GD

XXXX – XX – XX发布

XXXX - XX – XX实施

中华人民共和国广播电影电视行业暂行技术文件

GD/J  XXX—XXXX

国家广播电视总局科技司 发布

5G广播业务数据封装  
与传输技术要求

Technical Requirements for Encapsulation and Transmission of Data Broadcasting Service in 5G Broadcasting Network

（提纲）

项目编号：技科字[2020]xxx号

目  次

[GD/J XXX—XXXX I](#_Toc69288135)

[目次 I](#_Toc69288136)

[前 言 III](#_Toc69288137)

[引 言 IV](#_Toc69288138)

[1 范围 5](#_Toc69288139)

[2 规范性引用文件 5](#_Toc69288140)

[3 术语、定义、缩略语和符号 5](#_Toc69288141)

[3.1 术语和定义 5](#_Toc69288142)

[3.2 缩略语 5](#_Toc69288143)

[4 系统概述（广科院） 5](#_Toc69288144)

[5 业务协议（信令）概述 7](#_Toc69288145)

[5.1 协议栈（高通） 7](#_Toc69288146)

[5.2 业务发现/发布（Service Discovery/Announcement）（高研院） 7](#_Toc69288147)

[6 协议 7](#_Toc69288148)

[6.1 FLUTE/ROUTE（高通、中兴） 7](#_Toc69288149)

[6.2 SMT（上海交大） 8](#_Toc69288150)

[7 关联分发过程(Associated delivery procedures) 12](#_Toc69288151)

[7.1 概述 12](#_Toc69288152)

[7.2 文件修复（高通） 12](#_Toc69288153)

[7.3 统计上报（高通） 12](#_Toc69288154)

[8 FEC 12](#_Toc69288155)

[8.1 Raptor/RaptorQ（高通） 12](#_Toc69288156)

[8.2 基于Raptor的算法（上海交大） 12](#_Toc69288157)

[8.3 RaptorG（广科院） 16](#_Toc69288158)

[9 媒体编解码和格式（高通、广科院） 16](#_Toc69288159)

[附　录　A （规范性附录） FEC编码规范 17](#_Toc69288160)

[A.1 概述 17](#_Toc69288161)

[A.2 FEC信令消息 17](#_Toc69288162)

[A.2.1 语法 17](#_Toc69288163)

[A.2.2 语义 18](#_Toc69288164)

[A.3 编码符号块格式 19](#_Toc69288165)

[A.3.1概述 19](#_Toc69288166)

[A.3.2源数据包组 20](#_Toc69288167)

[A.3.3源符号块格式 20](#_Toc69288168)

[A.4 FEC的源数据包和恢复数据包格式 22](#_Toc69288169)

[A.4.1概述 22](#_Toc69288170)

[A.4.2 FEC源数据包格式与负载ID 22](#_Toc69288171)

[A.4.3 FEC恢复数据包格式与负载ID 23](#_Toc69288172)

[附　录　B （资料性附录） 终端业务接收流程 25](#_Toc69288173)

[参考文献 26](#_Toc69288174)

前 言

1. 本技术文件按照GB/T 1.1—2009给出的规则起草。
2. 请注意本技术文件的某些内容可能涉及专利。本技术文件的发布机构不承担识别这些专利的责任。
3. 本技术文件由国家广播电视总局科技司归口。

本技术文件起草单位：国家广播电视总局广播电视科学研究院、xx。

1. 本技术文件主要起草人：xx。

引 言

1. 本标准制定了5G广播业务数据封装与传输技术要求。本标准参考国内和国际相关标准，并结合国内网络的实际情况制定。

5G广播业务数据封装与传输技术要求

1. 范围

本技术文件规定了5G广播业务数据封装协议和传输方式。

1. 规范性引用文件

下列文件对于本技术文件的应用是必不可少的。凡是注日期的引用文件，仅所注日期的版本适用于本技术文件。凡是不注日期的引用文件，其最新版本（包括所有的修改单）适用于本技术文件。

GB/T 28160—2011 数字电视广播电子节目指南规范

1. 术语、定义、缩略语和符号
   1. 术语和定义

下列术语和定义适用于本技术文件。



数据广播 datacast(data broadcast)

xxxx。

业务发现(Service Discovery)

业务发现是指终端获取一组可用的MBMS用户服务以及用户服务信息的方法。

业务发布(Service Announcement）

业务发布是指MBMS服务提供商向终端发布一组可用的MBMS用户服务以及关于用户服务的信息的方法。

仅接收模式(Receive Only Mode）

设备在ROM下，能够接收MBMS广播服务并采用特定的过程选择网络，但不能与网络发起任何信令过程，且不能采用3GPP TS 23.122定义的PLMN选择过程。参见 3GPP TS 23.246。

