Synchronized Interactive Services for Mobile Devices over IPDC/DVB-H and UMTS

Philip Leroux, Vincent Verstraete, Filip De Turck and Piet Demeester

Ghent University
Department of Information Technology - IBBT
Gaston Crommenlaan 8 Bus 201
9050 Gent, Belgium

Abstract. DVB-H (Digital Video Broadcasting - Handheld) is a technical specification for bringing broadcast services to handheld receivers. IP Datacast integrates DVB-H in a hybrid network structure consisting of both a mobile communications network such as GPRS or UMTS and an additional DVB-H downstream. This paper introduces two different approaches for providing interactive services to DVB-H terminals. It is assumed that the terminals dispose of a MIDP compliant runtime environment. The Mobile Information Device Profile (MIDP) is a J2ME profile for the use of Java on embedded devices such as mobile phones and PDAs. It is shown how these services can be synchronized with the multimedia data that is broadcasted over the DVB-H channel. A first approach will only use the DVB-H broadcast channel as bearer of both the audiovisual data and the interactivity-related data. The second approach combines the broadcast path with a bidirectional unicast channel in order to implement the interactive services.

1 Introduction

In November 2004, the DVB consortium released the DVB-H (Digital Video Broadcasting- Handheld) standard [1]. This broadcast standard provides an efficient way of carrying multimedia services over digital terrestrial broadcasting networks to handheld terminals. Special requirements for this type of devices (low energy consumption, robust transmission, etc.) have been taken into account. The first commercial DVB-H services have already been rolled out. IP Datacasting (IPDC) combines digital content formats, software applications, programming interfaces and multimedia services are combined through IP with a digital broadcasting environment. An inherent part of such a datacast system is that it comprises of a unidirectional DVB broadcast path that may be combined with a bi-directional mobile/cellular interactivity path. IPDC can be used for enabling the convergence of services from broadcast/media and telecommunications domains. A typical IPDC over DVB-H terminal is a mobile phone with built-in DVB-H receiver and UMTS antenna. On most mobile phones a MIDP [2] compliant runtime environment is installed. The Mobile Information Device Profile (MIDP) is a J2ME profile for the use of Java on embedded devices such as

1-4244-1297-8/07/$25.00 ©2007 IEEE

mobile phones and PDAs. It is the de facto specification for application development when targeting different types of mobile phones. This paper introduces two different approaches for providing interactive services to DVB-H terminals that (only) dispose of a MIDP compliant runtime environment. Instead of user-device interactivity, this paper will focus on client-server interaction. These services may also be synchronized with the audio and video (A/V) data that is broadcasted over the DVB-H channel. Such services may enlarge the user's experience of mobile television by offering interactivity that is closely related or even frame accurately synchronized with what is happening on TV. A first approach will only use the DVB-H broadcast channel as bearer of both the A/V data and the interactivity-related data. The second approach combines the broadcast path with a mobile interactivity path such as UMTS in order to implement all the required services. In the presented approaches, several concepts of the Multimedia Home Platform (MHP) [3] standard have been used and translated to the "DVB-H/MIDP" environment. MHP is an open middleware standard designed by the DVB project for interactive digital television. The main objective however was not to present a generic MHP-compliant solution, but a client-server architecture which can be easily integrated into a DVB-H system. The development of the presented solutions is still ongoing and is part of the MADUF (MAximizing DVB Usage in Flanders) project [4]. The objective of MADUF is to generate an optimum model for providing mobile television services in Flanders via DVB-H and developing a proof of concept setup.
This paper is structured as follows: section 2 states some related work. In section 3 we introduce IP Datacasting over DVB-H and in section 4 the functional requirements for mobile interactivity are listed. The setup and concepts that are used to fulfill these functional requirements are described in section 5. Finally, section 6 states our conclusions and future work.

2 Related Work

Several projects are trying to define or implement synchronized, interactive services for broadcast and convergent networks.

The INSTINCT (IP-based Networks, Services and TermINals for Convering systTems) project [5] is a European project in line with the objectives of the newly established DVB-CBMS activities. It is committed to assist DVB in realising the commercial provision of convergent services in mobility with a special focus on the DVB-T, DVB-H and DVB-MHP standards in conjunction with the concept of wireless communications networks (notably GPRS and UMTS) combined with terrestrial DVB broadcast networks. The INSTINCT activities include the definition and implementation of the network infrastructure, the user devices and the generation of content, services and applications.

