

(19) 日本国特許庁(JP)

(12) 特 許 公 報(B2)

(11) 特許番号

特許第6247309号
(P6247309)

(45) 発行日 平成29年12月13日(2017.12.13)

(24) 登録日 平成29年11月24日(2017.11.24)

(51) Int.Cl.		F I
HO4N 21/235 (2011.01)		HO4N 21/235
HO4N 21/6332 (2011.01)		HO4N 21/6332
HO4N 21/435 (2011.01)		HO4N 21/435

請求項の数 4 (全 165 頁)

(21) 出願番号	特願2015-543996 (P2015-543996)	(73) 特許権者	502032105
(86) (22) 出願日	平成25年11月26日 (2013.11.26)		エルジー エレクトロニクス インコーポ レイティド
(65) 公表番号	特表2016-506114 (P2016-506114A)		大韓民国ソウル、ヨンドンポーク、ヨイ ーデロ、128
(43) 公表日	平成28年2月25日 (2016.2.25)	(74) 代理人	100099759
(86) 国際出願番号	PCT/KR2013/010788		弁理士 青木 篤
(87) 国際公開番号	W02014/084570	(74) 代理人	100123582
(87) 国際公開日	平成26年6月5日 (2014.6.5)		弁理士 三橋 真二
審査請求日	平成27年5月26日 (2015.5.26)	(74) 代理人	100165191
(31) 優先権主張番号	61/730, 937		弁理士 河合 章
(32) 優先日	平成24年11月28日 (2012.11.28)	(74) 代理人	100114018
(33) 優先権主張国	米国 (US)		弁理士 南山 知広
(31) 優先権主張番号	61/736, 529	(74) 代理人	100159259
(32) 優先日	平成24年12月12日 (2012.12.12)		弁理士 竹本 実
(33) 優先権主張国	米国 (US)		
前置審査			最終頁に続く

(54) 【発明の名称】 双方向サービス処理する装置及び方法

(57) 【特許請求の範囲】

【請求項1】

プライマリデバイスで放送信号を処理する方法であって、
前記プライマリデバイスが提供するUPnPサービスを広告するディスカバリメッセージをマルチキャストするステップであって、前記UPnPサービスはトリガサービスと双方向通信サービスを含む、ステップと、

コンパニオンデバイスでのセコンダリアプリケーションから前記UPnPサービスの記述 (description) に対する要求を受信するステップと、

前記UPnPサービスを記述するために前記セコンダリアプリケーションのために利用される前記記述を前記セコンダリアプリケーションに送信するステップと、

放送ストリーム又はインターネットサーバーから活性化トリガを受信するステップであって、前記活性化トリガは、第1のタームと第2のタームを含む、ステップと、

前記第1のタームは、アプリケーションテーブルの中のアプリケーションID、イベントID及びデータIDを参照して、前記アプリケーションテーブルからターゲットイベントに関連するターゲット情報を示し、

前記第2のタームは、前記ターゲットイベントのための活性化時間を示し、

前記アプリケーションテーブルは、アプリケーションエレメントを含み、前記アプリケーションエレメントのそれぞれは、アプリケーションID及びアプリケーションのメタデータを含み、

前記アプリケーションテーブルは、イベントエレメントを含み、前記イベントエレ

10

20

ントのそれぞれは、イベントIDとイベントのメタデータを含み、

前記アプリケーションテーブルは、データエレメントを含み、前記データエレメントのそれぞれは、データIDとイベントに利用されるデータセットを含み、

前記活性化トリガ及び前記アプリケーションテーブルを利用して増加したトリガを生成するステップであって、前記増加したトリガは、前記第1のタームにより前記アプリケーションテーブルに基づいて示される前記ターゲットイベントに関連したターゲット情報を含む、ステップと、

前記ターゲット情報は、前記第1のタームに参照されるアプリケーションエレメントのアプリケーションIDに基づく、前記ターゲットイベントによりターゲットとされるアプリケーションのメタデータをさらに含み、

10

前記ターゲット情報は、前記第1のタームに参照されるイベントエレメントのイベントIDに基づく、前記ターゲットイベントのメタデータを更に含み、

前記ターゲット情報は、前記ターゲットイベントにより利用され、前記第1のタームに参照されるデータエレメントのデータIDに基づく、データセットを更に含み、

前記トリガサービスを利用して前記セコンダリアプリケーションに前記増加したトリガを伝達するステップであって、前記増加したトリガは前記活性化トリガの前記第2のタームにより示された前記活性化時間で伝達される、ステップと、を含む、方法。

【請求項2】

前記双方向通信サービスを利用して前記セコンダリアプリケーションに状態情報を伝達するステップであって、前記状態情報は前記セコンダリアプリケーションとの通信に参加するように用意された前記プライマリデバイスで実行するプライマリアプリケーションがあるか否かを示す、ステップと、

20

前記セコンダリアプリケーションから前記通信のための要求を受信するステップと、前記プライマリアプリケーションと前記セコンダリアプリケーションの間に伝達されるデータを介する通信を確立するステップとをさらに含む、請求項1に記載の方法。

【請求項3】

プライマリデバイスとして放送サービスを処理する装置であって、

前記プライマリデバイスが提供するUPnPサービスを広告するディスカバリメッセージをマルチキャストするように構成されたネットワークインターフェースであって、前記UPnPサービスはトリガサービスと双方向通信サービスを含む、ネットワークインターフェースと、

30

前記ネットワークインターフェースは、コンパニオンデバイスにおいてセコンダリアプリケーションから前記UPnPサービスの記述のための要求を受信するようにさらに構成され、

前記ネットワークインターフェースは、前記UPnPサービスを記述するために前記セコンダリアプリケーションで利用される記述を、前記セコンダリアプリケーションに送るようにさらに構成され、

放送ストリーム又はインターネットサーバーから活性化トリガを受信するように構成されたチューナであって、前記活性化トリガは、第1のタームと第2のタームを含む、チューナと、

40

前記第1のタームは、アプリケーションテーブルの中のアプリケーションID、イベントID及びデータIDを参照して前記アプリケーションテーブルからターゲットイベントに関連するターゲット情報を示し、

前記第2のタームは、前記ターゲットイベントのための活性化時間を示し、

前記アプリケーションテーブルは、アプリケーションエレメントを含み、前記アプリケーションエレメントのそれぞれは、アプリケーションIDとアプリケーションのメタデータを含み、

前記アプリケーションテーブルは、イベントエレメントを含み、前記イベントエレメントのそれぞれは、イベントIDとイベントのメタデータを含み、

前記アプリケーションテーブルは、データエレメントを含み、前記データエレメント

50

のそれぞれは、データIDとイベントに利用されるデータセットを含み、

前記活性化トリガ及び前記アプリケーションテーブルを利用して増加したトリガを生成するように構成されたトリガモジュールであって、前記増加したトリガは、前記第1のチームにより前記アプリケーションテーブルに基づいて示される前記ターゲットイベントに関連したターゲット情報を含む、トリガモジュールとを含み、

前記ターゲット情報は、前記第1のチームに参照されるアプリケーションエレメントのアプリケーションIDに基づく、前記ターゲットイベントのターゲットとされる前記アプリケーションのメタデータをさらに含み、

前記ターゲット情報は、前記第1のチームに参照されるイベントエレメントのイベントIDに基づく、前記ターゲットイベントのメタデータをさらに含み、

前記ターゲット情報は、前記ターゲットイベントに利用され、前記第1のチームに参照されるデータエレメントのデータIDに基づく、データセットを更に含み、

前記ネットワークインターフェースは、前記トリガサービスを利用して前記増加したトリガを前記セコンダリアプリケーションに伝送するようにさらに構成され、前記増加したトリガは前記活性化トリガの前記第2のチームに示された前記活性化時間で伝送される、装置。

【請求項4】

前記ネットワークインターフェースは、前記双方向通信サービスを利用して前記セコンダリアプリケーションに状態情報を配信するようにさらに構成され、前記状態情報は前記セコンダリアプリケーションとの通信に参加するように用意された前記プライマリデバイスで実行するプライマリアプリケーションがあるか否かを示し、

前記ネットワークインターフェースは、前記セコンダリアプリケーションから前記通信のための要求を受信し、前記プライマリアプリケーションと前記セコンダリアプリケーションの間に配信されるデータを介する通信を確立するように構成された、請求項3に記載の装置。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、放送サービスを提供、受信及び処理する方法及び装置に関し、特に、放送コンテンツに関連する付加サービスを提供する方法及び装置に関する。

【背景技術】

【0002】

TVは、19世紀末に初めて登場して以来、画面表示方法やデザインなどにおいて発展を遂げ続き、20世紀後半からは最も大衆的な情報伝達機器として確立された。ただし、TVは、放送局から提供する単方向の情報を視聴者たちが一方的に受容するという根本的な形態をしていた。このようなTVの限界は、1990年代からPC(個人用コンピュータ)及びインターネットが大衆化するにつれてより明らかになり始めた。そこで、TVは双方向サービスが可能な方向に進化を続けてきている。

【0003】

しかしながら、現在、コンテンツプロバイダと視聴者との間には、上記のような双方向サービスを本格的に実行するシステムがない実情である。特に、上記のような双方向サービスの提供のために、放映中の放送コンテンツに関連するアプリケーションを特定時点に駆動し、情報の特定処理を通じて、視聴者に関連情報を提供する方法及びフォーマットに関する技術的支援が必要な状況である。

【発明の概要】

【発明が解決しようとする課題】

【0004】

本発明が遂げようとする技術的課題は、前述した問題点を解決するためのもので、放送コンテンツが再生される期間中に、適切な時期に放送コンテンツに関連する付加情報を提供することにある。

10

20

30

40

50

【課題を解決するための手段】

【0005】

本発明の目的は、デジタルサービス信号を処理する方法であり、前記デジタルサービス信号に対するプライマリアプリケーションを実行するステップであって、前記プライマリアプリケーションは、トリガーされたアプリケーション又はトリガーされていないアプリケーションである、ステップと、前記プライマリアプリケーションに関連したセカンダリアプリケーションに対するインタラクティブ（双方向）サービスのメッセージを送信するステップであって、前記インタラクティブサービスは、前記トリガーされたアプリケーションに関連した第1モード及び前記トリガーされていないアプリケーションに関連した第2モードを含む、ステップと、前記インタラクティブサービスを識別する記述（*description*）に対する要求を受信するステップと、前記インタラクティブサービスのそれぞれを識別するサービス*id*情報及び前記インタラクティブサービスのそれぞれのタイプを示すサービスタイプ情報を有する記述を送信するステップと、前記インタラクティブサービスに対する情報を送信することによって前記インタラクティブサービスを提供するステップを含む方法によって達成される。

10

【0006】

好適には、前記インタラクティブサービスに対する情報は、前記インタラクティブサービスのそれぞれによって定義され、前記インタラクティブサービスに対する情報は、トリガー、バイト及びURL情報のうちの一つである。

【0007】

好適には、前記インタラクティブサービスに対する情報がトリガーであれば、前記トリガーの少なくとも一つは、前記デジタルサービス信号のチャンネルの変更された番号を示す値を有するチャンネル変更情報を含む。

20

【0008】

好適には、前記インタラクティブサービスに対する情報がトリガーであれば、前記方法は、TPT内の情報と結合することによって前記トリガーを増加（*augment*）させるステップであって、前記TPTは、前記トリガーされたアプリケーションに対するメタデータを含む、ステップと、前記トリガー内で識別されたイベントを示すイベント指示子を有する増加したトリガーを送信するステップをさらに含む。

【0009】

好適には、前記インタラクティブサービスに対する情報がバイトであれば、前記インタラクティブサービスは、前記アプリケーションが実行されて前記セカンダリアプリケーションとの通信に参加する準備ができているかを示すステータス（*status*）状態変数を含み、前記方法は、前記*status*状態変数を設定するステップをさらに含む。

30

【0010】

好適には、前記プライマリアプリケーションは、APIを用いてインタラクティブサービスを提供する。

【0011】

好適には、前記APIは、バイト及びバイト引数（*argument*）に対するアドレス及びポートに関する情報を有するアドレス引数を含む。

40

【0012】

好適には、前記方法は、前記メッセージを送信する前に、前記プライマリアプリケーションに関連した前記セカンダリアプリケーションに対するインタラクティブサービスのメッセージに対する要求を受信するステップをさらに含む。

【0013】

本発明の他の形態において、デジタルサービス信号を処理する方法であって、前記デジタルサービス信号に対するアプリケーションを実行するステップと、前記実行されたアプリケーションに対するインタラクティブサービスのメッセージを受信するステップと、前記インタラクティブサービスを識別する記述（*description*）に対する要求を送信するステップと、前記インタラクティブサービスのそれぞれを識別するサービス*id*

50

情報及び前記インタラクティブサービスのそれぞれのタイプを示すサービスタイプ情報を有する記述を受信するステップと、前記記述を用いて少なくとも一つのインタラクティブサービスをアクセスするステップと、前記アクセスされたインタラクティブサービスに対するファイルをダウンロードするステップを含み、前記アクセスするステップは、前記少なくとも一つのインタラクティブサービスに対する通知のためのプロトコルに加入するステップと、前記通知を受信するステップと、前記受信された通知を用いて前記少なくとも一つのインタラクティブサービスに対する情報を受信するステップとを含む、方法が提供される。

【0014】

好適には、前記少なくとも一つのインタラクティブサービスに対する情報は、前記インタラクティブサービスによって定義され、前記少なくとも一つのインタラクティブサービスに対する情報は、トリガー、バイト及びURL情報のうちの一つである。

10

【0015】

好適には、前記探索されたインタラクティブサービスに対する情報がトリガーであれば、前記トリガーの少なくとも一つは、前記デジタルサービス信号のチャンネルの変更された番号を示す値を有するチャンネル変更情報を含む。

【0016】

好適には、前記アクセスされたインタラクティブサービスに対する情報がバイトであれば、前記方法は、前記受信された通知を用いて前記インタラクティブサービスが始まる準備ができているかを決定するステップと、前記受信された通知を用いてバイトを送信するための接続に対する要求を送信するステップをさらに含む。

20

【0017】

好適には、前記方法は、前記メッセージを受信する前に、前記実行されたアプリケーションに対するインタラクティブサービスのメッセージに対する要求を送信するステップをさらに含む。

【0018】

本発明の他の形態において、デジタルサービス信号を処理する装置であって、前記デジタルサービス信号に対するプライマリアプリケーションを実行するように構成された実行モジュールであって、前記プライマリアプリケーションは、トリガーされたアプリケーション又はトリガーされていないアプリケーションである、モジュールと、前記プライマリアプリケーションに関連したセカンダリアプリケーションに対するインタラクティブサービスのメッセージ及び前記インタラクティブサービスを識別する記述(d e s c r i p t i o n)を送信するように構成された送信モジュールであって、前記インタラクティブサービスモジュールは、前記トリガーされたアプリケーションに関連した第1モード又は前記トリガーされていないアプリケーションに関連した第2モードを含み、前記記述は要求によって送信される、送信モジュールと、前記インタラクティブサービスのそれぞれを識別するサービスid情報及び前記インタラクティブサービスのそれぞれのタイプを示すサービスタイプ情報を有する記述に対する要求を受信するように構成された受信モジュールと、前記インタラクティブサービスに対する情報を送信することによって前記インタラクティブサービスを提供するように構成された提供モジュールを備える装置が提供される。

30

40

【0019】

好適には、前記インタラクティブサービスに対する情報は、前記インタラクティブサービスによって定義され、前記インタラクティブサービスに対する情報は、トリガー、バイト及びURL情報のうちの一つである。

【0020】

好適には、前記インタラクティブサービスに対する情報がトリガーであれば、前記トリガーの少なくとも一つは、前記デジタルサービス信号のチャンネルの変更された番号を示す値を有するチャンネル変更情報を含む。

【0021】

好適には、前記インタラクティブサービスに対する情報がトリガーであれば、前記装置

50

は、TPT内の情報と結合することによって前記トリガーを増加(augment)させるように構成された増加モジュールをさらに備え、前記TPTは、前記トリガーされたアプリケーションに対するメタデータを含み、前記送信モジュールはさらに、前記トリガー内で識別されたイベントを示すイベント指示子を有する増加したトリガーを送信する。

【0022】

好適には、前記インタラクティブサービスに対する情報がバイトであれば、前記インタラクティブサービスは、前記アプリケーションが実行されて前記セカンダリアプリケーションとの通信に参加する準備ができているかを示すstatus状態変数を含み、前記装置はさらに、前記status状態変数を設定することを含む。

【0023】

好適には、前記プライマリAPIを用いてインタラクティブサービスを提供する。

【0024】

好適には、前記APIは、バイト及びバイト引数(argument)に対するアドレス及びポートに関する情報を有するアドレス引数を含む。

【0025】

好適には、前記受信モジュールはさらに、前記送信モジュールが前記メッセージを送信する前に、前記プライマリアプリケーションに関連した前記セカンダリアプリケーションに対するインタラクティブサービスのメッセージに対する要求を受信するように構成される。

【0026】

本発明の他の形態において、デジタルサービス信号を処理する装置であって、前記デジタルサービス信号に対するアプリケーションを実行するように構成された実行モジュールと、前記実行されたアプリケーションに対するインタラクティブサービスのメッセージ及び記述(description)を受信するように構成された受信モジュールと、前記インタラクティブサービスを識別する記述に対する要求を伝送するように構成された伝送モジュールであって、前記記述は、前記インタラクティブサービスのそれぞれを識別するサービスid情報及び前記インタラクティブサービスのそれぞれのタイプを示すサービスタイプ情報を有する、伝送モジュールと、前記記述を用いて少なくとも一つのインタラクティブサービスをアクセスするように構成されたアクセスモジュールと、前記アクセスされたインタラクティブサービスに対するファイルをダウンロードするように構成されたダウンロードモジュールを備え、前記アクセスモジュールは、前記少なくとも一つのインタラクティブサービスに対する通知のためのプロトコルに加入し、前記通知を受信し、前記受信された通知を用いて前記少なくとも一つのインタラクティブサービスに対する情報を受信する、装置が提供される。

【0027】

好適には、前記少なくとも一つのインタラクティブサービスに対する情報は、前記インタラクティブサービスによって定義され、前記少なくとも一つのインタラクティブサービスに対する情報は、トリガー、バイト及びURL情報のうちの一つである。

【0028】

好適には、前記探索されたインタラクティブサービスに対する情報がトリガーであれば、前記トリガーの少なくとも一つは、前記デジタルサービス信号のチャンネルの変更された番号を示す値を有するチャンネル変更情報を含む。

【0029】

好適には、前記アクセスされたインタラクティブサービスに対する情報がバイトであれば、前記アクセスモジュールはさらに、前記受信された通知を用いて前記インタラクティブサービスが始まる準備ができているかを決定し、前記受信された通知を用いてバイトを送信するための接続に対する要求を送信する。

【0030】

好適には、前記伝送モジュールは、前記受信モジュールが前記メッセージを受信する前に、前記実行されたアプリケーションに対するインタラクティブサービスのメッセージに

10

20

30

40

50

対する要求を伝送するように構成される。

【0031】

上述した一般的な説明と次の本発明の詳細な説明は、例示的で且つ説明的なものに過ぎず、請求された本発明の追加説明を提供するためのものである。

【発明の効果】

【0032】

本発明によれば、既存の放送システムを用いて放送コンテンツと関連した付加情報を提供することができる。

【0033】

本発明によれば、放送コンテンツと関連した付加情報が表示されなければならない時期を検出し、適切な時期に付加情報を使用者に提供することができる。

10

【0034】

本発明によれば、第2スクリーンデバイスに、放送コンテンツと関連した付加情報を提供することができる。

【図面の簡単な説明】

【0035】

添付の図面は次の通りである。

【図1】一般的な放送ストリームの一実施例を示す図である。

【図2】予め生成された(Pre-produced)コンテンツの場合に、トリガータイミングの一実施例を示す図である。

20

【図3】ライブ(Live)コンテンツの場合に、トリガータイミングの一実施例を示す図である。

【図4】トリガーシンタックスの一実施例を示す図である。

【図5】TDOパラメータテーブルの一実施例を示す図である。

【図6】TDOパラメータテーブルの一実施例を示す図である。

【図7】'Frequency of Use'属性値の意味の一実施例を示す図である。

【図8】'destination'属性値の意味の一実施例を示す図である。

【図9】TDOパラメータテーブルの2進形態のシンタックスの一実施例を示す図である。

【図10】TDOパラメータテーブルの2進形態のシンタックスの一実施例を示す図である。

30

【図11】TDOパラメータテーブルの2進形態のシンタックスの一実施例を示す図である。

【図12】TDOパラメータテーブルの2進形態のシンタックスの一実施例を示す図である。

【図13】TDOパラメータテーブルの2進形態のシンタックスの一実施例を示す図である。

【図14】活性化メッセージテーブル構造の一実施例を示す図である。

【図15】URLリスト構造の一実施例を示す図である。

【図16】TPTを含むプライベートセクション(private section)の2進フォーマットの一実施例を示す図である。

40

【図17】XML文書としてエンコードされたURL目録の一実施例を示す図である。

【図18】addTriggerEventListenerの一実施例を示す図である。

【図19】removeTriggerEventListenerの一実施例を示す図である。

【図20】EventListenerタイプの定義の一実施例を示す図である。

【図21】TriggerEventタイプの定義の一実施例を示す図である。

【図22】WM技法のためのアーキテクチャの一実施例を示す図である。

【図23】FP技法のアーキテクチャの一実施例を示す図である。

50

- 【図24】要求/応答ACR場合における静的活性化の一実施例を示す図である。
- 【図25】要求/応答ACR場合における静的活性化の一実施例を示す図である。
- 【図26】要求/応答ACR場合における動的活性化の一実施例を示す図である。
- 【図27】要求/応答ACR場合における動的活性化の一実施例を示す図である。
- 【図28】ACRサーバー活性化のためのアーキテクチャの一実施例を示す図である。
- 【図29】End Timeがない場合(a)と場合(b)における活性化トリガーの一実施例を示す図である。
- 【図30】End Timeがない場合(a)と場合(b)における活性化トリガーの一実施例を示す図である。
- 【図31】End Timeがある場合(a)における活性化トリガーの一実施例を示す図である。 10
- 【図32】End Timeがある場合(a)における活性化トリガーの一実施例を示す図である。
- 【図33】場合(c)に対する活性化トリガーの一実施例を示す図である。
- 【図34】場合(c)に対する活性化トリガーの一実施例を示す図である。
- 【図35】最後に伝達される動的活性化トリガーの一実施例を示す図である。
- 【図36】最後に伝達される動的活性化トリガーの一実施例を示す図である。
- 【図37】要求/応答の場合においてACRクライアントと他のサーバー間の順序図である。
- 【図38】イベント主導型ACR場合におけるACRクライアントと他のサーバー間の順序図である。 20
- 【図39】UPnP Remote UIクライアントサービスのアクションリストの一実施例である。
- 【図40】UPnP Remote UIクライアントサービスの一実施例である。
- 【図41】Trigger in DTVCC Service Number 6の一実施例である。
- 【図42】第2スクリーンシナリオ用システムアーキテクチャの一実施例である。
- 【図43】ATSC 2.0受信機及び第2スクリーン間のトポロジ(直接接続)の一実施例である。
- 【図44】ATSC 2.0受信機及び第2スクリーン間のトポロジ(間接接続)の一実施例である。 30
- 【図45】中央集中型実行モデル(Centralized Execution model)の動作概念図の一実施例である。
- 【図46】中央集中型実行モデルベース受信機及び第2スクリーン間の連動に対するフローの一実施例である。
- 【図47】分散型実行モデル(Distributed Execution model)の動作概念図の一実施例である。
- 【図48】分散型実行モデルベース受信機及び第2スクリーン間の連動に対するフローの一実施例である。
- 【図49】分散型実行モデルベース受信機及び第2スクリーン間の連動に対するフローの一実施例である。 40
- 【図50】第2スクリーンサービスアプリケーションのソフトウェア層の一実施例である。
- 【図51】第2スクリーンサービスアプリケーションのソフトウェア層の一実施例である。
- 【図52】第2スクリーンアプリケーションライフサイクル管理による送信順序と送信データの違いを示すテーブルの一実施例である。
- 【図53】受信機が第2スクリーンデバイスにUI情報を知らせる方法の一実施例である。
- 【図54】受信機が第2スクリーンデバイスにUI情報を知らせる方法の一実施例である。 50

- 。
- 【図55】受信機が第2スクリーンデバイスにTDO及びイベント情報を知らせる方法の一実施例である。
- 【図56】第2スクリーンデバイスがTPT及び第2スクリーンアプリケーション接近方法の一実施例である。
- 【図57】第2スクリーンデバイスがTPT及び第2スクリーンアプリケーション接近方法の一実施例である。
- 【図58】RemoteUIサーバーサービスのための放送シグナリングの一実施例である。
- 【図59】RemoteUIサーバーサービスのための放送シグナリングの他の実施例である。 10
- 【図60】第2スクリーンサービスのための装置ディスカバリ及び装置能力交換の一実施例である。
- 【図61】UPnPフォーラムのDeviceProfileXMLスキーマの一実施例である。
- 【図62】第2スクリーンデバイスのデバイスプロファイルの一実施例である。
- 【図63】第2スクリーンサービスに対するProtocolInfoの説明の一実施例である。
- 【図64】第2スクリーンデバイスでAddUIListing及びRemoteUIListingが実行される間のUIListingの一実施例である。 20
- 【図65】RemoteUIクライアントサービスのためのユニキャストシグナリングの一実施例である。
- 【図66】RemoteUIクライアントサービスのためのユニキャストシグナリングの一実施例である。
- 【図67】RemoteUIクライアントサービスのためのユニキャストシグナリングの一実施例である。
- 【図68】ProcessInputアクションで第2スクリーンデバイスに伝達になる“EventInfo”情報の一実施例である。
- 【図69】受信機及び第2スクリーンデバイス間の構成の一実施例である。
- 【図70】サービスのサービスタイプ及びサービスIDの一実施例である。 30
- 【図71】トリガー伝達サービスの動作概念図の一実施例である。
- 【図72】拡張された活性化トリガー(expanded activation trigger)生成過程の一実施例である。
- 【図73】増加した活性化トリガー(Augmented Activation Trigger)に対するXMLスキーマ記述(Schema Description)の一実施例である。
- 【図74】URI__data()のシンタックスの一実施例である。
- 【図75】URI__typeの意味の一実施例である。
- 【図76】増加していないトリガーに対するXMLスキーマ記述の一実施例である。
- 【図77】増加した活性化トリガーのフォーマットの一実施例である。 40
- 【図78】増加した活性化トリガーのフォーマットの一実施例である。
- 【図79】増加した活性化トリガーのフォーマットの一実施例である。
- 【図80】増加した活性化トリガーのフォーマットの一実施例である。
- 【図81】HTTPロングポーリング基盤アプローチ(long polling based approach)によるトリガーサービス状態変数の一実施例である。
- 【図82】HTTPロングポーリング基盤アプローチによるトリガーサービスアクションの一実施例である。
- 【図83】HTTPロングポーリング基盤アプローチによるGetTrigServerURLアクションの引数(Arguments)の一実施例である。
- 【図84】トリガーサービス状態変数の実施例を示す図である。 50

【図85】トリガーサービスアクションの実施例を示す図である。

【図86】GetLatestUnfilteredTriggerアクションの引数の実施例を示す図である。

【図87】GetLatestFilteredTriggerアクションの引数の実施例を示す図である。

【図88】UPnPイベントメカニズム基盤アプローチの実施例である。

【図89】トリガーサービス状態変数の実施例を示す図である。

【図90】トリガーサービスアクションの実施例を示す図である。

【図91】トリガー伝達サービスを通じてトリガーを取得する時、第2スクリーン上の動作の実施例を示す図である。

10

【図92】トリガー伝達サービスの動作概念を示す図である。

【図93】第1アプローチによる双方向(インタラクティブ)通信サービス状態変数の一実施例である。

【図94】第1アプローチによるClientList状態変数のXMLスキーマテーブルの一実施例である。

【図95】第1アプローチによるトリガーサービスアクションの一実施例である。

【図96】第1アプローチによるSignUpアクションの引数の一実施例である。

【図97】第1アプローチによるSignOffアクションの引数の一実施例である。

【図98】第2アプローチによる双方向通信サービス状態変数の一実施例である。

【図99】第2アプローチによるonBytesReceived関数の一実施例である

20

。【図100】第2アプローチによるsetStatusYes()の一実施例である。

【図101】第2アプローチによるsetStatusNo()の一実施例である。

【図102】第2アプローチによるsetStatus()の一実施例である。

【図103】第2アプローチによるsendBytes()の一実施例である。

【図104】AppURLサービス状態変数の一実施例である。

【図105】AppURLサービスアクションの一実施例である。

【図106】GetAppURLアクションの引数の一実施例である。

【図107】プロキシHTTPサーバーサービス(Proxy HTTP Service)の動作概念図の一実施例である。

30

【図108】プロキシサーバーサービス状態変数(Proxy Server Service State Variable)の一実施例である。

【図109】プロキシサーバーサービスアクション(Proxy Server Service Action)の一実施例である。

【図110】GetProxyURLアクションの引数の一実施例である。

【図111】RequestFiles()の一実施例である。

【図112】第2スクリーンデバイスアーキテクチャの一実施例である。

【図113】受信機の簡略化された構造図の一実施例である。

【図114】第2スクリーンデバイスの構造図の一実施例である。

【図115】第2スクリーンデバイスシナリオの一実施例である。

40

【図116】第1デバイスでインタラクティブサービスを処理する方法の一実施例である

。【図117】第2デバイスで実行される第2スクリーンアプリケーションでインタラクティブサービスを処理する方法の一実施例である。

【図118】第1デバイスとしてインタラクティブサービスを処理する装置の一実施例である。

【図119】第2デバイスで実行される第2スクリーンアプリケーションとしてインタラクティブサービスを処理する装置の一実施例である。

【発明を実施するための形態】

【0036】

50

発明のより容易な理解を助け、発明の実施例を示す添付の図面が、下記の記載と共に本発明の原理を説明するために提供される。

【0037】

本明細書で使われる用語としては、本発明においての機能を考慮したうえ、できるだけ現在広く使われている一般的な用語を選択したが、これは、当該分野に従事する技術者の意図、慣例又は新しい技術の出現などによって変更されてもよい。また、特定の場合、出願人が任意に選定した用語もあり、その場合には、該当する発明の説明の部分においてその意味を記載するものとする。したがって、本明細書で使われる用語は、単純な用語の名称ではなく、その用語が持つ実質的な意味と本明細書の全般にわたる内容に基づいて解釈されなければならないということは明らかである。

10

【0038】

本明細書において、メディア時間は、オーディオ/ビデオ又はオーディオコンテンツアイテムの再生 (playout) においていずれかの地点を参照するパラメータを表す。ACRは、自動コンテンツ認識 (Automatic Content Recognition) を表す。AMTは、活性化メッセージテーブル (Activation Messages Table) を表す。APIは、アプリケーションプログラミングインターフェース (Application Programming Interface) を表す。DAEは、宣言的応用環境 (Declarative Application Environment) を表す。DOは、宣言的客体 (Declarative Object) を表す。FLUTEは、単方向送信を通じたファイル伝達 (File Delivery over Unidirectional Transport) を表す。GPSは、位置確認システム (Global Positioning System) を表す。HTTPは、ハイパーテキスト送信プロトコル (Hypertext Transfer Protocol) を表す。IPは、インターネットプロトコル (Internet Protocol) を表す。IPTVは、インターネットプロトコルテレビ (Internet Protocol Television) を表す。iTVは、双方向テレビ (Interactive Television) を表す。MIMEは、インターネットメディアタイプ (Internet Media Type) を表す。NDOは、NRT宣言的客体 (NRT Declarative Object) を表す。NRTは、非実時間 (Non-Real Time) を表す。SMTは、サービスマップテーブル (Service Map Table) を表す。SSCは、サービスシグナリングチャンネル (Service Signaling Channel) を表す。TDOは、トリガーされた宣言的客体 (Triggered Declarative Object) を表す。TPTは、TDOパラメータテーブル (TDO Parameters Table) を表す。UDOは、非結合宣言的客体 (Unbound Declarative Object) を表す。UPnPは、ユーザープラグアンドプレイ (User Plug and Play) を表す。URIは、統合リソース識別子 (Uniform Resource Identifier) を表す。URLは、リソース位置表示子 (Uniform Resource Locator) を表す。XMLは、拡張性生成言語 (eXtensible Markup Language) を表す。TFTは、テキストフラグメントテーブル (Text Fragment Table) を表す。詳細な内容は後述する。

20

30

40

【0039】

この明細書で、DO、TDO、NDO、UDOは次のような意味を有することができる。

【0040】

DO (Declarative Object) は、双方向アプリケーション (例えば、HTML、JavaScript、CSS、XML及びマルチメディアファイル) を構成するコレクション (collection) であってもよい。

【0041】

50

トリガーされた双方向付加データサービスによって開始された宣言的客体又はトリガーによって開始されたDOによって実行されたDOなどを指定するために“トリガーされた宣言的客体(TDO)”を使用する。

【0042】

トリガーされた双方向データサービスではなくNRTサービスの一部として実行された宣言的客体を表すために、“NRT宣言的客体”(NDO)という用語を使用する。

【0043】

パッケージ型アップ(Packaged App)のようにサービスに結合されない宣言的客体、リンクによって実行されたDO、又はそのようなDOによって実行されたDOなどを表すために“非結合宣言的客体(UDO)”という用語を使用する。

10

【0044】

“リンク”は、現在のTVプログラム又はNRTサービスに関連するオンライン情報又は機能を提供するウェブサイトを示す放送局提供URLである。

【0045】

“パッケージ型アップ”は、放送局が視聴者に提供しようとする情報又は機能を提供し、視聴者がダウンロードして設置できる一つのファイルとしてまとめられた放送局提供宣言的客体(DO)である。

【0046】

その詳細は後述する。

【0047】

20

この明細書で、時間ベースメッセージ(time base message)は、時間ベーストリガー及びその等価物を含む。したがって、“時間ベースメッセージ”と“時間ベーストリガー”と同じ意味で使われてもよい。

【0048】

この明細書で、活性化メッセージは、活性化トリガー及び/又はAMT内の活性化エレメントのような、活性化を引き起こす全ての情報伝達を含む。

【0049】

図1は、一般的な放送ストリームの一実施例を示す図である。

【0050】

一般的な放送ストリームはTVプログラムのシーケンスで構成されている。各TVプログラムは、基本となるショー(show)で構成され、ショーは一般的に広告及び/又は他の介在物(interstitial material)によって分離されたブロックに分けられる。

30

【0051】

同図では、放送ストリームに、ショーA(Show A)セグメント、広告1(Ad1)、広告2(Ad2)、ショーB(Show B)セグメントなどが順次に含まれている。各ショーをなすセグメントをショー材料、広告を介在物と呼ぶことができる。

【0052】

各ショー又は介在物は、それと関連している双方向付加データサービスを有しても有さなくてもよい。

40

【0053】

本明細書では、統合ユニットとして放送局によって扱われる双方向付加サービスの一部を表すために“双方向サービスセグメント”、又は単純に“セグメント”という用語を使用する。双方向サービスセグメントは一般的に単一ショー又は1個の介在物と関連するが、必ずしもそうでない。

【0054】

このような双方向付加データサービスを実行するための2つのモデルがあり得る。直接実行モデルとトリガーされた宣言的客体(Triggered Declarative Object, TDO)モデルがそれである。

【0055】

50

直接実行モデルでは、仮想チャンネルが選択されると直ちに宣言的客体 (D e c l a r a t i v e O b j e c t , D O) は自動で開始され得る。宣言的客体は、スクリーン上の特定位置での画面の生成、世論調査の実行、他の特殊 D O の開始などのように、オーディオ - ビデオプログラムと同期化される双方向特徴の提供のための具体的な指示事項を得るために、インターネットを介してバックエンド (b a c k e n d) サーバーと通信することができる。

【 0 0 5 6 】

T D O モデルでは、T D O の開始、T O D の終了、又は T D O による一部作業の誘発のように T D O イベントを開始するために信号を放送ストリームで伝達したり又はインターネットを介して伝達することができる。これらのイベントは、特定時間に開始され、一般的にオーディオ - ビデオプログラムと同期化される。T D O が開始されると、プログラムされた双方向特徴を提供することができる。

10

【 0 0 5 7 】

T D O モデルの基本概念は、T D O を構成するファイル及びある動作を取るために T D O によって用いられるデータファイルはいずれも、その大きさによって受信機に伝達されるには一定量の時間がかかるということである。双方向エレメントに対するユーザ経験は、コンテンツの放送前に作成 (a u t h o r) されてもよいが、ある振舞いは、プログラム内のイベント、例えば、商業広告セグメントの発生と同時に起きるように注意して時間を決めなければならない。

【 0 0 5 8 】

T D O モデルでは、宣言的客体、関連しているデータ、スクリプト、テキスト及びグラフィックの伝達を、双方向イベントの再生に対する特定タイミングのシグナリングと区別する。

20

【 0 0 5 9 】

双方向イベントのタイミングを設定するエレメントはトリガー (T r i g g e r) である。

【 0 0 6 0 】

一つのセグメント内で用いられる T D O とトリガーによって開始される関連 T D O イベントに関する情報は、“ T D O パラメータテーブル ” (T P T) と呼ばれるデータ構造によって提供される。

30

【 0 0 6 1 】

図 2 は、既に生成されたコンテンツの場合に、トリガータイミングの一実施例を示す図である。

【 0 0 6 2 】

トリガーは、シグナリングを識別し、双方向イベントの再生タイミングを設定する機能を担うシグナリングエレメントである。

【 0 0 6 3 】

トリガーには、双方向サービスに関連するセグメントに対するメディア時間を示す役割を担う時間ベーストリガーと、双方向サービスに関連するアプリケーションのイベント発生時間を示す役割を担う活性化トリガーの 2 種類があり得る。

40

【 0 0 6 4 】

トリガーは、双方向サービスの支援のために、タイミングに関連する様々な機能を果たすことができる。トリガーはその機能によって多機能的であってもよく、特定トリガーインスタンス (i n s t a n c e) は、次の機能のうち一つ以上を行うことができる。

【 0 0 6 5 】

1 . T P T (放送ストリーム内の F L U T E セッション、インターネットサーバー、又はこれら両者を介して接近可能な T P T) の位置をシグナリングする。

【 0 0 6 6 】

2 . 次のプログラムセグメントに対する双方向コンテンツがプリロード (p r e - l o a d) されることを示す。

50

【 0 0 6 7 】

3 . 関連しているオーディオ/ビデオコンテンツ又はオーディオコンテンツの現在メディア時間を示す。

【 0 0 6 8 】

4 . T P T 内特定双方向イベントを参照し、当該イベントが現在又は特定の未来のメディア時間に実行されなければならないということを示す。

【 0 0 6 9 】

5 . 需要が最大となることを避けるために、インターネットサーバーに対する接続が特定の時間区間上でランダムに分散されることを示す。

【 0 0 7 0 】

図 2 は、2つのプログラミングセグメントと関連して伝達されるトリガーを示す。この例では、2つのセグメントとも“ 予め生成 ”されるが、これは、コンテンツがライブコンテンツではなく、双方向エレメントはポストプロダクション (p o s t - p r o d u c t i o n) で追加されたことを意味する。

【 0 0 7 1 】

図示のように、プログラミングセグメント 1 の発生前の短い時間に“ プリロード (p r e - l o a d) ”トリガーが伝達されて、受信機と関連している T P T 及び一方向コンテンツを取得できる機会を受信機に許容することができる。万一、プリロードトリガーが送信されないと、受信機は、コンテンツを取得するために該当のセグメント内で会う最初のトリガーを用いると考えることができる。

【 0 0 7 2 】

図示のように、トリガーは S e g m e n t 1 を通して送信されて、当該セグメントに関する現在メディア時間 (図では “ m ” と表記) を示すことができる。当該チャンネルに会うトリガーが双方向コンテンツを同期化して取得できるようにするためにメディア時間トリガーの周期的な伝達が必要なことがある。

【 0 0 7 3 】

S e g m e n t 2 の開始直前に当該セグメントに対するプリロードトリガーが送信される。

【 0 0 7 4 】

予め生成されたコンテンツ (非ライブ (n o n - l i v e) コンテンツ) の場合、受信機が最初のトリガーを処理した後に取得できる T P T は、当該セグメントに対する双方向経験の全てのエレメントのタイミングを定義することができる。当該受信機と T D O が双方向エレメントを再生するようにするために必要なものは、メディアタイミングの知識であってもよい。この T P T はメディア時間に関して双方向イベントを記述することができる。

【 0 0 7 5 】

図 3 は、ライブコンテンツの場合に、トリガータイミングの一実施例を示す図である。

【 0 0 7 6 】

ライブコンテンツの場合にも、T P T は、互いに異なる双方向イベントに関連するデータ及び情報を含むが、放送中にプログラム内の動作が展開するまでは、そのようなイベントの再生タイミングは知られないことがある。このようなライブの場合、トリガーの“ イベント - タイミング (e v e n t - t i m i n g) ”機能が活用される。このモードでトリガーは、特定の双方向イベントが新しい特定メディア時間値に再設定 (r e - t i m e d) されることを知らせることができる。又は、トリガーは、あるイベントが直ちに実行されることを示すこともできる。

【 0 0 7 7 】

図 3 で、S e g m e n t 3 に対する各トリガーの機能は次の通りである。

【 0 0 7 8 】

1 番トリガーは、プリロードトリガーであって、セグメント 3 に関するファイルを得ることができるディレクトリを参照する。

10

20

30

40

50

【0079】

2番トリガーは、メディア時間トリガーであって、Segment 3に対する再生タイミングを示すために用いられる。

【0080】

3番トリガーは、イベントリタイミング(re-timing)トリガーであって、TPTでevent ID = 2のイベントがメディア時間240に発生するように時間が再設定されることを示す。ハッチングされた領域は、3番トリガーが受信機に伝達され得る240以前の時間区間を示す。

【0081】

4番トリガーは、メディア時間トリガーである。

10

【0082】

5番トリガーは、イベントリタイミングトリガーであって、TPTでevent ID = 5のイベントがメディア時間444に発生するように時間が再設定されることを示す。

【0083】

6番、7番トリガーは、メディア時間トリガーである。

【0084】

8番トリガーは、イベントトリガーであって、TPTでevent ID = 12のイベントが直ちに実行されることを示す。

【0085】

9番トリガーは、イベントリタイミングトリガーであって、TPTでevent ID = 89のイベントがメディア時間900に発生するように時間が再設定されることを示す。

20

【0086】

以下、TDOのライフサイクル(life cycle)と状態、及び状態変化イベントについて説明する。

【0087】

TDOは、解除(Released)、準備(Ready)、活動(Active)、及び中止(Suspended)の異なる4つの状態で存在することができる。多数の異なる要素(トリガー、ユーザ動作、変更されるチャンネルなど)が一つの状態から他の状態への変動を引き起こしうる。

【0088】

TDOは、次の4つの状態を有することができる。4つの状態は、準備(Ready)、活動(Active)、中止(Suspended)及び解除(Released)である。準備状態は、TDOがダウンロードされて実行準備となっているが、まだ実行されてはいないことを意味する。活動状態は、TDOが実行中であることを意味する。中止状態は、その状態の保存と併せて、TDOの実行が一時的に中止することを意味する。解除状態は、TDOが準備、活動又は中止状態でないことを意味する。

30

【0089】

TDOの状態変化を起こし得るイベントの例には次のものがある。

【0090】

1. "prepare" をトリガーする - 装置は、TDOが実行(リソース割当、主メモリへのローディングなど)をするように準備されることを要求するトリガーを(現在選択された1次仮想チャンネルで)受信する。

40

【0091】

2. "execute" をトリガーする - 装置は、TDOが活性化されることを要求するトリガーを(現在選択された1次仮想チャンネルで)受信する。

【0092】

3. "suspend" をトリガーする - 装置は、TDOの中止を示すトリガーを(現在選択された1次仮想チャンネルで)受信する。

【0093】

4. "kill" をトリガーする - 装置は、TDOの終了を示すトリガーを(現在

50

選択された1次仮想チャネルで)受信する。

【0094】

図4は、トリガーシンタックスの一実施例を示す図である。

【0095】

活性化メッセージと時間ベースメッセージは両方とも、ある送信環境で次のような一般的な“トリガー”フォーマットを有することができる。

【0096】

ここで、シンタックス上の定義は、アルタネイティブ(alternative)を指定するために垂直パーシMBOL“|”が使われる場合以外は、ABNF(Augmented Backus-Naur Form)文法を用いて記述する。ルール(rule)は等号“=”によって定義と分離され、1行を超えてルール定義を続けるときは字下げ(indentation)を用い、リテラル(literal)は” ”で引用し、エレメントのグループ化には括弧“(”と”) ”を用い、選択的なエレメントは” [”と”] ”との間に入れる。そして、エレメントの前に<n>*がつくと、その次のエレメントのn回以上の反復を指定でき、nは基本値として0に設定される。そして、エレメントの前に<n>*<m>がつくと、その次にくるエレメントのn回以上m回以下の反復を指定することができる。

【0097】

このようなトリガーシンタックスは、<scheme>と“://”部分を除いて、付加的な制限事項と共にURI(Uniform Resource Identifier):ジェネリックシンタックス(Generic Syntax)に基づく。

【0098】

トリガーは、locator_partとtermsで構成することができる。termsは、省略可能な構成である。termsが存在するとlocator_partとtermsとを‘?’で連結することができる。

【0099】

locator_partは、hostname部分とpath_segments部分とで構成することができ、その間は‘/’で連結することができる。

【0100】

hostnameは、domainlabelとtoplabelとで構成することができ、domainlabelはその次に‘.’と共に0回以上反復され得る。すなわち、hostnameは、反復されたdomainlabelがtoplabelと連結した形態であってもよく、又は、toplabelのみで構成された形態であってもよい。

【0101】

domainlabelは、一つのalphanumericからなる形態であってもよい。又は、2つのalphanumericの間にalphanumeric又は‘-’が0回以上反復されたものが挿入された形態であってもよい。

【0102】

ここで、alphanumericは、alpha又はdigitを意味することができる。

【0103】

ここで、alphaは、lowalpha又はupalphaのいずれかであってもよい。

【0104】

ここで、lowalphaは、a、b、c、d、e、f、g、h、i、j、k、l、m、n、o、p、q、r、s、t、u、v、w、x、y、zのいずれか一つであってもよい。

【0105】

ここで、upalphaは、A、B、C、D、E、F、G、H、I、J、K、L、M、N、O、P、Q、R、S、T、U、V、W、X、Y、Zのいずれか一つであってもよい。

【0106】

10

20

30

40

50

ここで、`digit`は、0、1、2、3、4、5、6、7、8、9のいずれか一つであってもよい。

【0107】

`toplabel`は、一つの`alpha`からなる形態であってもよい。又は、`alpha`と`alphanumeric`との間に`alphanumeric`又は‘ - ’が0回以上反復されたものが挿入された形態であってもよい。

【0108】

`path_segments`は、セグメントが一つがあり、その次にセグメントが0回以上反復されたものが続く形態であってもよい。このとき、セグメントの間は‘ / ’で連結することができる。

10

【0109】

ここで、セグメントは、`alphanumeric`が1回以上反復された形態であってもよい。

【0110】

`terms`は、`event_time`又は`media_time`のいずれかからなる形態であってもよい。続いて`spread`又は`others`が来てもよい。`spread`と`others`は省略可能な構成であってもよい。`spread`と`others`が存在すると、これは‘ & ’を前に置いている形態であって、`event_time`又は`media_time`の次に位置することができる。

【0111】

ここで、`spread`は、‘ s = ’の次に`digit`が1回以上反復されたものが続く形態であってもよい。

20

【0112】

`event_time`は、‘ e = ’の次に`digit`が1回以上反復されたものが続く形態であってもよい。又は、その後、‘ & t = ’の次に`hexdigit`が1回以上7回以下反復されたものが続く形態であってもよい。‘ & t = ’とその次の部分は省略可能な構成であってもよい。

【0113】

ここで、`hexdigit`は、0、1、2、3、4、5、6、7、8、9、a、b、c、d、e、fのいずれか一つであってもよい。

【0114】

`media_time`は、‘ m = ’の次に`hexdigit`が1回以上7回未満反復されたものが続く形態であってもよい。

30

【0115】

`others`は、一つの`other`からなる形態であってもよい。又は、その次に‘ & ’と`other`が追加された形態であってもよい。

【0116】

ここで、`other`は、`resv_cmd`と`alphanumeric`が1回以上反復されたものが‘ = ’によって連結された形態であってもよい。

【0117】

ここで、`resv_cmd`は、‘ c ’、‘ e ’、‘ E ’、‘ m ’、‘ M ’、‘ s ’、‘ S ’、‘ t ’、‘ T ’を除く`alphanumeric`であってもよい。

40

【0118】

トリガーの長さは52バイトを超えなくてもよい。そして、トリガーの`hostname`部分は、登録されたインターネットドメイン名であってもよい。

【0119】

トリガーは、3つの部分からなることがわかる。

【0120】

<domain name part> / <directory path> [? <parameters>]

【0121】

50

< domain name part > は、登録されたドメイン名であり、< directory path > は、ドメイン名が URI において登場するようになる経路であってもよい。

【 0 1 2 2 】

< domain name part > は、登録されたインターネットドメイン名を参照することができる。< directory path > は、識別されたドメイン名に対する権限を持つエンティティ (entity) の制御及び管理下でディレクトリ経路を識別する任意の文字列であってもよい。

【 0 1 2 3 】

TDOモデルで、< domain name part > と < directory path > の組合せは、関連しているコンテンツに双方向性を追加するために受信機によって処理され得る TPT を固有に識別することができる。

10

【 0 1 2 4 】

< domain name part > と < directory path > の組合せは、現在セグメントに対する TPT が得られるインターネット位置の URL であってもよい。

【 0 1 2 5 】

すなわち、トリガーは、< domain name part > と < directory path > を用いて TPT を識別することができ、これによってトリガーの適用される TPT がわかる。当該 trigger が TPT に適用されてどのような役割を果たすかは、< parameters > 部分による。

20

【 0 1 2 6 】

以下、< parameters > 部分について説明する。

【 0 1 2 7 】

< parameters > は、“ event__time ”、“ media__time ”、又は“ spread ”の二つ以上で構成することができる。

【 0 1 2 8 】

次は、図 4 に示したシンタックスにおいて“ event__time ”、“ media__time ”、“ spread ”に関する部分である。

【 0 1 2 9 】

event_time = " e = " 1*digit[" &t = " 1*7hexdigit]

media_time = " m = " 1*7hexdigit

spread = " s = " 1*digit

30

【 0 1 3 0 】

“ event__time ” 項は、ターゲットとされたイベント (“ e = ” 項) と当該イベントが活性化されるべき時間 (“ t = ” 項) を識別するために活性化トリガーにおいて用いることができる。“ t = ” 項がないと、これは、トリガーが到達する時間にイベントが活性化されなければならないことを意味する。

【 0 1 3 1 】

すなわち、双方向イベント ID 項である “ e = ” は、当該イベントのターゲットである TDO の関連している TPT 内の app ID、当該特定イベントの event id e、及びこのイベント活性化のために用いられるデータエレメントの data ID を参照することができる。

40

【 0 1 3 2 】

選択的なタイミング値項である “ t = ” は、指定されたイベントのための新しいメディアタイミングを示すことができる。“ t = ” 部分が存在しないと、これは、指定されたイベントのためのタイミングがトリガーの到達時間であることを意味することができる。

【 0 1 3 3 】

“ media__time ” 項 (“ m = ” 項) は、時間ベーストリガーによって示される時間ベースに対する現在時間を識別するために時間ベーストリガーにおいて用いることが

50

できる。そして、現在表れているコンテンツを識別できるコンテンツ識別子情報（“c = ”項）がmedia_timeにさらに含まれてもよい。“c = ”項については、直接実行モデルに関する説明で後述する。

【0134】

すなわち、“m = ”は、その次に16進数を表示する1乃至8文字長の文字列が続くメディアタイムスタンプ(timestamp)項であって、現在のメディア時間を示すことができる。

【0135】

“spread”項は、サーバー上で作業負荷が分散するように、時間ベーストリガーに
10 応答してなされる動作（サーバーからのTPTの検索のような動作）又は活性化トリガーに
応答してなされる動作（TDOをサーバーに接続させるような動作）がランダムな時間
の間に遅延されなければならないということを示すために用いることができる。

【0136】

“s = ”項は、全ての受信機がトリガーで識別されたインターネットサーバーに接続を
試みるべき秒単位の時間数を示すことができる。それぞれの個別受信機が、指定された区
間内でランダムな時間を導出して、該当する量だけ接続要求を遅延させることによって、
受信機でトリガーが初めて現れる時に発生しう需要の最大値を時間上で分散させると考
えることができる。

【0137】

<media_time>パラメータを含むトリガーは、イベント時間に対する時間ベ
20 ースを設定するために用いられることから、時間ベーストリガーと呼ぶこともできる。

【0138】

<event_time>パラメータを含むトリガーは、イベントに対する活性化時間
を設定することから、活性化トリガーと呼ぶこともできる。

【0139】

図5及び図6は、TDOパラメータテーブルの一実施例を示す図である。

【0140】

TDOパラメータテーブル(TPT)は、一つのセグメントのTDO及びこれらのター
ゲットであるイベントに関するメタデータを含む。

【0141】

以下、テーブルに含まれる各フィールドについて説明する。各フィールドの大きさとテ
30 ーブルに含み得るフィールドの種類は設計者の意図によって追加又は変更されてもよい。

【0142】

TPT構造内のフィールドの具体的なセマンティクス(semantics)は次の
通りである。

【0143】

TDOパラメータテーブル(TPT)は、@majorProtocolVersion、@minorProtocolVersion、@id、@tptVersion、
@expireDate、@updatingTime、@serviceID、@baseURL属性、Capabilities、ライブトリガー(LiveTrigger)
40)、及び/又はTDOエレメントを含むことができる。

【0144】

TPTは、当該TPTのルートエレメントである。一つのTPTエレメントは、一つの
プログラミングセグメントの(時間上での)全体又は一部分を記述する。

【0145】

4ビット属性であってもよい@MajorProtocolVersionは、テーブル
定義の主バージョン番号を示すことができる。主バージョン番号は1に設定することが
できる。受信機はそれらが支援しない主バージョン値を示すTPTのインスタンスを捨て
ると考えることができる。

【0146】

10

20

30

40

50

4ビット属性であってもよい@MinorProtocolVersionが存在する場合、テーブル定義の副バージョン番号を示すことができる。存在しない場合、その値は0にデフォルトすることができる。副バージョン番号は0に設定することができる。受信機は、それらが支援するように備えられていない副バージョン番号値を示すTPTのインスタンスを捨てないと予想される。この場合、受信機はそれらが支援しない個別エレメント又は属性を無視すると予想される。

【0147】

URIである@idは、このTPTエレメントに関連している双方向プログラミングセグメントを固有に識別することができる。@idは、セグメントの識別子の役割を持つ。したがって、受信機がTPTをパーシングした後、@id情報を用いて、一つのセグメントに関連しているトリガー、AMTなどを該セグメントを識別するための@idを有するTPTとマッチングさせることができる。したがって、該トリガーとAMTが適用されるセグメントを探すことが可能になる。AMTに関する詳細な内容は後述する。

10

【0148】

8ビット整数であってもよい@tptVersionは、id属性によって識別されるTPTエレメントのバージョン番号を示すことができる。tptVersionは、TPTが変更される度に増分(increment)されてもよい。

【0149】

TPTエレメントの@expireDate属性は、存在する場合、当該TPTインスタンスに含まれた情報の満了日及び時間を示すことができる。万一、受信機がTPTをキャッシュに保存すると、満了日まで再使用することができる。

20

【0150】

16ビットエレメントであってもよい@updateTimeは、存在する場合、TPTが修正されることを示すことができ、TPTを再びダウンロードするための秒単位の推薦区間を提供する。

【0151】

16ビットエレメントであってもよい@serviceIDは、存在する場合、このTPTインスタンスで記述された双方向サービスと関連しているNRTservice_idを示すことができる。これは、この双方向サービスのためのファイルが放送ストリームで伝達されるとき、受信機がサービスマップテーブルからFLUTEパラメータを得るために必要である。

30

【0152】

@baseURL属性は、存在する場合、当該TPTで登場する任意の相対URLの前につくと、ベースURLを提供することができる。この属性は、ファイルの絶対URLを提供することができる。

【0153】

Capabilitiesエレメントは、存在する場合、当該TPTと関連している双方向サービスの意味ある表出のために必須である能力を示すことができる。要求される能力のうち一つ以上を持たない受信機はサービスの表出を試みないと予想される。

【0154】

LiveTriggerは、インターネットを介した活性化トリガーの伝達が利用可能な場合にのみ存在する。存在する場合、活性化トリガーを得るために受信機で必要とする情報を提供することができる。LiveTriggerのチャイルド(child)エレメントと属性については後述する。

40

【0155】

TPTエレメントのチャイルドエレメントであるTDOは、当該TPTインスタンスによって記述されたセグメントの間に双方向サービスの一部を提供するアプリケーション(例えば、TDO)を示すことができる。TDOのチャイルドエレメントと属性については後述する。

【0156】

50

LiveTriggerエレメントは、@URLと@pollPeriod属性を含むことができる。

【0157】

上述した通り、LiveTriggerエレメントは、インターネットを介した活性化トリガーの伝達が利用可能な場合にのみ存在する。存在する場合、活性化トリガーを得るために受信機で必要とする情報を提供することができる。

【0158】

LiveTriggerエレメントの属性である@URLは、インターネットを介して活性化トリガーを伝達し得るサーバーのURLを示すことができる。活性化トリガーは、双方向サービスプロバイダの選択によって、HTTPショートポーリング(short polling)、HTTPロングポーリング(long polling)、又はHTTPストリーミングを用いてインターネットを介して伝達することができる。

10

【0159】

LiveTriggerエレメントの属性である@pollPeriodは、存在する場合、活性化トリガーの伝達のためにショートポーリングが使用中であることを示すことができ、pollPeriod属性値は、受信機がポーリング周期として用いる秒単位の推薦時間を示すことができる。

【0160】

LiveTriggerエレメントが存在する場合、受信機は、当該TPTをパーシングして、活性化トリガーをインターネットを用いて伝達するための情報を得ることができる。@URL情報から、活性化トリガーを伝達してもらえらるサーバーのURLを得ることができる。@pollPeriod情報から、或いは@pollPeriod属性が存在しないという情報から、活性化トリガーがインターネットを介して伝達される方式とポーリング周期に関する情報を得ることができる。@pollPeriodの詳細な説明は後述する。

20

【0161】

TDOエレメントは、@appId、@appType、@appName、@globalID、@appVersion、@cookieSpace、@frequencyOfUse、@expireDate、@testTDO、@availInternet、@availBroadcast属性と、URL、Capabilities、ContentItem、及び/又はEventエレメントを含むことができる。

30

【0162】

上述した通り、TPTエレメントのチャイルドエレメントであるTDOは、当該TPTインスタンスによって記述されたセグメントの間に双方向サービスの一部を提供するアプリケーション(例えば、TDO)を示すことができる。

【0163】

16ビット整数であってもよい@appIdは、TPT範囲内でアプリケーションを固有に識別することができる。活性化トリガーはappIdを参照してトリガーに対するターゲットアプリケーションを識別することができる。@appIdの場合、アプリケーションの識別子である。一つのTPTには複数のアプリケーション(TDOのようなアプリケーション)があってもよい。したがって、TPTをパーシングした後に@appId情報を用いてアプリケーションを識別することができる。あるアプリケーションに適用されるトリガー又はAMTなどを、そのアプリケーションを識別する@appIdを有するアプリケーションとマッチングさせることができる。したがって、トリガーとAMTの適用されるアプリケーションを探すことが可能になる。AMTに関する詳細な内容は後述する。

40

【0164】

8ビット整数であってもよい@appTypeは、アプリケーションフォーマット類型を示すことができる。基本値は1に設定することができ、この値はTDOを示すことができる。他の値は他のフォーマットを示すことができる。

50

【0165】

TDOエレメントの属性である@appNameは、アプリケーションを開始するために視聴者の許可を求めるとき、視聴者に表示し得る人間にとって読み取り可能な名前であってもよい。

【0166】

TDOエレメントの属性である@globalIDは、アプリケーションの全域的に固有な識別子であってもよい。多くの場合に、受信機は近いうちに再び用いるアップをキャッシュに保存する。このことが有用となるには、受信機は当該アップが将来に登場するとそれを認識できなければならない。当該アップが新しいセグメントで再び登場すると、受信機がそれを認識できるようにするためにglobalIDが必要である。

10

【0167】

TDOエレメントの属性である@appVersionは、アプリケーションのバージョン番号であってもよい。appVersion値は、当該アプリケーション(globalIDによって識別されたアプリケーション)が変更される度に増分されてもよい。appVersion属性は、globalID属性が存在しない限り存在することができない。

【0168】

8ビット整数であってもよい@cookieSpaceは、各呼出し(invocation)間に持続性データ(persistent data)を保存するためにアプリケーションがどれくらい多くの空間を必要とするかを示すことができる。

20

【0169】

4ビット整数であってもよい@frequencyOfUseは、受信機にそのアプリケーションコードキャッシュ空間の管理に関する案内を提供するために概略的にどれくらい頻繁に当該アプリケーションが放送で使われるかを示すことができる。'@frequencyOfUse'の詳細な説明は後述する。

【0170】

TDOエレメントの属性である@expireDateは、受信機が当該アプリケーション及び関連するリソースを安全に削除できる日付及び時間を示すことができる。

【0171】

ブールの(Boolean)属性である@testTDOは、その値が“true”として存在する場合、当該アプリケーションがテスト用であり、一般受信機では無視されてもよいことを示すことができる。

30

【0172】

@availInternet属性に対する“true”値は、当該アプリケーションがインターネットを介したダウンロード用に利用可能であることを示すことができる。その値が“false”であれば、当該アプリケーションがインターネットを介したダウンロード用に利用可能でないことを示すことができる。属性が存在しない場合、基本値は“true”であってもよい。

【0173】

@availBroadcast属性に対する“true”値は、当該アプリケーションが放送からの抽出に利用可能であることを示すことができる。“false”値は、当該アプリケーションが放送からの抽出に利用可能でないことを示すことができる。属性が存在しない場合、基本値は“true”であってもよい。

40

【0174】

TDOエレメントのチャイルドエレメントであるURLの各インスタンスは、当該アプリケーションの一部であるファイルを識別することができる。インスタンスURLElementは@entry属性を含むことができる。URLElementの属性である@entryは“true”値を有すると、当該URLが当該アプリケーションのエントリーポイント(entry point)、すなわち、当該アプリケーションを開始するために開始され得るファイルであることを示すことができる。“false”値を有すると、当該UR

50

Lが当該アプリケーションのエントリポイントでないことを示すことができる。当該属性が登場しないとき、基本値は“false”であってもよい。TDOエレメントのチャイルドエレメントであるURLエレメントは、上述した通り、当該アプリケーションをなすファイルを識別する。受信端ではTPTをパーシングして当該URL情報を得た後、それを用いてサーバーに接続して、当該URL情報が示すアプリケーションをダウンロードすることとなる。

【0175】

TDOエレメントのチャイルドエレメントであるCapabilitiesは、存在する場合、当該アプリケーションの意味ある表出のために必須である能力を示すことができる。要求される能力のうち一つ以上を持たない受信機は当該アプリケーションの開始を試みないと予想される。

10

【0176】

TDOエレメントのチャイルドエレメントであるContentItemは、当該アプリケーションで必要とする一つ以上のデータファイルで構成されたコンテンツコンテンツアイテムを示すことができる。ContentItemエレメントは、このエレメントの属しているTDOエレメントが示すアプリケーションが必要とするデータファイルに関する情報を有する。受信端ではパーシングを行った後、ContentItemエレメントが存在する場合、ContentItem内のURL情報などを用いて、当該アプリケーションが必要とするデータファイルをダウンロードしたりできる。ContentItemのチャイルドエレメントと属性については後述する。

