An Integrated, Interactive Application Environment for Session-Oriented IPTV Systems, Enabling Shared User Experiences

vorgelegt von
Diplom-Ingenieur
Oliver Friedrich
aus Berlin

Von der Fakultät IV - Elektrotechnik und Informatik
der Technischen Universität Berlin
zur Erlangung des akademischen Grades

Doktor der Ingenieurwissenschaften
- Dr.-Ing. -

genehmigte Dissertation

Promotionsausschuß:

Vorsitzender: Prof. Dr. Olaf Hellwich
Berichter: Prof. Dr. Thomas Magedanz
Berichter: Prof. Dr. Philippe Palanque
Berichter: Prof. Dr. Axel Küpper

Tag der wissenschaftlichen Aussprache: 27. Juni 2011

Berlin 2011
D 83

Abstract

The Internet is continuously developing in the direction of becoming a platform for the delivery of video and entertainment services. Internet Protocol Television (IPTV) services, including Video-on-Demand (VoD), Linear TV and corresponding interactive TV services are already part of nearly all current Internet service offerings. Due to consumers' rising demands for Internet video portals like YouTube, which are not necessarily part of operators' edge networks, the entire Internet is currently heading towards an end-to-end media delivery platform.

Current IPTV offerings provide services already known from Digital TV. In addition to Linear TV and Video on Demand, a certain amount of interactivity using the IP back-channel is provided. In contrast to those, innovative services like Social Networking come from the Internet. What has been missing so far is an integrated approach combining both worlds, including the above-mentioned common streaming services, interactivity, rich media communications and novel aspects of Over-The-Top services like Social Networking.

In this thesis, such an extended approach to IPTV is introduced. Herewith the consumer, as well as third-party service providers will be enabled to create, manipulate and enhance services and interact with other consumers and services in the end. Based on a profound state-of-the-art and requirements analysis, the author proposes a novel session-oriented core system for IPTV. The platform therefore reuses concepts from session-oriented conversational services specified for Next Generation Networks (NGNs), incorporating the Session Initiation Protocol (SIP).

First, different approaches for so-called *Interactive Application Environments* are analyzed. Then two *Interactive Application Environments* are selected for integration into the proposed system called the *Open IPTV Ecosystem Core*. Second, the corresponding architecture will be derived through the specification of functional entities, interfaces and protocols, which allow for the consumption, creation, manipulation and interaction with other consumers and services.
Furthermore, the described architecture and exemplary interactive services are implemented. Finally, the resulting infrastructure is validated through different case studies.

The work on this thesis was carried out at the Fraunhofer Institute for Open Communication Systems (FOKUS). The results have been used within the scope of multiple international research projects either funded by the European Commission or research partners from industry. Furthermore, they have partly contributed to the IPTV standardization process within the Open IPTV Forum and ETSI TISPAN.

Zusammenfassung

Das Internet entwickelt sich derzeit zu einer Plattform für multimediale, hochauflösende Video- und Unterhaltungsdienste.

Fernsehen über das Internetprotokoll, sog. Internet Protocol Television (IPTV), ist bereits heute Bestandteil vieler Internetpakete. Dies beinhaltet sowohl klassische lineare TV-Inhalte, als auch Video-on-Demand–Angebote, kombiniert mit ersten interaktiven Diensten.

Die derzeit verfügbaren Angebote für IPTV unterscheiden sich kaum von klassischen Digital-TV-Angeboten, da diese derzeit nur bedingt von dem über das Internet verfügbaren Rückkanal für interaktive Dienste Gebrauch machen.

Im Gegensatz hierzu entwickeln sich im Internet neue und innovative Dienste, die vornehmlich unter dem Stichwort Social Media zusammengefasst werden. Eine Verschmelzung beider Ansätze erfolgt derzeit nur zögerlich, wobei der Kombination von Bewegtbild, Interaktivität, Kommunikationsdiensten und Sozialen Medien eine erfolgversprechende Zukunft vorausgesagt wird.

Die vorliegende Arbeit beschreibt einen solchen kombinierten Ansatz, mit dem der Nutzer und der Dienstanbieter in die Lage versetzt werden, kombinierte Mediendienste zu nutzen und bereitzustellen, sowie mit Kommunikationsdiensten anzureichern und zu beeinflussen.

Die technische Basis bilden hierbei Konzepte, abgeleitet von sog. Next-Generation-Networks (NGNs) zur Bereitstellung von multimedialen Diensten über das Internetprotokoll.

Das sog. Session-Initiation-Protokoll (SIP) stellt die entsprechenden Funktionalitäten zur Kontrolle und Signalisierung der Dienste bereit. Das im Rahmen der Arbeit entwickelte Kernsystem – der sog. Open IPTV Ecosystem Core – besteht weiterhin aus einer integrierten Laufzeitumgebung für interaktive Dienste, die aus existierenden Ansätzen abgeleitet und zu einem neuen und innovativen System erweitert wird.

Die hierbei entwickelte Architektur spezifiziert funktionale Komponenten, Schnittstellen und Protokolle, die den oben beschriebenen Konsum und die Erstellung von interaktiven Inhalten, sowie die Interaktion mit anderen Nutzern und Diensten erlauben.

Des Weiteren erfolgt die exemplarische Entwicklung von neuen interaktiven Diensten, sowie deren Validierung durch mehrere Fallstudien.

Die vorliegende Arbeit wurde im Rahmen der Tätigkeit des Autors als wissenschaftlicher Mitarbeiter am Fraunhofer Institut für Offene Kommunikationssysteme (FOKUS) erstellt. Die Ergebnisse wurden hierbei im Rahmen mehrerer internationaler Forschungsprojekte erarbeitet und publiziert. Die Projekte wurden durch Partner aus der Industrie bzw. der Europäischen Kommission gefördert.

Des Weiteren flossen Ergebnisse in offene Standards für interaktives IPTV der ETSI und des Open IPTV Forums (OIPF) ein.

Acknowledgements

This thesis was written at the Fraunhofer Institute for Open Communication System, between April 2007 and February 2011 under the supervision of Professor Thomas Magedanz.

First of all, I would like to thank Professor Magedanz for his encouragement and constant feedback during my work on this thesis and his assistance with completing this work.

Second, I would like to thank my colleagues Dr. Stefan Arbanowski, Robert Seeliger and Benjamin Zachey, for their feedback on and support in creating the test bed infrastructure for the concepts described in this thesis.

I also deeply appreciate the contributions of Dr. Adel Al-Hezmi, who shaped my understanding in this field of research from the beginning,

My special thanks go to Dr. Regina Bernhaupt and Professor Philippe Palanque for their very helpful early reviews of my work, instant feedback and ongoing support during writing on this thesis.

I also want to thank Professor Axel Küpper for his intermediate review and support during finalization of this work.

Finally, I want to thank my family and friends, especially Elisabeth, who always kept their faith in me during the work on this thesis.

Berlin, Germany, June 2011

Oliver Friedrich

Contents

List of Figures

List of Tables

1. Introduction

Chapter 1: Introduction

Background & Motivation
Problem Statement,Objectives & Scope
Research Questions & Outline

1.1. Background & Motivation

In the near future, services will integrate TV and video with telecommunication, social media and personalized interactivity.

From a user's perspective, the resulting infrastructures will allow for so-called cross-domain shared experiences, where the user can interact and share services with other people independent of location, device or access network.

From a content, service or broadcaster's perspective, these infrastructures will allow for the control, enrichment and modification of the described cross-domain shared experiences with additional information. This information might contain related audio or video contents, metadata or even personalized services like games or targeted advertisements. So called *Interactive Applications Environments* provide the technological basis for the execution of services in these environments.

The overall goal of this thesis is to realize, design, implement and evaluate such a framework for cross-domain shared experiences and to integrate it with a future media-enabled Internet.

The Internet is continuously developing in the direction of becoming a platform for the delivery of video and entertainment services. *Internet Protocol Television* (IPTV) services, including Video-on-Demand (VoD), Linear TV and corresponding interactive TV services, are part of nearly all Internet service offerings nowadays.

With the goal of overcoming the Internet's current limitations, the European Commission's Networked Media Research presented the so called *"Challenges for Mastering the Media Revolution"* [112] in 2007. Various groups such as the *Media Future Internet Workgroup*, the *Networked Media Task Force* [9], the *Media Delivery Platforms Cluster* [142] and the *User Centric Media Cluster* [79] stated their visions and defined the targeted outcomes as follows:

"Create an *interoperable multimedia network* and *service infrastructures* which offer a seamless, personalised and trusted experience of:

i) *multimedia services* and *applications.*
ii) *media content, for users in a variety of roles* (consumer, producer or manager of communication and media), *locations, contexts* and *mobility scenarios.*
iii) enable variable media distribution patterns between multiple users"

and

"End-to-end systems and application platforms that enable:

i) *intuitive, intelligent, professional and non-professional creation, manipulation, storage/handling/search, management and rendering of media*
ii) *new creative forms of interactive, immersive and high quality media* as well as
iii) *new forms of experiences for individual users or user communities.*"[20]

Altogether, the above-mentioned vision statement describes *the convergence of media and communications* emerging from existing *Technology Domains.*

As depicted in Figure 1.1, the key drivers of this vision are groups that previously did not need to interact in order to be successful [112]. This includes *Telecommunications*, *Information Technology*, *Digital Broadcast, Media & Content Providers* and *Consumer Electronics.*

To enable the vision described above, a combination of the five key drivers expertise is required:

Telecommunications provide their expertise in building (mobile) broadband networks, assuring the efficient delivery of new rich content services created by broadcaster, as well as media & content providers. Integrated offers and pervasive services can be created just in cooperation between these three drivers. The presentation of services will happen on devices designed and developed, in-line with these visions, by the consumer electronics industry. Furthermore, State-of-the-Art information technology assures the provisioning of innovative and pervasive services.

A detailed view concerning the involved *Technology Domains* as depicted in Figure 1.2 shows where and how new services in these rich media environments will be enabled:

- *Multimedia Networks* allowing a seamless access anywhere/anytime, and allowing a context-aware, collaborative any-to-any and ad-hoc communication.

- *Multimedia Content Handling and Management* enabling easy content distribution, personalization, adaptation, management, storage, retrieval and

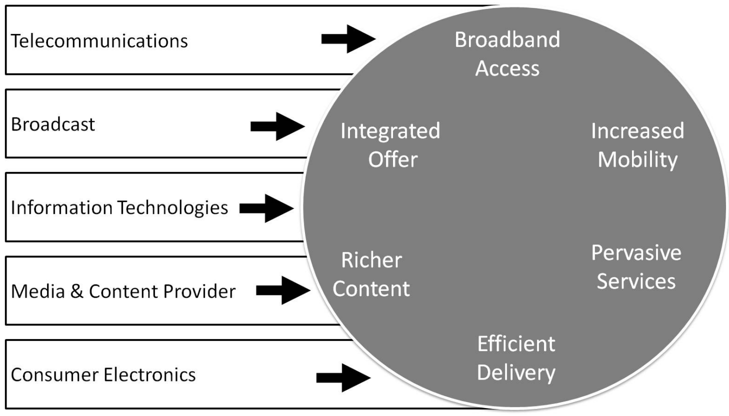

Figure 1.1.: Technology Domains in Networked Media [112]

search of contents and services provided by broadcasters and media & content providers.

- *Multimedia Applications and Experiences* allowing for user-centric applications, user-initiated content creation and sharing of interactive high quality interactive and immersive media.

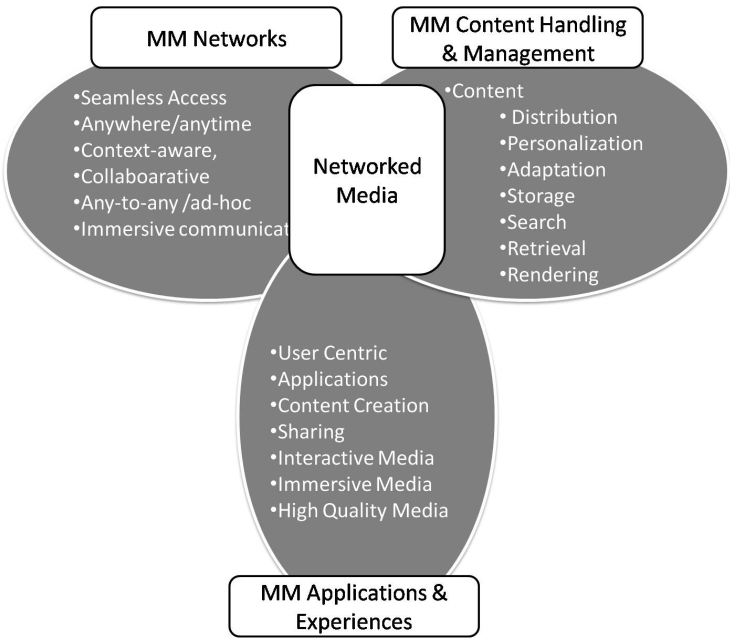

Figure 1.2.: Networked Media: Access, Services & Applications [112]

1.2. Problem Statement

The last sections described the current gap between existing technology domains involved in the development of interactive IPTV and related services. Furthermore, what kind of service experience a resulting integrated infrastructure might be able to deliver has been outlined. The question of how to realize such an interaction between the described technology domains shapes the main problem addressed by this thesis.

This problem will be addressed in detail by dealing with the different key aspects, allowing for a combinatorial approach to the various technology domains:

1. The business models and technological requirements of telecommunication, broadcasters and content providers and consumer electronics differ with respect to preferences for a certain technology, e.g. protocol or programming paradigm, over another when creating services. Nevertheless, services should be available and work across the boundaries of platform providers, also when offered from one or the other party.

2. An integrated architectural approach allowing for the generation of hybrid services embracing Web, telecommunication, content and broadcast technology for IPTV is lacking. This thesis combines the work of various *Standard Development Organization* (SDOs), working on this topic and provides one possible solution.

3. Runtime environments for interactive services do exist but do not reflect the requirements of a combinatorial approach like the one described above. This thesis provides one solution for the combination of two different application environments into one system.

1.3. Objectives and Scope

The overall objective of this thesis is to design, implement and evaluate a *session-oriented* ene-to-end platform for current IPTV networks and a future media-enabled Internet.

Furthermore an *Interactive Application Environment* for interactive IPTV services, combining session-oriented and Web-driven declarative approaches will be designed. On top, interactive and *Social IPTV* services for the developed system will be specified and evaluated. Interactive contents will be provided through the re-use and extension of Open Source technology.

The specified system will be named the *Open IPTV Ecosystem Core*.

It is out of the scope of this thesis to question how streaming services can be delivered seamlessly over multiple access networks, including necessary optimizations to the session principles and delivery networks. These aspects will be discussed in another thesis[1] created in parallel to this work.

Based on the functional and non-functional requirements stated as vision statements in the introduction, this thesis will describe one specific approach to the creation of a session-oriented platform for streaming, interactive and rich media communication services. Furthermore, how content providers, advertisers, telecom operators, consumer electronics manufacturers and service providers will become part of this platform is described.

From a technical point of view, the objective of this thesis is to exploit baseline technologies for the control and delivery of media and telecommunication services towards new and integrated experiences [55].

In order to achieve the above mentioned objectives, this thesis focuses on pointing out the implications of different technological approaches for the involved parties.

Figure 1.3 helps to identify the main aspects discussed in this thesis, by providing an overview of the variances of IPTV and highlighting the two main areas discussed here:

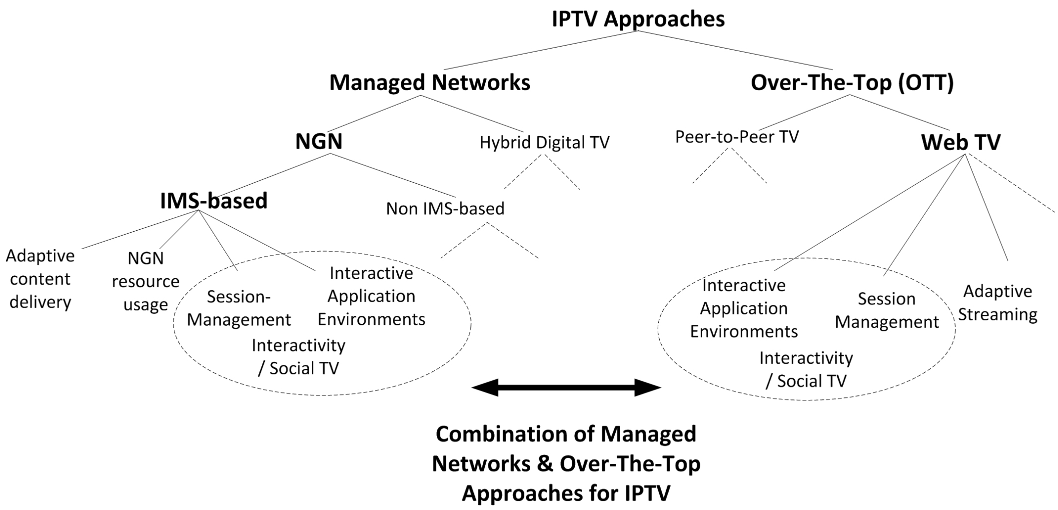

Figure 1.3.: Taxonomy for Interactive IPTV (Scope of this work highlighted)

IPTV is defined as multimedia services such as television, video, audio and the corresponding data for interactive services delivered over IP-based networks. Furthermore, a certain level of Quality-of-Service (QoS), Quality-of-Experience (QoS), security, interactivity and reliability will be provided.

Nowadays, both so called Managed Networks, i.e. closed telecom operator's networks, as well as Over-The-Top (OTT) approaches, using the open Internet, are

[1]Please refer to the PhD thesis "Efficient Session-based Multimedia Content Delivery in Next Generation Networks" by Adel Al-Hezmi. Technische Universität, Berlin 2011.

able to guarantee these metrics[2].

Below, IPTV over Next Generation Networks (NGNs) [35], Hybrid TV [37] using IPTV and conventional Digital TV in parallel, Peer-to-Peer approaches over IP and Web TV streaming services, can be identified.

This thesis will focus on the *Session Management* and *Interactive Application environments* enabling *Interactive* and *Social IPTV* in NGN and Web TV environments. The NGN approach chosen in this thesis will furthermore be limited to IMS-based IPTV. The IMS (IP Multimedia Subsystem) hereby represents an instantiation of an NGN system.

Session Management summarizes all aspects necessary for the maintenance of a user's requests for multimedia content, including interactive services. *Interactive Application Environments* are runtime engines for interactive services, like games, user polls or targeted advertisements, enriching plain streaming service consumption. *Social IPTV* combines interactive services, media and communication aspects into a new form of interactivity, enabling so-called shared experiences between groups of users.

1.3.1. IPTV Role Model

As part of the platform being developed, a set of entities, and the communication aspects between them, will specified. The developed session-oriented *IPTV Role Model* in Figure 1.4 provides a visualization of a blueprint of the overall scope of this thesis including the role of all involved parties and their mutual dependencies.

This drawing represents a logical separation between different roles, whereas a single party could fulfill different roles in parallel, e.g. acting as IPTV and Interactive Service Provider simultaneously. This approach follows the ideas of a *Future Media Internet*, where the roles of the involved entities are not as static as in the NGN domain [78].

IPTV Service Provider The IPTV Service Provider plays a central role in all IPTV networks. It is responsible for managing the contractual relationship with the customer and maintains incoming service requests from the user. Furthermore, it acts as a gatekeeper for *Content Providers* and *Interactive Service Providers*. Finally, it manages and controls the *Content Delivery Network* (Provider).

User The user consumes interactive IPTV services on one or multiple *End Devices*. This might include TVs, Set-Top-Boxes (STBs), mobile devices, tablets and Personal Computers. From a technical perspective, the user requests services through his *End Devices* at the *IPTV Service Provider*. These services will originate from a *Content*

[2] The original definition is maintained by ITU-T's IPTV Joint Coordination Activity (IPTV-JCA) at http://www.itu.int/ITU-T/jca/IPTV

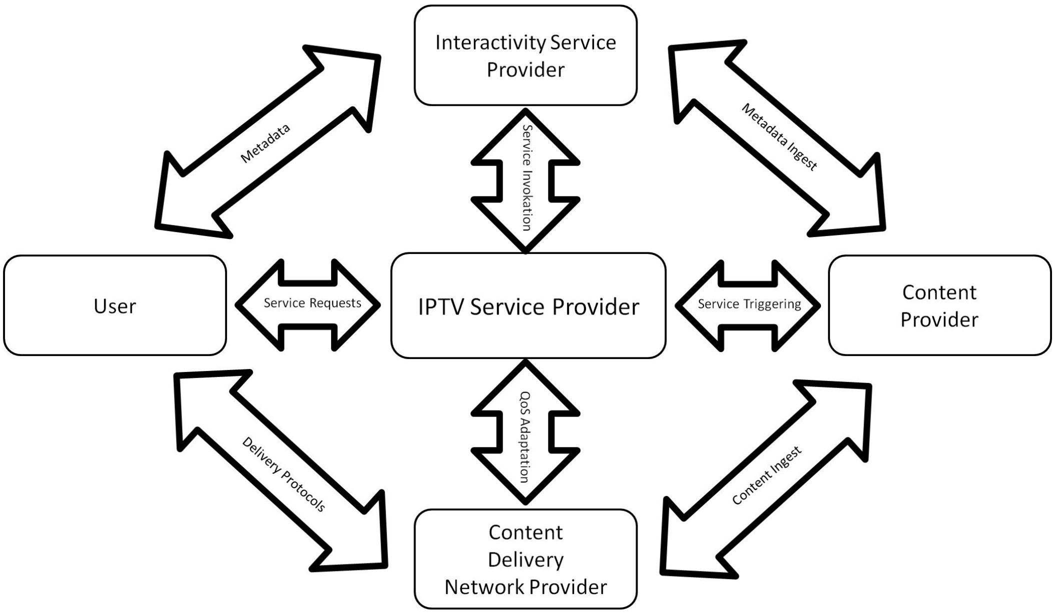

Figure 1.4.: Session-Oriented IPTV Role Model and Overall PhD Scope

Provider or *Interactivity Service Provider* or both, and will be delivered by a *Content Delivery Network Provider*.

Interactivity Service Provider *The Interactivity Service Provider* enriches streaming services with interactive, personalized features. For this reason, it has access to data collected by the *IPTV Service Provider*, allowing it to personalize its service usage. Furthermore, communication features and *Social Media* services will be combined with the plain content related services, e.g. to create shared experiences between different users.

Content Provider *The Content Provider* injects Live TV and/or video services into the *IPTV Service Providers's* platform. Furthermore, it provides necessary metadata. *Content Providers* could also participate in generating interactive services by providing the necessary hooks and trigger points for interactive services within their contents, i.e. for *Red Button* services.

Content Delivery Network Provider The *Content Delivery Network Provider* maintains the IP delivery network for all IPTV as well as interactive services. Content is ingested by the different *Content Providers*, and managed by the *IPTV Service Provider* through Quality-of-Service (QoS) metrics.

1.3.2. Generic Multimedia Service and Delivery Framework

Using the session-oriented *IPTV Role Model* from the last section, the derived entities and their relationships have been mapped onto a conceptual architecture visualized in Figure 1.5. The functional roles from above have been translated into functional entities described in the following. This architecture has been named the *Generic Multimedia Service and Delivery Framework*.

This reference architecture has been inspired by another reference model developed for Next Generation Networks[3] (NGNs).

The NGN reference model provides a layered approach divided into the *NGN Transport Stratum* for delivery aspects, the *NGN Service Stratum* for service control aspects and an application layer on top. Figure 1.5 also shows how the NGN stratums are reflected in the corresponding architecture developed in this thesis. Furthermore, the dotted line in Figure 1.5 outlines the main aspects discussed in this thesis compared to the aspects discussed by the second thesis created in parallel:

As this thesis concentrates on how interactive services and *Social Media* aspects can be realized on a session-oriented IPTV system, the thesis developed in parallel to this work concentrates on content delivery aspects and their optimization for different access networks. *IPTV Session Management* plays a key role in both works.

The baseline for this architecture has been developed jointly between the authors during various research projects as published in [4] and has furthermore been contributed to the IPTV standardization process within ETSI TISPAN in 2006 [52].

IPTV Session Management The *IPTV Session Management* acts as the central building block of the *Open IPTV Ecosystem Core,* managing users and their requests for content and interactive services. It therefore provides interfaces for the *Content Provisioning* and *Interactivity* component, allowing for the triggering and manipulation of service usage. These interfaces for *Third Party Access* are also used by the *Meta Session* functionality, enabling the Social IPTV features.

Interactivity The *Interactivity* building block contains runtime environments for interactive and *Social IPTV* applications and services. *Social IPTV* represents *Social Media* and social communications in IPTV environments, and enables shared user experiences connecting users, content and services.

End Device The *End Device* building block acts as a placeholder for various devices acting as endpoints for interactive and *Social IPTV* services. This might include TVs, Set-Top-Boxes (STBs), mobile devices, tablets and Personal Computers.

[3]http://www.itu.int/ITU-T/NGN

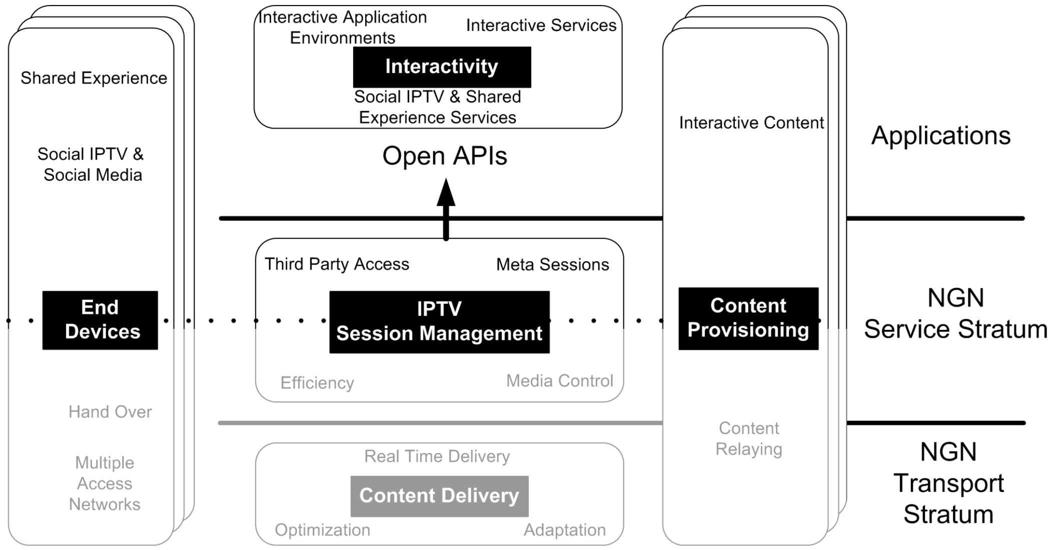

Figure 1.5.: Generic Multimedia Service And Delivery Framework

Content Provisioning The *Content Provisioning* building block represents *Content Providers* and corresponding services.

Content Delivery The *Content Delivery* building block represents different IP-based delivery networks.

1.3.3. Research Questions

To narrow the scope of this thesis, one general and four research questions have been postulated by the author. The following overview lists these five questions. These questions will be discussed again in Chapter 9 of this thesis to conclude the work presented here.

Furthermore Table 1.1 compares the current State-Of-The-Art with the achievements made in this thesis and differentiation from other work:

1. What are the limitations of current, first-generation, streaming and IPTV platforms and what kind of modifications are necessary to overcome these limitations?

2. How to provide a unified platform for the combination of content-oriented and communications-oriented services and then identify the roles of the various involved entities?

3. How can services for interactive IPTV services be realized based on the integrated *Session-Oriented Application Environment* and specified signaling principles for *IPTV*?

4. How to abstract a multi-user, multi-content platform from single user-to-network relationships?

5. How can dynamically composed interactive streaming media be generated and delivered?

Aspect	State-of-the-Art	Differentiation from State-of-the-Art
Service Silos	Current IPTV, communication as well as Social Networking systems provide their services separate from the other domains. Only Social TV systems propose an integration of these three domains.	The author specifies a core system which combines IPTV, communication services and Social Media aspects using standardized, session-oriented signaling protocols.The *IPTV Meta Session* extends the above-mentioned approach and adds capabilities for multi-user, multi-content consumption scenarios, named shared experience and shared consumption scenarios.
Social Media	Current IPTV systems do not provide capabilities for multimedia and multi-user session handling which limits these services when implementing blended services which require information concerning active user sessions.	The proposed core system introduces a new multi-protocol, session-oriented approach allowing for a standardized way to maintain user requests, as well as give access to third parties. Actual implementations are founded on session principles borrowed from the Session Initiation Protocol (SIP) developed originally for conversational services and extended during this thesis to support converged IPTV scenarios.
Interactivity	Current IPTV systems do not integrate a common approach for Interactive and Value Added Services. If interactive applications are provided they do rely on proprietary implementations.	The proposed core system proposes generic enablers for the provisioning of common interactive services as Televoting, Targeted Advertisement and Quiz-Shows. These services will be provided using a session-oriented signaling infrastructure and APIs for manipulating user sessions through these interactive services.

Table 1.1.: Targeted Architecture and Core System achievements compared to State-Of-The-Art

1.4. Methodology

The stated problems, objectives and defined scope addressed in Section 1.2 and 1.3 of this thesis will be discussed using the following methodology.

This methodology has been derived from working methods as conducted by different Standard Developments Organizations (SDOs) and is known as the *Staged Approach*[4], subdivided into requirements collection, architecture design and a detailed specification phase:

- Description and analysis of the State-of-The-Art on IPTV, providing a classification of the different approaches ranging from Over-The-Top (OTT) to managed networks.
- Analysis of approaches for *Interactive Application Environments* for IPTV as chosen and defined by the different Standard Development Organizations worldwide, or available as proprietary solutions from industry players.
- Definition of requirements for cross-domain, shared experiences in interactive IPTV.
- Design of an architecture for the *Open IPTV Ecosystem Core*, combining multiple *Interactive Application Environments* and fulfilling the requirements of the different players involved. The design hereby reflects State-of-the-Art technology and requirements analyzed earlier.
- Specification of the *Open IPTV Ecosystem Core* including *Application Programmable Interfaces*, interactive and *Social IPTV* services.
- Description of the concepts and methods used for the implementation of the most relevant building blocks.
- Validation of the proposed system throughout the evaluation of the implementations of the proposed system.
- Detailed assessment of and comparison with related works.

[4]http://portal.etsi.org/mbs/protocolStandards/stagedApproach.htm

1.5. Outline

This thesis is structured as follows. Figure 1.6 provides a graphical representation:

Chapter 2 describes the basic technologies and concepts from which the main contributions of this thesis have been created. This includes an overview of how Digital Television (DTV) evolved towards IP-based media delivery. Based on this information, the difference between so-called managed and Over-The-Top media delivery infrastructures will be presented. The role of *Content Delivery Networks* (CDNs) will then be discussed.

As current CDN approaches might serve as a reference, even for the Internet, a migration path is outlined in the next paragraph. In a further step, the various acronyms and definitions for IPTV used within the scope of this thesis will be defined. Next, the current standards and state-of-art technology and related works in research and academia on *IPTV and Interactive Application Environments* will be analyzed. This includes an analysis of and decision point for which of these technologies should become part of the designed platform. Furthermore, an initial overview of related work within the area of interactive and *Social IPTV* will be provided.

Chapter 3 analyzes and derives requirements for interactive IPTV. Different aspects such as session-oriented signaling principles, corresponding APIs, interactive service requirements, as well as content provisioning and delivery aspects, are taken under analysis.

Chapter 4 specifies a session-oriented architecture for IPTV, namely the *Open IPTV Ecosystem Core.* This includes the definition of functional entities and enablers, which will then be used in the following chapters for the the entities that form an end-to-end ecosystem. Furthermore, a combined session-oriented and declarative application environment is described.

Chapter 5 focuses on the specification of APIs and services for the *Open IPTV Ecosystem Core.* First, the so called *IPTV Session State API* will be introduced, allowing for the manipulation of IPTV services through third party application providers.

In another abstraction layer, the so-called *IPTV Meta Session* introduces concepts that fulfill the requirements for so-called *Shared Experiences* and *Shared Content Consumption* scenarios. This approach is driven by the idea of combining multiple contents, multiple users and communication services into one *Shared Experience* offered as a sort of channel to a group of potential users. Another section focuses on how contents like live streams and stored media might be provisioned, based on the introduced session-oriented approaches. This includes the specification of a session-oriented *Interactive Content Enabler*. The last section is focused on the introduction of different session-oriented interactive services that can be executed on the proposed architecture. This includes *Televoting*, *Targeted Advertisement*, and an interactive, *Virtual Quiz Show* scenario including *User Generated Content*.

Chapter 6 is dedicated to the description of the practical implementations conducted during work on this thesis.

Chapter 7 is dedicated to the verification of the described system, specifications and implementations. Based on the provided implementations, measurements and three case studies will be presented.

Chapter 8 provides a detailed assessment of and comparison with related works.

Chapter 9 summarizes the work conducted in this thesis and provides an outlook for future work.

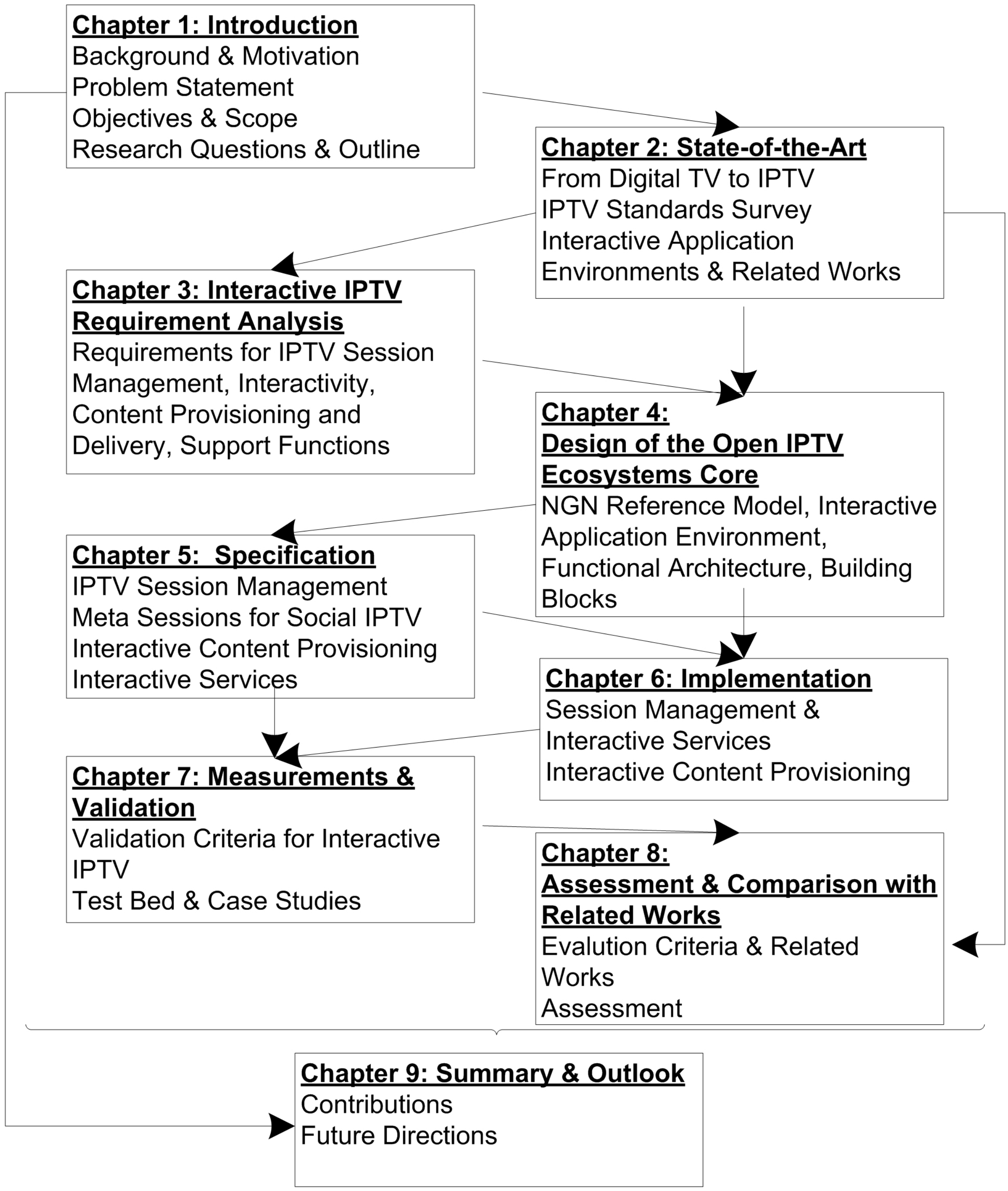

Figure 1.6.: Outline & Structure

2. State-of-the-Art

This chapter describes the common technologies and related works involved with Digital, IP-based and interactive Television.

First of all, the evolution of Digital TV (DTV) and its transition to Internet Protocol Television (IPTV) will be outlined. This is followed by definitions of the terms commonly used in the field of IPTV, and an IPTV standards survey.

Furthermore, different architectures and concepts for *Interactive Application Environments* will be analyzed by describing their overall approach, advantages and disadvantages, as well as their relevance for this thesis. Finally the different conceptual approaches will be compared.

2.1. From Digital TV to IPTV

2.1.1. Digital TV Evolution

High bit rate, high quality media and corresponding services with high bandwidth requirements are accelerating the development of a media-enabled Internet.

An intermediate step on the path to this vision is *Internet Protocol Television* (IPTV).

Just when we thought we had mastered communications- and home entertainment-related acronyms and managed to connect a Digital TV Set-Top-Box (STB) to our TV set, IPTV was introduced.

That acronym, IPTV, represents an emerging technology that could fundamentally change the manner in which we receive home entertainment, obtain communication services, use social networks, and even use our mobile phones.

At first, the acronym is nothing more than an abbreviation for television transmitted over the Internet Protocol (IP) network, but it can also represent a series of technologies that provide television and other corresponding services for different sizes of screens ranging in size from cell phone displays and personal computer monitors to large plasma and LCD televisions mounted on walls in homes [61].

2.1.2. Digital Interactive Television

An evolutionary step in the history of TV and a benchmark on the path towards IPTV was the introduction of *Digital Television* (DTV) in the early nineties, by the *Digital Video Broadcasting Project* (DVB)[1]. By introducing a framework for terrestrial (DVB-T), cable (DVB-C) and satellite (DVB-S), digital distribution channels for content and data services were made available.

The main advantage of digital formats is that it allows for multiple copies to be made without any degradation in quality [10]. In addition to the much more effective bandwidth consumption, this aspect has also been a main driver for the digitalization of TV.

As described for the European market in [108], Digital TV has been recognized as the starting point for any kind of interactivity with the TV set: In addition to just the pure broadcasting of pictures and sound, so-called data broadcasting and interaction channels were introduced.

With the specification of the *Multimedia Home Platform* (MHP) as described in [105], and [93] and specified in [36], the technological baseline for providing a standardized platform for downloadable Set-Top-Box (STB) applications was set.

Nevertheless, MHP and its variations have never been deployed on a large scale, due to missing broadband connections and follow-up versions of the standard that were incompatible with previous versions of the standard.

2.1.3. Transition from Digital TV to IPTV

With widely-available broadband connections over xDSL and cable broadband networks, the discussion concerning interactive and value-added IPTV re-started about five years ago.

Again, the *Digital Video Broadcasting Group* (DVB) took advantage of the opportunity and began initial work on a common standard for IPTV and released an initial version called DVB-IP [122, 34]. This specification relies mostly on the already well-known and approved standards for Digital Television [107].

Initially, it looked like IPTV would become nothing more than a revised version of interactive DTV, which had been mainly built around the *Multimedia Home Platform* (MHP).

MHP has some technological deficiencies, like missing broadband back channel connections. But, with the availability of broadband connections over xDSL, cable, and even mobile networks, this main drawbacks has disappeared. Today powerful end devices, video codecs that are optimized with regards to bandwidth consumption (e.g. H.264) and new service delivery platforms like the *IP Multimedia Subsystem* (IMS) have served as additional enablers, seeming to make it worth it to start a second approach towards interactive IPTV.

[1] http://www.dvb.org

In parallel to the DVB's work, which was and is mainly driven by broadcasters and content providers, Telco-driven forums like the *International Telecommunication Union* (ITU)[2] and the *ETSI* group for *Telecommunications and Internet converged Services and Protocols for Advanced Networking*[3] (TISPAN) also started to work on building IPTV standards [89].

In contrast to DVB, Telco-driven forums have, from the beginning, focused and pushed the idea of converging services and platforms based on IP technology.

With available specifications for all-IP communication services like Next Generation Networks (ITU-T) and the *IP Multimedia Subsystem* (ETSI TISPAN, Core IMS) in addition to the vision for the so-called fixed mobile broadcast convergence (FMBC) [101], new communication paradigms and ways of media consumption have been specified. The main paradigm of the before-mentioned approaches is enabling the use of services across the borders of terminals, offering both personal and terminal mobility.

Altogether this describes concepts like *Triple Play*, defining Internet, telephony services and IPTV. *Quadruple Play* adds the mobile domain to this concept.

Taking the technology one step further, the concept of multi-play characterizes a new vision for multiple services interaction by also integrating features like mobility, interactivity, community services, and personalization.

2.1.4. Definition for IP based Television (IPTV)

With the aforementioned technological backgrounds in mind, and the intention of creating a successor to the well-established standards for Digital Television, the collection of initial requirements was condensed into a definition of what IPTV is. The ITU-T's IPTV Focus Group defined IPTV as follows:

Definition 1 *IPTV is defined as multimedia services such as television/video/audio/text/graphics/data delivered over IP-based networks managed to provide the required level of quality of service (QoS)/quality of experience (QoE), security, interactivity and reliability.*[4]

This definition has also partly been used in the context of this thesis when defining architectures, signaling flows and services. Furthermore, a functional role model as depicted in 2.1 refines the ITU-T's definition by introducing four roles within the IPTV value chain:

- *Content Provider:* Video, audio (including voice), data, text, and applications
- *Service Provider:* Reception, manipulation, value-added processing, and transmission of content with security and management, according to the service provider

[2] http://itu.int/IPTV
[3] http://www.etsi.org/TISPAN
[4] Definition maintained by ITU-T's IPTV Joint Coordination Activity (IPTV-JCA) at http://www.itu.int/ITU-T/jca/iptv/

- *Network Service Provider:* Managed, controlled, and secured delivery of contents processed by platform using QoS controlled Broadband IP Network including both wired and wireless.
- *Multimedia Broadband Consumer:* TV, PDA, Cellular, Mobile TV with STB module or similar device for the customer.

The *Content Provider* role is composed of different providers like broadcast TV stations, providers of motion pictures, in addition to advertisers. The *Service Provider* is responsible for provisioning IPTV on a service level in addition to providing the network-situated middleware. The *Network Service Provider* role consists of an IP network provider that offers a quality-assured delivery channel. The *Multimedia Broadband Consumer* represents the end user and end devices.

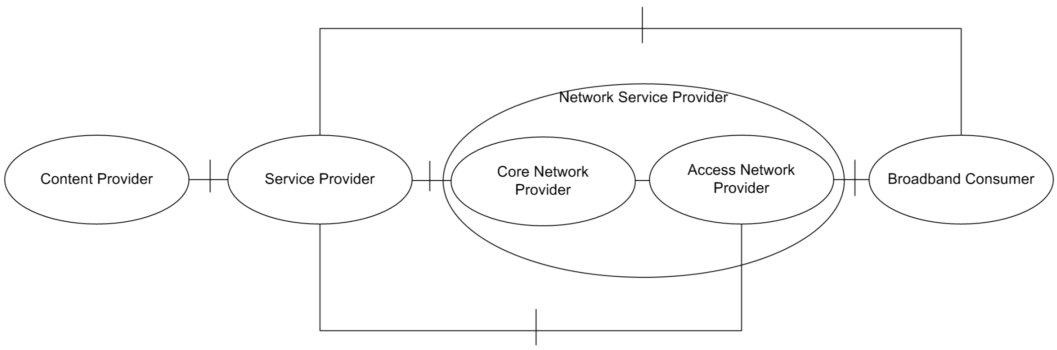

Figure 2.1.: Roles in an IPTV Ecosystem (ITU-T definition)

2.1.4.1. Variances of IP-based Television

In addition to IPTV, other expressions like Digital TV, Interactive TV, Web TV, Next Generation TV, TV 2.0, Hybrid TV are also ways of describing aspects of IP-based media delivery. These words are often used in the same context, or with the same meaning as IPTV. The following overview provides a summary of the most important definitions:

Digital TV Digital TV is a way of transmitting TV pictures and sound as computerized bits of information. Standards for Digital Television have been set by the Digital Video Broadcasting Group (DVB) for Europe, the Advanced Television Systems Committee (ATSC) for the North American market, and by the Association of Radio Industries and Businesses (ARIB) for the Japanese market.

Interactive TV Interactive TV describes an extension to the classical unidirectional model of TV with the inclusion of a return path from the user towards the broadcaster or service provider. Interactive TV systems can be realized through various hardware and middleware configurations. Standards for interactive TV were first developed by the DVB as the so called Multimedia Home Platform (MHP).

From a user's perspective, the difference between classical unidirectional Digital TV systems and interactive TV is often described as a shift in the viewing paradigm from lean-back to lean-forward TV. Interactive TV systems integrate the user into the viewing experience by allowing him to interact with the contents, e.g. in quiz shows and surveys, by offering telecommunication features or other so-called value-added services (e.g. in-channel shopping).

Web TV Web TV represents the current trend of using a broadband Internet connection for the delivery of streaming media or progressive downloads inside a Web browser. Contents are mostly short and often user-generated video clips. Web TV uses simple point-to-point connections between the user and the provider (e.g. YouTube). It does not require a managed network and the content is delivered on a best-effort basis.

Hybrid TV Whereas IPTV has been recognized as an intermediate step towards a Media Internet, Hybrid TV represents a migration from existing Digital TV broadcast networks towards IPTV. To achieve this, Hybrid TV combines Digital TV receivers and an Internet connection in one device. This technique creates an easy way to enrich plain broadcasts with Internet services. This includes portal pages (e.g. enhanced Teletext services), various kinds of personalization and Video on Demand.

Social TV Social TV combines multimedia content and multimedia communications. These two building blocks allow for the generation of interactive services that incorporate groups of users building so called immersive[5] shared experiences [14]. These terms say nothing about the technology behind them and can be mapped to various scenarios.

Mobile TV Mobile TV describes the delivery of IPTV services over mobile networks. Mobile TV services do not differ from services delivered over fixed networks, but might have other usage patterns and are limited by the capabilities of the used air interface, e.g. missing multicast capabilities.

Telco TV describes the idea of operator-driven IPTV platforms over managed networks. This includes proprietary black-box solutions, as well as IPTV systems

[5]Immersiveness describes the idea of multimodal communications, e.g. gaming, learning, training, simulating, social networking, participating, acting, doing, influencing and interacting. Presence and interaction are predicted to be essential characteristics in these environments.

currently under development and specified by various standardization bodies worldwide.

2.1.5. IPTV Characteristics

In general, IPTV has to be differentiated from Web TV or Internet TV (e.g. YouTube), because the latter works as a so-called Over-The-Top (OTT) service in open access networks (Internet) and offers no control over delivery, making it only best effort. IPTV, on the other hand, requires an infrastructure to offer at least a certain level of Quality of Experience (QoE) and uses QoS, or simply over-provisioning of resources, inside a controlled network.

The following characteristics try to define IPTV more precisely because, in order to succeed, IPTV solutions need to support a couple of features in order to constitute more than just the delivery of television, Video on Demand or audio content over a broadband network:

1. Interactivity made possible with the availability of an upstream channel, enabling the delivery of new kinds of services to the end user.

2. Personalization and Advertisement offer new kinds of business models to content providers and enhance the user's entertainment experience.

3. Portability/Mobility is important, because the increasing number of mobile devices in use needs to be considered.

4. Quality of Experience assurances and media transport over managed networks are necessary to provide the same or better quality as that achieved in DTV.

5. Converged Services combine features from different components (e.g. Telco and TV services) and enhance the consumer's rich media experience.

In [97] O'Driscoll defines additional differences between WebTV and IPTV, which are also affected by the fact that the Internet at its current stage is still not capable of transporting high quality video content:

- Different Platforms: In contrast to WebTV, IPTV uses secure, dedicated private networks to deliver video content to consumers. These private networks are managed and operated by the provider of the IPTV service.

- Access Mechanisms: A set-top box is generally used to access and decode well-defined services and the video content delivered via an IPTV system, whereas a PC is nearly always used to access the Internet.

- Network Ownership: IPTV is delivered over a networking infrastructure, which is typically owned by the service provider. Owning the networking infrastructure allows telecom operators to tailor engineer their systems in order to support the end-to-end delivery of high quality video.

2.1.5.1. Over-The-Top vs. Managed Networks

Experts' discussions concerning how future IPTV and streaming services should be delivered to the consumer has divided the involved parties into two categories:
The first, including Consumer Electronics and Broadcasters, currently prefers a model that uses the *Internet* as the delivery channel for streaming content, without the need for any operator-specific network. Telecom operators prefer a solution in which the network operator acts as the Service Provider and offers services with a certain amount of quality. The first approach is called the *Managed Network* approach, also referred to as a *Triple Play Walled Garden*, the other is called *Over-The-Top* (OTT) or *Portal Side Walled Garden* (see Figure 2.2)[6]. The next section outlines how content delivery takes place in managed environments.

	Managed Network	Over-The-Top (OTT)
Applicable IPTV Systems	Telco TV, Hybrid TV	OTT, Hybrid TV, Web TV
Delivery Network	NGN	Internet
Service Control	IMS, proprietary	none
Session Control	SIP, proprietary	none
Transport Protocols	RTP/RTSP, HTTP	HTTP
Quality-of-Service	yes	none
IP unicast	yes	yes
IP multicast	yes	no

Table 2.1.: Characteristics of Managed Networks & OTT deployments

2.1.5.2. IPTV Content Delivery Networks (CDN)

As currently constructed, the Internet supports all kind of IP traffic on a best-effort basis without noting any difference in what is sent over the channel. This implies that no *Quality-of-Service* (QoS) nor *Quality-of-Experience* (QoE) could be guaranteed. That is also the main reason why especially high quality video services cannot be delivered over the Internet with an assured level of quality.

In contrast to that, the operator's edge networks already fulfill these requirements by incorporating aspects and technologies as virtual networks using managed routers and caches for media content, forming so called *Content Delivery Networks* (CDN). For IPTV to succeed, carriers and service providers must offer a QoE equal or better to that offered by today's cable or satellite TV [143]. This requirement cannot be fulfilled without appropriate delivery networks that incorporate, at least, paths from

[6] From the author's point of view, these are more or less political discussions simply driven by the question of *"who owns the customer"* and not necessarily founded on a technical basis.

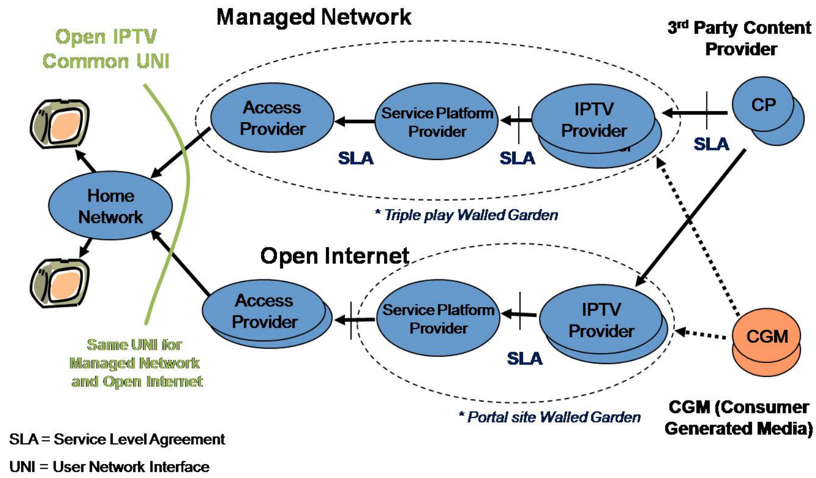

Figure 2.2.: Managed Network vs. OTT deployment options [99].

the TV/VoD head-end to the last mile or edge network and the user home network. At present, the last mile is realized either with a fiber connection, xDSL technology or over a cable network using DOCSIS standards. The backhaul[7] network is realized using either ATM over SONET/SDH, IP over MPLS or metro ethernet. A high bandwidth connection alone is not enough to guarantee that the content is delivered without any kind of degradation, as even a 50 MBit/s VDSL connection is still a shared medium in which any kind of service might interfere with another. For this reason, either virtual circuits in ATM, virtual paths in MPLS backbone networks and in the last mile, Ethernet Virtual Connections (EVN) are used. A VLAN is defined as a mechanism for the creation of a series of independent logical networks.

Under this approach and as depicted in Figure 2.3, a number of different VLANs can coexist together within the same network infrastructure, including VLANs for IPTV, Voice-Over-IP telephony and Web usage. A network management software that interfaces directly with the various network elements is used for the creation of VLANs and to support an IPTV implementation.

The main benefit of using VLAN is that different kind of services like voice, plain Internet IPTV broadcast and on-demand services can be deployed in parallel, using the same network infrastructure.

[7]In a hierarchical telecommunications network, the backhaul portion of the network comprises the intermediate links between the core, or backbone, of the network and the small subnetworks at the "edge" of the entire hierarchical network.

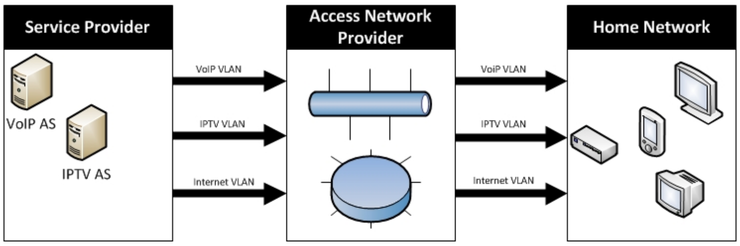

Figure 2.3.: Content Delivery Networks with VLANs for different service classes

2.2. IPTV Standards Survey

This IPTV standards survey collects the current status of the work and output of the five most active *Standard Development Organizations* (SDOs) and one de-facto industry standard in this field, from 2005 to present:

- DVB
- ETSI TISPAN
- ITU-T
- Open IPTV Forum (OIPF)
- HbbTV
- Microsoft Mediaroom

The author has hereby actively contributed to work in ETSI TISPAN from 2006 to 2008 and then in the Open IPTV Forum (OIPF), also becoming active in the forum's leadership as the deployment project manager and ambassador.

An earlier version of this survey has already been published in 2008 [69].

2.2.1. DVB

The Digital Video Broadcasting (DVB) standards have been designed to broadcast digital TV services and have been maintained by the DVB project since 1993. DVB systems transmit data using a variety of approaches: over satellite (DVB-S), cable (DVB-C), terrestrial (DVB-T) and hand held (DVB-H).

As DVB-T is mainly targeted at stationary receivers and is not suitable for mobile devices, the DVB-H standard was proposed, which enhanced the physical and link layers of DVB-T to reduce power consumption and improve performance in urban indoor environments.

While DVB specifications were initially designed to support only one way broadcast transmission, there are several standardized solutions such as DVB-RCS (Return Channel Satellite), DVB-RCT (Return Channel Terrestrial) or DVB-RCC (Return Channel Cable) or DOCSIS (Data Over Cable Service Interface Specification) that have been developed to facilitate bidirectional communication channels, thus supporting interactive applications (e.g. VoIP or VoD).

Specifications The efforts of DVB-IPTV, representing a collective name for a set of technical specifications that facilitate the delivery of digital TV, started in 2005. Using the Internet Protocol over bi-directional fixed broadband networks, DVB-IP is the DVB Project's answer to current activities in the sphere of IPTV. In this context, the DVB's interactive middleware specifications, namely DVB-MHP (Multimedia Home Platform) and GEM (Globally Executable MHP), which also include IPTV profiles have been adjusted to also support IP-based environments.

Beside others, the DVB has released a set of specifications[8] and so-called blue books for DVB-IPTV:

- An Architectural Framework for the Delivery of DVB-Services over IP-based Networks
- Transport of MPEG-2 TS Based DVB Services over IP Based Networks (and associated XML)
- Guidelines for the implementation of DVB-IP (Part 1-4)
- DVB-IPTV Profiles

Altogether, these specifications allow for an end-to-end implementation of a DVB-IP compliant IPTV system.

Research & Market Relevance The DVB's IPTV specifications offer a profound framework for the development of streaming services over an IP network. They mainly concentrate on transport and metadata issues, and do not integrate aspects relevant for Telco-operated managed IPTV networks using IMS and NGN and browser-oriented declarative interactive application environments.

The latest discussions within the DVB's Commercial Module IPTV (CM-IPTV), responsible for defining the commercial requirements for DVB-IP, try to evaluate if these gaps must be closed by adapting the work of the Open IPTV Forum or HbbTV.

[8] http://dvb.org/technology/standards/index.xml

2.2.2. ETSI TISPAN

TISPAN[9] (Telecoms & Internet converged Services & Protocols for Advanced Networks) is part of ETSI and responsible for integrating IPTV into the NGN.

Members range from telecom equipment manufacturers to network operators and compose over 700 companies and organizations globally. The customer network, service provider network and media content distribution are the critical elements of the IPTV ecosystem. ETSI works on standardizing use cases, functions and interfaces that allow interoperability and interworking.

As part of the TISPAN NGN Release 2, released in early 2008, the integration of IPTV services in a NGN architecture was specified and therefore enabled IPTV functions to use capabilities provided by the NGN subsystem.

In detail, this included traditional services like broadcasting TV, Content on Demand and Network-PVR. An overview is depicted in Figure 2.4.

There are two solutions for the integration of IPTV in the NGN architecture in order to serve the needs of both network service providers and equipment vendors. The *Dedicated IPTV Subsystem* focuses on the integration of existing market solutions in an NGN environment. However, it is a non-IMS-based approach and results in no guarantees for end-to-end QoS and might lead to performance and scalability problems.

On the other hand, the IMS-based IPTV solution allows blending of TV services with other telecommunication services (e.g. voice, presence, and data services) by reusing the same IP-based NGN components. Moreover, it offers full network control for Telcos. This might well be a disadvantage, because this walled garden approach might scare third parties off. A short overview on the role of NGNs and the IMS in managed IPTV can be found in Annex A.1.

Specifications ETSI TISPAN has released a set of specifications where:

- the *NGN Release 1* adopts the 3GPP IMS standard for SIP-based applications, and adds further functional blocks and subsystems to enable fixed access to IMS and to handle non-SIP applications, where the
- *NGN Release 2* provides a TISPAN and 3GPP agreed approved Common IMS platform, introducing new IMS enabled services and adds key elements to the NGN such as:
  - IPTV (both IMS and non-IMS based)
  - Home Networking
  - Corporate networks and the NGN
- the still-active NGN Release 3 will improve several aspects introduced in the previous releases such as:

[9] http://www.etsi.org/tispan

- IPTV service evolution (combining IMS and non-IMS-based approaches)
- Content Delivery Networks (CDN)
- Peer-to-Peer aspects for IPTV

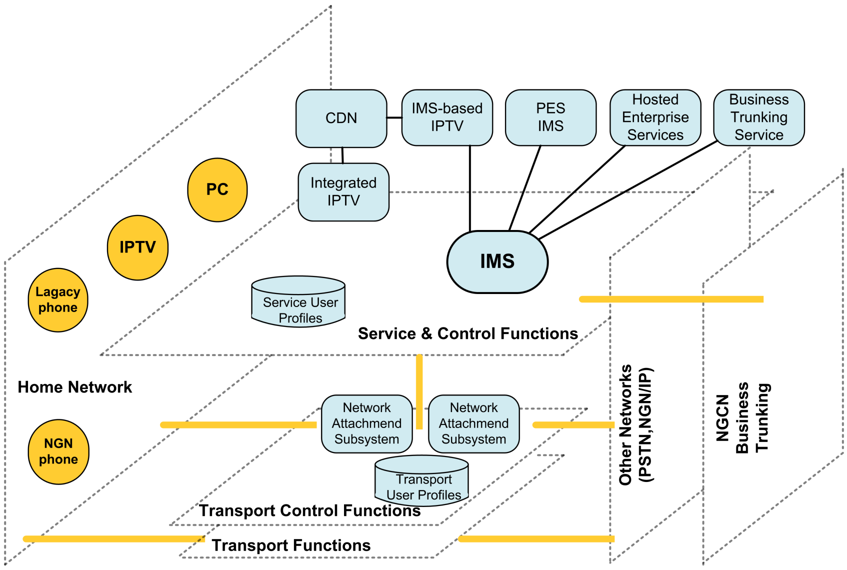

Figure 2.4.: IPTV in the ETSI TISPAN NGN Release 2 [33]

Research & Market Relevance ETSI TISPAN's IPTV specifications have reached a certain level of maturity and are recognized as implementable in an NGN environment.

In 2007, they were adopted by the Open IPTV Forum as part of their Release 1 specifications.

The lack of specifications for the end device with regards to an *Interactive Application Environment* have stopped the specifications from being adopted by the market. An alignment or merger with the Open IPTV Forum makes the most sense at this moment, allowing for a unified specification for managed NGN and IMS-based IPTV systems.

2.2.3. ITU-T

The International Telecommunication Union - Telecommunication Standardization (ITU-T) is a body for international Telecommunication standards.

Regarding IPTV, ITU takes the same approach as ETSI TISPAN. The NGN and IMS plays a key role when integrating telecommunication services.

Specifications A wide range of specifications for IPTV have been published by the ITU-T[10]. These specifications range from architectural specifications to signaling, home networking and metadata.

Furthermore, three different deployment models, including a non-NGN IPTV, a NGN without IMS IPTV and a NGN with IMS IPTV have been specified, allowing for a progressive migration.

Furthermore, a standard suite for a declarative Interactive Application Environment has been specified.

Research & Market Relevance The work on IPTV specification took place in parallel to ETSI TISPAN's standardization process, mostly during 2006 to 2008. Unfortunately, an alignment between both groups never took place, resulting in two specifications for IPTV.

In the end of 2008, the so-called ITU-T Focus Group on IPTV[11] (FG-IPTV) handed over its work to ITU-T's IPTV Global Specification Initiative (IPTV-GSI), which resulted in a subdivision of activities and loss of alignment.

Today, the ITU-T's IPTV specification do not provide a comprehensive framework allowing for the implementation of a common IPTV system.

2.2.4. Open IPTV Forum

A consortium of network operators, device vendors, public network infrastructure vendors, service content providers and technology providers compose the members of the Open IPTV Forum, which was founded in March 2007 with the goal of orchestrating the IPTV standardization landscape and working towards an interoperable end-to-end IPTV solution.