* 1. 缩略语

下列缩略语适用于本技术文件。

HTML 超文本标记语言（Hyper Text Markup Language）

SA业务发布（Service Announcement）

SACH业务发布频道（Service Announcement Channel）

MIME多用途互联网邮件扩展类型（Multipurpose Internet Mail Extensions）

ROM 仅接收模式（Receive-Only-Mode）

1. 系统概述（广科院）

本章主要描述以下内容：

（1）叙述业务特征，包括传输的内容、数据类型、传输方式、支持数据传输主要功能（统计上报、文件修复等）（可参照需求文档）

（2）实体位置，即此业务系统与5G广播整体系统架构的关联（图）；

（1）数据封装在5G广播架构中的位置

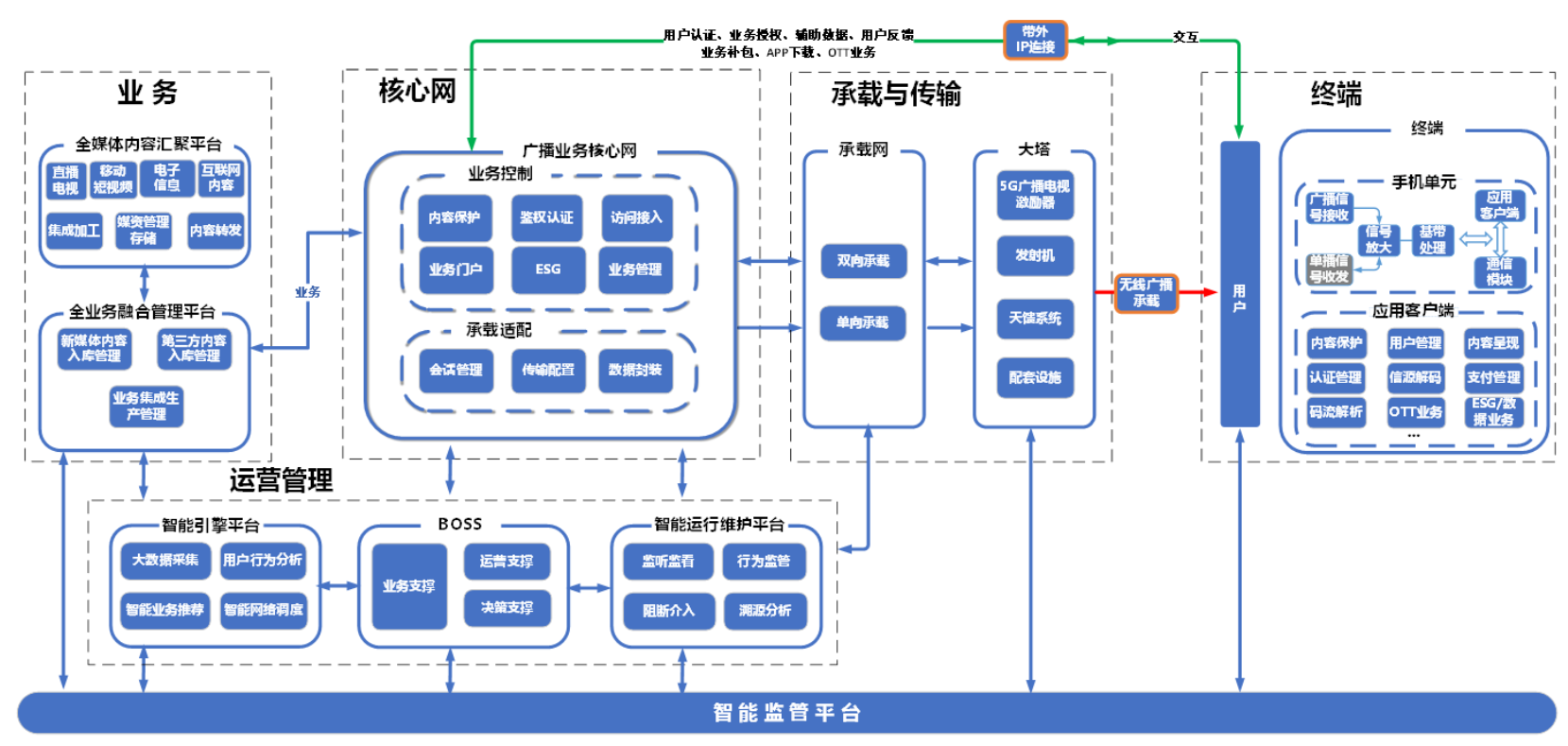


图4-1 5G广播系统架构

如上图所示，本标准所涉及部分为5G数据广播核心网承载适配层，主要功能是将前端业务数据进行封装，以适应多种类型信道的传输。

5G广播系列标准的本部分描述和约定5G广播业务的数据传输和封装方法。



图4-2 数据封装与传输层级图

本部分约定的技术指标主要包括传输模式、数据封装、更新方式、修复机制等几个方面。

5G多媒体广播业务的数据分发方式分为两种：一种为实时方式，一种为非实时方式。本部分约定的5G广播业务传输内容包括音视频、实时流、软件应用、ESG、文本、静态图像、二进制数据等基本类型。内容类型可分为传输实时流和文件等不同类型，同时采用实时方式和非实时方式。支持FLUTE/ROUTE协议和SMT协议，描述关联分发过程中的文件修复和统计上报数据，以及前向纠错编码。

（2）

此业务系统主要位于核心网与承载网，承接各类业务数据，进行打包封装，通过5G广播大塔、互联网和5G蜂窝网小塔对广播业务进行发送、补包传输和用户传输质量等信息的反馈。



图4-3 数据封装与传输示意图

1. 业务协议（信令）概述
   1. 协议栈（高通）

5G广播中协议栈（图）；



图x 5G广播业务协议栈框架

（该图将根据标准实际定义明确和细化）

本标准定义的5G广播业务协议栈包括广播传输协议和通过单播传输的辅助传输过程。广播协议栈包括SMT和ROUTE/FLUTE协议。

ROUTE/FLUTE协议请参见6.1.

SMT协议请参见6.2.

关联分发过程、文件修复和统计上报请参见7章。

FEC请参见第8章。

* 1. 业务发现/发布（Service Discovery/Announcement）（高研院）

（高通提供相关参考文档）

定义业务如何发现，如何区分多种协议。

本协议支持SMT、FLUTE和ROUTE协议，三种协议通过业务发布内“传输方法”字段区分。FLUTE和ROUTE协议是可以通过IETF定义的SDP（Session Description Protocol）协议标识区分。

业务发布的示例如下

Service announcement = [TMGI, SDP, MPD, initiate segment, 传输方法, ADP …]，其中传输方法 = SMT，FLUTE/ROUTE，和其他*.*

5.2.1简介Introduction

业务发现是指终端如何获取一组可用的MBMS用户服务以及用户服务信息的方法。业务发布是指MBMS服务提供商向终端发布一组可用的MBMS用户服务和用户服务包以及关于用户服务的信息的方法。

具体参见3GPP TS 26.346 5.2.1节。

5.2.2MBMS用户服务描述元数据片段

MBMS用户业务发现/发布的作用在于，在用户服务的会话启动之前或同时将服务信息告知终端。用户服务通过元数据（对象/文件）描述，元数据主要通过下载方式进行分发，也可以通过带外IP分发。MBMS用户业务发现/发布通过合适的方式向多个终端传送元数据片段（metadata fragments）。元数据描述了服务的细节，元数据片段是唯一可标识的元数据实体。

具体参见3GPP TS 26.346 5.2.2节。

对于SMT协议，MBMS用户服务描述元数据为MP（媒体呈现）表，具体参见GB/T 33475.6第8.3.3节。

5.2.3基于MBMS承载的用户业务发布

5.2.3.1概述 General

元数据封装文件和相关的元数据段作为同一个下载会话的文件对象传送。业务发布本身也被定义为一项用户服务，为了接收该项服务的客户端必须获取其相关的MBMS下载会话的会话参数。

对于FLUTE/ROUTE协议，下载会话的会话参数可通过以下几种方式获取

* 1. 通过预存在UE中获取
  2. 通过预存在MBMS应用中获取
  3. 通过OMA PUSH分发获取
  4. 通过从HTTP服务器下载获得

对于仅接收模式，仅支持方式1）和2）。具体参见3GPP TS 26.346的5.2.3.1节

对于SMT协议，承载MP表的下载会话参数获取方式与FLUTE/ROUTE协议相同。

5.2.3.1.1 ROM服务的业务发布

当采用FLUTE/ROUTE协议时，ROM服务通过业务发布（SA）用户服务描述，该SA用户服务的TMGI值在3GPP TS 24.116中定义的一段保留取值的范围之内。ROM的业务发布本身也被定义为一种特殊的服务，该服务被称为ROM业务发布信道，简称ROM SACH。（Receive-Only-ModeService AnnouncementCHannel)。当采用SMT协议时，ROM服务通过服务层信令（SLS）描述。ROM SACH和服务层信令通过不同的MBMS下载会话获取。

对于FLUTE协议，MBMS下载会话参数最小集为

* + IP多播地址 (IPv4 or IPv6)；
  + 目标UDP端口号；
  + 传输会话标识（TSI）；
  + 协议标识（Protocol\_ID）(i.e. FLUTE/UDP)。

具体参见3GPP TS 26.346的5.2.3.1.1节。

对于ROUTE协议，MBMS下载会话参数最小集为

* + 源IP地址 (IPv4 or IPv6)；
  + 目的IP地址；
  + 目标UDP端口号；
  + 传输会话标识（TSI）；
  + 协议标识（Protocol\_ID）(i.e. ROUTE/UDP)。

对于SMT协议，MBMS下载会话参数最小集为

* + IP多播地址 (IPv4 or IPv6)；
  + 目标UDP端口号；
  + 协议包标识（packet\_id）/类型（Type）；
  + 协议标识（Protocol\_ID）(i.e. SMT/UDP)。

5.2.3.1.2 非仅接收类服务的业务发布Service Announcement for non Receive-Only-Mode Services(5.2.3.1.2)

对于本规范定义的5G广播业务传输应用，非ROM为可选模式。非ROM的MBMS用户服务的描述方式与ROM的MBMS用户服务的相同。对于非ROM的MBMS终端，ROM SACH和服务层信令及相关会话参数的获取方法与ROM下的相同。

当采用FLUTE/ROUTE协议时，描述MBMS用户服务的SA服务的TMGI可以与运营商专用的MCC+MNC值相关联，或与3GPP TS 24.116中定义的保留值之一关联。参见3GPP TS 26.346 5.2.3.1.2节。

对于SMT协议，FFD。

5.2.3.2 元数据封装的传输Metadata Envelope Transport(5.2.3.2)

当采用FLUTE/ROUTE协议时，元数据封装对象作为一个对象，在同一个业务发布的MBMS下载会话中传送，并且作为该会话的一个元数据片段对象。参见3GPP TS 26.346 5.2.3.2节。

对于SMT协议，元数据封装的传输协议参见GB/T 33475.6 第7.3节。

5.2.3.3 FLUTE/ROUTE相关的元数据封装和元数据片段的传输Metadata Envelope and Metadata Fragment Association with FLUTE (5.2.3.3)

当采用FLUTE/ROUTE协议时，MBMS下载业务发布会话FDT实体为每个传输对象提供多个统一资源标识（URI）。元数据封装项的metadataURI字段应与该元数据片段文件的FDT实体中的元数据片段的URI相同。参见3GPP TS 26.346 5.2.3.2节。

