Overview of Open Standards for Interactive TV (iTV)

Günther Hölbling, Tilmann Rabl and Harald Kosch

Department of Distributed Information Systems, University of Passau
Innstraße 43, 94032 Passau, Germany
{guenther.hoelbling, tilmann.rabl, harald.kosch}@uni-passau.de

Television has become the most important mass medium. The digitalisation in progress gives new possibilities to enrich the television experience. The Electronic Program Guide - though still very limited - gives a first impression of future television. There are various attempts to give the audience a more active role. All these can be found under the topic interactive TV. Besides the world of television, there is a fast growing community for videos in the World Wide Web (e.g. YouTube[1] or MyVideo[2]). Until today interactivity in "WWW-videos" is of minor importance, but has a great potential for growth. It has not reached a similar level of professionalism as in iTV and no open standards have been introduced. Thus we will concentrate on interactivity in traditional TV.

In this chapter, we present different forms of interaction in television and give an extensive overview of existing platforms and standards like MHEG, DAVIC, Java TV, MHP, GEM, OCAP and ACAP. Also the underlying technology that forms the basis for these standards is presented in short. Finally, the relationships between the different standards and a look at future trends of iTV will be given.

1 Introduction

The term "interactive television" (iTV or ITV) is used for television systems in which the audience may interact with the television content. Interactivity in TV is usually not intended to mean that the viewer is enabled to change the storyline of a program. Rather he/she may participate in a quiz show, gather additional information on news topics or directly buy a product presented in a commercial. Also Electronic Program Guides (EPGs), Video on Demand (VoD) Portals or Telelearning can be realized with iTV Systems.

[1] YouTube - http://www.youtube.com/
[2] MyVideo - http://www.myvideo.de/

Interactivity in television is not as new as one might think. In the beginning, around 1960, there were shows, where viewers called in and participated in quiz shows. In 1974, the teletext was developed in the United Kingdom. It was the first form of additional content delivered with the broadcast programs. At the IFA[3] 1979 the Tele-Dialog was introduced. It was a televoting system which allowed viewers to participate in polls for TV shows by calling specific telephone numbers. It was not until the 1990s that more advanced forms of interactivity appeared in the media landscape. One of the first trials was the Full Service Network by Time Warner, which was launched in December 1994. This trail provided several interactive services, like video-on-demand, program guide, video games, and home-shopping to the customers. In 18 months only 65 people subscribed and therefore it was closed again. With the ongoing digitalisation of television, interactivity became an interesting topic again, and many broadcasters have been trying to enrich their product range by enhancing it with interactive services.

We start with a categorisation of interactivity in TV (section 2). iTV systems are complex systems that involve a long chain of successive processes from the broadcaster to viewers. Section 3 gives a short overview of the main components of such systems. One of the most interesting and important components to empower the viewer to use iTV applications, the middleware and different iTV standards are discussed in section 4. The last section concentrates on the interrelation between different standards.

2 Levels of Interactivity

Interactive applications have a widespread diversity of user interfaces and resource requirements but also different levels of interactivity. In this section we will introduce seven levels and for each level we will give some representative interactive applications. Our categorisation is based on [26].

Level 1 - Basic TV: Interaction on this level is defined as basic functionalities for watching TV such as switching channels or powering the TV set on and off.

Level 2 - Call-In-TV: At this level, the interaction between the audience and the broadcaster is established by techniques such as telephone calls or short message service (SMS). Examples of such TV shows are televoting shows where viewers can vote for a candidate or the next music clip.

Level 3 - Parallel TV: Parallel TV introduces alternative content on multiple channels. The viewer is empowered to change the way he/she consumes a broadcast program. Popular examples for broadcasts on this level are multilingual audio channels or subtitles. Another form of parallel TV are shows with different camera angles, well known from auto racing pro-

[3] Internationale Funkausstellung Berlin

grams. A very special form are movies that show the perspective of different characters on different channels.

Level 4 - Additive TV: This level is also known as "enhanced TV". Additionally to the TV program further content is broadcast. The content can be basic information or advanced services. A well-established service is teletext. Applications like EPG or synchronised computer programs are advanced examples. A return channel is not needed for applications on this level.

Level 5 - Service on Demand: The "Media on Demand" level enables the viewer to consume programs detached from the TV schedule. This level includes video on demand (VoD), upgrade services and other services that are provided on demand. The interaction between the user and the service provider requires a return channel. In TV environments, VoD is not very common because of its technical demands. Often, Near-VoD is used instead of VoD. Near-VoD uses several channels where multiple copies of a program are broadcast in short intervals.

Level 6 - Communicative TV: For such programs, content from other sources such as the Internet can be accessed in addition to broadcast content. Services that stem from the PC domain can also be used and combined with TV. At this level, TV may also be enriched by community functions: chats, online games or email. Another option is user-generated content that can be uploaded. Thus, user- or community-generated programs become possible.

Level 7 - Fully Interactive TV: The most enhanced level of interactivity enables the user to create her/his individual storyline for a program. A program on this level can be understood as a kind of video game, in which the user affects the proceeding of the program. The program can also be affected automatically based on personal profiles, which may include personalised commercials as well as personalised movies.

Today most iTV applications can be assigned to the level 4, 5 or 6. Nevertheless user-generated content, as mentioned on level 6, is hardly present in the world of iTV. Applications on level 7 are still in their infancy although several approaches for personalised commercials are in progress (e.g. Advertising that is relevant to a person[4]).

3 Basic Technologies for iTV

Before discussing the different iTV standards, we will give a brief introduction of the basic technology for iTV.