This solution is IMS, CE-HTML and Digital Living Network Alliance (DLNA) based and backed up by a full end-to-end specification set that reuses existing technologies and standards from other SDOs.

Furthermore, the Forum works on interoperability and certification to focus on the end user's perspective.

Specifications At the moment, the work is done on OIPF specification Release 2, while the architecture and solution specs are prepared for Release 3 and have been published in late 2010. Release 1, in general, features services access on TVs/STBs via fixed networks, broadcast in standard definition (SD) and high definition (HD) quality, on-demand services and recording services (local and network PVR). It is also concerned with the user interface/portal, user management and services interworking (IM, presence). Release 2 features services access on various end devices

[10] http://www.itu.int/ITU-T/IPTV/
[11] http://www.itu.int/publ/T-PROC-IPTVFG-2008/en

via fixed and mobile access networks, remote PVR control, and more improvements regarding service portability and continuity.

The Open IPTV Forum's architectural design, as depicted in Fig. 2.2, supports two service models, namely the Managed Network and Open Internet (OTT).

Although the models are essentially different, a single UNI (User Network Interface) is supposed to ensure services from both networks to the consumer's electronic devices. This approach has to be differentiated from the ETSI TISPAN's approach of the *Dedicated IPTV Subsystem* and *IMS based IPTV*, because the latter requires an underlying NGN architecture for both approaches. The Managed Network is an end-to-end network managed by an operator. In a typical model, the operator, for instance a Telecom Operator, plays the IPTV Service Provider, Service Platform Provider and Network Provider roles.

As a result, this walled garden approach can guarantee high quality services to the end user. The Open Internet/Unmanaged Model has the same technical roles as the managed model, but the roles are typically played by different players. Nevertheless, the Service Platform Provider and IPTV Provider are the same entity. Open Internet also refers to the ability to access any Service Provider using any Access Network Provider without any quality of service guarantees. This boosts the capability to keep up with the many technical innovations constantly occurring in the World Wide Web.

Similar to the description of ETSI TISPAN's architecture earlier, this description will also be done for OIPF's high-level architecture.

OIPF Release 1 Specifications The OIPF Release 1/2 specification are composed of seven volumes describing an end-to-end infrastructure for IPTV. The following specifications are available in detail[12]:

- **Volume 1** – Overview: description of the overall approach of the OIPF's specifications
- **Volume 2** – Media Formats: references to MPEG media formats
- **Volume 3** – Content Meta Data: references to ETSI and TVAnytime metadata specifications
- **Volume 4** – Protocols: specifications for the usage of protocols for managed, hybrid and OTT deployments
- **Volume 5** – Declarative Application Environment: definition of a browser-based runtime environment for interactive applications
- **Volume 6** – Procedural Application Environment: references for DVB's Multimedia Home Platforms (MHP) specifications.

[12] http://www.oipf.tv/specifications.html

- **Volume 7** – Authentication, Content Protection and Service Protection: specifications for the usage of AAA and content protection mechanisms.

End Devices Infrastructure The Open IPTV Forum's specifications contain detailed specifications for the home network infrastructure. Key element is the decoupling on protocol level of the termination end-point for managed networks and the TV/STB.

In detail, the so called Open IPTV Terminal Functional Entity (OITF) includes functionalities for accessing IPTV services for both the unmanaged and the managed network models through the HNI-INI and HNI-IGI interfaces. Through the HNI-IGI interface, it connects with the IMS Gateway Functional Entity (IG), which allows the OITF to connect to the IMS core networks. The OITF has its own direct user interaction (e.g. remote control, keyboard) and audio/video rendering and integrates browser technology for the generation of dynamic user interfaces.

The Application Gateway Functional Entity (AG) is an optional component and offers applications for remote downloading for execution. Examples of these applications are EPG, generation of a remote UI, insertion of personalized advertisements in media streams and full, blended person-to-person communication (videoconferencing using a TV).

The Content and Service Protection (CSP) Gateway Functional Entity (CSPG) is also optional and provides a conversion from a content and service protection solution in the network to a secure authenticated channel with the OITF.

The WAN Gateway function (WG) supports the physical connection between the residential LAN and the Access Network WAN.

Research & Market Relevance As with all IPTV standards and specifications, the OIPF specifications have also suffered from not being adopted by the industry. For this reason, and in contradiction to other IPTV standardization bodies, the OIPF is heavily driving different go-to-market activities, like proof-of-concept projects, live-interoperability-trials and certifications programs. In this context, the OIPF took part in its first proof-of-concept project in 2010, demonstrating the applicability of its specifications.

2.2.5. HbbTV

Hybrid Broadband Broadcast TV (HbbTV) is an industry-led forum of broadcasters and CE manufacturers. In contrast to the IPTV standards bodies, the HbbTV forum concentrates and limits its activities to a hybrid delivery channel using classical DTV transmission and an Over-The-Top approach to IP-based services. The HbbTV ecosystem has been as defined as:

IPTV Standard	OIPF	HbbTV
Typical service provider	IPTV service provider	Broadcaster
Typical TV content	Premium, VoD, other payTV	Free to air, catch-up TV
Typical User / Service provider relationship	Close relationship with one service provider	More distant relationship with multiple service providers
Geographical Focus	Global	European
IP connection	Always required	Useful but not essential
Application Environment	CE-HTML/HTML5, JavaScript, JavaScript APIs	CE-HTML, JavaScript, JavaScript APIs

Table 2.2.: Comparison of OIPF and HbbTV specifications

"a content distribution platform for signalling, transport and presentation of enhanced and interactive television services and related applications designed for using both a broadcast and internet networks and is running on hybrid terminals that include both a broadcast and internet connection."[128]

The HbbTV specifications explicitly do not touch managed networks as promoted by the IPTV standardization bodies and implies that HbbTV is not applicable to a telecom operators' infrastructure. Furthermore, this specification drives a broadcaster-controlled environment using an ISPs network without any service guarantee.

Specifications The HbbTV specifications have mostly been derived from the Open IPTV Forum's specifications and were published as an ETSI standard [37] in mid 2010.

Both share the same *Interactive Application Environment* based on CE-HTML. Furthermore, the OIPF's API specifications for a browser-based TV environment have been extended by HbbTV to allow for an interaction with the broadcast channel.

Research & Market Relevance The advantage of the HbbTV's specifications is their immediate applicability to currently deployed IP-network infrastructures (xDSL, cable) and DTV environments. In this context, they have a good chance to become the standard for hybrid environments in Europe and herewith also influence upcoming IPTV deployments.

2.2.6. De-facto Standard: Microsoft Mediaroom

The Microsoft Mediaroom platform is a complete end-to-end solution for IPTV in managed Telco and Cableco environments. At the point this thesis was written, Mediaroom has, without a doubt, the largest number of IPTV deployments worldwide, with a growing subscriber base of around 4 million users in various countries [85]. As Mediaroom is a complex system, just a brief introduction can be given here. The information provided here has mainly been taken from the so-called "Microsoft Mediaroom Architecture Guide for ADK Developers" [22] and will be limited to presenting the core system elements as well as technologies available for the creation of interactive IPTV applications.

Mediaroom Subsystems The Mediaroom platform is organized using a so-called subsystem approach, by which some of them perform acquisition and delivery directly, whereas others perform a supporting role in turn forming the whole ecosystem. In addition, a set of Application Programming Interfaces (APIs) are available, allowing service and content providers to access the platform directly (e.g. content ingest, OSS/BSS integration). In more detail, the following selected subsystems are available inside a Mediaroom infrastructure:

- **Live TV acquisition Subsystem** acquiring live broadcast services and encoding them into the required output formats (H.264 or VC-1) for full screen or Picture-In-Picture (PIP) delivery and encapsulation into RTP transport streams over IP unicast or multi cast.
- **Live TV Delivery Subsystem** managing the delivery of live TV channels in unicast mode to support the so-called Fast-Channel-Switching using a dedicated Distribution Server infrastructure,
- **Live Anytime Subsystem** recording live TV services and converting them into VoD assets (Personal Video Recorder Functionality)
- **VoD Acquisition / Delivery Subsystems** allowing for the creation of VoD assets and metadata which are then deployed through the Delivery Subsystem towards the users STBs
- **Application Subsystem** subdivided into three domains for so-called RDP applications, browser-based applications and applications developed using the so-called Media Presentation Foundation presented later in this section.
- **Service Information Subsystem**, **Bootstrap Web Service** and **Sync Discovery Service Windows Service** where the first provides clients with information about how to acquire video services. The second one authenticates Mediaroom STBs towards the back-end, and provides them with information

on how the STB can obtain configuration data. The latter provides connected STBs with information about how to recover from an error state.

- **Subscriber Management System (SMS)** The SMS hosts the central user database for the Mediaroom ecosystem and is used to check a user's authorization for specific services.
- **The Notification Subsystem** enables Mediaroom services to send notifications to a user's STB.

A short overview of the *Interactive Application Environments* provided with the Mediaroom platform will be provided later in this chapter, when comparing the different available *Interactive Application Environments*.

Research & Market Relevance The Mediaroom platform is on its way to becoming a de-facto standard for IPTV in certain areas of the world. Nevertheless, proprietary software and protocols are used throughout the solution. With respect to this thesis, scenarios as well as the Web-technology driven approaches for application deployment are relevant and have directly influenced the author's work in this context. This is reflected in the availability of a Declarative Application Environment (DAE) for the execution of Web applications in the proposed architecture.

2.2.7. Comparison of IPTV Standards

In the last sections, the five most relevant standards and specifications for IPTV and corresponding hybrid environments have been presented. In Table 2.3 these standards and their applicability for managed and OTT environments are compared. Furthermore, the integrated *Interaction Application Environment* is described.

IPTV Standard	DVB	ETSI TISPAN	ITU-T	Mediaroom	OIPF	HbbTV
Supporting Managed Networks	yes, but no NGN/IMS	both: NGN+IMS	both: NGN+IMS	yes, but propri-etary	yes, both IMS + NGN	n/a
Supporting OTT	no	no	no	not yet	yes	yes
Application Environ-ment	yes	session-oriented	no	proprietary	declarative + proce-dural	declarative

Table 2.3.: Comparison of IPTV standard

2.3. IPTV Architectures & Interactive Application Environments

Following the methodology introduced in Chapter 1, this section introduces the State-of-the-Art architectures and technologies that enable interactivity in IPTV environments. A classification of *Interactive Application Environments* and corresponding architectures will then be provided. In between these sections, related technologies and/or related work will be mapped to each class of *Interactive Application Environments*.

2.3.1. Classification of Interactive Application Environments

This section provides a classification and categorization of the different types of *Interactive Application Environments* used to create interactive IPTV services, and parts of which are used in combination to compose the *Open IPTV Ecosystem Core* presented in this thesis.

An application environment consists of frameworks, libraries, and services, along with the associated APIs necessary for the runtime execution of programs developed with those APIs. The application environments are dependent on all underlying layers of system software [71], examples of which include the existing *standard* Web application environment consisting of HTML, JavaScript and an ad-hoc collection of standard graphics file formats, processed by an HTML browser [28] or the Java Application Environment (JAE). When generating interactive IPTV applications, three different *Interactive Applications Environments*, as well as one conceptional approach can be distinguished, as depicted in Figure 2.5:

- *Session-oriented IPTV architectures & Application Environments (SAE):*

SAEs rely on a model in which the STB or TV is connected to Application Servers that contain application logic for interactive applications. The communication between STB/TV and the Application Server uses a session-oriented protocol like the *Session Initiation Protocol* (SIP) [114]. In contrast to Procedural (PAEs) or Declarative Applications Environments (DAEs), in which the application logic is either downloaded and executed on the STB or on part of the Web server and rendered by the Web browser, in *Session-oriented Application Environments* both the Application Server and the client contain a certain amount of application logic. The Application Server communicates with the client through the SIP protocol and then triggers the corresponding application logic. In the same manner, the client reacts to actions taken by the user, processes these inside its own application logic and forwards them, if necessary, to the corresponding Application Server.

- *Procedural Application Environments (PAE)*: PAEs use a conventional imperative programming style, in which applications are decomposed into computation steps, providing an algorithmic specification. In the case of Interactive Television, these applications are provided by a broadcaster or service provider, and downloaded and executed locally on a TV or STB. Most PAEs rely on Java technology and therefore implement a Java Virtual Machine on the end device.
- *Declarative Application Environments (DAE):* DAEs provide applications hosted on a Web server. Declarative applications emphasize the declarative description of a problem, rather than its decomposition into an algorithmic implementation [121]. Such declarative descriptions are closer to a high-level specification, and, thus, easier to design than the procedural ones, which usually require a programming expert. However, declarative languages are not all-purpose and usually have a specific focus in their design principles. With Web technologies like HTML, CSS and various scripting languages becoming more popular and Web browsers incorporated as part of every new TV set, the DAE approach is gaining more and more attention.
- In contradiction to the above-mentioned *Interactive Application Environments, Social IPTV* systems do not follow the classical definition mentioned above. They also comprise or provide a framework and services, but do not necessarily rely on a distinct software paradigm and can be composed using either procedural, session-oriented or declarative approaches or even a combination. The main goal of *Social IPTV* systems is the integration of communication services, e.g. enabling services like a television presence through a buddy list or ambient display, shared viewing experiences through program suggestions and recommendations, free-form communication through text or video chat and voice group calls, and the ability to gauge the viewing habits of other users.

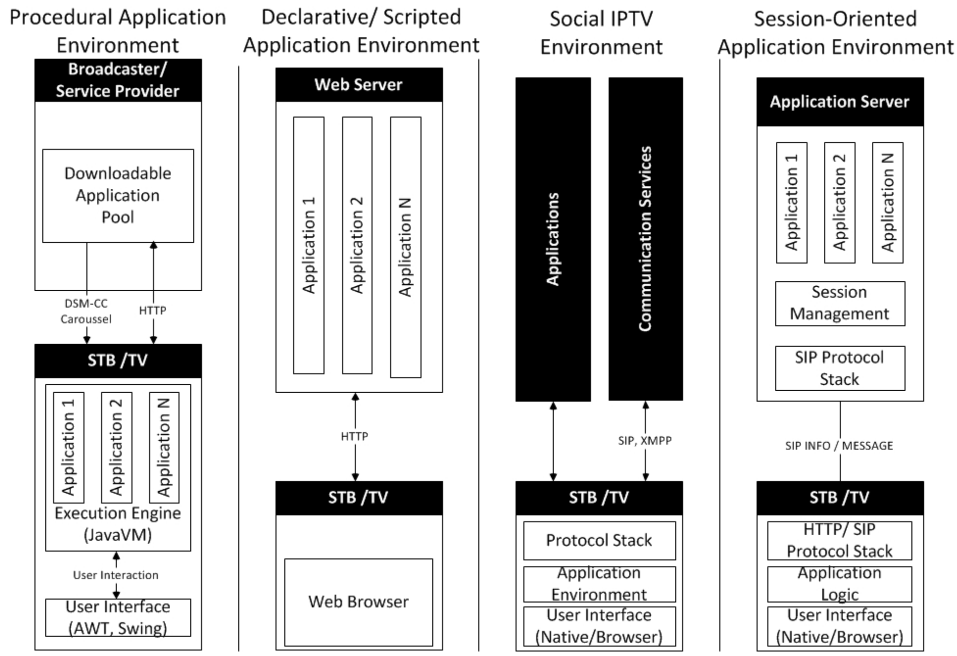

Figure 2.5.: Classification of Application Environments for Interactive IPTV

2.3.2. Session-Oriented Application Environments - Enabling interactive IPTV on signaling level

This section describes current research concerning *Session-Oriented Application Environments*, residing on session-oriented signaling principles for IPTV.

First, the protocols, general mechanisms and session models for rich communications will be introduced. In combination with this introduction, a description of typical session-oriented services like Voice-over-IP telephony and Instant Messaging will be discussed.

Second, how a session-oriented approach could be used to signal streaming media sessions will be shown. In this context, a corresponding role model, namely the *session-oriented IPTV Role Model*, describing all the entities involved will be introduced. Thirdly, related work on session-oriented IPTV systems will be described.

2.3.2.1. Relevance for this Thesis

Session-Oriented Application Environments, as described in this section, use a simple but effective approach, namely a SIP session, to allow for the easy control of common streaming services, combined with basic interactive functionality. For this reason, the SAE approach has been adopted by various SDOs that specify end-to-end infrastructures for IPTV.

During the period of working on this thesis, the SAE approach has already become relevant in the very early stages when shaping the overall architecture. Based on the existing work on the Next Generation Network and the IP Multimedia Subsystem, the author and his team have created an initial approach to a session-oriented middleware and ecosystem for IPTV. This has been achieved through a combination of the SAE approach with a DAE approach and scenarios for Social IPTV.

2.3.2.2. Background

Models and protocols for sessions and related services have a long history in the Internet and telecommunication domain. Taking the OSI model as a reference, sessions can be implemented that begin upwards from layer five in the Session, Presentation or Application layer and are described as a semi-permanent interactive information interchange, also known as a dialogue, conversation or meeting between two or more communicating devices, or between a computer and user.

From a technological point of view, session models can be used in several contexts and different technology domains, and help keep a persistent and identifiable connection between two or more endpoints. A Session-ID is used in most implementations in order to allow for a matching state between the sender and the endpoint. The most common session types and protocols are:

- HTTP sessions (originally stateless but made stateful throughout session-IDs, server side cookies and database),
- TCP sessions (transport layer), RTSP sessions for the control of media delivery and a full protocol relying on the session idea:
- The Session Initiation Protocol (SIP) initially described in [58, 114].

2.3.2.3. Session models for Rich Telecommunications

As highlighted earlier, sessions can be used in nearly all situations where information has to be interchanged and a certain level of history should be tracked during the session run time, or even after it has ended. In the context of this thesis, the author wants to focus on sessions used to deliver rich media contents and rich media telecommunications to the end user.

VoIP and Video Telephony In the telecommunication domain, the session model fits perfectly to the idea of stateful connections between users that represent a voice, video or even a combined multimedia conferencing session. With Voice over IP (VoIP) telephony becoming more and more popular, the basic SIP session model is used for any call initiated between two or more parties. Proprietary solutions like Skype or Microsoft Messenger also use basically the same concepts. Figure 2.6 depicts a simple Voice-Over-IP session setup dialogue between two users, two proxies in the middle and the resulting media session, i.e. a voice call.

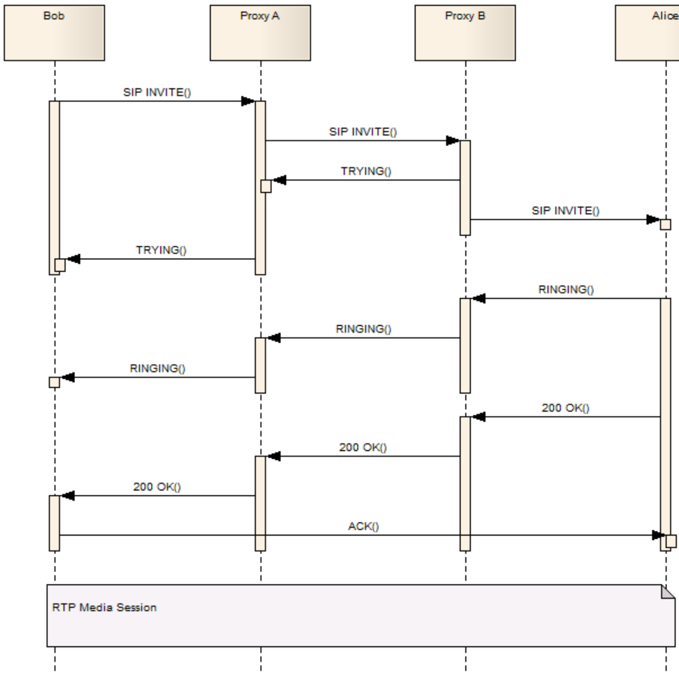

Figure 2.6.: Voice Over IP Session

Instant Messaging Instant Messaging has become a popular service throughout the Internet. SIP messaging was originally defined in RFC 3428 [17], but has not always been a session-oriented service. In any case, some kind of virtual session is always used when two users have a conversation.

However, a series of related instant messages between two or more parties can be viewed as part of a *messaging session*: a conversational exchange of messages with a definite beginning and end. This is in contrast to individual messages sent independently. Specifications for session-oriented instant messaging have been added to the corresponding RFCs. More specifications for event mechanisms and protocols can be found in [113, 16]. The Jabber or XMPP is another session-related instant messaging protocol that will be used in the context of this thesis to create a multi-protocol environment. This protocol is also defined by the Internet Engineering Task Force (IETF) as Request for Comments (RFCs) 3920 - 3923 [116, 117, 118, 115]. XMPP protocol is currently gaining more and more acceptance due to its simplicity and implementation in some semi-commercial systems and might soon become available in different versions from the consumer electronics industry.

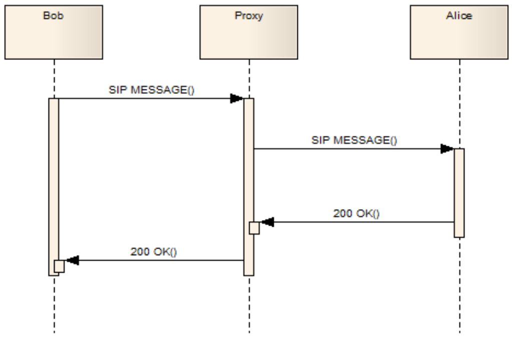

Figure 2.7.: Sessionless Instant Messaging

2.3.2.4. SIP for IPTV & Rich Media Signaling

Initial ideas for the creation of session-oriented IPTV environments (i.e. Linear TV and Video on Demand services) appeared for the first time during the IPTV standardization process that started in 2006 in ETSI TISPAN and ITU-T's Focus Group on IPTV. These ideas were described in detail in [111, 8] and [123] and practically implemented in [110].

Today, session models for IPTV have also been adapted by the Open IPTV Forum (OIPF) [102]. The concept behind an IPTV session is simple and relies on the idea that each session represents an active user inside the system. The session can easily be used to keep track of user behavior, allows for instant session modifications through the Service Provider, and enables other services and functions to use this information. As shown in [42] and later in this thesis, these kinds of architectures can be used to build new sophisticated, converged and personalized IPTV services.

Examples of these architectures include the integration of services like consumer oriented targeted advertisements, interactive gaming, quiz shows and voting scenarios.

While IPTV is not necessarily considered a session-related service, requesting high quality video streams can be described as setting up a persistent client-server connection for the duration of content consumption. Compared to the principles used for setting up a voice call in Figure 2.6 the only difference is that the recipient of the session setup request is, in that case, not another user but a so-called *Application Server* or specifically an *Session Manager*, residing inside the network infrastructure. Figure 2.8 depicts a simplified session setup process:

- The *User* requests content through a SIP request directed at the network infrastructure, including the so-called CRID[13] as the content identifier.

[13] A CRID or a Content Reference Identifier is a concept from the standardization work done by the TV-Anytime forum. It mirrors, or closely matches, the concept of the Uniform Resource Locator, or URL, as used on the World-Wide Web

- The *IPTV Session Management Enabler* inside the infrastructure resolves the user's request, creating a (dynamic) URL.
- The URL is sent back to the user embedded in a 200 OK response message.
- The *User* initiates a RTSP session setup directed at the network infrastructure, including the obtained content URL.
- After the establishment of the RTSP session, the content is streamed to the *User*.

These simple procedures are necessary for the maintenance of an IPTV session. Nevertheless, some additional actors are involved in this process and will be described in the following.

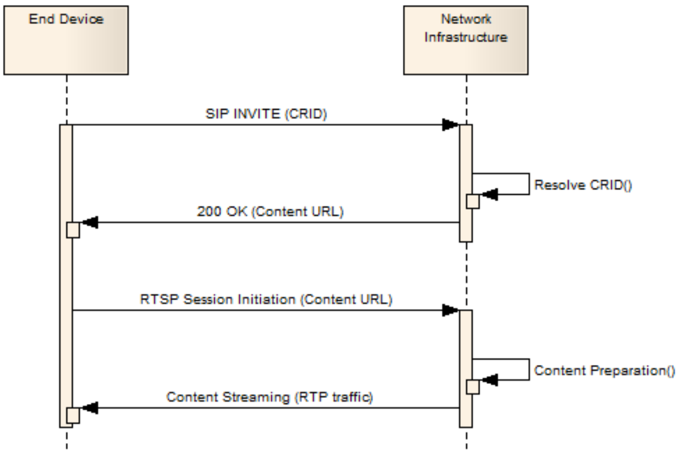

Figure 2.8.: Basic IPTV Session

2.3.2.5. IPTV Session Context

An*IPTV Session Context* is comprised of the different actors involved in an *IPTV session* as described below and depicted in Figure 2.9. All-in-all, the context can be described as the stateful information used by the different actors involved in order to set-up, modify or tear down the dialogue between them. Actors involved in this process are the *End Device* that consumes a service, the *Service Provider* that maintains the session and acts as a proxy for potential *Content Providers*. Additionally, a unique identifier for each content item, namely the *Content Resource Identifier* (CRID), is used to resolve the user's request. In addition to the maintenance of the ongoing session, the *Session Manager* also provides an API for other Support Functions (i.e. EPG, User Profile) and third party applications that make use of the users' session information.

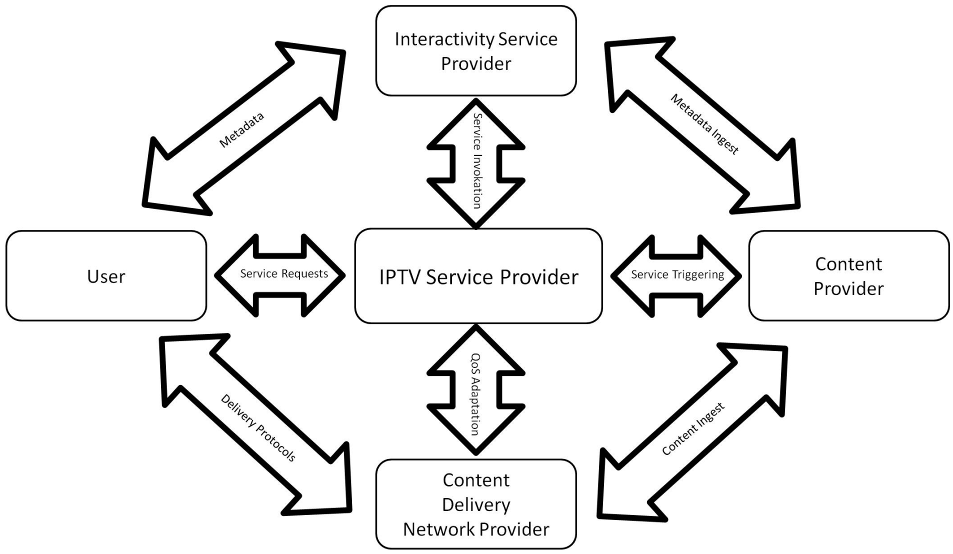

Figure 2.9.: Session-Oriented IPTV Role Model: Roles within an IPTV Session

End Device The *End Device* is the location in which services are consumed. An *End-Device* may be a single terminal used directly for service consumption, or a complex network of terminals and related devices, including consumer-operated mobile devices. Additionally *End-Devices* may be connected via two or more networks to a number of service providers that obtain content from multiple content providers.

IPTV Service Provider The *IPTV Service Provider* provides a service to the *End-Device* and the corresponding users behind it. The service provider acts as the gatekeeper between the *End-Device* and potential *Content Providers*. In this context, the *Service Provider* acquires or licenses content from one or more *Content Providers* and makes these services available. In this model, the *Service Provider* also maintains the *Session Manager*.

Interactivity Service Provider *Interactivity* and *Support Functions* are *Application Servers* with access to the current sessions maintained by the *IPTV Session Manager*.

Active sessions are accessible through well-defined interfaces and APIs.
In contradiction to the *Procedural* and the *Declarative Application Environments*, the building of applications for a *Session-Oriented Application Environment* is executed by creating application logic on both end points, namely the end device and the *Application Server*. On the one hand, this allows for a de-coupled development, e.g. with regards to the runtime or programming languages used on both sides. On the other hand, the signaling necessary between the two endpoints limits the level

of complexity of the created applications. Nevertheless, this is enough for the most successful interactive application scenarios like user votings, polls, and quizzes.

Content Provider The Content Provider is an entity that creates, owns or is licensed to sell content. In the presented model, the Content Necessary metadata is also provided.

Content Delivery Network Provider The *Content Delivery Network Provider* provides a Content Delivery Network (CDN) and is responsible for the efficient delivery of contents from the *Content Provide*r to the different *End Devices*.

2.3.3. Procedural Application Environments - Downloadable Applications for Interactive IPTV

A procedural application is an application that primarily makes use of procedural information to express its behavior; a Java program is an example of a procedural application [129]. The development of standardized *Procedural Application Environments* was very much influenced by the *Digital Video Broadcasting Project* with the so-called Multimedia Home Platform and a globally harmonized subset called Globally Executable MHP (GEM).

In this context, a number of sub-standards have been developed to fulfill regional requirements for Europe, Asia, the US and South America. These variances mostly consist of adaptations to different requirements in service information (metadata), content protections and application execution (see Figure 2.10 for details).

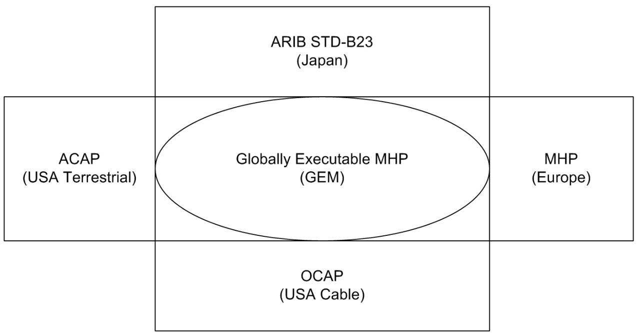

Figure 2.10.: Regional Variances of the GEM specifications [29]

2.3.3.1. Relevance for this thesis

The Multimedia Home Platform has heavily influenced developments in the field of interactive Television. As described in the previous sections, alongside the global version of MHP, namely GEM, various localized adaptations have been created to fit regional requirements. Its main paradigm, a procedural application environment that relies on a so-called thick client approach[14], seems to be outdated today [64]. Web technologies — also a part of the MHP specifications but not completely within the scope of MHP, have taken over current developments.

Within the scope of this thesis, the author has concentrated on more lightweight solutions that rely on session-oriented principles or incorporate browser-based, declarative environments.

2.3.3.2. Limitations of Procedural Application Environments

As described, the MHP platform defines a complete framework for the delivery of Interactive TV. Nevertheless, MHP is not widely available on the market nor it is present in any current IPTV deployments. The following points have been gathered from various sources [64, 105] and attempt to uncover the reasons why MHP is not widely deployed.

- Slow time to market due to the requirement of integrating multiple running applications (essential for the de-coupling of scenarios created by different application providers).
- Complicated profile specifications with many options for non-mandatory features (e.g. Web browser).
- Relatively high level of effort needed to create simple applications, especially when integrating communication aspects.
- Limitations on the generation of a high performance and aesthetically pleasing user interface mainly derived from the limitations of the underlying Java platform
- Extremely strict security model derived from a not-so-strict Java platform
- Absence of high DSL broadband penetrations during the time MHP was available and currently not competitive with evolved Web technologies.

2.3.4. Declarative Application Environments - Web technology for Interactive IPTV

This section first reviews different regional and technological alternatives for *Declarative Application Environments* (DAE) and then identifies the potential gaps in and drawbacks to this technological approach. A declarative application is an application that primarily makes use of declarative information to express

[14] A fat client or rich client is a computer (client) in client-server architecture networks that typically provides rich functionality independent of the central server.

its behavior; an XML document instance is an example of a declarative application [129]. The Open IPTV Forum's specification describes as DAE application as:

"...A DAE application is an associated collection of documents (typically JavaScript, CSS and HTML or SVG documents). ...The difference between a DAE application and a traditional web page is the context within which it is loaded and executes.....Both applications and traditional web pages have an initial document, almost always written in HTML, which can include the contents of other documents. These included documents can have a variety of types, including Cascading Style Sheets (CSS), JavaScript, SVG, JPEG, PNG and GIF." [98]

In the field of interactive TV, declarative environments have garnered more and more attention, as Web browsers become a mandatory feature in TV sets and Set-top-Boxes. This review describes the most relevant technologies available in this context and relevant for IPTV. This includes, the Mediaroom Presentation Foundation (MPF), CE-HTML and HTML5. Old standards as DVB-HTML or ACAP-X have been left out in this overview but have been added to the comparison of DAEs in Table 2.4.

2.3.4.1. Mediaroom Presentation Foundation

The Microsoft Mediaroom platform provides three different approaches to the creation of interactive applications, which have evolved during the development of the platform in the sequence listed below:

1. **Remote Desktop Protocol (RDP) Applications:** RDP Applications can be in the form of Web applications or stand-alone Windows applications and can interact with remote resources, such as Web servers and databases. RDP is well suited to the delivery of Web and stand-alone Windows application UIs, but it is not efficient for the delivery of motion graphics or sound.

2. **Browser Applications:** Microsoft Mediaroom Browser is an XHTML 1.0 browser with support for CSS and ECMAScript (JavaScript). Limitations appear through the environment itself (simple browser) and the platform's limited computational power on the STBs.

3. **Mediaroom Presentation Framework (MPF):** MPF is an application platform incorporating dynamic, server-created contents leveraging ASP.NET for developing applications to build a set of abstract Mediaroom controls on ASP.NET that map to native client controls. This paradigm allows for the execution of applications in the back-end and saves resources on the client side platform.

Mediaroom applications can be developed easily using an development process which is completely integrated into the common Microsoft IDEs. Additionally, a so-called

Personal Server environment is available, which basically simulates the whole network side infrastructure. In combination with developer STBs, a whole Telco IPTV ecosystem can be set up using one server computer.

Without a doubt, the available tool kit including the Personal Server and IDE integration offers a unique combination at the point this thesis was written.

2.3.4.2. CEA-2014 (CE-HTML)

Consumer Electronics HTML (CE-HTML) is part of the *Web-based Protocol and Framework for Remote User Interface on UPnP Networks and the Internet* [7], also known as Web4CE. CE-HTML has been defined by the Consumer Electronics Association (CEA) under the name CEA-2014.

Developments of CE-HTML have been driven mainly by the consumer electronics industry and were released to the public in 2006. CE-HTML mainly relies on approved Web standards and combines these standards towards the creation of Declarative Application Environments for TV sets and STB. Its most relevant feature is the definition of a *Audio/Video Object* (A/V Object), allowing the playback of streaming media inside its browser environment. In detail, the following Web standards have been combined to create CE-HTML:

- XHTML 1.0 Strict or Transitional
- JavaScript / ECMAScript
- DOM2
- CSS TV Profile 2.0

Additionally the *XMLHttpRequest* and *Notifsocke*t, allowing for bi-directional communication between a client and a Web server, local scripting support for controlling the A/V Object, as well as a save and restore mechanism for user profiles have been added. In the following sections, details concerning the modification to the used Web standards will be summarized

- XHTML represents the current State-Of-The-Art for building Web pages in the WWW. The eXtensible Hyper Text Markup Language was defined by the W3C and is identical with HTML 4.0 on a feature level.
  Concerning CE-HTML, XHTML has been modified through the removal of compatibility with Frames and Applets and addition of an OP-element that allows mapping between a key code on a remote control and its representation on the TV screen.
- CSS is supported in CE-HTML using the CSS 1.0 TV Profile. This profile is based on CSS 2, with some extension of CSS 3. This extension allows for

specific requirements to be supported on a TV or STB as opacity for overlay used for controls on the TV screen and includes recommendations for colors and fonts to be used on a TV.

- In addition to the standard JavaScript / ECMAScript objects, CE-HTML defines some supplementary events relevant for usage on a TV or STB. These include:
  - The A/V Scripting Object
  - NotifSocket
  - Save/Restore
  - XMLHttpRequest
- CE-HTML adapts the DOM 2 (Document Object Model Level 2) specifications of the W3C with regards to ECMAScript (Core, Style, events) and HTML. Minor modifications have been made, e.g. to Applets, that do not necessarily have to be supported.
- The *Save/ Restore* functionality is a server-oriented feature that allows clients to store the current state (e.g. the user interface) created by the server. This allows for terminal mobility or for resuming a UI on a end-device that has been shut-off and turned on again.
- *Profile Matching* is used for capability matching between a browser on a client and a Web server that offers CE-HTML pages. This feature is used to allow a server to create dynamic Web pages based on the client's capabilities like screen size, color depth, supported MIME-types, audio- and video-codecs and available fonts.
- CE-HTML defines two kinds of notifications that allow a Web server to communicate directly with the browser without the need for continuous polling through the client. The first realization is called In-session notifications and based on the so-called *NotifSocket* object. *NotifSocket* allows for a permanent TCP connection to the browser on an arbitrary port. This connection can be used to initiate a call back to the browser. The originating connection has to be set up by the client.
- The *CE-HTML Media Object* is used to render audio or video contents inside the browser environment and to be controlled via JavaScript. Looking at the standard, no specific audio or video codec is defined, and which codec is supported is just an implementation issue. Using the Profile Matching feature described above, a browser can negotiate its codec capabilities with a Web server.

2.3.4.3. HTML5

HTML5 [66] is the result of the fifth major revision of the core language of the World Wide Web, the Hyper Text Markup Language.

In contrast to technologies specifically created for TV and IPTV like CE-HTML or MHP, HTML5 cannot be considered to be part of a – in this case *Declarative Application Environment* – per se. Nevertheless, due to the current direction of the WWW as focusing increasingly on rich media contents and streaming audio and video, HTML5 contains various new features that help fulfill the requirements for a *Declarative Application Environment*, and therefore will become part of future middleware systems for IPTV. The following section describes the relevance of HTML5 for IPTV by describing new elements and features of the HTML5 standard that might have a relevance for this work and are limited therefore to aspects like rich media and streaming media.

The information provided here has mostly been extracted from a document published by the W3C entitled *"HTML5 differences from HTML4"*[15] This document describes the differences between HTML4 and HTML5 and provides some of the rationale for the changes. Since HTML4 became the defacto standard for the WWW in 1997, the Web has developed towards a platform where users can do more than just consume text and pictures or download files.

Streaming media, like audio and video contents, are also becoming increasingly popular. First, these requirements were met by the introduction of technologies for Rich Internet Applications (RIA) like Adobe's Flash, Microsoft's Silverlight or Sun's JavaFX. The main drawback to these technologies is that they require a plug-in for implementation in each browser and do not comply with open Web standards. To overcome these drawbacks, new elements have been added to the HTML language defined as the <audio> and <video>tag, allowing for a native representation of corresponding contents. Both provide an API enabling application authors to script their own user interface. By also implementing the newly-introduced <source> element in combination with the above-mentioned <audio> and <video>tags allows for the definition of multiple streams of different types, i.e. using another video codec or bit rate.

Listing 2.1 provides an example of a HTML5 video element containing two sources for a video file. The first one is encoded in MP4, the other one uses the Ogg codec.

Listing 2.1: Usage of the HTML5 <video> and <source> tags

```
...
<video controls autoplay>
 <source src='video.mp4' type='video/mp4; codecs="avc1.42E01E, mp4a.40.2"'>
 <source src='video.ogv' type='video/ogg; codecs="theora, vorbis"'">
</video>
...
```

[15] http://www.w3.org/TR/html5-diff/

2.3.4.4. Discussion

This section has introduced different technological approaches to the generation of *Declarative Application Environments* for IPTV, including DVB-HTML, ACAP-X, CE-HTML as well as HTML5.

Besides some major problems in reaching the mass market, as in the example of DVB-HTML, which is tightly linked with DVB-J and MHP and therefore suffers from limited deployments, *Declarative Application Environments* do represent the actual State-Of-The-Art in Interactive IPTV development. Their main advantage, in comparison to the *Procedural Application Environments* is mostly that they originate from the ongoing developments inside the Internet, and CE-HTML and HTML5 already support the features required to render rich media contents.

Due to this feature, one cannot create State-Of-The-Art environments for IPTV without integrating a declarative approach. In the scope of this thesis, CE-HTML and HTML5 are of particular relevance and became part of the architecture and middleware created during work on this thesis.

Application Environment	DVB-HTML	ACAP-X	Mediaroom	CEA-2014	HTML5
Programming Language	XHTML	XHTML	ASP.NET	CE-HTML	HTML5
Scripting Language	JavaScript	JavaScript	limited JavaScript	JavaScript	JavaScript
Native Video support	dvb-tv	yes	Video Application Object	CE-HTML Video Object	HTML5 Video Tag

Table 2.4.: Comparison of Declarative Application Environments

2.4. Related Works & Projects

This section describes the most relevant and influential projects concerning IPTV in research and academia.

2.4.1. Towards a Media Internet

The last sections have described how, beginning with Digital TV and initial interactive services, IP-based systems can be used to deliver rich media contents. After such a discussion, it becomes clear that the distinction between OTT and managed network approaches will decline, as the current state of the Internet accelerates towards the development of a platform capable of transporting high quality and high bit rate multimedia contents and corresponding services.

According to Minoli [91] one can anticipate (and his first and second anticipation have already proven accurate) several phases in the deployment of IPTV that reflect current technological developments. Minoli distinguishes these phases as the path toward the *Media Internet*:

- **Phase 1**: IPTV introduced by the Telcos for the commercial delivery of entertainment grade video over managed walled garden networks end edge devices (2007-2010).
- **Phase 2**: IPTV and hybrid broadcast/IP services introduced by the satellite and cable TV companies for the commercial delivery of entertainment-grade video over their broadband sat/cable infrastructure (speculative, 2010+).
- **Phase 3**: IPTV to morph to Internet TV for the commercial delivery of all video content except that of entertainment-grade quality (2011+).

The missing piece of these anticipations is the inclusion of additional services and corresponding platforms that would make the switch from Digital TV to an IP-based system worth it.

In addition to the plain video services, blended interactive communication and Social Networking will play a particularly key role in the decision-making process of the consumer.

This combination of technological approaches on the one hand, and the concentration on user needs on the other, opens up a needed field of research touched by this thesis. To identify the key drivers in research and academia, the next sections will provide an overview of the most relevant projects in this direction.

2.4.2. IPTV Architectures & IPTV Session Management

In addition to the author's work on implementations within the context of this thesis, other research groups have also influenced the research landscape concerning session-driven IPTV infrastructures. In [90], Mikoczy et. al. describe a prototype environment for session-oriented IPTV services developed in the context of a project called Scalenet[16].

In addition to the architectural issues and adaptations of draft standards for IPTV, a real world test bed was implemented. Its focus is the description of the prototypical realization of the platform. This includes a brief overview of some implementation details and some QoS measurements in the provided infrastructure.

In [84], Menezes describes his work on a SIP-based IPTV platform integrated into an IMS infrastructure. The main focus of his research centers around signaling aspects for session-oriented IPTV, the prototypical implementation of the Session Controller and measurements of session setup, channel change and QoS adaptation scenarios.

In [120], Sivasothy et al describe an approach to providing an extension to the SIP protocol for controlling IPTV sessions in the context of Next Generation Networks.

In [8], Chatras et al provide an overview of the current work of the ETSI TISPAN standardization body on IPTV and provide some details on IPTV service signaling.

In [27], Deventer et al present an approach allowing IPTV users to participate in a virtual game show by using mechanisms for uploading User Generated Content to a managed IPTV environment. This is achieved through the combination of SIP and RTSP for session signaling and media control, respectively. The authors present their proposed architecture, necessary signaling flows and a prototype for implementation based on Open Source and off-the-shelf components.

Related work on API-oriented approaches allowing for the manipulation of IPTV Sessions can be found in [81] and [88].

2.4.3. Interactive Application Environments, Shared Experiences & Social IPTV

Interactive Application Environments like the *Procedural Application Environment* (PAE) and the *Declarative Application Environment* (DAE) have been specified during standardization within the different Standard Development Organization (SDOs) as DVB for MHP/GEM [36] and the DAE within the Open IPTV Forum [98]. In contrast to that *Social TV* systems are still part of current research. In [104, 23], Pelt and Coppens present one of the earliest efforts in the area of Interactive TV and Social Television that can be recognized as a sort of reference project in this domain. The main idea consists of providing avatars for every single user, as a way of visualizing their presences. A variety of faces are available for use as avatars and express the user's emotions during content consumption. In addition, so-called shared video effects allow for another way to express emotions between users (e.g. a flaming ball whizzing across the screen). The XMPP or Jabber protocol [116] is used for communication purposes. The Jabber implementation used for AmigoTV allows for buddy list and presence functionality. A so-called Room Functionality

[16] http://www.scalenet.de

was added for the creation of a community scenario, in which multiple users can create a virtual private area for content consumption. In [94] Nathan et al. from the AT&T labs present their work on a Social TV system called CollabraTV. Unlike some other systems, like the below-discussed AmigoTV or STV, CollabraTV is focused on supporting asynchronous communication. Results were gained through extensive lab studies that tested various features of the platform.

The ConnecTV project was carried out within the so called B@Home project and is described in detail in [12]. ConnecTV is unique in the field of Social Television research because the prototype was implemented for a field test in about 50 households in the city of Enschede.

With regard to features, ConnecTV uses the XMPP protocol, like in the Amigo TV project, for communication purposes between users. This includes features like a *buddy list*, *rich presence* that shows other users' TV channels and a feature that allows for switching to channels other users are currently watching.

Additionally, *TV program recommendations*, *follow and invite a friend* scenarios have been implemented. A *most popular channel* service community feature was also available during the field test.

In the results, the *follow a friend* scenario in combination with the available buddy list was used by most of participants, showing the acceptance of communication features. The idea of the *virtual couch* is the best description of the research on the different iterations in the Social Television System (STV) project [60, 127, 67].

During the development of this project, throughout various prototypes and case studies, the design goal was to enable a small group of relatives or friends to share with each other while watching TV. The key elements each evolved during different phases of the project (STV1-3). The key elements of the current version are:

- A television presence that includes a buddy list and ambient display
- The creation of a Shared Viewing Experience though program suggestions from other buddies
- Two different communication channels, including text chat and group voice calls
- The integration of a user's viewing habits into the EPG of other buddies

When examining the results so far, it turns out that, in Shared Viewing scenarios, some communications channels are better suited than others. Especially in that voice chats are more accepted than text chat.

Additionally, most of the participants did not even want the video chat feature, a result that the author can also underline as a result of his own experiences. Several other aspects also seem to be unresolved like how multiple users in one household can be supported and CE equipment allowing for an easy integration of community features can be developed.

More theoretical assumptions on the evolution of interactive services as well as the different levels of interactivity can be found in the following publications: In [77], Körling et al. presents a model of the evolution and relative roles of interactivity realizations. This model describes how the current media break[17] in mixed TV and telecommunication environments will disappear through the use of IPTV: The starting point for interactivity is TV+telephony and TV+SMS (e.g. for voting and user-poll scenarios); and now TV+PC (e.g. video portals like the Mediathek introduced by the German public broadcasters ARD[18] and ZDF[19]) is growing as a way for broadcasters to attract more viewers.

In the end, (mobile) IPTV with integrated media and communication services, and without any kind of media break, will dominate.

Hjelm [64] identifies four levels of interactivity: the first three are directly addressed in the conclusions of this thesis and the fourth forms a current field of research for the author:

1. The first level of interactivity enables a user to interact with the meta-information about the content, such as the program guide. This includes Video-on-Demand, setting personal video recorders, and selecting content in an Electronic Program Guide (EPG).This level of interactivity had already been reached in DTV systems used in the late 20th century. A refinement is currently taking place because some of the functionalities (e.g. personal video recorders) are becoming services provided inside the network and are no longer based locally.

2. The second level is reached when the user is given access to external information not necessarily related to the program. This includes Teletext or portal pages. The user can obtain news and other information, but the type of interactivity is limited to pointing and clicking. This level of interactivity has also been implemented through different kinds of technologies (e.g. as Teletext in the mid-seventies of the last century or by other means, e.g. using MHP).

   Nevertheless, a breakthrough in portal driven approaches is taking place as this thesis is being written, through the integration of a Web browser into the TV sets. This approach is reflected in the section on *Declarative Application Environments* (DAE) presented later in this thesis. The main advantage of browser-driven environments is their ability to present contents (e.g. news pages, program guides, games) generated dynamically by corresponding Web servers on the content provider's side.

3. The third level consists of services that prompt the user to react to events embedded in and related to the content. In this case, the user is able to

[17] Whenever the medium changes during a work process, a media break takes place. In most cases, the result is a higher amount of work and a disruption of the work routine.

[18] www.ardmediathek.de

[19] www.zdf.de/ZDFmediathek

influence the program through the use of voting and betting services. In order to realize these kinds of scenarios, different technological approaches can be chosen.

As we will see later, session-oriented approaches play an especially key role in the realization of these services. The key advantage of the session-oriented for the means of personalization and direct reach ability. The session allows service providers to identify each individual user, as well as the current status (e.g. which channel the user is currently watching to).

4. The fourth level extends the above-mentioned type of interactivity by allowing actual story changes depending on how the user interacts with the content. This includes both explicit interactions, where the user makes a choice concerning how the program should proceed; and implicit interaction, where previous user actions are taken into account in shaping the program.

   Also, in this context, session-oriented approaches are best suited to the management of these kinds of scenarios as an individual user's status and feedback are needed to create this kind of experience.

2.4.4. Interactive IPTV Services

Addressing single users or user groups with personalized services has always been on the wish lists of broadcasters and content providers.

For example, Targeted Advertisement Services are an important topic inside various standardization bodies and research. Various SDOs have developed corresponding specifications: The Open Mobile Alliance (OMA) with the so called *OMA Mobile Advertisement (OMA Mob Ad)* [100] specification and the North American Society of Cable Telecommunications Engineers (SCTE) and their *SCTE-130* specification. The SCTE standards for targeted advertisement are additionally approved by American National Standards Institute (ANSI).

Another session-oriented IPTV service combines professional as well as User Generated Contents (UGC). This service, called User Participation on TV perfectly fits the idea of Social IPTV. Specifically, the idea of integrating *User Generated Content* is hereby mapped to a form of video chatting as described in various projects.

Additionally, these concepts are also relevant to IPTV standardization [39]. In this context, Deventer et al [27] have introduced a concept for studio controlled upload of User Generated Content in mixed SIP and RTSP signaling environments. The ideas presented in this paper have influenced the author's work on *Virtual Quiz Show* scenarios presented later in this thesis. Where Deventer et al. assume a heterogeneous infrastructure with either speaking the SIP or RTSP protocol, the author introduces a completely integrated approach using both protocols on all entities.

2.4.5. IPTV Meta Sessions

Research on the *IPTV Meta Session* model, as acting as an enabler for Social IPTV services, has been accompanied by different international research activities. Work on the *IPTV Meta Session* model has mainly contributed to the European Research Project iNEM4U[20], where the *IPTV Meta Session Model* has been used to connect the different partners developments into a unified infrastructure.

As the developed mechanisms present novel work and created a new field of research, related work on *IPTV Meta Sessions* does not exist. In principle, concepts from Social IPTV, as presented in Section 2.4.3 fit in certain aspects to the idea of *IPTV Meta Sessions*.

The latest developments within the Internet Engineering Taskforce (IETF) reflect concepts from *IPTV Meta Session*. The IETF draft "Disaggregated Media in the Session Initiation Protocol (SIP)"[21] edited by Camarillo et al describes the

"ability for a user to create a multimedia session combining different media streams, coming from different devices under his or her control, so that they are treated by the far end of the session as a single media session."

and show that the concepts developed within iNEM4U meet real world requirements. Gonzalo Camarillo was one of the reviewers of the iNEM4U project in April 2010.

2.4.6. Interactive Content Provisioning

Related work on session-oriented media delivery infrastructures for the provisioning of interactive contents, is a very specific field of research, and often linked to developments within IPTV standardization bodies like ETSI TISPAN and the Open IPTV Forum.

In [136], Waiting et al. describe developments at the University of Capetown (UCT), South Africa on the UCT IPtv server. This server is a SIP application server that streams up to three channels to multiple destinations. The server is built on top of the Open Source SIP library *eXosip* for the support of session- based SIP signaling and the *GStreamer Library* for media delivery and it is released free on the Internet under the GPLv3 licence. Its primary goal is to serve a limited number of packet-based media streams to as many clients as possible, similarly to regular digital terrestrial and satellite television broadcasts. The server is designed to fit tightly within IMS architectures therefore, unlike other IPTV solutions, it uses SIP exclusively for signalling.

In [84] Menezes describes, as part of his dissertation, an IP Television (IPTV) service architecture that applies the Session Initiation Protocol (SIP) for session and

[20] http://www.inem4u.eu
[21] https://datatracker.ietf.org/doc/draft-loreto-dispatch-disaggregated-media/

media control, while incorporating a design suitable for deployment in the context of an IP Multimedia Subsystem (IMS) architectural framework. The main features of the architecture include flexible delivery of personalized multimedia streams, adaptability to the characteristics of the networking infrastructures and channels used for transmission, and a modular design to facilitate implementation of new functionalities and services. The developed solution is specifically designed for live multimedia streaming, such as broadcasted events, independent of the cast mode (unicast or multicast). Private Video Recorder (PVR) functions and Video On Demand (VoD) services are supported, their control is ensured by standard SIP messages.

Menai et al. describe in [83] their work on standard SIP/RTSP based Content Delivery Networks (CDN). The system includes a central server (Content Delivery Network Controller) that analyzes all received content delivery requests. The Content Delivery Network Controller (CDNC) chooses the cluster of servers to which a request should be redirected. The choice is made depending on client location, content availability, location and servers' global load. Each cluster is controlled by a Cluster Controller (CC) that would choose the final VoD server to deliver the content based on a fine grained analysis of the load on the VoD servers it manages. The system proves the feasibility and flexibility of SIP interfaces when coupled with RTSP to organize redirections within a CDN.

Finally, Daher describes in his diploma thesis [25] an infrastructure which combines the benefits of a session-oriented approach for the dynamic provisioning of multimedia streaming services, with a Service Oriented Architecture (SOA). It provides a generic API, allowing third party services to make use of the provided infrastructure. Daher's work represents research activity parallel to the author's work, performed at the Fraunhofer Institute FOKUS.

2.5. Summary

This chapter has introduced the State-of-The-Art of IPTV and *Interactive Application Environments* relevant for this thesis. Beginning with an overview of the basics of IPTV, acronyms and technological background, *Interactive Applications Environments* and related technologies have been discussed. Beginning with *Procedural Application Environments* (PAEs) and *Declarative Application Environments* (DAEs), both *Social TV Environments* and *Session-Oriented Application Environments* were presented. Furthermore, an analysis of the advantages and drawbacks, and their role in future developments within this thesis have been provided.

Finally, related works and a set of reference projects have been presented.

In the following, the results of this chapter will be used to discuss requirements for interactive IPTV, design an overall architecture, specify interfaces, APIs and services and finally implement a novel IPTV system named the *Open IPTV Ecosystem Core*.

3. Interactive IPTV Requirements Analysis

Chapter 3: Interactive IPTV Requirements Analysis

Requirements for IPTV Session Management,
Interactivity,
Content Provisioning and Delivery,
Support Functions

In the previous chapter, the relevant standards and technologies allowing for the construction of interactive IPTV systems were introduced.

This chapter derives the corresponding functional and non-functional requirements, allowing for the fulfillment of the vision in the first chapter. The defined requirements will be then used to design and specify the *Open IPTV Ecosystems Core* later in the course of this thesis.

The requirements defined in this chapter have been derived from the work on and the author's active contributions to the requirements phase for the ETSI TISPAN Release 2 [38] as well as the Open IPTV Forums Release 2[1] specifications. Furthermore, requirements going beyond the scope of the above-mentioned standards have been published in [4] and [62] and selected from the European Commission's vision statements on a Future Media Internet [78, 9].

By using the defined requirements, it is furthermore described how these requirements might help to evaluate the different components of the *Open IPTV Ecosystem Core*.

The requirement analysis has been conducted according to the five main categories identified in the *Generic Multimedia Service and Delivery Framework* from Chapter 1. This includes:

- Requirements for *IPTV Session Management* including overall architectural aspects and extended with requirements for *Third Party Access* and *Multi-User*, *Multi-Content Sessions* (Meta Sessions).

[1] http://www.oipf.tv/docs/OIPF-T1-R2-Service-and-Platform-RequirementsV2_0-2008-12-12.pdf

- Requirements for *Interactivity*, incorporating extended *IPTV Sessions Management* capabilities and *Social IPTV* aspects based on the *IPTV Meta Session Model.*
- Requirements for *Dynamic Content Provisioning* allowing for adaptation and combination of different content sources.
- Basic requirements for *Content Delivery.*
- Requirements for *Support Functions* forming an end-to-end IPTV system.

Figure 3.1 depicts the five categories for which functional and non-functional requirements will be described in this chapter based on the Generic Multimedia Service and Delivery Framework from Chapter 1.

Furthermore, and combined with the session-oriented *IPTV Role Model* from Chapter 2, these requirements will allow for the derivation of the *Open IPTV Ecosystem Core* (OIEC) functional architecture in the next chapter.

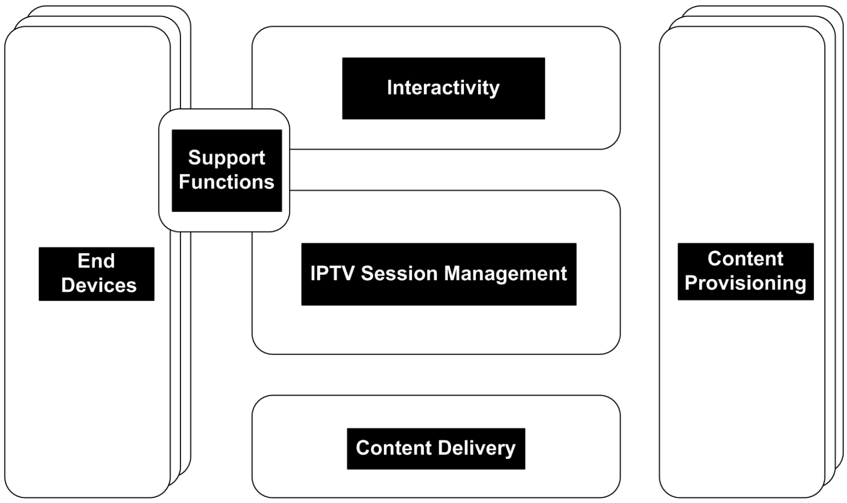

Figure 3.1.: Extended Generic Multimedia Service and Delivery Framework

3.1. Requirements for IPTV Session Management

As described in Chapter 2, the session-oriented *IPTV Role Model* enables Service as well as Content Providers to control service delivery to the end user. This includes plain streaming services, as well as the creation of more sophisticated interactive services.

These services are enabled by using exposed session data made available to third parties.

In this section, an emphasis has been put on the requirements necessary for interactive streaming services and how these services might be created using the *IPTV Session Management.*

3.1.1. General Requirements

Standards Compliance As described in the previous chapter, the session-oriented *IPTV Role Model* is part of different standards for IPTV including the Open IPTV Forum, ETSI TISPAN and ITU-T. For this reason, it must be assured that the concept presented in this thesis is in line with these standards to a certain degree. This allows for comparisons with other commercial and non-commercial systems. At the point this thesis was written, no commercial systems were available on the market, which makes standards compliance even more important. Ensuring that this requirement is met allows the author to act as a reference for and compatible with other systems.

Interoperability One of the main goals of, and directly linked to the requirement for standards compliance, is interoperability with other systems. This might include systems being part of the *Open IPTV Ecosystem Core* or systems from other research groups. Interoperability is assured by implementation according to specifications from (draft) standards or by defining and publishing descriptions of new interfaces and reference points. Furthermore, interoperability must be assured by conducting interoperability tests.

Integrability Integration and interaction with other commercial and non-commercial solutions is one of the most important requirements for *IPTV Session Management.* As the *IPTV Sessions Management* acts as some kind of central component in a IPTV system, the integration with or into other solutions has occurred several times during work on the *Open IPTV Ecosystems Core.* Integrability is ensured through enabling interoperability through standard compliant interfaces and reference points for relevant functionalities.

Scalability The *Open IPTV Ecosystems Core,* and therefore also the *IPTV Session Management* represents an academic and non-commercial implementation of a session-oriented IPTV system, party fulfilling open standards where available. In this context, the scalability of the designed components is also an important requirement when comparing the solutions with other approaches or analyzing the feasibility of the approach chosen. Scalability can be ensured by fulfilling different sub-requirements including:

- A clean and efficient implementation with fast memory allocation schemes, event handling and parsing mechanisms.

- Choosing an Application Server platform with accurate performance to avoid external bottlenecks. Load balancing mechanism should be deployable, where necessary.
- A fast database or corresponding non-persistent usage of local memory.
- A non-blocking network connection.
- Fast and reliable hardware, including fast CPUs and adequate availability of local memory.

Performance The *IPTV Session Management* represents the central component of all session-oriented IPTV approaches. Each user request, i.e. requests for content, trick functions and also interactive services are handled by the *IPTV Session Management*. For this reason, this component and corresponding implementations have special requirements with regards to performance.

The performance of SIP applications servers is mostly measured by testing their response time for a single session's request and challenging them with parallel requests.

In the context of this thesis, the response time has especially been taken under analysis, as this directly influences channel switching times and the delay for other service requests. Parallel usage, i.e. measurements on the possible service requests per second have also been taken under analysis but cannot be compared with commercial environments.

Performance metrics relevant for *IPTV Session Management* can be derived from corresponding standards documents, e.g. the ITU-T recommendation G.1080 *Quality of experience requirements for IPTV*[2] services defining IPTV Quality-of-Experience (QoE) metrics.

This document does not provide an exact range for the channel switching procedure as this is still under discussion. Contributing ITU-T members suggest a maximum channel switching time of around 2000ms.

3.1.2. Streaming Services

Basic streaming services like Linear TV or Video-on-Demand have already been described in Chapter 2, through the introduction of the session-oriented *IPTV Role Model*, which was originally derived from standards for Voice-over-IP (VoIP).

The Session Initiation Protocol (SIP), corresponding session-related methods like INVITE, INFO, UPDATE and BYE together with new specifications for the Session Description (SDP) are used in this context. These basic principles allow for monitoring and tracking of user requests and also provide enough flexibility for interactive features and scenarios described later.

[2] http://www.itu.int/md/T05-SG12-080522-TD-GEN-0370

3.1.3. Third Party Openness

Third party openness adds the functionalities of a Service-Oriented-Architecture (SOA) to the idea of session-oriented IPTV.

By introducing an abstraction layer on top of the plain *IPTV Session Management*, an Application Programmable Interface (API) can be provided. This API allows other applications and services to read and manipulate IPTV sessions data. To enable *Third Party Openness*, the specification of an API and data model that allows for the exposure of information concerning ongoing sessions, must be implemented on the Application Server.