对于SMT协议，元数据封装的传输协议参见GB/T 33475.6 第7.3节。

5.2.4交互式样用户业务发布User Service Announcement using Interactive Announcement Function(5.2.4)

当采用FLUTE/ROUTE协议时，在存在带外IP的条件下，用户服务描述（USD）还可以通过HTTP或其它交互式传输方式传送到UE。获得USD的URL的获取方法与MBMS下载会话的会话参数获取方法相同。具体参见3GPP TS 26.346 5.2.4节。

对于SMT协议，FFD。

5.2.5基于点对点推送承载的用户业务发布User Service Announcement over point-to-point push bearers.(5.2.5)

5.2.5.1概述General(5.2.5.1)

基于点对点推送承载的用户业务发布具有几个不同于基于MBMS承载的用户业务发布的特征。业务发布发送方提供的元数据封装不一定要发送到MBMS终端。在元数据封装和元数据片段都被传输的情况下，该业务发布方式存一个限制，即要么必须将元数据片段嵌入元数据封装中，要么必须通过元数据封装引用元数据段，并且元数据片段和元数据封装必须包含在一个multipart类型的MIME容器中。在这两种配置中，元数据封装和元数据段都作为文件对象在同一个下载会话中传输。

5.2.5.2支持的元数据与信令语法

对于FLUTE协议，支持的元数据语法参见3GPP TS 26.346 第11节。

对于SMT协议，支持的信令语法参见GB/T 33475.6 第8章。

5.2.5.3一致性控制和语法独立性Consistency Control and Syntax Independence(5.2.5.3)

对于FLUTE协议，一致性控制和语法独立性定义参见3GPP TS 26.346 第11.1节。

对于SMT协议，FDD。

5.2.5.4元数据封装定义Metadata Envelope Definition(5.2.5.4)

对于FLUTE协议，元数据封装定义参见3GPP TS 26.346 第11.1节。

对于SMT协议，元数据封装定义参见GB/T 33475.6 第6.4节。

5.2.5.5元数据封装文件分发Delivery of the Metadata Envelope(5.2.5.5)

对于FLUTE协议，元数据片段的实例应嵌入元数据封装中或与元数据封装一起包含在multipart类型的MIME容器中。元数据封装和元数据片段在同一传输会话的带内一起传送。元数据封装中包含对关联元数据片段的引用（metadataURI），该元数据片段采用与业务发布中标识的元数据片段文件相同的统一资源标识（URI）。具体参见3GPP TS 26.346 5.2.5.5节。

对于SMT协议，FFD。

5.2.5.6基于SMS承载的用户业务发布 User service announcement over SMSbearers(5.2.5.7)

对于FLUTE协议，基于SMS承载的用户业务发布的格式遵循OMA推送OTA规范。OTA-WSP应基于SMS承载，应用端口号遵循OMA推送OTA规范的定义，使用的应用ID是按照OMNA分配的0x9023。具体参见3GPP TS 26.346 5.2.5.6节。

对于SMT协议，FFD。

5.2.5.7基于HTTP 推送承载的用户业务发布 User service announcement over HTTP pushbearers(5.2.5.8)

对于FLUTE协议，基于HTTP push承载的用户业务发布的格式遵循OMA推送OTA规范。OTA-HTTP应基于HTTP承载。应用端口号遵循OMA推送OTA规范的定义，使用的应用ID是按照OMNA分配的0x9023。具体参见3GPP TS 26.346 5.2.5.7节。

对于SMT协议，FFD。

5.2.6汇聚业务发布文件的元数据片段封装Metadata Fragment Encapsulation to aggregate Service Announcement Documents(5.2.6)

元数据片段是一个唯一可识别的元数据块。通常，需要多于一个元数据片段以提供所有必要的参数以发起MBMS用户服务。典型情况下，元数据片段由多个独立的文档提供。每个元数据片段采用MIME类型标注。Multipart MIME可用于将元数据片段封装进一个汇聚业务发布文件。汇聚文件也许包含多个MBMS用户服务的多个元数据片段。汇聚业务发布文件应按照RFC 2557封装元数据片段。具体参见3GPP TS 26.346 5.2.6节。

对于SMT协议，FDD。

5.2.7 传输协议标识Transport protocol identification (7.3.2.13)

对于MBMS下载分发方式，不同传输协议的SDP的媒体描述相关字段设置为：

对于FLUTE协议，SDP的媒体描述("m=")行的<proto>字段应设置为'FLUTE/UDP'；

对于ROUTE协议，SDP的媒体描述("m=")行的<proto>字段应设置为'ROUTE/UDP'；

对于SMT协议，SDP的媒体描述("m=")行的<proto>字段应设置为'SMT/UDP'。

1. 协议

简要介绍。

* 1. FLUTE/ROUTE（高通、中兴）
     1. FLUTE

定义该协议的数据封装、传输协议和元数据/信令等要求(metadata/signaling）。

* + 1. ROUTE

定义该协议的数据封装、传输协议和元数据/信令等要求(metadata/signaling）。

* 1. SMT（上海交大）

定义该协议的数据封装、传输协议和元数据/信令等要求(metadata/signaling）。

本部分定义了智能媒体传输（Smart Media Transport, SMT），具体内容参见《信息技术 高效多媒体编码 第6部分：智能媒体传输》

SMT协议为面向包交换的应用层协议，旨在为异构网络中的媒体数据提供传输服务。异构网络即单向物理网络（如数字广播网络）和双向物理网络（如IP网络）组成的混合网络。媒体数据包括时序媒体数据（如视频，音频）、非时序媒体数据（如文本、图片）、以及用户反馈数据（如实时指令）。SMT协议基于IP协议，能动态地自适应于异构网络中的不同网络条件。

SMT协议从逻辑上可以分为三个逻辑功能区，分别为封装功能区、传送功能区和信令功能区，见图1。

封装功能区定义内容和服务的逻辑组织结构，实现了数据内容和数据描述的分离以及数据碎片化。以此形成了媒体内容的分布式布置，能够依靠数据描述形成的内容关联，完成服务的灵活组织和动态配置。信令功能区定义用于控制媒体内容消费和传送两方面的信息格式，能根据需求定制信令消息以实现动态配置服务、多种QoS类型、网络时钟的同步以及多屏设备之间的交互等，同时能够保证客户端实时反馈和超低延时控制。传送功能区定义异构网络下多媒体内容的传输，包括对多媒体数据包的封装与流式传送。传送功能区支持媒体数据的重要性分级、数据存储格式与传输格式的快速转换、数据内容的摘要和检索（如音视频指纹，检索特征向量等），同时加入应用层纠错保护机制，建立用于处理时延和抖动的接收端缓冲模型，以自适应于网络状态的变化。



1. SMT架构

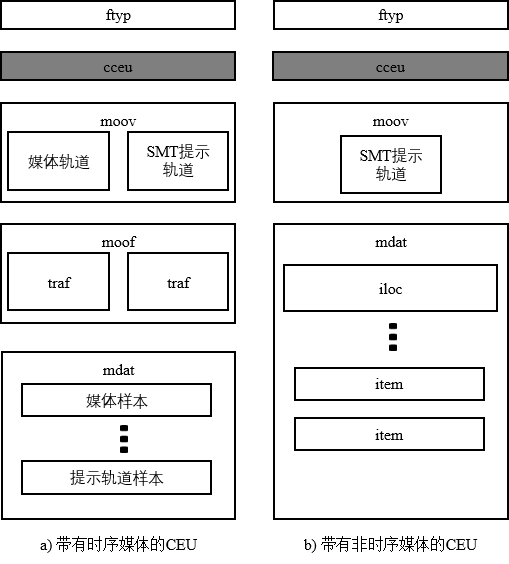
在传输过程中，SMT协议用信令消息保护数据模型。为满足SMT内容灵活组织的技术需求，SMT服务中内容的关联关系由描述文件指定，服务的改变不需要对数据流层级进行更改，只需要更新描述文件。同时为适配不同的网络环境，需要有单独的媒体传输控制信息。在异构网络中，传输控制信息能够根据网络环境的变化动态更新。分离信令消息与媒体数据，保证了内容自组织的动态灵活性，提供了更好的跨层优化。本章节将分别从数据封装、信令消息以及数据传输三个方面介绍SMT协议。



