

(19)日本国特許庁(JP)

(12)特許公報(B2)

(11)特許番号  
特許第7176588号  
(P7176588)

(45)発行日 令和4年11月22日(2022.11.22)

(24)登録日 令和4年11月14日(2022.11.14)

(51)国際特許分類		F I	
H 0 4 N	21/435 (2011.01)	H 0 4 N	21/435
H 0 4 N	21/434 (2011.01)	H 0 4 N	21/434
H 0 4 H	20/28 (2008.01)	H 0 4 H	20/28
H 0 4 H	20/93 (2008.01)	H 0 4 H	20/93
H 0 4 H	20/95 (2008.01)	H 0 4 H	20/95

請求項の数 3 (全34頁)

(21)出願番号	特願2021-56762(P2021-56762)	(73)特許権者	000002185 ソニーグループ株式会社 東京都港区港南1丁目7番1号
(22)出願日	令和3年3月30日(2021.3.30)	(74)代理人	100093241 弁理士 宮田 正昭
(62)分割の表示	特願2019-138969(P2019-138969) )の分割	(74)代理人	100101801 弁理士 山田 英治
原出願日	平成26年12月17日(2014.12.17)	(74)代理人	100095496 弁理士 佐々木 榮二
(65)公開番号	特開2021-106402(P2021-106402 A)	(74)代理人	100086531 弁理士 澤田 俊夫
(43)公開日	令和3年7月26日(2021.7.26)	(74)代理人	110000763 特許業務法人大同特許事務所
審査請求日	令和3年4月21日(2021.4.21)	(72)発明者	北里 直久 東京都港区港南1丁目7番1号 ソニー 最終頁に続く

(54)【発明の名称】 受信装置並びに受信方法

(57)【特許請求の範囲】

【請求項1】

所定のトランスポート方式に基づく伝送単位で伝送されるアプリケーションとアプリケーション情報テーブルを受信する受信部と、

前記アプリケーション情報テーブルに含まれる制御コードに基づいて、前記アプリケーションの状態を制御する制御部と、

を具備し、

前記アプリケーション情報テーブル内のアプリケーション情報のループに配置されるアプリケーションに関する記述子には、実行可能な状態にあるアプリケーションから自動起動アプリケーションへの切り替えに関する制御情報が含まれ、

前記制御部は、リモコン操作に基づいて指定されたアプリケーションを実行し、前記アプリケーション情報テーブルが更新された場合、前記制御情報に基づいて、前記リモコン操作に基づいて指定されたアプリケーションから前記自動起動アプリケーションへの切り替えを制御する、

受信装置。

【請求項2】

前記制御部は、前記アプリケーション情報テーブルが更新された場合、前記制御情報と前記制御コードに基づいて、前記リモコン操作に基づいて指定されたアプリケーションから前記自動起動アプリケーションへの切り替えを制御する、

請求項1に記載の受信装置。

## 【請求項 3】

所定のトランスポート方式に基づく伝送単位で伝送されるアプリケーションとアプリケーション情報テーブルを受信する受信ステップと、

前記アプリケーション情報テーブルに含まれる制御コードに基づいて、前記アプリケーションの状態を制御する制御ステップと、

を有し、

前記アプリケーション情報テーブル内のアプリケーション情報のループに配置されるアプリケーションに関する記述子には、実行可能な状態にあるアプリケーションから自動起動アプリケーションへの切り替えに関する制御情報が含まれ、

前記制御ステップでは、リモコン操作に基づいて指定されたアプリケーションを実行し、前記アプリケーション情報テーブルが更新された場合、前記制御情報に基づいて、前記リモコン操作に基づいて指定されたアプリケーションから前記自動起動アプリケーションへの切り替えを制御する、

10

受信方法。

## 【発明の詳細な説明】

## 【技術分野】

## 【0001】

本明細書で開示する技術は、所定のトランスポート方式でデータ伝送する送信装置及び送信方法、並びに、受信装置並びに受信方法に係り、特に、データ放送アプリケーション並びにその伝送に関する制御情報をMMT (MPEG Media Transport) 方式により伝送する送信装置及び送信方法、並びに、受信装置並びに受信方法に関する。

20

## 【背景技術】

## 【0002】

デジタル放送ではデータ放送と呼ばれる、ニュース、天気予報などの各種文字情報や、双方向性サービスを提供することができる。本出願時点において運用されているデータ放送アプリケーションは、ARIB (Association of Radio Industries and Broadcast) STD-B24によって規定されたBML (Broadcast Markup Language) 方式によって記述されている。

## 【0003】

データ放送アプリケーションには、放送番組に連動した番組連動型データ放送アプリケーションと、ニュースや天気予報のように放送番組とは特に連動しない番組非連動型データ放送アプリケーションに大別することができる。番組連動型データ放送アプリケーションは、基本的に、放送番組の切り替わりとともに内容が刷新される。したがって、番組非連動型データ放送アプリケーションを実行中に放送番組が切り替わったときには、強制的に番組連動型データ放送アプリケーションに切り替えて表示させたいという要望が放送局側にはある。従来 of BML (Broadcast Markup Language) 形式のデータ放送アプリケーションをデータ・カルセル伝送する放送システムでは、引き戻しフラグ (return to entry flag) というシグナリングにより、所望するデータ放送アプリケーションに強制的に引き戻す処理が可能である (例えば、特許文献1を参照のこと)。

30

40

## 【0004】

また、次世代のデジタル放送方式として、MPEGで新たなメディア・トランスポート方式として規格化されたMMT (MPEG Media Transport) 方式による超高解像度TV放送規格が検討されている。MMT方式では、異なる伝送路を組み合わせる利用することが容易であり、放送や通信の複数の伝送路に共通に用いることができる。

## 【先行技術文献】

## 【特許文献】

## 【0005】

【文献】WO2006/075590

## 【発明の概要】

50

## 【発明が解決しようとする課題】

【 0 0 0 6 】

本明細書で開示する技術の目的は、データ放送アプリケーションの伝送に関する制御情報を好適に伝送することができる、優れた送信装置及び送信方法、並びに、受信装置並びに受信方法を提供することにある。

## 【課題を解決するための手段】

【 0 0 0 7 】

本願は、上記課題を参酌してなされたものであり、請求項 1 に記載の技術は、放送サービスを構成する各コンポーネントを所定のトランスポート方式に基づく伝送単位にして送信する送信部と、

前記コンポーネントの伝送に関する制御情報を送信する情報送信部と、を具備し、

前記情報送信部は、前記送信部から送信されるデータ放送アプリケーションの引き戻しに関する制御情報を送信する、送信装置である。

【 0 0 0 8 】

本願の請求項 2 に記載の技術によれば、請求項 1 に記載の送信装置において、前記所定のトランスポート方式は M M T である。

【 0 0 0 9 】

本願の請求項 3 に記載の技術によれば、請求項 1 に記載の送信装置の前記情報送信部は、前記送信部から送信される映像及び音声コンポーネントで構成される放送番組に連動してエントリー・アプリケーションへの引き戻しを指示する制御情報を送信するように構成されている。

【 0 0 1 0 】

本願の請求項 4 に記載の技術によれば、請求項 1 に記載の送信装置の前記情報送信部は、前記送信部が送信するアプリケーションのコンポーネントの伝送に関する制御情報を示す第 1 のテーブル内で、エントリー・アプリケーションへの引き戻しを指示する引き戻し制御記述子をコンポーネント単位で格納して送信するように構成されている。

【 0 0 1 1 】

本願の請求項 5 に記載の技術によれば、請求項 4 に記載の送信装置の前記情報送信部は、引き戻し対象をコンポーネント単位で示す前記引き戻し制御記述子をコンポーネント毎に格納した前記第 1 のテーブルを送信するように構成されている。

【 0 0 1 2 】

本願の請求項 6 に記載の技術によれば、請求項 1 に記載の送信装置の前記情報送信部は、データ放送アプリケーションの提示単位に関する情報を示す第 2 のテーブル内でエントリー・アプリケーションへの引き戻しを指示する引き戻し制御記述子を提示単位で格納して送信するように構成されている。

【 0 0 1 3 】

本願の請求項 7 に記載の技術によれば、請求項 6 に記載の送信装置の前記情報送信部は、引き戻し対象を提示単位で示す前記引き戻し制御記述子をデータ・コンテンツ毎に格納した前記第 2 のテーブルを送信するように構成されている。

【 0 0 1 4 】

本願の請求項 8 に記載の技術によれば、請求項 1 に記載の送信装置の前記情報送信部は、データ・コンテンツ毎にデータ放送アプリケーションに関する情報を示す第 3 のテーブル内でエントリー・アプリケーションへの引き戻しを指示する引き戻し制御記述子をデータ・コンテンツ単位で格納して送信するように構成されている。

【 0 0 1 5 】

本願の請求項 9 に記載の技術によれば、請求項 8 に記載の送信装置の前記送信部は、データ放送アプリケーションを構成するファイルを前記所定のトランスポート方式に基づく伝送単位にグルーピングして送信し、前記情報送信部は、引き戻し対象を伝送単位で示す

10

20

30

40

50

前記引き戻し制御記述子をデータ・コンテンツ毎に格納した前記第 3 のテーブルを送信するように構成されている。

【 0 0 1 6 】

本願の請求項 1 0 に記載の技術によれば、請求項 1 に記載の送信装置の前記情報送信部は、アプリケーションの動作制御を示す第 4 のテーブル内でエントリー・アプリケーションへの引き戻しを指示する引き戻し制御記述子をアプリケーション単位で格納して送信するように構成されている。

【 0 0 1 7 】

本願の請求項 1 1 に記載の技術によれば、請求項 1 0 に記載の送信装置の前記情報送信部は、次に起動するアプリケーションを指定する前記引き戻し制御記述子をアプリケーション毎に格納した前記第 4 のテーブルを送信するように構成されている。

10

【 0 0 1 8 】

本願の請求項 1 2 に記載の技術によれば、請求項 1 に記載の送信装置の前記情報送信部は、アプリケーションの動作制御を示す第 4 のテーブル内のすべてのアプリケーションに適用される共通記述子としてエントリー・アプリケーションへの引き戻しを指示する引き戻し制御記述子を格納して送信するように構成されている。

【 0 0 1 9 】

本願の請求項 1 3 に記載の技術によれば、請求項 1 2 に記載の送信装置の前記情報送信部は、エントリー・アプリケーションへの引き戻しのために終了すべきアプリケーションの識別情報を示す前記引き戻し制御記述子を前記共通記述子として配置した前記第 4 の

20

【 0 0 2 0 】

また、本願の請求項 1 4 に記載の技術は、  
放送サービスを構成する各コンポーネントを所定のトランスポート方式に基づく伝送単位にして送信する送信ステップと、

前記コンポーネントの伝送に関する制御情報を送信する情報送信ステップと、  
を有し、

前記情報送信ステップは、前記送信ステップにおいて送信するデータ放送アプリケーションの引き戻しに関する制御情報を送信する、  
送信方法である。

30

【 0 0 2 1 】

また、本願の請求項 1 4 に記載の技術は、  
所定のトランスポート方式に基づく伝送単位で伝送される放送サービスの各コンポーネントを受信する受信部と、

前記コンポーネントの伝送に関する制御情報を受信する情報受信部と、  
を具備し、

前記情報受信部は、前記受信部で受信するデータ放送アプリケーションの引き戻しに関する制御情報を受信する、  
受信装置である。

【 0 0 2 2 】

40

また、本願の請求項 1 6 に記載の技術は、  
所定のトランスポート方式に基づく伝送単位で伝送される放送サービスの各コンポーネントを受信する受信ステップと、

前記コンポーネントの伝送に関する制御情報を受信する情報受信ステップと、  
を有し、

前記情報受信ステップでは、前記受信ステップで受信するデータ放送アプリケーションの引き戻しに関する制御情報を受信する、  
受信方法である。

【 発明の効果 】

【 0 0 2 3 】

50

本明細書で開示する技術によれば、データ放送アプリケーションの伝送に関する制御情報を好適に伝送することができる、優れた送信装置及び送信方法、並びに、受信装置並びに受信方法を提供することができる。

【 0 0 2 4 】

本明細書で開示する技術によれば、例えばMMT方式に基づいてデータ放送アプリケーションを伝送する放送システムにおいて、アプリケーションの表示切り替えに関する制御情報を好適に伝送することができる、優れた送信装置及び送信方法、並びに、受信装置並びに受信方法を提供することができる。

【 0 0 2 5 】

なお、本明細書に記載された効果は、あくまでも例示であり、本発明の効果はこれに限定されるものではない。また、本発明が、上記の効果以外に、さらに付加的な効果を奏する場合もある。