3.1.4. Meta Sessions

Meta Sessions allow for the combinations of single user sessions, and are therefore an explicit requirement for the enabling of multi-user scenarios and shared content consumptions. The *IPTV Meta Session* concept has been developed by the author in the context of various research projects.

The *IPTV Meta Session* concept enables interactive IPTV features, especially *Social IPTV*.

3.2. Requirements for Interactivity

Interactivity has been recognized as a general term for all services that extend the plain consumption of Linear TV, beginning with the availability of a remote control. For this reason, it is not possible to define requirements for all forms of interactivity within this thesis.

Due to the ongoing convergence between the mobile and fixed network domains, including corresponding services, the requirements for the executions of applications and the signaling of value added services, e.g. targeted advertisement have been specifically taken under analysis.

3.2.1. General Requirements

Simplicity TV service consumption differs from services and applications, e.g. consumed on mobile devices. So-called lean-forward services, requiring active user participation can only be successful to a certain extent, due to limited interaction capabilities through the remote control.

Quality-of-Experience Usability and ease of use were the most limiting factors in early interactive TV services. Even nowadays, interactive services suffer from slow hardware in Set-Top-Boxes and TVs. Furthermore, controlling interactive services on the TV is still a partly unsolved problem.

Composibility New services are often developed through a combination of services that originally were not meant for that. The most prominent example are incoming call signaling scenarios on the TV, and all scenarios in which Social Media services are combined with TV services. For this reason a general requirement when building an interactive TV platform is composability with other services, so-called mash-ups.

Technical Applicability Various concepts for interactive television have been developed in the past. As of today, interactive services have only partly reached mass markets. This is mainly due to concepts not being relevant or applicable to the current State-of-the-Art technology. The most limiting factors were missing back channel connections and still comparatively slow hardware and limited memory on consumer electronics devices.

Market Reach Sufficient market reach is a very basic requirement whenever the technology that might be chosen for interactive services must be decided upon. In the past, various approaches including the presented *Multimedia Home Platform*(MHP) have been brought to the market, but have not been successful due to the different reasons already discussed in Chapter 2.

Cross Platform Deployability So-called multi-screen scenarios gain more and more attention. In this context, interactive scenarios must also be executable on combination across multiple devices. From a technical perspective, two different paradigms can be implemented:

1. Thick client approaches, where interactive services are implemented as native applications for each platform and executed on the *End Device*. The *Multimedia Home Platform* (MHP) represents such an approach for a *Procedural Application Environments* (PAE) on Set-Top-Boxes.

2. Thin client approaches, using e.g. Web technologies to run applications from an Application Server and to render applications in a Web browser on the *End Device*. The *Declarative Application Environments* (DAE) are examples of such an approach.

3.2.2. Interactive Application Environment

An *Interactive Application Environment* allows for the execution of interactive services on TVs and Set-Top-Boxes. Three different approaches, namely the *Session-Oriented Application Environment* (SAE), the *Procedural Application Environment* (PAE) as well as the *Declarative Application Environment*(DAE) have been discussed and compared in Chapter 2. The creation of services for these application environments requires that certain technologies be available on the *End Device*. In contrast to the PAE , SAE and DAE approaches can be implemented in thin client environments using a Web browser, with some modifications for the TV environment and an adapted declarative description language like CE-HTML or HTML5. Furthermore, protocol stacks for HTTP, SIP, RTP and RTSP should be available.

3.2.3. Interactive Services

Interactive services represent implementations of use-case enriching the usage of streaming services. Technological requirements, potential technologies and different levels of interactivity have already been described in the preceding chapter. Based on these assumptions, different classes of interactive services can be implemented. Interactive Services for TV environments should strictly follow the basic requirements for interactivity defined in the beginning of this section.

3.2.4. Social IPTV

State-of-the-Art IPTV environments are required for the integration of Social Media services. The most prominent examples are Facebook and Twitter, representing a class of applications and use cases, in which users are enabled to share personal information and content and communicate with each other. In this context, Social IPTV is recognized as a specific form of interactivity on the TV set.

3.3. Requirements for Content Provisioning

Content Provisioning abstracts all network elements that provide the necessary functionalities for supplying and adapting contents. This includes Live TV channels and Video-on-Demand (VoD) assets. In the context of this thesis, the requirements for *Content Provisioning* are limited to relatively simple provisioning and adaptation aspects. This includes streaming capabilities in multiple bitrates, adapted through information provided during the session setup (SIP INVITE) or session modification (SIP re-INVITE/INFO). In addition, content mixing capabilities are required for the server end devices that are not able to render multiple streams at once.

These requirements can be fulfilled using and modifying Open Source tools like the VideoLan[3] software in combination with an Open Source SIP stack.

3.4. Requirements for Content Delivery

Content Delivery abstracts all network elements necessary for delivering content from a content source, e.g. a broadcaster or content provider, to the *End Device*. In the context of this thesis, the requirements are limited to network parameters necessary for the delivery of contents from the broadcaster or content provider to the *End Device*.

[3]http://www.videolan.org

Derived requirements contain a reliable IP network connection, where the available bandwidth per user should not exceed 8-10 MBit/s for standard definition contents, and 8-20 MBit/s for high definition contents. Multiple *End Devices* in one household extend the required bandwidth, respectively . Furthermore, IP Multicast capability must be available for the delivery of Live TV contents.

3.5. Requirements for IPTV Support Functions

IPTV Support Functions describe all components of an IPTV architecture required to bootstrap an IPTV service, provide necessary metadata allowing service selection or are responsible for content and service protection. These components have also partly been developed during work on this thesis, in-line with corresponding standards and specifications. As they do not represent a field of research, they are not further discussed and simply integrated as is.

3.5.1. Digital Rights Management

Protecting digital contents against illegal usage and unallowed distribution is the main purpose of Digital Management (DRM). Basically all commercial IPTV systems and Video-on-Demand platforms implement a certain DRM mechanism. DRM mechanisms and aspects are beyond the scope of this thesis.

3.5.2. Metadata

Metadata and corresponding formats are one of the largest fields for research and development on IPTV. For each service and process, beginning with bootstrapping mechanisms up to service consumption metadata is required. Different standardization bodies have specified metadata formats for standard processes. In the scope of this thesis, different metadata formats especially for information exchange in interactive service consumption have been defined.

3.5.3. Service Bootstrapping

Bootstrapping IPTV services subsume all processes that take place during the start-up sequence of a Set-Top-box of TV, when connected to an *IPTV Service Provider*. Two main processes have been defined, e.g. by DVB and are referenced by most other IPTV standardization bodies:

- *Service Provider Discovery* (SPD) describes processes where the STB obtains an IP address, e.g. through DHCP mechanisms and discovers available IPTV service providers. These mechanisms require corresponding network protocols and network entities providing necessary information.

- *Service Discovery* (SD) takes place after a STB has obtained necessary Service Provider Information during the SPD process. During the SD process, metadata on available contents and services is downloaded from the Service Provider. Service Discovery therefore requires corresponding network protocols for downloading metadata as well as a parser on the STB for the display of downloaded metadata.

3.6. Summary

This chapter discussed the requirements for composing an end-to-end system for interactive IPTV, the *Open IPTV Ecosystem Core*. Beginning with the requirements for *IPTV Session Management*, requirements for *Interactivity*, *Content Provisioning*, *Content Delivery* and other *Support Functions* have been analyzed.

The focus of this analysis has been on the *IPTV Sessions Management* and *Interactivity*, as they do represent the main areas of research discussed in the scope of this thesis.

In the following, State-of-the-Art technologies from Chapter 2 and the requirements from this section will be combined and mapped onto a functional architecture defining the *Open IPTV Ecosystem Core*.

4. Design of the Open IPTV Ecosystem Core

Chapter 4:
Design of the Open IPTV Ecosystem Core

NGN Reference Model
Interactive Application Environment
Functional Architecture
Building Blocks

In this chapter, a functional architecture that shapes an end-to-end IPTV system, named *Open IPTV Ecosystem Core* (OIEC) will be introduced[1].

The OIEC specified here combines the concepts for *Session-Oriented (SAE)* and *Declarative Interactive Application Environments (DAE)*. Furthermore, user-to-user communications aspects introduced as *Social IPTV* within Chapter 2 and requirements identified in the last chapter will be part of the composed architecture.

Furthermore, different other supplementary building blocks, which are not part of the core architecture, will be introduced.

The *Open IPTV Ecosystem Core* is a novel, end-to-end IPTV system and focused on how session-oriented streaming services can be realized, how the session state can be exposed to third parties in order to implement interactive services, and how the service usage can be personalized.

Functional architecture design as conducted in this chapter is the second step within a three-tier methodology known as a three stage specification process[2]:

Based on the state-of-the-art in research and technology introduced in Chapter 2 and the functional and non-functional requirements derived in Chapter 3, a functional architecture design will be developed in this chapter. A detailed specification of services and corresponding service signaling represents the last step of this methodology and will be discussed in the following chapter.

Section 4.1 provides background information on architecture design in *Next Generation Networks* (NGNs) and ongoing architectural research for a *Future Media Inter-*

[1]Results of this chapter and preliminary work have been published in [4, 42, 54, 44, 45, 51, 43, 52]

[2]http://portal.etsi.org/mbs/protocolStandards/stagedApproach.htm

net. A mapping of these concepts with the session-oriented *IPTV Role Model*, *Interactive Application Environments* (Chapter 2) and functional requirements (Chapter 3) onto a new functional architecture will be provided afterwards.

Section 4.2 describes the detailed functional architecture for the OIEC, introducing the functional building blocks and corresponding interfaces between them.

4.1. Architecture Design

This section discusses the approach chosen, when designing the *Open IPTV Ecosystem Core* functional architecture.

First, background information on the architecture design for *Next Generation Networks* (NGNs) and a *Future Media Internet* is presented. NGN architecture design heavily influenced the author's work on IPTV. Second, the general architectural approach for *Next Generation Networks* and ongoing work on the *Future Media Internet* is presented. Third, a description of how a novel *Interactive Application Environment*, combining session-oriented as well as declarative paradigms, has been designed for integration into the *Open IPTV Ecosystem Core* is provided.

4.1.1. Next Generation Networks Architectures

Functional architecture design was one of the major work items during the specification process for *Next Generation Networks* in Standard Development Organization as ITU-T (ITU-T Recommendation Y.2011[3]) and ETSI TISPAN (NGN Release 1[4]). Various other standards are built upon or reference these specifications, including work on IPTV as described in this thesis.

Figure 4.1 illustrates a simplified and schematic NGN architecture as specified in ITU-T's Y.2011 recommendation.

All services and functions are related to each other, since functions are used to build services. These services might include session-based services, such as Voice-over-IP or IPTV and also non-session related services. NGN functions are divided into *Service* and *Transport Stratum* and an *Application Layer* on top. So-called *End-User Functions* represent *End Devices* like telephones, computers, Set-Top-Boxes and TVs. More information on NGN architectures can be found in [76].

As becomes fairly clear, these high level architectural principles of *Next Generation Networks* do fit perfectly onto the requirements for an integration with IPTV services since IPTV can be treated like just another service within such an approach.

Furthermore, and with the introduction of some specific new components described later in this chapter, the *IPTV Role Model* from Chapter 2 can also be instantiated by choosing the NGN baseline architecture.

[3] http://www.itu.int/itudoc/itu-t/aap/sg13aap/history/y2001/index.html
[4] http://www.etsi.org/tispan/

For this reason, the use of an extended baseline NGN architecture for the composition of the *Open IPTV Ecosystem Core* functional architecture was decided upon.

4.1.2. Future Media Internet Paradigms

A missing key paradigm, namely a certain flexibility in and overlap between the the roles in the NGN model and as described for the *IPTV Role Model* in Chapter 2, has been derived from work on approaches for the *Future Media Internet*[5].

Here the term *tussle* describes the clash of interests between Internet stakeholders and especially the need for allowing users to choose Service and Content Providers in an instant manner [19]. In [78] Laso-Ballesteros et al. are stating that *"...NGNs do not seem to have been designed with tussle in mind. The roles of the network operators, content providers, and end users in NGNs are considered to be fairly static..."*

Taking this into account, the designed *Open IPTV Ecosystem Core* explicitly allows for the integration and connection of multiple Content and Service Providers, managed and Over-The-Top (OTT) approaches and the interaction between users across the boundaries of a certain service provider silos. This has been reached through the introduction of open APIs and a multi-protocol approach allowing for managed and Over-The-Top service creation and usage.

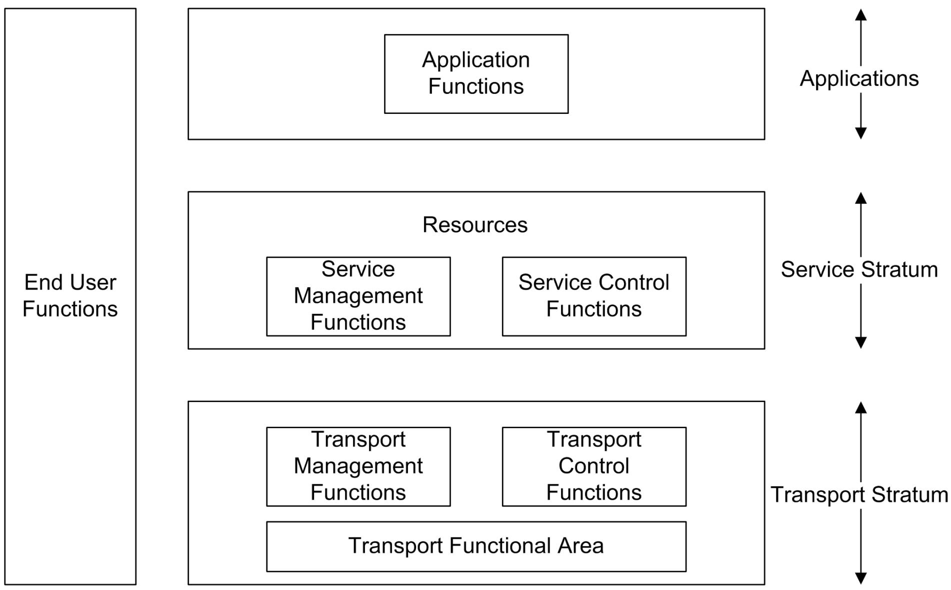

Figure 4.1.: *Open IPTV Ecosystem* Core; Functional Architecture

[5]http://www.future-internet.eu/publications/media.html

4.1.3. Interactive Application Environments

The NGN baseline architecture provides a session control core and introduces the flexibility of roles with approaches for the *Future Media Internet*, and allowed to form the basic framework for the *Open IPTV Ecosystem Core*.

In a next step, interactive applications and their current shape, denoted as *Social Media*, need to be enabled on the *Open IPTV Ecosystem Core*. For this reason, state-of-the-art on *Interactive Applications Environments* has been introduced in Chapter 2. As the outcome of the following analysis, two *Interactive Application Environments* will be chosen for integration into the OIEC.

4.1.3.1. Comparison of Interactive Application Environments

Procedural (PAE), *Declarative (DAE)* and *Session-Oriented Application Environments (SAE)* are compared in Table 4.1 with regard to their advantages when building State-Of-The-Art middleware for *IPTV*. This comparison has been derived from Chapter 2 and the author's research on these aspects.

Social TV Environments have been left out in this comparison, as they can be realized using either one or the other approach or a combination of the three. Development skills, either in a procedural or declarative language, do not necessarily represent a drawback for one or another approach. Nevertheless, Web applications are much easier to create, which might even enable non- or semi-professionals to contribute, e.g. to application stores known from the mobile domain.

Building *Session-Oriented Application Environments* requires a certain level of skill in a native language on protocol and signaling level, at least on the *End Device*. On the *Application Server*, a runtime environment often allows for an abstraction from the concrete implementation on protocol level. The complexity of the produced application does not vary when comparing PAEs or DAEs, as nearly every type of functionality can be integrated into both environments. Limitations exist when access to a certain hardware functionality is required. Especially browser-based environments like the DAE have some limitations here. Currently, projects like BONDI[6], JIL[7], WAC[8] or the W3C's DAP[9] group try to overcome this limitation through the creation of standardized JavaScript APIs that build a bridge between the browser and the native code and platform below. In SAEs, the SIP signaling between end device and the Application Server is the limiting factor when creating interactive applications. During the work on this thesis, complex scenarios have been created which require a huge amount of messages between the two endpoints. Nevertheless, the applications that successfully deal with voting, polls, targeted advertisements and quizzes can be realized with a SAE.

[6]BONDI Project: http://bondi.omtp.org/
[7]JIL Project: http://www.jil.org/
[8]Wholesale Application Community: http://www.wacapps.net/
[9]W3C Device APIs and Policy Working Group: http://www.w3.org/2009/dap/

Aspect	PAE	DAE	SAE
Required skills in a higher programming language	yes	no	yes
Required Skills in Web development & Scripting	no	yes	no
Potential Level of Application Complexity	high	high	low
Integration with Telecommunication Services	n/a	low	high
Richness of User Interfaces	low	high	n/a
In-Line with State-Of-The-Art Web technologies	low	high	medium

Table 4.1.: Comparison of Interactive Application Environments

Telecommunication services play a key role in the future environments for IPTV. Both the PAE and the DAE approach are still limited when integrating telephony, Instant Messaging or buddy list functionality. DAEs are currently making a huge step forward, but native protocol stacks and applications, as used for SAE implementations, are still more mature and reliable. Rich user interfaces are a key driver for a successful application. PAEs as MHP have suffered from their very limited graphical capabilities, whereas the Web including DAE is developing in the direction of *Rich Internet Applications* (RIA) allowing for animated 3D user interfaces and the integration of streaming media. SAEs are not directly affected, nor do they present any limitations, as user interfaces are mostly created as native applications or even in combination with Web and RIA technology. Finally, when looking at state-of-the-art Web technologies, it is again the DAE approach and the SAE that are part of current developments.

4.1.3.2. Interactive Applications in the Open IPTV Ecosystem Core

Based on the description and analysis above and the requirements from Chapter 3, *Declarative* as well as *Session-Oriented Application Environments* have some advantages when directly compared to the procedural approach.

The combination of the DAE and SAE approaches together with scenarios from *Social IPTV* research helps fulfill the user's functional and non-functional requirements for interactivity as well and are able to keep up with the development of the Web. For this reason, this section briefly describes an initial architectural sketch that incorporates SAE, DAE and Social IPTV approaches.

In more detail, and as depicted in Figure 4.2, the author has decided to compose a combination of *Social IPTV* with a session-oriented signaling layer that incorporates SIP communication services.

Besides that, the declarative approach has been chosen for the creation of user interfaces and also interactive applications, in part. This initial sketch will be used in the following sections for integration into the detailed architecture specifications allowing for the creation and execution of interactive services.

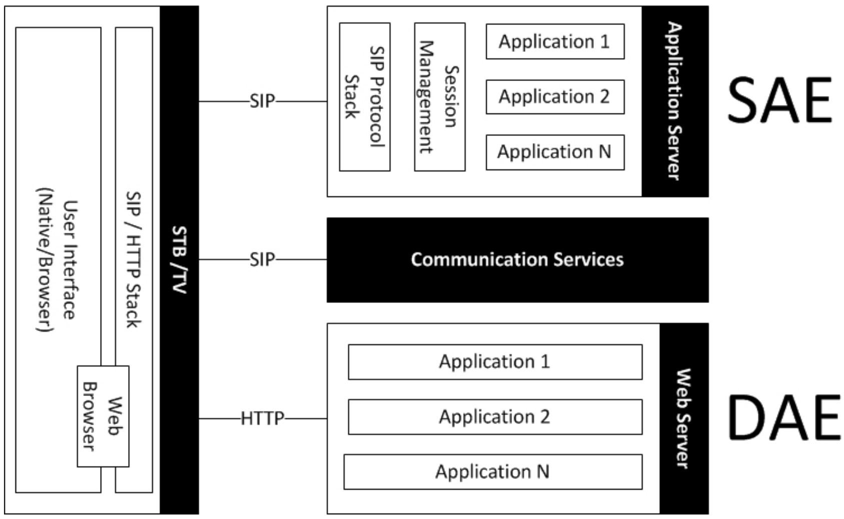

Figure 4.2.: Architectural Sketch for interactive IPTV

4.2. Open IPTV Ecosystem Core - Functional Architecture Composition

The *Open IPTV Ecosystem Core* functional architecture derived in this section has been composed based on the assumptions and decisions made in Chapters 2 & 3, the NGN reference model, approaches for a *Future Media Internet* and the analysis of *Interactive Application Environments* from the last section.

4.2.1. Architecture Composition

As depicted in Figure 4.3 the NGN Reference Model, work on a Future Media Internet, the session-oriented *IPTV Role Model*, requirements for interactive IPTV, as well as the combined session-oriented and declarative *Interactive Application Environment* have all been combined.

Table 4.2 provides details on each element used for the composition of the OIEC architecture. Furthermore, the reason for choosing each element is discussed and what kind of paradigm or functionality is brought in by the specific approach.

4.2.2. Core Building Blocks

The following building blocks for building the OIEC architecture have been identified. They furthermore correspond to the identified requirements for interactive IPTV from Chapter 3.

A distinction between components developed in the scope of this thesis (core functionalities) and other functions has been made:

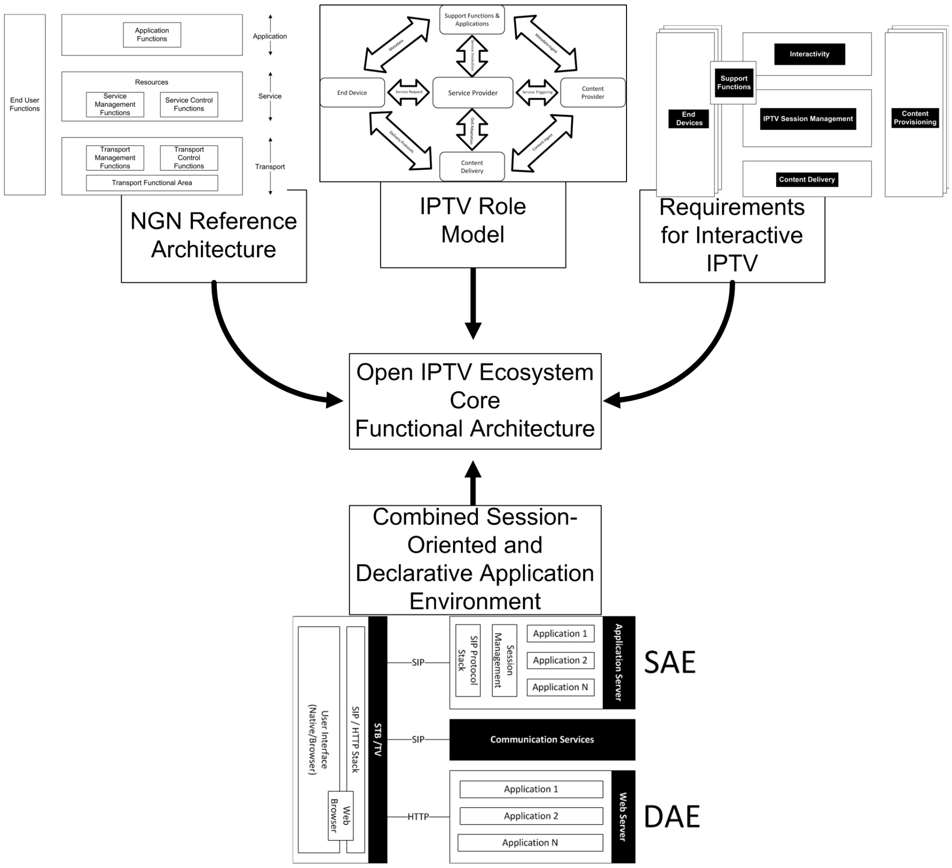

Figure 4.3.: Composition of the Open IPTV Ecosystem Core Functional Architecture

Composition Element	**Description**
State-of-the-Art on NGN and Future Media Internet paradigms	NGN and Future Media Internet paradigms provide the architectural baseline for the Open IPTV Ecosystem Core. This includes the layered NGN architecture principles and building blocks, as well as the more flexible roles of the involved entities from the Future Media Internet approach.
Session-Oriented IPTV Role Model	The Session-Oriented IPTV Role Model introduced in this thesis extends the basic NGN concepts for conversational services, by re-using its session concept for IPTV services as well. By adding a third party interface, the role model allows for the creation of interactive IPTV services.
Requirements for Interactive IPTV	Requirements for Interactive IPTV as derived in Chapter 3 allow for the identification of the necessary functional entities of the Open IPTV Ecosystems Functional Architecture. Furthermore, corresponding functional and non-functional requirements have been used to model the interaction between the architectural building blocks and later to validate the proposed architecture.
Integrated, Interactive Session-Oriented and Declarative Application Environments	The proposed combined session-oriented and declarative Interactive Application Environment allows for the creation of interactive services. The advantages of both worlds have been combined, allowing for session-oriented signaling and the usage of Web technologies and protocols to create interactive applications.

Table 4.2.: Composition of the Open IPTV Ecosystem Core Functional Architecture

Core Functionalities

- The *IPTV Session Management Enabler* maintaining all service requests and exposing them to other applications, if necessary. Furthermore, the *IPTV Meta Session Enabler*, providing *Social IPTV* functionality is also located here.
- The *Session-Oriented Application Environment Enabler* (SAE) is responsible for hosting session-oriented interactive applications.
- The *Declarative Application Environment Enabler* (DAE) is responsible for hosting browser-based applications.

Other Functions

- The *Interactive Content Enabler* (ICE) represents components responsible for interactive content delivery in the OIEC system. The ICE represents research activity conducted in parallel to the work on this thesis and has been re-used in the scope of several scenarios.
- The *End Device* serves the user by displaying and executing services.
- The *Support Functions* provide supplementary services like metadata, program guides, services to bootstrap the OIEC system.
- The *Content Providers* and *Broadcaster* ingest content and/or trigger services through the *IPTV Session Management Enabler*
- The *Communication Service Enabler* provides Rich Communication Services (RCS) as Voice-over-IP telephony, Instant Messaging (IM) and presence services.

4.2.3. Detailed Functional Architecture

In Chapter 2, the *IPTV Role Model* and three concepts for *Interactive Application Environments* were introduced. Two of them, namely the *Session-Oriented Application Environment* (SAE) and the *Declarative Application Environment* (DAE) were selected for integration into the OIEC functional architecture.

In addition to an *Interactive Application Environment*, responsible for the execution of interactive applications, various other functionalities must also be offered to the user.

The following overview lists all relevant architectural building blocks and describes their functionality, while Figure 4.4 provides a graphical representation of the building blocks and the corresponding interfaces between them.

Building blocks developed within the scope of this thesis have been highlighted. Relevant protocols are visualized through the different dashed lines.

Based on these high level descriptions, the following paragraphs will describe and analyze their detailed functionality and mutual interactions.

Table 4.3 again lists all components, and provides a short textual and interface description.

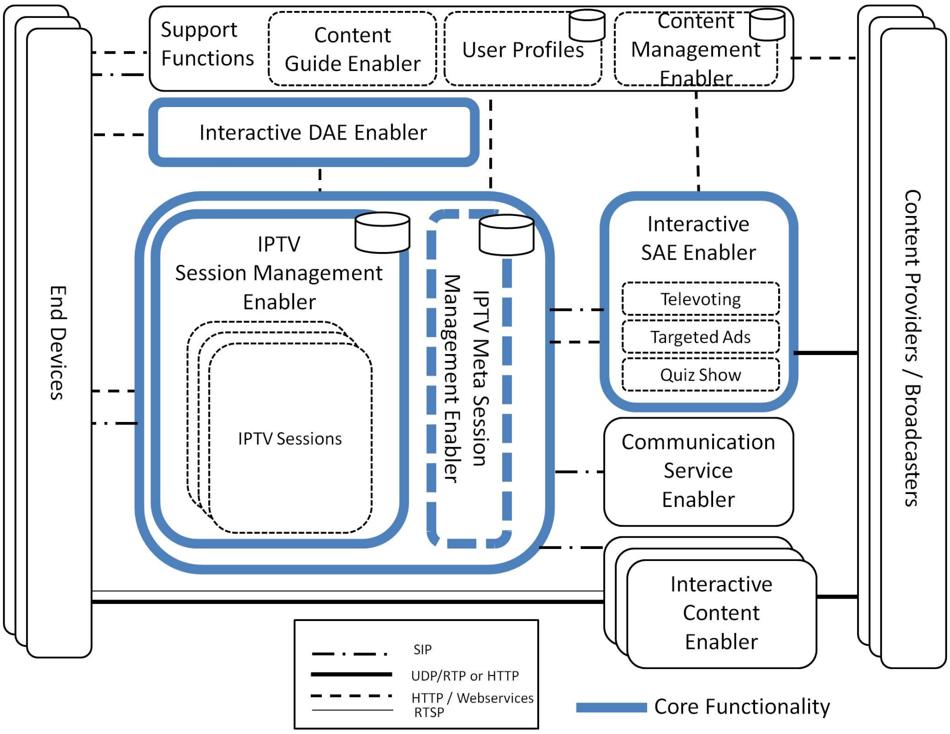

Figure 4.4.: *Open IPTV Ecosystem* Core; Functional Architecture

IPTV Session Management Enabler The *IPTV Session Management Enabler* represents the core building block for the described architecture. This entity is responsible for handling all incoming requests for content and interactive services requests issued by users.

First, a new session is generated for each request and then maintained within this entity. Second, the request is resolved by the *IPTV Session Management Enabler* to a serving *Interactive Content Enabler* for the delivery of the requested contents. Alternatively, an interactive application could also be triggered from the *Interactive SAE Enabler*.

An API allows for manipulation of the current session state in the *IPTV Session Management Enabler*. This can be used to allow SAE applications to access current session information. These concepts will be discussed again during specification of session-oriented streaming services in Section 5.1.

An additional building block is the *IPTV Meta Session Management Enabler*. *IPTV Meta Sessions* allow for the combination of individual sessions with merged Meta Sessions that contain multiple users and multiple contents. This is specifically discussed in Chapter 5.2. and enables the concept of Social IPTV.

Session-Oriented Application Environment Enabler The *Session-Oriented Application Environment Enabler* implements a *Session-Oriented Application Environment*, as described in Chapter 2. Specifically, this entity is composed as a converged SIP/HTTP Application Server, based on the JSR289 SIP Servlet[10] specification. This allows for the easy generation and deployment of new applications based on SIP Servlet technology. Three example applications designed and implemented on the architecture will be discussed in Chapter 5.4.

Declarative Application Environment Enabler The *Declarative Application Environment Enabler* provides server-side functionality for browser-based, interactive applications. From a technical perspective, the DAE acts as a Web server hosting applications, e.g. in the CE-HTML or HTML5 format to be rendered on the different *End Devices.*

Support Functions The *Support Functions* building block abstracts from certain baseline functionalities, which must be available in any kind of IPTV system:

Beginning with *Service Discovery* mechanisms, referred to as *Content Guide Enabler* in the architecture, a bootstrapping mechanism is provided. By making this information available, the STB or the user in front of it, respectively, are able to request information concerning available service offerings like Linear, Video-on-Demand or interactive services. Besides the *Content Guide*, a manageable *User Profile*, containing detailed subscription information for each user is part of this functional entity. The *Content Management Enabler* provides mechanisms allowing for content and metadata processing by *Content Providers* and *Broadcasters* through a dedicated interface.

Interactive Content Enabler The *Interactive Content Enabler* (ICE) represents the *Content Delivery Network* (CDN) infrastructure within the presented architecture. In general, this entity acts as an endpoint for all requests for content, forwarded by the *IPTV Session Management Enabler* and as a source for interactive contents streamed to the requesting users. Multiple ICEs could be deployed inside the architecture, e.g. fulfilling various purposes like basic Linear TV streaming, VoD streaming, acting as a networked Personal Video Recorder (nPVR) and serving as a source for interactive contents, e.g. during targeted advertisement scenarios.

[10]More details can be found at: http://jcp.org/en/jsr/detail?id=289

End Device The *End Device* functional building block represents a user's terminal in the form of a Set-Top-Box (STB) or TV. The *End Device* integrates the necessary protocol stacks for IP connectivity, as well as specific stacks for communication with the other building blocks of the system.

As depicted in Figure 4.4, this includes a HTTP and SIP stack, allowing for communication with *Support Functions* (Content Guide, User Profile, DAE). The *IPTV Session Management Enabler* and *Communication Service Enabler* are accessible via an interface using the SIP protocol, while the *Interactive Content Enabler* can be accessed using the Real Time Streaming Protocol (RTSP) for signaling and the Real Time Transport Protocol (RTP) for media transport.

Content Provider The *Content Provider/ Broadcaster* building block represents a placeholder for any type of content provider. This can be classical broadcasters that provide Linear TV services over IP multicast, providers of motion pictures, or even advertisers that feed their content into the system, i.e. into the *Content Management* (metadata) and the *Interactive Content Enabler* (files and streams). As *Content Providers* might also offer or control interactive applications, an interface for the SAE functional entity is also provided, allowing for the manipulation of ongoing sessions. This could be used e.g. for targeted advertisement services.

Communication Services Enabler The *Communication Services Enabler* fulfills the requirements for user-to-user and user-to-service communications. On a technical level, the *Communication Service Enabler* is realized as a combined SIP/XMPP presence and messaging server that allows for Instant Messaging, Buddy List and Rich Presence functionality.

Component / Entity	Description	Interface
IPTV Session Management Enabler	Managing user requests for content and services, Exposition of session data to third party	SIP, HTTP
IPTV Meta Session Management Enabler	Enabling Social IPTV scenarios	SIP, HTTP
Session-Oriented Application Environment Enabler	Runtime for session-oriented, interactive services	SIP
Declarative Application Environment Enabler	Runtime for browser-based interactive services	HTTP
Interactive Content Enabler (ICE)	Entity with Interactive Content Provisioning capabilities.	HTTP/SIP/ RTP/ RTSP
Support Functions	Collection of functionalities to bootstrap IPTV services	SIP/HTTP
Content Guide Enabler	Entity providing service description and metadata	HTTP
Content Management Enabler	Managing available content assets	HTTP
User Profiles	IPTV specific user data	SIP/HTTP
Content Providers	Placeholder for content providers and broadcasters	UDP, RTP, HTTP
Communication Services Enabler	Providing capabilities of a Rich Communication Server	SIP

Table 4.3.: Enablers & Entities of the Open IPTV Ecosystem Core

4.3. Summary

This chapter has introduced the high level architecture for the *Open IPTV Ecosystem Core (OIEC)*.

General architectural concepts for *Next Generation Networks* (NGNs), a *Future Media Internet* and *Interactive Applications Environments* have been described and mapped onto a new architecture fulfilling identified requirements from Chapter 3.

Furthermore, the main purpose and task of each functional building block has been analyzed.

In the following chapter, first the core functionality, namely the *IPTV Session Management Enabler* will be analyzed in more depth, describing its role and mutual interaction with other components enabling interactive services.

Furthermore, two different enablers for interactive and *Social IPTV* services will be specified, namely *IPTV Meta Session Management* and *Dynamic Content Provisioning*. Having this already available, three simple session-oriented interactive services will be specified.

5. Specification of the Open IPTV Ecosystem Core

Chapter 5:
Specification of the Open IPTV Ecosystem Core

IPTV Session Management
Meta Sessions for Social IPTV
Interactive Content Provisioning
Interactive Services

In the last chapter, the basic functional architecture for the *Open IPTV Ecosystem Core* was described. This chapter extends this description and specifies components developed in the course of this thesis or integrated as part of existing work. Figure 5.1 highlights the components that have been specified and will be described in Sections 5.1, 5.2, and 5.4, respectively:

Section 5.1 describes the design of the *IPTV Session Management Enabler*, specifies how session-oriented streaming services can be executed on the ecosystem, and how these services are signaled using the SIP protocol. Furthermore, *Third Party Openness* describes the concept of allowing third party access through a *Session State API* that has been designed to allow interactive applications to be integrated into the architecture.

These core functionalities are then used to specify mechanisms for *Social IPTV* called *IPTV Meta Session* in Section 5.2. This will be combined with the concepts for *Interactive Content Provisioning* in Section 5.3, representing the extensions for interactive functionality to the Open Source project VideoLAN. Finally, these specifications will be used to design interactive services in Section 5.4.

5.1. IPTV Session Management

The *IPTV Session Management Enabler* is a stateful entity that manages all user-to-content and content-to-user relationships. The building block presented is aware of which user consumes which content, and which content is consumed by which user on a particular *End Device.*

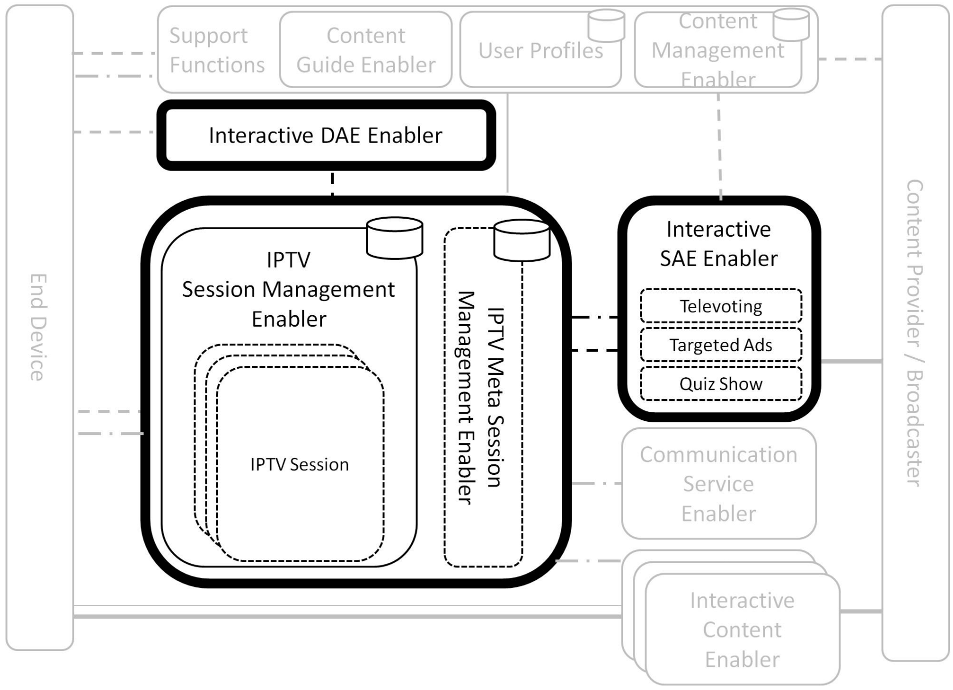

Figure 5.1.: Scope of Specifications conducted in this Thesis

Furthermore, the *IPTV Session Management Enabler* manages all signaling to the *Interactive Content Enablers*. The concept used in this context is borrowed from the concept of conversational services and services that have the ability to request and reserve transport resources and trigger interactive services. Therefore, it is possible to modify these parameters at a later stage (e.g. when zapping from SD channels to HD channels)[8].

In addition, the *IPTV Session Management Enabler* makes this information available to third parties through a well-defined API.

This functionality enables interactive and value-added services. Information about available content and access rights will be exchanged with the *Content Management Enabler*. Altogether, the *IPTV Session Management Enabler* fulfills the following tasks (see figure Figure 5.2 for a graphical representation):

- **IPTV Session Creation:** Instantiation of logical user to network connections based on content requests originating from a user's *End Device*.
- **Content Availability & Personalization :** Access to the *Content Management Enabler* and *User Profiles* allows for the resolution of content requests either directly or through forwarding these requests to the corresponding *Interactive Content Enabler*.
- **Maintain IPTV Session State (ISS):** Snapshots of the information related

to an ongoing *IPTV Session* are stored in the so-called *Session Repository* and are exposed through XML metadata and a well-defined API.

- **Trigger *Interactive Content Enabler (ICE)*:** Incoming requests will be forwarded to the designated *ICE* responsible for content delivery.

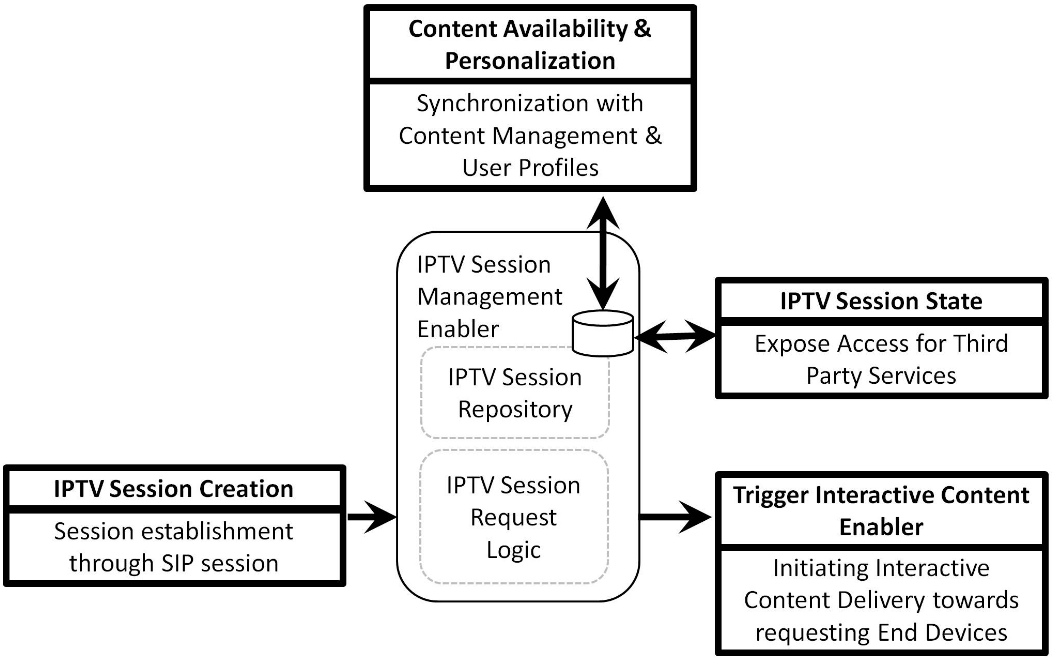

Figure 5.2.: Tasks for the IPTV Session Management Enabler

5.1.1. Executing IPTV Session Management

Following the above description, the practical instantiation of an *IPTV Session* involves the following functional building blocks:

- The *End Device*, e.g. a TV set or STB.
- The *IPTV Session Management Enabler*
- The *Interactive Content Enabler*

Figure 5.3 depicts their mutual interaction mapped onto the proposed architecture upon session creation:

The *End Device* initiates a request for content (Step #1), which is then processed by the *IPTV Session Management Enabler*(Step #2). Following that, the *Interactive Content Enabler* prepares the content (Step #3). In a last step, the requested content will be streamed to the *End Device*.

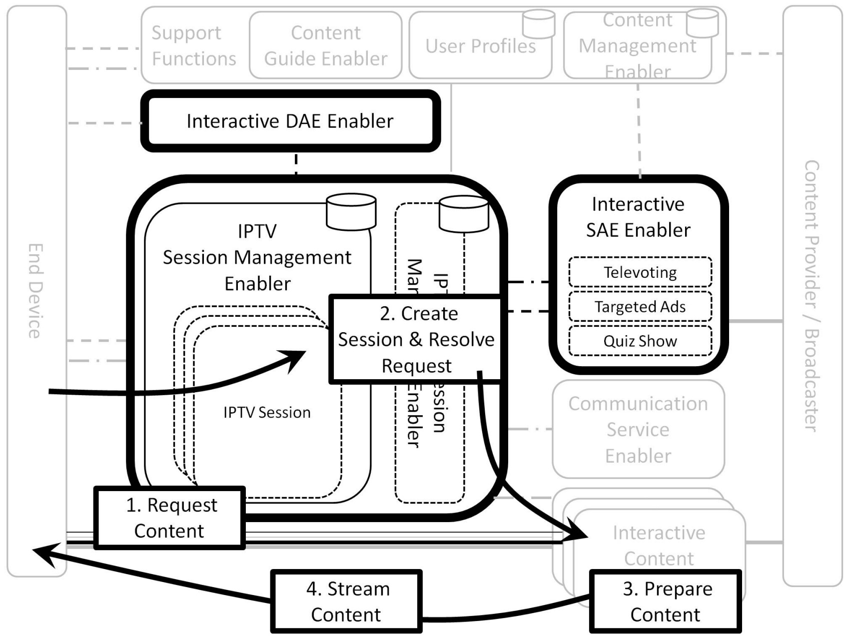

Figure 5.3.: IPTV Session instantiation (simplified)

5.1.2. IPTV Session Signaling Procedures

The last sections have described the session principles for *IPTV Sessions* in a high level format. This section provides a detailed description of the *IPTV Session* instantiation on protocol level for two specific services, namely Linear TV and Video-on-Demand (VoD).

5.1.2.1. Linear TV Session Setup Procedures

Before a TV or STB can join an IP multicast channel for Linear TV consumption, an *IPTV Session* has to be instantiated. The instantiation on a protocol level is visualized in Figure 5.4. In this case, session instantiation means that the TV requests access to a certain TV program. The necessary information like the channel identifier (CRID), i.e. the name of the TV channel, is assumed to already be available on the device and is selected by the user through his Electronic Program Guide (EPG). The following procedures describe a Linear TV session setup:

1. The user, in front of the *End Device*, selects a TV channel e.g. through the EPG and his remote control.

2. The session initiation request is then routed by the SIP proxy up to the *IPTV Session Management Enabler*.

3. The *IPTV Session Management Enabler* resolves the Content Resource Identifier (CRID) carried inside the request into an IP multicast address.

4. A response, including the TV channel's multicast address, is then sent back to the *End Device*

5. The *End Device* joins the corresponding multicast address and receives the TV channel as traffic either using the User Datagram Protocol (UDP) or the Real Time Protocol (RTP).

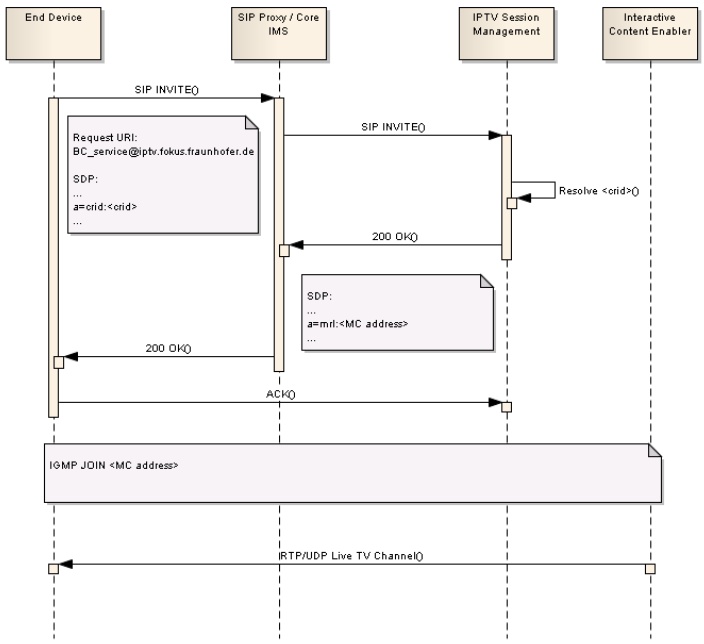

Figure 5.4.: Setting-up a Linear TV session

5.1.2.2. Video on Demand Session Setup Procedures

As described in the previous section, an *IPTV Session* has to be instantiated before any content consumption can take place. The signaling procedures for the VoD case differ from the Linear TV case, in making dynamic instead of static content addresses available. Thus the *IPTV Session Management Enabler* cannot directly resolve incoming requests, but rather must forward them to an appropriate *Interactive Content Enabler*. Figure 5.5 describes the setup of a Video-on-Demand (VoD) session and the necessary negotiations.

1. The user selects a content item from his listing on the *End Device*.

2. A session instantiation request is sent out by the *End Device* to the infrastructure.

3. The request is routed by a SIP proxy and the corresponding trigger point to the *IPTV Session Management Enabler*.

4. The *IPTV Session Management Enabler* analyzes the request. A direct resolution is impossible because VoD Uniform Resource Identifiers (URIs) have to be generated dynamically by the *Interactive Content Enabler*.

5. The *Interactive Content Enabler* resolves the CRID to a current file in his repository or database, and then generates a URI accessible through the Real Time Streaming Protocol (RTSP).

6. The RTSP URI is sent back within a 200 OK message as so-called Session Description Protocol information (SDP) to the *End Device*

7. The *End Device* initiates a RTSP session through a *RTSP DESCRIBE* (capability negotiation), *RTSP SETUP* (session initiation) and finally a *RTSP PLAY* that starts the streaming process on the server.

From this point on, trick functions (e.g. STOP, PAUSE, FAST FORWARD, FORWARD REWIND) are available through corresponding RTSP commands. The session will then be terminated by sending TEARDOWN messages to an RTSP, followed by a session initiation in reverse order.

Processes During VoD Session Setup A process-oriented view of the same procedure as above is depicted in Figure 5.6.:

A user selects content on the *End Device*, which results in a request being forwarded to the *IPTV Session Management Enabler*. Within this functional component, the incoming request is analyzed and, through reading the Content Resource Identifier (CRID), is mapped to a corresponding Interactive Content Enabler (ICE). While resolving the request, an *IPTV Session* is created in the *Session Repository*. The resolved request is then forwarded to an ICE, identified during the resolution phase. The ICE responds with a message containing a content URI. If necessary, the *IPTV Session Management* adds further information (e.g. additional information about interactive services) and finally forwards the message to the *End Device*.
At this point, the *IPTV Session* is established on the signaling level. In a last step, the *End Device* requests the contents directly from the ICE using the URI provided in the last message. The content is then streamed to the *End Device*. Alongside the setup of *IPTV Sessions*, the modification and termination is also possible through the *End Device*. A detailed description is not provided here as it is analogous to the process described above, just in a reverse order.

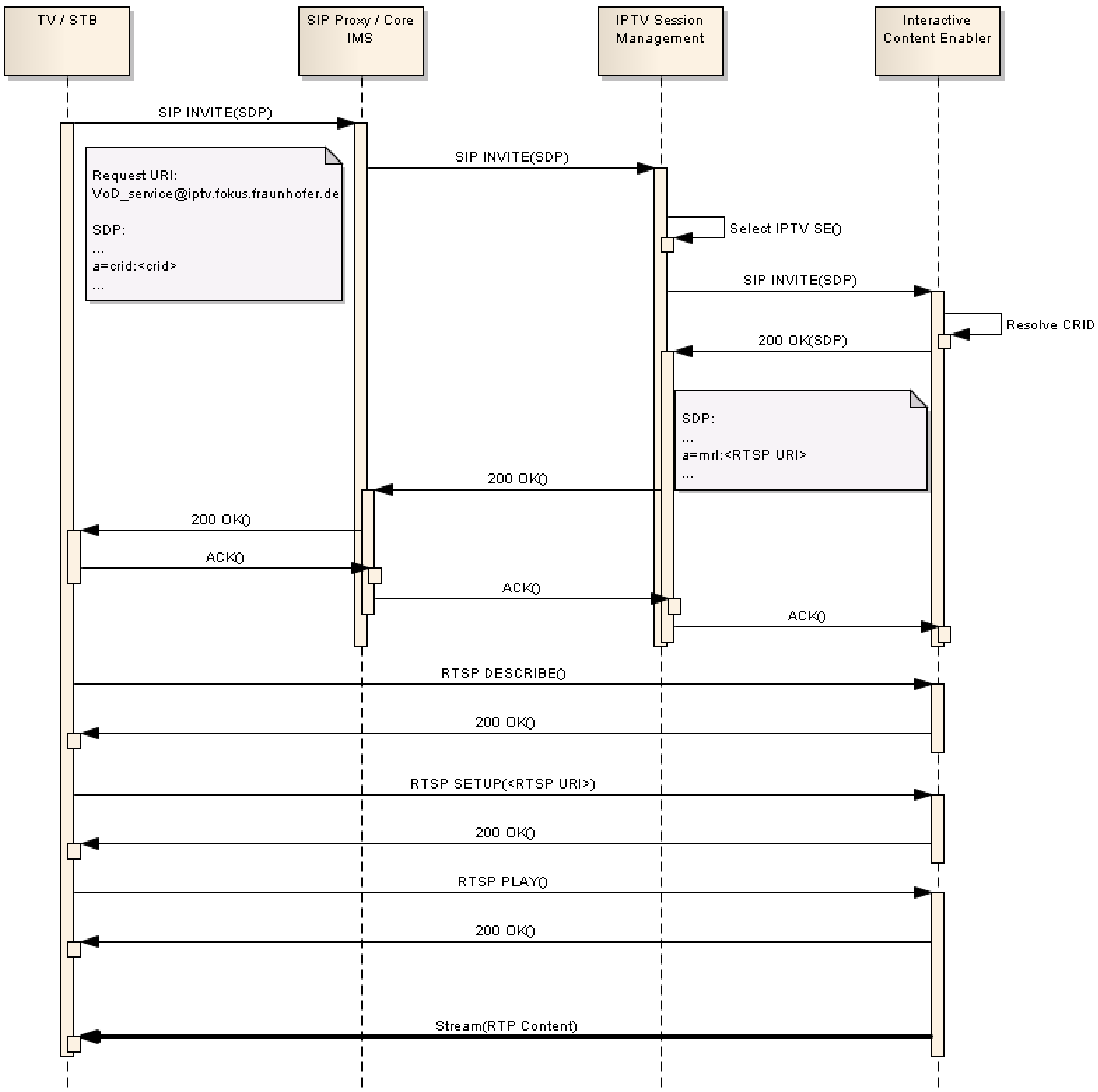

Figure 5.5.: Setting-up a VoD session

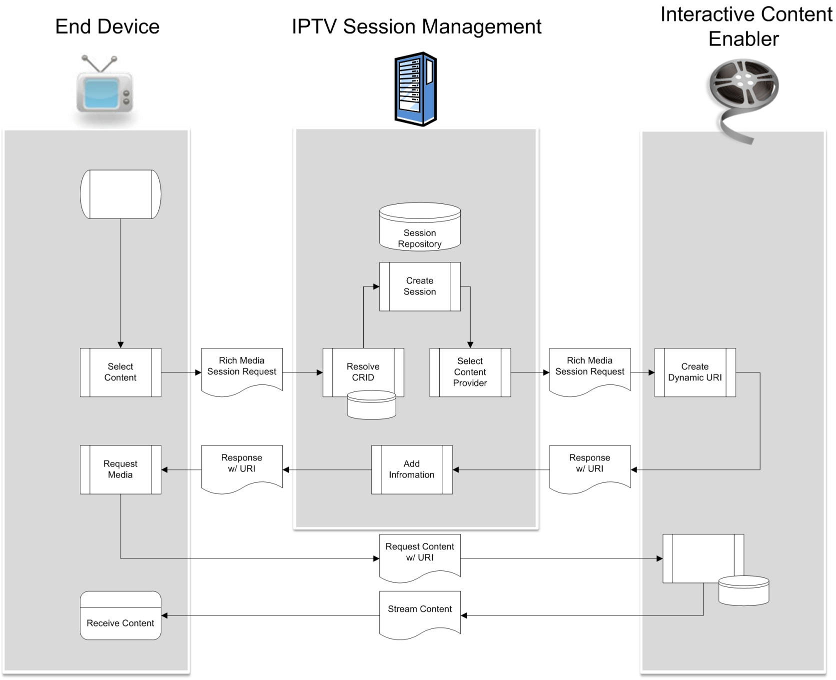

Figure 5.6.: IPTV Session instantiation (process view)

5.1.3. Low Level IPTV Session Management

This section provides a detailed view of the underlying infrastructure of the *IPTV Session Management* functional building block and outlines how and where *IPTV Sessions* are instantiated and how the application logic executes requests for content. A detailed architectural overview is depicted in Figure 5.7.

The *IPTV Session Management Enabler* consists of the *Media Request Logic*, which implements abstract functions that allow for the processing of incoming service requests. For example, these abstract functions allow for the creation, modification and termination of *IPTV Sessions* as described in Tables 5.1, 5.2 and 5.3, respectively.

These abstract functions will be used again in Section 5.1.4 when describing third party access mechanisms allowing for the manipulation of ongoing sessions. An additional functionality that resides on top of the *IPTV Session Management Enabler* allows access to the *Content Management* and *User Profiles* functional components as part of the *Support Functions*.

These functionalities allow for both synchronization with the content currently available, and the authorization upon service request. In addition to the service logic for controlling *IPTV Sessions*, a *Session Repository* also maintains sessions inside a database. Below the *Core IPTV Session Management Enabler*, the so-called *Media Signaling Protocol Layer* abstracts various protocols that can be used for session setup.

At the current point in time, this includes SIP, and for further consideration, skeletons for HTTP, Web services and JSON. On top of the core *IPTV Session Management*, a *Configuration & Management* interface allows browser-based access to the main functionalities, allowing for maintenance and setup. The *Third Party Access* functionality will be described in the next section.

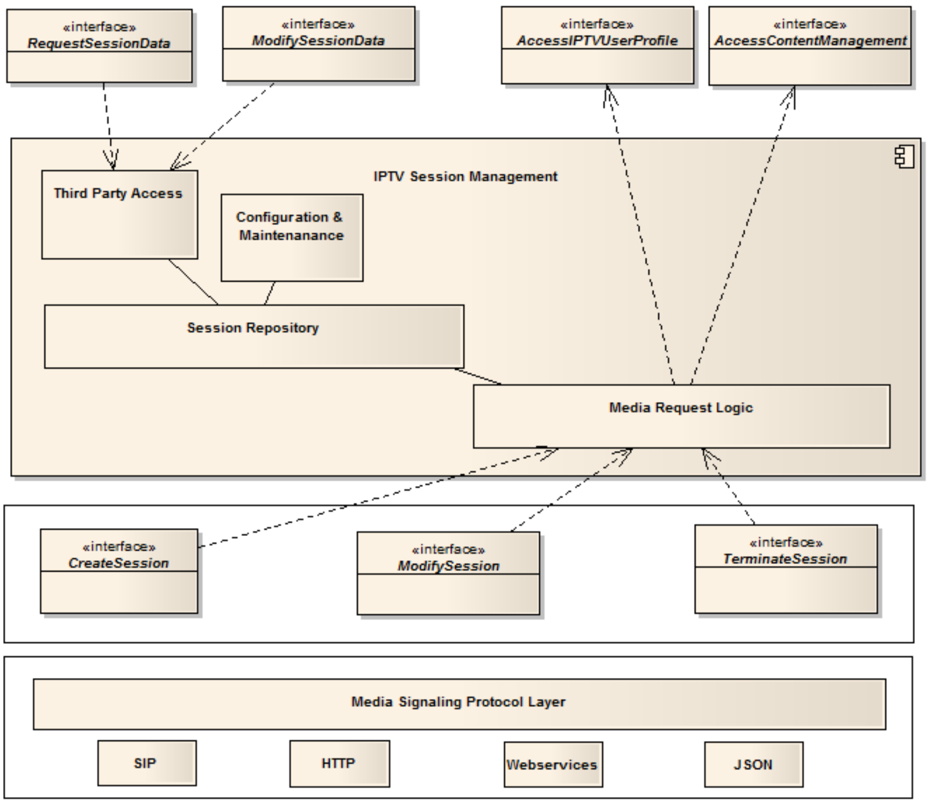

Figure 5.7.: IPTV Session Management Enabler architecture

5.1.4. Enabling Interactivity - Third Party Access API

The procedures for *IPTV Session Management* described in the last section allow the system to create *IPTV Sessions* based on a user's request for a certain IPTV service.

Going one step further, exposing the information about available sessions to third party applications, or even allowing them to manipulate this information, allows for completely new service models, including interactive services.

Method	CreateIPTVSession
Describes	Generates a SessionID and creates a session according to the delivered data. Forwards the request to the ICE (Video on Demand) or resolves the Multicast address for a Live TV channel.
Argument	Session Description, CRID or Service URI: ID to identify the content
Returns	Returns the RTSP URI or Multicast address of a streaming service.

Table 5.1.: Create IPTV Session

Method	ModifyIPTVSession
Description	Allows for the modification of session parameters. This includes adaptations to the required bandwidth or used codec.
Argument	updated Session Description (SDP)
Returns	Acknowledgement

Table 5.2.: Modify IPTV Session

Method	TerminateIPTVSession
Description	Terminates a session on the IPTV Session Management Enabler, forwards the termination requests towards the ICE and initiates a notification towards the TV
Argument	Session ID
Returns	Acknowledgement

Table 5.3.: Terminate IPTV Session

In particular, this enables scenarios, currently well known as so-called *Red Button* services, allowing a broadcaster or service provider to enrich the current plain consumption of TV or streaming video with supplementary services. This might include program-related information, voting or user polls or other interactive scenarios linked to the content.

This section describes how the *IPTV Session Management Enabler* exposes available *IPTV Session* information on a technical level.

Two key concepts presented here are the novel approaches to the integration of *IPTV Meta Sessions* (Chapter 5.2), as well as the specification of interactive applications, executed on the *Session Oriented Application Environment* (SAE) (Chapter 5.4). Both concepts require the *IPTV Session Management Enabler* for the exposure of information about active sessions through a dedicated API.

During the work on this thesis, different mechanisms for the exposure of session related information have been taken under analysis. Two of them will be described within this section:

5.1.4.1. Third Party Access: IPTV Session State

In Chapter 2, the *IPTV Session Context* was introduced as

"stateful information about a session created and used by different actors in a session-oriented system"

IPTV Sessions are maintained inside the *IPTV Session Management Enabler* and can be exposed to third parties through *IPTV Session State* (ISS) information. The *ISS* is dynamically composed and updated through current *IPTV Sessions* maintained by the *IPTV Session Management Enabler*. In the current implementation, the *IPTV Session State* is written into an XML schema and made available through a dedicated URL. The following data model is used to expose session information:

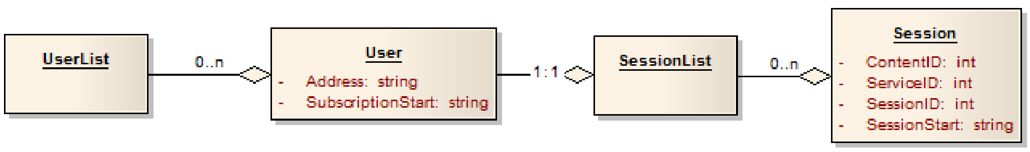

Figure 5.8.: Data model IPTV Session State

As depicted in Figure 5.8 the *IPTV Session State* allows for the listing of the registered users. This includes:

- The *End Device*' address containing its SIP URI.

- The start time of the users subscription (i.e. *IPTV Session* instantiation), containing the time stamp for when the user turned on his *End Device* and subscribed to a service.

- A list of ongoing *IPTV Sessions* including a:

  - ContentID referencing the currently consumed content.

