CHALMERS

# Technology Evaluation of future IPTV Services

*Master of Science Thesis in the Programme Computer Science and Engineering*

PETTER A. MAGNUSSON

Department of Computer Science and Engineering
CHALMERS UNIVERSITY OF TECHNOLOGY
UNIVERSITY OF GOTHENBURG
Göteborg, Sweden,  May 2009

Technology Evaluation of future IPTV Services

PETTER A. MAGNUSSON

Examiner: STAFFAN BJÖRK

Department of Computer Science and Engineering
Chalmers University of Technology
SE-412 96 Göteborg
Sweden
Telephone + 46 (0)31-772 1000

Work done in cooperation with:
Ericsson AB
Kista, Stockholm, Sweden May 2009

Department of Computer Science and Engineering
Göteborg, Sweden May 2009

# Abstract

The next home entertainment platform to receive an upgrade is the television. As one of the most important devices in the home today the struggle to provide TVs successor continues. On its journey through the modern world the Internet Protocol Television (IPTV) provides new technology, cheaper transmissions and a better TV experience.

Telefonaktiebolaget L. M. Ericsson, being a major part of the telecom industry provides underlying technology for IPTV as well as end-user solutions via service providers. Ericsson is looking towards the future in an effort to improve IPTV as a platform for end-users.

This thesis aims to evaluate Ericsson's current IPTV solution towards the new services that the platform will provide.

The thesis sets out to find a new service designed to take advantage of the technology of the platform and meet the needs of the customers with interactivity and personalization. The resulting service then serves as a base for two prototypes implemented using technology provided by Ericsson in an attempt to evaluate the platform as well as to demonstrate requirements for how future services should be designed.

From the implementation of the prototypes the thesis reveals potential issues with the current platform in regard to future services. Such problems include the complexity in how more advanced services are developed and the need for in-house knowledge to create them. Ways to solve the problems on different levels are presented, some that adds additional value to the platform in the process.

The thesis also presents a way for how to create new services for IPTV called Content Interactive Widgets and how they affect the requirements on the technology.

## Abstrakt

Nästa del i det moderna hemmet som kommer att uppgraderas är vår TV. TVn är nog en av de absolut viktigaste elektroniska apparaterna i dagens hem och utvecklarna arbetar febrilt med att ta fram dess ersättare. Ett steg på vägen är Internet Protocol Television (IPTV) som erbjuder ny teknologi, billigare sändningar och en bättre TV-upplevelse.

I sin roll i den allt större telekom-industrin är Telefonaktiebolaget L. M. Ericsson med och tillhandahåller teknologin som gör IPTV till en verklighet och erbjuder dessutom kundnära lösningar. Med detta ser Ericsson mot framtiden och arbetar för att förbättra den plattform som IPTV använder.

Detta examensarbete är utformat för att utvärdera Ericssons nuvarande IPTV-plattform utefter dess förmåga att hantera framtidens tjänster.

Projektet ger sig ut för att hitta och designa en ny tjänst som är utformad att dra nytta av allt som IPTV har att erbjuda och samtidigt uppfylla nya krav på interaktivitet och personlig anpassning. Den framtagna tjänsten används senare som bas för två prototyper som utvecklas med Ericssons nuvarande teknik för att på så sätt kunna utvärdera hur bra den stödjer nya idéer och tjänster. Prototyperna fungerar samtidigt som exempel på hur man bör utforma nya tjänster för IPTV.

Genom utvecklingen av prototyperna avslöjar projektet en del problem och begränsningar i den nuvarande plattformen med avseende på nya tjänster. Dessa problem innefattar bland annat komplexiteten i hur nya mer avancerade tjänster måste utvecklas och därmed behovet av intern kompetens. Lösningar på problemen presenteras för olika nivåer av tidsinvestering. En del förslag innebär även en generell förbättring av plattformens utvecklingspotential.

Examensarbetet presenterar dessutom en modell för att skapa nya tjänster utformade för IPTV kallat Content Interactive Widgets och förklarar hur de i sin tur påverkar kraven på teknologin.

# Acknowledgements

I would like to thank Ola Andersson for his incredible support as the thesis supervisor and for allowing me to do this thesis in the first place. It has been a great experience from the start to the end. Additionally I would like to thank Hans Byström and the entire Converged TV team for all the assistance and support that I got even during though times of tight deadlines and presentations.

I would also like to thank fellow student and colleague Ezequiel Lisak for all that I learned during our cooperative work and for being a friend during the time in Kista.

Finally I'd like to thank my mother for helping me find thesis projects, without you I never would have found the thesis to begin with.

## Note to the reader

The reader is expected to have some basic knowledge on IPTV as the report focuses on improvements on the platform. This report is aimed towards a broad audience with main focus towards project leaders and managers. The work includes parts from both interaction design and software development and both aspects will be described in a way to give an understanding of the motivations rather than to go into full detail.

The report is written to introduce the setting and the research question followed by the process in three stages. The experiences from the process are then presented in the Discussion chapter together with possible solutions and recommendations.

Those primarily interested in the results are recommended to read the Introduction (1), Discussion (6) and Conclusions (7) chapters. The information about the process, the target service found and the two prototypes made can be found in their respective chapters.

## Terminology

| | |
|---|---|
| IPTV | Internet Protocol Television (Interactive Personalized Television in some cases) |
| IP | Internet Protocol |
| STB | Set-top Box, Hardware usually used for services together with a TV. |
| SVG | Scalable Vector Graphics |
| Front-end | The part of the software that directly interacts with the user. |
| Back-end | The server part of the software, usually processing and computation. |
| GUI | Graphical user interface |
| PiP | Picture-in-picture, displaying one video on top of another video while both are visible. |
| HD | High Definition, Higher resolution video (compared to Standard Definition) |

## Contents

# 1 Introduction

Internet Protocol TV or IPTV [1] is the next step of television medium. It is a general term describing the use of IP-technology to send television information. In difference to regular broadcast TV it also allows for a television service with interactive functionality in addition to the regular services. Instead of just receiving a broadcasted transmission you can now also respond and send information back. This opens up the doors for new possible features but the technology also decreases the costs of hosting and transmitting a TV-show.

To this day there have been a number of larger tests with IPTV and 2007 there were a total of 14 million IPTV subscribers [2] using it as their main TV platform. It has yet to take the major step and entirely replace regular broadcasting television.

To take that next step IPTV needs to make use of the unique possibilities that the technology can provide and offer services that are interesting and fascinating enough to encourage a majority of customers to change to this new platform.

Ericsson is one of the companies developing and providing both a platform and a delivery system for IPTV and by running projects that focus on looking at the future of IPTV they can create systems that are well prepared for what is to come.

## 1.1 Thesis Goal

Today IPTV suffers from not being able to identify itself as unique and valuable enough to require the investments that are necessary for its deployment. There are already services similar to what IPTV offers on other platforms such as Satellite TV and Cable TV and the somewhat technical advantages with IPTV are difficult to sell to the major audience.

The goal of this thesis is to help Ericsson create a better platform for their IPTV solution. Such improvements would serve to better prepare it for future development of services that will be offered on it, services that utilize the unique technology that IPTV provides. This would all be in an effort to try to differentiate from the competing technologies and gaining more customers.