[4] United States Patent Application: 20070174117; Advertising that is relevant to a person by the Microsoft Corporation

3.1 Set-top Boxes

A set-top box (STB) is a device that forms a link between a TV-set and an external source[5]. The source usually is one of the following: cable, terrestrial antenna or satellite dish. The term set-top box stems from the fact that STBs usually are placed above the TV-set. STBs have multiple purposes, ranging from simple signal conversion (e.g. digital receiver) over personal video recorder functions to interactive media center functions.

3.2 Video Encoding and Transport

Audio and video encoding standards play a major role in the world of digital TV (DTV). The encoding and the resulting reduction of data leveraged the introduction of Digital TV. An important standard which is pervasive in DTV is the MPEG-2 standard by the Moving Picture Experts Group[6]. Also MPEG-4 Part 10 Advanced Video Codec (MPEG-4 AVC or H.264) is used in several recent DTV standards. Besides the video and audio encoding capabilities of MPEG-2, the system part of the standard is of significant importance in DTV. The MPEG-2 system part describes the combination of multiple encoded audio and video streams and auxiliary data into a single bitstream. This makes it possible to carry multiple TV channels, additional data and iTV applications.

3.3 Digital TV Standards

The term "Digital TV" (DTV) or "Digital Video Broadcasting" (DVB) is used for the transmission of digitised audio, video and auxiliary data. As for analog broadcasting, the digital TV market is fragmented. There are various standards developed by different organisations all over the world. This fragmentation is even extended by the accommodation of the DTV standards to different transmission channels used for broadcasting. For that reason, there are different standards for terrestrial, satellite, cable and mobile TV. In Europe, the DTV standards of the Digital Video Broadcasting Project[7] (DVB) are used. Other major players in the development of DTV standards are the Advanced Television Systems Committee[8] (ATSC), the CableLabs[9] in the US and the Association of Radio Industries and Businesses[10] (ARIB) in Japan.

[5] According to more general definitions all forms of electronic devices that are connected to a TV-set are called set-top boxes, but we will use the more restricted definition from now on.

[6] The Moving Picture Experts Group (MPEG) - `http://www.chiariglione.org/mpeg/index.htm`

[7] Digital Video Broadcasting Project - `http://www.dvb.org/`

[8] Advanced Television Systems Committee (ATSC) - `http://www.atsc.org/`

[9] CableLabs - `http://www.cablelabs.com/`

[10] The Association of Radio Industries and Businesses (ARIB) - `http://www.arib.or.jp/english/`

4 Middleware Platforms

In this section we will give an overview of several open iTV standards and middleware platform specifications. This overview is not exhaustive, but covers the major open standards.

4.1 MHEG

The Multimedia and Hypermedia Information Coding Expert Group[11] (MHEG), a subgroup of the International Organisation for Standardisation[12] (ISO), published the MHEG standard in 1997. It was designed as an equivalent to HTML for multimedia presentations [22]. Therefore, the aim of the group was to describe the interrelation between different parts of a multimedia presentation and to provide a common interchange format. The standard initially consisted of five parts, and three parts were added later on:

MHEG-1: MHEG Object Representation Base Notation (ASN.1) [15].
MHEG-2: Should provide an encoding based on SGML instead of ASN.1 but was never finished [25].
MHEG-3: MHEG Script Interchange Representation [16].
MHEG-4: MHEG Registration Procedure [14].
MHEG-5: Support for Base Level Interactive Applications [17].
MHEG-6: Support for Enhanced Interactive Applications [19].
MHEG-7: Interoperability and Conformance Testing for MHEG-5 [20].
MHEG-8: XML Notation for MHEG-5 [21].

In contrast to other standards, MHEG was designed to be only a description language for final-form interactive multimedia presentations. It provides neither a definition of a graphical user interface nor any architectural specification of the execution engine.

The first part of the standard defines the encoding of MHEG presentations in ASN.1 notation. A central aspect of the design was to build a generic standard. As such, it contains no specification about the application area or target platform. MHEG follows an object-oriented approach. Media elements of a presentation, such as text, audio and video, are represented by *Content* objects. These can contain information about the media elements, such as spatial and temporal attributes, as well as information about the actual content or a reference. *Action*, *Link*, and *Script* objects are used to describe behaviours. Simple objects can be grouped to *Composite* objects. The reason for this is to bundle objects that are needed together. Limited user interactivity is provided by the *Selection* and *Modification* class. The *Selection* class enables the user to select an item of a predefined set as in drop-down menus,

[11] Multimedia and Hypermedia Information Coding Expert Group (MHEG) - `http://www.mheg.org`
[12] International Organisation for Standardisation (ISO) - `http://www.iso.org`

while the *Modification* class provides free user input. For further information about these and the remaining classes please refer to the MHEG standard [15].

Part 2 was planned to provide an encoding based on SGML, but it was never finished.

The third part of the MHEG standard defines a virtual machine and an encoding model for scripting. The interactive elements of MHEG-1 are very limited. This part of the standard features advanced operations. The encoding is based on the Interface Definition Language of CORBA and is known as the Script Interchange Representation [9]. As for the other standard parts, there were no specifications made about the concrete execution environment. The encoding is mainly a intermediate representation for the platform-independent exchange of scriptware.

MHEG-4 describes the procedure to register identifiers for objects, for example for data formats or script types.

MHEG-1 and MHEG-3 had many features that were too complicated for the technology of their time. In order to overcome this problem and to support systems with minimal resources, MHEG-5 was developed. Although MHEG-5 is a simplification of MHEG-1, there are too many differences for the standards to be compatible. The class hierarchy itself is quite different. To reflect the interactive television domain, the naming was adapted. An MHEG-5 application consists of *Scenes*, that are composed of *Ingredients*.

MHEG-5, like MHEG-1, lacks possibilities for advanced processing of hypermedia data. Although MHEG-1 was already extended by the definition of a new script encoding standard and an according virtual machine in MHEG-3, the new standard MHEG-6 was specified. In contrast to MHEG-3, MHEG-6 builds upon existing solutions for data processing. MHEG-6 uses the Java programming language as a basis and defines an interface for the interoperability of MHEG and Java objects, the MHEG-5 API.

