TUCS

Muhammad Mohsin Saleemi

Towards Combining Interactive Mobile TV and Smart Spaces

Architectures, Tools and Application Development

Turku Centre *for* Computer Science

TUCS Dissertations
No 157, February 2013

Towards Combining Interactive Mobile TV and Smart Spaces

Architectures, Tools and Application Development

Muhammad Mohsin Saleemi

To be presented, with the permission of the Department of Information Technologies at Åbo Akademi University, for public criticism in Auditorium Gamma, the ICT building, on February 15th, 2013, at 12 noon.

Åbo Akademi University
Department of Information Technologies
Joukahaisenkatu 3-5
20520 Turku
Finland

2013

Supervisor

Professor Johan Lilius
Department of Information Technologies
Åbo Akademi University
Joukahaisenkatu 3-5
20520 Turku
Finland

Reviewers

Adjunct professor Jouni Smed
Department of Information Technology
University of Turku
FI-20014 Turku
Finland

Professor Tullio Salmon Cinotti
ARCES - Dipartimento di Informatica
University of Bologna
Via Toffano, 40125 Bologna
Italy

Opponent

Adjunct professor Jouni Smed
Department of Information Technology
University of Turku
FI-20014 Turku
Finland

ISBN 978-952-12-2839-1
ISSN 1239-1883

Dedicated to my beloved father Ghulam Mustafa Saleemi
who gave so much hope, encouragement and support for my education

Abstract

Technological developments in microprocessors and ICT landscape have made a shift to a new era where computing power is embedded in numerous small distributed objects and devices in our everyday lives. These small computing devices are fine-tuned to perform a particular task and are increasingly reaching our society at every level. For example, home appliances such as programmable washing machines, microwave ovens etc., employ several sensors to improve performance and convenience. Similarly, cars have on-board computers that use information from many different sensors to control things such as fuel injectors, spark plug etc., to perform their tasks efficiently. These individual devices make life easy by helping in taking decisions and removing the burden from their users. All these objects and devices obtain some piece of information about the physical environment. Each of these devices is an island with no proper connectivity and information sharing between each other. Sharing of information between these heterogeneous devices could enable a whole new universe of innovative and intelligent applications. The information sharing between the devices is a difficult task due to the heterogeneity and interoperability of devices.

Smart Space vision is to overcome these issues of heterogeneity and interoperability so that the devices can understand each other and utilize services of each other by information sharing. This enables innovative local mashup applications based on shared data between heterogeneous devices. Smart homes are one such example of Smart Spaces which facilitate to bring the health care system to the patient, by intelligent interconnection of resources and their collective behavior, as opposed to bringing the patient into the health system.

In addition, the use of mobile handheld devices has risen at a tremendous rate during the last few years and they have become an essential part of everyday life. Mobile phones offer a wide range of different services to their users including text and multimedia messages, Internet, audio, video, email applications and most recently TV services. The interactive TV provides a variety of applications for the viewers. The combination of interactive TV and the Smart Spaces could give innovative applications that are personalized, context-aware, ubiquitous and intelligent by enabling heterogeneous

systems to collaborate each other by sharing information between them. There are many challenges in designing the frameworks and application development tools for rapid and easy development of these applications. The research work presented in this thesis addresses these issues. The original publications presented in the second part of this thesis propose architectures and methodologies for interactive and context-aware applications, and tools for the development of these applications. We demonstrated the suitability of our ontology-driven application development tools and rule based-approach for the development of dynamic, context-aware ubiquitous iTV applications.

Sammanfattning

Den tekniska utvecklingen inom ICT landskapet och speciellt av mikroprocessorer har möjliggjort en övergång till en ny era där datorkraft är inbäddad i många distribuerade små enheter i vår vardag. Dessa små datorenheter är finjusterade för att utföra en viss uppgift och sträcker sig allt mera mot alla nivåer i vårt samhälle. Till exempel hushållsmaskiner såsom programmerbara tvättmaskiner, mikrovågsugnar osv. använder flera sensorer för att förbättra prestandan och bekvämligheten. Likaså bilar har datoren som använder information från många olika sensorer för att styra bränsleinjektorer, tändstift osv., för att utföra sina uppgifter effektivt. Dessa enskilda enheter gör vardagen enklare genom att hjälpa till med att fatta beslut och därmed lyfta bort bördan från användaren. Alla dessa föremål och apparater samlar in någon information om den fysiska miljön. Var och en av dessa enheter är isolerade utan någon ordentlig anslutbarhet och informationsutbyte mellan varandra. Utbyte av information mellan dessa heterogena enheter kan möjliggöra en mångfald innovativa och intelligenta tillämpningar. Informationsdelning mellan enheterna är en svår uppgift på grund av heterogenitet och interoperabilitet mellan utrustning.

Smart Space vision handlar om övervinna dessa frågor om heterogenitet och interoperabilitet så att enheterna kan förstå varandra och utnyttja tjänster av varandra genom informationsutbyte. Detta möjliggör innovativa lokala mashup applikationer baserade på delade data mellan heterogena enheter. Smarta hem är ett exempel på sådana smarta utrymmen som underlättar att få hälso-och sjukvårdssystemet till patienten genom intelligent sammankoppling av resurser och deras kollektiva beteende, i motsats till att få patienten in i hälso- och sjukvårdssystemet.

Dessutom har användningen av mobila handhållna enheter ökat i rasande takt under de senare åren och de har blivit en viktig del av vardagen. Mobiltelefoner erbjuder ett brett utbud av tjänster åt sina användare, inklusive SMS och MMS, Internet, audio, video, e-postprogram och på senare tid TV-tjänster. Interaktiv TV erbjuder en mängd olika tillämpningar för sina tittare. Kombinationen av interaktiv TV och smarta utrymmen kan ge innovativa tillämpningar som är personliga, kontextmedvetna och intelligenta. Det finns många utmaningar med att utforma ramverk och

utvecklingsverktyg för snabb och enkel utveckling av sådana tillämpningar. Forskningen som presenteras i denna avhandling behandlar dessa frågor. De ursprungliga publikationerna som presenteras i den andra delen av denna avhandling föreslår arkitekturer och metoder för interaktiv och kontextmedvetna applikationer och verktyg för utveckling av dessa tillämpningar. Vi visade lämplighet våra ontologi-drivna verktyg för applikationsutveckling och regelbaserad-strategi för utveckling av dynamiska, kontextmedvetna ubiquitous iTV applikationer.

Acknowledgements

Allah Alone is the creator of all knowledge and wisdom, and it is He who has bestowed the gift of understanding to human. I humbly offer my thanks to Almighty Allah, the most merciful and dispenser of grace, who gave me an opportunity to complete this PhD dissertation.

This thesis would not have completed without the support and encouragement of several people and it is my pleasure to appreciate all those who made this thesis a reality.

First of all, I am beholden to my supervisor Professor Johan Lilius whose valuable advices and devoted guidance in an affectionate and friendly atmosphere mainly contributed to the successful completion of the thesis. I truly appreciate his efforts for keeping me on right track and supporting me during tough times.

I would also like to thank the reviewers of my thesis, Adjunct Professor Jouni Smed from University of Turku, Finland, and Professor Tullio Salmon Cinotti from University of Bologna, Italy, for their time and valuable comments and useful suggestions to improve the quality of this thesis. I am very grateful to Adjunct Professor Jouni Smed for accepting to be the opponent in the public defense of my thesis.

I owe my gratitude to all my co-authors Kristian Nybom, Janne Kempe, André Kaustel, Thomas Rosqvist, Juuso Jokiniemi, Natalia Diaz, Jerker Björkqvist, Ivan Porres, and Johan Lilius, for the work we have accomplished together. Special thanks go to Kristian Nybom for his time and efforts to act as a proof reader to improve my research papers, and to Johan Ersfolk for providing me the translation of the thesis abstract in Swedish.

I wish to acknowledge the ideal and productive research environment provided by the Embedded Systems Laboratory at Department of Information Technologies and Turku Centre for Computer Science (TUCS). I am grateful to both of these for providing funding for my research. I would also like to thank Nokia Foundation and HPY Research Foundation for their generous grants.

I would like to express my gratitude to all my colleagues and friends at the Embedded Systems Laboratory especially Johan Ersfolk, Juuso Jokiniemi, Qaisar Malik, Kristian Nybom, Sébastien Lafond, Andreas Dahlin, Fareed

Jokhio and Natalia Diaz. I would extend my gratitude to all other friends and colleagues sitting in ICT building especially Moazzam Fareed, Ali Hanzala, Adnan Ashraf, Irum Rauf, Kashif Javed, Dragos Truscan, Espen and Nauman Ahmad Khan.

I am very grateful to all the secretaries, administrative and IT support staff at Åbo Akademi University and TUCS for their assistance throughout my doctoral studies. Special thanks to Christel Engblom, Britt-Marie Villstrand, Irmel Laine, Tiina Hanila, Tove Österroos and Christel Donner.

My infinite thanks go to my family for their constant support, prayers and love. I am truly indebted to my parents, whose reinforcement and encouragement has always been with me in every day and night, whose back up kept me aspiring in every darkest shade and brightest shine, and whose untiring efforts brought me up to here.

I would also like to thank my great friends in Turku for their company, good times and all the Halla Gulla for making my stay in Finland so wonderful.

Last but not least, I would like to acknowledge my wife Sumreen Mohsin and daughter Alizay for their understanding, unyielding devotion, support, patience and unwavering love to lighten up my spirit to complete this thesis.

List of Original Publications

I M. Mohsin Saleemi, Jerker Björkqvist and Johan Lilius. System Architecture and Interactivity Model for Mobile TV Applications. In *Proceedings of the 3rd ACM International Conference on Digital Interactive Media in Entertainment and Arts (DIMEA 2008).* ACM, pp.407-414, September 2008, Athens, Greece.

II M. Mohsin Saleemi and Johan Lilius. Interactive Applications for Mobile TV. In *Proceedings of International Conference on Multimedia Computing and Information Technology (MCIT 2010).* IEEE Xplore, pp.37-40, March, 2010, Sharjah, UAE.

III M. Mohsin Saleemi, Jerker Björkqvist and Johan Lilius. Content Scheduling in Multimedia Interactive Mobile Games. In *Loading...: The Journal of the Canadian Game Studies Association*, 3, December 2009, Toronto, Canada.

IV Andre Kaustell, M. Mohsin Saleemi, Thomas Rosqvist, Juuso Jokiniemi, Johan Lilius and Ivan Porres. Framework for Smart Space Application Development. In *International Workshop on Semantic Interoperability (IWSI 2011).* SciTePress, pp. 3-12, January 2011, Rome, Italy.

V M. Mohsin Saleemi and Johan Lilius. End-users Service Composition in Ubiquitous Computing using Smart Space Approach. In *Proceedings of the 6th International Conference on Internet and Web Applications and Services (ICIW 2011).* IARIA, pp. 214-221, March 2011, St. Maarten, The Netherlands Antilles.

VI M. Mohsin Saleemi, Natalia Diaz Rodriguez, Johan Lilius and Ivan Porres. A Framework for Context Aware Applications for Smart Spaces. In *Proceedings of the 4th conference on Smart Spaces (ruSMART 2011).* Springer-Verlag, pp. 14-25, August 2011, St.Petersburg, Russia.

VII M. Mohsin Saleemi, Natalia Diaz Rodriguez and Johan Lilius. Towards Ontology-driven Development of Ubiquitous and Intelligent interactive TV Applications. In *Proceedings of the 10th IEEE Interna-*

tional Conference on Pervasive, Intelligence and Computing (PiCom 2012). IEEE, December 2012, ChangZhou, China.

"Seek knowledge from the cradle to the grave."
Prophet Muhammad (Peace be upon him)

Contents

Part I

Research Summary

1 Introduction

Recent advances in ICT technologies have made available a considerably interactive information-rich environment with devices ranging from personal portable such as mobile phones, Personal Digital Assistant (PDA) etc., to domestic accessories such as TV and computers. People are able to interact with the environment using these devices. Mobile Phones have become an essential part of everyday life and, according to a study, there were more than five billion users of mobile phones in July 2010 [97]. The technological advances in the digital world enable a wide range of mobile devices to access a variety of content through a number of different networks. Mobile phones offer a wide range of different services to their users including text and multimedia messages, Internet, audio, video, email applications and most recently TV services. Moreover, the increase in the amount of data available on social and semantic Webs, and approaches to access and utilize that data, enable the development of innovative applications and services that link TV viewing with both the viewer's personal profiles on social networks, and with social activities across the Web [79]. This convergence results in the social and next generation ubiquitous iTV applications that are innovative, ambient, more personal and social. These ubiquitous applications employ a range of different devices in addition to mobile phones to provide a set of services that are personalized and context-aware to automatically adapt to the user's preferences. This requires more advanced methods for data handling and understating and approaches for data exchange and communication across heterogeneous sources. Semantic technologies can solve the essential data understating and interoperability issues across heterogeneous sources of data [22].

While the Semantic Web envisions better structured data enabling the new possibilities for the Internet, the semantic concept is also being adopted in other areas. One of these areas is the Smart Space, which, although very similar, comes with a different set of restrictions, challenges and possibilities. The reason is that Smart Spaces heavily depend on heterogeneous devices, systems and services and need to be made seamlessly interoperable to be used effectively in a device, vendor and domain independent manner.

We define *Smart Space* as an abstraction of space that encapsulates both the information in a physical space as well as access to this information, such that it allows devices to join and leave the space. In this way, a Smart Space becomes a dynamic environment whose identity changes over time when the set of entities interact with it to share information between them. This opens new cutting-edge aptitude for data owners, service providers, network operators, and device manufacturers to fill the gap by developing and providing novel added-value services and applications based on data exchange. The effective design and implementation of emerging Smart Space applications still have challenges and lack tools and development frameworks. The research work presented in this thesis addresses these issues.

Today, many broadcast TV shows are becoming interactive by enabling their viewers to participate using remote controls and mobile phones. The services and applications in the interactive TV environment are mainly limited to the TV domain only, which is rather passive as compared to highly dynamic web and Smart Space environments, which endorse active user participation. Therefore, one of the most important and evolving research areas in the ICT landscape is how to converge separate domains, networks, devices, services and applications to offer innovative services that are highly interactive, intelligent and give ultimate user experience.

The scenarios presented here are the examples of convergence of interactive TV and Smart Spaces that exploit additional interaction to offer innovative services and applications.

Personalized TV Recommendations: The success of social portals such as Facebook, Twitter, YouTube etc. indicates that an increasing number of people are using them and share their personal data and interest with other people. The Smart Spaces together with semantic web technologies, such as RDF/OWL, could be used to model the data and their relationships, could provide RDF storage, inference and querying of that data and give personalized recommendations based on a user's current context. While watching a program on TV, users might be interested to know further details about the program and interact using their mobile phones to obtain the required information. Furthermore, intelligent recording of a user's favorite TV program could be done according to his profile information or fan page on Facebook and the TV guide available on the broadcaster's web page.

Staged Participatory Multimedia Events (SPME): SPME are broadcasted television events that actively involve TV viewers to participate in interactive TV applications that are not commercially available [99]. The events are directed by TV viewers and any number of viewers can participate. One example of this kind of interactive application is AuctionTV application that allows participants to offer an item for sale. Smart Spaces could be ideal choice for this kind of application as the application data could be structured using RDF/OWL and can be driven by viewer actions. Moreover, participants could use any kind of device to interact with other participants through the Smart Space. The Smart Space could track the participants' sessions, subscriptions and handle the data as the application evolves.

1.1 Research problems

In this section, a list of research problems that motivated this thesis, are outlined.

- *What are the limitations of existing digital TV standards to support ubiquitous interactive applications?*

- *Which infrastructure is needed for interactive TV applications and what are the requirements for building it?*

- *What are design issues and problems related to development of a context-aware interactive application using broadcast technology and point-to-point network?*

- *How is it possible to achieve interoperability in Smart Space applications and what kind of methodologies and tools are needed to develop Smart Space based multi-domain scenarios?*

- *How is it possible to provide automatic service composition in the Smart Space for more complex scenarios?*

- *How is it possible to model and process context information in the Smart Spaces using ontologies and the inference rules?*

- *How is it possible to design and develop interactive TV applications based on the semantic and social Web data, iTV content and the Smart Spaces?*

This thesis addresses these research problems and proposes suitable solutions for them in the original publications in Part II of this thesis.

1.2 Research Methodology

The methodology used in this research work is mainly constructive research methodology. We first identified the research problems in this area and then constructed the solutions for them. In the next step, the proof-of-concept validation of the proposed solutions is given to check if the solutions work. The scope of applicability is then examined and the results are submitted to the international scientific conferences to demonstrate the research contributions. We started by analyzing the existing standards and technologies available for the interactive TV and identified their limitations to support innovative interactive applications. The results of this investigation along with the proposal for new software architecture are presented in Papers I and II. The implementation of a specific interactive application is carried out which is demonstrated in Paper III. A tool was developed for ontology-driven application development and a case study was used to analyze the feasibility of that tool as presented in Paper IV. The notion of services is introduced and implementation of a service composition algorithm was introduced in Paper V. A proposal for context aware intelligent applications was introduced in Paper VI and the developed tool presented in IV was used. We demonstrate the suitability of our ontology-driven application development tools and rule based-approach for the development of highly dynamic context-aware iTV applications in Paper VII.

The proof-of-concept demonstrations in our publications involved various software tools and languages. The most important were the following: the Smart-M3 platform [11], the Ptolemy [81] tool, the Protégé ontology editor [72], UML [73], Eclipse [41].