### 1.1.1 Research Question

The main research question for this thesis to answer would be directly taken from the thesis goal presented above. That main question would be as follows:

- What improvements can be done to the Ericsson IPTV platform?

The best way to evaluate the Ericsson IPTV solution would be to try to develop the type of service that would show of the true value of the platform and with such a prototype find weaknesses and areas of improvement. But to develop such a service that particular service has to first be found and defined. A thesis sub question would then be a follows:

- How could a next generation IPTV service be designed?

The first step to do this would be to come up with a target service for further development. This new service would be aimed towards showcasing the future of IPTV services by utilizing the technology that is available with IPTV.

### 1.1.2 Delimitation

With the research questions found it is clear that some target service will need to be found. Such a service could theoretically be anything associated with the television experience. To keep the results within the ideology of Ericsson the target service should be evaluated and matched towards Ericsson's vision of future television.

For practical reasons the target service should have use-cases realistically implementable and should utilize the big screen TV as a main actor. Any alternative devices should only be used to enhance the experience.

Direct feedback during the process of finding the target service would be limited to people within the Ericsson Company due to confidentiality reasons.

Prototypes made during the thesis aim towards finding weaknesses in the platform and to demonstrate the target service. This has an impact on the level of detail and polish that the final prototypes receive.

Resources used during the development process were limited to what Ericsson could provide and hardware was available mainly when not occupied by employees. This had some smaller impact on the prototype end result presentation.

### 1.1.3 Stakeholders

This thesis was done by Petter Magnusson, a Computer Science student of Chalmers University. The work was assisted by Solution Area TV Product Development Unit within Business Unit Multimedia and was positioned in the Multimedia Headquarters Building in Kista just outside of Stockholm.

The thesis was supervised by Ericsson through Ola Andersson and line manager Hans Byström. The academic supervision was carried out by Staffan Björk of Chalmers University. Some collaborative work was done together with Ezequiel Lisak, Telecommunications, Managements and Economics student of Chalmers University.

The thesis result was aimed towards Ericsson IPTV development teams for improvements of the current platforms but also towards end-users such as with the design of future services for IPTV. Assistance while working and continuous feedback was received by Ericsson co-workers, Ericsson Volvo Ocean Race team and Ericsson Research in Kista as well as some individuals in Business Unit Multimedia.

## 1.2 Planning / Process

This thesis started in June 2008 and was originally planned to end December the same year. Some parts of the work were to be shared with a separate thesis done by the student Ezequiel Lisak [3].

Following the thought process set up by the selected research questions in section 1.2.1 the first goal of the thesis was to find a target service. The following work would then focus around that selected service. Since that service would serve as a base not only for this thesis but also for the Business Analysis done by Ezequiel Lisak, the initial effort to find that service would be a co-operative work process.

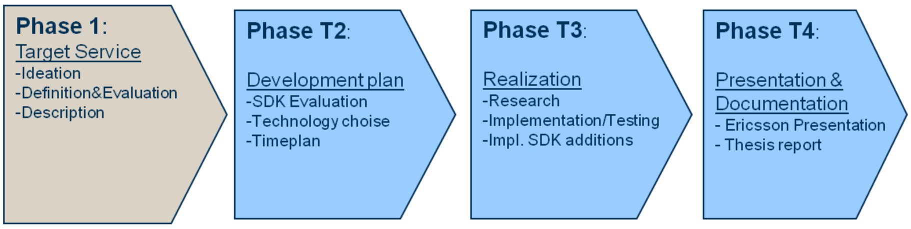
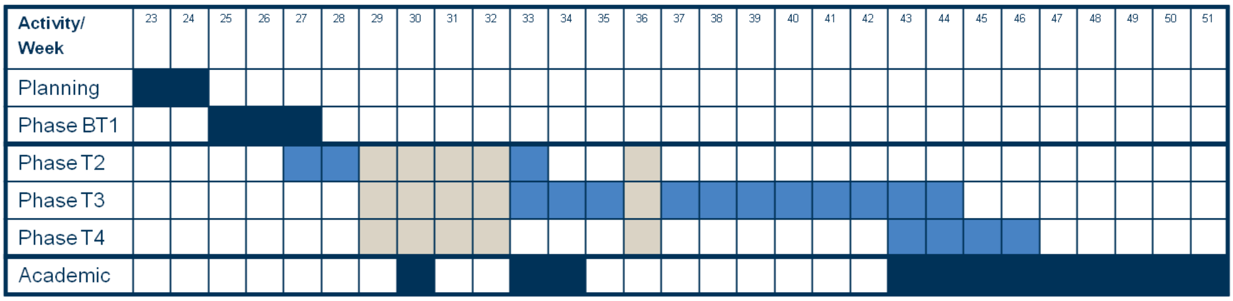


Figure 1. General plan for the whole thesis.

To be able to produce good results during the first phase of the thesis, such as a well motivated and interesting target service, a rather long period of time would be needed. This would however decrease the time available for further development and prototyping of that target service. The choice was made that it was more important to get a good prototype at the cost of a possibly less developed service concept. Therefore the selection process was planned to be no more than three weeks long. Requests for feedback during this process was instead collected and taken into consideration during the following work-process.

After selecting the target service a shorter period of time was set up to develop that service more thoroughly and to select what use-cases from it that should be used as a base for prototype development. This work was to be focused mainly on looking for realistic and interesting use-cases for the technology landscape and developing time span available.



| Activity/ Week | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 | 32 | 33 | 34 | 35 | 36 | 37 | 38 | 39 | 40 | 41 | 42 | 43 | 44 | 45 | 46 | 47 | 48 | 49 | 50 | 51 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Planning | ■ | ■ | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Phase BT1 | | | ■ | ■ | ■ | | | | | | | | | | | | | | | | | | | | | | | | |
| Phase T2 | | | | | 🟦 | 🟦 | ▒ | ▒ | ▒ | ▒ | 🟦 | | | ▒ | | | | | | | | | | | | | | | |
| Phase T3 | | | | | | | ▒ | ▒ | ▒ | ▒ | 🟦 | 🟦 | 🟦 | ▒ | 🟦 | 🟦 | 🟦 | 🟦 | 🟦 | 🟦 | 🟦 | 🟦 | | | | | | | |
| Phase T4 | | | | | | | ▒ | ▒ | ▒ | ▒ | | | | ▒ | | | | | | | 🟦 | 🟦 | 🟦 | 🟦 | | | | | |
| Academic | | | | | | | | ■ | | | ■ | ■ | | | | | | | | | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ | ■ |

Figure 2. Time plan for different phases of the thesis.

Due to the fact that the use-cases that were to be developed were not available during the main planning process further planning would be necessary at the beginning of the development phase. The development process was laid out to span all the way to late 2008 leaving some room for documentation before the preliminary deadline early December.

## 1.3 Background

Ericsson has a history of working towards staying in the front of new telecommunication technology. By investing in new areas with research and development Ericsson keep adding new features and systems to their portfolio.