MHEG-7 defines a test suite for interoperability and conformance testing for MHEG-5 engines. Additionally, a format for test cases is defined to allow more detailed or application specific tests.

MHEG-8 defines an alternative encoding format for MHEG-5 objects based on XML.

Although MHEG-1 was not able to gain wide acceptance in the iTV market the reduced specification of MHEG-5 is used in several systems. Today MHEG-5 has great industry support and MHEG-5 content (MHEG 5 UK Version 1.06 [3]) is broadcast in the UK and New Zealand. Also many extensions and profiles such as Euro MHEG have been developed and are in use today.

4.2 DAVIC

The Digital Audio-Video Council[13] (DAVIC) was founded in 1994 and completed its work in 1999. It was a non-profit organisation with a membership of

[13] The Digital Audio-Video Council (DAVIC) - `http://www.davic.org/`

over 220 companies. Its purpose was to promote interactive digital audio-visual applications and to maximise interoperability across countries and applications and services [6]. Since there were so many companies involved, a major target was to keep the specifications to a minimum. So, existing standards were used whenever possible, and new ones created only if none existed. For example for multimedia information delivery the MHEG-5 format was used. The versions 1.0 - 1.4 of the standard are a set of 11 (v1.0) to 14 (v1.4.1) parts that cover all areas of commercial interactive multimedia experience. Version 1.5 is an additional set of five parts that especially pay attention to IP-based audio-visual services. After 5 years, the DAVIC work was completed. Some concepts and parts of DAVIC were resumed by the TV-Anytime[14] organisation. Nowadays, not all of the DAVIC specifications are used, but major parts are referenced in many other standards.

4.3 Java TV

The Java TV API[15] is an extension of the Java platform. It provides functionality for using Java to control and run applications on TV receivers, such as set-top boxes. The main purpose of these extensions is to combine the platform independence of Java applications with a set of functions recommended for an iTV platform offered by TV-specific libraries. Furthermore, Java TV applications are independent of the underlying broadcast network technology. The JVM resides on the set-top box and allows a local execution of the applications, usually embedded in the broadcast content. Set-top boxes are often very resource-constrained devices. For that reason, the PersonalJava application environment, which is optimized for such devices, is used. PersonalJava offers a subset of the APIs introduced by the Java Standard Edition. PersonalJava applications are fully compliant with the Java standard edition. There are several packages of the PersonalJava application environment that are often used by Java TV applications. The *java.io* package is for input/output operations. It is used for file-based operations such as filesystem access (local and remote) and for stream-based operations such as broadcast data access. The *java.net* package is used for IP-based network access. These functions are often used for providing return channels or accessing IP data in the MPEG TS. Another important feature is security. For this purpose, Java TV makes use of the JDK 1.2 security model which lets operators define their own security model or policy. Most important for an iTV system, in terms of security, are areas like the conditional access sub-system, secured communication and the secure execution of code in the JVM. Based on the graphics toolkit, the abstract window toolkit (AWT) offered in the *java.awt* package, user interfaces (UI) can be build. AWT offers a set of basic UI components.

[14] The TV-Anytime Forum (`http://www.tv-anytime.org/`) developed open specifications for metadata with a focus on all participants in the TV value chain.

[15] The Java TV API - `http://java.sun.com/products/javatv/`

In the following several important aspects and functions of Java TV will be explained [5].

Service and Service Information (javax.tv.service)

A service is often used as a synonym for "TV channel". In Java TV, a service is handled as a unit. It represents a bundle of content (audio, video and data) that can be selected by the user and presented on the receiver. Service information (SI) describes the content and layout of a service. This information can be offered in several formats depending on the used standard such as DVB-SI or ATSC A56. Java TV offers a common API for accessing the service information. Services are composed by several service components. A service component represents one element of a service such as a video or a Java application. Besides these functions that are common to all services, more specialised features are available. The *navigation* subpackage offers classes to navigate through the existing services and to request detailed information about services and their components. The *guide* subpackage provides APIs for EPG. Basic EPG features like program schedules, program events, and rating information such as information for parental control are also included. The *transport* subpackage offers additional information about the transport mechanism that is used, such as MPEG-2 TS. The *selection* subpackage provides mechanisms to select discovered services for presentation. The service context, represented by the *ServiceContext* Class, provides an easy way for controlling the service and its presentation. The selection of a service context changes the presentation of the service and its components. For example a selection may cause the receiver to tune in to a desired service, demultiplex the necessary service components, present the audio and video, and launch any associated applications. A service context may exist in one of the following four states - "Not Presenting", "Presentation Pending", "Presenting", "Destroyed". Although the number of simultaneous ServiceContext objects is not limited by the specification, a limitation is often forced by resource constraints.

JMF

The Java Media Framework[16] (JMF), though not part of the Java TV API, is very important for Java TV. It provides the foundation for management and control of time-based media, such as video, audio and others, in Java TV. JMF offers a player component including a GUI for playback of audio and video streams which eases the integration and flexible placement of the video presentation. Also a set of controls such as a *GainControl* for manipulating audio signal gain is provided with JMF. In the Java TV API, only controls for video size and positioning (*AWTVideoSizeControl*) and for media selection

[16] The Java Media Framework (JMF) - `http://java.sun.com/products/java-media/jmf/`

(*MediaSelectionControl*) have been specified, but other controls may also be implemented. A set of useful additional controls was defined in DAVIC 1.4. Also the foundation for the synchronisation of the presentation is provided by a clock mechanism of JMF.