1.3 Research Setting

The research presented in thesis has been done as a collaborative work at the Turku Centre for Computer Science (TUCS) and Åbo Akademi University. The research work has both individual and teamwork context. The individual research work was done in the area of interactive mobile TV. The other part of the research work was in the context of two research and development projects, namely, ABOT (A location-based interactive game) and DIEM (Devices and Interoperability Ecosystems) [38]. The main objective of the ABOT project was to implement location-based interactive multimedia services using the broadcast channel combined with a point-to-point channel. The DIEM project aimed to create the concepts and implementation of Smart Space interoperability solution and platforms, which are adaptable to multi-domain and multi-vendor applications. DIEM project, funded by Tekes [42], is a consortium of 32 partners both from industry and academia.

1.4 Thesis Organization

This thesis consists of two parts. Part I is organized as follows: Chapter 1 describes the introduction and motivation of this research work, the research problems, the research setting and the research methodology. Chapter 2 presents the theory related to the interactive TV, the existing software architectures and the description of current interactive services along with a look towards future interactive applications. In Chapter 3, the overview of Smart Space approach is presented along with the requirements, design issues and the application development for the Smart Spaces. A description about the future interoperable applications using Smart-M3 is also presented in this chapter. In Chapter 4, we explore the potential of introduction of Smart Spaces in the design and development of interactive TV applications. Chapter 5 presents the major contributions of this thesis, gives an overview of the published papers along with the related work in this scientific area. The discussion and conclusion are presented in chapter 6. The original research papers are included in Part II of this thesis.

2 Interactive TV

In recent years, migration from analog to digital broadcasting has made significant and rapid progress in the sector of media consumption. In addition to other benefits, this migration has also resulted in giving birth to a new field of interactive television (iTV). It enables more active participation of the users in the broadcast content, hence, challenging the traditional media services. This active user participation originates the emergence of new standards, content, delivery platforms, services and applications.

The use of mobile handheld terminals has increased at a tremendous rate in recent years. The digitalization of television and content brings not only the distribution of high quality services but also the technology that has the capability to deliver content to the mobile terminals. It has provided ways to deliver a variety of content such as software applications, web pages, graphics and games other than just audio and video. Mobile handheld terminals are indeed intended to support digital transmission of multimedia content and it is considered as the next development step in the broadcasting technology. Wireless network operators already enable mobile phone users to download and play streaming video. Broadcasting digital videos over the existing telecommunication networks causes a number of technical problems which include high cost of distribution of digital content. This is because the cellular networks are point-to-point networks and low capacity of existing third generation mobile communication networks (3G). Even a moderate use of video streaming services can cause the 3G networks to reach their capacity limit [76].

2.1 Literature Review

The research in the area of interactive digital TV has been focused on a variety of digital content and interactive applications appropriate for digital TV [53]. These research works focused on ITV issues for wide range of applications. These include usability issues [24], human-computer interaction [30], graphics user interface [34], system architecture [85]. The different application domain that are focused include learning [13], entertainment [31][101], personalization [17], TV channels and adds [35], shopping [98], messaging and video streams [67]. These research works provide an overview of the emanating kinds of ITV contents and applications for digital TV. The new and exciting area of Interactive Mobile TV (iMTV) is quite immature and needs a considerable amount of work because of the absence of common platform for mobile interactive services. There is no extensive research on the system architecture of mobile TV, the impact of interactive applications on this architecture and the requirements imposed by truly interactive applications on terminal and server. Part of this thesis is focused on these issues.

In recent years, there have been coordinated efforts from multiple organizations and research groups towards inclusion of semantics in interactive TV services and platforms. Work presented in [77] describes a semantics-aware platform for interactive TV services in order to distribute, process and consume media content. They proposed an interactive TV receiver framework capable of collecting, extending and reasoning semantic data related to the broadcast multimedia content. The work presented in [25] outlines a video annotation technique, ontology-based modeling, multimedia metadata and user profiling through semantic reasoning. The main goal of this work is to create a personalized digital TV recommender based on metadata. Other works in the direction of personalized TV recommendation systems based on semantics include [50] and [23]. All of these approaches use semantic information for the reasoning purpose to deliver personalized TV content.

2.2 Multimedia Broadcast Standards

Digital broadcasting systems provide several mechanisms for transmitting data services to improve the conventional radio and TV services, and these systems also have the capability to support new multimedia services. There are several competing technologies for the broadcasting of digital content. In some parts of the world, we can tune into digital broadcasting on our digital TV set in the living room. The standards that provide this facility include Integrated Services Digital Broadcasting (ISDB), Digital Video Broadcasting (DVB), and Digital Multimedia Broadcast (DMB). These digital broadcast standards are being used in different parts of the world. In Japan, the ISDB standard is used to provide multimedia broadcast services to the terrestrial networks. The DMB standard is based on the Digital Audio Broadcast (DAB) standard and is deployed in South Korea. It allows mobile reception of the multimedia content. The DVB standard is adopted in Europe for digital video broadcasting, and the European Telecommunications Standards Institute (ETSI) has regulated it.
Based on these standards, there are several systems that can provide the broadcasting of digital contents. The most common systems include ISDB-T, DVB-T, DAB, DMB and DVB-H.

ISDB-T

ISDB-T [52] is an emerging digital TV broadcasting system developed in Japan to provide flexibility for multimedia broadcasting services using the terrestrial networks [70]. The ISDB-T system uses the Band Segmented Transmission technique (BST), which divides the channel into a set of frequency blocks called segments. The ISDB-T systems are capable of providing a variety of multimedia services to the stationary as well as the mobile receivers.

The main drawback of ISDB-T for mobile reception is that the digital TV transmission to the mobile terminals requires high power consumption.

DVB-T

DVB-T [1] is a flexible terrestrial system that supports various broadcast service environments ranging from a fixed-rooftop antenna to a portable service. DVB-T also supports mobile reception of the services and content. This standard has been accepted as a common standard for the digital television in Europe and it is the basis of the new emerging standard for reception of the digital television services on the handheld terminals i.e., the DVB-H standard [2]. DVB-T uses the MPEG-2 standard for the compression because it makes it possible to multiplex the separate elementary streams that are associated with the current service and it allows the encryption of all transport streams.

DVB-T has the same drawback as ISDB-T; it was not originally designed to target the mobile receivers, and the major problem in receiving the digital TV on handheld devices is the battery consumption. Neither of these systems takes into account the special requirements to support digital TV services on the mobile handheld terminals.

DMB

Digital Multimedia Broadcasting (DMB) [32] is a digital transmission system for the delivery of multimedia content and services to mobile phones. It has been developed in South Korea and is the further development of the DAB standard, with additional error correction capability to deliver television services to the mobile handheld receivers even at high speeds. For the error correction, the DMB uses the technique called Forward Error Correction as does DVB-H, but the DMB additionally uses the time interleaving to solve the problems of difficult conditions such as impulsive noise, typically found in the mobile environments. The time interleaving works by spreading the errors in time over logical frames so that the receiver can correct the errors, hence, it is very effective. The DMB offers the possibility to work using considerably less power.

All of these features make DMB a better choice to be used for delivering digital TV services to the mobile handheld terminals. The only problem with the DMB system is its limited capacity. Scaling up to a large amount of multimedia services on top of DAB/DMB would require frequency reallocation. For high-capacity demand, the DVB-H system would normally be the system of choice [32].

DVB-H

DVB-H (DVB-Handheld) [47] is an extension of the DVB-T, which is the current terrestrial digital TV broadcast standard. The DVB-H enables service reception in the handheld devices using IP data. DVB-H network parameters are optimized for mobile usage, and for the handheld devices that have low battery capacity. The DVB-T was particularly designed to deliver broadcast services to the living room TV connected to a continuous power supply with a large rooftop antenna. In contrast, the DVB-H was developed by taking into account the mobile handheld receivers and is optimized for the battery-powered receivers that have internal antennas and small screens. Although, the DVB-T has proved itself to be able to work in a mobile environment e.g., public transportation, the DVB-H overcomes two key limitations of the DVB-T when used for the handheld terminals. Firstly, it lowers the battery power consumption. Secondly, it improves robustness in very hard reception environments both indoor and outdoor.

DVB-H reduces power consumption by using a technique called time slicing. The data is transmitted in high-speed bursts and the mobile receiver only wakes up when it has to receive these bursts. The receiver remains in the sleep mode between these bursts, i.e., at all other times. This technique enables the receiver to save power up to 90 percent. DVB-H has the ability to converge with the GSM standard when using the IPDC technology.

Mobility and flexibility are obvious benefits of the mobile TV. DVB-H has full potential and capability to deliver large amount of multimedia and digital TV services to the mobile handheld terminals with high data transmission rates. Because of the related features and similarities among basic building blocks in DMB and DVB-H, it is unlikely that two handsets, each having a different mobile TV standard, will end up being vitally different in the reception quality or the power consumption.

In recent years, the DVB Project have developed new standards such as DVB-T2, DVB-SH and DVB 3D-TV [1]. A complete list of standards and the countries in which these are deployed can also be seen on the DVB project webpage.

2.2.1 Mobile TV: Broadcasting to Handheld Devices

Today's handheld devices such as mobile phones offer a wide range of different services to the users including text and multimedia messages, Internet, audio, video and email applications. These devices have become advanced entertainment and utility devices which can function as a radio, audio player and video player in addition to mobile phones. It seems that the TV is the only missing application from the mobile phones. Mobile TV is the broadcast transmission of TV programs or content in a coverage area for a wide range of wireless devices ranging from capable cellphones to PDAs and mobile receivers.

Mobile TV implementations are now more generic, based on Internet Protocol Datacasting, IPDC. IPDC is an effective means that is used in digital broadcasting technology to deliver the data and services to mobile users in a cost effective and efficient way. IPDC makes it possible to deliver data services to a large number of users simultaneously in point-to-multipoint fashion. This distribution of data and services are unidirectional and a separate interaction channel is needed for interactivity purpose. By using IP based technology, it is possible to deliver any type of digital content such as audio, video, graphics, web pages, games, and live TV to the end users. This means that IPDC expands the range of services and contents that can be delivered and provide the ways to receive these services and contents by the mobile users. Every type of digital content is encapsulated into IP packets, which is the format used to deliver digital contents on the Internet, and then broadcast to the users via digital broadcast networks such as DVB-H and DAB etc. Hence, all of the contents that can be delivered through the Internet can also be provided by IPDC. Furthermore, by using the interactive channels through other networks, interactivity is made possible so that the customers can also request the contents [12]. The main features of IPDC include the following:

- Use of digital broadcast networks for a large number of different services
- For interactivity purpose, mobile networks are used
- IP is used as abstraction layer to deliver any kind of digital content
- It enables users to access a wide range of multimedia and TV services
- It provides means to enable users to discover available services by ESG
- Uses of high bandwidth channels and enables high transmission rates

IPDC over DVB-H technology offers the advantage that all the existing IP-based digital content such as video streams, web pages, music files or game software, can be easily delivered to the mobile devices over mobile broadcast. The combination of DVB-H and IPDC enables the mobile users to access and consume music, web pages, games and the rich media, television and radio.

2.3 Software Platforms

The software platform is the key element for the vitality of digital systems as it provides the facility and environment for the development and execution of software applications that perform the valuable functions to make these systems work. To receive the broadcast services and mobile TV on a wide range of handheld devices, there must be a common software platform that provides the capability for these services to function across multiple networks. The software platforms of modern digital broadcast systems are responsible for providing efficient and high quality integration of software applications and the hardware platform that is used in the end user terminals to give effective management of multimedia services, data, and hardware resources. The software platform enables early and reliable delivery and handling of advanced video broadcasts and interactive services and applications. By using the software platform that is based on the open standards and specifications, the broadcast and multimedia services and applications can achieve full interoperability and could be independent of the specific hardware of user terminals. This means that the applications and services have the same constant behavior on a wide range of hardware environments in different kinds of terminals, which use that software platform.

This section briefly describes the common existing software platforms of digital TV and mobile phones, which are being used for the software application and service developments.

MHP

MHP (Multimedia Home Platform) is an open software platform that defines a generic interface between interactive digital applications and user terminals on which these applications execute. In this way, applications from various service providers are interoperable with different MHP implementations independent of the underlying hardware, vendor and software. MHP specifies a way to deliver interactive applications within digital TV stream and describes how these applications are executed on the digital TV receiver. The application model and lifecycle management functionality in MHP is provided by JavaTV API. The MHP standard specifies a software environment that must be offered by the receiver in order to be compatible with MHP. DVB produced the GEM (Globally Executable MHP) specification which is a cut-down version of MHP that removes any dependencies on other DVB technologies.

OCAP

OCAP (OpenCable Application Platform) is a software platform for data broadcasting defined by CableLabs and is largely based on MHP with cable specific extensions. OCAP specifies the application model and signaling information for broadcast applications. It extends MHP by supporting an additional monitor application to give network operators more control over the receiver. OCAP also provides support for the applications which can run at any time and are not related to the currently watched TV channel e.g., video-on-demand (VoD) applications.

DASE

DASE (DTV Application Software Environment) is the middleware platform based on the ATSC (Advanced Television Systems Committee) digital broadcast standard. It provides core functionalities required in digital TV including application model, life cycle signaling, data storage and service management. [62] describes the systems architecture and software environment for DASE systems.

ACAP

ACAP (Advanced Common Application Platform) is a new standard that is designed by integrating OCAP and ATSC-DASE. The goal of ACAP is to define a common application framework between Cable and terrestrial digital TV. It combines the Java part from OCAP and the HTML part from DASE. It uses the ATSC service information for handling broadcast services. ACAP includes elements from the full MHP specification and GEM, including GEM application signaling.

BML

BML (Broadcast Mark-up Language) is an XML based multimedia coding scheme defined by the ARIB STD-B24 standard. It provides multimedia data broadcasting to enable viewers to access multimedia content and to obtain detailed information about the TV programs. Similar to other digital TV middleware standards, the BML standard is also based on GEM to provide application model and life cycle management functionalities. The main difference between BML and the other standards is its transport protocol to carry applications and data. It uses the DSM-CC data-carousel based transport protocol instead of an object carousel.

MIDP

MIDP (Mobile Information Device Profile) is designed for mobile phones and entry-level PDAs. It offers core application functionality required by mobile applications including user interface, network connectivity, data storage and application management. MIDP provides a complete environment for creating applications for handheld devices while having minimum usage of memory and power consumption. The key feature of high-level MIDP applications is their portability across various cell phones by handling issues such as screen layout and button mapping. The MIDP solution is based on connection limited device configuration (CLDC) which relies on K virtual machine, a derivative of JVM for resource constrained mobile devices.

MIDP was not initially designed to handle the broadcast services and applications because it was targeted to be a common software solution for the resource-constrained mobile handheld devices. An optional package named Mobile broadcast service API for Handheld Terminals[8] in MIDP environment provides the functionality to handle broadcast content, e.g. view digital television and to consume rich multimedia services and features. The main objective of this API is to enable java applications to receive, execute and interact with the received content over digital broadcast link, hence the receiving and management of applications that are delivered through broadcast channel. It also provides the management of interactive broadcast services including the service search and discovery, service and content access and consumption, service scheduling, protection and purchasing. This API adds exciting new features to the existing MIDP software platform to support digital broadcast services through the consumer mobile handheld terminals that was not possible earlier because there was no framework addressing the specific needs of digital broadcast services like digital TV and other rich multimedia services.

Comparison

The table 1 represents the main elements that are included in the above mentioned platforms to give a precise comparison.

Table 1: Comparison between existing solutions

Features	**MHP**	**OCAP/ ACAP/ DASE**	**BML**	**MIDP+ JSR-272**
Underlying Platform	JavaTV/ GEM based	GEM-based	GEM-based	Java-Based
Specification	Receiver/ transmission	Receiver	Receiver	Receiver
GUI	AWT/ HAVi	AWT/ HAVi	AWT/ HAVi	LCDUI
Application model	DVB-J	DVB-J/ ACAP-J	ECMA-Script	MIDlet
Service selection	JavaTV API	JavaTV API	JavaTV API	JSR 272
Service information	DVB-SI API	JavaTV-SI API	JavaTV-SI API	JSR 272
Interactivity	Interactive profile	Interactive profile	Interactive profile	Limited
Signaling and Data Access	Object carousel	Object carousel	Data carousel	Not specified
Monitor application	No	Yes	No	No
Unbounded app.	No	Yes	Yes	Yes

2.4 Interactivity in Mobile TV

The potential to deliver live TV on a mobile is in itself a great achievement, but is now becoming overshadowed by the fact that the mobile users want greater control over the content rather than simply watching. Hence, interactive TV and iTV applications are recent research topics and currently we have no commercial deployments of truly interactive TV applications. Due to the limited hardware and software resources, it is more likely that the people will use interactive services such as voting, chatting, shopping, breaking news and games and interactivity related to TV programs, etc. rather than watching long movies on their mobile phones [89]. This requires new tools and methodologies that can cope with the complexity of dynamic iTV applications.

The term 'interactivity' has been defined in many different ways. A literature review presented by [57] pinpoints the definitions prepared by the researchers from different professional and academic perspectives. The definition that is most appropriate in the context of mobile interactive TV is three dimensional [51]. It is described as the experience involving 1) at least two actants (human or technology) participating in the exchange, 2) a technology is presented that facilitates the exchange and 3) the users have the possibility to modify the mediated environment, e.g., through making inputs and creating texts. From this perspective, we define interactive applications as software applications that run on the handheld terminals or media servers and require the user's involvement in the presence of some technology to affect the environment. These interactive TV applications are not limited to a single distribution method but involve different domains like 3G/UMTS, GPRS, broadband and WLAN, etc. Interactivity is not about simply having a return path for users to respond to. Rather it goes deep into the design of interactive applications. Hence, interactive applications are the sought-after features, along with capability to deliver TV.

2.5 Examples of value-added Services and Applications

The interactive applications that could possibly run over the Mobile TV can be broadly categorized in the following categories [88].

Entertainment: This class of interactive mobile applications is perhaps the most attractive for the viewers as it will provide them truly absorbing services they like to see on their mobile devices. This class would include interactive TV game and quiz shows that will allow real-time interaction with the live broadcast show. Other applications in this category would include Video on Demand (VoD), user generated content, enriched sports applications where the viewers can obtain additional content about the ongoing sport events [44][92].