A couple of years ago the television platform started moving towards the telecom industry by introducing more advanced video services over the internet as well as mobile television. Lately that development has lead to IPTV or Internet Protocol Television as the new platform for modern TV. IPTV is supplied mostly by a small number of companies worldwide including Imaginero, PCCW, Alcatel-Lucent and Microsoft [1].

Ericsson decided to follow the technology and by acquiring the digital video company Tandberg Television has as such started investing heavily in television and IPTV.

### 1.3.1 IP-Television

IPTV today is still a somewhat new concept to the general public. It is something many have heard about but few know the details on. This mainly comes from the technological foundation that the abbreviation stands for.

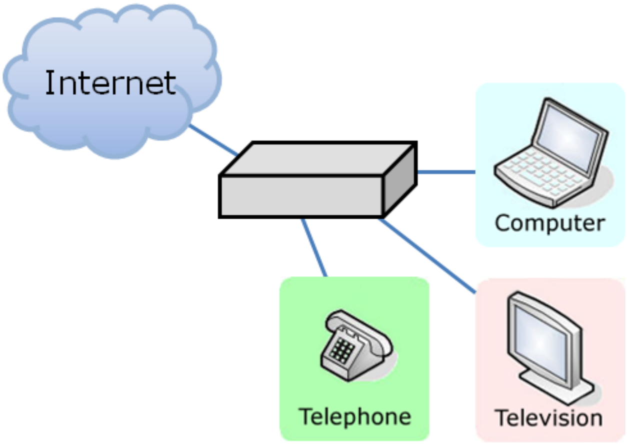


Figure 3. IPTV technology overview as part of the Triple Play solution.

IPTV means Internet Protocol TV [1] and it is a general term describing the use of an IP network to send television information. This is the same type of network as the one we already use today for internet distribution and telephone in some households. The possibility of serving the three together, TV – Telephone – Internet, in the same IP network is what the industry call Triple Play [4].

Figure 4. Set-top-box with remote control similar to what is used for IPTV.

The reasons for hosting a TV network over IP are many. The first and most simple one would be the sheer fact that providing Triple Play on one network would greatly simplify the distribution of media. One contact in the wall for all multimedia in the house is a rather nice vision. Secondly it would greatly lower the cost of hosting a channel and transmit TV content. Transmitting information over the air as is being done today is an expensive affair. To some extent this distribution cost has already been lowered in many countries. Sweden just recently transferred from sending analog TV information to digital transmissions instead. However, digital television in this regard is not IPTV. Digital TV [5] emphasizes the use of digital data, something that is used by IPTV as well, but says nothing about the channel that the data is transferred on. The third reason for using IPTV comes from this difference in distribution network. IP networks not only promote receiving of information but transmitting as well. This is a natural part of both internet surfing and a two-sided dialogue over the phone. For television this is considered to be something completely new.

### 1.3.2 New opportunities

Probably the largest difference between IPTV and regular television is the so called back channel. This emphasizes the availability to send information back from the user and his television to the content provider. The most common feature to make use of this functionality is Video on Demand (VoD) [6]. VoD is a common name for requesting private transmissions of video information. It can be the rental of a movie, done with just a few clicks from the sofa. It can also be viewing of recorded TV content such as passed shows or highlights. Examples of other applications that have been introduced with the possibility to send information are chat services and news feeds.

Beside the back channel IPTV also introduced a more application friendly environment in conjunction to the television content. With more powerful Set-top Boxes (STBs) more advanced logic can be run on the television. Although the power of these STBs might not be as great as modern day personal computers it still is enough to host a multitude of exciting features.

### 1.3.3 Interactive Personalized TV

Ever since Ericsson started working with television on the different platforms they have had plans to try to evolve the concept of it. With all the new technology that has been developed during the years since televisions introduction the possibilities for how it can be improved has practically exploded.

Ericsson’s ongoing Televisionary campaign [7] focuses on creating a personalized television experience, customized for every users unique desires. The campaign wants to give the power to the users to decide when, where and how they want to consume this service. By combining a number of different devices and platforms the vision presents users that can change the television medium to fit their needs and their wishes.

As a step on this road Ericsson has introduced an alternative version of the acronym IPTV. Instead of Internet Protocol TV the focus would be towards Interactive Personalized television, an experience that has been expanded with interactive functionality and that can be personalized to fit the user’s needs, all in line of the Televisionary campaign.

This thesis will try to find services that match this vision and find what technology may be needed to make them as good as possible.

# 2 IPTV Services – Finding the target service

The first step of the thesis, following the reasoning from the research questions, was to find out what the future of IPTV could look like. What could best bring the IPTV platform into the households of millions of customers and still keep what makes television such a powerful medium. Try to find not only the building blocks of these future services but also find one such service with high enough potential to lead the way for others.

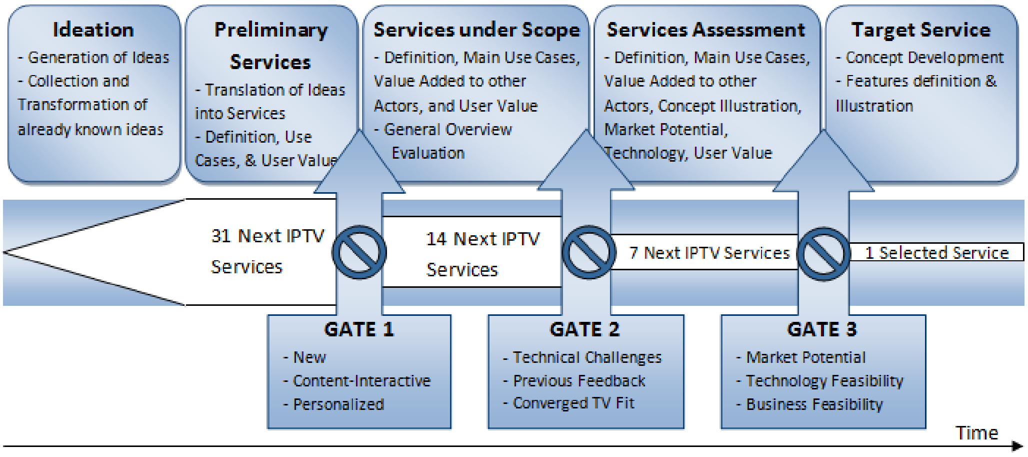


Figure 5. Detailed plan for finding the target service.

To find such a target service it was decided to use a divergence / convergence method similar to that of J.C. Jones [8]. By generating as much information as possible and generate services without constraints it would be possible to find new and interesting ideas. Those generated services would then pass a series of gates that would slowly converge to a target service.

For that convergence a target vision first had to be established. A set of attributes used as a goal and scale to evaluate the candidate services against each other. The attributes should follow Ericsson's vision of the future of television.

This part of the thesis was done as a co-operative effort between the thesis author and the parallel thesis student Ezequiel Lisak [3].

## 2.1 Service Requirements

To find the target service a number of requirements were needed to be able to compare found services with each other and to restrict the area of work. These requirements were partially set up as part of the thesis goal but had to be looked at and understood. The requirements for the target service were as follows:

- New: The service should not have been seen previously on other technologies such as Satellite TV, Cable TV or DTT.
- Content-Interactive: The service had to be interactive and the interaction had to be with the TV Content together with the TV and not just the TV as a device.
- Personalized: The experience should be customizable by the users to fit the needs and interests of the audience.