SAVANT (Synchronised and Scalable AV content Across NeTworks) [6] was a European Union co-funded project that was aiming at demonstrating novel types of services, which can be achieved when cross-linking different networks in the Internet as well as the broadcast domain. SAVANT has chosen DVB-

MHP, ISO/IEC MPEG (MPEG-4 & MPEG-7), IETF and W3C technologies as its cornerstones and demonstrated the capabilities of these standards in implementing novel services. Within this project, which has already been terminated, an end-to-end solution has been developed which is able to synchronize content delivered over an IP network with a TV programme delivered over DVB.

The porTiVity (portable interacTiVity) project [7] is developing some of the technical achievements of SAVANT. It develops and experiments a complete end-to-end platform providing Rich Media Interactive TV services for portable and mobile devices, realising direct interactivity with moving objects on handheld receivers connected to both a broadcast and unicast channel. The work that is presented in this paper differs from these projects as it is especially focussing on a IPDC over DVB-H setup for MIDP mobile devices.

3 Introduction to IP Datacasting over DVB-H

3.1 The DVB-H Broadcast Specification

DVB-H is a technical specification for bringing broadcast services to handheld receivers. It adapts the successful DVB-T (Digital Video Broadcasting - Terrestrial) system for digital terrestrial television to the specific requirements of handheld, battery-powered receivers. The most remarkable enhancements are:

- **Time Slicing**
  By transmitting chunks of data in bursts (using a high instantaneous bitrate), time-slicing allows the receiver to be switched off in inactive periods. This results in power savings of up to 90% while the same inactive receiver can be used to monitor neighboring cells for seamless handovers. Time-slicing is mandatory for DVB-H.
- **4K transmission mode**
  The additional 4K transmission mode is an intermediate mode between DVB-T's 2K and 8K modes. It aims to offer an additional trade-off between single frequency network (SFN) cell size and mobile reception performance, providing an additional degree of flexibility for network planning.
- **MPE-FEC**
  The objective of Multi-Protocol Encapsulation - Forward Error Correction (MPE-FEC) is to improve the mobile channel tolerance to impulse interference and Doppler effect. This is accomplished through the introduction of an additional level of error correction at the MPE layer. By adding parity information calculated from the datagrams and sending this parity data in separate MPE-FEC sections, error-free datagrams can be output (after MPE-FEC decoding) even under bad reception conditions.

3.2 IP Datacasting over DVB-H

IP Datacasting is a service where digital content formats, software applications, programming interfaces and multimedia services are combined through IP with digital broadcasting. Together with DVB-H, IP Datacast defines a fully-specified end-to-end solution with transport protocols, handover and signalling mechanisms, data structures, etc. With the optional bidirectional channel integration, IPDC can be used for enabling the convergence of services from broadcast/media and telecommunications domains.

The protocolstack of an IP Datacast over DVB-H system is shown in Fig.1. The integration of an IP layer in the broadcaststack is one of the key concepts of an IPDC system. These IP datagrams are encapsulated inside the MPEG Transport Stream (TS) using MPE to improve mobile performance. For the delivery of streaming media, IP Datacast specifies the use of the Real-time Transport Protocol (RTP) [8] and file delivery is performed by using File Delivery over Unidirectional Transport (FLUTE) [9]. FLUTE is a protocol for unidirectional delivery of files over the Internet.

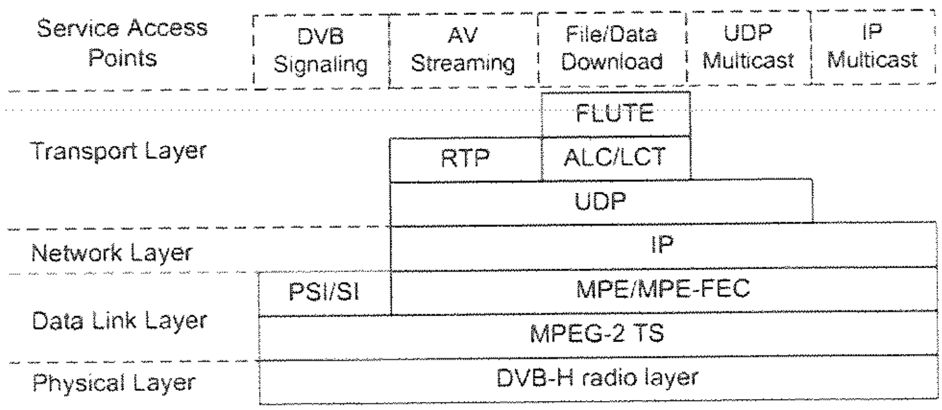

Fig. 1. The IPDC over DVB-H protocol stack.