Learning and Information: In a broader view, most of the interactive TV services could be used as learning and information resources. For example, in addition to the typical interactive information services such as news, weather, Electronic Service Guide (ESG), information of local theaters, hotels and bars etc., the learning services could include the audio visual courses a viewer is interested in. The important point here is that these applications should have minimal text and should be based on audio/video content because of the restricted set of features of the mobile handheld devices. These learning and information services can be provided as stand alone applications or they can be mixed with the entertainment category to give an absorbing experience of learning using entertainment. An example could be learning the commonly used words, their expressions and the delivery emphasis [69][86].

Advertisement and enterprise: Creative and advanced advertisement and enterprise applications will provide promising benefits to all the key players including cellular and broadcast network operators, and are an excellent source of revenue. These innovative applications may include the personalized interactive advertisements in which, based on the preferences, the viewers are given the opportunity to shop by making purchase requests using their mobile phones. They can also obtain additional information about their requests, and would also be able to receive related products or digital content. The enterprise applications are the business applications which would enable the users to handle their businesses on-the- move directly from their mobile phones [15][14].

Table 2: Application Catagories

Catagory	**Example Interactive Applications**
Entertainment	Games, broadcast shows, sports, VoD, user generated content portals
Information	news, weather, ESG, local information services
Learning	Audio/video courses, learning language and expressions
Advertisement and enterprise	Personalized classifieds, shopping, stock exchange, banking
Communication and voting	Chatting, community-based portals, voting, auction, gambling, bets

Communication and Voting: As the primarily usage of the mobile phones is for personal communications, the ultimate return channel provided by the mobile phones allows the development of interactive communication and voting applications over the Mobile TV. These applications include, for instance, chatting with friends, giving views regarding some TV programs, casting votes in a poll and the applications based on the instant messages from a group of users e.g., auction, gambling and betting applications. All of these applications would be possible over the Mobile TV using the messaging facility provided by the cellular network operators [16].

2.6 A particular Implementation of Multimedia Interactive Application

Given the background of interactive TV applications and the standards in the previous sections, we now discuss issues that arise when we implement a particular interactive multimedia application. We discuss how to implement interactive multimedia services using a DVB-H broadcast channel combined with a point-to-point channel, such as 3G or GPRS. We study the problem in the context of a location-based interactive mobile game. With the evolution of handheld devices, positioning techniques and new transmission technologies, a new class of location-based mobile applications has been realized [40]. These mobile location based games involve wireless networks, navigation techniques, movement and location of the players, and the hypermedia content used by the players while playing the game. Examples of some particular location-based games include [26] and [58]. These games are realized with the wireless handheld devices together with the GPS receivers in a PtP communication network over the Web infrastructure. In [71], the authors investigate the basic properties of the location-based games.

In our interactive location-based mobile game, we use a new wireless broadcast technology (DVB-H) with the old theme of traditional location-based games. The motivation for using DVB-H is that the substantially larger bandwidth will provide a more immersive environment for the players. The technical challenge is to schedule the delivery of data over the broadcast channel while maintaining the Quality-of-Service, i.e., sending the right data to the right user at the right time to provide a seamless interactive experience. These kinds of applications are sensitive to the delay as the players would need to wait unnecessarily for the content to identify their next tasks which may in turn be the difference between winning and losing. We explore design issues and problems related to the scheduling of content in the game and propose a content scheduling algorithm to solve these problems.

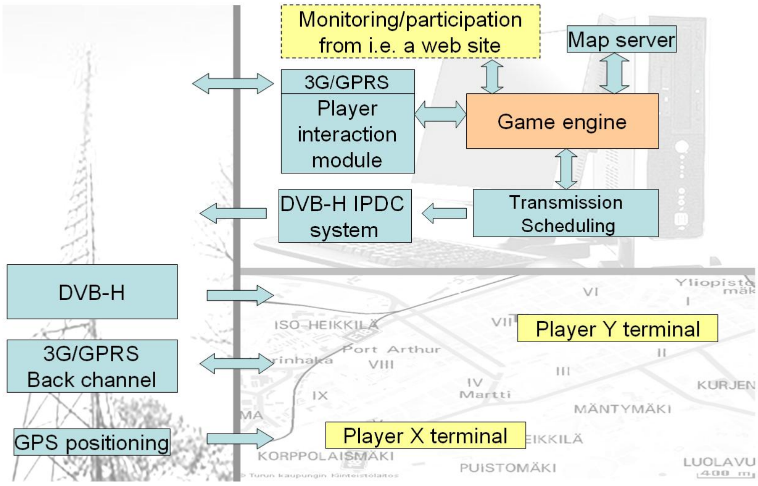

Figure 1: Game platform

2.6.1 Interactivity

In this application, an interactive service is defined as a service where the action of one user affects the internal status of the server providing the service, allowing all other users to experience the change. We can also define interaction from the point of view of the user. In this case we can distinguish between passive and active interaction. The passive interaction is used for interacting with the server periodically, i.e., sending the state information with a fixed interval. The active interaction, on the other hand, is initiated when the user has to make a decision that affects the state of the game and the other users. An active interaction can, for instance, be when a player requests some information from the server to perform an action or when a player makes a choice to accept or reject some action that has influence on other players. In this interactive game we are dealing with both active and passive interactions.

2.6.2 Game model

Åbot is a multiplayer interactive mobile game in which movements and locations of the players trigger all the actions. The GPS is used for obtaining the location information. The communication between the terminals and the game server is done using two different channels. The GPRS/3G is used as the uplink to the server to send information from the terminals to the server using an IP connection whereas a wireless broadcast medium (DVB-H) is used to communicate with the players. The real physical environment serves as playing area for the game. The conceptual game platform is shown in

Figure 1. The game provides bridging of physical locations and objects to a virtual map on players' handheld screens. The game has two different kinds of tasks for the players. The players can collect bots, which are virtual containers that are dispersed on the game map by the server. Bots contain hints, points, equipment etc., which are useful for solving the quests. Quests are the second kind of tasks in the game. Quests in their simplest form involve finding locations by solving puzzles, and they can also be constrained in time. Quests also form the story of the game. The players in the game are equipped with handheld terminals having mobile phone networks and capable of receiving wireless broadcast data. Additionally, the players are equipped with a GPS device to get accurate positions on their mobile phones.

The novelty of the game is that we use DVB-H broadcast medium to deliver data to the players. Our motivations for using DVB-H are the following. Firstly, DVB-H is optimized for the handheld terminals and offers much higher data rates than the cellular networks such as 3G or GPRS. Secondly, DVB-H is a relatively new technology. We want to use the context of a location based game to explore this technology in a converged environment where both cellular and broadcast technologies are used.

2.6.3 Server architecture

Figure 2 shows the logical block diagram of the server implementation. The game engine is the entity which keeps track of the state of the game. Its main tasks are to update the positions of the subscribers and based on that to decide on what new information each subscriber (or the group of subscribers) needs next. The media engine is the entity which handles all the objects, in the media database, associated with the service. The media engine produces all messages which contain service objects. The game engine may compose dynamically directed live video and audio feeds reflecting, for instance, the state of the game. The system encourages the subscribers to produce own content to the service by uploading digital images or video clips. The subscriber manager keeps track of users or players that are currently using the service interactively. The subscriber manager assigns correct IP addresses to the packets encapsulating data destined to a certain subscriber or group. In addition, the subscriber manager handles congestion control for individual terminals. During login, the memory capacity of each terminal is negotiated to the server and taken into account before transmission, in order to avoid buffer overflow in the receiving terminals. The message handling in the communication subsystem is realized using message queues for input and output messages, a message carousel, and a message router. Messages may be of any length and have arbitrary deadline for delivery. This imposes the need for a pre-emption mechanism for message transmission. In other words, an urgent message must be able to pre-empt a less important message.

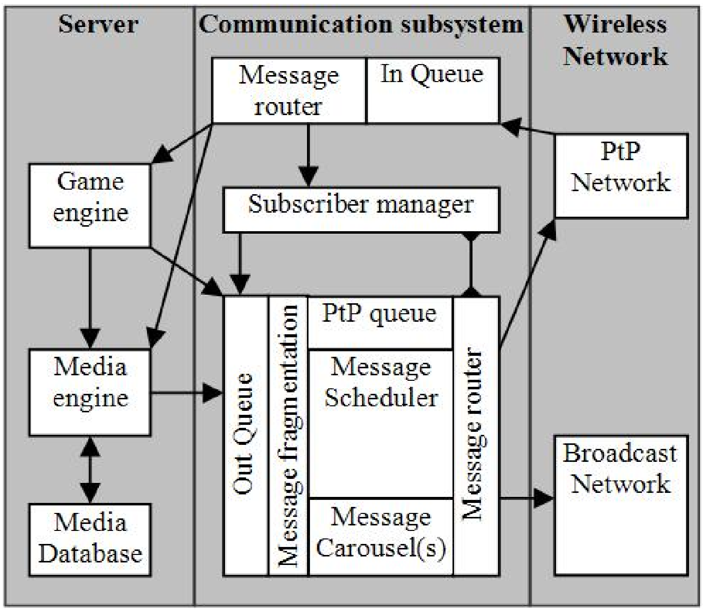

Figure 2: Server logical block diagram

The solution to this problem in the presented system is to use a message scheduler.

2.6.4 Scheduling of content

One challenge for this game environment is to provide all user terminals with data that is needed at the particular moment. The assumption is that the terminal's memory capacity is limited, hence the user terminals must be provided with data on demand. Additionally, some of the data sent is dependent on the status of the game; hence it is not available before it is needed. The demand for data is assumed to be predicted by the game server. Hence, the problem of providing media to the players at the right time is solved by proposing a content scheduling algorithm. Using this algorithm, it can be made sure that every player receives the data intended to him/her before its deadline. The server predicts movements of the player towards the possible quest locations, calculates the distances and expected times of arrival (ETA) of the players to these quest locations based on the received position messages.

The scheduling algorithm must be efficient enough to schedule the data for each player according to its deadlines and reschedule the data if the player chooses one option to go further among many available options. In that case, when the game server predicts the player movements towards the next quest location, the scheduled data for not-taken options will be discarded by the scheduler.

The main steps and pseducode of the scheduling algorithm are mentioned below.

Program 1 Algorithm

```
For every position message, do:

1. Update position of the player
2. Calculate the distances to next quest locations
3. Iterate through quest locations list
   if ( ! ( questLocList[index] == player.Loc )
    {
     calculate ETA to next quest locations
       if ( ETA <= thresholdETA)
        {
         // Prepare and schedule data for player
         outQueue.pushMessage(questData)
        }
    }
4. For each message to be inserted in outQueue,
  // iterate through the outQueue
  for (index=0 ; index < this.index ; index++)
  {
  insert message according to ascending ETA
  }
5. Send data with the highest priority to carousel
     while (outQueue.length>0)
         {
          Carousel.send(outQueue.pop())
         }
```

In this section we have presented how interactivity can be implemented in an environment where content is delivered using a broadcast medium. We have shown how to express interactivity as a scheduling problem and how to solve this problem. Performance, feasibility and scalability issues of the algorithm are discussed in detail in Paper III and simulation results are presented to show how different game parameters influence the in-time delivery of the multimedia content to the players and to study the optimization problem for the presented algorithm.

2.7 Today's Interactive TV applications

Over the last few years, the introduction of widgets (by e.g., Yahoo!, Opera and GoogleTV) on connected television sets opens up a new era for a range of possible social TV applications. There are several social video applications on the web that have been launched recently (e.g., Watchitoo [106], Clipsync, Sofanatics)[60]. Even more applications for iPhone and iPad are being created (e.g., Miso or Tunerfish, Broadcastr), bringing into practice the use of a secondary screen as a social interaction medium, and exploring the potentials of social Web, semantic Web, Linked Data and semantic TV.

Figure 3: Broadcastr Application

Below are the short descriptions of a few interactive applications that are evolved in recent years. In this section, we are not focusing on widely known commercial iTV applications that are related to TV programs such as BBC Test the Nation, BBC Come and Have a Go. Instead, the focus is on social and interactive applications showing that the definition of interactive applications should not depend on the devices used for reception and display, and should not be limited to only TV content.

Broadcastr

Broadcastr is a social-media platform for location-based audio and stories. This application enables people to easily create and share recordings on an interactive map. The users can share oral histories, restaurant reviews, walking tours, personal messages, funny anecdotes, and much more. Each Broadcastr story is pinned to a GPS location. On the Web, the users can access a digital archive of human memory and narrative; on the mobile application, the users can take a GPS-enabled walk while stories about their surroundings stream automatically into their headphones. The users can record their own content, create playlists, follow their friends, and share on the Facebook. Broadcastr is currently in beta [3]. Figure 3 shows screenshot of the application on iPhone.

SPME

In interactive broadcast TV, the viewers can take part in the broadcast shows from their homes using their remote controls. The SPME on TV takes this approach a step further where the viewers not only participate in the broadcast show but also direct the show by exploiting the additional interaction capabilities. The SPME actively engages TV viewers and provides a stage for them to participate in the interactive TV applications that are not yet commercially available. The viewers need some means of communication to the server such as a remote control, a mobile phone etc., and for full participation webcams, microphones, and Internet connection are required. The involvement of heterogeneous devices and information in the SPME makes it an interesting research problem to explore. Auction TV is an example of such an application [99].

The term interactive TV application means different things to different people. No single definition is presently accepted by all researchers since the nature of iTV services is evolving rapidly with the constant and continuous technological changes. The European Broadcasting Union defines iTV applications as enhanced or interactive services with digital Television [7]. BBC defines iTV as follows: iTV is the content and services (in addition to linear TV channels) which are available for digital viewers to navigate through on their TV screens. In practice, this means giving viewers control over some video, audio, graphical and text elements, or allowing them to use simple games and quizzes or send simple communication back to broadcasters [6]. These definitions are usually supported by the broadcasters providing iTV applications and are generally related to and bounded to the TV programs.

Due to the technological developments in the last few years, the landscape of TV has been evolved rapidly. The IPTV and webTV have dramatically changed the definition of the digital TV. IPTV is the next generation TV delivered over a managed network, usually a closed network. It is delivered through Internet and telecommunication lines but not through a browser as the WebTV does. IPTV also gives the ability to track what each user is watching. Due to the IP based TV services, TV and web came closer to each other and the distinction between the iTV applications and the web services is becoming even harder as the operators combine and develop different technologies to serve specific situations. Moreover, the iTV could use web as information space i.e., utilizing web as an ultimate information source by means of a variety of technologies such as semantic web technologies. In the view of current changes in the digital and web technologies and their use, the term iTV application might not be the thing as it was originally a few years ago.

2.8 A Look Towards Future iTV Applications

Due to the advances in the digital technologies, the iTV applications are not bounded to the TV set but a range of devices could also be used for display and interaction purposes. Moreover, after the advent of IPTV, the web and the TV came closer to each other which originate the possibility of a mesh-up of web knowledge in the iTV applications. The future of interactive TV applications based on the convergence of different devices, networks and services is hard to predict. This is an industry in which new giants can emerge in a short time. The trends of the current TV viewing behavior are changing quickly. The success of the social networking portals such as YouTube, Facebook, Twitter etc., have offered exciting new choices for watching TV, and the interactive TV applications "sandbox" is expanding rapidly. The increase in the amount of data available on the social and semantic Webs, APIs and methods to access and manipulate that data, and the new trends in social awareness allow future television a range of new opportunities. It has become possible to develop innovative applications that link TV viewing with both the viewers' personal profiles on social networks, and the social activities across the Web [79]. This convergence results in the social and next generation ubiquitous iTV applications that are innovative and more personal and more social. We believe that the future interactive TV applications would involve not only a wide range of digital devices in a highly interactive, dynamically changing and context aware environment but also data/information from different sources such as Web, advertising networks and physical environment. Connecting up all kinds of information and content by being much more dependent on the environment (physical and social environment) could enable a whole universe of converged iTV applications.

SPME extended

We extended this basic scenario in a number of ways to recommend users only the particular items for bidding which are of their interest. This is done by observing user's TV viewing history, content consumption behavior, personal preferences etc., and mapping this knowledge to user's profile. We assume that various digital devices in a particular user space (e.g., home) can exchange information with each other. Assuming that there are different TV stations and programs, embedding their schedules on their WebPages by using some common semantics for TV program description, the TV can then recommend TV programs for the particular user based on the user's profile. It can further recommend particular interactive applications based on the preferences and the profile. For example, consider a user has been watching Shakira's new video song and has liked her page on the Facebook. The system can recommend him an iTV application of bidding for her latest album based on harvesting and querying his content consumption behavior and the personal preferences given on the social networks. It can add an event to the user's calendar about this auction. At the time of the auction, the banking application on user's smartphone can check the account detail and the user can decide based on the information if he has to join the auction. The system can identify different users and give recommendations according to their own profiles. In this scenario, the information between different sources and devices needs to be communicated and the mesh-up of all this information is required to enable intelligent ambient iTV applications which are not limited to only TV.

We believe that the future interactive TV applications will take benefits from the pervasive computing environment to deliver highly personalized context-aware TV applications to the users. This is due to the increase in the number of digital appliances embedded in the users' surrounding. This enables pervasive interactive space that interconnects users, physical resources and the computational entities. For example, assume an iTV banking application that shows user's account balance and other details on TV screen (or mobile TV screen). User can use this application to pay his bills. The application could detect the presence of other persons in the room and could automatically hide the balance details when the living room TV is being used to display application content. The same application while using on a smartphone could detect it as a personal device and show all banking details.

It is clear that creating these kinds of interactive applications would require establishing some concrete development infrastructures and methodologies which can provide a sufficient level of abstraction to hide the complexity and the details and also provide the interoperability. A common development methodology could provide benefits such as (i) abstracting underlying platform (ii) porting applications to different devices and platforms (iii) reducing efforts in learning APIs.