It was realized that matching found services to these requirements on a true / false basis would be very difficult. Instead they would have to be graduated and compared to each other using the requirements as guidelines for the selection process instead of a direct filtering method.

The requirement regarding content interaction was initially only requiring interaction in itself. This was later changed because it was found that interaction alone did not help to improve the general experience. Some types of interaction did improve the experience but not all of it. During the process of searching for future services it was revealed that one of the major factors with interaction that made it successful together with the television medium was when it had a strong connection to the content already available.

## 2.2 Generation

To be able to find the best possible service to continue working with the first step of reaching that service was to generate a large set of ideas or partial services. To make sure that no ideas were left out very few restrictions were present on what was added to the set. The generation process was then set up in two stages, first an ideation stage where the thesis workers gathered ideas from existing material and brainstormed around them. The second stage introduced brainstorming sessions where participants was given sort of role-playing scenario in where to come up with new ideas and service suggestions.

### 2.2.1 Ideation

Many services in the realm of IPTV or similar technologies have been around for quite some time and many new ideas have already been introduced. To consume the value of this pre-existing material the ideation stage involved browsing around published reports and websites scavenging ideas that had yet to be made into services [9]. Even texts concerning the general future of television was used to trigger creative thinking as a base for coming up with new ideas.

A problem that emerged in this process was concerning how ideas should be noted down. One could try to create complete ideas or solutions that could possibly be sold as services and later on get them down into use-cases that could be used in different solutions. Another other option would be to only note down small use-case oriented ideas directly and later on merge them together into services to pick from.

The issue was discussed and the decision was made to go for the second option as that would result in even less limitation to what would be added to the list and not concentrate solely on things that could easily be fit into a service.

In the end this did not work out as planned. With most of the content being produced during brainstorming sessions and the fact that the people present there was not aware of the earlier choice, new additions ended up as a mixture of small ideas and more complete services.

Another problem that was found during this stage was finding the format of the resulting ideas. For someone with a technical background and a base in logical thinking it is easy to get stuck in terms of use-cases and functionalities. This might be good later on for developing the tools to be used by future services but it does not help define a new service. Instead ides are better thought of inside of a context and describe not only the actions to be taken but the motivation and reasons.

*Record the TV program at some point and be able to*

*access the recorded content with any device.*

This idea is very general it could be used in a million different solutions and it doesn't really help with finding the next generation IPTV application.

*Check the temperature of your steak that is in oven via a widget on the TV.*

This idea instead shows how to put a use-case in a context, something that can later lead to more ideas around for example cooking or general widgets on the TV. It still is a rather simple functionality but it's not nearly as general.

### 2.2.2 Brainstorming Sessions

As a way to generate more services a brainstorming method was introduced. The method was influenced by experience from school projects. By gathering a number of participants to sessions where they were involved in a creative process guided by customized cards a large amount of services and ideas could be found.

The participants were placed in groups of two or three and then instructed to take one customized card from two of the three different categories at random. The cards were divided into the categories TV Viewers, TV Content and TV Places. The pairs then began to brainstorm using the cards as a catalyst. After about 5-10 minutes or when the groups would get stuck they were to change cards and continue.

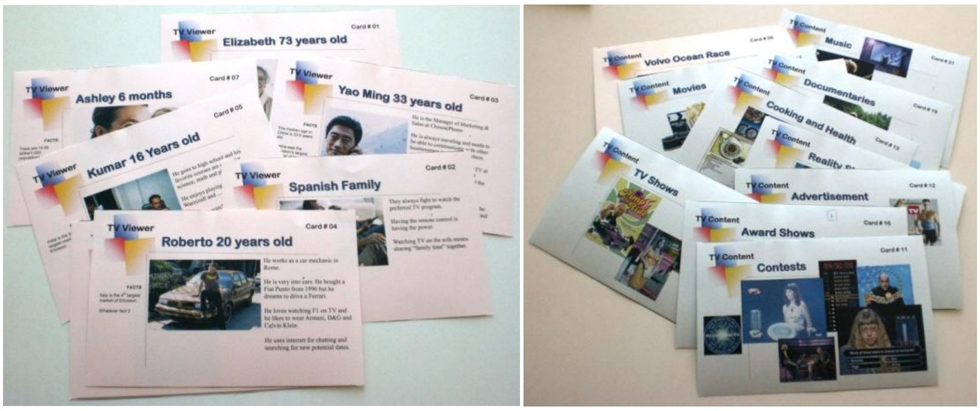


Figure 6. Paper cards showing the categories TV Viewers and TV Content.

At the end of the session all groups choose one of their ideas as the most promising idea and another as the craziest idea. These were then presented to the other groups and a winner of both categories was elected by voting. This way a discussion emerged concerning how the groups had generated ideas differently. New features could be added to them and improve them further.

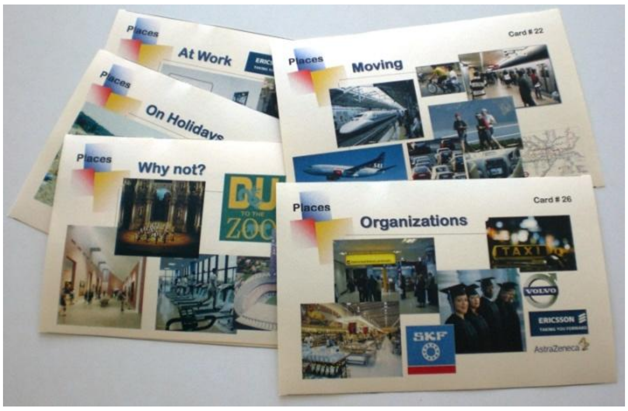


Figure 7. Paper cards showing TV Places.

After the sessions were complete all the produced material was gathered and sent out to the participants via mail. This way feedback could be gathered and the resulting ideas and services could potentially be improved even further.

The brainstorming sessions were quite successful in terms of productivity. We managed to get a list of useful ideas from them all. The only part that could have gone better would be the attendance of invited participants. With a total of about 20 invites being sent out for each of the sessions no more than 5 showed up on either of them.

The sessions were a fun way to engage more people in the creative process, they were however quite time consuming. The hour of time that was used for the sessions in this thesis produced enough material but both participants and thesis workers agreed that longer sessions could have been even more productive.

### 2.2.3 Service compilation

After generating ideas both through regular brainstorming and brainstorming sessions a rather long list of content had been produced. The decision taken earlier to keep the generated content in the form of smaller ideas had failed with the introduction of the session results as the participants of the brainstorming sessions had noted down anything that they could come up with. While it was a bit unfortunate it never affected the results in the long run.

With the quite rough mixture of different formats within the list of ideas a compilation process was necessary to get results in the form of services. Those services could later be further detailed and evaluated towards the requirements set up earlier.

The definition of a service in this case was decided to be a collection of features that would enhance the TV-experience. A discussion that came up during this work was the question of how broad and general such services should be. At start the focus was aimed towards quite specific packages such as a soccer service that would have features aimed towards maybe one larger soccer event. Such services would be directed towards the end-user.