4 Functional Requirements for Mobile Interactivity

4.1 Interactivity on Mobile Devices

The presented solutions are specifically targeting mobile devices which have limited computational resources. There are three general approaches to interactivity on such devices:

- **Generic Browser**
  When using a generic browser, all interactivity is defined declaratively in the broadcast stream. The standardized information elements (such as text fields, buttons, logos, pictures) can be customized and arranged in simple, predefined layouts and all elements can be synchronized with the A/V stream. Only a generic rendering application is required on the terminal, as all aspects of the application are defined in the broadcast stream. This type of interaction is easy and inexpensive to deploy and frequent changes and relaunches are easy to handle.

- **Rich Media Streaming**
  Rich-media is content composed of audio, video, graphics and text with user interactivities and dynamic updates. Again, the application is entirely defined in the broadcast stream, all elements can be synchronized with the A/V stream and only a generic runtime component is required on the terminal. This solution provides more powerful control over graphics, animation and interaction. An example of this type of mobile interaction is the MPEG LASeR (Lightweight Application Scene Representation) standard [10].
- **Downloadable (Java) applications**
  This approach uses fullfeatured applications with complex logic which allow full control over the interaction. These applications may be downloaded over the broadcast channel or via the interaction channel, they are expensive to develop and must be verified against all terminal types the operator wants to support for the application.

In order to support a maximum of functionality, the presented solutions are based on the latest approach, specifically downloading applications on the user's device. As already mentioned in the introduction, we will only focus on applications that are compliant with the MIDP specification.

4.2 Functional Requirements

First of all, the applications that will provide the interactive services should be linked to one or more television programs and a signalling mechanism is required to inform the user that a program is available. A transport mechanism is also required to transmit these file-based objects to the mobile terminal.

In order to manage these downloaded applications a middleware framework should be installed on the user's device. This framework shoud be able to detect applications that are available. These applications must then be loaded in memory, and started when requested by the user or when the application is linked to the television program that is currently playing on the user's device. When the service is no longer needed, the applications should be stopped and (if necessary) removed from the memory to free space.

Without enhancements, the (stand-alone) applications will have no immediate coupling with the audiovisual data that is broadcasted. This makes them initially not suitable to support synchronized interactivity, i.e. generating actions that occur on the user's device when a specific scene (or frame) is shown on the screen. To support this type of interactivity a mechanism will be needed that synchronizes the A/V data stream with the interactive applications that are running on the mobile device.

By providing a synchronization mechanism, the applications can perform the right actions at the right time. However, they still need to know which action this should be and when exactly it should be performed. A static solution would be the integration of these parameters into the application logic. But this would not be suitable for live shows and it would also raise development costs. For instance, an application that is linked to a daily quiz should have to be adapted and recompiled on a daily basis. And the user would also experience more overhead (re-downloading, reinstalling, etc.). Therefore, a mechanism is required to inform the client-applications about when they should perform a specific action. Along with this information, sending extra data to perform an action should also be possible.

As a summary, we can state that the solution for providing synchronized, interactive services through downloadable MIDP applications should support:

1. Signalling in the broadcast stream in order to link the applications to certain television programs.
2. A transport mechanism for delivering the applications to the handheld terminals.
3. Middleware on the user's device to load, start, stop and unload applications.
4. A mechanism that synchronizes the downloaded applications with the broadcasted content.
5. A server-side mechanism that can steer all client applications in order to perform the right actions at the right time.

5 Integrating Interactive Services into a IPDC over DVB-H System

5.1 Setup Description

In order to fulfill the functional requirements that where listed in Sec.4.2, a setup is built as is shown in Fig. 2. This architecture can easily be mapped to the functional architecture that is specified in the DVB-CBMS specification [11]. Starting from a general explanation of the work flow of this architecture, we will go into more detail about how the functional requirements are met by this setup.