Currently, there is no commonly agreed suitable method for the development of the iTV applications and several companies such as Aircode, Alticast, ItVBox and Cardinal etc., are using their own tools for the development of iTV applications. Their tools provide graphical environment to easily create simple iTV applications. These environments and tools are too limited for the creation of complex iTV applications that involve information and resources from many sources, and hence, interoperable systems are required that can provide device and information level interoperability for the development of dynamic and context-aware future iTV applications.

3 Interoperable Systems

The recent evolution in the digital information and communication technologies has created an environment which contains a wide range of devices with computers embedded in them. For instance, a modern house or office environment has a number of digital devices. Some of these devices, e.g., mobile phones, PDAs, etc. are actually designed to interact with the user and the digital environment. However, in order to take full advantage of this modern device-rich environment, the interaction between the devices is required. The vision of *ubiquitous computing* was first envisioned by Mark Weiser in 1991 [103]. According to him, the ubiquitous computing is the way of enhancing computer use by making many computers available in the physical environment, but making them invisible to the users. From the technology point of view, the ubiquitous computing is the mixture of intelligent environments, networking, mobile computing, appliances, context awareness, etc. From the user's point of view, the computing is invisible and is done in the background, yet provides assistance in their everyday life.

This vision of ubiquitous computing has been accredited by many research fields such as mobile computing, wireless communication, sensor networks and human computer interaction [36]. For example, wireless communication standards such as WiFi, Bluetooth, etc., have enabled users to interact with each other using their handheld terminals, and the sensing technologies such as RFID, NFC, GPS, etc., enabled context-awareness in the applications. Similarly, today's modern appliances have a range of sensors in them to improve the efficiency of the appliances and to provide more convenience to the users. However, all these technical developments provide a limited form of interactivity between the devices e.g., information exchange using Bluetooth is only possible between the devices conforming to the Bluetooth standard. This could be the initial step of the ubiquitous computing vision but a framework for ubiquitous computing that enables the sharing of information between different devices or systems is still required.

Interoperability is one of the key issues in ubiquitous computing as multi-vendor devices offer increasing number of services and applications by exploiting information from the physical environment [64]. To attain efficient interoperability devices need to be able to find and communicate with each other. To achieve this, a device must be able to find data from another device, and also be able to understand this data. Today this is a major problem due to the lack of common standard for semantic interoperability. There is no understanding on a common protocol, or on how data should be described in such a way that it could be conveniently shared across the devices. There is a need for some infrastructure that makes the co-existence of multi-vendor devices and services easy, rapid and efficient both for the developers and for the end-users. This will result in a simplified and speedy design, development and deployment of cross-domain applications. One possible solution of this problem is to use semantic technologies together with the smart environment.

3.1 Smart Environment

A Smart Environment can be defined as the combination of a physical environment, an infrastructure for data storage and management (e.g., Smart Space), a collection of embedded systems gathering heterogeneous data and information from the environment, and a communication solution to deliver this data to the Smart Space [91]. As defined by Cook [33], a Smart Environment is a world where all kinds of smart devices are continuously working to make users' lives more comfortable. The vision of the smart environment is to use the information about the surrounding environment to create a variety of applications where the end-users must be empowered to easily modify the applications as well as mash-up the information to create new applications when needed.

3.2 Smart Space Approach

The ubiquitous computing paradigm has realized information processing by the entities available in our environment [80]. It enables the vision of the Smart Spaces. The Smart Space is the fundamental component of smart environment as it is the digital representation of the smart environment where relevant real-world information is stored and kept up-to-date. A *Smart Space* is an abstraction of space that encapsulates both the information in a physical space as well as access to this information, such that it allows devices to join and leave the space. In this way, a Smart Space becomes a dynamic environment whose membership changes over time when the set of entities interact with it to share information between them.

Smart Spaces provide several benefits to make themselves suitable for the ubiquitous computing. (1) Smart Space provides anonymous data exchange as the data can be retrieved independently of a particular publisher of that data. (2) The data exchange is independent of the time as it does not require direct point-to-point connection between the devices. It means that the data exchange can happen over a longer period and the devices do not need to be present in the Smart Space at the same time. (3) Smart Space enables simultaneous data access requests of multiple devices in a transparent manner. (4) High scalability can be achieved as the information in the Smart Space is independent of the number of receivers retrieving it. (5) The information remains accessible in the Smart Space until it is explicitly removed. By exposing the internal data and functionality of the devices and ensuring interoperability of data, a whole new universe of applications will be possible. Smart Space provides integration of heterogeneous devices by providing access protocols to access information and hence, enabling a wide range of applications for all domains.

3.2.1 Requirements and Challenges

In designing a Smart Space, there are many requirements and challenges that need to be taken into account. For example, the user should be able to access and use the best available services and resources in the mobile and dynamically changing environment in which he is currently present. In this section, we outline the requirements that are crucial in designing and implementing the Smart Spaces [19].

Interoperability: The first and the most important design requirement is the interoperability of devices and services in the Smart Space. Interoperability means that the devices and services from different manufacturers and vendors should be able to understand each other, and exchange information between them to perform various tasks [63]. This is the most crucial requirement of a Smart Space because Smart Spaces aim to incorporate a range of devices from different domains and are not specific to a single particular architecture. The Smart Space must provide an interoperability mechanism for interoperating components.

Scalability: As Smart Spaces are primarily multifaceted due to the fact that a unique combinations of devices and services participate in a particular context, it raises the requirement that the new and cutting-edge digital devices should be able to join the Smart Space. The functionality of the Smart Space should be easily extensible as it emerges over time when new devices join the Smart Space.

Portability: The Smart Space infrastructure includes and employs devices, technologies and services from different application domains. Since it is not clear which technology is actually employed for a certain kind of service, the services in the Smart Space should be portable to a wide range of devices and application domains.

Security: The Personal Smart Spaces must have some security mechanism so that the services which are personal to the user of the Smart Space cannot be available to other people. When a Personal Smart Space of one user interacts with the Smart Space of another person, this interaction should be secure and only the services and functionalities which are explicitly defined by the user should be available to the other person.

Complexity: The complexity of designing and implementing the Smart Spaces should not be high because the increase in complexity will make it difficult to maintain and upgrade. As Smart Spaces are dynamic environments and new devices and services can join at any time, it should be easy to develop and manage the operations.

3.2.2 Smart Space enabling Technologies

The Smart Space is not a completely new phenomenon that incorporates a whole new range of technologies, domains and standards to provide different services to the users. It uses already available technologies and standards and incorporates them in a way that gives a wide range of cross-domain intelligent and adaptive services and applications to its users. Some of the widely used technologies and standards in the development of Smart Spaces are given below.

Sensors: Sensors are probably one of the most important technologies that play a crucial role in the development of Smart Spaces. This is because the Personal Smart Spaces are pervasive or context-aware and the sensors are the most used sources for providing information about the user's current environment. A range of sensors can provide continuous and accurate characteristics of the Space in which users are currently present. The information from multiple sensors are collected and processed by the different devices in the Smart Space in order to take some actions based on the current context. In this way, the Smart Space can automate many services which otherwise require users' involvements. In order to make Smart Space services more adaptive to the users, there must be a standard solution for representation and exchanging information between the sensors and processing devices. There are a range of ambient applications based on sensors.

Semantic web: The Semantic web enables well structured data representation and interpretation than the World Wide Web (WWW), and it has been predicted that the semantic web will be a key enabler of future Internet. The enabling underlying technologies of the semantic web such as the Resource Description Framework (RDF) [9] and OWL provide data representation in the machine readable form. This could be an enabling technology for the Smart Spaces as the information structure and semantics of information should be represented in a way that could be readable by all the devices in the Smart Space. In this way, the interoperability issue can be solved. The semantic web technologies can provide the interoperability solution for the Smart Space applications.

Mobile devices: In Smart Spaces, the devices are responsible for information processing and for providing the resulting services to the users. The mobile devices available today are capable of performing complex tasks due to the boost in processing power, data storage and the communication technologies. As mobile devices are the personal devices that are available to the users anywhere and anytime, these devices could be an essential component of Personal Smart Spaces. Existing inter-device solutions could be applied for the communication between the mobile devices and other smart objects in the Personal Smart Space.

Interfaces and protocols: There must be optimized interfaces and protocols for the communication between the devices that are available in the Personal Smart Space. There are currently many interface solutions for the PC world e.g., PCI and USB. For Personal Smart Spaces, there is a strong need for this kind of interfaces and communication protocols that can be used to exchange information between heterogeneous devices from different domains. One fundamental technology for this purpose could be the Network on Terminal Architecture (NoTA) [4] that is based on the Device Interconnect Protocol (DIP). DIP can be implemented for many distinct physical interfaces such as high speed serial interfaces or wireless transport interfaces such as Bluetooth.

3.3 Smart-M3: A Particular Implementation of Smart Spaces

Our approach for rapid development of interoperable Smart Space applications (Paper IV) is based on Nokia's open source Smart-M3 architecture [74][11]. The Smart-M3 architecture provides a particular implementation of the Smart Space where the central repository of information is the Semantic Information Broker (SIB). The smart-M3 space is composed of one or more SIBs where information may be distributed over several SIBs for the later case. The information in the M3 space is the union of information stored in the SIBs associated with that space. The set of SIBs in a M3 space are completely routable and the devices see the same information. Hence, it does not matter to which particular SIB in a M3 space a device is connected. The information is accessed and processed by the entities called Knowledge Processors (KPs). KPs interact with the M3 space by inserting, retrieving or querying the information in any of the participating SIBs using access methods defined by the Smart Space Access Protocol (SSAP). Smart-M3 provides information level interoperability to the objects and devices in the physical space by defining common information representation models, such as, RDF. In this way, it provides a device, vendor and domain independent solution for interoperability.

Figure 4 shows the overall Smart-M3 architecture. It comprises the following components and concepts that work together to provide a functional framework.

The *SIB* is the core component in Smart-M3 as it is responsible for information storage, sharing and management. The information is stored in the SIB as RDF graphs. The SIB acts as information broker between the devices for storing, receiving, or querying information in the RDF triplestore contained in the SIB. For information sharing, devices do not communicate directly with each other, instead information is passed via the SIB through the SSAP communication protocol.

SSAP defines the following operations to provide access to the M3 space.

Join : Join the smart-M3 space to access information

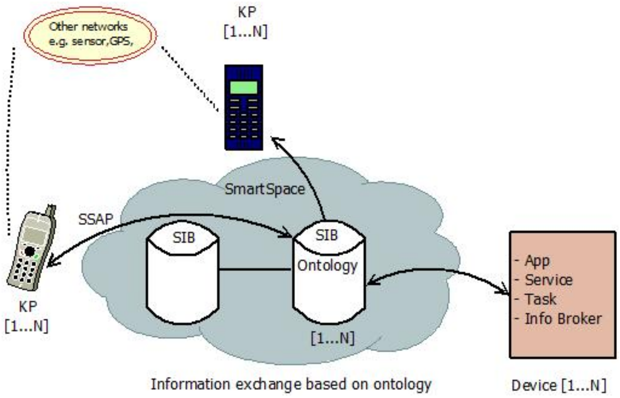

Figure 4: Smart-M3 Architecture

Insert : Insert information into the SIB
Remove : Remove information from the SIB
Update : Update the information in the SIB
Query : Query for information in the SIB
Subscribe : Information updates are reported to the subscriber
Leave : Leave the smart-M3 space

The *Knowledge Processor (KP)* is the component that performs only a single task. A KP must first join the M3 space in order to access the information. It can then manipulate the information in the SIB by accessing it through the operations such as insert, remove, update, query, and subscribe, defined by the SSAP communication protocol. A single device can run any number of KPs. The functionality of each KP is defined by the ontology it is related to and the KP accesses and processes the information according to its application or service ontology.

The *Ontology* represents the knowledge and its structure in the Smart Space and each application has its own ontology. The ontology enables the KPs to access and process the information related to their functionality from the Smart Space and hence it directs the KPs through the Smart Space.

The *Application* is constructed by the composition of several KPs where each KP performs a single task. The application design in this approach differs from the traditional single device control-oriented application.

The Smart-M3 platform supports features like asynchronous queries from the back-end database. This feature is important when optimizing the semantic environment for low-end embedded devices with limited processing capabilities, and potentially battery constraints for cellular phones and sensors of different types. These kinds of devices are being targeted by the Smart-M3 environment and this infrastructure has been employed in many case studies [49][100] and approaches [75][43]for interoperable application development [54][90]. Other examples of ontology driven application development for smart environment are provided in [55][18]. In [94], the authors presented an approach for anonymous agent coordination in Smart Spaces.

.

3.4 Ontologies

W3C has been working on the specifications for semantic web for many years. The goal is to develop standards to make the information machine readable for sharing and using across multiple platforms. Ontologies facilitate the common understanding of information being shared [27]. The W3C chose the Resource Description Language, RDF, as its representation model upon which the Web Ontology Language, OWL has been built.

3.4.1 RDF

RDF is based on the idea of making statements about resources. These statements are in the form of triples as subject-predicate-object. For example, the sky can be described as blue by using the statement "Sky-hasColor-Blue". Using this approach any kind of data can be described in such a way that a developer and a device both can understand. The RDF provides a way for expressing the semantics of the information.

RDF solves the data interchange problem on syntactic level but does not provide enough information for expressing the agreement between several different concepts. Applications must have the same interpretation of data in order to provide interoperability between devices from different domains. Ontology driven approaches solve this problem by representing the information and its relationship.

3.4.2 OWL

The Web Ontology Language, OWL [37], is developed by W3C, for empowering the semantic web. It is divided into three sub-languages, OWL Lite, OWL DL and OWL Full. OWL is always a correct RDF. The OWL sub-languages can be seen as formal extensions to the RDF Schema.

Since smart-M3 is not constraint to a specific structure of information, it enables the usage of ontologies to express the information and relations in an application. The ontology enables the KPs to access and process the information related to their functionality from the M3 space and hence it directs the KPs through the space.

In our application development approach, we have chosen the OWL DL sub-language as it contains additional expressions that we find to be important, compared to the less expressive sub-language OWL Lite. The `oneOf`

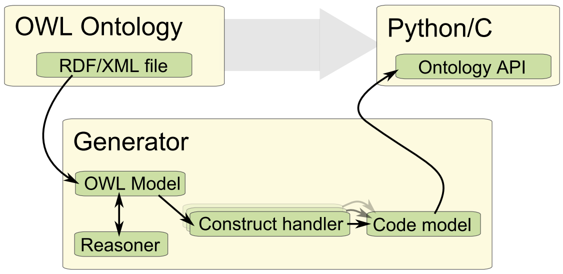

Figure 5: Framework overview.

class axiom makes it possible to define a class as an enumeration set, e.g. a `week` is characterized by containing only a very specific set of `day` instances. The `disjointWith` constructor makes it possible to define rules such that the classes `Man` and `Woman` are subclasses of `Human`, but a `Man` cannot be a `Woman` and vice versa. Full support for cardinality is important and is defined in OWL DL, whereas OWL Lite is limited to defining cardinality restrictions to being either 0 or 1, or no restriction at all. It is important to be able to specify, for example that a class `week` is defined by seven instances of `day`. The property restriction `hasValue` is powerful, as it makes it possible to have a property require a certain individual as a value. For example, instances of `dutchCitizen` can be characterized to have `theNetherlands` as their nationality.

OWL Full has the same constructs as OWL DL, but OWL DL has some restrictions in order to be decidable. OWL DL requires type separation, i.e. a class cannot also be an instance or a property. OWL DL also requires that a property is an `ObjectProperty` or a `DatatypeProperty`.

3.5 Application Development Tools for Smart-M3

We chose the ontology-driven application development approach for smart-M3 (Paper IV) and developed tools [56] for mapping of ontologies to Object Oriented Programming (OOP). Our approach consists of two parts.

I The first part is the code generator that creates a static API from an OWL ontology as illustrated in Figure 5. This mapping is done according to a set of static mappings. These mappings generate native Python classes, methods and variable declarations which can then be used by the KP developer to access the data in the SIB as structured and specified in the OWL ontology. The generator loads an OWL ontology into a Java ontology model which provides interfaces for accessing the RDF graph. A reasoner is connected to the model

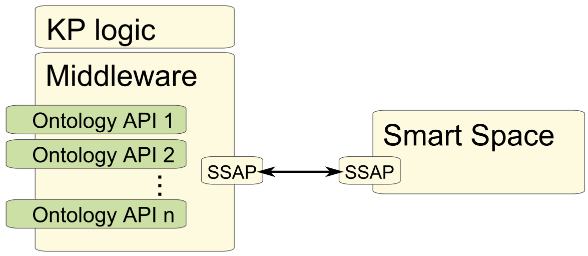

Figure 6: Runtime middleware for *Smart-M3*.

to complete the inferred part of the ontology. The generator then lists all named classes in the ontology and the handler creates a counterpart OWL class in Python which is added to the code model. The class handler will list all properties and call the *ObjectProperty* and *DatatypeProperty* handlers which, in turn, translate every restriction that the property may have, e.g., `Cardinality` and `Range` restrictions.

II The second part is the middleware layer which abstracts the communication with the SIB as shown in Figure 6. Its main functionality to the generated API is triple handling. This consists of inserting, removing, and updating triples and committing changes to the smartspace. It also provides functionality for synchronous and asynchronous querying. Our approach enables application developers to use the generated API to develop new KPs and applications without worrying about the SIB interface as the generated API takes care of the connection to the SIB each time an object is created [10]. From the Smart-M3 point of view, the proposed framework simplifies the development of KPs by making the Smart Space interface more abstract and hiding the underlying complexity involved in ontology-driven approaches.