20

【0177】

TDOエレメントのチャイルドエレメントであるEventは、当該アプリケーションのためのイベントを示すことができる。Eventエレメントは、このエレメントの属しているアプリケーションのイベントを示す。いかなるイベントを有しているか、該当データにはどのようなものがあるか、該当動作にはどのようなものがあるか等に関する情報を含んでいる。これをパーシングして受信端ではアプリケーションのイベントに関する情報を得ることができる。Eventのチャイルドエレメントと属性については後述する。

【0178】

受信端でTPTを受けてパーシングすると、TDO及び上述したTDOのチャイルドエレメントと属性の情報を得ることができる。

30

【0179】

TDOエレメントのチャイルドエレメントの一つである、contentItemエレメントは、@updateAvail、@pollPeriod、@size、@availInternet、@availBroadcastattribute及び/又はURLエレメントを含むことができる。

【0180】

ここで、URLエレメントは、@entry属性を含むことができる。ContentItemエレメントのチャイルドエレメントであるURLの各インスタンスは、コンテンツアイテムの一部であるファイルを識別することができる。URLエレメントは、@entry属性を含むことができる。URLエレメントの属性である@entryは、“true”値を有すると、当該URLが当該コンテンツアイテムのエントリポイント、すなわち、当該コンテンツアイテムを開始するために開始されうるファイルであることを示すことができる。この属性が“false”値を有すると、当該URLが当該コンテンツアイテムのエントリポイントでないことを示すことができる。この属性が登場しないとき、基本値は“false”であってもよい。受信機は、パーシング後にContentItemのURL情報を用いて当該アプリケーションの要求されるデータファイルをダウンロードすることができる。この過程で、上述した他の属性のような情報を用いることができる。

40

【0181】

ContentItemエレメントのプールの属性である@updatesAvail

50

は、当該コンテンツアイテムがたまにアップデートされるか否か、すなわち、コンテンツアイテムが固定ファイル (static file) で構成されるか、それとも、当該アイテムが実時間データフィード (feed) であるかを示すことができる。その値が "true" であれば、当該コンテンツアイテムはたまにアップデートされ、その値が "false" であれば、当該コンテンツアイテムはアップデートされない。この属性が登場しないと、基本値は "false" であってもよい。

【0182】

ContentItem要素の属性である@pollPeriodは、updatesAvail属性の値が "true" の場合にのみ存在することができる。@pollPeriodが存在する場合、活性化トリガーを伝達するためにショートポーリングが用いられていることを示すことができ、@pollPeriod属性の値は、受信機がポーリング周期として使用するための秒単位の推薦時間を示すことができる。

10

【0183】

ContentItem要素の属性である@Sizeは、当該コンテンツアイテムの大きさを示すことができる。

【0184】

@availInternet属性は、その値が "true" であれば、当該コンテンツアイテムがインターネットを介したダウンロードに利用可能であることを示すことができる。その値が "false" であれば、当該コンテンツアイテムがインターネットを介したダウンロードに利用可能でないことを示すことができる。この属性が存在しない場合、基本値は "true" であってもよい。

20

【0185】

@availBroadcast属性は、その値が "true" であれば、当該コンテンツアイテムが放送からの抽出に利用可能であることを示すことができる。"false" 値は、当該コンテンツアイテムが放送からの抽出に利用可能でないことを示すことができる。属性が存在しない場合、基本値は "true" であってもよい。

【0186】

Event要素は、このEvent要素の属するTDO要素によって示されたアプリケーションのイベントに関する情報を含む。受信機は、当該イベントに関する情報を得るためにこのEvent要素をパースングすることができる。

30

【0187】

TDO要素のチャイルド要素であるEvent要素は、@eventId、@action、@destination、@diffusion属性及び/又はData要素を含むことができる。ここで、Data要素は、@dataID属性を含むことができる。

【0188】

上記Event要素の16ビット整数属性であってもよい@eventIdは、これを含むTDO要素の範囲内で当該イベントを固有に識別することができる。活性化トリガー (又は、AMT内の活性化要素) は、appIdとeventIdの組合せによって当該トリガーに対するターゲットアプリケーション及びイベントを識別することができる。イベントが活性化されると、受信機は、このイベントを当該アプリケーションに渡す。@eventIdは、イベントの識別子の役割を持つ。@eventId情報を用いて、当該イベントを活性化させるためのトリガー、AMTなどを、このイベントを識別するための@eventIdを有するアプリケーションとマッチングさせることができる。すなわち、活性化トリガー (又は、AMT内の活性化要素) は、appIdとeventIdの組合せによって当該トリガーに対するターゲットアプリケーション及びイベントを識別することができる。イベントが活性化されると、受信機は、このイベントを当該アプリケーションに渡す。AMTの詳細は後述する。

40

【0189】

Event要素の属性である@actionは、当該イベントが活性化されると

50

きに適用される動作の類型を示すことができる。許容される値は、“ p r e p ”、“ e x e c ”、“ s u s p ”及び“ k i l l ”であってもよい。

【 0 1 9 0 】

“ p r e p ”は、“ T r i g p r e p ”動作に対応できる。ターゲットとなったアプリケーションの状態が“ R e l e a s e d ”であれば、この動作は“ R e a d y ”への状態変更を起こすことができる。

【 0 1 9 1 】

“ e x e c ”は、“ T r i g e x e c ”動作に対応できる。このトリガーを受信すると、ターゲットとなったアプリケーションの状態は“ A c t i v e ”になり得る。

【 0 1 9 2 】

“ s u s p ”は、“ T r i g s u s p ”動作に対応できる。このトリガーを受信すると、ターゲットとなったアプリケーションの状態が“ A c t i v e ”である場合、“ S u s p e n d e d ”に変更可能であり、そうでない場合は変更されない。

【 0 1 9 3 】

“ k i l l ”は、“ T r i g k i l l ”動作に対応できる。このトリガーを受信すると、ターゲットとなったアプリケーションの状態は“ R e l e a s e d ”になり得る。

【 0 1 9 4 】

@ a c t i o n は、当該イベントが活性化される時に適用される動作の類型を示すことができる。

【 0 1 9 5 】

E v e n t エレメントの属性である @ d e s t i n a t i o n は、当該イベントのためのターゲット装置を示すことができる。@ d e s t i n a t i o n に関する詳細な内容は後述する。

【 0 1 9 6 】

E v e n t エレメントの 8 ビット整数属性であってもよい @ d i f f u s i o n は、存在する場合、秒単位の時間周期 T を示すことができる。この拡散パラメータの目的は、サーバーローディングの最大値を均一にすることである。受信機は 0 ~ T の範囲で 10 ミリ秒単位でランダム時間を算出し、T P T で U R L によって参照されたコンテンツを検索するためにインターネットサーバーに接続する前にこの時間量だけ遅延されると予想

【 0 1 9 7 】

E v e n t エレメントのチャイルドエレメントである D a t a は、存在する場合、当該イベントに関連するデータを提供することができる。E v e n t の互いに異なる活性化は、これらの活性化と関連している互いに異なる D a t a エレメントを有することができる。ここで、D a t a エレメントは、@ d a t a I D 属性を含むことができる。16 ビット整数属性である @ d a t a I D は、これを含む E v e n t エレメントの範囲内で D a t a エレメントを固有に識別することができる。イベントの活性化がこれと関連するデータを有する場合、活性化トリガーは、A p p I D 、E v e n t I D 及び D a t a I D の組合せによって当該 D a t a エレメントを識別することができる。D a t a エレメントは、当該イベントに用いられるデータを示す。一つのイベントエレメントは複数のデータエレメントを有することができる。データエレメントの @ d a t a I D 属性を用いてデータが識別される。受信端では該当データに関連するイベントが活性化された場合、活性化トリガー（又は、A M T 内活性化エレメント）は、A p p I D 、E v e n t I D 、及び D a t a I D の組合せによって D a t a エレメントを識別することができる。A M T に関する詳細な内容は後述する。

【 0 1 9 8 】

図 7 は、‘ F r e q u e n c y o f U s e ’ 属性値の意味の一実施例を示す図である。

【 0 1 9 9 】

“ 意味 ” 列は、当該アプリケーションを含むセグメントの発生頻度を示す。（もちろん

10

20

30

40

50

、一つのセグメント内で一つの属性が複数回登場可能である。) `frequencyOfUse` 属性は、`globalID`が存在しなければ存在することができない。万一、アップがキャッシュに保存されて後で用いられるとすれば、受信機は、該当アップが後で再び登場すると、同一のアップとして認識しなければならない。そのために `globalID` 属性が必要である。

【0200】

図8は、'destination'属性値の意味の一実施例を示す図である。

【0201】

図8に示すように、目的地属性の値が0のときに予約され、1のときに主装置のみを意味し、2のときに一つ以上の補助装置のみを意味し、3のときに主装置及び/又は一つ以上の補助装置を意味することができる。

10

【0202】

図9、図10、図11、図12、図13は、TDOパラメータテーブルの2進形態のシタックスの一実施例を示す図である。

【0203】

これは、上述したTPT構造の2進フォーマットである。この構造は、NRTでTPTを送信する場合に必要なフォーマットであって、TPTのXML構造をNRTで送信するのに適するように作成しておいたものである。

【0204】

TPTのXMLバージョンに含まれた次のエレメント及び/又は属性、すなわち、`@protocolVersion(major/minor)`、`@serviceID`及び`@tptVersion`は、2進テーブルを伝達するためのカプセル化ヘッダー(`encapsulation header`)によって放送ストリームで提供できるため、2進バージョンから省略することができる。

20

【0205】

フィールドのセマンティックスは、次の通りである。図9、図10、図11、図12、図13の順に続くTDOパラメータテーブルの2進フォーマットにおいて、上から下に登場する順に記述した。

【0206】

1ビットフィールドであってもよい`expire_date_included`は、`expire_date_field`が含まれるか否かを示すことができる。その値が'1'であれば、当該フィールドが含まれることを意味でき、その値が'0'であれば、含まれないことを意味できる。

30

【0207】

5ビットフィールドであってもよい`segment_id_length`は、`segment_id`のバイト長を示すことができる。

【0208】

可変長フィールドである`segment_id`は、セグメントIDのバイトを含むことができるが、これは、TPTXMLフォーマットの" id "属性と同じセマンティックスを有することができる。

40

【0209】

8ビットフィールドであってもよい`base_URL_length`は、`base_URL`フィールドのバイト長を示すことができる。

【0210】

可変長フィールドである`base_URL`はベースURLのバイトを含むことができるが、これは、TPTXMLフォーマットの`baseURL`属性と同じセマンティックスを有することができる。

【0211】

32ビットフィールドであってもよい`expire_date`は、当該TPTインスタンスに含まれた情報の満了日及び時間を示すことができる。受信機がTPTをキャッシュ

50

に保存すると、満了日まで再使用することができる。符号無し整数 (`unsigned integer`) は、1980年1月6日00:00:00 UTC以後、GPS秒 (`second`) の数から `GPS - UTC_offset` を引いたものと解釈することができる。GPS UTC オフセットはGPSとUTC時刻標準間現在の全体秒単位オフセットを定義する8ビット符号無し整数であってもよい。

【0212】

8ビットフィールドであってもよい `trigger_server_URL_length` は、`trigger_server_URL` フィールドのバイト長を示すことができる。このフィールドの値が0であれば、個々の活性化トリガーのインターネット伝達が利用可能でないことを示すことができる。

10

【0213】

`trigger_server_URL_length` フィールドが0でなければ、`trigger_server_URL` は、`Trigger Server URL` のバイトを含むことができ、これはTPTXMLフォーマットのLiveTriggerエレメントのURL属性と同じセマンティックスを有することができる。

【0214】

1ビットフィールドであってもよい `trigger_delivery_type` は、インターネットを介した個別活性化トリガーの伝達モードを示すことができる。'0' 値を有すると、HTTPショートポーリングが使用中であることを示すことができ、'1' 値を有すると、HTTPロングポーリング又はHTTPストリーミングが使用中であることを示すことができる。

20

【0215】

8ビット整数であってもよい `poll_period` は、HTTPショートポーリングが使用中であるとき、ポール間の推奨される秒の数を示すことができる。

【0216】

8ビットフィールドであってもよい `num_apps_in_table` は、当該TPTインスタンスで記述されたアプリケーション (TDO) の数を示すことができる。

【0217】

16ビットフィールドであってもよい `app_id` は、当該アプリケーション (`num_apps_in_table` ループ (`loop`) の反復で記述されるアプリケーション) に対する識別子を含むことができる。これは、当該TPTインスタンス内で固有であってもよい。

30

【0218】

1ビットフィールドであってもよい `app_type_included` は、`app_type` フィールドが当該アプリケーションに対して含まれるか否かを示すことができる。その値が'1'であれば、当該フィールドが含まれることを意味し、その値が'0'であれば、含まれないことを意味できる。

【0219】

1ビットフィールドであってもよい `app_name_included` は、`app_name` フィールドが当該アプリケーションに対して含まれるか否かを示すことができる。その値が'1'であれば、当該フィールドが含まれることを意味し、その値が'0'であれば、含まれないことを意味できる。

40

【0220】

1ビットフィールドであってもよい `global_id_included` は、`global_id` フィールドが当該アプリケーションに対して含まれるか否かを示すことができる。その値が'1'であれば、当該フィールドが含まれることを意味し、その値が'0'であれば、含まれないことを意味できる。

【0221】

1ビットフィールドであってもよい `app_version_included` は、`app_version` フィールドがこのアプリケーションに対して含まれるか否かを示す

50

ことができる。その値が ' 1 ' であれば、当該フィールドが含まれることを意味し、その値が ' 0 ' であれば、含まれないことを意味できる。

【 0 2 2 2 】

1ビットフィールドであってもよい `cookie__space__included` は、`cookie__space` フィールドがこのアプリケーションに対して含まれるか否かを示すことができる。その値が ' 1 ' であれば、当該フィールドが含まれることを意味し、その値が ' 0 ' であれば、含まれないことを意味できる。

【 0 2 2 3 】

1ビットフィールドであってもよい `frequency__of__use__included` は、`frequency__of__use` フィールドがこのアプリケーションに対して含まれるか否かを示すことができる。その値が ' 1 ' であれば、当該フィールドが含まれることを意味し、その値が ' 0 ' であれば、含まれないことを意味できる。

10

【 0 2 2 4 】

1ビットフィールドであってもよい `expire__date__included` は、`expire__date` フィールドがこのアプリケーションに対して含まれるか否かを示すことができる。その値が ' 1 ' であれば、当該フィールドが含まれることを意味し、その値が ' 0 ' であれば、含まれないことを意味できる。

【 0 2 2 5 】

8ビットフィールドであってもよい `app__type` は、存在する場合、このアプリケーションのフォーマット類型を示すことができる。その値が 0 であれば、そのアプリケーションが TDO であることを示すことができる。万一、このフィールドが存在しない場合、その値は 0 にデフォルトされる。他の値は他のフォーマットを示すことができる。

20

【 0 2 2 6 】

8ビットフィールドであってもよい `app__name__length` は、存在する場合、その直後にくる `app__name` フィールドのバイト長を示すことができる。このフィールドの値が 0 であれば、該当アプリケーションのための `app__name` フィールドが存在しないことを示すことができる。

【 0 2 2 7 】

可変長フィールドである `app__name` は、TPT XML フォーマットである TDO エLEMENT の `appName` 属性と同じセマンティックスを有することができる。

30

【 0 2 2 8 】

8ビットフィールドであってもよい `global__id__length` は、その直後にくる `global__id` フィールドのバイト長を示すことができる。このフィールド値が 0 であれば、当該アプリケーションのための `global__id` フィールドが存在しないことを示すことができる。

【 0 2 2 9 】

可変長フィールドである `global__id` は、TPT XML フォーマットの TDO エLEMENT の `globalId` 属性と同じセマンティックスを有することができる。

【 0 2 3 0 】

8ビットフィールドであってもよい `app__version` は、TPT XML フォーマットの TDO エLEMENT の `appVersion` 属性と同じセマンティックスを有する。

40

【 0 2 3 1 】

8ビットフィールドであってもよい `cookie__space` は、TPT XML フォーマットの TDO エLEMENT の `cookieSpace` 属性と同じセマンティックスを有することができる。

【 0 2 3 2 】

8ビットフィールドであってもよい `frequency__of__use` は、存在する場合、TPT XML フォーマットの TDO エLEMENT の `frequencyOfUse` 属性と同じセマンティックスを有することができる。

50

【0233】

8ビットフィールドであってもよい`expire_date`は、TPT XMLフォーマットのTDOエレメントの`expireDate`属性と同じセマンティックスを有することができる。

【0234】

1ビットフィールドであってもよい`test_app`は、当該アプリケーションが一般受信機に無視されるようになっているテストアプリケーションであるか否かを示すことができる。その値が‘1’であれば、テストアプリケーションであることを意味でき、その値が‘0’であれば、テストアプリケーションでないことを意味できる。

【0235】

1ビットフィールドであってもよい`available_on_internet`は、当該アプリケーションがインターネットを介して利用可能か否かを示すことができる。その値が‘1’であれば、インターネットを介して利用可能であることを意味でき、その値が‘0’であれば、インターネットを介して利用可能でないことを意味できる。

10

【0236】

1ビットフィールドであってもよい`available_in_broadcast`は、当該アプリケーションが放送を介して利用可能か否かを示すことができる。その値が‘1’であれば、放送を介して利用可能であることを意味でき、その値が‘0’であれば、放送を介して利用可能でないことを意味できる。4ビットフィールドであってもよい`number_URLs`は、当該アプリケーションを含むファイルの数を示すことができる。

20

【0237】

8ビットフィールドであってもよい`URL_length`は、それに続くURLフィールドの長さを示すことができる。

【0238】

可変長フィールドであるURLは、TPT XMLフォーマットであるTDOエレメントのURL属性と同じセマンティックスを有することができる。

【0239】

8ビットフィールドであってもよい`number_content_items`は、このアプリケーションによる使用のためにダウンロードしなければならないコンテンツアイテムの数を示すことができる。

30

【0240】

1ビットフィールドであってもよい`updates_avail`は、このコンテンツアイテムがたまにアップデートされる否か、すなわち、コンテンツアイテムが固定ファイルの集合であるか、それとも、実時間データフィールドであるかを示すことができる。その値が‘1’であれば、アップデートされることを示すことができ、その値が‘0’であれば、アップデートされないことを示すことができる。

【0241】

1ビットフィールドであってもよい`avail_internet`は、このコンテンツアイテムを含むファイルをインターネットを介してダウンロードできるか否かを示すことができる。その値が‘1’であれば、インターネットを介したダウンロードに利用可能であることを意味でき、その値が‘0’であれば、インターネットを介したダウンロードに利用可能でないことを意味できる。

40

【0242】

1ビットフィールドであってもよい`avail_broadcast`は、このコンテンツアイテムを含むファイルを放送を介してダウンロードできるか否かを示すことができる。その値が‘1’であれば、放送を介したダウンロードに利用可能であることを意味でき、その値が‘0’であれば、放送を介したダウンロードに利用可能でないことを意味できる。

【0243】

1ビットフィールドであってもよい`content_size_included`は、

50

このアプリケーションのために `content_size` フィールドが含まれるか否かを示すことができる。その値が '1' であれば、当該フィールドが含まれることを意味でき、その値が '0' であれば、含まれないことを意味できる。

【0244】

4ビットフィールドであってもよい `number_urls` は、このコンテンツアイテムを含むファイルの数を示すことができる。

【0245】

8ビットフィールドであってもよい `url_length` は、それに続く URL フィールドの長さを示すことができる。

【0246】

可変長フィールドである URL は、TPT XML フォーマットである TDO エLEMENT のチャイルドELEMENT である `ContentItem` の URL 属性と同じセマンティックスを有することができる。

【0247】

24ビットフィールドであってもよい `content_size` は、TPT XML フォーマットである TDO エLEMENT のチャイルドELEMENT である `ContentItem` の `contentSize` 属性と同じセマンティックスを有することができる。

【0248】

8ビットフィールドであってもよい `num_content_descriptors` は、その直後にくるデスクリプタループ内のコンテンツデスクリプタの数を示すことができる。

【0249】

可変長フィールドである `content_descriptor()` は、MPEG-2 デスクリプタフォーマット(タグ、長さ、データ)に従うデスクリプタであってもよい。このフィールドは、当該コンテンツアイテムに関する付加情報を提供することができる。このデスクリプタループに含み得るデスクリプタの中に、このコンテンツアイテムの意味ある表出のために必要な受信機能力を示す `Capabilities` デスクリプタがあってもよい。

【0250】

8ビットフィールドであってもよい `number_events` は、このTDOに対して定義されたイベントの数を示すことができる。

【0251】

16ビットフィールドであってもよい `event_id` は、当該イベント(`number_events` ループの反復で記述されたイベント)のための識別子を含むことができる。このフィールドは、当該アプリケーションの範囲内で唯一であってもよい。当該イベントは `app_id` と `event_id` の組合せによって活性化トリガーら内で参照されてもよい。

【0252】

5ビットフィールドであってもよい `action` は、TPT XML フォーマットである TDO ELEMENT の `Event` チャイルドELEMENT の `action` 属性と同じセマンティックスを有することができる。

【0253】

1ビットフィールドであってもよい `destination_included` は、当該イベントのために `destination` フィールドが含まれるか否かを示すことができる。その値が '1' であれば、このフィールドが含まれることを示すことができ、その値が '0' であれば、含まれないことを示すことができる。

【0254】

1ビットフィールドであってもよい `diffusion_included` は、当該イベントのために `diffusion` フィールドが含まれるか否かを示すことができる。その値が '1' であれば、このフィールドが含まれることを示すことができ、その値が '0'

10

20

30

40

50

’であれば、含まれないことを示すことができる。

【0255】

1ビットフィールドであってもよい`data__included`は、当該イベントのために`data__size`と`data__bytes`フィールドが含まれるか否かを示すことができる。その値が‘1’であれば、このフィールドらが含まれることを示すことができ、その値が‘0’であれば、含まれないことを示すことができる。

【0256】

`destination`フィールドのセマンティックスは、存在する場合、TPT XMLフォーマットであるTDOエレメントのEventチャイルドエレメントの`destination`属性のセマンティックスと同一であってもよい。

10

【0257】

`diffusion`フィールドのセマンティックスは、存在する場合、TPT XMLフォーマットであるTDOエレメントのEventチャイルドエレメントの`diffusion`属性のセマンティックスと同一であってもよい。

【0258】

`data__size`フィールドは、存在する場合、その直後にくる`data__bytes`フィールドの大きさを示すことができる。

【0259】

`data__bytes`フィールドは、存在する場合、当該イベントに関連するデータを提供することができる。当該イベントが活性化される度にターゲットアプリケーションはそのデータを読み取り、それを用いて所望の動作の実行を助けることができる。このフィールドは、原始(`raw`)の2進値を含み得るということを除けば、その内容がTPT XMLフォーマットである該当のTDOエレメントの該当のEventチャイルドエレメントの該当のDataチャイルドエレメントの内容と同一なあってもよく、TPT XMLフォーマットであるこのDataエレメントはその2進値のベース64エンコーディングを含むことができる。

20

【0260】

8ビットフィールドであってもよい`num__app__descriptors`は、その直後にくるデスクリプタループ内のデスクリプタの数を示すことができる。

【0261】

可変長フィールドである`app__descriptor()`は、MPEG-2デスクリプタフォーマット(タグ、長さ、データ)に従うデスクリプタであってもよい。このフィールドは、当該アプリケーション(TDO)に関する付加情報を提供することができる。このデスクリプタループに含み得るデスクリプタの中に、このアプリケーションの意味ある表出のために必要な受信機能力を示す`Capabilities`デスクリプタがあってもよい。

30

【0262】

8ビットフィールドであってもよい`num__TPT__descriptors`は、その直後にくるデスクリプタループ内のデスクリプタの数を示すことができる。

【0263】

可変長フィールドである`TPT__descriptor()`は、MPEG-2デスクリプタフォーマット(タグ、長さ、データ)に従うデスクリプタであってもよい。このフィールドは、当該TPTに関する付加情報を提供することができる。このデスクリプタループに含み得るデスクリプタの中に、このTPTによって示される双方向サービスの意味ある表出のために必要な受信機能力を示す`Capabilities`デスクリプタがあってもよい。

40

【0264】

図14は、活性化メッセージテーブル構造の一実施例を示す図である。以下、テーブルに含まれる各フィールドについて説明する。各フィールドの大きさでテーブルに含み得るフィールドの種類は、設計者の意図によって追加又は変更されてもよい。

50

【0265】

活性化メッセージテーブル (Activation Messages Table: AMT) は、セグメントのための活性化トリガーに相応するものを含むことができる。ある環境では活性化トリガーに代えてこのAMTが受信機に伝達されてもよい。トリガーは、“livetrigger”サーバーによって、ACRサーバーによってAMTを介してクローズドキャプション (closed caption) ストリームで伝達されてもよい。

【0266】

AMT構造内のフィールドの具体的なセマンティックスは、次の通りである。

【0267】

活性化メッセージテーブル (AMT) は、@majorProtocolVersion、@minorProtocolVersion、@segmentId、@beginMT属性、及び/又はActivationエレメントを含むことができる。

10

【0268】

AMTエレメントの4ビット属性であってもよい@majorProtocolVersionは、AMT定義の主バージョン番号を示すことができる。主バージョン番号は1に設定することができる。受信機は、それらが支援するように備えられていない主バージョン値を示すAMTのインスタンスを捨てると予想できる。

【0269】

AMTエレメントの4ビット属性であってもよい@minorProtocolVersionは、存在する場合、AMT定義の副バージョン番号を示すことができる。存在しない場合、その値は0にデフォルトすることができる。副バージョン番号は0に設定することができる。受信機は、それらが支援するように備えられていない副バージョン値を示すAMTのインスタンスを捨てないと予想され得る。この場合、受信機はそれらが支援しない個別エレメント又は属性を無視すると予想されてもよい。

20

【0270】

AMTの識別子である@segmentIDは、AMT内Activationが適用されるアプリケーション及びイベントを含むTPTの識別子とマッチングされる。@segmentIdは、AMTの識別子の役割を担うことができる。したがって、受信端でAMTを受信した後にこれをパースングし、@segmentId情報を用いてAMTを識別することができる。@segmentIdは、AMTがどのセグメントに適用されるべきかに関する情報を含んでおり、そのセグメントに関連するTPTが有している@idとマッチングされてAMTとTPTとを連結する役割を担うことができる。さらに、まず当該segmentを識別することによって、AMT内のそれぞれのActivationエレメントがターゲットとするTDO、イベントを識別するために必要な基本的な情報を提供することができる。

30

【0271】

AMTエレメントの属性である@beginMTは、存在する場合、AMTインスタンスが活性化時間を提供するセグメントのメディア時間の開始を示すことができる。@beginMTは、属しているAMTが適用されるセグメントに対してメディア時間の開始を示すことができる。これによって、Activationエレメントが示す活性化が発生する時間の基準を定めることができる。したがって、@beginMTがある場合、Activationエレメント内の@startTime属性は、@beginMTが示すメディア時間の開始に影響を受けることができる。

40

【0272】

AMTのActivationエレメントの各インスタンスは、あるイベントをこのイベントと関連しているあるデータと共にある時間に活性化する命令を示すことができる。Activationエレメントは、AMT内に複数個が存在してもよい。各Activationエレメントは、活性化トリガーと同様の役割を担うことができる。Activationエレメントはいずれも、AMT内の@segmentIdが示すセグメントに

50

適用することができる。Activationエレメント内の属性に、該当活性化がどのアプリケーションの、どのイベントに、いつ起きるかに関する情報などを含めることができる。Activationエレメント内の属性に関する詳細な説明を次に説明する。

【0273】

Activationエレメントは、@targetTDO、@targetEvent、@targetData、@startTime、@endTimeの属性を含むことができる。

【0274】

Activationエレメントの属性である@targetTDOは、AMTに関連しているTPT内のTDOエレメントのappID属性とマッチングされることによって、活性化命令に対するターゲットアプリケーションを識別することができる。@targetTDOは、AMT内のActivationエレメントがどのアプリケーションに適用されるかに関する情報を含むことができる。受信端ではAMTを受信してパーシングして@targetTDOを得、マッチングされるTPT内のTDOエレメント内のappIDを探して、Activationエレメントの適用されるアプリケーションを識別することができる。

10

【0275】

Activationエレメントの属性である@targetEventは、targetTDO属性によって識別されるTDOエレメントに含まれたEventエレメントのeventID属性とマッチングされることによって、当該活性化命令に対するターゲットイベントを識別することができる。@targetEventは、AMT内のActivationエレメントがどのアプリケーションのどのイベントに適用されるかに関する情報を含むことができる。受信端ではAMTを受信しパーシングして@targetEventを得、マッチングされるTPT内のTDOエレメント内のeventIDを探して、Activationエレメントの適用されるイベントを識別することができる。

20

【0276】

Activationエレメントの属性である@targetDataは、targetTDOとtargetEvent属性によって識別されるEventエレメントに含まれたDataエレメントのdataIDとマッチングされることによって、活性化命令が適用される時にターゲットイベントと関連しているデータを識別することができる。@targetDataは、活性化命令が適用される時、ターゲットとなったイベントに関連するデータを識別することができる。受信端ではAMTを受信しパーシングして@targetDataを得、TPT内の該当Eventエレメント内のdataIDを探することができる。

30

【0277】

イベントエレメントの属性である@startTimeは、メディア時間に対するイベントの有効期間の開始を示すことができる。受信機は、メディア時間がstartTime内の値に到達する時又はその後のできるだけ近いうちに当該命令を実行すると予想され得る。@startTimeは、当該イベントが起きる開始時間を示すことができる。この開始時間はメディア時間を基準とする。受信端ではAMTをパーシングして@startTime情報を得、これを用いてイベントの起きる時間が把握できる。受信端では、@segmentIdによって識別されたセグメントのメディア時間を基準にメディア時間が開始時間(startTime)に達すると、イベントを活性化させることができる。万一、既に開始時間が過ぎた場合は、できるだけ早くイベントを活性化させることができる。

40

【0278】

イベントエレメントの属性である@endTimeは、メディア時間に対する当該イベントのための有効期間の終わりを示すことができる。受信機は、メディア時間がendTimeの値を経過すると当該命令を実行しないと予想され得る。@endTimeは、当該イベントが終わる時間を示すことができる。メディア時間が終わる時間に達すると、受

50

信端ではイベントを実行させなくてもよい。

【0279】

AMT内Activationエレメントは、startTime値の昇順で登場できる。

【0280】

受信機がAMT内Activationによってイベントを活性化させるとき、受信機は、各活性化をその開始時間に適用したり、又はその後に行えるだけ近いうちに適用（例えば、受信機がサービスに加入して、開始時間以降と終了時間以前との間におけるある時間にAMTを受信する場合）すると予想され得る。TPT内イベントエレメントの“action”属性が“exec”であれば、受信機はTriggerEventをターゲットアプリケーションに渡すと予想できる。TriggerEventについては、APIに関する部分で後述する。

10

【0281】

図15は、URLリスト構造の一実施例を示す図である。

【0282】

URLリストは、受信機に対して使用潜在性のあるURLを含むことができる。次のようなURLなどを含むことができる。

【0283】

1．一つ又はそれ以上の未来セグメントに対するTPTのためのURL。受信機がファイルを予めダウンロードできるようにする。

20

【0284】

2．放送ストリーム内の独立型NRTサービスに関する情報が検索され得るNRTシグナリングサーバーのURL。受信機に放送ストリーム内のNRTサービスシグナリングの伝達に対する接近権限がなくてもこれらのサービスに接近可能になる。

【0285】

3．仮想チャンネルに対する使用報告が送信される使用報告サーバーのURL。受信機に放送ストリーム内のこのURLの伝達に対する接近権限がなくてもこのような報告を送信可能にする。

【0286】

4．仮想チャンネルに対するPDI-QテーブルのURL。受信機に放送ストリームで伝達されたPDI-Qテーブルに対する接近権限がなくても視聴経験を個人化可能にする。（PDI-Qテーブルは双方向サービスを提供するにあたり、ユーザにカスタマイズサービスを提供するための個人化（personalization）に関連するテーブルである。PDI-Qテーブルを介してユーザに個人化のための質問ができる。）。

30

【0287】

そのうち、UsrUrlエレメントに関して、URLリストを作ってさらに使用報告のためのサーバーのURLを知らせる必要があり得る。これは、現在、受信機が視聴し消費しているコンテンツの種類と好むデータを活用してビジネスに活用するためであってもよい。URLリストにあるUsrUrlエレメントは、次のように様々に解釈可能である。

【0288】

第一は、使用報告サーバーである場合は、受信機はこの使用報告サーバーのURLをもって、予め定められたプロトコル（例、データ構造、XMLファイルなど）によって受信機の使用報告機能を果たすことができる。

40

【0289】

第二は、受信機のウェブブラウザで実行されるTDOであってもよい。ここでは、使用報告TDOの位置を知らせ、この場合、TDOが直接、受信機のウェブブラウザのAPI（例、ファイルAPI、使用報告API）を用いて受信機に保存された或いは現在消費しているコンテンツの情報を収集して報告することができる。TDOは、独自でXMLHttpRequestというジャバスクリプトAPIを活用して収集されたデータを送信することができる。

50

【0290】

U r l l i s t は、U r l L i s t、T p t U r l、U r s U r l、及び/又はP d i U r l などを含むことができる。これらのエレメントのセマンティックスは次の通りである。

【0291】

U r l l i s t エレメントのエレメントであるT p t U r l は、現在の双方向付加サービス内の未来セグメントに対するT P T のU R L を含むことができる。複数のT p t U r l エレメントが含まれた場合、放送内のセグメントの登場順に配列することができる。

【0292】

U r l l i s t エレメントのエレメントであるN r t S i g n a l i n g U r l は、受信機が現在のトランスポートストリーム内の全ての仮想チャンネルに対するN R T シグナリングテーブルを取得し得るサーバーのU R L を含むことができる。

10

【0293】

U r l l i s t エレメントのエレメントであるU r s U r l は、受信機が現在の仮想チャンネルに対する使用（視聴率調査）報告を送信し得るサーバーのU R L を含むことができる。

【0294】

U r l l i s t エレメントのエレメントであるP d i U r l は、現在の仮想チャンネルに対するP D I - Q テーブルのU R L を含むことができる。

【0295】

図16は、T P T を含むプライベートセクションの二進フォーマットの一実施例を示す図である。図16は、後述する伝達メカニズムにおいて放送ストリームでT P T が伝達される場合を示す。詳細な内容は後述する。

20

【0296】

トリガー、T P T など伝達する伝達メカニズムについて説明する。双方向サービス生成からの結果、放送ストリームでのトリガーの伝達、インターネットを介した時間ベーストリガーの伝達、インターネットを介した活性化トリガーの伝達（A C R シナリオ）、放送ストリームでのT P T の伝達、インターネットを介したT P T の伝達、T D O 及びコンテンツアイテムの移動、複数セグメントの一つのセグメントへの結合（c o m b i n i n g）などを順に説明する。

30

【0297】

以下、双方向サービス生成からの結果について説明する。

【0298】

一つのセグメントに対するサービス生成過程は、全てのT D O 及び他のコンテンツアイテムを含むフォルダ、X M L f o r m a t のT P T ファイル、及びX M L f o r m a t のA M T ファイルを生成することができる。その他の結果物も生成することができる。

【0299】

次に、放送ストリーム内でのトリガーの伝達について説明する。

【0300】

放送ストリームで伝達される場合、トリガーはU R L S t r i n g 命令内でサービス#6でD T V クローズドキャプションチャンネルで伝達することができる。

40

【0301】

万一、トリガーの長さが26文字以下であれば、トリガーはセグメント化せずに送信することができる（Type = 11）。万一、トリガーの長さが27乃至52文字であれば、2つのセグメント（the first segment in a Type = 00セグメントである一番目のセグメントと、Type = 10セグメントである2番目のセグメント）で送信されてもよい。

【0302】

当該命令の与えられた任意のインスタンスで伝達されたU R I のタイプは、インスタンス8ビットパラメータによって与えることができる。

50

【0303】

TDOモデルを使用する双方向サービスの場合、URIデータ構造のURIタイプは、0 (TDOモデルのための双方向Interactive TVトリガー) に設定することができる。この伝達メカニズムは、時間ベーストリガーも活性化トリガーも含む。

【0304】

時間ベーストリガーが(クローズドキャプションサービス#6で)放送ストリームで伝達される場合に、“m=”項が不在であれば、時間ベーストリガーは単純にシグナリングサーバーのURLを伝達することができる。そして、“m=”項が不在であれば、“t=”項は活性化トリガーに不在でなければならない。

【0305】

活性化トリガーが(クローズドキャプションサービス#6で)放送ストリームで伝達される場合に、すなわち、“Trigger”フォーマットが“e=”項を有しており、“t=”項は有しているかないない場合に、“t=”項が存在すれば、活性化時間は時間ベースに対するタイムスタンプであってもよい。そして、“t=”項が不在であれば、活性化時間はメッセージの到達時間になってもよい。

【0306】

時間ベーストリガーと活性化トリガーがCCサービス#6を介して伝達される場合、放送局で時間ベーストリガーと活性化トリガーを扱う方法には3つの可能な方法がある。この3つの方法は、‘明示的な時間ベースがないセグメントモード’、‘明示的な時間ベースがあるセグメントモード’、そして‘明示的な時間ベースがないサービスモード’である。

【0307】

これらは放送内でセグメント単位で混合されてもよい。

【0308】

明示的な時間ベースがないセグメントモードでは、活性化メッセージがタイムスタンプを含まないため、各メッセージの活性化時間はそのメッセージの伝達時間となり、また、時間ベースメッセージがタイムスタンプを含まないため、その唯一の目的はTPTファイルを伝達し得るシグナリングサーバーのURLを提供することであり得る。このモードでは、シグナリングサーバーのURLを提供するための活性化メッセージ内のURLによって時間ベースメッセージが完全に省略されてもよいが、この場合受信機は、最初の活性化メッセージが登場するまでTPTを検索したり、TDOのダウンロードを開始できなくなり、活性化メッセージに対する応答をだいぶ遅延することになる。

【0309】

このような場合、CCサービス#6に登場し得る時間ベースメッセージは、“Trigger”フォーマットの“locator_part”を含むことができ、“spread”項も含むことができるが、“media_time”項は含まなくてもよい。そして、CCサービス#6で登場し得る活性化メッセージは、“Trigger”フォーマットの“locator_part”と“event_time”項を含むことができ、“spread”項も含むことができるが、“イベント_time”項には“t=”部分がなくてもよい。時間ベース及び活性化メッセージの“locator_part”は現在のsegmentIdであってもよい。このURLは、インターネットを介してセグメントのためのTPTを検索するために用いられてもよい。

【0310】

明示的な時間ベースがあるサービスモードにおいて、時間ベースメッセージは時間ベースを定義するためにこのタイムスタンプを含む。活性化メッセージは、時間ベースに対する活性化時間を定義するために時間スタンプを含む。又は、これらのメッセージはタイムスタンプを含まなくてもよく、この場合、活性化時間はメッセージの到達時間である。

【0311】

この場合、CCサービス#6で登場し得る時間ベースメッセージは、Trigger”フォーマットの“locator_part”と“media_time”項を含むこと

10

20

30

40

50

ができ、“spread”項も含んでもよく、CCサービス#6で登場し得る活性化メッセージは“Trigger”フォーマットの“locator_part”と“event_time”項を含むことができ、“spread”項も含んでもよく、このとき、“event_time”項内の“t=”部分は存在してもしなくてもよい。時間ベース及び活性化メッセージの“locator_part”は現在のsegmentIdであってもよく、該当時間ベースは該当セグメントに特定である。このURLは、インターネットを介してセグメントのためのTPPを検索するために用いられてもよい。

【0312】

明示的な時間ベースがあるサービスモードにおいて、時間ベースメッセージは時間ベースを定義するためにこのタイムスタンプを含み、活性化メッセージは時間スタンプを含んでも含まなくてもよい。時間ベースは特定のセグメントに特定されるよりは、複数のセグメントにわたって延長することができる。時間ベースメッセージの“locator_part”は、時間ベースの識別子であってもよく、インターネットを介してサービスのためのTPPを検索するために使用可能なURLであってもよい。

10

【0313】

いずれの場合であれ、CCサービス#6にトリガーを挿入するトリガー挿入サーバーは、AMTから作業をして、活性化メッセージをAMT内のXMSフォーマットからCCサービス#6での伝達に特定されたトリガーフォーマットに変えなければならない。endTime属性がないActivationエレメントの場合は、活性化時間がstartTime属性と同一であり、単一トリガーが挿入されてもよい。startTimeとendTimeエレメントの両方を有するActivationエレメントの場合は、同一のターゲットをもってトリガーシーケンスが挿入されてもよい。シーケンス内の最初のトリガーは、startTime属性と同じ活性化時間を有することができ、当該シーケンス内の最後のトリガーは、endTime属性と同じ活性化時間を有することができ、シーケンス内のトリガーの活性化時間の間には固定した時間区間が存在できる(例外として、シーケンス内の最後のトリガーとその直前のトリガーとの間の区間はより短くてもよい)。この固定した時間区間の長さは設定可能である。

20

【0314】

時間ベース及び活性化メッセージがセグメントモードであるとき、時間ベースは、セグメントに対して特定することができる。このベースは、当該セグメントの開始時に“beginMT”値で始まって、セグメントを通じて行うことができる。個々の活性化の“startTime”と“endTime”値は、“beginMT”値に対して相対的であってもよい。時間ベース及び活性化メッセージがサービスモードにあるとき、時間ベースはセグメントにわたっていてもよく、各セグメントに対する“beginMT”値は、サービス時間ベースと放送スケジュールを考慮して調整することができる。

30

【0315】

次に、インターネットを介した時間ベーストリガーの伝達について説明する。

【0316】

時間ベーストリガーインターネット伝達は、いわゆる自動コンテンツ認識(Automatic Content Recognition、ACR)状況で有用であり得るが、このような状況で、時間ベーストリガーの受信側はクロードキャプションサービス#6に対する接近権限がない。このような状況で受信機はビデオフレームを認識し、時間ベースをこれらのフレームと同期化させるためにACRを使用しなければならない。ACR状況で時間ベースメッセージはウォーターマーク又はACRサーバーから得ることができる。ACRサーバーから受ける場合、時間ベースメッセージはACRサーバーから応答として伝達されてもよい。

40

【0317】

次に、インターネットを介した活性化トリガーの伝達(ACRシナリオ)について説明する。

【0318】

50

活性化メッセージは、ショートポーリング、ロングポーリング又はストリーミングを通じて伝達することができるが、これらのメッセージはいずれも受信機とサーバーに相当のオーバーロードを発生させ得る。活性化メッセージは、AMTの形態で伝達されてもよいが、セグメントの長さに関する多量の情報を提供し、広告サイト遮断器 (ad killer) を可能にすることができる。他の方法も可能である。

【0319】

活性化メッセージが活性化トリガーの形態で伝達される場合に、すなわち、“Trigger” formatにおいて“e=”項はあり、“t=”項はあるかない場合に、これは、HTTPショートポーリング、ロングポーリング又はストリーミングを通じて伝達され得る。

10

【0320】

インターネットを介して伝達される場合、活性化メッセージは、個別的活性化トリガー伝達メカニズムや一括的活性化トリガー伝達メカニズムのいずれか一方又は両方を用いて伝達することができる。

【0321】

次に、個別的活性化トリガー伝達について説明する。

【0322】

前述したように、個別活性化トリガーは、インターネットを介して伝達されるとき、HTTPショートポーリング、ロングポーリング又はストリーミングを用いて伝達することができる。活性化トリガーのフォーマットはD TV C Cサービス#6を通じて伝達されるときと同一であってもよい。

20

【0323】

ショートポーリングの場合は、そのポーリング周期を指定しなければならない。この周期は、次の説明のように、TPTにあるpollPeriodを用いてショートポーリング動作を設定することができる。

【0324】

活性化トリガーのインターネット伝達が利用可能な場合、TPT内LiveTriggerエレメントのURL属性は、活性化トリガーを伝達し得る活性化トリガーサーバーを示すことができる。LiveTriggerエレメントのpollPeriod属性がTPTに存在する場合、これはHTTPショートポーリングが使用中であることを示すことができ、この属性は、受信機が用いるべきポーリング周期を示すことができる。LiveTriggerエレメントのpollPeriod属性がTPTに存在しない場合、これは、HTTPロングポーリングとHTTPストリーミングのいずれか一方が使用中であることを示すことができる。

30

【0325】

いずれのプロトコルが用いられるかにかかわらず、受信機は質疑形式で活性化トリガーサーバーにHTTP要求をすると予想できる。

【0326】

?mt=<media_time>

ここで、<media_time>は、視聴されているコンテンツの現在メディア時間であってもよい。

40

【0327】

ショートポーリングが使用中である場合、活性化トリガーサーバーからの応答は、<media_time>で終わるpollPeriod長の時間区間で発生した全てのトリガーを含むことができる。一つ以上の活性化トリガーがリターンされる場合、これら一つ以上の空白 (white space) 文字によって分離することができる。万一、リターンされる活性化トリガーがないと、当該応答は空いてもよい。

【0328】

HTTPロングポーリング又はHTTPストリーミングが使用中である場合は、活性化トリガーサーバーは、活性化トリガーが放送ストリームで伝達されるメディア時間まで応

50

答のリターンを待機することができる。このとき、活性化トリガーをリターンすることができる。

【0329】

HTTPロングポーリングが使用中である場合、活性化トリガーサーバーは、活性化トリガーをリターン後にセッションを閉じることができる。受信機はアップデートされたメディア時間を用いて直ちに他の要求をすると予想できる。

【0330】

HTTPストリーミングが使用中である場合、活性化トリガーサーバーは、各活性化トリガーをリターンした後に当該セッションを開いた状態に維持し、追加の活性化トリガーを伝達する時間が到達すると、当該セッションでそれらのトリガーを伝達することができる。

10

【0331】

いずれの場合においても、HTTP応答は、伝達モードを知らせるために、次の形態のいずれかの形態からなるHTTP Response Headerフィールドを含むことができる。

【0332】

ATSC-Delivery-Mode:ShortPolling[<poll-period>]

ATSC-Delivery-Mode:LongPolling

ATSC-Delivery-Mode:Streaming

【0333】

<poll-period>パラメータは、連続するポールに対してポール間の推奨間隔を示すことができる。<poll-period>は省略されてもよい。

20

【0334】

次に、活性化トリガー一括伝達について説明する。

【0335】

活性化トリガーがインターネットを介して一括的に伝達される場合、一つのセグメントに対する活性化トリガーを、当該セグメントに対するTPTをメッセージの一番目の部分とし、AMTをメッセージの二番目の部分とする多重部分MIMEメッセージの形態としてHTTPを通じて当該セグメントに対するTPTと共に伝達することができる。

【0336】

以下、放送ストリームでのTPT伝達について説明する。

30

【0337】

放送ストリームで伝達時に、TPTを、それらのXMLフォーマットから対応する2進NRT-スタイルシグナリングフォーマットに変換して、テーブルインスタンス当たり一つずつNRT-スタイルプライベートセクションにカプセル化することができる。現在セグメントに対するTPTは常に存在する。一つ以上の未来セグメントにに対するTPTも存在することができる。TPTインスタンスは、そのsegment_idフィールドの値によって定義される。参考として、TDOパラメータテーブルの2進フォーマットは、上述した通りである。ここで、NRT-スタイルプライベートセクションは、図16のtp_t_section()に該当してもよい。

40

【0338】

要するに、TPTの2進構造をNRTで送信するために、NRT送信に合うようにセクション構造を有することができる。以下、この過程について詳しく説明する。

【0339】

各TPTをブロックに分割し、それらのブロックをtable_id、protocol_versionTPT_data_versionとsequence_numberフィールドの共通値を持つセクションのtp_t_bytes()フィールドに挿入することによって、各TPTをNRT-スタイルプライベートセクション内にカプセル化することができる。これらのブロックは、section_numberフィールド値の昇順でセクションに挿入することができる。プライベートセクションは、TPTに関連する仮

50

想チャンネルのIPサブネットのサービスシグナリングチャンネル (Service Signaling Channel; SSC) で送信することができる。ここで、“Service Signaling Channel” は、ATSC-NRT 標準に定義されており、特定IPアドレスとポートナンバーを持つチャンネルを意味する。セクション内の `sequence_number` フィールドは、同一SSCで送信される互いに異なるTPTインスタンスを区別するために用いることができる。

【0340】

以下、図16のフィールドについて説明する。

【0341】

プライベートセクション (`tpt_section()`) は、`table_id`、`protocol_version`、`sequence_number`、`TPT_data_version`、`current_next_indicator`、`section_number`、`last_section_number`、及び/又は `service_id`、`tpt_bytes()` 情報を含むことができる。

10

【0342】

8ビットフィールドであってもよい `table_id` は、当該テーブルセクションをTDOパラメータテーブルインスタンスに属するものとして識別することができる。

【0343】

`protocol_version` は2つの部分に分けることができる。8ビット符号無し整数フィールドのうち上位4ビットは、当該テーブルの定義の主バージョン番号と、ここに含まれて送信されるTPTインスタンスを示すことができ、下位4ビットは、副バージョン番号を示すことができる。主バージョン番号は1に設定することができる。受信機は、それらが支援するように備えられていない主バージョン値を示すAMTのインスタンスを捨てると予想され得る。副バージョン番号は0に設定することができる。受信機は、それらが支援するように備えられていない副バージョン値を示すAMTのインスタンスを捨てないと予想され得る。この場合、受信機はそれらが認識できないデスクリプタと、それらが支援しないフィールドを無視すると予想できる。

20

【0344】

`sequence_number` は、8ビットフィールドであってもよい。`sequence_number` は、当該TPTインスタンスの他の全てのセクションの `sequence_number` と同一であり、当該サービスシグナリングチャンネル内の他のTPTインスタンスの全てのセクションの `sequence_number` とは異なってよい。したがって、他のTPTインスタンスと区別する役割を担うことができる。`sequence_number` フィールドは、このセクションが現れるサービスシグナリングチャンネルに関連しているIPサブネットを示すことができる。互いに異なるTPTインスタンスの `sequence_number` フィールド値は、放送ストリームでセグメントが登場する順序を反映することができる。

30

【0345】

`TPT_data_version` は、5ビットフィールドであってもよく、当該TPTインスタンスのバージョン番号を示すことができるが、ここで、TPTインスタンスは、その `segment_id` によって定義することができる。TPTバージョンをあらかじめ知ってこそ受信されたTPTセクションデータが最新バージョンのTPTであるか否かがわかるため、セクションテーブルに `TPT_data_version` フィールドがあってもよい。バージョン番号は、TPTインスタンス内の任意のフィールドに変更がある場合、 $1 \text{ modulo } 32$ 分ずつ増分できる。

40

【0346】

`current_next_indicator` は1ビット指示子であってもよく、TPTセクションに対して常に '1' に設定されて送信されるTPTが `segment_id` によって識別されるセグメントに対して常に現在TPTであることを示すことができる。

50

【0347】

`section_number`は8ビットフィールドであってもよく、当該TPTインスタンスセクションのセクション番号を提供することができるが、TPTインスタンスは`segment_id`によって識別可能である。TPTインスタンス内の最初のセクションの`section_number`は0x00であってもよい。`section_number`は、TPTインスタンス内の各追加セクションと共に1ずつ増分可能である。

【0348】

`last_section_number`は8ビットフィールドであってもよく、最後のセクション(すなわち、`section_number`が最も高いセクション)が、その一部であるTPTインスタンスの最後のセクション番号を提供することができる。

10

【0349】

`service_id`は、当該テーブルインスタンスに記述されたコンテンツアイテムを提供する双方向サービスと関連している`service_id`を特定することができる。

【0350】

`tpt_bytes()`は、可変長フィールドであって、当該セクションによって部分的に送信されるTPTインスタンスブロックで構成することができる。当該テーブルインスタンスの全てのセクションの`tpt_bytes()`フィールドが`section_number`フィールドの順に連結されると、その結果として一つの完全なTPTインスタンスが作成できる。

20

【0351】

すなわち、TPTの2進フォーマットを用いたり、XMLフォーマットを2進フォーマットに変えた後、NRT送信に合うようにそれを分けて、プライベートセクションのうち`tpt_bytes()`フィールドに入れてNRT方式で送信することができる。このとき、一つのTPTが複数のプライベートセクションに分けられると、それぞれのプライベートセクションは、同一の`table_id`、`protocol_version`、`TPT_data_version`、`sequence_number`値を有することができる。分けられたTPTブロックを`section_number`フィールド値に従う順に挿入することができる。

【0352】

受信端では、受信したプライベートセクションに対してパーシングできる。一つのTPTに再びまとめるために、同一の`table_id`、`protocol_version`、`TPT_data_version`、`sequence_number`値を持つプライベートセクションを用いることができる。このとき、`section_number`と`last_section_number`情報から得られる順序情報を用いることができる。同一の`table_id`、`protocol_version`、`TPT_data_version`、`sequence_number`値を持つ全てのプライベートセクションの`tpt_bytes()`が`section_number`の順に連結されると、その結果として一つの完全なTPTを作成できる。

30

【0353】

インターネットを介したTPTの伝達について図17を参照して詳しく説明する。

40

【0354】

次に、TDO及びコンテンツアイテムの移動について説明する。

【0355】

ネットワークと基地局は、TDO及びこれらのTDOによって用いられるコンテンツアイテム(ファイル)の伝達のために、自体HTTPサーバーを頻繁に提供しなければならない。提供がなされると、TPT内の`baseURL`はサーバーの位置を反映して調整されてもよい。

【0356】

次に、複数のセグメントを一つのセグメントへの結合について説明する。

50

【0357】

セグメント間の境界を徹底的に曖昧にさせるために、複数のセグメントに対するTPTとAMTを一つのTPT及びAMTに結合することができる。その段階は、次の通りである。

【0358】

1. 結合されるセグメントの集合を識別する。

【0359】

2. 新しいsegmentIdを有する新しいTPTを生成する。

【0360】

3. 結合されるセグメントの中でライブ活性化を有するものがあれば、それらの活性化の全てに対して接近権限を付与するリレーサーバーを提供し、このサーバーに対するパラメータを“LiveTrigger”エレメントに入れる。

10

【0361】

4. 必要によって各セグメントに対してbaseURLを適用して全体TDOとContentItemURLを得る(結合される全てのセグメントに共通するより短いbaseURLを識別し、それを、結合されたセグメントに対するbaseURLとして維持する)。

【0362】

5. 必要によって値を修正して衝突を除去する。

【0363】

6. 結合される全てのセグメントに対する全ての修正されたTDOエレメントを新しいTPTに挿入する。

20

【0364】

7. 結合されたTPTの新しいsegmentIdに相応するsegmentIdを有する新しいAMTを生成する。

【0365】

8. 新しいAMTに対して適切な新しい“beginMT”値を選択する。

【0366】

9. appId値における変更を反映して結合されるセグメントに対するAMTファイル内の全てのActivation要素のtargetId値を調整する。

30

【0367】

10. 新しい“beginMT”値と結合されるセグメントに対する放送スケジュールを反映して結合されるセグメントに対するAMTファイル内の全てのActivationエレメントのstartTimeとendTime値を調整する。

【0368】

11. 修正された全てのActivationエレメントを新しいAMTに挿入する。

【0369】

図17は、XML文書としてエンコードされたURL目録の一実施例を示す図である。

【0370】

以下、インターネットを介したTPTの伝達について説明する。

40

【0371】

インターネットで伝達される場合、TPTはHTTPを通じて伝達することができる。時間ベースメッセージで現在セグメントに対するTPTのURLは“<domainnamepart>/<directorypath>”であってもよい。TPTに対する要求に対する応答は、単純にTPTで構成されてもよく、要求されたTPTは第1部分にあり、URLリストは第2部分にあってXML文書としてエンコードされる2-部分MIMEメッセージとして構成されてもよい(要求に対する応答は常に現在セグメントに対するTPTを含む。一つ以上の未来セグメントらに対するTPTも含むことができる)。

【0372】

上述した応答の第2部分としてのURLは、図17のような形態であってもよい。

50

【0373】

図17の要素のセマンティクスについて説明する。

【0374】

“`Urllist`”は、受信機に有用なURLリストを含むことができる。

【0375】

“`TptUrl`”は、未来セグメントに対するTPTのURLを含むことができる。複数の`TptUrl`要素が含まれる場合、それらは、放送でセグメントの登場順に配列することができる。

【0376】

“`NrtSignalingUrl`”は、受信機が現在の放送ストリームで全ての仮想チャンネルに対するNRTシグナリングテーブルを取得し得るサーバーのURLを含むことができる。

10

【0377】

図18は、`addTriggerEventListener`の一実施例を示す図である。

【0378】

以下、DOを実行するための環境のための、ATSCJavaScriptAPIについて説明する。

【0379】

放送プログラムに対する宣言的客體動作の同期化を支援するために、ビデオ/放送客體に対して追加の方法が支援されてもよい。

20

【0380】

DTVC C 或いはインターネットでTPTが受信されると、TPT内にはDOを実行するための複数のイベントがあってもよく、これらのイベントは活性化トリガーによって活性化されてもよい。

【0381】

このイベントを処理するようにするために、リスナー(`Listener`)という関数を`eventId`別に登録しておくことができる。これによって、上述の‘追加の方案’として、リスナー関数を登録できる次の2つの関数があり得る。

【0382】

図18には、`addTriggerEventListener`についての説明とフォーマット、アークギュメント(`argument`)などが示されている。

30

【0383】

“`addTriggerEventListener`”関数は、`eventId`ごとに生成されたイベントを処理するコールバック(`callback`)関数(リスナー関数)を登録することができる。“`addTriggerEventListener`”関数は、`EventListener`タイプの`listener`及び`Number`タイプの`eventId`をアークギュメントとして受け取ることができる。`EventListener`タイプについて後述する。“`addTriggerEventListener`”関数は返還値を有しなくともよい(`void`)。ここで、`eventId`アークギュメントは、TPTのイベント要素においてイベントIDであってもよい。ここで、`listener`アークギュメントは、イベントに対するリスナーであってもよい。

40

【0384】

受信機のトリガー処理モジュールでは活性化メッセージを受信するすと直ちに“`addTriggerEventListener`”関数を用いて`eventId`ごとにリスナー関数を登録することができる。後で当該イベントに活性化が起きると、登録されたリスナー関数を呼び出すことができる。このとき、`TriggerEvent`タイプの客體をリスナー関数として伝達することができる。`TriggerEvent`タイプについては後述する。

【0385】

50

図19は、`removeTriggerEventListener`の一実施例を示す図である。

【0386】

図19には、`removeTriggerEventListener`に対する説明とフォーマット、アーギュメントなどが示されている。

【0387】

”`removeTriggerEventListener`”関数は、`eventId`ごとに生成されたイベントを処理するコールバック関数(リスナー関数)の登録を取消することができる。 ”`removeTriggerEventListener`”関数は、`EventListener`タイプの`listener`及び`Number`タイプの`eventId`をアーギュメントとして受け取ることができる。`EventListener`タイプについて後述する。 ”`removeTriggerEventListener`”関数は返還値を有しなくてもよい(`void`)。ここで、`eventId`アーギュメントは、TPTのイベントエレメントでイベントIDであってもよい。ここで、`listener`アーギュメントは、イベントに対するリスナーであってもよい。

10

【0388】

ジャバスクリプトプログラムでは、当該`eventId`別に発生し得るイベントをそれ以上受信しないことを希望したり、“`DestroyWindow`”のようなプログラム終了時には、“`removeTriggerEventListener`”を用いて、登録されたリスナー関数を取り消すことができる。

20

【0389】

図20は、`EventListener`タイプの定義の一実施例を示す図である。

【0390】

ここで、`EventListener`タイプの定義は、`WebIDL`(`Web interface definition language`)(インターフェース定義言語)に従う。`WebIDL`は、ウェブブラウザで実行されるように意図的インターフェースを記述するために用いることができる。`WebIDL`は、ウェブプラットフォームにおける共用スクリプトオブジェクトの動作をより読みやすく明細化するための多数の機能を有するIDLの変種である。

【0391】

`EventListener`タイプは、`TriggerEvent`タイプを持つイベントをアーギュメントとすることができる。`EventListener`はインターフェースであってもよい。

30

【0392】

図21は、`TriggerEvent`タイプの定義の一実施例を示す図である。

【0393】

`TriggerEvent`タイプは、イベントに関する情報を含むことができる。

【0394】

`TriggerEvent`タイプは、`eventId`、`data`、`status`を属性として有することができる。ここで、`eventId`は、TPTのイベントエレメント中の`eventId`であってもよい。ここで、`data`は、イベントの活性化のためのものであってもよい。ここで、`data`は、16進法となってもよい。ここで、`status`は、イベントの状態を意味することができる。ここで、`status`値が“`trigger`”の場合は、イベントが活性化トリガーによって活性化された状態を意味できる。ここで、`status`値が“`error`”の場合は、エラーが発生した状態を意味できる。

40

【0395】

TDOモデルについて説明してきた。以下、直接実行モデルについて説明する。

【0396】

直接実行モデルでは、仮想チャンネルが選択されると、宣言的客体(`Declarati`

50

ve Object ; DO) が自動で開始され得る。DO は、スクリーン上の特定位置での画面の生成、世論調査の実行、他の特殊 DO の開始などのようにオーディオ - ビデオプログラムと同期化される双方向特徴の提供のための具体的な指示事項を得るために、バックエンド (back end) サーバーとインターネットを介して通信することができる。

【0397】

以下直接実行モデルにおけるトリガー動作について説明する。

【0398】

トリガーの役割、機能、シンタックスは、直接実行モデルにおいても大きく変わらない。

【0399】

トリガーの性能については、上述した通りである。

【0400】

トリガーシンタックスについては、上述した通りである。

【0401】

トリガーは、3つの部分で構成されたと見なすことができる。

【0402】

<ドメインネーム部分> / <ディレクトリ経路> [? <パラメータ>]

直接実行モデルにおいて、<ドメインネーム部分> と <ディレクトリ経路> の組合せは、開始される DO を固有に識別することができる。

【0403】

<パラメータ> は、“event__time”、“media__time”、又は“spread”のうち一つ以上からなることができる。

【0404】

直接実行モデルにおいて、仮想チャンネルが選択されると直ちにアプリケーションが自動で開始されうる。アプリケーションは、“同期化されたコンテンツプロトコル”を介してバックエンドサーバーとインターネットで通信することができる。このサーバーは、オーディオ - ビデオプログラムと全て同期化された双方向特徴の提供のための具体的な指示事項を提供することができる。

【0405】

直接実行モデルの場合、直ちにアプリケーションが実行されるために、時間ベーストリガーが伝達されるやいなや、現在実行中のアプリケーションに情報を伝達することができる。このモデルにおいて、アプリケーションは、サーバーに、同期化のために、現在放映中のコンテンツに関する情報を続けて伝達する必要がある。そのために、時間ベーストリガーには、TDOモデルとは異なる特別な情報がさらに含まれてもよい。この特別な情報は、現在放映中のコンテンツに対する識別子であってもよい。

【0406】

同様に、上記コンテンツ識別子は、トリガーのパラメータ部分にパラメータの形態で存在できる。

【0407】

同様に、上記コンテンツ識別子は、トリガーのmedia__timeに一つの項の形態で存在してもよい。コンテンツ識別子項は、文字列を伴う“c=”によって指定されうるcontent__idと呼ぶことができ、現在見られているコンテンツに対する識別子を示すことができる。

【0408】

content__id項は、双方向サービス実行に対する直接実行モデルを支援するためのものであってもよい。

【0409】

上述した通り、このモデルにおいて、content__id項を有する時間ベーストリガーは、アプリケーションの開始後にこのアプリケーションに伝達されてもよく、このアプリケーションは、双方向作用のためのコンテキスト(context)の識別のために