  - ServiceID describing the type of content

  - the associated SIP SessionID

  - the session start time

An example of a current *IPTV Session State* is shown in Listing 5.1 below:

Listing 5.1: Example for a user's IPTV Session State

```
<<UserList>
   ...
   <User>
      <Address>sip:bob@milab.fokus.fraunhofer.de</Address>
      <SessionList>
         <Session>
            <SessionId>1261923923115</SessionId>
            <ContentId>23</ContentId>
            <ServiceId>3</ServiceId>
            <SessionStart>2009-12-27 15:25:23</SessionStart>
         </Session>
      </SessionList>
      <SubscriptionStart>2009-12-27 15:25:11</SubscriptionStart>
   </User>
   ...
</UserList>
```

5.1.4.2. Third Party Access: Session State API

The relatively straight-forward concept offered in the *IPTV Session State* through an XML document, has some disadvantages, especially when building a scalable service and managing a growing number of subscribers. Moreover, new requirements arise when client or server side functions should be allowed to modify session-related data or create new sessions on the *IPTV Session Management Enabler*. For this reason, it has been decided to define an API that can be made accessible to any kind of third party service through different protocols. At the time this thesis was written, the current implementation of this API was carried out in a diploma thesis that contributes to the author's work. Due to this parallel activity, only the abstract functions will be presented here.

Abstract ISS API Functions Tables 5.4 and 5.5 describe the currently available API calls for network-oriented components, namely:

- `GetSessionData()`
- `ModifySessionData()`

The API allows for the reading out and manipulation of active *IPTV Sessions*.

Method	GetSessionData
Description	Function that returns session related information. This function is synchronized with the IPTV Session State and can be used to provide subsets of information from the IPTV Session State
Argument	UserID, GroupID, ChannelID
Returns	Active session metadata

Table 5.4.: Get Sessions Data

Method	ModifySessionData
Description	Allows third-party access to active sessions. Results in modifying a session with respect to codec and bitrate
Argument	UserID, GroupID
Returns	Acknowledgement upon session modification

Table 5.5.: Modify Session Data

5.1.5. Summary

This section has introduced details of a session-oriented IPTV architecture, namely *Open IPTV Ecosystem Core*.

An outline for how *IPTV Sessions* can be executed on the proposed architecture was delineated. Detailed signaling flows for Linear TV as well as Video-on-Demand sessions were then presented. Finally, two novel mechanisms for access to *IPTV Session* information through third parties were discussed. The author hereby contributes to the idea of open APIs for IPTV systems, as described in the State-of-the-Art section of this thesis. These mechanisms will be used in the following chapters to create multi-user and multi-content scenarios enabling Social IPTV, as well as different session-oriented interactive services.

5.2. Social IPTV: Meta Sessions

In this section, the session principles for IPTV will be extended towards *IPTV Meta Sessions.* Therefore, the defined single user to service relationship will be extended towards a multi-user, multi-content environment.

The idea behind the so-called *IPTV Meta Session* is to bring together both: the idea of multimedia communications between multiple users on the one hand, and on the other the usage of multiple rich media contents like TV channels, videos and pictures under the umbrella of one shared *IPTV Meta Session.* Additionally, information on how to synchronize consumption as well as layout information could also be included. Altogether, a *IPTV Meta Session* is responsible for the maintenance of the persistence of the use case and allowing participating users to manipulate this session by adding and removing contents.

The foundation for this work was developed during the European Commission's Framework Programme 7 project iNEM4U. Corresponding concepts have been developed by the author and were part of different deliverables [53, 50] and scientific papers [55, 62, 63] created during the lifetime of the project.

5.2.1. Introduction

The goal of the *IPTV Meta Session* and the corresponding functional building block, namely the *IPTV Meta Session Management Enabler* serving as the manager of these kind of sessions, is the delivery of shared user experiences. This shared user-experience involves providing a variety of multimedia services to end users that are connected from different locations, devices and networks.

These experiences may be comprised of several types of multimedia contents, e.g. broadcast, IPTV, Video on Demand, video conferencing, chatting or voice calls and also interactive multimedia services.

Users expect to consume all these services and combinations of them together and to share the impression of being at the same location even though they may be miles apart.

The proposed *IPTV Meta Session* concept is an environment where several network domains are connected, and where users want to consume services and share content from any of these domains. The domains include home networks, DTV broadcaster networks, broadband access networks, telecom service provider networks, 3G / mobile networks, managed IPTV networks and the Internet.

The *IPTV Meta Session* concept provides a logical representation of interactive multimedia sessions that span multiple domains. An *IPTV Meta Session* contains all the information that is required by a client to connect to a cross-domain session or to replay an archived session. It includes information about the services and content items that are consumed, as well as information about the users who participate in the session. For example, if a session between two users consists of a Video on Demand and a Voice over IP session, the *IPTV Meta Session* would describe the two media-level sessions (VoD session and VoIP session) and the involved users. An *IPTV Meta Session* can furthermore contain additional metadata information that may be required for synchronization of media streams across domains and additional layout information to create a similar experience across domains. The *IPTV Meta Session* description itself is independent of any given domain. The actual delivery of the content is achieved within each domain by *IPTV Meta Session* enabled *End Devices* using domain-specific technology (e.g. SIP/IMS, CE-HTML or Webservices) to establish connections to the content sources described in the *IPTV Meta Session* description.

In the *IPTV Meta Session* model, a so-called *IPTV Meta Session Service Provider* entity is responsible for hosting and administrating *Meta Session* information. The system that administers *IPTV Meta Session* information will typically be hosted like a network service, the only constraint being that the *IPTV Meta Session* Service Provider is accessible from all network domains. It is likely that it is hosted by an IPTV Service Provider in addition to his common IPTV services, but it can also be provided by an independent Over-The-Top (OTT) service provider hosting services and enriching existing IPTV services provided by others.

5.2.1.1. IPTV Meta Sessions as Part of the Open IPTV Ecosystem Core

As stated previously, the work on the *IPTV Meta Session* model has been accompanied by different international research activities. Work on the *Meta Session* model has mainly been contributed to the project European Research Project iNEM4U, where the session-oriented *IPTV Role Model* has been used to connect the different partners' developments into a unified infrastructure.

As depicted in Figure 5.9, the realized infrastructure fits perfectly into the designed architecture for the *Open IPTV Ecosystem Core*:

Users sitting on top of their formerly siloed entertainment and communication infrastructures incapable interacting with each other are enabled to interact through a so-called *Shared Experience*. This means that they are able to share, consume, interact and communicate with each other without an awareness of the technology used.

The composed iNEM4U Platform (Figure 5.10) integrates the *IPTV Meta Session Model* (called iSession in the context of the project) and defines various use-cases and scenarios on top.

5.2.2. IPTV Meta Session Management

This section provides an overview of *IPTV Meta Session* management. *IPTV Meta Session Management* is done on an abstract level, irrespective of the underlying content delivery technologies. A *IPTV Meta Session* contains a high-level description of the media contents of and participants in the session. The description of delivery specific information like content encoding and transport is completed one

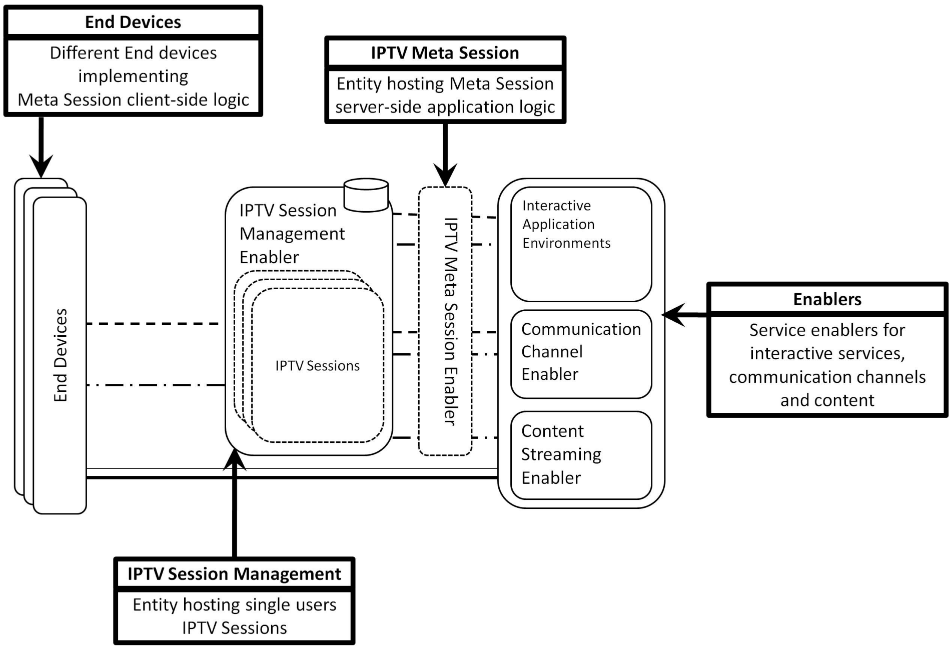

Figure 5.9.: Meta Session Management Enabler as part of the session-oriented IPTV architecture

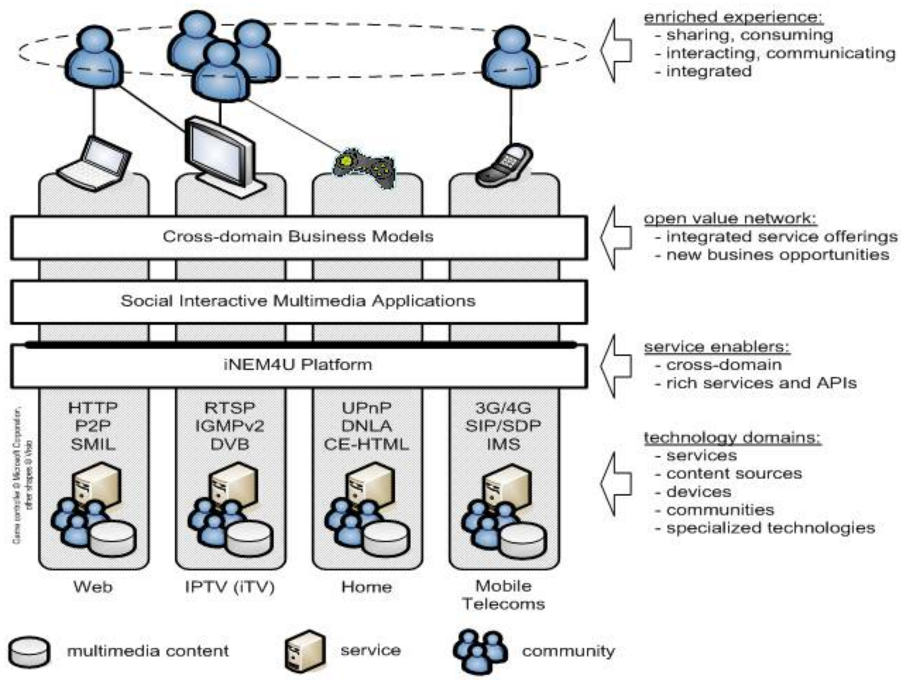

Figure 5.10.: IPTV Meta Session Domains

step earlier at the *IPTV Session* level as described earlier in this chapter. On the *IPTV Meta Session* level, content (e.g. personal or public content) is only given by a unique *Content Reference Identifier* (CRID) based on the TVAnytime specification [32] or URL which allows the client to retrieve more detailed information from the *Content Guide Enabler*.

Contents can be accessible over the Internet (e.g. Web TV, YouTube) or locally accessible, e.g. as managed content from an operator network (managed IPTV or Digital TV Video Broadcast). Contents can furthermore be *User Generated Content* (UGC) like live Web cam or in-home media servers. An *IPTV Meta Session* is managed by a so-called *IPTV Meta Session Service Provider*. A *IPTV Meta Session* Service Provider has to be located on the Internet to be accessible for participants but may also, in parallel, be part of a managed network e.g. acting like an Application Server in IMS environments.

5.2.2.1. IPTV Meta Session Classification

Two basic *IPTV Meta Session* classes can be distinguished:

- User-generated *IPTV Meta Sessions*
- Service Provider-generated *IPTV Meta Session*

In the first case, a Service Provider gives users the possibility to create an *IPTV Meta Session* themselves and (possibly) share this created session with other users. In the second case, the *IPTV Meta Session* is created by the Service Provider and users are able to join such a session. Users can join a session when invited by other users, by receiving a notification caused by a subscription to specific types of *IPTV Meta Sessions*, or by selecting a *IPTV Meta Session* from an EPG containing *IPTV Meta Session* information or by downloading it from a Web server. An overview of the basic *IPTV Meta Session Management Architecture* is given in Figure 5.11.

It shows the distinction between the client with integrated application logic, and a potential service provider with an *IPTV Meta Session* server entity. As depicted, the content source is independent of the provider of the *IPTV Meta Session Management.*

The following section describes the structure of the metadata used to describe an *IPTV Meta Session.*

5.2.2.2. Session Description

A *IPTV Meta Session* is described by a so-called *IPTV Meta Session Description* (IMSD) and transferred between the network infrastructure and the various clients as payload inside one of the supported protocols.

It contains basic information about the content associated with an *IPTV Meta Session*, the participants in the *IPTV Meta Session*, and layout and timing information needed to render the session at the client side. An *IPTV Meta Session* does

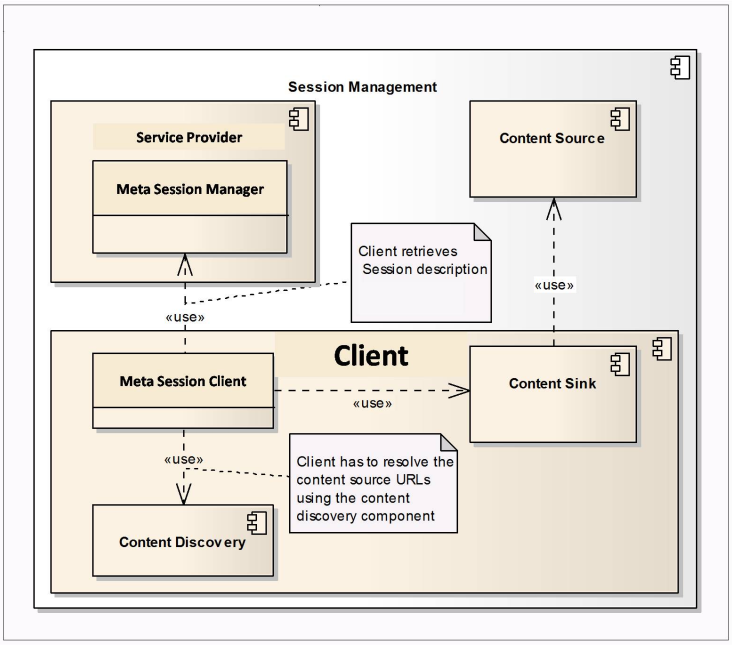

Figure 5.11.: IPTV Meta Session Architecture

not contain media specific information like media format or media delivery type. This information is negotiated between the client creating or receiving a IMSD and the content sources providing the actual content of the *IPTV Meta Session*.

The IMSD description contains the following elements as visualized in Figure 5.12:

- A unique *IPTV Meta Session ID*
- Human-readable name or description
- List of content items in the *IPTV Meta Session* given by unique Content IDs. Content IDs can be TV-Anytime CRIDs, URLs of public resources, URLs for User Generated Content or an *IPTV Meta Session ID* (for hierarchical *IPTV Meta Sessions*)
- A list of User IDs (possibly pseudonyms)
- Mapping of Content IDs to Users (e.g. which user consumes which contents)
- High-level description of a media resource
- Media type like audio, video, picture, Web page, etc. live-streamed, streamed, downloaded
- Access control list for session modification, i.e. a list of users which are allowed to modify a session

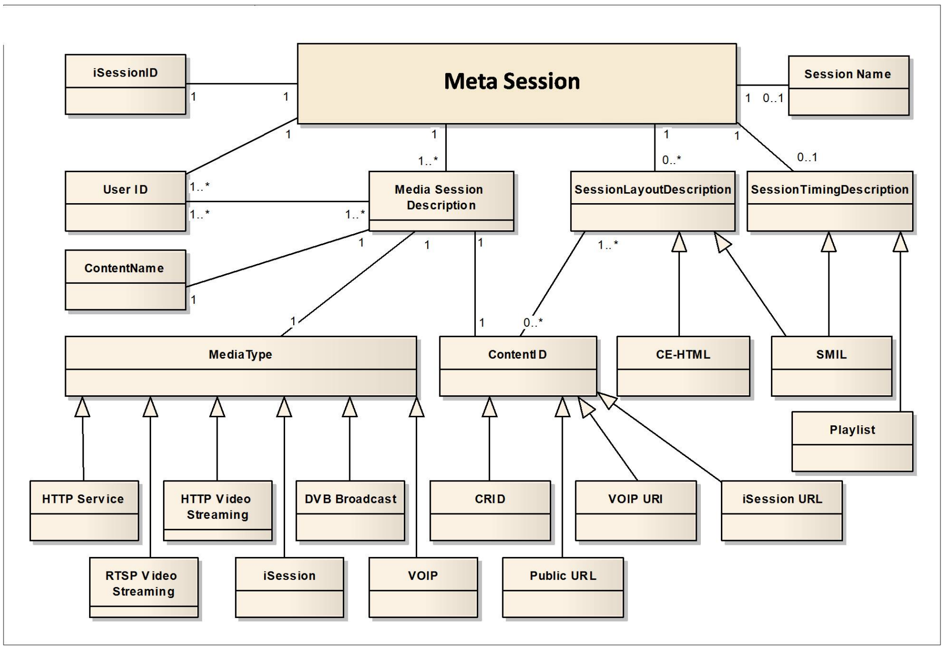

Figure 5.12.: IPTV Meta Session Composite Structure

- Timing information or playlist. This may be necessary if a session can be recorded
- Layout information (or a URL providing the information, e.g. CE-HTML)

Hierarchical IPTV Meta Sessions An *IPTV Meta Session* may contain a reference to an embedded *IPTV Meta Session* in its content list. This allows for the creation of hierarchical *IPTV Sessions*. Thus, for example a *IPTV Meta Session* can describe a channel containing multiple media resources defined and published by a Service Provider. If two users decide to consume this *IPTV Meta Session* together privately adding other media sources, other users of the channels should not be included anymore. In that case, the two users can create a private *IPTV Meta Session* containing the public *IPTV Meta Session* plus the private content sources.

IPTV Meta Session Documents The *IPTV Session Description* document is used to set up an *IPTV Session* at an corresponding client. The IMSD can be split over multiple documents linked together using document URLs. The following logical sub-documents can be identified:

- User document: contains the list of user IDs, a link to the session description document and the mapping of user ID and content. Additionally, it may contain access control information

- Session Description document contains a list of (high-level) media descriptions and may link to additional layout and timing documents
- Layout document
- Timing document

By making available the concepts and necessary metadata specifications for the creation of a *IPTV Meta Session*, the next section will describe the life-cycle of a *IPTV Meta Session*.

5.2.3. Meta Session Life-Cycle

The life-cycle of a *IPTV Meta Session* is depicted in Figure 5.13. Session creation can either be performed on the client side (user-generated *IPTV Meta Session*) or on the Service Provider side.

During the creation phase, the creator can add media resources and *Users* to a session and is able to define the layout and timing of the content within the session. A description of an *IPTV Meta Session* can then be downloaded to the client device that initialized the session.

During initialization, current media resources have to be detected (if not given by a globally unique URI). The client may not be able to find an appropriate media resource because there might be none that match the device's capabilities. In such a case, the *IPTV Meta Session* may be modified.

Modification might only affect one user (who can then simply disable that media resource) or all users, in which case the media resource is completely removed from the session. The modifications and the completed initialization are signaled back to the server.

After that, the server is able to indicate to all involved clients that they should start the actual rendering of the session. Subject to access control, a running *IPTV Meta Session* can still be modified by clients and the Service Provider.

This includes addition and removal of users, changing layout and also timing of a session. Furthermore, an active session can be stopped by a user or the Service Provider, which causes all involved ends to stop the session.

If not stored, a stopped session can be directly destroyed, i.e. all resources dedicated to a session can be freed. Running sessions can be resumed and paused, which will trigger the resuming and pausing of all media resources. Live content (whether broadcast or *User Generated Content*) might not be paused at all ends and thus will not be affected by the pausing of the session. If all clients have PVR functionality, pausing of live content can be possible, but it is not reasonable to assume that level of functionality for all types of live content and clients.

Paused and stopped sessions can be stored and loaded at a later time. The actual meaning of storing a session is not defined at this point and leaves space for later research. As it may involve the storing of all content sources of a session this will often not be feasible. A simple solution to the problem of storing an *IPTV Meta Session* is to store the IMSD only.

In this case, media resources are just stored by reference and, if they disappear (e.g. their URI changes or the source is offline), the session cannot be played back. This is especially problematic for live content. The extreme opposite of this approach is a complete recording of the session and all its involved media resources in order to replay the whole session later.

This involves the storage of all content which was shown during a session (also live broadcast and video chat), as well as timed relationships if the session was been modified during runtime (e.g. a content item has been added at a given time after the start of the session).

5.2.4. Meta Session Manager Enabler

The *IPTV Meta Session Manager* (IMSM) is the core component for the creation, updating and handling *IPTV Meta Sessions* and their comprising metadata. To fulfill the requirements that the server is supposed to support a n interoperability between a variety of clients, the server needs to include a multi protocol stack.

This includes SIP for communication with NGN/IMS-enabled clients, HTTP/-SOAP for communication with CE-HTML-enabled clients and the XMPP protocol for a general notification process. For the implementation of the different functions, the IMSM consists of four layers, the Meta Session Interface Layer, the Meta Session Logic, Meta Session Repository and a Notification component. An overview of the *IPTV Session Management Enabler* is shown in Figure 5.14.

5.2.4.1. Meta Session Manager API

Three client interfaces have been designed within the *IPTV Meta Session Manager* for client communication. This includes a SOAP (Webservices) interface, a HTTP interface and a SIP interface. These different interfaces allow a variety of clients to access the different functions of the IMSM. In the actual implementation of the IMSM, the specific protocol-related implementations are mapped corresponding to abstract API calls inside the underlying IMSM functionality.

This offers two main advantages:

- the business logic is the same for all interfaces and
- it is easy to add other interfaces as well, whenever new technologies arise. The different functions currently supported by the API are listed in Tables 5.6-5.17. During work on this thesis, the API was still under development and additional API calls, e.g. for synchronization issues were due to be added.

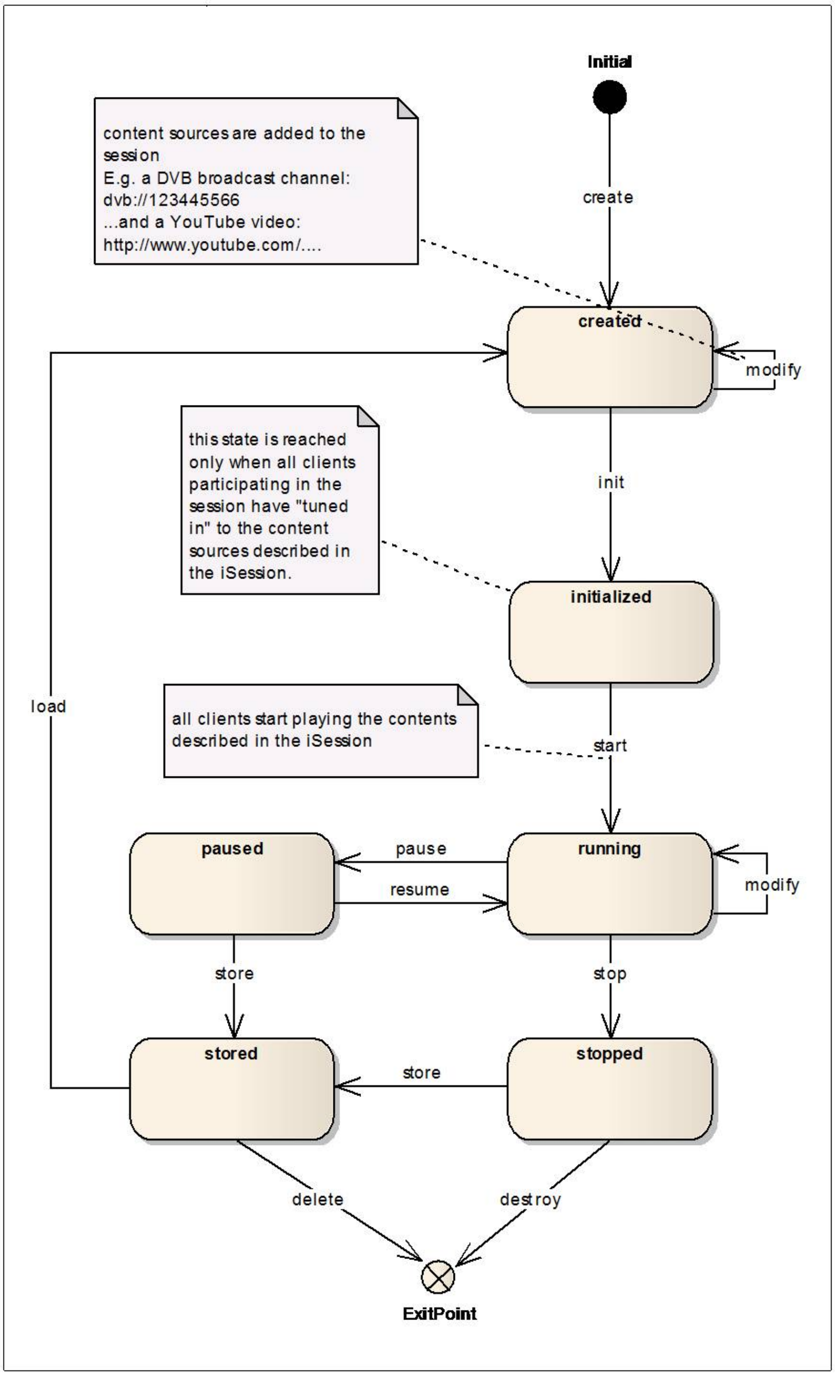

Figure 5.13.: IPTV Meta Session Lifecycle

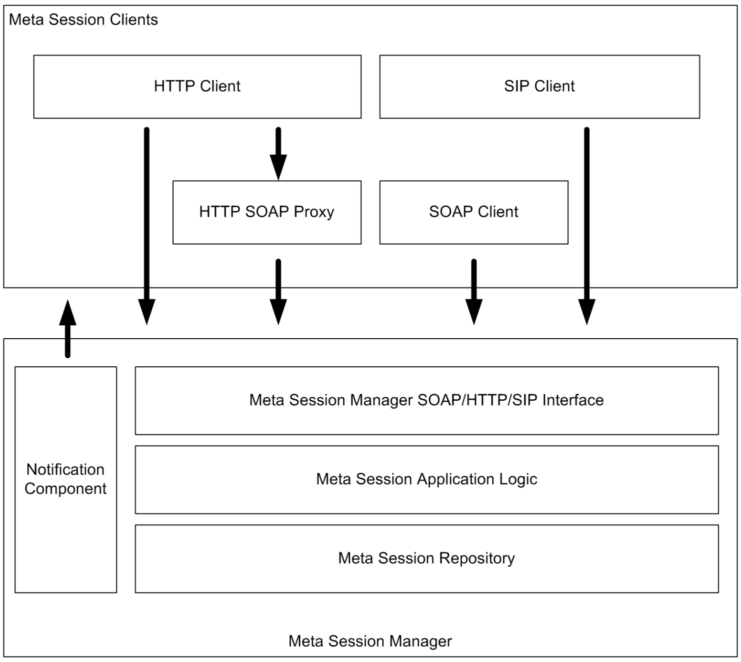

Figure 5.14.: IPTV Meta Session Manager and Interfaces

Method	**CreateIPTVMetaSession**
Describes	Generates a SessionID and creates a session according to the delivered data.
Argument	UserID: The User ID of the session owner. tva: URI to the associated media of the session. Description: A name or description of the session.
Returns	Returns the Session Description XML of the new session.

Table 5.6.: Create Meta Session

Method	**AddUser**
Describes	Adds the user to the session and notifies all other user in the session.
Argument	Session: Session ID of the session the user needs to be added. UserID: The User ID of the user to add. NotificationType: Additional Information for the Notification.
Returns	Returns the Session Description XML of the new session.

Table 5.7.: Add User to Meta Session

Method	**RemoveUser**
Describes	Removes the User from the session and notifies all other User in the Session. This also involves deleting the content that the user owns.
Argument	Session: Session ID of the session the user needs to be removed. UserID: The User ID of the user to be removed.
Returns	Returns the iSession Description XML of the modified session.

Table 5.8.: Remove User From Meta Session

Method	**ChangeState**
Describes	Changes the state of the Session.
Argument	Session: Session ID of the session the state needs to be changed. State: The new state of the Session. (can be: CREATED, RUNNING, STOPPED, PAUSED)
Returns	Returns the Session Description XML of the modified session.

Table 5.9.: Change the State of a Meta Session

Method	**AddContent**
Describes	Creates and adds the content into the session. Also notifies all users in the session (or excluding the given User of the UserID). Additionally, it is possible to add content with a UserID (of the content owner) and a description.
Argument	Session: SessionID of the session the content needs to be added. URI: Link to the content. MediaType: Type of the media (e.g. audio, video, image). UserID: The User ID of the content owner. Description: A name or description of the content.
Returns	Returns the Session Description XML of the modified session.

Table 5.10.: Add Content (Media Resource) to a Meta Session

Method	**Remove Content**
Describes	Removes the content from the session and notifies all users in the session.
Argument	Session: Session ID of the session the content needs to be removed. ContentID: Content ID of the content to remove.
Returns	Returns the Session Description XML of the modified session.

Table 5.11.: Remove Content from a Meta Session

Method	AddUserContent
Describes	Adds the content to the user, showing that the user is consuming that content.
Argument	Session: Session ID of the session to which the content needs to be added. UserID: The User ID of the content consumer. ContentID: Content ID of the content to add.
Returns	Returns the Session Description XML of the modified session.

Table 5.12.: Creates an Association between a User and Content

Method	RemoveUserContent
Describes	Removes the content from the user, showing that the user is not consuming that content anymore.
Argument	Session: Session ID of the session the content needs to be removed. UserID: The User ID of the content consumer. iContentID: Content ID of the content to remove.
Returns	Returns the Session Description XML of the modified session.

Table 5.13.: Removes the Association between Content and a User

Method	AddLayout
Describes	Creates and adds the layout to the session according to the given arguments. Also notifies all users within the session.
Argument	SessionID: Session ID of the session the layout needs to be added. Type: Type of the layout document (CE-HTML or a SMIL). iURI: Link to the layout properties document. iResolution: Resolution for the layout (in HxW, height to width in pixel). iResolutionFormat: Format of the resolution (e.g. PAL, NTSC, 720P, VGA).
Returns	Returns the iSession Description XML of the modified session.

Table 5.14.: Add Layout Information to a Meta Session

Method	RemoveLayout
Describes	Removes the layout from the session and notifies all users in the session.
Argument	String SessionID: Session ID of the session from which the layout needs to be removed. String type: Type of layout to remove.
Returns	Returns the Session Description XML of the modified session.

Table 5.15.: Removes Layout from a Meta Session

Method	PollForSession
Describes	Gives back all information about the Session with the given iSession ID.
Argument	Session: Session ID of the session to pool for.
Returns	Returns the iSession Description XML of the new session.

Table 5.16.: Polls for Available Meta Sessions

Method	Remove Session
Describes	Removes the Session from the repository.
Argument	iSession: Session ID of the session to remove.
Returns	n/a

Table 5.17.: Removes a Meta Session from the Server

5.2.4.2. Session Signaling Protocols

SOAP Interface The SOAP interface allows clients to call the different *IPTV Meta Session Manager* functions over standard SOAP requests, including the function name and any properties and with a SOAP response, including an *IPTV Media Meta Session Description* XML. Listings 5.2 and 5.3 show an example of a SOAP request and response:

Listing 5.2: Create Meta Session Request

```
<?xml version="1.0" encoding="UTF-8"?>
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
    <S:Header/>
    <S:Body>
        <ns2:CreateSession xmlns:ns2="http://soap.inem4u.fokus.fhg.de/">
            <iUserId>bob@milab.fokus.fraunhofer.de</iUserId>
            <tva>crid://fokus/10</tva>
            <description>A Test iNEM4U Session.</description>
        </ns2:CreateSession>
    </S:Body>
</S:Envelope>
```

Listing 5.3: Create Meta Session Response

```
<?xml version="1.0" encoding="UTF-8"?>
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
    <S:Body>
        <ns2:CreateSessionResponse xmlns:ns2="http://soap.inem4u.fokus.fhg.de/">
<return><?xml version="1.0"?>
    <iSDP xmlns="urn:schemas-iNem4U-org">
        <iSessionID>b7996479-6aa3-4eba-be05-1588da473b63</iSessionID>
            <specVersion>
                 <major>1</major>
                <minor>0</minor>
            </specVersion>
        <iContentList/>
        <iDescription>A Test iNEM4U Session.</iDescription>
        <iLayoutList/>
        <iState>
            <iState>CREATED</iState>
        </iState>
        <iUserList/>
        <tva>crid://fokus/10</tva>
    </iSDP>
</return>
        </ns2:CreateSessionResponse>
    </S:Body>
</S:Envelope>
```

SIP Interface Via the SIP interface, it is possible for clients to call the different *IPTV Meta Session Manager* functions using the Session Initiation Protocol and corresponding definitions for payload carried using XML data. In this case, the client sends so-called *IPTVMetaSessionQueries* as an *SIP INFO* request or *SIP MESSAGE* request depending on whether or not the client's already joined a session or not.

This means that if the client is not in a session, it can call the functions *PollForSessions* and *CreateSession* with a SIP MESSAGE request, and if the client is in a session, it can call all other functions with an SIP INFO request (e.g. *AddContent* function). The SIP response contains either the Session Description XML or the list of Session Descriptions embedded in the <SessionList> tag. The joining of a client to a session is done via a SIP INVITE method, creating the current session on a protocol level as well. Both types of SIP requests need to have the following structure (see Listing 5.4) according to the function to be called:

Listing 5.4: Meta Session Signaling with SIP

```
<iSessionQuery>
        <iQueryCall>Function</iQueryCall>
        <iQueryValueList>
                <iQueryValue Name="Property1">Value1</iQueryValue>
                <iQueryValue Name="Property2">Value2</iQueryValue>
        </iQueryValueList>
</iSessionQuery>
```

5.2.5. Discussion

This section has described *IPTV Meta Sessions* and finalizes the theoretical assumptions and compositions made for this topic.

For the time being, the defined *IPTV Meta Session Model* has reached a certain maturity through practical demonstrators developed in parallel to ongoing specification work. A focus has been set on the signaling aspects and multi-protocol support for connecting multiple client technologies. Future work will concentrate on extending the *IPTV Meta Session Model* to visualization aspects using CE-HTML and HTML5. This approach will deliver a platform independent methodology for the execution of *IPTV Meta Sessions.*

5.3. Interactive Content Provisioning

5.3.1. Introduction

Controlling the delivery of streaming media like Video on Demand and Linear TV in high quality, multiple formats, and for a large number of users is the current key to success for operator controlled edge networks for IPTV.

As already pointed out in the introduction chapter of this thesis, the developments towards a *Media Internet* have just begun and are enabled first in the last mile from the customer to Internet Service Provider (ISP).

During research on this thesis, one of the key requirements was to generate an end-to-end infrastructure for IPTV, allowing for the execution of interactive and Social IPTV services, while the generation, adaptation and network agnostic delivery of contents is beyond the scope of this work.

Within Section 5.1 how content will be provided throughout the described *IPTV Session* and the related signaling was described. For reasons of simplification the *back-end*, or rather the media server infrastructure behind it, has been considered and described as a black box, providing any type of requested media. This section will present and approach how interactivity can be added to these contents, describing a distinct functional building block inside the proposed architecture, namely the *Interactive Content Enabler* (ICE).

The work on the ICE represents a research activity conducted in parallel to this thesis, and adds a required interactive functionality to the Open Source project *VideoLAN Project [133]*. For this reason, only a limited overview will be presented here. Results have been contributed to the related projects or have been re-published as Open Source.

5.3.2. Interactive Content Enabler

Common media server implementations do not necessarily fulfill the requirements of the *IPTV Session Model* presented in this thesis. To address these issues, the author has decided to specify some minor extensions to the existing Open Source project VideoLAN. The main goal was to re-use the existing project within the proposed architecture, enabling both interactive and Social TV scenarios.

5.3.2.1. Core Functionalities of the VideoLAN project

As depicted in Figure 5.15, the VideoLAN project supports the following scenarios. This includes a number of input sources like :

- **Stored Content** (flat files).
- **Linear Television** (classic Broadcast TV).
- **IP-based uni- or multicast streams** as UGC or others.

All inputs can be transformed to various outputs by controlling the VideoLAN engine, including the following scenarios :

- **Scheduling** functionality for timer-based play out.
- **Time shifting** allowing the deferred play-out of contents.
- **Transcoding** and **Transration** used for any kind of content adaptations to another codec or bitrate.
- **Recording** functionalities allowing for the creation of a *Networked Personal Video Recorder* (nPVR).
- **Mixing and Insertion** capabilities used for overlays or mixing different streams to one output.

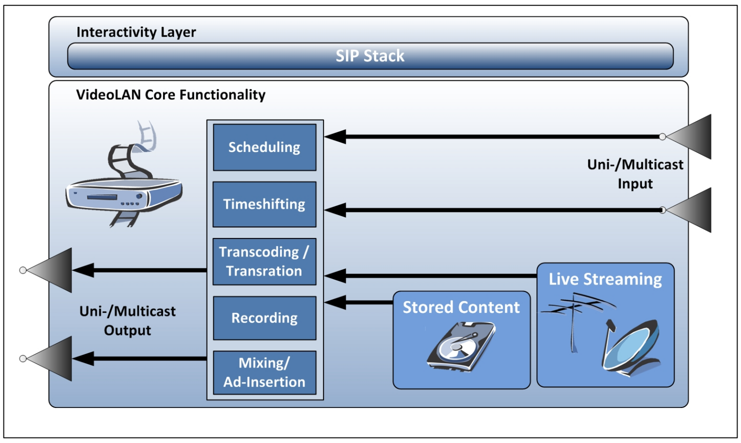

Figure 5.15.: Interactive Content Enabler - Basic Architecture

5.3.2.2. Architectural Approach

The *VideoLAN* has been has been extended analog to related works of Waiting [136], Menezes [84] and Menai [83] with an open Source SIP stack, allowing for the control of the VideoLAN core functionality. The limitation to the SIP protocol has been made, at this point, in order to allow for the evaluation of session-oriented, interactive scenarios.

Hence, as depicted in Figure 5.15, the *ICE* has been divided into two logical components:

- The *Interactivity Layer* allowing for the control of the VideoLAN using an Open Source SIP stack.

- The *VideoLAN Core Functionality* providing all functionalities for the adaptation and streaming of media.

The *Interactivity Layer* will be controlled by the *IPTV Session Management Enabler* through SIP signaling. It receives SIP messages from the *IPTV Session Management Enabler* and runs an internal logic to translate them into streaming commands for the *VideoLAN*.

Interactivity Layer The *Interactivity Layer* contains application logic for the maintenance of incoming interactive session requests, e.g. for targeted advertisements.

The engine receives messages from the requesting *End Devices* and creates internal states for all initiated sessions. Information about existing session states are retrievable through an API accessible with the VideoLAN Core Functionality. Based on this information, the VideoLAN will process the corresponding service logic. Incoming requests will be evaluated according to requested bandwidth, codec, content title or streaming service type.

A modification of an existing session implies the modification of the media streaming parameters of the corresponding media session.

VideoLAN Core Functionality To fulfill interactive service processing, the VideoLAN is fulfilling the described scenarios from above and used in the context of interactive and Social IPTV service generation, as specified in the following sections.

5.4. Session-Oriented Interactive Services

"Interactive programs have not been a success in most of the world. In general (apart from the UK), there has not been a widespread deployment of interactive video applications, although there is one exception: programs where the viewers can vote [64]."

Beginning with this statement from Johan Hjelm, this chapter is focused on demonstrating the extensibility of the proposed IPTV architecture through the addition of interactive applications based on the *Session-Oriented Application Environment* (SAE).

As we will see, the ideas presented take Hjelm's remark into account and try to pursue this warning by keeping the interaction model simple and user interaction intuitive. This implies that, especially currently successful interactive scenarios – mostly combining TV and telephone or the Short Message Service – will be mapped onto IPTV.

In the following sections, different session-oriented interactive services that make use of the introduced *IPTV Session State* and *IPTV Session State API* will be discussed. This includes a short recapitulation of the above-mentioned mechanisms and their combination with the so-called *Session-Oriented Interactive Services for IPTV* introduced here.

5.4.1. Introduction

As described earlier in this chapter, an *IPTV Session* contains stateful information about a user's service consumption, maintained in the so-called *IPTV Session State*. This section will focus on describing how these third party access mechanisms can be used to create additional services on top of the session-oriented infrastructure. Beside that, the methodologies for *Interactive Content Provisioning*, as presented in Chapter 5.3, will be used to provide and adapt content inside these scenarios.

5.4.2. Basic Principles

The basic idea of all scenarios discussed specified in this chapter lies in reading or manipulating a user's *IPTV Session State*. A session-oriented signaling based on the SIP protocol is used additionally, for all information transferred between *End Devices* and the infrastructure.

- Read-only operations for the *IPTV Session State* are sufficient when no interaction or manipulation with the content is required, e.g. for voting scenarios or user polls.
- Session parameters have to change using the *IPTV Session State API*, whenever the interaction model requires a different content or content format, e.g. for targeted advertisements.

Furthermore, scenarios requiring switching between contents can either contain a session modification between the user's *End Device* and the *IPTV Session Management Enabler* (client side switching) or transparent for the *End Device* taking place just between the *IPTV Session Management Enabler* and the *Interactive Content Enabler* (ICE) (content injection at network side).

Figure 5.16 outlines how services can be added to and benefit from the proposed architecture by using generic API calls to read the *User Profile* or *IPTV Session State* or manipulate the latter, using the corresponding API calls. The following three services have been composed making use of the APIs provided by the *IPTV Session Management Enabler*:

- The *Televoting Service* (Section 5.4.3) represents the simplest interaction model by posing content related or content independent questions and possible answers to a set of specified users. Feedback can be given by choosing an answer from the pool of provided answers blended on the user's screen.
- The *Targeted Advertisement Service* (TAD) (Section 5.4.4) represents a mechanism for content provider-driven session modifications and dynamic content insertion. The results are advertisement clips inserted into the running contents. Two models, either inserting the media into the current stream at network level or an STB driven approach, will be presented.
- The *Virtual Quiz Show Service* (Section 5.4.5) represents a so-called user participation scenario. The scenario requires that the user's feedback, like in the Televoting scenario, as well as *User Generated Content* (UGC) be inserted into the other participant's stream coming from other users.

Table 5.18 summarizes the requirements of the three interactive services towards the *IPTV Session State* and *IPTV Session State API*, respectively.

	Televoting	Targeted Advertisements	Virtual Quiz Show
Read-only operation through ISS	sufficient	not sufficient	not sufficient
Session modification through ISS API	not required	required	required
Content inception	not required	required	required
User Generated Content	not available	not available	essential

Table 5.18.: Comparison of Requirements for Session-Oriented Interactive IPTV scenarios

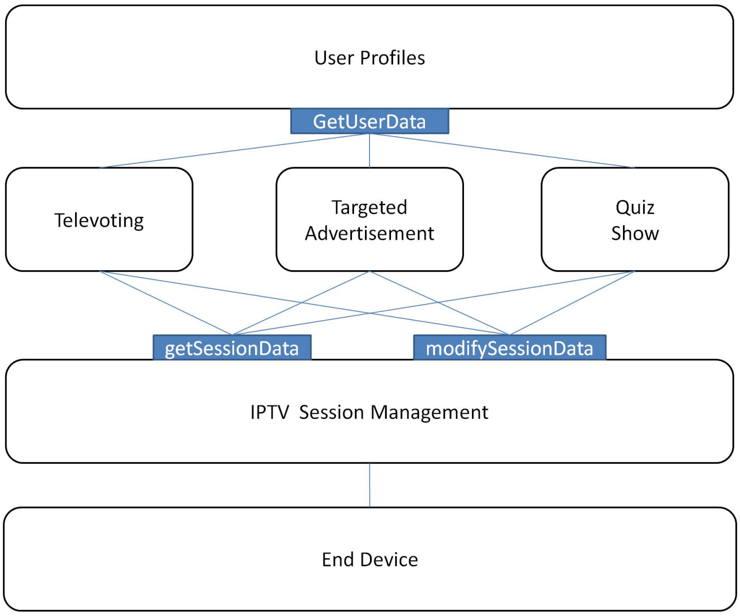

Figure 5.16.: Session modification in a Session-Oriented Application Environment

From a service perspective, the three services make use of three abstract API functionalities, as depicted in Figure 5.16 and have to be implemented by all of them:

- *getUserData:* giving feedback on a user's TV or VoD subscription data from its *User Profile*, which also contains additional data on a user's behaviour and collected habits over a longer period of time.
- *getSessionData:* providing the *IPTV Session State* of a set of users specified within the parameter when issuing the request towards the *IPTV Session Management*.
- *modifySessionData:* is used to insert new content or messages into a running session.

5.4.3. Televoting Service

This section describes the mapping of the most successful interactive service onto the proposed architecture, with respect to signalization and implementation complexity: Televoting, allowing a user to participate in a poll using his remote control.

This work has been published by the author in [109] and represents a basic use case, also composing a part of different draft IPTV standards. The implementation has been evaluated in the FOKUS Open IPTV Ecosystem test bed.

5.4.3.1. Scenario Description

Current interaction schemes in terms of voting during TV shows rely on the interaction of the user using his mobile device (e.g. via SMS) and the TV. This circumstance poses a not inconsiderable barrier for a lot of users (e.g. usability and expensive charging for SMS). However, the bidirectional TV experience allows the user to interact with the system directly via the remote control.

A key feature is participation in television game shows like e.g. the British quiz show: *Who Wants to Be a Millionaire*? Here, the *Ask the Audience* question could be realized by a real audience of thousands of users. The user could then take part directly from his TV in a lottery with a customized prize (according to his or her preferences and usual behavior).

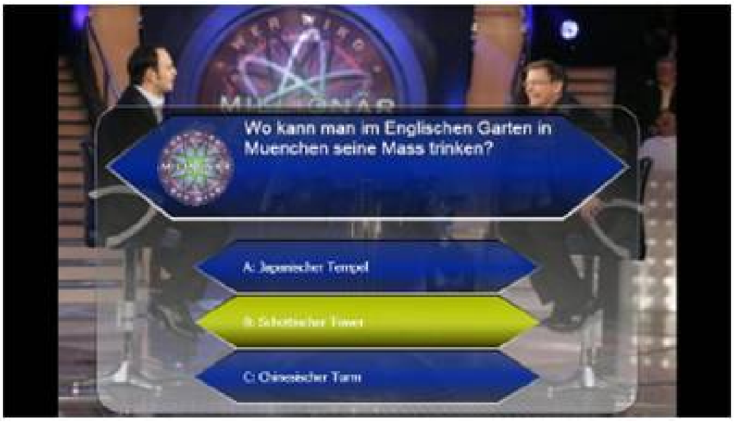

Figure 5.17.: Televoting example

5.4.3.2. Metadata

A well-defined XML metadata schema is used to carry the necessary information like payload during signaling the Televoting Service. This metadata is used inside the messages exchanged during the scenario and carries, on the one hand, information to be displayed to the user on the screen and, on the other, application level signaling to trigger the application logic on the *End Devices* and Application Server.

Listing 5.5 and 5.6 show the defined metadata schemas used for the Televoting Scenario. Beside the questions posed to the user, the price for participation in this poll, as well as potential answers are provided using the schema. Each question and answer is combined with a unique identifier. The user's response contains the corresponding identifiers for the current poll as well the identifier of the answer given by the user.

Listing 5.5: Voting Metadata

```xml
<?xml version="1.0"?>
<Vote xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
   xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <Id>dummyVote</Id>
  <Text>Who is the best?</Text>
  <Price>
   <Currency>$</Currency>
     <Value>0.5</Value>
   </Price>
  <SelectionItem>
    <Id>dummyItem1</Id>
    <Text>I</Text>
   </SelectionItem>
  <SelectionItem>
    <Id>dummyItem2</Id>
    <Text>You</Text>
   </SelectionItem>
  <SelectionItem>
    <Id>dummyItem3</Id>
    <Text>Nobody</Text>
   </SelectionItem>
</Vote>
```

Listing 5.6: Voting Response Metadata

```xml
<?xml version="1.0"?>
<MyVote xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
 xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <VoteId>dummyVote</VoteId>
  <MySelectionId>dummyItem3</MySelectionId>
</MyVote>
```

5.4.3.3. Service Signaling

As described above, the Televoting Scenario has been kept as simple as possible with regards to the used metadata, as well as on the signaling level. Figure 5.18 depicts the simplified signaling procedures used within this scenario.

First, the *Voting Service Enabler*, representing an Application Server implementing the service logic for the Televoting Service, issues a session modification request using the *IPTV Session State API* (ISS API). This request contains information about a certain user group (e.g. users on a specific channel) as well as the necessary metadata for this poll. A session modification is not mandatory in this scenario but ensures that the *IPTV Session Management Enabler* is aware of the ongoing Televoting

Second, the *IPTV Management Enabler* issues a SIP INFO message towards subscribed users, fulfilling the session modification request criteria. This message results in triggering the application logic on the corresponding *End Devices*, putting the question and possible answers onto the screen (see Figure 5.16). In a next step, the user chooses from the presented pool of answers, resulting in a SIP INFO message that is returned to the *IPTV Session Management Enabler* and the corresponding Application Server. Finally, the poll results can be displayed to the user.

Table 5.19 describes these steps according to Figure 5.16.

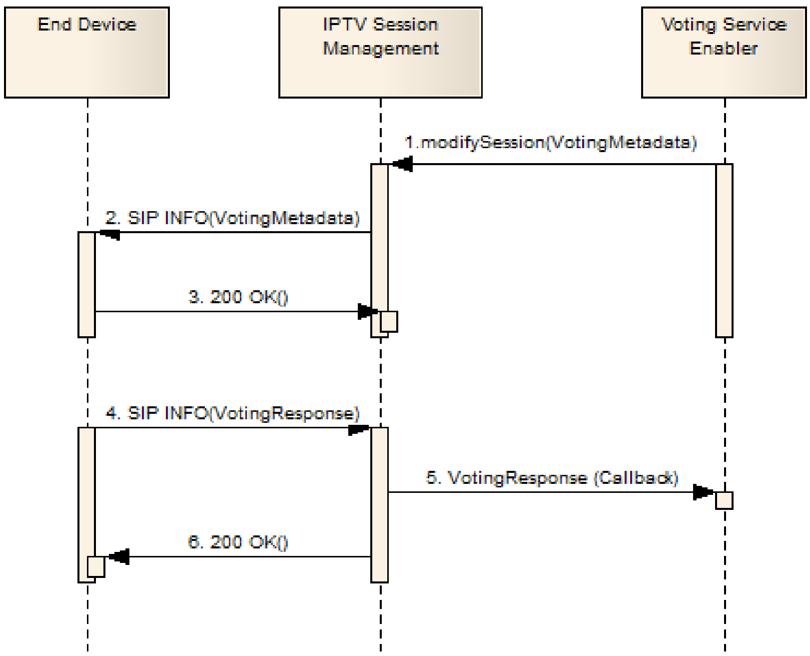

Figure 5.18.: Procedures for Voting Service

5.4.3.4. Discussion & Limitation

With the specification of the Televoting Service, a first interactive service using the *Session Oriented Application Environment* (SAE) has been mapped onto the proposed architecture. Due to its simplicity, the resulting user experience is much better, because no media break occurs during service usage. Answering a poll only requires a click on the remote control. The following services will describe more sophisticated services, resulting in more complex signaling, as well as metadata

Procedure/Message	Operation
1. **ModifySession**	This operation sets the Ad Engine to operational mode with enabled Ad Points.
2. **SIP INFO**	Representing users on a TV channel.
3. **200 OK**	A trigger has been activated by the scheduler.
4. **SIP INFO**	Targeted users are being selected.
5. **Callback**	An ad clip is selected and associated with each single user or user group.
6. **200 OK**	Generating the XML data to be sent to individual users.

Table 5.19.: Signaling Procedures for Televoting Service

structures.

5.4.4. Targeted Advertisement Service

Addressing single users or user groups with personalized, so-called targeted advertisements has always been on the wish lists of broadcasters and content providers.

From a technical perspective, advertisements are messages and content sent to the consumer, which are not part of the main program [64].

These messages can be realized in different formats and inserted either in a classical way, inside a so-called ad break, as an overlay on top of the main content, or through various other procedures, which will be presented in the following. Personalized Advertisement Services go one step beyond this, and try to place different advertisements for different users, using one or more of these mechanisms. The process of finding the right advertisement for a single user or user group is called targeting. The technical challenge is obvious and lies in finding the perfect match between a pool of advertisements, the right time slot or ad location, and the different users. Within the scope of this thesis, the exact targeting has not been the main focus, rather the generic procedures and signaling that make such a service available.

The process of providing a Targeted Advertisement Service involves several parties. This includes:

- The Advertiser or service and product owner who wants his product to be advertised, e.g. while a user is consuming content.
- The IPTV Service Provider who is offering the service.
- The consumer who receives advertisement during content consumption.

The following paragraphs will describe how a targeted advertisement system integrates and enriches these entities into the proposed *Open IPTV Ecosystem Core*.

5.4.4.1. Standards Frameworks for Targeted Advertisements

Targeted Advertisement Services are an important topic inside various standardization bodies and research. This section will describe the two most important bodies still working on their final specification: The Open Mobile Alliance (OMA) with the so called *OMA Mobile Advertisement (OMA Mob Ad)* [100] specification and the North American Society of Cable Telecommunications Engineers (SCTE) and their *SCTE-130* specification. The SCTE standards for targeted advertisement are additionally approved by the *American National Standards Institute* (ANSI).

OMA Mob Ad The OMA's Mob Advertisement specifications have been created to enable targeted advertisement services in mobile networks. Nevertheless, the specifications also fit into IPTV environments. For this reason, they also play a role within IPTV standardization. Figure 5.19 illustrates the general idea of the

OMA's approach, which shares some similarities with the author's work as we will see starting with Section 5.4.4.4:

- The Ad Server is responsible for ad selection (mapping ads to user profiles), ad delivery (ads and corresponding metadata), ad metric handling and user data management (gathering data for contextualization and personalization).
- The Ad Engine is the Ad Server's counterpart on the device and fulfills the same functionalities at the endpoint including ad acquisition, ad selection, ad metrics handling, user data handling.
- SP App is a network-located application providing any kind of service that could be combined with advertisements like Web portals, e.g. a browser-based EPG or a media server as described in Chapter 5.3. It connects with the Ad Server to obtain ads.
- An AD App can be any kind of client-side application that requests advertisements. Within the scope of this thesis, this could be a TV or STB.

Beside these architectural specifications, the actual implementation and details concerning protocol and and metadata specification were still under discussion when this thesis was written. A specification focusing on metadata has been provided by the SCTE and is discussed in the next section.

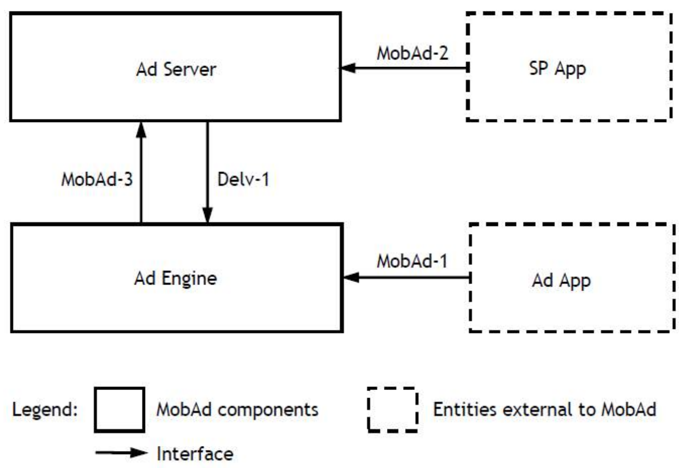

Figure 5.19.: OMA Mobile Advertisement Architecture [100]

Cablelabs SCTE-130 In contrast to OMA Mob Ad, the SCTE 130 standard supports a unified platform for addressable advertising, providing inventory and placement definitions while merging content and subscriber metadata for targeting zones – or, in a unicast environment, for targeting individuals [125].

As a complete framework, SCTE-130 defines the language spoken between participating machinery, and which messages they'll exchange. Likewise, it defines how the machines will be connected during the creation of addressable and interactive advertising.

SCTE 130 doesn't define how that targeting and campaign work should be done – that's the job for innovation [30].

The SCTE specifications are subdivided into six different parts, in which part one and two contain interface specifications and three to six focus on normative specifications.

5.4.4.2. Target Advertisement Schemes

Advertisement Schemes define scenarios for ad-insertion. Different so-called Targeted Advertisement Schemes have been described in the literature and will be presented within this section [124, 103]. Besides the classical way of inserting advertisements, which is based on a fixed pre-defined scheme (see Figure 5.20) Targeted Advertisements add a number of scenarios as depicted in Figure 5.21.

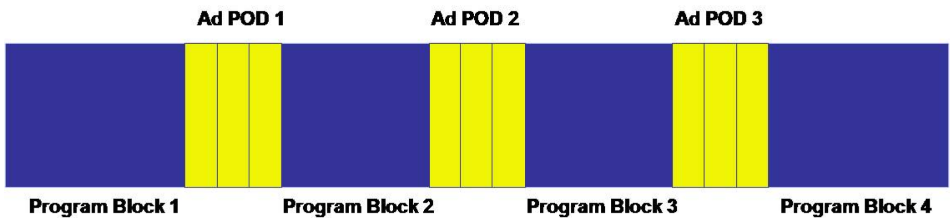

Figure 5.20.: Classical Advertisement Schema [124]

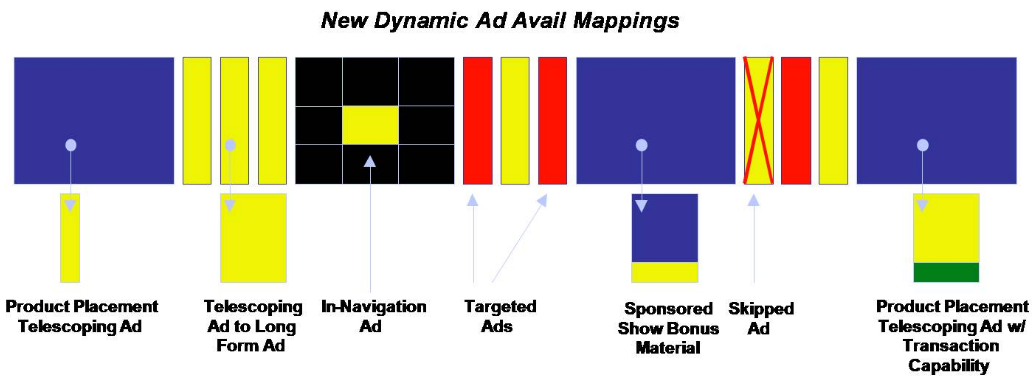

Figure 5.21.: Targeted Advertisement Schemes [124]

Classical Ad-Insertion The classical ad-insertion schema is based on a static model defining so called Ad Pods, which represent free space in between, or at the end of a program block. A Ad Pod could be filled with at least one advertisement. The schedule for ad insertion is defined before the actual play-out, and a dynamic switching is not possible. Nevertheless, regional ad-insertion was also previously possible. Especially in US cable networks, so-called ad splicing[1] technologies are common for the insertion of ads in local cable offices.

Telescoped Advertisement The first scenario enabled in IP-enabled TV systems is Telescoped Advertisement allowing the Content or Service Providers to point to external resources containing additional advertisement and information made available to the consumer. This may be web-based information portals, or hooks to special advertisement streams.

Targeted Ad Blocks The most sophisticated advertisement approach, and also used for the prototype, is placing advertisements based on the users profile, which contains information about age, sex and other personal information enriched with information about personal viewing and consuming habits.

Overlay Advertisements This feature is often used for so-called in show sponsorship, putting a special focus on a specific item that apparent in the content. The presented banners may be enhanced with links to extended content or advertisements.

Transaction Capabilities All described scenarios can be enriched by adding more interactivity to the advertisements through providing access to the products presented in the different spots. This allows the consumer to buy the presented or related products through his IPTV interface, and charge them either through his Service Provider on his monthly bill, or through credit card credentials also stored in the user profile.

Of course, a mixture of nearly all the presented scenarios is possible, like placing a targeted Picture-In-Picture Advertisement, telescoped targeted Ads or mixing targeted and non-targeted advertisements.

The following section will describe where advertisements will be inserted, either on the client or inside the network.

[1] A splicer works by watching the river of MPEG-2 video bits blasting in from the satellite. When it sees a "digital cue tone," it executes the task of splicing in the local ad, housed on the video server [31].

5.4.4.3. Approaches for TAD Insertion

The insertion of single or multiple advertisement blocks can be realized at different locations, either:

- inside the network using the *Interactive Content Provisioning* mechanisms from Chapter 5.3 or
- through each single client and a switching between different content items, respectively.

The concepts presented in this thesis and beginning in the next section support both approaches, whereas the prototypical implementation made on the context of this thesis only supports the client side insertion of advertisements.

5.4.4.4. TAD Service Architecture

The targeted advertisement integration follows the basic principles introduced alongside the *IPTV Service State* and *Third Party Access*. This implies that a trigger-based mechanism is used to activate a targeted advertisement scenario by modifying active sessions.

In addition to the triggering method, a complex scheme for the generation of targeted advertisements has been designed and will be described in the following sections. This includes the basic work flow, including an entity specification, the TAD Compositions Lifecycle, necessary metadata formats, a description of signaling flows, and finally, a description of the reference implementation.

TAD Service Workflow In order to compose a Targeted Advertisement scenario, different entities have been designed that help to model the process. All-in-all, these entities form the Targeted Advertisement Engine presented in the next section. A general overview called *TAD Service Workflow* is depicted in Figure 5.22:

- The Advertisement Enabler controlled by an administrator for the creation, editing and setting of Advertisement Schedule Data using a Web interface.
- The Advertisement Engine, which uses a trigger-based mechanism, access to current session using the *IPTV Session Stat*e and the protocol stack to generate advertisement messages.
- The User in front of his TV watching content and inserted personalized advertisements, looking at product pages in telescoped advertisements and making transactions by buying products.