In this application development approach, the concept of application is not the traditional control-oriented application running on a single device but the application is constructed from a number of independently operated KPs which may run on different devices and group together to be perceived as a single application. For instance, chat, calendar synchronization and multiplayer games are examples of applications using this approach where a set of KPs each handling a single task run on multiple smart devices and coordinate and interact with each other through the SIB to make a complete application. This coordination between KPs are done in the form of data exchange through the SIB where KPs subscribe to or query for specific data to perform their specified task. Application ontologies are used to describe data in the SIB and direct the KPs to access and manipulate data related to their functionality.

3.6 A Look Towards Future Smart Space Applications

Pervasive Computing or ambient intelligence [84] aims to provide services transparently to the users in their everyday life anywhere and anytime. Since the Smart Spaces can address the needs of the users on-the-move, they require pervasiveness to offer the services to the users wherever the users go. In this way, a whole new universe of services could be provided which are otherwise not possible. In this section, we give some example scenarios which could be possible with the Smart Spaces.

Smart Spaces in Health Care Infrastructure

The Smart Spaces could be used in the health care infrastructure to automate the tasks involved in it. For example, a patient has its own Personal Smart Space (PSS) which includes different devices to regularly monitor his heart beat, blood pressure etc and if there is something wrong, the patient's PSS could interact with doctors PSS to book an appointment. After the appointment is confirmed, the PSS will inform the patient about the appointment by sending an SMS to his mobile phone. At the appointment day, the patient's data would be transfered to the doctors PSS so that the doctor could diagnose and suggest some medicine to the patient. In this way, the patient does not need to monitor and call the doctor manually as his PSS will do it automatically and inform him about it. These kinds of services provided by interacting different PSS and sharing information between each other would be revolutionary in the health care infrastructure [100][96].

Smart Spaces in Office Infrastructure

Smart Spaces could also provide intelligent services and applications in the office domain. For example, a person with his PSS in the office is about to give a presentation in the meeting room. His PSS will interact with the PSS of the meeting room and could automatically use the projector in the meeting room to display the slides he has in his laptop. At the time he is presenting, his PSS could switch his mobile phone to silent mode and any incoming calls could be routed to his voice mail. If the person wants to print his slides, the PSS could locate the nearest printer to send the print command and then guide him to that printer's location automatically.

Smart Spaces in Travel and Tourism Infrastructure

The Smart Spaces could also provide a range of intelligent services to their users during their traveling and tourism. For example, if a person goes to a foreign city, his PSS could log all the places he visits during his stay there. His PSS could interact with, for instance, the fort's PSS he visits and retrieve the history from there and present it to the user. In this way, it could also act as the travel guide for the user. At the end of his journey, his mobile phone could give him a report of all the places he visited, and at what time he visited them. He could share this report with his friends or family.

Smart Spaces in Learning Infrastructure

The Smart Spaces could also be used in the learning infrastructure. Each student has his own PSS and he can choose learning service from heterogeneous sources by interacting with their SS. For example, the PSS of a user A contains his background profile information such as his previous education, degrees, jobs and the location etc., and his profile is updated dynamically with his learning process. His PSS could search for his required course and interact with the PSS of the relevant learning service provider for an enrollment process [93].

There could be a wide range of services and applications in addition to the above scenarios that could be provided using the Smart Spaces. The SS could make life of the users very easy by automating many of their everyday tasks. The SS learns from the previous events and improve itself based on this history information to adapt to the users preferences as accurately as possible. The scenarios described above illustrate how useful the SS is for providing personal services to the users in different domains.

3.7 Literature Review

The Smart Space infrastructure is explored mainly by the area of context aware computing. The research in the context-aware computing provides a number of context-aware systems and approaches for application development. Starting with the context modeling, there are plenty of different points of view, but since the ontology model wins the rest regarding simplicity, flexibility, extensibility and expressiveness, we focus on the comparison of ontology-based systems. A good compendium of pros and cons in relation to the design architectures and context models [20] presents several ontology-based context models.

CoBrA and SOCAM are some of them using their own OWL-based approach for context processing while others such as Context Managing Toolkit describe context in RDF. CoBrA [29] is proposed as an agent-based infrastructure for context modeling, context reasoning and knowledge sharing using context acquisition components. The Soupa and CoBrA-Ont Ontologies are some of the related tools. They also provide techniques for user's privacy control. SOCAM [46] introduces another architecture for building context-aware services focused on information sensing and context providers. A central server or context interpreter gains context data through distributed context providers and processes it before sending it to the clients. These and other projects such as [105] focus basically on creating ontologies for context-representation but they do not intend to build a framework for creating location- and context-aware development of services or applications based on the semantic back end.

In [87], the authors extend typical operating system concepts to include context-awareness. Gaia presents a similar representation to our RDF Triples with 4-ary predicates (the 4th one is context-type) and it does not use OWL but DAML + OIL. Gaia's MVC model also differs from our blackboard architecture. In [59] and [48], the authors presented a framework that targets only smart phone platform and technologies.

Context Toolkit [39] presents an approach to enable application development by using reusable components. However, its attribute-value tuples are not clearly meaningful enough making the application programming restricted. Other approaches for context aware applications and services are presented in [61][82]. In [21], the authors describe a semantic approach for context-aware adaptation and decision taking. The research challenges in mobile context-aware computing and service development are presented in [78]. In [45], the authors describe an approach for analyzing sensed context information to help in the generation, documentation and assessment of the designs of context-aware interactive applications. None of these approaches deal with the Smart Spaces and smart environment which have different challenges than traditional context-aware approaches in terms of scalability, heterogeneity and dynamicity of information in the physical environment.

Service discovery and service composition in the dynamic smart environment are among the greatest challenges. There are different approaches, tools, and systems that take the user's request and try to discover the services that best matches to fulfill the task. In [104], the authors proposed a system consisting of a middleware and user-level tools that enable the end-users to combine, configure and control the services using their home devices.

In [83], the author presented an approach of macroprograming of wireless sensor networks (WSN) using the mobile agents. The WSN makes an infrastructure and queries from the end-users are provided over this infrastructure. Due to the technological advances, non-expert users have to accomplish non-trivial tasks in application and device-rich computing environments [68]. Task computing [66][95] is an approach that reduces these difficulties by shifting focus to what users want to do (i.e., on the tasks at hand) rather than on the specific means for doing those tasks. Other approaches for end-user task composition [28] are presented in [65] and [102].

4 Combining Smart Spaces and Digital TV

4.1 Motivation

Due to the recent technological developments, ICT landscape is evolving into a highly interactive distributed environment that demands integration of information in heterogeneous technologies and systems. Information in this environment is accessed using a range of different devices. These devices include portable devices (such as mobile phones, PDAs, smart phones, tablets) and fixed or non-portable devices (such as TV, set-top boxes, desktop computers, Personal video recorders). These devices provide new possibilities of interaction, and all of them have the capacity to execute applications and share information with each other. With the birth of IPTV, Television and Web came closer to each other by sharing a substantial set of methodologies to provide the users an immerse interactive experience. The users now have more control over data and content creation, consumption and sharing. It is foreseen that the future interactive TV applications would involve not only a wide range of digital devices in a highly interactive, dynamically changing and context-aware environment but also data/information from different sources such as web. Connecting up all kinds of information and content by being much more dependent on the environment (physical and social environment) could enable a whole universe of converged iTV applications. For example, assume an iTV banking application that shows user's account balance and other details on TV screen (or mobile TV screen). The user can use this application to pay his bills. The application could detect the presence of other persons in the room and could automatically hide the balance details when the living room TV is being used to display application content. The same application while using on the smartphone could detect it as a personal device and show all banking details.

It is clear that creating such interactive applications requires establishing concrete development infrastructures and methodologies that can provide a sufficient level of abstraction to hide the complexity. Currently, there is no commonly agreed suitable method for the development of iTV applications and different organizations have their own platforms, approaches and APIs (JavaTV API, MHP, OCAP API). Several companies are using their own tools for the development of iTV applications. Their tools provide graphical environment to easily create simple iTV applications. These environments and tools are too limited for the creation of complex iTV applications that involve information and resources from many sources. In order to make full use of the power of interactivity and content consumption, data and device interoperability issues must be solved and data must be structured in a way that could enable multiple devices to consume and share data between them. Smart Spaces provide solution for the interoperability problem by standardizing how to describe data formats. Smart Spaces could take advantage of the digital TV / IPTV technologies to deliver content/data to the receiving devices and this data could be shared between heterogeneous devices present in the Smart Space. The Smart Space application development tools could be used to develop interoperable interactive TV applications that employ mixture of information from different sources and devices rather than a standard remote control. Moreover, traditional iTV application development methods and techniques must be replaced by scalable, agile and configurable methodologies.

We propose to use the ontology driven iTV applications development because

- Ontologies are perfect candidates for modeling context information which is highly desirable in the future iTV applications to provide personal services dependent on the environment. Users at different contexts have different needs and expectations and ontologies can model the users' context in an effective way.

- As the concept of iTV has been evolving after the advent of IPTV, users are expecting highly dynamic systems where they can join and leave anytime to consume the services. Smart Space provides this dynamic environment, and ontologies are important concept in the Smart Space infrastructure.

- Reasoning and inferring new knowledge from the available information and searching and querying for the desired services are becoming essential part of iTV usages as TV came closer to the web in recent years. Ontology driven architectures can provide reasoning capabilities in an effective way.

- Users now have a range of devices in addition to the TV in their personal space to interact and consume iTV applications. It enables an ubiquitous system in which information from heterogeneous information sources such as sensors, digital appliances, web, smartphones, TV, PVR etc., is used for realization of truly interactive applications. Ontologies are immediate solution for handling heterogeneity and provide information level interoperability to the users.

It is perceived that the use of the ontologies will essentially change the way in which software systems/applications are built, and the software designers will have libraries of ontologies from which they can choose relevant ones. Use of ontologies in application development provides competitive advantages over the traditional approach, enabling greater information sharing and reuse. The ontology-driven development (ODD) additionally exploits knowledge exploitation using reasoning over the maintained ontology.

In this section, we explore the potential of introduction of Smart Spaces in the design and development of interactive TV applications. We have previously developed ontology-driven tools and frameworks for rapid application development for the Smart Spaces. We are now applying our ideas and methodologies of Smart Spaces to interactive TV domain as we believe this convergence could provide potential benefits in terms of value-added applications to the users. Our tools and methodologies provide benefits which are not currently realized in the iTV domain such as (i) abstracting underlying platform (ii) porting applications to different devices and platforms (iii) reducing efforts in learning APIs. Our approach for developing highly interactive applications deals with the key issues such as flexibility with respect to adding new devices and services to the Smart Space, high level of abstractions, rule-based reasoning, task-based and recommendation-based design and automatic code generation from the application ontology.

4.2 Enhanced iTV Application Scenario

We chose AuctionTV example scenario presented in [99] because it exploits additional interaction and participation by the iTV users. This application allows one of the participants to offer some item on sale through the auction. This auctioneer gets the role master and the other users join afterwards get the role participant. Whenever a participant p bids on the item, the auctioneer raises the price confirming the bid. When the acceptable bid has been made and confirmed, the bidding process can be ended and the participant with the highest bid gets the role winner.

We extended this basic scenario in number of ways to recommend users only the particular items for bidding which are of their interest. This is done by observing user's TV viewing history, content consumption behavior, personal preferences etc., and mapping all this knowledge to the user's profile ontology. We assume that various digital devices in a particular user space (e.g., home) can exchange information through the Smart Space. Assuming that there are different TV stations and programs, embedding their schedules on their WebPages by using some common semantic for program description, the TV can then recommend TV programs for the particular user based on the user's profile ontology. It can further recommend the particular interactive applications based on the preferences and profile ontology. For example, consider a user has been watching Shakira's new video song and has liked her page on the Facebook. The system can recommend him an iTV application of bidding for her latest album based on harvesting and querying his content consumption behavior and the personal preferences given on the social networks. It can add an event to the user's calendar about the auction. At the time of the auction, the banking application on user's smartphone can check the account detail and the user can decide based on the information if he has to join the auction. The system can identify different users and give recommendations according to the particular user's profile. In this scenario, information between different sources and devices is communicated through the Smart Space. Smart Space allows the fusion of all this information to enable intelligent ambient iTV applications which are not limited to only TV.

This interactive application exhibits important properties which enable it to be modeled and developed using our ontology driven Smart Space approach. Firstly, the user's preferences can be inferred by semantically matching user's profile with metadata of the content provided by the content providers. This activates appropriate services for the user from the available resources. Secondly, heterogeneous devices could be used for the interaction with the system, making it a multi-device environment. Thirdly, the application is driven by the user's actions and the time-based events could also be used. Fourthly, subscriptions could be used in the situations where one action could be performed before any other action e.g., after a bid is made, the amount of the next bid should be raised.

All these properties make the Smart Space an ideal choice for the development of these kinds of interactive TV applications. The Smart Space addresses the issues of reasoning, heterogeneous devices, interoperability, subscription based and user-driven actions. Our approach for the application development provides higher level of abstraction by automatically generating ontology API from the application ontology by mapping OWL ontology concepts into Object Oriented programming language concepts. This enables application developers to create innovative Smart Space applications using traditional Object Oriented programming concepts without worrying about the complexity of OWL ontologies.

4.3 Implementation

We have developed a Python Module for easy definition of the rules in our approach for Smart-M3. The interaction with the Semantic Information Broker (SIB) is made so that the Python developer does not have to deal with the RDF triples or semantic technologies such as query languages to access the central repository of shared information. The Python Rule module makes use of the Ontology Library Generator (OWL to Python) and its framework as an abstraction of the interface with the SIB. The *Python-Rules* module allows the programmer to write rules on the fly i.e., they can be executed directly and interpreted as any other Python statement. The execution of the rules is achieved through the subscription capability of the SSAP protocol to the *Smart Space*. This generates asynchronous notifications when the changes occur in the *Smart Space*. However, this is a concrete implementation of *Smart Space* broker (*Smart-M3*) but the *PythonRules* module aims at being independent of the information broker or the repository used.

Our application development tools are used as follows:

I *Smart-M3 Ontology to Python API Generator*: First of all, the ontologies to be used need to be converted automatically to their corresponding Python classes. For this purpose, our Ontology Library [56] is used, generating classes for each Ontology class together with their properties and methods.

II *Programming Knowledge Processors*: When the Ontology Library has generated the needed classes with the included middleware, containing getters and setters methods, this middleware already abstracts the communication with the SIB allowing programming of the KPs. The generated `EmptyKP.py` file can be used as a starting template; instance declarations automatically translate to RDF insertions in the SIB (after committing changes). This allows other applications connected to the same *Smart Space* to know about the existence of those individuals and to interact with them.

III *Python Rules for Smart Space programming*: Since the previous middleware still requires a considerable number of calls before achieving interaction with the repository, as well as working with specific namespaces, *PythonRules* provides a higher abstraction layer for fast specification and configuration of the *Smart Space*'s behavior.

The rules can either be executed synchronously (when declared in real time) or stored together in the class *RuleSoup*. In the latter case they can run all at once and executed asynchronously (when their conditions are satisfied).

4.3.1 Ontology Development

As our approach is based on ontology driven application development, the first step is to create application and domain ontologies. We developed an application ontology for the application scenario described in section IV. Figure 7 illustrates the excerpt from the application ontology. The ontology shows the semantic relationships between different concepts.

As we are dealing with the interactive TV domain in this particular scenario, the domain ontology consists of the concepts related to the TV content such as category, title, actors, schedule etc. The domain ontology can be automatically generated using the metadata available for each TV

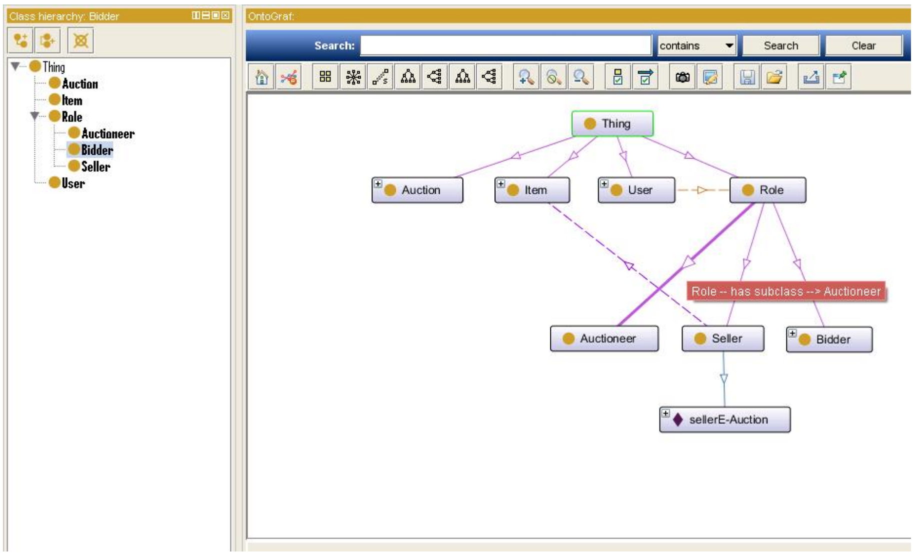

Figure 7: Application Ontology

program or it can be created manually. In this use case, we used a simplified version of BBC program ontology [5].

4.3.2 Programming Knowledge Processors: iTV Use case

When the application programmer does not deal with the RDF Triples directly, but mainly with the Python ordinary statements, the translation of problems described with natural language into programs becomes much easier. This section describes two knowledge processors that are created for the evaluation of the example scenario. The *TVBroadcasterKP* creates a new Calendar and a new Event. The *AuctionItemsManagerKP* is an example Class that creates new Items for sale at an Auction. These knowledge processors use the APIs that are generated from the ontologies by our tool. Detailed explanation and implementation of these two KPs are presented in Paper VII of this thesis.