The content provider streams its multimedia content to the encoders that trans/encode both audio and video to the requested format (e.g. for video this will usually be the H.264/AVC format). Afterwards, these encoders packetize both the encoded audio and the video stream into RTP packets. Besides these RTP streams, the encoders may also provide extra information concerning these streams by generating a Session Description Protocol (SDP) [12] file for every session. These files are used by both the ESG server and our interactivity server. The generated RTP streams are sent over a multicast IP network to the IP Encapsulator. There, the incoming IP datagrams are encapsulated into datagram sections which are compliant to the DSM-CC [13] section format for private data.

The content provider will offer its scheduling information to the Electronic Service Guide (ESG) server by building up a schedule on the server via a graphical user interface or by sending the data in an appropriate format to the server.

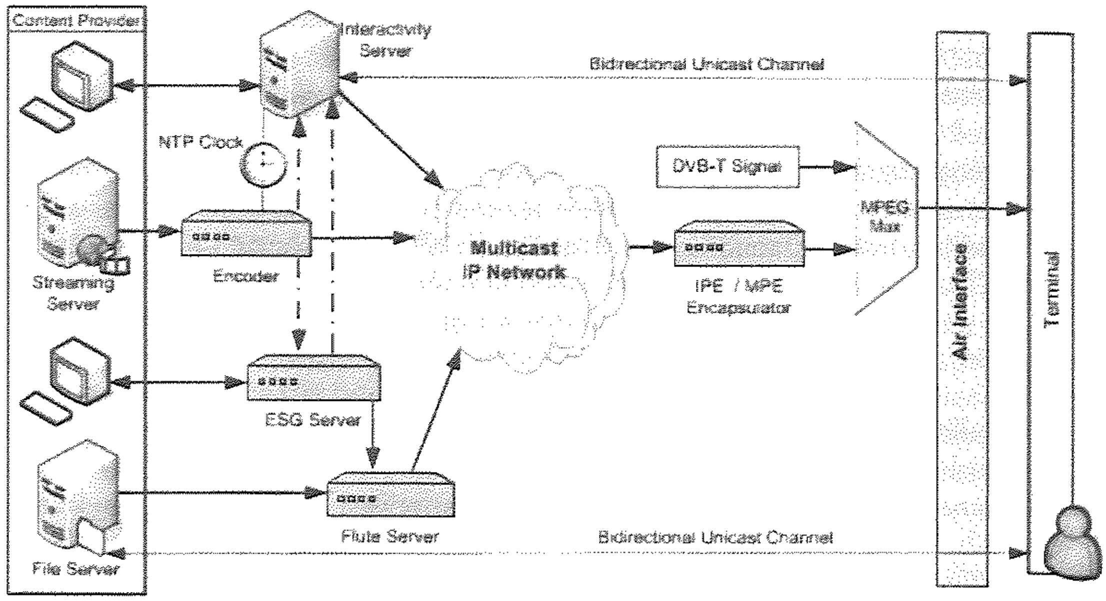

Fig. 2. A DVB-H over IPDC setup extended with interactivity components.

By combining the input of the content provider and the extra info that is generated by the encoders, the ESG server builds a DVB-H over IPDC compliant ESG [14].

Both the ESG data that is generated by the server and the extra data (non streaming data such as files) that is delivered by the content provider are delivered to the end user by using the FLUTE data carousel.

For the signalling of data services in the DVB signal, Program Specific Information (PSI) and Service Information (SI) will be added to the transport stream. PSI consists of data enabling a decoder to demultiplex DVB services. In addition to the PSI, SI data is needed to provide identification of DVB services to the user. PSI and SI information are often generated in the IP encapsulator. For more information concerning the use of these tables in a IPDC over DVB-H system we refer to [15].

The interactivity server is the central component for the delivery of the synchronized, interactive services. This server is able to generate data streams to the multicast IP network and it can also handle incoming client requests in order to send all the data via a unicast channel. The next sections detail about how the interactive services are provided.

5.2 Signalling and Transportation of the Applications

Linking applications to a specific service can be done in the ESG. The IP Datacast ESG provides a flexible data structure for a large number of use cases. In order to deliver Java-based software applications, the extensions that are proposed in [16] may be used.