10

20

30

40

50

、 `content_id` をバックエンドサーバーに伝達することができる。詳細な動作については後述する。

【0410】

直接実行モデルにおけるトリガーの伝達メカニズムについては、上述した通りである。

【0411】

ただし、放送ストリームによるトリガーの伝達の場合において、トリガーは、`URLString` 命令でサービス # 6 で DTV クローズドキャプションチャンネルに伝達されてもよい。そして、直接実行モデルを用いる双方向サービスは、`URI_type` フィールドが 2 (直接実行モデルのための双方向 TV トリガー) に設定されてもよい。

【0412】

以下、直接実行モデルの全体動作について説明する。

【0413】

双方向サービスを実行する一つのモデルとして、直接実行モデルでは、仮想チャンネルが選択されると直ちにアプリケーションが自動で開始されうる。アプリケーションは“同期化されたコンテンツプロトコル”を介してインターネットでバックエンドサーバーと通信することができる。このサーバーは、スクリーン上の特定位置での画面の生成、世論調査の実行、他の特殊 DO の開始などのようにオーディオ - ビデオプログラムと同期化される双方向特徴の提供のための具体的な指示事項を提供することができる。

【0414】

上記の動作を次のように行うことができる。

【0415】

まず、アプリケーションが開始されうる。その後、時間ベーストリガーが受信される。時間ベーストリガーは、アプリケーションが実行された後にアプリケーションに伝達される。時間ベーストリガーの `content_id` 項には、現在見られているコンテンツに関するコンテンツ識別情報が含まれていてもよい。アプリケーションは、相互作用のためのコンテキストを識別し、現在見られているコンテンツを識別するために `content_id` をバックエンドサーバーに伝達することができる。

【0416】

直接実行モデルについて説明してきた。

【0417】

図 22 は、WM 技法のためのアーキテクチャの一実施例を示す図である。

【0418】

他のインターフェースを介した伝達について説明する。

【0419】

非圧縮されたオーディオ及びビデオにのみ接近可能な (例えば、ケーブル又は衛星セットトップボックスから受信される) 環境で双方向サービスの取得を可能にするプロトコル及びアーキテクチャが定義される。アーキテクチャ及びプロトコルは、インターネット接続を有し、放送ストリームからの非圧縮されたオーディオ及びビデオにのみ接近できる受信機による使用のために設計することができる。勿論、アーキテクチャ及びプロトコルは、双方向サービスプロバイダがその支援を選択する場合は、成功的に用いることができる。

【0420】

このアーキテクチャは、視聴されているコンテンツの識別のための 2 つの基本技法を支援するように設計され、関連双方向サービスデータ改善事項がインターネットを介して伝達されうるようにすることができる。2 つの基本技法は、ウォーターマーキング (`Watermarking`) とフィンガープリンティング (`Fingerprinting`) であってもよい。

【0421】

ウォーターマーキング、フィンガープリンティング技法両方とも、その目的は、どのプログラムが現在視聴されているかを受信機が探して、当該プログラムのための双方向サー

10

20

30

40

50

ビスに関する付加情報を得る出発点として用いられうるURLを取得できるようにすることである。

【0422】

図22は、WM技法のためのアーキテクチャの一実施例を示す図である。

【0423】

WM技法のためのアーキテクチャにおいて、アーキテクチャは、放送局22010、ウォーターマーク挿入部22011、MVPD 22020、STB 22030、受信機22040、WMクライアント22050、TPTサーバー22060及びコンテンツサーバー22070で構成することができる。

【0424】

放送局22010は、オーディオ/ビデオストリーム及びこれと関連する双方向サービスを送出するソースであってもよい。TV放送局などをその例に挙げることができる。

放送局22010は、放送コンテンツの生産者又は流通業者であってもよい。放送局22010は、放送ストリーム、オーディオ/ビデオコンテンツ、双方向データ、放送スケジュール又はAMTなどを伝達することができる。

【0425】

ウォーターマーク挿入部22011は、放送されるオーディオ/ビデオフレームにウォーターマークを入れることができる。ウォーターマーク挿入部22011は、放送局22010と統合されてもよく、別個のモジュールであってもよい。ウォーターマークとは、受信機に必要な一種の情報であってもよい。ウォーターマークは、URLのような情報であってもよい。ウォーターマークについての詳細は後述する。

【0426】

MVPD 22020は、多重チャンネルビデオプログラム流通業者(Multiprogram Video Program Distributor)の略語である。MVPD 22020は、ケーブル事業者、衛星事業者又はIPTV事業者であってもよい。MVPD 22020は、放送局/ウォーターマーク挿入部から放送ストリームを受信することができる。ウォーターマーキングACRシステムの場合、ウォーターマーク挿入部22011によってウォーターマークが挿入される。MVPD 22020は、たびたびオーディオ及びビデオトラック以外の全てのプログラムエレメントを除くストリームを顧客の宅内のセットトップボックス(STB)に送る。

【0427】

STB 22030は一般にオーディオ及びビデオをデコーティング(圧縮解除)し、これらを、視聴者が視聴するTV受像機に送る。STBは、非圧縮されたオーディオ/ビデオコンテンツを受信機22040に送ることができる。STBは、本発明の一実施例に係る外部のデコーティング部であってもよい。

【0428】

受信機22040は、WMクライアント22050を含むことができる。WMクライアント22050は、受信機22040の外部に位置してもよい。ここで、受信機は、ウォーターマーキングができる受信機である。受信機22040の構造については後述する。

【0429】

WMクライアント22050は、該当用途のAPIを用いてACRサーバー(図示せず)から活性化トリガーを受け、これらをメイン受信機コードに送ることができる。一般に、WMクライアント22050は、受信機内に構成されるが、他の構成も可能である。WMクライアント22050は、非圧縮されたオーディオ/ビデオコンテンツから、挿入されたウォーターマークを抽出することができる。ウォーターマークはURLなどの情報であってもよい。

【0430】

TPTサーバー22060は、TPTのようなアプリケーションをダウンロードできるサーバーであってもよい。受信機22040は、抽出されたウォーターマークをACRサーバーに伝送する。ウォーターマークがデータベース(図示せず)に保存されたウォータ

10

20

30

40

50

ーマークとマッチングされると、受信機 2 2 0 4 0 は、一トリガー又は複数トリガーを応答として受信することができる。受信された一トリガー又は複数トリガーが、上述した新しい `locator_part` を有したり、又は新しいバージョンの T P T 又はアプリケーションパラメータテーブルが発見されると、受信機 2 2 0 4 0 は T P T サーバ 2 2 0 6 0 に新しい T P T 又はアプリケーションパラメータテーブルをダウンロードすることを要求することができる。

【 0 4 3 1 】

コンテンツサーバ 2 2 0 7 0 は、双方向サービスを提供するために必要なアプリケーション及び T D O を提供することができる。新しいアプリケーション又は T D O が必要な場合、新しいアプリケーションは、T P T 又はアプリケーションパラメータテーブルにある URL を用いてダウンロードすることができる。

10

【 0 4 3 2 】

ウォーターマーキング (W M) 技法において、放送局 / ウォーターマーク挿入部は、放送されるオーディオ / ビデオフレームにウォーターマークを入れることができる。これらのウォーターマークは、視聴者に感知されないか、又はできるだけ邪魔とならないような量の情報を受信機に送信するように設計することができる。このようなウォーターマークは、受信機が必要とする情報を直接提供してもよく、又は受信機が必要とする情報を得るために、遠く離れたサーバにインターネット接続を通して送ることができるコード値のみを提供してもよい。

【 0 4 3 3 】

図 2 3 に、F P 技法のアーキテクチャの一実施例が示されている。

20

【 0 4 3 4 】

F P 技法のアーキテクチャにおいて、アーキテクチャは、放送局 2 3 0 1 0、M V P D 2 3 0 2 0、S T B 2 3 0 3 0、受信機 2 3 0 4 0、F P クライアント 2 3 0 5 0、T P T サーバ 2 3 0 6 0、コンテンツサーバ 2 3 0 7 0、シグネチャー (`signature`) 抽出機 2 3 0 8 0 及び F P サーバ 2 3 0 9 0 で構成することができる。

【 0 4 3 5 】

放送局 2 3 0 1 0 は、オーディオ / ビデオストリーム及びオーディオ / ビデオストリームと関連した双方向サービスを出力するソースであってもよい。放送局 2 3 0 1 0 の一例として T V 放送局が挙げられる。放送局 2 3 0 1 0 は、放送コンテンツ生産者や配給者であってもよい。放送局 2 3 0 1 0 は、放送ストリーム、オーディオ / ビデオコンテンツ、双方向データ、放送スケジュール、又は A M T を伝達することもできる。

30

【 0 4 3 6 】

M V P D 2 3 0 2 0 は、多重チャンネルビデオプログラム流通業者 (`Multiprogram Video Program Distributor`) の略語である。M V P D 2 3 0 2 0 は、ケーブル事業者、衛星事業者又は I P T V 事業者であってもよい。M V P D 2 3 0 2 0 は、たびたびオーディオ及びビデオトラック以外の全てのプログラムエレメントを除くストリームを顧客の宅内のセットトップボックス (S T B) に送る。

【 0 4 3 7 】

S T B 2 3 0 3 0 は一般にオーディオ及びビデオをデコーティング (圧縮解除) し、これらを視聴者に視聴される T V 受像機に送る。S T B は、非圧縮されたオーディオ / ビデオコンテンツを受信機 2 3 0 4 0 に送ることができる。S T B 2 3 0 3 0 は、本発明の一実施例に係る外部のデコーティング部であってもよい。

40

【 0 4 3 8 】

受信機 2 3 0 4 0 は、F P クライアント 2 3 0 5 0 を含むことができる。F P クライアント 2 3 0 5 0 は、受信機 2 2 0 4 0 の外部に位置してもよい。ここで、受信機 2 3 0 4 0 は、フィンガープリンティングができる受信機である。受信機 2 3 0 4 0 の構造については後述する。

【 0 4 3 9 】

F P クライアント 2 3 0 5 0 は、当該用途の A P I を用いて、F P サーバ 2 3 0 9 0

50

から活性化トリガーを受けてそれらをメイン受信機コードに送ることができる。一般に、FPクライアント23050は受信機内に構成されるが、他の構成も可能である。FPクライアント23050は、非圧縮されたオーディオ/ビデオコンテンツから、フィンガープリントを抽出することができる。フィンガープリントについては後述する。

【0440】

TPTサーバー23060は、TPTのようなアプリケーションをダウンロードできるサーバーであってもよい。受信機23040は、抽出されたフィンガープリントをFPサーバー23090に伝送する。フィンガープリントがシグネチャー抽出機23080のシグネチャーとマッチングされると、受信機23040は、一トリガー又は複数トリガーを応答として受信することができる。受信された一トリガー又は複数トリガーが上述の新しいlocator_partを有したり、又は新しいバージョンのTPT又はアプリケーションパラメータテーブルが発見されると、受信機23040はTPTサーバー23060に新しいTPT又はアプリケーションパラメータテーブルをダウンロードすることを要求することができる。

10

【0441】

コンテンツサーバー23070は、双方向サービスを提供するために必要なアプリケーション及びTDOを提供することができる。新しいアプリケーション又はTDOが必要な場合、新しいアプリケーションは、TPT又はアプリケーションパラメータテーブルにあるURLを用いてダウンロードすることができる。

【0442】

シグネチャー抽出機23080は、放送局23010からメタデータを受信することができる。シグネチャー抽出機23080は、受信されたメタデータからフレームのシグネチャーを抽出することができる。FPサーバー23090に伝送されたフィンガープリントがシグネチャー抽出機23080のシグネチャーとマッチングされると、シグネチャー抽出機23080は、シグネチャーと関連したメタデータをFPサーバー23090に伝送することができる。

20

【0443】

FPサーバー23090は、シグネチャー抽出機23080とシグネチャーマッチング動作を行うことができる。FPサーバー23090は、シグネチャーを受信機23040から受信されたフィンガープリントとマッチングさせることができる。シグネチャーがフィンガープリントとマッチングされると、FPサーバー23090は、シグネチャー抽出機23080からシグネチャーと関連したメタデータを受信することができる。FPサーバー23090はメタデータを受信機23040に伝送することができる。

30

【0444】

フィンガープリンティング(FP)技法において、FPクライアント23050は、オーディオ又はビデオフレームからフィンガープリント(シグネチャーと呼ぶこともできる。)を抽出し、多重の放送局からの放送フレームに対するフィンガープリントデータベースに対してこのフィンガープリントを検査し、受信機23040が必要とする情報を探すことができる。このような検査は、遠く離れたサーバーに対するシグネチャーによってなされてもよく、所望の情報を有する記録を受けることができ、一部の場合は、受信機23040にダウンロードされたシグネチャーのデータベースに対して検査することによって検査がなされてもよい。ここで、遠く離れたサーバーは、FPサーバー23090であってもよい。

40

【0445】

ウォーターマーキングとフィンガープリンティングは互いに区別される技術であるが、必ずしも互いに排他的なものではない。これらの両技術の組合せを用いることも考えることができる。自動コンテンツ認知(automatic content recognition; ACR)は、これらの両技術のいずれか一方を別途に指したり、又はこれらの両技術の組合せを指すために用いることができる。

【0446】

50

受信機が放送ストリームからの非圧縮されたオーディオ及びビデオにのみ接近できる環境を“ACR環境”と呼ぶ。

【0447】

WMの場合もFPの場合も、受信機は、トリガーを含む双方向サービスコンテンツを得るための出発点としてURLを用いることができる。

【0448】

WM、FPの両方の場合とも、インターネットを介して伝達されたトリガーの活性化タイムスタンプなどのように、当該チャンネルに対して時間臨界的(time critical)イベントの放送側クロック(broadcast side clock)に対するタイムスタンプの形態であってもよい。

10

【0449】

放送局は、一般にアンテナからTV信号を受ける受信機のために放送ストリームでの双方向サービスの直接伝達を支援でき、非圧縮されたオーディオ及びビデオを受ける受信機のために、前述した通り、インターネットを通じた双方向サービスの伝達を支援するものの、インターネット接続を有することができるものと仮定する。しかし、放送局は、この2つの伝達メカニズムのいずれか一方のみ支援できる。

【0450】

ウォーターマークがコード値のみを提供する場合、ウォーターマーキング技法のための代表的なアーキテクチャは、図22及び図23の2つのアーキテクチャの組合せのように見える。図22のようにウォーターマーク挿入部が存在するようになるが、この挿入部は、受信機が必要とする情報ではなくコードを挿入する。また、図23のFPサーバーと同じ役割を担うWMサーバーが存在する。受信機は、このサーバーにシグネチャーではなくコードを送り、これら受信機が必要とする情報を受ける。

20

【0451】

双方向サービス接近について説明する。

【0452】

双方向サービス接近についての説明は、直接実行モデル、ACRサーバーに独立した活性化を有するTDOモデル、ACRサーバーから受信された活性化を有するTDOモデルについての説明を含む。モデルを示してはいないが、モデルは、説明に限定されず、設計者の意図によって変更されてもよい。

30

【0453】

放送局選択とACRシステムの特性によってACR環境にある受信機が双方向サービスに接近できる方法には、様々なものがある。双方向サービスモデルは、直接実行モデル又はTDOモデルであり、TDOモデルの場合、活性化とトリガーは、ACRサーバーに独立して又はACRサーバーによって伝達することができる。

【0454】

直接実行モデルについて説明する。

【0455】

直接実行モデルを有する双方向サービスを含む仮想チャンネルのためのACR過程は、このチャンネルを見ている受信機に、media_time(“m=”)項とcontent_id(“c=”)項を含む時間ベーストリガーに相応するものを提供することができる。これらのトリガーは、直接実行モデルを有する双方向サービスのためのトリガーと確認できる。

40

【0456】

まず、受信機が新しいlocator_partを有するこのようなトリガーを受信すると、そのブラウザーに、トリガーのlocator_partが指す宣言的客体(DO)をローディングすると考えることができる。一般に、DOは、あらかじめ設置されていたり、あらかじめダウンロードされてキャッシュに格納されたりする。そうでなければ、受信機がHTTP GET要求を用いてDOをダウンロードすると考えることもできる。

【0457】

50

その後、DOは、適宜のバックエンドサーバーと連絡して、バックエンドサーバーが指示する双方向サービスを提供することができる。

【0458】

受信機は、ACRサーバーから新しいlocator_partを有する及び/又はTDOモデルを有する双方向サービスのためのトリガーとして確認されるトリガーを得るまで(これらのいずれか一つの場合は、一般的にチャンネルの変更を指示する)取得された初期トリガーと続くトリガーをDOが利用できるようにするものと考えられることができる。

【0459】

以下、ACRサーバーに独立した活性化を有するTDOモデルについて説明する。

【0460】

TDOモデルを有する双方向サービスを含むことができ、ACRサーバーと独立してイベント活性化を提供する仮想チャンネルのためのACR過程は、media_time(“m=”)項を含み得る時間ベーストリガーに相応するものを、このチャンネルを見ている受信機に提供することができる。これらのトリガーは、TDOモデルを有する双方向サービスのためのトリガーと確認できる。

【0461】

まず、受信機が新しいlocator_partを有するこのようなトリガーを受信すると、当該トリガーのlocator_partが指し得るTPTサーバーで現在のTDOパラメータテーブル(TPT)を検索し、このトリガー及び続くトリガー内のメディア時間を用いて、ACR過程によって識別されうるオーディオ又はビデオフレームに対してイベント活性化のための基準時間ベースを設定すると考えることができる。

【0462】

AMT(活性化メッセージテーブル)がTPTと併せて伝達される場合、受信機は、トリガー内のメディア時間によって設定された時間ベースを基準に正確な時間にイベントを活性化させるためにこのテーブル内のそれぞれの活性化エレメントを使用すると考えることができる。(これらのイベントは、TDO DMLローディング及び実行、TDOが同期化される特定動作を行うようにすること、TDOを中止させること、などを含むことができる。)

【0463】

TPTにLiveTriggerエレメントが含まれた場合、受信機はLiveTriggerエレメントでシグナリングされたポーリング方法を用いてLiveTriggerエレメント内のURLによって識別されたライブトリガーサーバーから活性化トリガーを検索し、この活性化トリガーを用いて、このトリガー内のメディア時間項によって設定された時間ベースを基準に、正確な時間にイベントを活性化させることができる。

【0464】

AMTとライブトリガーサーバーの両方とも同一のサービスのために用いることができるが、前者は静的活性化を提供し、後者は動的活性化を提供する。これと違い、セグメントの全ての活性化が静的活性化である場合は、AMTのみを用いてもよく、ライブトリガーサーバーのみを用いて静的活性化及び動的活性化の両方を伝達してもよい。

【0465】

ACRサーバーから受信された活性化を有するTDOモデルについて説明する。

【0466】

TDO双方向サービスモデルのための活性化トリガーが、ACR環境において、別途のトリガーサーバー無しでどのように伝達されるかについて記述する。

【0467】

フィンガープリンティングACRシステムは、ACRサーバーを含むことができる。受信機は、ACRサーバーにフレームシグネチャーを送ることができ、ACRサーバーは、当該シグネチャーが示すフレームを識別して、再び受信機で必要とする情報を送ることができる。ウォーターマークが、受信機で必要とする情報を得るためにACRサーバーに伝達され得るコードのみで構成された場合、ウォーターマーキングACRシステムはACR

10

20

30

40

50

サーバーを含むことができる。ウォーターマーク自体が受信機で必要とする情報を含む場合、ウォーターマーキング ACR システムは ACR サーバーを含まなくてもよい。ACR サーバーを含む ACR システムでは、サーバーと受信機間の通信に互いに異なる 2 のモデル、すなわち、要求 / 応答モデルとイベント主導型モデルを用いることができる。

【0468】

放送局で TDO 双方向モデル (TDO interaction model) を支援していると仮定する。

【0469】

要求 / 応答モデルを用いる ACR サーバー、イベント主導型モデルを用いる ACR サーバー、情報を直接挿入するウォーターマーキング ACR システムのような 3 つの場合を想定することができる。

10

【0470】

ACR サーバーがある場合、ACR 方式は、フィンガープリンティングであってもよく、この場合、受信機は一種のオーディオ又はビデオフレームのシグネチャー (又は、フィンガープリント) を算出し、識別のためにそれらを ACR サーバーに提出する。又は、ACR 方式がウォーターマーキングであってもよく、この場合、受信機はオーディオ又はビデオフレームからウォーターマーク形態のコードを抽出し、これらのコードを識別のために ACR サーバーに提出する。

【0471】

便宜のためにフィンガープリンティングシグネチャーの用語を説明する。ただし、システムはウォーターマーキングコードの場合と同一の方式で動作し、本発明はフィンガープリンティングに限定されない。

20

【0472】

図 24 は要求 / 応答 ACR 場合の静的活性化の一実施例を示す図である。

【0473】

ACR サーバーが要求 / 応答モデルを使用する場合について説明する。

【0474】

要求 / 応答 ACR モデルで、受信機は周期的に (例えば、5 秒ごとに (これは、例示に過ぎず、設計者によって変更されてもよい)) コンテンツのシグネチャーを生成し、シグネチャーを含む要求を ACR サーバーに送ると考えることができる。ACR サーバーは、受信機から要求を受けると応答を送ることができる。要求 / 応答インスタンス間には通信セッションが開放状態に維持されない。このモデルでは、ACR サーバーが受信機にメッセージを開始することが不可能であってもよい。

30

【0475】

この要求 / 応答モデルを用い、受信機に活性化トリガーを伝達する ACR サーバーの場合、ACR サーバーからの各応答は、ヌル (Null)、時間ベーストリガー又は活性化トリガーのうち一つであってもよい。

【0476】

ヌル応答は、シグネチャーが認識されないことを示したり、又は、(ACR インジェストモジュール (ACR Ingest Module) がフレームに対するシグネチャーを双方向サービスのないプログラムセグメントに含める場合に) シグネチャーは関連している双方向サービスのないセグメントに属するフレームを意味することを示すことができる。ACR インジェストモジュールについては後述する。

40

【0477】

時間ベーストリガー応答は、クライアントの次の要求がある前にイベント活性化が発生するようにスケジューリングされていないことを示すことができる。クライアントは、時間ベーストリガーを用いてメディア時間クロック (media-time clock) を維持すると考えることができる。

【0478】

活性化トリガー応答は、活性化がまもなく発生する予定であることを示すことができ、

50

活性化の時間は、トリガー内の“ t = ”項によって示される。

【 0 4 7 9 】

受信機は、新しい `locator_part` を有するトリガーを受け度、以前の TPT と共に伝達された `URLList` を用いて新しい TPT が検索されていないと、それを直ちにダウンロードすると考えることができる。

【 0 4 8 0 】

受信機は、活性化トリガーを受け度、メディア時間クロックに対して当該トリガー内の“ t = ”項によって示された時間に、当該イベントを活性化させると考えることができる。

【 0 4 8 1 】

図 2 4 は、静的活性化に対して (ACR システムが予定よりも十分に早く動的活性化について知っている場合は動的活性化に対して)、この方式がどのように作用するかを示している。

【 0 4 8 2 】

図 2 4 で、受信機は、ACR サーバーでメディア時間 `MT1`、`MT2` 及び `MT3` を有すると判断したフレームに対するシグネチャーを伝達することができる。メディア時間 `MT1` を有するフレームの場合、受信機は、時間ベーストリガーを含む応答を受け度。メディア時間 `MT2` を有するフレームの場合、静的活性化がメディア時間 `Mta` に予定され、受信機は“ t = `MTa` ”項を有する活性化トリガーを含む応答を受け度。メディア時間 `MT3` を有するフレームの場合、受信機は、時間ベーストリガーを含む応答を受け度。

【 0 4 8 3 】

受信機が同一のイベント活性化に対して一つ以上の活性化トリガーを受信する場合が発生しうる。しかし、これらのそれぞれに対するメディア時間は同一であるため、受信機はそれらの活性化トリガーが重複していると把握し、これらのうち一つのみを適用することができる。

【 0 4 8 4 】

図 2 5 は、要求 / 応答 ACR ケースにおける静的活性化の一実施例を示す図である。

【 0 4 8 5 】

ACR サーバーが要求 / 応答モデルを使用する場合について説明する。

【 0 4 8 6 】

図 2 5 で、受信機は、ローカルクロック (`local_clock`) 時間 `LC1`、`LC2`、`LC3` で見られたフレームに対するシグネチャーを伝達することができる。ローカルクロック時間 `LC1` に見られたフレームに対する `media_time` は ACR サーバーによって `MT1` であると判断でき、受信機は、`media_time` 又は `event_time` のないトリガーを含む応答を受信する。ローカルクロック時間 `LC2` に見られたフレームに対する `media_time` は ACR サーバーによって `MT2` であると判断でき、ACR サーバーは、メディア時間 `Mta` に静的活性化が予定されていることを知っているため、ACR サーバーは、この活性化のための活性化時間 `Mta` が `MT2` 以降の < オフセット > 時間であることを意味する“ d = < `offset` > ”項を有する活性化トリガーを含む応答を伝達する。すると、受信機は < オフセット > を時間 `LC2` に追加し、当該イベントを活性化させるべきローカル時間である `Lca` を得る。

【 0 4 8 7 】

図 2 6 は、要求 / 応答 ACR ケースにおける動的活性化の一実施例を示す図である。

【 0 4 8 8 】

以下、要求 / 応答 ACR 場合において動的活性化が起きる場合について説明する。

【 0 4 8 9 】

あらかじめ受信機にトリガーを送るには手遅れになるまでずっと、ACR システムでイベント活性化について知らない状況での動的活性化の場合には、ACR サーバーが次の要求があるまで待ってから活性化トリガーを送らなければならない。このような場合を図 2 6 に示す。そのため、一つの要求区間だけ動的活性化が遅延されうる。

10

20

30

40

50

【0490】

図26で、受信機は、ACRサーバーでメディア時間MT1、MT2及びMT3を有すると判断したフレームに対するシグネチャーを伝達することができる。メディア時間MT1及びMT2を有するフレームの場合、受信機は、時間ベーストリガーを含む応答を受ける。活性化時間Mtaを有する動的活性化がメディア時間Mtaで又は直前に登場する場合、ACRサーバーは、メディア時間MT3を有するフレームに対して発生する受信機からの次の要求があるまで、受信機にそれを通知することができない。このとき、ACRサーバー応答は、活性化時間Mta（やや去の時間）を有する活性化トリガーを含む。このような状況で、受信機は、その応答が到達すると直ちに当該活性化トリガーを適用すると考えることができる。

10

【0491】

ここで、受信機は、同一のイベント活性化に対して一つ以上の活性化トリガーを受信することが可能である。しかし、それらのそれぞれのメディア時間は同一であるため、受信機はそれらが重複したと把握し、それらのうち一つのみを適用できる。

【0492】

図27は、要求/応答ACRケースにおける動的活性化の一実施例を示す図である。

【0493】

要求/応答ACRケースにおいて動的活性化が起きる場合について説明する。

【0494】

図27で、受信機は、ローカルクロック時間LC1、LC2、LC3で見られたフレームに対するシグネチャーを伝達することができる。ローカルクロック時間LC1に見られたフレームに対するmedia__timeはACRサーバーによってMT1であると判断でき、受信機はmedia__time又はevent__timeのないトリガーを含む応答を受信する。ローカルクロック時間LC2に見られたフレームに対するmedia__timeはACRサーバーによってMT2であると判断でき、ACRサーバーは、メディア時間Mtaに動的活性化が登場する予定であることを知っておらず、受信機は、media__time又はevent__timeのないトリガーを含む応答を受信する。動的活性化がメディア時間Mtaに登場すると、ACRサーバーは、ローカル時間LC3に発生する受信機からの次の要求があるまで、受信機にそれに対して通知することができない。このとき、ACRサーバー応答は、負の<オフセット>値を有する活性化トリガー又は“今施行”の活性化トリガーを含むことができる。

20

30

【0495】

イベント主導型モデルを使用するACRサーバーについて説明する。

【0496】

イベント主導型ACRモデルで、受信機はACRサーバーに対する永久的な接続を開始し、周期的に（例えば、5秒ごとに）コンテンツのシグネチャーを生成し、シグネチャーを接続を通して提出すると考えることができる。ACRサーバーは各シグネチャーに反応しない。それは、新しいセグメントが発見されたり、又はイベント活性化が受信機に伝達される必要があるとき、受信機にメッセージを送ることができる。当該モデルで、ACRサーバーはいつでもクライアントに対するメッセージを開始することができる。

40

【0497】

該イベント主導型モデルを使用し、活性化を受信機に伝達するACRサーバーに対して、次の規則はACRサーバーからのメッセージを申し込むことができる。

【0498】

第一に、ACRサーバーは、新しいセグメントに対応する受信機からシグネチャーを受信すると、単に受信機が関連のTPTを取得できるようにするために、直ちに時間ベーストリガーを受信機に送ることができる。

【0499】

第二に、イベントが活性化される予定である度に、ACRサーバーは活性化トリガーを受信機に送ることができる。可能ならば、受信機が適用すべき時期よりもやや早く活性化

50

トリガーを送ることができる（これは、要求/応答モデルにおける動作と略同様である）。ACRサーバーは、活性化をあまりにも遅く知って活性化トリガーを事前に早く送ることができなくても（これは、動的イベント活性化の場合に発生しうる）、活性化トリガーを可能な限り早く送ることができる。後者の場合に、メッセージ遅延によってクライアントは活性化時期よりもやや遅れてメッセージを受け取ることができ、この場合、受信機はメッセージを受け次第イベントを活性化すると考えることができる。

【0500】

受信機は、新しいlocator_partを有するトリガーを受信する度に、以前のTPTと一緒に伝達されるURLListを用いて既に検索していない限り、新しいTPTを直ちにダウンロードすると考えることができる。

10

【0501】

情報を直接挿入するウォーターマーキングACRシステムについて説明する。ウォーターマーキングACRシステムを示してはいないが、ウォーターマーキングACRシステムは、次の説明に限定されず、設計者によって変更されてもよい。

【0502】

受信機が必要とする情報を直接挿入するウォーターマーキングシステムの場合、フレームと関連しているウォーターマークは、このフレームに対して要求/応答ACRサーバーがリターンすることについて、前述したのと同じ規則を、次のように従うことができる。要求/応答ACRサーバーは、ヌル、時間ベーストリガー、活性化トリガーのうち一つをリターンすることができる。

20

【0503】

ヌル応答は、シグネチャーが認識されないことを示したり、又は（ACRインジェストモジュールがフレームに対するシグネチャーを双方向サービスのないプログラムセグメントに含める場合）シグネチャーは関連した双方向サービスのないセグメントに属するフレームを意味することを示すことができる。

【0504】

時間ベーストリガー応答は、クライアントの次の要求がある前にイベント活性化が発生するようにスケジューリングされていないことを示すことができる。クライアントは、時間ベーストリガーを用い、メディア時間クロックを維持すると考えることができる。

【0505】

活性化トリガー応答は、活性化がまもなく発生する予定であることを示すことができ、活性化の時間はトリガー内の“t=”項によって示される。

30

【0506】

ACRサーバーが必要とされないように、受信機が必要とする情報をウォーターマークに直接含めて伝達するウォーターマーキングACRシステムの場合、インジェストモジュールは、各フレームと関連付けるトリガーを決定し、このトリガーをデータベース内の該当フレームと関連付けずに当該フレームのためのウォーターマークに含める要求/応答サーバーモデルについて、前述したのと同じ規則に従うことができる。インジェストモジュールとデータベースについては後述する。

【0507】

独立型NRTサービスの支援について説明する。これを示してはいないが、本発明は、次の説明に限定されず、設計者によって変更されてもよい。

40

【0508】

ACR環境にある受信機が独立型NRTサービスを受けるために、放送局は、NRTサービスへのインターネット接続を支援する必要がある、受信機は、それらのサービスに対するSMTとNRT-ITインスタンスを得る必要がある。

【0509】

インターネットを介してPSIPテーブルとNRTテーブルを得るための質疑(query)プロトコルについて説明する。

【0510】

50

放送局は、特定放送ストリームに対してこのプロトコルを支援し、このとき、この放送ストリームに対する放送局のシグナリングサーバーのURLを知っている受信機は、次の段階を行う。

【0511】

第一に、受信機は、指定された未来時間区間（例：次の時間）に当該放送ストリームのためのテーブルの“基本NRT集合”に対して質疑を発生させることができる。

【0512】

第二に、上記の段階で各独立型NRT仮想チャネルに対するSMT及びILT、並びに指定された時間区間を含むNRT-IT及びTFTインスタンスを生成する。

【0513】

受信機が放送ストリームに対するシグナリングサーバーのURLを発見できる一つの方法は、この放送ストリーム内の双方向サービスセグメントのプロバイダが、TPTと併せて伝達されたURLListエレメント内でシグナリングサーバーURLを提供するように選択することができる。

10

【0514】

受信機が放送ストリームに対するシグナリングサーバーのURLを発見できるもう一つの方法は、事前設定によるものであってもよい。DTV受信機メーカーが、特定放送領域に到達するACRサーバーをどのように探すかを知るようにDTV受信機をあらかじめ設定するのと同じ方式で、DTV受信機メーカーが、特定放送領域に到達する“NRT発見サーバー”をどのように探すかを知るようにDTV受信機をあらかじめ設定することができる。このようなNRT発見サーバーは、各放送ストリームに対するシグナリングサーバーURLと共に、独立型NRTサービスを含む放送ストリーム目録を受信機に提供することができる。

20

【0515】

図28は、ACRサーバー活性化のためのアーキテクチャの一実施例を示す図である。

【0516】

あるACRシステムはACRサーバーを含み、あるACRシステムは含まない。フィンガープリンティングACRシステムでは、受信機はフレームシグネチャーを算出してACRサーバーに送ることができ、ACRサーバーは、受信機で必要とする情報を送る。したがって、フィンガープリンティングACRシステムは、ACRサーバーを含む。ウォーターマーキングACRシステムでは、ウォーターマークにフレームを固有に識別するコードのみが含まれてもよく、又は受信機が必要とする全体情報が含まれてもよい。ウォーターマークがコードのみを含む場合は、受信機がコードを抽出し、それらをACRサーバーに送ることができ、ACRサーバーは、受信機で必要とする情報を送る。ウォーターマークが全体情報を含む場合は、受信機が自分に必要な情報をウォーターマークから直接抽出でき、ACRサーバーが必要でない。

30

【0517】

ACRサーバーを含むACRシステムでは、ACRサーバーと受信機間の通信のために2種類の互いに異なるモデル、すなわち、要求/応答モデルとイベント主導型モデルを共通に使用することができる。

40

【0518】

要求/応答ACRサーバーモデルにおいて、受信機は周期的に（例えば、5秒ごとに）コンテンツのシグネチャーを算出したりコンテンツからコードを抽出し、このシグネチャーやコードが含まれた要求をACRサーバーに送ると考えることができる。ACRサーバーは受信機から要求を受信すると、応答をリターンすることができる。要求/応答インスタンス間には通信セッションが開いた状態で維持されない。このモデルでは、ACRサーバーが受信機にメッセージを開始することが実行可能でなくともよい。

【0519】

処理しているチャンネルの放送局がTDO双方向モデルを支援すると仮定する。

【0520】

50

2種類の一般的な種類のイベント活性化、すなわち、セグメントの放送が開始される前に活性化時間が知らされる静的活性化と、セグメントの放送中に活性化時間が動的に決定される動的活性化がありうる。あらかじめ記録されたセグメントでは、全てのイベント活性化が静的であってもよい。ライブショーを放送するセグメントでは、イベント活性化の一部又は全てが動的であってもよい。静的活性化は活性化トリガーの形態で受信機に伝達されてもよいが、一般的に活性化メッセージテーブル(Activation Messages Table、AMT)に記載される。動的活性化は、これらのタイミングがAMTの発生時点に知らされないため、活性化トリガーの形態で伝達されてもよい。

【0521】

図28は、ACRサーバーを使用するACRシステムを支援するアーキテクチャを示している。これは論理構成図であって、実行アーキテクチャではない。例えば、ACRインジェストモジュールは、放送ソースと同じ位置にあってもよく、別途の位置にあってもよい。

10

【0522】

ACRサーバーを使用するACRシステムを支援するアーキテクチャにおいて、アーキテクチャは、放送ソース28010、ACRインジェストモジュール28020、MVPD 28030、STB 28040、受信機28050、ACRクライアント28060、ACR設定サーバー28070、ACRサーバー28080及びデータベース28090で構成することができる。

【0523】

放送ソース28010は、A/Vストリーム及び関連した双方向サービスが送出される地点であって、例えば、ネットワーク分配地点又はTV放送局であってもよい。

20

【0524】

ACRインジェストモジュール28020は、フィンガープリンティングACRシステムの場合、フレームのシグネチャー(フィンガープリント)を算出したり、コードに基づくウォーターマーキングACRシステムの場合、コードからなるウォーターマークをフレームに挿入することができる。このモジュールは、他のメタデータと併せてシグネチャー又はコードに関連している各フレームのmedia_timeをデータベース28090に保存することができる。ACRインジェストモジュール28020は、一つの放送ストリーム、全体放送ストリーム、又は複数の放送ストリーム又はこれらの組合せにおいて単一チャンネルを扱うことができる。そのために、ACRインジェストモジュール28020は、双方向サービスを含むプログラムセグメントのためのフレームを処理すると仮定する。しかし、全てのフレームが処理されるが、双方向サービスを有するセグメントの一部でないものは、これらが双方向サービスを有するセグメントの一部でないことを示す指示子がデータベース28090エントリにあるACRシステムも可能である。

30

【0525】

MVPD 28030は一般に、ケーブル事業者、衛星事業者又はIPTV事業者である。MVPD 28030は、ウォーターマーキングACRシステムの場合、ACRインジェストモジュール28020によって挿入されたウォーターマークと共に放送ストリームをいずれかの方式で放送ソースから受信することができる。このようなシステムはたびたびオーディオ及びビデオトラック以外の全てのプログラムエレメントを除くストリームを顧客の宅内のSTB 28040に送る。

40

【0526】

STB 28040は一般にオーディオ及びビデオをデコーティング(圧縮解除)し、これらを、視聴者に視聴されるTV受像機に送る。双方向サービストリガーを含むDTVクロードキャプションサービス#6は、TV受像機が利用できないと仮定する。

【0527】

受信機28050は、ACRクライアント28060を含むことができる。ACRクライアント28060は、受信機28050の外部に位置してもよい。受信機28050の構造については後述する。

50

【 0 5 2 8 】

受信機 2 8 0 5 0 内の A C R クライアント 2 8 0 6 0 は、当該用途の A P I を用いて A C R サーバ ー 2 8 0 8 0 から活性化トリガ ーを受け、これらをメイン受信機コードに送ることができる。一般に、A C R クライアント 2 8 0 6 0 は受信機 2 8 0 5 0 内に構成されるが、他の構成図可能である。

【 0 5 2 9 】

A C R 設定サーバ ー 2 8 0 7 0 は、A C R クライアント 2 8 0 6 0 が適切な A C R サーバ ー 2 8 0 8 0 の位置を決定する方式を提供することができる。この発見過程は他の方式で得ることもできる。

【 0 5 3 0 】

A C R サーバ ー 2 8 0 8 0 は、受信機からシグネチャ ー又はコードを受け、適切な時点で活性化トリガ ーをリターンすることができる。

【 0 5 3 1 】

データベース 2 8 0 9 0 は、A C R サーバ ー 2 8 0 8 0 の使用のためにオーディオ又はビデオフレームのいずれか一方（又は、両方）に関する情報が格納される一種のデータ格納所であって、厳密にはデータベースでなくてもよい。

【 0 5 3 2 】

ウォーターマ ーク内の情報の直接伝達を用いる A C R システムのアーキテクチャは、データベースと A C R サーバ ーを有しなくてもよい。A C R インジェストモジュールは、情報をフレームの識別子及びこれらと関連している情報を含むデータベース記録に挿入する代わりに、ウォーターマ ークの形態で放送ストリーム内のフレームに直接挿入してもよい。その後、受信機はその情報を A C R サーバ ーから取得せず、放送内のフレームから抽出することができる。

【 0 5 3 3 】

以下、要求 / 応答 A C R サーバ ーを介した活性化トリガ ーの伝達について段階別に説明する。

【 0 5 3 4 】

この A C R サーバ ーの挙動を具現する効果的な方法は、以下に説明する過程に従うものであるが、この過程の動作の数は、上記のアーキテクチャダイアグラムである図 2 8 内の数字に対応する。

【 0 5 3 5 】

1) 双方向サービスセグメントと A M T 又は各セグメントに相当するものに対する放送スケジュールを、セグメントが放送される前にあらかじめ A C R インジェストモジュールに伝達することができる。放送スケジュールは、セグメント I D、関連する双方向サービスを含み得る各セグメントの G P S 開始時点及び G P S 終了時点を含むことができる。放送スケジュールに最後の瞬間の変化がある場合、A C R インジェストモジュールにそれらの変化が直ちに通知されてもよい。放送スケジュールは、各セグメントに対する T P T のバージョンナンバーも含むことができ、T P T バージョンのスケジュールされていない変更は、実時間で A C R インジェストモジュールに通知され、必要時に “ v e r s i o n ” (“ v = ”) 項をトリガ ーに挿入することができる。インジェストモジュールは、各セグメントの開始時に特定区間（多数の受信機が新しい T P T を同時に要求する可能性がある時）のように適切な時間に “ s p r e a d ” (“ s = ”) 項をトリガ ーに挿入するように構成されてもよい。

【 0 5 3 6 】

2) 動的活性化が存在する場合、動的活性化のソースから A C R インジェストモジュールにリンクが設定されてもよい。

【 0 5 3 7 】

3) 放送ストリームは A C R インジェストモジュールに送信されてもよい。

【 0 5 3 8 】

4) A C R インジェストモジュールは、フレームからシグネチャ ーを抽出したり（フィ

10

20

30

40

50

ンガープリント A C R システムの場合)、関連している双方向サービスを持つセグメントに含まれた全てのフレームに対してコードをフレームに挿入(ウォーターマーク A C R システムの場合)することができる。(A C R インジェストモジュールは、 G P S クロックと放送スケジュール内のセグメントの開始時間及び終了時間を用いてフレームがそのようなセグメントにあるかを判断できる。)そのような各フレームに対して A C R インジェストモジュールは、トリガーとシグネチャー又は当該フレームと関連しているコードを含み得るデータベースに記録を挿入することができる。トリガーが挿入されることに関する規則は、過程上、このような動作リストの終端に記述される。

【 0 5 3 9 】

5) 放送ストリームは、 M V P D に続いてよい。

10

【 0 5 4 0 】

6) M V P D 加入者の位置にある S T B に放送ストリームを送信できる。(一般に、双方向コンテンツはまず除く。)

【 0 5 4 1 】

7) S T B は、 A / V をデコーティングして非圧縮された A / V を D T V 受信機に送ることができる。

【 0 5 4 2 】

8) まず、受信機が O N になると、受信機は自身の位置を A C R 設定サーバーに送ることができる。(A C R 設定サーバーの U R L は、この受信機内に構成されてもよい。)

【 0 5 4 3 】

20

9) A C R 設定サーバーは、受信機が使用する A C R サーバーの U R L を送ることができる。

【 0 5 4 4 】

10) 受信機内の A C R クライアントは、フィンガープリントシグネチャー又はウォーターマークコードを抽出し、それらを A C R サーバーに送ることができる。

【 0 5 4 5 】

11) シグネチャー又はコードを受信した A C R サーバーは、データベース内でシグネチャー又はコードのマッチングを試みることができる。

【 0 5 4 6 】

12) シグネチャー又はコードがデータベースのいずれのシグネチャー又はコードともマッチングされないと、 A C R サーバーは “ n o m a t c h ” 指示子を受けることができる。シグネチャー又はコードがデータベース内のシグネチャー又はコードとマッチングされると、 A C R サーバーは、マッチングされるシグネチャー又はコードを有するフレームに対する記録を受けることができる。後者の場合、データベース内の記録は時間ベーストリガーを含むことができ、及び/又は A C R インジェストモジュールによってフレームの記録に何が挿入されたかによって一つ以上の活性化トリガーを含むことができる。

30

【 0 5 4 7 】

13) A C R サーバーがデータベースから “ n o m a t c h ” 指示子を受けると、 A C R クライアントにヌル応答をリターンすることができる。そうでない場合、 A C R サーバーは、得られたトリガー又はトリガーを A C R クライアントにリターンすることができる。

40

【 0 5 4 8 】

下記の規則を用いて、 A C R インジェストモジュールがどの一つ又は複数のトリガーをデータベース内の各フレーム記録に挿入するかを判断できる。

【 0 5 4 9 】

図 29 は、 E n d T i m e が ない 場合 (a) と 場合 (b) に お ける 活 性 化 トリガーの一実施例を示す図である。

【 0 5 5 0 】

受信機内の個別 A C R クライアントによって使われた要求区間の長さ上に、ある上限 (u p p e r b o u n d) L 1 が存在すると仮定できる。(A C R クライアントが実際に

50

この上限内で動作する限り、この境界を知っているかは重要でない。) L2をフレームが受信機に到達した時点から計算して、一般のACRクライアントがシグネチャーを算出したり、このフレームと関連しているウォーターマークを抽出するためにかかる時間の長さとしよう。メッセージがACRクライアントからACRサーバーに送られてから帰ってくるためにかかる一般的な往復移動時間をL3としよう。M = L1 + L2 + L3とする。(ややより大きいMの値が用いられもよい。 - ややより大きい値の長所は、活性化トリガーに反応する上で若干の余分の時間を得るということである。短所は、受信機が同一のイベント活性化に対して複数の活性化トリガーを受けの可能性がややより高いということである。受信機は、後述するように、これらが重複することを感知し、活性化を1回のみ適用できるため、この短所は大きく問題にならない。)

10

【0551】

ACRインジェストモジュールは、次の3つの条件のうち少なくとも一つが成立しないと、時間ベーストリガーのみをフレームと関連している記録に挿入することができる。

【0552】

(a) 活性化エレメントは、フレームのmedia__timeが活性化エレメントのstartTime前の時間長さMから始まって活性化エレメントのendTimeで終わる時間区間にあるようにAMTに存在する。(活性化がendTimeを有しないと、endTimeはstartTimeと同一のものと見なされる。)

【0553】

(b) トリガーの活性化時間(“t = <event__time>”)直前の時間長さMの時間区間以前に動的活性化トリガーがインジェストモジュールによって受信されており、当該フレームはこの区間内に存在する。

20

【0554】

(c) トリガーの活性化時間直前の時間長さMの区間の開始よりも遅れて動的活性化トリガーがインジェストモジュールによって受信されており、フレームのmedia__timeはトリガーの受信の直後の時間長さL1の区間に存在する。

【0555】

これらの条件(a)、(b)及び(c)のいずれか一つでも成立すると、活性化トリガーは、活性化されるイベントを識別する“e = ”項とAMT内の活性化エレメントのstartTimeを指示したり(条件(a)の場合)、動的トリガーのevent__timeを指示する(条件(b))“t = ”項と共に記録に含まれる。トリガーはバージョン(“v = ”)項も含むことができる。

30

【0556】

もちろん、(a)の場合、startTimeからendTimeまでの区間を通じて続けて活性化トリガーをフレームと関連付ける理由は、この区間を通じてチャンネルに参加する受信機をある程度受容するためである。

【0557】

この接近法は、ACRサーバー側に追加的な知能を要求しないという点に留意されたい。このサーバーは、単純にデータベースで発見する情報をACRクライアントにリターンする。全ての知能はACRインジェストモジュールに存在できる。しかも、ACRインジェストモジュールが行うべき演算が非常に簡単でありうる。

40

【0558】

この方式を用いると、受信機で同一のイベント活性化に対して一つ以上の活性化トリガー(互いに異なるフレームと関連しているトリガー)を受信することが可能である。しかし、受信機は、“t = ”値から、それらのトリガーがいずれも同一の活性化時間を有することがわかるため、受信機はそれらが重複すると判断し、イベントを1回のみ活性化させることができる。

【0559】

上記の場合のうち二つの場合に、活性化トリガー内の“t = ”項は、関連しているフレームのmedia__timeより早いevent__timeを有することができる。(a

50

) の状況で、活性化エレメントの `endTime` が `startTime` よりも非常に遅いと、受信機は一般的に `startTime` と `endTime` の間の区間を通じて複数の活性化トリガーを受け取ることができ、これらはいずれも活性化時間として `startTime` を有することができる。(c) の状況では、活性化のための活性化トリガーがあまりにも遅くフレーム記録に挿入され、受信機が受ける活性化トリガーが要求に対する応答として、活性化時間以降の `media_time` を有するフレームに対するシグネチャーと共にくることがある。受信機が、関連しているフレームの `media_time` よりも早い `event_time` を有する活性化トリガーを受けると、このトリガーが既に確認して当該イベントの活性化のために使用した活性化トリガーと同じものと認識しない場合、当該イベントを直ちに活性化すると考えることができる。

10

【0560】

フレームの `media_time` が `event_activation_time` よりも遅い場合に“今施行”トリガーではなく過去の `event_time` 値を用いる目的は、受信機がこのような“after the fact”活性化トリガーを一つ以上受け取ることができるためである。“t=”値は、受信機でこれらのトリガーがいずれも同一の活性化時間を有すると判断し、当該イベントを1回のみ活性化させ得るようにする。

【0561】

図29は、AMT内の活性化エレメントに `endTime` 属性がない状況(a)と状況(b)を説明する。

【0562】

20

図29は、AMT内の活性化エレメントが `endTime` を有しないとき、上記の動作(4)での状況(a)の一例を示している。これは、ACRインジェストモジュールに活性化時間よりも少なくともM時間単位以前に動的活性化トリガーが送られる上記の段階(4)での状況(b)の例であってもよい。

【0563】

図29は、タイムライン(`timeline`)上にイベント活性化時間とこれよりも先行する長さL1、L2及びL3の区間を含む長さMの区間を示している。タイムラインの下の垂直矢印は、個別フレームの時間を表す。長さMの区間の開始よりも先行したり、イベント活性化時間後にくる各フレームは、データベースで時間ベーストリガーと関連している。長さMの区間内の各フレームは、データベースにおいて同図の下における2つの例(f1、f2)のような活性化トリガーと関連している。各フレームに対する“t=”項は、`media_time` に対するイベント活性化時間を示すことができる。(丸囲みのf1、f2で表示されている。)

30

【0564】

丸囲みの4つの垂直矢印は、一般の受信機が要求を送る時の例示である。この例で、受信機は、同一のイベント活性化に対して2つの活性化トリガーを受けようになるが、これらのトリガーは同一のイベント活性化時間を有するため、受信機はこれらを同一のものと認識し、最初のトリガーのみを適用する。受信機の要求間の間隔がL1未満であるため、受信機は図示のL1区間でフレームに対するシグネチャーと併せて少なくとも一つの要求をするように保障される。これによって活性化時間以前にシグネチャーを算出してACRサーバーに要求を送り、その応答として活性化トリガーを受け取る時間を有することとなる。この例で、受信機が受ける一番目の活性化トリガーはあらかじめ伝達されるが、受信機が受ける二番目の活性化トリガーは、かろうじて間に合って到達する(このトリガーは重複トリガーである)。

40

【0565】

図30は、`EndTime` がない場合(a)と場合(b)における活性化トリガーの一実施例を示す図である。

【0566】

以下、`EndTime` がない場合(a)と場合(b)における活性化トリガーを説明する。これは、上記の図29に関する説明に似ている。

50

【0567】

図30は、AMT内の活性化エレメントがendTimeを有しない場合に、上の動作(4)で状況(a)の例を示している。これは、ACRインジェストモジュールに活性化時間よりも少なくともM時間単位以前に動的活性化トリガーが送られる上の段階(4)で状況(b)の例である。

【0568】

図30は、タイムライン(timeline)上にイベント活性化時間とこれよりも先行する長さL1、L2及びL3の区間を含む長さMの区間を示している。タイムラインの下矢印は、個別フレームの時間を表す。長さMの区間の開始よりも先行したりイベント活性化時間後にくる各フレームは、データベースで<media__time>又は<event__time>項のないトリガーと関連している。長さMの区間内の各フレームは、データベースで同図の下における2つの例と同一の活性化トリガーと関連している。各フレームに対する"d="項は、このフレームとイベント活性化時間との時間の長さを示すことができる。(丸囲みのf1、f2で表示されている)。

10

【0569】

丸囲みの4つの垂直矢印は、一般の受信機が要求を送る時の例示であってもよい。この例で、受信機は、同一のイベント活性化に対して2つの活性化トリガーを受け、<d1>値をフレームf1に対する受信機のローカル時間に加えたり、<d2>値をフレームf2に対する受信機のローカル時間に加えて算出した活性化時間は、同一の結果を有するため、受信機はそれらを重複したものと認識し、最初のトリガーのみを適用する。受信機要求間の間隔がL1未満であるため、受信機は図示のL1区間でフレームに対するシグネチャーと併せて少なくとも一つの要求をするように保障される。これによって活性化時間以前にシグネチャーを算出してACRサーバーに要求を送り、その応答として活性化トリガーを受け時間有することとなる。この例で、受信機で受信された二番目の活性化トリガーは、活性化時間以降に到達する。

20

【0570】

図31は、EndTimeがある場合(a)における活性化トリガーの一実施例を示す図である。

【0571】

図31は、AMT内の活性化エレメントがstartTimeの他、endTimeも持つ場合に、上の動作(4)で状況(a)の例を示している。

30

【0572】

同図は、タイムライン上にイベント活性化のstartTimeとendTime、及びstartTimeよりも先行する長さMの区間を示している。タイムラインの下矢印は、個別フレームの時間を表す。長さMの区間の開始よりも先行したりイベント活性化のendTime後にくる各フレームは、データベースで時間ベーストリガーと関連している。長さMの区間内又はイベント活性化のstartTimeとendTime間の各フレームは、データベースで同図の下における3つの例で示す形態で関連している活性化トリガーを有する。各フレームに対する"t="項は、media__timeに対するイベント活性化時間を示すことができる。(丸囲みのf1、f2、f3で表示されている)。

40

【0573】

丸囲みの3つの垂直矢印は、一般の受信機が要求を送る時の例示であってもよい。この場合、受信機は同一のイベント活性化に対して3つの活性化トリガーを受け、活性化時間が同一であるため、受信機はそれらを同一のものと認識し、最初のトリガーのみを適用する。

【0574】

もちろん、同図における最初の2つの活性化トリガーは、startTime以降にチャンネルに参加して最初の要求と併せてフレームf3のシグネチャーを送る受信機にとっては認知できない。

50

【 0 5 7 5 】

図 3 2 は、E n d T i m e がある場合 (a) における活性化トリガーの一実施例を示す図である。

【 0 5 7 6 】

以下、E n d T i m e がある場合 (a) における活性化トリガーを説明する。

【 0 5 7 7 】

図 3 2 は、A M T 内の活性化エレメントが s t a r t T i m e の他、e n d T i m e も有する場合に上の動作 (4) での状況 (a) の例を示している。

【 0 5 7 8 】

同図は、タイムライン上にイベント活性化の s t a r t T i m e と e n d T i m e 、及び s t a r t T i m e よりも先行する長さ M の区間を示している。タイムラインの下矢印は、個別フレームの時間を表す。長さ M の区間の開始よりも先行したりイベント活性化時間後にくる各フレームは、データベースで < m e d i a _ t i m e > 又は < e v e n t _ t i m e > 項のないトリガーと関連している。長さ M の区間内の各フレームは、データベースで同図の下における 2 例で示す形態の活性化トリガーを有する。各フレームに対する “ d = ” 項は、このフレームとイベント活性化時間との間の時間長を示すことができる。(丸囲みの垂直矢印で示されている)。

10

【 0 5 7 9 】

丸囲みの垂直矢印は、一般の受信機が要求を送る時の例示であってもよい。この場合、受信機は同一のイベント活性化に対して 3 つの活性化トリガーを受けるが、< d 1 > 値をフレーム f 1 に対する受信機のローカル時間に加えたり、< d 2 > 値をフレーム f 2 に対する受信機のローカル時間に加えたり、(負の) < d 3 > 値をフレーム f 3 に対する受信機のローカル時間に加えたりして算出した活性化時間は、同一の結果を有するため、受信機はそれらを重複したものと認識し、最初のトリガーのみを適用する。

20

【 0 5 8 0 】

もちろん、図示の最初における 2 つの活性化トリガーは、s t a r t T i m e 以降にチャンネルに参加して最初の要求と併せてフレーム f 3 のシグネチャーを送る受信機にとっては認知できない。

【 0 5 8 1 】

図 3 3 は、場合 (c) に対する活性化トリガーの一実施例を示す図である。

30

【 0 5 8 2 】

図 3 3 は、活性化時間以前に M 時間単位よりも遅く A C R インジェストモジュールに動的活性化トリガーが送られる上記の動作 (4) での状況 (c) を示している。

【 0 5 8 3 】

図 3 3 は、タイムライン上に、動的イベント活性化時間と、A C R インジェストモジュールでイベントの活性化を知るようになるイベント活性化時間の直前の時間を示す図で、A C R インジェストモジュールでイベントの活性化を知る時間後に長さ L 1 の区間が続く。タイムラインの下における垂直矢印は、個別フレームの時間を表す。長さ L 1 の区間の開始よりも先行したり、長さ L 1 の区間の終わりに続く各フレームは、データベースでこのフレームと関連している時間ベーストリガーを有する。長さ L 1 の区間内の各フレームは、データベースで同図の下における例のような活性化トリガーを有する。各フレームに対する “ t = ” 項は、メディア時間ラインに対するイベント活性化時間を示す。(丸囲みの垂直矢印で表示されている。)丸囲みの垂直矢印は、一般の受信機が要求を送る時の例示であってもよい。この場合、受信機はイベント活性化に対して一つの活性化トリガーを有する。活性化トリガーの活性化時間は、このトリガーを受信した時間前であるから、受信機はトリガーを受信すると直ちに適用する。

40

【 0 5 8 4 】

図 3 4 は、場合 (c) に対する活性化トリガーの一実施例を示す図である。

【 0 5 8 5 】

以下、場合 (c) に対する活性化トリガーを説明する。これは、上記の図 3 3 に関する

50

説明に似ている。

【0586】

図34は、活性化時間以前にM時間単位よりも遅くACRインジェストモジュールに動的活性化トリガーが送られる上記の動作(4)での状況(c)を示している。

【0587】

図34は、タイムライン上に、動的イベント活性化時間と、ACRインジェストモジュールでイベントの活性化を知るようになるイベント活性化時間の直前の時間を示す図であり、ACRインジェストモジュールでイベントの活性化を知る時間後に長さMの区間が続く。タイムラインの下における矢印は、個別フレームの時間を表す。長さMの区間の開始よりも先行したり、長さMの区間の終わりに続く各フレームは、データベースで<media_time>又は<event_time>項のないトリガーを有する。長さMの区間内の各フレームは、データベースで同図の下における2つの例のような活性化トリガーを有する。各フレームに対する“d=”項は、このフレームとイベント活性化時間との間の時間長を示すことができる。(丸囲みの垂直矢印で表示されている。)丸囲みの垂直矢印は、一般の受信機が要求を送る時の例示であってもよい。この場合、受信機は、同一のイベント活性化に対して2つの活性化トリガーを受けるが、(負の)<d1>値をフレームf1に対する受信機のローカル時間に加えたり、(負の)<d2>値をフレームf2に対する受信機のローカル時間に加えたりして算出した活性化時間はいずれも同一の結果を有するため、受信機はそれらを重複したものと認識し、受信した最初のトリガーのみを適用する。最初のトリガーの活性化時間は、それを受信した時間以前であるから、受信機はこのトリガーを受信すると直ちに適用する。

【0588】

図35は、最後に伝達される動的活性化トリガーの一実施例を示す図である。

【0589】

イベント主導型ACRモデルで、受信機は、ACRサーバーに対する持続的な接続を開始し、一定の間隔をおいて(例えば、5秒ごとに)フレームと関連したシグネチャーを生成し、シグネチャーを接続を通して提出すると考えることができる。ACRサーバーは各シグネチャーに反応しない。それは、新しいセグメントが発見されたり、又はイベント活性化が受信機に伝達される必要があるとき、受信機にメッセージを送ることができる。当該モデルで、ACRサーバーはいつでも持続的な接続を通してクライアントに対するメッセージを開始することができる。

【0590】

また、受信機からの最も最近の提出に対応するセグメントID(トリガーの<locator_part>)及び受信機に送られる最近活性化トリガーのような各受信機に関する一定量の情報を維持することはサーバーにとって簡単である。

【0591】

当該イベント主導型モデルを使用し、活性化を受信機に伝達するACRサーバーに対して、次の規則はACRサーバーからのメッセージを申し込むことができる。

【0592】

第一に、ACRサーバーは、新しいセグメントでのフレームに対応する受信機からシグネチャーを受信すると、受信機が関連のTPTを取得できるようにするために、直ちに時間ベーストリガーと共にメッセージを受信機に送ることができる。

【0593】

第二に、ACRサーバーは(受信機が見た最も最近のバージョンとは異なる)TPTに対する新しいバージョンナンバーを有するセグメントの一部でのフレームに対応する受信機からシグネチャーを受信すると、受信機が関連のTPTの新しいバージョンを取得できるようにするために、直ちに“v=”項を有する時間ベーストリガーと共にメッセージを受信機に送ることができる。

【0594】

第三に、イベントが活性化される予定である場合、ACRサーバーは活性化トリガーを

受信機に送ることができる。可能ならば、メディア時間ラインと関連した活性化時間を示すために活性化トリガーでの " t = " 項と共に、受信機が適用すべき時期よりもやや早く活性化トリガーを送ることができる（これは、要求/応答モデルでの動作と略同様である）。ACRサーバーは、活性化をあまりにも遅く知って活性化トリガーを平素より遥かに早く送ることはできなくても、活性化を知るとできるだけ早く活性化トリガーを送ることができる。（後者の場合に、受信機は、活性化時期後に活性化トリガーを受けることができる、この場合、受信機は活性化トリガーを受け次第イベントを活性化すると考えることができる。）

【0595】

図28に示した要求/応答ケースに対するアーキテクチャも、一つの差異点以外は、当該イベント主導型ケースに適している。その差異点は、イベント主導型ケースにおいて新しいアクション(2a)がありうるということである。動的活性化トリガーが存在すると、ACRインジェストモジュールが選択された動的活性化トリガーをACRサーバーに送ることができるように、ACRインジェストモジュールとACRインジェストモジュールによって実装されたデータベースを使用する全てのACRサーバーとの間に接続が設定されてもよい。

10

【0596】

イベント主導型ケースに対する番号付きアクションは、要求/応答ケースと類似可能である。新しいアクション(2a)の他に、アクション(4)がすこし異なり、アクション(13)がすこし異なり、新しいアクション(14)が追加される。

20

【0597】

アクション(4)で、ACRインジェストモジュールはフレームからシグネチャーを抽出したり(フィンガープリントACRシステムの場合)、又は関連した双方向サービスを有するセグメントに含まれた全てのフレームに対してコードをフレームに挿入(ウォーターマークACRシステムの場合)することができる。(ACRインジェストモジュールは、GPSクロックと放送スケジュール内セグメントの開始時間及び終了時間を用いて、フレームがそのようなセグメントにあるか否かを判断できる。)そのような各フレームに対してACRインジェストモジュールは、トリガーとシグネチャー又は当該フレームと関連したコードを含み得るデータベースに記録を挿入することができる。次の2つの条件の少なくとも一つを満たさないと、ACRインジェストモジュールによって記録に含まれたトリガーは、時間ベーストリガーであってもよい。

30

【0598】

(a) 活性化エレメントは、フレームの `media_time` が活性化エレメントの `startTime` 前の時間長 `M` から始まって活性化エレメントの `endTime` で終わる時間区間にあるようにAMTに存在する。(活性化が `endTime` を有しないと、`endTime` は `startTime` と同一であると見なす。)(要求/応答ACRモデルに対して条件(a)と同一)

【0599】

(b) トリガーの活性化時間(" t = < event_time > ") 直前の時間長 `M` の時間区間以前に動的活性化トリガーがインジェストモジュールによって受信されており、当該フレームはこの区間内に存在する。(要求/応答ACRモデルに対して条件(b)と同一)