4.3.3 Inference Rules

We created two simple rules for the evaluation purpose as described in Paper VII.

Rule 1: If in an electronic auction, the new gadget Cool100in1 is offered for sale, Natalia would like to be notified as soon as this item appears in the Auction. If this occurs, an event on her calendar should be created immediately to remind her to bid. The particular implementation of this rule using our PythonRule module is given in the listing 4.3.3

```
withClause = With([newGadgetItem])
whenClause = When(newGadgetItem.getProperty("ItemName") == "Cool100in1")
thenClause = Then([remindToBidEvent.new(Event),
   remindToBidEvent.setProperty(Title = XSDString("Cool100in1 in e-Auction, Remember to Bid!")),
   remindToBidEvent.setProperty(DtStart = XSDDateTime(newGadgetItem.getProperty("HasDateOfStart"))),
   remindToBidEvent.setObject("MemberOf", nataliaCalendar.get()),
   GoogleCalendar("smartspacecalendar@gmail.com", "smartspace").addEvent("Remember to Bid for Cool100in1!","","e-Auction", dayBefore(XSDDateTime(newGadgetItem.getProperty("DateOfStart"))),
         dayBefore1hourAfter(XSDDateTime(newGadgetItem.getProperty("DateOfStart"))), None)])

rule = PythonRule(withClause, whenClause, thenClause)
```

Rule 2: If there is a new event in the broadcaster calendar which includes Natalia's favorite documentary, *Spanish people around the world*, happening in Finland, she would like to be notified on her calendar not to miss it.

```
withClause = With([favouriteDocumentaryEvent])
whenClause = When(favouriteDocumentaryEvent.getProperty("Title") == "Spanish people around the world: Finland")
thenClause = Then([remindDocumentaryEvent.new(Event),
    remindDocumentaryEvent.setProperty(Title = XSDString("Spanish people around the world in Finland!")),
    remindDocumentaryEvent.setProperty(DtStart = XSDDateTime(favouriteDocumentaryEvent.getProperty("DtStart")-oneDay)),
    remindDocumentaryEvent.setObject("MemberOf", nataliaCalendar.get()),
    GoogleCalendar("smartspacecalendar@gmail.com", "smartspace").addEvent("Tomorrow is your favourite documentary","","BBC Broadcaster", dayBefore(XSDDateTime(favouriteDocumentaryEvent.getProperty("DtStart"))), dayBefore1hourAfter(XSDDateTime(favouriteDocumentaryEvent.getProperty("DtStart"))), None)])

rule = PythonRule(withClause, whenClause, thenClause)

# Running the whole RuleSoup...
ruleSoup = RuleSoup()
ruleSoup.addRule(rule)
```

```
ruleSoup.runAllRules()
# Waiting for creation of new Events...
sys.exit(app.exec_())
```

For this use case, we first developed the ontologies in Protégé and then these ontologies are fed as input to our tool to generate ontology libraries. We then implemented the knowledge processors that reflect the functionality of the application. Knowledge processors use the generated ontology APIs. We then defined the rules using PythonRules Module which describes the rule expressions embedded into Python language.

In this section we presented how to develop interactive TV application using ontology-driven Smart Space approach. We have developed a context-aware service in the domain of interactive TV and evaluated it using our developed ontology-driven tools. Our approach for ontology-driven iTV application development worked well for these simple rules. We aim to extend it to evaluate more complex scenarios and context-aware iTV applications.

5 Contribution of the Thesis

The contribution of the thesis is described in the context of the individual research papers, which are presented in Part II of this thesis. This chapter presents a summary of the original publications along with the description of the author's contribution in each publication.

5.1 Overview of the Papers

5.1.1 Paper I: System Architecture and Interactivity Model for Mobile TV Applications

Paper I was the result of the research in the emerging area of interactive mobile TV. We investigated the existing software platform solutions to explore their capabilities and limitations to support the interactive mobile TV applications. Based on this research study, we identified the demands and constraints for the software structure of mobile TV, suggested possible solutions for them and proposed prototype software architecture for interactive mobile TV applications. The Interactivity model for the interactive mobile TV applications was proposed that facilitates the evaluation of these applications to prospect the terminal and server side's requirements. The proposed interactivity model provides the basis to explore many worthwhile aspects related to the interactive applications for mobile TV and specifies the overall image of the mobile interactive system. The evaluation of the specific interactive applications provides the idea of how and when the viewers of the mobile TV interact with the content.

Author's contribution: The basic idea presented in this paper was initially developed by the author while working on his Master's thesis under the supervision of Assistant Professor Jerker Björkqvist. The author then refined the idea, proposed the model, and wrote the paper.

5.1.2 Paper II: Interactive Applications for Mobile TV

Paper II addresses the issues of usability and user experience in interactive TV applications. It explores the concepts of interactive applications, their usability and system architectural aspects in interactive mobile TV. The paper evaluates selected application scenarios for different interactive applications that can be supported by future Mobile TV and demonstrates how the proposed architecture in Paper I supports these scenarios.

Author's contribution: The author further extended the ideas presented in Paper I and wrote this paper.

5.1.3 Paper III: Content Scheduling in Multimedia Interactive Mobile Games

Paper III presents an implementation of a particular interactive application described in Paper I and Paper II. We present how to implement interactive multimedia services using a DVB-H broadcast channel combined with a point-to-point channel, such as 3G or GPRS. We study the problem in the context of a location-based interactive mobile game and introduce a data scheduler for this game, which relies on a wireless broadcasting technology for delivering the data. The proposed scheduler uses the receivers' GPS locations to predict the most probable destination to where each user is headed. Through this information, the scheduler derives the latest point in time, i.e., the threshold, when the data should be transmitted to meet the deadline. We have shown that the threshold is dependent on the number of players in the game, on the number of possible special locations, and on the size of the data that should be transmitted for every location. Simulations for the scheduler have been performed and the experimental results are presented to show how different game parameters influence the in-time delivery of the multimedia content to the players. This paper gives the idea of context-aware applications, which is then explored later under the topic of Smart Space infrastructure and presented in Paper VI of this thesis.

Author's contribution: Paper III is related to the work done in the bot projectin which this interactive application was developed. The author participated in the implementation of the game server. The idea of content scheduling came during the project and was implemented by the author. The author also performed the simulations and wrote the paper.

5.1.4 Paper IV: Framework for Smart Space Application Development

Paper IV presents our approach for rapid and easy development of the Smart Space ubiquitous applications. The approach presented in this paper is based on Nokia's open source Smart-M3 architecture. We have developed ontology-driven user level tools that enable application developers to create innovative Smart Space applications using traditional object-oriented (OO) programming concepts. Our approach consists of the OWL ontology for application description, a developed tool for mapping OWL to OO languages (Python and C) to give complete control over ontologies for application development, and a middleware for encapsulating communication with the Smart Space. The paper also presents a case study implementation of a home automation system to illustrate the functionality of the proposed framework. The results show that the agents communicating implicitly through the Smart-M3 spaces can achieve the interoperability, and

our approach works well for the development of agent-based interoperable applications for the Smart Spaces.
Author's contribution: Paper IV presents the work done in the ICT-SHOCK DIEM project where Åbo Akademi University participated with a large number of ICT companies and research units. The approach was jointly developed by the contributors listed in the paper. The author took part in all the discussions and meetings throughout the development of this framework and case study, and also contributed to writing of the paper.

5.1.5 Paper V: End-users Service Composition in Ubiquitous Computing using Smart Space Approach

In this paper, we express our ideas for introducing the concept of services to the Smart-M3 approach presented in Paper IV.This paper introduces a structured view of the Smart-M3. The paper presents our prototype architecture and overall process for creating end-user service compositions using the Smart Space approach. We have used the OWL-S ontology language to describe the service capabilities semantically. As part of our solution, we implemented a composition algorithm that finds a set of candidate services which could be part of the composition. This composition conforms to semantic graph-based techniques where atomic services are composed iteratively based on OWL-S service properties. The complete realization is obtained by the grounding of the selected services.
Author's contribution: The idea in this paper came during the regular discussions with Professor Johan Lilius. The author was responsible for developing this idea, implementing the proposed algorithm, and the writing of the paper

5.1.6 Paper VI: A Framework for Context-aware Applications for Smart Spaces

Paper VI presents an approach for developing context-aware intelligent applications for a Smart Space-based infrastructure. The goal is to model and process context information using our development tool presented in Paper IV and Nokia's Smart-M3 architecture. We propose an adaptable and scalable context ontology, an ambient computing framework based on Smart Spaces and a rule based reasoning to infer high level context. Our approach deals with key issues in context aware ubiquitous computing such as adaptive and proactive changes in the environment, incorporation of novel sources of context information and automatic code generation from the context ontology to provide seamless interoperability. The paper shows that Smart Spaces are well suited for ambient applications to adapt to the user's preferences because they can provide information about the physical environment

which can be shared and reused by many dynamic applications.
Author's contribution: The approach presented in this paper has its origins in Paper III where the approach of content scheduling based on a player's location was presented. The idea of context-aware applications for Smart Spaces was developed by the author and wrote this paper.

5.1.7 Paper VII: Towards Ontology-driven Development of Ubiquitous and Intelligent interactive TV Applications

Paper VII presents how the ontology driven approach can help to design and develop iTV applications. We demonstrate the suitability of our ontology-driven application development tools and rule based-approach for the development of highly dynamic context-aware iTV applications. We have developed a context-aware service in the domain of interactive TV and evaluated it using our developed ontology-driven tools.
Author's contribution: The idea presented in this paper was initially developed by the author while working in the DIEM Project parallel to his individual research under the supervision of Professor Johan Lilius. The author then applied the tools and architectures developed in the DIEM project to interactive TV domain and wrote this paper.

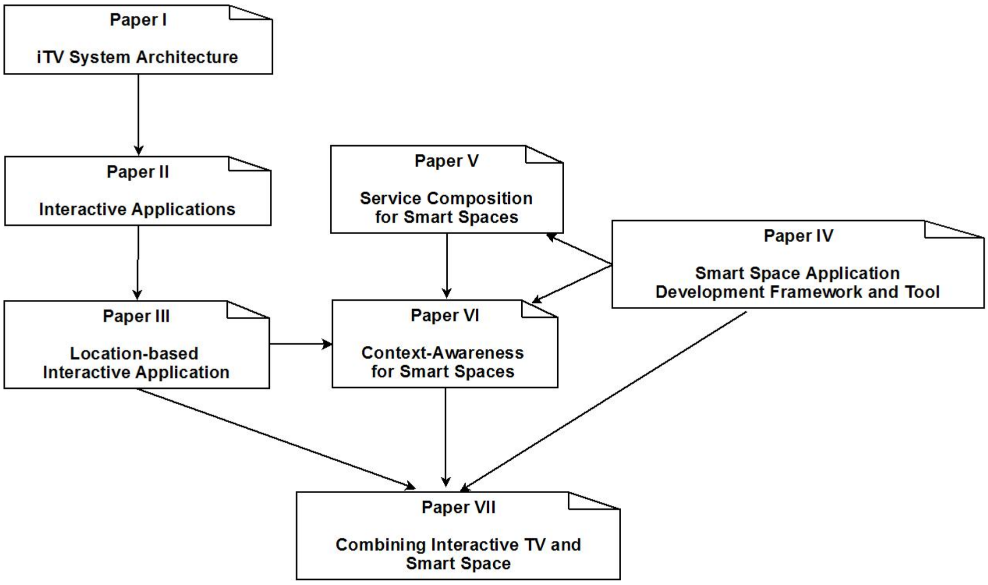

Figure 8: Relationship of Previous Publications

Figure 8 shows the relationship of publications that are part of this thesis. Paper I, Paper II and paper III are the results of research in the area of interactive Mobile TV. They describe proposed iTV system architectures and several interactive applications that could be possible over the Mobile TV. Paper III presents detailed implementation of a particular iTV application. Paper IV, Paper V and Paper VI presents frameworks, approaches and tools developed as a result of research in the area of Smart Spaces. The concept of context-aware applications (Paper VI) was originated from the location-aware application we presented in Paper III. Location is one important concept of context where context can be defined as any information relating to the state of an individual. We further developed this concept of context-aware computing in Paper VI by presenting context ontology and rule-based inference. It discusses not only the location of the user but also other concepts such as time, activity etc. Paper VII is a step towards combining these two research areas, and presents ubiquitous iTV scenario that is developed and evaluated using Smart Space application developed tools and approaches presented in Paper IV, Paper V and paper VI.

6 Conclusions

The vision of Smart Spaces is to enable heterogeneous systems to collaborate with each other by information sharing. Smart Spaces provide storage and access of information about physical environments and this information is shared by inherently dynamic applications using heterogeneous devices. The design of a Smart Space system needs to solve many issues of which interoperability is the most important one. Semantic web technologies such as RDF and ontologies are the key enablers for interoperable Smart Space systems as they provide information interoperability by common understanding of information shared between the other components of the system. OWL ontologies give the expressiveness, flexibility, and extendibility which are required to develop interoperable Smart Space systems. This thesis addresses ontology driven application development for Smart Spaces, which is one step forward in application development for interoperable systems. Designing and dealing with complex ontologies are difficult tasks for software developers as thorough knowledge about the domain and information hierarchy in the applications is required. This thesis presents the solution for this problem by presenting a tool for mapping OWL ontologies to programming languages (Python and C) and hiding the details of complex ontologies from the application developers. This enables rapid and easy interoperable application development for the Smart Spaces.

Smart-M3 can contain information about the environment, which can be published using some description languages such as OWL-S. A promising scenario, where a person joining the smart-M3 would be able to compose an application on-the-fly matching his needs, is addressed in this thesis. The OWL-S specifies Services as Processes, which makes it possible to express more complex services as compositions of services. By storing the service descriptions in the SIB, it becomes possible to query the Smart Space for its Services. Thus, by extending the framework with OWL-S information available in the SIB, we add support for service interaction and composition. The convergence of Smart Spaces and digital TV creates these kinds of scenarios where the viewer of the TV program can request more information about the program from the Smart Space. Information about the TV shows, other metadata related to them and the Electronic Service Guide (ESG) could be stored in Smart Space and the viewer could request them anytime.

The thesis also shows the suitability of Smart Spaces for context-aware ubiquitous applications. The context-aware intelligent applications can be realized by exposing the context information, internal data and functionality of the devices, and ensuring data interoperability between them. This requires modeling of the users and their context in order to enable applications to adapt dynamically to the user's desires. This requirement is due to the variety of devices to be used and the need for interacting with each other within the context. We proposed an adaptable and scalable context ontology, an ambient computing framework based on the Smart Spaces and a rule based reasoning to infer high level context. Our approach deals with the key issues in context aware ubiquitous computing such as adaptive and proactive changes in the environment, incorporation of novel sources of context information, and automatic code generation from the context ontology to provide seamless interoperability. This approach could be applied when the user wants to receive TV programs according to his preferences and his current context. The convergence of the Smart Spaces, semantic web and digital TV could enable a whole new universe of applications. It remains to be seen how long it will take to make these kinds of interoperable systems a reality. This thesis is a contribution in this direction.

6.1 Future Works

Designing interoperable systems and context-aware interactive TV applications using Smart Space is a relatively new and immature research area and there are number of important issues that need to be resolved. At one side, ontology-driven service discovery and composition must be better envisioned so that the system may automatically search and list the components and services that are applicable given a particular situation and application. Since the search space is very big, we need to use the ontology hierarchy to restrict the set of services considered for matching. Moreover, the security, privacy and access control issues must be handled. On the other side, a greater degree of multimedia integration needs to be realized within the systems. Important research question for future work includes; how to use Smart Space development tools when integrating streaming audio and video and other multimedia content related to interactive TV applications. The solution for this research problem requires a way of sharing resources that need greater memory and bandwidth requirements. Future work also includes development and evaluation of more complex application scenarios that could benefit from pervasiveness. Moreover, performance and scalability of the developed approaches and tools will be further evaluated by developing new application scenarios and case studies.

Table 3: Relation of Research Questions and Publications

Research Question	**Answer**	**Publication**
1. What are the limitations of existing digital TV standards to support ubiquitous interactive applications?	Requirement analysis and solutions, Proposed model	PI, PII
2. What infrastructure is needed for interactive TV applications and what are the requirements for building it?	Proposed model	PI, PII
3. What are design issues and problems related to development of a context-aware interactive application using broadcast technology and point-to-point network?	Application concept implementation and evaluation, proposed algorithm and validation	PIII
4. How to achieve interoperability in Smart Space applications and what kind of methodologies and tools are needed to develop Smart Space based multi-domain scenarios?	Tool implementation, Case study implementation for validity of the tool	PIV
5. How to provide automatic service composition is Smart Space for more complex scenarios?	Proposed service composition Algorithm and use-case	PV
6. How to model and process context information in Smart Spaces using ontologies and inference rules?	Development of module for context and rule processing	PVI
7. How to design and develop interactive TV applications based on semantic and social Web data, iTV content and Smart Spaces?	Applicability of our tools to iTV application development	PVII

Bibliography

[1] Home page of DVB project. URL: http://www.dvb.org/.

[2] Digital Video Broadcasting Organization (DVB). DVB-H: IP broadcasting to handheld devices using DVB and mobile telecom networks.

[3] A Social Media platform for location-based stories, http://beta.broadcastr.com/.

[4] Network on Terminal Architecture (NoTA), Available online, http://www.notaworld.org.

[5] BBC Programmes Ontology. Available online: `http://www.bbc.co.uk/ontologies/programmes/2009-09-07.shtml`.

[6] British Broadcasting Corporation: Enhanced TV Formats: http://www.bbc.co.uk/ commissioning/interactive/.

[7] European Broadcasting Union: Multimedia homepage of the European Broadcast Union (EBU), Geneva, Switzerland. http://www.ebu.ch/en/multimedia/index.php.

[8] JSR 272: Mobile broadcast service API for handheld terminals. JSR 272. [Available online], http://www.jcp.org/en/jsr/detail?id=272.