As already mentioned, we present two solutions for delivering synchronized interactive services to a DVB-H terminal. The first solution uses the FLUTE data carousel in order to send the applications to the broadcast channel. To download and install these applications, the application manager on the terminal should be able to access this carousel. The second solution uses a bidirectional unicast connection to transport the applications. In the proposed setup, this is a UMTS network connection as this offers sufficient band width for a fast delivery of MIDP applications. In order to install new application logic via this unicast connection, the application manager only needs to invoke a MIDP *platformrequest* to the URI that refers to the requested application.

5.3 Action Mechanism

Analogue to the MHP specification, we use an event-trigger mechanism for steering all the client applications. An event is the description of a specific action that can be executed by a downloaded application. A trigger will inform the application logic on the client device when a specific event needs to be executed. There are two types of triggers:

- Do-It-Now Event Triggers where the associated event occurs immediately.
- Scheduled Event Triggers, these triggers contain a clock value that indicates when the event has to occur. They are sent in advance, and possibly more than once.

Additionally, a trigger may contain extra data that is needed to execute an event. The Interactivity Server that is shown on Fig.2 generates the appropriate triggers and events, based on a simple description of the content provider. Contrary to the MHP solution, the interactive data is not sent via the data carousel but depending on the solution that is used:

- The server will generate a multicast IP socket for every service that may support interactivity. The events and triggers are thus delivered to the user as a separate DVB-H service.
- The interactivity related data is sent via a bidirectional unicast channel to the client devices. Each client has to specifically request such a unicast connection.

Both solutions have the advantage that the on time arrival of triggers is not subject to the size of the data carousel. In order to work on terminals that "logged in" after the start of a television show, the server will resent all required information in well defined cycles.

With regard to linking the data that is generated by the interactivity server to a specific service, extra information should be provided in the ESG. When using the broadcast solution, the interactive service should be linked to the multimedia content service. In case of the unicast solution, the IP address and port number of the interactivity server should be provided in the ESG.

5.4 Client Middleware for Application Management

As already mentioned, a MIDP application (called a MIDlet) is used to provide the interactive services. However, the currently available MIDP profile MIDP 2.0 imposes some serious restrictions:

- There is no support for the Java reflection mechanism.
- Inter-MIDlet communication and auto-launching of MIDlets are very hard to implement.
- Running multiple concurrent MIDlets in the same VM is only possible if these MIDlets are part of the same group (called a MIDlet suite).
- Due to the severe security restraints of MIDP's virtual sandbox, the MIDlets cannot be installed without requiring the user to intervene several times.

Because of the absence of a reflection mechanism or an existing OSGi [17] framework on a MIDP terminal, it is impossible to add application logic to an already installed framework or application. This has a major impact on the architecture of the client's application manager. Instead of running a manager that may be extended with extra functionality, new application logic is provided to the user by means of a fully installable MIDP application. Fortunately, MIDP 3 (JSR 271) [18], which is still in early stage of development, will tackle most of the issues that are listed above. Another important specification that is under development is JSR 272: "Mobile Broadcast Service API for Handheld Terminals"[19]. This specification will define an optional package in a J2ME/MIDP/CLDC environment to provide functionality to handle broadcast content, e.g. to view digital television and to utilize its rich features and services. In order to implement a fully working application manager that is able to load the requested applications or interpret the broadcasted ESG information, the usage of JSR 272 will be required.

5.5 Synchronization Mechanism

RTP Synchronization Overview

RTP provides tools for synchronizing the seperate audio and video stream that are generated by the encoders. For that purpose, an RTP time stamp is present in the RTP header. As RTP's synchronization mechanism will be used by the interactivity server to link events to audio and video data, it will be explained first:

- RTP time stamps convey the sampling instant of access units at the encoder. An access unit is the smallest entitiy to which timing information can be attributed. The RTP time stamp is expressed in units of a clock, which is required to increase monotonically and linearly. The frequency of this clock is specified for each payload format. We assume that the frequency of the sampling clock is used. In Fig.3, TSa(i) and TSv(j) are RTP time stamps that are used to present the access units at the correct timing at the receiver. This requires that the receiver reconstructs the video clock and audio clock with the same mutual offset in time as at the sender.