10

【 0 0 2 6 】

本明細書で開示する技術のさらに他の目的、特徴や利点は、後述する実施形態や添付する図面に基づくより詳細な説明によって明らかになるであろう。

【図面の簡単な説明】

【 0 0 2 7 】

【図 1】図 1 は、本明細書で開示する技術を適用したデジタル放送システム 1 0 の構成例を模式的に示した図である。

【図 2】図 2 は、MMT方式を用いる放送システムのプロトコル・スタック 2 0 0 を示した図である。

20

【図 3】図 3 は、図 2 に示した放送信号を送出する放送送出システム 1 1 の構成例を示した図である。

【図 4】図 4 は、図 2 に示した放送信号を受信する受信機 1 2 の構成例を示した図である。

【図 5】図 5 は、MMT / TLV方式に従って放送送出システム 1 1 から放送伝送路に送出される放送信号 5 0 0 のイメージを示した図である。

【図 6】図 6 は、PAメッセージ内のMPテーブルからパッケージの各アセットを指定する仕組みを示した図である。

【図 7】図 7 は、MMT伝送されるデータ放送アプリケーションを構成するファイルを取得する仕組みを説明するための図である

30

【図 8】図 8 は、アプリケーションの強制引き戻しを行なう動作例を示した図である。

【図 9】図 9 は、データ伝送メッセージで伝送されるデータ・アセット管理テーブル(DAMT)のシンタックス例 9 0 0 を示した図である。

【図 1 0】図 1 0 は、データ・アセット管理テーブルに配置されるエントリー・アプリケーションへの引き戻し制御の記述子のシンタックス例 1 0 0 0 を示した図である。

【図 1 1】図 1 1 は、データ・コンテンツ管理テーブル(DCCT)のシンタックス例 1 1 0 0 を示した図である。

【図 1 2】図 1 2 は、第 3 の実施例で利用するデータ・コンテンツ管理テーブルのシンタックス例 1 2 0 0 を示した図である。

【図 1 3】図 1 3 は、データ・コンテンツ管理テーブルのデータ・コンテンツの情報領域に配置されるエントリー・アプリケーションへの引き戻し制御の記述子のシンタックス例 1 3 0 0 を示した図である。

40

【図 1 4】図 1 4 は、MH - A I T (アプリケーション情報テーブル)のシンタックス例 2 0 0 0 を示した図である。

【図 1 5】図 1 5 は、MH - A I Tのアプリケーション情報のループ内に配置されるエントリー・アプリケーションへの引き戻し制御の記述子(Forced\_\_Return\_\_descriptor)のシンタックス例 1 5 0 0 を示した図である。

【図 1 6】図 1 6 は、MH - A I Tの共通記述子に配置する引き戻し制御記述子(Forced\_\_Return\_\_descriptor)のシンタックス例 1 6 0 0 を示した図である。

50

【図 17】図 17 は、受信機がデータ放送アプリケーションの実行を制御するための処理手順を示したフローチャートである。

【発明を実施するための形態】

【0028】

以下、図面を参照しながら本明細書で開示する技術の実施形態について詳細に説明する。

【0029】

図 1 には、本明細書で開示する技術を適用したデジタル放送システム 10 の構成例を模式的に示している。図示のデジタル放送システム 10 は、放送送出システム 11 と、受信機 12 で構成される。

【0030】

放送送出システム 11 は、放送信号の伝送に MMT 方式を適用しており、放送サービスを構成する各コンポーネントを IP パケットにして伝送する。具体的には、放送番組の映像信号や音声信号の符号、並びに、放送番組に関連するコンテンツ（データ放送アプリケーションなど）や字幕の信号は、MMTP ペイロードに乗せて MMTP パケット化され、IP パケットで伝送される。また、これらの IP パケットは、放送伝送路では TLV パケットの形式で伝送される。ここで、映像や音声、字幕などの放送番組本体に関わるコンポーネントは、タイムド・メディアである。また、データ放送に利用されるコンテンツ（HTML: Hyper Text Transfer Protocol）形式で記述されるデータ放送アプリケーションなどはノンタイムド・メディアである。

【0031】

一方、受信機 12 は、放送送出システム 11 から放送伝送路で送られてくる IP パケットを受信する。受信機 12 は、そして、受信機 12 は、受信パケットから映像や音声、字幕などの伝送メディアを復号して、画像や音声を提示する。また、受信機 12 は、受信パケットからデータ放送用の各ファイル・データを取得すると、HTML ブラウザなどのアプリケーション・エンジンを起動して、放送番組に連動したデータ放送の提示を行なう。

【0032】

図 2 には、MMT 方式を用いる放送システムのプロトコル・スタック 200 を示している。

【0033】

1 つの放送サービスは、映像 201、音声 202、字幕 203、アプリケーション 204、コンテンツ・ダウンロード 205 の各コンポーネントで構成される。映像 201 は HVC (High Efficiency Video Coding) 形式で符号化 211 され、音声 202 は AAC (Advanced Audio Coding) 形式で符号化 212 され、字幕 203 は次膜符号化 213 される。また、アプリケーション 204 は、EPG (Electric Program Guide) を含むが、HTML 5 形式で符号化 214 される。

【0034】

MMT レイヤー 220 上では、これらタイムド・メディア及びノンタイムド・メディアの符号化コンポーネント 211 ~ 214 は、MPU フォーマットにして、MMTP ペイロードに乗せて MMTP パケット化される。また、メディア・トランスポート方式である MMT に関わる（放送番組の構成などを示す）制御情報である MMT-SI (signaling Information) 221 も、MMTP ペイロードに乗せて MMTP パケット化される。なお、コンテンツ・ダウンロード 205 のデータ伝送方式 215 として、字幕・文字スーパー伝送方式、アプリケーション伝送方式、イベント・メッセージ伝送方式、汎用データ伝送方式の 4 種類が挙げられるが、詳細な説明は省略する。

【0035】

UDP (User Datagram Protocol) / IP レイヤー 230 では、MMTP パケットは IP パケット化される。また、タイムド・メディアのための現在時刻の情報を含む NTP (Network Time Protocol) パケット 206 も、IP パケット化される。さらに、これらの IP パケットは、TLV レイヤー 240 で TL

10

20

30

40

50

Vパケット化され、最下層の物理レイヤーである放送伝送路250で伝送される。また、IPパケットの多重のためのTLV多重化形式に関わるTLV-SI241も、TLVパケット化され、放送伝送路250で伝送される。TLVパケットを多重した伝送スロットは、伝送路のTMCC(Transmission and Multiplexing Configuration Control)信号251から、TLVストリーム識別情報(TLV\_\_stream\_\_id)を用いて特定される。

#### 【0036】

図3には、図2に示した放送信号を送出する放送送出システム11の構成例を示している。放送送出システム11は、例えば放送番組本体の制作元であるキー局(番組制作局)に相当する。図示の放送送出システム11は、時計部301と、信号送出部302と、ビデオ・エンコーダー303と、オーディオ・エンコーダー304と、キャプション・エンコーダー305と、シグナリング・エンコーダー306と、ファイル・エンコーダー307と、電子データ処理システム(Electronic Data Processing System:EDPS)308と、TLVシグナリング・エンコーダー309と、IPサービス・マルチプレクサー(MUX)310と、TLVマルチプレクサー(MUX)311と、変調・送信部312を備えている。

#### 【0037】

時計部301は、NTPサーバー(図示しない)から取得した時刻情報に同期した時刻情報を生成し、この時刻情報を含むIPパケットをIPサービス・マルチプレクサー310に送る。

#### 【0038】

信号送出部302は、例えばTV放送局のスタジオやVTRなどの記録再生機であり、タイムド・メディアである映像、音声、字幕などのストリーム・データや、ノンタイムド・メディアであるデータ放送アプリケーション用のファイル・データ(HTML文書データなど)をそれぞれ、ビデオ・エンコーダー303、オーディオ・エンコーダー304、キャプション・エンコーダー305、ファイル・エンコーダー307に送る。

#### 【0039】

EDPS308は、TV放送局のスケジューラー並びにファイルの供給源であり、ノンタイムド・メディアであるデータ放送アプリケーションと、放送番組の構成などを示す制御情報と、IPパケットの多重に関する制御情報をそれぞれ、ファイル・エンコーダー307、シグナリング・エンコーダー306、TLVシグナリング・エンコーダー309に送る。

#### 【0040】

ビデオ・エンコーダー303は、信号送出部302から送出される映像信号をHEVC符号化し、さらにパケット化して、映像信号のMMTパケットを含むIPパケットをIPサービス・マルチプレクサー310に送る。また、オーディオ・エンコーダー304は、信号送出部302から送出される音声信号をAAC符号化し、さらにパケット化して、音声信号のMMTパケットを含むIPパケットをIPサービス・マルチプレクサー310に送る。また、キャプション・エンコーダー305は、信号送出部302から送出される字幕信号を字幕符号化し、さらにパケット化して、字幕のMMTパケットを含むIPパケットをIPサービス・マルチプレクサー310に送る。

#### 【0041】

シグナリング・エンコーダー306は、EDPS308から送出される情報に基づいて、放送番組の構成などを示す制御情報を記述したシグナリング・メッセージ(MMT-SI)を生成し、ペイロード部にこのシグナリング・メッセージが配置されたMMTパケットを含むIPパケットをIPサービス・マルチプレクサー310に送る。本実施形態では、シグナリング・メッセージは、PA(Package Access)メッセージ、M2セクション・メッセージ、データ伝送メッセージの3種類に大別される。

#### 【0042】

ファイル・エンコーダー307は、信号送出部302又はEDPS308から送出され

10

20

30

40

50

るデータ放送アプリケーションをHTML5形式のファイル・データに符号化し、さらにパケット化して、このMMTパケットを含むIPパケットをIPサービス・マルチプレクサー310に送る。

【0043】

放送送出システム11は、送出するチャンネル（放送番組）毎にIPサービス・マルチプレクサー310を装備する。1つのチャンネルのIPサービス・マルチプレクサー310は、各エンコーダー303～307から送られてくる映像、音声、字幕、シグナリング・メッセージ（MMT-SI）、及びデータ放送アプリケーションの各々を含むIPパケットをマルチプレクスして、1つの放送サービス（チャンネル）を構成するTLVパケットを生成する。

10

【0044】

TLVシグナリング・エンコーダー309は、EDPS308から送出される情報に基づいて、上記のIPパケットの多重に関する制御情報（TLV-SI）をペイロード部に配置するTLVパケットを生成する。

【0045】

TLVマルチプレクサー311は、各IPサービス・マルチプレクサー310-1～310-N及びTLVシグナリング・エンコーダー309で生成されるTLVパケットをマルチプレクスして、TLVストリーム識別情報で識別されるTLVストリームを生成する。

【0046】

変調・送信部312は、TLVマルチプレクサー311で生成されたTLVストリームに対してRF変調処理を行なって、放送伝送路に送出する。

20

【0047】

図3に示した放送送出システム11の動作について説明しておく。

【0048】

時計部301では、NTPサーバー（図示しない）から取得した時刻情報に同期した時刻情報が生成され、この時刻情報を含むIPパケットが生成される。

【0049】

信号送出部302から送出される映像信号は、ビデオ・エンコーダー303に供給される。ビデオ・エンコーダー303では、映像信号がHEVC符号化され、さらにパケット化されて、HEVC符号化映像信号のMMTパケットを含むIPパケットが生成される。このIPパケットは、IPサービス・マルチプレクサー310に送られる。

30

【0050】

また、信号送出部302から送出される音声信号並びに字幕信号に対しても、同様の処理が行なわれる。すなわち、オーディオ・エンコーダー304で生成されるAAC符号化音声信号のMMTパケットを含むIPパケットがIPサービス・マルチプレクサー310に送られるとともに、キャプション・エンコーダー305で生成される字幕符号化信号のMMTパケットを含むIPパケットがIPサービス・マルチプレクサー310に送られる。

【0051】

また、シグナリング・エンコーダー306では、EDPS308から送出される情報に基づいて放送番組の構成などを示す制御情報を記述したシグナリング・メッセージ（MMT-SI）が生成され、ペイロード部にこのシグナリング・メッセージが配置されたMMTパケットを含むIPパケットが生成される。このIPパケットは、IPサービス・マルチプレクサー310に送られる。

40

【0052】

また、信号送出部302又はEDPS308から送出されるデータ放送アプリケーションは、ファイル・エンコーダー307に供給される。ファイル・エンコーダー307では、データ放送アプリケーションがHTML5形式に符号化され、さらにパケット化され、このMMTパケットを含むIPパケットが生成される。このIPパケットは、IPサービス・マルチプレクサー310に送られる。

【0053】

50



各IPサービス・マルチプレクサー310では、各エンコーダ303～307から送られてくる映像、音声、字幕、シグナリング・メッセージ(MMT-SI)、及びファイル・データ(HTML5文書)の各々を含むIPパケットがマルチプレクスされて、1つのチャンネルを構成するTLVパケットが生成される。

【0054】

TLVシグナリング・エンコーダ309では、EDPS308から送出される情報に基づいて、上記のIPパケットの多重に関する制御情報(TLV-SI)をペイロード部に配置するTLVパケットが生成される。

【0055】

TLVマルチプレクサー311では、各IPサービス・マルチプレクサー310-1～310-N及びTLVシグナリング・エンコーダ309で生成されるTLVパケットがマルチプレクスされて、TLVストリームが生成される。変調・送信部312では、TLVマルチプレクサー311で生成されたTLVストリームに対してRF変調処理が行なわれ、そのRF変調信号が放送伝送路に送出される。

【0056】

図4には、図2に示した放送信号を受信する受信機12の構成例を示している。図示の受信機12は、チューナー・復調部401と、デマルチプレクサー(DEMUX)402と、時計回復部403と、ビデオ・デコーダ404と、オーディオ・デコーダ405と、キャプション・デコーダ406と、システム制御部407と、アプリケーション制御部408と、キャッシュ・メモリー408と、アプリケーション・エンジン409と、IPインターフェース(I/F)410と、合成部411を備えている。図示の受信機12は、例えば家庭内に設置されるテレビ受信機やセット・トップ・ボックスの他、IPTVやCATVの再送信機を含むものとする。