This however might be problematic because it would overpopulate the list of services with quite similar entries. It would also pose a possible problem with selling such services since they would be extremely narrow in its targeted market.

Because of this the target was changed towards service packages that would be complete enough to be sold to providers that in turn could market them towards users directly. Instead of having for example a soccer service we would design it as a sports service that would include a set of user values that could enhance the experience of watching a larger amount of TV-content.

The services were created by going through the ideas and writing down all services that came to mind using the ideas for inspiration. For each service that was noted down all ideas were cycled through again to find anything that could fit into that service. If something new was found in the process it was of course added as well.

For different reasons some of the ideas from the list did not make it into any of the services. Most of these ideas were later added as a separate package as possible additions to the IPTV core service.

## 2.3 Gate 1

After the first step towards the target future service the ideation process had produced a list with a total of 31 different services. As mentioned earlier no technical restrictions had been in effect during this process so any ideas and services that had been produced were taking up equal space in the list. To slim down the list a series of gates were used with different requirements.

The objective of the first gate was to apply the main requirements set up for the target service, namely that the service has to be new, interactive, personalized and content interactive. Since the different requirements would be difficult to apply on a yes/no basis they were all simply weighted towards the services and any service with an overall weak preference towards more than one requirement were killed.

In total 14 services passed the requirements of the first gate and passed on to the next step. The killed services were mainly ideas that would clearly fall outside the scope of the thesis and technology as well as services lacking any connection to regular television content.

## 2.4 Detailed definitions

To get a better idea of what potential the different services had work was done to further describe them. All services were listed with a general descriptive text as well as their technical and business feasibility. To further evaluate the usefulness and potential general use-cases were given for each service as well. This not only helped out with technical feasibility but also showed a possible value in implementing a service with multifunctional and popular use-cases.

## 2.5 Gate 2

For the final selection feedback was required from a number of experts within the field of IP-television. For that purpose detailed descriptions would be needed of the candidate services. With the 14 services that passed the first gate it was not practical within set timeframes to create such descriptions and any feedback received would not reach the same quality as with a smaller amount of services. A second gate was introduced to reduce the list of candidates further for those reasons.

The aspects that were evaluated in this gate process focused mainly around practical uses. Did the services provide any technical challenges impossible to overcome in the timeframe of this thesis or were they too easy to implement? Previous feedback on similar projects was taken into consideration to get an idea of what would be most valuable to develop and demonstrate. Ongoing projects in the office were also matched with the services to try to pick out those that could make use available knowledge and technology.

The selection process of the second gate was done by the thesis workers together with colleagues knowledgeable within the subject of IPTV-services. The gate effectively halved the list of alternatives down to 7 candidates.

The services that did not pass the second gate were picked out for a number of different reasons. The main motivation to not continue with some was that they focused on technology that was deemed difficult with the available resources. Even if it was technically feasible it might not be practical in the scope of the thesis. To further shorten the list of candidates a preliminary selection based on business feasibility was made using knowledge of the participating colleagues.

## 2.6 Service exploration

To get qualitative feedback in preparation for the final selection an editable presentation was sent out to a number of experts within the company. This presentation contained all the remaining 7 services condensed down to a simple format describing the four main features of each service. The selected use-cases were each presented with a short text and a concept Photoshop montage or PowerPoint figure, all made to create a realistic view of how the service could look in real life.

## 2.7 Gate 3

With the final gate selection the target service for further development could be found. This process was executed by the thesis workers together with the company supervisor.

The result from the experts was taken into consideration both qualitative and quantitative and the final service was selected to be a service called IPTVSports. It was selected for having a high market potential as well as good technical and business feasibility. For creating prototypes it also enabled use of video content available through Volvo Ocean Race as well as possible demonstration opportunities.

# 3 IPTVSports – Target service definition

IPTVSports is a service that enhances the experience when watching sports events on the TV. Such sports events would include games from different areas such as soccer, motor racing and ice hockey as well as more regional sports such as sailing and skiing.

## 3.1 Concept

In preparation of the expert feedback that lay as a base for the final service selection four main use-cases were selected and expanded with image concepts as well as short text descriptions. The concept was later expanded by Ezequiel Lisak in his business analysis work [3]. The four main use-cases were expanded into four categories and served as a base for the later total of 15 use-cases.

MultiviewTV lets the user select different camera views on different devices. FurtherInfo allows additional relevant information to be shown in conjunction to the normal game. Replays&Highlights focuses on control and availability of important events. RevenueGenerators allows for the addition of new ways of gaining revenue on the platform such as betting and voting.

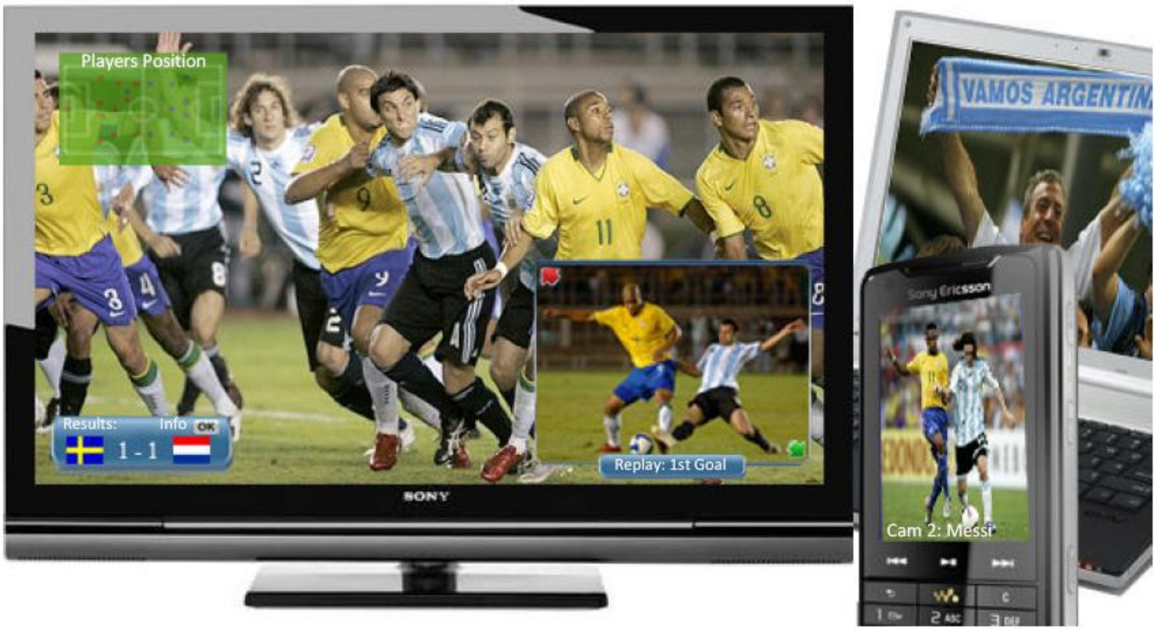


Figure 8. IPTVSports concept.

### 3.1.1 MultiviewTV

MultiviewTV presents a way for the user to take the role of the TV director. Instead of being forced to view only what the content provider decides to show, a new set of options allow the selection of a multitude of video streams to watch in addition to the standard overview. This personal camera view can then be presented either on the television together with the regular stream, or it can be sent and viewed on a separate device of the user’s choice.

