在通过两个 AES-3 接口传送上述音频信号的情况下， **transportTrackFormat** 和 **audioTrackUID** 的示例如下所示：

```
<!-- ##### -->
<!-- Transport Track -->
<!-- ##### -->
<transportTrackFormat transportID="TP_0001" transportName="AES3-A" numIDs="3" numTracks="2">
  <audioTrack trackID="1">
    <audioTrackUIDRef>ATU_00000001</audioTrackUIDRef>
    <audioTrackUIDRef>ATU_00000002</audioTrackUIDRef>
  </audioTrack>
  <audioTrack trackID="2">
    <audioTrackUIDRef>ATU_00000003</audioTrackUIDRef>
  </audioTrack>
</transportTrackFormat>
<transportTrackFormat transportID="TP_0002" transportName="AES3-B" numIDs="1" numTracks="1">
  <audioTrack trackID="1">
    <audioTrackUIDRef>ATU_00000004</audioTrackUIDRef>
  </audioTrack>
</transportTrackFormat>

<!-- ##### -->
<!-- Audio Track UUIDs -->
<!-- ##### -->
<audioFormatExtended>
  ...
  <audioTrackUID UID="ATU_00000001" sampleRate="48000" bitDepth="24">
    <audioTrackFormatIDRef>AT_00031001_01</audioTrackFormatIDRef>
    <audioPackFormatIDRef>AP_00031001</audioPackFormatIDRef>
  </audioTrackUID>
  <audioTrackUID UID="ATU_00000002" sampleRate="48000" bitDepth="24">
    <audioTrackFormatIDRef>AT_00031002_01</audioTrackFormatIDRef>
    <audioPackFormatIDRef>AP_00031002</audioPackFormatIDRef>
  </audioTrackUID>
  <audioTrackUID UID="ATU_00000003" sampleRate="48000" bitDepth="24">
    <audioTrackFormatIDRef>AT_00031003_01</audioTrackFormatIDRef>
    <audioPackFormatIDRef>AP_00031001</audioPackFormatIDRef>
  </audioTrackUID>
  <audioTrackUID UID="ATU_00000004" sampleRate="48000" bitDepth="24">
    <audioTrackFormatIDRef>AT_00031004_01</audioTrackFormatIDRef>
    <audioPackFormatIDRef>AP_00031003</audioPackFormatIDRef>
  </audioTrackUID>
  ...
</audioFormatExtended>
```

A2.5 timeReference标志用法示例

本示例显示了如何在“总计”或“本地” **timeReference** 模式下将ADM XML转换为S-ADM。来自初始ADM的**audioBlockFormat**定时参数（**rtime**和**duration**）保留在“总计”版的S-ADM XML中；在“本地”版中，这些定时参数被转换为**lstart**和**lduration**。

初始ADM中描述的XML代码示例如下所示（为了提高清晰度，一些属性和元素没有纳入进来）：

```
<audioFormatExtended>
...
<audioObject audioObjectID="AO_1001" start="00:00:01.00000">
  <audioPackFormatIDRef>AP_00031001</audioPackFormatIDRef>
</audioObject>
<audioPackFormat audioPackFormatID="AP_00031001">
  <audioChannelFormatIDRef>AC_00031001</audioChannelFormatIDRef>
</audioPackFormat>
<audioChannelFormat audioChannelFormatID="AC_00031001">
  <audioBlockFormat      audioBlockFormatID="AB_00031001_00000001"      rtime="00:00:00.00000"
duration="00:00:01.00000">
    <position coordinate= "azimuth">30.0</position>
    <position coordinate= "elevation">0.0</position>
    <jumpPosition>1</jumpPosition>
  </audioBlockFormat>
  <audioBlockFormat      audioBlockFormatID="AB_00031001_00000002"      rtime="00:00:01.00000"
duration="00:00:01.00000">
    <position coordinate="azimuth">0.0</position>
    <position coordinate="elevation">0.0</position>
    <jumpPosition>0</jumpPosition>
  </audioBlockFormat>
</audioChannelFormat>
...
</audioFormatExtended>
```