Broadcast Data API

The Java TV API provides access to different kinds of broadcast data. Broadcast data is transmitted beside the video and audio components embedded in the television broadcast signal. The first kind of broadcast data is the broadcast file system. For transmission, broadcast carousel mechanisms are usually used. In a broadcast carousel, all files are repeatedly transmitted in a cyclic way. The data access in Java TV is modeled as the access to a conventional read-only filesystem with high access latency. Predominant protocols in the area of broadcast filesystems are the Digital Storage Media Command and Control (DSM-CC) data carousel protocol, well known from the teletext service, and the DSM-CC object carousel protocol [18]. DSM-CC is an extension of the MPEG-2 standard. The other two kinds of broadcast data are IP datagrams and streaming data. IP datagrams, unicast and multicast, are accessed via the conventional functions of the *java.net* package. Streaming data is extracted and accessed via JMF.

Application Lifecycle

Java applications for digital receivers that use the Java TV API are called Xlets. Xlets have a similar concept to Java applets. Unlike normal Java applications, Xlets have to share the JVM with other Xlets, like applets do. Therefore Xlets have a special application life cycle model and a component, the application manager, that controls and manages their life cycle. There are four states defined in the life cycle of an Xlet. Xlets are optimized for the use on TV receivers. An Xlet also has an associated context, the *XletContext*. This context is an interface between the Xlet and its environment, similarly to the *AppletContext* for applets. This interface allows the Xlet to discover information about its environment, via properties, and to inform its environment about its state changes. JavaTV provides many interesting concepts for iTV systems. Especially the application model, introduced in JavaTV, is used in all major iTV standards. The Xlet concept paved the way for interoperable iTV applications.

4.4 MHP

The Multimedia Home Platform[17] was specified by the MHP group, a subgroup of DVB. This group was created in 1997 with the goal of developing a standard for a hardware and vendor independent execution environment for

[17] Multimedia Home Platform - `http://www.mhp.org/`

digital applications and services in the context of DVB standards. In July 2000 the first version of the MHP standard (MHP 1.0) was published by European Telecommunications Standards Institute[18] (ETSI) [12, v1.0.3]. Only one year later MHP 1.1 became an ETSI standard [11, v1.1.1]. The latest specification is version 1.2 [7], in which support for DVB-IPTV was added. Every new version of MHP extends the previous versions.

The MHP specification defines four profiles for different classes of functionalities [24, pages 337-339]. The profiles build upon each other. MHP 1.0 specifies only the first and the second profile. The third profile was introduced with MHP 1.1 and the fourth profile with MHP 1.2. (see also [23, chapter 1]).

1. Enhanced Broadcast Profile (Profile 1): The simplest version of an MHP environment supports the Enhanced Broadcast Profile. It is aimed for set-top boxes without a return channel in a low-cost area. This profile allows the development of local interactivity applications. Because of the lack of a return channel, applications may only be downloaded from the broadcast stream - the MPEG-2 Transport Stream. Typical applications based on this profile are electronic program guides, news tickers or enhanced teletext applications.
2. Interactive Broadcast Profile (Profile 2): Additional to the functions of Profile 1 this profile includes support for a standardised return channel. Based on the return channel an interaction between the audience and the broadcast becomes possible. This enables support for applications like televoting, T-commerce or pay-per-view applications. Another advantage of the return channel is that MHP 1.1 applications can also be downloaded over an Internet connection.
3. Internet Access Profile (Profile 3): In the Internet Access Profile, Profile 2 is extended with support for Internet applications. Only APIs for accessing different Internet services and applications rather than concrete services have been specified. By using this profile, typical point-to-point services like email and WWW can be combined with the broadcast world. Online games, chat- and email-applications are often provided based on it.
4. IPTV Profile (Profile 4): The most enhanced profile is the IPTV Profile. This profile integrates support for DVB-IPTV into the MHP platform. DVB-IPTV is formed by a collection of various specifications for the delivery of DTV using IP. There are various options such as the broadband content guide (BCG) available for extending the IPTV Profile. BCG specifies signalling and delivery of TV-Anytime [13] information.

Figure 1 presents an overview of the MHP architecture. According to this figure and the presented distinction into three layers the components will be described in the following paragraphs.

[18] European Telecommunications Standards Institute (ETSI) - `http://www.etsi.org/`

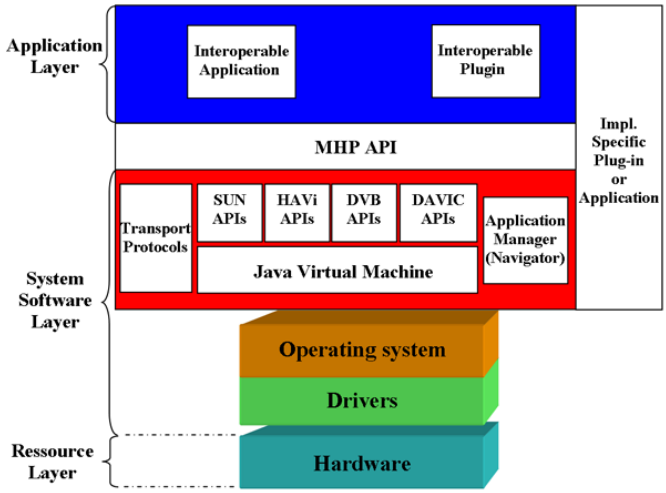

Fig. 1. MHP Architecture

Ressource Layer

It represents the different hardware platforms of set-top boxes. Besides different components like CPU, network interface and memory, the TV specific components like the DVB frontend and the MPEG-2 decoder module also reside on this layer.