Examples of possible views could be cameras following only specified players, important checkpoints or audience views.

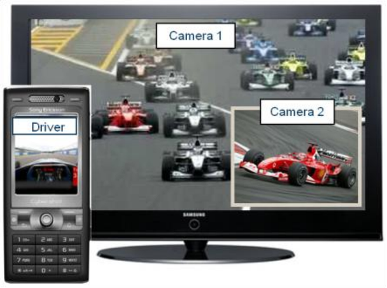


Figure 9. MultiviewTV concept.

### 3.1.2 FurtherInfo

For most types of sports there are large amounts of information that can be interesting to see, both for the amateur and the expert. It could be statistics of the current game, information about similar events within the sport, history of the participating sportsmen or tools to help the viewer understand the games.

An important aspect of adding information of this type is to keep the strong connection to the original content. Most of this type of information already exists on other platforms so keeping that connection is what makes it unique for the TV platform.

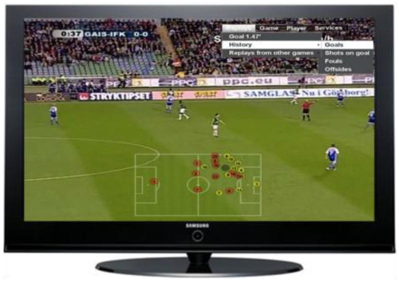


Figure 10. FurtherInfo concept.

### 3.1.3 Replays&Highlights

Replays&Highlights takes the tools of MultiviewTV and utilizes them together with the availability of clips showing popular events such as scoring or checkpoints. Today most replays are inserted shortly after the actual occurrence of the event for a more detailed view but no service is available to show them to newly arrived watchers and users that simply missed them. This new category of use-cases would allow users to see these events whenever they wish and on any device so that they do not miss out on new ongoing content.

### 3.1.4 RevenueGenerators

This group of features focuses around adding new ways for the Operators besides advertising. With interactive functions such as betting, users can spend money on the winning team, individual players or who the next foul will go to. RevenueGenerators also includes direct voting of the users using the remote control.

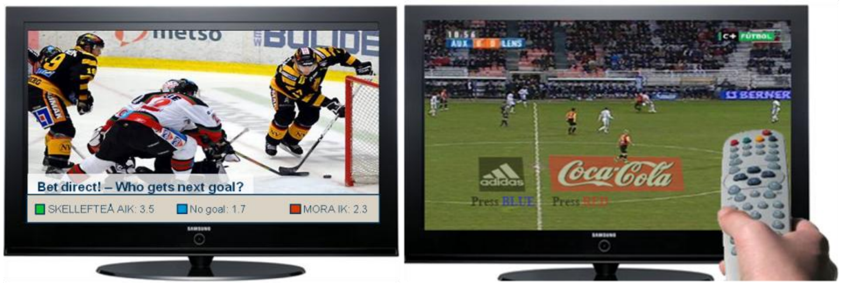


Figure 11. RevenueGenerators concept.

## 3.2 Context

IPTVSports was designed around features that can be applied individually and on a multitude of different types of sports events. Not all features may be fit for all types of sports but the creation of a service like this drastically increases the value if it can be packaged and applied to more content.

The initial idea for the service focused around soccer as the main sports event. This was chosen because of feedback during the generation process and inspiration from earlier projects such as Interactive Soccer [10]. Soccer however introduced a problem for creation of prototypes and the inclusion of video content due to digital rights. To circumvent this problem and to add an additional motivation for modularity of the service features a second context was added.

The added context was the international sailing event Volvo Ocean Race (VOR) [11]. Since Ericsson was the main sponsor of two of the teams in the competition as well as owner of the full rights to all recorded video material, VOR had been regarded as an option and motivation for the selection of IPTVSports in the first place.

In addition Ericsson was planning to have a multimedia pavilion in conjunction to the VOR events to show of media material and technology. Such events would be a good way to demonstrate any prototypes produced within the thesis work.

## 3.3 GUI Concept

The interface concept for IPTVSports was designed from the perspective of the different functionalities that was set up as the main parts of the service. Instead of focusing on the whole service at once each individual feature was looked at and iteratively improved to an initial design concept.

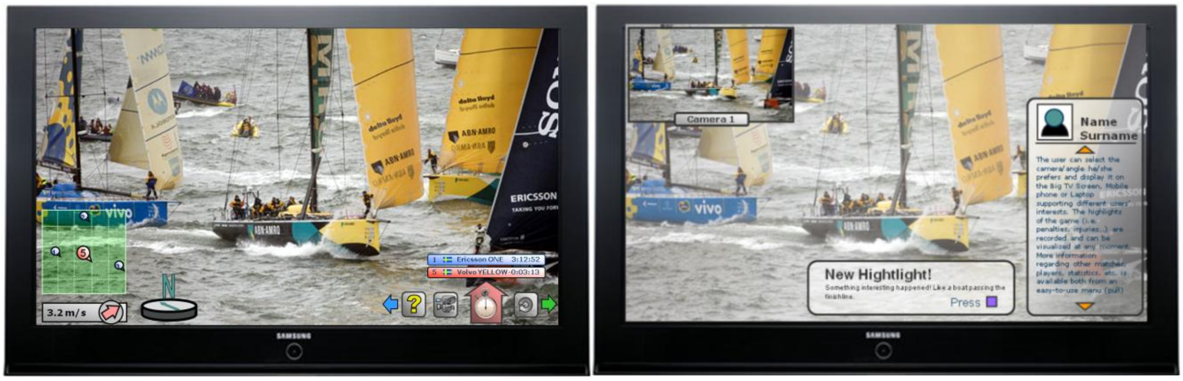
The only parts of the service that required any serious graphical design was the features categorized under FurtherInfo as well as the general menu system. The MultiviewTV and Replays&Highlights focused mainly on separate devices as well as picture-in-picture displays and as such left little room for actual design. Together with the rest of the features they did however enforce somewhat of a requirement for a more global system for interaction and position.

The target audience for the designed interfaces was recognized as anyone with access to the rest of the system. The interaction design would be required to match this and allow for almost anyone to understand and use the system. By using tools and methods as presented by Cooper and Reimann [12] the design focused on simplicity and usability.

While coming up with ideas for the design all focus was put on the Volvo Ocean Race as the main concept. It was recognized that parts of the concepts would have to be remade if this context changed but the objective was to only have a VOR version of the prototypes. Some consideration was made to allow for these different contexts in the future though.

The level of detail in the IPTVSports definition concerning what features would be available was quite low at this time so the design of the interface simultaneously allowed for introduction of new ideas. When focusing on what could be available in the FurtherInfo category a couple of new features were introduced.

The idea of an interactive time display, similar to what is available in most sprinting sports was tested. With a minimal and clear menu the user could change between some alternatives for what to view. It could be possible to bookmark certain participants in the list so instead of showing only the leader and the individuals in view everyone would be compared to the bookmarked player. While the idea was interesting it never reached a state where it was enough to prioritize over the other features.