1. 数据模型
   * 1. 数据封装

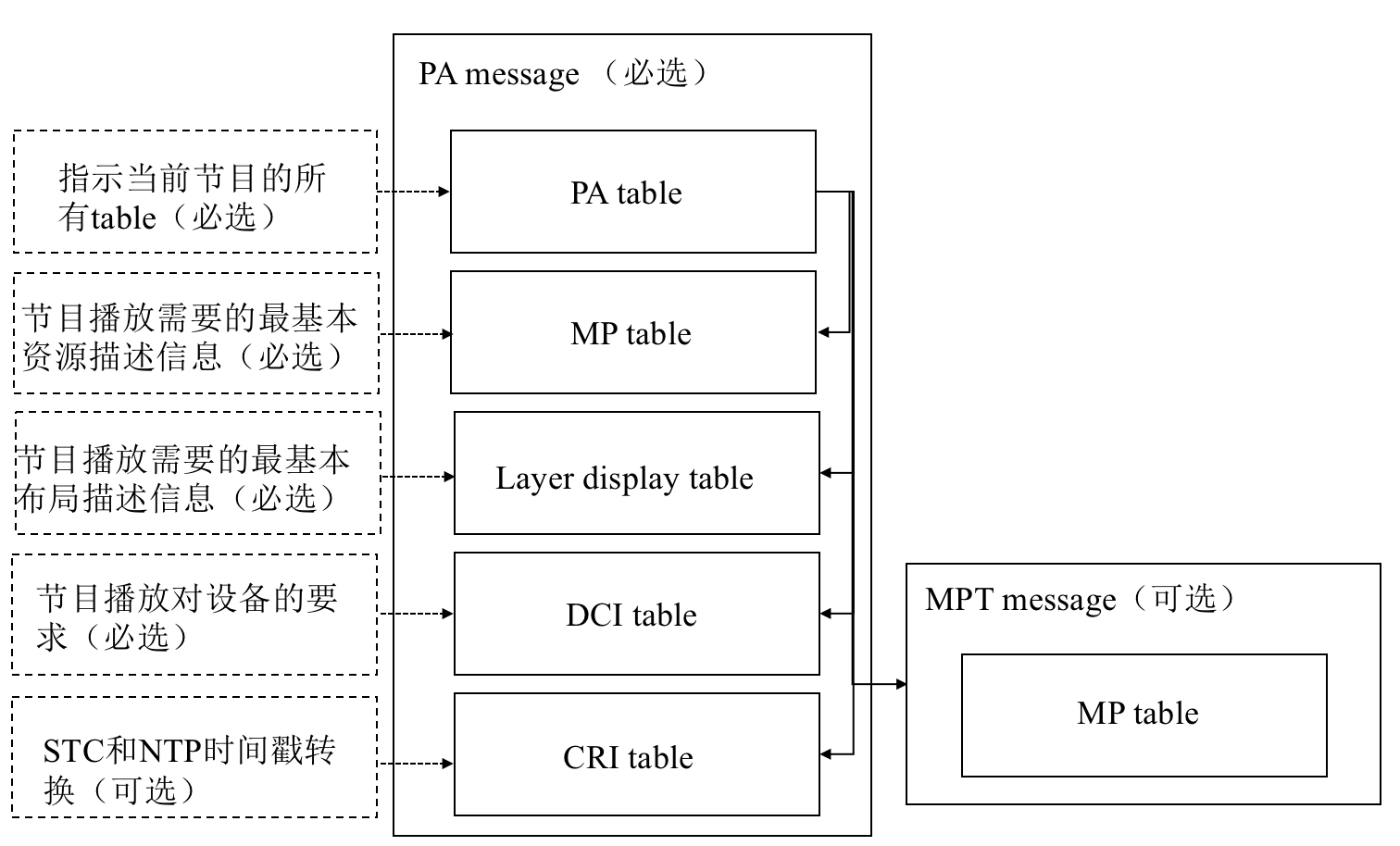
SMT协议在ISO BMFF文件封装格式的基础上，通过增加‘cceu’盒子，形成SMT的数据封装格式—通用封装单元（CEU）。该盒子提供Asset标识、CEU序列号以及相关信息，以明确地标识出封装进CEU文件中的媒体数据。‘moov’盒子包含所有编码器配置信息，以解码和呈现媒体数据。

时序媒体数据作为ISO BMFF的轨道存储，CEU中允许单媒体轨道。非时序媒体作为ISO BMFF的元数据的部分存储。图3描绘了两个SMT封装的例子，一个是时序媒体，另一个是非时序媒体。对于封包化的CEU传送，SMT提示轨道提供将封装的CEU转化成SMTP负载和SMTP包的信息。



1. CEU封装结构
   * 1. 信令消息

SMT信令功能区中包含了一整套消息格式传达用于传输和消费SMT数据包的必要信令消息，分别称为传输信令和消费信令。接收端在接入系统时会首先收到PA消息，通过PA消息解析PA表以后可以得到其他信令表的信息。如图4所示，一个PA消息包含一个PA表，PA表中指示了构成该节目的所有表。一个PA消息中应包含MP表，一个完整的MP表具有关于包含所有媒体资源的媒体数据包的信息，如媒体数据的位置信息、呈现时间以及媒体数据之间的关系信息等。同时，一个PA消息还应包含媒体呈现层信令表（Layer display table），采用分层的方式指示媒体数据呈现的基本布局。DCI表用来指示节目播放对设备的要求，而CRI表用以指示STC和NTP时间戳的转换关系，用以同步异构网络下的媒体资源。其中，PA表、MP表、Layer display表以及DCI表是节目正常播放所必须的，而CRI表是可选的。



1. 信令消息组成
   * 1. 数据传输

SMT传输模型如图5所示，SMT信令文件和传输包均根据媒体服务逻辑包进行组织，媒体组织和传输都与媒体服务相关，有利于根据不同服务类型差异化实现媒体内容组织和传输的自适应性。SMT逻辑包可以序列化为SMT文件，支持媒体文件式的存储、传输和下载；也可以打包为SMT传输包，以支持媒体的流化传输。针对数据与信令的流化传输，SMT传输协议（SMTP）分别设计了CEU模式和信令模式下的封包格式，在SMTP包type字段分别定义两种模式，如表1所示。



1. 传输模型

表1 数据类型与数据单元定义

| **值** | **数据类型** | **数据单元定义** |
| --- | --- | --- |
| 0x00 | CEU | 媒体感知的CEU片段 |
| 0x01 | 信令消息 | 一条或多条信令消息或信令消息的片段 |
| 0x02~0x03 | 为其它数据类型保留 |  |
| 0x04 ~ 0x3F | 私有用途保留 |  |

6.2.3.1 CEU模式

通过SMT协议传输CEU要求在SMT发送与接收实体分别配置封包与解包程序。封包程序将CEU封包成一组被SMTP包携带的SMTP负载。SMTP负载格式支持SMTP负载分段传输，以使大容量负载可以传输。SMTP负载格式也支持将多个SMTP负载的数据单元整合到一个SMTP负载中，以便于小容量数据单元聚合传输。接收实体解包以恢复原始CEU数据。图6和图7分别描述了CEU与时序媒体负载和非时序媒体负载的关系。



1. 时序媒体的有效负载产生



1. 非时序媒体有效负载的产生

6.3.2.2 信令模式

SMTP信令消息模式用于定义信令消息的传输。信令消息可用其它格式编码，如二进制格式或XML格式。因此在传输层能够快速访问与过滤信令消息很重要，且在过滤时希望尽量避免解析信令消息。