40

【0600】

これらの条件(a)及び(b)のうち一つでも成立すると、活性化トリガーは、活性化されるイベントを識別する " e = " 項とAMT内活性化エレメントの `startTime` を示したり(条件(a)が場合)、動的トリガーの `event_time` を示す(条件(b)) " t = " 項と共に記録に含まれうる。

【0601】

動的活性化トリガーがトリガーの活性化時間直前の時間長 `M` の区間でインジェストモジュールによって受信されると(このとき、`M` は要求/応答サーバーにおけると同じ意味を

50

有する)、インジェストモジュールは、動的活性化トリガーと関連してデータベースに如何なるものも置かず、インジェストモジュールが記録を挿入できるデータベースを使用する全てのACRサーバーに活性化トリガーを伝達することができる。(このアーキテクチャに対して次のような変形があってもよい。インジェストモデルを経由せずに動的活性化トリガーが動的活性化トリガーソースからACRサーバーに直接伝達されるが、関連した受信機に直ちにメッセージを伝送できるように、ACRサーバーは活性化時間以前のM時間単位よりも遅く到着する動的活性化トリガーを得る。次の受信機提出まで待つと遅すぎるはずである。)

【0602】

アクション(13)で、ACRサーバーは直前の提出に対して受信しなかった後にデータベースから"no match"指示子を返してもうと、受信機にNULLメッセージを送ることができる。直前の提出に対して返してもらった<locator_part>と異なる<locator_part>と共にトリガーを返してもらうと、受信機にトリガーを送ることができる。両方の場合において、受信機に視聴中のチャンネルが変更されたり視聴中のセグメントが終わったことを知らせ、受信機が現在実行しているTDOを終了し、必要なら、新しいTPTをダウンロードするようにすることができる。ACRサーバーは、一つ以上の活性化トリガーを返してもらうと、既に受信機に送られた活性化トリガーと重複するものは捨てて受信機に送ることができる。それとも、ACRサーバーは何にもしなくてもよい。

【0603】

新しいアクション(14)で、ACRサーバーは動的活性化トリガーを受信すると、動的活性化トリガーの<locator_part>と各ACRクライアントに対する現<locator_part>とを比較することができる。(ここでクライアントに対する現<locator_part>は、ACRサーバーがACRクライアントからの最も最近の提出に対してデータベースから受けたトリガーの<locator_part>である。<locator_part>がマッチングされる各クライアントに対して、ACRサーバーは動的トリガーをクライアントに送ることができる。)

【0604】

図29及び図31は、活性化時間よりも少なくともM時間単位以前にACRインジェストモジュールに伝達される静的活性化及び動的活性化に対するトリガーの処理を示す。差異点は、ACRサーバーが重複する活性化トリガーを受信機に送らずに捨てることのできるという点である。

【0605】

図35は、急に(活性化時間よりもM時間単位以前に)受信された動的活性化トリガーの処理の一実施例を示す。

【0606】

図35は、時間ライン上の動的イベント活性化時間及びACRインジェストモジュールがイベント活性化について知る時であるイベント活性化時間よりも少し前の時間を示す。時間ラインの下の垂直矢印は、個別フレームの回数を表す。ACRサーバーは、ACRインジェストモジュールから活性化トリガーを受信し次第、(トリガーの<locator_part>から確認されるように)現在活性化トリガーと関連したセグメントを視聴している全ての受信機に活性化トリガーを送ることができる。

【0607】

図36は、最後に伝達される動的活性化トリガーの一実施例を示す図である。

【0608】

最後に伝達される動的活性化トリガーについて説明する。

【0609】

図36は、急に(活性化時間よりもM時間単位以前に)受信された動的活性化トリガーの処理の一実施例を示す。

【0610】

10

20

30

40

50

最後に伝達される動的活性化トリガーを示す同図は、時間ライン上の動的イベント活性化時間及びACRインジェストモジュールがイベント活性化について知る時であるイベント活性化時間よりも少し前の時間を示す。時間ラインの下の垂直矢印は、個別フレームの回数を表す。ACRサーバーは、ACRインジェストモジュールから活性化トリガーを受信し次第、(トリガーの<locator_part>から確認されるように)現在活性化トリガーと関連したセグメントを視聴している全ての受信機に活性化トリガーを送ることができる。各受信機に対して、トリガーのevent_timeを、受信機から最も最近に提出されたシグネチャーに対応するフレームと関連した<delay_time>に調節する。

【0611】

10

図37は、要求/応答ACRケースにおけるACRクライアントと他のサーバー間のシーケンスダイアグラムを示す図である。

【0612】

図37は、要求/応答モデルにおいてACRサーバーと受信機(ACRクライアント)の動作プロトコルによってトリガー及びTPTを効果的に送信する実施例を示すシーケンスダイアグラムである。

【0613】

以下、要求/応答の順で要求/応答モデルの動作の一例を説明する。

【0614】

要求/応答ACRケースにおいて、ACRクライアントと他のサーバー間のシーケンスダイアグラムは、ACR要求1(S37010)、ACR応答1(S37020)、ACR要求2(S37030)、ACR応答2(S37040)、HTTP要求1(S37050)、HTTP応答1(S37060)、HTTP要求2(S37070)、HTTP応答2(S37080)、ACR要求3(S37090)、ACR応答3(S37100)、ACR要求4(S37110)、及び/又はACR応答4(S37120)を含むことができる。

20

【0615】

ACR要求1(S37010)は、受信機が現在視聴中のプログラムのシグネチャーをサーバーに送る段階である。サーバーは、前述したACRサーバーであってもよい。シグネチャーはフィンガープリントシグネチャー又はウォーターマーキングコードであってもよい。

30

【0616】

ACR応答1(S37020)は、シグネチャーが認識されなかったり、又は関連した双方向サービスが存在しないとき、ACRサーバーがヌルをリターンする段階である。これは、ヌル応答がリターンされる上述したケースに該当する場合であってもよい。

【0617】

ACR要求2(S37030)は、チャンネル又はプログラムが変更されるとき、受信機が変更されたプログラムのシグネチャーをACRサーバーに伝送する段階である。

【0618】

ACR応答2(S37040)は、ACRサーバーが該当のプログラムに関連した双方向サービスを取得できるアドレスを含むトリガー(例えば、xbc.com/tpt504)をリターンする段階である。この段階は、ACR応答1(S37020)と違い、シグネチャーが認識され、関連した双方向サービスが存在するケースに該当できる。すなわち、トリガーを利用可能である場合であってもよい。この場合、リターンされるトリガーは、media_timeに関する情報がない時間ベーストリガーであってもよい。

40

【0619】

HTTP要求1(S37050)は、受信機がTPTサーバー(例えば、http://xbc.com/tpt504)にhttpプロトコルを通してACR応答2(S37040)を受信したアドレスを用いて該当するTPTを提供することを要求する段階である。

50

【0620】

HTTP 応答 1 (S37060) は、TPT サーバーが受信機の要求に応じて XML に表した TPT を伝送する段階である。

【0621】

HTTP 要求 2 (S37070) は、受信機がコンテンツサーバーに http プロトコルを用いて TDO のようなアプリケーションを提供することを要求する段階である。受信機は、TPT 内に含まれた TDO 関連情報をパースングすることができる。TDO 関連情報は、TDO をダウンロードできるコンテンツサーバーのアドレスであってもよい。コンテンツサーバーのアドレスを用いて要求を送ることができる。

【0622】

HTTP 応答 2 (S37080) は、コンテンツサーバーが受信機の要求に応じて該当する TDO を伝送する段階である。

【0623】

ACR 要求 3 (S37090) は、受信機が現在視聴中のプログラムのフレームのシグネチャーを ACR サーバーに送る段階である。

【0624】

ACR 応答 3 (S37100) は、ACR サーバーが該当のプログラムに関連した双方向サービスを取得できるアドレスを含むトリガー（例えば、xbc.com/tp504）をリターンする段階である。この場合、ACR 応答 1 (S37020) と違い、シグネチャーが認識され、関連した双方向サービスが存在する。すなわち、この段階でトリガーは使用可能である。

【0625】

ACR 要求 4 (S37110) は、受信機が現在視聴中のプログラムのフレームのシグネチャーを ACR サーバーに伝送する段階である。

【0626】

ACR 応答 4 (S37120) は、ACR サーバーが、受信機から伝送されたシグネチャーと関連した双方向サービスと関連した活性化トリガーを伝送する段階である。特定イベントが活性化トリガーによって特定時間に活性化されうる。

【0627】

図 38 は、イベント主導型 ACR ケースにおける ACR クライアントと他のサーバー間のシーケンスダイアグラムを示す図である。

【0628】

図 38 は、イベント主導型モデルにおいて ACR サーバーと受信機 (ACR クライアント) の動作プロトコルによってトリガー及び TPT を効果的に伝送する実施例に係るシーケンスダイアグラムを示す。

【0629】

要求、要求に対する応答、イベントの順にイベント主導型モデルの動作についての一例を説明する。

【0630】

イベント主導型 ACR ケースにおいて、ACR クライアントと他のサーバー間のシーケンスダイアグラムは、ACR 要求 1 (S38010)、ACR 要求 2 (S38020)、ACR イベント 1 (S38030)、HTTP 要求 1 (S38040)、HTTP 応答 1 (S38050)、HTTP 要求 2 (S38060)、HTTP 応答 2 (S38070)、ACR 要求 3 (S38080)、ACR イベント 2 (S38090)、及び / 又は ACR 応答 4 (S38100) を含むことができる。

【0631】

ACR 要求 1 (S38010) は、受信機が現在視聴中のプログラムのシグネチャーをサーバーに送る段階である。サーバーは、上述した ACR サーバーであってもよい。シグネチャーは、フィンガープリントシグネチャー又はウォーターマークであってもよい。ここで、シグネチャーが認識されなかったり、又は関連した双方向サービスが存在しないと

10

20

30

40

50

き、図37のシーケンスとは違い、ACRサーバーはヌルをリターンせずにACR要求1に何ら応答も送らなくてもよい。

【0632】

ACR要求2(S38020)は、チャンネル又はプログラムが変更されるとき、受信機が、変更されたプログラムのシグネチャーをACRサーバーに伝送する段階である。

【0633】

ACRイベント1(S38030)は、ACRサーバーが、該当のプログラムに関連した双方向サービスを取得できるアドレスを含むトリガー(例えば、xbc.com/tp504)をリターンする段階である。この段階は、シグネチャーが認識され、関連した双方向サービスが存在するケースに該当できる。すなわち、この段階でトリガーは使用可能である。この場合、リターンするトリガーは、media_timeと関連した如何なる情報も有しない時間ベーストリガーであってもよい。

10

【0634】

HTTP要求1(S38040)は、受信機がTPTサーバー(例えば、http://xbc.com/tp504)にhttpプロトコルを通してACRイベント1(S38030)で受信したアドレスを用いて該当するTPTを提供することを要求する段階である。

【0635】

HTTP応答1(S38050)は、TPTサーバーが受信機の要求に応じてXMLに表したTPTを伝送する段階である。

20

【0636】

HTTP要求2(S38060)は、受信機がコンテンツサーバーにhttpプロトコルを用いてTDOのようなアプリケーションを提供することを要求する段階である。受信機は、TPT内に含まれたTDO関連情報をパーシングすることができる。TDO関連情報は、TDOをダウンロードできるコンテンツサーバーのURLであってもよい。コンテンツサーバーのURLを用いて要求を送ることができる。

【0637】

HTTP応答2(S38070)は、コンテンツサーバーが受信機の要求に応じて該当するTDOを伝送する段階である。

【0638】

ACR要求3(S38080)は、受信機が現在視聴中のプログラムのフレームのシグネチャーをACRサーバーに送る段階である。

30

【0639】

ACRイベント2(S38090)は、ACRサーバーが受信機から伝送されたシグネチャーと関連した双方向サービスと関連した活性化トリガーを伝送する段階である。特定イベントが活性化トリガーによって特定時間に活性化される。

【0640】

ACR応答4(S38100)は、受信機が現在視聴中のプログラムのフレームのシグネチャーをACRサーバーに送る段階である。

【0641】

図39は、UPnP Remote UIクライアントサービスのアクションリストの一実施例である。

40

【0642】

第2スクリーン支援は、受信機で、放送社のサービス或いは放送社で提供し、プログラムに適した付加的なサービスを、第2スクリーンデバイスに提供するための方法に関するものである。付加的なコンテンツは、アプリケーションを用いて提供することができ、アプリケーションは、TV画面に表示されてユーザがリモコン操作で付加コンテンツを使用できるようにすることができる。しかし、最近では、個人化情報化機器(スマートホン、スマートパッド及び小型化しつつあるラップトップ)などによってTV画面で再生されているコンテンツはそのまま表示させながら、付加的なサービスを第2スクリーンデバイスに

50

表示させることができる。これによって、既存の放送コンテンツを妨害なく個人的に利用できるようにすることができる。このように付加的なデータや情報を第2スクリーンデバイスに表示して処理をすると、TV画面を複数の人が一緒に見る状況で、付加サービスに関心のある人だけが他の人のTV視聴を妨害することなく付加コンテンツを得ることができる効果がある。第2スクリーン支援は、前述した効果の他にも様々な用途に活用することができる。

【0643】

一つの装置を他の装置に接続させて制御するためには、まず、ホームネットワークにどのような装置があるかを知らなければならない。このため、一つ以上の装置が自身の存在を周期的にホームネットワークにメッセージを送信して広く知らせることができる。一方では、一つの装置が新しくホームネットワークに接続しながら自身を紹介する前に、自身以外にどのような装置があるかを要求することができる。これら2つの接近方式は、互いに装置を容易に且つ迅速に認識できるように手伝えることができる。これを、UPnPデバイスディスカバリ(Device Discovery)と呼ぶ。装置を発見すると、発見された装置に関心のある他の装置は、その装置にどのようなサービスを提供できるかに関する情報が必要でありうる。UPnPフレームワーク(Framework)が内蔵されている家電機器は、互いにどんな機能を有しているか、そしてそれらの機能がどの範囲まで支援するかが把握できる。この情報をUPnPデバイスプロファイル(Device Profile)と定義しておき、互いに提供できるサービスの属性を把握することができる。特定サービスを提供する装置を常に待っていてもよいが、その特定サービスを提供する装置が発見される場合、発見された装置内にどのようなサービスがあるかを問うことができる。発見された装置内に適合なサービスがあると、すぐに接続して互いに通信することができる。また、UPnPサービス記述語(Service Descriptor)に定義されたとおりにサービスを定義して交換することができる。

【0644】

一つの装置には、単数又は複数個のサービスがあり、このサービスを互いに異なる機器で制御して使用するようになっている。装置の一つ以上の機能があると、その機能別に定義された複数個のサービスが含まれ、他の機器で制御することができる。装置の定義は、装置固有の情報を有することができる。例えば、モデル名(Model Name)、一連番号(Serial Number)、モデル番号(Model Number)、CEメーカー名(Manufacturer Name)及びウェブサイト(Web Site)などの情報が装置固有の情報であってもよい。これらの情報は装置別に固有の値を有し、一般的には変更されない。サービスは、装置が行える動作であり、各動作はアクション(Action)と呼び、一つ以上のアクションを有することができる。アクションは、関数のようにそれぞれパラメータ値を有し、一つ以上のパラメータ値を有することができる。アクションは、装置内のサービスで行われ、アクションが行われた後には、必要によって、定義されたリターン値を返還することができる。

【0645】

アクションに対する例示として、図39は、例示されたサービスであるUPnP RemoteUIクライアントサービスに対して、そのサービスのアクションの種類及び記述(Description)を示している。Connection/disconnectionは、現在接続状態を返還するアクションであってもよい。GetCurrentConnectionは、現在接続目録を返還するアクションであってもよい。GetDeviceProfileは、デバイスプロファイルをXMLの形態として返還するアクションであってもよい。GetUIListingは、互換可能な(compatible)UIの目録をXMLの形態として返還するアクションであってもよい。AddUIListing/RemoveUIListingは、remoteUIのURLをUI目録に追加したり除去する動作であってもよい。ProcessInputは、RemoteUIクライアントサービスにデータを送るアクションであってもよい。DisplayMessageは、RemoteUIクライアントサービスを含む装置にメッセージを

10

20

30

40

50

ディスプレイするアクションであってもよい。

【0646】

図40は、UPnP RemoteUIクライアントサービスの一実施例である。

【0647】

UPnPでは、前述したアクションと必要なパラメータ値を送信するために、SOAPというXMLメッセージ形式を使用し、SOAPは、遠隔手順(Remote Procedure)を遠隔で行うために用いることができる。これらのメッセージは、HTTPプロトコルを用いて伝送することができる。

【0648】

前述したRemoteUIクライアントのアクションは、図40のように動作することができる。図40に示す動作は、前述したアクションの一実施例にすぎない。

10

【0649】

UPnP RemoteUIクライアントサービスの一実施例は、UPnP探索プロセスとRemoteUIクライアントアクションとに分けることができる。

【0650】

UPnP探索プロセスは、第2スクリーンサービスのためのUPnPアプリケーションを実行する段階(s40001)、UPnPデバイスを探す段階(s40010)、RemoteUIClientを探す段階(s40020)、デバイス記述(Device Description)を要求する段階(s40030)、デバイス記述を受信する段階(s40040)、サービス制御記述を要求する段階(s40050)及び/又はサービス制御記述を受信する段階(s40060)を含むことができる。

20

【0651】

RemoteUIクライアントアクションは、デバイスプロファイルを要求する段階(s40070)、デバイスプロファイルを受信する段階(s40080)、遠隔UIのURLを送る(put)段階(s40090)、応答1を送信する段階(s40100)、RemoteUIクライアントサービスにメッセージを送信する段階(s40110)、応答2を送信する段階(s40120)、及び/又はユーザがスクリーン上のボタンをクリックする段階(s40130)を含むことができる。

【0652】

UPnPアプリケーションを実行する段階(s40001)は、第2スクリーンデバイス(RemoteUIクライアント)でan UPnPアプリケーションを実行させる段階であってもよい。

30

【0653】

UPnPデバイスを探す段階(s40010)は、プライマリデバイスが第2スクリーンデバイスで実行中のアプリケーションにディスカバリメッセージ(discovery message)をマルチキャストする段階であってもよい。これによってネットワーク内の第2スクリーンデバイスを探すことができる。プライマリデバイスは、ディスカバリメッセージを用いて、自身が提供する第2スクリーン支援サービスを知らせることができる。

【0654】

40

RemoteUIClientを探す段階(s40020)は、第2スクリーンデバイスがディスカバリメッセージを受け、プライマリデバイスに通知メッセージ(notification message)を送る段階であってもよい。

【0655】

デバイス記述を要求する段階(s40030)は、プライマリデバイスが第2スクリーンデバイスに第2スクリーンデバイスに対するデバイス記述を要求する段階であってもよい。

【0656】

デバイス記述を受信する段階(s40040)は、第2スクリーンデバイスがデバイス記述を要求する段階(s40030)に対する応答として第2スクリーンデバイスのデバ

50

イスプロファイルを送ると、プライマリデバイスがこれを受信する段階であってもよい。

【0657】

サービス制御記述を要求する段階 (s 4 0 0 5 0) は、プライマリデバイスが第2スクリーンデバイスにサービス制御記述を要求する段階であってもよい。

【0658】

サービス制御記述を受信する段階 (s 4 0 0 6 0) は、プライマリデバイスが第2スクリーンデバイスから、要求したサービス制御記述を受信する段階であってもよい。

【0659】

UPnP探索プロセスによって、ネットワーク上に存在するプライマリデバイスと第2スクリーンデバイスはお互い探索して認識することができる。また、これによってプライマリデバイスと第2スクリーンデバイスは互いに接続することができる。

10

【0660】

デバイスプロファイルを要求する段階 (s 4 0 0 7 0) は、第2スクリーンデバイスのデバイスプロファイルを要求する段階であってもよい。前述したGetDeviceProfileアクションを用いることができる。

【0661】

デバイスプロファイルを受信する段階 (s 4 0 0 8 0) は、プライマリデバイスが要求した第2スクリーンデバイスのデバイスプロファイルを受信する段階であってもよい。“RemoteUIクライアントサービス”を有する装置(第2スクリーンデバイス、RemoteUIクライアント)は、遠隔の装置(プライマリデバイス)でUIのURLアドレスを知らせると、このアドレスを探して画面に表示する役割を果たすことができる。また“RemoteUIServerサービス”を有する装置は、UIのURLアドレスをマルチキャストし、それに関心のある装置に知らせるサービスであってもよい。しかし、この場合には、RemoteUIが特定装置のために作られるため、特定装置に合う遠隔UIを提供しなければならない。このため、デバイスプロファイル情報を共に送信する必要があり、デバイスプロファイルを要求する段階 (s 4 0 0 7 0) 及びデバイスプロファイルを受信する段階 (s 4 0 0 8 0) が必要でありうる。

20

【0662】

遠隔UIのURLを送る段階 (s 4 0 0 9 0) は、遠隔UIのURLアドレスを第2スクリーンデバイスに知らせる段階であってもよい。前述したAddUIListingアクションを用いることができる。第2スクリーンデバイスは、これをUIListingに追加することができる。

30

【0663】

応答1を送信する段階 (s 4 0 1 0 0) は、遠隔UIのURLを送る段階 (s 4 0 0 9 0) に対する結果を送る段階であってもよい。状況によって、HTTP/1.1 200 OK (要求が成功裏に処理される)、HTTP/1.1 501 Method Not Implemented (要求を処理するために必要な機能が具現されていない)、HTTP/1.1 400 Not Found (要求したファイル/リソースが存在しない)などの応答を送ることができる。

【0664】

40

RemoteUIクライアントサービスにメッセージを送信する段階 (s 4 0 1 1 0) は、プライマリデバイスが第2スクリーンデバイスにディスプレイメッセージを送信する段階であってもよい。ディスプレイメッセージは、メッセージタイプなどの情報を含むことができる。前述したDisplayMessageアクションを用いることができる。第2スクリーンデバイスは、ディスプレイメッセージによって、メッセージを第2スクリーンにディスプレイすることができる。

【0665】

応答2を送信する段階 (s 4 0 1 2 0) は、RemoteUIクライアントサービスにメッセージを送信する段階 (s 4 0 1 1 0) に対する結果を送る段階であってもよい。前述した応答1を送信する段階 (s 4 0 1 0 0) と同様に、状況に応じて、HTTP/1.

50

1 2 0 0 O Kなどの応答を送ることができる。

【0666】

ユーザがスクリーン上のボタンをクリックする段階 (s 4 0 1 3 0) は、第2スクリーンにディスプレイされたメッセージなどによって、ユーザが選択をする段階であってもよい。

【0667】

R e m o t e U I クライアントアクションは、前述したアクションなどによって R e m o t e U I クライアントサービスが動作する過程を記述したものであってもよい。

【0668】

遠隔ユーザインターフェースに関する記述は、画面に表示される H T M L 記述ベースのアプリケーションを家電機器で使用できるように機能を追加したり制限された家電機器のリソースを勘案して、既存の P C システムで使用可能な機能を簡素化した。この記述には、大きく、2つの観点が含まれている。すなわち、画面に表示する H T M L を家電製品で使用できるようにしたことと、各家電製品で使用できるブラウザ A P I を定義することである。ブラウザ A P I は、多く使われているスクリプト言語であるジャバスクリプト (J a v a S c r i p t) で呼び出しして使用することができる。ジャバスクリプトで呼び出された A P I は、結局として受信機の関数を呼び出すようになる。

【0669】

この中、前述した U P n P デバイスとサービスは、ブラウザで動作するジャバスクリプトによっても動作できるはずである。そのためには、ブラウザで各 U P n P デバイス別に異なるパラメータを伝達する新しい A P I が必要であろう。

【0670】

図41は、D T V C C S e r v i c e N u m b e r 6 におけるトリガーの一実施例である。

【0671】

前述したとおり、D T V C C (D i g i t a l T V C l o s e d C a p t i o n S e r v i c e) サービスを用いて前述のリガーを送信することができる。トリガーは、一つの U R L スtring (S t r i n g) 値を有することができる。D T V C C サービスのうちの6番目のサービスとして受信することができる。M P E G - 2 送信プロセッサ 4 1 0 1 0 はトリガーを D T V C C サービスシリーズ # 6 として受信し、送信プロセッサドライバ 4 1 0 2 0 を通して D T V - C C モジュール 4 1 0 4 0 に伝達できる。このとき、U s e r d a t a は、バッファ 4 1 0 3 0 を経由することができる。D T V - C C モジュール 4 1 0 4 0 は、D T V C C デコーダの役割を果たすことができる。D T V - C C モジュール 4 1 0 4 0 は、U R I スtring 値を有するトリガーを付加サービスモジュール (A d j u n c t S e r v i c e m o d u l e) 4 1 0 5 0 に伝達することができる。付加サービスモジュール 4 1 0 5 0 はトリガーモジュールであり、U R I スtring 値を把握し、前述した T P T (T D O P a r a m e t e r s T a b l e) を取得することができる。これによって、前述したとおり、トリガー、T P T 及び T D O を用いて付加サービスを提供することができる。

【0672】

図42は、第2スクリーンシナリオのためのシステムアーキテクチャの一実施例である。

【0673】

第2スクリーンの支援のためにいくつかの要求事項がありうる。1) 受信機は、一つ或いはそれ以上の第2スクリーンデバイスと接続することができる。2) 第2スクリーンデバイスは、ラップトップ、タブレット、スマートホン、P D A などの携帯機器を対象とすることができる。3) 第2スクリーンの主要コンテンツは、H T M L 、テキスト、ビデオ、オーディオなどとすることができる。4) 第2スクリーンで再生されるコンテンツは、プライマリデバイス (受信機) の放送プログラムに邪魔とならない設計とすることができる。5) 第2スクリーンは直接的に A T S C 2 . 0 受信機に接続することができ、間接的

10

20

30

40

50

には3GPP網のように迂回して接続することもできる。6)プロバイダが、特定コンテンツが第2スクリーンでのみ表示されるようにシグナリングすることもできる。7)場合によっては、受信機で再生可能なコンテンツも、第2スクリーンで再生可能に設計することができる。8)認証(Authentication)及び許可(Authorization)された第2スクリーンで第2スクリーンサービスを用いることができる。9)第2スクリーンコンテンツの聴衆使用量(audience usage)を測定することができる。(この場合、このような情報を得るためにユーザから同意を得、その情報を報告できる機能が存在してもよい。)10)第2言語又はコンテンツエンハンスメント(Content Enhancement)可能な装置を用いて提供することもできる。

10

【0674】

本明細書は、放送されるプログラムと同期したコンテンツを有する第2スクリーンデバイス(スマートホン、タブレット、ラップトップなど)上で実行されるアプリケーションによるA/V放送に関連したコンテンツのディスプレイを支援するために受信機によって提供されるサービスを記述することができる。サービスに関する説明はUPnPデバイスアーキテクチャに基づいて記述された。

【0675】

この明細書で、ATSC2.0受信機とは、TV受信機又は一般的な受信機を意味することができる。また、DVB、ATSCなどにおける受信機を意味することができる。

【0676】

UPnPデバイスアーキテクチャは、サービスを提供する“被制御デバイス”及びそのサービスを利用する“制御ポイント”間のIPネットワークにおける通信のためのプロトコルを定義する。“第2スクリーン”シナリオで、TV受信機は“被制御デバイス”の役割を担い、第2スクリーンデバイスは“制御ポイント”の役割を担うことができ、その逆であってもよい。

20

【0677】

UPnPデバイスアーキテクチャは、関心のある被制御デバイスを探す制御ポイントのための“探索”プロトコル、被制御デバイス及びサービスに対する細部事項を学習する制御ポイントのための“記述(description)”プロトコル、被制御デバイス上の“アクション”(方法)を実行(involve)する制御ポイントのための“制御”プロトコル、及び制御ポイントに非同期通知を伝達する被制御デバイスのための“イベントイング(eventing)”プロトコルを特定する。アクション及びイベントは、デバイスサービスによって提供することができる。

30

【0678】

UPnP被制御デバイスがネットワークにジョイン(join)すると、“よく知られた”IPマルチキャストアドレス及びポートにディスカバリメッセージをマルチキャストすることができる。それらのメッセージは、デバイスによって提供されるデバイスタイプ及びサービスタイプを識別することができ、デバイス及びサービスの記述を取得できるURLを提供することができる。

【0679】

UPnP制御ポイントがネットワークにジョインすると、被制御デバイスに自身を知らせるかを問う序歯(search)メッセージをマルチキャストすることができる。サーチメッセージは、関心のあるデバイスタイプ及び/又はサービスタイプを特定することができる。関連デバイスは、制御ポイントにユニキャストディスカバリメッセージを送ることによって応答することができる。

40

【0680】

制御ポイントが関心のあるデバイス及びサービスに対するディスカバリメッセージを得ると、メッセージ内のURLを用いてデバイス及びサービスの記述(description)を要求することができる。それらの記述は、アクションを実行(involve)し、サービスに対するイベントを申し込むために用い得るURLを含むことができる。

50

【0681】

一般的な第2スクリーン探索シナリオは、シナリオAとシナリオBとに大別できる。シナリオAの場合、ユーザは、TV受信機が(TV受信機がオンになる時又はそのネットワークインターフェースがイネーブルされる時に発生しうる)ネットワークにジョインする時、自身の第2スクリーンデバイスで実行される第2スクリーンアプリケーションを有する。シナリオBの場合、ユーザは、TV受信機がネットワークにジョインする時、自身の第2スクリーンデバイスで実行される第2スクリーンアプリケーションを有しない。

【0682】

シナリオAは、次の通りでありうる。1)第2スクリーン支援を提供するTV受信機はネットワークにジョインする。2)TV受信機は、その第2スクリーン支援サービスを広告するディスカバリメッセージをマルチキャストする。3)第2スクリーンデバイスで実行される第2スクリーンアプリケーションは、マルチキャストされたディスカバリメッセージを受信し、TV受信機にそのサービスの記述に対する要求を送信する。4)TV受信機は、そのサービスの記述で応答する。5)第2スクリーンアプリケーションは、記述中に含まれた情報を用いて適切なサービスにアクセスして、TVプログラミングと同期したインタラクティブ経験を提供する。

【0683】

シナリオBは、次の通りでありうる。1)TV受信機上で見ているTV番組は、第2スクリーン支援を提供するプログラムセグメントに入る。(これは、TV受像機がオンになるやいなや又はチャンネルが第2スクリーン支援を有するインタラクティブデータサービスを提供しないチャンネルからインタラクティブデータサービスを提供するチャンネルに変更される時、又は視聴しているチャンネルが第2スクリーン支援を有するインタラクティブデータサービスを提供しないプログラムセグメントからインタラクティブデータサービスを提供するプログラムセグメントに変更される時でありうる。)2)これは、TV受信機が第2スクリーン支援が利用可能な任意の方式で、例えば、スクリーンのコーナーにある小さいアイコンによって、ユーザに知らせるようにする。3)視聴者は、第2スクリーン支援を用いて自身の第2スクリーンデバイス上で第2スクリーンアプリケーションを活性化する。すると、第2スクリーンアプリケーションは、第2スクリーン支援を提供する装置を探すメッセージをマルチキャストする。TV受信機は、自身のディスカバリメッセージでこのメッセージに応答する。4)第2スクリーンアプリケーションがディスカバリメッセージを探すと、TV受信機にそのサービスの記述に対する要求を送信する。5)TV受信機は、そのサービスの記述で応答する。6)第2スクリーンアプリケーションは、記述中に含まれた情報を用いて適切なサービスにアクセスし、TV番組と同期したインタラクティブ経験を提供する。

【0684】

シナリオAを書き換えると、次の通りでありうる。1)第2スクリーン支援を提供するTV受像機はネットワークにジョインする。(これは、TVセットがオンになる時又はそのネットワークインターフェースがイネーブルされる時であってもよい。)2)これは、TV受信機が自身及びその第2スクリーン支援サービスを広告する自身のディスカバリメッセージをマルチキャストするようにする。3)この時、ユーザが自身の第2スクリーンデバイスで実行される第2スクリーンアプリケーションを有すると、アプリケーションは、マルチキャストされたディスカバリメッセージを受信し、段階(7)に進行する。4)TV受信機上で見ているTV番組が第2スクリーン支援を提供するプログラムセグメントに入る。(これは、TV受像機がオンになるやいなや又はチャンネルが第2スクリーン支援を有するインタラクティブデータサービスを提供しないチャンネルからインタラクティブデータサービスを提供するチャンネルに変更される時又は視聴しているチャンネルが第2スクリーン支援を有するインタラクティブデータサービスを提供しないプログラムセグメントからインタラクティブデータサービスを提供するプログラムセグメントに変更される時であってもよい。)5)これは、TV受信機が、第2スクリーン支援が利用可能な任意の方式で、例えば、スクリーンのコーナーにある小さいアイコンによって、ユーザに知らせるよ

10

20

30

40

50

うにする。6) 視聴者が自身の第2スクリーンデバイスで実行される第2スクリーンアプリケーションを有していない場合、視聴者は、第2スクリーン支援を用いて自身の第2スクリーンデバイス上で第2スクリーンアプリケーションを活性化する。すると、第2スクリーンアプリケーションは、第2スクリーン支援を提供する装置を探すメッセージをマルチキャストする。TV受信機は、自身のディスカバリメッセージでこのメッセージに応答する。7) 第2スクリーンアプリケーションがディスカバリメッセージを探すと、TV受信機にそのサービスの記述に対する要求を送信する。8) TV受信機は、そのサービスの記述で応答する。これらは、トリガーサービス及び選択的にHTTPプロキシサーバーサービスであろう。9) 第2スクリーンアプリケーションは、トリガーサービスの“イベントリング”特徴を申し込むはずである。提供されるインタラクティブデータサービスの相互作用モデルが直接実行モデルであれば、全てのトリガーがTV受信機によって受信される時、トリガーサービスは全てのトリガーを第2スクリーンデバイスに送信するはずである。インタラクティブデータサービスの相互作用モデルがTDOモデルであれば、新しい活性化トリガーの活性化時間に到達する度に、トリガーサービスは活性化トリガーを第2スクリーンアプリケーションに送信するはずである。10) 第2スクリーンアプリケーションは、必要によってTDO及び他のファイルを、インターネットリンク又はTV受信機によって提供されたHTTPプロキシサーバーサービスからダウンロードし、インタラクティブサービスを第2スクリーン装置ユーザに提供しながらトリガーに対して動作するはずである。11) TV受信機上のTV番組が、第2スクリーン支援を有するインタラクティブデータサービスを提供しないセグメントに進行すると、トリガーサービスは、“ヌルトリガー (null trigger)” を第2スクリーンアプリケーションに送信し、第2スクリーンデバイスに対するインタラクティブデータサービスがそれ以上利用可能でないということを知らせる。12) その後、第2スクリーンデバイスのユーザが、第2スクリーンアプリケーションを閉じたり、そのまま第2スクリーンアプリケーションが実行されるようにし、TV受信機上の番組が第2スクリーン支援を提供する他のセグメントに入る度に相互作用を再開するようにすることができる。

【0685】

これらの両シナリオで、仮定は、ホームネットワーク上に1よりも多いTV受信機を有することができる。この場合、第2スクリーンアプリケーションは、多数の異なった受信機からディスカバリメッセージを受信することができる。この場合、第2スクリーンアプリケーションは(ユーザの決定を助けるために記述メッセージからの情報をディスプレイしながら) どれをもって相互作用をするかをユーザに問うことができる。

【0686】

第2スクリーン支援を提供するTV受信機は、第2スクリーンアプリケーションの使用のためのいくつかのUPnPサービスを提供することができる。UPnPサービスは、受信機から第2スクリーンアプリケーションへの“トリガー伝達サービス”、受信機で実行されるDO (Declarative Objects) 及び第2スクリーンアプリケーション間の“双方向通信サービス”及び“HTTPプロキシサーバーサービス”であってもよい。それらのサービスは、構成によって選択的であってもよい。

【0687】

それらのサービスは、様々な異なったソースから得られ、様々な異なったタイプの第2スクリーンデバイスで様々な互いに異なる動作環境下に実行される様々な異なったタイプの第2スクリーンアプリケーションを支援するように設計されてもよい。

【0688】

一般的な第2スクリーンパッケージアプリケーションシナリオは、次の通りである。1) 第2スクリーンデバイス上の制御ポイントは、第1スクリーン装置上のパッケージアプリケーションサービスを申し込む。2) 消費者は、第1スクリーン装置上のパッケージアプリケーションを始める。3) パッケージアプリケーションは、第2スクリーンアプリケーションのネーム及び第2スクリーンアプリケーションのURLをパッケージアプリケーションサービスに利用可能にする。4) 第2スクリーンデバイス上の制御ポイントは、同

10

20

30

40

50

伴者 (companion) アプリケーションのネーム及び URL を受信する。5) 制御ポイントは、必要な消費者アクションを指示する第2スクリーン上のマーカーを設定する。6) 消費者は、第2スクリーンアプリケーションのネームを検討し選択する。7) 制御ポイントは、指示された第2スクリーンアプリケーションを開始する。

【0689】

第2スクリーン支援を提供する第1スクリーンは、第2スクリーンアプリケーションの利用のためにいくつかのUPnPサービスを提供することができる。UPnPサービスの一つは、第2スクリーンデバイス上で実行される同伴者第2スクリーンアプリケーションのネーム及びURLを提供している“アプリケーションURLサービス”である。

【0690】

第2スクリーンシナリオに対するシステムアーキテクチャを説明する。

【0691】

第2スクリーンシナリオに対するシステムアーキテクチャは、放送システム42010、ACRサーバー42020、放送社ATSC2.0iTVサーバー42030、ATSC2.0受信機42040及び/又は第2スクリーンデバイス42050を含むことができる。

【0692】

放送システム42010は、一般的な放送システムであってもよい。放送ネットワークを介してトリガー、A/V、TPT及び/又はその他データファイルなどを伝達することができる。トリガーの場合、前述したとおり、DTVCCを用いて伝達することができる。

【0693】

ACRサーバー42020は、放送コンテンツに対するACR関連データを管理するサーバーであり、要求或いは必要な場合、ATSC2.0受信機42040がインタラクティブサービスを取得できるように、これと関連したトリガーをATSC2.0受信機42040に送信することができる。前述したACRサーバーと同一であってもよい。

【0694】

放送社ATSC2.0iTVサーバー42030は、インタラクティブサービスを生成及び管理するサーバーであり、これと関連したTDO/ファイル进行管理し、それに関する情報を含むTPT (T D O Parameter Table) を生成及び管理することができる。

【0695】

ATSC2.0受信機42040は、放送A/V及びインタラクティブサービスと関連したトリガーなどを受信し、それを用いてインタラクティブサービスを取得及び画面上に提供することができる。前述した受信機と同一であってもよい。

【0696】

第2スクリーンデバイス42050は、第2スクリーンデバイスであるモバイルホン、タブレット、ラップトップなどを含むことができる。前述した第2スクリーンデバイスと同一であってもよい。

【0697】

トリガーは、DTVCCチャンネル又はACRプロセスを通してATSC2.0受信機42040に伝達されてもよく、放送インタラクティブTV (iTV) サーバー42030から伝達されてもよい。いずれの場合も、適切な時間にトリガーはATSC2.0受信機42040から第2スクリーンデバイス42050に伝達される。本明細書は、第2スクリーンデバイス42050に伝達されるトリガーに対するメカニズムを説明する。また、ATSC2.0受信機42040上で実行されて第2スクリーンデバイス42050と双方向通信を確立するDOに対するメカニズムを説明する。

【0698】

インターネットを介して利用可能なアプリケーション及び他のファイルは、そのサービスを提供する場合には、ホームネットワークを通じて、別途の3GPPネットワークを通

10

20

30

40

50

じて、又はA T S C 2 . 0 受信機 4 2 0 4 0 上のH T T P プロキシサーバー（図示せず）を通じて第 2 スクリーンデバイス 4 2 0 5 0 によって検索されうる。第 1 スクリーンデバイス上で実行されるアプリケーションは、放送によって送信されるアプリケーション又はインターネットからダウンロードしたパッケージアプリケーションであってもよい。

【 0 6 9 9 】

放送でF L U T E セッションを通じてのみ利用可能なアプリケーション及び他のファイルは、A T S C 2 . 0 受信機 4 2 0 4 0 が要求時にF L U T E ファイルを伝達するH T T P プロキシサーバーを提供する時にのみ（第 2 スクリーンデバイス 4 2 0 5 0 はT V 受信機を含まないと仮定）、第 2 スクリーンデバイス 4 2 0 5 0 によって検索されうる。

【 0 7 0 0 】

本明細書はまた、第 2 スクリーンデバイス 4 2 0 5 0 上でのアプリケーションの開始を支援するためにA T S C 2 . 0 受信機 4 2 0 4 0 上で実行されるパッケージアプリケーションに対するメカニズムを説明する。

【 0 7 0 1 】

図 4 3 は、A T S C 2 . 0 受信機及び第 2 スクリーン間のトポロジ（直接接続）の一実施例である。

【 0 7 0 2 】

図 4 3 は、A T S C 2 . 0 受信機と第 2 スクリーンデバイス間の直接接続を説明する。

【 0 7 0 3 】

A T S C 2 . 0 受信機及び第 2 スクリーン間のトポロジ（直接接続）の一実施例は、放送システム 4 3 0 1 0、放送社 A T S C 2 . 0 サーバー 4 3 0 2 0、A T S C 2 . 0 受信機 4 3 0 3 0 及び / 又は第 2 スクリーンデバイス 4 3 0 4 0 を含むことができる。

【 0 7 0 4 】

放送システム 4 3 0 1 0 は、放送システム 4 2 0 1 0 と同一であってもよい。

【 0 7 0 5 】

放送社 A T S C 2 . 0 サーバー 4 3 0 2 0 は、放送社 A T S C 2 . 0 i T V サーバー 4 2 0 3 0 と同一であってもよい。

【 0 7 0 6 】

A T S C 2 . 0 受信機 4 3 0 3 0 は、A T S C 2 . 0 受信機 4 2 0 4 0 と同一であってもよい。

【 0 7 0 7 】

第 2 スクリーンデバイス 4 3 0 4 0 は、第 2 スクリーンデバイス 4 2 0 5 0 と同一であってもよい。

【 0 7 0 8 】

A T S C 2 . 0 受信機 4 3 0 3 0 は、放送網に接続して直接に地上波放送受信が可能であってもよい。したがって、A T S C 2 . 0 受信機 4 3 0 3 0 では、D T V C C を通じて送信される i T V メッセージの U R L スtring を D T V C C サービス番号 6 で抽出可能であってもよい。また、ホームゲートウェイに接続していつでも必要によってインターネットに接続可能であってもよい。そして、ホームネットワーク記述（U P n P）を用いて、他のホームネットワーク内に接続した装置と通信することができる。

【 0 7 0 9 】

第 2 スクリーンデバイス 4 3 0 4 0 はいずれもホームゲートウェイと接続してそれぞれの装置間の通信が可能であり、自由にインターネット接近が可能であってもよい。ホームゲートウェイは、イーサネット（E t h e r n e t）及びW i - F i を支援する機能を全て備えていてもよい。

【 0 7 1 0 】

放送社は、インタラクティブサービスのためにインターネットにサーバーをおいて付加サービスを提供することができる。A T S C 2 . 0 受信機 4 3 0 3 0 や第 2 スクリーンデバイス 4 3 0 4 0 が接続して、T D O やモバイル用に作られたウェブページをダウンロードすることができる。A T S C 2 . 0 受信機 4 3 0 3 0 である場合には、T V の解像度に

10

20

30

40

50

合わせて作られたウェブページであり、第2スクリーンデバイス43040である場合には、TVと違い、解像度及び支援するAPIsが異なってもよいため、それに適合するように製作される。

【0711】

図44は、ATSC2.0受信機及び第2スクリーン間のトポロジ(間接接続)の一実施例である。

【0712】

図44は、ATSC2.0受信機及び第2スクリーンデバイス間の間接接続を説明する。

【0713】

ATSC2.0受信機及び第2スクリーン間のトポロジ(間接接続)の一実施例は、放送システム44010、放送社ATSC2.0サーバー44020、放送社セッション管理44030、ATSC2.0受信機44040及び/又は第2スクリーンデバイス44050を含むことができる。

【0714】

放送システム44010は、放送システム42010と同一であってもよい。

【0715】

放送社ATSC2.0サーバー44020は、放送社ATSC2.0iTVサーバー42030と同一であってもよい。

【0716】

放送社セッション管理44030は、間接接続において、第2スクリーンデバイス44050とATSC2.0受信機44040間の間接接続を管理する役割を担うことができる。

【0717】

ATSC2.0受信機44040は、ATSC2.0受信機42040と同一であってもよい。

【0718】

第2スクリーンデバイス44050は、第2スクリーンデバイス42050と同一であってもよい。

【0719】

図44の間接接続は、第2スクリーンデバイス44050がホームゲートウェイを通じてATSC2.0受信機44040に接続される場合ではなく、3GPP網に直接接続してホームネットワークにあるATSC2.0受信機44040と接続される場合であってもよい。この場合には、第2スクリーンデバイス44050がホームネットワークに接続し難い場合或いはホームネットワークを支援するネットワークインターフェースがない場合に該当する。

【0720】

この場合には、第2スクリーンデバイス44050は、外部のインターネットサーバーが有しているATSC2.0受信機44040の情報が必要でありうる。また、ATSC2.0受信機44040は、動作時に、接続アドレスを外部にあるインターネットサーバーに報告して知らせなければならないことがある。

【0721】

第2スクリーンデバイス上にインタラクティブサービスを効果的に伝達するための方案として、中央集中型(centralized)実行モデル及び分散型(distributed)実行モデルがある。

【0722】

図45は、第2スクリーンサービスアプリケーションのソフトウェア層の一実施例を示す図である。

【0723】

第2スクリーンデバイスは、一般的に、OSを用いてアプリケーションを実行する。一

10

20

30

40

50

般に、モバイル装置用OSの一部の機能は、軽量化の目的から含まれなくてもよい。したがって、OSによって支援されない機能がアプリケーションに含まれる必要がある。

【0724】

図45のソフトウェア層は、第2スクリーンサービスに必要な第2スクリーンサービスアプリケーションのソフトウェア構造を示す。図45は、受信機が第2スクリーンアプリケーションのライフサイクルを管理する場合を示すことができる。

【0725】

第2スクリーンデバイスがインターネットコンテンツを再生するので、OSはプラットフォームSDKを用いてウェブアプリケーションが実行され得る環境をプログラマーに提供することができる。環境は、OSによって実行されるアプリケーションがインターネットコンテンツをディスプレイするようにSDKの形態で提供することができる。

10

【0726】

図45で、第2スクリーンサービスアプリケーションは、第2スクリーンデバイス上で実行されうる。このアプリケーションは、アップストア(App Store)(又は、任意の種類アプリケーションマーケット)からダウンロードされうる。第2スクリーンサービスアプリケーションは、ACRクライアントを含むことができ、マイクロホン及びカメラを用いてTV受像機からビデオ及びオーディオデータをキャプチャーすることができる。また、ACRクライアントは、ビデオ及びオーディオデータをサンプリングし、メディアシグネチャーをACRサーバーに送信することができる。第2スクリーンサービスアプリケーションは、OS上で実行可能であり、UPnPフレームワークが何を動作させ得るかによって、HTTP、SSDP又はマルチキャストイベントなどのプロトコルを含み、ブラウザモジュールは、ウェブアプリケーションを実行及びディスプレイするために含まれうる。

20

【0727】

ブラウザモジュールは、インターネットウェブページをレンダリングするモジュールであってもよい。ブラウザモジュールは、ウェブアプリケーション(例えば、ECMAScript、HTML及びXHTML)をレンダリングするウェブブラウザエンジンの核心部分である。また、ブラウザモジュールは、OSによって提供されるブラウザと同一又は類似の機能を提供するソフトウェアモジュールを意味できる。ブラウザ及び制御可能な機能を利用する方法は、OS別に異なる。

30

【0728】

第2スクリーンアプリケーションは、第2スクリーンデバイス用に設計されたウェブアプリケーションであってもよい。第2スクリーンアプリケーションは、ECMAScript又はウェブアプリケーションによって機能呼び出すウェブアプリケーションであってもよく、そのサイズは、モバイルOSによって提供されるブラウザモジュールで実行されるように制御されうる。このウェブアプリケーションは、第2スクリーンサービスアプリケーションで実行されうる。

【0729】

モバイル装置用オペレーティングシステムは、ハードウェアを支援するドライバで構成され、カーネル特定(kernel-specific)サービス用ドライバを含むことができる。基本的に、IP、TCP及びUDPプロトコルを支援することができる。ネットワークプロトコルスタックは、OS上に存在する。UPnPの場合、HTTPは、UDPを用いて送信することができる。

40

【0730】

UPnP DCPフレームワークは、UPnPフォーラムに定義されたデバイス制御ポイントを意味することができる。

【0731】

SSDP(Simple Service Discovery Protocol)は、ホームネットワークでネットワークに接続されたデバイスによって一つ以上のサービスを探すことができる。

50

【0732】

図46は、第2スクリーンサービスアプリケーションのソフトウェア層を示す図である。

【0733】

図46のソフトウェア層は、第2スクリーンサービスに必要な第2スクリーンサービスアプリケーションのソフトウェア構造を示す。図46は、第2スクリーンサービスアプリケーションが第2スクリーンアプリケーションのライフサイクルを管理する場合を示すことができる。

【0734】

図46のそれぞれのエンティティの説明は、その役割及び機能において図45と同一である。

10

【0735】

ATSC 2.0トリガークライアントは、上述したトリガー、TPT、AMTなどを受信し処理するモジュールを意味できる。このモジュールは、状況によって第2スクリーンデバイスに含まれてもよい。このモジュールが第2スクリーンデバイスに含まれると、このモジュールはTPT及びAMTを直接処理し、アプリケーションのライフサイクルを直接チェックすることができる。

【0736】

図47は、第2スクリーンアプリケーションライフサイクル管理による送信順序及び送信されたデータ間の違いを示すテーブルである。

20

【0737】

受信機は、放送ネットワークを通じてDTVCCを用いてTPT及びAMTを直接受信したり、インターネットネットワークを通じてTPT及びAMTをダウンロードし、TPT及びAMTを処理する。上述した通り、TPTはイベント情報を含み、イベント情報は、EventId、Action、Destination及びData情報を含むことができる。“Event@Destination”は、このイベント情報が発生する装置を示すことができる。例えば、あて先は、受信機又は第2スクリーンデバイスであってもよい。あて先が第2スクリーンデバイスに設定されると、受信機は、第2スクリーンデバイスにイベント情報を伝達しなければならない。

【0738】

30

このため、この情報を第2スクリーンデバイスに伝達する方法が必要である。第2スクリーンアプリケーションのライフサイクルは、受信機又は第2スクリーンサービスアプリケーションで管理することができる。

【0739】

図47のテーブルは、どのエンティティが第2スクリーンアプリケーションのライフサイクルを支援するかによる情報伝達方法の概要を示す。

【0740】

第1行は、図51の後述する場合の説明の概要であってもよく、その詳細な説明は後述する。

【0741】

40

第2行は、受信機が第2スクリーンアプリケーションのライフサイクルを管理する場合の説明の概要であってもよく、その詳細な説明は後述する。

【0742】

第3行は、受信機が第2スクリーンデバイスに必要なトリガー情報をフィルタリング(増加)し伝達する場合の説明の概要であってもよく、その詳細な説明は後述する。

【0743】

第4行は、図58の後述する場合の説明の概要であってもよく、その詳細な説明は後述する。

【0744】

第5行は、図57の後述する場合の説明の概要であってもよく、その詳細な説明は後述

50

する。

【0745】

図48は、中央集中型実行モデルの動作概念を示す図である。

【0746】

インタラクティブサービスを第2スクリーンデバイスに効率的に伝達する方法は、中央集中型実行モデル及び分散型実行モデルを含むことができる。

【0747】

中央集中型実行モデルの動作概念を示す図に示すように、TV受像機は、UPnPのRUIメカニズムを用いて、各第2スクリーンデバイス上にディスプレイされるRUIを生成及びアップデートし、ネットワークを通じてRUIを送信することができる。それぞれの第2スクリーンデバイスは、RUIクライアントを通じて実時間でRUIを受信し、スクリーン上にRUIをディスプレイすることができる。分散型実行モデルと違い、DAEは、プライマリデバイスに存在しうる。

10

【0748】

図49は、分散型実行モデルベース受信機及び第2スクリーン間の連動に対するフローの一実施例である。

【0749】

図49のフローは、中央集中型実行モデルの動作概念の実施例で受信機が放送ネットワークを通じてインタラクティブサービスを得ることができ、第2スクリーンデバイスと連動する環境で中央集中型実行モデルメカニズムが適用される時の動作であってもよい。

20

【0750】

中央集中型実行モデルベース受信機及び第2スクリーン間の連動に対するフローの実施例は、時間ベーストリガーを送信する段階(s49010)、TPTを要求する段階(s49020)、応答としてTPTを送信する段階(s49030)、TPTをパーシング(parse)する段階(s49040)、第2スクリーンに対する実行トリガーを送信する段階(s49050)、TDO/ファイルをダウンロードする段階(s49060)、RUIを生成する段階(s49070)、RUIプロトコルを通じてRUIを送信する段階(s49080)、スクリーン上にUIをディスプレイする段階(s49090)、ユーザが新しいページに対するリンクをクリックする段階(s49100)、入力データを送信する段階(s49110)、新しいTDO/ファイルをダウンロードする段階(s49120)、RUIをアップデートする段階(s49130)、RUIプロトコルを通じてRUIを送信する段階(s49140)、及び/又はスクリーン上にUIをディスプレイする段階(s49150)を含むことができる。

30

【0751】

時間ベーストリガーを送信する段階(s49010)は、受信機がDTVCC又はACRサーバーを通じて時間ベーストリガーを受信する段階を含むことができる。時間ベーストリガーは上述した通りである。

【0752】

TPTを要求する段階(s49020)は、受信機が、受信されたトリガーを解釈する段階、TPTを取得できるサーバーのURLを取得する段階、及びTPTを要求できる段階を含むことができる。

40

【0753】

応答としてTPTを送信する段階(s49030)は、TPTを要求する段階(s49020)における要求に応答してTPTを送信する段階を含むことができる。

【0754】

TPTをパーシングする段階(s49040)は、受信機が、要求されたTPTをパーシングする段階を含むことができる。

【0755】

第2スクリーンに対する実行トリガーを送信する段階(s49050)は、受信機がDTVCC又はACRサーバーを通じて第2スクリーンに対する実行トリガーを受信する段

50

階を含むことができる。実行トリガーは、上述した活性化トリガーと同一であってもよい。

【0756】

TDO/ファイルをダウンロードする段階(s49060)は、受信機が、受信したTPTから実行トリガー又は時間ベーストリガーと関連したTDO及びファイルに関する情報を取得する段階、及びコンテンツサーバーからトリガーで指示されたTDO及びファイルをダウンロードする段階を含むことができる。

【0757】

RUIを生成する段階(s49070)は、ダウンロードされたTDO及びファイルに対するRUIを生成する段階を含むことができる。

10

【0758】

RUIプロトコルを通じてRUIを送信する段階(s49080)は、受信機がRUIプロトコルを用いて、第2スクリーン上に生成されたRUIを送信する段階を含むことができる。

【0759】

スクリーン上にUIをディスプレイする段階(s49090)は、第2スクリーン上に、受信されたRUIをディスプレイする段階を含むことができる。

【0760】

ユーザが新しいページに対するリンクをクリックする段階(s49100)は、第2スクリーン上でユーザの選択によってRUI上の特定リンクをクリックする段階を含むことができる。

20

【0761】

入力データを送信する段階(s49110)は、クリックされた特定リンクが他のページに接続されると、受信機にユーザの入力情報を伝達する段階を含むことができる。

【0762】

新しいTDO/ファイルをダウンロードする段階(s49120)は、受信機がユーザ入力と関連したTDO及びファイルをダウンロードする段階を含むことができる。

【0763】

RUIをアップデートする段階(s49130)は、ダウンロードされたTDO及びファイルによってRUIをアップデートする段階を含むことができる。

30

【0764】

RUIプロトコルを通じてRUIを送信する段階(s49140)は、受信機がRUIプロトコルを用いて、アップデートされたRUIを第2スクリーンに送信する段階を含むことができる。

【0765】

スクリーン上にUIをディスプレイする段階(s49150)は、第2スクリーン上にアップデートされたRUIをディスプレイする段階を含むことができる。

【0766】

図50は、受信機で第2スクリーンデバイスにUI情報を知らせる方法の実施例を示す図である。

40

【0767】

図50は、受信機が放送社からトリガーを受信してトリガーを第2スクリーンデバイスに伝達する論理的順序を示す。

【0768】

このプロセスは、受信機に対するトリガーを受信する段階(s50010)、第2スクリーンサービスに対するトリガーを受信する段階(s50020)、アップデートされたUI情報に対する通知を送信する段階(s50030)、アップデートされたUI情報を要求する段階(s50040)、応答として互換可能なUI情報を送信する段階(s50050)、及び受信機に対する他のトリガーを受信する段階(s50060)を含むことができる。

50

【 0 7 6 9 】

受信機に対するトリガーを受信する段階 (s 5 0 0 1 0) は、プライマリデバイスが放送ストリームを通じて放送社から受信機、すなわち、プライマリデバイスに対するトリガーを受信する段階を含むことができる。

【 0 7 7 0 】

第2スクリーンサービスに対するトリガーを受信する段階 (s 5 0 0 2 0) は、プライマリデバイスが放送ストリームを通じて放送社から第2スクリーンサービスに対するトリガーを受信する段階を含むことができる。

【 0 7 7 1 】

アップデートされたUI情報に対する通知を送信する段階 (s 5 0 0 3 0) は、アップデートされたUIに対して通知する段階を含むことができる。上述した通り、放送プログラムを見る間にトリガーが受信されると、受信機は、トリガーが第2スクリーンデバイス又はプライマリデバイスに対するものか否かをチェックすることができる。この時、トリガーが第2スクリーンデバイスに対するものであれば、全てのUPnPデバイス又は特定UPnPデバイスにのみ新しいUI情報アップデートを通知することができる。これは、第2スクリーンデバイスがUPnPプロトコルを用いて加入した場合に該当しうる。

10

【 0 7 7 2 】

アップデートされたUI情報を要求する段階 (s 5 0 0 4 0) は、第2スクリーンデバイスがプライマリデバイスに、アップデートされたUI情報を要求する段階を含むことができる。

20

【 0 7 7 3 】

応答として互換可能なUI情報を送信する段階 (s 5 0 0 5 0) は、プライマリデバイスが第2スクリーンデバイスに互換可能なUI情報を送信する段階を含むことができる。

【 0 7 7 4 】

受信機に対する他のトリガーを受信する段階 (s 5 0 0 6 0) は、受信機に対するトリガーを受信する段階と同一であってもよい。すなわち、上述した動作が再び行われてもよい。

【 0 7 7 5 】

受信機がトリガーマジュールを含んでもよいため、受信機は、放送ネットワーク又はインターネットネットワークを通じてTPT及びAMTなどのXMLファイルを受信し、XMLファイルをパーシングし、適切な動作を行うことができる。第2スクリーンデバイスを探すと、第2スクリーンデバイスに伝達されるアクション、TDOURL及びデータは、Event@Destinationフィールドを用いて認識することができる。第2スクリーンデバイスは、TPT及びAMTなどのXMLファイルを直接受信しなくてもよく、よって、トリガーマジュールを含まなくてもよい。RemoteUIクライアントサービスが含まれると、受信機は、全ての第2スクリーンアプリケーションのライフサイクルを管理することができる。逆に、いくつかの第2スクリーンデバイスが接続されると、第2スクリーンアプリケーションを制御するデータが数回送信されなければならない。

30

【 0 7 7 6 】

図51は、受信機で第2スクリーンデバイスにUI情報を通知する方法の実施例を示す図である。

40

【 0 7 7 7 】

図50と違い、図51は、第2スクリーンデバイスに適切に用いられると決定された第2スクリーンデバイスに対する全てのUI情報が直接伝達される場合を示す。すなわち、第2スクリーンデバイスに対するTDOURLを送信することができる。

【 0 7 7 8 】

このプロセスは、受信機に対するトリガーを受信する段階 (s 5 1 0 1 0)、第2スクリーンサービスに対するトリガーを受信する段階 (s 5 1 0 2 0)、UI情報に対して通知する段階 (s 5 1 0 3 0)、応答 " o k " を送信する段階 (s 5 1 0 4 0)、受信機に対するトリガーを受信する段階 (s 5 1 0 5 0)、受信機に対するトリガーを受信する段

50

階 (s 5 1 0 6 0)、第 2 スクリーンサービスに対するトリガーを受信する段階 (s 5 1 0 7 0)、UI 情報に対して通知する段階 (s 5 1 0 8 0)、及び / 又は応答 “ o k ” を送信する段階 (s 5 1 0 9 0) を含むことができる。

【 0 7 7 9 】

受信機に対するトリガーを受信する段階 (s 5 1 0 1 0) は、受信機に対するトリガーを受信する段階 (s 5 0 0 1 0) と同一であってもよい。

【 0 7 8 0 】

第 2 スクリーンサービスに対するトリガーを受信する段階 (s 5 1 0 2 0) は、第 2 スクリーンサービスに対するトリガーを受信する段階 (s 5 0 0 2 0) と同一であってもよい。

10

【 0 7 8 1 】

UI 情報に対して通知する段階 (s 5 1 0 3 0) は、UI 情報アップデートに対して通知する段階を含むことができる。

【 0 7 8 2 】

応答 “ o k ” を送信する段階 (s 5 1 0 4 0) は、UI 通知が受信されたことを示すメッセージを送信する段階を含むことができる。

【 0 7 8 3 】

受信機に対するトリガーを受信する段階 (s 5 1 0 5 0)、及び受信機に対するトリガーを受信する段階 (s 5 1 0 6 0) は、受信機に対するトリガーを受信する段階 (s 5 0 0 1 0) と同一であってもよい。すなわち、上述した動作が再び行われてもよい。

20

【 0 7 8 4 】

第 2 スクリーンサービスに対するトリガーを受信する段階 (s 5 1 0 7 0) は、第 2 スクリーンサービスに対するトリガーを受信する段階 (s 5 0 0 2 0) と同一であってもよい。すなわち、上述した動作が再び行われてもよい。

【 0 7 8 5 】

UI 情報に対して通知する段階 (s 5 1 0 8 0) は、UI 情報に対して通知する段階 (s 5 1 0 3 0) と同一であってもよい。すなわち、上述した動作が再び行われてもよい。

【 0 7 8 6 】

応答 “ o k ” を送信する段階 (s 5 1 0 9 0) は、応答 “ o k ” を送信する段階 (s 5 1 0 4 0) と同一であってもよい。すなわち、上述した動作が再び行われてもよい。

30

【 0 7 8 7 】

図 5 1 の方法で、受信機は、どの U P n P デバイスが第 2 スクリーンデバイスであるか、及びいかなるデバイスプロファイルが含まれるかを知っていなければならない。特に、受信機は、第 2 スクリーンデバイスが第 2 スクリーンサービスを支援するかを知っていなければならない。

【 0 7 8 8 】

トリガーが第 2 スクリーンデバイス又はプライマリデバイスに関するものかを決定した後、UI 情報に対して通知する方法、及び第 2 スクリーンデバイスに適切に用いられると決定された第 2 スクリーンデバイスに対する全ての UI 情報を伝達する方法 (図 5 1 の場合) は、受信機が T P T 及びトリガーを処理し、T D O U R L のみを第 2 スクリーンデバイスに伝達するという点で同一でありうる。これら 2 つの方法は、受信機が T D O を第 2 スクリーンデバイスに間接的に伝達したり、受信機が各デバイスのデバイスプロファイルを直接記録し、第 2 スクリーンデバイスにのみ第 2 スクリーンアプリケーションの位置を通知するという点で異なりうる。

40

【 0 7 8 9 】

図 5 2 は、R e m o t e U I サーバサービスに対する放送シグナリングの実施例を示す図である。

【 0 7 9 0 】

R e m o t e U I サーバサービスに対する放送シグナリングの一実施例は、U P n P デバイス及びサービスディスカバリ段階 (s 5 2 0 1 0)、U I L i s t i n g を要求す