下面的例子是用“总计”时间在S-ADM中写成的。**audioBlockFormat**中的**rtime**相对于引用**audioObject**的开始时间：

```
<frame>
  <frameHeader>
    <frameFormat   frameFormatID="FF_00000003"   start="00:00:01.00000"   duration="00:00:00.50000"
timeReference= "total" type="full">
    <changedIDs>
      <audioChannelFormatIDRef status="changed">AC_00031001</audioChannelFormatIDRef>
    </changedIDs>
  </frameFormat>
</frameHeader>

<audioFormatExtended>
...
  <audioObject audioObjectID="AO_1001" start="00:00:01.00000">
```

```

    <audioPackFormatIDRef>AP_00031001</audioPackFormatIDRef>
  </audioObject>
  <audioPackFormat audioPackFormatID="AP_00031001">
    <audioChannelFormatIDRef>AC_00031001</audioChannelFormatIDRef>
  </audioPackFormat>
  <audioChannelFormat audioChannelFormatID="AC_00031001">
    <audioBlockFormat audioBlockFormatID="AB_00031001_00000001" rtime="00:00:00.00000"
duration="00:00:01.00000">
      <position coordinate="azimuth">30.0</position>
      <position coordinate="elevation">0.0</position>
      <jumpPosition>1</jumpPosition>
    </audioBlockFormat>
  </audioChannelFormat>
  ...
</audioFormatExtended>
</frame>

```

下面的示例是以“本地”时间在S-ADM中写成的。由于**audioBlockFormat**中的**ltime**是相对于帧开始时间的，因此引用**audioObject**不需要开始时间：

```

<frame>
  <frameHeader>
    <frameFormat frameFormatID="FF_00000003" start="00:00:01.00000" duration="00:00:00.50000"
timeReference="local" type="full">
      <changedIDs>
        <audioChannelFormatIDRef>AC_00031001</audioChannelFormatIDRef>
      </changedIDs>
    </frameFormat>
  </frameHeader>

  <audioFormatExtended>
    ...
    <audioObject audioObjectID="AO_1001">
      <audioPackFormatIDRef>AP_00031001</audioPackFormatIDRef>
    </audioObject>
    <audioPackFormat audioPackFormatID="AP_00031001">
      <audioChannelFormatIDRef>AC_00031001</audioChannelFormatIDRef>
    </audioPackFormat>
    <audioChannelFormat audioChannelFormatID="AC_00031001">
      <audioBlockFormat audioBlockFormatID="AB_00031001_00000000" initializeBlock="1">
        <position coordinate="azimuth">30.0</position>
        <position coordinate="elevation">0.0</position>
      </audioBlockFormat>
      <audioBlockFormat audioBlockFormatID="AB_00031001_00000002" ltime="00:00:00.00000"
lduration="00:00:00.50000">
        <position coordinate="azimuth">15.0</position>
        <position coordinate="elevation">0.0</position>
        <jumpPosition>0</jumpPosition>
      </audioBlockFormat>
    </audioChannelFormat>
  </audioFormatExtended>

```

```

    </audioChannelFormat>
    ...
  </audioFormatExtended>
</frame>

```

A2.6 广播元数据用法示例

本示例显示了广播元数据的使用方式。

```

<frame>
  <frameHeader>
    <frameFormat frameFormatID = "FF_00000001" start = "00:00:00.00000" duration = "00:00:00.50000"
    flowID = "12345678-abcd-4000-a000-112233445566" type = "header"/>
    <transportTrackFormat/>
  </frameHeader>

  <coreMetadata>
    <date>
      <created statDate="2000-10-10" startTime="12:00:00"/>
    </date>
    <format>
      <audioFormatExtended>
        <!--the XML code of the ADM is written here -->
        <audioProgramme/>
        ...
        <audioChannelFormat/>
      </audioFormatExtended>
    </format>
  </coreMetadata>
</frame>

<frame>
  <frameHeader>
    <frameFormat frameFormatID="FF_00000002" start="00:00:00.50000" duration="00:00:00.50000"
    flowID="12345678-abcd-4000-a000-112233445566" type="full"/>
    <transportTrackFormat/>
  </frameHeader>

  <coreMetadata>
    <format>
      <audioFormatExtended>
        <!--the XML code of the ADM is written here -->
        <audioProgramme/>
        ...
        <audioChannelFormat/>
      </audioFormatExtended>
    </format>
  </coreMetadata>
</frame>

```