信令消息负载格式提供分块与整合功能以支持高效封包。信令消息模式下SMTP负载头部结构见图8。



1. 信令消息模式下SMTP负载头部结构
2. 关联分发过程(Associated delivery procedures)
   1. 概述

简单描述本节内容。

* 1. 文件修复（高通）

描述文件修复的方式和方法以及对应所需的数据接口。

文件修复功能请参照3GPP TS26.346第9.3节

* 1. 统计上报（高通）

描述统计上报的功能和方式以及对应所需的数据接口。

统计上报功能请参照3GPP TS26.346第9.4节

1. FEC

FEC算法、编解码结构及应用等。

（如有发布的标准，可以直接引用，没有标准则在此章节或附录中具体说明）

* 1. Raptor/RaptorQ（高通）

适用于Raptor编码[RFC 5053]和RaptorQ编码[RFC 6330]。

* 1. 基于Raptor的算法（上海交大）

（适用于SMT）

* + 1. FEC编码算法

如表1-1所示，表示应用于SMT中的前向纠错编码算法。

**表1 FEC编码码字算法**

|  |  |
| --- | --- |
| 码字编号 | FEC编码算法 |
| 0～6 | 保留 |
| 7 | 自适应前向纠错编码 |
| 7～255 | 保留 |

* + 1. 自适应前向纠错编码码字

8.2.2.1概述

本节规定了自适应前向纠错编码码字。自适应前向纠错编码码字在标准RaptorQ码字上进行了扩展，可以实现在单层中的自适应FEC保护。

根据DU头中定义的不同优先级，可以将源符号划分为不同的类别。 所有分类的符号都可以由一个FEC编码矩阵保护，并在一个流中生成源符号和修复符号。

8.2.2.2编码方法

步骤1：根据DU头中定义的不同优先级，每个源符号被划分到不同编码符号块中，每个编码符号块中源符号的数目为D1，D2，D3，...，Dr。 这样，D1个源符号形成第一类，接下来的D2个源符号形成第二类，以次类推。并且，类的重要性随类索引而降低。

步骤2：根据源符号的分类，可以设计相应的编码矩阵A来产生中间码字。图1为2类优先级下的编码矩阵A的结构。



图1 编码矩阵结构

其中G\_p矩阵由’G\_LDPC,1’、’I\_S’、’G\_LDPC,2’、’G\_HDPC’和’I\_H’组成，G\_ENC1和G\_ENC2是LT矩阵。根据该编码矩阵A可以编码得到中间码字。

对于优先级数量超过2个的情况，其编码矩阵A和生成中间码字如下所示：



步骤3：采用扩展窗的方式使得优先级高的源符号所对应的中间码字具有更高的恢复概率，从而提高了优先级高的源符号的恢复概率。其编码过程如下所示。 当信道条件较差时，可以减少*G\_ENC12*和*G\_ENC21*的行数，增加*G\_ENC11*的行数，以增加更多重要中间码字的冗余度并保证优先级高的源符号的恢复。

其中，*R1*是由中间码字*C1*编码产生的恢复符号，*R2*是由中间码字*C1*和*C2*编码产生的恢复符号。

若共有l类源符号，则恢复符号可以如下产生：



8.2.2.3解码方法

解码过程类似于一组接收到的编码符号的编码过程。

步骤1：由于某些数据包可能在传输过程中丢失，因此需要根据ISI确定数据丢失情况生成编码矩阵A。编码矩阵A的行应基于ISI删除。用于解码的编码矩阵A如下。可以从源符号和编码矩阵A逆矩阵生成中间码字。

矩阵A可以进行如下变换：

由于中间符号*C1*与恢复符号*R1*和*R2*有关，因此具有更高的恢复概率。接收到编码后的符号，可以根据不同情况恢复得到中间码字。

情况1：接收到的编码符号总数小于源数据生成的编码符号的个数，并且的行数小于列数时；*C1*、*C2*均无法解出；

情况2：接收到的编码符号总数小于源数据生成的编码符号的个数，且的行数大于等于列数时，*C1*可以成功解码，但*C2*不能成功解码。

情况3：接收到的编码符号总数大于源数据生成的编码符号的个数，的行数大于等于列数，的行数大于等于列数，*C1*和*C2*可以使用IETF RFC rfc6330定义的方法解出。

情况4：接收到的编码符号总数大于等于源数据生成的编码符号的个数，的行数大于等于列数，的行数小于列数，*C1*可以成功解码，但*C2*无法成功解码。

情况5：接收到的编码符号总数大于等于源数据生成的编码符号的个数，且的行数小于列数，的行数大于等于列数时，使用高斯消元法直接对矩阵进行求解。

步骤2：得到中间码字后，源符号可以根据中间码字译码得到。

对于优先级多于2个的情况，同样可以根据中间码字的恢复情况得到源符号。



步骤3：得到源符号后，数据包可以根据packet\_sequence\_number恢复原始顺序。

* 1. RaptorG（广科院）
     1. 参数语法定义

具体实施方式中用到的基本符号及基本函数如表8.1所示。

表8.1. 基本符号及基本函数

|  |  |
| --- | --- |
| 定义 | 说明 |
| C[0],...,C[L-1] | 中间符号 |
| S | G\_LDPC行数，也是冗余符号个数 |
| K | 源符号个数 |
| L | 中间符号个数，L=K+S |
| CE | G\_LDPC每列非零元素个数 |
| base | GF域元素个数，本发明为256 |
| bitxor(x,y) | x和y的比特异或 |
| mod(x,y) | x对y取余 |
| floor(x) | x向下取整 |
| x/y | x除以y |
| G\_LDPC(x,y) | G\_LDPC x行y列值 |
| x\*y | x乘以y |
| L’ | 大于等于L的最小质数 |
| Q | 小于2的16次方最小质数（65521） |
| SI | systemtic index 系统索引，随K值不同而不同 |
| x^^y | x的y次幂 |
| ESI | 编码符号ID |
| S\_Max | 源符号和冗余符号的个数和 |
| A | RaptorG的构成矩阵 |
| Min(x,y) | x和y最小值 |

* + 1. 编码矩阵生成

具体的编码矩阵生成包括以下步骤：

步骤1、构造G\_LDPC矩阵，首先初始化G\_LDPC矩阵为全零矩阵，然后按照算法每一列赋予CE个非零元素。

步骤1.1、初始化G\_LDPC矩阵，其中行数为S，列数为L，整个矩阵为零矩阵；

步骤1.2、初始化变量i为0，其后每次转入此步骤i=i+1，当i=L时算法结束，G\_LDPC构造完成，步骤S1结束；

步骤1.3、设变量a，其值为1+mod(floor(i/S),(S-1))；

步骤1.4、设变量b，其值为mod(i,S)；

步骤1.5、计算G\_LDPC矩阵中横坐标为b+1，纵坐标为i+1的符号值，其值等于bitxor(G\_LDPC(b+1,i+1),mod((b+1)\*(i+b+1)+i\*i,base-1)+1)；

步骤1.6、若由1.5转入此步骤，则初始化变量j为1；若由1.8转入此步骤，则j=j+1，当j的值为CE时，转步骤1.2；

步骤1.7、设置b值为mod(b+a,S)；

步骤1.8、计算G\_LDPC矩阵中横坐标为b+1，纵坐标为i+1的符号值，其值等于bitxor(G\_LDPC(b+1,i+1),mod((b+1)\*(i+b+1)+i\*i,base-1)+1)，转步骤1.6；

步骤2、构造G\_LT度生成函数Deg函数，函数的输入参数为v，输出为度d。此函数与Raptor10的构造方法相同。

步骤2.1、定义度分布表，通过索引j的值查表得到d[j]，如表8.2所示。

表8.2. G\_LT度分布表

|  |  |  |
| --- | --- | --- |
| 索引j | F[j] | d[j] |
| 0 | 0 | -- |
| 1 | 10241 | 1 |
| 2 | 491582 | 2 |
| 3 | 712794 | 3 |
| 4 | 831695 | 4 |
| 5 | 948446 | 10 |
| 6 | 1032189 | 11 |
| 7 | 1048576 | 40 |

步骤2.2、初始化变量j为0，其后每次转入此步骤j=j+1；

步骤2.3、通过查表2判断，如果v<f(j),令d=d(j)，函数返回d,步骤2结束。否则转步骤2.2；

步骤3、构造随机数生成函数Rand函数，函数的输入参数为一个三元组X，i和m，其中X和i是一个非负整数，m为一个正整数，输出为生成值。

步骤3.1、设变量InputV0的值为V0(mod(X+i,256)+1)。

步骤3.2、设变量InputV1的值为V1(mod(floor(X/256)+i,256)+1)。

步骤3.3、计算mod（bitxor(InputV0, InputV1),m）的值，得到Rand函数的返回值，步骤3结束。

步骤4、构造生成三元组Triple函数，函数的输入参数为一个二元组SI和ESI的值，输出为一个三维向量d，a和b。

步骤4.1、设变量A的值为mod(53591+SI\*997,Q)；

步骤4.2、设变量B的值为mod(10267\*(SI+1),Q)；

步骤4.3、设变量Y的值为mod(B+ESI\*A,Q);