50

る段階 (s 5 2 0 2 0)、U I L i s t i n g を送信する段階 (s 5 2 0 3 0)、イベントを申し込む段階 (s 5 2 0 4 0)、H T T P メッセージを返還する段階 (s 5 2 0 5 0)、U I L i s t i n g U p d a t e メッセージを送信する段階 (s 5 2 0 6 0)、U I L i s t i n g を要求する段階 (s 5 2 0 7 0) 及び/又は U I L i s t i n g を返還する段階 (s 5 2 0 8 0) を含むことができる。

【 0 7 9 1 】

U P n P デバイス及びサービスディスカバリ段階 (s 5 2 0 1 0) は、受信機及び第 2 スクリーンデバイスを探す段階を含むことができる。ホームネットワークに新しく接続されたデバイス又はホームネットワークに既に接続されたデバイス (受信機又は第 2 スクリーンデバイス) は、ディスカバリのためのメッセージをマルチキャストすることができる。このとき、所望のサービスをマルチキャストを用いて探索することができ、複数の不特定 U P n P デバイスに対して全てのサービスを探査することができる。これは、デバイスによってどのサービスが提供されるかによって変更されてもよい。この段階で、第 2 スクリーンデバイスは、プライマリデバイスのデバイスプロファイルを知っていてもよい。プライマリデバイスは、デバイスプロファイルを処理し、適切な U I L i s t i n g を生成することができる。R e m o t e U I サーバーは、C o m p a t i b l e U I S X S D スキーマ (s c h e m a) を第 2 スクリーンデバイスに提供することができる。このスキーマは、アプリケーションの URL、UI に対する固有 id (“ u i I D ”)、ネーム、p r o t o c o l I n f o などを含むことができる。

【 0 7 9 2 】

U I L i s t i n g を要求する段階 (s 5 2 0 2 0) は、第 2 スクリーンデバイスが自身のデバイスプロファイルを送信する段階、及び U I L i s t i n g を要求する段階を含むことができる。互換可能な UI が得られる G e t C o m p a t i b l e U I S アクションを用いることができる (U I F i l t e r は選択的であってもよい)。その詳細な説明は後述する。

【 0 7 9 3 】

U I L i s t i n g を送信する段階 (s 5 2 0 3 0) は、プライマリデバイスが、要求によって、適切な UI リスティングを第 2 スクリーンデバイスに送信する段階を含むことができる。その詳細な説明は後述する。

【 0 7 9 4 】

イベントを申し込む段階 (s 5 2 0 4 0) は、プライマリデバイスの適切な露出されたイベントを申し込む段階を含むことができる。イベントを申し込む段階 (s 5 2 0 4 0) は、R e m o t e U I サーバーサービスによって提供されたイベント情報である “ U I L i s t i n g U p d a t e ” を受信するために選択的に行うことができる。I n U I L i s t i n g を返還する段階 (s 5 2 0 8 0) で、第 2 スクリーンデバイスは、R e m o t e U I のアドレスが変更されたり又は UI が変更されたという通知を受けすることができる。すなわち、第 2 スクリーンデバイスが UI が変更された通知のみを受けると、詳細な位置又は追加の情報が要求されると、“ G e t C o m p a t i b l e U I S ” アクションが受信機に送信され、“ U I L i s t i n g ” 値を得ることができる。

【 0 7 9 5 】

H T T P メッセージを返還する段階 (s 5 2 0 5 0) は、イベントを申し込む段階 (s 5 2 0 4 0) の結果を送信する段階を含むことができる。応答 1 を送信する段階 (s 4 0 1 0 0) と同様に、H T T P / 1 . 1 2 0 0 O K などの応答が状況によって伝送される。

【 0 7 9 6 】

U I L i s t i n g U p d a t e メッセージを送信する段階 (s 5 2 0 6 0) は、加入者に “ U I L i s t i n g U p d a t e ” メッセージを送信する段階を含むことができる。

【 0 7 9 7 】

U I L i s t i n g を要求する段階 (s 5 2 0 7 0) は、第 2 スクリーンデバイスが自

10

20

30

40

50

身のデバイスプロファイルを送信する段階、及びUI Listingを要求する段階を含むことができる。互換可能なUIが得られるGet Compatible UIsアクションを用いることができる。UI Listing Updateメッセージを送信する段階(s 52060)及びUI Listingを要求する段階(s 52070)は、第2スクリーンアプリケーションが変更されず、サーバーによって支援される時にのみ選択的に可能なように時間設定を行う段階を含むことができる。この段階は選択的だ。この方法は、受信機でUPnPイベントを用いて、第2スクリーンサービスが存在することを全ての第2スクリーンデバイスに通知する方法であってもよい。すなわち、ホームネットワーク内の全ての第2スクリーンデバイスは第2スクリーンサービスと関連してUIが変更されたとの通知を受けることができる。全ての第2スクリーンデバイスにUPnPイベントにて“UI Listing Update”変数が通知されると、第2スクリーンデバイスは、UIが新しくアップデートされたということを確認することができる。その後、第2スクリーンデバイスは、“Get Compatible UIs”アクションを行い、ハードウェア装置に適したUIを選択し、デバイスプロファイルを参考にすることによってUIを実行することができる。

10

【0798】

UI Listingを返還する段階(s 52080)は、プライマリデバイスが要求によって適切なUIリスティングを第2スクリーンデバイスに送信する段階を含むことができる。上述した通り、UI Listingを返還する段階(s 52080)は、Remote UIのアドレスが変更されたりUIが変更されたことを申請した第2スクリーンデバイスに通知する段階を含むことができる。すなわち、第2スクリーンデバイスは、UIが変更されたことのみが通知されるため、詳細な位置又は追加の情報が要求されると、“UI Listing”を得るために“Get Compatible UIs”アクションを受信機に送信しなければならない。

20

【0799】

Remote UIサーバーサービスが動作すると、受信機は、要求する装置のデバイスプロファイルによって遠隔UI情報を提供しなければならない。逆互換性の観点から、デバイスプロファイル“Device Info”が交換され、後で受信機に“Get Compatible UIs”アクションを要求する時、Remote UIサーバーサービスによって送信されたURLが、要求するクライアントのプロファイルによって変更されうる。レガシーUPnPデバイスがRemote UIサーバーに“Get Compatible UIs”アクションを要求すると、レガシーUPnPデバイスのDevice Infoに適したUIが提供されなければならない。しかし、第2スクリーンサービスを支援するデバイスが“Get Compatible UIs”アクションを要求すると、Remote UIサーバーサービス装置はさらにURL情報を送信しなければならない。

30

【0800】

受信機は、DTVCCを通じてトリガー情報を得、トリガー情報からメディア時間及びイベント時間情報を得ることができる。このとき、“appID”、“URL”(TDO_URL)、“eventID”、“action”、及び、選択的に、TDO又は実行時間に対する“Data”は、“t=media_time”シンタックスを用いて確認できる。受信機が第2スクリーンアプリケーションのライフサイクルを管理すると、第2スクリーンサービスアプリケーションによって処理される情報の量が減少しうる。逆に、第2スクリーンサービスアプリケーションが第2スクリーンアプリケーションのライフサイクルを直接管理すると、必要な情報が増加しうる。

40

【0801】

図53は、分散型実行モデルの動作概念を示す図である。

【0802】

第2スクリーンデバイスにインタラクティブサービスを効率的に伝達する方法として、分散型実行モデルを説明する。

【0803】

50

分散型実行モデルの動作概念を示す図に示すように、TV受像機は、UPnPなどを用いて第2スクリーンデバイスにトリガーなどの情報を伝達することができる。それぞれの第2スクリーンデバイスはトリガーハンドラーを有し、トリガーハンドラーを通じて受信されたトリガーなどの情報を実時間で処理することができる。ブラウザは、それに関連したインタラクティブサービスを実行することができる。

【0804】

中央集中型実行モデルと違い、各第2スクリーンデバイスにDAEが存在してもよい。

【0805】

分散型実行モデルに基づいて第2スクリーンデバイスにインタラクティブサービスを伝達するプロセスは、次の順序で行うことができる。1.受信機は、第2スクリーン支援を有するセグメントを提示している。2.受信機は、第2スクリーン支援が利用可能な任意の指示をディスプレイする。3.ユーザは、第2スクリーンデバイス上で第2スクリーンアプリケーションを開始する。4.受信機及び第2スクリーンデバイスは、(受信機上のネイティブコード及びデバイス上の第2スクリーンアプリケーションによって行われる)UPnPメカニズムを通じてお互い発見する。5.TV受像機は、(DTVCCストリーム又はACRから)トリガーを得、第2スクリーンデバイス上でTDOを開始する。6.TV受像機は、トリガーをTPTからの情報と結合し、活性化時間に第2スクリーンデバイスに送信する。7.第2スクリーンデバイスは、TDO(一つ以上のファイル)をダウンロードしてブラウザで実行する。8.ユーザはTDOと相互作用してTDOが追加のファイルをダウンロードできるようにする。9.受信機は、トリガーを得、第2スクリーンデバイス上でTDOに対するイベントを生成する。10.受信機はトリガーとTPTからの情報とを結合し、活性化時間に第2スクリーンデバイスに送信する。11.第2スクリーンデバイスは、ブラウザでTDOに対する(DVB Stream Eventのような)Trigger Eventを生成する。12.TDOは、Trigger Eventによって要求されたアクションを行ってデータファイルをダウンロードすることができる。

【0806】

図54は、分散型実行モデルベース受信機及び第2スクリーン間の連動に対するフローを示す図である。

【0807】

分散型実行モデルのフローは、受信機がDTVCC又はACRサーバーを通じて受信された情報をそのまま第2スクリーンに送信する場合のデータフローであってもよい。以下、詳細なフローを説明する。

【0808】

分散型実行モデルベース受信機及び第2スクリーン間の連動に対するフローの実施例は、時間ベーストリガーを送信する段階(s54010)、時間ベーストリガーを送信する段階(s54020)、TPTを要求する段階(s54030)、TPTを応答として送信する段階(s54040)、TPTをパーシングする段階(s54050)、第2スクリーンに対する実行トリガーを送信する段階(s54060)、実行トリガーを送信する段階(s54070)、TDO/ファイルをダウンロードする段階(s54080)、ブラウザでTDOを実行する段階(s54090)、ユーザが新しいページに対するリンクをクリックする段階(s54100)、新しいTDO/ファイルをダウンロードする段階(s54110)、及びブラウザで新しいTDOを実行する段階(s54120)を含むことができる。

【0809】

時間ベーストリガーを送信する段階(s54010)は、時間ベーストリガーを送信する段階(s49010)と同一であってもよい。

【0810】

時間ベーストリガーを送信する段階(s54020)は、受信機が受信されたトリガーをそのまま第2スクリーンに送信する段階を含むことができる。

【0811】

10

20

30

40

50

TPTを要求する段階(s54030)は、第2スクリーンが受信されたトリガーを解釈する段階、TPTが得られるサービスのURLを取得する段階、及びTPTを要求できる段階を含むことができる。

【0812】

応答としてTPTを送信する段階(s54040)は、TPTを要求する段階(s54030)における要求に応答してTPTを送信する段階を含むことができる。

【0813】

TPTをパーシングする段階(s54050)は、要求されたTPTをパーシングする段階を含むことができる。

【0814】

第2スクリーンに対する実行トリガーを送信する段階(s54060)は、第2スクリーンに対する実行トリガーを送信する段階(s49050)と同一であってもよい。

【0815】

実行トリガーを送信する段階(s54070)は、受信機が受信されたトリガーをそのまま第2スクリーンに送信する段階を含むことができる。

【0816】

TDO/ファイルをダウンロードする段階(s54080)は、TPTからのトリガーと関連したTDO/ファイルを得る段階、及びコンテンツサーバーからTDO/ファイルをダウンロードする段階を含むことができる。

【0817】

ブラウザでTDOを実行する段階(s54090)は、ダウンロードされたTDO及びファイルをブラウザ上で実行する段階を含むことができる。

【0818】

ユーザが新しいページに対するリンクをクリックする段階(s54100)は、ユーザが、実行されたTDO上で特定リンクをクリックする段階を含むことができる。

【0819】

新しいTDO/ファイルをダウンロードする段階(s54110)は、他のTDOへの特定リンクが確立されると、関連したTDO/ファイルをダウンロードする段階を含むことができる。

【0820】

ブラウザで新しいTDOを実行する段階(s54120)は、ブラウザ上で関連したTDO及びファイルを実行する段階を含むことができる。

【0821】

図55は、分散型実行モデルベース受信機及び第2スクリーン間の連動に対するフローを示す図である。

【0822】

分散型実行モデルのフローは、受信機がDTVC C又はACRサーバーを通じて受信された情報をそのまま第2スクリーンに送信せず、第2スクリーンに適したトリガーによって必要な情報を挿入し、その情報を拡張されたトリガーに変更し、その情報を第2スクリーンに送信する場合のデータフローであってもよい。以下、詳細なフローを説明する。

【0823】

分散型実行モデルベース受信機及び第2スクリーン間の連動に対するフローの実施例は、時間ベーストリガーを送信する段階(s55010)、TPTを要求する段階(s55020)、応答としてTPTを送信する段階(s55030)、TPTをパーシングする段階(s55040)、第2スクリーンに対する実行トリガーを送信する段階(s55050)、拡張されたトリガーを生成する段階(s55060)、拡張されたトリガーを送信する段階(s55070)、TDO/ファイルをダウンロードする段階(s55080)、ブラウザでTDOを実行する段階(s55090)、ユーザが新しいページに対するリンクをクリックする段階(s55100)、新しいTDO/ファイルをダウンロードする段階(s55110)、及び/又はブラウザで新しいTDOを実行する段階(s551

10

20

30

40

50

20)を含むことができる。

【0824】

時間ベーストリガーを送信する段階(s55010)は、時間ベーストリガーを送信する段階(s54010)と同一であってもよい。

【0825】

TPTを要求する段階(s55020)は、TPTを要求する段階(s54030)と同一であってもよい。

【0826】

応答としてTPTを送信する段階(s55030)は、応答としてTPTを送信する段階(s54040)と同一であってもよい。

【0827】

TPTをパーシングする段階(s55040)は、TPTをパーシングする段階(s54050)と同一であってもよい。

【0828】

第2スクリーンに対する実行トリガーを送信する段階(s55050)は、第2スクリーンに対する実行トリガーを送信する段階(s54060)と同一であってもよい。

【0829】

拡張されたトリガーを生成する段階(s55060)は、受信機がTPTからのトリガーと関連したTDO及びファイルに関する情報を取得する段階、及びそれを含む拡張されたトリガーを生成する段階を含むことができる。

【0830】

拡張されたトリガーを送信する段階(s55070)は、受信機が拡張されたトリガーを第2スクリーンに送信する段階を含むことができる。

【0831】

TDO/ファイルをダウンロードする段階(s55080)は、TDO/ファイルをダウンロードする段階(s54080)と同一であってもよい。

【0832】

ブラウザでTDOを実行する段階(s55090)は、ブラウザでTDOを実行する段階(s54090)と同一であってもよい。

【0833】

ユーザが新しいページに対するリンクをクリックする段階(s55100)は、ユーザが新しいページに対するリンクをクリックする段階(s54100)と同一であってもよい。

【0834】

新しいTDO/ファイルをダウンロードする段階(s55110)は、新しいTDO/ファイルをダウンロードする段階(s54110)と同一であってもよい。

【0835】

ブラウザで新しいTDOを実行する段階(s55120)は、ブラウザで新しいTDOを実行する段階(s54120)と同一であってもよい。

【0836】

図56は、受信機が第2スクリーンデバイスにTDO及びイベント情報を通知する方法の一実施例を示す図である。

【0837】

図56は、受信機がトリガー及びTPTを受信し、トリガーによって処理を行い、第2スクリーンデバイスに対するトリガーが到達すると、トリガーをTPTと比較して、第2スクリーンデバイスに認識される必要がある情報を抽出してXMLファイルの形態で構成し、その情報を第2スクリーンデバイスに送信する方法を示す。この方法は、第2スクリーンデバイスが活発に動作して予測を行えるという点で有利である。

【0838】

このプロセスは、受信機に対するトリガーを受信する段階(s56010)、第2スク

10

20

30

40

50

リーンデバイスに対するトリガーを受信する段階 (s 5 6 0 2 0)、T P T 及びトリガーを変換 (t r a n s l a t e) し、イベント情報を生成する段階 (s 5 6 0 3 0)、T D O 及びイベント情報を送信する段階 (s 5 6 0 4 0)、応答 “ o k ” を送信する段階 (s 5 6 0 5 0)、受信機に対するトリガーを受信する段階 (s 5 6 0 6 0)、受信機に対するトリガーを受信する段階 (s 5 6 0 7 0)、第 2 スクリーンデバイスに対するトリガーを受信する段階 (s 5 6 0 8 0)、T P T 及びトリガーを変換してイベント情報を生成する段階 (s 5 6 0 9 0)、T D O 及びイベント情報を送信する段階 (s 5 6 1 0 0)、及び応答 “ o k ” を送信する段階 (s 5 6 1 1 0) を含むことができる。

【 0 8 3 9 】

受信機に対するトリガーを受信する段階 (s 5 6 0 1 0) は、受信機に対するトリガーを受信する段階 (s 5 0 0 1 0) と同一であってもよい。

10

【 0 8 4 0 】

第 2 スクリーンデバイスに対するトリガーを受信する段階 (s 5 6 0 2 0) は、第 2 スクリーンデバイスに対するトリガーを受信する段階 (s 5 0 0 2 0) と同一であってもよい。

【 0 8 4 1 】

T P T 及びトリガーを変換してイベント情報を生成する段階 (s 5 6 0 3 0) は、T P T 及びトリガーを解釈する段階、及びイベントを生成する段階を含むことができる。生成された情報は、新しいデータ構造を生成するために、T P T に含まれた情報及びトリガーを結合するのに用いられ、どのような T D O が生成されたか、又はいつ T D O が生成されたかに関する情報を含むことができる。このデータ構造は後述する。新しいデータ構造が決定されると、T D O U R L に加えて、様々な必要な情報が送信されうる。

20

【 0 8 4 2 】

T D O 及びイベントを送信する段階 (s 5 6 0 4 0) は、生成されたイベント情報及び T D O を第 2 スクリーンデバイスに送信する段階を含むことができる。

【 0 8 4 3 】

応答 “ o k ” を送信する段階 (s 5 6 0 5 0) は、受信された T D O 及びイベント情報が受信されたことを示すメッセージを送信する段階を含むことができる。

【 0 8 4 4 】

受信機に対するトリガーを受信する段階 (s 5 6 0 6 0)、及び受信機に対するトリガーを受信する段階 (s 5 6 0 7 0) は、受信機に対するトリガーを受信する段階 (s 5 0 0 1 0) と同一であってもよい。

30

【 0 8 4 5 】

第 2 スクリーンデバイスに対するトリガーを受信する段階 (s 5 6 0 8 0) は、第 2 スクリーンデバイスに対するトリガーを受信する段階 (s 5 0 0 2 0) と同一であってもよい。

【 0 8 4 6 】

T P T 及びトリガーを変換してイベント情報を生成する段階 (s 5 6 0 9 0) は、T P T 及びトリガーを変換してイベント情報を生成する段階 (s 5 6 0 3 0) と同一であってもよい。

40

【 0 8 4 7 】

T D O 及びイベント情報を送信する段階 (s 5 6 1 0 0) は、T D O 及びイベント情報を送信する段階 (s 5 6 0 4 0) と同一であってもよい。

【 0 8 4 8 】

応答 “ o k ” を送信する段階 (s 5 6 1 1 0) は、応答 “ o k ” を送信する段階 (s 5 6 0 5 0) と同一であってもよい。

【 0 8 4 9 】

受信機に対するトリガーを受信する段階 (s 5 6 0 6 0)、受信機に対するトリガーを受信する段階 (s 5 6 0 7 0)、第 2 スクリーンデバイスに対するトリガーを受信する段階 (s 5 6 0 8 0)、T P T 及びトリガーを変換してイベント情報を生成する段階 (s 5

50

6090)、TDO及びイベント情報を送信する段階(s56100)、及び応答“ok”を送信する段階(s56110)は、上述した動作と同一であってもよい。

【0850】

図57は、第2スクリーンデバイスがTPT及び第2スクリーンアプリケーションにアクセスする方法の実施例を示す図である。

【0851】

第2スクリーンデバイスは、受信機がTPT及びAMTなどのXMLファイルを受信したり、インターネットを通じてTPT及びAMTサーバーアドレスを知ると、第2スクリーンアプリケーションを直接実行できる独立装置である。この場合、トリガーモジュールが第2スクリーンデバイスに含まれる。第2スクリーンデバイスは、受信機によって受信されたiTVメッセージに対するURISTRINGを受信することができる。このメッセージは、1)RemoteUIサーバーサービスの場合、iTVメッセージ(トリガー)に対するURISTRINGを送信する方法、及び2)RemoteUIクライアントサービスの場合、iTVメッセージ(トリガー)に対するURISTRINGを送信する方法の両方に適用することができる。

【0852】

まず、RemoteUIサーバーサービスの場合、iTVメッセージ(トリガー)に対するURISTRINGを送信する方法を説明する。

【0853】

このプロセスは、受信機に対するトリガーを受信する段階(s57010)、第2スクリーンサービスに対するトリガーを受信する段階(s57020)、アップデートされたUI情報に関する通知を送信する段階(s57030)、アップデートされたUI情報を要求する段階(s57040)、応答としてUI情報を送信する段階(s57050)、TPTXMLファイルを要求する段階(s57060)、TPTXMLファイルを返還する段階(s57070)、第2スクリーンサービスに対するトリガーを受信する段階(s57080)、アップデートされたUI情報に対する通知を送信する段階(s57090)、アップデートされたUI情報を要求する段階(s57100)、応答としてUI情報を送信する段階(s57110)、第2スクリーンサービスに対するトリガーを受信する段階(s57120)、アップデートされたUI情報に対する通知を送信する段階(s57130)、アップデートされたUI情報を要求する段階(s57140)、応答としてUI情報を送信する段階(s57150)、第2スクリーンアプリケーションを要求する段階(s57160)、及び/又は第2スクリーンアプリケーションを返還する段階(s57170)を含むことができる。

【0854】

受信機に対するトリガーを受信する段階(s57010)は、受信機に対するトリガーを受信する段階(s50010)と同一であってもよい。第1トリガーは、第2スクリーンデバイスに対するものでないから、受信機はトリガーを第2スクリーンデバイスに伝達しない。

【0855】

第2スクリーンサービスに対するトリガーを受信する段階(s57020)は、第2スクリーンサービスに対するトリガーを受信する段階(s50020)と同一であってもよい。トリガーは、第2スクリーンサービスに対するメディア時間に関する情報を有することができる。第2トリガーは、上述したプリロード(pre-load)トリガーとして機能し得る。

【0856】

アップデートされたUI情報に対する通知を送信する段階(s57030)は、アップデートされたUI情報を通知する段階を含むことができる。第2トリガーは、第2スクリーンデバイスに対するものであるから、受信機は受信されたURISTRINGを第2スクリーンデバイスに送信することができる。このとき、第2スクリーンデバイスは、新しい情報が存在するという通知を受け、この情報を直接読み取ることができる

10

20

30

40

50

。

【0857】

アップデートされたUI情報を要求する段階(s57040)は、アップデートされたUI情報を要求する段階(s50040)と同一であってもよい。

【0858】

応答としてUI情報を送信する段階(s57050)は、プライマリデバイスがUI情報を第2スクリーンデバイスに送信する段階を含むことができる。この時、第2トリガーが送信されてもよい。

【0859】

TPT XMLファイルを要求する段階(s57060)は、第2スクリーンデバイスがプライマリデバイスから受信された情報(第2トリガー)をパーシングする段階、及びTPTサーバーにTPT XMLファイルを要求する段階を含むことができる。

10

【0860】

TPT XMLファイルを返還する段階(s57070)は、第2スクリーンデバイスがTPTサーバーから要求されたTPT XMLファイルをダウンロードする段階を含むことができる。

【0861】

第2スクリーンサービスに対するトリガーを受信する段階(s57080)は、第2スクリーンサービスに対するトリガーを受信する段階(s50020)と同一であってもよい。第3トリガーは、第2スクリーンデバイスと関連しており、メディア時間に関する情報を有することができる。第3トリガーは、上述した時間ベーストリガーと同一の機能を有し得るメディア時間トリガーである。

20

【0862】

アップデートされたUI情報に関する通知を送信する段階(s57090)は、アップデートされたUI情報を通知する段階を含むことができる。第3トリガーが第2スクリーンデバイスと関連したものと決定されるため、受信機は、第3トリガーを第2スクリーンデバイスに通知することができる。

【0863】

アップデートされたUI情報を要求する段階(s57100)は、アップデートされたUI情報を要求する段階(s50040)と同一であってもよい。

30

【0864】

応答としてUI情報を送信する段階(s57110)は、プライマリデバイスがUI情報を第2スクリーンデバイスに送信する段階を含むことができる。このとき、第3トリガーが送信されてもよい。しかし、第3トリガーは、メディア時間トリガーであるから、第2スクリーンデバイスのメディア時間のみを制御することができる。このため、第2スクリーンデバイス及び受信機間のメディア時間同期化を行うことができる。

【0865】

第2スクリーンデバイスに対するトリガーを受信する段階(s57120)は、第2スクリーンデバイスに対するトリガーを受信する段階(s50020)と同一であってもよい。第4トリガーは、第2スクリーンデバイスと関連しており、イベント時間に関する情報を有することができる。第4トリガーは、上述した活性化トリガーと同一の機能を有し得るイベント時間トリガーである。

40

【0866】

アップデートされたUI情報に対する通知を送信する段階(s57130)は、アップデートされたUIを通知する段階を含むことができる。第4トリガーは、第2スクリーンデバイスと関連したものと決定されるため、受信機は第2スクリーンデバイスに第4トリガーを通知することができる。

【0867】

アップデートされたUI情報を要求する段階(s57140)は、アップデートされたUI情報を要求する段階(s50040)と同一であってもよい。

50

【 0 8 6 8 】

応答としてUI情報を送信する段階 (s 5 7 1 5 0) は、プライマリデバイスがUI情報を第2スクリーンデバイスに送信する段階を含むことができる。このとき、第4トリガーが送信されてもよい。

【 0 8 6 9 】

第2スクリーンアプリケーションを要求する段階 (s 5 7 1 6 0) は、第2スクリーンデバイスが第4トリガーに関する情報をチェックする段階、及びTDO URLの位置の第2スクリーンアプリケーションをダウンロードするためにアプリケーションサーバーに第2スクリーンアプリケーションを要求する段階を含むことができる。

【 0 8 7 0 】

第2スクリーンアプリケーションを返還する段階 (s 5 7 1 7 0) は、要求によって第2スクリーンアプリケーションをダウンロードする段階を含むことができる。第2スクリーンアプリケーションがダウンロードされ、Event@Actionを行うことができる。このとき、アプリケーションサーバーに第2スクリーンデバイスのブラウザに関する情報を通知するので、アプリケーションサーバーは、どの第2スクリーンデバイスが接続されるかを容易にチェックすることができる。したがって、アプリケーションは自動で選択され、サーバーからダウンロードされることが可能である。

【 0 8 7 1 】

要約すると、iTVメッセージに対するURIストリングがDTVCCを通じて受信されると、受信機は、新しいUIがアップデートされたということを示すイベントを、発見された第2スクリーンデバイスに送信することができる。第2スクリーンデバイスは、イベント情報をチェックし、新しいUI情報を得るために“GetCompatibleUIs”アクションを送信することができる。受信機は、TPTサーバーの受信されたURI情報を送信することができる。この方法によって、第2スクリーンデバイスはURI情報を受信し、TPT及びAMT情報を直接処理し、第2スクリーンアプリケーションを直接制御することができる。

【 0 8 7 2 】

このプロセスは、ATSC 2.0トリガークライアントが第2スクリーンサービスアプリケーションに含まれる場合に可能である。

【 0 8 7 3 】

図58は、第2スクリーンデバイスがTPT及び第2スクリーンアプリケーションをアクセスする方法の実施例を示す図である。

【 0 8 7 4 】

上述した2つの方法のうち、RemoteUIクライアントサービスの場合にiTVメッセージ(トリガー)に対するURIストリングを送信する方法を説明する。

【 0 8 7 5 】

このプロセスは、受信機に対するトリガーを受信する段階 (s 5 8 0 1 0)、第2スクリーンサービスに対するトリガーを受信する段階 (s 5 8 0 2 0)、トリガーを通知する段階 (s 5 8 0 3 0)、応答“ok”を送信する段階 (s 5 8 0 4 0)、TPT XML ファイルを要求する段階 (s 5 8 0 5 0)、TPT XML ファイルを返還する段階 (s 5 8 0 6 0)、第2スクリーンデバイスに対するトリガーを受信する段階 (s 5 8 0 7 0)、トリガーを通知する段階 (s 5 8 0 8 0)、応答“ok”を送信する段階 (s 5 8 0 9 0)、第2スクリーンサービスに対するトリガーを受信する段階 (s 5 8 1 0 0)、トリガーを通知する段階 (s 5 8 1 1 0)、応答“ok”を送信する段階 (s 5 8 1 2 0)、第2スクリーンアプリケーションを要求する段階 (s 5 8 1 3 0) 及び/又は第2スクリーンアプリケーションを返還する段階 (s 5 8 1 4 0) を含むことができる。

【 0 8 7 6 】

受信機に対するトリガーを受信する段階 (s 5 8 0 1 0) は、プライマリデバイスが放送ストリームを通じて放送社から受信機、すなわち、プライマリデバイスに対するトリガーを受信する段階を含むことができる。第1トリガーが受信機に対するものであるから、

10

20

30

40

50

第1トリガーは第2スクリーンデバイスに伝達されない。

【0877】

第2スクリーンサービスに対するトリガーを受信する段階(s58020)は、第2スクリーンサービスに対するトリガーを受信する段階(s50020)と同一であってもよい。第2トリガーは、第2スクリーンサービスに対するトリガーであり、メディア時間に関する情報を有することができる。第2トリガーは、上述したプリロードトリガーとして機能し得る。

【0878】

トリガーを通知する段階(s58030)は、図57と違い、第2スクリーンデバイスに直ちにトリガーを送信する段階を含むことができる。iTVメッセージに対するURIストリングがD TVCCを通じて受信されると、受信機は、“AddU I L i s t i n g”アクションを用いて、受信機によって受信されたURI値を第2スクリーンデバイスに伝達することができる。

10

【0879】

応答“ok”を送信する段階(s58040)は、トリガーが受信されたことを示すメッセージを送信する段階を含むことができる。

【0880】

TPTXMLファイルを要求する段階(s58050)は、TPTXMLファイルを要求する段階(s57060)と同一であってもよい。第2スクリーンデバイスに含まれるトリガーモジュールは、受信されたURI値を用いてTPTサーバーにアクセスし、TPT及びAMTファイルをダウンロードし、第2スクリーンアプリケーションを直接制御することができる。

20

【0881】

TPTXMLファイルを返還する段階(s58060)は、TPTXMLファイルを返還する段階(s57070)と同一であってもよい。

【0882】

第2スクリーンサービスに対するトリガーを受信する段階(s58070)は、第2スクリーンサービスに対するトリガーを受信する段階(s50020)と同一であってもよい。第3トリガーは、第2スクリーンデバイスと関連しており、メディア時間に関する情報を有することができる。第3トリガーは、上述した時間ベーストリガーと同一の機能を有し得るメディア時間トリガーである。

30

【0883】

トリガーを通知する段階(s58080)は、トリガーを通知する段階(s58030)と同様に、トリガーを伝達する段階を含むことができる。

【0884】

応答“ok”を送信する段階(s58090)は、応答“ok”を送信する段階(s58040)と同一であってもよい。

【0885】

第2スクリーンサービスに対するトリガーを受信する段階(s58100)は、第2スクリーンサービスに対するトリガーを受信する段階(s50020)と同一であってもよい。第4トリガーは、第2スクリーンデバイスと関連しており、イベント時間に関する情報を有することができる。第4トリガーは、上述した活性化トリガーと同一の機能を有し得るイベント時間トリガーである。

40

【0886】

トリガーを通知する段階(s58110)は、トリガーを通知する段階(s58030)と同様に、トリガーを伝達する段階を含むことができる。

【0887】

応答“ok”を送信する段階(s58120)は、応答“ok”を送信する段階(s58040)と同一であってもよい。

【0888】

50

第2スクリーンアプリケーションを要求する段階(s 5 8 1 3 0)は、第2スクリーンアプリケーションを要求する段階(s 5 7 1 6 0)と同一であってもよい。

【0889】

第2スクリーンアプリケーションを返還する段階(s 5 8 1 4 0)は、第2スクリーンアプリケーションを返還する段階(s 5 7 1 7 0)と同一であってもよい。

【0890】

すなわち、受信機は、常に、第2スクリーンサービスに関連したイベント情報を有するトリガーを第2スクリーンデバイスに伝達することができ、第2スクリーンデバイスは、ダウンロードされたTPT情報をを用いて直ちに動作することができる。メディア時間トリガーがDTVCを用いて周期的に伝達されるため、受信機は、この情報を連続して伝達しなければならない。

10

【0891】

プライマリデバイス又は第2スクリーンデバイスがTPTXMLを有するので、イベントトリガーがライブトリガーサーバーから実時間で伝送される時にAMTも受信される。

【0892】

上述した2つの方法は、いかなるURL値が適用されるかによって異なるように行われ、受信機の動作又は第2スクリーンサービスアプリケーションの構造及び動作は変更されてもよい。

【0893】

20

第2スクリーンサービスのシグナリングメカニズムを説明する。

【0894】

第2スクリーンサービスは、2つの方法、すなわち、受信機が、第2スクリーンサービスが存在するということを知り、及び第2スクリーンデバイスがホームネットワークに接続されると、第2スクリーンサービスを提供する電子装置を検出するために装置及びサービスを探し第2方法を用いてシグナルすることができる。代案として、受信機は、新しい電子装置のデバイス記述語を確認し、サービス記述語を要求することができる。

【0895】

以下、放送シグナリング及びユニキャストシグナリングを説明する。

30

【0896】

放送シグナリングの場合、第2スクリーンデバイスは、RemoteUIサーバーサービスを検出し、デバイス記述語を確認し、第2スクリーンデバイスと互換可能なDeviceInfoプロファイルを要求する。イベントを受信でき、受信機に見られるプログラムによって変更されるTDO(インタラクティブアプリケーション)のURLを受信することができる。

【0897】

逆に、ユニキャストシグナリングの場合、受信機は、第2スクリーンデバイスのDeviceInfoが互換可能か否かを確認し、互換可能なTDOURLを送信することができる。RemoveUIListingアクションは、現在実行される第2スクリーンアプリケーションを終了してディスプレイメッセージなどのアクションを支援し、現在実行されるUIを終了するように送信されうる。受信機でパーシングされ処理される追加のEvent@dataが、ProcessInputアクションによって第2スクリーンサービスアプリケーションに伝達されうる。

40

【0898】

以下、放送シグナリング及びユニキャストシグナリングを説明する。

【0899】

図59は、RemoteUIサーバーサービスに対する放送シグナリングの他の実施例である。

【0900】

50

放送シグナリングで、最初に受信機がホームネットワークに接続されると、自身のデバイス記述語とサービス記述語を全ての家電機器に自分で知らせたり、他の制御ポイントからの要求に応じて送信することができる。

【0901】

Remote UIサーバーサービスに対する放送シグナリングの他の実施例は、UPnPデバイス及びサービスディスカバリ段階(s59010)、UIListingを送信する段階(s59020)、UIListingを送信する段階(s59030)、UIListingを要求する段階(s59040)、及び/又はUIListingを返還する段階(s59050)を含むことができる。

【0902】

UPnPデバイス及びサービスディスカバリ段階(s59010)は、UPnPデバイス及びサービスディスカバリ段階(s58010)と同一の段階であってもよい。

【0903】

UIListingを送信する段階(s59020)は、“UIListingUpdate”メッセージを全てのUPnPデバイスに送信する段階であってもよい。プライマリデバイスは、UIListingUpdateのアナウンスメント(announcement)を、固有“uiID”リストを有するUPnPデバイスに送信することができる。第2スクリーンデバイスは、このメッセージを受けた後、uiIDをチェックすることができる。しかし、特定uiIDとマッチングされず、この第2ストリーム装置ではUIアップデートが行われなくてもよい。

【0904】

UIListingを送信する段階(s59030)は、“UIListingUpdate”メッセージを全てのUPnPデバイスに送信する段階であってもよい。この時は、UIListingを送信する段階(s59020)の場合とは違い、マッチングされるuiIDが存在し得る。

【0905】

UIListingを要求する段階(s59040)は、UIListingを送信する段階(s59030)でuiIDがマッチングされるため、第2スクリーンデバイスが自身のデバイスプロファイルを送信し、UIListingを要求することができる。互換可能なUIを取得できるGetCompatibleUIsアクションを用いることができる。

【0906】

UIListingを返還する段階(s59050)は、プライマリデバイスが第2スクリーンデバイスに、要求に応じた適切なUIListingを送信する段階であってもよい。

【0907】

図60は、第2スクリーンサービスのための装置ディスカバリ及び装置能力交換の一実施例である。

【0908】

前述したとおり、最初に受信機がホームネットワークに接続されると、自身のデバイス記述語とサービス記述語を全ての家電機器に自分で知らせたり、他の制御ポイントからの要求に応じて送信することができる。

【0909】

受信機のデバイス記述語とサービス記述語を受信した、全てのホームネットワークに接続したUPnPデバイスは、自身のデバイス記述語の位置をHTTPの“LOCATION”ヘッダーで送ることができる。すなわち、UPnPデバイス記述語の位置を送信して自身を知らせることができる。“LOCATION”にHTTP GET要求をすると、当該装置に関する情報がわかるXMLファイルを受信することができる。

【0910】

UPnPデバイス及びサービスディスカバリ過程で、プライマリデバイスが、第2スク

10

20

30

40

50

リーンサービスが可能な第2スクリーンデバイスを探す方法を紹介する。この方法は、2つに大別することができる。その第一は、第2スクリーンデバイスが2種類のデバイスプロファイルを準備しておき、このデバイスプロファイルのXMLファイルをHTTPヘッダーで知らせる方法である。このとき、互換されないプライマリデバイスは、理解できないHTTPヘッダーはそのまま無視すると仮定することができる。その第二は、デバイスプロファイルのうち、第2スクリーンサービスを提供する第2スクリーンデバイスという情報を“Protocol Info”に含む方法である。

【0911】

図60は、第一の方式を例示するものである。

【0912】

第2スクリーンサービスに対する装置ディスカバリ及び装置能力交換の一実施例は、第2スクリーンサービスに対するUPnPアプリケーションを実行する段階(s60001)、UPnPデバイスを探す段階(s60010)、RemoteUIClientを探す段階(s60020)、デバイス記述を要求する段階(s60030)、デバイス記述を受信する段階(s60040)、サービス制御記述(description)を要求する段階(s60050)、サービス制御記述を受信する段階(s60060)、デバイスプロファイルを要求する段階(s60070)、デバイスプロファイルを受信する段階(s60080)、遠隔UIのURLを送る(put)段階(s60090)、応答1を送信する段階(s60100)、RemoteUIクライアントサービスにメッセージを送信する段階(s60110)、応答2を送信する段階(s60120)、及び/又はユーザがスクリーン上のボタンをクリックする段階(s60130)を含むことができる。

【0913】

第2スクリーンサービスに対するUPnPアプリケーションを実行する段階(s60001)、UPnPデバイスを探す段階(s60010)、RemoteUIClientを探す段階(s60020)、デバイス記述を要求する段階(s60030)、デバイス記述を受信する段階(s60040)、サービス制御記述を要求する段階(s60050)、サービス制御記述を受信する段階(s60060)、デバイスプロファイルを要求する段階(s60070)、デバイスプロファイルを受信する段階(s60080)、遠隔UIのURLを送る(put)段階(s60090)、応答1を送信する段階(s60100)、RemoteUIクライアントサービスにメッセージを送信する段階(s60110)、応答2を送信する段階(s60120)、及びユーザがスクリーン上のボタンをクリックする段階(s60130)は、それぞれ順に、第2スクリーンサービスに対するUPnPアプリケーションを実行する段階(s40001)、UPnPデバイスを探す段階(s40010)、RemoteUIClientを探す段階(s40020)、デバイス記述を要求する段階(s40030)、デバイス記述を受信する段階(s40040)、サービス制御記述を要求する段階(s40050)、サービス制御記述を受信する段階(s40060)、デバイスプロファイルを要求する段階(s40070)、デバイスプロファイルを受信する段階(s40080)、遠隔UIのURLを送る(put)段階(s40090)、応答1を送信する段階(s40100)、RemoteUIクライアントサービスにメッセージを送信する段階(s40110)、応答2を送信する段階(s40120)、及びユーザがスクリーン上のボタンをクリックする段階(s40130)と同一の段階であってもよい。

【0914】

第一の方式は、図60に示すように、UPnPデバイスを探す段階(s60010)段階後にHTTPヘッダーから得る“X-ATSC-COMPANION-LOCATION”ヘッダーで第2スクリーンサービスを支援するデバイスプロファイルの位置を知らせることができる。X-ATSC-COMPANION-LOCATION: http://10.177.56.36:37900/ATSC2ndScreenRemoteUIClient1.xml¥r¥nと表示された部分が“X-ATSC-COMPANION-LOCATION”ヘッダーである。受信機は、“LOCATION”ヘッダーは無

10

20

30

40

50

視し、その代わりに“ X - A T S C - C O M P A N I O N - L O C A T I O N ”ヘッダーから、受信機の所望する情報である第2スクリーンデバイスのデバイスプロファイルを得ることができる。

【0915】

プライマリデバイスが、第2スクリーンサービスが可能な第2スクリーンデバイスを探す方法のうち、前述した第二の方式は、“ L O C A T I O N ”ヘッダーの位置でデバイスプロファイルを得、デバイスプロファイル内で第2スクリーンデバイスの“ P r o t o c o l I n f o ”エレメントの値をパーシングして処理することができる。これは、前述した R e m o t e U I サーバサービス # 1 に対する放送シグナリングの一実施例のうち、 U I L i s t i n g 要求 (s 5 8 0 2 0) 段階で行うことができる。

10

【0916】

特定 U P n P デバイスのデバイス記述語には提供可能なサービスのリストがあり、一つ或いは複数個のサービスを提供するという事実がわかる。U P n P デバイスが有しているそれぞれのサービスは、遠隔の他の U P n P デバイスで制御可能であり、イベントを受信することができる。ただし、イベント受信は、提供するサービスがイベントで知らせ得る機能がある場合のみ可能である。U P n P デバイスが提供するサービスは、“制御”，“コントロール”そして“イベント”ができるように U R L を知らせることができる。

【0917】

図61は、U P n P フォーラムの D e v i c e P r o f i l e X M L スキーマの一実施例である。

20

【0918】

前述した R e m o t e U I サーバサービス # 1 に対する放送シグナリングの一実施例のうち、U I L i s t i n g 要求 (s 5 8 0 2 0) 段階についてさらに説明する。この段階は、第2スクリーンデバイスから R e m o t e U I サーバサービスのある受信機に、第2スクリーンデバイスに合う条件の U I があるか問うことであり、“ I n p u t D e v i c e P r o f i l e ”には、図61のような X S D ファイルのような形式を有しなければならない。

【0919】

したがって、第2スクリーンデバイスは受信機に、図61のような形式で自身のデバイスプロファイルと合う情報を入れて送信しなければならない。

30

【0920】

図62は、第2スクリーンデバイスのデバイスプロファイルの一実施例である。

【0921】

前述した R e m o t e U I サーバサービス # 1 に対する放送シグナリングの一実施例のうち、U I L i s t i n g 要求 (s 5 8 0 2 0) 段階についてさらに説明する。この段階で、図62は、デバイスプロファイルの一例を示している。U I L i s t i n g 要求 (s 5 8 0 2 0) 段階で、図示のデバイスプロファイルのような情報を受信機に送ることができる。

【0922】

第2スクリーンデバイスが探そうとするデバイスプロファイルの第2スクリーンサービスが、受信機にあるか否かを検索することであり、図示の情報を“ I n p u t D e v i c e P r o f i l e ”変数に格納して送信することができる。受信機では“ I n p u t D e v i c e P r o f i l e ”情報を確認し提供すべきサービスタイプ、バージョン、第2スクリーンデバイスの解像度、受信機可能なイメージエンコーディング方式等を定義することができる。

40

【0923】

図63は、第2スクリーンサービスに対する P r o t o c o l I n f o の説明の一実施例である。

【0924】

前述した R e m o t e U I サーバサービス # 1 に対する放送シグナリングの一実施例

50

のうち、U I L i s t i n g を要求する段階 (s 5 8 0 2 0) 及び U I L i s t i n g を送信する段階 (s 5 8 0 3 0) についてさらに説明する。

【 0 9 2 5 】

前述したとおり、受信機では “ I n p u t D e v i c e P r o f i l e ” 情報を確認し提供すべきサービスタイプ、バージョン、第2スクリーンデバイスの解像度、受信機可能なイメージエンコーディング方式などを定義することができる。“第2スクリーンサービスに対する P r o t o c o l I n f o の記述の一実施例” は、“ I n p u t D e v i c e P r o f i l e ” 情報の一実施例であってもよい。

【 0 9 2 6 】

図示のような情報を用いて受信機に伝達した時 (U I L i s t i n g を要求する段階 (s 5 8 0 2 0))、XML 形態となっている U I L i s i n g 情報を第2スクリーンデバイスに送信する (U I L i s t i n g を送信する段階 (s 5 8 0 3 0))。 10

【 0 9 2 7 】

もし、R e m o t e U I サーバサービスで所望する第2スクリーンデバイスを支援しない他の装置であると、U I L i s t i n g を送信する段階 (s 5 8 0 3 0) でエラー情報をリターンして、第2スクリーンサービスを支援できない R e m o t e U I サーバサービスであることを知らせることができる。

【 0 9 2 8 】

図 6 4 は、第2スクリーンデバイスで A d d U I L i s t i n g 及び R e m o t e U I L i s t i n g が実行される間の U I L i s t i n g の一実施例である。 20

【 0 9 2 9 】

前述した R e m o t e U I サーバサービス # 1 に対する放送シグナリングの一実施例のうち、U I L i s t i n g を要求する段階 (s 5 8 0 2 0) 及び U I L i s t i n g を送信する段階 (s 5 8 0 3 0) についてさらに説明する。

【 0 9 3 0 】

U I L i s t i n g を要求する段階 (s 5 8 0 2 0) 後に、受信機は、互換される R e m o t e U I 情報を探し、要求した第2スクリーンデバイスに伝達することができる (U I L i s t i n g を送信する段階 (s 5 8 0 3 0))。U I L i s t i n g を送信する段階 (s 5 8 0 3 0) で受信された情報は、図 6 4 に示す。 30

【 0 9 3 1 】

この受信された情報は、受信機が第2スクリーンに送信する時、T P T 及び A M T を処理して T D O U R L を入れるのもであってもよい。したがって、第2スクリーンデバイスは、R e m o t e U I の情報のみを受信機から得、第2スクリーンアプリケーションは、ホームネットワーク外のインターネットアプリケーションサーバからダウンロードして実行することができる。このとき、実行することは、第2スクリーンサービスアプリケーションが担当することができる。

【 0 9 3 2 】

一方では、放送社からあらかじめ放送網や受信機に第2スクリーンデバイスのための第2スクリーンアプリケーションを送信してもよい。この場合には、受信機が第2スクリーンサービスのためのアプリケーションを保存しておき、直接サービスすることができる。 40
この場合、ウェブサーバが受信機で動作しなければならず、関連したイメージ及びデータは、受信機或いはホームネットワーク以外の外部アプリケーションサーバからダウンロードすることができる。D O が N D O である場合に、A T S C - N R T で N D O 及び関連リソースファイルを放送網に送信しておき、第2スクリーンサービスを提供することができる。

【 0 9 3 3 】

放送シグナリングの場合、第2スクリーンアプリケーションのライフサイクルは、受信機又は第2スクリーンサービスアプリケーションが管理することができる。

【 0 9 3 4 】

まず、受信機が第2スクリーンアプリケーションのライフサイクルを管理する方法につ 50

いて説明する。

【0935】

受信機は、DTVCCで伝送されるメディア時間トリガーの情報を処理後に、第2スクリーンと関連したイベントがTPTに存在し、このイベントが実行されなければならない時、すぐに“UIListingUpdate”変数を、全てのホームネットワークにある機器に知らせることができる。このようなイベントに関連して申請した機器は、UI情報がアップデートされたことを認知し、必要な情報を得るために“GetCompatibleUIs”アクションを行うことができる。この時、受信機のRemoteUIサーバーでは、すぐに実行しなければならないTDOアドレスを渡すことができる。

【0936】

次に、第2スクリーンサービスアプリケーションが第2スクリーンアプリケーションのライフサイクルを管理する方法について説明する。

【0937】

この場合には、受信機が受信した情報を第2スクリーンデバイスに伝達し、第2スクリーンサービスアプリケーションが自ら動作できるようにすることができる。放送の場合、第2スクリーンサービスとなるクライアントが“GetCompatibleUIs”アクションを要求すると、受信機がDTVCCで直接受信したiTVメッセージ(トリガー)のURIStringをリターンすることができ、第2スクリーンデバイスが、受信したURIStringを用いてTPTファイルをダウンロードし、受信機と同様にTPTを解釈して動作することができる。受信機はDTVCCでメディア時間トリガーやイベント時間トリガーを受信する度にすぐに“UIListingUpdate”変数を変更して全ての機器に送信することができ、新しいiTVメッセージ(トリガー)のURIStringを受信できるように、機器が“GetCompatibleUIs”アクションをするように誘導することができる。

【0938】

図65は、RemoteUIクライアントサービスのためのユニキャストシグナリングの一実施例である。

【0939】

前述したユニキャストシグナリングについて説明する。

【0940】

この方法は、第2スクリーンデバイスがRemoteUIClientサービスを装置内に内蔵している場合といえる。UPnPデバイスが、最初にホームネットワークに接続すると、自身の情報を自分で通知し、他の装置にも新しい装置がホームネットワークに接続したという事実を知らせることができる。もう一つの方法は、周期的に新しい装置があったら知らせるように要求するメッセージを、全てのホームネットワークに接続した装置に伝達すると、全てのホームネットワークに接続した装置はこの情報に応答して通知(NOTIFY)メッセージをさらに全ての装置に知らせる方法である。まず、第2スクリーンデバイスが第2スクリーンサービスを支援するか否かについて調べなければならない。

【0941】

RemoteUIクライアントサービスに対するユニキャストシグナリングの一実施例は、UPnPデバイス及びサービスディスカバリ段階(s65010)、デバイスプロファイルを要求する段階(s65020)、デバイスプロファイルを返還する段階(s65030)、TDOURL情報を送信する段階(s65040)、HTTPメッセージを返還する段階(s65050)、ディスプレイメッセージを送信する段階(s65060)、及び/又はHTTPメッセージを送信する段階(s65070)を含むことができる。

【0942】

UPnPデバイス及びサービスディスカバリ段階(s65010)は、UPnPデバイス及びサービスディスカバリ段階(s58010)と同一の段階であってもよい。

【0943】

10

20

30

40

50

デバイスプロファイルを要求する段階 (s 6 5 0 2 0) は、受信機が新しく発見された第2スクリーンデバイスに第2スクリーンサービスを支援できるかを確認するために “ S t a t i c D e v i c e I n f o ” 情報を送信する段階であってもよい。

【 0 9 4 4 】

デバイスプロファイルを返還する段階 (s 6 5 0 3 0) は、デバイスプロファイルを要求する段階 (s 6 5 0 2 0) に対する応答として、第2スクリーンデバイスのデバイスプロファイルを得る段階であってもよい。もし、新しく発見された第2スクリーンデバイスが送ったデバイスプロファイルが “ S t a t i c D e v i c e I n f o ” と一致しないか、又は第2スクリーンデバイスを支援しない装置と判断されると、第2スクリーンサービスを開始しなくてもよい。“ S t a t i c D e v i c e I n f o ” は、上記で定義した “ D e v i c e I n f o ” と同一であってもよい。“ S t a t i c D e v i c e I n f o ” と D e v i c e I n f o 情報が一致するかは受信機で判断しなければならない。もし、ホームネットワークに新しく発見された第2スクリーンデバイスが一つ以上である場合、このような過程は繰り返される。一度発見された第2スクリーンデバイスで第2スクリーンサービスを支援しないからといって、このような過程を経ないというわけにはいかない。第2スクリーンデバイスには一つ以上の複数のソフトウェアが設置及び削除されてもよく、ユーザが実行するか否かによってこの過程に対する結果値が変わってもよいからである。第2スクリーンサービスアプリケーションを行った状態になると、T D O U R L 情報を送信する段階 (s 6 5 0 4 0) に移行することができる。

【 0 9 4 5 】

T D O U R L 情報を送信する段階 (s 6 5 0 4 0) は、受信機が T P T 及び A M T をパーシングした結果である T D O U R L 情報を送信する段階であってもよい。U P n P R e m o t e U I クライアントサービスで定義された “ A d d U I S t r i n g ” を用いることができる。“ A d d U I S t r i n g ” は、第2スクリーンデバイスで第2スクリーンサービスの D e v i c e I n f o 情報を満たす場合に実行されるアクションであってもよい。T D O U R L は、一つ或いは一つ以上の URL になってもよい。一つ以上の URL である場合、すぐに実行するための URL を送信することができる。このとき、選択的にディスプレイメッセージを送信する段階 (s 6 5 0 6 0) を行うことができる。

【 0 9 4 6 】

H T T P を返還する段階 (s 6 5 0 5 0) は、T D O U R L 情報を送信する段階 (s 6 5 0 4 0) に対する結果を送る段階であってもよい。前述した応答 1 を送信する段階 (s 4 0 1 0 0) と同様に、状況によって、H T T P / 1 . 1 2 0 0 O K などの応答を送ることができる。

【 0 9 4 7 】

ディスプレイメッセージを送信する段階 (s 6 5 0 6 0) は、第2スクリーンデバイスにディスプレイメッセージを送信する段階であってもよい。R e m o t e U I クライアントサービスは、第2スクリーンデバイス画面にメッセージを表示できる D i s p l a y M e s s a g e アクションを有しうるため、第2スクリーンデバイスに T D O U R L を送信し、D i s p l a y M e s s a g e アクションを用いて、画面に現在視聴中の放送プログラムと関連付けられた第2スクリーンアプリケーションがあるというメッセージを表示することができる。ユーザがこのメッセージを確認するとすぐに第2スクリーンアプリケーション実行となり得る。

【 0 9 4 8 】

H T T P メッセージを返還する段階 (s 6 5 0 7 0) は、ディスプレイメッセージを送信する段階 (s 6 5 0 6 0) に対する結果を送る段階であってもよい。前述した応答 1 を送信する段階 (s 4 0 1 0 0) と同様に、状況によって、H T T P / 1 . 1 2 0 0 O K などの応答を送ることができる。

【 0 9 4 9 】

図 6 6 は、R e m o t e U I クライアントサービスのためのユニキャストシグナリングの一実施例である。

10

20

30

40

50

【0950】

新しいUIのURLが第2スクリーンデバイスのUIListingに追加される度に、第2スクリーンサービスアプリケーションは新しい第2スクリーンアプリケーションを実行する環境を提供することができる。

【0951】

図66のRemoteUIクライアントサービスのためのユニキャストシグナリングの一実施例は、実行中の第2スクリーンアプリケーションの実行を中断し、新しい第2スクリーンアプリケーションを実行するための順序を示している。

【0952】

RemoteUIクライアントサービスのためのユニキャストシグナリングの一実施例は、RemoveUIListingアクションを送信する段階(s66010)、HTTPメッセージを返還する段階(s66020)、ディスプレイメッセージを送信する段階(s66030)、HTTPメッセージを返還する段階(s66040)、TDOURL情報を送信する段階(s66050)、HTTPメッセージを返還する段階(s66060)、ディスプレイメッセージを送信する段階(s66070)、及び/又はHTTPメッセージを返還する段階(s66080)を含むことができる。

10

【0953】

RemoveUIListingアクションを送信する段階(s66010)は、RemoteUIクライアントサービスに定義されている“RemoveUIListing”アクションを用いて、第2スクリーンサービスアプリケーションで実行中の第2スクリーンアプリケーションを中止させることができる。第2スクリーンサービスアプリケーションは、受信機で現在実行中であるUIのアプリケーション使用中止をする時、RemoveUIListingアクションを送るということを知っていてもよい。したがって、第2スクリーンサービスアプリケーションは、RemoveUIListingactionが第2スクリーンデバイスに受信されるとすぐに現在動作中の第2スクリーンアプリケーションの動作を中止しなければならない。

20

【0954】

HTTPメッセージを返還する段階(s66020)は、RemoveUIListingアクションを送信する段階(s66010)に対する結果を送る段階であってもよい。前述した応答1を送信する段階(s40100)と同様に、状況によって、HTTP/1.1 200 OKなどの応答を送ることができる。

30

【0955】

ディスプレイメッセージを送信する段階(s66030)は、第2スクリーンデバイスにディスプレイメッセージを送る段階であってもよい。RemoveUIListingアクションを送信する段階(s66010)の後に、場合によっては、実行が中止されるというメッセージを第2スクリーンデバイス画面に表示する必要がある。この場合には、DisplayMessageアクションを用いて適切なメッセージを画面に表示することができる。

【0956】

HTTPメッセージを返還する段階(s66040)は、ディスプレイメッセージを送信する段階(s66030)に対する結果を送る段階であってもよい。前述した応答1を送信する段階(s40100)と同様に、状況によって、HTTP/1.1 200 OKなどの応答を送ることができる。

40

【0957】

TDOURL情報を送信する段階(s66050)は、受信機が、新しく行われるUIのTDOURLを送信するためにAddUIListingアクションを行う段階であってもよい。新しい第2スクリーンアプリケーションが行われる時は、UIListingにTDOURLが追加されるやいなや第2スクリーンサービスアプリケーションを実行させることができる。参考として、TDOURLは、受信機で直接ダウンロードして実行し得る第2スクリーンアプリケーションに関するものであってもよく、部分的なりソ

50

ースデータだけをインターネットサーバーからダウンロードして行うものであってもよい。また、T D O U R L そのものが外部インターネットサーバーアドレスを指してもよい。外部インターネットサーバーアドレスを指す場合には、第2スクリーンアプリケーション及び実行に必要な全てのデータは、インターネットサーバーからダウンロードして実行することができる。

【0958】

H T T Pメッセージを返還する段階 (s 6 6 0 6 0) は、T D O U R L 情報を送信する段階 (s 6 6 0 5 0) に対する結果を送る段階であってもよい。前述した応答1を送信する段階 (s 4 0 1 0 0) と同様に、状況によって、H T T P / 1 . 1 2 0 0 O K などの応答を送ることができる。

10

【0959】

ディスプレイメッセージを送信する段階 (s 6 6 0 7 0) は、第2スクリーンデバイスにディスプレイメッセージを送る段階であってもよい。T D O U R L 情報を送信する段階 (s 6 6 0 5 0) 後に、新しい第2スクリーンサービスが実行されるとのメッセージを第2スクリーンデバイス画面に表示することができる。ディスプレイメッセージを送信する段階 (s 6 6 0 3 0) と同様に、D i s p l a y M e s s a g e アクションを用いて適切なメッセージは、画面に表示することができる。

【0960】

H T T Pメッセージを返還する段階 (s 6 6 0 8 0) は、ディスプレイメッセージを送信する段階 (s 6 6 0 7 0) に対する結果を送る段階であってもよい。前述した応答1を送信する段階 (s 4 0 1 0 0) と同様に、状況によって、H T T P / 1 . 1 2 0 0 O K などの応答を送ることができる。

20

【0961】

ユニキャストシグナリングの場合も同様、第2スクリーンアプリケーションのライフサイクルは、受信機又は第2スクリーンサービスアプリケーションが管理することができる。

【0962】

まず、受信機が第2スクリーンアプリケーションのライフサイクルを管理する方法について説明する。

【0963】

第2スクリーンサービスアプリケーションでは“U R L ” (T D O _ U R L) 情報だけを知っていればよい。したがって、受信機で第2スクリーンデバイスにあるR e m o t e U I クライアントサービスに“ A d d U I L i s t i n g ” アクションを行い、第2スクリーンデバイスで動作する第2スクリーンアプリケーションを、定められた時間に行うことができる。逆に、第2スクリーンアプリケーションを中断させなければならない場合には、“ R e m o v e U I L i s t i n g ” アクションを行い、定められた時間内に第2スクリーンアプリケーションを中断させることができる。

30

【0964】

さらに、細部の時間まで合わせるためには、U R L の他に、T D O を行うメディア時間情報まで与えると、第2スクリーンサービスアプリケーションを指定された時間に行うことができる。しかし、メディア時間は、現在受信機で再生されているメディアに対する相対的な時間であるから、この時間は各機器が理解し得る絶対時間に変更する必要がある。N T P (N e t w o r k T i m e P r o t o c o l) 或いは他の時間情報を有することができる。この時間に、未来の時間情報を有するメディア時間を加えると、アクションを行う時間となる。第2スクリーンデバイスにおいてアクションは実行と中止しかないため、これは、A d d U I L i s t i n g 及びR e m o v e U I L i s t i n g a c t i o n の具現によって異なり得る。

40

【0965】

次に、第2スクリーンサービスアプリケーションが第2スクリーンアプリケーションのライフサイクルを管理する方法について説明する。

50

【0966】

第2スクリーンサービスアプリケーションが第2スクリーンデバイスで動作する第2スクリーンアプリケーションを実行し中断する時間やイベントに関する情報を知っていなければならない。したがって、前述したように、第2スクリーンサービスアプリケーションがトリガークライアントを含んでおり、受信機がDTVCCで受信されたiTVメッセージ(トリガー)のURIS string情報を受信機のDeviceInfo(Device Profile)に合わせて送信することができる。

【0967】

送信方法は、2種類に大別できる。その第一の方法は、DTVCCで送信されて受信機で処理したトリガーマッセージをそのまま送信する方法であり、第二の方法は、受信機で第2スクリーンデバイスに必要な情報のみをさらに整理し、必要な情報とデータのみをXML形態で伝達する方法である。

10

【0968】

まず、受信機がトリガーを第2スクリーンにそのまま伝達する方法について説明する。

【0969】

この方法は、受信されたトリガーをそのまま“AddUIListing”アクションすることによって処理を終えることができる。RemoteUIクライアントサービスでは、この情報をもってすぐにトリガークライアントが処理できるように渡し、トリガークライアントは、受信機と同様に、TPTをダウンロードして処理することができる。ただし、受信機は、DTVCCでメディア時間トリガーが受信される度に、第2スクリーンデバイスにこのように伝達しなければならない。受信機が“appID”、“eventID”及び“event@destination”を確認し、第2スクリーンデバイスで受信しなくてもよいトリガーである場合には、自分で消費し、第2スクリーンデバイスには伝達しなくてもよい。

20

【0970】

次に、受信機で第2スクリーンデバイスに必要なトリガー情報をフィルタリングして伝達する方法について説明する。

【0971】

この方法は、受信機でTPTファイルとトリガーを処理し、第2スクリーンデバイスが処理するデータにして伝達することができる。この方法は、第2スクリーンサービスアプリケーションで、トリガークライアントが動作しなくてもよく、“ProgressInput”アクションを通じてXMLファイルを受け取って処理することができる。すなわち、第2スクリーンサービスアプリケーションが全体TPTを直接処理するのではなく、現在或いは将来に処理しなければならない必要なデータのみを受信機がフィルタリングして伝達すると、これを受けて処理することができる。この方法の長所は、“Event@Data”フィールドがあるが、このフィールドまで伝達できるという点である。

