Experimental Characterisation of an Open Out-of-home MHP-based DVR Service

Philip Leroux
Ghent University-IBBT-INTEC
G. Crommenlaan 8 bus 201,
B-9050 Gent, Belgium
philip.leroux@intec.ugent.be

Kris Vanhecke
Ghent University-IBBT-INTEC
G. Crommenlaan 8 bus 201,
B-9050 Gent, Belgium
kris.vanhecke@intec.ugent.be

Klaas Roobroeck
Ghent University-IBBT-INTEC
G. Crommenlaan 8 bus 201,
B-9050 Gent, Belgium
klaas.roobroeck@intec.ugent.be

Chris Develder
Ghent University-IBBT-INTEC
G. Crommenlaan 8 bus 201,
B-9050 Gent, Belgium
chris.develder@intec.ugent.be

Filip De Turck
Ghent University-IBBT-INTEC
G. Crommenlaan 8 bus 201,
B-9050 Gent, Belgium
filip.deturck@intec.ugent.be

ABSTRACT

MHP (Multimedia Home Platform) is an open standard for interactive TV middleware that was designed by the Digital Video Broadcasting (DVB) alliance and is found on many commercially deployed set-top boxes (STBs). In this paper, a detailed system architecture is presented which enables customers of digital television providers to use their MHP-based DVR from an out-of-home location through their mobile phone. Building further on the openness of the MHP platform, special focus was on the overall scalability, security and extensibility of the system and the efficient and fast communication between mobile phone and set-top box. Technical challenges are imposed by the limited hardware resources that are available on most of the currently deployed STBs and by the delays that are introduced by the mobile communication networks. In order to characterise the system's performance, extensive throughput and scalability tests were performed on commercial DVRs. It is shown that the overall system performance meets the predefined requirements even under heavy load.

Categories and Subject Descriptors

H.5.1 [**Information Systems Applications**]: Multimedia Information Systems; D.2.11 [**Software Engineering**]: Software Architectures— *Domain-specific architectures, Service-oriented architecture (SOA)*

General Terms

Design

Keywords

Distributed Software design, Web Services, Java ME, MHP

1. INTRODUCTION

Due to the abundance of media, people have become more selective with regard to what they want to watch on television. This is confirmed by the growing sales of video recorders and the increasing popularity of services such as video-on-demand. In spite of their success, the current products and services on the market still provide no conclusive solution for a number of problems. The hardware capabilities of the devices are often limited (especially with respect to the CPU and RAM) and it is still necessary to be at home when programming the DVR. However, people often remember to record a program when outdoors and as such are forced to pay for view it on demand later on.

In this article, a generic architecture is presented that tackles this problem. It enables the user to configure his DVR remotely by using his mobile phone while taking into account the limited resources of both the set-top boxes and (most) mobile phones. This out-of-home recording may be provided as a service by the television provider, which implies that no additional configuration has to be done by the user nor does he has to purchase additional hardware.

The rest of the paper is organized as follows. In Section II, related work is discussed and we explain how our system architecture differs from other implementations. In Section III, we amplify on some requirements that our architecture at least should fulfill. Section IV provides an architecture overview, detailing each specific component and the interaction between these components. In section V, the interaction mechanism between the different components is described. In section VI results of the extensive performance tests that were performed on the implemented prototype are detailed. Finally, section VII states our conclusions.

2. RELATED WORK

Recently, several commercial operators started with offering similar out-of-home configuration options. Contrary to our system architecture, all these implementations are specifically targeting one specific architecture and one specific end user platform. Popular examples of these closed source implementations are DirecTV [1] in the USA and Sky [2] in the United Kingdom. Both implementations allow their clients to configure their DVR through a web application or by using an iPhone application. Additionally, Sky customers can configure their DVR by sending an SMS. The work that is presented in this paper differs from these closed source implementations as it is an open and extensible framework that is built upon international standards and for more generic platforms, both for the mobile client side as on the STB. For the latter, all components were compliant with the MHP specification. MHP (Multimedia Home Platform) [3], is the collective name for a compatible set of middleware specifications developed by the Digital Video Broadcasting (DVB) alliance [4] and is currently deployed on many commercial STBs.

Other related work focuses on the out-of-home configuration of the DVR by deploying a home media center that is running at the customer premises [5]. Contrary to this work, we propose an architecture that will require no installation or configuration of additional components at the user's houses. Instead, a central deployable component with a general interface to the operator's management and payment components is introduced.