【0057】

チューナー・復調部401は、放送信号を選局受信し、復調処理を行なって、TLVストリームを得る。デマルチプレクサー402は、このTLVストリームに対して、デマルチプレクス処理及びデパケット化処理を行なう。本実施形態では、デマルチプレクサー402は、TLVフィルター402-1と、IPフィルター402-2と、UDPフィルター402-3と、MMTフィルター402-4と、SIフィルター402-5を備えている。

【0058】

TLVフィルター402-1は、TLVストリーム識別情報に基づいて、放送伝送されるTLVパケットをフィルタリングする。IPフィルター402-2は、IPアドレスに基づいて、TLVパケットからIPパケットをフィルタリングするとともに、IPインターフェース410経由で受信したIPパケットのフィルタリングも行なう。また、UDPフィルター402-3は、UDPパケットをフィルタリングする。MMTフィルター402-4は、MMTPヘッダ(後述)内の情報に基づいて、IPパケットからMMTPパケットをフィルタリングして、映像、音声、並びにアプリケーションの各符号化コンポーネントを乗せたMMTPパケットを、それぞれビデオ・デコーダ404、オーディオ・デコーダ405、キャプション・デコーダ406、アプリケーション・エンジン409に振り分ける。SIフィルター402-5は、シグナリング情報SIをフィルタリングして、システム制御部407及びアプリケーション制御部408にそれぞれ振り分ける。SIフィルター402-5は、MMTストリームからMMT-SIをフィルタリングするMMT-SIフィルターと、TLVストリームからTLV-SIをフィルタリングするTLV-SIフィルターを含むものとする。

【0059】

時計回復部403は、デマルチプレクサー402内のIPフィルター402-2並びにUDPフィルター402-3でフィルタリングされたNTPパケットに含まれる現在時刻の情報に基づいて、この時刻情報に同期した時刻情報を生成して、各タイムド・メディアをデコードするにビデオ・デコーダ404、オーディオ・デコーダ405、キャプシ

10

20

30

40

50

ョン・デコーダー 406 にそれぞれ出力する。

【0060】

ビデオ・デコーダー 404 は、デマルチプレクサー 402 で得られる符号化映像信号をデコードして、ベースバンドの映像信号を得る。また、オーディオ・デコーダー 405 は、デマルチプレクサー 402 で得られる符号化音声信号をデコードして、ベースバンドの音声信号を得る。また、キャプション・デコーダー 406 は、デマルチプレクサー 402 で得られる字幕符号化信号をデコードして、字幕の表示信号を得る。

【0061】

アプリケーション制御部 408 は、SI フィルター 402 - 5 を介して受け取るシグナリング情報に基づいて、データ放送アプリケーションの処理を制御する。例えば、アプリケーション制御部 407 は、MMT - SI を解析して、デフォルト・エントリーに設定されているデータ放送アプリケーションを見つけると、アプリケーション・エンジン 409 に対してデータ放送の提示処理を指示する。

【0062】

本実施形態に係る放送システム 10 では、放送信号並びに IP ネットワークの 2 系統からデータ放送アプリケーションが伝送されることを想定している。前者の系統ではチューナー・復調部 401 で受信し、後者の系統では IP インターフェース 410 で受信し、いずれもデマルチプレクサー 402 内でパケット化された MMT パケットが MMT フィルター 402 - 4 によってアプリケーション・エンジン 409 に振り分けられる。

【0063】

アプリケーション・エンジン 409 は、例えば HTML ブラウザーなどであり、データ放送アプリケーションのエンティティであるファイル・データ (HTML 5 文書など) の処理を行なって、データ放送の表示信号を生成する。また、アプリケーション・エンジン 409 は、データ放送の表示に必要なファイル・データ (データ放送の表示に使用するモノメディアや、リンク先のアプリケーションなど) を IP インターフェース 410 経由で IP ネットワークから取得することもできる。

【0064】

システム制御部 410 は、SI フィルター 402 - 5 を介して受け取るシグナリング情報や、ユーザー操作部 (図示しない) を介したユーザーからの操作情報などに基づいて、当該受信機 12 の各部の動作を制御する。また、システム制御部 410 は、各デコーダー 404 ~ 406 におけるデコード・タイミングをシグナリング情報に基づいて制御し、映像、音声、及び字幕の提示タイミングを調整する。合成部 411 は、ベースバンドの映像信号に、字幕の表示信号及びデータ放送の表示信号を合成して、映像表示用の映像信号を得る。また、オーディオ・デコーダー 405 で得られるベースバンドの音声信号は、音声出力用の音声信号となる。映像信号及び音声信号からなる放送番組本編は、図示しないモニター・ディスプレイから映像及び音声出力される。また、データ放送アプリケーション・エンジン 409 が処理したデータ放送も、モニター・ディスプレイ上で放送番組本編の画面に重畳して表示される。

【0065】

IP インターフェース 410 は、例えばネットワーク・インターフェース・カードで構成され、インターネットやホーム・ネットワークなどの IP ネットワークに接続して、IP パケットの送受信処理を行なう。

【0066】

また、本実施形態では、IP フィルター 402 - 2 で IP アドレスに基づいてフィルタリングした IP パケットを、IP インターフェース 410 から IP ネットワークへ送信若しくは再送信することも想定される。また、放送サービスを IP アドレスだけでフィルタリングできることが判明すると、デマルチプレクサー 402 内の IP フィルター 402 - 2 だけで特定サービスを抽出して、受信機 12 から外部へ転送することができる。

【0067】

図 4 に示した受信機 12 の動作について説明しておく。

10

20

30

40

50

## 【 0 0 6 8 】

チューナー・復調部 4 0 1 では、放送信号が受信され、復調処理が行なわれて、T L V ストリームが得られる。デマルチプレクサー 4 0 2 では、この T L V ストリームに対して、デマルチプレクス処理及びパケット化処理を行なわれ、N T P 時刻情報、映像、音声、字幕、データ放送の各符号化信号、並びに、シグナリング情報が抽出され、ビデオ・デコーダー 4 0 4、オーディオ・デコーダー 4 0 5、キャプション・デコーダー 4 0 6、アプリケーション・エンジン 4 0 9、システム制御部 4 0 7、アプリケーション制御部 4 0 8 にそれぞれ振り分けられる。また、I P インターフェース 4 1 0 で受信した I P パケットについても同様に、デマルチプレクス処理及びパケット化処理を行なわれ、各部に振り分けられる。

10

## 【 0 0 6 9 】

また、デマルチプレクサー 4 0 2 で抽出された N T P パケットは、時計回復部 4 0 3 に振り分けられる。時計回復部 4 0 3 では、N T P パケットに載せられた時刻情報に基づいて、この時刻情報に同期した時刻情報が生成される。つまり、時計回復部 4 0 3 では、放送送出システム 1 1 側の時計部 3 0 1 で生成された時刻情報に合った時刻情報が生成される。

## 【 0 0 7 0 】

デマルチプレクサー 4 0 2 で抽出された符号化映像信号は、ビデオ・デコーダー 4 0 4 に送られてデコードされ、ベースバンドの映像信号が得られる。また、デマルチプレクサー 4 0 2 で抽出された字幕符号化信号はキャプション・デコーダー 4 0 6 に送られてデコードされ、字幕の表示信号が得られる。

20

## 【 0 0 7 1 】

アプリケーション制御部 4 0 8 では、S I フィルター 4 0 2 - 5 を介して受け取るシグナリング情報に基づいて、データ放送アプリケーションの処理が制御される。H T M L ブラウザなどからなるアプリケーション・エンジン 4 0 9 では、アプリケーション制御部 4 0 8 からの指示に従って、デマルチプレクサー 4 0 2 で抽出されたデータ放送アプリケーションの符号化信号 ( H T M L 5 文書 ) の処理が行なわれ、データ放送の表示信号が得られる。

## 【 0 0 7 2 】

合成部 4 1 1 では、ベースバンドの映像信号に、字幕の表示信号及びデータ放送の表示信号が合成され、画面表示用の映像信号が得られる。また、デマルチプレクサー 4 0 2 で抽出された符号化音声信号はオーディオ・デコーダー 4 0 5 に送られてデコードされ、音声出力用のベースバンドの音声信号が得られる。そして、映像信号及び音声信号は、図示しないモニター・ディスプレイから映像及び音声出力される。

30

## 【 0 0 7 3 】

図 1 に示したデジタル放送システム 1 0 では、放送送出システム 1 1 から受信機 1 2 へ、M M T 方式により放送信号を伝送することを想定している。図 5 には、M M T 方式に従って放送送出システム 1 1 から放送伝送路に送出される放送信号 5 0 0 のイメージを示している。

## 【 0 0 7 4 】

1 つのサービス ( チャンネル : 放送番組 ) の放送信号は、映像、音声、字幕などの放送番組本編に関わるタイムド・メディアと、放送番組に連動するデータ放送に利用されるファイル・データのようなノンタイムド・メディアで構成される。これらを符号化したメディア・データは、M P U フォーマットにして M M T P パケット化され、I P パケットで伝送される。また、メディア・トランスポート方式である M M T に関わる ( 放送番組の構成などを示す ) シグナリング情報 ( M M T - S I ) も、I P パケットで伝送される。これらの I P パケットは、放送伝送路では T L V パケットの形式で T L V ストリームとして伝送される。I P パケットの多重のための T L V 多重化形式に関わるシグナリング情報 ( T L V - S I ) も、T L V パケットの形式で伝送される。

40

## 【 0 0 7 5 】

50

MMT方式では、1つのチャンネル(放送番組)を構成するタイムド・メディア及びノンタイムド・メディアのデータを異なる伝送路の組み合わせで利用することが容易である。図5に示す例では、放送信号500として、映像、音声、字幕、ファイル・データ、シグナリング情報など、データのタイプ毎のMMT伝送路501~504が利用されている。各MMT伝送路は、それぞれ1つのIPデータ・フローに相当する。ここで言うIPデータ・フローとは、IPヘッダー及びUDPヘッダーの送信元IPアドレス、宛先IPアドレス、IPヘッダーのプロトコル種別、送信元ポート番号、宛先ポート番号の5種類のフィールドの値がすべて同じとなるIPパケットの集合である。なお、図中、字幕データ用の伝送路は便宜上、図示を省略している。また、TLV-SIのストリームについても、図5では省略している。

10

#### 【0076】

MMT方式の放送システム11は、放送伝送路でIPパケットを伝送する方式であるが、放送サービス毎(若しくは、放送局毎)に1つのIPアドレスをマッピングするという運用が可能である。このような場合、受信機側では、IPアドレスに基づいて放送信号500をフィルタリングすることで、所望する放送サービス(若しくは、所望する放送局)の各MMT伝送路501~504にアクセスすることができる。同じIPアドレス内の各MMT伝送路501~504で伝送されるMMTP(MMTプロトコル)パケットは、パケット識別情報(packet\_id:PID)で一意に指定することができる。また、異なるIPアドレス上のMMTPパケットは、パケット識別情報と、IPアドレスと、ポート番号の組み合わせにより指定することができる。

20

#### 【0077】

1つのチャンネル(放送番組)は、映像、音声、字幕、ファイル・データ(データ放送アプリケーション)などタイプの異なる複数のアセットで構成される「パッケージ」ということができる。ここで言う「パッケージ」は、MMT伝送路を使って伝送されるメディア・データの論理集合である。また、ここで言う「アセット」は、固有のアセット識別情報に関連付けられる、マルチメディアのプレゼンテーションを構成するために使用されるデータのエンティティである。

#### 【0078】

各アセットは、同じアセット識別情報を共有する1又はそれ以上のMPUの集合(論理グループ)で構成される。MPUは、MMT方式における伝送単位となるフォーマットとすることができる。各MPUは、それぞれのアセットに専用のES(Elementary Stream)すなわちMMT伝送路501~503上で伝送される。すなわち、伝送路501では共通のアセット識別情報を持つ映像信号のMPU論理グループからなる符号化映像信号のMMTPパケットが伝送され、伝送路502では、共通のアセット識別情報を持つ音声信号のMPU論理グループからなる符号化音声信号のMMTパケットが伝送され、伝送路503では共通のアセット識別情報を持つデータ放送アプリケーションのMPU論理グループからなる符号化アプリケーションのMMTパケットが伝送される。各MPUは、アセット識別情報と、該当する伝送路上でのMPUのシーケンス番号で特定される。また、各メディアを伝送するMMT伝送路は、アセット識別情報で識別することができる。

30

40

#### 【0079】

付言すれば、1つのパッケージ(放送番組)で、タイプが同じ複数の(すなわち、アセット識別情報が異なる)アセットが伝送されることもある。例えば、同じ放送番組に対して、2以上のデータ放送アプリケーションが提供される場合である。例えば、放送番組に連動する番組連動型データ放送アプリケーションと、放送番組に連動しない番組非連動型データ放送アプリケーション(例えば、天気予報やニュースなど)は、通常、別のアセットとして別々のアセット識別情報が割り振られ、別々のMPU論理グループとして異なるMMT伝送路で伝送される。図5では、放送番組連動型データ放送アプリケーションの伝送路503-1と放送番組非連動型データ放送アプリケーションの伝送路503-2を描いている。

50

## 【 0 0 8 0 】

また、MMT方式は、放送や通信の複数の伝送路に共通に用いることができる。例えば、データ放送用アプリケーション（HTML5文書など）のようなノンタイムド・メディアは、図5に示したように放送信号の伝送路503を用いてタイムド・メディアとともに伝送される以外に、IPネットワークなど通信伝送路（図示しない）を介して提供することもできる。