– When transporting RTP packets, the RTCP Control Protocol [8] is used for purposes such as monitoring and control. RTCP data is generated by the encoder and carried in RTCP packets. There are several RTCP packet types, one of which is the Sender Report (SR) RTCP packet type. Each RTCP SR packet contains an RTP time stamp and an NTP [20] time stamp. Both time stamps correspond to the same instant in time. However, the RTP time stamp is expressed in the same units as RTP time stamps in data packets, while the NTP time stamp is expressed in "wall-clock" time. In Fig.3, NTPa(k) and NTPv(n) are the NTP time stamps of the audio and video RTCP packets. At(k) and Vt(n) are the values of the audio and video clock at the same instant in time as NTPa(k) and NTPv(n), respectively.

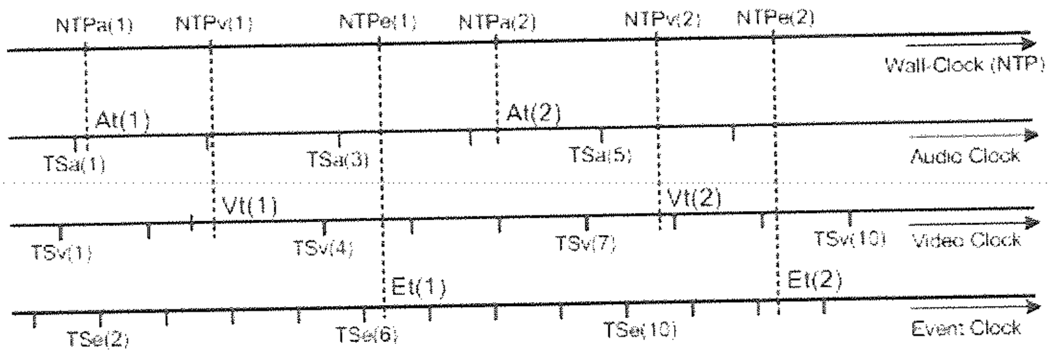

Fig. 3. RTP synchronization

– If the streams use the same wall-clock time, receivers can achieve synchronization by using the correspondence between RTP and NTP time stamps. To synchronize an audio and a video stream, one needs to receive an RTCP SR packet relating to the audio stream, and an RTCP SR packet relating to the video stream. These SR packets provide a pair of NTP timestamps and their corresponding RTP timestamps that are used to align the media. For example, in Fig.3 [NTPv(k) - NTPa(n)] represents the offset in time between Vt(k) and At(n), expressed in wall-clock time. As such inter-stream synchronization is accomplished.

Event Synchronization Mechanism

In order to synchronize the events with the A/V content, we use this RTP synchronization mechanism. At the user's terminal, besides a video and an audio decoder clock, a third clock is used as time basis for the events. We call this time basis the "Event Clock". The functionality of this clock can be compared to that of the Normal Playing Time (NPT). This is a timebase, defined in the DSM-CC MPEG-2 Extensions [13] in order to support Video-On-Demand and it is reused in MHP as the timebase for interactive services. Analogue to the NPT clock, the event clock can be used to reflect pausing, fast forwarding, rewinding and discontinuities in the multimedia stream.

The interactivity server steers the clients' event clock by transmitting RTCP SR analogue packets which map the event clock values Et(i) to the NTP time stamps NTPe(i). As such, the offset between the event clock and the audio or video decoder clock can be calculated. By using this mechanism, server side synchronizing of the client's event clock to the decoder's video or audio clock is possible. However, to do this, the interactivity server itself needs to know the exact link between these clocks. This can be accomplished by offering a graphical user interface to the content provider in order to inform the interactivity server about the mapping between the occurrence of an event and the video or audio time basis. Still, frame accurate synchronization is only possible in a scenario where the link between the time basis that is used by the content provider and its mapping of events to this basis can be kept during transport, encoding and/or transrating (i.e changing the frame rate of the video stream). In order to keep this link, the interactivity server shall use the same wall-clock time as the encoders to perform the calculations needed to re-map the specified events to the clock of the encoders. This is possible by interpreting the SDP files and SR RTCP packets that are generated by the encoders. By interpreting the SDP file, the interactivity server knows the exact start NTP time of the session. As it is supposed that the (usually uncompressed) video data, offered by the content provider, has a constant frame rate, knowledge of the NTP timestamp of the first frame makes it possible to map the occurence of events to the NTP time. If the encoder doesn't generate (appropriate) SDP files, then a initial requirement is that the start value of the encoder clock is set to zero. In order to support the nonstop streaming of concurrent television programs, the interactivity server should also be synchronized with the ESG data. As such, an interactivity server is built that can synchronize events with the decoding clock of the client's terminal while only sending the interactive data that is related to the program that is currently playing.