The user can select the camera/angle he/she prefers and display it on the Big TV Screen, Mobile phone or Laptop supporting different users' interests. The highlights of the game (i.e. penalties, injuries..) are recorded and can be visualized at any moment. More information regarding other matches, players, statistics, etc. is available both from an easy-to-use menu (pull)

Something interesting happened! Like a boat passing the finishline.

Figure 12. IPTVSports concept in the Volvo Ocean Race context.

The main feature in the FurtherInfo was the Radar. It was introduced earlier in the service description and had a history from earlier projects in the department. The main challenge with designing it was the change of concept compared to earlier designs. Further details on the concept development can be found in section 4.2.2.

A simpler part of FurtherInfo to receive some design work was the display of text information. Displaying such information offered less value with mostly information and not much interaction but it would require less work to implement as well. It was an important part of the whole service to consider because it did still take up some space on the screen and even small amounts of interaction require some keys that then would be occupied.

While coming up with new ideas and features it became clear that all added features would introduce some type of new interaction, partially because of the project requirements to enhance just that. This new interaction clearly introduced problems with designing what keys on the remote would correspond to the different features. Due to the size and functionality of all features it also showed that some sort of menu system would be needed for running them since having all up at the same time would be quite messy.

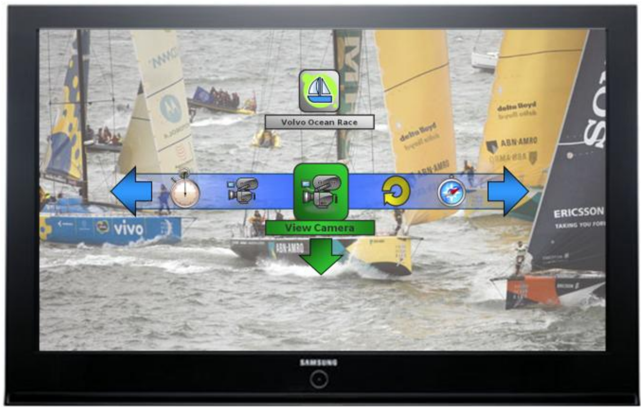
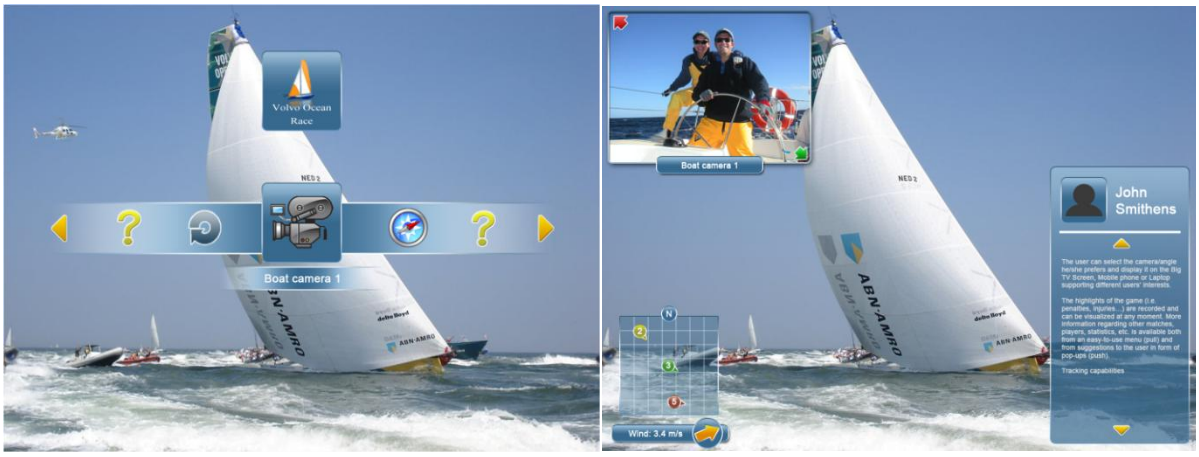


Figure 13. Improved menu-system concept.

A design for a menu system was created and designed. It was influenced by menu systems on modern video game systems such as the Playstation 3 [13] due to the similar nature of the platforms. It was however rather clear that such a system would offer little value in a prototype so the priority of actually implementing it was rather low, but it did introduce the problem.

With all the different features competing for interaction it became clear that there wouldn't be enough keys on the remote to serve them all at the same time. Instead a focus system was introduced, a system similar to window managers used on PC systems. Only one part of the service would have focus at a time and all interaction would be with that part. At the same time there would be a way to circulate that focus in the same way as the now standardized Alt-Tab functionality of Microsoft Windows [14]. With such a system it would be easy to use a limited amount of keys for all the different features. For this design the colored keys that are now almost standard on all remotes could easily be enough for most features. One thing that should be noted is the problem with color blindness with using such a system. A way of circumventing that could be to not only use the four different colors on the buttons but let them have different shapes as well. For further details see Appendix.

When a design for most of the different parts of IPTVSports had reached a good enough state the Power Point concepts were replicated in Photoshop and at the same time further enhanced with a color theme. The theme that was used was mimicked from earlier concepts of different VOR applications used on several different platforms.



The user can select the camera/angle he/she prefers and display it on the Big TV Screen, Mobile phone or Laptop supporting different users' interests.

The highlights of the game (i.e. penalties, injuries...) are recorded and can be visualized at any moment. More information regarding other matches, players, statistics, etc. is available both from an easy-to-use menu (pull) and from suggestions to the user in form of pop-ups (push).

Tracking capabilities

Figure 14. Final IPTVSports concept in the Volvo Ocean Race context.

## 3.4 Prototype candidates

For the next step of the thesis one or more use-cases were to be chosen as candidates for prototyping. The objective of the prototype was to find limitations in the available technology but it was also a way of demonstrating what can actually be done with it as well. Therefore it was important to select use-cases that did not pose any clear problems that could not be solved in the given time-plan. Neither did it do any good to select only use-cases that was relatively easy to implement.

An additional goal with the prototypes was to focus on technology close to the market releases. There had already been work done to create prototypes for a pure demonstrative purpose showing how one could interact with the television medium. These kinds of demonstrations had mainly been done using flash or simple 2D images without functionality.

Out of the use-cases that was available for IPTVSports there was two that best fit the objectives. A radar overview found in the FurtherInfo feature and the MultiviewTV personal camera. In addition to them additional functionality of FurtherInfo such as news and statistics could be implemented if the main use-cases were to be finished ahead of schedule.

The radar was chosen to be the first use-case mainly because it best fits the description of new IPTV services found during the first phase of the thesis. A similar prototype had been made during the Interactive Soccer [10] thesis work earlier but the product there had only been made as a Flash demonstration. Though the demonstration had received good feedback in general it was quite far from being seen as a reality on the market due to technical requirements of Flash. A possible option to pass this problem was available in the use of SVG, a 2D vector graphics format that was being investigated by the Ericsson Research department.

MultiviewTV was selected as the second main use-case. With its close connection to the content as well as the modularity it fit the same requirements as the radar when matching the found description of future IPTV services. An ongoing project in the Ericsson office could also prove to be an asset in using the convergence with multiple devices that it provided. At the initiation of the development phase that project could not yet supply the technology needed by the MultiviewTV use-case. This was partially solved by planning the work so that the radar prototype would be developed first giving more time to them.