TAD Data Model In addition to the basic work flow presented in the last section, a data model is used to compose the whole application life cycle. The entities of the data model are the following:

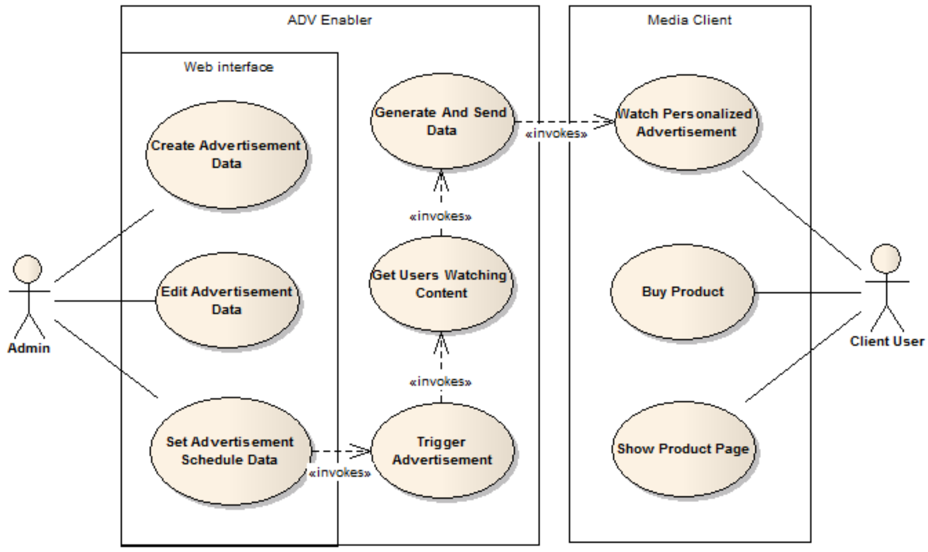

Figure 5.22.: TAD Workflow

User The single user entity, finally consuming targeted advertisements.

User Group The User Group Entity is responsible for allowing the personalization of the different advertisements. In this case, a user group represents more general user-specific information on which a personalization can take place, like for example: age groups, hobby-related group or gender specific groups.

Advertisement The Advertisement Entity specifies the core attributes for an advertising spot. This includes, most importantly, a link to the advt. media file (URL), but also additional information like a name, the vendor, an icon (URL) and the duration of the media file.

Product Clip The Product Clip Entity adds a (product and advertisement would not normally be capitalized) Product into an Advertisement. This means that it contains a start and an end time for the occurrence of a specific Product within a specific Advertisement. The Product Entity contains all important information about a specific product. This includes a name, an icon (URL), the vendor, a price and, a Product Page (not mandatory).

Product Page The Product Page Entity represents a web page with additional information about the product. This gives the user the possibility to access further information about a product he is interested in directly, while watching the Advertisement.

Advertisement To Group The Advertisement to Group Entity associates one Advertisement with one User Group. With this association. it is possible to determine the right Advertisement for a User, according to the User Groups a User is in, and end up with a personalized Advertisement.

Advertisement Point The Advertisement Point is the major Entity for triggering the personalized advertisement process. Mainly it associates a specific content (Media CRID) with a number of AdvtToGroup Entities. In addition to some descriptive attributes, it includes a start time when the advertisement is supposed to be triggered.

The relationship between the described entities is visualized in Figure 5.23.

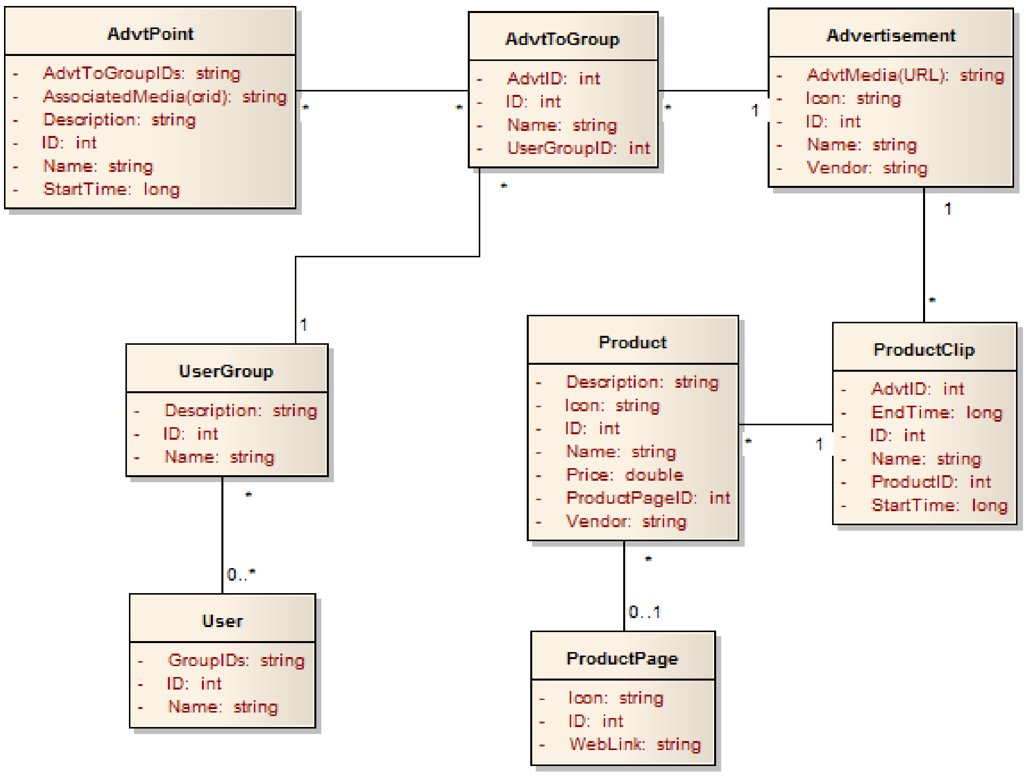

Figure 5.23.: Class Diagram TAD engine

5.4.4.5. TAD Service Composition Lifecycle

Using the data model and its relations, the so-called *Compositions Lifecycle* describes the control of the Advertisement Engine and its corresponding state machine.

As depicted in Figure 5.24 the Ad Engine has an operational and a maintenance state. Before the operational mode can be set to active, an initial maintenance cycle has to be conducted, which is described in Table 5.20. After this initial preparation – which of course could be repeated whenever necessary – the so-called Ad Points could be set to active and triggered by the defined schedule.

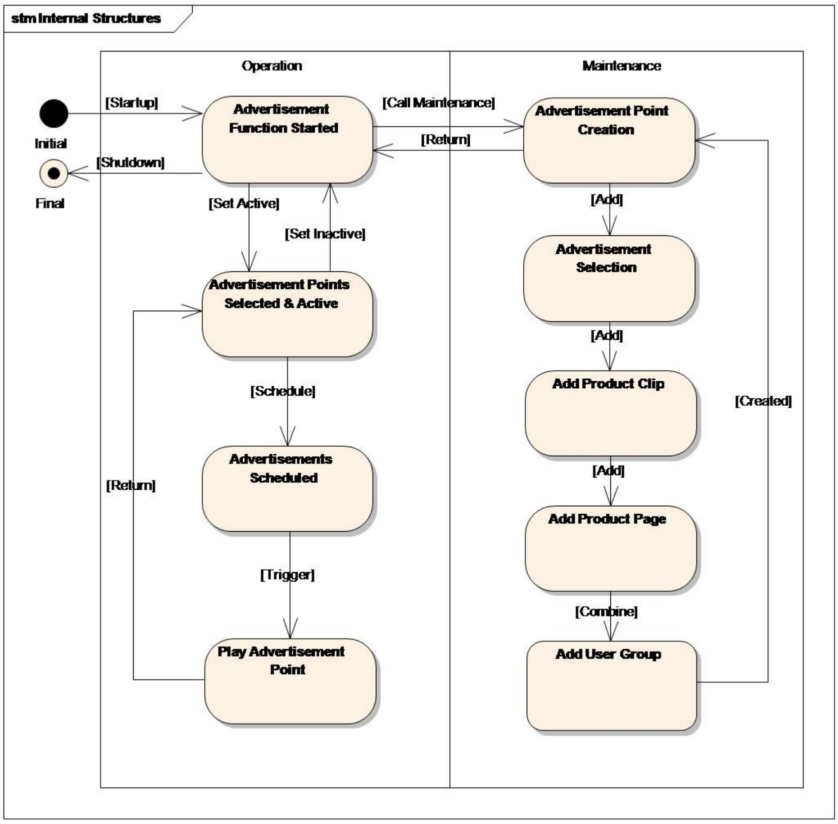

Figure 5.24.: State Diagram – Targeted Advertisement Composition Lifecycle

5.4.4.6. TAD Metadata

After making work flow, data model and the composition life cycle for targeted advertisement generation available, the next step is defining the metadata used when a client or media server is triggered to play a targeted ad. Listing 5.7 provides an example of the information used in this context for client side ad insertion. The XML data consists of the following elements:

- A link or URL to at least one ad clip, i.e. a video file.
- Time stamps and clip duration to allow the client to insert an ad at the right position.

State	Operation
1. **Ad Engine Started**	The Ad Engine is started by the administrator. A web interface allows for switching to either maintenance mode or operational mode.
2. **Advertisement Point Creation**	In maintenance mode new Advertisement Points can be created. These points represent an add trigger point, including advertisement schemes, schedules, the targeting and mapping to users and corresponding collections of ad clips.
3. **Advertisement Selection**	The first step after having created an Ad Point consists of adding a collection of ads. These clips are provided externally, e.g. by a Content Provider
4. **Add Product Clip**	Each ad consists of at least a product clip, i.e. a video file.
5. **Add Product Page**	A product page is used for telescoped ads and added in the form of a URL.
6. **Add User Group**	Finally user groups representing a targeted audience are added to the ad point. User groups are provided either manually or by a targeting engine (beyond the scope of this thesis).

Table 5.20.: Ad Engine Maintenance Operations

- Metadata describing the product to allow the in-ad transactions as described in Section 5.4.4.7 (optional).

Listing 5.7: Advertisement Metadata

```
<?xml version="1.0"?>
  <Advertisement xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        xmlns:xsd="http://www.w3.org/2001/XMLSchema">
    <Uri>http://193.175.134.181/advertisement/advertisement1.mpg</Uri>
      <Duration />
      <Clip>
        <StartPosition>00:00:15</StartPosition>
        <Length>00:00:13</Length>
        <Product>
          <Id>becks</Id>
          <Name>Becks Beer</Name>
          <Description>Taste the BECKS experience - Order your crate of
                       beer now and save 3$!
          </Description>
         <ImageUrl>http://193.175.134.181/advertisement/pics/becks.png</
             ImageUrl>
         <Link>nolink</Link>
         <Price>
           <Currency>$</Currency>
           <Value>10</Value>
           </Price>
        </Product>
      </Clip>
      <Clip>
        ...
      </Clip>
</Advertisement>
```

5.4.4.7. TAD Service Signaling

Within the last two sections, how an advertisement campaign is composed, what the general work flow looks like, and afterwards which kind of metadata will be delivered, to either the media server or client for ad-insertion, have been described. This section will go into the details of ad signaling using the SIP protocol inside the proposed model. Finally, so-called in-ad transactions allowing for the purchase of the advertised product will be described.

Starting on a procedural level, Figure 5.25 illustrates the transactions between two users while viewing content. Table 5.21 below describes the process. The involved entities are:

- The Targeted Ad Enabler.
- The *IPTV Session Management Enabler*.
- A media server hosting the content (e.g. Interactive Content Enabler (ICE) from Chapter 5.3).
- Two users belonging to different target groups, namely Alice and Bob.

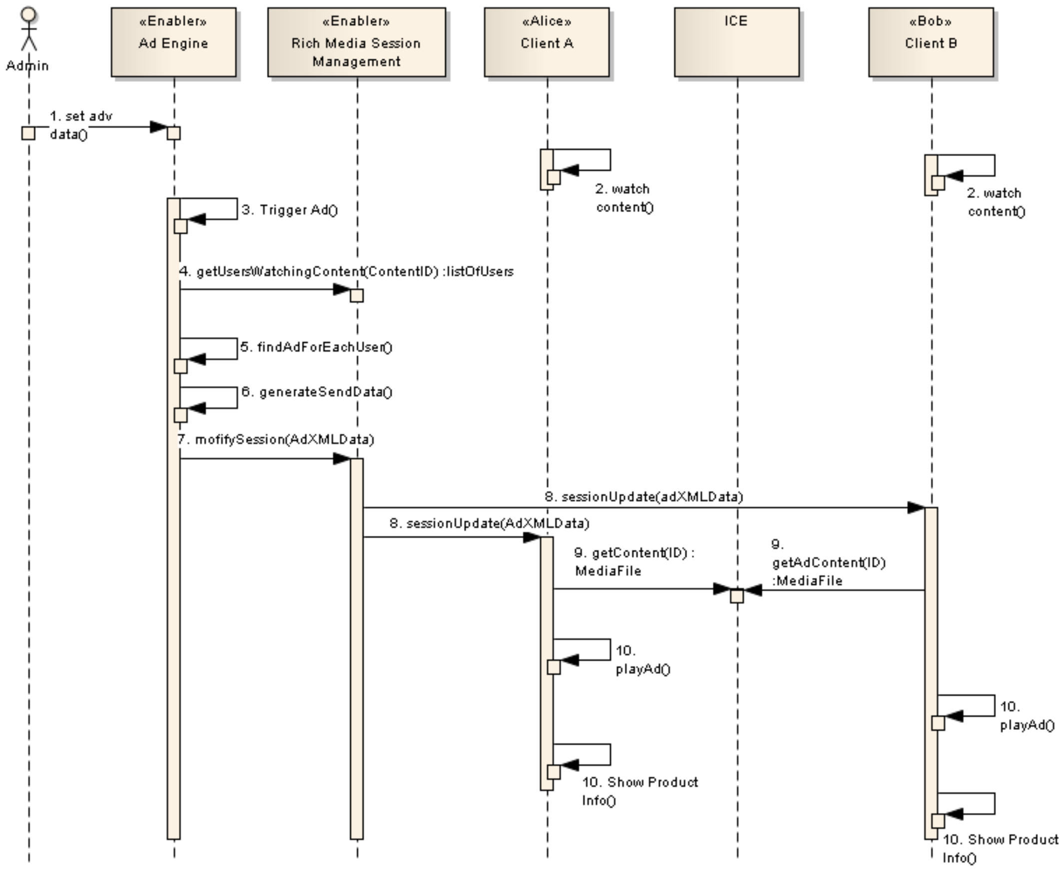

Figure 5.25.: Procedures for Targeted Advertisement Signaling

Procedure	Operation
1. **setAdData**	This operation sets the Ad Engine to operational mode with enabled Ad Points.
2. **watchContent**	Representing the users on a TV channel.
3. **triggerAd**	A trigger has been activated by the scheduler.
4. **getUsersOnChannel**	Targeted users selected.
5. **findAdForUser**	An ad clip is selected and associated with each single user or user group.
6. **generateSendData**	Generating the XML data to be sent to individual users.
7. **mofifySession**	Triggering the session modification on the IPTV Session Management.
8. **sessionUpdate**	Updating active sessions on the end devices.
9. **getContent**	Representing a client's request for an ad clip
10. **playAd**	A TV or STB playing a targeted ad .
11. **showProductInfo**	Optional representation of product purchase information.

Table 5.21.: Operations upon Targeted Advertisement Signaling

In contrast to the signaling flow in Figure 5.25, Figure 5.26 illustrates necessary signaling on the protocol level using SIP. Table 5.22 contains the necessary description information.[2]

Telescoped Ads & In-Advertisement transaction As already mentioned during the description of the various concepts from above, during play out of a targeted advertisement clip, additional metadata allows the product purchase inside the advertisement as a special form of telescoped advertisements. Figure 5.27 contains the procedures necessary to enable such a service. The first process demonstrates how a user interacts with the client to put a product in the shopping basket. The second process describes the process of a user opening a telescoped Web site.

5.4.4.8. Discussion & Limitations

The scenarios and specifications described in this section integrate a targeted advertisement service into the proposed architecture. The different ideas presented here have been verified through prototypical implementations, which have shown that the specifications meet the requirements on a functional level. Current limitations of the service were identified upon implementation. This includes:

[2] For reasons of simplification: Acknowledgement (ACKs) in SIP signaling are not displayed. Also, session setup procedures between IPTV SC and IPTV SE are not visualized

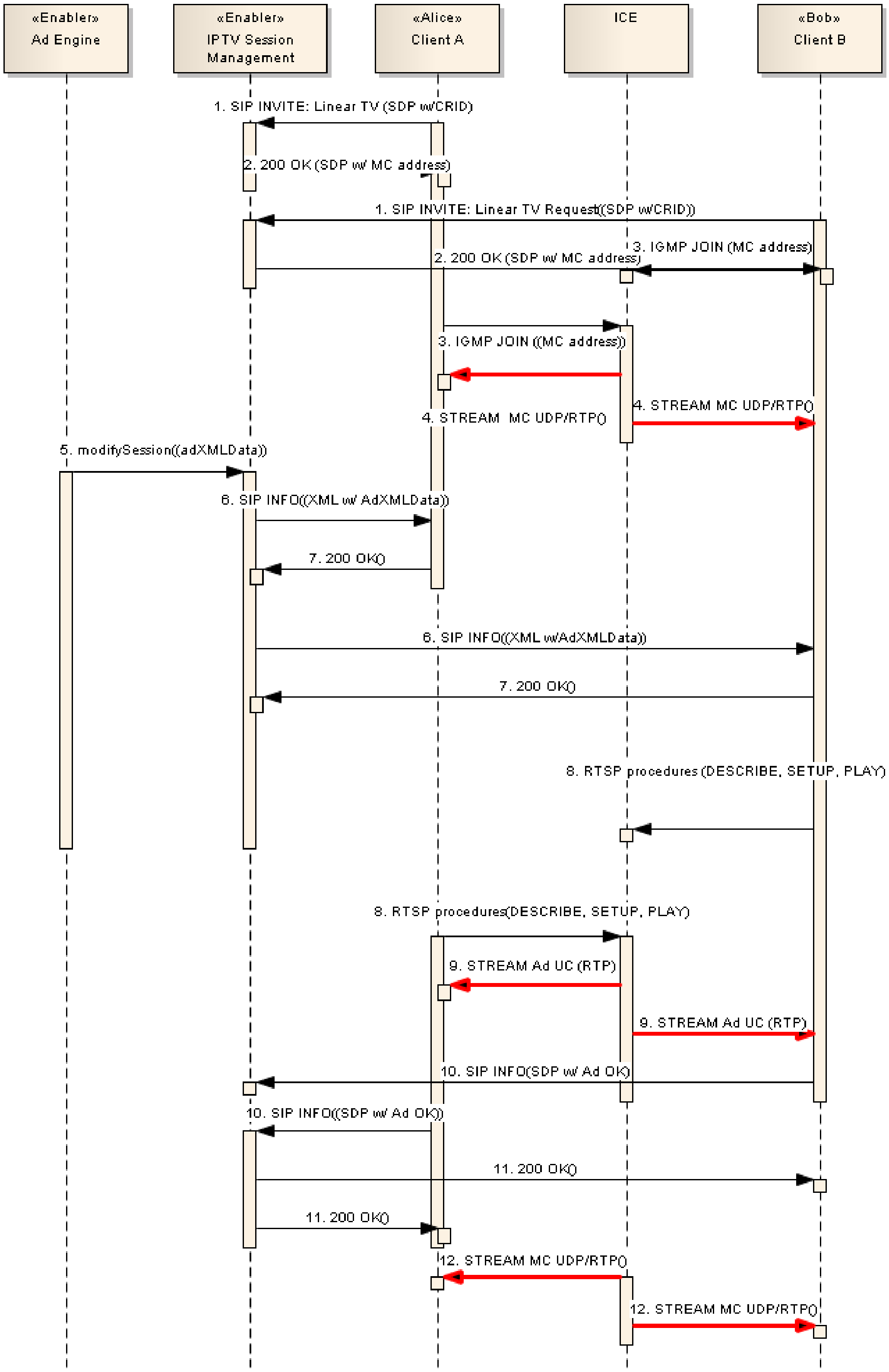

Figure 5.26.: Targeted Advertisement Signaling with SIP

Procedure	Operation
1. **Linear TV Channel request**	A SIP INVITE message is used by two users to request a Linear TV channel.
2. **IPTV SC response**	The IPTV Session Management resolves the request into a multicast address and sends back a 200 OK message including the multicast address.
3. **Joining a multicast address**	The user's clients join the Linear TV channel on the obtained multicast address.
4. **UDP/RTP streaming**	After joining a multicast address, channels are streamed using the UDP or RTP protocol.
5. **Modify Session Request**	The Targeted Advertisement Enabler initiates a request to modify the targeted users' sessions including the necessary metadata.
6. **SIP INFO**	SIP INFO messages are sent as in-session signaling to modify the two user's Linear TV sessions using IP multicast to IP unicast for the targeted advertisement clips .
7. **200 OK**	The clients acknowledge the SIP INFO message with a 200 OK message.
8. **RTSP procedures**	The clients initiate a unicast stream from the IPTV Session Management using RTSP signaling. (In before and not visualized: The IPTV Session Management has initiated a session towards the ICE.
9. **RTP streaming**	The advertisement clips are streamed using RTP unicast.
10. **SIP INFO (2)**	The user's clients inform the IPTV Session Management on active unicast streaming using a SIP INFO message.
11. **200 OK (2)**	The IPTV Session Management responds with a 200 OK message.
12. **Stream MC UDP/RTP**	After successful play out of all ads, the client switches back to IP multicast displaying the original Linear TV channel-

Table 5.22.: Targeted Advertisement Signaling with SIP

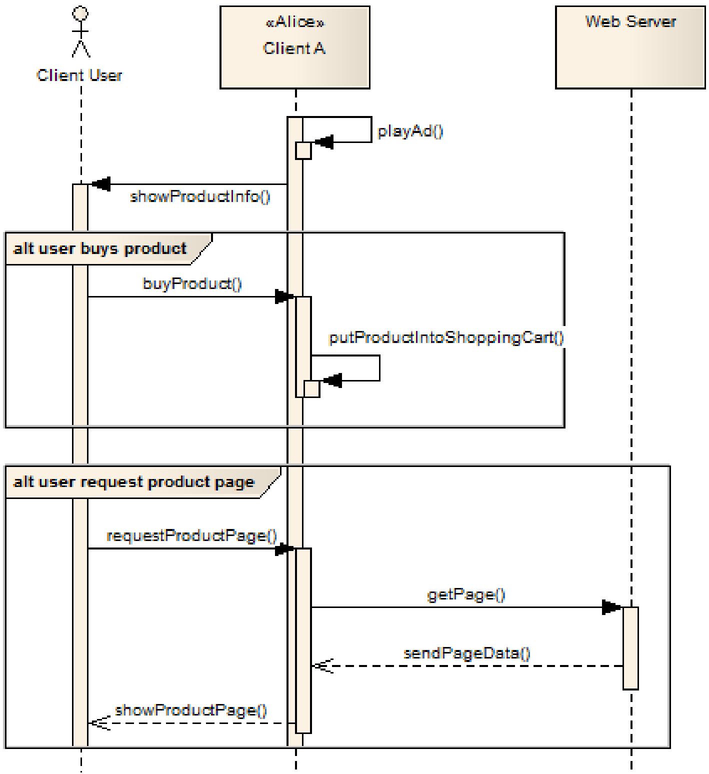

Figure 5.27.: State Diagram: TAD transaction

- The missing integration with a targeting engine incorporating contextual information and usage data. Future developments shall integrate work on recommendation engines for automation the targeting process.
- The usage of the *IPTV Session State* upon identification of active user sessions creates a huge overhead in large deployments, or when only small groups of users shall receive targeted advertisements. The full integration of the *IPTV session State API* will solve this problem.
- The missing implementations of the server-side ad insertion feature limit the possibilities for comparing both approaches. With advancing implementation status in the ICE, a comparative analysis should be possible in the near future.
- Future developments should harmonize metadata and signaling with corresponding standards like SCTE 130 or OMA Mobile Ad. To push the developments inside these bodies, the results of this work will be contributed to ETSI TISPAN, integrating both approaches.

While this section has analyzed how single users could be addressed by a third party service from the outside, the next section will introduce a service that involves multiple users that can also interact with each other.

5.4.5. User Participation on TV: The Virtual Quiz Show

Today, television shows increasingly ask for user participation through telecommunication services.

In various scenarios, user feedback is collected through the Short Messaging Service (SMS) or by dialling in through telephone (see Section 5.4.3 on Televoting). Through these services, users are already live participants in TV events but very limited in their actual possibilities for interaction.

By removing these barriers one could even go a step further in providing a platform where users can take part in TV shows or online gaming live from their living rooms.

IPTV users will provide live streaming film footage through their Web cams or their mobile phones, enabling fast and real-time interaction. In the context of this thesis, this is referred to as *User Participation on IPTV*.

Debates, competitions or (virtual) quiz shows would become more dynamic and attractive by adding a face to a voice. We can distinguish between two kinds of User Participation on TV:

- In the first case, user participation relies on a shared video content that all participating users are watching simultaneously. Two friends could therefore agree to watch a TV event together while each sitting in his own living room, i.e. in geographically separated spaces. Both friends would watch the TV event while also seeing each other on a dedicated part of the TV screen and interacting with each other. This kind of service can be referred to as *Watching Apart Together*.

- The second case does not include any shared video content watched by all users. Rather the final video consists of Web cam videos from all participating users with values. This kind of user participation is suitable for gaming services where users can see and compete with each other on TV and can be referred to as *Gaming Apart Together*.

On a conceptual level, the service could be implemented in two different ways: by either mixing the content on the media server or on the *End Device*, which is similar to the Targeted Advertisement concepts presented in the last section:

- The user's *End Device* could receive all video streams (the professional video stream from the ICE and the Web cam live streams from IPTV subscribers STBs), mix them up and display them on the TV. This requires a TV or STB with the ability to receive and render multiple streams from different sources. Figure 5.28 describes this concept in the case of user participation with shared video.

- Alternatively, all streams are first fetched by the ICE (Chapter 5.3) where they get mixed into a single video stream and sent to the participating users.

The TV therefore receives a single video stream containing the video from the different sources. Figure 5.29 illustrates this process in the case of user participation with shared video.

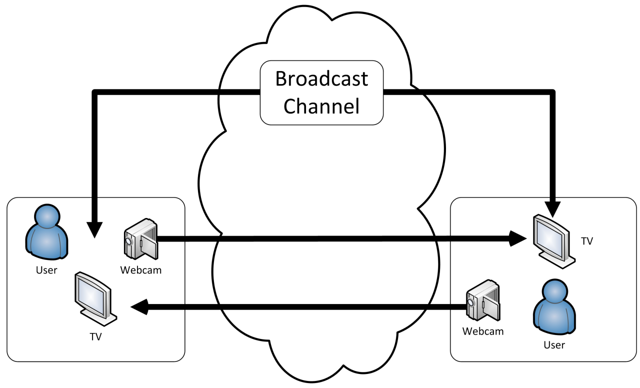

Figure 5.28.: Client-side multiple channels mixing

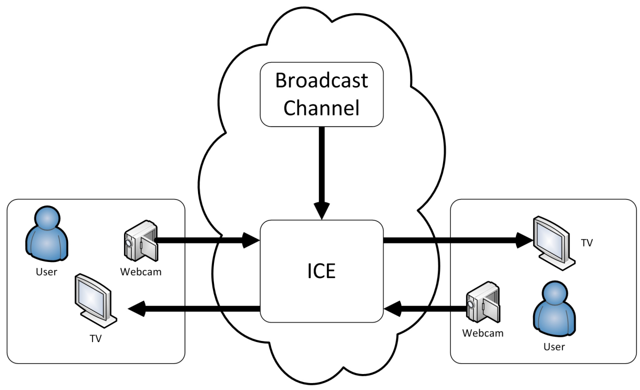

Figure 5.29.: Server-side multiple channels mixing

In both cases, the resulting mixed video appears in a mosaic shape or PiP (Picture in Picture).

Both options have advantages and disadvantages, but the first option is easier on an implementation level, as it doesn't require any media processing on the provider side. Solely, the signaling part of the service should provide the TVs with the correct physical links in order to retrieve the different Web cam live video streams from the other user Web cams. This option is best suited to user participation with shared video, as it requires little control from the service provider and the number of participating users is typically low. The direct access to Web cam live video streams may reduce the delay between the source Web cams and the shared video, contributing to a synchronous display of the shared video and the Web cam video. The main drawback however, is the need for some multi-rendering capabilities on the client side, to display all incoming streams on the TV.

The second option requires the ICE to mix up the final video stream on the network side. Mixing the final stream on an additional node before forwarding it to users may cause an additional delay between the live sources and the final destination. Because of the media processing overhead, this option may best suit scenarios involving several participating users, with no shared video that would require an additional synchronization process. However, in contrast to the previous option, the TV receives a single stream and therefore does not require multi-rendering functionalities.

Within the next sections the *User Participation on TV* idea will be described on a technical level, introducing a blueprint for the realization of such a kind of service.

Beginning with related work, the scenario will be described in detail. The following sections describe the architectural approach chosen, including the designed work flow, the composed data model, metadata and service signaling. A corresponding prototypical implementation will be analyzed in Chapter 7.

5.4.5.1. Related Work

User Participation on TV fits the idea of Social IPTV already introduced in Chapter 2 perfectly. Specifically, the idea of integrating *User Generated Content* is hereby mapped to a form of video chatting as described in various projects.

Additionally, the concepts are also relevant in IPTV standardization [39]. In this context, Deventer et al [27] have introduced a concept for studio-controlled upload of User Generated Content in mixed SIP and RTSP signaling environments. The ideas presented in this paper have influenced the author's work on *User Participation TV*. However, while Deventer et al assume a heterogeneous infrastructure speaking either the SIP or RTSP protocol, the author introduces a completely integrated approach using both protocols on all entities.

The corresponding detailed scenario description is provided in the next section.

5.4.5.2. Service Architecture

First, a new network entity, the so-called *Quizshow Application Server* is introduced and implemented within the *Session-Oriented Application Environment* (SAE). This entity follows the basic principles for *Third Party Access*, giving access to user's *IPTV Sessions*. The necessary application logic for the creation of metadata and specific service signaling is implemented here.

The Application Server resides on top of the *IPTV Session Management Enabler*, and is able to read and manipulate current session data. A minimum of two participating users is needed to create the scenario.

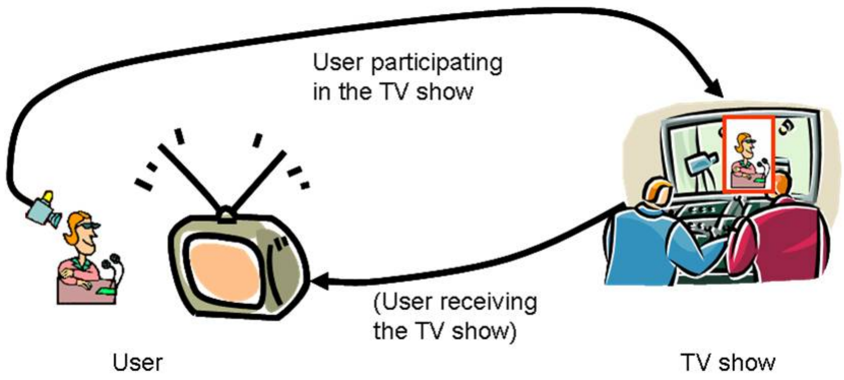

Figure 5.30.: Feedback Loop in User Participation TV [27]

The ICE is involved as long as the professional context is mixed and combined inside the network.

Workflow Composition Figure 5.31 illustrates the different processes of service preparation and service usage:

- A Web interface allows for the creation, editing and beginning of a virtual quiz show scenario
- The application logic is responsible for the execution of the service upon it, as it has been published and triggered within a specific context.
- The participating users are able to join announced quizzes, receive questions, publish answers and see the final results.

Data Model Creating an interactive TV experience – as in this case, a virtual quiz show – requires a corresponding data model. This data model is used by both the network and client side application logic. On the Application Server, the application logic interacts with a database to store the application data where the necessary protocol stacks are used to create the matching signaling to communicate with the client-side application logic. Figure 5.32 illustrates the used data model, describing a potential quiz show and its attributes. One show consists of a variable set of questions and associated with a variable set of potential answers.

Metadata The data model presented in the last section, and finally instantiated on an Application Server, will be used with the corresponding metadata whenever a new scheme is added to the Virtual Quiz Show enabler. Parts of the metadata are composed dynamically and sent to users in the service signaling described in the

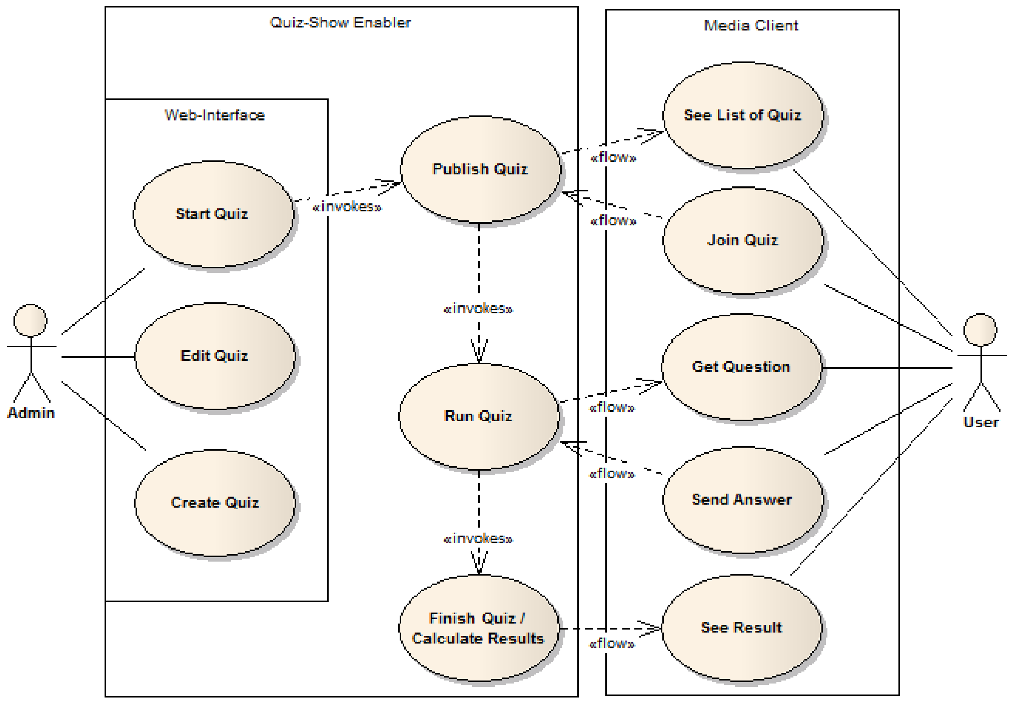

Figure 5.31.: Workflow User Participation TV

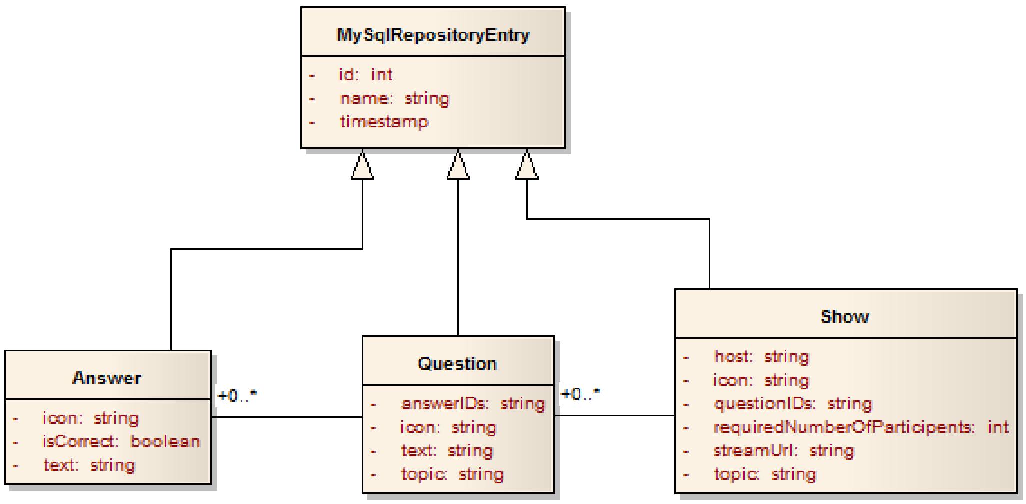

Figure 5.32.: Data Model User Participation TV

next section. Listing 5.8 illustrates the metadata used for the announcement of a Virtual Quiz Show. This data is carried inside a SIP INFO message. Various other XML Schemas are also used in this context.

Listing 5.8: Service Announcement Metadata

```xml
<?xml version="1.0"?>
  <QSAnnouncement xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        xmlns:xsd="http://www.w3.org/2001/XMLSchema">
<Name>Die Super Quizshow</Name>
<Topic>Naturwissenschaften</Topic>
<Host>Peter Müller</Host>
<Icon>http://193.175.134.101:8080/qs/images/1.png</Icon>
<NumberOfParticipants>2</NumberOfParticipants>
<NumberOfQuestions>3</NumberOfQuestions>
</QSAnnouncement>
```

5.4.5.3. Service Signaling

Being a complex service, the *Virtual Quiz Show* also requires a relatively complex service signaling compared to that of the target advertisement scenario.

The service signaling process consists of two major steps which include:

1. The Announcement Phase.

2. The iterative Question-Answer Phase.

Announcement Phase This is the first step in the *Virtual Quiz Show* scenario. As in all scenarios built on the SAE, the application selects users with active *IPTV Sessions* using the *IPTV Session State*. This happens according to previously created policies. Creation upon contextualization and personalization will not be discussed here as they are beyond the scope of this work. For example, both users might have previously subscribed to a *Virtual Quiz Show* session, or may belong to the same community sharing the same interests.

The service provider initiates this process using a simple Web interface.

Table 5.23 and Figure 5.33 describe the following steps in detail, beginning with the Application Server issuing a so called Quiz Show Announcement message:

5.4.5.4. Question-Answer Phase

After finishing the Announcement Phase, both users are prepared to participate in the actual scenario. The following procedures within the so called Question-Answer Phase are repeated until the pre-defined number of questions has been reached. Figure 5.34 and Table 5.24 illustrate and describe the process in detail. Acknowledgements have been left out for reasons of simplification.

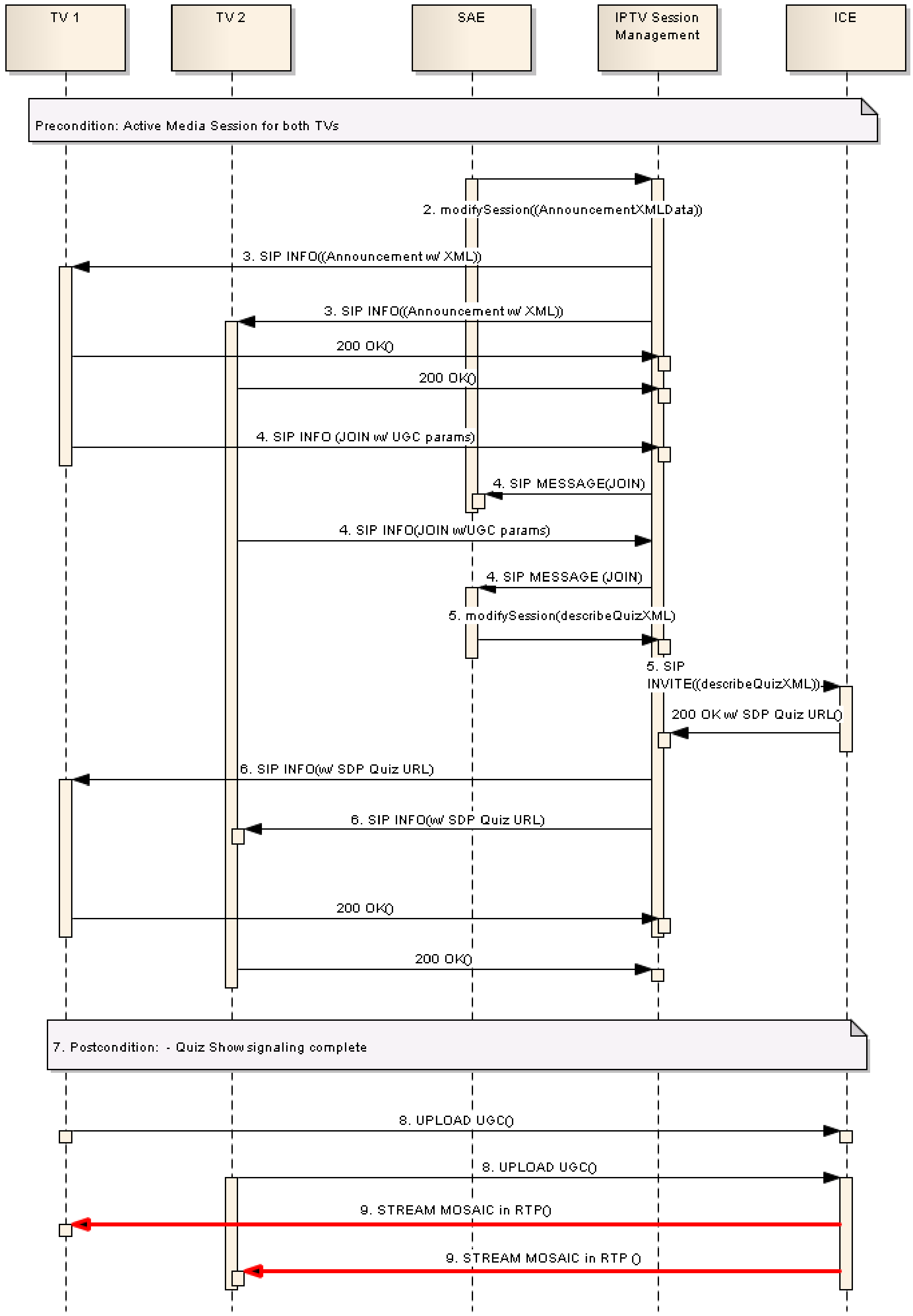

Figure 5.33.: Virtual Quiz Show Announcement

Procedure	Operation
2./3. **Start**	The AS sends a *Quiz-show Announcement Message* (QS announcement) in form of an API call (*modifySession*) to the IPTV Session Management Enabler which forwards this message as an in-session SIP INFO message to the candidates A and B. The message contains XML-data providing quiz-show related Information. This includes: • the Virtual Quiz Show title. • the topic of the actual quiz. • a potential moderator's name. • metadata describing the UI. • the number of participants. • the number of questions in the session.
4. **IPTV SM response**	Upon reception of the announcement message, both users answer with another SIP INFO message including their session parameters, i.e. pull address for UGC upload. These messages are forwarded as SIP MESSAGES to the AS.
5./6. **Media Session**	The AS collects the session parameters in a new XML-file and forwards this information towards the ICE for the integration of the UGC into the original content. The information about how to receive the mixed PiP stream is then forwarded as SIP INFO messages to the participants.
7./8. **Streaming**	Having completed the actual signaling for the scenario, both clients start sending their UGC data to the media server for insertion and, on the other hand, receive the mixed stream including the other user's content.

Table 5.23.: Virtual Quiz Show Announcement

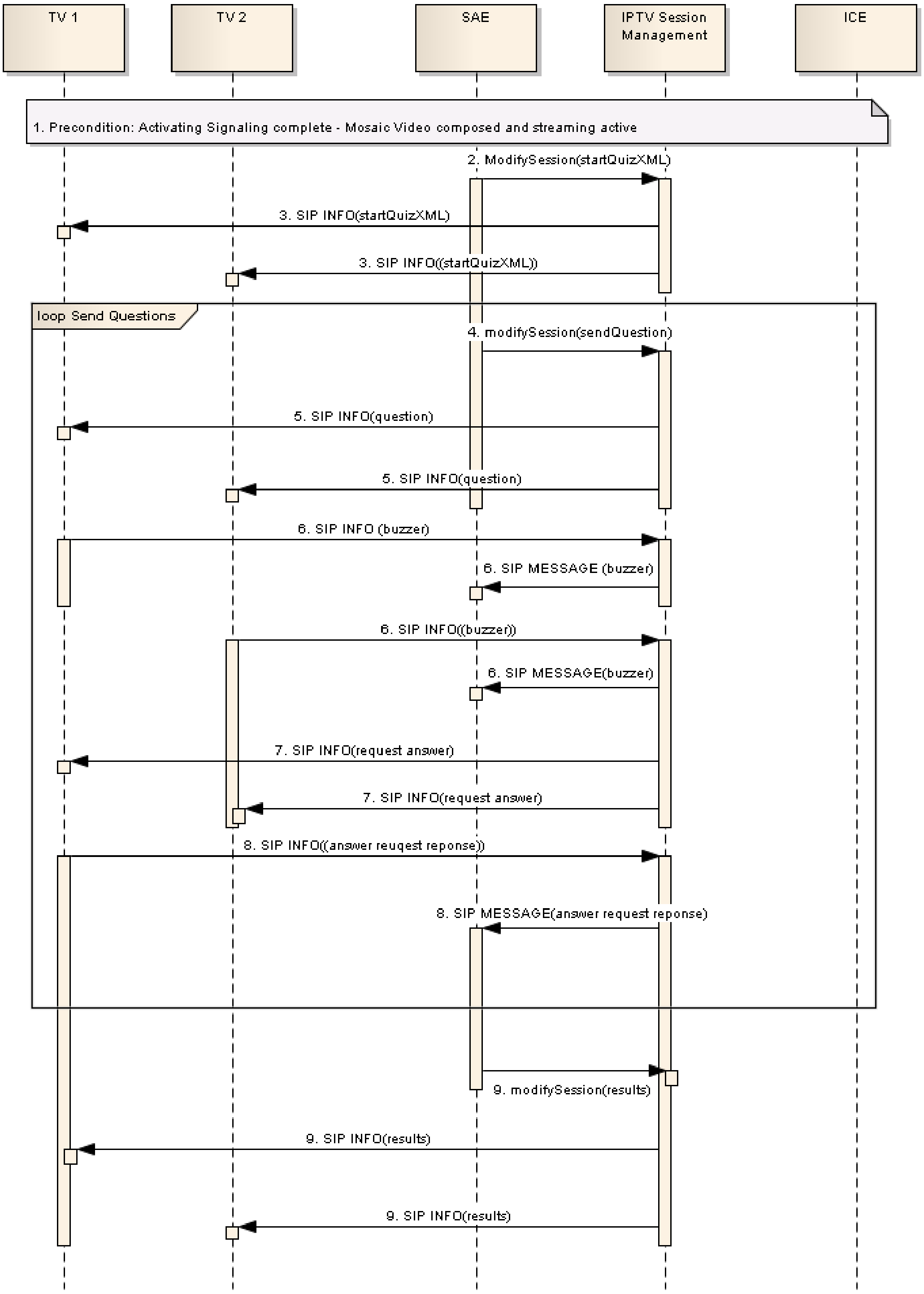

Figure 5.34.: Virtual Quiz Show Active

Procedure	Operation
3. **Start**	After candidate A and B have confirmed their participation in the session, the AS issues an modifySession request towards the IPTV Session Management Enabler. This results in an SIP INFO message towards A and B. This message contains metadata in detail as a countdown duration untill the next question.
4./5. **Question**	The next message coming from the AS starts the questions-answer loop. A SIP INFO message containing the first question and possible answers is issued. In parallel, it internally starts a timer which is set to a fixed time, in which both candidates are expected to react to the question. Both End Devices receive the INFO message synchronously and display them on the screen, allowing the candidates to read questions and answers on the TV screen and react to them.
6. **Buzzer**	By pressing a buzzer (e.g. the red button on the remote control) each client is able to initiate an INFO message towards the IPTV Session Management Enabler and the AS. The AS identifies the initiator of the first received INFO message and stops the timer.
7./8. **Answer**	The AS issues a request asking both clients to answer the question. Client A, as the first candidate to press the buzzer, is initially given the chance to issue an answer. The client will send his selection to the AS in a SIP INFO message.
9. **Result**	Finally, the AS stops the timer and sends the result back to both candidates in a new SIP INFO message, setting the end of the current iteration and the procedure restarts for the second and all other questions.

Table 5.24.: Virtual Quiz Show Active

5.4.6. Discussion

How easily successful interactive TV scenarios can be mapped on to the proposed architecture has been demonstrated. Especially the Televoting and Targeted Advertisement scenarios have been created by modeling plain signaling schemes.

This is also where the limitations of the presented signaling SIP signaling-oriented approaches become visible:

The Virtual Quiz Show scenario and the the corresponding signaling reaches a high complexity with regards to the number of messages to be exchanged and the payload to be carried. In addition, the necessary implementation of client side application logic also grows exponentially.

For this reason, Web browser-oriented solutions or a combination might be better suited for complex interactive scenarios.

5.5. Summary

This chapter provided the specifications for services implemented on the *Open IPTV Ecosystem Core* architecture. First, it has been demonstrated how session-oriented streaming services, including Live TV and Video-on-Demand are executed on the proposed architecture. Mechanisms exposing IPTV session data through the so-called *IPTV Session State* and a corresponding API have been presented.

Based on these principles, the *IPTV Meta Session* concept has been specified, enabling Social IPTV services on top of the proposed architecture.

An enabler for the provisioning of dynamically generated and adapted contents was then discussed and used by the interactive services presented.

In the following chapter, implementation details for certain aspects of the specified infrastructure will be discussed.

6. Implementation

Chapter 6:
Implementation

Session Management & Interactive Services
Interactive Content Provisioning

Following the specification of functional building blocks, APIs and services for the *Open IPTV Ecosystem Core*, as described in Chapters 4 and 5, in this chapter details about how the specification has been used to implement distinct prototypical components for the *Open IPTV Ecosystem Core* will be presented.

The implemented system consists of two core building blocks, namely a convergent runtime environment using Java technology and based on the concept of SIP and HTTP Servlets. This modular and composite runtime environment hosts *the IPTV Session Management Enabler*, *IPTV Meta Session Management* and *Interactive Services*. Second, implementations for *Interactive Content Provisioning* and therefore the *Interactive Content Enabler* have been realized by extending the source code of the Open Source media suite VideoLAN[1].

Clients for service tests and evaluations have been realized through various methods, either as native implementation using .NET technology or as Web browser-based solutions implemented in HTML, CSS and JavaScript and will not discussed in detail.

6.1. Session Management & Interactive Services

6.1.1. Runtime Environment

The common runtime environment used to implement the *IPTV Session Management Enabler*, *IPTV Meta Session Management* (Social IPTV), as well as specified scenarios for *Targeted Advertisement* insertion and the *Virtual Quiz Show* resides on the concept of convergent SIP/HTTP Servlets. These Servlets are programmed in Java technology and can be executed on any *Application Server* fulfilling the corresponding specifications for SIP[2] and/or Java EE HTTP Servlets[3].

[1] http://www.videolan.org
[2] SIP Servlet Specification 1.1 (JSR289)
[3] Java Servlet Specification 3.0 (JSR315)

Servlets are server-side objects that process incoming requests and send an appropriate response to the client. In the case of an IPTV scenario, this client might be a Set-Top-Box (STB), TV or even a mobile device. Servlets are deployed in a so-called Servlet container that manages resources to related or integrated components and technologies like databases and network stacks.

In the case of convergent SIP/HTTP Servlets, both the SIP and the HTTP protocol are supported resources. The corresponding JSR specifications contain information about the basic requirements to fulfill including:

- An API for the SIP as well as HTTP programming model
- The responsibilities of the SIP/HTTP Servlet container
- Information about how SIP/HTTP Servlets interface with other Servlets and Java EE components

Figure 6.1 illustrates a reference design for a convergent SIP/HTTP Servlet in an IPTV environment. The *End Devices* are connected either through the SIP or HTTP protocol or via both to the *Application Server*. Incoming messages are then forwarded to the corresponding SIP or HTTP Servlet container.

The *Servlet Context* defines specific context attributes used to store and retrieve information specific to a Servlet and interfaces from the context. The context can be shared with other Servlets within the same application.

The *Deployment Descriptor* (SIP.xml/Web.xml) is based on an XML file and is used to describe a Servlet. This description contains information about how a specific Servlet is invoked, as well as used environment properties and other resources.

Convergent applications like the interactive services and *Meta Session* service specified in the last chapter may use both a SIP and HTTP Servlet to create a service. Through a so-called *Convergent Context*, information can be exchanged and resources can be shared.

In the scope of this thesis, this concept has been used to create all services described in this chapter. An implementation of such a convergent Application Server is the *SailFin* Project[4] which has been used throughout work on this thesis. *SailFin* is a sub project of the Open Source Application Server project *GlassFish*[5].

6.1.2. Composite Java Application Library

Modular and loosely-coupled application design, e.g. following a *Model-View-Presenter* (MVP) pattern, represents the current state-of-the-art in software engineering. Different frameworks have been implemented, or are already part of various programming languages nowadays.

[4] http://sailfin.java.net/
[5] https://glassfish.dev.java.net/

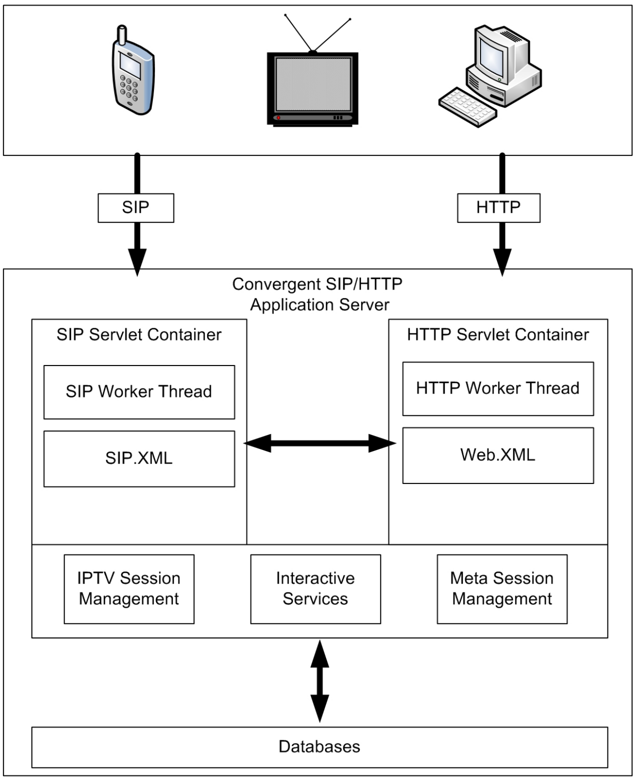

Figure 6.1.: Convergent SIP/HTTP Servlet Model

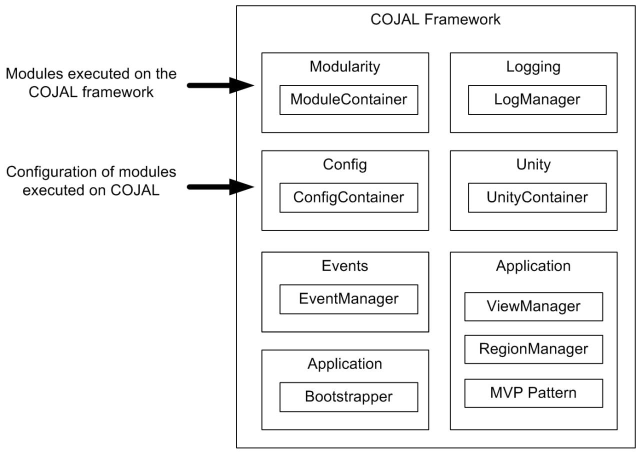

Figure 6.2.: Composite Java Application Library Overview

In the realm of this thesis and related projects, it was decided to use the *Composite Java Application Library* (COJAL).

The *COJAL* framework as visualized in Figure 6.2 supports composite Java application development. The library, developed at Fraunhofer Institute FOKUS, is inspired by Microsoft's *Composite Application Library* (CAL)[6].

A COJAL instance is being configured in the *Modularity* and *Config* container allowing for adjustments to module registration and module dependency management and other application-related information. Furthermore, COJAL offers a eventing mechanism that allows events to be spread across all modules. Furthermore, User Interface (UI) Support synchronizes user interface threads with the rest of the application. A bootstrapper is available for application startup, a *Log Manager* for application-related logging and a so-called *Unity Container* able to carry common services and share these services with other modules.

6.1.3. Component Interaction

By using the convergent SIP/HTTP Servlet environment and a composite framework for application development, namely the COJAL framework, developed application logic can easily be implemented and deployed. Figure 6.3 shows the resulting infrastructure, including deployed Servlets for *IPTV Session Management Enabler*, *IPTV Meta Session Management*, the *Targeted Advertisement* service and the *Virtual Quiz Show* service.

Furthermore, four other modules providing common and general communication functionalities, and allowing for interaction in between the components, have been developed and are jointly used by the above-mentioned services.

- The *Common Module*, is a library referenced by all other modules. It contains all shared and common objects that need to be accessible throughout the entire implementation. Global commands, global event definitions and payloads, commands and common-used services are stored in the common module.
- The *Web Communication Module* deals with incoming requests using the HTTP stack used by OTT clients or Web browser-based service maintenance. Furthermore, the *IPTV Session State* API is implemented using this module.
- The *NGN Communication Module* manages communication towards the managed IPTV network. Incoming SIP requests are evaluated by this component and forwarded to the corresponding application logic.
- The *Database Module* provides common functionality for database access. This includes wrappers for mapping application data to database schemas as well as the corresponding standard connectors, e.g. JDO or JDBC.

[6]http://msdn.microsoft.com/en-us/library/ff647752.aspx

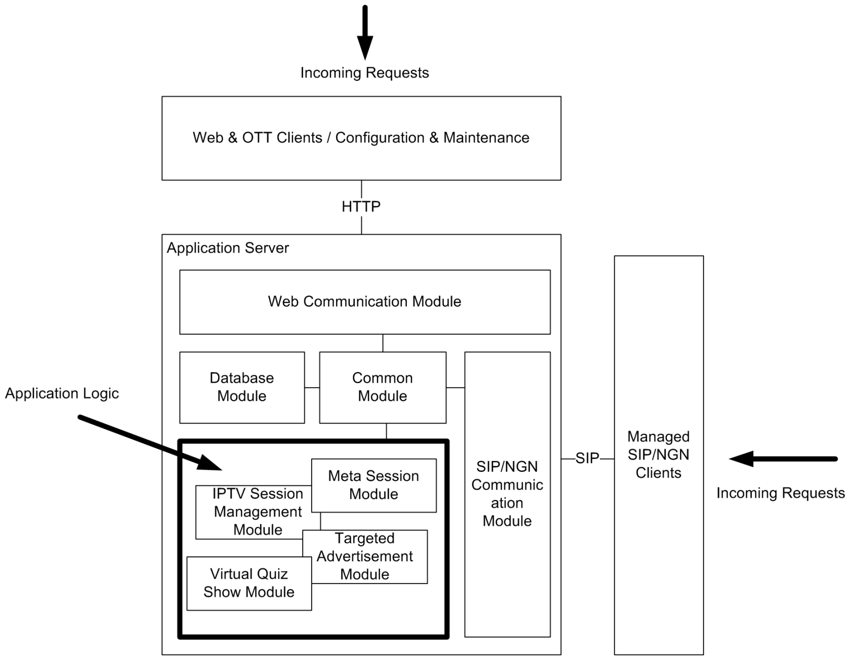

Figure 6.3.: Servlet Deployment & Component Interaction

6.1.4. Application Model

Each service, e.g. the *IPTV Session Management Enabler,* has been developed according to the guidelines and principles discussed within the last sections. Taking a deeper look into the application logic itself, this section finally describes the application model for the *IPTV Session Management*. The other services have been implemented analog to their corresponding specifications.

Figure 6.4 provides an overview of the implemented structures and corresponding data models. Beside the *Common Module*, *Database Module* and the two *Communication Modules* for Web and NGN communication, the application logic is subdivided into general, enhanced and so-called repository classes for database access.

The general classes provide functionality for starting the service to maintain and log IPTV sessions.

The enhanced classes are responsible for implementing the session handling itself. According to the corresponding data model, each session, representing a user's service request, contains exactly one single *Media Item*. A *Media Item* corresponds to a TV channel or a video stored in the connected content management system. Furthermore, each session is mapped onto a *Service Type* object. This service object describes the service type, e.g. Live TV or Video-on-Demand. Finally, each *Media Item* can be mapped onto a *Media Container*. A *Media Container* collects multiple *Media Items* to a specific *Service Type*. *Media Containers* are used to build channel bouquets, which are made available during service discovery processes whenever an *End Device* is started and searches for available services.

The repository classes implement all aspects necessary for the storage and retrieval of session-related data from a connected database or local memory.

Database access is necessary whenever a user requests content in the form of a Live TV channel, or a video to resolve corresponding usage rights, or the URL pointing to the media server hosting the content.

Furthermore, data is written to the databases during a session allowing interactive scenarios implemented using the *IPTV Session State* to be driven, in order to restore a session upon system errors or during session handover scenarios to other devices.

After a session is terminated the session data is written to a log file. This data is used to create statistical data concerning service and application usage.

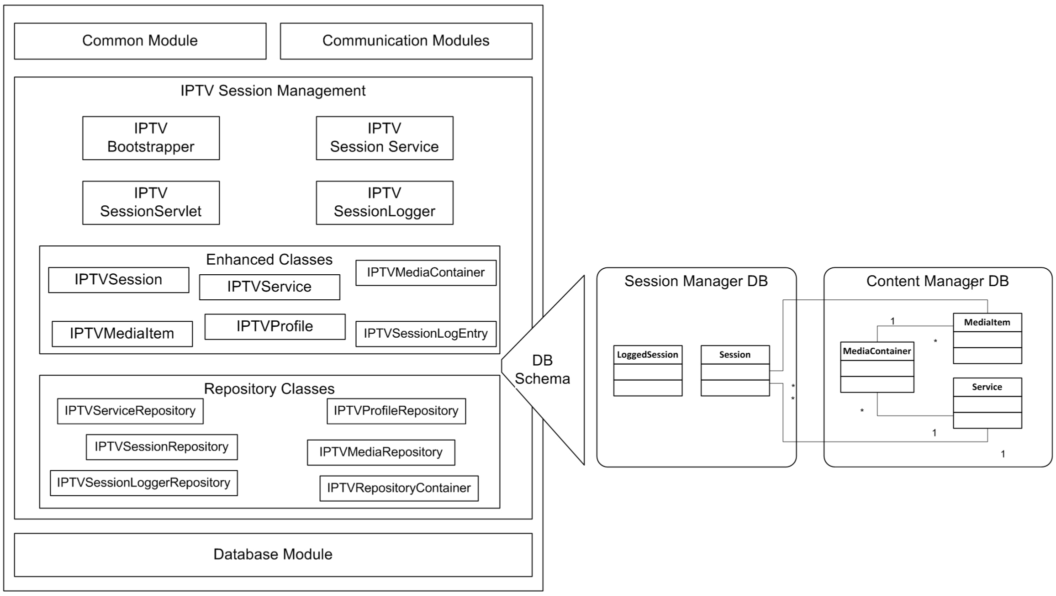

Figure 6.4.: IPTV Session Management Enabler implementation: Object & Data Model

6.2. Interactive Content Enabler Prototype

This section describes the modifications to the Open Source project VideoLAN to enable interactive content provisioning in the context of interactive and Social IPTV scenarios. The goal of this work was to create a component providing as much flexibility as possible with regards to the supported scenarios. The main focus has been set on the signaling of the scenarios. For this reason, the VideoLAN Project [133] has been chosen for all media adaptation and streaming issues, as it already provides a complete functionality on this level.