6 Conclusions and Future Work

In this paper, we introduced two possible solutions for delivering interactive services that are synchronized with the multimedia content that is broadcasted via a IP Datacast over DVB-H system. As the focus is on using MIDP application logic to implement these services, complex logic can be supported while targeting a wide diversity of mobile terminals. The first solution maximizes the usage of the broadcast channel. The advantage of this approach is that no additional channel is required. When evaluating the functional requirements for synchronized interactivity on mobile devices, it is clear that adding extra services to the broadcast channel will require new Java middleware to be available on the MIDP terminal. The JSR 272 specification should provide a decent solution for handling broadcasted data. The second solution combines the broadcast channel with a bidirectional unicast channel. As this solution is less dependent on the available MIDP middleware, it is currently easier to implement but it requires a DVB-H terminal that also disposes of a bidirectional unicast connection such as

UMTS. In this paper a synchronization mechanism was presented that makes it possible to map events to the decoder clock of the broadcasted audio or video stream. As this mechanism is independent of the type of channel that is used to transmit the interactivity related data, it can be used in both the unicast and broadcast solution. Currently we are focussing on the development of the server side component (i.e the interactivity server) in order to synchronize the event clock with the decoder clocks. A demo client application that handles the interactivity server's data via a unicast connection has already been developed.

References

1. ETSI EN 302304 v1.1.1:Digital Video Broadcasting (DVB), Transmission System for Handheld Terminals (DVB-H), November 2004.
2. J2ME Mobile Information Device Profile (MIDP), JSR 37, JSR 118, "http://java.sun.com/products/midp/".
3. Digital Video Broadcasting (DVB), Multimedia Home Platform (MHP) Specification 1.1.1, ETSI TS 102 812 v1.2.1.
4. MADUF, "https://projects.ibbt.be/maduf".
5. INSTINCT, "http://www.ist-instinct.org/".
6. SAVANT, "http://dea.brunel.ac.uk/project/savant/".
7. porTiVity, "http://www.portivity.org/".
8. IETF RFC 3550: RTP, A Transport Protocol for Real-Time Applications, July 2003.
9. IETF RFC 3926: FLUTE- File Delivery over Unidirectional Transport, 2004.
10. MPEG LASeR, "http://www.mpeg-laser.org/".
11. ETSI TR 102 469 v1.1.1: Digital Video Broadcasting (DVB): IP Datacast over DVB-H: Architecture, May 2006.
12. IETF RFC 2327: SDP, Session Description Protocol, April 1998.
13. ISO, Generic coding of moving pictures and associated audio information: Extensions for Digital Storage Media Command and Control, 1996.
14. ETSI TR 102 471 v1.1.1: Digital Video Broadcasting (DVB): IP Datacast over DVB-H: Electronice Service Guide (ESG), April 2006.
15. ETSI TS 102 470 v1.1.1: Digital Video Broadcasting (DVB): IP Datacast over DVB-H: Program Specific Information (PSI)/Service Information (SI).
16. Steckel P, The Handheld Software Platform (HSP), A software platform for DVB-H and IP Datacast.
17. OSGi, "http://www.osgi.org".
18. JSR 271: Mobile Information Device Profile 3, "http://jcp.org/en/jsr/detail?id=271".
19. JSR 272: Mobile Broadcast Service API for Handheld Terminals, "http://jcp.org/en/jsr/detail?id=272".
20. Network Time Protocol (Version 3) Specification, Implementation and Analysis,1992.

IEEE Catalog Number: 07EX1820
ISBN: 1-4244-1297-8
Library of Congress: 2007926696

The 2nd International Workshop on
Broadband Convergence Networks
IEEE/IFIP BcN2007

Standards, Architectures and Managements of Broadband convergence Networks

Editors:
Young-Tak Kim
Dong-Myun Lee
Nazim Agoulmine

May 21, 2007
Munich, Germany

IEEE