System Software Layer

Based on the hardware platform the operating system manages the integration of the hardware and offers basic functions such as process management to the MHP middleware. The first component of the middleware is represented by the Java virtual machine. The use of Java offers a hardware independent common ground for the MHP API. The Java platform of MHP is also known as DVB-Java (DVB-J) [27, pages 199-236]. PersonalJava forms the SUN API part of DVB-J. The main cause for the use of PersonalJava in MHP is the small footprint of PJVMs which fits perfectly with the limited resources of most set-top boxes. Other important Java components are the Java Media Framework (JMF), the Java TV API and the Home Audio Video Interoperability (HAVi) specification. JMF is used for controlling audio and video content. HAVi (HAVi Level 2 GUI) forms the UI components of MHP because the abstract window toolkit of Java and its UI elements were not suitable for TV UIs. Another important component of the middleware is the Navigator which forms an application manager and offers all essential functions for a user to watch TV, e.g. listing and switching channels. The application model and many APIs providing access to DTV-specific functionality of MHP stem from the Java TV specification. The applications of MHP are called DVB-J

applications. The transport protocol's component in the figure consists of all parts and protocols that are necessary for communicating through different networks. TV specific protocols such as DVB-SI, DSM-CC and MPEG-2-TS and network protocols such as IP, TCP, UDP and HTTP can be found in this component.

Application Layer

As shown in figure 1, MHP is able to handle multiple applications in one JVM, on top of the MHP API. Besides the normal applications also plugins can be implemented for MHP, which are used to extend the functionality of the platform. There are two categories of applications and plugins in MHP: interoperable and non-interoperable. Interoperable applications and plugins may be used across all kinds of MHP receiver, on top of the MHP API. These DVB-J applications have been specified in Java TV. The application model of DVB-J applications follows the model of Java TV Xlets. In contrast to normal Java applications, Xlets have a similar concept to Java applets. Implementation specific applications and plugins are not interoperable. Native code or special Java APIs, which are not available in MHP, are commonly used. For that reason, such applications and plugins lose the ability of running on any kind of MHP receiver.

Besides DVB-J, there is another method for building interoperable MHP applications. The declarative language DVB-HTML for interactive TV applications has been specified since MHP version 1.1. The basic concepts of DVB-HTML such as the application life cycle had already been introduced in MHP 1.0 but has not been formalised in a specific language. The basic framework of DVB-HTML is based on a selection of XHTML 1.0 modules. For formatting, CSS level 2 is used. ECMAscript and DOM level 2 support form the basis of the dynamic aspects of DVB-HTML applications. The main reason for the introduction of DVB-HTML was that many companies had an expertise in HTML that they were willing to reuse, and that Java is not the best choice for creating presentation driven applications [23, chapter 15]. DVB-HTML is available for each of the three MHP Profiles but is most important for the Internet Access Profile.

MHP is already in use worldwide. Especially in Europe, MHP is the dominant iTV platform. Big supporters in Europe are Italy, Finland, Germany and others. Globally MHP gains importance by the fact that many other iTV specifications relate to MHP.

4.5 GEM

The Globally Executable Multimedia Home Platform (GEM) represents a subset of MHP. GEM 1.0 [10, v1.0.2] relates to MHP 1.0. It was published in the year 2003. GEM 1.1 relates to MHP 1.1 and the recent specification of GEM (1.2) [8] published in 2007 relates to MHP 1.2. The main purpose of

GEM is to enable organisations such as the CableLabs or the ATSC to define specifications based on MHP with the help of DVB. The goal is to guarantee that applications can be written in a way to be interoperable across all different GEM-based specifications and standards. GEM is not a standalone specification but a framework aimed at letting other organisations to define GEM-based specifications. GEM defines the APIs and content formats that can be used as a common basis in all interactive television standards. GEM also identifies and lists the components of MHP which are, from a technical or market perspective, specific to DVB. Other organisations are enabled to use GEM and define their own replacement of the DVB specific components as long as they are functionally equivalent. A specification in which only such DVB specific components are replaced is GEM compliant and capable of running MHP applications. Many organisations around the world have adopted GEM as the core of their middleware specification. Figure 2 shows the relationship and the functional replacements between GEM and ARIB, OCAP (cf. section 4.6) and ACAP (cf. section 4.7). GEM is referenced in its entirety in specifications like ACAP and OCAP. Differences and extensions have to be defined in detail.

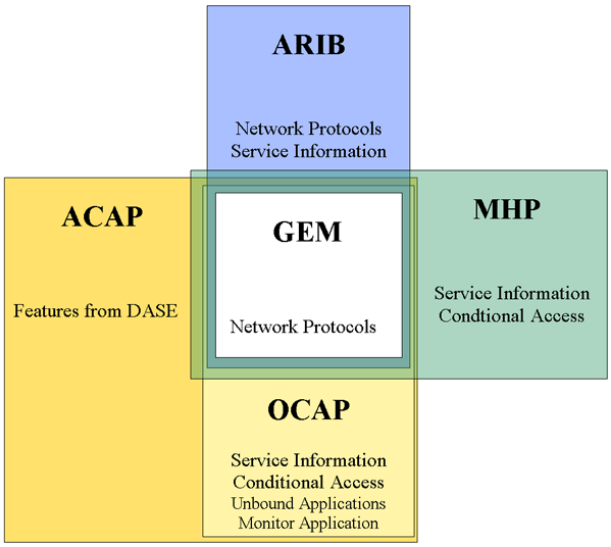

Fig. 2. Relationship between GEM and ARIB, OCAP and ACAP.

GEM seems to have the power to harmonise the iTV middleware market around the world and significantly facilitate the development of interoperable applications.