[9] Resource Description Framework (RDF): Concepts and abstract syntax. W3C Recommendation. [Available online], http://www.w3.org/TR/2004/REC-rdf-concepts-20040210/.

[10] Smart-M3 Ontology to Python API Generator:. http://sourceforge.net/projects/smart-m3/files/smart-m3-ontology_to_Python-API_generator_v0.9.1beta.tar.gz/.

[11] Smart-M3 software at sourceforge.net. Release 0.9.4beta, May 2010. [Online]. Available: http://sourceforge.net/projects/smart-m3/.

[12] IP Datacasting - Bringing TV to the mobile phone, white paper, Nokia. 2004.

[13] Paivi Aarreniemi-Jokipelto. Instant messaging in informal learning via interactive television: online communities among children in a "get along" program. *Comput. Entertain.*, 5(2):9, 2007.

[14] BNS Showcases Interactive TV Shopping Application. Prnewswire. 2009.

[15] Interactive TV Applications. Available online from http://www.nextgenitv.com/applications.htm.

[16] Sofia Digital Interactive TV Applications. Available online from http://www.sofiadigital.com/products/all-products/64-interactive-tv-application-suite.

[17] Liliana Ardissono, Alfred Kobsa, and Mark T Maybury. *Personalized Digital Television*, volume 3. Springer, 2004. 13th International Workshop, DSVIS 2006, Dublin, Ireland, July 26-28, 2006.

[18] S. Balandin, I. Oliver, and S. Boldyrev. Distributed architecture of a professional social network on top of M3 Smart Space solution made in PCs and mobile devices friendly manner. In *Mobile Ubiquitous Computing, Systems, Services and Technologies, 2009. UBICOMM '09.*

[19] Sergey Balandin and Heikki Waris. Key properties in the development of Smart Spaces. In *Proceedings of the 5th International on ConferenceUniversal Access in Human-Computer Interaction. Part II: Intelligent and Ubiquitous Interaction Environments*, UAHCI '09, pages 3–12, Berlin, Heidelberg, 2009. Springer-Verlag.

[20] Matthias Baldauf, Schahram Dustdar, and Florian Rosenberg. A Survey on Context-Aware systems. In *International Journal of Ad Hoc and Ubiquitous Computing*, volume 2, 2007.

[21] Vitor Barbosa and Maria Teresa Andrade. Multicao: A semantic approach to context-aware adaptation decision taking. In *WIAMIS*, pages 133–136, 2009.

[22] Nick Bassiliades. Semantic web: Vision and technologies. In *Microsoft Academic Research.* http://academic.research.microsoft.com/Publication/5051009/.

[23] Pieter Bellekens, Kees van der Sluijs, Lora Aroyo, Geert-Jan Houben, and Annelies Kaptein. Semantics-based Framework for Personalized Access to TV Content: Personalized TV Guide Use Case.

[24] Regina Bernhaupt, Marianna Obrist, and Manfred Tscheligi. Usability and usage of iTV services: lessons learned in an Austrian field trial. *Comput. Entertain.*, 5(2), 2007.

[25] Yolanda Blanco-Fernandez, Jose J. Pazos-Arias, Alberto Gil-Solla, Manuel Ramos-Cabrer, Belen Barragans-Martinez, Martin Lopez-Nores, and Jorge Garcia-Duque. Avatar: An advanced multi-agent recommender system of personalized TV contents by semantic reasoning. In *Web Information Systems WISE 2004*.

[26] Susanne Boll, Jens Krsche, and Christian Wegener. Paper chase revisited - a real world game meets hypermedia. In *In Proc. der Intl. Conference on Hypertext (HT03*, 2003.

[27] B. Chandrasekaran, John R. Josephson, and V. Richard Benjamins. What are ontologies, and why do we need them? *IEEE Intelligent Systems*, 14:20–26, January 1999.

[28] Yasmine Charif and Nicolas Sabouret. An overview of semantic web services composition approaches. *Electron Notes Theoretical Computer Science*, 146, January 2006.

[29] Harry Chen, Tim Finin, and Anupam Joshi. An Ontology for context-aware pervasive computing environments. In *Proceedings of the Workshop on Ontologies in Agent Systems*, 2003.

[30] Konstantinos Chorianopoulos. *Conceptual model, user interface design and affective usability evaluation.* PhD thesis, Athens University of Economics and Business, May 2004.

[31] Konstantinos Chorianopoulos and George Lekakos. Learn and play with interactive tv. *Comput. Entertain.*, 5(2):4, 2007.

[32] DMB Comparison of the DAB and DVB-H Systems. available online from `http://www.digitalradiotech.co.uk/dvb-h_dab_dmb.htm`.

[33] D. J. Cook and S. K. Das. Smart environments: Technology, protocols and applications. In *Wiley-Interscience, 2005.*

[34] Pablo Csar and Petri Vuorimaa. A graphical user interface framework for digital television. In *Proceedings of the 10th WSCG International Conference in Central Europe on Computer Graphics, Visualization, and Computer Vision,WSCG*, pages 1–4, Czech Republic, February, 2002. IEEE Computer Society.

[35] Michael J. Darnell. How do people really interact with TV? naturalistic observations of digital TV and digital video recorder users. *Comput. Entertain.*, 5(2), 2007.

[36] Oleg Davidyuk, Nikolaos Georgantas, Valrie Issarny, and Jukka Riekki. Medusa: Middleware for end-user composition of ubiquitous

applications. In *Handbook of Research on Ambient Intelligence and Smart Environments: Trends and Perspectives*, 2009.

[37] Mike Dean and Guus Schreiber. OWL web ontology language reference. W3C recommendation, feb 2004. http://www.w3.org/TR/2004/REC-owl-ref-20040210/.

[38] Devices and Interoperability Ecosystem. Tivit project, available online from `http://www.tivit.fi/en/devices`,.

[39] Anind K. Dey and Gregory D. Abowd. The Context Toolkit: Aiding the development of context-aware applications. In *Workshop on Software Eng. for Wearable and Pervasive Computing*, 2000.

[40] Alan Dix, Tom Rodden, Nigel Davies, Jonathan Trevor, Adrian Friday, and Kevin Palfreyman. Exploiting space and location as a design framework for interactive mobile systems. In *In ACM Transactions on Computer-Human Interaction (TOCHI)*.

[41] Eclipse IDE for C/C++ Developers. available online from `http://www.eclipse.org`,.

[42] The Finnish Funding Agency for Technology and Innovation. Tekes, `http://www.tekes.fi`,.

[43] Kary Främling, Ian Oliver, Jukka Honkola, and Jan Nyman. Smart spaces for ubiquitously smart buildings. In *Proceedings of the 2009 Third International Conference on Mobile Ubiquitous Computing, Systems, Services and Technologies*, UBICOMM '09, pages 295–300. IEEE Computer Society, 2009.

[44] A. Gil, F. Fraile, M. Ramos, I. Fez, and J. Guerri. Personalized multimedia touristic services for hybrid broadcast/broadband mobile receivers. *Consumer Electronics, IEEE Transactions on*, 56(1):211 –219, february 2010.

[45] Philip D. Gray and Daniel Salber. Modelling and using sensed context information in the design of interactive applications. In *Proceedings of the 8th IFIP International Conference on Engineering for Human-Computer Interaction*, London, UK, 2001. Springer-Verlag.

[46] Tao Gu, Hung Keng Pung, and Da Qing Zhang. A middleware for building context-aware mobile services. In *Proceedings of IEEE Vehicular Technology Conference*, 2004.

[47] Digital Video Broadcasting Handheld. available online from `http://www.dvb-h.org/technology.htm`.

[48] Yun Her, Su-Kyoung Kim, and YoungTaek Jin. A context-aware framework using ontology for smart phone platform. In *International Journal of Digital Content Technology and its Applications, Volume 4, Number 5*, 2010.

[49] Jukka Honkola, Hannu Laine, Ronald Brown, and Ian Oliver. Cross-domain interoperability: A case study. In *Smart Spaces and Next Generation Wired/Wireless Networking*, Lecture Notes in Computer Science, pages 22–31. Springer Berlin / Heidelberg, 2009.

[50] F. Hopfgartner and J.M. Jose. Semantic user profiling techniques for personalised multimedia recommendation. *Multimedia Systems*, 16(4-5):255–274, 2010.

[51] Anne-Katrin Hubel, Johannes Theilmann, and Ulrich Theilmann. I just want to see the news: Interactivity in mobile environments. In *Proceedings of the 5th EuroITV Conference*, volume 4471. Springer, 2007.

[52] Integrated Services Digital Broadcasting Terrestrial (ISDB-T). available online from `http://www.dibeg.org/`.

[53] Jens F. Jensen. Interactive television: new genres, new format, new content. In *IE2005: Proceedings of the second Australasian conference on Interactive entertainment*, pages 89–96, Sydney, Australia, Australia, 2005. Creativity and Cognition Studios Press.

[54] A. Katasonov and M. Palviainen. Towards ontology-driven development of applications for smart environments. In *Proceedings of 8th IEEE International Conference on Pervasive Computing and Communications Workshops (PERCOM Workshops)*, April 2010.

[55] Artem Katasonov. Enabling non-programmers to develop smart environment applications. In *Proceedings of the The IEEE symposium on Computers and Communications*. IEEE Computer Society, 2010.

[56] Andre Kaustell, M. Mohsin Saleemi, Thomas Rosqvist, Juuso Jokiniemi, Johan Lilius, and Ivan Porres. Framework for Smart Space Application Development. In *Proceedings of the International Workshop on Semantic Interoperability, IWSI*, 2011.

[57] Spiro Kiousis. Interactivity: a concept explication. *New Media and Society*, 4(3), 2002.

[58] Palle Klante, Jens Kroesche, and Susanne C. J. Boll. Evaluating a mobile location-based multimodal game for first-year students. In *In proceedings of the SPIE 2005.*

[59] Laszlo Kovacs, Peter Matetelki, and Balazs Pataki. Service-oriented context-aware framework. In *Young Researchers Workshop on Service-Oriented Computing*, 2009.

[60] Amitabh Kumar. *Implementing Mobile TV.* Focal Press, 2010.

[61] L. Lamorte, C. A. Licciardi, M. Marengo, A. Salmeri, P. Mohr, G. Raffa, L. Roffia, M. Pettinari, and Salmon T. Cinotti. A platform for enabling context aware telecommunication services. *Third Workshop on Context Awareness for Proactive Systems*, 2007.

[62] Kwangkee Lee, Jungho Lee, and Hyogun Lee. A middleware architecture for itv receivers. In *Consumer Electronics, 2002. ICCE. 2002 Digest of Technical Papers. International Conference on*, pages 88–89, 2002.

[63] Grace A. Lewis, Edwin Morris, Soumya Simanta, and Lutz Wrage. Why standards are not enough to guarantee end-to-end interoperability. In *Proceedings of the Seventh International Conference on Composition-Based Software Systems (ICCBSS 2008)*. IEEE Computer Society, 2008.

[64] Daniele Manzaroli. *Architectures for Context Aware Services in Smart Environments.* PhD thesis, University of Bologna, 2009.

[65] Ryusuke Masuoka, Yannis Labrou, Bijan Parsia, and Evren Sirin. Ontology-enabled pervasive computing applications. *IEEE Intelligent Systems*, 18, September 2003.

[66] Ryusuke Masuoka, Bijan Parsia, and Yannis Labrou. Task computing - the semantic web meets pervasive computing. In *In Proceedings of 2nd International Semantic Web Conference (ISWC2003), Sanibel Island*, pages 866–881, 2003.

[67] Samuel Miller. Taking on the masses with mobile messaging tv. *Comput. Entertain.*, 3(2):6–6, 2005.

[68] Sonia Mokhtar, Nikolaos Georgantas, and Valrie Issarny. Ad hoc composition of user tasks in pervasive computing environments. In Thomas Gschwind, Uwe Amann, and Oscar Nierstrasz, editors, *Software Composition*, Lecture Notes in Computer Science, pages 31–46. Springer Berlin / Heidelberg, 2005.

[69] Nashid Nabian, Giusy Di Lorenzo, Francesco Calabrese, and Carlo Ratti. Context-based experience of the city via broadcasted content retrieval. In *Proceedings of the 5th International ICST Mobile Multimedia Communications Conference*, Mobimedia '09, 2009.

[70] S. Nakahara, M. Okano, M. Takada, and T. Kuroda. Digital transmission scheme for isdb-t and reception characteristics of digital terrestrial television broadcasting system in japan. In *Consumer Electronics, 1999. ICCE. International Conference on*, 1999.

[71] Daniela Nicklas, Christoph Pfisterer, and Bernhard Mitschang. Towards Location-based Games. In *Proceedings of the International Conference on Applications and Development of Computer Games in the 21st Century: ADCOG 21; China, November 22-23 2001*, 2001.

[72] NF Noy, M. Sintek, S. Decker, M. Crubezy, RW Fergerson, and MA Musen. Creating semantic web contents with protege-2000. *IEEE Intelligent Systems*, 16(2):60–71, 2001.

[73] Documentation of the Unified Modeling Language (UML). Available from the object management group (OMG). `http://www.omg.org`.

[74] Ian Oliver and Jukka Honkola. Personal semantic web through a space based computing environment. In *Proceedings of the 2nd Interntional Conference on Semantic Computing*, 2008.

[75] Ian Oliver, Esko Nuutila, and Seppo Torma. Context gathering in meetings: Business processes meet the agents and the semantic web. In *Proceedings of the The 4th International Workshop on Technologies for Context-Aware Business Process Management (TCoB 2009)*, 2009.

[76] T. Paila. Mobile internet over ip data broadcast. In *10th International Conference on Telecommunications, 2003. ICT 2003.*, volume 1, pages 19 – 24 vol.1.

[77] Antonis Papadimitriou, Christos Anagnostopoulos, Vassileios Tsetsos, Sarantis Paskalis, and Stathes Hadjiefthymiades. A semantics-aware platform for interactive tv services. In *In the Proceedings of the 1st International Conference on New Media Technology (I-MEDIA 07) Graz Austria*, 2007.

[78] Julien Pauty, Davy Preuveneers, Peter Rigole, and Yolande Berbers. Research challenges in mobile and context-aware service development. In *Proceedings of the Workshop on Research Challenges in Mobile and Context-Aware Service Development*, 2006.

[79] FutureTV-2011: Making Television Personal and Social. 2nd international workshop on the future of television.

[80] Stefan Poslad. *Ubiquitous Computing: Smart Devices, Environments and Interactions.* Wiley, 2009.

[81] Ptolemy project heterogeneous modeling and design. Available online from http://ptolemy.eecs.berkeley.edu,.

[82] Anand Ranganathan and Roy H. Campbell. A middleware for context-aware agents in ubiquitous computing environments. In *Proceedings of the ACM/IFIP/USENIX 2003 International Conference on Middleware*, Middleware '03, pages 143–161. Springer-Verlag New York, Inc., 2003.

[83] Reza Razavi. Dynamic macroprogramming of wireless sensor networks with mobile agents. In *In 2nd Workshop on Artificial Intelligence Techniques for Ambient Intelligence*, pages 43–55, 2007.

[84] P. Remagnino and G.L. Foresti. Ambient intelligence: A new multidisciplinary paradigm. *Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Transactions on*, 35(1):1 – 6, 2005.

[85] Marta Rey-Lopez, Rebeca P. Diaz-Redondo, Ana Fernandez-Vilas, and Jose J. Pazos-Arias. Entercation: engaging viewers in education through tv. *Comput. Entertain.*, 5(2):7, 2007.

[86] Martin Richartz. The future of interactive mobile broadcast. In *Vodafone GROUP R and D Germany.*

[87] M. Roman, C. Hess, R. Cerqueira, and A. Ranganathan. A middleware infrastructure for active spaces. In *IEEE Pervasive Computing*, 2002.

[88] M.M. Saleemi and J. Lilius. Interactive applications for mobile tv. In *Multimedia Computing and Information Technology (MCIT), 2010 International Conference on*, pages 37 –40, march 2010.

[89] Mohsin M. Saleemi, Jerker Björkqvist, and Johan Lilius. Software platform for mobile tv. 2007. Workshop on Interactive Applications for Mobile TV, In association with the 5th European Interactive TV Conference - EuroITV 2007, Amsterdam, May 23-25, 2007.

[90] Cinotti Tullio Salmon, Bartolini Sara, D. Alfredo, Roffia Luca, Spadini Federico, Trevisan Riccardo, Vergari Fabio, and Zamagni Guido. Approaching the design of interoperable smart environments applications. In *NoTA Conference*, 2009.

[91] Cinotti Tullio Salmon, Bartolini Sara, Delia Alfredo, Roffia Luca, Spadini Federico, Trevisan Riccardo, Vergari Fabio, and Zamagni Guido. Approaching the design of interoperable smart environments applications. In *NoTA 2009 Conference.*

[92] Raimund Schatz, Norbert Jordan, and Siegfried Wagner. Beyond broadcast a hybrid testbed for Mobile TV 2.0 services. In *In Proc. ICN07*, 2007.

[93] B. Simon, Z. Miklos, W. Nejdl, M. Sintek, and J. Salvachua. Smart space for learning: A mediation infrastructure for learning services. In *In Proceedings of 12th International Conference on World Wide Web*, 2003.

[94] Alexander Smirnov, Alexey Kashevnik, Nikolay Shilov, Ian Oliver, Sergey Balandin, and Sergey Boldyrev. Anonymous agent coordination in smart spaces: State-of-the-art. In *Proceedings of the 9th International Conference on Smart Spaces and Next Generation Wired/Wireless Networking and Second Conference on Smart Spaces*, NEW2AN '09 and ruSMART '09, pages 42–51. Springer-Verlag, 2009.