## 【 0 0 8 1 】

伝送路504では、MMTのパッケージの構成や放送サービスに関連する情報を示す伝送制御信号であるMMT-SIを含んだMMTPパケットが、カルーセル方式により繰り返し伝送される。伝送路504で伝送されるMMT-SIのシグナリング・メッセージとして、PAメッセージ510、M2セクション・メッセージ520、データ伝送メッセージ530を挙げることができる。

10

## 【 0 0 8 2 】

例えば、PAメッセージ510は、放送番組の構成などを示す制御情報であり、アセットのリストやその位置などパッケージを構成する情報を記述するMP（MMT Package）テーブル511が含まれている。

## 【 0 0 8 3 】

PAメッセージ510は、放送サービスのエントリー・ポイントであり、PAメッセージ510を伝送するMMTPパケットには、固定のパケット識別情報（例えば、0x0000）が割り当てられている。したがって、受信機側では、MMT伝送路504上で、上記固定のパケット識別情報を指定してPAメッセージ510を取得することができる。そして、PAメッセージ510で伝送されるMPテーブル511を参照して、パッケージ（放送番組）を構成する各アセット（映像、音声、字幕、ファイル・データ（データ放送アプリケーション）など）を指定することができる。

20

## 【 0 0 8 4 】

また、M2セクション・メッセージ520は、MPEG-2 Systemsのセクション拡張形式を伝送するメッセージである。MH-AIT（Application Information Table）521などのシグナリング・テーブルがM2セクション・メッセージ520に格納される。MH-AIT521は、アプリケーションに関する動的制御情報及び実行に必要な付加情報を伝送するテーブルであり、具体的には、MMT伝送路で送られてくるデータ放送アプリケーション（ファイル・データ）の処理方法（アプリケーションに適用される起動状態など）、並びにロケーション（URL）を指定する。アプリケーションの処理方法として、アプリケーションの自動起動を示すAUTOSTART（AS）、アプリケーションが実行可能の状態であることを示すPRESENT（PR）、アプリケーションの終了を示すKILL、アプリケーションの取得及び保持（事前キャッシュ）を示すPREFETCHを挙げることができる。

30

## 【 0 0 8 5 】

また、データ伝送メッセージ530は、データ放送アプリケーションの伝送に関する制御情報を伝送するためのメッセージである。1つのデータ伝送メッセージ530内には、データ・ディレクトリー管理テーブル531、データ・アセット管理テーブル532、データ・コンテンツ管理テーブル533の各シグナリング・テーブルが格納される。

40

## 【 0 0 8 6 】

データ・ディレクトリー管理テーブル531は、ディレクトリー単位（言い換えれば、データ放送アプリケーションの制作単位）でデータ放送アプリケーションを管理するためのテーブルである。同テーブル内は、1つのパッケージに含まれるディレクトリー並びにディレクトリーに含まれるサブディレクトリーやファイル（アイテム）に関するディレクトリー構造を記述しているので、アプリケーションのファイル構成とファイル伝送のための構成を分離することができる。

## 【 0 0 8 7 】

また、データ・アセット管理テーブル532は、アセット単位でデータ放送アプリケー

50

ションを管理するためのテーブルであり、アセット内のMPUの構成とのMPU毎のバージョン情報を記述している。

【0088】

また、データ・コンテンツ管理テーブル533は、各データ・コンテンツを構成するデータ放送アプリケーションの提示単位(Presentation Unit:PU)やファイルの情報を管理するためのテーブルである。ここで言う「データ・コンテンツ」は、一般に、1つの放送番組の連携するアプリケーション・データ全体に相当する。同テーブルは、提示単位(PU)の情報である場合、データ放送アプリケーションのファイルの構成情報をデータ放送の提示単位(PU)で記述しており、データ放送アプリケーション用のファイル・データの柔軟で有効なキャッシュ制御に利用することができる。

10

【0089】

MMTによるデータ放送アプリケーションの伝送方式において、データ伝送メッセージで伝送する上記3種類のシグナリング・テーブル531~533を活用することにより、ファイル単位の伝送データ構造やコンテンツ(データ放送アプリケーション)制作におけるディレクトリー構造とは独立して、受信機におけるキャッシュ・メモリーの有効活用のために、アプリケーション単位、提示単位といった利用単位のデータ構造を表現することができる(例えば、本出願人に既に譲渡されている特願2014-88630号明細書を参照のこと)。

【0090】

MMT-SIとして伝送されるメッセージやテーブルのパケット識別情報は、固定されているものや、他のテーブルから間接指定されるものがある。このうち、PAメッセージは、放送サービスのエントリー・ポイントであり、固定のパケット識別情報(例えば、0x0000)が割り当てられている。PAメッセージで伝送されるMPテーブルでは、パッケージ(放送番組)を構成する各アセット(映像、音声、字幕、ファイル・データ(データ放送アプリケーション)など)を指定している。したがって、図12に示すように、MPテーブルを参照して、パッケージ(放送番組)を構成する各アセット(映像、音声、字幕、ファイル・データ(データ放送アプリケーション)など)を指定することができる。

20

【0091】

図7には、同一のIPデータ・フローに多重されたデータ放送アプリケーションのアイテムを取得する仕組みを図解している。

30

【0092】

データ放送アプリケーションを構成するファイルは、HTML5などのアプリケーション記述内でパス名を指定される。ここで言うパス名は、ディレクトリー・ノード名とファイル名の組み合わせで記述される。また、ディレクトリー・ノードとファイルを統合した記述子としてノード・タグを規定し、各シグナリング・テーブルをリンクする情報として使用する。

【0093】

データ放送アプリケーションからパス名を指定すると、参照番号701で示すように、データ伝送メッセージ内のデータ・ディレクトリー管理テーブルから、指定されたパス名のファイルのノード・タグを得ることができる。

40

【0094】

次いで、参照番号702で示すように、同じくデータ伝送メッセージ内のデータ・アセット管理テーブルから、データ・ディレクトリー管理テーブルで得られたノード・タグを持つアイテムが伝送されるアセットのコンポーネント・タグ、ダウンロード識別情報、MPUシーケンス番号、及びアイテム識別情報を得ることができる。

【0095】

さらに、参照番号703で示すように、MPテーブルから、データ・アセット管理テーブルで得られたコンポーネント・タグを持つアセットのロケーション情報を取得すると、参照番号704で示すように、該当するファイルが実際に伝送されるデータ・アセットを特定することができる。

50

## 【 0 0 9 6 】

そして、特定されたデータ・アセット内で、データ・アセット管理テーブルから得られたダウンロード識別情報とアイテムを伝送するMMTPパケットのヘッダー領域に記載されたダウンロード識別情報とにより、カールセルに対応するファイルの繰り返し伝送の単位を一意に識別することができる。参照番号705で示すように、繰り返し伝送されるアイテムのうち、データ・アセット管理テーブルから得られたMPUシーケンス番号及びアイテム識別情報を持つアイテムを所望のファイルとして指定することができる。ノード・タグは、データ伝送メッセージ内で、MPUシーケンス番号はアセット（IPデータ・フロー）内で、アイテム識別情報はサービス事業者内で、それぞれ一意であるものとする。

## 【 0 0 9 7 】

データ放送アプリケーションには、放送番組に連動した番組連動型データ放送アプリケーションと、ニュースや天気予報のように放送番組とは特に連動しない番組非連動型データ放送アプリケーションに大別することができる。放送局側には、放送番組に連動して、番組連動型データ放送アプリケーションに表示を強制的に切り替えたいという要求がある。「放送番組に連動して」とは、放送番組が切り替わったときや、放送番組内でコーナーが切り替わったときのことなどを意味する。このようなタイミングで、番組非連動型データ放送アプリケーションが実行されていたとしても、これを終了して、放送番組（若しくは番組内のコーナー）でエントリーに指定された番組連動型データ放送アプリケーションに強制的に引き戻したい、という要求が放送局にはある。

## 【 0 0 9 8 】

図8には、エントリーに指定された番組連動型データ放送アプリケーションへの強制引き戻しが行なわれるケースを模式的に示している。

## 【 0 0 9 9 】

図8ではMMT方式の放送伝送路を想定しており、映像アセット801や音声アセット802、データ放送アプリケーションのアセット803、804がそれぞれ専用のMMT伝送路で伝送される。但し、参照番号803は番組連動型データ放送アプリケーションのアセット、参照番号804は番組非連動型データ放送アプリケーションのアセットとする。

## 【 0 1 0 0 】

また、アセットのMMT伝送に併せて、MMTに関わる制御情報（MMT-SI）として、参照番号811～816で示すように、イベント情報テーブル（EIT）、アプリケーション情報テーブル（AIT）、データ・ディレクトリー管理テーブル（DDMT）、データ・アセット管理テーブル（DAMT）、データ・コンテンツ管理テーブル（DCT）、イベント・メッセージ・テーブル（EMT）といった各種シグナリング・テーブルが伝送される。このうちイベント情報テーブル811は、番組名、放送日時、番組内容などが記述され、番組の変更や終了によって更新される。また、アプリケーション情報テーブルは、各アプリケーションの処理方法を示し、例えば自動起動（AS）すべきアプリケーション、実行可能な状態（PR）にあるアプリケーションを記述している。実行可能なアプリケーションは、アセット803、804内で伝送されている、番組連動型及び番組非連動型のすべてのアプリケーションである。また、自動起動すべきアプリケーションはその放送番組（若しくは番組内のコーナー）でエントリーとなる放送連動型データ放送アプリケーションに相当する。また、イベント・メッセージ・テーブル816は、イベント・メッセージに関する情報を伝送する。

## 【 0 1 0 1 】

図8の横方向は時間軸に相当するものとし、時刻t1で放送番組P1のオンエアが開始し、時刻t2でその放送番組P1内でコーナーが変更し、時刻t3で放送番組P1が終了して次の放送番組P2が開始するものとする。

## 【 0 1 0 2 】

アセット803では、番組の切り替わり並びに番組コーナーの切り替わりに連動して、伝送されるアプリケーションが変更される。すなわち、時刻t1～t2の期間に番組連動型アプリケーションapp1、app2が伝送され、時刻t2～t3の期間に番組連動型

10

20

30

40

50

アプリケーション a p p 6、a p p 7 が伝送される。また、時刻 t 3 以降では、番組連動型アプリケーションの伝送が停止される。一方、アセット 8 0 4 では、番組の切り替えや番組コーナーの切り替えには連動せずに、番組非連動型アプリケーション a p p 3、a p p 4 が伝送され続ける。

【 0 1 0 3 】

アプリケーション情報テーブル 8 1 1 は、時刻 t 1 ~ t 2 の期間では、自動起動すべきアプリケーション ( A S ) に a p p 1 を示すとともに、実行可能な状態にあるアプリケーション ( P R ) としてアセット 8 0 3 及び 8 0 4 で同期間内に伝送される他のすべてのアプリケーション a p p 2、a p p 3、a p p 4 を示している。また、時刻 t 2 ~ t 3 の期間では、自動起動すべきアプリケーション ( A S ) に a p p 6 を示すとともに、実行可能な状態にあるアプリケーション ( P R ) としてアセット 8 0 3 及び 8 0 4 で同期間内に伝送される他のすべてのアプリケーション a p p 7、a p p 3、a p p 4 を示している。また、時刻 t 3 以降では、自動起動すべきアプリケーション ( A S ) に番組非連動アプリケーション a p p 3 を示すとともに、実行可能な状態にあるアプリケーション ( P R ) としてアプリケーション a p p 4 を示している。

10

【 0 1 0 4 】

図 8 の最下段では、各期間におけるアプリケーションの参照関係並びにアプリケーションの表示が遷移する様子を示している。時刻 t 1 ~ t 2 の期間では、a p p 1 がエントリー・アプリケーションに指定されており、a p p 1 は番組連動型アプリケーション a p p 2 と番組非連動型アプリケーション a p p 3 を参照している。さらに a p p 3 は、同じく番組非連動型であるアプリケーション a p p 4 を参照している。また、時刻 t 2 ~ t 3 の期間では、a p p 6 がエントリー・アプリケーションに指定されており、a p p 1 は番組連動型アプリケーション a p p 7 と番組非連動型アプリケーション a p p 3 を参照している。さらに a p p 3 は、時刻 t 2 以前と同様に、番組非連動型であるアプリケーション a p p 4 を参照している。

20

【 0 1 0 5 】

時刻 t 1 で放送番組 P 1 のオンエアが開始されたことに連動して、エントリー・アプリケーション a p p 1 が自動起動する。その後、ユーザーのリモコン操作などに応じて、データ放送の表示が番組非連動型アプリケーション a p p 3 に遷移したとする。そして、時刻 t 2 に到達して、放送番組 P 1 内でコーナーが変更したことに連動して、エントリー・アプリケーションが a p p 6 に切り替わる。このような場合、放送番組 P 1 を放送する放送局には、番組非連動型アプリケーション a p p 3 を強制的に終了して、エントリーとしての番組連動型アプリケーション a p p 6 の表示に引き戻したいという要求がある。

30

【 0 1 0 6 】

従来の B M L 形式のデータ放送アプリケーションの伝送方式では、引き戻しフラグ ( r e t u r n t o \_ e n t r y \_ f l a g ) というシグナリングにより、所望するデータ放送アプリケーションに強制的に引き戻す処理が可能である。

