MPEG-21 SESSION MOBILITY IN A BROADCASTING ENVIRONMENT

Davy De Schrijver, Frederik De Keukelaere, and Rik Van de Walle
Multimedia Lab, Departement of Electronics and Information Systems,
Ghent University, Belgium
Interuniversity MicroElectronics Center (IMEC), Leuven, Belgium
Interdisciplinary Institute for Broadband Technology (IBBT), Belgium
davy.deschrijver@ugent.be

Abstract Interactive digital television (iDTV) will take an important role in our future life. The world of (digital) television is based on a broadcasting environment. The impact of that kind of environment is very important during the development of applications for digital television. Another remarkable trend of the last years is the growth of new (mobile) devices that can interact with multimedia content like GSMs and Personal Digital Assistants (PDA). Each of those devices has different network and terminal capabilities. Nowadays, television producers not only want to create programs for television but they also want to offer their content to other devices and to the World Wide Web (WWW). In this paper, we will talk about session mobility when we mean switching between different devices. In this paper, we will discuss what kind of problems arises during the development of session mobility in a broadcasting environment. As implementation platform, we will use Multimedia Home platform as middleware of the set top box. For the realization of session mobility, we also use a part of the MPEG-21 technology.

Keywords: Digital Video Broadcasting, Session Mobility, MPEG-21, Multimedia Home Platform.

1. Introduction

Today, people are using a wide set of devices to run applications. For example, people look at multimedia content on their personal computer, PDA, television (through a set top box), GSM, and so on. Of course, the content provider does not want to create different versions of the presentation for each possible device. Additionally, each device may need several terminal specific adjustments. Therefore, the content provider wants to create the content once and wants to publish it everywhere at

any time. This idea of universal multimedia access [1] is presented in figure 1.
When we have several devices at our disposal, we may want to switch from one device to another transparently. To realize such session mobility, we will use the MPEG-21 framework [2].
In this paper, we describe how we can integrate session mobility into a broadcasting environment. First, we will give a small introduction on both MPEG-21 and Digital Video Broadcasting (DVB). Then, we describe how we have implemented the session mobility application. Finally, we discuss the performance results of the implementation.

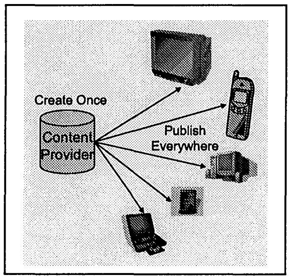

Figure 1. Universal Multimedia Access

2. MPEG-21

MPEG-21, also known as ISO/IEC 21000, is the latest standard of the Moving Picture Experts Group (MPEG) [3]. Most MPEG standards are targeted at audio and video compression, in contrast to MPEG-21 where the core of the standardization is the creation of a generic framework for multimedia production and consumption.

The key concept of the MPEG-21 framework is the "Digital Item". The communication within the framework is based on the transaction of Digital Items (DIs). A Digital Item is a structured object with a standard XML (Extensible Markup Language) representation and associated metadata. How to create a DI is standardized in the second part of MPEG-21, the Digital Item Declaration (DID) [4]. A DI can be seen as a composition of different resources like audio, video, synthetic data, text, right expressions, and metadata such as MPEG-7 descriptions [5]. In figure 2, we give an example DI for a music album. The "Digital Item" contains different "Components" representing album tracks. The "Components" contain "Resources" with a reference to the place where the actual music is stored. To describe the content of the "Item", we use the "Descriptor". In this example, the user has the ability to choose

```
<?xml version="1.0" encoding="UTF-8"?>
<DIDL xmlns="urn:mpeg:mpeg21:2002:02-DIDL-NS">
    <Item>
        <Descriptor>
            <Statement mimeType="text/plain">
                AC-DC: We Salute You
            </Statement>
        </Descriptor>
        <Choice choice_id="select_track">
            <selection select_id="track_1"/>
            <selection select_id="track_2"/>
        </Choice>
        <Component>
            <Condition require="track_1"/>
            <Resource ref="Snowballed.mp3" type="audio/mp3"/>
        </Component>
        <Component>
            <Condition require="track_2"/>
            <Resource ref="Spellbound.mp3" type="audio/mp3"/>
        </Component>
    </Item>
</DIDL>
```

Figure 2. Example of a music album DI

the track of the album he prefers to play. Through the condition (in the "Component" tag) the chosen resource will be available.