4.6 OCAP

The OpenCable Application Platform[19] (OCAP) is an open middleware standard for interactive TV. The first steps towards this standard were made by a non-profit organisation formed by many US cable television system operators called Cable Television Laboratories (CableLabs). The main goal of this initiative was to develop a middleware suitable across all set-top boxes of different vendors and all major cable TV system operators. When the work on OCAP started, its European counterpart, DVB-MHP, was still right on its way to standardisation. For that reason, DVB-MHP was investigated by the CableLabs and many parts where found to be suitable for OCAP. OCAP is largely based on MHP 1.0. Nevertheless, there are major differences between the distribution standards for digital TV used in the US and in Europe, leading to major differences in the distribution related middleware components. Also some restrictions made by the Federal Communications Commission[20] (FCC), an independent United States government agency, led to changes and extensions of the middleware components. The OCAP platform defines two profiles, OCAP 1.0 [4] and OCAP 2.0.

1. The first profile of OCAP (OCAP 1.0) was published in 2001. OCAP 1.0 defines the basic functionalities of OCAP. Over the years several versions of this profile have been published. In some versions changes were made on substantial components which led to the loss of backward compatibility between the versions. The most recent version of this profile is I16, which was released in August 2005.
2. OCAP 2.0 was first published in 2002. This profile extended OCAP 1.0 in several aspects. The most important difference with 1.0 was the inclusion of DVB-HTML support, based on the DVB-HTML extension of MHP 1.1.

In the following paragraphs, the middleware architecture of OCAP will be described. Figure 3 shows an overview of the OCAP architecture. The *Hosted Device Hardware* component of the figure represents the hardware platform of an OCAP set-top box. OCAP set-top boxes are typically hybrid analog/digital devices. This means, that such set-top boxes are able to support analog as well as digital services. The *Operating System* offers basic services such as task/process scheduling, memory management and forms a middleware layer between the hardware and the OCAP components. The major functionality of OCAP is offered by the *Execution Engine* and its various modules. Moreover, the engine provides a platform-independent interface built upon the JVM and a set of additional Java APIs. Java support in OCAP, also known as OCAP-J, is based on the DVB-J platform which was described earlier in section 4.4. We will now take a closer look at the following modules:

[19] OpenCable Application Platform (OCAP) - `http://www.opencable.com/ocap/`
[20] Federal Communications Commission (FCC) - `http://www.fcc.gov/`

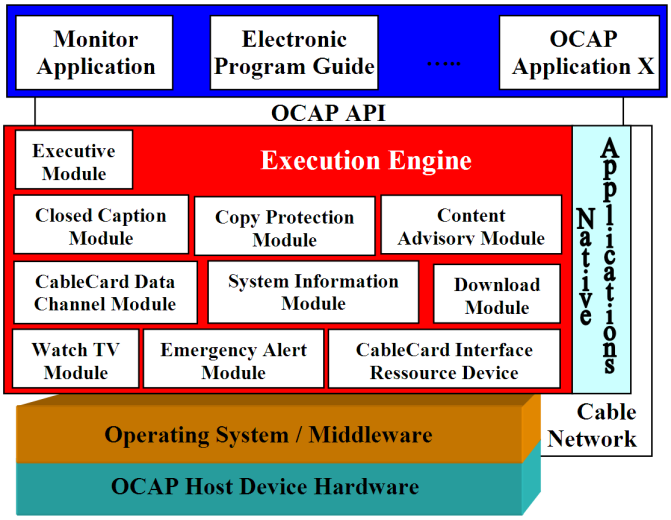

Fig. 3. OCAP Architecture.

Watch TV Module: This module offers the minimum functionality for watching TV such as switching channels. This module allows the user to watch all unencrypted channels.

Emergency Alert Module: This module is used to broadcast local or national emergency messages. Alert messages provided by the cable network operators force all receivers to show the alert message. This module is, based on the FCC rules for emergency alert system (EAS), a mandatory component of a receiver.

CableCard Interface Ressource Device: This module handles all messages of the CableCard hardware that require user-interaction, such as requesting the pin number, and satisfying the communication needs of applications via the module according to the CableCard Interface 2.0 Specification.

CableCard Data Channel Module: This module offers baseline functionality for processing data on the CableCard Data channel.

System Information Module: The System Information Module keeps track of service information. After parsing the service information, it makes the information accessible to other modules and OCAP applications. Special types of information such as emergency alert system messages are directly forwarded to the appropriate module for further processing.

Download Module: The Download Module keeps track of new available updates for the set-top box. It empowers network providers to update their set-top boxes. This is a very important function and represents the only way to get rid of erroneous firmware versions when the set-top boxes are already at the customer's home.

Closed Caption Module: The presentation of closed caption text is the main purpose of this module. It is part of the core functions and should work

regardless of any extension of the network operator. It is also mandated by the FCC for all analog TV services.

Copy Protection Module: The module controls copying of analog and digital content. It controls the storage and the output of digital and analog content according to the copy control information (CCI) delivered by the conditional access (CA) system.

Content Advisory Module: The V-Chip functionality, mandated by the FCC, is handled by the Content Advisory Module. The V-Chip provides the ability to block the display of television programs based upon its rating. This offers a parental control upon the TV consumption of children. The module decodes the V-Chip signal provided via the analog broadcast and offers the rating to other modules.

Executive Module: The Executive Module is responsible for launching and controlling applications. Also the management of stored applications is handled by it. It plays a major role during the boot-up of the set-top box and starts the initial Monitor application, if one is available. If no Monitor application is available, it is responsible for controlling the receiver. While a Monitor application is running, the executive module monitors the life cycle of the Monitor application and re-launches it if it is destroyed.