【 0 1 0 7 】

これに対し、本明細書では、M M T 方式を採用する放送システムにおいて、データ伝送に関するシグナリング・メッセージでデータ放送アプリケーションの強制切り替えを指示する幾つかの実施例について説明する。

40

【実施例 1】

【 0 1 0 8 】

第 1 の実施例では、データ伝送メッセージで伝送するデータ・アセット管理テーブル ( 図 5 を参照のこと ) 内に、エントリー・アプリケーションへの引き戻し制御の記述子を配置することにより、エントリー・アプリケーションへの強制引き戻しを実現する。

【 0 1 0 9 】

図 9 には、データ伝送メッセージで伝送されるデータ・アセット管理テーブル ( D A M T ) のシンタックス例 9 0 0 を示している。データ・アセット管理テーブルは、アセット単位でデータ放送アプリケーションを管理するためのテーブルであり、アセット内の M P

50



Uの構成とのMPU毎のバージョン情報を記述している。

【0110】

`table__id` (テーブル識別) は、各種シグナリング情報においてデータ・アセット管理テーブルであることを示す8ビットの固定値である。`version__` (バージョン) は、このデータ・アセット管理テーブルのバージョンを示す8ビットの整数値のパラメーターである。例えばデータ・アセット管理テーブルを構成する一部のパラメーターでも更新した場合には、`version`は+1だけインクリメントされる。`length`は、このフィールドの直後からカウントされる、このデータ・アセット管理テーブルのサイズをバイト単位で示す、16ビット長のパラメーターである。

【0111】

`number__of__data__components` は、パッケージに含まれるデータ・コンポーネントの数(すなわち、データ放送アプリケーションのアセット数)を示す、8ビットのパラメーターである。例えば、1つのパッケージ(放送番組)で番組連動型データ放送アプリケーションと番組非連動型データ放送アプリケーションの2種類のデータ・コンポーネントが伝送されることが想定される。`number__of__data__components`の数分だけ、以下のデータ・コンポーネント(すなわち、アセット)のループが配置され、データ・コンポーネント毎の情報が格納される。各データ・コンポーネントのループ(若しくは、アセットのループ)内には、データ・コンポーネントの属性情報と、データ・コンポーネントに含まれるMPUの情報が書き込まれる。

【0112】

データ・コンポーネントの属性情報として、`transaction__id` (トランザクション識別情報) と、`component__tag` と、`download__id` (ダウンロード識別情報) が含まれる。`transaction__id`は、当該データ・コンポーネントのバージョン機能を持つ識別子である。`component__tag`は、当該データ・コンポーネントのストリームを識別するためのラベルである。`component__tag`は、MPテーブル内にアセット記述子として配置されるMHストリーム識別記述子内の`component__tag`と同一の値であるとする。`download__id`は、データ・コンテンツを一意に識別するためのラベルの役割をする。アプリケーション(ノンタイムド・メディア)を伝送するMMTPパケットには、必要に応じて拡張ヘッダーにダウンロード識別情報が書き込まれる。

【0113】

`num__of__mpus` は、当該データ・コンポーネントに含まれるMPUの数を示す。そして、`num__of__mpus`の数分だけ配置されるMPUのループ内には、各MPUの属性情報が格納される。`MPU__sequence__number`は、MPUに割り振られるMPUシーケンス番号である。`num__of__items`は、MPUに含まれるアイテム(ファイル・データ)の数を示す。そして、`num__of__items`の数分だけ配置されるアイテムのループ内には、各アイテムの情報が格納される。

【0114】

1つのアイテムのループ内には、アイテムの属性情報とアイテムに関する情報が格納される。アイテムの属性情報として、`item__id`、`node__tag`、`item__size`、`item__version`、`item__checksum`が格納される。`item__id`は、MMT伝送路上でアイテムを一意に識別する32ビットの値である。`node__tag`は、アイテムに対応するノード・タグとしてアイテムを識別する16ビットの値である。シグナリング情報としては、32ビットの`item__id`に代えて16ビットの`node__tag`を使用することで、データ伝送メッセージ上のアイテムの識別に必要なビット・サイズを削減することができる。`item__size`は、アイテムのサイズをバイト単位で表す。`item__version`は、アイテムのバージョンを示し、アイテムの内容が更新される度に`version`は+1だけインクリメントされる。`item__checksum`は、アイテムのチェックサムを示す。なお、チェックサムは、すべてのファイルに対して必ず設定するのは情報量が多いと考えられるので、1ビットの`check`

10

20

30

40

50

`_sum_flag`を設定し、これに1が代入された場合にのみ32ビットの`item_checksum`が現れる。`checksum_flag`はチェックサムの記載があるか否かを示すフラグであり、このフラグが1のときには`item_checksum`が記載される。`item_info_length`は後続のアイテム情報領域のバイト長を示し、このバイト数分のループからなる一連の領域にアイテムに関する情報がバイト単位(`item_info_byte`)で書き込まれる。

【0115】

また、MPUのループ内には、各MPUの情報が格納される。具体的には、`MPU_info_length`は後続のMPU情報領域のバイト長を示し、このバイト数分のループからなる一連の領域にMPUに関する情報がバイト単位(`item_info_byte`)で書き込まれる。

10

【0116】

アセットのループの最後には、アセット単位での情報を記述する記述子が配置される。具体的には、`descriptor_loop_length`は、`descriptor`の全バイト長を示す。`descriptor`は、`descriptor_loop_length`の数分のループからなる一連の領域に、アセット単位での情報を記述する記述子(`descriptor`)を格納する。格納される記述子は別途定義する。

【0117】

第1の実施例では、アセット単位での情報を記述する記述子(`descriptor`)として、エントリー・アプリケーションへの引き戻し制御の記述子(`returnToEntry_descriptor`)を配置する。

20

【0118】

図10には、データ・アセット管理テーブルに配置されるエントリー・アプリケーションへの引き戻し制御の記述子(`returnToEntry_descriptor`)のシンタックス例1000を示している。

【0119】

`descriptor_tag`は、当該記述子1000を識別する、8ビットの整数値である。`descriptor_length`は、このフィールドより後に続く当該記述子1000のデータのバイト長を書き込む領域である。そして、`return_to_entry_flag`には、強制引き戻しを行なうか否かが示される。

30

【0120】

第1の実施例においても2通りのエントリー・アプリケーションへの引き戻し制御の記述子の適用方法がある。第1の方法はデータ・アセット管理テーブルの、自動起動アプリケーションを含むデータ・アセットのループの記述子領域には図10に示した引き戻し制御記述子を配置し、それ以外のデータ・アセットのループには引き戻し制御記述子を配置しない。これにより、受信機は自動起動アプリケーションを含むデータ・アセットで伝送されたアプリケーション以外のアプリケーションを実行している場合には当該アプリケーションを終了してエントリー・アプリケーションを起動する。この場合の受信機動作について考察してみる。例えば、時刻 $t_2$ に到達してデータ・アセット管理テーブルが更新されると、受信機はその内容をチェックする。受信機は、時刻 $t_1 \sim t_2$ の期間中に番組非連動型アプリケーションである`app3`の実行に切り替えているが、更新されたデータ・アセット管理テーブルでは、引き戻し制御記述子が含まれるので、当該記述子が記述されたデータ・アセットとは異なるデータ・アセットで伝送された`app3`を直ちに終了し、エントリー・アプリケーションである`app6`を起動する。

40

【0121】

第2の方法はデータ・アセット管理テーブルの引き戻し対象となるデータ・アセットのループに引き戻し制御記述子を配置する。これにより受信機は当該記述子が配置されたデータ・アセットに含まれるアプリケーションを実行中の場合には当該アプリケーションを終了してエントリー・アプリケーションを起動する。この場合の受信機動作について考察してみる。例えば、時刻 $t_2$ に到達してデータ・アセット管理テーブルが更新されると、

50

受信機はその内容をチェックする。受信機は、時刻  $t_1 \sim t_2$  の期間中に番組非連動型アプリケーションである  $app_3$  の実行に切り替えているが、更新されたデータ・アセット管理テーブルでは、 $app_3$  が伝送されるデータ・アセットに引き戻し制御記述子が配置されるので、 $app_3$  を直ちに終了し、エントリー・アプリケーションである  $app_6$  を起動する。

【0122】

このように、データ・アセット管理テーブルを用いることで、アセット単位でアプリケーションの強制引き戻しを制御することができる。

【実施例2】

【0123】

第2の実施例では、データ伝送メッセージで伝送するデータ・コンテンツ管理テーブル（図5を参照のこと）内に、エントリー・アプリケーションへの引き戻し制御の記述子を配置することにより、エントリー・アプリケーションへの強制引き戻しを実現する。

【0124】

図11には、データ伝送メッセージで伝送されるデータ・コンテンツ管理テーブル（DCCCT）のシンタクス例1100を示している。データ・コンテンツ管理テーブルは、提示単位（Presentation Unit：PU）毎にデータ放送アプリケーションを管理するためのテーブルである。同テーブルは、データ放送アプリケーションのファイルの構成情報をデータ放送の提示単位（PU）で記述している。

【0125】

`table_id`（テーブル識別子）には、各種シグナリング情報においてデータ・コンテンツ管理テーブルであることを示す8ビットの固定値が書き込まれる。`version_`（バージョン）は、当該データ・コンテンツ管理テーブルのバージョンを示す8ビットの整数値のパラメーターである。例えば当該テーブルを構成する一部のパラメーターでも更新した場合には、`version`は+1だけインクリメントされる。`length`は、このフィールドの直後からカウントされる、当該データ・コンテンツ管理テーブルのサイズをバイト単位で示す、16ビット長のパラメーターである。

【0126】

`number_of_contents`は、パッケージ（放送番組）で伝送されるデータ・コンテンツの数を示す、8ビットのパラメーターである。`number_of_contents`の数分だけ、以下のデータ・コンテンツのループが配置され、データ・コンテンツ毎の情報が格納される。データ・コンテンツは、一般に、1つの放送番組の連携するアプリケーション・データ全体に相当する。

【0127】

1つのデータ・コンテンツのループ内には、当該データ・コンテンツの属性情報として、`content_id`と、`content_version`と、`content_size`、`PU_info_flag`を記載するとともに、当該データ・コンテンツを構成するデータ放送アプリケーションの提示単位（PU）又はファイルに関する情報が書き込まれる。`content_id`（コンテンツ識別情報）は、当該データ・コンテンツを一意に識別するラベルである。`content_version`は、当該データ・コンテンツのバージョン番号を書き込む領域である。`content_size`は、当該データ・コンテンツのサイズを書き込む領域である。また、`PU_info_flag`は、当該データ・コンテンツのループが提示単位（PU）の情報を示すか否かを示すフラグである。

【0128】

`PU_info_flag = 1`は、当該データ・コンテンツのループが提示単位（PU）の情報であることを示す。この場合の後続の領域には、当該データ・コンテンツを構成するデータ放送アプリケーションの提示単位（PU）の情報が格納される。`number_of_PUs`に当該データ・コンテンツに含まれるPUの数が書き込まれ、これに続いて、`number_of_PUs`の数分だけPUのループが配置される。

【0129】

10

20

30

40

50

1つのPUのループ内には、PUの識別情報であるPU\_\_tagと、PUのサイズを書き込む領域であるPU\_\_sizeと、当該PUを構成するファイル又はディレクトリーのノード指定の数を示すnumber\_of\_member\_nodesが書き込まれる。そして、number\_of\_member\_nodesの数分だけ配置されるノードのループ内では、当該PUを構成するファイル又はディレクトリーを識別するノード・タグが書き込まれる。

【0130】

また、1つのPUのループ内には、さらにPUの情報が書き込まれる。具体的には、PU\_\_info\_\_lengthにPU情報のバイト長を示し、このバイト数分のループからなる一連の領域にPUの情報がバイト単位(PU\_\_info\_\_byte)で書き込まれる。

10

【0131】

一方、PU\_\_info\_\_flag = 0は、当該データ・コンテンツのループがファイルの情報であることを示す。この場合の後続の領域には、当該データ・コンテンツを構成するファイル又はディレクトリーの情報が書き込まれる。具体的には、当該データ・コンテンツを構成するファイル又はディレクトリーのノード指定の数を示すnumber\_of\_nodesが書き込まれる。そして、number\_of\_nodesの数分だけ配置されるノードのループ内で、当該データ・コンテンツを構成するファイル又はディレクトリーを識別するノード・タグが書き込まれる。

【0132】

図11に示すように、データ・コンテンツ管理テーブルのPUのループ内には、データ・コンテンツを構成する提示単位PU毎に、PUの情報を格納することができる(但し、PU\_\_info\_\_flag = 1の場合)。第2の実施例では、この情報領域を利用して、エントリー・アプリケーションへの引き戻し制御の記述子(return\_to\_entry\_descriptor)を配置して、アプリケーションの提示単位でアプリケーションの強制引き戻しを制御する。

20

【0133】

PU毎の情報領域に配置する引き戻し制御記述子のシンタックスは、図10と同様でよい。引き戻し制御記述子のreturn\_to\_entry\_flagには、強制引き戻しを行なうか否かが示される。