步骤4.3、设变量v的值为Rand(Y,0,2^^20)，这里的Rand为步骤3的生成流程;

步骤4.5、设变量d的值为Deg(v);

步骤4.6、设变量a的值为1+Rand(Y,1,L’-1);

步骤4.7、设变量b的值为Rand(Y,2,L’);

步骤4.8、返回三元组(d,a,b);

步骤5、构造G\_LT矩阵，此步骤参照Raptor10，具体实现方法为：

步骤5.1、构造G\_LT矩阵，其中行数为K，列数为L，初始化矩阵为全零矩阵；

步骤5.2、构造d\_Triple矩阵，其中行数为1，列数为S\_Max，初始化矩阵为全零矩阵；

步骤5.3、构造a\_Triple矩阵，其中行数为1，列数为S\_Max，初始化矩阵为全零矩阵；

步骤5.4、构造b\_Triple矩阵，其中行数为1，列数为S\_Max，初始化矩阵为全零矩阵；

步骤5.5、初始化变量ii为1，其后每次转入此步骤ii=ii+1，当ii=S\_Max时，转步骤5.7；

步骤5.6、执行函数Triple（SI,ii-1），返回的三元组分别赋值给d\_Triple(ii)，a\_Triple(ii)和b\_Triple(ii)并转步骤5.5；

步骤5.7、初始化变量ESI为1，其后每次转入此步骤ESI=ESI+1，当ESI=K+1时步骤3结束；

步骤5.8、设变量d\_i，d\_i的值为d\_Triple(ESI);

步骤5.9、设变量a\_i，a\_i的值为a\_Triple(ESI);

步骤5.10、设变量b\_i，b\_i的值为b\_Triple(ESI);

步骤5.11、当b\_i大于等于L时，b\_i的值重设为mod(b\_i+a\_i,L),直至b\_i≤L;

步骤5.12、赋值G\_LT(ESI,b\_i+1)为1；

步骤5.13、若由5.12转入此步骤，则初始化变量j为1；若由5.16转入此步骤，则j=j+1，当j= min(d\_i-1, L-1)+1时，转步骤5.7；

步骤5.14、赋值b\_i为mod(b\_i+a\_i,L)；

步骤5.15、当b\_i大于等于L时，b\_i的值重设为mod(b\_i+a\_i,L) ,直至b\_i≤L;

步骤5.16、赋值G\_LT(ESI,b\_i+1)为bitxor(G\_LT(ESI,b\_i+1),1)；

1. 媒体编解码和格式（高通、广科院）

音频编码：aacPlus, Enhanced aacPlus,AMR narrow-band/wide-band speech, AC-3

视频编码：H.264(AVC), H.265 (HEVC)

封装编码：CMAF, [MP4](http://zh.wikipedia.org/wiki/MPEG-4),

**请补充或修改以上章节及小标题，并分别简述每个章节或小标题下要描述的内容。**

**以下为附录内容，按同上要求补充或修改。**

1. （规范性附录）  
   FEC编码规范
   1. 概述
   2. FEC信令消息

A.2.1 语法

表1 FEC信令

|  |  |  |  |
| --- | --- | --- | --- |
| **语法** | **值** | **比特位数** | **备注** |
| AL\_FEC ( ){  message\_id  version  length  message\_payload{  fec\_flag  private\_fec\_flag  reserved  if (fec\_flag==1) {  length\_of\_fec\_flow\_descriptor  fec\_flow\_descriptor() {  number\_of\_fec\_flows  for ( i=0; i<N1 ; i++) {  fec\_flow\_id  source\_flow\_id  number\_of\_assets  for ( j=0; j<N2 ;j++) {  packet\_id  }  fec\_coding\_structure  ssbg\_mode  reserved  length\_of\_repair\_symbol  if (ssbg\_mode == 2) {  num\_of\_repair\_symbol\_per\_packet  num\_of\_symbol\_element\_per\_source\_symbol  }  if (fec\_coding\_structure == 0001) {  number\_of\_class  if (private\_fec\_flag == 1) {  private\_flag  private\_field\_length  private\_field  }  repair\_flow\_id  fec\_code\_id\_for\_repair\_flow  maximum\_k\_for\_repair\_flow  maximum\_p\_for\_repair\_flow  protection\_window\_time  protection\_window\_size  }  }  }  }  }  } | ‘111111’  N1  N2  ‘1’  N3  N4 | 16  8  16  1  1  6  16  8  8  8  8  16  4  2  1  16  16  16  8  1  7  N4\*8  8  8  24  24  32  32 | uimsbf  uimsbf  uimsbf  bslbf  bslbf  bslbf  uimsbf  uimsbf  uimsbf  uimsbf  uimsbf  uimsbf  uimsbf  uimsbf  bslbf  uimsbf  uimbsf  uimsbf  uimsbf  bslbf  bslbf  uimsbf  uimbsf  uimbsf  uimbsf  uimbsf  uimbsf  uimbsf |

A.2.2 语义

message\_id：表示FEC信令的标识符，见表 20信息ID值。

version：表示FEC信令的版本。SMT接收实体利用此字段废弃旧版本，采纳新版本的配置。。

length：表示FEC信令的长度。这个字段的长度是16位。以字节表示FEC消息的长度，从下一字段的第一字节到本信令的最后字节，不能为0。

fec\_flag：表示是否有至少一个源码流需要进行FEC保护。如果每个源码流均不需FEC保护，则本字段后的所有字节均不出现。

表2 fec\_flag的值

|  |  |
| --- | --- |
| **值** | **描述** |
| 0 | 没有FEC源码流 |
| 1 | 至少有一个FEC源码流 |

private\_fec\_flag：标识专有FEC配置信息是否定义。若该字段为0， 则没有相应字段；若为1，则存在私有FEC相关字段。

length\_of\_fec\_flow\_descriptor：fec\_flow\_descriptor 所占字节数。

number\_of\_fec\_flows：FEC编码流的数量。

fec\_flow\_id ：用于识别FEC编码流的任意整数。每个FEC编码流由有关联的一个FEC的源码流和一个或多个FEC恢复码流组成。

source\_flow\_id：用于识别FEC源码流的任意整数。每个FEC源码流与一个FEC编码流相关。

number\_of\_assets：属于同一个FEC源码流的Asset的数量。

packet\_id：SMTP中包头的packet\_id。

ssbg\_mode：表示已提供的SSBG模式，该字段的值表示的含义见表3。

表3 ssbg\_mode的值

|  |  |
| --- | --- |
| **值** | **描述** |
| b00 | ssbg\_mode0 (SMTP包大小固定) |
| b01 | ssbg\_mode1 (SMTP包大小可变) |
| b10 | ssbg\_mode2 (SMTP包大小可变) |
| b11 | 保留 |

length\_of\_repair\_symbol：以字节表示恢复符号的长度。

num\_of\_repair\_symbol\_per\_packet：在使用ssbg\_mode2模式时，一个FEC恢复数据包中所携带的恢复符号的数量。

num\_of\_symbol\_element\_per\_source\_symbol：在ssbg\_mode2模式下，一个源符号所包含的源符号项的数量。

fec\_coding\_structure：表示对FEC源码流采用的FEC编码结构，该字段的值表示的含义见表4。

表4 fec\_coding\_structure的值

|  |  |
| --- | --- |
| **值** | **描述** |
| b0000 | 不采用AL-FEC编码模式 |
| b0001 | 应用自适应FEC编码模式 |
| b0010 ~ b1111 | 保留 |

number\_of\_class：用于指示划分优先级的数目。

repair\_flow\_id：用于识别FEC恢复码流的任意整数。它表示FEC恢复数据包的SMTP包头中packet\_id。恢复码流的packet\_id表示为repair\_flow\_id。