OCAP applications are in many ways similar to MHP applications. The *Monitor* application, also called Monitor, plays a special role in an OCAP set-top box and represents a unique feature of OCAP. It is implemented as a single OCAP-J application or is made up of a set of applications cooperating with each other. The Monitor application has a privileged access to several APIs which are not accessible for normal applications. It helps to manage and control the life cycle of OCAP applications, cares about resource management and security issues. Monitor applications are provided by the cable television system operators and downloaded to the set-top box when it connects to the cable network for the first time. The full functionality of an OCAP set-top box is just available when the Monitor application of the network operator is present. In most cases, just the basic functions for watching DTV and using unencrypted services are available without a Monitor. By providing this application, a system operator gains much control over the set-top boxes. Further customisation can be made by the network operator because of several assumable modules of the Execution Engine such as the Watch TV Module, the Emergency Alert System Module, and so on. The Monitor application may assume the functionality of these modules. By using this feature a network operator is able to replace several functions of the box by his own implementations and create his own, branded OCAP platform. For example, the Emergency Alert System could present alert messages with a special layout or allow switching over to a channel where further information on the alert is provided[23, pages 47-52].

As mentioned before the OCAP standard has a clear focus on cable networks. For that reason OCAP is widely used by US cable TV system opera-

tors. Although OCAP is an open and mature standard a global use is unlikely because of several US-market specific characteristics.

4.7 ACAP

The Advanced Common Application Platform[21] (ACAP) is a middleware specification for iTV applications [2]. It was developed and standardised by the Advanced Television Systems Committee (ATSC), a US non-profit organisation dedicated to the development of standards for DTV. The ATSC is formed by members of the television industry ranging from broadcasters to the semiconductor industry. The ACAP standard (A/101[22]) was published in 2005. ACAP is primarily based on GEM and DTV Application Software Environment Level 1 (DASE-1) [1] but also makes use of OCAP functionalities. Many sections of the ACAP specification are references to parts of these iTV standards. An important capability of ACAP is the support of all US DTV systems, cable, satellite and terrestrial television networks. ACAP was the first attempt to harmonise the US iTV market for cable and terrestrial TV.

There are two different types of ACAP applications, procedural applications called ACAP-J and declarative applications called ACAP-X. In general ACAP-J applications are Java TV Xlets and ACAP-X, the "X" stems from XHTML, applications are very similar to DVB-HTML applications. ACAP is structured in two profiles. The first profile supports only the first application type, ACAP-J. The second profile adds support for ACAP-X applications. ACAP is based on GEM and DASE. For that reason the architecture of ACAP and its components look very similar to DASE and parts of GEM/MHP. Nevertheless there are differences between the broadcast system specific parts and parts stemming from OCAP.

4.8 Other platforms

Besides the presented open standards, several proprietary solutions for iTV middleware exist. A widespread solution is OpenTV[23]. Other products are MSTV[24] by Microsoft and MediaHighway[25] by NDS. Since these platforms are proprietary, a further description is out of scope of this paper.

5 History and Future of iTV Standards

In their latest or most advanced versions all of the presented standards feature comparable capabilities. The main differences lie in the initial objectives of

[21] Advanced Common Application Platform (ACAP) - `http://www.acap.tv/`
[22] ACAP Standard - `http://www.atsc.org/standards/a101.html`
[23] OpenTV - `http://www.opentv.com/`
[24] MSTV - `http://www.microsoft.com/tv`
[25] MediaHighway - `http://mediahighway.nds.com/`

the specifications. Goals were the development of representation languages, middleware specifications and all-round standards. Also regional distinctions led to differences and additional components. Nevertheless all the standards are strongly interrelated, as shown in figure 4. Besides several iTV standards and the HAVi specification Java plays an special role in figure 4. The application model for nearly all iTV applications is Java-based. For that reason a relation between Java and all other presented iTV standards is present. In figure 4 this relation is only indicated by arrows from Java to MHEG, DAVIC and JavaTV and the relation of these standards to all other standards. The first open standard for iTV was MHEG, initially only planned as a description language for multimedia presentations. A scripting language was included soon, in order to allow the implementation of advanced applications. In spite of other versions of the MHEG specifications, MHEG-5 and its extension MHEG-6 are the most relevant parts. These were partly included in the DAVIC specifications, the industry standard for interactive digital audiovisual applications and broadcast. Like DAVIC, MHEG-6 also makes use of the Java programming language to maximise interoperability. The JavaTV API was developed to provide a pure Java environment to control and run applications on TV receivers, such as set-top boxes. Although figure 4 does not

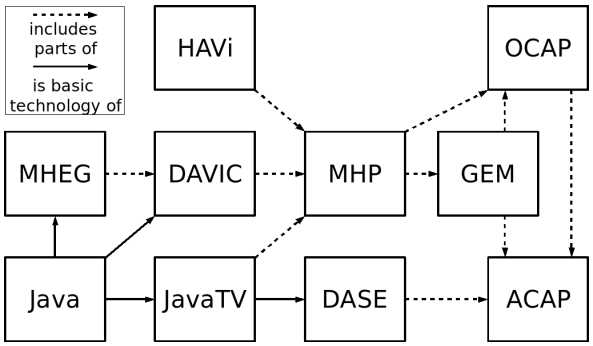

Fig. 4. Relationships between the Presented Standards.

indicate a relation between DAVIC and JavaTV, several concepts of DAVIC (e.g. controls defined in DAVIC) are used. MHP is a comprehensive specification for iTV middleware platforms. It includes parts and concepts of DAVIC, JavaTV and several parts of HAVi (e.g. UI components). Also a knowledge transfer in both directions between the standardisation of MHP and JavaTV took place. Specifically for cable networks in the US, the OCAP standard was introduced. OCAP specifies a middleware platform with several cable network and FCC specific components. The OCAP standard reuses major parts of the MHP specification. Since many standards reference MHP, a subset - GEM - was defined. GEM forms a common core for many MHP related standards. As indicated in figure 4 GEM is fully included in new versions of the OCAP

specification and ACAP. ACAP is a very young standard which supports all common DTV systems in the US. Since ACAP was developed by the ATSC the former ATSC standard DASE was included in ACAP.

The emergence of the iTV standards and especially GEM shows the trend to a harmonisation of the iTV market. In the near future the slogan of Java - "Write Once, Run Anywhere" - will be true also for applications in the world of open iTV standards.