Session mobility is a part of Digital Item Adaptation (DIA), part 7 of MPEG-21 [6]. In figure 3, the adaptation process is sketched [6]. The first input is the Digital Item, which is declared in the DID Language and contains the multimedia data. The second input (a DI with a DIA Description) contains information about the context in which the original data will be used. This information is standardized within the DIA specification of MPEG-21 and it contains information like usage environments, terminal capabilities, and network characteristics. The

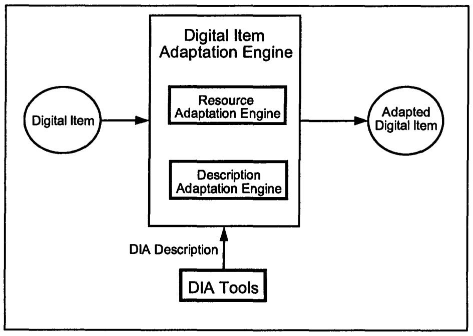
Digital Item Adaptation Engine
Resource Adaptation Engine
Description Adaptation Engine
Digital Item
Adapted Digital Item
DIA Description
DIA Tools

Figure 3. Digital Item Adaptation

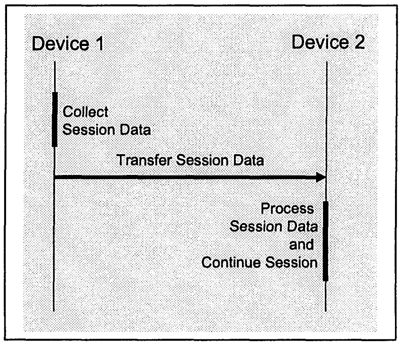

Figure 4. Session Mobility

adaptation engine produces, based on the adaptation information, a new adapted Digital Item.
To explain the architecture for session mobility, let us consider the situation where a session must be transferred between two devices and where the devices can handle all kinds of resources (we are working in ideal circumstances). In figure 4, the simplest architecture for session mobility is given. In the first step, we will collect the session data of the sending device. This data will contain the location of the resource (the DI), the state of the data (played, stopped...), and the configuration state of the selections in the choice of the DI. This DIA Description will be transferred to the other device. Once the second device has received the DIA Description, it processes the received data and continues the session. In figure 2, a possible DI was given. Suppose we are listening to track 2. A possible session mobility DIA Description for this scenario is given in figure 5. In this figure, first we see the status of the data, next the place where the DI (with a reference to the music files) is located, and lastly the configuration information for the choice of the DI is given. The second device runs an adaptation engine that uses the incoming (session mobility) DIA Description and (content) DI to generate a new adapted DI. This new DI contains necessary information in order to continue the session and receive the correct data from the referred location [7].

3. Digital Video Broadcasting

Digital Video Broadcasting (DVB) is the standard in Europe for broadcasting digital television (DTV) over all kinds of transmission media from the content provider to the consumer [8]. Other protocols for the same purpose exist, such as Advanced Television Systems Committee (ATSC), used in the United States [9]. DVB is based on MPEG-2 Transport Streams (MPEG-2 TS) [10]. An MPEG-2 TS contains video, audio, and data from different content providers (e.g. television stations) together with metadata about the composition of the TS itself (and nothing more). For a receiver, it is impossible to find out, based on MPEG-2 TSs, the name of the television station for a certain channel and how to tune to a different channel or steering a dish to a different satellite. DVB extends MPEG-2 by defining additional metadata, the service information (DVB-SI), in the standard.