【0134】

第2の実施例においても、2通りのエントリー・アプリケーションへの引き戻し制御の記述子の適用方法がある。第1の方法は、データ・コンテンツ管理テーブルの、自動起動アプリケーションを含むPUのループのPU情報領域には引き戻し制御記述子を配置するが、それ以外のPUのループには引き戻し制御記述子を配置しない。自動起動アプリケーションを含むPUは、PU\_\_tag = 0で識別することができるので、PU\_\_tag = 0のPUのループにのみ引き戻し制御記述子を配置し、PU\_\_tagが0でないPUのループには引き戻し制御記述子を配置しない。自動起動アプリケーションへの強制引き戻し時には、引き戻し制御記述子を配置しないPUに含まれるアプリケーションを終了すべきことを示す。第2の方法は、逆に、自動起動アプリケーションを含まないPUのループで、引き戻し対象となるPUのPU情報領域に引き戻し制御記述子を配置する。自動起動アプリケーションへの強制引き戻し時には、引き戻し制御記述子を配置したPUに含まれるアプリケーションを終了すべきことを示す。

30

40

【0135】

PU\_\_tag = 0のPUのループにのみ引き戻し制御記述子を配置する場合の受信機側の動作について考察してみる。例えば時刻t2に到達して、データ・コンテンツ管理テーブルが更新されると、受信機はその内容をチェックする。受信機は、時刻t1~t2の期間中に番組非連動型アプリケーションであるapp3の実行に切り替えているが、更新されたデータ・コンテンツ管理テーブルでは、引き戻し制御記述子がPU\_\_tag = 0のPUのループに配置されているので、app3を直ちに終了する。その結果、受信機は、時刻t2以降においてエントリー・アプリケーションに指定されているapp6を、放送局

50

の要求通りに自動起動することができる。

【0136】

このように、データ・コンテンツ管理テーブルのPUのループ内にエントリー・アプリケーションへの引き戻し制御の記述子(`returnToEntryDescriptor`)を配置することで、提示単位(PU)でアプリケーションの強制引き戻しを制御することができる。

【実施例3】

【0137】

第3の実施例でも、第2の実施例と同様に、データ・コンテンツ管理テーブル内にエントリー・アプリケーションへの引き戻し制御の記述子を配置することにより、エントリー・アプリケーションへの強制引き戻しを実現する。但し、第3の実施例では、PUのループではなく上位のデータ・コンテンツのループに引き戻し制御記述子を配置して、PUよりも上位のデータ・コンテンツの単位でエントリー・アプリケーションへの強制引き戻しを制御する。

10

【0138】

第3の実施例では、データ・コンテンツ管理テーブルのデータ・コンテンツのループに、引き戻し制御記述子を配置するためのデータ・コンテンツの情報領域を新たに定義する。図12には、第3の実施例で利用するデータ・コンテンツ管理テーブルのシンタックス例1200を示している。図11のシンタックス例1100と相違する箇所を点線で囲んで示した。以下、このシンタックス例1200について説明するが、図11と重複するパラメーターについては説明をする。

20

【0139】

参照番号1201で示すように、データ・コンテンツのループ内に、当該データ・コンテンツの属性情報として、`PU_info_flag`の後に、`content_info_flag`を定義する。`content_info_flag`は、当該データ・コンテンツのループがデータ・コンテンツの情報を含むか否かを示すフラグである。

【0140】

`content_info_flag = 1`は、当該データ・コンテンツのループがデータ・コンテンツの情報を含むことを示す。この場合、当該データ・コンテンツのループの最後にデータ・コンテンツの情報を格納する領域が定義される。具体的には、参照番号1202で示すように、`content_info_length`にデータ・コンテンツの情報のバイト長を示し、このバイト数分のループからなる一連の領域にデータ・コンテンツの情報がバイト単位(`content_info_byte`)で書き込まれる。

30

【0141】

第3の実施例では、このデータ・コンテンツの情報領域を利用して、エントリー・アプリケーションへの引き戻し制御の記述子(`returnToEntryDescriptor`)を配置して、アプリケーションの提示単位でアプリケーションの強制引き戻しを制御する。

【0142】

図13には、データ・コンテンツ管理テーブルのデータ・コンテンツの情報領域に配置される、エントリー・アプリケーションへの引き戻し制御の記述子(`returnToEntryDescriptor`)のシンタックス例1300を示している。

40

【0143】

`descriptor_tag`は、当該記述子1300を識別する、8ビットの整数値である。`descriptor_length`は、このフィールドより後に続く当該記述子1300のデータのバイト長を書き込む領域である。`PU_number_of_target_to_return`には、当該データ・コンテンツに含まれる、アプリケーションの強制引き戻し時に終了することが指定されるPUの数が書き込まれる。そして、`PU_number_of_target_to_return`の数分だけ配置されるループ内では、アプリケーションの強制引き戻し時に終了すべき各PUを識別するPUタグが

50

書き込まれる。

【0144】

この場合の受信機側の動作について考察してみる。例えば時刻  $t_2$  に到達して、データ・コンテンツ管理テーブルが更新されると、受信機はその内容をチェックする。受信機は、時刻  $t_1 \sim t_2$  の期間中に番組非連動型アプリケーションである  $app_3$  の実行に切り替えているが、更新されたデータ・コンテンツ管理テーブルでは、番組非連動型アプリケーション  $app_3$  を含むデータ・コンテンツのループに配置される引き戻し制御記述子内で  $app_3$  を含む提示単位 (PU) のタグが指定されているので、 $app_3$  を直ちに終了する。その結果、受信機は、時刻  $t_2$  以降においてエントリー・アプリケーションに指定されている  $app_6$  を、放送局の要求通りに自動起動することができる。

10

【0145】

このように、データ・コンテンツ管理テーブルのデータ・コンテンツのループに新たに定義するデータ・コンテンツ内にエントリー・アプリケーションへの引き戻し制御の記述子 (`returnToEntry_descriptor`) を配置することで、データ・コンテンツ毎に提示単位 (PU) でアプリケーションの強制引き戻しを制御することができる。

【実施例4】

【0146】

第4の実施例では、M2セクション・メッセージで伝送するMH-AIT (アプリケーション情報テーブル) (図5を参照のこと) 内に、エントリー・アプリケーションへの引き戻し制御の記述子を配置することにより、エントリー・アプリケーションへの強制引き戻しを実現する。

20

【0147】

図14には、M2セクション・メッセージで伝送されるMH-AITのシンタックス例1400を示している。MH-AITは、アプリケーションに関する動的制御情報及び実行に必要な付加情報を伝送するテーブルであり、具体的には、MMT伝送路で送られてくるデータ放送アプリケーション (ファイル・データ) の処理方法 (アプリケーションに適用される起動状態など)、並びにロケーション (URL) を指定する。

【0148】

`table_id` (テーブル識別) は、各種シグナリング情報においてアプリケーション情報 (AI) テーブルであることを識別する8ビットの固定値であり、本実施形態では  $0x89$  とする。`section_syntax_indicator` (セクション・シンタックス指示) は、1ビットのフィールドで、常に「1」とする。`section_length` (セクション長) は、12ビットのフィールドで、セクション長フィールドからCRC32を含むセクションの最後までセクションのバイト長を規定する。この値は  $4093$  ( $16$ 進数で  $0xEF D$ ) を超えないものとする。`application_type` (アプリケーション形式) は、16ビットのフィールドで、AITで伝送しているアプリケーションの値を示す。DVBでは、DVB-Jアプリケーションに対して  $0x0001$  が割り当てられている。ARIB-Jアプリケーションにおいても  $0x0001$  とする。`version_number` (バージョン番号) は、5ビットのフィールドで、サブテーブルのバージョン番号である。`version_number` は、当該MH-AITテーブルのバージョン番号であり、サブテーブル内の情報に変化があった場合に+1だけインクリメントされる。また、バージョン番号の値が「31」になったとき、その次は「0」に戻る。`current_next_indicator` (カレント・ネクスト指示) は、常に「1」とする。`section_number` (セクション番号) は、8ビットのフィールドで、セクションの番号を表す。サブテーブル内で最初のセクションのセクション番号は  $0x00$  である。セクション番号は、同一のテーブル識別及びアプリケーション形式を持つセクションが追加される度に+1だけインクリメントされる。`last_section_number` (最終セクション番号) は、8ビットのフィールドであり、そのセクションが属するサブテーブルにおける最後のセクション番号を規定する。

30

40

50

【0149】

common\_descriptor\_length (共通記述子ループ長) は、8ビットのフィールドで、後続の descriptor (記述領域内記述子) のバイト長を示し、このバイト数分のループからなる一連の領域に descriptor (記述領域内記述子) が書き込まれる。この共通記述子領域内の descriptor は、AITサブテーブル内のすべてのアプリケーションに適用される。

【0150】

application\_loop\_length は、この MH AI テーブルに含まれるアプリケーション情報の数を書き込む領域である。そして、application\_loop\_length が示す数分だけ、アプリケーション情報のループが配置される。そして、当該テーブルの最後に、ITU-T 勧告 H.222.0 に従う巡回冗長符号 CRC32 (CRC) が付加される。

10

【0151】

1つのアプリケーション情報のループ内には、application\_identifier (アプリケーション識別子) と、application\_control\_code (アプリケーション制御コード) と、アプリケーション情報が配置される。

【0152】

application\_identifier (アプリケーション識別子) は、アプリケーションを識別するパラメータである。また、application\_control\_code (アプリケーション制御コード) は、8ビットのフィールドで、アプリケーションの状態を制御する制御コードを規定する。アプリケーション制御コードのセマンティックスは、アプリケーション形式の値に依存する。アプリケーション形式に依存しない場合のアプリケーション制御コードのセマンティックスを表1に示しておく。また、application\_descriptor\_loop\_length (アプリケーション情報記述子ループ長) はアプリケーション情報記述子のバイト長を示し、このバイト数分のループからなる一連の領域に descriptor (アプリケーション情報記述子) が書き込まれる。この記述子領域内のアプリケーション情報記述子は、共通記述子とは相違し、このループ内の application\_identifier で指定したアプリケーションのみに適用される。

20

【0153】

【表1】

30

アプリケーション制御コードの値	識別名	意味
0x01	AUTOSTART	アプリケーションを起動する。
0x02	PRESENT	アプリケーションが実行可能の状態であることを示す。
0x04	KILL	アプリケーションを終了する。
0x05	PREFETCH	アプリケーションを取得し保持する。

40

【0154】

図14に示すように、MH-AITのアプリケーション情報のループ内には、該当するアプリケーションの情報を格納するアプリケーション情報記述子(descriptor)が配置される。第4の実施例では、この記述子領域にエントリー・アプリケーションへの引き戻し制御の記述子を配置して、アプリケーション単位でアプリケーションの強制引き戻しを制御する。第4の実施例においては、2通りの引き戻し制御記述子のシンタックスが考えられる。第1の方法は、前述の図10に示す引き戻し制御記述子(return\_to\_entry\_descriptor)を適用する方法である。これにより、当該記述子により指定されたアプリケーションの実行を終了し自動起動(autostart)が

50

指定されたエントリー・アプリケーションを起動する。この場合の受信機側の動作について考察してみる。例えば時刻  $t_2$  に到達して、MH-AITが更新されると、受信機はその内容をチェックする。受信機は、時刻  $t_1 \sim t_2$  の期間中に番組非連動型アプリケーションである  $app_3$  の実行に切り替えているが、更新されたMH-AITでは、番組非連動型アプリケーションのアプリケーション情報のループに引き戻し制御記述子が配置されているので  $app_3$  を直ちに終了する。その結果、受信機は、時刻  $t_2$  以降においてエントリー・アプリケーションに指定されている  $app_6$  を、放送局の要求通りに自動起動することができる。本実施例は、変形例としてMH-AITのアプリケーション情報ループやアプリケーション制御記述子に存在する `reserved_for_future` のうちの1ビットを `returnToEntry` として利用することも考えられる。

10

【0155】

第2の方法は、図15に示す引き戻し制御の記述子 (`Forced_Return_descriptor`) を適用する方法である。

【0156】

`descriptor_tag` は、当該記述子1500を識別する、8ビットの整数値である。`descriptor_length` は、このフィールドより後に続く当該記述子1500のデータのバイト長を書き込む領域である。また、`application_identifier` は、アプリケーションの強制引き戻し時に、次に実行するアプリケーションを識別するアプリケーション識別情報を示す。

【0157】

20

MH-AITのアプリケーション情報のループ内では、アプリケーションの強制引き戻し時に終了すべきアプリケーションに対して引き戻し制御記述子 (`Forced_Return_descriptor`) を配置して、該当するアプリケーションに対して個別に終了を指示する。図8に示した動作例に当てはめて説明すると、番組非連動型アプリケーション  $app_3$  や  $app_4$  に対して引き戻し制御記述子 (`Forced_Return_descriptor`) を配置する。

【0158】

この場合の受信機側の動作について考察してみる。例えば時刻  $t_2$  に到達して、MH-AITが更新されると、受信機はその内容をチェックする。受信機は、時刻  $t_1 \sim t_2$  の期間中に番組非連動型アプリケーションである  $app_3$  の実行に切り替えているが、更新されたMH-AITでは、番組非連動型アプリケーション  $app_3$  のアプリケーション情報記述子の格納領域に引き戻し制御記述子 (`Forced_Return_descriptor`) が配置されているので、 $app_3$  を直ちに終了する。そして、受信機は、時刻  $t_2$  以降において、引き戻し制御記述子 (`Forced_Return_descriptor`) でアプリケーション識別情報が示されているアプリケーション  $app_6$  を、放送局の要求通りに自動起動することができる。