[95] Z. Song, Y. Labrou, and R. Masuoka. Dynamic service discovery and management in task computing. In *proceedings of The First Annual International Conference on Mobile and Ubiquitous Systems: Networking and Services, MOBIQUITOUS*, 2004.

[96] Nicholas Taylor. Self-improving personal smart spaces. *PerAda Magazine*, 2009.

[97] BBC News Technology. available online from `http://www.bbc.co.uk/news/10569081`.

[98] Sabine Thieme, Ansgar Scherp, Melanie Albrecht, and Susanne Boll. Media gallery tv: view and shop your photos on interactive digital television. In *MULTIMEDIA '05: Proceedings of the 13th annual ACM international conference on Multimedia*, pages 269–270. ACM Press, 2005.

[99] Jan Van Den Bergh, Bert Bruynooghe, Jan Moons, Steven Huypens, Koen Handekyn, and Karin Coninx. Model-driven creation of staged participatory multimedia events on tv. In *Proceedings of the 5th European conference on Interactive TV: a shared experience*, Berlin, Heidelberg, 2007. Springer-Verlag.

[100] F. Vergari, S. Bartolini, F. Spadini, A. D'Elia, G. Zamagni, L. Roffia, and T.S. Cinotti. A smart space application to dynamically relate medical and environmental information. In *Design, Automation Test in Europe Conference Exhibition (DATE), 2010*, march 2010.

[101] Peter Vorderer, Silvia Knobloch, and Holger Schramm. Does entertainment suffer from interactivity? The impact of watching an inter-

active TV movie on viewers' experience of entertainment. *Comput. Entertain.*, 3(4):343–363, 2001.

[102] David W. Walker. Automated web service composition using semantic web technologies. In *Proceedings of the First International Conference on Autonomic Computing*, pages 306–307. IEEE Computer Society, 2004.

[103] Mark Weiser. The computer for the 21st century. In *Scientific American*, 1991.

[104] Paul Wisner and Dimitris N. Kalofons. A framework for end-user programming of smart homes using mobile devices. In *Proceedings of the Consumer Communications and Networking Conference, CCNC*, 2007.

[105] X.H.Wang, D.Q. Zhang, T. Gu, and H.K. Pung. Ontology based context modeling and reasoning using OWL. In *Workshop Proceedings of the 2nd IEEE Conference on Pervasive Computing and Communications*, 2004.

[106] Watchitoo Embed your Interactive Live Event. available online from `http://watchitoo.com/`.

Turku Centre for Computer Science
TUCS Dissertations

1. **Marjo Lipponen**, On Primitive Solutions of the Post Correspondence Problem
2. **Timo Käkölä**, Dual Information Systems in Hyperknowledge Organizations
3. **Ville Leppänen**, Studies on the Realization of PRAM
4. **Cunsheng Ding**, Cryptographic Counter Generators
5. **Sami Viitanen**, Some New Global Optimization Algorithms
6. **Tapio Salakoski**, Representative Classification of Protein Structures
7. **Thomas Långbacka**, An Interactive Environment Supporting the Development of Formally Correct Programs
8. **Thomas Finne**, A Decision Support System for Improving Information Security
9. **Valeria Mihalache**, Cooperation, Communication, Control. Investigations on Grammar Systems.
10. **Marina Waldén**, Formal Reasoning About Distributed Algorithms
11. **Tero Laihonen**, Estimates on the Covering Radius When the Dual Distance is Known
12. **Lucian Ilie**, Decision Problems on Orders of Words
13. **Jukkapekka Hekanaho**, An Evolutionary Approach to Concept Learning
14. **Jouni Järvinen**, Knowledge Representation and Rough Sets
15. **Tomi Pasanen**, In-Place Algorithms for Sorting Problems
16. **Mika Johnsson**, Operational and Tactical Level Optimization in Printed Circuit Board Assembly
17. **Mats Aspnäs**, Multiprocessor Architecture and Programming: The Hathi-2 System
18. **Anna Mikhajlova**, Ensuring Correctness of Object and Component Systems
19. **Vesa Torvinen**, Construction and Evaluation of the Labour Game Method
20. **Jorma Boberg**, Cluster Analysis. A Mathematical Approach with Applications to Protein Structures
21. **Leonid Mikhajlov**, Software Reuse Mechanisms and Techniques: Safety Versus Flexibility
22. **Timo Kaukoranta**, Iterative and Hierarchical Methods for Codebook Generation in Vector Quantization
23. **Gábor Magyar**, On Solution Approaches for Some Industrially Motivated Combinatorial Optimization Problems
24. **Linas Laibinis**, Mechanised Formal Reasoning About Modular Programs
25. **Shuhua Liu**, Improving Executive Support in Strategic Scanning with Software Agent Systems
26. **Jaakko Järvi**, New Techniques in Generic Programming – C++ is more Intentional than Intended
27. **Jan-Christian Lehtinen**, Reproducing Kernel Splines in the Analysis of Medical Data
28. **Martin Büchi**, Safe Language Mechanisms for Modularization and Concurrency
29. **Elena Troubitsyna**, Stepwise Development of Dependable Systems
30. **Janne Näppi**, Computer-Assisted Diagnosis of Breast Calcifications
31. **Jianming Liang**, Dynamic Chest Images Analysis
32. **Tiberiu Seceleanu**, Systematic Design of Synchronous Digital Circuits
33. **Tero Aittokallio**, Characterization and Modelling of the Cardiorespiratory System in Sleep-Disordered Breathing
34. **Ivan Porres**, Modeling and Analyzing Software Behavior in UML
35. **Mauno Rönkkö**, Stepwise Development of Hybrid Systems
36. **Jouni Smed**, Production Planning in Printed Circuit Board Assembly
37. **Vesa Halava**, The Post Correspondence Problem for Market Morphisms
38. **Ion Petre**, Commutation Problems on Sets of Words and Formal Power Series
39. **Vladimir Kvassov**, Information Technology and the Productivity of Managerial Work
40. **Frank Tétard**, Managers, Fragmentation of Working Time, and Information Systems

41. **Jan Manuch**, Defect Theorems and Infinite Words
42. **Kalle Ranto**, Z_4-Goethals Codes, Decoding and Designs
43. **Arto Lepistö**, On Relations Between Local and Global Periodicity
44. **Mika Hirvensalo**, Studies on Boolean Functions Related to Quantum Computing
45. **Pentti Virtanen**, Measuring and Improving Component-Based Software Development
46. **Adekunle Okunoye**, Knowledge Management and Global Diversity – A Framework to Support Organisations in Developing Countries
47. **Antonina Kloptchenko**, Text Mining Based on the Prototype Matching Method
48. **Juha Kivijärvi**, Optimization Methods for Clustering
49. **Rimvydas Rukšėnas**, Formal Development of Concurrent Components
50. **Dirk Nowotka**, Periodicity and Unbordered Factors of Words
51. **Attila Gyenesei**, Discovering Frequent Fuzzy Patterns in Relations of Quantitative Attributes
52. **Petteri Kaitovaara**, Packaging of IT Services – Conceptual and Empirical Studies
53. **Petri Rosendahl**, Niho Type Cross-Correlation Functions and Related Equations
54. **Péter Majlender**, A Normative Approach to Possibility Theory and Soft Decision Support
55. **Seppo Virtanen**, A Framework for Rapid Design and Evaluation of Protocol Processors
56. **Tomas Eklund**, The Self-Organizing Map in Financial Benchmarking
57. **Mikael Collan**, Giga-Investments: Modelling the Valuation of Very Large Industrial Real Investments
58. **Dag Björklund**, A Kernel Language for Unified Code Synthesis
59. **Shengnan Han**, Understanding User Adoption of Mobile Technology: Focusing on Physicians in Finland
60. **Irina Georgescu**, Rational Choice and Revealed Preference: A Fuzzy Approach
61. **Ping Yan**, Limit Cycles for Generalized Liénard-Type and Lotka-Volterra Systems
62. **Joonas Lehtinen**, Coding of Wavelet-Transformed Images
63. **Tommi Meskanen,** On the NTRU Cryptosystem
64. **Saeed Salehi**, Varieties of Tree Languages
65. **Jukka Arvo**, Efficient Algorithms for Hardware-Accelerated Shadow Computation
66. **Mika Hirvikorpi**, On the Tactical Level Production Planning in Flexible Manufacturing Systems
67. **Adrian Costea**, Computational Intelligence Methods for Quantitative Data Mining
68. **Cristina Seceleanu**, A Methodology for Constructing Correct Reactive Systems
69. **Luigia Petre**, Modeling with Action Systems
70. **Lu Yan**, Systematic Design of Ubiquitous Systems
71. **Mehran Gomari**, On the Generalization Ability of Bayesian Neural Networks
72. **Ville Harkke**, Knowledge Freedom for Medical Professionals – An Evaluation Study of a Mobile Information System for Physicians in Finland
73. **Marius Cosmin Codrea**, Pattern Analysis of Chlorophyll Fluorescence Signals
74. **Aiying Rong**, Cogeneration Planning Under the Deregulated Power Market and Emissions Trading Scheme
75. **Chihab BenMoussa**, Supporting the Sales Force through Mobile Information and Communication Technologies: Focusing on the Pharmaceutical Sales Force
76. **Jussi Salmi**, Improving Data Analysis in Proteomics
77. **Orieta Celiku**, Mechanized Reasoning for Dually-Nondeterministic and Probabilistic Programs
78. **Kaj-Mikael Björk**, Supply Chain Efficiency with Some Forest Industry Improvements
79. **Viorel Preoteasa**, Program Variables – The Core of Mechanical Reasoning about Imperative Programs
80. **Jonne Poikonen**, Absolute Value Extraction and Order Statistic Filtering for a Mixed-Mode Array Image Processor
81. **Luka Milovanov**, Agile Software Development in an Academic Environment
82. **Francisco Augusto Alcaraz Garcia**, Real Options, Default Risk and Soft Applications
83. **Kai K. Kimppa**, Problems with the Justification of Intellectual Property Rights in Relation to Software and Other Digitally Distributable Media
84. **Dragoş Truşcan**, Model Driven Development of Programmable Architectures
85. **Eugen Czeizler**, The Inverse Neighborhood Problem and Applications of Welch Sets in Automata Theory

86. **Sanna Ranto**, Identifying and Locating-Dominating Codes in Binary Hamming Spaces
87. **Tuomas Hakkarainen**, On the Computation of the Class Numbers of Real Abelian Fields
88. **Elena Czeizler**, Intricacies of Word Equations
89. **Marcus Alanen**, A Metamodeling Framework for Software Engineering
90. **Filip Ginter**, Towards Information Extraction in the Biomedical Domain: Methods and Resources
91. **Jarkko Paavola**, Signature Ensembles and Receiver Structures for Oversaturated Synchronous DS-CDMA Systems
92. **Arho Virkki**, The Human Respiratory System: Modelling, Analysis and Control
93. **Olli Luoma**, Efficient Methods for Storing and Querying XML Data with Relational Databases
94. **Dubravka Ilić**, Formal Reasoning about Dependability in Model-Driven Development
95. **Kim Solin**, Abstract Algebra of Program Refinement
96. **Tomi Westerlund**, Time Aware Modelling and Analysis of Systems-on-Chip
97. **Kalle Saari**, On the Frequency and Periodicity of Infinite Words
98. **Tomi Kärki**, Similarity Relations on Words: Relational Codes and Periods
99. **Markus M. Mäkelä**, Essays on Software Product Development: A Strategic Management Viewpoint
100. **Roope Vehkalahti**, Class Field Theoretic Methods in the Design of Lattice Signal Constellations
101. **Anne-Maria Ernvall-Hytönen**, On Short Exponential Sums Involving Fourier Coefficients of Holomorphic Cusp Forms
102. **Chang Li**, Parallelism and Complexity in Gene Assembly
103. **Tapio Pahikkala**, New Kernel Functions and Learning Methods for Text and Data Mining
104. **Denis Shestakov**, Search Interfaces on the Web: Querying and Characterizing
105. **Sampo Pyysalo**, A Dependency Parsing Approach to Biomedical Text Mining
106. **Anna Sell**, Mobile Digital Calendars in Knowledge Work
107. **Dorina Marghescu**, Evaluating Multidimensional Visualization Techniques in Data Mining Tasks
108. **Tero Säntti**, A Co-Processor Approach for Efficient Java Execution in Embedded Systems
109. **Kari Salonen**, Setup Optimization in High-Mix Surface Mount PCB Assembly
110. **Pontus Boström**, Formal Design and Verification of Systems Using Domain-Specific Languages
111. **Camilla J. Hollanti**, Order-Theoretic Mehtods for Space-Time Coding: Symmetric and Asymmetric Designs
112. **Heidi Himmanen**, On Transmission System Design for Wireless Broadcasting
113. **Sébastien Lafond**, Simulation of Embedded Systems for Energy Consumption Estimation
114. **Evgeni Tsivtsivadze**, Learning Preferences with Kernel-Based Methods
115. **Petri Salmela**, On Commutation and Conjugacy of Rational Languages and the Fixed Point Method
116. **Siamak Taati**, Conservation Laws in Cellular Automata
117. **Vladimir Rogojin**, Gene Assembly in Stichotrichous Ciliates: Elementary Operations, Parallelism and Computation
118. **Alexey Dudkov**, Chip and Signature Interleaving in DS CDMA Systems
119. **Janne Savela**, Role of Selected Spectral Attributes in the Perception of Synthetic Vowels
120. **Kristian Nybom**, Low-Density Parity-Check Codes for Wireless Datacast Networks
121. **Johanna Tuominen**, Formal Power Analysis of Systems-on-Chip
122. **Teijo Lehtonen**, On Fault Tolerance Methods for Networks-on-Chip
123. **Eeva Suvitie**, On Inner Products Involving Holomorphic Cusp Forms and Maass Forms
124. **Linda Mannila**, Teaching Mathematics and Programming – New Approaches with Empirical Evaluation
125. **Hanna Suominen**, Machine Learning and Clinical Text: Supporting Health Information Flow
126. **Tuomo Saarni**, Segmental Durations of Speech
127. **Johannes Eriksson**, Tool-Supported Invariant-Based Programming

128. **Tero Jokela**, Design and Analysis of Forward Error Control Coding and Signaling for Guaranteeing QoS in Wireless Broadcast Systems
129. **Ville Lukkarila**, On Undecidable Dynamical Properties of Reversible One-Dimensional Cellular Automata
130. **Qaisar Ahmad Malik**, Combining Model-Based Testing and Stepwise Formal Development
131. **Mikko-Jussi Laakso**, Promoting Programming Learning: Engagement, Automatic Assessment with Immediate Feedback in Visualizations
132. **Riikka Vuokko**, A Practice Perspective on Organizational Implementation of Information Technology
133. **Jeanette Heidenberg**, Towards Increased Productivity and Quality in Software Development Using Agile, Lean and Collaborative Approaches
134. **Yong Liu**, Solving the Puzzle of Mobile Learning Adoption
135. **Stina Ojala**, Towards an Integrative Information Society: Studies on Individuality in Speech and Sign
136. **Matteo Brunelli**, Some Advances in Mathematical Models for Preference Relations
137. **Ville Junnila**, On Identifying and Locating-Dominating Codes
138. **Andrzej Mizera**, Methods for Construction and Analysis of Computational Models in Systems Biology. Applications to the Modelling of the Heat Shock Response and the Self-Assembly of Intermediate Filaments.
139. **Csaba Ráduly-Baka**, Algorithmic Solutions for Combinatorial Problems in Resource Management of Manufacturing Environments
140. **Jari Kyngäs**, Solving Challenging Real-World Scheduling Problems
141. **Arho Suominen**, Notes on Emerging Technologies
142. **József Mezei**, A Quantitative View on Fuzzy Numbers
143. **Marta Olszewska**, On the Impact of Rigorous Approaches on the Quality of Development
144. **Antti Airola**, Kernel-Based Ranking: Methods for Learning and Performace Estimation
145. **Aleksi Saarela**, Word Equations and Related Topics: Independence, Decidability and Characterizations
146. **Lasse Bergroth**, Kahden merkkijonon pisimmän yhteisen alijonon ongelma ja sen ratkaiseminen
147. **Thomas Canhao Xu**, Hardware/Software Co-Design for Multicore Architectures
148. **Tuomas Mäkilä**, Software Development Process Modeling – Developers Perspective to Contemporary Modeling Techniques
149. **Shahrokh Nikou**, Opening the Black-Box of IT Artifacts: Looking into Mobile Service Characteristics and Individual Perception
150. **Alessandro Buoni**, Fraud Detection in the Banking Sector: A Multi-Agent Approach
151. **Mats Neovius**, Trustworthy Context Dependency in Ubiquitous Systems
152. **Fredrik Degerlund**, Scheduling of Guarded Command Based Models
153. **Amir-Mohammad Rahmani-Sane**, Exploration and Design of Power-Efficient Networked Many-Core Systems
154. **Ville Rantala**, On Dynamic Monitoring Methods for Networks-on-Chip
155. **Mikko Pelto**, On Identifying and Locating-Dominating Codes in the Infinite King Grid
156. **Anton Tarasyuk**, Formal Development and Quantitative Verification of Dependable Systems
157. **Muhammad Mohsin Saleemi**, Towards Combining Interactive Mobile TV and Smart Spaces: Architectures, Tools and Application Development

Turku Centre *for* Computer Science

Joukahaisenkatu 3-5 B, 20520 Turku, Finland | www. tucs.fi

University of Turku
Faculty of Mathematics and Natural Sciences
- Department of Information Technology
- Department of Mathematics and Statistics

Turku School of Economics
- Institute of Information Systems Science

Åbo Akademi University
Division for Natural Sciences and Technology
- Department of Information Technologies

ISBN 978-952-12-2839-1
ISSN 1239-1883

Towards Combining Interactive Mobile TV and Smart Spaces

Muhammad Mohsin Saleemi

Towards Combining Interactive Mobile TV and Smart Spaces