Digital broadcasting, certainly digital television broadcasting, has different distribution and transmission requisites. In figure 6, we show a part of the transport chain from content producer to consumer (e.g., from television station to television viewer). Every television station can feed more than one channel. Therefore, a TV station will produce a

```
<?xml version="1.0" encoding="UTF-8"?>
<DIDL   xmlns="urn:mpeg:mpeg21:2002:02-DIDL-NS"
        xmlns:sm="urn:mpeg:mpeg21:200:01-DIA-SM-NS"
        xmlns:dia="urn:mpeg:mpeg21:2003:01-DIA-NS"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
    <Item>
        <Descriptor>
            <Statement mimeType="text/xml">
                <dia:DIADescriptionUnit xsi:type="
                    sm:SessionMobilityAppInfoType">
                    <sm:ItemInfoList>
                        <sm:ItemInfo target"http://192.168.0.1/cdi.xml#
                            track_2">
                            <PlayerStatus xmlns="urn:be:mmlab:sm">
                                <mediatime>2.2</mediatime>
                            </PlayerStatus>
                        </sm:ItemInfo>
                    </sm:ItemInfoList>
                </dia:DIADescriptionUnit>
            </Statement>
        </Descriptor>
        <Component>
            <Resource mimeType="text/xml">
                <dia:DIADescriptionUnit xsi:type ="
                    sm:SessionMobilityTargetType" ref="http:
                    //192.168.0.1/cdi.xml"/>
            </Resource>
        </Component>
        <Annotation target="http://192.168.0.1/cdi.xml">
            <Assertion target="http://192.168.0.1/cdi.xml#select_track"
                true="track_2"/>
        </Annotation>
    </Item>
</DIDL>
```

Figure 5. Example of a session mobility DI for the album DI

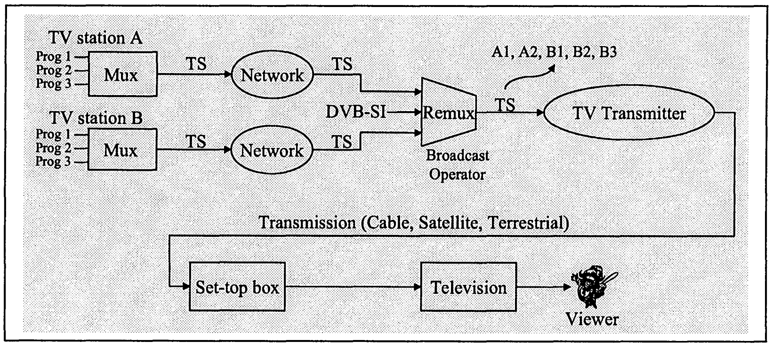

Figure 6. Transport chain in a broadcast environment

transport stream with one or more TV programs. The broadcast operator receives different TSs from the TV stations or from other broadcast operators and produces a new transport stream containing several TV programs. The broadcast operator (for example a cable operator) decides which programs will be encapsulated in the TS; in other words, it does a re-multiplexing of multiple TSs. Another important job of the broadcaster is adding the metadata about the transport stream. The service information must also be encoded (multiplexed) into the TS.

A last important issue for digital television, in particular iDTV, is about how we can broadcast applications to the users. DVB has adopted the standard MPEG-2 data encapsulation methods with only minor variations [12]. MPEG-2 data broadcasting is defined in part 6 of the standard namely Digital Storage Medium - Command and Control (DSM-CC). For data broadcasting in DVB, we use data and object carousels. The data files that are broadcasted are ordered in a directory structure. This directory structure is included in the object carousel. The carousel periodically delivers the data files to the set top box of the consumer.

4. Multimedia Home Platform

In section 2, we have seen the framework that allows us to realize session mobility (in MPEG-21). In section 3, we described how we could broadcast television programs together with applications. Of course, it is impossible to run applications on regular televisions. Therefore, we need some platform that understands the applications and can generate television signals so that we see the application together with the TV program on the TV screen. For the latter, we need a set top box (STB) that runs applications and presents the generated information to the viewer. To run applications on a STB, a STB has a software API sitting between the application and the drivers. This software layer is known as the "middleware".

The middleware of a STB can be seen as the operating system of the STB. There are many different (open) standards of middleware, each having a clearly defined API. Multimedia Home Platform (MHP) is such an open standard and will be used in most European countries [13]. This is why we have decided to implement our session mobility application on MHP.

The MHP standard is based on the Java Virtual Machine (PersonalJava1.2). MHP extends the PersonalJava1.2 with other APIs like JavaTV, HAVi (Home Audio/Video Interoperability), DAVIC (Digital Audio Visual Council) and DVB specific APIs (for example for controlling the carousels).