30

【0159】

このように、MH-AITを用いることで、アプリケーション単位でアプリケーションの強制引き戻しを制御することができる。

【実施例5】

40

【0160】

第5の実施例でも、第4の実施例と同様に、MH-AITに引き戻し制御記述子 (`Forced_Return_descriptor`) を配置することにより、エントリー・アプリケーションへの強制引き戻しを実現する。但し、第5の実施例では、特定のアプリケーションにのみ適用されるアプリケーション情報記述子ではなく、AITサブテーブル内のすべてのアプリケーションに適用される共通記述子 (`common_descriptor`) として引き戻し制御記述子 (`Forced_Return_descriptor`) を配置する。

【0161】

図16には、MH-AITの共通記述子に配置する引き戻し制御記述子 (`Forced`

50



`_Return_descriptor`) のシンタックス例 1600 を示している。

【0162】

`descriptor_tag` は、当該記述子 1600 を識別する、8 ビットの整数値である。`descriptor_length` は、このフィールドより後に続く当該記述子 1600 のデータのバイト長を書き込む領域である。`number_of_target_to_return` にアプリケーションの強制引き戻し時に終了することが指定されるアプリケーションの数が書き込まれる。そして、`number_of_target_to_return` の数分だけ配置されるループ内では、アプリケーションの強制引き戻し時に終了することが指定された各アプリケーションを識別するアプリケーション識別情報 (`application_identifier`) が書き込まれる。

10

【0163】

MH-AIT の共通記述子として引き戻し制御記述子 (`Forced_Return_descriptor`) を配置して、AIT サブテーブル内の図 8 に示した動作例に当てはめて説明すると、アプリケーションに対して個別に終了を指示する。図 8 に示した動作例に当てはめて説明すると、引き戻し制御記述子 (`Forced_Return_descriptor`) に、番組非連動型アプリケーション `app3` や `app4` のアプリケーション識別情報を示す。

【0164】

この場合の受信機側の動作について考察してみる。例えば時刻  $t_2$  に到達して、MH-AIT が更新されると、受信機はその内容をチェックする。受信機は、時刻  $t_1 \sim t_2$  の期間中に番組非連動型アプリケーションである `app3` の実行に切り替えているが、更新された MH-AIT に共通記述子として配置されている引き戻し制御記述子 (`Forced_Return_descriptor`) では、番組非連動型アプリケーション `app3` のアプリケーション識別情報が示されているので、`app3` を直ちに終了する。その結果、受信機は、時刻  $t_2$  以降においてエンターテインメント・アプリケーションに指定されている `app6` を、放送局の要求通りに自動起動することができる。

20

【0165】

このように、MH-AIT を用いることで、アプリケーション単位でアプリケーションの強制引き戻しを制御することができる。

【実施例 6】

30

【0166】

図 17 には、上述した第 1 ~ 第 5 の各実施例において、受信機がデータ放送アプリケーションの実行を制御するための処理手順をフローチャートの形式で示している。

【0167】

放送番組のオンエア中、M2 セクション・メッセージ並びにデータ伝送メッセージを含む各シグナリング・メッセージは、基本的にカルセル伝送されており、受信機はシグナリング・メッセージから最新のシグナリング・テーブルを取得する (ステップ S1701)。受信機が取得するシグナリング・テーブルは、MH-AIT やデータ・アセット管理テーブル、データ・コンテンツ管理テーブルを含むものとする。

【0168】

40

そして、受信機は、データ放送アプリケーションを起動する (ステップ S1702)。受信機は、例えば、MH-AIT で自動起動 (`autostart`) が指定されたアプリケーションを自動起動する。また、受信機は、ユーザー (放送番組の視聴者) のリモコン操作などで指定されたアプリケーションを実行する。なお、以下では、説明の簡素化のため、ユーザーが起動中のアプリケーションの終了を指示することはないものとする。

【0169】

シグナリング・テーブルが更新されない限り (ステップ S1703 の No)、受信機は、アプリケーションの動作を継続する。

【0170】

ここで、MH-AIT やデータ・アセット管理テーブル、データ・コンテンツ管理テー

50

ブルなどのシグナリング・テーブルが更新されると(ステップS 1703のYes)、動作中のアプリケーションに対して、MH-AIT内で「kill(終了)」が指定されるなどの記述がないかをチェックする(ステップS 1704)。

【0171】

また、更新されたシグナリング・テーブルに動作中のアプリケーションに対する「kill」などの記述がない場合には(ステップS 1704のNo)、動作中のアプリケーションに対してエントリー・アプリケーションへの強制引き戻しのために引き戻し制御記述子に基づいて終了が指示されているかどうかをチェックする(ステップS 1705)。

【0172】

そして、動作中のアプリケーションに対して、MH-AIT内で「kill(終了)」が指定されている場合や(ステップS 1704のYes)、引き戻し制御記述子に基づいて動作中のアプリケーションの終了が指示されている場合には(ステップS 1705のYes)、動作中のアプリケーションを終了する(ステップS 1706)。

【0173】

動作中のアプリケーションを終了させた後、自動起動するアプリケーション(エントリー・アプリケーション)が指定されている場合には(ステップS 1707のYes)、受信機は、そのアプリケーションを実行する。

【産業上の利用可能性】

【0174】

以上、特定の実施形態を参照しながら、本明細書で開示する技術について詳細に説明してきた。しかしながら、本明細書で開示する技術の要旨を逸脱しない範囲で当業者が該実施形態の修正や代用を成し得ることは自明である。

【0175】

本明細書で開示する技術は、トランスポート方式としてMMTを採用するさまざまな放送システムに適用することができる。また、本明細書で開示する技術は、データ放送アプリケーションの伝送に関する制御情報を伝送するさまざまな放送システムに適用することができる。

【0176】

要するに、例示という形態により本明細書で開示する技術について説明してきたのであり、本明細書の記載内容を限定的に解釈するべきではない。本明細書で開示する技術の要旨を判断するためには、特許請求の範囲を参酌すべきである。

【0177】

なお、本明細書の開示の技術は、以下のような構成をとることも可能である。

(1) 放送サービスを構成する各コンポーネントを所定のトランスポート方式に基づく伝送単位にして送信する送信部と、  
前記コンポーネントの伝送に関する制御情報を送信する情報送信部と、  
を具備し、  
前記情報送信部は、前記送信部から送信されるデータ放送アプリケーションの引き戻しに関する制御情報を送信する、  
送信装置。

(2) 前記所定のトランスポート方式はMMTである、  
上記(1)に記載の送信装置。

(3) 前記情報送信部は、前記送信部から送信される映像及び音声コンポーネントで構成される放送番組に連動してエントリー・アプリケーションへの引き戻しを指示する制御情報を送信する、  
上記(1)に記載の送信装置。

(4) 前記情報送信部は、前記送信部が送信するアプリケーションのコンポーネントの伝送に関する制御情報を示す第1のテーブル内で、エントリー・アプリケーションへの引き戻しを指示する引き戻し制御記述子をコンポーネント単位で格納して送信する、  
上記(1)に記載の送信装置。

10

20

30

40

50

(5) 前記情報送信部は、引き戻し対象をコンポーネント単位で示す前記引き戻し制御記述子をコンポーネント毎に格納した前記第1のテーブルを送信する、

上記(4)に記載の送信装置。

(6) 前記情報送信部は、データ放送アプリケーションの提示単位に関する情報を示す第2のテーブル内でエントリー・アプリケーションへの引き戻しを指示する引き戻し制御記述子を提示単位で格納して送信する、

上記(1)に記載の送信装置。

(7) 前記情報送信部は、引き戻し対象を提示単位で示す前記引き戻し制御記述子をデータ・コンテンツ毎に格納した前記第2のテーブルを送信する、

上記(6)に記載の送信装置。

(8) 前記情報送信部は、データ・コンテンツ毎にデータ放送アプリケーションに関する情報を示す第3のテーブル内でエントリー・アプリケーションへの引き戻しを指示する引き戻し制御記述子をデータ・コンテンツ単位で格納して送信する、

上記(1)に記載の送信装置。

(9) 前記送信部は、データ放送アプリケーションを構成するファイルを前記所定のトランスポート方式に基づく伝送単位にグルーピングして送信し、

前記情報送信部は、引き戻し対象を伝送単位で示す前記引き戻し制御記述子をデータ・コンテンツ毎に格納した前記第3のテーブルを送信する、

上記(8)に記載の送信装置。

(10) 前記情報送信部は、アプリケーションの動作制御を示す第4のテーブル内でエントリー・アプリケーションへの引き戻しを指示する引き戻し制御記述子をアプリケーション単位で格納して送信する、

上記(1)に記載の送信装置。

(11) 前記情報送信部は、次に起動するアプリケーションを指定する前記引き戻し制御記述子をアプリケーション毎に格納した前記第4のテーブルを送信する、

上記(10)に記載の送信装置。

(12) 前記情報送信部は、アプリケーションの動作制御を示す第4のテーブル内のすべてのアプリケーションに適用される共通記述子としてエントリー・アプリケーションへの引き戻しを指示する引き戻し制御記述子を格納して送信する、

上記(1)に記載の送信装置。

(13) 前記情報送信部は、エントリー・アプリケーションへの引き戻しのために終了すべきアプリケーションの識別情報を示す前記引き戻し制御記述子を前記共通記述子として配置した前記第4のテーブルを送信する、