fec\_code\_id\_for\_repair\_flow：FEC恢复码流的FEC编码标识符。

private\_flag：当设置为“1”时，表示存在专有FEC的描述信息，比如特定FEC算法的参数等。当设置为“0”时，表示没有这样的专有字段。

表5private\_flag的值

|  |  |
| --- | --- |
| **值** | **描述** |
| b0 | 指定的FEC架构没有专有字段 |
| b1 | 指定的FEC架构含有专有字段 |

private\_field\_length：以字节表示专有字段的长度。若private\_flag等于0，则该值应置0表示没有专有字段。

private\_field：该字段包含特定FEC算法参数的专有FEC信息。

maximum\_k\_for\_repair\_flow：表明与FEC恢复码流相关的源符号块中所允许的最大源符号数量。

maximum\_p\_for\_repair\_flow：表明与FEC恢复码流相关的恢复符号块中所允许的最大恢复符号数量。

protection\_window\_time：该字段以毫秒表示从第一个FEC源或恢复数据包块到最后一个FEC源或恢复数据包块的最大持续时间。若该值置0，将会使用“protection\_window\_size”。

protection\_window\_size：该字段表示该FEC编码流（一个FEC源码流和对应的一个或多个FEC恢复码流）的编码块（源符号块和恢复符号块）的最大字节数。若该值置0，则将会使用“protection\_window\_time”。

* 1. 编码符号块格式

A.3.1概述

前向纠错的编码块由源符号块和恢复符号块组成。源符号块根据给定的SSBG（source symbol block generation）模型，从源数据包中生成。相关的源符号块通过特定的FEC编码方式编码产生恢复符号块。编码符号块格式见图19。



图19 编码符号块格式

A.3.2源数据包组

为FEC要保护的数据。一个FEC源码流被划分为1个或多个源数据包组。一个源数据包组由指定数量的FEC字段值为1的SMTP包组成。源数据包组被转化为源符号组。

A.3.3源符号块格式

一个源符号块由指定数量的源符号组成，源符号由源数据包生成。目前有三种生成方式（SSBG）：ssbg\_mode0、 ssbg\_mode1 和 ssbg\_mode2。ssbg\_mode0适用于固定大小的SMTP包，而ssbg\_mode1和ssbg\_mode2适用于可变大小的SMTP包。

采用ssbg\_mode0模式时，因所有SMTP包大小相同，源符号块和源数据包组完全相同。此时源数据组中SMTP包的数量与源符号块中源符号数量一致，且对应编号相同，见图20。



图20 不填充生成源符号块(ssbg\_mode0)

ssbg\_mode1模式生成源符号块时，与ssbg\_mode0基本一致，源符号长度相同，但因源SMTP包大小不同，因此需要添加填充字节，此时每个源符号的开始2字节为SMTP包的大小（为网络字节序）， 接着是SMTP包的数据，之后为全0的填充字节，见图21。



图21 填充生成源符号块（ssbg\_mode1）

ssbg\_mode2模式时，源符号块仍然由源数据包块进行字节填充生成，但填充方式与ssbg\_mode1不同。ssbg\_mode1时，源符号的大小必须大于最长的SMTP数据包的长度，且一个源SMTP数据包与一个源符号一一对应。ssbg\_mode2时源符号的大小则可以小于SMTP数据包的长度，此时一个源SMTP数据包可能对应多个源符号。单个源数据包块由KSS 个源符号组成，为了减少字节填充数目，源符号进一步划分为N个等长的符号项（长度为T’）。因此源符号的长度（T）与符号项的关系为： T= T’ x N；一个源符号块中包含N\*Kss个符号项。处理源数据包组的第一个SMTP数据包时，将其长度以网络字节序存放在第一个源符号的第一个符号项的前2字节中，然后存放SMTP数据包的数据（可能占据多个源符号的多个符号项），若所占的最后一个符号项没有填满，则以0填充该符号项的剩余空间；其他数据包以此类推。最后如果所占空间不足源符号长度的整数倍（也即占用的符号项的个数si， si %N != 0），则以0填充剩余空间。

图22中介绍了一个形成源符号块的例子。该例中，长度分别为34，30，56，40，48的SMTP包生成一个源符号块，其由8个源符号组成。每个源符号长度为32字节，其中含有2个符号项（T＝32，N＝2），每个符号项长度为16字节。



图22 带有填充的源符号块示例（ssbg\_mode2）

* 1. FEC的源数据包和恢复数据包格式

A.4.1概述

下面将分别介绍传输SMTP包时的FEC的源数据包格式和传输恢复符号时的恢复数据包包格式，二者均为符合7.3.2.2 SMTP包结构的SMTP包。

在原始SMT数据包上添加source FEC payload ID，就形成了FEC源数据包，且需将数据包的FEC字段设为1，FEC恢复包包含一个repair FEC payload ID，和恢复符号块的一个或者多个恢复符号，需将数据包的FEC字段设为2。源数据包和恢复数据包实际上都是SMTP包。FEC源数据包的source FEC payload ID标识FEC源数据包携带的源符号和符号项，FEC恢复数据包的repair FEC payload ID标识FEC恢复数据包携带的恢复符号和相应的源数据包组。

A.4.2 FEC源数据包格式与负载ID

FEC源数据包是为了传输受到FEC保护的SMTP包而生成的包，它由SMTP包头，SMTP负载头，负载数据和source FEC payload ID组成。

FEC源数据包的格式见图23，其与SMTP包结构一致。



图23 FEC源数据包格式

图24规定了source FEC payload ID的格式。



图24 source FEC payload ID格式

SS\_ID （32位），该序列号在FEC源数据包内识别源符号。初始值从任意值开始增加，达到最大值后归0，重新开始递增。如果ssbg\_mode == 0或者ssbg\_mode == 1，SS\_ID每个FEC源数据包加1。如果ssbg\_mode == 2，SS\_ID每个符号项加1（包括最后一个源符号后的填充的符号项），且应被设置成第一个符号项的SS\_ID。

FFSRP\_TS（32位），指示值，FEC源码块或者恢复码块中发送的第一个包的SMTP包头中时间戳。

A.4.3 FEC恢复数据包格式与负载ID

FEC恢复数据包是用来传输一个或者多个恢复符号的包，它可以恢复相关的源码块。它由SMTP包头，repair FEC payload ID和一个或者多个恢复码组成。

FEC恢复数据包的格式见图7。其中SMTP packet header与图7和图8中定义的SMTP包结构一致，为了方便快捷地获得repair FEC payload ID，将其放置在恢复符号的前面。对于ssbg\_mode0, ssbg\_mode1两种模式，FEC恢复数据包应带有一个恢复符号，对于ssbg\_mode2，携带一个或者多个恢复符号。此外，一个FEC源码块或者恢复码块中的所有恢复数据包中（除去最后一个），应含有相同数量的连续恢复符号。



图25 FEC恢复数据包格式

repair FEC payload ID的定义见图26。

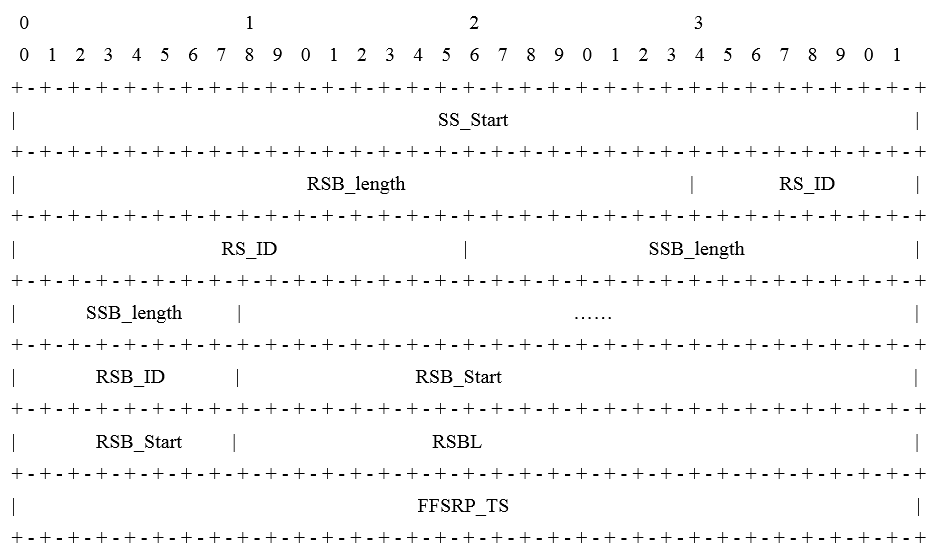


图26 repair FEC payload ID格式

SS\_Start（32位），指出源符号块的边界。若ssbg\_mode == 0或ssbg\_mode == 1，它被置为相关的源符号块中第一个源符号的SS\_ID。若ssbg\_mode== 2，它被置为相关源符号块中第一个符号项的SS\_ID。