30

【0972】

図67は、RemoteUIクライアントサービスのためのユニキャストシグナリングの一実施例である。

【0973】

図67は、前述した第二の方法である、“受信機で第2スクリーンデバイスに必要なトリガー情報をフィルタリングして伝達する方法”を示している。

40

【0974】

RemoteUIクライアントサービスのためのユニキャストシグナリングの一実施例は、イベント1情報を送信する段階(s67010)、HTTPメッセージを返還する段階(s67020)、イベント2情報を送信する段階(s67030)、HTTPメッセージを返還する段階(s67040)、イベント3情報を送信する段階(s67050)、及び/又はHTTPメッセージを返還する段階(s67060)を含むことができる。

【0975】

イベント1情報を送信する段階(s67010)は、受信機が第2スクリーンデバイス

50

に、受信したイベント情報を伝達する段階であってもよい。前述した通り、受信機でTPTファイルとトリガーを処理し、第2スクリーンデバイスが処理するデータにして伝達することができる。第2スクリーンサービスアプリケーションでは、トリガークライアントが動作しなくてもよく、“ProgressInput”アクションを通じてXMLファイルを受信して処理することができる。

【0976】

HTTPメッセージを返還する段階(s67020)は、イベント1情報を送信する段階(s67010)に対する結果を送る段階であってもよい。前述した応答1を送信する段階(s40100)と同様に、状況によって、HTTP/1.1 200 OKなどの応答を送ることができる。

10

【0977】

イベント2情報を送信する段階(s67030)は、イベント1情報を送信する段階(s67010)と同様に、イベントに関する情報を伝達する段階であってもよい。TPTXMLにおける同様に、各イベント別にデータをTDOに伝達することができるが、この段階は、イベント2に関する情報を伝達する段階であってもよい。

【0978】

HTTPメッセージを返還する段階(s67040)は、イベント2情報を送信する段階(s67030)に対する結果を送る段階であってもよい。前述した応答1を送信する段階(s40100)と同様に、状況によって、HTTP/1.1 200 OKなどの応答を送ることができる。

20

【0979】

イベント3情報を送信する段階(s67050)は、同様に、イベント3に関する情報を伝達する段階であってもよい。

【0980】

HTTPメッセージを返還する段階(s67060)は、イベント3情報を送信する段階(s67050)に対する結果を送る段階であってもよい。前述した応答1を送信する段階(s40100)と同様に、状況によって、HTTP/1.1 200 OKなどの応答を送ることができる。

【0981】

前述した第二の方法では、受信機が第2スクリーンアプリケーションのライフサイクルを管理することができる。第2スクリーンアプリケーションのライフサイクルは、第2スクリーンデバイスで動作するTPT及びAMT情報をパーシングし、処理する過程を意味することができる。

30

【0982】

図68は、ProcessInputアクションで第2スクリーンデバイスに伝達される“EventInfo”情報の一実施例である。

【0983】

図68は、前述した“受信機で第2スクリーンデバイスに必要なトリガー情報をフィルタリングして伝達する方法”において、受信機がTPTファイルとトリガーを処理し、第2スクリーンデバイスが処理するデータにして伝達する時、イベント情報のXMLデータ構造の一例を示している。

40

【0984】

すなわち、受信機で第2スクリーンアプリケーションのライフサイクルを管理する場合には、ProcessInputアクションでは、イベント情報を、図68のデータ構造を有するXMLファイルとしてRemoteUIクライアントに伝達することができる。受信機がライブトリガー(Live Trigger)を受けると同様に、ライブトリガーを処理し、実行する時間とTDO URLなどの情報のみを第2スクリーンデバイスに伝達することもできる。

【0985】

図68のテーブルで、@appID、URL、Event、@eventID、@ac

50

t i o n及びD a t aは、受信機が受信したトリガー又はA M Tの情報と同一であってもよい。しかし、“@m e d i a T i m e”、“@s t a r t T i m e”及び“@e n d T i m e”は、受信機で特別に処理することができる。“@m e d i a T i m e”に合わせて、D T V C Cで送信されたトリガーの“t i m e =”シンタックスの情報（或いは、A M Tの“@b e g i n M T”情報）を処理し、再び“@s t a r t T i m e”と“@e n d T i m e”を決定することができる。受信機と第2スクリーンデバイスが、同一の絶対時間情報（w a l l c l o c k t i m e）を使用するとはいえないことから、コンテンツのメディア時間に合わせて第2スクリーンデバイスを動作させるためである。すなわち、受信機と第2スクリーンデバイス間にアクションに対する実行開始時間及び終了時間に関する時間情報が実際N T P（N e t w o r k T i m e P r o t o c o l）に合わせられているとすれば、送信する前に受信機が変換する作業を行わなければならない。

10

【0986】

送信遅延及びメディア時間推定について説明する。

【0987】

前述した通り、@s t a r t T i m eと@e n d T i m eは、受信機で第2スクリーンに送る時、現在のm e d i a T i m e値を基準に相対的に作られて送信されるため、送信時に発生する時間損失は、送信プロトコルであるH T T Pのヘッダーの“D a t e”値とH T T P応答であってもよい到着した時間値との差を用いて、第2スクリーンデバイスで調整してより正確な時間値を探ることができる。現在メディア時間と@m e d i a T i m e値との差は送信遅延時間となるため、次のような数式によって現在メディア時間を得ることができる。

20

【0988】

現在メディア時間 = 送信遅延時間（受信時の時間値 - H T T Pヘッダー“D a t e”値） + @m e d i a T i m e

これによって、現在受信機の現在メディア時間値と略同一に類推することができる。

【0989】

図69は、受信機及び第2スクリーンデバイス間の構成の一実施例である。

【0990】

受信機及び第2スクリーンデバイス間の構成が示されている。受信機が被制御デバイス（サーバー）であり、第2スクリーンデバイスが制御ポイント（クライアント）であってもよい。

30

【0991】

ディスカバリ段階で、受信機は、サービスリストをマルチキャストしてネットワークにジョインし、第2スクリーンアプリケーションはサービスリストに対する要求をマルチキャストして開始することができる。

【0992】

記述（D e s c r i p t i o n）段階で、第2スクリーンアプリケーションは、受信機のサービス記述を要求することができる。

【0993】

本発明では、U P n Pに基づいてTV受像機、すなわち、受信機を通じて受信されるA / V放送コンテンツと関連したインタラクティブサービスを第2スクリーンデバイスを通じて取得できるようにするために、新しいU P n Pデバイス及びサービスを定義することができる。

40

【0994】

図70は、サービスのサービスタイプ及びサービスIDの一実施例である。

【0995】

受信機は、トリガーサービス、双方向通信サービス及びA p p U R Lサービスを支援することができる。また、H T T Pプロキシサーバーサービスを支援することができる。サービスタイプ及びサービスIDは、図70に示している。

【0996】

50

双方向通信サービスは、第2スクリーンデバイス内のアプリケーションと双方向通信に参加するように準備されたプライマリデバイスで実行されるDOが存在するかを、第2スクリーンデバイスで決定するようにすることができる。

【0997】

トリガーサービス、AppURLサービス及びHTTPプロキシサーバーサービスについては後述する。

【0998】

図71は、トリガー伝達サービスの動作概念図の一実施例である。

【0999】

トリガー伝達サービスは、UPnPに基づいてTV受信機に受信されるA/V放送コンテンツと関連したインタラクティブサービスを第2スクリーンを通じて取得できるようにするためのサービスを意味できる。

10

【1000】

図71は、受信機がDTVCC或いはACRサーバーなどからトリガーを取得し、当該トリガーをそのまま或いは第2スクリーンデバイスに適した形態に変形(拡張されたトリガー)して第2スクリーンデバイスに送信する過程を示している。

【1001】

トリガーが変化する度に、受信機から第2スクリーンデバイスに当該トリガー或いは変形された拡張トリガーを実時間で送信することができる。

【1002】

20

図72は、拡張された活性化トリガー(expanded activation trigger)生成過程の一実施例である。

【1003】

拡張されたトリガー(拡張された活性化トリガー)は、DTVCCストリーム或いはACRサーバーなどから取得したトリガーとTPT内の情報とを組み合わせる生成することができる。トリガーに含まれたTDO IDとイベントID、データID及び活性化時間と、TPT内のTDO URL、TDO属性、イベント、アクション、ディフュージョン(Diffusion)及びデータなどの情報を組み合わせる生成することができる。TDO内の情報は、トリガー内の情報と関連したTDO及びイベントに関する情報であってもよい。

30

【1004】

ここで、拡張されたトリガーは、増加したトリガー(augmented trigger)と呼ぶこともできる。

【1005】

図73は、増加した活性化トリガー(Augmented Activation Trigger)に対するXMLスキーマ記述(Schema Description)の一実施例である。

【1006】

トリガーサービスの説明書(Specification)について説明する。

【1007】

40

トリガーサービスは、トリガーを伝達することができる。トリガーは、(a)ACRプロセス、(b)現在見ているチャンネルのDTVCCサービス#6、(c)遠隔“ライブトリガー”サーバー、又は(d)活性化メッセージテーブル(AMT)からTV受像機によって取得することができる。これは状況による。

【1008】

トリガーサービスはまた、チャンネルが変更される度に特殊“チャンネル変更”トリガーを伝達することができる。チャンネル変更トリガーについては後述する。伝達される4つのタイプのトリガーが基本的に存在し得る。すなわち、1)TDOインタラクティブサービスモデルのための時間ベーストリガー、2)TDOインタラクティブサービスモデルのための活性化トリガー、3)直接実行インタラクティブサービスモデルのためのトリガー、及

50

び4) 特殊“チャンネル変更”トリガーである。

【1009】

最大の柔軟性のために、第2スクリーンデバイスに全てのタイプのトリガーを伝達するオプションを有し、受信機でトリガーが到達するやいなや伝達することが好ましい。

【1010】

しかし、受信機が相互作用を提供するのと同じ方式で相互作用を提供するように設計された第2スクリーンアプリケーションに対して、TDO相互作用モデルのための時間ベーストリガーを省略し、その活性化時間にTDO相互作用モデルのための各活性化トリガーを伝達することが好ましい。これは、時間ベースを維持し、活性化時間を算出する必要性から、それらの第2スクリーンアプリケーションを求めることができる。これはまた、トリガーからの情報をトリガーで参照されたTDOエレメント及びそのイベントチャイルド(child)エレメントに関するTPTからの情報と結合することによってそれぞれの活性化トリガーを増加させ、これらの第2スクリーンアプリケーションがTPTを扱う必要性から求めることが好ましい。

10

【1011】

したがって、トリガースービスは、トリガー伝達のための2つのオプションを提供することができる。その一つは、受信機が(増加していない)全てのトリガーを伝達する“フィルタリングされていないストリーム(Unfiltered stream)”オプションでよい。もう一つは、TDO相互作用モデルのための増加した活性化トリガー、TDO相互作用モデル以外の相互作用モデルのための全てのトリガー及び特殊チャンネル変更トリガーのみを伝達する“フィルタリングされたストリーム(Filtered stream)”オプションでよい。

20

【1012】

TDO相互作用モデルのためのそれぞれの増加した活性化トリガーのためのターゲット伝達時間は、その活性化時間であってもよい。(フィルタリングされていないストリームオプションで増加無しで伝達された活性化トリガーを含む)全ての他のトリガーに対するターゲット伝達時間は、受信機によって受信される時間であってもよい。それぞれの特殊チャンネル変更トリガーのターゲットトリガー時間は、チャンネル変更の時間であってもよい。

【1013】

トリガースービスのトリガー伝達フォーマットは、増加した活性化トリガーに対する伝達フォーマット及び他の全てのトリガーに対する伝達フォーマットとに区別できる。

30

【1014】

図73は、増加した活性化トリガー(Augmented Activation Trigger)に対する伝達フォーマットを示す。他の全てのトリガーに対する伝達フォーマットは後述する。

【1015】

増加した活性化トリガーに対する伝達フォーマットは、@interactionModel、@appURL、及び/又は@cookieSpace属性と能力及び/又はイベントエレメントを含むことができる。

40

【1016】

イベントエレメントは、@action、@destination、@diffusion及び/又は@data attributesを含むことができる。

【1017】

各フィールドのセマンティックス(semantics)は、次の通りである。

【1018】

@interactionModel属性の値は、トリガーを伝達するために用いられるDTVCチャンネルのサービス#6内のSDOPrivateData命令内のcmdIDフィールドに対して同一のコーディングを用いてトリガーと関連した相互作用モデルのための数値コードであってもよい。本発明の一実施例で、数値コードは、トリガーを

50

伝達するために用いられるDTV CCチャンネルのサービス6内のURLString命令内のURI__data()構造内のURI__typeに対するものと同じのコーディングを用いることができる。

【1019】

@appURL属性の値は、トリガー内のイベント(“e=”)用語によって識別されるTPT TDOエレメントの第1URLチャイルドエレメントの値であってもよい。

【1020】

@cookieSpace属性は、トリガー内のイベント(“e=”)用語によって識別されるTPT TDOエレメントが@cookieSpace属性を含む度に存在でき、その属性と同じ値を有することができる。

10

【1021】

Capabilitiesエレメントは、トリガー内のイベント(“e=”)用語によって識別されるTPT TDOエレメントがCapabilitiesエレメントを含む度に存在でき、そのエレメントと同一であってもよい。

【1022】

Eventエレメントは、トリガー内のイベント(“e=”)用語によって識別されるEventエレメントを示すことができる。(厳密に言えば、トリガー内のイベント用語は、TPT内のTDOエレメント及びそのTDOエレメントのイベントチャイルドエレメントを識別する。これは、ここでトリガー内のイベント用語によって識別されたEventエレメントという。)

20

【1023】

@action属性の値は、トリガー内のイベント(“e=”)用語によって識別されるイベントエレメントのアクション属性の値と同一であってもよい。

【1024】

@destinationは、トリガー内のイベント(“e=”)用語によって識別されるイベントエレメントがあて先属性を含む度に存在でき、その属性と同じ値を有することができる。本発明の一実施例で、@destination属性は、トリガー内の“e=”用語によって識別されたTDOエレメントがあて先属性を含む度に存在することができる。

【1025】

@diffusion属性は、トリガー内のイベント(“e=”)用語によって識別されるイベントエレメントがディフュージョン属性を含む度に存在でき、その属性と同じ値を有することができる。本発明の一実施例で、@diffusion属性は、トリガー内の“e=”用語によってTDOエレメントがディフュージョン属性を含む度に存在することができる。

30

【1026】

@data属性は、イベントエレメントのデータチャイルドエレメントがトリガー内のイベント(“e=”)用語によって識別される度に存在でき、そのエレメントと同じ値を有することができる。

【1027】

前述した通り、増加したトリガーは、DTVCCストリーム或いはACRサーバーなどから取得したトリガーとTPT内の情報とを組み合わせる生成することができる。

40

【1028】

前述した通り、拡張されたトリガーは、増加したトリガーと呼ぶこともできる。

【1029】

図74は、URI__data()のシンタックスの一実施例である。

【1030】

図74は、前述したAugmentedTriggerの@interactionModel属性に関する説明のうち、URI__data()に関するものであってもよい。

【1031】

50

URI_data()構造は、URI_type及び/又は複数個のURI_characterを含むことができる。また、URI_data()構造は予約ビットを含むことができる。

【1032】

4ビット符号無し整数であってもよいURI_typeは、命令内で伝送されるURIのタイプを示すことができる。URI_typeの意味は、図75に示すとおりにすることができる。受信機は、認識されないタイプを示すURIString命令のインスタンスを無視すると期待することができる。URIが2個のセグメントで伝送されると、URI_typeフィールドは両セグメントにおいて同一であってもよい。

【1033】

URI_characterは、URIに対して許容されるものに制限される値を有する8ビットASCII文字であってもよい。URIが2個のセグメントで伝送される場合の再組立(reassembly)後に、URI_character値のシーケンスによって形成された文字ストリングが有効URIであってもよい。

【1034】

図75は、URI_typeの意味の一実施例である。

【1035】

図75は、URI_data()のシンタックスの一実施例で前述したURI_data()のURI_typeの意味に関するものであってもよい。

【1036】

URI_typeが0である場合、URI_data()がインタラクティブTVトリガーに関するものであることを意味することができる。

【1037】

URI_typeが1である場合、URI_data()がサービス使用報告サーバー(SURS)ロケータ(locator)に関するものであることを意味できる。

【1038】

URI_typeが2-15である場合、URI_data()がどのtypeを有するかはまだ定められておらず、将来の使用のために予約されていることを意味できる。

【1039】

図76は、増加した活性化トリガー(Augmented Activation Trigger)に対するXMLスキーマ記述(Schema Description)の一実施例である。

【1040】

増加していないトリガーのXMLフォーマットであってもよい。前述した特殊“チャンネル変更”トリガーもこのフォーマットに従うことができる。

【1041】

増加した活性化トリガーに対するXMLスキーマ記述の一実施例は、@interactionModel属性及び/又は@triggerString属性を含むことができる。

【1042】

各フィールドのセマンティックスは、次の通りである。

【1043】

@interactionModel属性は、特殊“チャンネル変更”トリガーのために存在することができない。他のトリガーのために、interactionModelが存在でき、その値は、DTVCCチャンネル内のSDOPrivateDataコマンド内のcmdIDフィールドに対するものと同じコーディングを用いてトリガーと関連した相互作用モデルのための数値コードであってもよい。ライブトリガーサーバーから取得されたりAMTから導出されたトリガーのための@interactionModelは、TDOモデルと見なすことができる。本発明の一実施例において、数値コードは、DTVCCチャンネルのサービス6のURLString命令内のURI_data()構造内の

10

20

30

40

50

URI__typeフィールドに対するものと同じのコーディングを用いることができる。

【1044】

@triggerString属性の値は、トリガーのSTRING表示であってもよい。トリガーのSTRING表示は、前述したトリガーシンタックスで説明したものと同一であってもよい。ただし、特殊“チャンネル変更”トリガーの場合には異なってもよい。特殊“チャンネル変更”トリガーのための@triggerString属性は、値“**<major__num>.<minor__num>”を有することができ、ここで、<major__num>は、(TVステーションによって放送されたため)新しいチャンネルの本来の主要チャンネル番号であり、<minor__num>は、(TVステーションによって放送されたため)新しいチャンネルの本来のマイナーチャンネル番号であってもよい。チャンネル番号を知っていないと、<major__num>及び<minor__num>値は“0”であってもよい。

10

【1045】

図77は、増加した活性化トリガーのフォーマットの一実施例である。

【1046】

前述の増加したトリガーのフォーマットの他の実施例であってもよい。

【1047】

@cookieSpace、Capabilities、Event、@action、@destination、@diffusion及び@dataは、前述したものと同一であってもよい。

20

【1048】

@activationTimeは、メディア時間スケール上の活性化時間であってもよい。

【1049】

@tdoURLは、前述した@appURLと同じであってもよい。

【1050】

図78は、増加した活性化トリガーのフォーマットの一実施例である。

【1051】

前述の増加したトリガーのフォーマットの更に他の実施例であってもよい。

【1052】

@activationTime、@tdoURL、@cookieSpace、Capabilities、Event、@action、@destination、@diffusion及び@dataは、前述したものと同一であってもよい。

30

【1053】

@availInternet及び@availBroadcastはTPTからなってもよい。@availInternet及び@availBroadcastも、前述したものと同一であってもよい。

【1054】

図79は、増加した活性化トリガーのフォーマットの一実施例である。

【1055】

前述の増加したトリガーのフォーマットの更に他の実施例であってもよい。

40

【1056】

@activationTime、@tdoURL、@cookieSpace、Capabilities、Event、@action、@destination、@diffusion及び@dataは、前述したものと同一であってもよい。

【1057】

ContentItemelement及び、@updatesAvail、@pollPeriod、@size、@availInternet、@availBroadcast属性は、増加したトリガーを生成する時、TPTにおける同名のelement及びattributesから由来のものであってもよい。ContentItemelement

50

lement及び、@updatesAvail、@pollPeriod、@size、@availInternet、@availBroadcastも、前述したものと同一であってもよい。

【1058】

図80は、増加した活性化トリガーのフォーマットの一実施例である。

【1059】

前述の増加したトリガーのフォーマットの更に他の実施例であってもよい。

【1060】

@cookieSpace、@availInternet、@availBroadcastCapabilities、ContentItem、@updatesAvail、@pollPeriod、@size、@availInternet、@availBroadcast、Event、@action、@destination及び@dataは、前述したものと同一であってもよい。

10

【1061】

@appId、@appType、@appName、@globalId、@appVersion、@frequencyOfUse、@expireDate、@testTDO、TDOエレメント内のURL及びContentItemエレメント内のURLは、増加したトリガーを生成する時、TPTにおける同名のエレメント及び属性から由来するものであってもよい。@appId、@appType、@appName、@globalId、@appVersion、@frequencyOfUse、@expireDate、@testTDO、TDO element内のURL及びContentItem element内のURLも、前述したものと同一であってもよい。

20

【1062】

図81は、HTTPロングポーリング基盤アプローチによるトリガーサービス状態変数の一実施例である。

【1063】

ここで、トリガーサービスを支援するいくつかのアプローチが存在できる。いくつかのアプローチとしては、HTTPロングポーリング基盤アプローチとUPnPイベントインゲメカニズム基盤アプローチを挙げることができる。

【1064】

以下、HTTPロングポーリング基盤アプローチについて説明する。

30

【1065】

HTTPロングポーリング基盤アプローチにおいて、トリガーサービスは一つの状態変数を有することができる。

【1066】

TrigServerURL状態変数の値は、プロトコル用語無しでクライアントによってトリガーを検索するために用いられるURLであってもよい。

【1067】

トリガーを検索するとの要求は、HTTP要求又はWebSocket要求を用いてなされてもよい。HTTP要求が(先頭に付け加えられたプロトコル用語http://を有するTrigServerURLを用いて)なされると、サーバーはHTTPロングポーリング伝達モードで応答することができる。WebSocket要求が(先頭に付け加えられたプロトコル用語“ws://”を有するTrigServerURLを用いて)なされると、サーバーは、WebSocket伝達モードで応答することができる。

40

【1068】

要求URLがクエリー用語を含まないか、又はクエリー用語“?level=unfilter”を含むと、サーバーは、フィルタリングされていないストリームオプションで応答することができる。要求URLがクエリー用語“?level=filter”を含むと、サーバーは、フィルタリングされたストリームオプションで応答することができる。

50

【1069】

トリガーがフィルタリングされていないストリームオプションで伝達され、トリガーがTDOモデル以外の相互作用モデルに対して伝達されると、応答のボディは、図76で表示されたXMLスキーマに従うXML文書の形態を有することができる。

【1070】

活性化トリガーがフィルタリングされたストリームオプションで伝達されると、応答のボディが、図73に表示されたXMLスキーマに従うXML文書の形態を有することができる。

【1071】

図82は、HTTPロングポーリング基盤アプローチによるトリガーサービスアクションの一実施例である。

10

【1072】

HTTPロングポーリング基盤アプローチにおいて、トリガーサービスは、単一アクション、即ちGetTriggerServerURLアクションを有することができる。GetTriggerServerURLアクションは、クライアントがTriggerServerURLの値を得るために用いることができる。

【1073】

図83は、HTTPロングポーリング基盤アプローチによるGetTriggerServerURLアクションの引数(Arguments)の一実施例である。

【1074】

TriggerServerURL出力引数は、TriggerServerURL状態変数の現在値であってもよい。

20

【1075】

この実施例で、第2スクリーンアプリケーションは、GetTriggerServerURLアクションを用いてクライアントによってトリガーを検索するために用いられるURLを得ることができる。

【1076】

図84は、トリガーサービス状態変数の実施例を示す図である。

【1077】

図84の実施例は、UPnPイベントングメカニズム基盤アプローチによるものである。

30

【1078】

以下、UPnPイベントングメカニズム基盤アプローチについて説明する。

【1079】

トリガーサービス状態変数の一実施例は、図示のトリガーサービス状態変数を定義することができる。トリガーサービスは、図84に列挙された状態変数を有することができる。

【1080】

LatestUnfilteredTrigger状態変数の値は、最も最近のターゲット伝達時間を有するフィルタリングされていないストリーム内のトリガーを示すことができる。この状態変数のフォーマットは、図76に記載されたスキーマに従うXML文書であってもよい。

40

【1081】

LatestFilteredTrigger状態変数の値は、最も最近のターゲット伝達時間を有するフィルタリングされたストリーム内のトリガーを示すことができる。活性化トリガーであれば、TPT内の情報を有する活性化トリガー内の情報を結合して、図73に記載されたテーブルで表示されたXMLスキーマに従うXML文書を生成することによって増加させることができる。TDO以外の相互作用モデルを有するトリガーであれば、図76に記載されたスキーマに従うXML文書の形態を有することができる。特殊チャンネル変更トリガーであれば、フォーマットは、上述したように“**<major_n

50

um>.<minor_num>”であってもよい。

【1082】

UnfilteredTriggerDeliveryTime状態変数の値は、最も最近のターゲット伝達時間を有するフィルタリングされていないストリーム内のトリガーの伝達時間であってもよい。“dateTime”データタイプは、1970年1月1日00:00:00以降ミリ秒の数を示す数であってもよい。

【1083】

FilteredTriggerDeliveryTime状態変数の値は、最も最近のターゲット伝達時間を有するフィルタリングされたストリーム内のトリガーの伝達時間であってもよい。

10

【1084】

図85は、トリガーサービスアクションの実施例を示す図である。

【1085】

図85の実施例は、UPnPイベントングメカニズム基盤アプローチによるものであってもよい。

【1086】

アクションは、第2スクリーンデバイス又は受信機がトリガーサービス状態変数の値を任意に判読するように定義することができる。

【1087】

UPnPイベントングメカニズム基盤アプローチにおいて、トリガーサービスアクションは、GetLatestUnfilteredTrigger及びGetLatestFilteredTriggerアクションを定義することができる。

20

【1088】

図86は、GetLatestUnfilteredTriggerアクションの引数の実施例を示す図である。

【1089】

図86の実施例は、UPnPイベントングメカニズム基盤アプローチによるものであってもよい。

【1090】

LatestUnfilteredTrigger出力引数の値は、LatestUnfilteredTrigger状態変数の値であってもよい。

30

【1091】

第2スクリーンアプリケーションは、GetLatestUnfilteredTriggerアクションを用いてLatestUnfilteredTrigger状態変数の値、すなわち、最も最近のターゲット伝達時間を有するフィルタリングされていないストリーム内のトリガーを得ることができる。

【1092】

図87は、GetLatestFilteredTriggerアクションの引数の実施例を示す図である。

【1093】

図87の実施例は、UPnPイベントングメカニズム基盤アプローチによるものであってもよい。

40

【1094】

LatestFilteredTrigger出力引数の値は、LatestFilteredTrigger状態変数の値であってもよい。

【1095】

第2スクリーンアプリケーションは、GetLatestFilteredTriggerアクションを用いてLatestFilteredTrigger状態変数の値、すなわち、最も最近のターゲット伝達時間を有するフィルタリングされたストリーム内のトリガーを得ることができる。

50

【1096】

図88は、UPnPイベントメカニズム基盤アプローチの実施例である。

【1097】

UPnPイベントメカニズム基盤アプローチの実施例は、トリガーを受信する段階(s88010)、UnfilteredTriggerDeliveryTimeを変更する段階(s88020)、非同期通知を伝達する段階(s88030)、GetLatestUnfilteredTriggerアクションを用いる段階(s88040)、増加していないトリガーで応答する段階(s88050)、活性化トリガーを受信する段階(s88060)、FilteredTriggerDeliveryTimeを変更する段階(s88070)、非同期通知を伝達する段階(s88080)、GetLatestFilteredTriggerアクションを用いる段階(s88090)及び/又は増加した活性化トリガーで応答する段階(s88100)を含むことができる。

10

【1098】

図面で、受信機は、前述した受信機であってもよい。第2スクリーンデバイス#1は、“unfiltered stream option”を支援する第2スクリーンデバイスであってもよい。第2スクリーンデバイス#2は、“フィルタリングされたストリームオプション”を支援する第2スクリーンデバイスであってもよい。

【1099】

トリガーを受信する段階(s88010)は、前述したとおり、受信機がDTVCCチャンネル、ACRプロセス又は放送社インタラクティブTV(iTV)サーバーなどを介してトリガーを受信する段階であってもよい。このとき、受信したトリガーは、増加していない全てのタイプのトリガーであってもよい。

20

【1100】

UnfilteredTriggerDeliveryTimeを変更する段階(s88020)は、前述したUnfilteredTriggerDeliveryTime状態変数の値が変更される段階であってもよい。

【1101】

非同期通知を伝達する段階(s88030)は、前述した“イベントメカニズム”プロトコルによるものであってもよい。第2スクリーンアプリケーションは、トリガーサービスの“イベントメカニズム”特徴に加入することができる。

30

【1102】

GetLatestUnfilteredTriggerアクションを用いる段階(s88040)は、第2スクリーンアプリケーションがトリガーを得るために、前述したGetLatestUnfilteredTriggerアクションを用いる段階であってもよい。

【1103】

増加していないトリガーで応答する段階(s88050)は、GetLatestUnfilteredTriggerアクションに対する応答として、増加していないトリガーを送信する段階であってもよい。第2スクリーンアプリケーションは、GetLatestUnfilteredTriggerアクションを用いて適切なトリガーを適切な時間に得ることができる。

40

【1104】

活性化トリガーを受信する段階(s88060)は、前述したとおり、受信機がDTVCCチャンネル、ACRプロセス又は放送社インタラクティブTV(iTV)サーバーなどを介して活性化トリガーを受信する段階であってもよい。

【1105】

FilteredTriggerDeliveryTimeを変更する段階(s88070)は、前述したFilteredTriggerDeliveryTime状態変数の値が変更される段階であってもよい。

【1106】

50

非同期通知を伝達する段階 (s 8 8 0 8 0) は、前述した非同期通知を伝達する段階 (s 8 8 0 3 0) と同一であってもよい。

【 1 1 0 7 】

GetLatestFilteredTriggerアクションを用いる段階 (s 8 8 0 9 0) は、第2スクリーンアプリケーションがトリガーを得るために、前述したGetLatestFilteredTriggerアクションを用いる段階であってもよい。

【 1 1 0 8 】

増加した活性化トリガーで応答する段階 (s 8 8 1 0 0) は、GetLatestFilteredTriggerアクションに対する応答として、増加した活性化トリガーを送信する段階であってもよい。第2スクリーンアプリケーションは、GetLatestFilteredTriggerアクションを用いて適切なトリガーを適切な時間に得ることができる。

10

【 1 1 0 9 】

図 8 9 は、トリガーサービス状態変数の実施例を示す図である。

【 1 1 1 0 】

前述したトリガーサービスを支援するためのいくつかのアプローチには、HTTPロングポーリング基盤アプローチ及びUPnPイベントメカニズム基盤アプローチだけでなく、他の実施例もある。以下、その実施例を、図 8 9 及び図 9 0 を参照して説明する。

【 1 1 1 1 】

この実施例において、トリガーサービス状態変数は、図 8 9 のとおりであってもよい。

20

【 1 1 1 2 】

CurrentTrigger状態変数の値は、次の3つの状況のいずれかが施行されるかに依存することができる。1)プライマリスクリンで現在視聴している番組と関連したインタラクティブ付加 (adjunct) データサービスがない。2)プライマリスクリンで現在視聴している番組と関連したインタラクティブ付加データサービスが存在し、直接実行相互作用モデルを有する。3)プライマリスクリンで現在視聴している番組と関連したインタラクティブ付加データサービスが存在し、TDO相互作用モデルを有する。

【 1 1 1 3 】

(1) の場合、CurrentTrigger状態変数の値は、ヌルストリング (null string) であってもよい。

30

【 1 1 1 4 】

(2) の場合、CurrentTrigger状態変数の値は、プライマリスクリンで現在視聴している番組に対してTV受像機によって受信された最も最近のトリガーであってもよい。

【 1 1 1 5 】

(3) の場合、CurrentTrigger状態変数の値は、プライマリスクリンで現在視聴している番組に対してTV受像機によって受信された活性化トリガーのうち最も最近に活性化された活性化トリガーの増加した形態であってもよい。(すなわち、活性化トリガーは、活性化時間に到達する時にCurrentTrigger状態変数に対する基礎となり、他の活性化トリガーがその活性化時間に到達する時まで基礎として残っていることができる。)活性化トリガーの増加した形態は、活性化トリガー内の情報をTP内の情報と結合することによって得ることができる。増加した形態は、上述した増加したトリガーフォーマットのの一つと同一であってもよい。

40

【 1 1 1 6 】

CurrentTrigger状態変数の定義は、TDO相互作用モデルに対して活性化トリガーがトリガー活性化時間にUPnPクライアントに伝達されることを意味することができる。

【 1 1 1 7 】

50

トリガーサービス状態変数の他の実施例において、トリガーが変更される度に、受信機から第2スクリーンデバイスにトリガー又は拡張されたトリガーを実時間に伝送するために、ATSCTrigger及びATSCExpandedTriggerが状態変数として定義されてもよい。

【1118】

ATSCTrigger状態変数は、DTVCCストリーム又はACRサーバーから受信されたトリガーに対する参照をURIの形態で含むことができる。この変数は、TPT(TDO parameters table)に対するURL、TDO ID、Event ID、Data ID、メディア時間、コンテンツID、ターゲットイベントに対する活性化時間などを含むことができる。

10

【1119】

ATSCExpandedTrigger状態変数は、第2スクリーンデバイスに対するTDOと関連したメタデータを、XMLフラグメントの形態で含むことができる。このメタデータは、DTVCCストリーム又はACRサーバーから受信されたTPT及びトリガーから抽出することができる。この変数は、図80の実施例と同様のXMLスキーマを有することができる。

【1120】

上記で定義したATSCTrigger及びATSCExpandedTriggerの変更された値は、受信機がUPnPのイベントングメカニズムに基づいて状態変数を変更する時に実時間に取得することができる。

20

【1121】

図90は、トリガーサービスアクションの実施例を示す図である。

【1122】

ATSCTrigger及びATSCExpandedTriggerが状態変数として定義された上述のトリガーサービス状態変数の他の実施例において、トリガーサービスアクションを説明する。

【1123】

このイベントにおいても、アクションは、第2スクリーンデバイス又は受信機が任意にATSCTrigger及びATSCExpandedTriggerの値を記録又は判読するように定義することができる。

30

【1124】

SetTrigger()アクションは、ATSCTriggerの使用を可能にすることができる。CurrentTriggerは、引数であってもよい。

【1125】

SetExpandedTrigger()アクションは、ATSCExpandedTriggerの値の利用を可能にすることができる。CurrentTriggerは増加してもよい。

【1126】

GetTrigger()アクションは、ATSCTriggerの値の判読を可能にすることができる。CurrentTriggerは引数であってもよい。

40

【1127】

GetExpandedTrigger()は、ATSCExpandedTriggerの値の判読を可能にすることができる。CurrentTriggerは引数であってもよい。

【1128】

図91は、トリガー伝達サービスを用いてトリガーを取得する時に第2スクリーン上の動作の実施例を示す図である。

【1129】

第2スクリーンデバイスがトリガー伝達サービスを通じて第2スクリーンデバイスから受信されたトリガー又は拡張された活性化トリガーに含まれるアクション情報によって動

50

作する方法を示すことができる。

【1130】

実行/終了アクションのトリガーは、トリガー伝達サービスを通じて取得することができる。

【1131】

第2スクリーンデバイスは、トリガー伝達サービスを通じて実行/終了アクションのトリガーを取得し、取得したトリガーからの関連した情報及びターゲットTDOのURLをDAE/ブラウザに伝達することができる。ブラウザは、実行又は終了などのトリガーに含まれるアクションを行うことができる。

【1132】

図92は、トリガー伝達サービスの動作概念を示す図である。

【1133】

トリガー伝達サービスを通じて第2スクリーンデバイスから受信されたトリガー又は拡張された活性化トリガー内に含まれるアクション情報によって動作する方法を示すことができる。

【1134】

第2スクリーンデバイスは、トリガー伝達サービスを通じてトリガーイベントアクションのトリガーを取得し、取得されたトリガーからData IDなどの情報を抽出することができる。その後、データは、AJAXを用いて現在実行されるTDOに伝達することができる。TDOは、受信したデータによって適切な動作を行うことができる。

【1135】

ATSC Trigger及びATSC Expanded Triggerが状態変数として定義される上述したトリガーサービス状態変数の他の実施例において、トリガー伝達サービスの動作概念を説明する。

【1136】

この実施例において、直接実行モデルの場合、コンテンツid、すなわち、“c=”がDTVCCストリーム又はACRサーバーを介して受信されたトリガーに含まれると、受信機は、受信された時間ベーストリガー値をATSC Trigger状態変数の値に設定することができる。時間ベーストリガーが受信機に到達すると、状態変数の値が直ちに変更されたり、Set Triggerアクションによって第2スクリーンデバイスに伝達されうる。

【1137】

本実施例で、TDOモデルの場合、コンテンツid、すなわち、“c=”がDTVCCストリーム又はACRサーバーを介して受信されたトリガー内に含まれないと、受信機は、活性化トリガーを受信し、TPTからの関連した情報及びトリガー情報を抽出及び結合して拡張されたトリガーを生成する。その後、拡張されたトリガーの活性化時間に（又は、その前に）、ATSC Expanded Trigger状態変数の値が設定されたり、Set Expanded Triggerアクションによって第2スクリーンデバイスに伝達されうる。

【1138】

図93は、第1アプローチによる双方向通信サービス状態変数の一実施例である。

【1139】

双方向通信サービスは、第2スクリーンデバイス内のアプリケーションとの双方向通信に参加するように準備されたプライマリデバイスで実行されるDOがあるかを、クライアントデバイスが決定するようにすることができる。

【1140】

双方向通信サービスは、クライアントデバイスが受信機で実行されるDOから双方向通信を受信するようにすることができる。これを支援するために、2個の互いに異なるアプローチが存在しうる。

【1141】

10

20

30

40

50

以下、双方向通信サービスを実現する第1アプローチを説明する。

【1142】

まず、クライアントデバイスが登録すると、(一般に、クライアントデバイス上で) Web SocketサーバーのIPアドレス及びTCPポートを提供することができる。DOがクライアントデバイスと通信することを希望すると、登録時に提供されたアドレス及びポートを用いてクライアントデバイスへのWeb Socket接続を確立することができる。

【1143】

第1アプローチにおいて、双方向通信サービスは、図93に示す状態変数を有することができる。

10

【1144】

ClientList状態変数は、双方向通信のために登録されたクライアントのリストであってもよい。また、Name、IPAddress及びPortは、各クライアントに対するネーム、IPアドレス及びTCPポートを意味することができる。

【1145】

図94は、第1アプローチによるClientList状態変数のXMLスキーマテーブルの一実施例である。

【1146】

ClientListは、図94のテーブルによって記載されたXMLスキーマに従うXML文書の形態であってもよい。

20

【1147】

Name状態変数は、双方向通信のために登録されたクライアントのネームであってもよい。

【1148】

IPAddress状態変数は、双方向通信のために登録されたクライアントに対するWeb SocketサーバーのIPアドレスであってもよい。

【1149】

TCPPort状態変数は、双方向通信のために登録されたクライアントに対するWeb SocketサーバーのTCPポートであってもよい。

【1150】

図95は、第1アプローチによるトリガーサービスアクションの一実施例である。

30

【1151】

第1アプローチにおいて、双方向通信サービスは、図95のテーブルによって記載されたように、2個のアクションを有することができる。

【1152】

SignUpアクションは、クライアントが双方向通信に参加することに関心があることを示し、用いられるネーム及びIPアドレス及びポートを提供して接続を確立するために用いることができる。

【1153】

SignOffアクションは、クライアントが双方向通信に参加することにもう関心がないことを示すために用いることができる。

40

【1154】

図96は、第1アプローチによるSignUpアクションの引数の一実施例である。

【1155】

Name入力引数は、登録されたクライアントと関連したネームであってもよい。これは、装置のホストネームであってもよい。これは、既に登録された任意の他のクライアントのネームと別個であってもよい。

【1156】

IPAddress入力引数は、接続を確立する時にDOによって用いられるWeb SocketサーバーのIPアドレスであってもよい。

50

【1157】

T C P P o r t 入力引数は、接続を確立する時に D O によって用いられる W e b S o c k e t サーバーの T C P ポートであってもよい。

【1158】

図97は、第1アプローチによる S i g n O f f アクションの引数の一実施例である。

【1159】

N a m e 入力引数は、双方向通信サービスのために登録される時にクライアントが提供したネームであってもよい。その目的は、現在送信終了 (s i g n i n g o f f) するクライアントを識別することである。

【1160】

図98は、第2アプローチによる双方向通信サービス状態変数の一実施例である。

【1161】

以下、双方向通信サービスを実現する第2アプローチを説明する。

【1162】

プライマリデバイスで実行されるこのような D O が存在すると、第2スクリーンデバイス内のアプリケーションは、第2スクリーンデバイス内のアプリケーションが双方向通信に参加する準備ができていることを示しながら、T C P / I P 接続を要求することができる。接続の第2スクリーンデバイス側上の T C P / I P アドレス及びポートが第2スクリーンデバイスの固有識別子としての機能を持つことができる。バイトは、T C P / I P 接続を通じて第2スクリーンデバイス及びプライマリデバイス内の D O 間で自由に伝送される。

【1163】

双方向通信サービスは、図98に示す2個の状態変数を有することができる。

【1164】

C o n n e c t i o n A d d r e s s 状態変数は、< a d d r e s s > : < p o r t > のフォーマットで、第2スクリーンデバイスからの接続要求が向く I P アドレス及び T C P ポートを含むことができる。

【1165】

S t a t u s 状態変数は、第2スクリーンデバイスとの双方向通信に参加するように準備された実行 D O が存在するかを示し、ここで、“真”は、実行されるこのような D O が存在するということを意味し、“偽”は、このような D O が実行されないということを意味する。

【1166】

本発明の一実施例において、S t a t u s 状態変数のデータタイプは、“ストリング”であってもよい。この実施例で、S t a t u s 状態変数の可能な値は“はい”及び“いいえ”であってもよい。

【1167】

第2アプローチにおいて、双方向通信サービスは任意のアクションを有しなくてもよい。

【1168】

図99は第2アプローチによる o n B y t e s R e c e i v e d 関数の一実施例である。

【1169】

以下、メッセージを伝送及び受信するためにプライマリデバイス内の D O によって用いられる A P I を説明する。

【1170】

次の A P I は、双方向通信サービスを用いてプライマリデバイスで実行される D O が第2スクリーンデバイスで実行されるアプリケーションとの双方向通信に参加することができる。

【1171】

10

20

30

40

50

TV受信機は、DOが終了する度に双方向通信サービスのためにUPnP “ Status ” 状態変数を “ 偽 ” に設定し、通信する前に新しいDOが事前にその状態変数を “ 真 ” に設定することができる。

【 1 1 7 2 】

前述した “ Status ” 状態変数のデータタイプがストリングである実施例の場合、TV受信機は、DOが実行を始める度にUPnP状態変数 “ Status ” を “ いいえ ” に設定し、通信前にDOが事前にその状態変数を “ はい ” に設定しなければならない。

【 1 1 7 3 】

新しい特性である onBytesReceived 関数は、NetworkInterface クラスに付加することができる。

10

【 1 1 7 4 】

コールバック関数である onBytesReceived 関数は、双方向通信サービスを通じてDOに対するバイトが受信される時に呼び出すことができる。2個の引数、すなわち、 “ address ” 及び “ bytes ” を定義することができる。

【 1 1 7 5 】

“ address ” は、 < address > : < port > のフォーマットで受信されたバイトの送信者のIPアドレス及びUDPポートを含むストリングであってもよい。

【 1 1 7 6 】

“ bytes ” は、受信されたバイトを示すストリングであってもよい。実施例によって、UDP及び/又はIPヘッダーを除くものであってもよい。

20

【 1 1 7 7 】

図100は、第2アプローチによる setStatusYes () の一実施例である。

【 1 1 7 8 】

新しい方法、すなわち、 setStatusYes () を NetworkInterface クラスに追加することができる。

【 1 1 7 9 】

setStatusYes () は、双方向通信サービスのUpnPブール (Boolean) 状態変数 “ Status ” の値を “ 真 ” に設定し、DOが通信に参加する準備ができていないことを示すことができる。

【 1 1 8 0 】

setStatusYes () は、任意の引数を有しなくてもよい。

30

【 1 1 8 1 】

図101は、第2アプローチによる setStatusNo () の一実施例である。

【 1 1 8 2 】

新しい方法、すなわち、 setStatusNo () を NetworkInterface クラスに追加することができる。

【 1 1 8 3 】

setStatusNo () は、双方向通信サービスのUpnPブール状態変数 “ Status ” の値を “ 偽 ” に設定し、DOが通信に参加する準備ができていないということを示すことができる。

40

【 1 1 8 4 】

setStatusNo () は、任意の引数を有しなくてもよい。

【 1 1 8 5 】

図102は、第2アプローチによる setStatus () の一実施例である。

【 1 1 8 6 】

前述した “ Status ” 状態変数のデータタイプがストリングである実施例の場合、 setStatus () は NetworkInterface クラスに追加されてもよい。

【 1 1 8 7 】

setStatus () は、双方向通信サービスのUpnP状態変数 “ Status ”

50

の値を設定することができる。

【1188】

`setStatus()` は状態引数を有することができる。

【1189】

状態引数は、DOが第2スクリーンデバイスとの双方向通信に参加できると“はい”に設定し、そうでないと“いいえ”に設定することができる。

【1190】

図103は、第2アプローチによる`sendBytes()`の一実施例である。

【1191】

新しい方法、すなわち、`sendBytes()`を`NetworkInterface`クラスに追加することができる。 10

【1192】

`sendBytes()` は双方向通信サービスを用いてバイトを伝送することができる。

【1193】

`sendBytes()` は、2個の引数、すなわち、アドレス及びバイトを有することができる。

【1194】

“`address`”引数は、`<address> : <port>`のフォーマットでバイトに対するあて先TCP/IPアドレス及びポートを意味することができる。 20

【1195】

“`bytes`”引数は、伝送されるバイトを意味することができる。

【1196】

図104は、`AppURL`サービス状態変数の一実施例である。

【1197】

`AppURL`サービスは、第2スクリーンデバイスが現在実行中のDOと関連した第2スクリーンアプリケーションのネーム及びベースURLを決定するようにすることができる。

【1198】

`UPnPAppURL`サービスは、2個の状態変数、すなわち、`AppURL`及び`AppName`を有することができる。 30

【1199】

`AppURL`状態変数の値は、現在実行中のDOと関連した第2スクリーンアプリケーションのベースURLであってもよい。第1スクリーンデバイス上に実行される関連した第2スクリーンアプリケーションを有するDOがないと、`AppURL`状態変数の値はヌルストリングであってもよい。

【1200】

`AppName`状態変数の値は、現在実行中のDOと関連した第2スクリーンアプリケーションのネームであってもよい。第1スクリーンデバイス上で実行される関連した第2スクリーンアプリケーションを有するDOがないと、`AppName`状態変数の値はヌルストリングであってもよい。 40

【1201】

図105は、`AppURL`サービスアクションの一実施例である。

【1202】

`AppURL`サービスは、一つのアクション、すなわち、`GetAppURL`を有することができる。

【1203】

図106は、`GetAppURL`アクションの引数の一実施例である。

【1204】

`GetAppURL`アクションは、2個の引数、すなわち、`AppURL`及び`AppN` 50

ameを有することができる。

【1205】

AppURL出力引数は、AppURL状態変数の現在値であってもよい。AppName出力引数は、AppName状態変数の現在値であってもよい。

【1206】

したがって、GetAppURLアクションを通じて、AppURL状態変数の現在値及びAppName状態変数の現在値を得ることができる。

【1207】

図107は、プロキシHTTPサーバーサービス(Proxy HTTP Service)の動作概念図の一実施例である。

10

【1208】

受信機は、プロキシHTTPサーバーサービスを支援することができる。プロキシHTTPサーバーサービスは、第2スクリーンデバイスで必要とするTDO/ファイルが放送網を通じて伝送される場合、これをTV受信機を通じて取得できるようにするサービスを意味することができる。

【1209】

プロキシHTTPサーバーサービスの動作概念図は、放送システム107010、ACRサーバ(107020)、放送社ATSC2.0 iTVサーバ(107030)、ATSC2.0受信(107040)、及び/又は第2スクリーンデバイス107050を含むことができる。

【1210】

放送システム107010は、放送システム42010と同一であってもよい。

20

【1211】

ACRサーバ107020は、ACRサーバ42020と同一であってもよい。

【1212】

放送社ATSC2.0 iTVサーバ107030は、放送社ATSC2.0 iTVサーバ42030と同一であってもよい。

【1213】

ATSC2.0受信機107040は、放送A/V及びインタラクティブサービスと関連したトリガーなどを受信し、これを用いてインタラクティブサービスを取得及び画面上に提供することができる。前述した受信機と同一であってもよい。プロキシHTTPサーバーサービスは、ATSC2.0受信機107040がプロキシサーバと同様の役割を持つようにするサービスであってもよい。第2スクリーンデバイスで要求したファイルを第2スクリーンデバイスに効率的に提供できるようにするためのものであってもよい。

30

【1214】

第2スクリーンデバイス107050は、第2スクリーンデバイス42050と同一であってもよい。

【1215】

プロキシHTTPサーバーサービスは、受信機が放送ストリーム或いはインターネットに存在するファイルに第2スクリーンデバイスを通じて接近できるようにすることによって、第2スクリーンデバイスが放送ストリームを通じて伝送されるコンテンツにも接近できるようにする。そして、複数の第2スクリーンデバイスがインターネットに存在する同一ファイルを接近する場合、各第2スクリーンデバイスが個別的に同じファイルに接近する重複性を最小化することができる。

40

【1216】

図108は、プロキシサーバーサービス状態変数(Proxy Server Service State Variable)の一実施例である。

【1217】

UPnPプロキシサーバーサービスは、HTTPプロキシサーバを提供し、第2スクリーンデバイスがFLUTEセッションを通じて放送でTV受信機に伝達されるファイルにアクセスできるようにすることができ、家庭内の複数の第2スクリーンデバイスが同時に

50

同一ファイルを検索するとき、第2スクリーンデバイスによってインターネットサーバーからより効率的にファイルを検索できるようにする。

【1218】

これを可能にさせるために、状態変数及びアクションを定義することができる。

【1219】

UPnP HTTPプロキシサーバーサービスは、単一状態変数、すなわち、ProxyServerURLを有することができる。

【1220】

ProxyServerURL状態変数の値は、HTTPプロキシサーバーのURL、すなわち、プロキシサーバーを通じて要求をルーティング(route)するためにHTTP要求が向かうURLであってもよい。

10

【1221】

図109は、プロキシサーバーサービスアクション(ProxyServerServiceAction)の一実施例である。

【1222】

UPnPプロキシサーバーサービスは、単一アクション、すなわち、GetProxyURLを有することができる。

【1223】

図110は、GetProxyURLアクションの引数の一実施例である。

【1224】

GetProxyURLアクションは、単一引数、すなわち、ProxyURLを有することができる。

20

【1225】

ProxyURL出力引数は、ProxyServerURL状態変数の現在値であってもよい。

【1226】

したがって、GetProxyURLアクションによって、ProxyServerURL状態変数の現在値を得ることができる。

【1227】

図111は、RequestFiles()の一実施例である。

30

【1228】

UPnP HTTPプロキシサーバーサービス状態変数の他の実施例は、ATSCProxyServerURLというUPnP HTTPプロキシサーバーサービス状態変数を定義することができる。また、この実施例で、GetProxyServerURL()というアクションを定義することができる。

【1229】

ATSCProxyServerURL状態変数は、URIの形態で受信機内のプロキシサーバーに対する参照を含むことができる。プロキシサーバーは、第2スクリーンデバイスからファイルに対するHTTP要求を受け、インターネット又は放送ストリーム内のFLUTEセッションからファイルを検索する。その後、プロキシサーバーは、HTTP応答として検索されたファイルを第2スクリーンデバイスに送信する。

40

【1230】

GetProxyServerURL()は、ProxyServerURLの値を読み込むようにするアクションであってもよい。ProxyServerURLを引数として有することができる。

【1231】

UPnP HTTPプロキシサーバーサービス状態変数の更に他の実施例は、さらに、ATSCFileListというUPnP HTTPプロキシサーバーサービス状態変数を定義することができる。また、この実施例で、RequestFiles()というアクションを定義することができる。

50

【1232】

放送ストリームに存在するファイルは、放送コンテンツの受信が可能な受信機でのみ取得することができる。したがって、放送コンテンツを受信できない第2スクリーンデバイス上で放送コンテンツに含まれたファイルに接近可能にするために必要なUPnP状態変数及びアクションを定義することができる。すなわち、FLUTEセッションを通じてダウンロードしたいファイルリストであるATSCFileListをUPnP状態変数として定めたり、第2スクリーンデバイスが受信機に該当のファイルを要求するRequestFiles()アクションを通じて特定ファイル或いは複数のファイルを取得できるようにすることができる。

【1233】

ATSCFileList状態変数は、受信機内にプロキシサーバーへの要求されたファイルのCSVリストを含むことができる。

【1234】

RequestFiles()は、放送ストリーム或いはインターネット上に存在する特定ファイルをダウンロードするように要求するアクションであってもよい。特に、放送ストリームに存在するファイルの場合、要求したファイルをFLUTEセッションを通じてダウンロードするようにすることができる。

【1235】

図112は、第2スクリーンデバイスアーキテクチャの一実施例である。

【1236】

動作の理論について説明する。

【1237】

2つのモードの動作が存在することができ、その一つの動作モードでは、トリガされたアプリケーション(TDO)がTV受信機上で実行され、他の動作モードでは、ノン-トリガされたアプリケーション(パッケージアプリケーション)がTV受信機上で実行される。

【1238】

TV受信機上で実行されるトリガされたアプリケーションの場合、TV受信機上に現在表示されている番組が、第2スクリーン支援を有する関連したインタラクティブデータサービスを有すると、第2スクリーンデバイスのユーザは、デバイス上の適切なアプリケーションを活性化させることができる。このアプリケーションは、UPnPディスカバリ及び記述プロセスを経てTV受信機上のトリガーサービス、双方向通信サービス及びプロキシサーバーサービスを発見することができる。

【1239】

すると、第2スクリーンアプリケーションは、トリガーサービスに対するUPnP“イベントリング”を申し込み、伝達のために準備されたトリガーの通知を得、GetLatestUnfilteredTrigger又はGetLatestFilteredTriggerアクションを用いて使用するよう設計されたトリガーを取得することができる。この結果、第2スクリーンアプリケーションが適切な時間に適切なトリガーを得ることができる。すると、アプリケーションは、いずれの方式で用いられるよう設計されても、それらのトリガーに対して動作することができる。

【1240】

第2スクリーンアプリケーションはまた、双方向通信サービスに対するUPnP“イベントリング”を申し込み、双方向通信のための接続を要求するために用いられるTCP/IPアドレス及びポートの通知、及び通信する準備がされるプライマリデバイスで実行されるDOが存在する時の通知を得ることができる。すると、アプリケーションは、このような通信を支援する任意のDOとの双方向通信に参加することができる。

【1241】

トリガー及び/又は双方向通信によるアクションはたまたま、第2スクリーンアプリケーションがファイルをダウンロードするように要求することができる。それらのファイル

10

20

30

40

50

は、インターネット上のHTTPサーバーから利用可能であるか、又はTV放送内のセッションから利用可能でありうる（又は、両者から利用可能であってもよい）。

【1242】

所望の全てのファイルがインターネットを介して利用可能であり、第2スクリーンデバイスがネットワークへの良好な接続性を有すると、アプリケーションは簡単にファイルを直接検索することができる。

【1243】

所望のファイルの一部又は全てが、TV放送のみを通じて利用可能であり、受信機がHTTPプロキシサーバーサービスを提供すると、アプリケーションは、GetProxyURLアクションでプロキシサーバーURLを得、プロキシサーバーを通じて所望のファイルを検索することができる。アプリケーションはまた、直接的な方式よりは、かえってそのようにファイルを検索することがより便利な他の状況ではプロキシサーバーを用いるように選択してもよい。

10

【1244】

TV受信機上で実行されるノン-トリガーアプリケーションの場合、現在表示されている番組と関係なく、ユーザは、その中でも、AppURLサービスを通じて第2スクリーンデバイス上で開始される同伴者アプリケーションのネーム及び位置を利用可能にするTV受信機上のDOを活性化させることができる。

【1245】

第2スクリーンデバイス上の制御ポイントは、AppURLサービスと関連したUPnP“イベントング”を申し込み、AppURL及びAppName属性に対する変更の通知を得ることができる。すると、制御ポイントは、利用可能なサービスが係留(pending)中であるということを示すことを第2スクリーンデバイスのユーザに知らせる。次に、ユーザは、AppNameを見てサービスを選択し、第2スクリーンデバイス上で同伴者第2スクリーンアプリケーションを開始させる。

20

【1246】

第2スクリーンアプリケーションは、そのDOが受信機に以前にダウンロードされた放送社提供パッケージアプリケーションである時にも、ATSC2.0受信機上で実行されるDOと関連付けられてもよく、ここで、放送社提供パッケージアプリケーションのライフサイクルは、放送社の代わりに消費者によって制御される。同伴者第2スクリーンアプリケーションを識別するトリガーの不在時に、受信機は、第2スクリーンデバイス上の制御ポイントがGetAppURLアクションを用いて、公開された第2スクリーンアプリケーションURLにアクセスして第2スクリーンデバイス上で開始するようにするAppURLサービスを提供する。

30

【1247】

第2スクリーンデバイスアーキテクチャの一実施例は、受信機と相互作用する第2スクリーンアプリケーションと第2スクリーンデバイスを示している。

【1248】

第2スクリーンデバイスアーキテクチャの一実施例は、遠隔コンテンツサーバー112010、TV受信機112020及び/又は第2スクリーンデバイス112030を含むことができる。

40

【1249】

遠隔コンテンツサーバー112010は、コンテンツを提供できるサーバーであってもよい。受信機112020又は第2スクリーンデバイス112030にコンテンツを提供することができる。

【1250】

TV受信機112020は、UPnPトリガーサービス及びUPnPプロキシサーバーサービスを提供することができる。前述した受信機と同一であってもよい。

【1251】

第2スクリーンデバイス112030は、第2スクリーンアプリケーション(ATSC2

50

. 0 App)を有することができ、第2スクリーンアプリケーションは、トリガーハンドラー内にUPnP制御ポイント(CP)モジュールを有することができる。UPnP制御ポイント(CP)モジュールは、第2スクリーンアプリケーション(ATSC2.0 App)及びTV受信機112020上のトリガーサービス及びプロキシサーバーサービス間の全てのUPnP通信をハンドリングする。UPnP制御ポイント(CP)モジュールは、探索プロセスを管理し、プロキシサーバーURLを得、トリガーサービスイベントを申し込むことができる。

【1252】

UPnPトリガーサービスからのトリガーはトリガーハンドラーに伝達されうる。新しいDOがDAE(すなわち、ブラウザ)でダウンロードされて実行されるということをトリガーが示すと、トリガーハンドラーはそれを処理する。ブラウザで既に実行されたDOが任意のアクションを取らなければならないとトリガーが指示すると、トリガーハンドラーはトリガーをDOに伝達することができる。第2スクリーンデバイスのユーザは、DOと相互作用をすることができる。トリガーハンドラーはユーザに見えなくてもよい。

10

【1253】

第2スクリーンアプリケーションは、必要によって次の機能を有することができる。1)UPnPディスカバリ及び記述プロトコルを用いて、TV受信機上でトリガーサービス、及び、利用可能であれば、プロキシサーバーサービスを探す。2)UPnPSUBSCRIBEプロトコルを用いてトリガーサービスイベントを申し込む。3)プロキシサーバーサービスのGetProxyURLアクションを実行(invoked)して(利用可能であれば)HTTPプロキシサーバーのURLを得る。4)トリガーサービスイベントを通じて伝達されたトリガーを受信する。5)直接実行トリガーを受信され、トリガー内で識別されたDOがまだDAEで実行されていないと、(必要なら、まずダウンロードし)DAEで実行し始める。6)(必要なら、DOをダウンロードして開始した後)各受信された直接実行トリガーを、トリガーで識別されたDOに伝達する。7)TDOトリガーを受信され、トリガーのアクション属性が“TriggerEvent”以外の任意のものであれば、アクションを実行する(例えば、TDO実行を準備、TDO実行、TDO終了など)。8)TDOトリガーを受信され、トリガーのアクション属性が“TriggerEvent”であれば、トリガーを、トリガーのターゲットとして識別されたTDOに伝達する。TDOがDAEでまだ実行されていないと、トリガーを廃棄する。9)必要によって、インターネットに直接接続してTV受信機上のプロキシサーバーサービスを通じて(TDOを含む)ファイルをダウンロードする。

20

30

【1254】

直接実行トリガー及びTDOトリガーは、“スプレッド”又は“ディフュージョン”属性を含まないと、受信するとすぐに動作することができる。トリガーが“スプレッド”又は“ディフュージョン”属性を含むと、アクションはランダム間隔だけ遅延されうる。

【1255】

図113は、受信機の簡略化した構造図の一実施例である。

【1256】

受信機の簡略化した構造図は、アンテナ(rcvr2010)、チューナー(rcvr2020)、EPG(rcvr2030)、VSB又はDVB復調器(rcvr2040)、チャンネルマネジャー(rcvr2050)、SIモジュール(rcvr2051)、MPEG-2TSシステムデコーダ(rcvr2060)、キャプションモジュール(rcvr2070)、トリガーモジュール(rcvr2080)、ウェブブラウザ(rcvr2090)、UPnPフレームワーク(rcvr2091)、ネットワークプロトコルスタック(rcvr2100)、ネットワークインターフェース(rcvr2110)、UIモジュール(rcvr2120)、オーディオデコーダ(rcvr2140)、ビデオデコーダ(rcvr2150)、スピーカー(rcvr2160)、ディスプレイモジュール(rcvr2170)、グラフィックプロセッサ(rcvr2180)、遠隔制御器受信器(rcvr2190)及び/又は遠隔制御器(rcvr2200)を含むことが

40

50

できる。

【1257】

アンテナ (rcvr2010) は、放送ストリームによって放送信号を受信することができる。ここで、アンテナ (rcvr2010) は、技術の分野で一般に用いられるものであってもよい。

【1258】

チューナー (rcvr2020) は、受信機のチャンネルを探したり同調することができる、無線周波数増幅器、ローカルオシレータ、周波数変換及び入力回路、探知器 (seeker) などを含むことができる。チューナー (rcvr2020) は、技術の分野で一般に用いられるものであってもよい。

10

【1259】

EPG (rcvr2030) は、TV放送の空いている周波数帯や付加チャンネルを用いてTV番組の放送時間と内容、出演者情報などを送る放送プログラム案内サービスであってもよい (Electronic program guide)。受信されたEPGデータは記憶しておき、視聴者がリモコンでこのEPGを操作してプログラムを選択して予約し、ペア・パー・ビュー方式 (pay per view) プログラム注文、主題又は種別プログラム検索、ビデオ録画などを可能にする。受信したEPGは、UIモジュールにも伝達されてもよい。

【1260】

VSB又はDVB復調器 (rcvr2040) は、VSB信号又はDVB信号を復調することができる。VSB又はDVB復調器 (rcvr2040) は、変調されたVSB又はDVB (e.g., OFDM - 変調信号) を本来の信号に回復することができる。VSB又はDVB復調器 (rcvr2040) の復調プロセスは、技術の分野で一般に用いられるものであってもよい。

20

【1261】

チャンネルマネジャー (rcvr2050) は、受信されたEPGを用いて、チャンネル管理を行うことができる。SIモジュールで処理されたEPGを伝達することができる。一般のチャンネルマネジャー役割を担うことができる。

【1262】

SIモジュール (rcvr2051) は、MPEG-2TSシステムデコーダ (rcvr2060) を通じて放送ストリームが入ってくると、プログラム構成情報 (Program Specific Information) を把握することができる。このように把握された情報から、この放送プログラムにいくつのキャプションがあり、トリガーがサービス#6にあるか否かなどがわかる。