3. FUNCTIONAL REQUIREMENTS

After evaluation of related implementations and patents, it was defined that our architecture should enable the following functional requirements:

- By using his mobile device, the user should be able to securely login to the system and select a DVR from a list of DVRs that are linked to his account.
- An Electronic Program Guide (EPG) can be retrieved and browsed on the device.
- Based on the program selection in this EPG, the user can send a recording request to the DVR.
- Furthermore, a list of all scheduled records and recorded programs on a specific DVR can be retrieved.
- Scheduled records from this list can be deleted.
- Finally, the client application can also retrieve capacity information from every DVR linked to his account.

The main quality attribute requirements of the system, listed by decreasing importance, are: usability, performance, scalability, interoperability and modifiability.

4. ARCHITECTURE DESCRIPTION

One of the initial design decisions was the use of a central, management component that works as an intermediate layer between the user's mobile phone and STB. The use of such an intermediate layer was chosen, as direct communication between STB and mobile phone would introduce some inconveniences. First of all, in order to communicate with the STB, the STB's IP address needs to be known and it has to be accessible from outside the service provider's network where the STB is connected to. However, for security reasons, most service providers don't allow direct communication to their STBs from devices outside their private network. Second, the user would need to configure the mobile application to connect with that specific IP address.

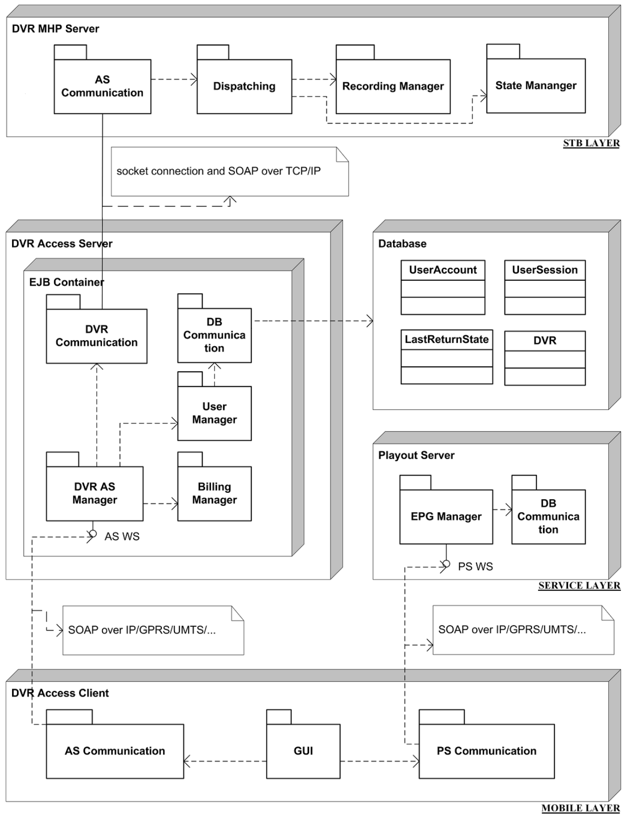
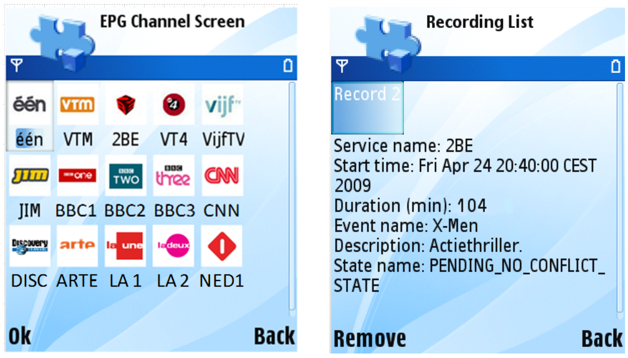


Figure 1: Low level system architecture.

As shown in the low level system architecture of Figure 1, three layers can be distinguished in the the general architecture: the client layer, the central service layer and the STB layer. The client layer consists of the client application which provides the presentation and communication to the user. The second layer contains the centralized processes. It consists of the DVR Access Server, which authenticates the user, manages all DVRs that are linked with this service and forwards all instructions to the correct DVR. The other components in the service layer are the Playout Server, which provides EPG data to the client application and a database component. This central layer is typically found in the backend of the service provider who manages all the STBs and most of the time also the digital video services. In the STB layer, an MHP application will run on the STB and process all incoming instructions. This is the DVR MHP Server application.