RSB\_length（24位），相关恢复符号块中恢复符号的数量。

RS\_ID（24位），一个用来标示FEC恢复数据包中第一个恢复符号的整数。从0开始，对相应的恢复数据包里每一个恢复符号递增1。

SSB\_length[N]（N\*24位）， 如果fec\_coding\_structure=0001，N与划分优先级的个数相关。如果ssbg\_mode == 0或ssbg\_mode == 1, 表示源符号块中划分每个优先级下的子源符号块中源符号的数量。如果ssbg\_mode == 2，表示源符号块中符号项的数量。

RSB\_ID（8位），指示每个优先级下恢复符号块的序列号。

RSB\_start（24位），指示第i个优先级下恢复符号块的第一个恢复符号的RS\_ID。

RSBL（24位），指示第i个优先级下恢复符号的数目。

FFSRP\_TS（32位），指示值，FEC源码块或者恢复码块中发送的第一个包的SMTP包头中时间戳。

1. （资料性附录）  
   终端业务接收流程
   1. 基于FLUTE协议的MBMS用户业务发现/发布Profile1a MBMS User Service Discovery / Announcement Profile 1a(L.2)
      1. 简介 Introduction(L.2.1)

业务发现功能是允许UE找到5G广播网络定义的可用MBMS用户服务以及相关的服务接入信息（如FLUTE会话参数，TMGI，文件修复服务等；SMT的MP表）。此外，UE需要其它的服务接入信息以发起接收特定MBMS用户服务，并找到与空口上与该MBMB用户服务相关的数据。

在此Profile的一个典型部署场景下，MBMS元数据片段在MBMS会话之前发布给UE。UE按一定的周期监测SACH信道，并为潜在后续的应用存储接收的元数据，以便在大多数情况下，当UE激活MBMS用户服务接收时，已具备所有相关的接入信息。还有些场景，有必要在MBMS文件下载分发完成过程中，或有时之后，更新元数据片段（fragment）。

本小节规定了基于MBMS承载服务的业务发现/发布的一种Profile，该Profile的目标是限制可选特性的数量，并定义提供的元数据片段集。

具体参见3GPP TS 26.346 L.2节。

* + 1. 业务发布频道(SACH)的定义 Definition of a Service Announcement Channel (SACH)(L.2.2)

MBMS用户业务发现/发布需要以适当的方式向多个UE传送元数据片段。对一个服务的访问，需要多个元数据段来描述。

业务发布频道（SACH）是一种特殊类型的MBMS用户服务（即一种业务发布用户服务），只提供发布的服务的元数据片段。MBMS终端可以在SACH上找到针对每个服务的相关元数据段。MBMS终端一旦激活了某个MBMS用户服务的接收，就可以通过该MBMS用户服务数据找到附加的元数据段或更新的元数据段。从系统的角度来看，创建和管理SACH的方式本质上和其他MBMS分发会话相同。从终端的角度来看，SACH的MBMS承载类似于其他MBMS承载。与其他MBMS用户服务类似，SACH也通过元数据段进行描述。获取描述SACH的元数据段的过程称为SACH引导过程(bootstrap)。

SACH通过单独的MBMS下载推送会话传送。SACH和相关MBMS承载在业务发布的生存期内始终被激活。所有潜在的MBMS服务区域都会提供SACH。

对于已经编排或正在进行的每个服务，其元数据片段被合并为一个或多个业务发布文件（SA文件）。SA文件在SACH上连续的不间断的重复传送，而每个SA文件由唯一的URL标识。为了保持元数据片段始终是最新的版本（造成元数据版本更新的原因，包括已排定服务的更新，或有新加入的服务），终端需要定期检查SACH。SACH的总体目标是确保尽可能多的客户端能够在服务的MBMS会话之前接收到其业务发布信息。

具体参见3GPP TS 26.346 L.2.2节。

* + 1. 业务发布文件结构(L.2.3)

业务发布元数据应通过MBMS下载分发传输。SA文件应作为一个汇聚Multipart MIME文件安排格式。一个MBMS用户服务的所有元数据片段应包含在同一个SA文件中。SA文件或许包含多于一个MBMS用户服务的多个元数据片段。一个SA文件应由其URL唯一的标识，并作为FDT实体的Content-Location字段取值提供。SA文件应仅包含一个元数据封装，并且对于每个服务，SA文件应包含至少一个USBD（具有一个userServiceDescription元素），一个会话描述和一个编排描述片段。

具体参见3GPP TS 26.346 L.2.3节。

* + 1. MBMS用户业务发现/发布Profile1b MBMS User Service Discovery/Announcement Profile 1b(L.3)

MBMS用户服务发现Profile 1b除了遵循Profile 1a的规定之外，还遵循以下限制：

1. SA元数据片段应作为一个SA文件分发；
2. 在DASH直播服务场景下，SA文件应包含；
   * MBMS用户服务的媒体呈现描述（MPD）片段（fragment）；
   * 所有MBMS用户服务必要的初始化分段描述（Initialization Segment Description ISD）片段；
3. 在HLS直播服务场景下，SA文件应包含
   * 应用服务描述(ASD)片段；
   * 所有MBMS用户服务的其内容为媒体初始化分区（Media Initialization Section）的ISD片段；
   * 在混合DASH/HLS直播服务场景下，SA文件应包含上述MPD和ASD片段，以及相应的ISD片段；
4. 每个带内片段应包含在一个元数据封包（Envelope）中，每个元数据封包应包含一个完整的元数据片段。

具体参见3GPP TS 26.346 L.3节。

* + 1. MBMS用户业务发现/发布Profile 1c(L.3A)

MBMS用户服务发现Profile 1c除了遵循Profile 1b的规定之外，还遵循以下限制：

1. 多用户服务场景下，用户服务通过SA信息描述，并且用户服务的所有内容成分（content components）应通过单一MBMS承载分发；
2. 不使用SACH。包含元数据片段的SA文件不通过MBMS下载分发会话传输，而是由内容提供商自己选择一种方法方法方法给UE。

具体参见3GPP TS 26.346 L.3A节。

* 1. 业务发现的引导 (L.2.9 + SMT)

对于FLUTE/LOUTE协议，SACH作为一种特殊的MBMS用户服务，由一组元数据片段描述。引导业务发现即为获取SACH服务元数据段的方法与过程。FLUTE/LOUTE协议的业务发现的引导基于5.2.3.1.1定义的会话参数实现。具体参见3GPP TS 26.346 L.2.9节。

对于SMT协议，引导业务发现即为获取包含服务列表的MP表的方法与过程。该MP表封装为SMTP包，基于5.2.3.1.1定义的会话参数进行传送。

MBMS终端根据采用的传输协议，分别在相应的专用的IP地址与端口号接收业务引导信息。

* 1. SMT业务发布/发现机制

基于SMT协议，MBMS用户业务发布/发现需要以适当的方式向多个客户端发送包含服务列表的MP表来实现。业务发布端基于5.2.3.1.1定义的会话参数，采用专用IP地址与端口号，周期性发送包含服务列表的MP表。

MBMS用户选择一项服务，并确定与该服务对应的SMT\_Package\_id。MBMS终端基于获取的MBMS下载会话参数，采用专用IP地址与端口号，监测接收的SMTP包；通过SMTP包报头中的"type"字段来识别其携带的是否是MPT信令消息。如果是，则可以从该MPT信令消息中提取MP表，并根据选定服务对应的SMT\_Package\_id，获取相应的MBMS服务的媒资数据。

参 考 文 献

[1]

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_