5. Running session mobility in a broadcast environment

5.1 Scenario

To demonstrate how to run session mobility in a broadcasting environment, we use a television with set top box and a PDA as devices. As data content, we use video fragments. The set up is given in figure 7. In situation (a), we transfer a movie from the PDA to the television; in situation (b), we transfer the television program to the PDA.

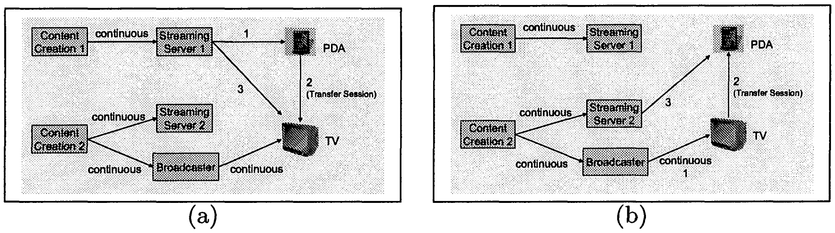

Figure 7. Session Mobility in a broadcast environment

We need two implementations of the session mobility application, one for the PDA and one for the set top box. The implementation on the PDA was done in C#, and on the set top box in Java, because the middleware of the set top is MHP.

As we can see in figure 7, there are two possible scenarios to have session mobility. In the first situation (figure 7(a)), we are viewing a movie on our PDA. At a certain moment, we want to continue the rest of the movie on the TV screen. Therefore, we need to transfer the session from the PDA to the television. The television (STB) receives the DIs and starts the player on the TV screen. The STB receives the video through the interactive channel and the TV program through the broadcast channel. In the second situation (figure 7(b)), we are watching to a TV program and want to continue watching on a PDA. The STB receives the TV program together with the session mobility application from the broadcaster. After the session is transferred, the PDA receives the video (with a lower quality and resolution) from a streaming server and not from the broadcast server. In contrast with the first situation where we receive the video from the same streaming server, now we need two servers (a broadcast and a streaming server) to watch to the TV program.

5.2 Implementation details

As previously stated every middleware has a restricted (well-defined) set of APIs. The most important APIs for MHP are JavaTV, HAVi,

DAVIC, and DVB specific APIs. Thus limiting us to this set of APIs, other libraries must be transported through the broadcast channel. Another important restriction for a DTV program developer is the limited resources: a typical STB has 32 Mb RAM and a processor of 166MHz. This means that we must optimize our programs as much as possible and keep the resource consumption low.
Combining these two issues gives us a small space to move during the development. We must avoid using libraries not included in the APIs. The application of session mobility needs several libraries (MPEG-21, SOAP and QuickTime libraries) that are transported over the broadcast channel and stored in the memory of the set top box when needed. To test our implementation, we have been using an emulator (which is written in Java). An interactive TV display can be split into three layers: a background layer (is usually capable of displaying a single color or a still image); a video layer (where video is shown), and a graphics layer (for the graphics operations of an application). In our application, we had a still image in the background, a broadcasting program in the video layer and the application in the graphics layer.

6. Performance results of the implementation

In this last section, we will discuss the results of the implementation of the session mobility application on an MHP set top box (tested in an emulator).

To run the session mobility application, we need some extra libraries that must be transported over the broadcast channel (in an object carousel) from the broadcaster to the set top box. Because the files (of the carousel) is periodically delivered to the set top box, it is important to know how big the libraries are. In table 1, we give the needed libraries (not included in the MHP APIs) together with the size of the library. We give the size of the compressed (.jar or .zip file) and the uncompressed size. The uncompressed size is the most important size because it will be this size that must be transported over the channel. When we need a class of the library, we must store the (uncompressed) class in the memory. We are also interested in the memory consumption of the session mobility application. In table 2, we give the memory consumption of the application. First, we have measured the memory of the emulator whitout any running application. Then we have started the session mobility application and measured the memory after we have received a video from the PDA on the set top box. After that, we started again the application and measured the memory after sending the TV program session to the PDA. Finally, we have measured the memory consumption of the application without any image in the background layer. In table 2, we see a difference in the memory consumption between the transfer of a session in situation (a) and (b) in figure 7. The reason for the difference is that in situation (a), the QuickTime library (for playing the video) is needed and not in situation (b). In the last row of the table, we see a remarkable conclusion. We will use 3MB less memory when we render no picture in the background. On a set top box with 32MB memory, 3MB is significant. From this table, we can conclude that it will be possible to run the application on a set top box (with limited resources) and that a set top box can be used as an MPEG-21 terminal. For the whole application with an MPEG-21 terminal but without an image in the background, we need only 1MB memory.