4.1 DVR Access Client

The main component (GUI) of the mobile application defines and initializes all graphical components and determines the flow of the user application. Two additional packages containing Web Service clients were added to communicate with the service layer. More specifically, *AS Communication* manages all communication with the DVR Access Server while the *PS Communication* packet is responsible for the Playout Server related communication. In both cases, the Simple Object Access Protocol (SOAP) was used. Once the user has logged into the system, he receives a list with all DVRs that are coupled with his user account. After selecting



(a) (b)

Figure 2: Screen shots of the client application ((a): EPG overview, (b): Recording information).

a specific DVR he can consult the program guide, requesting the current capacity of this DVR, consult and manage all scheduled recordings on this DVR or choose another DVR. Figure 2 shows two screen shots of the prototype client application. In order to target as many devices as possible, the mobile client was developed by using the Java Platform, Micro Edition (Java ME).

4.2 DVR Access Server

The DVR Access Server is the central component of the system and will be deployed in the back end layer at the server park of the service provider. This component is responsible for user authentication and distribution of the instructions to the various DVRs. The prototype that was tested during the performance analysis (detailed in Section VI), was developed by making use of Java EE (Java Enterprise Edition). It consists of several packages containing a number of stateless session beans, all providing different functionalities. One package is responsible for user authentication and management of sessions, while others retrieve database information or communicate with the DVRs. Two Web Services are designed, one to communicate with the DVR Access Client application and one with the DVR MHP Server.

4.3 Playout Server

The Playout Server is a Java EE application that is also part of the back end layer and whose Web Service can be called to retrieve all EPG data from a database. During the performance tests, live EPG data was used. Note that in operational systems, this component is already present in the backend of the service provider.

4.4 DVR MHP Server

Apart from the main module, that is controlled by the MHP Application Manager to start, pause and stop the application, four other modules are present. When the main application is started, the *AS Communication* module is initialized and starts listening to incoming messages from the DVR Access Server. When an instruction is received through this channel, it will be forwarded to the *Dispatching* unit for quick handling of the messages. All capacity related requests are then handled by the *State Manager* while all record related instructions are finally handled by the *Record Manager*. The prototype for the DVR MHP Server was developed using the MHP 1.1.2 Specification [3].

5. COMMUNICATION DETAILS

One of the main challenges for this use case, was the communication between the central service layer and the STB layer, due to the limited hardware resources of the STB. As a result, the communication between the DVR Access Server and DVR MHP Server applications is performed in an atypical manner. While the application client of the mobile layer uses the services of the service layer applications by addressing their Web Service, another approach has to be followed in this case. Due to the limited resources of the STB, no Web Service engine can run on the recording device.

In order to keep all advantages of a Web Service, following communication method was used: when an instruction arrives at the DVR Access Server application, the corresponding parameters are stored locally. An asynchronous socket message is sent to the DVR MHP Server, which triggers the device to call the server's Web Service and retrieve the necessary information to successfully execute the instruction. The result is then sent to the DVR MHP server using the same Web Service. To complete the process, the DVR MHP Server sends a socket message back to the DVR Access Server. The DVR Access Server will then evaluate the success of the operation and provide feedback to the user.

6. PERFORMANCE RESULTS

In order to evaluate the performance and scalability of the system, the developed prototype was subject to several tests. The prototype application was running on a commercial STBs that is currently used in many Belgian households. In cooperation with Belgian cable operator Telenet NV [8], this STB was running and fully functioning in their commercial cable network, which offers digital TV to hundreds of thousands subscribers. This had a major impact on the measurement results as on such commercial and fully functioning STBs, operators are typically running their own middleware processes in the background, already taken up a large portion of the STB's (limited) processing power. The main advantage of this approach, is that all results are representable for real life (out of the lab) situations. As all television channels were live data streams, automatic parsing of an up to date (external) program guide was performed in the Playout Server. During our tests, the Round Trip Time (RTT) of a request is measured in order to identify the numerous influences that cause a certain amount of delay.

In a first category of tests, only one request was sent to the DVR Access Server. The tests indicate that the data rate of the mobile communication channel, has an important impact on the RTT of a request. This is shown in Figure 3 where the average results of 10 tests are shown. For the UMTS connection, the network of the Belgian telecommunications operator Proximus was used. This UMTS network has a download link with an average data speed of 384 Kbps, which is a much lower data rate than a typical WiFi connection (802.11g) coupled with broadband internet access. As a result the RTT when using the UMTS network is significantly longer than when using the WiFi connection. Still, it is shown that even the heaviest operation *addRecord()* remains under the 5 seconds barrier. This 5 seconds bar-

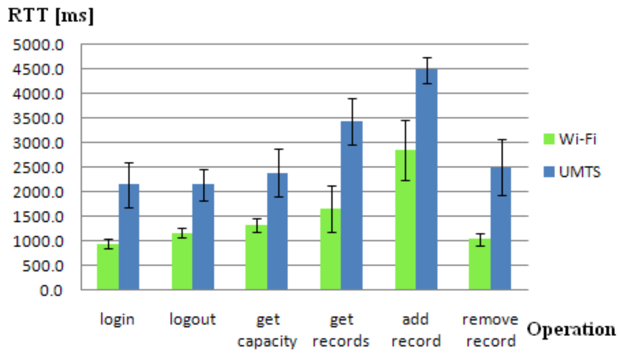


Figure 3: RTT (ms) for different mobile data transmission protocols.