6.2.1. Technology Mapping

The implementations for the *Interactive Content Enabler* reside on two Open Source projects, which have been combined for this purpose: The *VideoLAN Project* and the Open Source SIP stack *EXopsip2*, which has been integrated for session signaling purposes.

6.2.1.1. VideoLAN Project

VideoLAN is an open-source media player and server maintained by the VideoLAN Project[7]. It is a portable multimedia player, encoder, and server, supporting many audio and video codecs, file format and various streaming protocols. It can be used to stream files over networks, to transcode them in real-time and synchronize playback of multiple clients. It is platform-independent with versions for every major operating system and written in the C programming language.

Figure 6.5 provides a schematic overview of the VideoLAN. Supported input types, streaming engines, and client platforms are visualized.

6.2.1.2. SIP Stack

The Open Source project *EXosip*[8] has been used for all session related issues. *Exosip2* is a simple and easy-to-use SIP stack that can be seen as an extended version of the more complex and flexible oSIP2 library from the same project.

Exosip2 hides the complexity of the SIP protocol for multimedia session establishment. It is relatively small in terms of code lines, and provides an API allowing for a straight implementation of SIP user agents including SIP-phones and SIP-Proxies.

6.2.2. Content Composition

The *VideoLAN Manager* (VLM) is a media manager inside VLC that is designed to control multiple streams with only one instance of VLC. It allows for multiple parallel streaming sessions, which is particularly necessary for the ICE to be able

[7] http://www.videolan.org
[8] http://www.antisip.com/

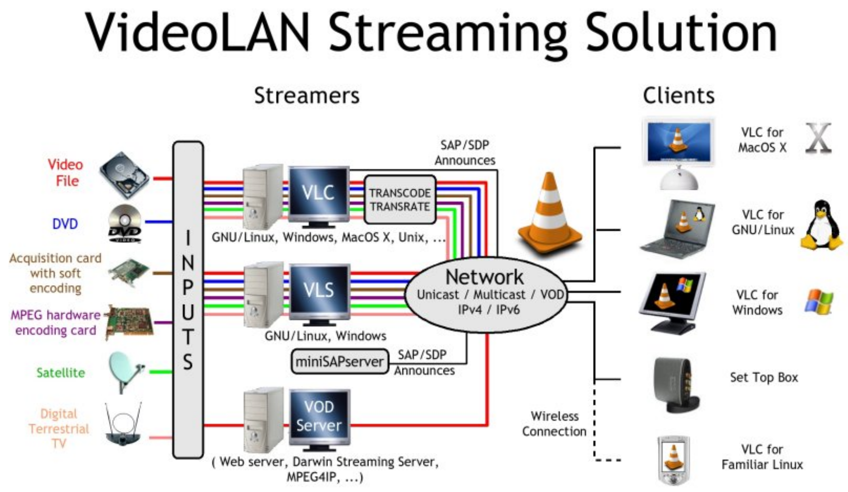

Figure 6.5.: Overview of the VideoLAN Project [133]

to serve a large number of clients concurrently. VLM allows for the composition of dynamic contents allowing the creation of a chain constructed by:

- An *Input Element* (video and audio files or and network video/audio streams).
- a corresponding *Output Element*, defining how and where a content should be streamed and
- different *Actions* in between, e.g. for transcoding, transration or mixing.

The VLM can either be triggered via multiple protocols like Telnet and a HTTP interface, or by using a low-level API.

This API was used in the context of the implementation described here, and was connected with the SIP stack presented above.

Figure 6.6 shows the structure of the designed application, combining both Open Source projects.

The designed extensions have been added in the form of a plug-in to the VideoLAN and are named *VideoLan Core*.

When the plugin is loaded and started, the *Interactive Content Enabler* (ICE) is listening for incoming SIP requests. Upon reception of a SIP INVITE message, the ICE identifies the message and decides, according to the information carried inside the SDP, which streaming service is actually requested by the *End Device*.

The ICE then uses the so-called *Advanced Streaming API* to start the appropriate streaming function running the corresponding logic. The *Advanced Streaming API*

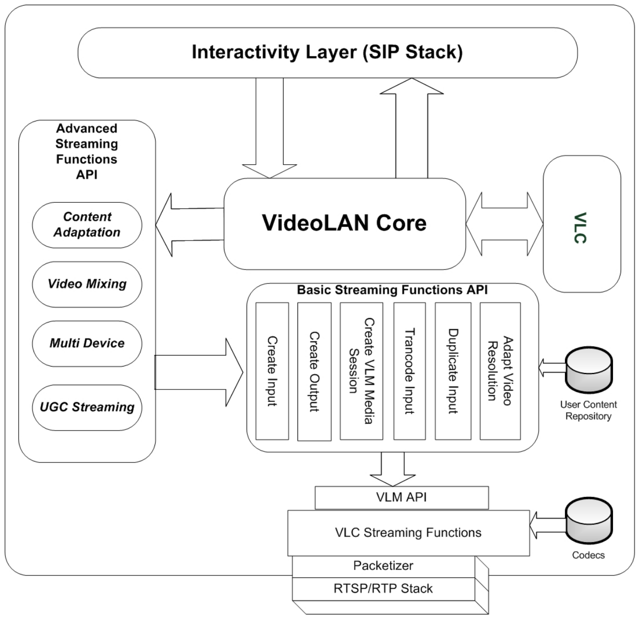

Figure 6.6.: Interactive Content Enabler Prototype Architecture

in turn makes uses of a second more basic API, the *Basic Streaming API* that realizes the stream pipeline towards the requesting clients.

In the following, the realized modules and APIs will be described in more detail.

6.2.3. VideoLAN Core

The *VideoLAN Core* is the central component within the ICE. It implements the required functions in order to start, run or stop the plug-in.

It provides all descriptive information about the SIP plug-in itself and its parameters. It is also the place all plug-in parameters are parsed in order to set up the server configurations. Additionally, the VideoLAN Core uses a service configuration file in order to take into account specific service information.

The most important process in the VideoLAN Core is the input loop waiting for requests. It waits for SIP messages from the SIP stack, originating from the *IPTV Session Management Enabler* and End Devices. It uses the *Advanced Streaming API* to trigger the required streaming service, as well as functionalities of the SIP Utilities component that cover more specific SIP-related tasks.

Because of its centralized loop, the VideoLAN Core is single threaded, which implies that just a single request can be handled at a time. While this design option might appear to be inappropriate for an application intended to serve a theoretically unlimited number of *End Devices* asynchronously, it has the advantage of keeping the server easy to implement, preventing unexpected bugs. Furthermore, in case of high load on the ICE, load balancing mechanisms could be implemented on the *IPTV Session Management Enabler* connected to multiple independent ICE instances. Finally, it is worth mentioning that only the signaling part of the ICE is single threaded. The media delivery part however, reuses the multi-threading capabilities of VLC to serve multiple stream *End Devices* simultaneously.

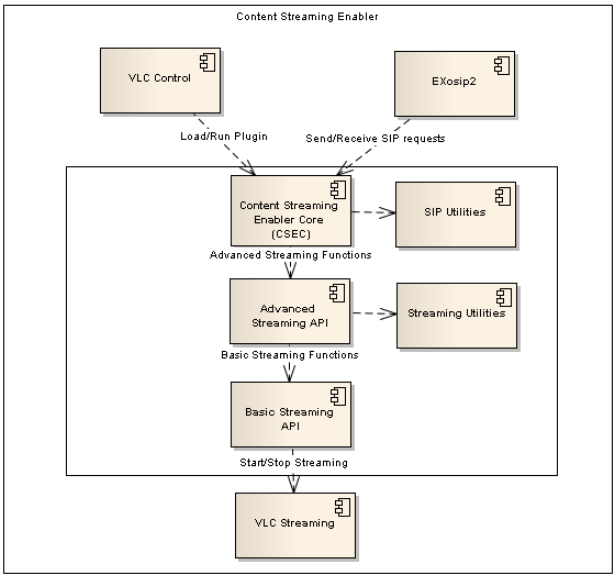

Figure 6.7.: Interactive Content Enabler Prototype

6.2.4. Advanced Streaming API

The Advanced Streaming API (AS API) is composed of a set of functions covering all streaming services available on the ICE. It provides high-level functions to implement the server streaming services, wrapping all necessary steps in the handling of the media streaming session. Additionally, it performs tasks necessary to support basic administration and monitoring functionalities.

The implemented functions allow for the following scenarios:

- Content Adaptation, allowing for on-the-fly transcoding and transration of content, e.g. from a high to a low bitrate or from one video codec to another.
- Video Mixing, allowing for the combination of multiple input sources into a single stream, e.g. used for the *Virtual Quiz Show* scenario.
- Multi Device Streaming, allowing for streaming content in multiple formats to multiple outputs.
- UGC Streaming, allowing the fetching of User-Generated-Content sources.

6.2.5. Basic Streaming API

Each of the *Advanced Streaming API* functions leverages the streaming functionalities of the *Basic Streaming* API (BS API). The BS API is a low level API wrapping the VLM API. It is composed of functions providing a direct access to VLM and allowing specific tasks in the process of creation of media sessions towards the requesting user.

- Create Input, allowing for the selection of a specific input source, e.g. a URL a video file, a DVB device.
- Create Output, allowing for the creation of an output stream
- Create VLM Media Session, allowing for the creation of a managed VLM session
- Transcode Output, allowing for the modification of a content's video codec
- Duplicate Output, allowing the streaming of content to multiple destinations.
- Adapt Video Resolution, allowing for the transration of a video on-the-fly.

7. Measurements & Validation

Chapter 7: Measurements & Validation
Validation Criteria for Interactive IPTV
Test Bed
Case Studies

This chapter re-addresses the concepts and scenarios discussed and specified in the last chapters and demonstrates their applicability to laboratory and partial real world conditions.

The main purpose of this chapter is to show that the ideas fit the requirements of technical feasibility, as well as user expectations.

First, validation criteria for *Interactive IPTV Services* will be defined and mapped onto the different tests and case studies described in this chapter.

Various – at first glance proprietary and incompatible technologies – will be combined and coordinated under the umbrella of the proposed architecture from Chapter 4 and shown in interaction.

The outcome is then used to validate this work against products already on the market or facing upcoming release.

7.1. Validation Criteria for Interactive IPTV Services

The evaluation of convergent technologies like IPTV that combine different technology domains, including telecommunication, broadcast and *Social Media* services, represents the most significant challenge and often the most complex to test and verify.

For this reason, it is of utmost importance to concentrate on certain specific aspects of the *System Under Test* (SUT). In the following, the testing methods and corresponding test metrics used will be described.

7.1.1. Testing Methods & Targets

Feasibility Testing Practical feasibility tests have been used during work on this thesis as an early tool for the verification of each specification through implementation. Most of the time, an early proof-of-concept implementation has been created, allowing for the conducting of tests on functionality, usability and the taking of initial measurements.

Feasibility tests represented the most important tool for making decisions on next steps and future developments. Furthermore, demonstrators and prototypes developed in this phase have been used to demonstrate concepts and technologies during academic conferences and corresponding demo sessions. Feasibility tests can be considered a combination of performance, usability and integration testing.

Performance Testing Where applicable, detailed performance tests have been conducted in order to analyze implemented systems under laboratory and test bed conditions. As described above, the metrics for IPTV performance tests have to be selected very carefully, to allow for a comparison with other approaches. For this reason, performance tests have been limited to aspects of:

- Application Server and database performance when testing *IPTV Session Management Enabler.*
- Channel or content switching delay when testing *Targeted Advertisement* (TAD) scenarios.
- Media Server performance when testing the *Virtual Quiz Show* scenario.

Integration Testing The systems and prototypes developed in the course of this thesis garnered a high level of interest whenever integration tests with other research groups have been conducted. In more detail, this was applicable especially to the research projects funded by the European Commission, and during standardization activities. This included official interoperability events[1] as well as IPTV proof-of-concept activities[2] conducted with industry partners.

During these interoperability events, implementations according to contributions made to *Standard Development Organizations* (SDOs) have been tested against other equipment, as well as introducing new features.

Usability Testing Usability Testing allows for the evaluation of a system or service by testing it with users [96]. In the course of this thesis, no systematic usability tests have been undertaken by the author's research group itself, nonetheless the developed features and services had a strong user focus. For this reason, different activities concerning usability testing have been conducted with external partners, from the broadcasting and media industry.

In Table 7.1 each *System Under Test* (SUT), as derived from the specifications and implementations from Chapters 5 and 6 has been listed. Furthermore, for each SUT an analysis of how much focus has been put onto a dedicated testing method has been performed.

[1] http://www.oipf.tv/PRESS/pressrelease_160610.html
[2] http://www.oipf.tv/PRESS/pressrelease_07092010_Bis.html

	Feasibility Test	Performance Test	Usability Test	Integration Test
Session Management	++	++	−	-
Targeted Advertisement	++	+	+	-
Virtual Quiz Show	++	+	+	-
iSession/ Meta Sessions	++	-	+	++
IPTV Session State API	++	-	-	++
Singapore Proof-of-Concept	+++	-	+	+++

Table 7.1.: Systems Under Test (SUT) & Testing Method

7.1.2. Targets for Feasibility Testing

During work on the prototypes for the specifications made in Chapter 5, each development has been subject to feasibility testing. Feasibility tests were conducted inside an overall test bed described later. Normally, new functionalities have been added as a new Application Server component, an update to an existing entity or as a new *End Device* integrated into the test bed.

7.1.3. Targets for Performance Testing

As described above, performance testing has been applied to three dedicated entities within the overall test bed, which play critical roles with regards to performance and real time aspects when building interactive IPTV services. This includes the:

- IPTV Session Management Enabler
- Targeted Advertisement Service (TAD)
- Virtual Quiz Show

Each service has different performance requirements, which will be discussed in the following in more detail:

7.1.3.1. IPTV Session Management

During testing on the *IPTV Session Management Enabler* it has been concentrated on measurements of signaling delays, Application Server and database performance. First, tests were limited to pure feasibility tests, then later extended to performance testing, checking the applicability of the specifications made. This included the session-oriented *IPTV Role Model* as specified in Chapter 2 and the *Third Party Access* mechanism introduced in Chapters 4 and 5.

7.1.3.2. Targeted Advertisement

This evaluation allows for the measurement of delays on media level, i.e. switching between a Live TV channel received via IP multicast and a sequence of ads received as unicast streams. All switching took place on the client.

The targeted advertisement scenario is equal to a channel change scenario but from multicast to unicast and back. Indirectly, it also allows for measurements of the IP multicast switching performance. SIP session setup delays have not been analyzed, but the delays during the process of joining a multicast channel and ads streamed via RTSP/RTP or HTTP have.

7.1.3.3. Virtual Quiz Show

This test setup has been used to analyze media server performance. In more detail, this test case analyzes the content adaptation and mixing capabilities of the *Interactive Content Enabler*. The practicability of server-side content mixing has been analyzed by measuring the delays for server side *User Generated Content* insertion. This scenario can also be used to predict server side ad-insertion performance as an alternative to the client side ad insertion scenario from the last section.

7.1.4. Targets for Integration Testing

Integration tests allow for testing the interaction with other systems and components, e.g. other (commercial) IPTV middleware systems, commercial and non-commercial clients. The main emphasis has been put on testing standards compliance with regards to Open IPTV Forum Release 1 and 2 specifications. Furthermore, commercial TV sets from Phillips and Samsung, running a CE-HTML browser and the so called Yahoo! Widget Channel have been tested in two other case studies. The three case studies discussed in this context can be summarized as follows:

- Singapore Proof-of-Concept; Integration test of the *Open IPTV Ecosystem Core* with other commercial middleware systems (Ericsson IAP), commercial TV sets (Philips) and browsers from Opera and ANT.

- IPTV Meta Session & Social TV features in iNEM4U; Integration test of the *Meta Session* from the concept, as specified in Chapter 5 with semi-commercial TV prototypes from Philips in the scope of a European research project.

- Third Party Access API: Integration and feasibility test of the *IPTV Session State API* as specified in Chapter 5, using corresponding implementations from Chapter 6. Client systems and own prototypes from Samsung and a Microsoft Mediaroom test client have been interconnected, showing a user's cross domain *IPTV Session State* on multiple devices.

7.1.5. Targets for Usability Testing

Systematical usability tests are beyond the scope of this thesis and might be a target for future work.

7.2. Overall Test Bed

The *Open IPTV Ecosystem Core* has been under development since 2006 and represents the author's and his research group's practical work on IPTV.

The system has been revised several times and consequently, it has served as the basis for various demonstrations and evaluations. It therefore has served to support the continuous testing and revising of ideas and concepts. During the time this thesis was written, a broadness with respect to available technology from the IPTV domain was the main unique characteristic of the *Open IPTV Ecosystem Core*. However, the main focus has now been steered to standardization, openness and innovation. From an evolutionary point of view, the lab is advancing towards an infrastructure that mainly emphasizes NGN and IMS aspects within IPTV. Nowadays, a mixed environment that also includes Web and future Media Internet aspects is available. At this point in time, the developed infrastructure has reached a certain level of maturity and is now recognized as a reference platform for the most important worldwide IPTV standardization activities, namely ETSI TISPAN and the Open IPTV Forum. The next section will provide an overview of the lab's architecture from both the client and infrastructure perspective.

7.2.1. Lab Architecture

The demonstration platform can be subdivided into the client environments described in the next section and the back end described later. Figure 7.1 provides an overview of both sides.

7.2.1.1. End Devices

One key element in the building of an *IPTV* ecosystem lies in the design and development of applications for the user front end, called *End Devices* in the scope of this thesis. Within the *Open IPTV Ecosystem Core*, these front ends can be divided into two categories:

- Thick clients and applications running on PCs or STBs that implement protocol stacks and required functionality in native code. Mostly used for managed network approaches requiring a SIP stack.

- Thin clients or, running browser-based applications hosted within the network and executed inside a Web browser. Mostly used for Over-The-Top (OTT) services

Component / Entity	Description	Interface
Managed End Device	NGN/SIP enabled client. Either .NET implementation or simulated through SIPp	SIP/HTTP
OTT End Device	Browser based End Device; e.g. Philips NetTV	HTTP
Open IMS Core	Open Source IMS, partly used to forward SIP messages and user authentication	SIP
IPTV Session Management Enabler	Prototypical implementation of the IPTV Session Management as a convergent SIP/HTTP Servlet	SIP/HTTP*
Meta Session Application Server	Server-side Implementations of Meta Session functionality for Social IPTV scenarios (SIP/HTTP Servlet)	SIP/HTTP*
Targeted Advertisement Application Server	Server-side business logic and implementations for Targeted Advertisement scenarios (SIP/HTTP-Servlet)	SIP/HTTP*
Virtual Quiz Show Application Server	Server-side business logic and implementations for Virtual Quiz Show scenario	SIP/HTTP**
Interactive Content Enabler (ICE)	Enabler for interactive content provisioning capabilities.	HTTP/SIP/ RTP/ RTSP

*Signaling via HTTP using Webservices/JSON-RPC and management console
**Management console

Table 7.2.: Components of the Open IPTV Ecosystem Core Test Bed

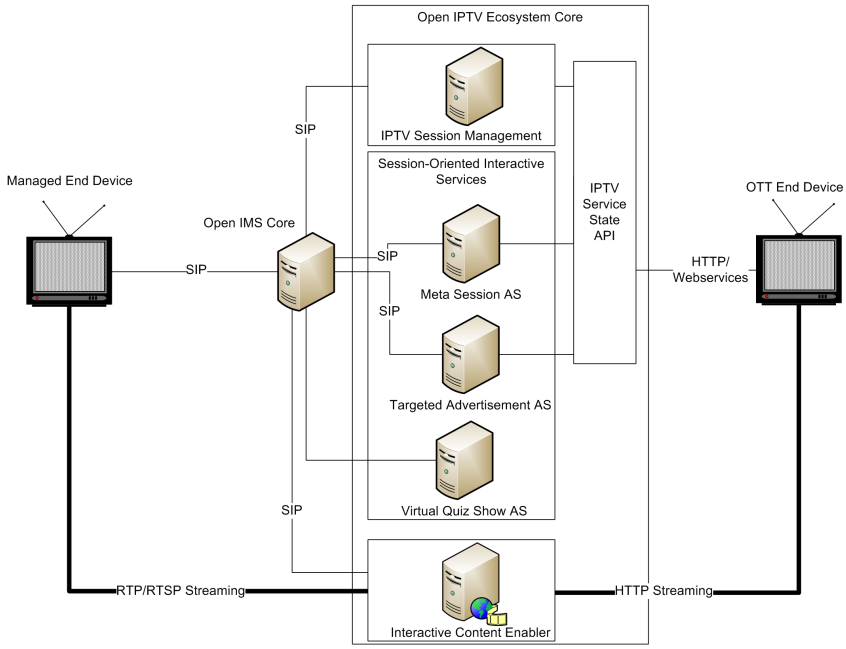

Figure 7.1.: FOKUS Open IPTV Lab Architecture

While the lab's prototypes implement both approaches, the latter category represents the more current development. This follows the latest trends in industry and academia. Currently available implementations have been realized and connected to the infrastructure described in the last section:

- .NET based native IPTV application for PC environments.
- Linux based IPTV client implemented in C and C++.
- browser-based IPTV clients in Silverlight, CE-HTML and JavaFX and HTML5.

Future work will concentrate on HTML5 and corresponding technologies like Rich Internet Applications.

7.2.1.2. Infrastructure & Services

The lab's infrastructural components can be classified into:

- The signaling infrastructure, using either managed NGN/IMS approaches or Over-The-Top (OTT) Internet-oriented signaling schemes, and based mainly on HTTP and Webservices.

- The different service enablers for interactive services running in a *Session-Oriented Application Environment* (SAE) and the *IPTV Session Management Enabler* described in Chapters 4, 5 and 6 of this thesis.
- The delivery infrastructure called *Interactive Content Enabler* (ICE) for providing static content or dynamic delivery schemes.

Table 7.2 collects all components including *End Devices* and network side enablers, including a brief overview of available interfaces.

7.2.1.3. System Tests

By making the test bed description from the last section available, the following section concentrates on feasibility and performance tests conducted on elements of the infrastructure beforehand. First performance tests on the *IPTV Session Management Enabler* will be discussed. This is followed by an analysis of measurements of *Targeted Advertisement* discussing content switching issues. Finally, the *Virtual Quiz Show* service is analyzed with regards to server-side content adaptation and mixing.

7.2.2. IPTV Session Management

As described numerous times during the design and specification of services within the last chapters, the *IPTV Session Management Enabler* acts as the central database, managing all user requests and exposing them to third party.

Having this information available, it enables the full set of interactive IPTV services, from basic behavioral user tracking to the most sophisticated services, e.g. *Targeted Advertisement* scenarios or *Social IPTV* through *IPTV Meta Sessions.*

For this reason, it is indispensable to prove the feasibility of specifications and implementations of the *IPTV Session Management Enabler* under laboratory conditions. In this section, the *IPTV Session Management Enabler* functional entity will be therefore challenged with a number of test cases allowing for the derivation of a proof of feasibility. These tests have been limited to tests on SIP session setup and signaling, i.e. no tests on a media session level have been conducted, however they will be conducted in the next test case of *Targeted Advertisement* insertion.

7.2.2.1. Testing Scenario Setup

Testing the implementations for the *IPTV Session Management Enabler* requires the definition of a controlled environment allowing for the creation of reproducible test results. Figure 7.2 shows such a test environment, including three options for performance testing:

1. A setup consisting of the full end-to-end signaling chain for session-oriented IPTV. This includes a test computer running a SIP traffic generator, namely

SIPp[3] allowing for the simulation of user requests for IPTV services. Furthermore, an instance of the *Open IMS Core* and the *IPTV Session Management Enabler*. The *IPTV Session Management* can be subdivided into the Servlet container managing incoming requests on protocol level, the application logic managing request on service level and the Session Repository acting as a persistent or non-persistent database for all managed sessions.

2. Testing setup (2) contains the same components as (1) but no instance of the *Open IMS Core*. In this setup, SIP messages are sent directly from the test tool to the *IPTV Session Management Enabler*.

3. Testing setup (3) is meant for directly testing the interface from the application logic to the connected database or local memory.

For reasons of simplification and to keep the number of variables as small as possible, it was decided to use test setup (2) and (3) for the performance tests described in this section. This implies that the SIP-based signaling does not pass the Open IMS Core but is sent directly to the *IPTV Session Management Enabler*.

Furthermore, and as depicted in Figure 7.3, another variation of the tests has been realized through switching between different *Session Repositories*. Option (a) represents a non-persistent repository using the Application Server's local memory to store IPTV session data. Option (b) uses a relational database, i.e. a plain MySQL in this specific case, where option (3) connects to a object-oriented database. All tests have been repeated with a different *Session Repository*, respectively.

Figure 7.4 describes the exact test procedure on signaling level. The test case therefore consists of the following test messages generated by the test tool and corresponding internal messages over the Java Data Object (JDO) interface:

1. A SIP INVITE message is sent from the test tool to the IPTV Session Management Enabler

2. The *IPTV Session Management Enabler* evaluates the incoming request and reads information concerning the requesting user' access rights , the selected service and the specific media item from the database (three steps depicted as a single message in the diagram). In parallel, a *100 TRYING* message is sent back to the test tool, indicating the ongoing evaluation.

3. The connected database responds with information about the user's access rights, service information and a media URL, if necessary.

4. The *IPTV Session Management Enabler* creates a new session object.

5. A *200 OK* message carrying the information from the previous step is sent back to the test tool.

[3]http://sipp.sourceforge.net/

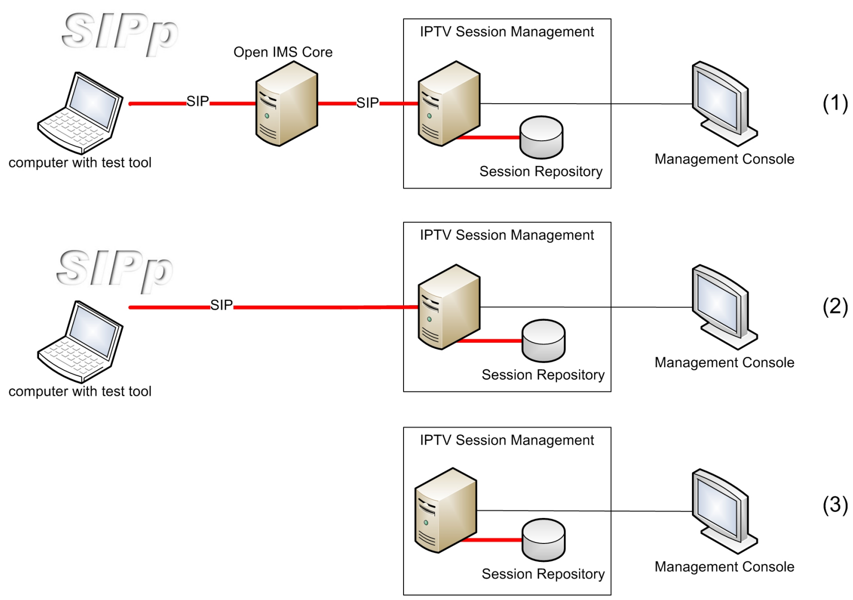

Figure 7.2.: Options for System under Test; IPTV Session Management Enabler

6. The test tool issues an *ACK* message upon reception.
7. A log entry is added to the database.
8. The test tool sends a session termination request in the form of a *BYE* message
9. The database state is updated on the *IPTV Session Management Enabler*
10. A 200 OK message is sent back to the test tool. The IPTV session is terminated.

7.2.2.2. Results

The performance and feasibility analysis conducted on the described environment has been conducted according to the testing scheme introduced in the last section by issuing a certain number of IPTV service requests from the test tool to the *IPTV Session Management Enabler*, and the connected persistent or non-persistent database or memory, respectively. Another variation has been introduced by modifying the number of service requests per test run.

The SUT is based on a Windows 2008 Server 64 Bit running on a IBM x3650, Typ 7979 machine. The system consists of 2x Intel Xeon E5420@2.5Ghz Quad-Core CPU with a total of 16 GB RAM.

Figure 7.5 shows the average results of the conducted system test, i.e. IPTV service requests issued towards the *IPTV Session Management Enabler*. Results

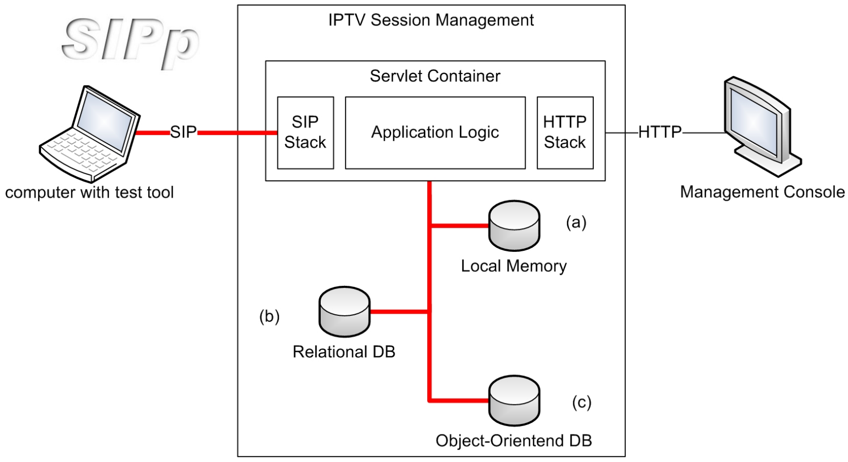

Figure 7.3.: Options for System under Test; Switching Session Repositories

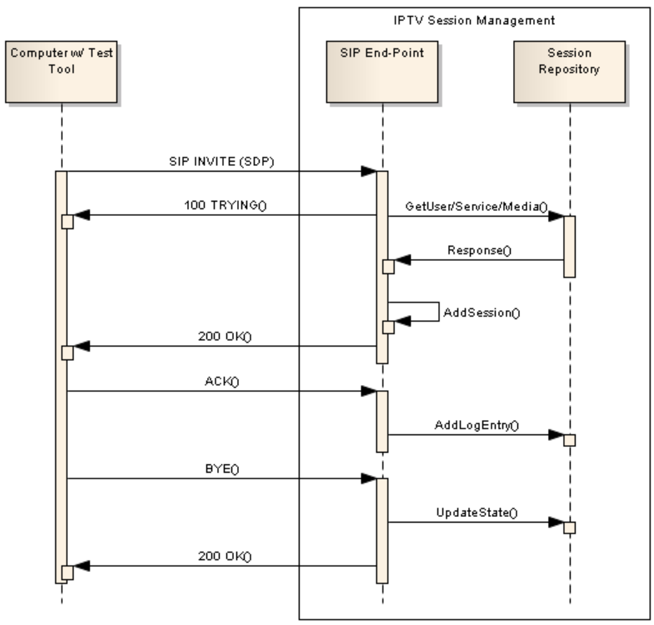

Figure 7.4.: SIP and Internal Application Level Signaling During Test Case

#User	Local Memory		Relational DB		Object DB	
	T_I	T_E	T_I	T_E	T_I	T_E
1	3	57	25	63	15	61
100	5	60	31	68	20	66
10.000	4	59	47	88	27	77

Table 7.3.: Session Setup and Termination Performance Measurements

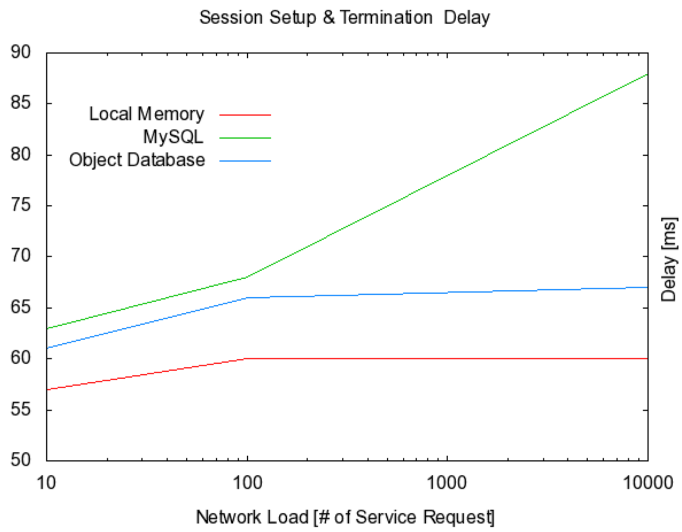

Figure 7.5.: Measurement of IPTV Session Setup and Termination Delay Local Memory, Relational Database and Object-Oriented Database using different Load Levels.

have been collected on a SIP signaling level (T_E) and for internal computation time (T_I) aggregating the delay for each single message. Furthermore, this data has been collected for all three storage types separately. The resulting graph, depicted as Figure 7.5, shows the results for measurements on a signaling level.

Taking this into account, the signaling delay has been proven to be between around 60 and up to 90ms and therefore fulfills the requirements under laboratory conditions.

Even with the number of users increased to a certain limit, the local memory approach (red line) and the object-oriented database (blue line) show nearly constant behavior, while the relational database shows some performance limitations.

7.2.3. Targeted Advertisement Insertion

As described and specified in Chapter 5 a *Targeted Advertisements* (TAD) service allows for providing personalized advertisements clips to single users or groups of users sharing the same interests. From a technical perspective, two different options have been discussed in the corresponding chapter:

- Client-side ad-insertion, done by the *End Device* based on provided metadata and corresponding resources to advertisements.
- Server-side ad insertions managed by a media server.

Both solutions have their advantages and disadvantages, and will not be discussed here again but rather the corresponding prototypical implementation for client-side *Targeted Advertisement* as specified in Chapter 5.

7.2.3.1. Test Bed

Figure 7.6 shows the subsystem of the *Open IPTV Ecosystem Core* used for the test case. It contains a managed *End-Device* using a Home Theater PC (HTPC; Acer Aspire Idea 510, Intel Core Duo 1,6GHz, 2GB RAM) running a Windows Vista Enterprise environment. A .NET based NGN-IPTV client implementing the necessary client-side application logic for the targeted advertisement scenario has been connected to the infrastructure using a dedicated 100 MBit/s Ethernet connection through a private IP network. The client furthermore implements a NGN/SIP-stack, allowing for communication with the Application Servers in the network. All media handling is implemented through a modified *VideoLan* library used by the .NET NGN-IPTV client. Measurement points have been implemented in the source code.

The infrastructure consists of an instance of the *Open IMS Core*[4] used for SIP signaling purposes. Furthermore, the *IPTV Session Management Enabler* and *Targeted Advertisement* AS use the same machine represented by an IBM Blade Server System X3550 Xeon DP 2.0GHz running Windows Server 2003 and the *Project SailFin* convergent SIP/HTTP Application Server. The *Interactive Content Enabler* serving Live TV channels via IP multicast and advertisements via unicast is running on a dedicated server with the same specifications, but a Linux Operating System.

7.2.3.2. Add Insertion Testing Scenario

To test the media switching delay when receiving a targeted advertisement and switching back from the targeted ad to the Live TV channel, the following test procedure was developed and conducted 30 times for each test run. Figure 7.7 shows the test setup procedure:

[4]http://www.openimscore.org

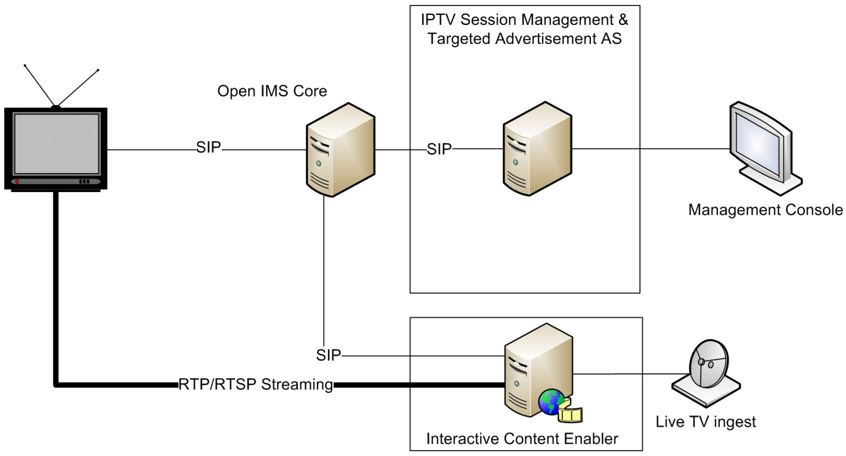

Figure 7.6.: System under Test for Targeted Advertisement

1. Pre-Condition: The .NET-based IPTV client is started on the *End Device*. The test user has joined an IPTV channel in MPEG2 format and MPEG2-TS container format, directly feeded via IP multicast into the test network. A corresponding SIP session has been set up, allowing it to bind a SIP INFO message with corresponding advertisement metadata. Measurement points inside the client's source code are fired in two cases:

   - Received targeted advertisement metadata and wall-clock time are matched. In this case, the client switches from Live TV to an advertisement. The switching time T_{MU} is computed automatically using the following equation and written to a log file.

   $$T_{MU} = t_{MU2} - t_{MU1} \tag{7.1}$$

   - The second timer is triggered whenever the client reports an *isPlaying* state back to the test code. The switching time from unicast back to multicast T_{UM} is computed as follows and also added to the corresponding log file.

   $$T_{UM} = t_{UM2} - t_{UM1} \tag{7.2}$$

2. A test manager triggers a targeted advertisement in the managements console. Metadata containing exact switching times and and a URL for the advertisement clip is transmitted as a SIP INFO message.

3. The client's application logic analyzes the received metadata and starts to request the targeted advertisements when wall clock time and metadata match. The first measurement (t_{MU1}) is written to the log file upon request, the second (t_{MU2}) when the player reports the *isPlaying* event.

4. When the advertisements clip has reached its end, the client initiates a switch back to the Live TV signal (t_{UM1}), by joining the corresponding multicast resource again. t_{UM2} is taken when the client reports a *isPlaying* event for the multicast channel.

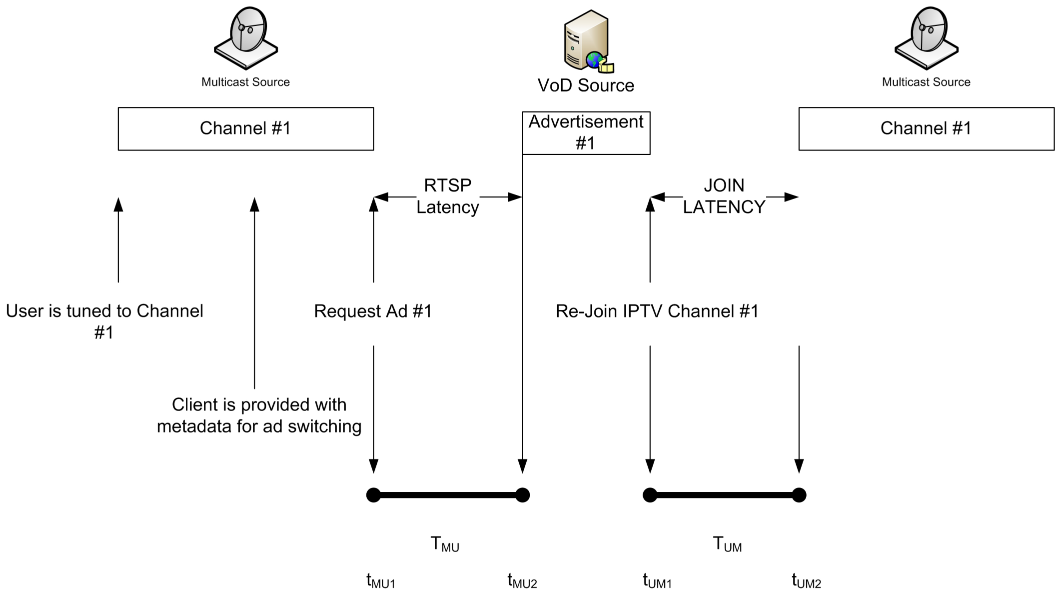

Figure 7.7.: Targeted Advertisement Test Scenario

7.2.3.3. Results

The measurements taken in this test case were designed to analyze the feasibility of the approach chosen for targeted ad insertion.

First, it must be denoted that the measurements are only representative under ideal conditions: The network load can be assumed as zero without taking the test user into account. This means that no interference with other services or parallel service usage has been taken into account.

Furthermore, the simple network infrastructure with just a single switch or hop allows for fast signaling, e.g. when joining a multicast group.

In Table 7.4, the average values for switching from Live TV to a unicast advertisement, and back from the advertisement to the corresponding Live TV channel have been denoted. Furthermore, Figure 7.8 depicts the absolute measurements for thirty test runs.

As denoted in Table 7.4, the switching times are relatively constant for each sub test case, e.g. around 150ms for switching to multicast and around 550ms when switching to a unicast signal.

Switching Type	Average Switching Time (ms)
LiveTV-to-Advertisement (Multicast-to-Unicast)	555
Advertisement-to-LiveTV (Unicast-to-Multicast)	150

Table 7.4.: Average switching time measurements for Targeted Ad Insertion

Furthermore, these numbers are within the limits of corresponding recommendations for channel switching, e.g. in the ITU-T's G.1080 recommendation, which proposes an overall *Zap Delay* of around 2000ms for good *Quality-of-Experience*.

The relatively high numbers for requesting a unicast stream can be explained through the much more complex signaling procedures and the number of messages exchanged, when requesting content via the used RTSP protocol and necessary handshakes between the network entities. For the Live TV channel, only a single IGMP JOIN message is sent to the next router.

Furthermore, the video codec and bitrate of the advertisement clip and Live TV channel play a certain role, as buffering and de-packetizing delay increase with higher numbers. Unfortunately, it was not possible to exchange the integrated *VideoLan* media stack of the used test client with another media player as the above mentioned delays are extremely dependent on the implementation of the video stacks.

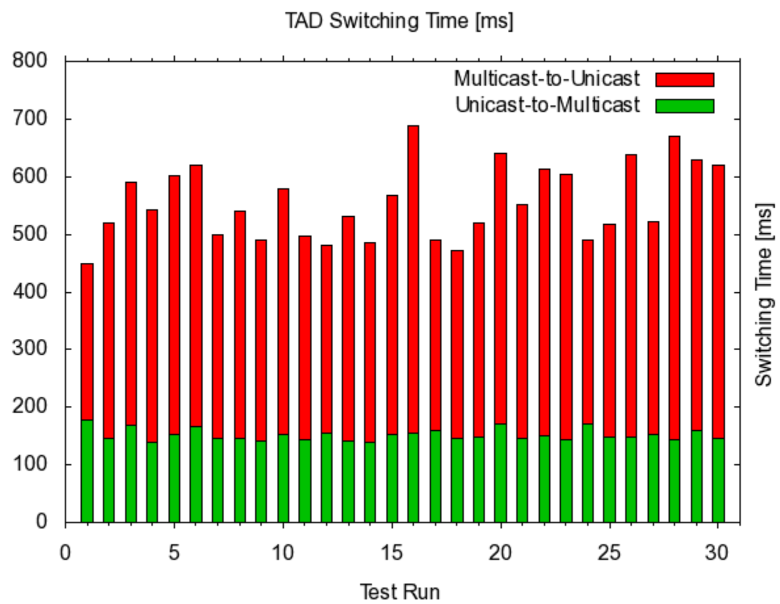

Figure 7.8.: Targeted Advertisement Test Scenario

7.2.4. Virtual Quiz Show: Server-side Content Mixing

This section takes up the concepts for a *Virtual Quiz Show* (VQS) scenario as specified in Section 5.4.5 of Chapter 5 and acts as a Proof-of-Concept validation.

The *Virtual Quiz Show* represents a class of scenarios in which a media server is responsible for composing an output stream using multiple input sources. In the specific case of the *Virtual Quiz Show*, these input sources must be combined, ideally with minimal processing delay, into a single video stream, which is then sent out to the participants via IP multicast. Hereby the stream is composed from a Live TV channel also fed via IP multicast and a video source provided by each user, having been recorded with a Web camera.

The advantages of such a solution are obvious:

1. Network resources are saved in scenarios with multiple users, as the downlink stream can be provided via IP multicast.
2. Many devices are not capable of rendering multiple video sources simultaneously. Server-side content mixing solves this problem, as just a single video stream with multiple blended videos is provided.

The scenario has been used to create a real time content-related gaming experience, posing a number of question to the users, alongside with the Live TV content played in the background. The *User Generated Content* was used to create a video chat situation between the participants, when participating in the *Virtual Quiz Show*.

An analog scenario would also fit the requirements of the *Targeted Advertisement Service* described in the last section, in which the content mixing is done sequentially and not into a joint stream.

Figure 7.9 illustrates the general approach chosen for this specific test case:

Two users are connected to the infrastructure and receive the broadcast channel and, in each case, the Web camera video of the other participant is rendered onto the broadcast channel, respectively.

7.2.4.1. Testing Scenario Setup

To validate the specifications for the *Virtual Quiz* Show scenario, a test system as depicted in Figure 7.10 has been created. The test system therefore consists of two clients using the same *End Devices and* and software as described during the *Targeted Advertisement* scenario, namely the Acer Aspire Idea 510 HTPC with 1,6GHz Core Duo CPU, 2GB RAM and Windows Vista operating system. Furthermore, the *Interactive Content Enabler* (ICE) as well as the combined *IPTV Session Management Enabler* with integrated application logic for the *Virtual Quiz Show* scenario has also been reused in this scenario.

Following the specifications from Section 5.4.5 four different measurements have been conducted on the test system:

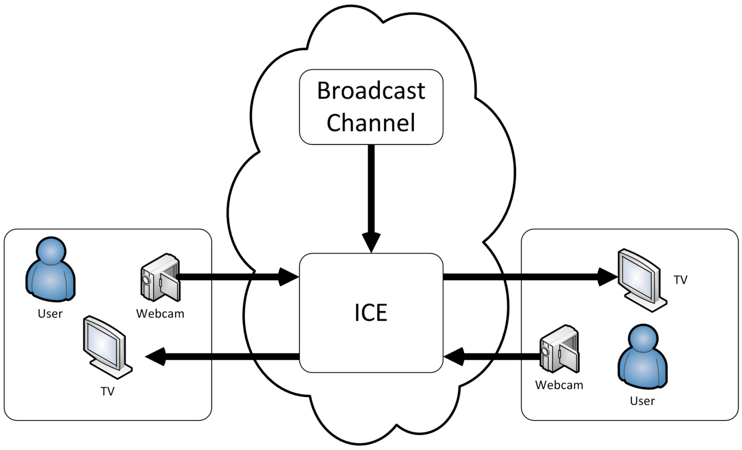

Figure 7.9.: Server-Side Multiple Input Source Mixing

1. Measurements of CPU load before and during the mixing process on the *Interactive Content Enabler*: Video mixing is a time, memory and especially computational power consuming process. The CPU load measurements have been taken on the *End Device* and the ICE.

2. An indirect measurements of the *Uplink Delay*, when generating *User Generated Content* on the end device upon its arrival on the ICE. Two measurements points, one on the *End Device* and another one directly on the ICE have been implemented. The second time stamp taken is identical to the first time stamp taken in the next step.

3. Measurements of the delay when mixing the different input sources on the ICE. Corresponding measurement points have been added to the source code and automatically compute the mixing delay. The mixing delay is computed by matching the sequence number and wall clock time of the first incoming packet with its counterpart, represented by the first outgoing packet.

4. An indirect measurement of the down link delay, i.e. measuring the time the first video packet leaves the the mixing engine and the *End Device* reports a *isPlaying* event.

7.2.4.2. Results

The measurements taken in the context of this scenario have been taken to validate the approach chosen, namely the server-side content mixing approach for the *Virtual Quiz Show* scenario.

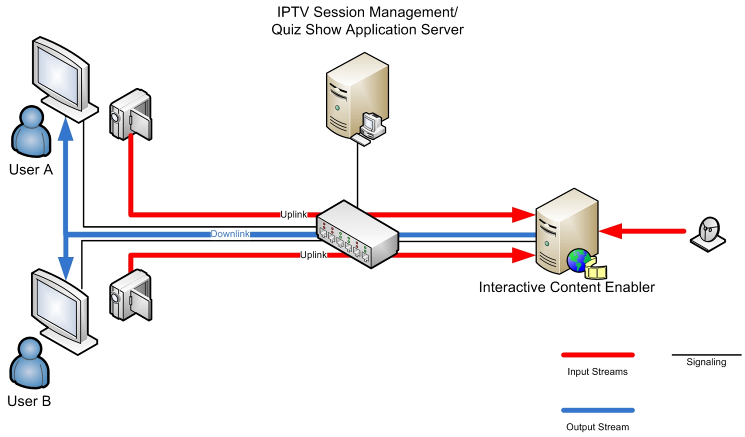

Figure 7.10.: System under Test: Virtual Quiz Show

The goal of the tests and measurements taken was to verify if such a scenario is feasible, especially with regards to different delays occurring on the *End Device* as well as on the ICE.

From a plain functional point of view, the specified scenarios worked flawlessly. When analyzing the results from Table 7.5 it becomes obvious that a total delay of around 2500ms occurs when transporting a user's Web camera signal to the other user, where a peer-to-peer scenario with multiple content rendering on the *End Device* would create a delay of around 1500ms.

As described earlier, the perceptual quality of IPTV scenarios is dependent on the occurring delays, e.g. zap or channel switching, with limits of around 2000ms. As the *Virtual Quiz Show* scenario is even more time critical and relates to conversational services like video telephony a delay of 2500ms is not acceptable for the maintenance of a high Quality-of-Experience. Even 1500ms using peer-to-peer video might be a critical number in real time gaming scenarios. As denoted in Table 7.5 delays occur especially during content generation on the *End Device* and mixing on the ICE and cannot be lowered easily because various elements including camera hardware, camera driver and corresponding video stack on client and server side are critical elements.

Another issue is client and server performance, as depicted in Figure 7.11. Even with powerful hardware, the ICE nearly reaches its limit with regards to available CPU resources. In this specific case, one standard definition TV stream in MPEG2 file format and Web camera streams in Motion JPEG format with a resolution of 640*480 have been used.

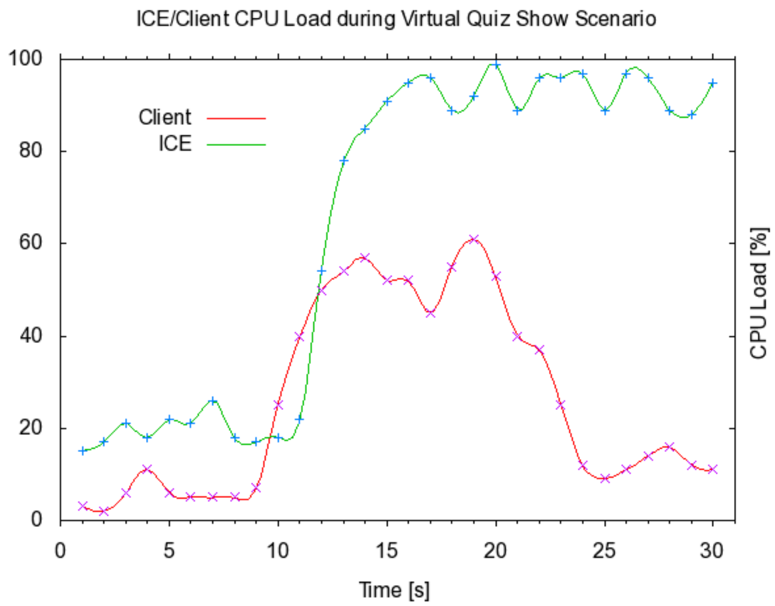

Figure 7.11.: CPU Load of Interactive Content Enabler & End Device during Test Case (Intel Xeon DP CPU 2,5GHz, Intel Core Duo 1,6GHz)

Delay Type	Average Delay (ms)
Uplink Delay (video Capture, encoding, transport)	1250
Mixing Delay (combination of input streams)	1000
Down Link Delay (transport, decoding, rendering)	250
Total Delay	2500

Table 7.5.: Measurements of Mixing Delay for the Virtual Quiz Show Scenario

7.2.5. Discussion

The last sections were used to validate three different scenarios, as specified in Chapter 5. First, the implementations for *IPTV Session Management Enabler* have been taken under analysis, followed by the *Targeted Advertisement* scenario and the Virtual Quiz Show.

It has been demonstrated that all specifications fit the requirements from Chapter 3 to a certain degree, where limitations occur especially for real time aspects, which are beyond the scope of this thesis. Furthermore, how the services integrate into the designed architecture from Chapter 4 has been demonstrated.

The next section will present three different case studies conducted during work on this thesis in the course of different international research projects. These case studies will also take up the concepts and enablers verified in this section, especially the *IPTV Session Management Enabler*. The presented case studies will furthermore concentrate on interactive services developed on top of the *IPTV Session Management Enabler*. Validation for these case studies is limited to a plain feasibility check. No measurements have been taken.

7.3. Case Study iSession: Social IPTV Scenario

In this study, the first prototype that incorporates the *IPTV Meta Session* concept from Chapter 4 was designed. The prototype, called *"iSession Demo Part 1"*, was developed as part of the evaluation scenarios in the Framework Program Seven Project iNEM4U [21].

It therefore also represents the author's main contribution to this project. In this context, the scenario has also been demonstrated as one of the main outcomes of the iNEM4U project for the European Commission, at the end of the project in April 2010.

Scenario Description The iSessions application is based on the concept of shared viewing of content between multiple users in the context of a session. The shared viewing experience is enabled by the *IPTV Meta Sessions* concept introduced in Chapter 4 and specified in Chapter 5. Potential users are able to participate in this experience through the use of the basic principles of a *IPTV Meta Session.* This concept enables them to interact directly with other users through their TVs. In further detail, the following functionalities are now available:

- Creating an iSession
- Joining an existing iSession
- Modifying an iSession by adding content or inviting additional users
- Synchronizing the viewing experience

On top of that, the novelty of iSessions is that they support multi-user sessions across domains. In this context, multi-domain implies that multiple user front ends incorporating different technological aspects have been integrated, and will be described in the next section. Figure 7.13 depicts the three different states of an iSession, including the main menu screen of two different UIs, the iSession listing in the middle and an active iSession on the right.

7.3.1. Infrastructure Setup

The iSessions scenario makes use of two TV clients, the iSession Server and the *FOKUS Open IPTV Ecosystem* signaling infrastructure, used to demonstrate the cross-domain aspect of the integrated technology. The following components have been integrated into the setup :

- An off-the-shelf Phillips NetTV with an embedded CE-HTML browser.
- An IMS-Based IPTV client from the FOKUS Open IPTV Ecosystem.

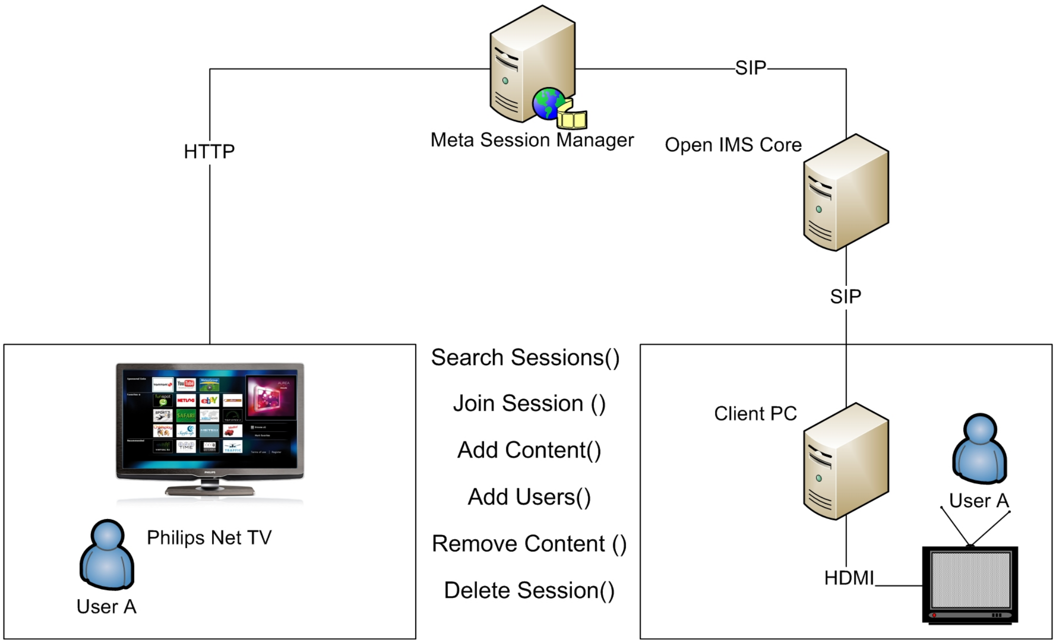

Figure 7.12.: iSession Demo

- The *Meta Session Application Server*: a converged SIP/HTTP Servlet running on the a Sun Sailfin [87] application server.
- The *Open IPTV Ecosystem Core*, i.e. parts of its IMS signaling infrastructure that use the *Open IMS Core*.

The Philips NetTV client is a TV with CE-HTML browser in which the client application is running. It offers personalized home portals to the users. Users can then browse and play any sessions offered by the iSession manager/repository.

The FOKUS IMS-Based IPTV client is a PC-based native application with a similar functionality offered to the user. There is no difference between these clients from a user perspective, except for the HTTP protocol for communication with the *Meta Session Application Server*, while the IMS client uses SIP. The *Meta Session Manager* offers implementations for both interfaces and thereby acts as a mediator between them and enables cross domain functionality.

The XMPP protocol is used for communication with and synchronization of clients. Each active iSession has one chat room through which all clients can communicate. E.g. if one client enters a session, an event is sent to all other clients already inside the chat room, giving them an opportunity to also pause the playback. All synchronization messages are exchanged via this chat-room. The following scenario shows the way in which one user chooses to watch iSession. The system detects that his friend (member of aggregated buddy list) is already watching the same iSession,

Figure 7.13.: iSession Demo

and offers the user the opportunity to invite the friend to watch the session in a synchronised manner. If friends decide to do this, sync agents will synchronize that playback on all clients. Users can also invite other friends to join the same iSession.

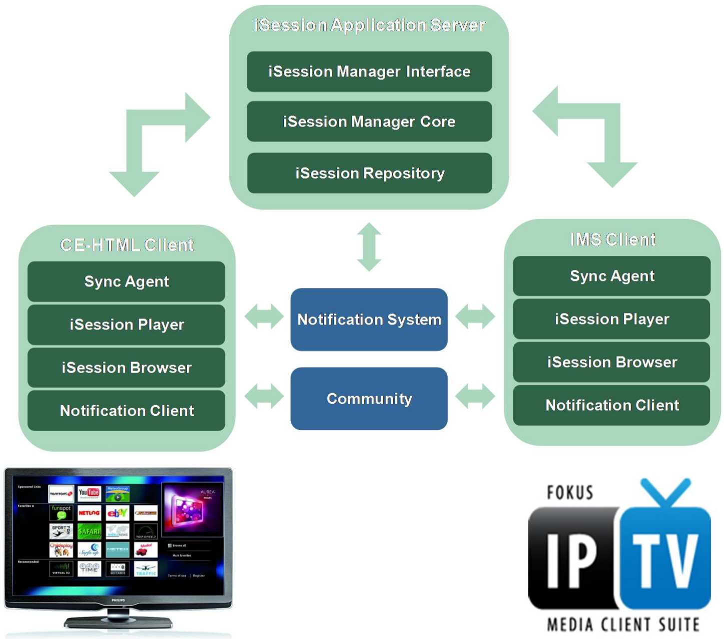

Figure 7.14.: iSession Architecture

7.3.2. Test Procedure

Both clients are connected to the *Meta Session Applications Server* as depicted in Figure 7.12. The following functions can be called from the user interface:

- Search Session: The *Meta Session Manager* acts as a *Session Repository* and allows connected clients to search for active sessions and then list them in the user interface

- Join Sessions: The user can select a session from the UI and join a session, i.e. starting contents belonging to a session.

- Add Content: Users who have joined an *iSession* can add content like pictures, videos or audio content to an *iSession*.

- Add Users: Each user might add new users to an iSession by inviting them through a dedicated message. The invited user gets a notification (e.g. a SIP MESSAGE) and can join the *iSession* through his user interface.

- Remove Content: Each user might have the right to delete content from a session. This content then also disappears in the list of available content for other users.

- Delete Session: *iSessions* can be deleted by a user. A deleted *iSession* is not available anymore in the corresponding listings.

Figure 7.13 depicts certain aspects of the prototype described in this case study. The first picture (upper left corner) shows the main menu of both clients, allowing *iSession* usage to start. The second (upper right) shows the *iSession* menu of the IMS-based implementation. The third picture (lower left) shows the *iSession* listing of both clients, including a single listed session. The fourth picture illustrates a started *iSession* with a video playing.

7.3.3. Discussion

Today, consumers are able to access multimedia content and services through different types of networks: Broadcast TV, internet, home networks and mobile. It is not currently possible for users and service providers to create media experiences that make use of all of the different content types and services available in these domains. The *iSession* and corresponding *Meta Session* model bridges that gap. In this case, it demonstrates the use of the cross-domain *Meta Session* model, synchronization algorithm and cross-domain buddy list. The benefits for the user are:

- Remote consumption of content together with others, regardless of the type of their end-device.

- Creation of live content for others across operators' and service providers' domains.
- Addition of communication services like voice or chat in order to support the shared experience.

In summary, the described scenario has shown that an interconnection of different – and at first glance perhaps incompatible solutions – is possible. In this specific case, it has been shown that an IMS-Based IPTV client and an OTT-approach like the Net TV platform can be used together. The *Open IPTV Ecosystem Core* confirms the effectiveness of the chosen approach. The platform acts as a mediator and demonstrates technology independency driving *Social IPTV* scenarios.

7.4. Case Study: Cross Domain IPTV Session State

In this study, the interaction of communication and content consumption will be analyzed, representing another *Social IPTV* scenario.

In contrast to the *iSession* approach from the previous section, in which the *IPTV Meta Session* model was used to create a shared viewing experience, this study will focus on the plain *IPTV Sessions* This second case study was conducted with a main focus on the so-called *IPTV Session State* introduced in Chapter 5.

The main purpose is to demonstrate again the interoperability of different, apparently incompatible technologies through the use of the proposed architecture for the *Open IPTV Ecosystem Core* described in Chapter 4.