References

1. Advanced Television Systems Committee. *DTV Application Software Environment Level 1 (DASE-1) Part 1: Introduction, Architecture, And Common Facilities*. Advanced Television Systems Committee, Washington, D.C., US, 2003.
2. Advanced Television Systems Committee. *Advanced Common Application Platform (ACAP)*. Advanced Television Systems Committee, Washington, D.C., US, 2005.
3. British Bradcasting Corporation. *Digital Terrestrial Television MHEG-5 Specification*. British Bradcasting Corporation, London, UK, 2003.
4. Cable Television Laboratories. *OpenCable Application Platform Specification - OCAP 1.0 Profile*. Cable Television Laboratories, Louisville, USA, 2005.
5. B. Calder, J. Courtney, B. Foote, L. Kyrnitszke, D. Rivas, C. Saito, J. VanLoo, and T. Ye. Java TV API Technical Overview: The Java TV API Whitepaper. Technical report, Sun Microsystems, Inc., 2000.
6. Digital Audio-Visual Council. *Statutes of The Digital Audio-Visual Council*. Digital Audio-Visual Council, Geneva, Switzerland, 1994.
7. DVB Project. *Digital Video Broadcasting (DVB); Multimedia Home Platform (MHP) Specification 1.2*. DVB Project, Geneva, Switzerland, 2007.
8. DVB Project. *GEM 1.2 (including IPTV)*. DVB Project, Geneva, Switzerland, 2007.
9. European Telecommunications Standards Institute. *ETS 300 715: Terminal Equipment (TE); MHEG script interchange representation (MHEG-SIR)*. European Telecommunications Standards Institute, Sophia Antipolis, France, 1994.
10. European Telecommunications Standards Institute. *ETS ES 102 819: Globally Executable MHP version 1.0.2 (GEM 1.0.2)*. European Telecommunications Standards Institute, Sophia Antipolis, France, 2005.
11. European Telecommunications Standards Institute. *ETS ES 102 812: Digital Video Broadcasting (DVB); Multimedia Home Platform (MHP) Specification 1.1.1*. European Telecommunications Standards Institute, Sophia Antipolis, France, 2006.
12. European Telecommunications Standards Institute. *ETS ES 201 812: Digital Video Broadcasting (DVB); Multimedia Home Platform (MHP) Specification 1.0.3*. European Telecommunications Standards Institute, Sophia Antipolis, France, 2006.
13. European Telecommunications Standards Institute. *ETSI TS 102 822-3-1: Broadcast and On-line Services: Search, select, and rightful use of content on personal storage systems ("TV-Anytime"); Part 3: Metadata; Sub-part 1: Phase 1 - Metadata schemas*. European Telecommunications Standards Institute, Sophia Antipolis, France, 2006.

14. International Organisation for Standardisation. *ISO/IEC 13522-4:1996: Information technology – Coding of multimedia and hypermedia information – Part 4: MHEG registration procedure.* International Organisation for Standardisation, Geneva, Switzerland, 1996.
15. International Organisation for Standardisation. *ISO/IEC 13522-1:1997: Information technology – Coding of multimedia and hypermedia information – Part 1: MHEG object representation – Base notation (ASN.1).* International Organisation for Standardisation, Geneva, Switzerland, 1997.
16. International Organisation for Standardisation. *ISO/IEC 13522-3:1997: Information technology – Coding of multimedia and hypermedia information – Part 3: MHEG script interchange representation.* International Organisation for Standardisation, Geneva, Switzerland, 1997.
17. International Organisation for Standardisation. *ISO/IEC 13522-5:1997: Information technology – Coding of multimedia and hypermedia information – Part 5: Support for base-level interactive applications.* International Organisation for Standardisation, Geneva, Switzerland, 1997.
18. International Organisation for Standardisation. *ISO/IEC 13818-6:1998: Information technology – Generic coding of moving pictures and associated audio information – Part 6: Extensions for Digital Storage Media Command and Control (DSM-CC).* International Organisation for Standardisation, Geneva, Switzerland, 1997.
19. International Organisation for Standardisation. *ISO/IEC 13522-6:1998: Information technology – Coding of multimedia and hypermedia information – Part 6: Support for enhanced interactive applications.* International Organisation for Standardisation, Geneva, Switzerland, 1998.
20. International Organisation for Standardisation. *ISO/IEC 13522-7:2001: Information technology – Coding of multimedia and hypermedia information – Part 7: Interoperability and conformance testing for ISO/IEC 13522-5.* International Organisation for Standardisation, Geneva, Switzerland, 2001.
21. International Organisation for Standardisation. *ISO/IEC 13522-8:2001: Information technology – Coding of multimedia and hypermedia information – Part 8: XML notation for ISO/IEC 13522-5.* International Organisation for Standardisation, Geneva, Switzerland, 2001.
22. T. Meyer-Boudnik and W. Effelsberg. MHEG Explained. *IEEE MultiMedia*, 2(1):26–38, 1995.
23. S. Morris and A. Smith-Chaigneau. *Interactive TV Standards.* Focal Press, Burlington, USA, 2005.
24. U. Reimers. *The Family of International Standards for Digital Video Broadcasting.* Springer, Heidelberg, Germany, 2004.
25. Roger Price. MHEG: an introduction to the future international standard for hypermedia object interchange. In *Proceedings of the First ACM International Conference on Multimedia*, pages 121–128, New York, NY, USA, 1993. ACM Press.
26. G. Ruhrmann and J.-U. Nieland. *Interaktives Fernsehen.* Westdeutscher Verlag GmbH, Wiesbaden, Germany, 1997.
27. S. Rupp and G. Siegmund. *Java in der Telekommunikation.* dpunkt.verlag, Heidelberg, Germany, 2003.