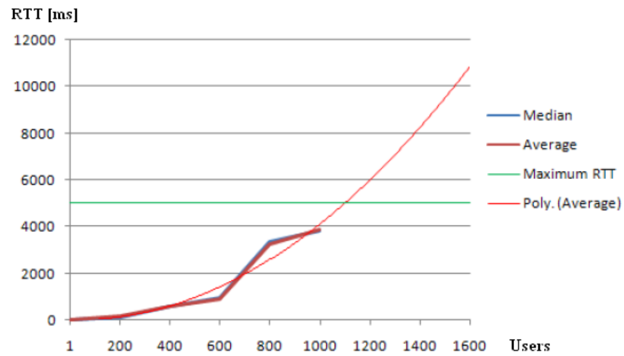


Figure 4: RTT (ms) of the login procedure as a function of the number of simultaneous login requests.

Table 1: Time difference of instructions (ms) depending on the recording activity.

Operation	Not Recording	Recording
get capacity	97,7	105,0
get records	87,1	138,4
add record	1429,0	1675,8
remove record	192,0	196,0

rier follows from a study performed by the Aberdeen Group where they wanted to investigate the correlation between the delay in a web application and the consumer's satisfaction of the service. Their report [7] states that the RTT should be limited to 5.1 seconds.

The impact of the recording activity is shown 1. These measurements were performed starting from the DVRCommunication module in order to filter out other influences. It is shown that the recording activity of the DVR will have a small negative impact on the execution time of all the instructions. The biggest absolute time difference (i.e. 246.8 ms) was measured when executing the *addRecords()* instruction. This is due to the fact that this relatively heavy instruction is imposed to the RecordManager of the MHP application, which is already responsible for the dispatching of the current recording.

Figure 4 shows a test using login requests. The RTTs were measured in every run, which differed in the amount of simultaneous login requests that were executed. A trend is added to predict the RTT using more requests. The green line represents the maximum RTT of 5 seconds considered acceptable by the user. The graph shows that the RTT is still acceptable when 1097 people log in at the same time.

7. CONCLUSIONS

In this paper, an open system architecture is described for the out-of-home configuration of a DVR by using a mobile phone. The system architecture requires no installation of additional components at the user's premises. Instead, a central component was installed at the Service provider's backend. This central component may interface with the Service provider's existing management system. One of the main challenges of the system was the implementation of efficient and reliable communication between different components with limited resources. This was implemented by combining synchronous and asynchronous communication calls. Additionally, decent user feedback is supported throughout the whole architecture. All functional requirements were fulfilled and the architecture was designed with usability, modifiability, interoperability, performance and scalability in mind. In order to evaluate the performance and scalability of the architecture, a prototype system was designed and thoroughly and successfully tested using live EPG data, a live transport stream and currently deployed commercial devices.

Acknowledgment

The authors would like to thank Bart Catrysse from Telenet N.V.[8] for his valuable input. Chris Develder is a Post-Doctoral Fellow of the Fund for Scientific Research, Flanders (FWO-V).

8. REFERENCES

[1] DIRECTV, `http://www.directv.com/`.
[2] Sky remote record service, `http://www.sky.com/`.
[3] Digital Video Broadcasting, *Multimedia Home Platform Spec. 1.1.2*, DVB Bluebook A068 Rev. 1, April 2005.
[4] Digital Video Broadcasting, `http://www.dvb.org/`.
[5] Shih-Hsiang Lo, Chi-Chia Lin, and Ming-Syan Chen, *Controlling Digital TV Set-Top Box with Mobile Devices via an IP Network*, IEEE Trans. on Consumer Electronics, vol. 152, number 3, 2006.
[6] Digital Video Broadcasting, *Digital Recording Extension to Globally Executable Multimedia Home Platform*, ETSI TS 102 817 V1.1.1, September 2007.
[7] Bojan Simic, Aberdeen Group, *Monitoring the End-User Experience for Web Applications: What Gets Measured, Gets Done*, January 2008.
[8] Telenet N.V., `http://telenet.be/`.