Table 1. Size of the used libraries

Description	*Size Library (MB[1])*	*Uncompressed Size (MB)*
MPEG-21 library	1,02	2,12
SOAP library	1,91	3,6
QuickTime library	0,88	1,66
Total	3,81	7,39

[1] 1MB = 2^{20} bytes

Table 2. Memory consumption of the applications

Application	*Memory Consumption (MB)*
Emulator without applications	2,09
SM from PDA to STB	6,45
SM from STB to PDA	6,08
SM from STB to PDA without still image	2,9

7. Conclusions and future work

In this paper, we have first explained how we can transfer a session from one device to another. To implement such a session mobility application, we have used the MPEG-21 framework. After a short introduction on MPEG-21 and an explanation of the session mobility architecture, we have introduced DVB and MHP. Then we have explained how we could integrate the idea of session mobility into a broadcasting environment.

We have tested our application in an emulator measured the memory consumption and the size of transported data. We concluded that it would be possible to run session mobility (and an MPEG-21 terminal) on a set top box.
In the future, we want to test our application in a real broadcasting environment.

Acknowledgments

The research activities that have been described in this paper were funded by Ghent University, the Interdisciplinary Institute for Broadband Technology (IBBT), the Institute for the Promotion of Innovation by Science and Technology in Flanders (IWT), the Fund for Scientific Research-Flanders (FWO-Flanders), the Belgian Federal Science Policy Office (BFSPO), and the European Union.

References

[1] A. Vetro, C. Christopoulos, T. Ebrahimi, *Universal Multimedia Access*, IEEE Signal Processing magazine, 20 (2) 16-16, March 2003

[2] I. Burnett, R. Van de Walle, K. Hill, J. Bormans and F. Pereira, *MPEG-21 Goals and Achievements*, IEEE Multimedia, IEEE Computer Society, October-December 2003, p. 60-70

[3] Moving Picture Experts Group, *MPEG Home Page*, http://www.chiariglione.org/mpeg/

[4] ISO/IEC, *ISO/IEC 21000-2:2003 Information technology – Multimedia framework (MPEG-21) – Part 2: Digital Item Declaration*, March 2003

[5] B. S. Manjunath, P. Salembier, T. Sikora, P. Salembier, *Introduction to MPEG 7: Multimedia Content Description Language*, John Wiley & Sons Ltd., West Sussex, England, 2002, ISBN 0-471-48678-7

[6] Moving Picture Experts Group, *ISO/IEC 21000-7 FDIS Part 7: Digital Item Adaptation*, ISO/IEC JTC1/SC29/WG11 N6168, Waikoloa, December 2003

[7] F. De Keukelaere, D. De Schrijver, S. De Zutter R. Van de Walle, *MPEG-21 Session Mobility For Heterogeneous Devices*, Euromedia, 2004

[8] Digital Video Broadcasting, *DVB Home Page*, http://www.dvb.org/

[9] Advanced Television Systems Committee (ATSC), *ATSC Home Page*, http://www.atsc.org/

[10] ISO/IEC, *ISO/IEC 13818-1:2000 Information technology – Generic coding of moving pictures and associated audio information: Systems*, December 2000

[11] Tektronix, *A Guide to MPEG Fundamentals and Protocol Analysis*, http://www.tektronix.com/video_audio, 2002

[12] ETSI, *ETSI EN 301 192 Digital Video Broadcasting (DVB), Specification for Data Broadcast, edition 1.3.1*, May 2003

[13] ETSI, *ETSI TS 101 812 Digital Video Broadcasting (DVB); Multimedia Home Platform (MHP) Specification 1.0.3*, June 2003