In this specific case, three implementation have been created:

- A Microsoft Mediaroom[5] client created as an ASP.NET application. This application uses the *Mediaroom Presentation Framework* (MPF) representing a *Declarative Application Environment* (DAE) with a browser-like rendering engine. The connection to the *Open IPTV Ecosystem Core* and the *IPTV Session State API* is therefore realized trough a JSON-RPC API.
- A Yahoo! TV Widget[6] available on commercial TV devices, e.g. the 2009 Samsung LCD TV series. The provided middleware and SDK allows for the generation of application and widgets using JavaScript and proprietary runtime engine generating the UI (Konfabulator). The Yahoo! TV Widget also uses the *IPTV Session State API* through a JSON-RPC interface.
- The FOKUS IPTV Media Client, acting as a native IMS-based IPTV client using managed signaling (SIP) towards the *Open IPTV Ecosystem Core.*

7.4.1. Scenario Description

The *Cross Domain IPTV Session State* study has been designed to allow users of different IPTV ecosystems to share information about their current viewing behavior. In more detail, this scenario follows concepts gathered from different Instant Messaging programs. These concepts include a buddy list and so-called rich presence information. In this specific case, all participants will be provided with at least the following information and functionality on their TV:

- A buddy list showing a list of friends.
- The current status of their friends (online / offline).
- An enriched status feature that shows the content currently consumed by each user (TV channel / VoD asset)

[5] http://www.microsoft.com/mediaroom/
[6] http://connectedtv.yahoo.com/

In addition to the passive consumption of the the information above, two features that make use of the *IPTV Session State API* are also available:

- *"See What I See"* feature that allows participants to invite other users to the content currently being viewed.
- *"Watch The Same"* feature that allows a user to tune to the content viewed by another user.

Figure 7.15 depicts the system's overall approach with the *Open IPTV Ecosystem Core* in the center and the Yahoo! Widget Channel client, the Mediaroom client and the IMS-based IPTV client connected via the *IPTV Session Management API*.

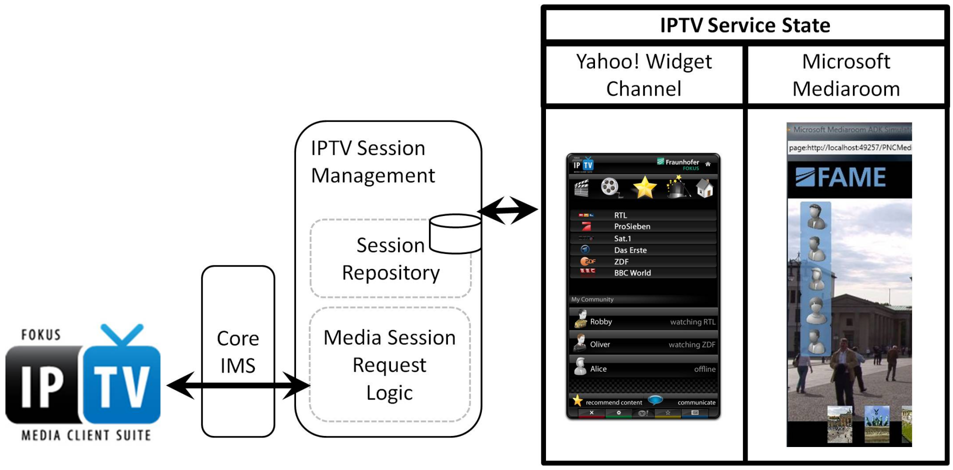

Figure 7.15.: IPTV Session State Evaluation Architecture

7.4.2. Infrastructure Setup

As described above, the *Cross Domain Service State* implementation makes use of three different clients connected to the lab infrastructure. The detailed test bed setup has been depicted in Figure 7.16:

- The FOKUS Media Client, as already described in the first study, is a native IMS-Based IPTV application. This application uses the SIP protocol to interconnect with the infrastructure provided. It uses the communication path through the Core IMS in order to trigger content from the *IPTV Session Management* Enabler. An established session inside the Media Session Request Logic instantiates a new entry inside the Session Repository. This session also updates the *IPTV Session State* information.

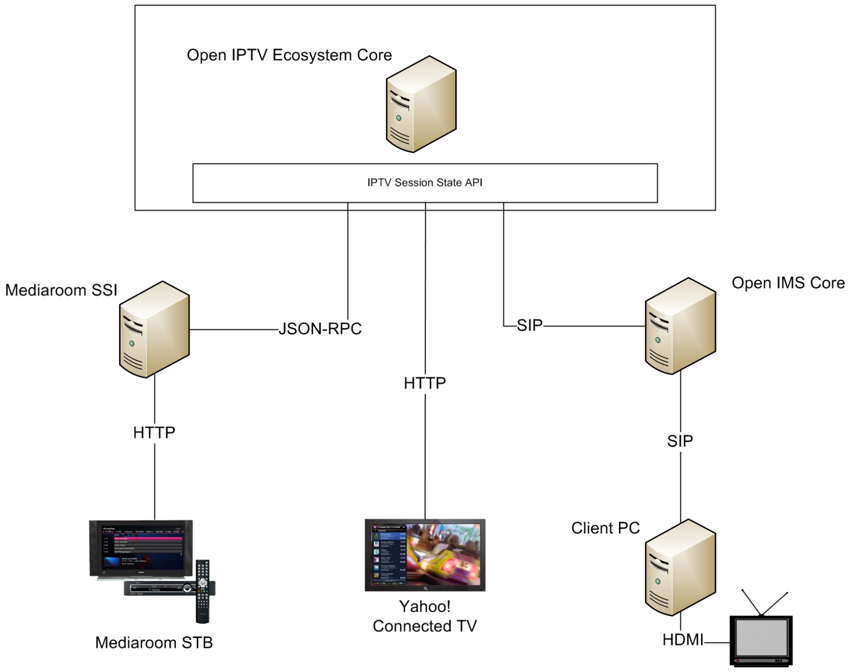

Figure 7.16.: IPTV Session State Test Bed

- An over-the-top prototype client based in the Yahoo! Widget Channel, and implemented mostly using JavaScript. This client implements an interface for the *IPTV Session State* information and *IPTV Session State API* provided by the *IPTV Session Management Enabler*. This allows the client to read data concerning available sessions, as well as update its status through the IPTV Session State API.

- A native implementation for the Microsoft Mediaroom platform that uses the so called Media Presentation Framework (MPF) and is written in ASP.NET and JScript, respectively. This implementation follows the same principles as the Yahoo! Widget Channel client.

Through the extension of the *IPTV Session State API* in the near future, various other scenarios will be enabled and thereby enrich the current implementation.

7.4.3. Discussion

The two case studies described in the second part of this chapter were created using three different commercial and non-commercial frameworks, and therefore run in three different IPTV ecosystems.

Nevertheless, all implementations can share information with little implementation effort. Without assuming anything in advance, it has to be concluded, that even

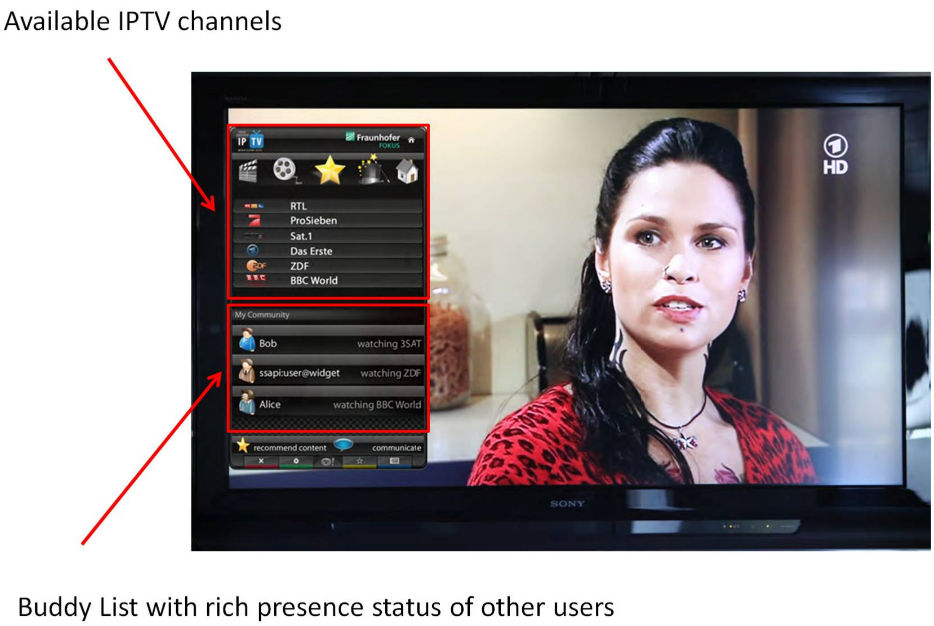

Figure 7.17.: Prototypical widget running on a Sony TV and Intel STB with Yahoo! TV Widget Channel

though the currently existing IPTV ecosystems use silo-oriented approaches, the interconnection and perhaps even harmonization of interfaces is also possible. The author will try to advance this idea through the presentation of new and updated demonstrations in the near future.

The limitations of the current solution rest mainly on the fact that the scenarios might not fit the business models of solution providers like Microsoft. From a technical perspective, the ideas described here are also more or less relevant to different IPTV standardization bodies.

Nevertheless the *IPTV Session State*, without the use of an appropriate API, does not make sense for connecting a third party, unless these third party solutions are used in standardized environments where the interoperability is required. For this reason, work in this area has to be continued, especially with regard to open APIs that allow for the control of IPTV scenarios.

7.5. Case Study: Singapore Proof-of-Concept

This case study summarizes the author's and his team's participation in the so-called Singapore NIMS Proof-of-Concept activity – a lab trial of standard compliant IPTV systems.

Within the context of the contribution of ten member companies from the Open IPTV Forum, the author's research group provided the *Open IPTV Ecosystem Core*, a Service Provider Discovery Server and various interactive applications using the combined DAE and SAE environment discussed in this thesis. Furthermore, research prototypes of a HTML5-based portal for the *Open IPTV Ecosystem Core* and so-called T-Governmental applications have been showcased.

7.5.1. Background

In 2006, the Singaporean government, and on its behalf the *InfoComm Development Authority*[7] (IDA), identified the need to develop a Fiber-To-The-Home (FTTH) infrastructure, called the *Next Generation National Broadband Network* (NGNBN). The nationwide offer is intended to reach across Singapore with an initial downlink speed of 100 Mbps and scalable to 1 Gbps. A 60 per cent coverage of all residential premises and nonresidential buildings by end of 2010, and 95 per cent by 2012 would be reached. Following a regulated Public Private Partnership (PPP) approach, a three-tier deployment model has been developed. This includes:

- A NetCo responsible for designing, building and operating the passive infrastructure layer of the NGNBN.
- An OpCo responsible for building and operating the active infrastructure of Singapore's network.
- Multiple Retail Service Providers (RSPs) offering services as IPTV on top of the fiber network.

With the ongoing deployment of the NGNBN infrastructure, IDA and the *Media Development Authority* (MDA) had created the Next Generation Interactive Multimedia, Applications and Services (NIMS) project by mid-2009.

The goal of this project is to create an ecosystem with a focus on interactive IPTV. The project has been structured in two phases in which Phase 1 was focused on understanding the requirement for interactive IPTV, collecting key considerations and industry trends. Phase 2 targeted creating a dialogue with specific participants to understand proposed models and key regulatory considerations as well as to develop functional requirements for NIMS Common Featured (CF) Set Top Box (STB). As part of Phase 2 in December 2009, the so called NIMS Panel was formed

[7] http://www.ida.gov.sg

by IDA/MDA, consisting of eleven industry players taking part in the dialogue. The NIMS panel is responsible for recommending the appropriate IPTV standards.

In parallel to the NIMS Industry Dialogue, IDA/MDA initiated the NIMS IPTV Proof-of-Concept sub-project. Different IPTV Standard Development Organization (SDOs) including DVB, ETSI TISPAN, ITU-T and OIPF presented their viewpoints. OIPF decided to take part in this practical activity by asking the membership for contributions to the requirements and test cases issued by IDA/MDA.

7.5.2. System Requirements

Requirement: IPTV Middleware for a Managed Network The focus of the NIMS project on a managed fibre network infrastructure requires the integration of corresponding middleware systems. As already described in the beginning of this chapter, the *FOKUS Open IPTV Ecosystem* fulfills these requirements and is compliant with multiple managed network environments.

Requirement: Multi Service Provider Support One of the main advantages of standardized Digital TV (DTV) and also future standardized IPTV infrastructures is that it permits an easy consumer choice of Service Providers. The derived requirements to fulfill this need can be ensured by offering mechanisms for Service Provider Discovery (SPD), using standard compliant Consumer Premise Equipment (CPE) as Set-Top-Boxes or TV sets.

As part of the author's contribution, a research prototype of a SPD server has been contributed to the project. A similar approach has already been described by the author in [49].

Requirement: Interactive Applications Providing services as interactive applications in a managed environment requires the integration of at least one *Interactive Application Environment* into the above-mentioned middleware system. In the scope of this project, the author provided several applications following the DAE and SAE approach.

Requirement: Common Content Protection Solution IDA/MDA requested a common Conditional Access (CA) and Digital Rights Management (DRM) solution, usable in an environment where protected services are offered by multiple RSPs. The system should not require user intervention when changing RSP. Content license can be utilised on multiple devices and should cover license reuse for Live TV, VoD and content sharing on supported user devices. The author and his team have not contributed to this requirement.

7.5.3. Infrastructure Setup

The OIPF's contribution to the NIMS IPTV PoC ecosystem was composed by the ten contributing member companies in a joint effort during two interoperability workshops in Sweden and Singapore.

The composed PoC Setup as depicted in Figure 7.18 represents an end-to-end managed IPTV solution integrating a *Service Provider Discovery Server*, two IPTV middleware systems, a content protection solution, two interactive application providers, and a home network containing CPEs from four manufacturers running two different embedded Web browser solutions.

The author's contributions are marked with the *Fraunhofer* label including SPD server, middleware and interactive applications.

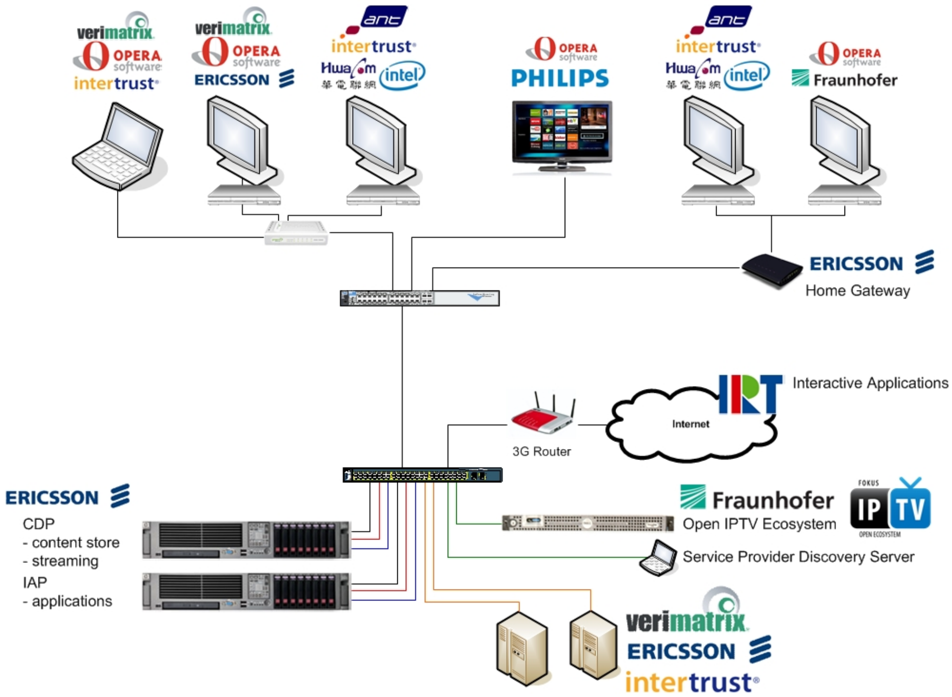

Figure 7.18.: Singapore Proof-of-Concept network setup

Figure 7.19 depicts the current setup including HTML5 middleware (second left), Service Provider selection menu (second right) and interactive applications (right).

7.5.4. Discussion

The Singapore Proof-of-Concept has proven to be a challenge for the participating companies and research facilities as, for the first time, an end-to-end reference implementation of the Open IPTV Forum's Release 1 specifications needed to be

Figure 7.19.: Singapore Proof-of-Concept setup

created.

Finally, the selection of the OIPF's specification by IDA/MDA for further consideration and adoption in Singapore has been recognized as a huge success, especially for the involved industry partners. For research facilities like Fraunhofer FOKUS and the Institut für Rundfunktechnik (IRT), the success was twofold in helping and facilitating industry with regard to standards adoption, as well as demonstrating features not yet part of the specifications but that go beyond the current requirements like the demonstrated HTML5 capable middleware and T-Governmental applications.

7.6. Requirement Validation

In Chapter 3, a set of requirements was defined for the different aspects relevant to the design and specification of the *Open IPTV Ecosystems Core* in Chapters 4 and 5, respectively.

These requirements have been grouped into five main areas, in which especially the first two groups of requirements for *IPTV Session Management* and *Interactivity* are relevant within the scope of this thesis and will be discussed here again. These requirements have been mapped onto the evaluations provided in the last sections.

7.6.1. Non-Functional Requirements

Interoperability Building an interoperable solution, according to open industry standards, was one of the main goals during work on this thesis. Furthermore, the active contribution to the standardization process in ETSI TISPAN and the Open IPTV Forum has been focused on. As an outcome of this thesis, the *Open IPTV Ecosystem Core* was used in the first three interoperability events InteroP TV#1, #2, #3 in Berlin and Stockholm.

Furthermore, the Open IPTV Ecosystem Core was contributed to the first Proof-of-Concept (PoC) activity at the Open IPTV Forum in July 2010.

Integrability Integrability is directly linked to interoperability and describes the successful integration of components coming from different sources. In the case of the *Open IPTV Ecosystem Core*, other components have mostly been integrated into the system and not the other way around, i.e. in the case of database systems. This has been demonstrated in this chapter in the test case on *IPTV Session Management* or in the *Virtual Quiz Show* scenario integrating the *VideoLAN* project.

Scalability Scalability has not been tested explicitly when evaluating the *Open IPTV Ecosystems Core.* The test cases on *IPTV Session Management* partly tested the scalability of the system, when doing database performance test.

Composibility The composibility of different types of services has been specified and tested within different implemented use cases. This includes the *IPTV Meta Session Management* as well as services integrating different kinds of IPTV middleware solutions from Microsoft or Yahoo!, as described earlier in this chapter. All *Social IPTV* services can be recognized as newly composed services.

Technical Applicability As discussed in Chapter 3 a missing real world application of interactive TV services has been the main drawback of various technological approaches, e.g. the *Multimedia Home Platform* (MHP).

The year 2010 has been the starting point for various commercial activities for so-called *Connected TVs,* integrating the technologies and concepts discussed in the course of this thesis. Especially *Social IPTV* scenarios using SAEs and DAEs as application environments have been already brought to the market.

Market Reach A specific commercial requirement when building a new product is market reach. In the IPTV domain, it has been extremely difficult to build a product e.g. for the pan-European or a worldwide market so far. A lack of standards has stopped the deployment of these services.

In the near future, specifically browser-based TV sets driving DAE-based OTT services have a good chance to be successful.

Cross Platform Deployability Cross platform deployments and multi-device usage play an important role when building successful interactive IPTV services. The evaluated services abstract from the underlying platform or operating system by either providing a multi-protocol approach (e.g. SAE) or by choosing a runtime environment available on multiple platforms (DAE).

7.6.2. Functional Requirements

Standards Compliance Standards compliance allows for integration and interoperability with other systems. As described above, these aspects have been most relevant during the specification of the *Open IPTV Ecosystem Core.* The test bed created during work on this thesis has demonstrated its applicability to the requirements described during different international research projects with industry partners and partners from academia.

Performance Performance tests allow for the comparison of a system against the requirement defined beforehand, to verify the applicability to real world or market conditions. In the course of this thesis, the *IPTV Session Management* represents the most critical component of the overall system. Performance tests have shown that the system fulfills requirements under certain conditions. For a real world application, more specific load and parallel usage tests must be conducted. Under laboratory and test bed conditions, the system fully fulfills the described requirements, e.g. derived from ITU-T or ETSI specifications.

Streaming Services Streaming services represent a basic service in each IPTV system. Different approaches have been discussed in the course of this thesis. In Chapter 5, a session-oriented approach for Live TV and Video-on-Demand has been specified and furthermore implemented into the *Open IPTV Ecosystem Core* in Chapter 6. In this chapter, the *IPTV Session Management* has been evaluated, fulfilling the baseline requirements.

Third Party Openness/APIs Third party openness is an enabler for interactive services and Social IPTV, allowing third party service providers to build new sophisticated services. In conjunction with the requirements for a standardized and interoperable environment, how proprietary services can also be integrated and interconnected by providing open APIs and using open Web technologies has been demonstrated.

Meta Sessions The Meta Session concept acts as an enabler for all Social IPTV aspects discussed in this thesis. A key element is the availability of interfaces for multiple platforms. This flexible mechanism allows for the connection of OTT and Managed IPTV clients as demonstrated for a Philips NetTV and the *Open IPTV Ecosystems Core*, including corresponding managed clients.

Interactive Application Environment The *Open IPTV Ecosystem Core* implements two *Interactive Applications Environments*, a session-oriented environment (SAE) and a declarative environment (DAE). Both environments have been evaluated in this chapter. Especially interactive services, like targeted advertisement services, have good chances for practical and commercial deployments in the near future. During the evaluation of this service, it has been shown that the application environment itself and the service specifications are applicable to real world conditions.

Interactive Services Different interactive services have been specified in Chapter 5 of this thesis. Furthermore, the targeted advertisement service and the virtual quiz show service have been evaluated in this chapter.

It has been shown that the specification can be applied to real world conditions based on the available Interactive Application Environments, i.e. SAE and DAE.

Social IPTV represents the latest trend for IPTV at the point this thesis was written. Various market players have announced services in this area, ranging from CE manufacturers using OTT approaches up to Telco TV providers integrating already available communication services. The concepts presented in this thesis therefore have a high relevance for research and academia and might be implemented into upcoming standards and products.

8. Assessment & Comparison with Related Works

Chapter 8:
Assessment & Comparison with Related Works

Evaluation Criteria
Related Works
Assessment

Interactive IPTV systems and adjunct topics are a wide field of research. Numerous research groups have, and are still actively contributing to the specific areas described in this section.

With respect to the different areas referenced in this thesis, mostly framework-oriented approaches, providing end-to-end IPTV systems, have been analyzed. In more detail, this includes architectural approaches, frameworks for *Application Programmable Interfaces* (APIs), allowing for access to these architectures from the outside, as well as systems for the creation and delivery of contents. Furthermore, the most important frameworks for shared content consumption and Social TV are presented.

This section analyzes the 25 most relevant contributions to this field by providing a description and categorization of the work. Afterwards, the analyzed contributions will be compared to each other and with the author's work.

8.1. Evaluation Criteria for Interactive IPTV Systems

8.1.1. Categories

When selecting the most relevant contributions to the field that influenced the author's work directly, five main categories can be distinguished. This includes:

Architectural Approaches which mostly concentrate on architectural frameworks providing an end-to-end infrastructure for IPTV. All analyzed infrastructures rely on frameworks for Next Generation Networks (NGN) and/or the IP Multimedia Subsystem (IMS).

Open APIs allow for the extension of the above-mentioned architectural approaches with interfaces for third party applications or content providers. The selected contributions either provide theoretical contributions to the field or extend existing infrastructures and demonstrators.

Content Provisioning is a specific field of research in the IPTV domain. Nevertheless, the contributions presented here mostly extend existing IPTV architectures and allow for the generation and delivery of contents adapted to user or network requirements. The main goal of all contributions is to extend the Quality-of-Experience (QoE) for the user on one or even multiple devices and through different access networks.

Interactive Services play a key role in IPTV environments. the contributions summarized in this section concentrate on the technical aspects necessary to realize interactive services.

Social IPTV and corresponding contributions mostly concentrate on how various communication services help to enrich the user experience and enable shared experiences between groups of users. These contributions therefore place much more focus on user and case studies, analyzing the impact on the consumer, rather than describing technical aspects.

8.1.2. Criteria

Beside this general categorization and mapping of each contribution into the corresponding category described above, each of them has been analyzed with regards to the following evaluation criteria, summarized in the following:

Approach The contribution is analyzed with regards to the approach chosen by the authors, e.g. test bed, plain specification or combination of both.

System Setup The system setup is analyzed with regards to the technological approach chosen, its maturity and potential real world applicability.

Features The contributions features are analyzed with regards to:

- Level of interactivity and services
- Third party APIs
- Communication services, e.g. chat, IM, buddy list, presence
- Social TV aspects like shared service consumption and shared service experience

8.2. Architectural Approaches & Services for Session-Oriented IPTV Systems

With the begin of standardization activities for IPTV in DVB and ETSI TISPAN in the year 2006, several research groups, including the author's team, began to concentrate their efforts on next generation IPTV architectures.

This section collects the contributions to the field:

8.2.1. Bodzinga06

Approach Bodzinga et al. provided in [11] one of the first contributions to the new field of research concerning the integration of IPTV into NGN/IMS infrastructures.

For this purpose, they analyzed different aspects and advantages relevant to the integration of entertainment services into a telecommunication infrastructure.

Contributions The key points of the their analysis can be summarized as follows and are limited to theoretical assumptions and contribution to IPTV standardization.

- Simplifying IPTV architectures, by integrating them into the framework of the IP Multimedia Subsystem (IMS). This allows for the re-use of building blocks already available for other, e.g. conversational but also session-oriented services.
- Service Blending allows for the combination of existing services, e.g. Instant Messaging (IM), voice or video telephony with IPTV.
- Common subscriber management information, i.e. re-using protocols and existing user databases for IPTV subscribers as well.
- Common network resource control, i.e. making use of NGN/IMS mechanisms for Quality-of-Service (QoS) control.
- Common service signaling, e.g. through re-using the SIP protocol for IPTV session signaling as well.

8.2.2. Fasbender06

Approach In [123] Fasbender et al. describe an approach towards a personalized, interactive IPTV solution based on open standards.

They contribute to the field by providing an overview of potential scenarios for convergent IPTV solutions, a technology mapping onto NGN infrastructures and an update of IPTV standardization activities.

Contributions As an early contribution to the field, Fasbender's work does not include any technical details concerning the development of such an infrastructure. Theoretical assumptions are made describing a mapping of IPTV services onto different elements of an next generation network. As also discussed in [11] this mainly includes aspects concerning re-using the session concept, components for user authentication and personalization, combined and blended services.

Furthermore, they elaborate on the client environment and home network issues, as an integration with services available from the standards of the Digital Network Living Alliance (DLNA).

Another point taken under analysis is the use of IMS Application Servers as an enabling platform also for IPTV services.

They conclude their contributions by stating that the integration of IPTV with NGN/IMS networks and services is currently the most promising approach to building convergent IPTV infrastructures.

8.2.3. Chatras07

Chatras et al. [8] continue describing the advantages of NGN/IMS-based infrastructures for IPTV. In the same manner as previous contribution described earlier, advantages like a common identity, resource and charging management are described. Furthermore, signaling principles for session-oriented IPTV services are taken under analysis.

Contributions Chatras describes in great detail, and in parallel to his work within standardization in ETSI TISPAN, how IPTV services in NGN infrastructures are bootstrapped through so-called Service Discovery Mechanisms, how IPTV streaming and broadcast sessions are established and how services like a network Personal Video Recorder (nPVR) can be realized.

This information is provided by describing high level signaling flows. Practical implementations were beyond the scope of the author's work.

8.2.4. Riede07

Approach The team at Fraunhofer FOKUS started very early with valuable, and practical analysis and implementations of next generation IPTV infrastructures. The resulting works concerning mainly infrastructural ideas, session concepts and implementations are reflected in Riede et al.'s work presented in [111] and [110].

Contributions Riede provides a detailed view of NGN-based IPTV architectures by describing an IPTV research prototype developed jointly with this thesis's author. The so-called Session Management Enabler (SME) represents Riede's key contribution to the field by implementing *IPTV Session Management* on an IMS

Application Server. The developed component integrated seamlessly into the IPTV test bed at Fraunhofer FOKUS.

In addition to implementation details, the signaling of session-oriented IPTV services is also described in this work. This includes detailed signaling flows for Video-on-Demand services.

Finally, some measurements of IPTV session setups through fixed Local Area Networks (LAN) or WiFi networks are presented.

8.2.5. Mas08

Approach In [82], Mas et al. describe AT&T and Ericsson's approach to an NGN/IMS-based IPTV ecosystem. They describe session flows for some important use-cases, such as for accessing the EPG, Video–on-Demand and fast channel change. Furthermore, some screens of a prototypical implementation are presented.

Contributions The author's contribution to the field is recognized to be providing a detailed system architecture and descriptions of its building blocks. They differ from other approaches by introducing a home gateway responsible for terminating all operator-related signaling and providing them as DLNA services to the different devices in the home network.

Detailed signaling flows are discussed, allowing for scenarios like:

- VoD/Broadcast TV
- Change of User Profile
- Remote Authorization
- Interactivity

Finally, Mas describes different use-cases implemented in a prototypical IPTV system. This includes a user portal, access control, an EPG and an interactive voting application.

8.2.6. Mikoczy08

Approach In [90], Mikoczy et al. describe a prototype environment for session-oriented IPTV services developed in the context of a project called Scalenet[1].

The developed system reflects standardization efforts within ETSI TISPAN, as parts of the team were active in the standardization process.

[1] http://www.scalenet.de

Contributions In addition to the architectural issues and adaptations of draft standards for IPTV, a real world test bed has been implemented. Its focus lies on implementing key aspects of the ETSI TISPAN standard. In more detail, this included work on the following components of such an infrastructure:

- Prototypical IPTV client for the consumption of session-oriented, NGN-based IPTV services
- IPTV session management for session monitoring.
- IPTV Media Delivery Function with integrated QoS adaptation capabilities

The implemented end-to-end IPTV infrastructure has then been used to connect clients through different access networks and measure the QoS adaption performance.

8.2.7. Volk08

Approach Volk et al.[134] present a theoretical approach to a policy-based NGN-IPTV quality assurance model. The described assumptions have been added to existing specification s coming from ETSI TISPAN.

Contributions Volks's main contribution lies in proposing a quality-related profile containing service layer, as well as transport layer policies. The profile collects necessary information for quality assurance from the user, service and content perspective. A set of metrics facilitate the adaption of this information. This includes metrics for perceptual quality, video/audio stream metrics and transport/network metrics.

8.2.8. Menezes09

Approach In [84], the author describes his work on a SIP-based IPTV platform integrated into an IMS infrastructure. The main focus of his research centers around signaling aspects for session-oriented IPTV, the prototypical implementation of the Session Controller and measurements of session setup, channel change and QoS adaptation scenarios.

Contributions The author describes, as part of his dissertation, an IP Television (IPTV) service architecture that applies the Session Initiation Protocol (SIP) for session and media control, while incorporating a design suitable for deployment in the context of an IP Multimedia Subsystem (IMS) architectural framework. The main features of the architecture include flexible delivery of personalized multimedia streams, adaptability to the characteristics of the networking infrastructures and channels used for transmission, and a modular design to facilitate implementation of new functionalities and services. The developed solution is specifically designed for live multimedia streaming, such as broadcasted events, independent of the cast mode (unicast or multicast). Private Video Recorder (PVR) functions and Video On Demand (VoD) services are supported, their control is assured by standard SIP messages.

8.3. Third Party Openness

8.3.1. Lyu07

Approach & Contributions Lyu et al.'s approach for an open API in IPTV environments [81] presents one of the first approaches to open-up IPTV infrastructures and systems for third party. Three exemplary services suitable for exposition to third party, namely a personalized EPG, networked Personal Video Recorder (nPVR) and a targeted advertisement service, have been chosen for integration in the so-called Open API approach using ParlayX Webservices.

In contrast to other approaches, only the user benefits from these Open APIs, which allow him to control the described service from a Web browser.

8.3.2. Mikoczy09

Approach In [88], Mikoczy et al. describe the concept of combinational or blended IPTV services and their exposition to third parties in the concept a Service Oriented Architecture (SOA). Basic service enablers from the NGN domain like presence, messaging, buddy list and notification are used in combination with IPTV services creating new service and business models.

Contributions By referring to the concepts of next generation, all-IP networks Mikoczy introduces SOA concepts for IPTV environments, allowing them to change the current implementation of a service, without changing the interfaces. The *Presence* service is used as the core service enabling combinational services. A service broker or interaction manager is introduced for mediation between the different services and enabling scenarios, like pausing a Video-on-Demand upon an incoming call. The specifications are furthermore verified through prototypical implementations.

8.3.3. Daher09

Approach & Contributions Daher describes in his diploma thesis [25] an infrastructure that combines the benefits of a session-oriented approaches for the dynamic provisioning of multimedia streaming services, with a Service Oriented Architecture (SOA). It provides a generic API allowing third party services to make use of the provided infrastructure. Daher's work represents a parallel research activity to the author's work, performed at the Fraunhofer Institute FOKUS.

8.3.4. Yang09

Approach An open service control platform for IPTV, following the principles of a Service Oriented Architecture (SOA) has been chosen by Yang et al. for their work on IPTV platforms [139] [138].

Contribution Bridging Web and telecommunication services has a long tradition in the telecommunication domain. OSA/Parlay and later ParlayX Web Services have evolved towards the idea of Service Oriented Architectures (SOA). Yang proposes the extension of these ideas towards the concept of IPTV, allowing third parties to access IPTV services. This allows for service blending, e.g. through broadcasters or content providers that are not necessarily part of the IPTV service provider's network.

8.3.5. Yoon10

Approach and Contributions Yoon et al. describe in [140] the convergence of broadcasting and telecommunication through IPTV. They furthermore outline the relevance of open APIs in these environments, also building the bridge to the Web domain.

8.4. Content Provisioning

Related work on session-oriented media delivery infrastructures is a very specific field of research, and often linked to developments within IPTV standardization bodies like ETSI TISPAN and the Open IPTV Forum.

8.4.1. Waiting08

Approach & Contributions In [136] Waiting et al describe developments at the University of Capetown (UCT), South Africa on the UCT IPtv server. This server is a SIP application server that streams up to three channels to multiple destinations. The server is built on top of the Open Source SIP library *eXosip* to support session-based SIP signalling and the *GStreamer Library* for media delivery and it is released free on the Internet under the GPLv3 licence. Its primary goal is to serve a limited number of packet-based media streams to as many clients as possible, similarly to regular digital terrestrial and satellite television broadcasts. The server is designed to fit tightly within IMS architectures, therefore unlike other IPtv solutions, it uses SIP exclusively for signalling.

8.4.2. Kadlic09

Approach & Contributions In [75], Kadlic et al. analyze the concept of a networked Personal Video Recorder (nPVR) as part of the IPTV infrastructures of the University of Capetown (UCT) and the NGNLab[2].

Kadlic provides details about the proposed architectural integration of the nPVR functionality into the test bed infrastructure and some information on how broadcast assets might be stored into a database.A detailed description of the prototypical implementation is lacking.

8.4.3. Menai09

Approach Menai et al describe in [83] their work on standard SIP/RTSP-based Content Delivery Networks (CDN).

Contributions The system includes a central server (Content Delivery Network Controller) that analyzes all received content delivery requests. The Content Delivery Network Controller (CDNC) chooses the cluster of servers a request should be redirected to. The choice is made depending on client location, content availability, location and servers' global load. Each cluster is controlled by a Cluster Controller (CC) that makes the choice of the final VoD server to deliver the content, based on a fine-grained analysis of the load of the VoD servers it manages. The system proves the feasibility and flexibility of SIP interfaces when coupled with RTSP to organize redirections within a CDN.

8.4.4. Arnaud10

Approach Arnaud et al. [6] propose new functionalities for IMS-based IPTV architectures enhancing the level of user satisfaction, as well as limiting the resource utilization of the operator's network.

Contributions This was reached through a new context-sensitive user profile model for the delivery of adapted streams to the user's environment. Furthermore, a Multimedia Content Management System (MCMS) is proposed for the performance of cross-layer adaption of the provided contents.

8.4.5. Cruz10

Approach Cruz et al. describe an approach for the development, implementation and evaluation of a SIP-based IPTV architecture with a new dynamic QoS adaptation method and signaling structure [24].

[2] http://www.ngnlab.eu

Contribution The author's main contribution lies in providing mechanisms for dynamic QoS adaptations through the modification of session parameters. Detailed descriptions of the signaling flows, as well as the necessary session description data are provided. Furthermore, performance measurements have been conducted on the proposed infrastructure.

8.5. Interactive Services

8.5.1. Niamut08

Approach & Contributions In [27], Deventer et al. present an approach to allowing IPTV users to participate in a virtual game show by using mechanisms for uploading User Generated Content to a managed IPTV environment. This is achieved through the combination of SIP and RTSP for session signaling and media control, respectively. The authors present their proposed architecture, necessary signaling flows and a prototype for implementation based on Open Source and off-the-shelf components.

8.5.2. Wilson09

Approach According to Wilson et al. in [137], session-oriented IPTV systems are qualified to deliver personalized advertisement services.

The advertisement system has been implemented as an additional component in the end-to-end IPTV test bed of the University of Cape Town (UCT), South Africa.

Contributions Wilson et al.'s main contribution lies in the practical realization of an advertisement system using session-oriented signaling principles.

Upon service request, an advertisement evaluation engine adds contextual advertisements fitting the user profile. These advertisements are then inserted into the current contents. Evaluations on the presented work were performed by measuring the call setup delay with or without advertisement.

8.6. Social IPTV

Throughout technological convergence in the field of rich media content and rich communications, research on the so-called *Social IPTV* is gaining more and more attention as this thesis is being written. This section will summarize the work in progress in this area of research.

Social IPTV extends the consumption of (rich) contents by adding communication services that can be used by the consumer to interact with his friends and family. The current trend of *Social Networking* and *Social Media* is very much reflected in this field of research, blurring the borders between TV, communication and Social

Media. Altogether, Social IPTV scenarios provide a fruitful baseline for the system described in the scope of this thesis, when combined with technologies for user front ends (e.g. *Declarative Application Environments*) and the Session-Oriented Application and IPTV environments presented in the next section.
Social IPTV is used to refer to a variety of experimental systems that claim to support social experiences for television viewers, and research such experiences [14]. In more detail, Montpetit et al [92] define Social IPTV as follows :

"video services that integrate other communication services like voice, chat, context awareness, and peer ratings to support a shared TV experience interactivity with peer groups (shared viewing) and peer recommendations are driven by the recent rise of social networks"

As described above, communication aspects combined with consumption of content are driving Social IPTV and forging a path towards interactive IPTV. Social TV, however, is not necessarily directly linked to a specific technology, but concentrates on the concepts and use-cases behind it. As Social IPTV reflects a portion of the ideas and use-cases presented by this thesis, it is worth understanding the drivers behind this area of research and its related projects. The next section is dedicated to an overview of current State-Of-The-Art work on Social IPTV. Different projects concerning Social IPTV will be presented in this context.

8.6.1. Coppens04, Pelt04

Approach In [104, 23], the authors present one of the earliest efforts in the area of Social Television, which can be recognized as a sort of reference project in this domain. The main idea consists of providing avatars for every single user, as a way of visualizing its presence. A variety of faces are available to be used as avatars and express the user's emotions during content consumption. In addition, so-called shared video effects create another way to express emotions between users (e.g. a flaming ball whizzing across the screen).

Contributions The XMPP or Jabber protocol [116] is used for communication purposes. The Jabber implementation used for AmigoTV allows for buddy list and presence functionality. A so-called Room Functionality was added for the creation of a community scenario in which multiple users can create a virtual private area for content consumption.

8.6.2. Nathan08

Approach In [94] the authors from the AT&T labs present their work on a Social TV system called CollabraTV. Unlike some other systems, like the previously discussed AmigoTV or STV below, CollabraTV is focused on supporting asynchronous communication.

Contributions Results were gained through extensive lab studies that tested the various features of the platform, including:

- Temporally-Linked Annotations allow users to attach comments to videos. In a next step, these attachments are shown to the next viewer at the corresponding time index. Additionally, a so-called Interest Point containing a user's positive or negative reaction to the content can be added.
- The Virtual Audience is an extension to the Temporally-Linked annotation feature that adds a buddy list. This buddy list contains on-line as well as offline users and their corresponding comments.

8.6.3. Boertjes08

Approach The ConnecTV project was carried out within the so called B@Home project and is described in detail in [12]. ConnecTV is unique in the field of Social Television research because the prototype was implemented for a field test in about 50 households in the city of Enschede.

Contribution With regard to features, ConnecTV uses the XMPP protocol, like in the Amigo TV project, for communication purposes between users. This includes features like a *buddy list*, *rich presence* that shows other users' TV channels and a feature that allows for switching to channels other users are currently watching.

Additionally, *TV program recommendations*, *follow and invite a friend* scenarios have been implemented. A *most popular channel* service community feature was also available during the field test.

In the results, the *follow a friend* scenario in combination with the available buddy list was used by most of participants, showing that the communication features were accepted.

8.6.4. Harboe08, Tullio08, Huang

Approach The idea of the *virtual couch* is the best description of the research concerning the different iterations in the Social Television System (STV) project [60, 59, 127, 67].

Contributions During the development of this project, throughout various prototypes and case studies, the design goal was to enable a small group of relatives or friends to share with each other while watching TV. The key elements evolved during different phases of the project (STV1-3). The key elements of the current version are:

- A television presence that includes a buddy list and ambient display
- The creation of a Shared Viewing Experience though program suggestions from other buddies
- Two different communication channels, including text chat and group voice calls
- The integration of a user's viewing habits into the EPG of other buddies

When examining the results so far, it turns out that in Shared Viewing scenarios some communications channels are better suited than others. Especially in that voice chats are more accepted than text chat.

Additionally, most of the participants did not want the video chat feature, a result that the author can also underline as a result of his own experiences. Several other aspects also seem to be unresolved like how multiple users in one household can be supported and CE equipment allowing for an easy integration of community features can be developed.

8.6.5. Zhang10

Approach The work presented by Zhang et al [145] describes the vision of a so-called IPTV 2.0 system developed within the ITEA2 project *CAM4Home*[3]. The system combines both: TV-enriched communication through standardized platforms and services provided by NGN/IMS environments, as well as sociability-enhanced TV, i.e. Social TV aspects.

Combining these two aspects in a convergent system is in-line with the approach chosen by the author of this thesis.

Contributions The presented framework and developed architecture for IPTV 2.0 relies on open standards including ETSI TISPAN, DVB and the Open IPTV Forum. A key aspect of the project was the specification of a new, open metadata framework with the ability to encapsulate existing metadata technologies for multiple types of content and also able to reference to content related services. With respect to the Social TV aspects, it includes a so-called social tag allowing for user comments and ratings.

A second target was the integration of NGN-based telecommunication services using two enablers for presence and instant messaging (IM). This feature is called TV Buddy. Two scenarios have been added to the traditional instant message service:

- TV metadata of the friends currently logged into the system is presented in the other user's screen as a sort of rich presence feature.

[3] http://www.cam4home-itea.org/

- A resource sharing feature, allowing other users to request the content displayed as rich presence in the buddy list.

Another feature of the described system is the so-called Social EPG. Social EPG represents an extended version of the traditional EPG which allows the user to share, save, organize and search TV resources based on the concept of tags, user ratings and comments.

8.6.6. Assessment & Comparison

The overall goal of this PhD was to design, implement and evaluate a session-oriented IPTV system, enabling interactive services and Social IPTV.

The related works presented in this chapter represent research and development in this field, in which each research group mostly concentrates on a specific aspect, e.g. end-to-end architecture design, third party APIs, interactivity, Social IPTV or content delivery networks and their optimization.

Furthermore, the related works either concentrate on theoretical assumptions, a test bed architecture or even real world test.

This categorization makes it easy to provide a rating for each approach as described in Table 8.1.

First, if the provided approach describes an end-to-end architectural framework, it is rated (++), only parts of an architecture (+), touches on the topic of architecture but does not describes the architecture itself (o) or does not make any assumption on architectural issues (–).

The second rating checks the availability of mechanisms allowing third parties to access the system (+/–), e.g. for the provisioning of interactive services. As this is a very specific field of research a certain limitation on the availability of this feature is obvious and the absence of a positive rating does not necessarily imply a negative rating of the overall contribution.

The rating of Interactivity checks on the availability of specifications for interactive services in the described system, where either no interactive services are available (–), the potential integration of interactive services is denoted (o), basic services are described (+) or the contribution is focused on interactivity (++).

Social IPTV aspects present another very specific field of research and are therefore reflected only in a subset of the presented related works. The levels distinguished between are no Social IPTV aspects available (–), early stage or somehow related, e.g. though available communication features (o) or fully available (++). work on content delivery aspects and their optimization have been conducted by different research groups. As observable in Table 8.1, these works often do concentrate heavily on this aspect and do not touch on e.g. interactive services nor Social IPTV aspects.

The last rating provided here checks the availability of a real world test bed, e.g. allowing for performance, integration or usability testing.

As visualized in Table 8.1, most early contributions do not provide system tests nor a real world test bed. It must be stated that this overview of related works presents major contributions to the field by having selected only relevant material. Tables 8.2-8.4 again provide a detailed analysis for each contribution presented in this chapter.

	End-to-End Architecture	Third Party APIs	Interactivity	Social IPTV	Content Delivery/ Optimization	Validation & Test Bed
Bodzinga06	+	–	–	–	o	–
Fasbender06	++	–	o	o	o	+
Chatras07	+	–	+	o	o	
Riede07	+	–	–	–	+	++
Mas08	+	–	+	o	o	++
Mikoczy08	++	–	o	–	+	++
Volk08	+	–	–	–	++	+
Menezes09	++	–	–	–	++	++
Lyu09	+	+	++	–	o	o
Mikoczy09	++	+	++	o	o	++
Daher09	+	+	o	–	++	++
Yang09	+	+	++	o	–	o
Yoon10	+	+	+	–	–	–
Waiting08	++	–	–	–	++	++
Kadlic09	+	–	+	–	++	++
Menai09	++	–	+	–	++	++
Arnaud10	++	–	+	–	++	+
Cruz10	++	–	–	++	++	++
Niamut08	+	o	++	++	+	+
Wilson09	+	–	++	–	++	++
Coppens04	o	–	++	++	–	++
Nathan08	o	–	++	++	–	++
Boertjes08	+	–	++	++	–	++
Harboe08	o	–	++	++	–	++
Zhang10	++	–	++	++	o	++
OIEC	++	+	++	++	+	++

Table 8.1.: Assessment & Comparison of Related Works

	Bodzinga06	Fasbender06	Chatras07	Riede07	Mas08	Mikoczy08	Volk08	Menezes08	Lyu09
Approach	baseline IMS-based IPTV architecture	Ericsson research IMS-based IPTV infrastructure	Theoretical evaluation of ETSI TISPAN IPTV	FOKUS NextGenTV test bed	NGN-based IPTV architecture and test bed	NGNLab/ Scalenet IPTV test bed	Policy and profile-based QoS adaptation in NGN-based IPTV	IMS-based IPTV test bed with focus on content provisioning	Proprietary IPTV system with open third party interfaces
System Setup	blueprint architecture	blueprint architecture and prototype	blueprint architecture	test bed based on Open IMS Core	architecture and test bed	architecture and test bed	blueprint architecture	architecture and test bed	architecture/ specification
Third Party API	n/a	n/a	n/a	n/a	n/a	n/a	n/a	n/a	ParlayX
Content Adaptation	n/a	n/a	n/a	client SDP+ access network	session parameters	session parameters	profile/ policies	session parameters	n/a
Communications Features	NGN-based	NGN-based	NGN-based	out of scope	out of scope	out of scope	out of scope	out of scope	n/a
IM	yes	yes	yes	n/a	n/a	n/a	n/a	n/a	n/a
Presence	yes	yes	yes	n/a	n/	n/a	n/a	n/a	n/a
Buddy List	yes	yes	yes	n/a	n/a	n/a	n/a	n/a	n/a
Interacticity Enabler	n/a	proposed for targeted ads	n/a	out of scope	out of scope	out of scope	out of scope	out of scope	Expositions of pers. EPG
Social IPTV	n/a	n/a	n/a	n/a	n/a	n/a	n/a	n/a	n/a
Shared Experience	n/a	n/a	n/a	n/a	n/a	n/a	n/a	n/a	n/a
Content Sharing	n/a	n/a	n/a	n/a	n/a	n/a	n/a	n/a	n/a

Table 8.2.: Comparison of Related Works

	Mikoczy09	Daher09	Yang09	Yoon10	Waiting08	Kadlic09	Menai09	Arnaud10	Cruz10
Approach	NGNLab/ Scalenet/ TNO IPTV test bed	FOKUS NextGen-Media Lab	NGN-based IPTV using ITU-T standards	NGN-based IPTV using ITU-T standards	UCT IPTV test bed	NGNLab/ UCT test bed	IMS-based IPTV/ CDN test bed @ FT	IMS-based IPTV, user profile and CMS	IMS-based IPTV test bed
System Setup	prototypal implementation and test bed	prototypical implementation	concept and prototype	concept and prototype	IMS test with IPTV integration	IMS test bed	IMS and CDN test bed	test bed infrastructure	test bed allowing dynamic QoS adaptation
Third Party API	SOA, Web-services	SOA, Web-services	SOA, Web-services	SOA Web-services	n/a	n/a	n/a	n/a	n/a
Content Adaptation	n/a	device agnostic through	n/a	n/a	SDP	storage; nPVR functionality	SDP	yes, limited on signaling	yes, limited on signaling
Communications Features	yes, but out of scope	yes, but out of scope	n/a	n/a	n/a	yes, but out of scope	yes, but out of scope	n/a	n/a
IM	n/a	n/a	n/a	n/a	n/a	n/a	n/a	n/a	n/a
Presence	n/a	n/a	n/a	n/a	n/a	n/a	n/a	n/a	n/a
Buddy List	n/a	n/a	n/a	n/a	n/a	n/a	n/a	n/a	n/a
Interacticity Enabler	n/a	n/a	n/a	n/a	n/a	nPVR	n/a	n/a	n/a
Social IPTV	n/a	n/a	n/a	n/a	n/a	n/a	n/a	n/a	n/a
Shared Experience	n/a	n/a	n/a	n/a	n/a	n/a	n/a	n/a	n/a
Content Sharing	n/a	n/a	n/a	n/a	n/a	n/a	n/a	n/a	n/a

Table 8.3.: Comparison of Related Works cont.

	Niamut08	Wilson09	Coppens/ Pelt04	Nathan08	Boertjes08	Harboe/ Tullio8/Huang09	Zhang10	Friedrich
Approach	Services for the TNO IMS-based IPTV test bed	UCT IPTV test bed w/ targeted ad system	Social IPTV system, AmigoTV	Social IPTV system; ColabraTV	Social IPTV system; ConnecTV	Social IPTV system, STV1-3	IMS-IPTV and Social TV system	End-to-end IPTV systems with session-oriented (NGN) and declarative application environment.
System Setup	IMS-IPTV partial implementation	integration into UCT test bed	proprietary with XMPP server	proprietary system	proprietary with XMPP server	proprietary with XMPP server	IMS infrastructure	Test bed and proof-of-concept infrastructure
Third Party API	n/a	n/a	n/a	n/a	n/a	n/a	n/a	SIP/Webservices/JSON-RPC
Content Adaptation	prot. translation SIP-to-RTSP	ad-insertion	n/a	n/a	n/a	n/a	n/a	Dynamic content provisioning through Content Streaming Enabler (CSE)
Communications Features	in-game	out of scope	yes, Social IPTV	yes, Social IPTV	yes, Social IPTV	yes, Social IPTV	yes, Social IPTV+NGN	NGN-based Rich communications, XMPP
IM	n/a	n/a	no	no	no	yes (STV3)	yes	Yes
Presence	n/a	n/a	yes	yes	yes	yes	yes	Rich Presence
Buddy List	n/a	n/a	yes	yes	yes	yes	yes	Yes
Interacticity Enabler	Quiz show	Ad-enabler	n/a	no	no	no	no	Targeted Advertisement, Voting, Quiz
Social IPTV	related to	no	yes	yes	yes	yes	yes	Content sharing and synchronized shared experience
Shared Experience	yes	no	yes	yes	yes	yes	yes	yes, see above
Content Sharing	yes	no	no	no	no	yes, recommendation	yes	yes, see above

Table 8.4.: Comparison of Related Works cont.

8.7. Discussion

Together with the author's work, 25 contributions to the field have been presented in this chapter. Tables 8.2-8.4 allow for a detailed mutual comparison with the author's work.

During analysis of both Social as well as Session-oriented IPTV systems, a strong overlap with regards to discussed scenarios and use-case becomes clear.

Both approaches focus on different research areas, either on the user perspective or technology, respectively.

As described in the contribution of Zhang et al. and finally also through the author's work provided in this thesis, a combination of approaches to and technologies for session-oriented IPTV systems, Social IPTV and open third party APIs allows for the generation of a new TV experience.

9. Summary & Outlook

In the course of this thesis, a session-oriented end-to-end system for IPTV has been presented.

The proposed *Open IPTV Ecosystem Core* (OIEC), describes a session-oriented signaling approach to Linear TV and Video-On-Demand services that incorporates the session principles of the SIP protocol, two different *Interactive Application Environments* and integrated telecommunication services.

First, the fundamentals of IPTV and the transition from the current Internet to a delivery platform for rich media have been discussed. A detailed requirement analysis allowed for the derivation of functional as well as non-functional requirements for the OIEC. This was followed by the design of the OIEC system. Starting with a comparison of three different *Interactive Application Environments,* two of them, namely the *Session-Oriented-Application Environment* (SAE) and the *Declarative Application Environments* (DAE) were then selected for integration into the IPTV system. Through the specification of an architecture, the baseline infrastructure was defined. This allowed for the specification of additional services like the *IPTV Meta Session* concept enabling Social IPTV features and three exemplary interactive applications using the *Session-Oriented Application Environment* (SAE).

Finally, the specifications have been grounded by providing an implementation and evaluation of the proposed concepts through multiple case studies, implementing different aspects as discussed in the course of this thesis.

9.1. Contributions

Results of the work conducted during the process of writing this thesis have been published in the proceedings of over 20 international academic conferences, three journals, one book chapter and over ten tutorials on IPTV and related topics.

From the author's perspective, the most significant achievements of this work are the definition of the end-to-end architecture for IPTV, allowing for session-oriented streaming services, enriched by corresponding interactive services incorporating SAE as well as DAE approaches. Additionally, the introduction of a novel *IPTV Meta Session* concept that enables multi-user, multi-content sessions and Social IPTV has also been recognized as a valid contribution throughout academia.

Finally, the practical implementation of most of the aspects presented in this thesis created a certain visibility in this field of research for the author's and his team's work. These contributions are summarized in this section.

9.1.1. End-to-end Architecture for IPTV

Based on the concept of session-oriented conversational services like VoIP and frameworks for converged all-IP telecommunication infrastructures like *Next Generation Networks* and the *IP Multimedia Subsystem*, a novel architecture for IPTV presents an approach that integrates two *Interactive Application Environments*, namely the SAE and DAE. In addition to that, the architecture named the *Open IPTV Ecosystem Core,* provides mechanisms for extended communication services that allow for shared consumption and/or user-to-user interaction during service consumption.

9.1.2. IPTV Meta Sessions

The concepts concerning, as well as the implementations of, *IPTV Meta Sessions* have mostly contributed to the European research project *"interactive Networked Experiences in Multimedia for You"* or iNEM4U for short[1]. The main innovation achieved in this context lies in the cross-domain capabilities of the specified infrastructure. This infrastructure allows for the integration of and interaction between components and *End Devices* with different capability sets, e.g. on a protocol level with native NGN clients or off-the-shelf TVs with an integrated standard Web browser.

The *Meta Session* concepts have been recognized as an enabler for *Social Media* and *Social IPTV*.

9.1.3. Implementation of the FOKUS Open IPTV Ecosystem

The prototypical implementations conducted for evaluation and validation purposes during work on this thesis have been used to create the *Fraunhofer FOKUS Open IPTV Ecosystem*[2]. The ecosystems represent a reference implementation of various standards for IPTV and go beyond them in certain aspects, e.g. with the integration of two different *Interactive Application Environments*, the *IPTV Meta Session* concept, as well as work on the *IPTV Session State API*. At the time that work on this thesis was being finished, the realized ecosystem was recognized within research and academia, used during interoperability events with various SDOs (e.g. Open IPTV Forum and ETSI TISPAN) and became part of a first Proof-of-Concept deployment in eastern Asia.

[1] Project Website: http://www.inem4u.eu
[2] Project Website: http://www.open-iptv-ecosystem.org

9.2. Discussion of Research Questions

In Chapter 1, a set of research questions were stated, which have been answered within the different chapters of this thesis. This section summarizes the answers to the research questions.

What are the limitations of current, first-generation, streaming and IPTV platforms and what kind of modifications are necessary to overcome these limitations?

- As discussed in Chapter 2 current IPTV platforms mostly exist as siloed solutions, not interacting with other technology domains.

  While telecommunication services have been partly combined with content related services, the integration with professional *Social Media* still exists only on a prototypical level. The biggest challenge in this context is on the one hand, the usability of such combined services on the TV screen, on the other, a technical framework that allows for the described combinatorial services. Furthermore, this framework needs to be accepted by relevant market players, e.g. supported by standardization efforts.

How to provide a unified platform for the combination of content-oriented and communications-oriented services and then identify the roles of the various involved entities?

- Unified platforms for Digital TV services do exist but their reach is often limited to a certain area of the world. Bearing this in mind, it becomes obvious that a single standard or platform for IPTV might not fulfill the different local requirement for such a service. For all that, Internet standards as specified by the IETF or W3C are used worldwide, and might also provide the technologies relevant for IPTV.

  For this reason, the author believes that the approach chosen in this thesis might be correct: Using well established signaling and transport protocols as used in the Internet and declarative languages like CE-HTML and later HTML5 for all visual aspects.

  This approach should provide enough flexibility to allow adoptions in various markets but still keep a baseline technology for all services.

How can services for *interactive IPTV services* be realized based on the integrated *Session-Oriented Application Environment* and specified signaling principles for *IPTV*?

- The session-oriented *IPTV Role Model*, as introduced in this thesis, provides an ideal platform not only for the control of plain streaming services. In fact, the exposure of *IPTV Session* information through the specified *IPTV Service State* and the corresponding *IPTV Service State API* provides an elegant mechanism for building interactive services on top.

An *IPTV Session* allows for the identification and tracking of a user's behaviour during service usage. Furthermore, the *Session Context* can be analyzed and modified by interactive services. This allows for service personalization, e.g. through targeted services, including advertisement, user polls or games. Finally, a complete *Service Oriented Architecture* allows for the integration of access mechanisms for IPTV services into frameworks for telecommunication services. This might result in open APIs allowing third parties to build their telecommunications services enriched with content-related services.

Furthermore, Social Media services could also take part in this opportunity to create integrated shared user-experiences.

How to abstract a multi-user, multi-content platform from single user-to-network relationships?

- According to the MIT's Technology Review[3], *Social Media* and in 2010 also *Social TV* have been recognized as the key drivers for innovation in the IPTV domain, and as one the most emerging technologies in the near future. *Social TV* is bringing together groups of users and content through rich communications.

  During work on this thesis, Social TV concepts have been developed on conceptual and technical levels, providing the necessary infrastructure and enablers to connect users and contents. The developed *Meta Session* model enables the spanning of a virtual network on top of a single user's *IPTV Sessions* and connecting them with other users. An important requirement in this context is the ability to enable *Meta Sessions* independently of the user's access network or *End Device*. This requirement has been fulfilled by implementing connectors using various state-of-the-art protocols and the application logic for *Meta Session Management* loosely coupling these connectors.

How can dynamically composed interactive streaming media be generated and delivered?

- Multiple access networks, multi-screen scenarios and the variety of a user's *End Devices* is growing constantly. The dynamic generation of contents for this heterogeneous puzzle is one of the main challenges for content providers and broadcasters. Contents must be available in multiple formats and multiple bitrates, content mixing and insertion scenarios become more and more relevant with a rising amount of personalized services.

[3]http://www.technologyreview.com/communications/25084/

In the course of this thesis, the requirements for a dynamic provisioning of contents have been analyzed carefully, resulting in a prototypical implementation. It has been demonstrated that the smart combination of a multi-purpose streaming engine, namely the *VideoLAN* project with advanced signaling can provide such a solution.

9.3. Future Directions

The applicability of the proposed system to current state-of-the-art infrastructures has been demonstrated in the discussion of the different case studies in the last chapter. Nevertheless, the speed of innovation in this field of research is high, as interactive IPTV and the Web are both subject to ongoing developments and standardization within telecommunications research.

The presented IPTV system fulfills the requirements of an end-to-end infrastructure for IPTV and interactive services. Nevertheless, and as already mentioned in the comparison of the different *Interactive Application Environments* in Chapter 2, and their mapping onto the proposed architecture, *Session-Oriented Application Environments* are limited with regards to the complexity of applications that can be realized. At this point, *Declarative Applications Environments* create much easier mechanisms for the generation of dynamic applications.

Taking this into account, the harmonization between the integrated SAE and DAE application environments is the main task for continuing work on IPTV topics. In more detail, this implies that browser-driven approaches like the DAE must be completely integrated with the SAE approach, and rely on native signaling stacks and native user interfaces. Initial developments in this direction have already been included in the future specifications from different Standard Development Organizations (SDOs).

A second field of future research lies in the extension of the current focus on the Session Initiation Protocol (SIP) when instantiating sessions on the *IPTV Session Management Enabler*. The concepts for a certain *"Third Party Openness"* as discussed in Chapter 5 can be extended. While this thesis was being written, only SAE applications like the above-discussed Televoting, Targeted Advertisement and Quiz-Show services were able to read and manipulate session data. The author's current research extends this approach and also allows Web technology-oriented front ends to connect to the described IPTV infrastructure. This connection has been achieved by also allowing clients with other mechanisms and protocols like JSON-RPC or Webservices to establish an *IPTV Session.* Details are described in the author's current publications, including *"An IPTV Session State API For Converging Managed And Unmanaged IPTV Infrastructures"* [43] and other ongoing work within the author's team at Fraunhofer FOKUS.

A. Annex

A.1. The Role of NGNs & IMS in Managed IPTV

Managing identities, making sure that sessions get established and ensuring that media flows through the network is quite a set of tasks, and it is the purpose of the IP Multimedia Subsystem (IMS) to offer the mechanism that reaches from the session to the link layer in the traditional OSI model to establish communications paths between users and services [64]. The IMS was originally developed by the Third Generation Partnership Project (3GPP) to provide an architecture for all-IP based mobile networks.