30

【1263】

MPEG-2TSシステムデコーダ (rcvr2060) は、復調された信号の送信ストリーム (TS) をデコードすることができる。MPEG-2TSシステムデコーダ (rcvr2060) は、送信ストリームからキャプションストリームを得てキャプションモジュール (rcvr207) に伝達することができる。MPEG-2TSシステムデコーダ (rcvr2060) は、デコードされたオーディオ及びビデオ信号をオーディオ/ビデオデコーダに送信することができる。

40

【1264】

キャプションモジュール (rcvr2070) は、キャプションストリームを受信することができる。キャプションモジュール (rcvr2070) は、サービス#6又は他のサービスをモニタし、サービス#6又はトリガーを送信するトリガーが選択されてトリガーモジュール (rcvr2080) に伝送されるか、又はキャプションテキストが処理されてスクリーン上にディスプレイされるかを決定することができる。トリガーデータは、トリガーモジュール (rcvr2080) に伝達することができる。他のキャプションサービスは、キャプションテキスト処理してグラフィックプロセッサ (rcvr2180) に伝送することができる。キャプションモジュール (rcvr2070) は、前述したS

50

Iモジュール (rcvr2051) が把握した情報に基づいて、現在受信されているデジタルキャプション中にトリガーがある場合にはトリガーモジュール (rcvr2080) に伝達する。

【1265】

トリガーモジュール (rcvr2080) は、トリガー、TPT及び/又はAMT情報をパーシングし、パーシングされたデータを処理することができる。トリガーモジュール (rcvr2080) は、トリガーのURI情報値を用いて、ネットワークプロトコルスタック (rcvr2100) を通じてネットワークにアクセスすることができる。URI情報値は、HTTPサーバーのアドレスであってもよい。トリガーモジュール (rcvr2080) は、TPTファイルコンテンツを分析してTDO URLを得ることができる。また、トリガーモジュール (rcvr2080) は、AMTをパーシングしてデータを処理することができる。他の情報は、パーシングを通じて得ることができる。AMTメッセージが受信された後、ウェブブラウザに対応するTDO URLは、所定の時間及び動作によって伝達されたり、現在動作中のTDOが所定の時間に停止することができる。これは、TDOアクションに対応し、トリガーモジュール (rcvr2080) は動作するウェブブラウザにコマンドを送信することができる。本発明に係るトリガーモジュール (rcvr2080) の動作は、以下に詳述する。トリガーモジュール (rcvr2080) は、直間接的にネットワークプロトコルスタック (rcvr2100) を通じてインターネットにおけるサーバーに接続することができる。トリガーモジュール (rcvr2080) は直ちに、DTVCで受信されたトリガーを分析し、必要な場合、ネットワークインターフェース (rcvr2110) を介して、TPTファイルをインターネットからダウンロードして処理することができる。TPTファイルをダウンロードするとすぐに処理して必要な動作を取ることができる。このとき、もし、第2スクリーンに関連したトリガーであり、関連したイベントがあると、前述したように、様々な方式で第2スクリーンデバイスと通信することができる。このような通信は、後述するUPnPフレームワーク (rcvr2091) で行うことができる。

【1266】

ウェブブラウザ (rcvr2090) は、トリガーモジュール (rcvr2080) からコマンドを受信し、UIモジュール (rcvr2120) からブラウザキーコードを受信し、遠隔制御器受信器 (rcvr2190) からブラウザキーコードを受信し、ネットワークプロトコルスタック (rcvr2100) と通信することができる。

【1267】

UPnPフレームワーク (rcvr2091) は、第2スクリーンデバイスを発見してトリガーを送信したり、或いは第2スクリーンデバイスに必要な情報を再生成して送信することができる。前述したように、ネットワークインターフェース (rcvr2110) を通じてトリガーを受けたとき、第2スクリーンに関連したトリガーであり、関連したイベントがあると、UPnPフレームワーク (rcvr2091) は、第2スクリーンデバイスと通信することができる。

【1268】

ネットワークプロトコルスタック (rcvr2100) は、トリガーモジュール (rcvr2080) 及びウェブブラウザと通信し、ネットワークインターフェース (rcvr2110) を介してサーバーにアクセスすることができる。

【1269】

ネットワークインターフェース (rcvr2110) は、いくつかの他の装置の共同接続又はネットワークコンピュータ及び外部ネットワークの接続を行う。ネットワークインターフェースはサーバーに接続してTDO、TPT、AMTなどをダウンロードすることができる。ここで、ネットワークインターフェース (rcvr2110) の動作は、技術の分野で一般に用いられるネットワークインターフェース (rcvr2110) の動作であってもよい。本発明に係るネットワークインターフェース (rcvr1090) の動作は、以下に詳述する。

【1270】

UIモジュール（rcvr2120）は遠隔制御器受信器（rcvr2190）から、遠隔制御器（rcvr2200）を用いて視聴者によって入力された情報を受信することができる。受信された情報がネットワークを用いたサービスと関連するものであれば、ブラウザキーコードをウェブブラウザに伝達することができる。受信された情報が現在ディスプレイされるビデオと関連するものであれば、信号を、グラフィックプロセッサ（rcvr2180）を通じてディスプレイモジュール（rcvr2170）に伝達することができる。

【1271】

オーディオデコーダ（rcvr2140）は、MPEG-2TSシステムデコーダ（rcvr2060）から受信されたオーディオ信号をデコードすることができる。その後、デコードされたオーディオ信号は、スピーカーに伝送し、視聴者に出力することができる。ここで、オーディオデコーダ（rcvr2140）は、技術の分野で一般に用いられるものであってもよい。

10

【1272】

ビデオデコーダ（rcvr2150）は、MPEG-2TSシステムデコーダ（rcvr2060）から受信されたビデオ信号をデコードすることができる。その後、デコードされたビデオ信号は、ディスプレイモジュール（rcvr2170）に伝送し、視聴者に出力することができる。ここで、ビデオデコーダ（rcvr2150）は、技術の分野で一般に用いられるものであってもよい。

20

【1273】

スピーカー（rcvr2160）は、オーディオ信号を視聴者に出力することができる。スピーカーは、技術の分野で一般に用いられるものであってもよい。

【1274】

ディスプレイモジュール（rcvr2170）は、ビデオ信号を視聴者に出力することができる。ディスプレイモジュール（rcvr2170）は、技術の分野で一般に用いられるものであってもよい。

【1275】

グラフィックプロセッサ（rcvr2180）は、キャプションモジュール（rcvr2070）から受信されたキャプションテキスト、及びUIモジュール（rcvr2120）から受信された視聴者入力情報に対してグラフィック処理を行うことができる。処理された情報は、ディスプレイモジュール（rcvr2170）に伝達することができる。グラフィックプロセッサ（rcvr2180）は、技術の分野で一般に用いられるものであってもよい。

30

【1276】

遠隔制御器受信器（rcvr2190）は、遠隔制御器（rcvr2200）から情報を受信することができる。この時、キーコードはUIモジュール（rcvr2120）に伝達し、ブラウザキーコードはウェブブラウザに伝達することができる。

【1277】

遠隔制御器（rcvr2200）は、視聴者によって入力された信号を遠隔制御器受信器（rcvr2190）に伝達する。遠隔制御器（rcvr2200）は、仮想チャンネルを変更する視聴者入力を受信することができる。また、遠隔制御器は、アプリケーションに対して視聴者によって選択された情報を受信することができる。遠隔制御器（rcvr2200）は、受信された情報を遠隔制御器受信器（rcvr2190）に伝達することができる。このとき、情報は、所定範囲外の距離で赤外線（IR）光を用いて遠隔で伝達することができる。

40

【1278】

図114は、第2スクリーンデバイスの構造図の一実施例である。

【1279】

第2スクリーンデバイスの構造図の一実施例は、IOサブシステム114010、ファイ

50

ルシステ(11402)、ネットワークプロトコルサブシステム114030、及び/又は基本モジュール114040を含むことができる。

【1280】

IOサブシステム114010は、第2スクリーンデバイスに装着可能な、外部接続される全ての装置が接続されていてもよい。IOサブシステム114010には、キーパッド、ディスプレイ、マイクロホン、スピーカー、ステレオジャッキ、動きセンサー、光センサー、GPS及び/又はカメラなどを接続することができる。各入出力装置は、それぞれ、キーモジュール、ディスプレイ制御、オーディオ制御、センサー制御及び/又はカメラ制御によって制御される。各入出力装置は、デバイスドライバでコントロールされ、それぞれのセンサーやカメラなどは、IOサブシステム114010によってSDK形態で関数

10

【1281】

ファイルシステム114020は、外部SDカードに対するファイルを読み書き可能になっており、USBに接続してデータ通信可能である。SDカード、フラッシュメモリー、USBなどの装置が接続可能である。それぞれSDインターフェース、フラッシュメモリー制御、USBバス制御によって制御される。

【1282】

ネットワークプロトコルサブシステム(114030)では、3GPP、Wi-Fi及び/又はブルートゥースなどを介して通信可能である。

20

【1283】

基本モジュール(114040)は、その他、第2デバイスで基本的なモジュールなどがありうる。バッテリーマネージャーは、第2スクリーンデバイスのバッテリー量などを管理し、通知は、通信社或いはメーカーで第2スクリーンデバイスに通知する場合に用いられるモジュールであってもよい。また、ストアモジュールは、第2スクリーンデバイスで提供するオープンSDKを用いて作られた第2スクリーン用アプリケーションを購入するモジュールであってもよい。アプリケーションマネージャーは、ストアモジュールを用いて設置されたアプリケーションを管理し、設置されたアプリケーションがアップグレードされた場合にもそれを知らせることができる。また、第2スクリーンデバイスでインターネット接続できるようにウェブモジュールが内蔵されていてもよい。個人の好みによって第2

30

【1284】

基本モジュール114040は、その他、様々な機能を有し、基本的に、第2スクリーンデバイスに内蔵されていなければならないプログラムである。しかし、点線で表示されたモジュールは、選択的にユーザが設置すると動作するソフトウェアモジュールであってもよい。

【1285】

例えば、UPnPフレームワークモジュールは、基本的に第2スクリーンデバイスで支援しない場合にも、内蔵されていてもよい。内蔵される場合には、ネイティブアプリケーションも併せて内蔵され、UPnP動作を円滑にしなければならない。本受信機の構造図では、UPnPフレームワークは、選択的にユーザが第2スクリーンサービスアプリケーション或いはUPnPフレームワークを支援するアプリケーションを設置することができる。UPnPフレームワークモジュールを介して、地上波を受信し得る受信機を探すことができる。

40

【1286】

図115は、第2スクリーンデバイスシナリオの一実施例である。

【1287】

第2スクリーンデバイスシナリオの一実施例について説明する。

【1288】

50

第2スクリーンデバイスシナリオの一実施例は、トリガーを受信する段階(s115010)、TPTを要求する段階(s115020)、応答としてTPTを送信する段階(s115030)、TDOを要求する段階(s115040)、応答としてTDOを送信する段階(s115050)、TDOを実行する段階(s115060)、scanDeviceListメッセージを送信する段階(s115070)、ディスカバリ機能呼び出す段階(s115080)、UPnPアプリケーションを実行する段階(s115090)、サーチメッセージをマルチキャストする段階(s115100)、UPnPフレームワークを通知する段階(s115110)、装置リストを返還する段階(s115120)、装置ハンドルを返還する段階(s115130)、scanServiceListメッセージを送信する段階(s115140)、サービスリスト機能呼び出す段階(s115150)、GetServiceDescriptionメッセージを送信する段階(s115160)、HTTPメッセージを送信する段階(s115170)、サービスリストを返還する段階(s115180)、サービスハンドルのリストを返還する段階(s115190)、hnHandleメッセージを送信する段階(s115200)、サービスリスト機能呼び出す段階(s115210)、GetDeviceProfileメッセージを送信する段階(s115220)、HTTPメッセージを送信する段階(s115230)、soap応答を返還する段階(s115240)、soap応答XMLを返還する段階(s115250)、soap応答をパーシングする段階(s115260)、hnHandleメッセージを送信する段階(s115270)、サービスリスト機能呼び出す段階(s115280)、InputUIListingメッセージを送信する段階(s115290)、HTTPメッセージを送信する段階(s115300)、soap応答を返還する段階(s115310)、TimeToLiveを返還する段階(s115320)、hnHandleメッセージを送信する段階(s115330)、サービスリスト機能呼び出す段階(s115340)、DisplayMessageを送信する段階(s115350)、HTTPmessageを送信する段階(s115360)、ヌル(null)を返還する段階(s115370)、ヌルを返還する段階(s115380)、及び/又は第2スクリーンアプリケーションを実行する段階(s115390)を含むことができる。

【1289】

トリガーを受信する段階(s115010)は、放送社から放送ストリームを通じてDTVCC iTVメッセージでトリガーを受信する段階であってもよい。

【1290】

TPTを要求する段階(s115020)は、受信したトリガーをパーシングし解釈して、放送社インターネットサーバーに関連したTPTを要求する段階であってもよい。

【1291】

応答としてTPTを送信する段階(s115030)は、要求されたTPTを送る段階であってもよい。

【1292】

TDOを要求する段階(s115040)は、関連したTDOをダウンロードする必要がある場合、放送社インターネットサーバーに関連TDOを要求する段階であってもよい。

【1293】

応答としてTDOを送信する段階(s115050)は、要求されたTDOを送る段階であってもよい。

【1294】

TDOを実行する段階(s115060)は、トリガーマジュールがTDOを実行させる段階であってもよい。

【1295】

scanDeviceListメッセージを送信する段階(s115070)は、実行されたDO(又はTDO)がデバイススキャンのためにデバイスリストをスキャンすることを要求するメッセージを送る段階であってもよい。

【1296】

10

20

30

40

50

ディスカバリ機能呼び出す段階 (s 115 0 8 0) は、デバイスディスカバリのために UPnP フレームワークのディスカバリ機能呼び出す段階であってもよい。

【 1 2 9 7 】

UPnP アプリケーション (s 115 0 9 0) は、第 2 スクリーンデバイスで第 2 スクリーンサービスランチャ用 UPnP アプリケーションを実行する段階であってもよい。この段階は、プライマリデバイスと独立した段階であり、UPnP アプリケーションを実行する段階 (s 115 0 9 0) 前に実行されてもよい。

【 1 2 9 8 】

サーチメッセージをマルチキャストする段階 (s 115 1 0 0) は、UPnP フレームワークが第 2 スクリーンデバイスを探すサーチメッセージをホームネットワーク内にマルチキャストする段階であってもよい。

10

【 1 2 9 9 】

UPnP フレームワークを通知する段階 (s 115 1 1 0) は、サーチメッセージを受けた第 2 スクリーンデバイスが通知メッセージをプライマリデバイスに送信する段階であってもよい。これで、第 2 スクリーンデバイスの存在を知らせることができる。

【 1 3 0 0 】

デバイスリストを返還する段階 (s 115 1 2 0) は、UPnP フレームワークがデバイスディスカバリ後に、デバイスリストを返還する段階であってもよい。

【 1 3 0 1 】

デバイスハンドルを返還する段階 (s 115 1 3 0) は、受信したデバイスリストを DO に伝達する段階であってもよい。

20

【 1 3 0 2 】

scanServiceList メッセージを送信する段階 (s 115 1 4 0) は、実行された DO (又は、TDO) がサービススキャンのためにサービスリストをスキャンすることを要求するメッセージを送る段階であってもよい。

【 1 3 0 3 】

サービスリスト機能呼び出す段階 (s 115 1 5 0) は、サービスディスカバリのために、UPnP フレームワークのサービスリスト機能呼び出す段階であってもよい。

【 1 3 0 4 】

getServiceDescription メッセージを送信する段階 (s 115 1 6 0) は、第 2 スクリーンデバイスのサービス記述を得るために第 2 スクリーンデバイスに要求する段階であってもよい。

30

【 1 3 0 5 】

HTTP メッセージを送信する段階 (s 115 1 7 0) は、getServiceDescription メッセージを送信する段階 (s 115 1 6 0) における要求に対する結果を送る段階であり、前述したように、HTTP / 1 . 1 2 0 0 OK (成功) のようなメッセージを送ることができる。サービス記述を XML 形態で受信することもできる。

【 1 3 0 6 】

サービスリストを返還する段階 (s 115 1 8 0) は、UPnP フレームワークがサービス記述を受けた後、サービスリストを返還する段階であってもよい。

40

【 1 3 0 7 】

サービスハンドルのリストを返還する段階 (s 115 1 9 0) は、受信したサービスリストをリターンする段階であってもよい。

【 1 3 0 8 】

hnHandle メッセージを送信する段階 (s 115 2 0 0) は、デバイスプロファイルを得るためにメッセージを送る段階であってもよい。

【 1 3 0 9 】

サービスリスト機能呼び出す段階 (s 115 2 1 0) は、サービスディスカバリのために、UPnP フレームワークのサービスリスト機能呼び出す段階であってもよい。

【 1 3 1 0 】

50

GetDeviceProfileメッセージを送信する段階(s115220)は、第2スクリーンデバイスのサービス記述を得るために第2スクリーンデバイスに要求する段階であってもよい。前述したGetDeviceProfileアクションを用いることができる。HTTPメッセージを送信する段階(s115230)は、GetDeviceProfileメッセージを送信する段階(s115220)における要求に対する結果を送る段階であり、前述したようにHTTP/1.1 200 OK(成功)のようなメッセージを送ることができる。デバイスプロファイルを受信することもできる。

【1311】

soap応答を返還する段階(s115240)は、受信したデバイスプロファイルを返還する段階であってもよい。

10

【1312】

soap応答XMLを返還する段階(s115250)は、受信したデバイスプロファイルをXML形態で返還する段階であってもよい。

【1313】

soap応答をパーシングする段階(s115260)は、DOが受信したSOAP messageをパーシングする段階であってもよい。

【1314】

hnHandleメッセージを送信する段階(s115270)は、第2スクリーンサービスのURLアドレスを第2スクリーンデバイスに知らせるためにメッセージを送る段階であってもよい。

20

【1315】

サービス呼び出しリスト機能呼び出す段階(s115280)は、UPnPフレームワークのサービスリスト機能呼び出す段階であってもよい。

【1316】

InputUIListingメッセージを送信する段階(s115290)は、第2スクリーンサービスのURLアドレスを第2スクリーンデバイスに知らせる段階であってもよい。前述したAddUIListingアクションを用いることができる。第2スクリーンデバイスは、取得したURLをUIListingに追加することができる。

【1317】

HTTPメッセージを送信する段階(s115300)は、InputUIListingメッセージを送信する段階(s115290)における要求に対する結果を送る段階であり、前述したようにHTTP/1.1 200 OK(成功)のようなメッセージを送ることができる。

30

【1318】

soap応答を返還する段階(s115310)は、URLの送信がなされたことを知らせるためのメッセージを返還する段階であってもよい。

【1319】

TimeToLiveを返還する段階(s115320)は、HTTP messageを送信する段階(s115300)と同様に、InputUIListingメッセージを送信する段階(s115290)における要求に対する結果を送る段階であってもよい。

40

【1320】

hnHandleメッセージを送信する段階(s115330)は、ディスプレイメッセージを第2スクリーンデバイスに伝達するために、メッセージを送る段階であってもよい。

【1321】

サービスリスト機能呼び出す段階(s115340)は、UPnPフレームワークのサービスリスト機能呼び出す段階であってもよい。

【1322】

DisplayMessageを送信する段階(s115350)は、第2スクリーンデバイスにディスプレイメッセージを送信する段階であってもよい。ディスプレイメッセー

50

ジは、メッセージタイプなどの情報を含むことができる。前述したDisplayMessageアクションを用いることができる。第2スクリーンデバイスは、ディスプレイメッセージによって、メッセージを第2スクリーンにディスプレイすることができる。

【1323】

HTTPメッセージを送信する段階(s115360)は、DisplayMessageを送信する段階(s115350)に対する結果を送る段階であり、前述したようにHTTP/1.1 200 OK(成功)のようなメッセージを送ることができる。

【1324】

ヌルを返還する段階(s115370)は、HTTP/1.1 200 OKを受けた場合、ヌルを返還する段階であってもよい。

【1325】

ヌルを返還する段階(s115380)は同様に、ヌルを応答として受けた場合、DOにヌルを返還する段階であってもよい。

【1326】

第2スクリーンアプリケーションを実行する段階(s115390)は、第2スクリーンデバイスで第2スクリーンアプリケーションを実行する段階であってもよい。

【1327】

図116は、プライマリデバイスでデジタルサービス信号を処理する方法の実施例を示す図である。

【1328】

プライマリデバイスでデジタルサービス信号を処理する方法の実施例は、プライマリアプリケーションを実行する段階(s116010)、インタラクティブサービスのメッセージを送信する段階(s116020)、記述に対する要求を受信する段階(s116030)、記述を送信する段階(s116040)、及び/又はインタラクティブサービスを提供する段階(s116050)を含むことができる。

【1329】

プライマリアプリケーションを実行する段階(s116010)は、プライマリデバイスがデジタルサービス信号に対するプライマリアプリケーションを実行する段階であってもよい。ここで、プライマリアプリケーションは、トリガーされたアプリケーション又はトリガーされないアプリケーションであってもよい。ここで、プライマリデバイスは、上述したTV受信機であってもよい。ここで、プライマリアプリケーションは、TV受信機によって実行されるDOを意味することができる。ここで、プライマリアプリケーションは、DO、TDO、NDO、又はUDOであってもよい。

【1330】

インタラクティブサービスのメッセージを送信する段階(s116020)において、メッセージは、プライマリアプリケーションに関連したセカンダリアプリケーションに対するインタラクティブサービスのメッセージを意味することができる。ここで、上述したメッセージは、プライマリデバイスから第2スクリーンデバイス(又は、第2スクリーンデバイスによって実行されるアプリケーション)に送信することができる。ここで、インタラクティブサービスは、トリガーされたアプリケーションに関連した第1モード又はトリガーされていないアプリケーションに関連した第2モードを含むことができる。ここで、セカンダリアプリケーションは、上述したプライマリアプリケーション及び第2スクリーンデバイスによって実行されるアプリケーションと関連することができる。アプリケーションは、DO、TDO、NDO、又はUDOであってもよい。ここで、トリガーされたアプリケーションはTDOを意味し、トリガーされていないアプリケーションはNDOを意味することができる。

【1331】

記述に対する要求を受信する段階(s116030)は、プライマリデバイスが第2スクリーンデバイス(又は、第2スクリーンデバイスによって実行されるアプリケーション)から記述を受信する段階であってもよい。ここで、記述は、上述したインタラクティブ

10

20

30

40

50

サービスを識別する記述を意味することができる。

【1332】

記述を送信する段階 (s 1 1 6 0 4 0) は、プライマリデバイスが第2スクリーンデバイス (又は、第2スクリーンデバイスによって実行されるアプリケーション) に、要求された記述を送信する段階であってもよい。ここで、記述は、記述に記述されたインタラクティブサービスのそれぞれのサービスタイプ情報又はサービス i d 情報を含むことができる。ここで、サービス i d 情報は、インタラクティブサービスのそれぞれを識別することができる。ここで、サービスタイプ情報は、インタラクティブサービスのそれぞれのタイプを示すことができる。

【1333】

インタラクティブサービスを提供する段階 (s 1 1 6 0 5 0) は、プライマリデバイスが第2スクリーンデバイス (又は、第2スクリーンデバイスによって実行されるアプリケーション) にインタラクティブサービスを提供する段階であってもよい。インタラクティブサービスは、インタラクティブサービスに対する情報を提供することによって提供することができる。

【1334】

本発明の一実施例において、上述したインタラクティブサービスに対する情報は、インタラクティブサービスのそれぞれによって定義することができる。また、上述したインタラクティブサービスに対する情報は、トリガー、バイト及びURL情報のうちの一つであってもよい。ここで、インタラクティブサービスに対する上述した情報がトリガーに対応すると、プライマリデバイスは、トリガーサービスを提供することができる。ここで、インタラクティブサービスに対する上述した情報がバイトに対応すると、プライマリデバイスは双方向通信サービスを提供することができる。ここで、インタラクティブサービスに対する上述した情報がURL情報であれば、プライマリデバイスはA p p U R L サービスを提供することができる。

【1335】

本発明の他の実施例において、上述したインタラクティブサービスに対する情報はトリガーであってもよい。また、トリガーの少なくとも一つは、デジタルサービス信号のチャネルの変更された番号を示す値を有するチャネル変更情報を含むことができる。本実施例は、上述したトリガーサービスが上述したチャネル変更トリガーを送信する場合に対応できる。チャネル変更情報を含む少なくとも一つのトリガーは、上述したチャネル変更トリガーを意味することができる。チャネル変更情報は、変更されたチャネル番号を示す値を有することができる。

【1336】

本発明の他の実施例において、上述したインタラクティブサービスに対する情報がトリガーであれば、方法は、T P T 内の情報と結合することによってトリガーを増加 (a u g m e n t) させる段階であって、T P T は、トリガーされたアプリケーションに対するメタデータを含む、段階と、トリガー内の識別されたイベントを示すイベント指示子を有する増加したトリガーを送信する段階をさらに含むことができる。ここで、“インタラクティブサービスに対する情報がトリガーであれば”は、プライマリデバイスがトリガーサービスを提供する場合を意味することができる。本実施例は、上述した増加したトリガーが生成及び送信される場合に対応できる。“T P T 内の情報と結合することによってトリガーを増加させる段階”は、T P T 内に含まれる情報及びトリガーを結合する段階であってもよい。ここで、イベント指示子は、上述したA u g m e n t e d T r i g g e r 内のイベントエレメントを意味することができる。

【1337】

本発明の他の実施例において、上述したインタラクティブサービスに対する情報がバイトであってもよい。また、インタラクティブサービスは、アプリケーションが実行されてセカンダリアプリケーションとの通信に参加する準備ができていないかを示すs t a t u s 状態変数を含むことができる。また、方法は、s t a t u s 状態変数を設定する段階をさ

10

20

30

40

50

らに含むことができる。本実施例は、プライマリデバイスが双方向通信サービスを提供する場合を意味することができる。ここで、`status`状態変数は、上述した双方向通信サービスの第2アプローチにおける`Status`状態変数を意味することができる。ここで、“`status`状態変数を設定する段階”は、上述した`setStatusYes()`、`setStatusNo()` or `setStatus()` APIを用いる場合を意味することができる。

【1338】

本発明の他の実施例において、プライマリアプリケーションは、APIを用いてインタラクティブサービスを提供する。本実施例は、プライマリデバイスが双方向通信サービスを提供する場合を意味することができる。ここで、APIは、上述した双方向通信サービスのAPIのうち、`sendBytes` APIを意味することができる。

10

【1339】

本発明の他の実施例において、APIは、バイト及びバイト引数に対するアドレス及びポートに関する情報を有するアドレス引数を含むことができる。ここで、アドレス引数は、`sendBytes` APIのアドレス引数を意味することができる。ここで、バイト引数は、`sendBytes` APIのバイト引数を意味することができる。

【1340】

本発明の他の実施例において、方法は、メッセージを送信する前にプライマリアプリケーションに関連したセカンダリアプリケーションに対するインタラクティブサービスのメッセージに対する要求を受信する段階をさらに含むことができる。第2スクリーンデバイスは要求をプライマリデバイスに伝送することができる。要求は、プライマリアプリケーションに関連したセカンダリアプリケーションに対するインタラクティブサービスのメッセージに対するものである。ここで、メッセージは、上述したディスカバリメッセージを意味することができる。プライマリデバイスがメッセージに対する要求を受信すると、プライマリデバイスはメッセージを第2スクリーンデバイスに送信することができる。本実施例は、デバイスディスカバリ又はサービスディスカバリに対応できる。

20

【1341】

図117は、セカンダリアプリケーションでデジタルサービス信号を処理する方法の実施例を示す図である。

【1342】

セカンダリアプリケーションでデジタルサービス信号を処理する方法の実施例は、アプリケーションを実行する段階(`s117010`)、インタラクティブサービスのメッセージを受信する段階(`s117020`)、記述に対する要求を送信する段階(`s117030`)、記述を受信する段階(`s117040`)、通知のためのプロトコルに加入する段階(`s117050`)、通知を受信する段階(`s117060`)、受信された通知を用いて情報を受信する段階(`s117070`)、及び/又はファイルをダウンロードする段階(`s117080`)を含むことができる。

30

【1343】

アプリケーションを実行する段階(`s117010`)は、第2スクリーンデバイスがデジタルサービス信号に対するアプリケーションを実行する段階であってもよい。ここで、アプリケーションは、第2スクリーンデバイスによって実行されうるDO、TDO、NDO、又はUDOを意味することができる。

40

【1344】

インタラクティブサービスのメッセージを受信する段階(`s117020`)において、メッセージは、実行されたアプリケーションに対するインタラクティブサービスのメッセージを意味することができる。ここで、メッセージは、プライマリデバイスから第2スクリーンデバイスに送信することができる。

【1345】

記述に対する要求を送信する段階(`s117030`)は、第2スクリーンデバイス(又は、第2スクリーンデバイスによって実行される第2スクリーンアプリケーション)がプ

50

ライマリデバイスに記述を要求する段階であってもよい。ここで、記述はインタラクティブサービスを識別することができる。

【1346】

記述を受信する段階（s 117040）は、第2スクリーンデバイス（又は、第2スクリーンデバイスによって実行される第2スクリーンアプリケーション）がプライマリデバイスによって要求された記述を受信する段階であってもよい。ここで、記述は、インタラクティブサービスのそれぞれを識別するサービスid情報及びインタラクティブサービスのそれぞれに対するタイプを示すサービスタイプ情報を有することができる。

【1347】

通知のためのプロトコルに加入する段階（s 117050）、通知を受信する段階（s 117060）、及び受信された通知を用いて情報を受信する段階（s 117070）は、第2スクリーンデバイス（又は、第2スクリーンデバイスによって実行される第2スクリーンアプリケーション）が上述の記述を用いて少なくとも一つのインタラクティブサービスに接近する段階に含まれてもよい。

10

【1348】

通知のためのプロトコルに加入する段階（s 117050）は、第2スクリーンデバイス（又は、第2スクリーンデバイスによって実行される第2スクリーンアプリケーション）がプライマリデバイスによって提供された少なくとも一つのインタラクティブサービスに対する通知のためのプロトコルに加入する段階であってもよく、これは、“イベントング”プロトコル”に加入する上述した段階であってもよい。

20

【1349】

通知を受信する段階（s 117060）は、プライマリデバイスからインタラクティブサービスに対する通知を受信する上述した段階であってもよい。通知は、上述した非同期通知であってもよい。

【1350】

受信された通知を用いて情報を受信する段階（s 117070）は、受信された通知を用いてインタラクティブサービスに対する情報を受信する段階であってもよい。

【1351】

ファイルをダウンロードする段階（s 117080）は、アプローチインタラクティブサービスに対するファイルをダウンロードする段階であってもよい。ファイルは、上述したアプリケーションであってもよい。

30

【1352】

本発明の一実施例において、上述したインタラクティブサービスに対する情報は、インタラクティブサービスのそれぞれによって定義することができる。また、上述したインタラクティブサービスに対する情報は、トリガー、バイト及びURL情報のうちの一つであってもよい。ここで、インタラクティブサービスに対する上述した情報がトリガーに対応すると、プライマリデバイスはトリガーサービスを提供することができる。ここで、インタラクティブサービスに対する上述した情報がバイトに対応すると、プライマリデバイスは双方向通信サービスを提供することができる。ここで、インタラクティブサービスに対する上述した情報がURL情報であれば、プライマリデバイスはAppURLサービスを提供することができる。

40

【1353】

本発明の他の実施例において、探索されたインタラクティブサービスに対する情報がトリガーであれば、トリガーの少なくとも一つは、デジタルサービス信号のチャンネルの変更された番号を示す値を有するチャンネル変更情報を含むことができる。本実施例は、上述したトリガーサービスが上述したチャンネル変更トリガーを送信する場合に対応できる。チャンネル変更情報を含む少なくとも一つのトリガーは、上述したチャンネル変更トリガーを意味することができる。チャンネル変更情報は、変更されたチャンネル番号を示す値を有することができる。

【1354】

50

本発明の他の実施例において、上述したアクセスされたインタラクティブサービスに対する情報がバイトであれば、方法は、受信された通知を用いてインタラクティブサービスが始まる準備ができていないかを決定する段階、及び受信された通知を用いてバイトを送信するための接続に対する要求を送信する段階をさらに含む。本実施例は、上述した双方向通信サービスが提供される場合に対応できる。本実施例で、通知は、プライマリデバイスによって実行されるDOが通信する準備ができていないかを示すためのものである。本実施例で、バイトを送信するための接続は、TCP/IP接続であってもよい。また、本実施例は、プライマリデバイスへのTCP/IP接続を要求する動作に対応できる。

【1355】

本発明の他の実施例において、方法は、メッセージを受信する前に実行されたアプリケーションに対するインタラクティブサービスのメッセージに対する要求を送信する段階をさらに含む。第2スクリーンデバイスは、要求をプライマリデバイスに伝送することができる。要求は、セカンダリアプリケーションに対するインタラクティブサービスのメッセージに対するものである。メッセージは、上述したディスカバリメッセージを意味することができる。プライマリデバイスがメッセージに対する要求を受信すると、プライマリデバイスは、メッセージを第2スクリーンデバイスに送信することができる。本実施例は、デバイスディスカバリ又はサービスディスカバリに対応できる。

【1356】

図118は、プライマリデバイスとしてデジタルサービス信号を処理する装置の実施例を示す図である。

【1357】

プライマリデバイスとしてデジタルサービス信号を処理する装置の実施例は、実行モジュール118010、送信モジュール118020、受信モジュール118030、及び/又は提供モジュール118040を備えることができる。

【1358】

実行モジュール118010は、デジタルサービス信号に対するプライマリアプリケーションを実行するように構成することができる。ここで、プライマリアプリケーションは、トリガーされたアプリケーション又はトリガーされていないアプリケーションであってもよい。ここで、プライマリデバイスは、上述したTV受信機であってもよい。ここで、プライマリアプリケーションは、TV受信機によって実行されるDOを意味することができる。ここで、プライマリアプリケーションは、DO、TDO、NDO、又はUDOであってもよい。

【1359】

送信モジュール118020は、プライマリアプリケーションに関連したセカンダリアプリケーションに対するインタラクティブサービスのメッセージを送信するように構成することができる。また、送信モジュール118020は、インタラクティブサービスを識別する記述を送信するように構成することができる。ここで、インタラクティブサービスは、トリガーされたアプリケーションに関連した第1モード又はトリガーされていないアプリケーションに関連した第2モードを含むことができる。ここで、記述は、要求によって送信することができる。上述したメッセージは、第2スクリーンデバイス(又は、第2スクリーンデバイスによって実行されるアプリケーション)に送信することができる。ここで、セカンダリアプリケーションは、上述したプライマリアプリケーションに関連したアプリケーション及び第2スクリーンデバイスによって実行されたアプリケーションであってもよい。アプリケーションは、DO、TDO、NDO、又はUDOであってもよい。ここで、トリガーされたアプリケーションはTDOを意味し、トリガーされていないアプリケーションはNDOを意味することができる。

【1360】

受信モジュール118030は、インタラクティブサービスのそれぞれを識別するサービスid情報及びインタラクティブサービスのそれぞれのタイプを識別するサービスタイプ情報を有する記述に対する要求を受信するように構成することができる。ここで、記述

10

20

30

40

50

は、記述に記述されたインタラクティブサービスのそれぞれのサービスタイプ情報及びサービスid情報を含むことができる。

【1361】

ここで、送信モジュール118020及び受信モジュール118030は、設計者の意図によって一つのモジュールに具備することができる。

【1362】

提供モジュール118040は、インタラクティブサービスに対する情報を送信することによってインタラクティブサービスを提供するように構成することができる。提供モジュール118040は、インタラクティブサービスを第2スクリーンデバイス(又は、第2スクリーンデバイスによって実行されるアプリケーション)に提供することができる。

10

【1363】

本発明の実施例において、上述したインタラクティブサービスに対する情報は、インタラクティブサービスのそれぞれによって定義することができる。また、上述したインタラクティブサービスに対する情報は、トリガー、バイト及びURL情報のうちの一つであってもよい。ここで、インタラクティブサービスに対する上述した情報がトリガーに対応すると、プライマリデバイスはトリガーサービスを提供することができる。ここで、インタラクティブサービスに対する上述した情報がバイトに対応すると、プライマリデバイスは双方向通信サービスを提供することができる。ここで、インタラクティブサービスに対する上述した情報がURL情報であれば、プライマリデバイスはAppURLサービスを提供することができる。

20

【1364】

本発明の他の実施例において、上述したインタラクティブサービスに対する情報はトリガーであってもよい。また、トリガーの少なくとも一つは、デジタルサービス信号のチャネルの変更された番号を示す値を有するチャンネル変更情報を含むことができる。本実施例は、上述したトリガーサービスが上述したチャンネル変更トリガーを送信する場合に対応できる。チャンネル変更情報を含む少なくとも一つのトリガーは、上述したチャンネル変更トリガーを意味することができる。チャンネル変更情報は、変更されたチャンネル番号を示す値を有することができる。

【1365】

本発明の他の実施例において、上述したインタラクティブサービスに対する情報がトリガーであれば、装置は、増加モジュールをさらに備えることができる。増加モジュールは、TPT内の情報と結合することによってトリガーを増加させるように構成することができる。TPTは、トリガーされたアプリケーションに対するメタデータを含むことができる。上述した送信モジュールはさらに、トリガー内の識別されたイベントを示すイベント指示子を有する増加したトリガーを送信することができる。ここで、“インタラクティブサービスに対する情報がトリガーであれば”は、プライマリデバイスがトリガーサービスを提供する場合を意味することができる。本実施例は、上述の増加したトリガーが生成及び送信される場合に対応できる。増加モジュールは、TPT内に含まれる情報及びトリガーを結合することができる。ここで、イベント指示子は、上述したAugmentedTrigger内のイベントエレメントを意味することができる。

30

40

【1366】

本発明の他の実施例において、上述したインタラクティブサービスに対する情報がバイトであってもよい。また、インタラクティブサービスは、アプリケーションが実行されてセカンダリアプリケーションとの通信に参加する準備ができていないかを示すstatus状態変数を含むことができる。また、装置は、status状態変数を設定する段階をさらに含むことができる。本実施例は、プライマリデバイスが双方向通信サービスを提供する場合を意味することができる。ここで、status状態変数は、上述した双方向通信サービスの第2アプローチにおけるStatus状態変数を意味することができる。ここで、“status状態変数を設定する段階”は、上述したsetStatusYes()、setStatusNo() or setStatus() APIを用いる場合を意

50

味することができる。

【1367】

本発明の他の実施例において、プライマリアプリケーションは、APIを用いてインタラクティブサービスを提供する。本実施例は、プライマリデバイスが双方向通信サービスを提供する場合を意味することができる。ここで、APIは、上述した双方向通信サービスのAPIのうちのsendBytes APIを意味することができる。

【1368】

本発明の他の実施例において、APIは、バイト及びバイト引数に対するアドレス及びポートに関する情報を有するアドレス引数を含むことができる。ここで、アドレス引数は、sendBytes APIのアドレス引数を意味することができる。ここで、バイト引数は、sendBytes APIのバイト引数を意味することができる。本発明の他の実施例において、受信モジュールはさらに、送信モジュールがメッセージを送信する前にプライマリアプリケーションに関連したセカンダリアプリケーションに対するインタラクティブサービスのメッセージに対する要求を受信するように構成することができる。第2スクリーンデバイスは要求をプライマリデバイスに伝送することができる。要求は、プライマリアプリケーションに関連したセカンダリアプリケーションに対するインタラクティブサービスのメッセージに対するものである。ここで、メッセージは、上述したディスカバリメッセージを意味することができる。プライマリデバイスがメッセージに対する要求を受信すると、プライマリデバイスはメッセージを第2スクリーンデバイスに送信することができる。本実施例は、デバイスディスカバリ又はサービスディスカバリに対応できる。

【1369】

図119は、セカンダリアプリケーションとしてデジタルサービス信号を処理する装置の実施例を示す図である。

【1370】

セカンダリアプリケーションとしてデジタルサービス信号を処理する装置の実施例は、実行モジュール119010、受信モジュール119020、伝送モジュール119030、アクセスモジュール119040、及び/又はダウンロードモジュール119050を備えることができる。

【1371】

実行モジュール119010は、デジタルサービス信号に対するアプリケーションを実行するように構成することができる。ここで、アプリケーションは、第2スクリーンデバイスによって実行されうるDO、TDO、NDO、又はUDOを意味することができる。

【1372】

受信モジュール119020は、実行されたアプリケーション及び記述に対するインタラクティブサービスのメッセージを受信するように構成することができる。ここで、メッセージはプライマリデバイスから送信されてもよい。ここで、記述はプライマリデバイスによって要求されてもよい。ここで、記述はインタラクティブサービスを識別することができる。

【1373】

伝送モジュール119030は、インタラクティブサービスを識別する記述に対する要求を伝送するように構成することができる。ここで、記述は、インタラクティブサービスのそれぞれを識別するサービスid情報及びインタラクティブサービスのそれぞれに対するタイプを示すサービスタイプ情報を有することができる。

【1374】

アクセスモジュール119040は、記述を用いて少なくとも一つのインタラクティブサービスをアクセスするように構成することができる。アクセスモジュール119040はまた、少なくとも一つのインタラクティブサービスに対する通知のためのプロトコルに加入することができる。これは、“イベントング”プロトコルに加入する上述した動作に対応する。アクセスモジュール119040はまた、通知を受信することができる。通知

10

20

30

40

50

は、上述した非同期通知であってもよい。アクセスモジュール 119040 はまた、受信された通知を用いて少なくとも一つのインタラクティブサービスに対する情報を受信することができる。

【1375】

ダウンロードモジュール 119050 は、アクセスされたインタラクティブサービスに対するファイルをダウンロードするように構成することができる。ここで、ファイルは、上述したアプリケーションであってもよい。

【1376】

本発明の一実施例で、上述したインタラクティブサービスに対する情報は、インタラクティブサービスのそれぞれによって定義することができる。また、上述したインタラクティブサービスに対する情報は、トリガー、バイト及びURL情報のうちの一つであってもよい。ここで、インタラクティブサービスに対する上述した情報がトリガーに対応すると、プライマリデバイスはトリガーサービスを提供することができる。ここで、インタラクティブサービスに対する上述した情報がバイトに対応すると、プライマリデバイスは双方向通信サービスを提供することができる。ここで、インタラクティブサービスに対する上述した情報がURL情報であれば、プライマリデバイスはAppURLサービスを提供することができる。

10

【1377】

本発明の他の実施例において、探索されたインタラクティブサービスに対する情報がトリガーであれば、トリガーの少なくとも一つは、デジタルサービス信号のチャンネルの変更された番号を示す値を有するチャンネル変更情報を含むことができる。本実施例は、上述したトリガーサービスが上述したチャンネル変更トリガーを送信する場合に対応できる。チャンネル変更情報を含む少なくとも一つのトリガーは、上述したチャンネル変更トリガーを意味することができる。チャンネル変更情報は、変更されたチャンネル番号を示す値を有することができる。

20

【1378】

本発明の他の実施例において、上述したアクセスされたインタラクティブサービスに対する情報がバイトであれば、アクセスモジュール 119040 はまた、受信された通知を用いてインタラクティブサービスが始まる準備ができていないかを決定することができる。また、アクセスモジュール 119040 は受信された通知を用いてバイトを送信するための接続に対する要求を送信することができる。本実施例は、上述した双方向通信サービスが提供される場合に対応できる。本実施例で、通知は、プライマリデバイスによって実行されるDOが通信する準備ができていないかを示すためののものであってもよい。本実施例において、バイトを送信するための接続は、TCP/IP接続であってもよい。また、本実施例は、プライマリデバイスへのTCP/IP接続を要求する動作に対応できる。

30

【1379】

本発明の他の実施例において、伝送モジュールはまた、受信モジュールがメッセージを受信する前に実行されたアプリケーションに対するインタラクティブサービスのメッセージに対する要求を伝送するように構成することができる。第2スクリーンデバイス（又は、伝送モジュール）は要求をプライマリデバイスに伝送することができる。要求は、セカンドリアプリケーションに対するインタラクティブサービスのメッセージに対するものである。ここで、メッセージは、上述したディスカバリメッセージを意味することができる。プライマリデバイスが上述したディスカバリメッセージを受信すると、プライマリデバイスはメッセージを第2スクリーンデバイスに送信することができる。本実施例はデバイスディスカバリ又はサービスディスカバリに対応できる。

40

【1380】

UPnPフレームワークは、電子装置をホームネットワークに接続し、電子装置によって機能を提供することによって提供することができる。本発明は、既存のUPnP RemoteUIクライアントサービス及びUPnP RemoteUIサーバークライアントの拡張に関するものである。したがって、既存のUPnP装置と互換可能であるとも

50

に、第2スクリーンサービスを受信することができる。

【1381】

添付の図面を参照して本発明を明瞭に説明してきたが、添付の図面に示す実施例を組み合わせる新しい実施例を設計することもできる。上述した実施例を実行するプログラムが記録されたコンピュータ読み取り可能な記録媒体が当業者の必要によって設計されると、これは添付した特許請求の範囲及びその同等物に属することができる。

【1382】

本発明の装置及び方法は、上述した実施例の構成及び方法に制限されるものではない。また、上述した実施例は、一部又は全部が選択的に結合されて様々な変更が可能な方式で構成されてもよい。

【1383】

また、本発明に係る方法は、ネットワークデバイスに提供されるプロセッサ読み取り可能記録媒体においてプロセッサ読み取り可能コードとして具現することもできる。プロセッサ読み取り可能媒体は、プロセッサによって読み取り可能なデータを記憶できるあらゆる種類の記録装置を含む。プロセッサ読み取り可能記録媒体の例には、ROM、RAM、CD-ROM、磁気テープ、フロッピーディスク、光データ記憶装置などがあり、また、インターネットを介した伝送のようなキャリアウェーブの形態として具現されてもよい。また、プロセッサ読み取り可能記録媒体は、ネットワークを通じて接続されたコンピュータシステムに分散され、分散方式でプロセッサ読み取り可能コードが記憶されて実行されてもよい。

【1384】

当業者にとって本発明の思想又は範囲から逸脱することなく本発明の様々な変形及び変更が可能であるということは明らかである。したがって、本発明は、添付した特許請求の範囲及びその同等物の範囲内で提供される本発明の変形及び変更をカバーする。

【1385】

そして、当該明細書では物の発明と方法の発明の両方とも説明されており、必要によって両発明の説明は相互補充的に適用されてもよい。

【実施例】

【1386】

発明の実施例は、前述した通り、発明を実施するための形態で詳述された。

【産業上の利用可能性】

【1387】

本発明は、放送サービス提供に関連した一連の産業分野で利用可能である。

【1388】

本発明の思想及び範囲から逸脱することなく、本発明の様々な変形及び変更が可能であるということが当業者にとっては明らかである。したがって、本発明は、添付の特許請求の範囲及び同等物の範囲内で提供される本発明の変形及び変更も含む。

10

20

30

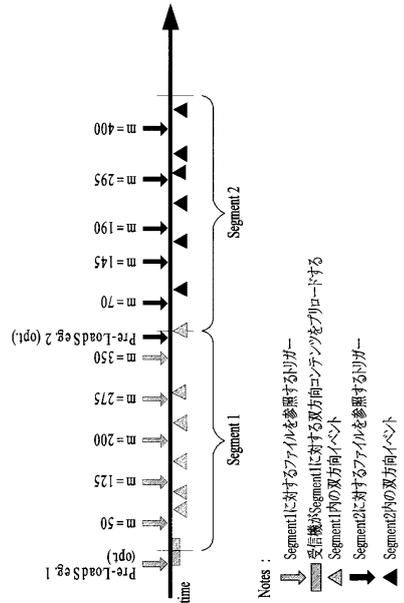
【 図 1 】

[Fig. 11]

Segment of Show A	Ad 1	Ad 2	Segment of Show B	Ad 3	Ad 4	Segment of Show B	Ad 5
-------------------	------	------	-------------------	------	------	-------------------	------

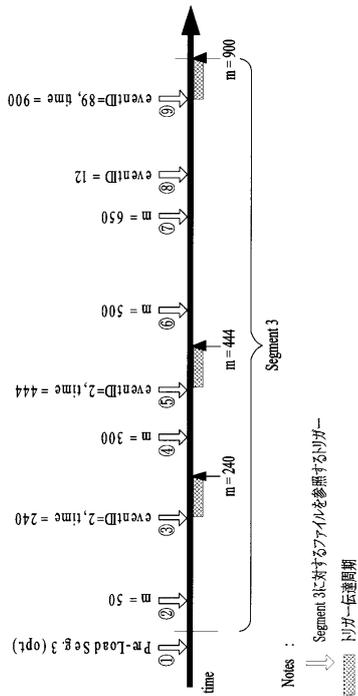
【 図 2 】

[図 2]



【 図 3 】

[図 3]



【 図 4 】

[Fig. 4]

```

Trigger = locator_part [ "m" terms ]

locator_part = hostname "m" path_segments

hostname = *( domainlabel " " toplabel
domainlabel = alphanum | alphanum *( alphanum | "-" ) alphanum
toplabel = alpha | alpha *( alphanum | "-" ) alphanum

path_segments = segment *( "m" segment )
segment = 1 * alphanum

terms = ( event_time | media_time ) [ "&" spread ] [ "&" others ]
event_time = "e" 1 * digit [ "&" { * 7 hexdigit }
media_time = "m" 1 * 7 hexdigit
others = other [ "&" other ]
other = resv_cmd "m" 1 * alphanum
spread = "s" 1 * digit
resv_cmd = <any>

alphanum = alpha | digit
alpha = lowalpha | upalpha

lowalpha = "a" | "c" | "d" | "e" | "f" | "g" | "h" | "i" | "j" | "k" | "l" | "m" | "n" | "o" | "p" | "q" | "r" | "s" | "t" | "u" | "v" | "w" | "x" | "y" | "z" | "0" | "1" | "2" | "3" | "4" | "5" | "6" | "7" | "8" | "9"
upalpha = "A" | "B" | "C" | "D" | "E" | "F" | "G" | "H" | "I" | "J" | "K" | "L" | "M" | "N" | "O" | "P" | "Q" | "R" | "S" | "T" | "U" | "V" | "W" | "X" | "Y" | "Z"
digit = "0" | "1" | "2" | "3" | "4" | "5" | "6" | "7" | "8" | "9"
hexdigit = "0" | "1" | "2" | "3" | "4" | "5" | "6" | "7" | "8" | "9" | "a" | "b" | "c" | "d" | "e" | "f"

```

【 図 5 】

[図 5]

表8.1 TDOパラメータテーブル構造

Element/Attribute (with @)	No. allowed	Data Type	Description & Value
TPT			
@majorProtocolVersion	1	Integer, range 0..15	主プロトコルバージョン
@minorProtocolVersion	0..1	integer, range 0..15	副プロトコルバージョン
@jid	1	anyURI	segment_id = domain_name/program_id
@tptVersion	1	unsignedByte	このTPTのデータバージョン
@expireDate	0..1	dateTime	将来このTPTが用いられないようになる日付
@updatingTime	0..1	unsignedShort	TPTアップデートを確認するための時間区間
@serviceID	0..1	unsignedShort	NRT service_id
@baseURL	0..1	anyURI	TPT内全ての相対的URLに対するベースURL
Capabilities	0..1	nrt:CapabilitiesType	このTPTに関連されたセグメントのための必須能力
LiveTrigger	0..1		インターネットライブトリガー伝達に関する情報
@URL	1	anyURI	秒単位ライブトリガーショートローリング
@pollPeriod	0..1	unsignedByte	周期に対するサーバーのURL
TDO			
	1..N		TDO (app) for the e segment associated with this TPT
@appID	1	unsignedShort	このアプリのアプリケーションID。このTPTの範囲内で固有
@appType	0..1	integer, range 0..15	Application type (default: 1="TDO")
@appName	0..1	string	(視聴者開始コンテンツに対する)ディスプレイ名
@globalID	0..1	anyURI	全球的に固有なアプリID
@appVersion	0..1	unsignedByte	このアプリのバージョン
@cookieSpace	0..1	unsignedByte	持続的保存必要、default=0
@frequencyOfUse	0..1	integer, range 0..15	表8.2によるコード値
@expireDate	0..1	dateTime	このアプリをキャッシュに保存するための満了日

【 図 6 】

[図 6]

@testTDO	0..1	boolean	テストアプリのためのフラグ; default="false"
@availInternet	0..1	boolean	Default="true"
@availBroadcast	0..1	boolean	Default="true"
URL	1..N	anyURI	App URL(s)
@entry	0..1	boolean	エン트리ポイント指示子; default = "false"
Capabilities	0..1	nrt:CapabilitiesType	このアプリEL表示するための必須能力
ContentItem	0..N		このアプリによって用いられるコンテンツアイテム
URL	1..N	anyURI	コンテンツアイテムのURL
@entry	0..1	boolean	Indicator of entry point; default = "false"
@updatesAvail	0..1	boolean	Default="false"
@pollPeriod	0..1	unsignedByte	秒単位のショートローリング周期
@size	0..1	24-bit integer	キロバイト単位のコンテンツアイテムの大きさ
@availInternet	0..1	boolean	Default="true"
@availBroadcast	0..1	boolean	Default="true"
Event			
	1..N		Event targeted to this TDO
@eventID	1	unsignedShort	Unique identifier of this Event element within the scope of the TDO element.
@action	1	string	Allowed values are "prep", "exec", "susp", and "kill"
@destination	0..1	unsignedByte	Device to which the event is directed (primary screen, second screen, or both)
@diffusion	0..1	unsignedByte	Period for applying diffusion, in seconds
Data	0..N	base64Binary	Data to be used for this event
@dataID	1	unsignedShort	Unique identifier of this Data element within the scope of the Event element.

【 図 7 】

[図 7]

表8.2 使用頻度属性の値の意味

"frequencyOfUse" value	Meaning
0	1回のみ使用
1	毎時
2	毎日
3	毎週
4	毎月
5-15	予約される

【 図 8 】

[図 8]

表8.3 "destination"属性値の意味

"destination" value	意味
0	予約される
1	主装置のみ
2	一つ以上の補助装置のみ
3	主装置及び/又は一つ以上の補助装置

【 図 9 】

[Fig. 9]

Syntax	No. Bits	Format
binary_16_parameters_table {		
reserved	2	"11"
expire_date_included	1	bit
segment_id_length	5	uint8
segment_id	var	bit
base_URL_length	8	uint8
base_URL	var	bit
if (expire_date_included=="1") {		
expire_date	32	uint32
}		
trigger_server_URL_length	8	uint8
trigger_server_URL	var	bit
if (trigger_server_URL_length != 0) {		
reserved	7	"1111"
trigger_delivery_type	1	uint8
if (trigger_delivery_type == 0) {		
poll_period	8	uint8
}		
}		
num_apps_in_table	8	uint8
for (k=0; k<num_apps_in_table; k++) {		
app_id	16	uint16
app_type_included	1	bit
app_name_included	1	bit

【 10 】

[Fig. 10]

global_id_included	1	bool
app_version_included	1	bool
cookie_space_included	1	bool
frequency_of_use_included	1	bool
expire_date_included	1	bool
reserved	1	'1'
If (app_type_included == '1') {		
app_type	8	uint8
}		
If (app_name_included == '1') {		
app_name_length	8	uint8
app_name	var	bool
}		
If (global_id_included == '1') {		
global_id_length	8	uint8
global_id	var	bool
}		
If (app_version_included == '1') {		
app_version	8	uint8
}		
If (cookie_space_included == '1') {		
cookie_space	8	uint8
}		
If (frequency_of_use_included) {		
frequency_of_use	8	uint8

【 11 】

[Fig. 11]

}		
if (expire_date_included) {		
expire_date	32	uint32
}		
test_app	1	bool
available_on_internet	1	bool
available_in_broadcast	1	bool
reserved	1	'1'
number_URLs	4	
for (i=0; i<number_URLs; i++) {		
URL_length	8	uint8
URL	var	bool
}		
number_content_items	8	uint8
for (i=0; i<number_content_items; i++) {		
update_avail	1	bool
avail_internet	1	bool
avail_broadcast	1	bool
content_size_included	1	bool
number_URLs	4	uint8
for (i=0; i<number_URLs; i++) {		
URL_length	8	uint8
URL	var	bool
}		
}		
if (content_size_included) {		

【 12 】

[Fig. 12]

content_size	24	uint32
}		
num_content_descriptors	8	uint8
for (i=0; i<num_content_descriptors; i++) {		
content_descriptor	var	bool
}		
number_events	16	uint16
for (i=0; i<num_events; i++) {		
event_id	16	uint16
action	5	uint5
destination_included	1	bool
diffusion_included	1	bool
data_included	1	bool
If (destination_included == ??) {		
destination	8	
}		
If (diffusion_included == ??) {		
diffusion		
}		
If (data_included == ??) {		
data_size	8	
data_bytes	var	bool
}		
}		
}		

【 13 】

[Fig. 13]

num_app_descriptors	8	uint8
for (i=0; i<num_app_descriptors; i++) {		
app_descriptor()	var	bool
}		
num_TPT_descriptors	8	uint8
for (i=0; i<num_TPT_descriptors; i++) {		
TPT_descriptor()	var	bool
}		

【 14 】

【 14 】

表8.4 活性化メッセージテーブル構造

Element/Attribute (with @)	No. allowed	Data Type	Description & Value
AMT			
@majorProtocolVersion	1	integer 0-15	主プロトコルバージョン
@minorProtocolVersion	0..1	Integer 0-15	副プロトコルバージョン
@segmentId	1	anyURI	domain_name/program_id = segment id
@beginMT	0..1	unsignedInt	このセグメント時間範囲の開始時間
Activation	1..N		活性化メッセージ
@targetTDO	1	unsignedShort	ターゲットTDOのappID
@targetEvent	1	unsignedShort	ターゲットTDO内ターゲットEventのeventid
@targetData	0..1	unsignedShort	dataID of target Data in target Event
@startTime	1	unsignedInt	動作周期の開始時間
@endTime	0..1	unsignedInt	動作周期の終了時間

【 図 15 】

[図 15]

表8.5 URLリストの構造図

Element/Attribute (with @)	No. allowed	Data Type	説明&値
UrlList			List of potentially useful URLs
TptUri	0..N	anyURI	未来セグメントのためのTPTのURL
NrtSignalingUri	0..1	anyURI	NRTシグナリングサーバーのURL
UrsUri	0..1	anyURI	使用報告サーバーのURL
PdiUri	0..1	anyURL	URL of PDI-Q table

【 図 16 】

[Fig. 16]

Syntax	No. Bits	Format
tpt_section () {		
table_id	8	TBD
section_syntax_indicator	1	'0'
private_indicator	1	'1'
reserved	2	'11'
section_length	12	uimsbf
table_id_extension {		
protocol_version	8	uimsbf
sequence_number	8	uimsbf
}		
reserved	2	'11'
TPT_data_version	5	uimsbf
current_next_indicator	1	'1'
section_number	8	uimsbf
last_section_number	8	uimsbf
service_id	16	uimsbf
tpt_bytes()	var	bsibf
}		

【 図 17 】

[Fig. 17]

```

<?xml version='1.0' encoding='UTF-8'?>
<xs:schema xmlns:xsi="http://www.w3.org/2001/XMLSchema" >
<xs:element name="UrlList">
<xs:complexType>
<xs:sequence>
<xs:element name="TptUri" type="xs:anyURI" minOccurs="0" maxOccurs="unbounded"/>
<xs:element name="NrtSignalingUri" type="xs:anyURI" minOccurs="0"/>
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:schema>

```

【 図 18 】

[図 18]

Void addTriggerEventListener(Number eventId, EventListener listener)					
Description	TPT内で現在実行中であるTDOの範囲内でeventIdによって指定されたイベントのためのリスナーを追加する。活性化トリガーによってこのイベントが活性化されると、このリスナーが呼び出され、TriggerEventタイプの客体がこのリスナーに伝達される。				
Arguments	<table border="1"> <tr> <td>eventId</td> <td>TPT内Event要素のeventId</td> </tr> <tr> <td>listener</td> <td>当該イベントのリスナー</td> </tr> </table>	eventId	TPT内Event要素のeventId	listener	当該イベントのリスナー
eventId	TPT内Event要素のeventId				
listener	当該イベントのリスナー				

【 図 21 】

[図 21]

Interface TriggerEvent : Event {		
	readonly attribute String eventId;	
	readonly attribute String data;	
	readonly attribute DOMString status;	
	}	
Properties	eventId	TPT内Event要素のeventId
	data	16進法の当該イベントの活性化のためのデータ
	status	活性化トリガーに応じてイベントが活性化されると、イベントの状態は"trigger"と同一であり、ある種類のエラーが発生すると"error"と同一である。

【 図 19 】

[図 19]

Void addTriggerEventListener(Number eventId, EventListener listener)					
Description	eventIdによって指定されたイベントのための指定されたリスナーを除去。				
Arguments	<table border="1"> <tr> <td>eventId</td> <td>The eventId of the Event element in the TPT</td> </tr> <tr> <td>listener</td> <td>The listener for the event</td> </tr> </table>	eventId	The eventId of the Event element in the TPT	listener	The listener for the event
eventId	The eventId of the Event element in the TPT				
listener	The listener for the event				

【 図 20 】

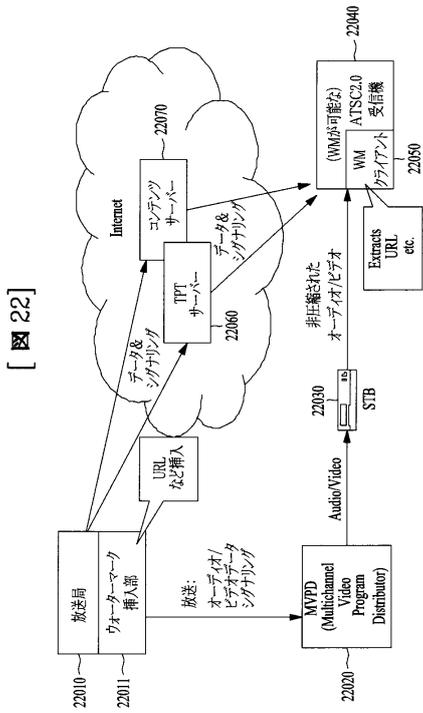
[Fig. 20]

```

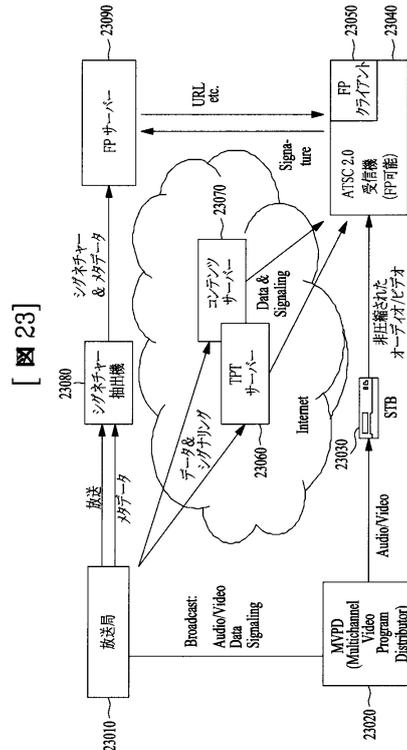
Interface EventListener {
  handleEvent(in TriggerEvent event);
};