上記(12)に記載の送信装置。

(14) 放送サービスを構成する各コンポーネントを所定のトランスポート方式に基づく伝送単位にして送信する送信ステップと、

前記コンポーネントの伝送に関する制御情報を送信する情報送信ステップと、を有し、

前記情報送信ステップは、前記送信ステップにおいて送信するデータ放送アプリケーションの引き戻しに関する制御情報を送信する、送信方法。

(15) 所定のトランスポート方式に基づく伝送単位で伝送される放送サービスの各コンポーネントを受信する受信部と、

前記コンポーネントの伝送に関する制御情報を受信する情報受信部と、を具備し、

前記情報受信部は、前記受信部で受信するデータ放送アプリケーションの引き戻しに関する制御情報を受信する、受信装置。

(16) 所定のトランスポート方式に基づく伝送単位で伝送される放送サービスの各コンポーネントを受信する受信ステップと、

10

20

30

40

50

前記コンポーネントの伝送に関する制御情報を受信する情報受信ステップと、  
を有し、

前記情報受信ステップでは、前記受信ステップで受信するデータ放送アプリケーション  
の引き戻しに関する制御情報を受信する、  
受信方法。

【符号の説明】

【 0 1 7 8 】

1 0 ... デジタル放送システム

1 1 ... 放送送出システム、 1 2 ... 受信機

3 0 1 ... 時計部、 3 0 2 ... 信号送出部

10

3 0 3 ... ビデオ・エンコーダー、 3 0 4 ... オーディオ・エンコーダー

3 0 5 ... キャプション・エンコーダー

3 0 6 ... シグナリング・エンコーダー

3 0 7 ... ファイル・エンコーダー、 3 0 8 ... E D P S

3 0 9 ... T L V シグナリング・エンコーダー

3 1 0 ... I P サービス・マルチプレクサー

3 1 1 ... T L V マルチプレクサー、 3 1 2 ... 変調・送信部

4 0 1 ... チューナー・復調部、 4 0 2 ... デマルチプレクサー

4 0 2 - 1 ... T L V フィルター、 4 0 2 - 2 ... I P フィルター

20

4 0 2 - 3 ... U D P フィルター、 4 0 2 - 4 ... M M T フィルター

4 0 2 - 5 ... S I フィルター、 4 0 3 ... 時計回復部

4 0 4 ... ビデオ・デコーダー、 4 0 5 ... オーディオ・デコーダー

4 0 6 ... キャプション・デコーダー、 4 0 7 ... システム制御部

4 0 8 ... アプリケーション制御部

4 0 9 ... データ放送アプリケーション・エンジン

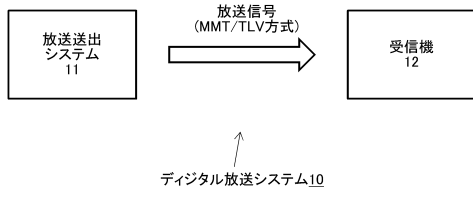
4 1 0 ... I P インターフェース、 4 1 1 ... 合成部

30

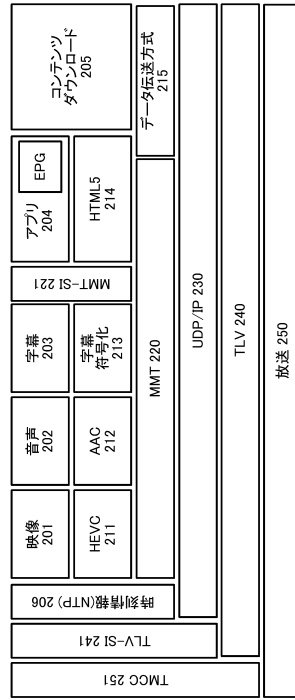
40

50

【図面】  
【図 1】



【図 2】



10

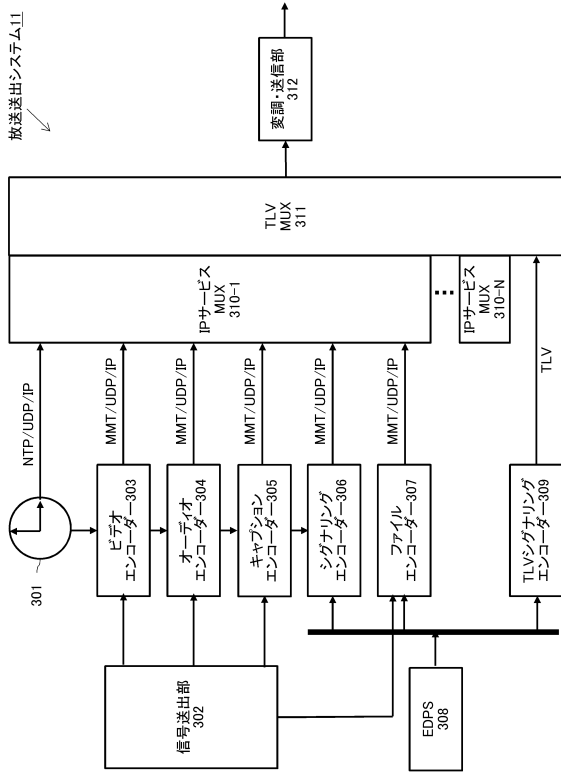
20

30

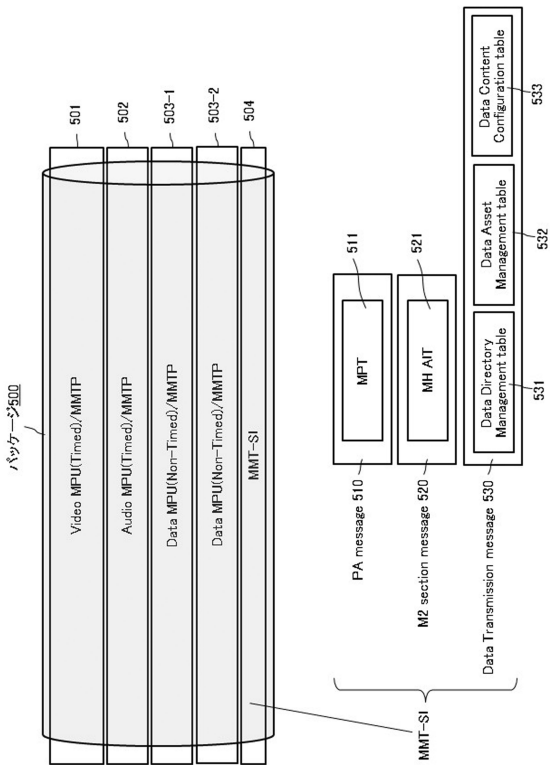
40

50

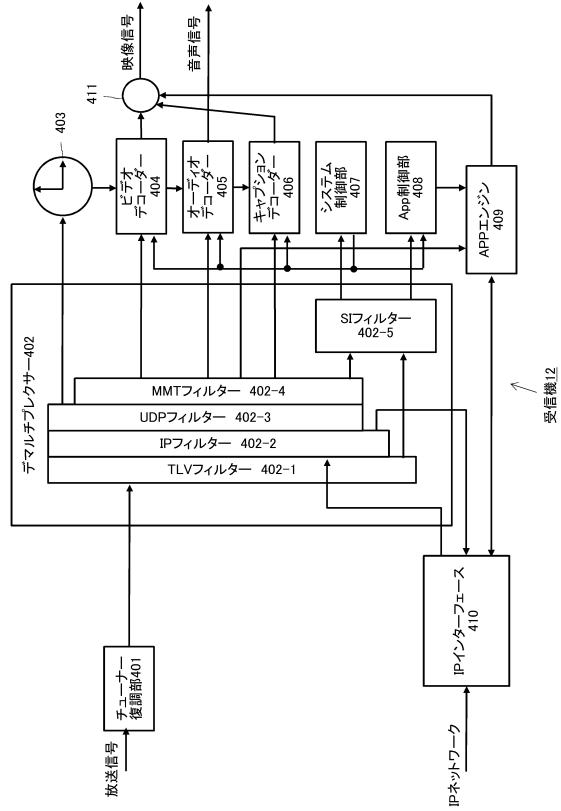
【図 3】



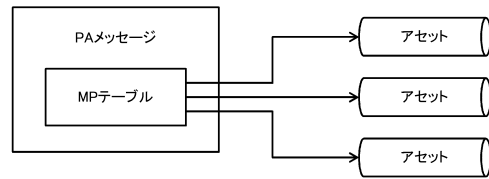
【図 5】



【図 4】



【図 6】



10

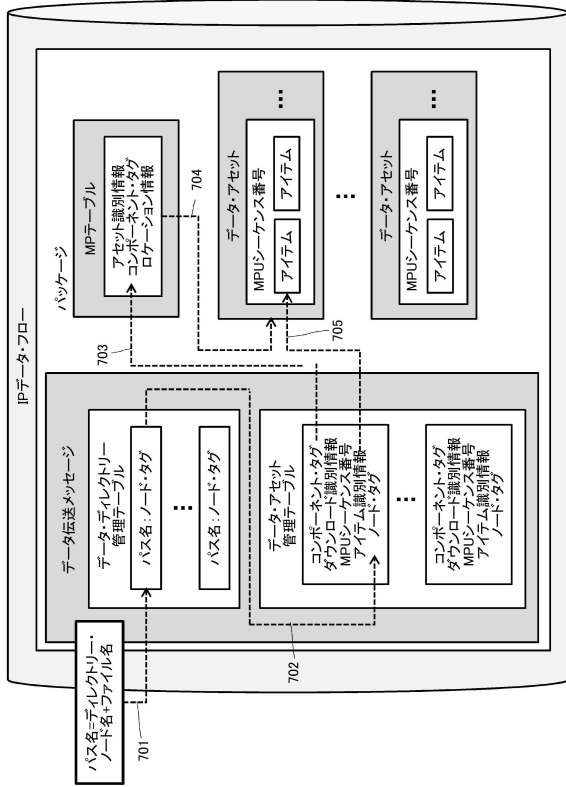
20

30

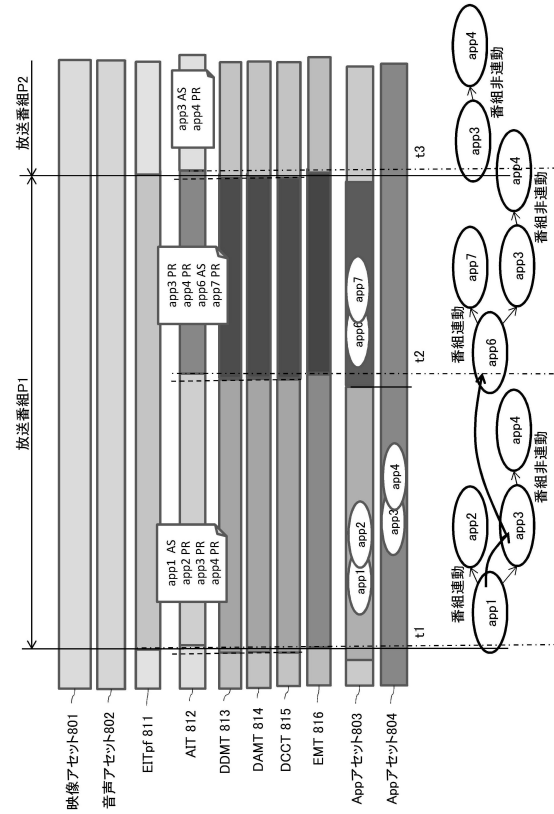
40

50

【 図 7 】



【 図 8 】



10

20

【 図 9 】

データ・アセット管理テーブル900

データ構造	ビット数	データ表記
Data_Asset_Management_Table{		
table_id	8	uimbsf
version	8	uimbsf
length	16	uimbsf
number_of_data_components	8	uimbsf
for(i=0; i<number_of_data_components; i++){		
transaction_id	32	uimbsf
component_tag	16	uimbsf
download_id	32	uimbsf
num_of_mpus	8	uimbsf
for(j=0; j<num_of_mpus; j++){		
mpu_sequence_number	32	uimbsf
num_of_items	8	uimbsf
for(k=0; k<num_of_items; k++){		
item_ID	32	uimbsf
node_tag	16	uimbsf
item_size	32	uimbsf
item_version	8	uimbsf
checksum_flag	1	bslbf
reserved_future_flag	7	bslbf
if (checksum_flag==1){		
item_check_sum	32	uimbsf
}		
item_info_length	8	uimbsf
for(l=0; l<item_info_length; l++){		
item_info_byte	8	uimbsf
}		
mpu_info_length	8	uimbsf
for(m=0; m<mpu_info_length; m++){		
mpu_info_byte	8	uimbsf
}		
}		
descriptor_loop_length	8	uimbsf
for(n=0; n<N; n++){		
descriptor()	16	uimbsf
}		
}		

引き戻し制御記述子を配置

【 図 10 】

引き戻し制御記述子1000

データ構造	ビット数	データ表記
returnToEntry_descriptor {		
descriptor_tag	16	uimbsf
descriptor_length	8	uimbsf
return_to_entry_flag	1	bslbf
reserved_future_use	7	bslbf
}		

30

40

50

【 図 1 1 】

データ・コンテンツ管理テーブル1100

データ構造	ビット数	データ表記
Data_Content_Configuration_table ()		
table_id	8	uimsbf
version	8	uimsbf
length	16	uimsbf
number_of_contents	8	uimsbf
for (i=0; i<number_of_contents; i++){		
content_id	16	uimsbf
content_version	8	uimsbf
content_size	32	uimsbf
PU_info_flag	1	bslbf
reserved_for_future_use	7	bslbf
if (PU_info_flag==1){		
number_of_PUs	8	uimsbf
for(j=0; j<number_of_PUs; j++){		
PU_tag	8	uimsbf
PU_size	32	uimsbf
number_of_member_nodes	8	uimsbf
for(k=0; k<number_of_member_nodes; k++){		
node_tag	16	uimsbf
}		
PU_info_length	8	uimsbf
for(k=0; k<PU_info_length; k++){		
PU_info_byte	8	uimsbf
}		
}		
}		
else {		
number_of_nodes	16	uimsbf
for (l=0; l<number_of_nodes; l++){		
node_tag	16	uimsbf
}		
}		

引き戻し制御記述子(図10を参照のこと)を配置

【 図 1 2 】

データ・コンテンツ管理テーブル1200

データ構造	ビット数	データ表記
Data_Content_Configuration_table ()		
table_id	8	uimsbf
version	8	uimsbf
length	16	uimsbf
number_of_contents	8	uimsbf
for (i=0; i<number_of_contents; i++){		
content_id	16	uimsbf
content_version	8	uimsbf
content_size	32	uimsbf
PU_info_flag	1	bslbf
content_info_flag	1	bslbf
reserved_for_future_use	6	bslbf
if (PU_info_flag==1){		
number_of_PUs	8	uimsbf
for(j=0; j<number_of_PUs; j++){		
PU_tag	8	uimsbf
PU_size	32	uimsbf
number_of_member_nodes	8	uimsbf
for(k=0; k<number_of_member_nodes; k++){		
node_tag	16	uimsbf
}		
PU_info_length	8	uimsbf
for(k=0; k<PU_info_length; k++){		
PU_info_byte	8	uimsbf
}		
}		
}		
else {		
number_of_nodes	16	uimsbf
for (l=0; l<number_of_nodes; l++){		
node_tag	16	uimsbf
}		
}		
if (content_info_flag==1){		
content_info_length	8	uimsbf
for (m=0; m<content_info_length; m++){		
content_info_byte	8	uimsbf
}		

引き戻し制御記述子を配置

【 図 1 3 】

引き戻し制御記述子1300

データ構造	ビット数	データ表記
returnToEntry_descriptor ()		
descriptor_tag	16	uimsbf
descriptor_length	8	uimsbf
PU_num_of_target_to_return	8	uimsbf
for (i=0; i<PU_num_of_target_to_return; i++){		
PU_tag	8	bslbf
}		

【 図 1 4 】

MH-AIT1400

データ構造	ビット数	データ表記
MH-Applicatin_Information_Table()		
table_id	8	uimsbf
section_syntax_indicator	1	bslbf
reserved_future_use	1	bslbf
reserved	2	bslbf
section_length	12	uimsbf
application_type	16	uimsbf
reserved	2	bslbf
version_number	5	uimsbf
current_next_indicator	1	bslbf
section_number	8	uimsbf
last_section_number	8	uimsbf
reserved_future_use	4	bslbf
common_descriptor_length	12	uimsbf
for(i=0; i<N; i++){		
descriptor ()		
}		
reserved_future_use	4	bslbf
application_loop_length	12	uimsbf
for(i=0; i<N; i++){		
application_identifier ()		
application_control_code	8	uimsbf
reserved_future_use	4	bslbf
application_descriptor_loop_length	12	uimsbf
for (j=0; j<M; j++){		
descriptor ()		
}		
}		
CRC32	32	rpchof

引き戻し制御記述子を配置

10

20

30

40

50





## フロントページの続き

株式会社内

審査官 長谷川 素直

- (56)参考文献 特開2013-009351(JP,A)  
国際公開第2013/080450(WO,A1)  
デジタル放送におけるMMTによるメディアトランスポート方式、標準規格 ARIB STD-B60  
，1.1版，日本，一般社団法人 電波産業会，2014年12月16日，p.124-146
- (58)調査した分野 (Int.Cl., DB名)  
H04N 21/00 - 21/858  
H04H 20/00 - 60/98