Basically the IMS is intended to keep track of users and manage their traffic, not handle the services they use, or user interactions with the services. This is the reason why it also fits perfectly into the idea of IPTV. The reason for that is that a baseline infrastructure for session handling, identity management and media delivery is also needed here. The following paragraphs will give a brief introduction to the basic ideas behind NGN and IMS, as well as IMS-Based IPTV ecosystems as part of current research and standardization.

A.1.1. NGN Working Definition

Current literature defines NGN in different ways. A working definition can be derived by looking at existing definitions, like the one stated below derived from the specification of the ITU-T, which was one of the main driving bodies of the NGN initiative [73]:

Definition 2 *A Next Generation Network (NGN) is a packet-based network able to provide services including telecommunication services and able to make use of multiple broadband, QoS-enabled transport technologies and in which service-related functions are independent from underlying transport-related technologies. It offers unrestricted access by users to different service providers. It supports generalized mobility which will allow consistent and ubiquitous provision of services to users.*

A.1.2. Development of NGN

The main developments during NGN standardization have taken place within ITU-T. which assigned various working groups to address research areas and create corresponding specifications reflected in [95].

In parallel, the European Telecommunication Standards Institute (ETSI) started its work on NGN standardization. This was inspired by the well-known developments for the Global System for Mobile Communication (GSM). ETSI combined its TIPHON working group – responsible for the development of interconnecting Voice-Over-IP (VoIP) and the PSTN – with the Signaling Protocols and Networks Group (SPAN). This formed the so-called TISPAN group, being still active at the point this thesis was written.

The outcome of both bodies are specifications finished in 2005 are mainly for future conversational services integrating also legacy aspects like the PSTN and also emulating ISDN services.

Adding IPTV to the idea of NGN Going one step beyond and starting in 2006, work on ITU-T and ETSI specifications has been extended to also support multimedia streaming services and IPTV. For this reason, both forums created specific working groups on this topic:

- ITU-T Focus Group on IPTV (FG IPTV) as a separate undertaking and independent of the NGN activities.
- ETSI TISPAN work on Release 2 with special tracks throughout all eight working groups to add IPTV functionality to the R1 specifications.

Having the capability to describe the work needed to add IPTV functionality, the next section will outline the basic architectural approach chosen to build an NGN infrastructure. To simplify this overview from this point forward, only ETSI TISPAN references will be used.

A.1.3. NGN Architecture

The NGN functional architecture, as depicted in Figure A.1 is structured into a service layer and an IP-based transport layer. The service layer is comprised of the following components, called subsystems:

- The Core IP Multimedia Subsystem (IMS), also called Core IMS, described in the next section.
- the PSTN/ISDN Emulation Subsystem (PES), providing access to legacy services.
- other multimedia subsystems (e.g. IPTV) and applications.
- A User Profile hosting user specific data
- So-called Applications making use of the subsystems, the User Profile and the underlying Transport Layer

This subsystem-oriented architecture allows for the easy addition of new subsystems. IP-connectivity is provided to NGN user equipment by the transport layer, under the control of the network attachment subsystem (NASS) and the resource and admission control subsystem (RACS). These subsystems hide the transport technology used in access and core networks below the IP layer [33]. The most important subsystem inside the NGN specifications is the so-called IP Multimedia Subsystem (IMS) and will be presented in the next section.

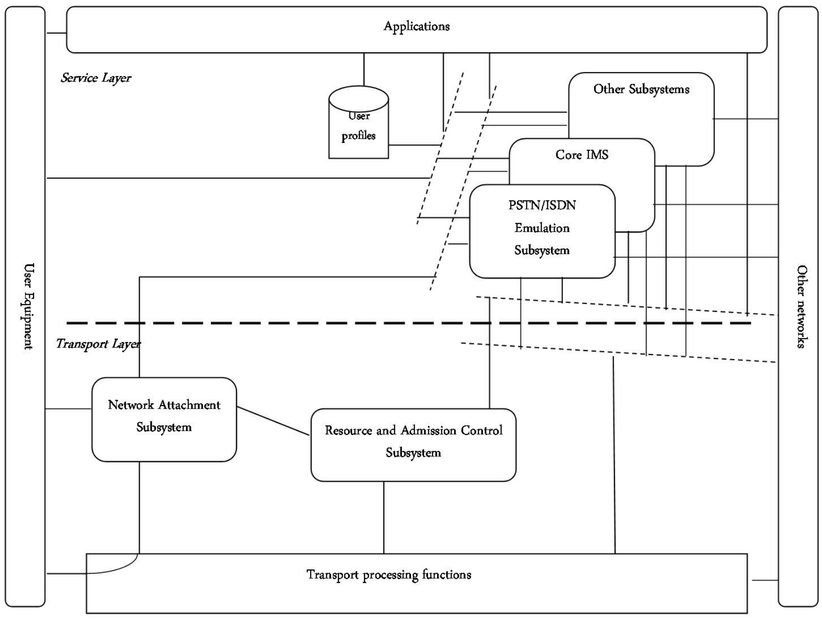

Figure A.1.: ETSI TISPAN NGN Overall Architecture [35]

A.1.3.1. The IMS inside a NGN

The basic idea behind the IP Multimedia Subsystem is the introduction of a unified *Service Delivery Platform* on top of an IP Network. This SDP is able to handle a set of elementary functions required to establish a relationship between a user and a Service Provider. The relationship is founded on setting up a session between the user and the networks using the Session Initiation Protocol (SIP). Details concerning how the SIP protocol will be used for multimedia session signaling and IPTV will be described in Chapter 5.1.1.

The SDP consists of a set of elementary functions which, on the one hand, help to maintain the session and on the other, help to provide basic and more variable services on top. The elementary functions can be identified as:

- A user register, namely the *Home Subscriber Server* (HSS) or *User Profile Serving Function* (UPSF) acting as a central repository for the maintenance of user-related information. Each user has to register in this entity before a service can be obtained.

- The so-called *IMS Core* consisting of a set of so-called *Call Session Control Functions* (CSCFs) is responsible for forwarding user requests for a specific service to the correct entity, namely the correct Application Server.

- An Application Server model based on the idea that each service will be routed to them. The decision-making process takes place within the CSCFs, acting as a trigger point based on a so-called Initial Filter Criteria or a specific service identifier, the so-called *Public Service Identifier* (PSI). The Application Server Model will be discussed in the next section.

In addition to the above-mentioned, other elementary services in the context of the IMS are:

- A *Presence* mechanism based on an Application Server which maintains the current status of a user and informs subscribed users about changes.

- *Multimedia Telephony* incorporating the Session Initiation Protocol for the handshake between two endpoints and the Real Time Streaming Protocol (RTP) to carry the payload.

- *Instant Messaging* allowing for the exchange of text messages between users and acting as the predecessor of the well-known Short Message Service (SMS).

An extremely simplified view of an IMS infrastructure is depicted in Figure A.2. already containing both: conversational clients and potential IPTV end devices and the corresponding Application Servers on top as described later in Section A.1.3.3.

A.1.3.2. Session Principles

Sessions and the corresponding Session Initiation Protocol (SIP) [114] play a key role in any NGN and are described extensively in [15]. SIP defines different methods that allow for the setup of durable states for communication. In the context of this thesis, two methods are especially relevant because they can be used to set up multimedia content sessions and create user to user communication, respectively:

- SIP INVITE sent within the protocol headers and hereby enables session establishments between two endpoints implementing this method. During a simple session setup, only information on IP address and (incoming) ports will be exchanged. Additionally, various other information might be negotiated which might be part of the SIP INVITE messages' bodies and carried either as so-called Session Description (SDP) or as XML.

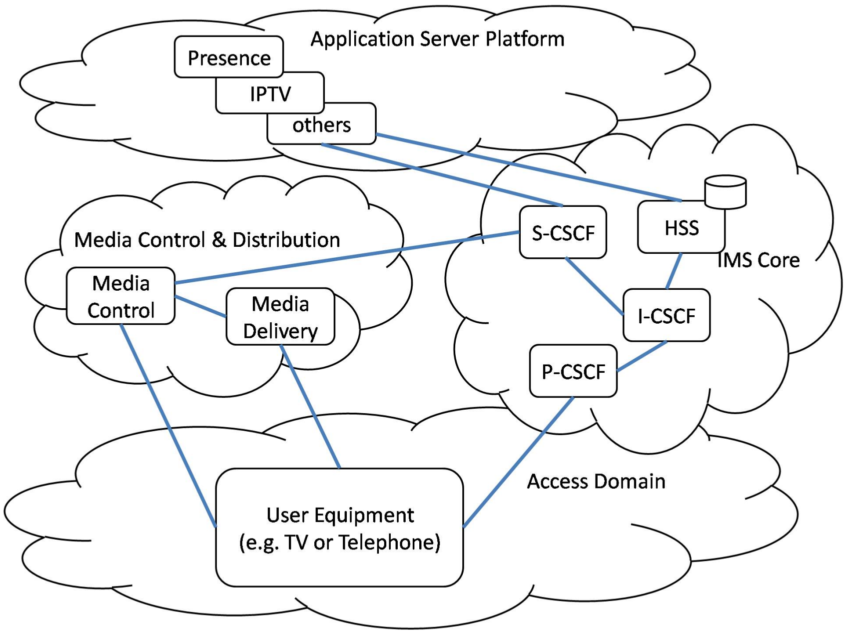

Figure A.2.: Simplified IMS Based IPTV Architecture

- SIP SUBSCRIBE / NOTIFY extends the plain session principles from the SIP INVITE scenario. As sessions might become long-lasting and have a state, updates during an established session might become necessary. For this reason, the so-called SUBSCRIPTION-NOTIFICATION mechanism has been added to the protocol. By subscribing to a certain event, corresponding notification messages will be send out upon session change.

Session Description As already pointed out above, setting up service-specific sessions requires more than the plain protocol header.. For this reason, the SIP protocol allows for session descriptions using the Session Description Protocol (SDP) as well as XML data to be carried inside its body. A detailed description of how SDP and XML will be used can be found in Chapter 5.1.1 in the discussion of the session model for IPTV.

A.1.3.3. Application Server Model

Application Servers play the key role in the IMS when creating new services. Basically, the IMS defines three different types of Application Server, but only the so-called *SIP Application Server* is relevant in the context of this thesis. An Application Server is, at first, nothing more than a SIP endpoint located inside the network. Application Servers are connected to the Core IMS through the so -called ISC interface which uses the SIP protocol.

A SIP Application Server can host and execute services, and additionally also trigger other (Application) servers.

Remembering what was already said in the last sections when talking about Trigger Points , the concepts used to forward incoming SIP requests towards the right Application Server is quiet similar: A so-called Initial Filter Criteria (IFC) is specified inside the Core IMS for service routing, while a Communication Service Identifier (CSID) is carried inside the SIP traffic to identify a service. For IPTV, no such CSID was available at the point this thesis was written.

A.1.3.4. Communication Services

One of the crucial points when creating services, as described later in this thesis (e.g. Session Related Services in Chapter 5.4), is the availability of a basic set of communication enablers. In contrast to other multimedia environments, the IMS relies on services built upon the SIP protocol, which have already been described in Section A.1.3.1. and are built-in features of the IMS.

A.1.4. Summary

The last sections have introduced the basic ideas behind Next Generation Networks and the IMS. Admittedly, just a very brief overview could be provided on this complex topic. What should be kept in mind are especially the basic principles of the SIP session concept, the overall NGN architecture and the Application Server model. These three concepts will be reflected within the next chapters, when building IPTV services on top.

A.2. Conducted Own R&D Projects

This section summarizes the author's work on the four most relevant research projects conducted during work on this thesis. These projects have therefore influenced the author's work on the topic of IPTV and have helped him to develop specifications and prototypes.

A.2.1. Next Generation Television Research Project

As discussed earlier in Chapter 2, when introducing the fundamentals of DTV and IPTV, it has been stated that the transition from Digital Television to IPTV was driven by the telecommunication industry.

While initial standards were set by the DVB, Triple and Quadruple Play scenarios also require broadcast and communications, and therefore telecommunication networks must be integrated in this context. The Next Generation TV (NGTV) project has been recently initiated as an cooperative industry research project with partners from eastern Asia. Its scope had been defined in line with current and future work being carried out for the ETSI TISPAN Release 2 [33] on IMS-based IPTV solutions. The main objective of the project was to prepare both partners for the standardization process and create intellectual property in advance. The project's vision is depicted in Figure A.3, representing a blueprint architecture for Next Generation TV service, and is also described in the following paragraph. The project outcome can be summarized as follows, and corresponds to the typical three-stage process known from ETSI standardization. The project started with a scenario and requirements phase, then one for the definition of an overall architecture and a third phase to specify used protocols and signaling flows:

NextGen TV Stage 1: Scenarios & Requirements Scenarios for NextGenTV focused mainly on the adaption of basic use cases known from Digital Television and advanced use cases allowing for an interaction between TV and telecommunication services – so-called cross-fertilization [1]. Typical use cases in this context have been identified as:

- Linear TV
- Video on Demand
- Incoming call on TV

NextGen TV Stage 2: Architecture The overall NextGenTV architecture has been composed by incorporating the ETSI TISPAN Release 1 [35] architecture and then extended with IPTV functionalities. The extensions to the existing architecture necessary for the support of heterogeneous IPTV access networks has also been analyzed. Furthermore, different IPTV enablers were designed as well. These enablers handle the establishment of the Electronic Program Guide, Content Delivery, and IPTV Session.

Additionally, the impact on existing IMS standards and potential new functional entities has been analyzed.

NextGen TV Stage 3: Service definition In Stage 3, the main interfaces and reference points, including a detailed specification of information to be exchanged, have been defined. In the following interactive message sequence, charts have been created for all designed scenarios.

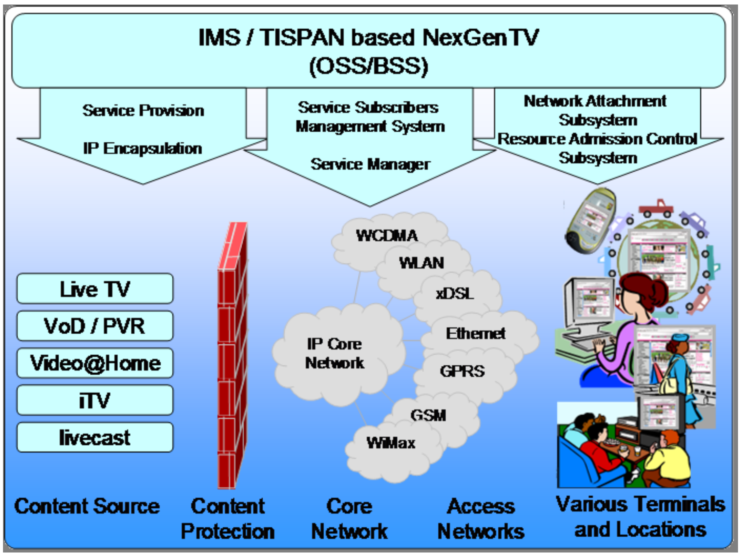

Figure A.3.: Scope of NGTV project

A.2.2. Next Generation Interactive Television Research Project

After making the initial specifications for TV services controlled by and delivered over Next Generation Networks available, the next obvious step was to extend this Service Delivery Platform with enhanced features. A first step in this direction has already been taken, through the introduction of the so-called cross-fertilization scenarios described in Section A.2.1. The Next Generation Interactive Television Project (NGiTV) project took the work performed in the NGTV project into account and the, at that time current, work on scenarios and requirements for the ETSI TISPAN Release 2. Table A.1 presents an overview of basic use cases from NGTV and enhanced use cases from NGiTV. In general, the project proceeded with the same three-stage process used in NGTV. In a first step, enhanced IPTV scenarios for interactive features were created. This was followed by a phase devoted to the creation of a revised architecture. The project outcome was composed of these two phases and a third phase that provided detailed service flows for service signaling.

NextGen iTV Stage 1: Scenarios & Requirements Phase 1 was driven by the analysis of existing technological approaches to interactive television. In this context, it was decided to not concentrate on existing middleware solutions like MHP. The main goal was the derivation of requirements for services that do not require a specific middleware on the end-device or inside the network, but rather use an abstract signaling path with SIP signaling. Defined service scenarios are listed in Table A.1. The main purpose of these scenarios was to create new or revised business models that lower the so-called media break[1] during service usage.

NextGen iTV Stage 2: Architecture The NextGen iTV architecture has been composed as an evolution from NextGen TV (Figure A.4) and has been extended with regard to functional components. This includes additional application logic on the end devices, as well as new components inside the network.
A high level view is depicted in Figure A.5. The main extensions are represented through the introduction of mechanisms for interactive application signaling from a third-party provider, throughout the network in the direction of the user.

NextGen iTV Stage 3: Service Definition The service definition stage included the specification of various interactive services on a protocol level. These have later been used as input for corresponding processes in the IPTV standardization process in ETSI TISPAN.

A.2.3. IMS-IPTV Project

Achieving interoperability between network elements is one of the main goals of standardization. The Next Generation Networks and IPTV project (IMS-IPTV) has been set up with a partner from the industry and is focused on demonstrating the interoperability of different IPTV-related network elements and middleware.
The project's main goal has been defined as follows:

"Connecting the Fraunhofer FOKUS IPTV Media Client with a commercial IMS core network and a legacy IPTV infrastructure."

In order to meet the project requirements, the project started with the identification of necessary network entities and the relationship between them. A high-level architecture is presented in Figure A.6. The main building blocks touched on during work for this architecture have been identified as:

[1]Whenever the medium changes during a work process, a media break takes place. In most cases, the result is a higher amount of work and a disruption of the work routine. E.g. switch from a mobile device to the TV's RC.

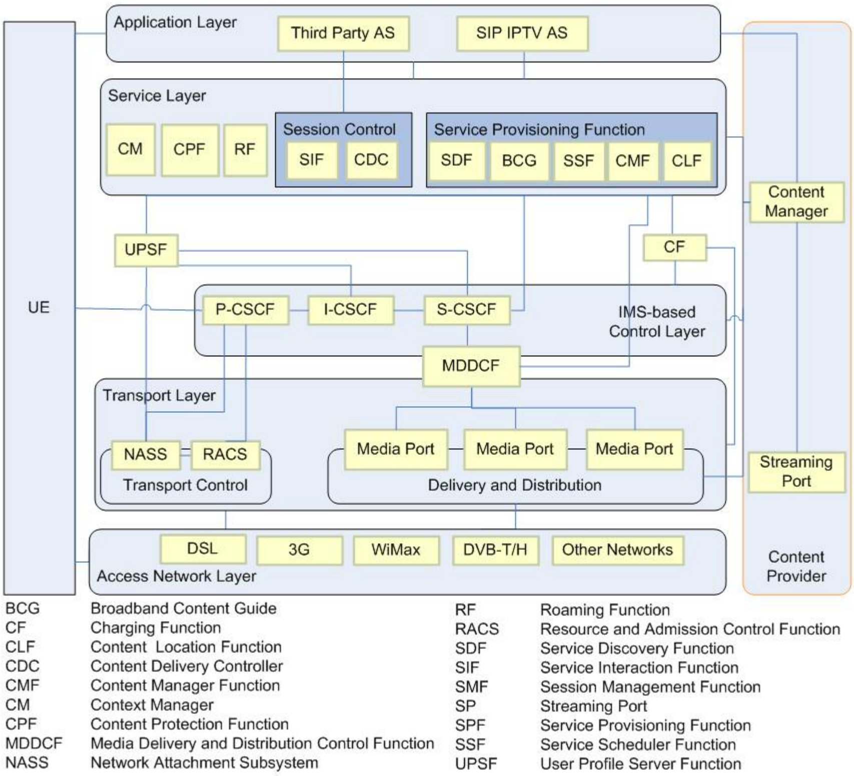

Figure A.4.: NextGenTV & NextGen TV Architecture

Use-Case	NextGen TV	NextGen iTV
Live TV	✔	✔
Video on Demand	✔	✔
Incoming Call on TV	✔	✔
Video on Demand	✔	✔
Interactive Voting	✗	✔
Targeted Advertisements	✗	✔
Interactive Shopping	✗	✔
Push to Share Content	✗	✔
Video Follow Me	✗	✔
Event driven notification	✗	✔
Targeted Advertisements	✗	✔

Table A.1.: NextGen TV vs. NextGen iTV use-cases

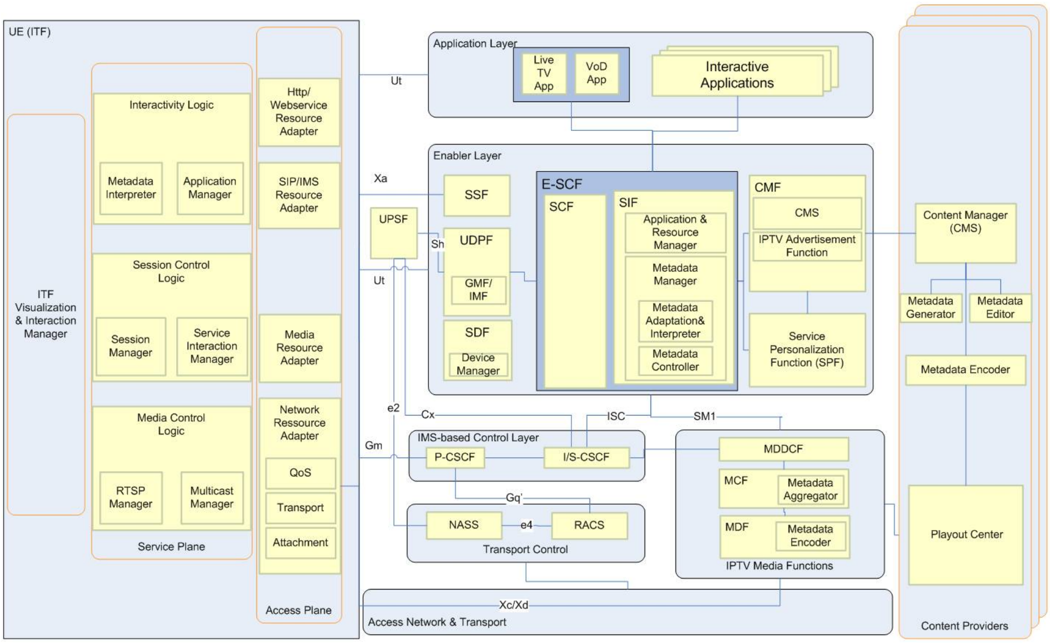

Figure A.5.: NextGen iTV Architecture

The FOKUS Media Client acting as an IMS IPTV User Agent (UA) combining IPTV related features for the consumption of streaming media while enabling IMS communication features.

The FOKUS IPTV Application Server acting as an Application Server within the commercial IMS system and providing session management and connectivity towards the legacy IPTV infrastructure. By making these two components available, the interworking between both systems could be achieved. The project's results also play a key role in the validation of the infrastructure presented in the scope of this thesis.

A.2.4. FP 7 Research Project iNEM4U

Consumers have a wide variety of (media) services at their disposal, but these services are traditionally bound to specific devices and networks, such as IPTV networks, the Internet, and in-home and mobile networks.

As a result, it can be very difficult for users and service providers to combine content and services from different networks into a single interactive experience [62]. To overcome this problem, a novel service platform has been developed within the iNEM4U project. This platform allows applications to manage sessions across networks easily, in particular for the sharing of services and content in relatively

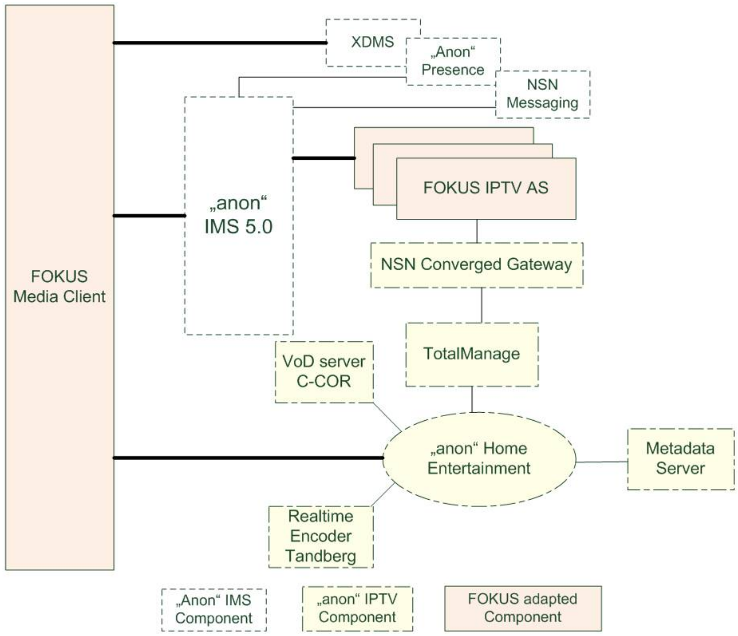

Figure A.6.: IMS-IPTV architecture

small groups of users, such as friends or families. To accomplish this, the platform allows for the management of multi-user sessions across networks. In addition, the platform supports various other innovative enabling services, such as context-aware recommendations and cross-network identity management.

The author's main contribution to the project was in the reference model and implementations for the so-called *Meta Session Manager* that acts as the core enabler for the above-mentioned Shared Experience. iNEM4U scenarios and, in parallel, the high level architecture has also been visualized in Figure A.7. Beside the iNEM4U platform in the center of the picture, a variety of different so-called iNEM4U clients are connected to the iNEM4U network through their dedicated and platform-specific protocols. This includes an NGN-IPTV-Client using NGN/IMS signaling, a CE-HTML TV-set and a mobile client making use of various Webservices offered by the infrastructure. In summary, these clients have later been used to drive different scenarios like:

- A cross domain EPG
- Smart Advertising Service
- Concert Community Channel
- VoD E-Shop

- iNEM4U Shared & Synchronized Experience

The iNEM4U results, including the author's contributions to the iNEM4U iSession and the NGN-driven IPTV Client, have been published in various papers.

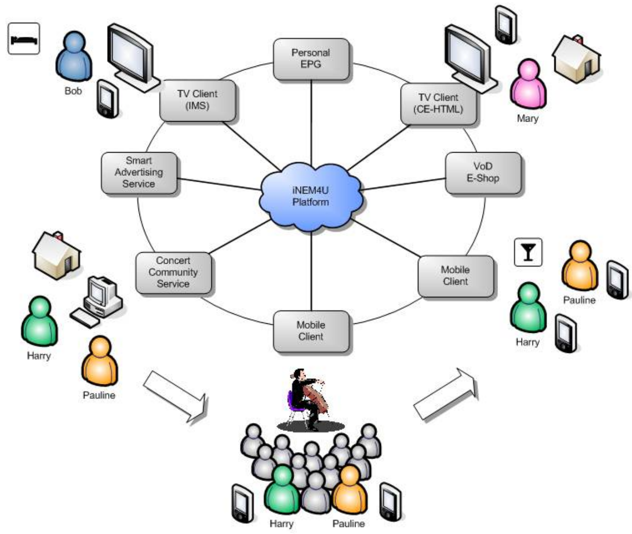

Figure A.7.: FP 7 iNEM4U High Level Architecture

List of Abbreviations

ANSI American National Standards Institute

ARIB Association of Radio Industries and Businesses

ATM Asynchronous Transfer Mode

ATSC Advanced Television Systems Committee

BSS Billing Support System

CA Conditional Access

CDN Content Delivery Network

CSCF Call Session Control Function

DAE Declarative Application Environment

DLNA Digital Network Living Alliance

DOCSIS Data Over Cable Service Interface Specification

DRM Digital Rights Management

DTV Digital Television

DVB Digital Video Broadcasting Group

EVN Ethernet Virtual Connection

FMBC Fixed Mobile Broadcast Convergence

GEM Globally Executable MHP

HSS Home Subscriber Server

HTPC Home Theater PC

IDE Integrated Development Environment

IM Instant Message

IMS IP Multimedia Subsystem

IP Internet Protocol

ISP Internet Service Provider

ITU-T International Telecommunication Union Telecommunication Sector

JDO Java Data Object

MPLS Multi Protocol Label Switching

nPVR networked Personal Video Recorder

OMA Open Mobile Alliance

OSA Open Service Architecture

OSS Operational Support System

OTT Over The Top

PAE Procedural Application Environment

PiP Picture in Picture

PoC Proof of Concept

PSTN Public Switched Telephone Network

RIA Rich Internet Application

RSP Retail Service Provider

SAE Session-Oriented Application Environment

SDH Synchronous Digital Hierarchy

SPD Service Provider Discovery

SUT System Under Test

TISPAN Telecommunications and Internet converged Services and Protocols for Advanced Networking

UA User Agent

VoIP Voice Over Internet Protocol

VQS Virtual Quiz Show

Bibliography

[1] A. Al-Hezmi et al. "Cross-fertilization of IMS and IPTV services over NGN". In: *Innovations in NGN: Future Network and Services, 2008. K-INGN 2008. First ITU-T Kaleidoscope Academic Conference*. 2008, pp. 153 –160. DOI: 10.1109/KINGN.2008.4542261.

[2] A. Al-Hezmi et al. "Provisioning IMS-based seamless triple play services over different access networks". In: *Network Operations and Management Symposium, 2008. NOMS 2008. IEEE*. 2008, pp. 927 –930. DOI: 10.1109/NOMS.2008.4575249.

[3] A. Al-Hezmi et al. "Provisioning of an Open NGN/Triple Play Toolkit and Testbed". In: *Testbeds and Research Infrastructure for the Development of Networks and Communities, 2007. TridentCom 2007. 3rd International Conference on*. 2007, pp. 1 –6. DOI: 10.1109/TRIDENTCOM.2007.4444713.

[4] A. Al-Hezmi et al. "Requirements for an IMS-based Quadruple Play Service Architecture". In: *Network, IEEE* 21.2 (2007), pp. 28 –33. ISSN: 0890-8044. DOI: 10.1109/MNET.2007.334309.

[5] Adel Al-Hezmi et al. "Interactive multimedia services over open NGN testbed". In: *TridentCom '08: Proceedings of the 4th International Conference on Testbeds and research infrastructures for the development of networks & communities*. Innsbruck, Austria: ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), 2008, pp. 1–6. ISBN: 978-963-9799-24-0.

[6] Julien Arnaud et al. "Adaptive IPTV services based on a novel IP Multimedia Subsystem". In: *Multimedia Tools and Applications* (2010). 10.1007/s11042-010-0576-1, pp. 1–20. ISSN: 1380-7501. URL: http://dx.doi.org/10.1007/s11042-010-0576-1.

[7] Consumer Electronics Association. *Web-based Protocol and Framework for Remote User Interface on UPnP Networks and the Internet (Web4CE)*. 2010. URL: http://electronics.ihs.com/document/abstract/XUTZACAAAAAAAAAA.

[8] M. Said B. Chatras. "Delivering Quadruple Play with IPTV over IMS". In: *International Conference on Intelligence in Networks (ICIN) 2007, Bordeaux,France*. 2007.

[9] Alexandros Bakalakos et al. *Future Internet and NGN Design requirements and principles for a Future Media and 3D Internet*. Feb. 2009. URL: http://cordis.europa.eu/fp7/ict/netmedia/publications_en.html.

[10] Herve Benoit. *Digital Television Satellite, Cable, Terrestrial,IPTV, Mobile TV in the DVB Framework*. Focal Press, 2006. ISBN: 9780240520810.

[11] Anne Bodzinga and Susan White. "Interworking IPTV Services with IMS". In: *Telecommunications Network Strategy and Planning Symposium, 2006. NETWORKS 2006. 12th International*. 2006, pp. 1 –5. DOI: 10.1109/NETWKS.2006.300364.

[12] E. BOERTJES. "ConnecTV: Share the experience". In: *In Proceedings of EuroITV 2007 (TICSP)*. 2007, pp. 139–140.

[13] David Boswarthick. *ETSI IPTV Standards - Visible Benefits for your Business*. 2008. URL: http://portal.etsi.org/.

[14] D. C. A. Bulterman et al. "Television Content Enrichment And Sharing: The Ambulant Annotator". en. In: *Social Interactive Television: Immersive Shared Experiences and Perspectives*. Ed. by P. S. Cesar Garcia, D. Geerts, and K. Chorianopoulos. first. Hershey (PA), USA: IGI Global Publishing, 2009, pp. 67 –75. ISBN: 978-1-60566-656. URL: http://books.google.com/books?id=YMEEsVPysyAC\&pg=PA67\&lpg=PA67#v=onepage\&q=\&f=false.

[15] Gonzalo Camarillo and Miguel-Angel Garcia-Martin. *The 3G IP Multimedia Subsystem (IMS): Merging the Internet and the Cellular Worlds, Second Edition*. John Wiley & Sons, 2006. ISBN: 0470018186.

[16] B. Campbell, R. Mahy, and C. Jennings. *The Message Session Relay Protocol (MSRP)*. RFC 4975 (Proposed Standard). Internet Engineering Task Force. Sept. 2007. URL: http://www.ietf.org/rfc/rfc4975.txt.

[17] B. Campbell et al. *Session Initiation Protocol (SIP) Extension for Instant Messaging*. RFC 3428 (Proposed Standard). Internet Engineering Task Force. Dec. 2002. URL: http://www.ietf.org/rfc/rfc3428.txt.

[18] Hu Mas Ivars Cedervall Horn and Näsström. *Open IPTV Forum Toward an open IPTV standard*. 2000. URL: http://www.ericsson.com/ericsson/corpinfo/publications/review/2007_03/01.shtml.

[19] David D. Clark et al. "Tussle in cyberspace: defining tomorrow's internet". In: *Proceedings of the 2002 conference on Applications, technologies, architectures, and protocols for computer communications*. SIGCOMM '02. Pittsburgh, Pennsylvania, USA: ACM, 2002, pp. 347–356. ISBN: 1-58113-570-X. DOI: http://doi.acm.org/10.1145/633025.633059. URL: http://doi.acm.org/10.1145/633025.633059.

[20] European Commission. *ICT INFORMATION AND COMMUNICATION TECHNOLOGIES A Theme for research and development under the specific programme Cooperation Work Programme 200708*. 2007. URL: http://cordis.europa.eu/fp7/ict/.

[21] iNEM4U Consortium. *iNEM4U | interactive Networked Experiences in Multimedia for You Sharing interactive multimedia experiences in a converged world*. http://www.inem4u.eu. 2009.

[22] Microsoft Cooperation. *Microsoft Mediaroom Architecture Guide for ADK Developers*. 2009. URL: http://www.microsoft.com/mediaroom/default.aspx.

[23] Coppens, Trappeniers, and Godon. "AmigoTV: towards a social TV experience". In: *In Proceedings of the European Interactive Television Conference (EuroITV'04)*. 2004.

[24] Rui Cruz et al. "IPTV architecture for an IMS environment with dynamic QoS adaptation". In: *Multimedia Tools and Applications* (2010). 10.1007/s11042-010-0537-8, pp. 1–33. ISSN: 1380-7501. URL: http://dx.doi.org/10.1007/s11042-010-0537-8.

[25] Gayass Daher. "Entwurf und Entwicklung eines generischen Audio/Video Streaming Enablers für SOA-basierte NGN Dienstplattformen". MA thesis. Technische Universität Berlin, Institut für Telekommunikationssysteme, Lehrstuhl Architektur der Vermittlungsknoten, 2009.

[26] Walter Dees and Paul Shrubsole. "Web4CE: accessing web based applications on consumer devices". In: *Proceedings of the 16th international conference on World Wide Web (WWW'07)*. 2007, pp. 1303–1304.

[27] Omar A. Niamut Oskar van Deventer Fabian Walraven Hans Stokking. "Interworking SIP and RTSP to achieve studi controlled uploading of live streaming user generated content". In: *NGNM 2008 5th International Workshop on Next Generation Networking Middleware*. 2007.

[28] Doug Dominiak. *Standardizing a Web-based Application Environment*. Sept. 2000. URL: http://www.w3.org/2000/09/Papers/Motorola.html.

[29] DVB. *MHP platform harmonisation. White Paper, April 2004*. 2004. URL: http://www.dvb.org.

[30] Leslie Ellis. *Advanced Advertising and SCTE 130*. Jan. 2010. URL: http://www.translation-please.com/column.cfm?columnid=232.

[31] Leslie Ellis. *Local-into-Digital Ad Splicing: Part 2*. Jan. 2010. URL: http://www.translation-please.com/column.cfm?columnid=150.

[32] ETSI. *Broadcast and On-line Services: Search, select and rightful use of content on personal storage systems*. Ed. by ETSI Secretariat. ETSI, 2006.

[33] ETSI. *ETSI ES 282 001: Telecommunications and Internet converged Services and Protocols for Advanced Networking (TISPAN); NGN Functional Architecture Release 2*. ETSI secretariat, 2008.

[34] ETSI. *ETSI TS 102 034. DVB over IP-based Networks (DVB-IPI)*. ETSI secretariat, 2005.

[35] ETSI. *Telecommunications and Internet converged Services and Protocols for Advanced Networking (TISPAN);NGN Release 1;Release definition*. Tech. rep. ETSI, 2006.

[36] ETSI. *TS 101 812. Digital Video Broadcasting (DVB); Multimedia Home Platform (MHP) Specification. ETSI secretariat*. ETSI secretariat, 2003,2005,2007.

[37] ETSI. *TS 102796 Hybrid Broadcast Broadband TV (HbbTV)*. ETSI secretariat, 2010.

[38] ETSI. *TS 181016 Telecommunications and Internet converged Services and Protocols for Advanced Networking (TISPAN); Service Layer Requirements to Integrate NGN Services and IPTV*. ETSI secretariat, 2008.

[39] ETSI. *TS 182027 Telecommunications and Internet converged Services and Protocols for Advanced Networking (TISPAN); IPTV Architecture; IPTV functions supported by the IMS subsystem*. ETSI secretariat, 2008.

[40] ETSI. *TS 183063 Telecommunications and Internet converged Services and Protocols for Advanced Networking (TISPAN); IMS based IPTV Stage 3 Specification*. ETSI secretariat, 2008.

[41] O. Friedrich and S. Arbanowski. "Enhanced IPTV service control media delivery in Next Generation Networks". In: *Internet Multimedia Services Architecture and Applications (IMSAA), 2009 IEEE International Conference on*. 2009, pp. 1 –5. DOI: `10.1109/IMSAA.2009.5439466`.

[42] O. Friedrich, R. Seeliger, and S. Arbanowski. "Interactive and Personalized Services for an Open IMS-Based IPTV Infrastructure". In: *Networking, 2008. ICN 2008. Seventh International Conference on*. 2008, pp. 302–307. DOI: `10.1109/ICN.2008.65`.

[43] O. Friedrich, D. Thatmann, and S. Arbanowski. "An IPTV Service State API for converging managed and unmanaged IPTV infrastructures". In: *Multimedia and Expo (ICME), 2010 IEEE International Conference on*. 2010, pp. 1493 –1498. DOI: `10.1109/ICME.2010.5583284`.

[44] O. Friedrich et al. "A Platform and Services for Standardized IPTV Ecosystems". In: *Consumer Communications and Networking Conference, 2009. CCNC 2009. 6th IEEE*. 2009, pp. 1 –2. DOI: `10.1109/CCNC.2009.4784836`.

[45] O. Friedrich et al. "IPTV User Equipment for IMS-Based Streaming Services". In: *Network Operations and Management Symposium Workshops, 2008. NOMS Workshops 2008. IEEE*. 2008, pp. 21 –25. DOI: 10.1109/NOMSW.2007.6.

[46] O. Friedrich et al. "Next Generation IPTV services for an extended IMS architecture". In: *Autonomous Decentralized Systems, 2007. ISADS '07. Eighth International Symposium on*. 2007, pp. 429 –436. DOI: 10.1109/ISADS.2007.52.

[47] O. Friedrich et al. "Service Discovery & Selection for Next Generation IPTV Environments". In: *Digital Telecommunications, 2008. ICDT '08. The Third International Conference on*. 2008, pp. 152–156. DOI: 10.1109/ICDT.2008.21.

[48] O. Friedrich et al. "User equipment for converged IPTV and telecommunication services in next generation networks". In: *Human System Interactions, 2008 Conference on*. 2008, pp. 879 –883. DOI: 10.1109/HSI.2008.4581559.

[49] Oliver Friedrich, Stefan Arbanowski, and Thomas Magedanz. "Enabling Multimedia Services over Distributed NGNs". In: *The Second International Conference on Communications and Electronics (HUT-ICCE)*. Hoi An, Vietnam 2008.

[50] Oliver Friedrich et al. *Deliverable: D3.1 Detailed Architecture of the iNEM4U Cross-Domain Service Infrastructure*. 2008. URL: http://www.inem4u.eu.

[51] Oliver Friedrich et al. "Evolution of Next Generation Networks towards an Integrated Platform for IMS-Based IPTV Services". In: *SAINT-W '07: Proceedings of the 2007 International Symposium on Applications and the Internet Workshops*. Washington, DC, USAO: IEEE Computer Society, 2007, p. 10. ISBN: 0-7695-2757-4. DOI: http://dx.doi.org/10.1109/SAINT-W.2007.46.

[52] Oliver Friedrich et al. *High Level Architecture Overview for IMS-based Solution for IPTV Standardization*. Oct. 2006. URL: http://www.slideshare.net/RockyS11/11ttd245docdoc.

[53] Oliver Friedrich et al. *iNEM4U - Interactive Networked Experience in Multimedia for You - D1.3 Overall Architecture of the iNEM4U Platform*. 2008. URL: http://www.inem4u.eu.

[54] Oliver Friedrich et al. "Prototyping Interactive and Personalized IPTV-Services on Top of Open IMS Infrastructures". In: *EUROITV '08: Proceedings of the 6th European conference on Changing Television Environments*. Salzburg, Austria: Springer-Verlag, 2008, pp. 204–208. ISBN: 978-3-540-69477-9. DOI: http://dx.doi.org/10.1007/978-3-540-69478-6_27.

[55] D. Goergen et al. "A Session Model for Cross-Domain Interactive Multi-User IPTV". In: *Consumer Communications and Networking Conference (CCNC), 2010 7th IEEE*. 2010, pp. 1 –6. DOI: 10.1109/CCNC.2010.5421792.

[56] R. Grafe and O. Friedrich. “NGN based IPTV and telecommunication services for the Vista Media Center”. In: *Telecommunications, 2009. ConTEL 2009. 10th International Conference on*. 2009, pp. 429 –434.

[57] gstreamer. *gstreamer open source multimedia framework*. Mar. 2010. URL: `http://www.gstreamer.net`.

[58] M. Handley et al. *SIP: Session Initiation Protocol*. RFC 2543 (Proposed Standard). Obsoleted by RFCs 3261, 3262, 3263, 3264, 3265. Internet Engineering Task Force. Mar. 1999. URL: `http://www.ietf.org/rfc/rfc2543.txt`.

[59] Gunnar Harboe et al. “Ambient social tv: drawing people into a shared experience”. In: *Proceeding of the twenty-sixth annual SIGCHI conference on Human factors in computing systems*. CHI ’08. Florence, Italy: ACM, 2008, pp. 1–10. ISBN: 978-1-60558-011-1. DOI: `http://doi.acm.org/10.1145/1357054.1357056`. URL: `http://doi.acm.org/10.1145/1357054.1357056`.

[60] Gunnar Harboe et al. “The uses of social television”. In: *Computers in Entertainment* 6.1 (2008). DOI: `http://doi.acm.org/10.1145/1350843.1350851`.

[61] Gilbert Held. *Understanding IPTV (Informa Telecoms & Media)*. Boston, MA, USA: Auerbach Publications, 2006. ISBN: 0849374154.

[62] C. Hesselman et al. “Sharing enriched multimedia experiences across heterogeneous network infrastructures”. In: *Communications Magazine, IEEE* 48.6 (2010), pp. 54 –65. ISSN: 0163-6804. DOI: `10.1109/MCOM.2010.5473865`.

[63] Christian Hesselmann, Oliver Friedrich, et al. “An Open Service Infrastructure for Enriching Networked Interactive Multimedia Experiences in a Converged World”. In: *NEM Summit 2008, St. Malo, France*. 2008.

[64] Johan Hjelm. *Why IPTV: Interactivity, Technologies, Services*. Wiley Publishing, 2008. ISBN: 0470998059, 9780470998052.

[65] Paola Hobson et al., eds. *Knowledge-Based Media Analysis for Self-Adaptive and Agil Multi-Media, Proceedings of the European Workshop for the Integration of Knwoledge, Semantics and Digital Media Technology, EWIMT 2004, November 25-26, 2004, London, UK*. QMUL, 2004. ISBN: 0-902-23810-8.

[66] *HTML5 W3C Working Draft*. Apr. 2010. URL: `http://dev.w3.org/html5/spec/Overview.html`.

[67] Elaine M. Huang et al. “Of social television comes home: a field study of communication choices and practices in tv-based text and voice chat”. In: *CHI*. Ed. by Dan R. Olsen Jr. et al. ACM, 2009, pp. 585–594. ISBN: 978-1-60558-246-7.

[68] International Engineering Consortium (IEC). *Intelligent Network (IN)*. 2007. URL: `http://www.iec.org`.

[69] Mohammad Ilyas and Syed A. Ahson. *IP Multimedia Subsystem (IMS) Handbook*. Boca Raton, FL, USA: CRC Press, Inc., 2008. ISBN: 1420064592, 9781420064599.

[70] International MHEG Promotion Alliance (IMPALA). *ETSI ES202184 2.1.1*. Tech. rep. ETSI, 2009.

[71] Apple Inc. *Definition for Application Environment*. 2003. URL: docs.info.apple.com/article.html.

[72] International Standardization Organization (ISO). *SO/IEC DSM-CC Draft International Standard MPEG-2 Digital Storage Media. Command and Control. ISO/IEC JTC1/SC29/WGll Nll00*. 1995. URL: http://webstore.iec.ch/preview/info_isoiec13818-6%7Bed1.0%7Den.pdf.

[73] ITU-T. *NGN Working Definition*. Dec. 2009. URL: http://www.itu.int/ITU-T/workingdefinition.html.

[74] Dan R. Olsen Jr. et al., eds. *Proceedings of the 27th International Conference on Human Factors in Computing Systems, CHI 2009, Boston, MA, USA, April 4-9, 2009*. ACM, 2009. ISBN: 978-1-60558-246-7.

[75] R. Kadlic, E. Mikoczy, and P. Podhradsky. "Advance PVR Applications in IMS Based IPTV Environment". In: *Systems, Signals and Image Processing, 2009. IWSSIP 2009. 16th International Conference on*. 2009, pp. 1 –4. DOI: 10.1109/IWSSIP.2009.5367740.

[76] K. Knightson, N. Morita, and T. Towle. "NGN architecture: generic principles, functional architecture, and implementation". In: *Communications Magazine, IEEE* 43.10 (2005), pp. 49 –56. ISSN: 0163-6804. DOI: 10.1109/MCOM.2005.1522124.

[77] M. Korling. "Evolution of open IPTV standards and services". In: *Innovations in NGN: Future Network and Services, 2008. K-INGN 2008. First ITU-T Kaleidoscope Academic Conference*. 2008, pp. 11 –14. DOI: 10.1109/KINGN.2008.4542243.

[78] Isidro Laso-Ballesteros et al. *Future Internet and NGN Design requirements and principles for a Future Media and 3D Internet*. Tech. rep. 2009. URL: ftp://ftp.cordis.europa.eu/pub/fp7/ict/docs/netmedia/20090220-fid-rp-3-dg_en.pdf.

[79] Isidro Laso-Ballesteros et al. *User Centric Future Media Internet*. Tech. rep. 2008. URL: ftp://ftp.cordis.europa.eu/pub/fp7/ict/docs/netmedia/ucm-white-paper_en.pdf.

[80] A. Liotta and Ling Lin. "Managing P2P services via the IMS". In: *Integrated Network Management, 2007. IM '07. 10th IFIP/IEEE International Symposium on*. 2007, pp. 586–600. DOI: 10.1109/INM.2007.374822.

[81] Jihye Lyu et al. "Design of Open APIs for Personalized IPTV Service". In: *Advanced Communication Technology, The 9th International Conference on*. Vol. 1. 2007, pp. 305–310. DOI: 10.1109/ICACT.2007.358361.

[82] I. Mas et al. "IMS-TV: An IMS-based architecture for interactive, personalized IPTV". In: *Communications Magazine, IEEE* 46.11 (2008), pp. 156 –163. ISSN: 0163-6804. DOI: 10.1109/MCOM.2008.4689259.

[83] M.F. Menai et al. "Demonstration of Standard IPTV Content Delivery Network Architecture Interfaces: Prototype of Standardized IPTV Unicast Content Delivery Server Selection Mechanisms". In: *Consumer Communications and Networking Conference, 2009. CCNC 2009. 6th IEEE*. 2009, pp. 1–2. DOI: 10.1109/CCNC.2009.4785012.

[84] Leandro Santana Menezes. "SIP Based IPTV Architecture for Heterogenous Networks". PhD thesis. Instituto Superior Técnico Universidada Técnica de Lisboa, 2009.

[85] Microsoft. *Mediaroom: Subscriber base nears 4M as new deployments continue and operators begin to introduce interactive TV apps and triple-screen services*. 2009. URL: http://www.telecomtv.com/comspace_newsDetail.aspx?n=45548&id=4e4f5c10-ac11-4415-b27b-5f1c7a4fe545.

[86] Sun Microsystems. *Personal basis profile specification (JSR 217),ver. 1.1*. 2005. URL: http://java.sun.com/products/personalbasis/.

[87] SUN Microsystems. *Project SailFin*. https://sailfin.dev.java.net. 2009.

[88] E. Mikoczy et al. "Combinational services of NGN based IPTV". In: *Testbeds and Research Infrastructures for the Development of Networks Communities and Workshops, 2009. TridentCom 2009. 5th International Conference on*. 2009, pp. 1 –8. DOI: 10.1109/TRIDENTCOM.2009.4976211.

[89] E. Mikoczy et al. "IPTV Systems, Standards and Architectures: Part II - IPTV Services over IMS: Architecture and Standardization". In: *Communications Magazine, IEEE* 46.5 (2008), pp. 128–135. ISSN: 0163-6804. DOI: 10.1109/MCOM.2008.4511659.

[90] Eugen Mikoczy et al. "IMS based IPTV services: architecture and implementation". In: *MobiMedia '07: Proceedings of the 3rd international conference on Mobile multimedia communications*. Nafpaktos, Greece: ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), 2007, pp. 1–7. ISBN: 978-963-06-2670-5.

[91] D. Minoli. *IP MULTICAST WITH APPLICATIONS TO IPTV AND MOBILE DVB-H*. John Wiley Sons, 2008, p. 280.

[92] Marie José Montpetit and Natalie Klym. *Innovation at the Edge: Social TV and Beyond*. Aug. 2008. URL: http://cfp.mit.edu/publications/CFP_Papers/Social_TV_Final_2008.08.01.pdf.

[93] Steven Morris and Anthony Smith Chaigneau. *Interactive TV Standards: A Guide to MHP, OCAP, and JavaTV*. Focal Press, 2005. ISBN: 0240806662.

[94] Mukesh Nathan et al. "CollaboraTV: making television viewing social again". In: *UXTV '08: Proceeding of the 1st international conference on Designing interactive user experiences for TV and video*. Silicon Valley, California, USA: ACM, 2008, pp. 85–94. ISBN: 978-1-60558-100-2. DOI: http://doi.acm.org/10.1145/1453805.1453824.

[95] ITU-T NGN-GSI. *Recommendation Y.2001, General Overview of NGN. Technical*. Tech. rep. ITU-T, 2004.

[96] Jakob Nielsen. *Usability Engineering*. San Francisco, California: Morgan Kaufmann Publishers, 1994. ISBN: 0125184069.

[97] Gerard O'Driscoll. *Next Generation IPTV Services and Technologies*. WILEY INTERSCIENCE, 2008.

[98] OIPF. *Open IPTV Release 1 Specification Volume 5 Declarative Application Environment V1.0, January 5, 2009*. Open IPTV Forum, 2008.

[99] OIPF. *Open IPTV Release 1 Specification Volume 5 Overview V1.0, January 5, 2009*. Open IPTV Forum, 2008.

[100] Open Mobile Alliance (OMA). *Mobile Advertising Architecture, Candidate Version 1.0 07 Jul 2009*. Tech. rep. OMA, 2009.

[101] Tadashi Onodera. *Towards Age of Convergence*. 2009. URL: http://www.ituaj.jp/pb/files/NB21-2_KDDI.pdf.

[102] *Open IPTV Forum Release 1 Specification*. Open IPTV Forum. Dec. 2008. URL: http://www.oipf.tv/docs/Release1/Release1_1/OIPF-T1-R1-Specification-Volume-1-Overview-V1_1-2009-10-08.pdf.

[103] OpenTV. *Interactive Advertising Whitepaper*. 2005. URL: http://jobfunctions.bnet.com/abstract.aspx?docid=241872.

[104] M. Pelt et al. "AmigoTV: A Medianet Application". In: *EWIMT*. Ed. by Paola Hobson et al. QMUL, 2004. ISBN: 0-902-23810-8.

[105] J. Piesing. "The DVB Multimedia Home Platform (MHP) and Related Specifications". In: *Proceedings of the IEEE* 94.1 (2006), pp. 237–247. ISSN: 0018-9219. DOI: 10.1109/JPROC.2005.860997.

[106] M.A. Qadeer and A.H. Khan. "Multimedia Distribution over IPTV and its Integration with IMS". In: *Data Storage and Data Engineering (DSDE), 2010 International Conference on*. 2010, pp. 101 –105. DOI: 10.1109/DSDE.2010.64.

[107] U. H. Reimers. "DVB The Family of International Standards for Digital Video Broadcasting". In: *Proceedings of the IEEE* 94.1 (Jan. 2006), 173 to 182. DOI: 10.1109/JPROC.2005.861004.

[108] Ulrich Reimers. *DVB The Family of International Standards for Digital Video Broadcasting*. Springer Verlag, 2004. ISBN: 354043545X.

[109] C. Riede et al. "Interactive IMS-based IPTV plunge into the reality game show". In: *Internet Multimedia Services Architecture and Applications, 2008. IMSAA 2008. 2nd International Conference on*. 2008, pp. 1–6. DOI: 10.1109/IMSAA.2008.4753918.

[110] Christian Riede. "Design and Implementation of an IMS Session Management Enabler for Quadruple Play". MA thesis. Technical University of Berlin, 2007.

[111] Christian Riede, Adel Al-Hezmi, and Thomas Magedanz. "Session and media signaling for IPTV via IMS". In: *MOBILWARE '08: Proceedings of the 1st international conference on MOBILe Wireless MiddleWARE, Operating Systems, and Applications*. Innsbruck, Austria: ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), 2008, 1 to 6. ISBN: 978-1-59593-984-5.

[112] Luis Rodriguez Rosello. *Networked Media Research challenges for mastering the media revolution*. Oct. 2007. URL: http://www.nem-initiative.org/public/event/3rdGA/2ndDay_presentations/LRR.pdf.

[113] J. Rosenberg. *A Presence Event Package for the Session Initiation Protocol (SIP)*. RFC 3856 (Proposed Standard). Internet Engineering Task Force. Aug. 2004. URL: http://www.ietf.org/rfc/rfc3856.txt.

[114] J. Rosenberg et al. *SIP: Session Initiation Protocol*. RFC 3261 (Proposed Standard). Updated by RFCs 3265, 3853, 4320, 4916, 5393, 5621. Internet Engineering Task Force. June 2002. URL: http://www.ietf.org/rfc/rfc3261.txt.

[115] P. Saint-Andre. *End-to-End Signing and Object Encryption for the Extensible Messaging and Presence Protocol (XMPP)*. RFC 3923 (Proposed Standard). Internet Engineering Task Force. Oct. 2004. URL: http://www.ietf.org/rfc/rfc3923.txt.

[116] P. Saint-Andre. *Extensible Messaging and Presence Protocol (XMPP): Core*. RFC 3920 (Proposed Standard). Internet Engineering Task Force. Oct. 2004. URL: http://www.ietf.org/rfc/rfc3920.txt.

[117] P. Saint-Andre. *Extensible Messaging and Presence Protocol (XMPP): Instant Messaging and Presence*. RFC 3921 (Proposed Standard). Internet Engineering Task Force. Oct. 2004. URL: http://www.ietf.org/rfc/rfc3921.txt.

[118] P. Saint-Andre. *Mapping the Extensible Messaging and Presence Protocol (XMPP) to Common Presence and Instant Messaging (CPIM)*. RFC 3922 (Proposed Standard). Internet Engineering Task Force. Oct. 2004. URL: http://www.ietf.org/rfc/rfc3922.txt.

[119] H. Schulzrinne, A. Rao, and R. Lanphier. *Real Time Streaming Protocol (RTSP)*. RFC 2326 (Proposed Standard). Internet Engineering Task Force. Apr. 1998. URL: http://www.ietf.org/rfc/rfc2326.txt.

[120] S. Sivasothy, Gyu Myoung Lee, and N. Crespi. "A unified session control protocol for IPTV services". In: *Advanced Communication Technology, 2009. ICACT 2009. 11th International Conference on*. Vol. 02. 2009, pp. 961–965.

[121] L.F.G. Soares, M.F. Moreno, and C. De Salles Soares Neto. "Ginga-NCL: Declarative middleware for multimedia IPTV services". In: *Communications Magazine, IEEE* 48.6 (2010), pp. 74 –81. ISSN: 0163-6804. DOI: 10.1109/MCOM.2010.5473867.

[122] A.J. Stienstra. "Technologies for DVB Services on the Internet". In: *Proceedings of the IEEE* 94.1 (2006), pp. 228–236. ISSN: 0018-9219. DOI: 10.1109/JPROC.2005.861016.

[123] A. Fasbender T. Cagenius. *Evolving the TV experience: Anytime, anywhere, any device*. 2006. URL: http://www.ericsson.com/ericsson/corpinfo/publications/review/2006_03/04.shtml.

[124] Jim Théberge. *Advanced Advertising Standards Proposals*. Sept. 2004. URL: www.tv-anytime.org/ftp/San-Jose_IDE_Sept04/ATSC.ppt.

[125] Interactive Tv Today. *News Source on Interactive Mutiplatform Television*. Jan. 2010. URL: http://www.itvt.com/story/3995/.

[126] Manfred Tscheligi, Marianna Obrist, and Artur Lugmayr, eds. *Changing Television Environments, 6th European Conference, EuroITV 2008, Salzburg, Austria, July 3-4, 2008, Proceedings*. Vol. 5066. Lecture Notes in Computer Science. Springer, 2008. ISBN: 978-3-540-69477-9.

[127] Joe Tullio, Gunnar Harboe, and Noel Massey. "Investigating the Use of Voice and Text Chat in a Social Television System". In: *EuroITV*. Ed. by Manfred Tscheligi, Marianna Obrist, and Artur Lugmayr. Vol. 5066. Lecture Notes in Computer Science. Springer, 2008, pp. 163–167. ISBN: 978-3-540-69477-9.

[128] European Broadcasting Union. "Requirements for the Standardization of Hybrid Broadcast/Broadband (HBB) Television Systems and Services". In: *EBU TECH 3338*. 2010.

[129] International Telecommunication Union. *Worldwide common core - application environment for digital interactive television services. ITU-T J.2001*. 2001.

[130] ETSI Doc. No. ES 201 812 V1.1.1. *Digital video broadcasting (DVB); Multimedia Home Platform*. Tech. rep. 2003.

[131] ETSI Doc. No. TS 102 812 V1.1.1. *Digital video broadcasting (DVB); Multimedia Home Platform (MHP) specification 1.1*. Tech. rep. ETSI Secretariat, 2001.

[132] I. Vaishnavi et al. "From IPTV services to shared experiences: Challenges in architecture design". In: *Multimedia and Expo (ICME), 2010 IEEE International Conference on*. 2010, pp. 1511 –1516. DOI: 10.1109/ICME.2010.5583263.

[133] VideoLAN. *The VideoLAN Project*. Mar. 2010. URL: http://www.videolan.org.

[134] M. Volk et al. "IPTV Systems, Standards and Architectures: Part II - Quality-Assured Provisioning of IPTV Services within the NGN Environment". In: *Communications Magazine, IEEE* 46.5 (May 2008), pp. 118 –126. ISSN: 0163-6804. DOI: 10.1109/MCOM.2008.4511660.

[135] W3C. *W3C Working Draft: XMLHttpRequest*. Nov. 2009. URL: http://www.w3.org/TR/XMLHttpRequest/.

[136] David Waiting et al. "Open source development tools for IMS research". In: *TridentCom '08: Proceedings of the 4th International Conference on Testbeds and research infrastructures for the development of networks & communities*. Innsbruck, Austria: ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), 2008, pp. 1–10. ISBN: 978-963-9799-24-0.

[137] P.R. Wilson, V.G. Oziyani, and N. Ventura. "Monetizing IMS-based IPTV through personalized advertising". In: *Ultra Modern Telecommunications Workshops, 2009. ICUMT '09. International Conference on*. 2009, pp. 1 –7. DOI: 10.1109/ICUMT.2009.5345437.

[138] Jinhong Yang, Hyojin Park, and Jun Kyun Choi. "A study on providing Open IPTV in next generation network service platform". In: *Communications and Information Technology, 2009. ISCIT 2009. 9th International Symposium on*. 2009, pp. 290 –293. DOI: 10.1109/ISCIT.2009.5341242.

[139] Jinhong Yang et al. "Interactive Control Platform for IPTV service in the IMS environment". In: *Advanced Communication Technology, 2009. ICACT 2009. 11th International Conference on*. Vol. 03. 2009, pp. 1703 –1706.

[140] Changwoo Yoon, Hyunwoo Lee, and Won Ryu. "Next generation IPTV platform". In: *Optical Internet (COIN), 2010 9th International Conference on*. 2010, pp. 1 –3. DOI: 10.1109/COIN.2010.5546631.

[141] Benjamin Zachey. "Personalisierung in Next Genration IPTV-Infrastrukturen - Entwurd und Implementierung eines personalisierten Programmführers". MA thesis. Technische Fachhochschule Berlin, 2008.

[142] Theodore Zahariadis et al. *Media Delivery Platforms Cluster: Multimedia Delivery in the Future Internet*. Oct. 2008. URL: www.ist-sea.eu/Dissemination/MDP_WhitePaper.pdf.

[143] Theodore Zahariadis et al. *Towards Future 3D Media Internet*. Oct. 2008. URL: http://www.ist-sea.eu/Dissemination/SEA_FIA.pdf.

[144] M. Zahid, M.A. Qadeer, and A. Iqbal. "Deployment of IPTV over IMS architecture". In: *Internet Multimedia Services Architecture and Applications, 2008. IMSAA 2008. 2nd International Conference on*. 2008, pp. 1 –6. DOI: 10.1109/IMSAA.2008.4753940.

[145] Hongguang Zhang et al. "IPTV 2.0 from Triple Play to social TV". In: *Broadband Multimedia Systems and Broadcasting (BMSB), 2010 IEEE International Symposium on*. 2010, pp. 1 –5. DOI: 10.1109/ISBMSB.2010.5463095.