```

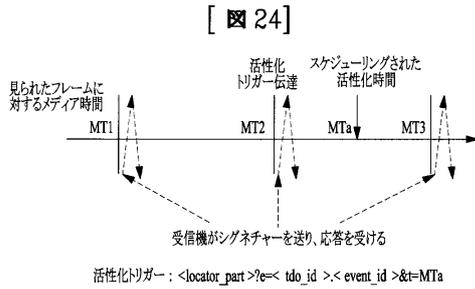
【図22】



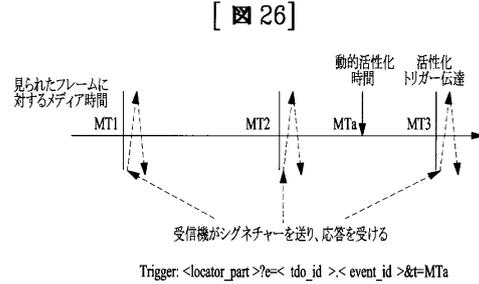
【図23】



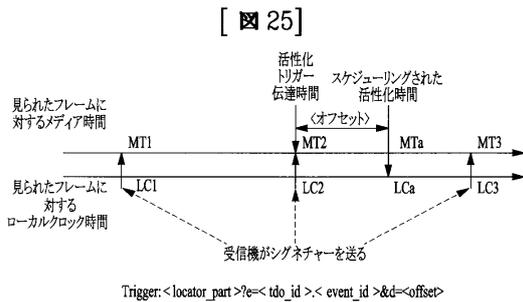
【図24】



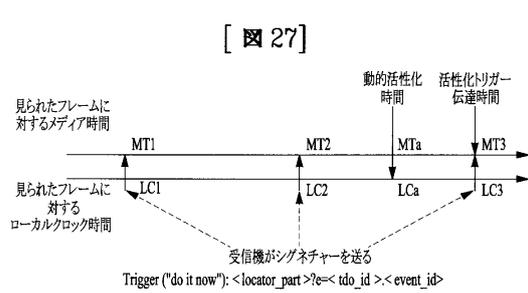
【図26】



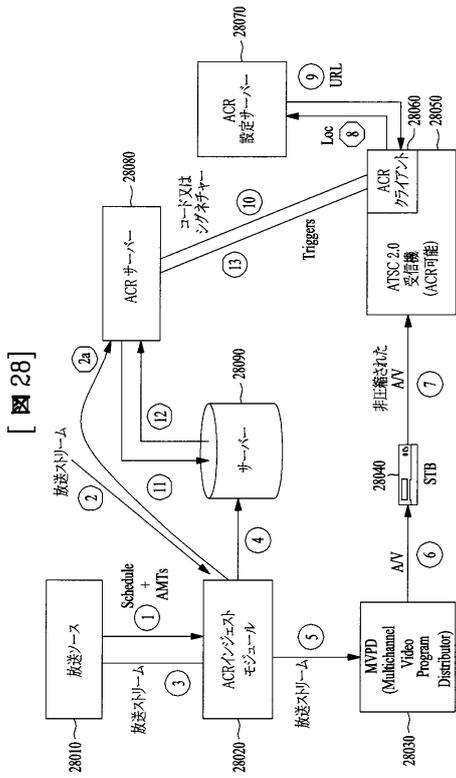
【図25】



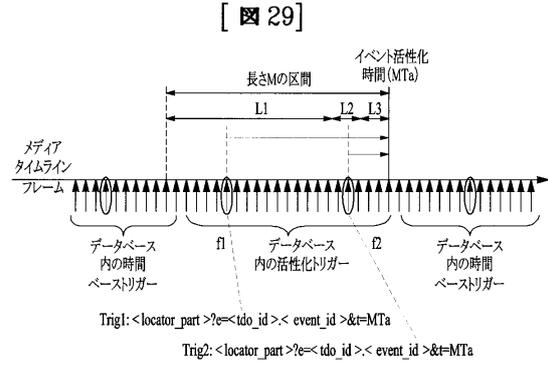
【図27】



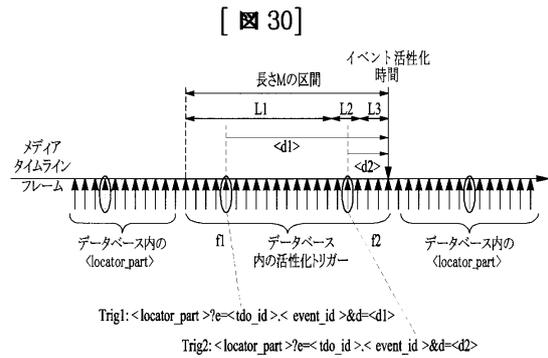
【図28】



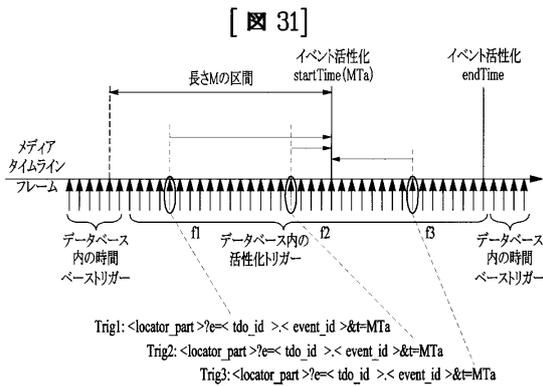
【図29】



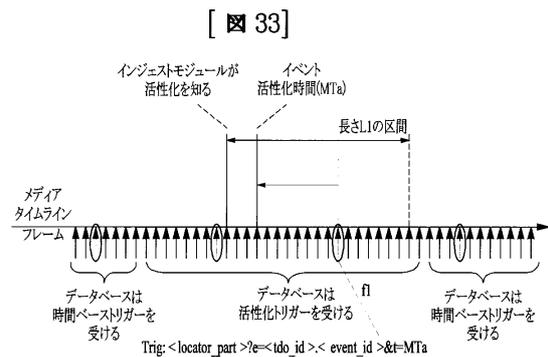
【図30】



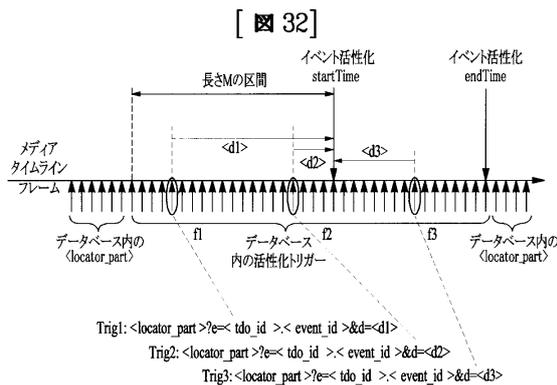
【図31】



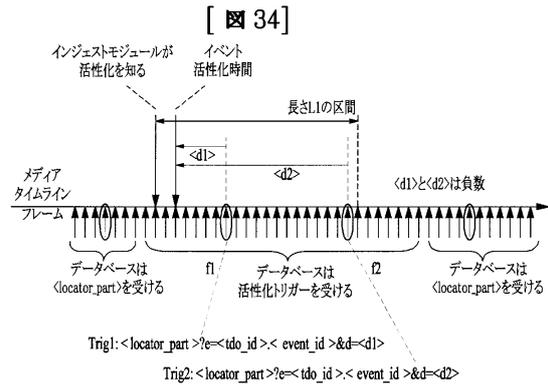
【図33】



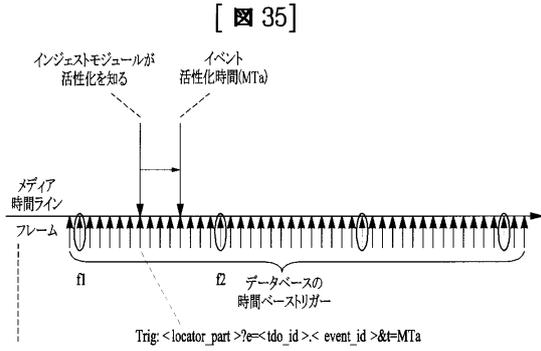
【図32】



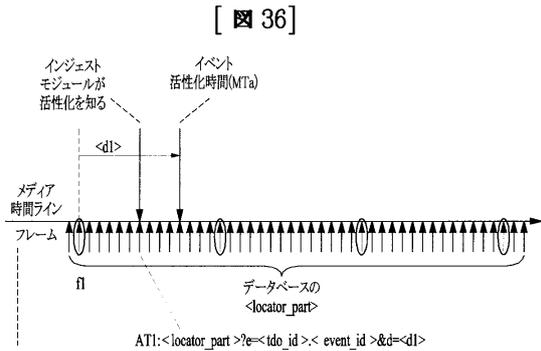
【図34】



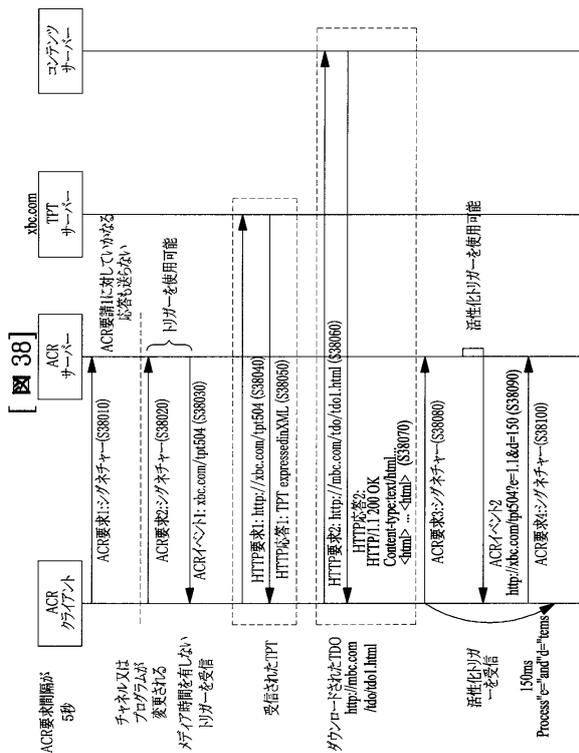
【 図 3 5 】



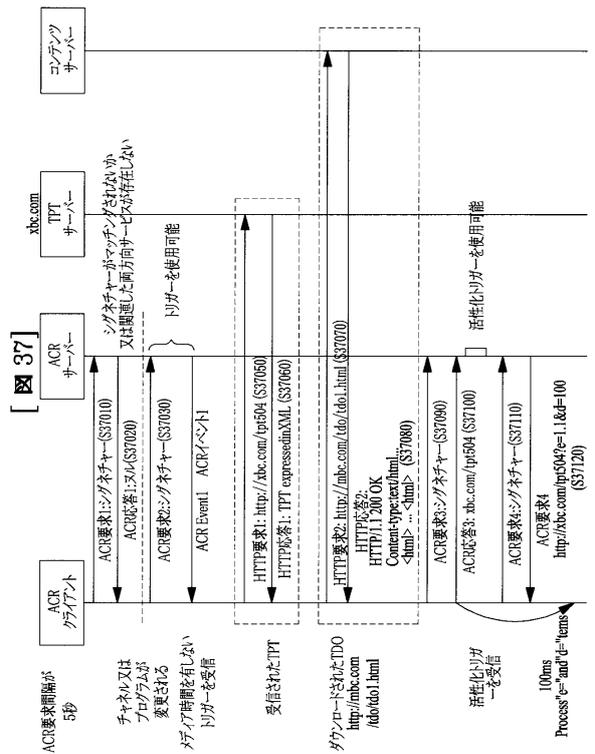
【 図 3 6 】



【 図 3 8 】



【 図 3 7 】

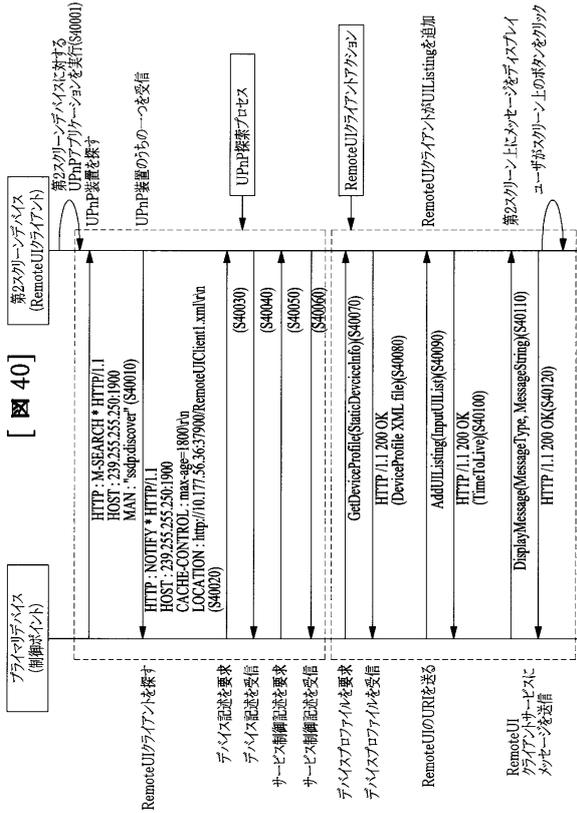


【 図 3 9 】

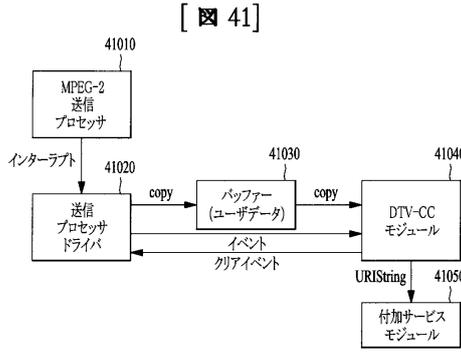
[図 39]

アクション名	説明
Connection / Disconnection	現在接続を返還
GetCurrentConnection	現在接続リストを返還
GetDeviceProfile	XMLで表現された"DeviceProfile"を返還
GetUIListing	XMLで表現された"compatibleUIs"を返還
AddUIListing / RemoveUIListing	遠隔UIのURIを追加し、装置内のUIリスティングから除去
ProcessInput	RemoteUIクライアントサービスにデータを送信
DisplayMessage	DisplayMessage RemoteUIクライアントサービスを含む装置上でメッセージをディスプレイ

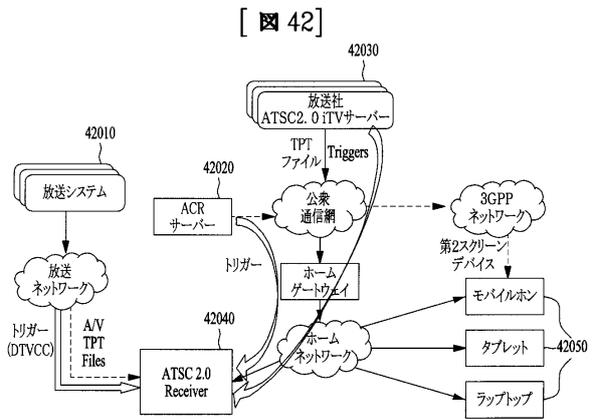
【 図 40 】



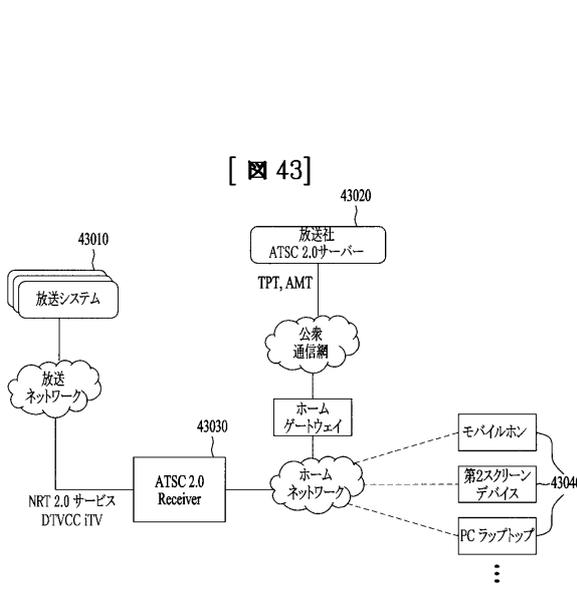
【 図 41 】



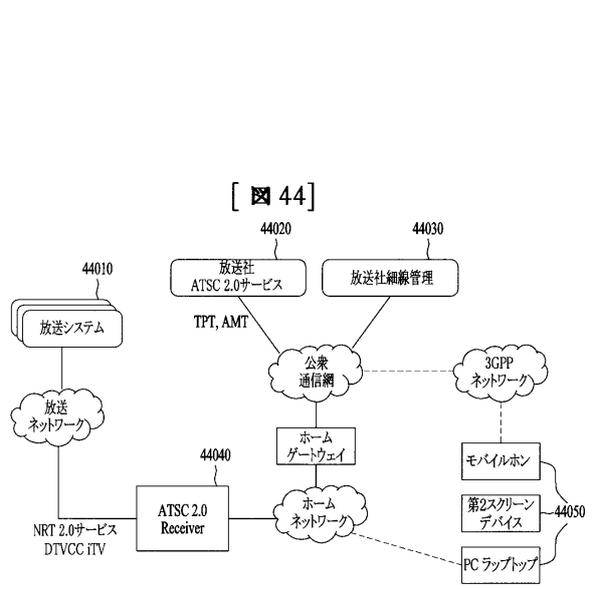
【 図 42 】



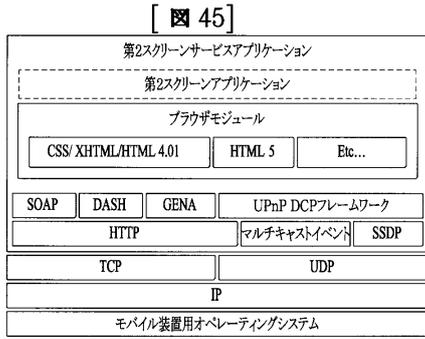
【 図 43 】



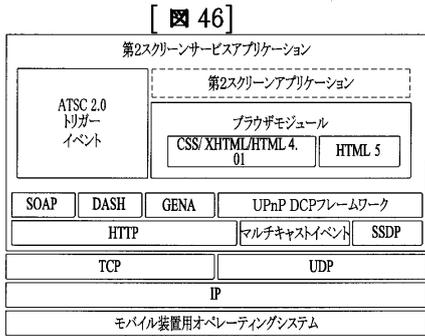
【 図 44 】



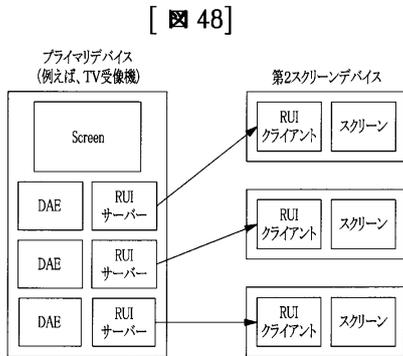
【 図 4 5 】



【 図 4 6 】



【 図 4 8 】

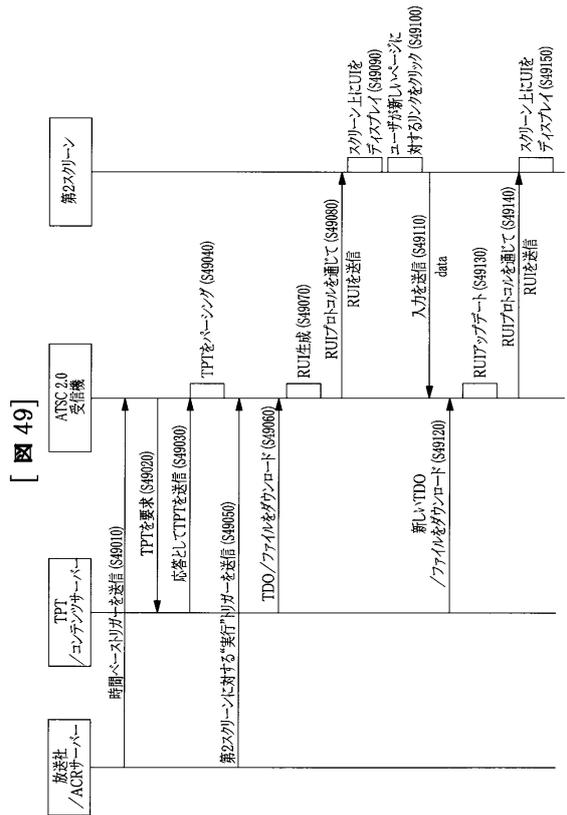


【 図 4 7 】

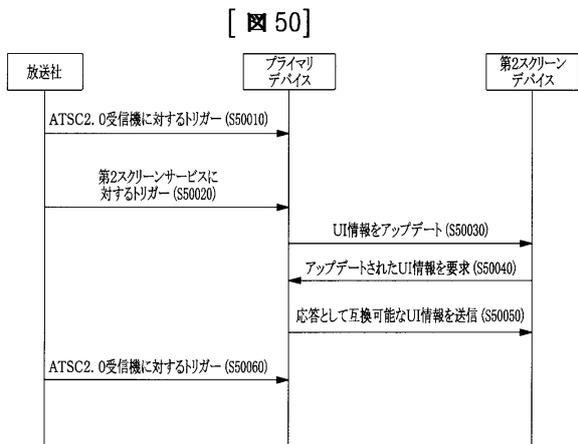
第2スクリーンアプリケーション アプリケーション 管理エンティティ	送信シーケンス	送信データ	
		URIShing(D TVCC)	URL
受信機	放送社 → (RemoveUI)サーバーサービス、TDO URL ↔ 第2スクリーンサービスアプリケーション	X	0
受信機	放送社 → 受信機 → 第2スクリーンサービスアプリケーション (RemoveUI)クライアントサービス、AddUIListing、TDO URL	X	0
第2スクリーンサービスアプリケーション	放送社 ↔ 受信機 ↔ 第2スクリーンサービスアプリケーション (RemoveUI)クライアントサービス、ProcessInput、第2スクリーン用IPTTからXMLを再生	X	0
第2スクリーンサービスアプリケーション	放送社 → 受信機 → 第2スクリーンサービスアプリケーション (RemoveUI)クライアントサービス、AddUIListing、DTVCC URIShing	0	X
第2スクリーンサービスアプリケーション	放送社 → 受信機 (RemoveUI)サーバーサービス、UIUpdate Event、DTVCC URIShing → 第2スクリーンサービスアプリケーション	0	X

【 図 47 】

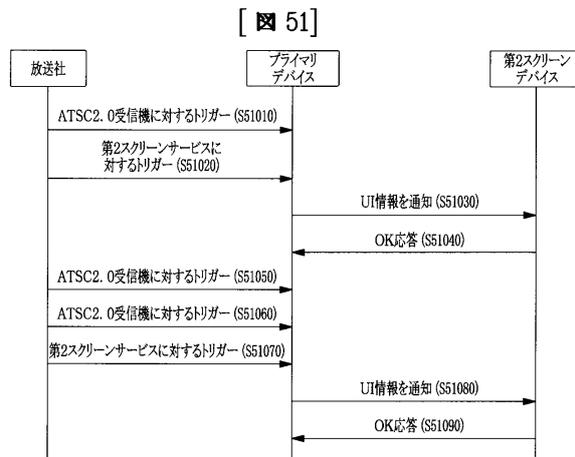
【 図 4 9 】



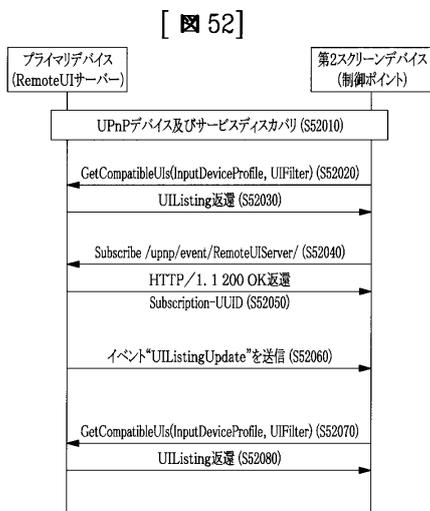
【 図 5 0 】



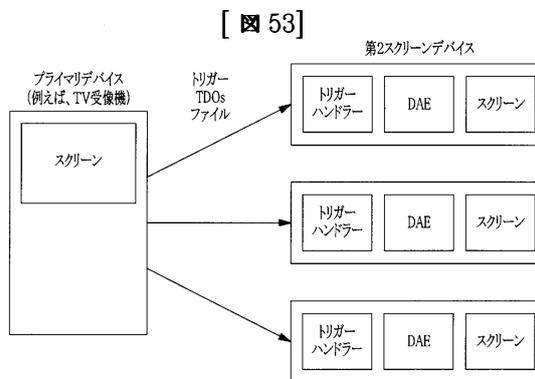
【 図 5 1 】



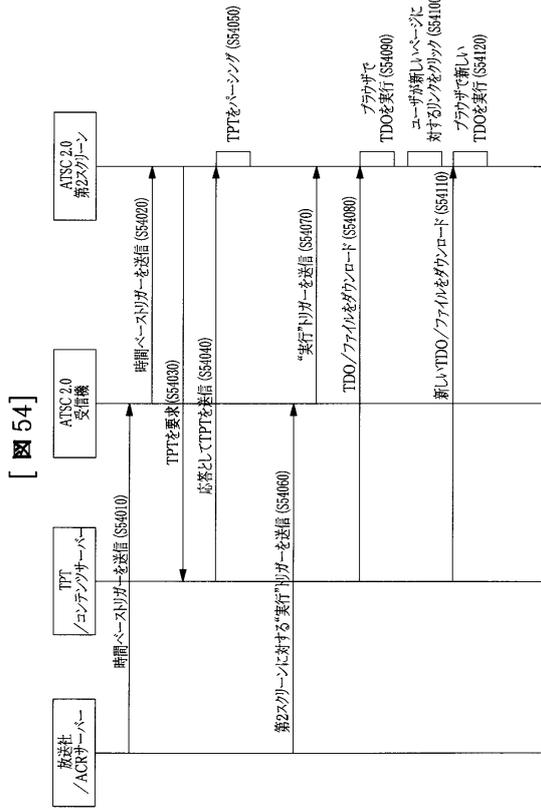
【 図 5 2 】



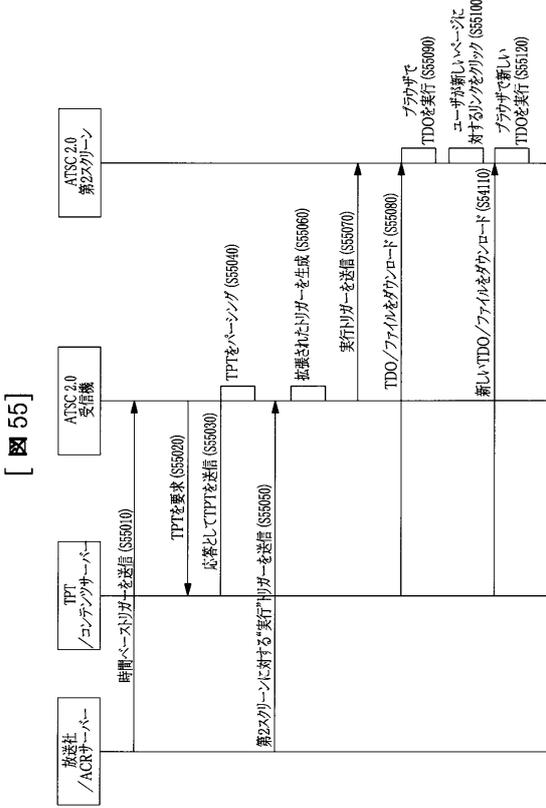
【 図 5 3 】



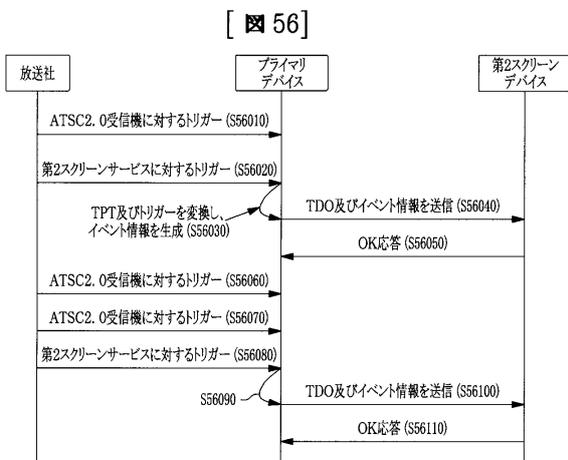
【 図 5 4 】



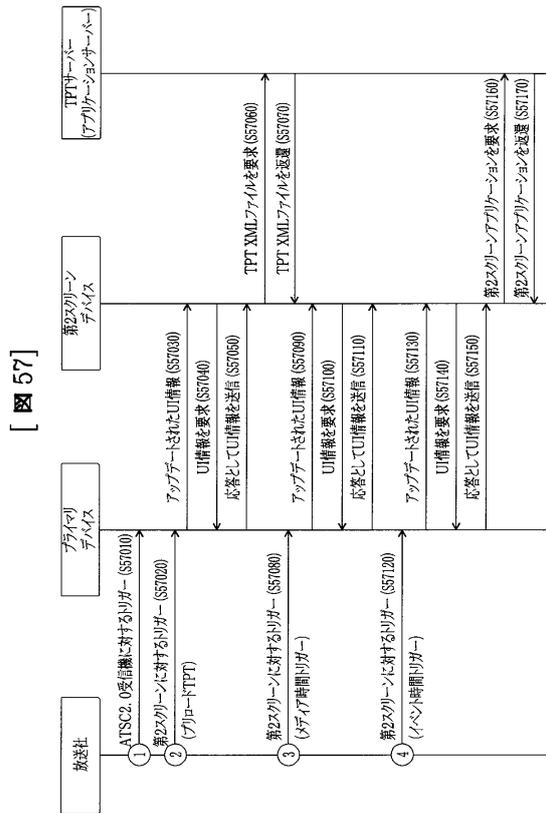
【 図 5 5 】



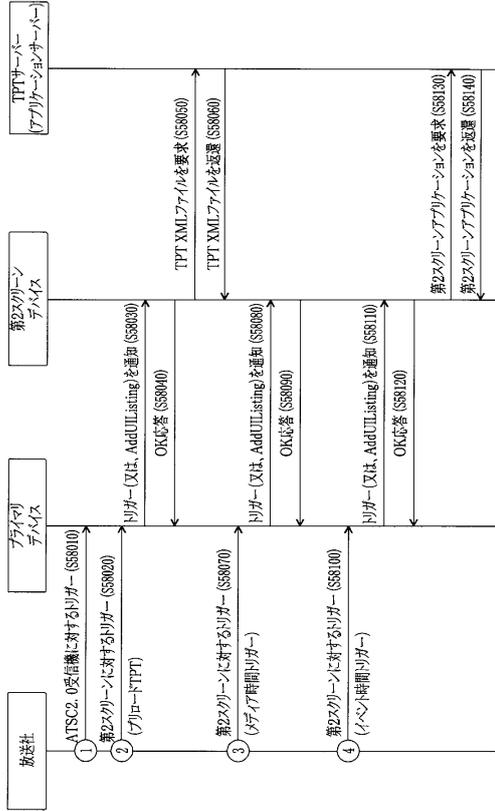
【 図 5 6 】



【 図 5 7 】

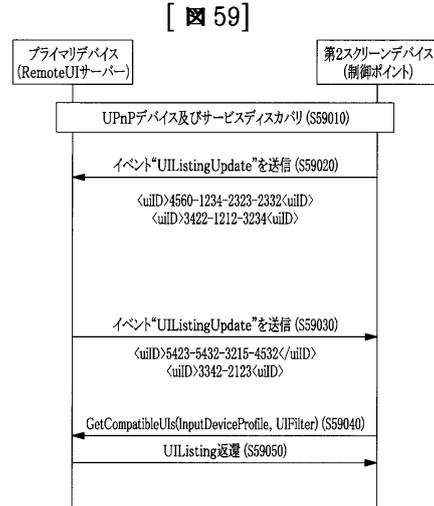


【 図 5 8 】



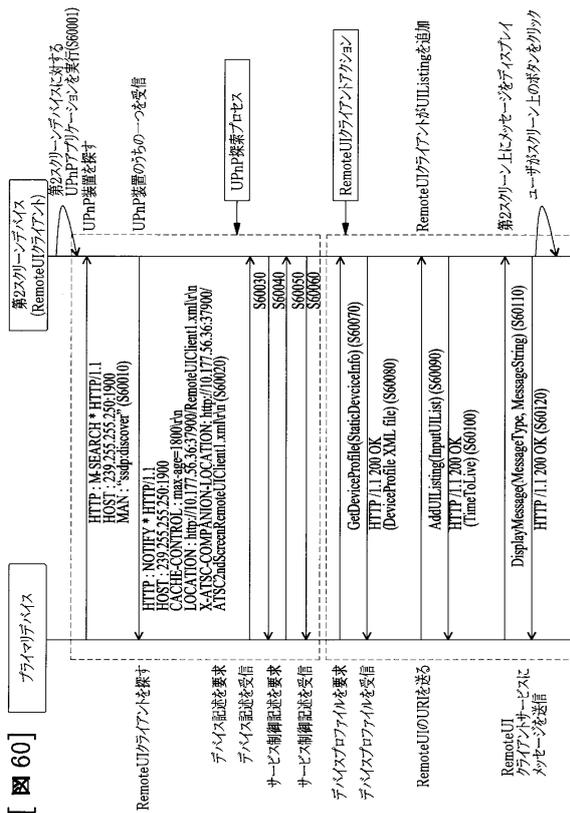
【 図 5 8 】

【 図 5 9 】



【 図 5 9 】

【 図 6 0 】



【 図 6 0 】

【 図 6 1 】

```

<?xml version='1.0' encoding='UTF-8'?>
<xs:schema xmlns="urn:schemas-app.org:remoteui:deviceprofile-1.0"
xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified" id="deviceprofile">
<xs:element name="deviceprofile">
<xs:complexType>
<xs:sequence>
<xs:element name="maxHoldUI" type="xs:string" default="0" minOccurs="0"/>
<xs:element name="protocol" minOccurs="unbounded"/>
<xs:complexType>
<xs:sequence>
<xs:element name="protocolInfo" type="xs:anyType" minOccurs="0" maxOccurs="1"/>
<xs:sequence>
<xs:attribute name="shortName" type="xs:string" use="required" form="unqualified"/>
<xs:complexType>
<xs:sequence>
<xs:complexType>
</xs:element>
</xs:schema>

```

【 図 6 2 】

```

<?xml version='1.0' encoding='UTF-8'?>
<deviceprofile xmlns="urn:schemas-app.org:remoteui:deviceprofile-1.0"
xmlns:xs="http://www.w3.org/2001/XMLSchema" instance">
<maxHoldUI>0</maxHoldUI>
<protocol shortName="HTTPHTML">
<protocolInfo>
<protocolInfo>
<protocolInfo>
<protocolInfo>
</protocolInfo>
</protocolInfo>
</deviceprofile>

```

【 図 6 3 】

[図 63]

タグ名	変数のタイプ	定義	説明
version	ストリング	プロトコルバージョン	これは、第2スクリーンデバイスのプロファイルのバージョンを示す。プロファイルの少なくとも一つがアップデートされる時に値が変更される。UIは、プロトコルの低いバージョンと互換可能である。
displayWidth	符号無し整数	幅(ピクセルの数)	第2スクリーンの幅解像度
displayHeight	符号無し整数	高さ(ピクセルの数)	第2スクリーンの高さ解像度
imageEncoding	ストリング	MIMEタイプ又はファイル拡張子	第2スクリーンデバイスによって受信され、ディスプレイされ得るイメージ
serviceProfile	ストリング	支援する標準プロファイルのネーム(ATSC2. 0、ATSC2. 1、...)	第2スクリーンプロファイルのネームによって支援され得るプロファイル
serviceType	ストリング	第2スクリーンアプリケーション、使用測定、など	これが第2スクリーンサービスであるか否かを示す。
tptVersion	ストリング	支援するTPTバージョン	第2スクリーンによって支援され得るTPTのバージョン
browserProfile	ストリング	支援するブラウザバージョン	HTML4、CSS、EMCAScriptのバージョンは、第2スクリーンによって支援されるブラウザプロファイルを用いて明示される。
supportTDO	ブール	TDOを動作可能な第2スクリーンがあるか(はい、いいえ)	第2スクリーンがTDOを動作及び処理できるか否かを明示する。
enableTriggerClient	ブール	トリガーを処理できるトリガークライアントモジュールが存在するか(はい、いいえ)	第2スクリーンがトリガーシグナル、パラメータ及びTDOパラメータテーブルを処理できるモジュールを有するか否かを明示する。

【 図 6 4 】

[Fig. 64]

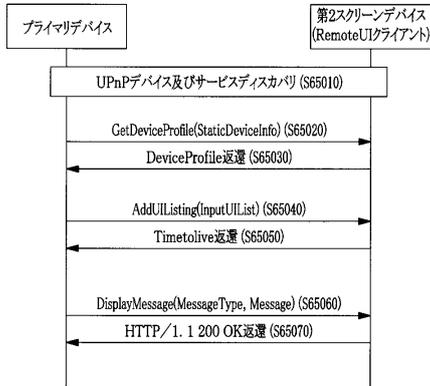
```

<?xml version="1.0" encoding="UTF-8">
<ulist xmlns:ns="http://www.w3.org/2001/XMLSchema-instance">
  <ul>
    <ulID>456023233267-7642</ulID>
    <name>ATSC 2.0 Second Screen Service</name>
    <description>ATSC 2.0 Second Screen Application</description>
    <iconList>
      <icon>
        <mime-type>image/png</mime-type>
        <width>160</width>
        <height>160</height>
        <depth>24</depth>
        <url>icon160.png</url>
      </icon>
    </iconList>
    <lifetime>1</lifetime>
    <protocolName>HTTP:HTML</protocolName>
    <url>http://1.8.7.2.333/SecondScreen/XBC/programA.html</url>
    <serviceProfileInfo>
      serviceProfile=ATSC2.0.serviceType=SecondScreen.displayWidth=640.displayHeight=480.version=1.0
    </serviceProfileInfo>
    </ul>
  </ulist>

```

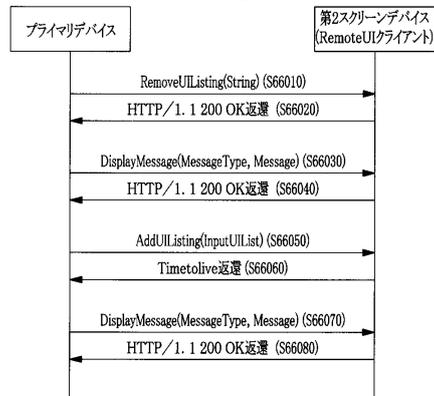
【 図 6 5 】

[図 65]

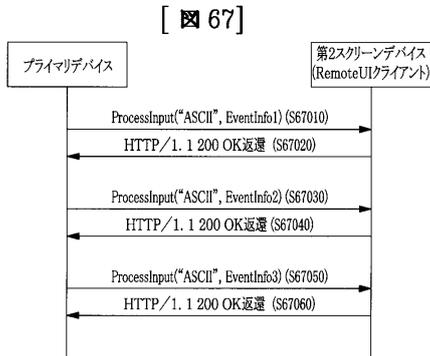


【 図 6 6 】

[図 66]



【図 67】



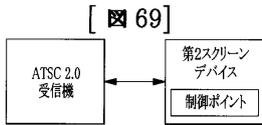
【図 67】

【図 68】

【図 68】

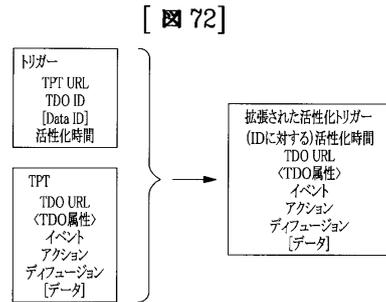
XML要素名/属性 (@と共に)	許容される番号	データタイプ	説明
TDO			
@appId	0..1	符号無し整数	トリガーによる参照のための第2スクリーンアプリケーションの固有識別子
@mediaTime	1	符号無し整数	受信機でコンテンツを再生するメディア時間
URL	1..N	任意のURI	TDO URL
Event	0..N	複雑数型	このTDOにリガされたイベント
@eventId	1	符号無しショート	トリガーによる参照用イベントID
@action	1	符号無しバイト	アクションコード(例えば、登録、実行、終了など)
Data	0..1	ベース64バイナリ	ストリームイベントのためにこのTDOによって用いられるデータ
@startTime	1	符号無し整数	@mediaTimeに関連した第2スクリーンサービスのイベントに対する有効期間の開始を示す
@endTime	0..1	符号無し整数	@mediaTimeに関連した第2スクリーンサービスのイベントに対する有効期間の終了を示す

【図 69】



【図 69】

【図 72】



【図 72】

【図 70】

サービス	サービスタイプ	サービスID
トリガー	atsc2.0trig:1	urn:atsc.org:serviceid:atsc2.0trig1
両方向通信	atsc2.0twocomm:1	urn:atsc.org:serviceid:atsc2.0twocomm1
AppURL	atsc2.0urls:1	urn:atsc.org:serviceid:atsc2.0urls1
HTTPプロキシサーバー	atsc2.0hps:1	urn:atsc.org:serviceid:atsc2.0hps1

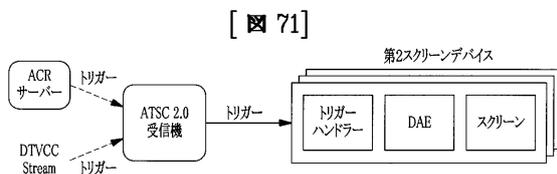
【図 70】

【図 73】

【図 73】

XML要素名/属性	基数	XMLデータタイプ	説明
AugmentedTrigger	1		
@interactionModel	1	符号無しバイト	トリガーと関連した相互作用モデル
@appURL	1	任意のURI	トリガーで識別されたアプリケーションのURI
@cookieSpace	0..1	符号無しバイト	必要な持続的な保存空間
Capabilities	0..1	nrt:CapabilitiesType	アプリケーションを提示するために必要な能力
Event	1		トリガーで識別されるイベント
@action	1	符号無しバイト	イベントのアクションフィールド
@destination	0..1	符号無しバイト	イベントによってターゲットになるデバイスのタイプ
@diffusion	0..1	符号無しバイト	イベントの活性化のために拡散された時間
@data	0..1	ベース64バイナリ	(存在すると)トリガーで識別されたデータ

【図 71】



【図 71】

【 図 7 4 】

[図 74]

	ビットの数	フォーマット	バイトの数
URI_data() {			
Reserved	4	bslbf	1
URI_type	4	uimsbf	
for (k=0; k<L-1; K++) {			1
URI_character	8	uimsbf	
}			

【 図 7 6 】

[図 76]

XMLエレメント/属性	基数	XMLデータタイプ	説明
Trigger	1		
@interactionModel	0..1	符号無しバイト	トリガーと関連した相互作用モデル
@triggerString	1	ストリング	トリガーストリング

【 図 7 5 】

[図 75]

URI_type	意味
0	インタラクティブTVトリガー
1	サービス使用報告サーバー(SURS)ロケータ
2-15	将来使用のために予約

【 図 7 7 】

[図 77]

XMLエレメント/属性	基数	XMLデータタイプ	説明
AugmentedTrigger	1		
@activationTime	1	符号無し整数	メディア時間スケール上での活性化時間
@tdoURL	1	任意のURI	トリガーで識別されたTDOのURI
@cookieSpace	0..1	符号無しバイト	必要な持続的な保存空間
Capabilities	0..1	nrt:Capabilitystyp	アプリケーションを提示するために必要な能力
Event	1		トリガーで識別されるイベント
@action	1	符号無しバイト	イベントのアクションフィールド
@destination	0..1	符号無しバイト	イベントによってターゲットとなるデバイスのタイプ
@diffusion	0..1	符号無しバイト	イベントの活性化のために拡散された時間
Data	0..1	ベース64バイナリ	(存在すると)トリガーで識別されたデータ

【 図 7 8 】

[図 78]

XMLエレメント/属性	基数	XMLデータタイプ	説明
AugmentedTrigger	1		
@activationTime	1	符号無し整数	メディア時間スケール上での活性化時間
@tdoURL	1	任意のURI	トリガーで識別されたTDOのURI
@cookieSpace	0..1	符号無しバイト	必要な持続的な保存空間
@availInternet	0..1	ブール	デフォルト="真"
@availBroadcast	0..1	ブール	デフォルト="真"
Capabilities	0..1	nrt:Capabilitystyp	アプリケーションを提示するために必要な能力
Event	1		トリガーで識別されるイベント
@action	1	符号無しバイト	イベントのアクションフィールド
@destination	0..1	符号無しバイト	イベントによってターゲットとなるデバイスのタイプ
@diffusion	0..1	符号無しバイト	イベントの活性化のために拡散された時間
Data	0..1	ベース64バイナリ	(存在すると)トリガーで識別されたデータ

【 図 7 9 】

[図 79]

XMLエレメント/属性	基数	XMLデータタイプ	説明
AugmentedTrigger	1		
@activationTime	1	符号無し整数	メディア時間スケール上での活性化時間
@tdoURL	1	任意のURI	トリガーで識別されたTDOのURI
@cookieSpace	0..1	符号無しバイト	必要な持続的な保存空間
Capabilities	0..1	nrt:Capabilitystyp	アプリケーションを提示するために必要な能力
ContentItem	0..N		このTDOによって用いられるコンテンツアイテム
URL	1..N	任意のURI	コンテンツアイテムのURI
@updatesAvail	0..1	ブール	デフォルト="偽"
@pollPeriod	0..1	符号無しバイト	秒単位の短いポーリング期間
@size	0..1	24ビット整数	キロバイト単位のコンテンツアイテムのサイズ
@availInternet	0..1	ブール	デフォルト="真"
@availBroadcast	0..1	ブール	デフォルト="真"
Capabilities	1	nrt:Capabilitystyp	アプリケーションを提示するために必要な能力
Event	1		トリガーで識別されるイベント
@action	1	符号無しバイト	イベントのアクションフィールド
@destination	0..1	符号無しバイト	イベントによってターゲットとなるデバイスのタイプ
@diffusion	0..1	符号無しバイト	イベントの活性化のために拡散された時間
Data	0..1	符号無しバイト	(存在すると)トリガーで識別されたデータ

【 図 8 0 】

[図 80]

活性化時間	1	符号無し整数	イベントの活性化時間
TDO	1		このセグメントに対するTDO
@appId	1	符号無しショート	トリガーによる参照のためのアプリケーションID
@appType	1	任意のURI	アプリケーションタイプ (デフォルト:1="TDO")
@appName	0..1	ストリング	(視聴者開始同意のための)ディスプレイ名
@globalId	0..1	任意のURI	全般的に固有のアプリケーションID
@appVersion	0..1	任意のURI	このアプリケーションのバージョン
@cookieSpace	0..1	符号無しバイト	必要な持続的な保存空間; デフォルト=0
@frequencyOfUse	0..1	整数、範囲0...15	コード値TBD
@expireDate	0..1	一時(date Time)	キャッシングのための満了日
@testTDO	0..1	ブール	テストTDOのためのフラグ; デフォルト="偽"
@availInternet	0..1	ブール	デフォルト="真"
@availBroadcast	0..1	ブール	デフォルト="真"
URL	1..N	任意のURI	TDO URL- 第1URIは エントリーポイント
Capabilities	0..1	nrt:CapabilitiesType	このTDOを提示する 必須の能力
ContentItem	0..N		このTDOによって用いられる コンテンツアイテム
URL	1..N	任意のURI	コンテンツアイテムのURI
@updatesAvail	0..1	ブール	デフォルト="偽"
@pollPeriod	0..1	符号無しバイト	秒単位の短いポーリング期間
@size	0..1	24ビット整数	キロバイト単位のコンテンツ アイテムのサイズ
@availInternet	0..1	ブール	デフォルト="真"
@availBroadcast	0..1	ブール	デフォルト="真"
Event	1		このTDOをターゲットとする イベント
@action	1	符号無しバイト	"登録", など;コード値TBD
@diffusion	0..1	符号無しバイト	秒単位のデフュージョンを 適用する期間
Data	0..1	符号無しバイト	ストリームイベントのためにこの TDOによって用いられるデータ

【 図 8 1 】

[図 81]

変数ネーム	必要/選択	データタイプ	変数ネーム	調整されたイベント	最小イベント 間隔
TrigServerURL	必要	ストリング	いいえ	N/A	N/A

【 図 8 2 】

[図 82]

ネーム	必要/選択
GetTrigServerURL	必要

【 図 8 3 】

[図 83]

引数	方向	関連した状態変数
TrigServerURL	OUT	TrigServerURL

【 図 8 4 】

[図 84]

変数ネーム	必要/選択	データタイプ	イベント?	調整された イベント	最小イベント 間隔
LatestUnfilteredTrigger	必要	ストリング	いいえ	N/A	N/A
LatestFilteredTrigger	必要	ストリング	いいえ	N/A	N/A
UnfilteredTriggerDeliveryTime	必要	日時	はい	いいえ	N/A
FilteredTriggerDeliveryTime	必要	日時	はい	いいえ	N/A

【 図 8 7 】

[図 87]

引数	方向	関連した状態変数
LatestFilteredTrigger	OUT	LatestFilteredTrigger

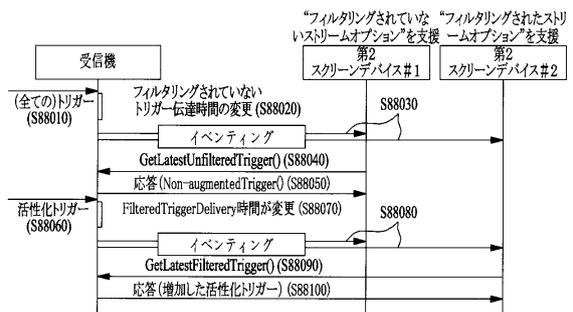
【 図 8 5 】

[図 85]

ネーム	必要/選択
GetLatestUnfilteredTrigger	必要
GetLatestFilteredTrigger	必要

【 図 8 8 】

[図 88]



【 図 8 6 】

[図 86]

引数	方向	関連した状態変数
LatestUnfilteredTrigger	OUT	LatestUnfilteredTrigger

【 図 8 9 】

[図 89]

変数ネーム	必要/選択	データタイプ	イベント	調整されたイベント	最小イベント間隔
CurrentTrigger	必要	ストリング	はい	はい	1秒

【 図 9 0 】

[図 90]

SetTrigger()

引数	方向	関連した状態変数
CurrentTrigger	IN	ATSCTrigger

SetExpandedTrigger()

引数	方向	関連した状態変数
CurrentTrigger	IN	ATSCExpandedTrigger

GetTrigger()

引数	方向	関連した状態変数
CurrentTrigger	OUT	ATSCTrigger

GetExpandedTrigger()

引数	方向	関連した状態変数
CurrentTrigger	OUT	ATSCExpandedTrigger

【 図 9 4 】

[図 94]

XMLエレメント/属性

XMLエレメント/属性	基数	XMLデータタイプ	説明
ClientList	1		登録されたクライアントのリスト
Client	0..N		単一登録クライアント
@name	1	ストリング	クライアントのネーム
@ipAddress	1	符号無し整数	クライアントウェブソケットサーバーのIPアドレス
@tcpPort	1	符号無しバイト	クライアントウェブソケットサーバーのIPアドレス

【 図 9 5 】

[図 95]

変数ネーム	必要/選択
SignUp	必要
SignOff	必要

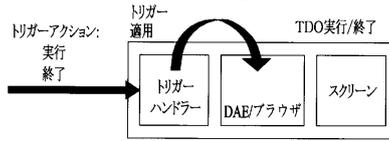
【 図 9 6 】

[図 96]

引数	方向	関連した状態変数
Name	IN	ネーム
IPAddress	IN	IPアドレス
TCPPort	IN	TCPポート

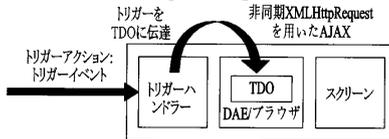
【 図 9 1 】

[図 91]



【 図 9 2 】

[図 92]



【 図 9 3 】

[図 93]

変数ネーム	必要/選択	データタイプ	イベント?	調整されたイベント	最小イベント間隔
ClientList	必要	ストリング	いいえ	N/A	N/A
Name	選択	ストリング	いいえ	N/A	N/A
IPAddress	選択	ストリング	いいえ	N/A	N/A
Port	選択	ストリング	いいえ	N/A	N/A

【 図 9 7 】

[図 97]

引数	方向	関連した状態変数
Name	IN	ネーム

【 図 9 8 】

[図 98]

変数ネーム	必要/選択	データタイプ	イベント?	調整されたイベント	最小イベント間隔
ConnectionAddress	必要	ストリング	はい	いいえ	N/A
Status	必要	ブール	はい	いいえ	N/A

【 図 9 9 】

[図 99]

変数ネーム	必要/選択	データタイプ	イベント?	調整されたイベント	最小イベント間隔
onBytesReceived	必要	ストリングアドレス, ストリングバイト	はい	いいえ	N/A

このコールバック関数は、両方向通信サービスを通じてDOに対するバイトを受信すると呼び出される。2個の引数は次のように定義される。
 ストリングアドレス - {address}:{port}フォーマットの受信されたバイトの送信者のIPアドレス
 アドレス及びTCPポートを含むストリング
 ストリングバイト - 受信されたバイト

【 図 1 0 0 】

[図 100]

void setStatusYes()	
説明	両方向通信サービスのUPnPプール状態変数“Status”の値を、DOが通信に参加する準備ができていないことを示す“真”に設定
引数	無し

【 図 1 0 1 】

[図 101]

void setStatusNo()	
説明	両方向通信サービスのUPnPプール状態変数“Status”の値を、DOが通信に参加する準備ができていないことを示す“偽”に設定
引数	無し

【 図 1 0 2 】

[図 102]

void setStatus(String state)		
説明	両方向通信サービスのUPnP状態変数“Status”の値を設定	
引数	状態	DOが第2スクリーンデバイスとの両方向通信に参加できると“はい”に設定し、そうでないと“いいえ”に設定

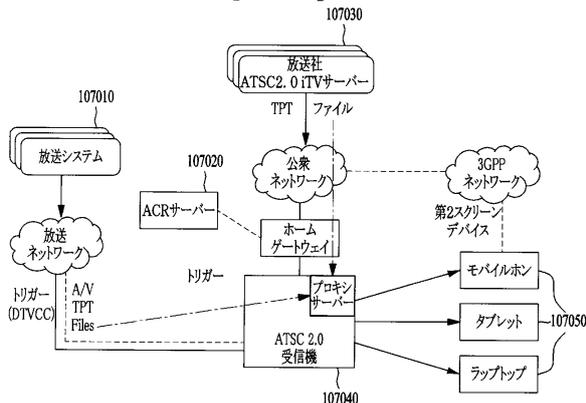
【 図 1 0 6 】

[図 106]

引数	方向	関連した状態変数
AppURL	OUT	AppURL
AppName	OUT	AppName

【 図 1 0 7 】

[図 107]



【 図 1 0 3 】

[図 103]

void sendBytes(String address, String bytes)		
説明	両方向通信サービスを用いてバイトを送信	
引数	アドレス	<address>:<port>フォーマットのバイトに対するあて先TCP/IPアドレス及びポート
	バイト	伝送されるバイト

【 図 1 0 4 】

[図 104]

変数名	必要/選択	データタイプ	イベント?	調整されたイベント	最小イベント間隔
AppURL	必要	ストリング	はい	N/A	N/A
AppName	必要	ストリング	はい	N/A	N/A

【 図 1 0 5 】

[図 105]

ネーム	必要/選択
GetAppURL	必要

【 図 1 0 8 】

[図 108]

変数名	必要/選択	データタイプ	イベント?	調整されたイベント	最小イベント間隔
ProxyServerURL	必要	ストリング	いいえ	N/A	N/A

【 図 1 0 9 】

[図 109]

ネーム	必要/選択
GetProxyURL	必要

【 図 1 1 0 】

[図 110]

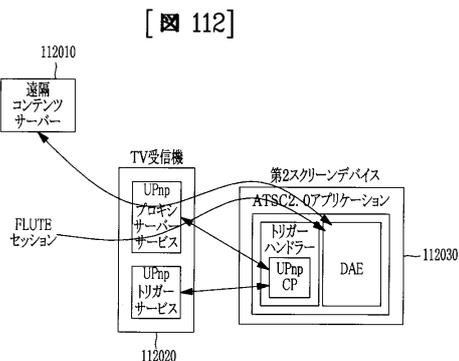
引数	方向	関連した状態変数
ProxyURL	OUT	ProxyServerURL

【 図 1 1 1 】

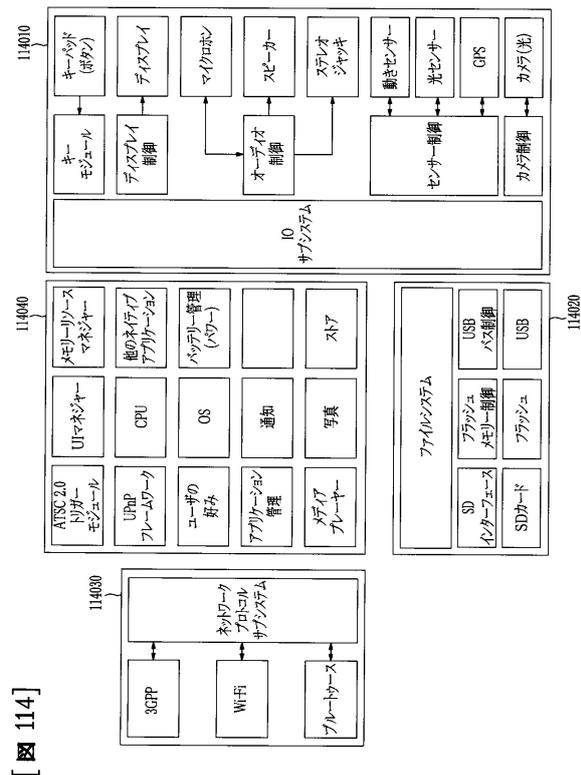
[図 111]

引数	方向	関連した状態変数
FileList	IN	ATSCFileList

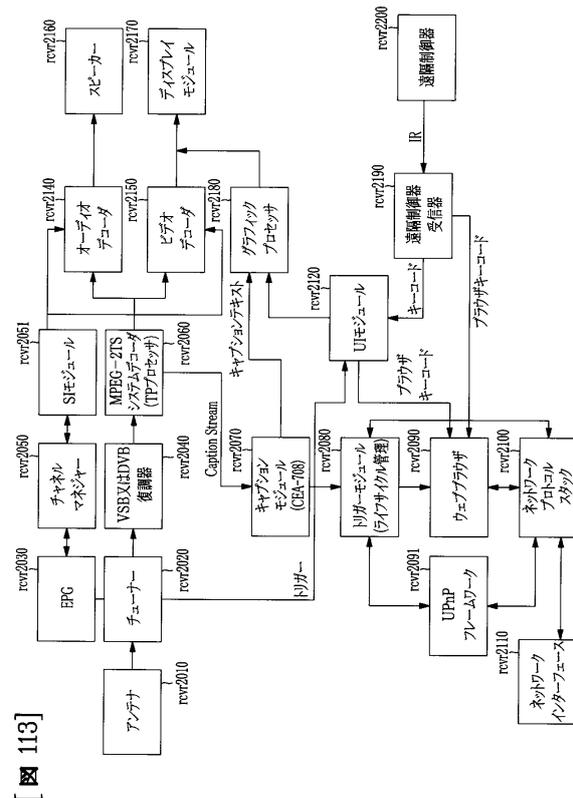
【 図 1 1 2 】



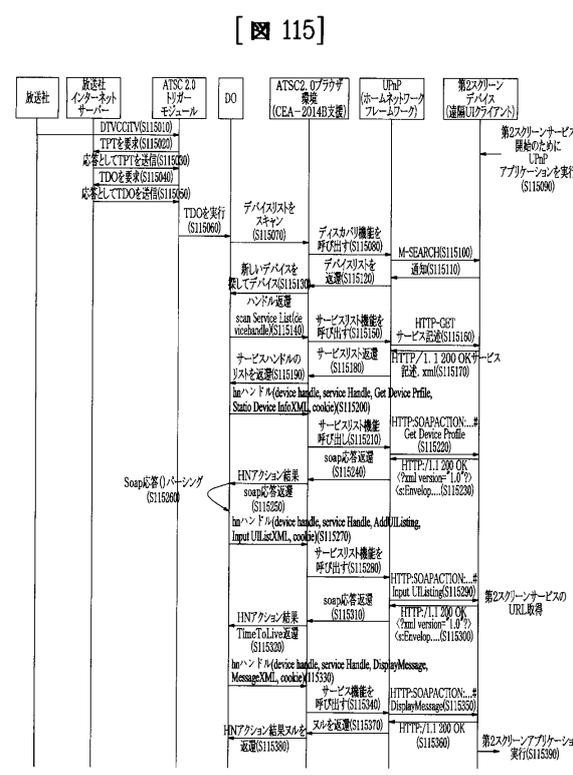
【 図 1 1 4 】



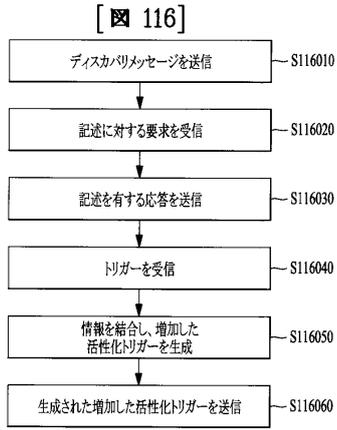
【 図 1 1 3 】



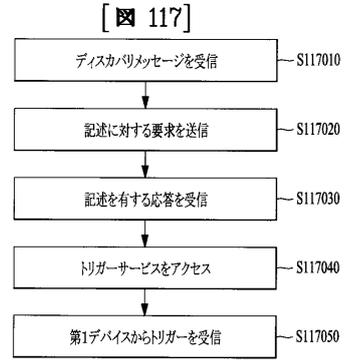
【 図 1 1 5 】



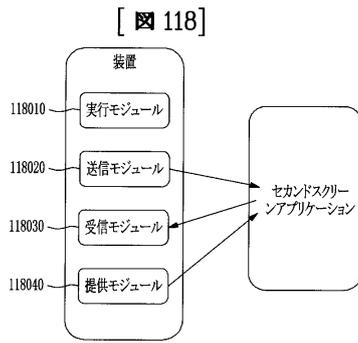
【図116】



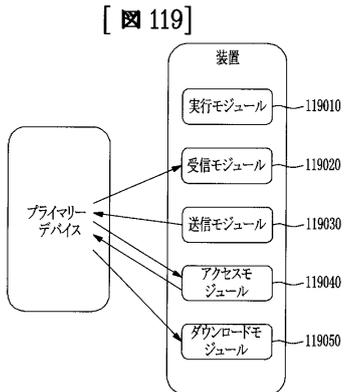
【図117】



【図118】



【図119】



フロントページの続き

- (72)発明者 オ セチン
大韓民国, ソウル 137-724, ソチョ-ク, ウーミョン-ドン 16, エルジー エレクトロニクス インコーポレイティド, インテレクチュアル プロパティ センター
- (72)発明者 ヤン スンリュル
大韓民国, ソウル 137-724, ソチョ-ク, ウーミョン-ドン 16, エルジー エレクトロニクス インコーポレイティド, インテレクチュアル プロパティ センター
- (72)発明者 キム チンピル
大韓民国, ソウル 137-724, ソチョ-ク, ウーミョン-ドン 16, エルジー エレクトロニクス インコーポレイティド, インテレクチュアル プロパティ センター
- (72)発明者 ペ チャンフン
大韓民国, ソウル 137-724, ソチョ-ク, ウーミョン-ドン 16, エルジー エレクトロニクス インコーポレイティド, インテレクチュアル プロパティ センター
- (72)発明者 アン スンチュ
大韓民国, ソウル 137-724, ソチョ-ク, ウーミョン-ドン 16, エルジー エレクトロニクス インコーポレイティド, インテレクチュアル プロパティ センター
- (72)発明者 キム キョンホ
大韓民国, ソウル 137-724, ソチョ-ク, ウーミョン-ドン 16, エルジー エレクトロニクス インコーポレイティド, インテレクチュアル プロパティ センター
- (72)発明者 ムン キョンソ
大韓民国, ソウル 137-724, ソチョ-ク, ウーミョン-ドン 16, エルジー エレクトロニクス インコーポレイティド, インテレクチュアル プロパティ センター

審査官 梅本 達雄

(56)参考文献 欧州特許出願公開第02343881(E P, A1)

(58)調査した分野(Int.Cl., DB名)

H04N	7/10		
H04N	7/14	-	7/173
H04N	21/00	-	21/858