## 3.5 Prototype planning

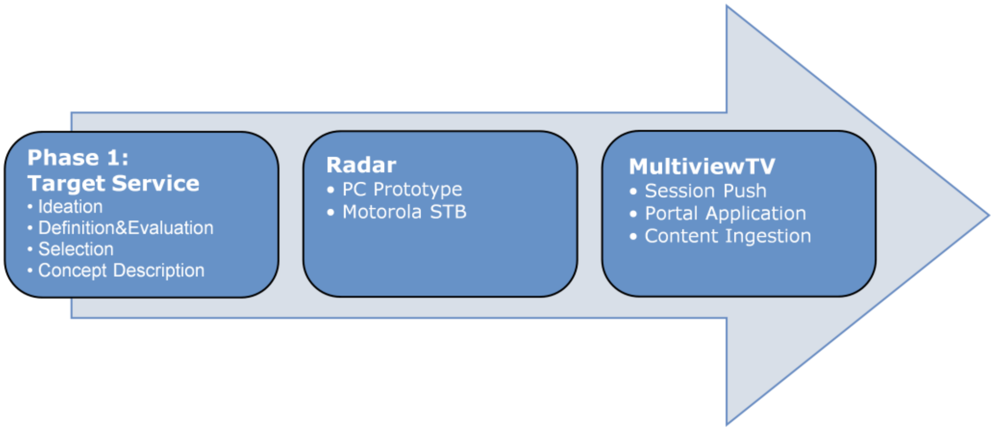


Figure 15. Development phase planning.

When both the target service and the prototype candidates had been found a more detailed planning was produced for the implementation phase of the thesis. The two major prototype use-cases that had been selected were the FurtherInfo use-case Radar and MultiviewTV.

The prototype for MultiviewTV involved using multiple devices to show content on. This technology was made available by a separate project. This was however not scheduled to be available until later the same year. Because of this it was clear that work should be initiated on the Radar.

The plan for the FurtherInfo Radar was to design and implement it in stages. An iterative approach was used by starting out on the PC platform and slowly moving over to real TV hardware. There was no set deadline for the Radar prototype development. The last stage would be to move it over to the same system as Ericsson's IPTV solution. This would require some outside development as well to complete so by halting development of the Radar and initiating work with MultiviewTV there should be time enough to finish of the work later on.

The focus of the work was to shift over to the MultiviewTV prototype after the Radar had reached a good enough state using the available technology. Additionally a prerequisite to the shift of focus was that the outside project team had reached far enough in their work to support the use-cases needed by MultiviewTV

The planning for MultiviewTV development would depend slightly on the complexity of the platform which would be explored later. The first step would be to help out with the use-cases within the outside project to ensure that they could be used for the thesis as well. It would also serve as training for working with the IPTV system. When the relevant parts would be complete a separate portal application would be made implementing the MultiviewTV use-case.

# 4 Radar – FurtherInfo use-case implementation

One of the features found in the FurtherInfo package is the Radar. The Radar is a graphical representation of positional data, a way of showing more detailed position information than the regular video stream can show. The idea originates from elements found in modern videogames where it is essential to have a simple overview of the playfield showing where the different actors are while the camera has to be fixed on whatever the player is controlling.

The Radar features lots of information in a visual format and requires the display of movements so animation is a big part of it. While animation can be very powerful for good looking applications it is also a big obstacle with its requirements for computational and rendering powers, something that has so far been missing on the Television platform.

## 4.1 Specification

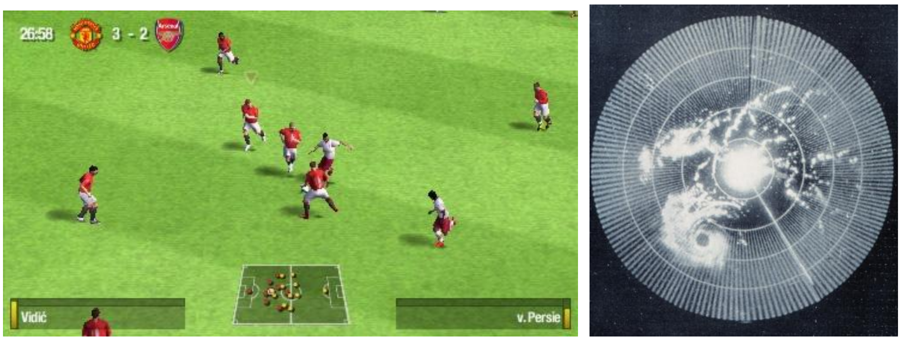


Figure 16. Fifa 2009 video game radar [16] and 1960’s weather radar.

The original concept of the Radar is the same as with scientific radar systems for airplanes or boats. It should show the relative position of entities that are relevant to the user in the context of sports. For a football game this could mean showing the playfield and all the players as well as the ball.

With the basic concept of IPTVSports being modular it was important for the features to fit as many different contexts as possible. This modularity fits the Radar as well and it can be applied to most types of sports. For a vehicle race such as Formula 1 the user could possibly be interested in each cars position around the racetrack. If instead watching cross-country skiing a view of the contestants closest to the focus of the camera could be of interest since the tracks often lie in areas with obstructed view such as forests.

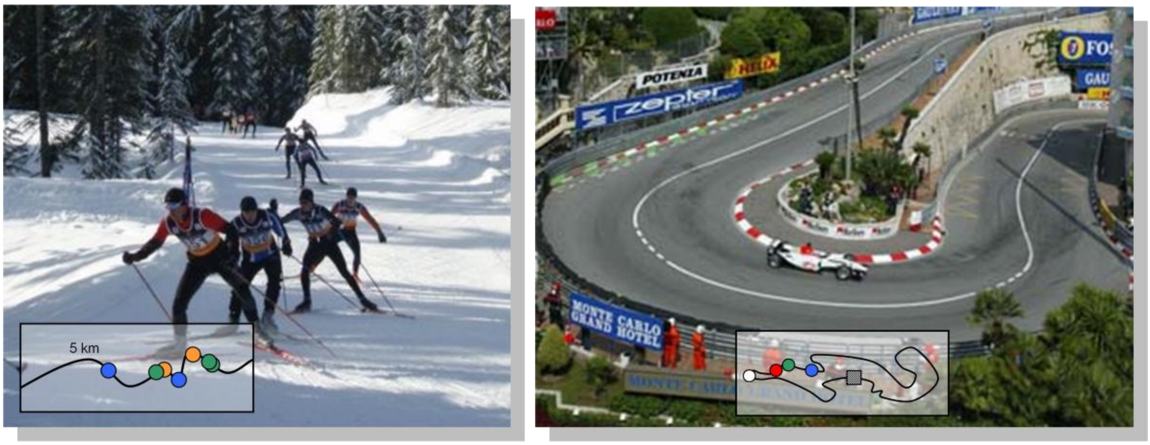


Figure 17. Radar concept for cross-country skiing and formula 1 racing.

The context of the prototypes in this thesis was decided to be ocean sailing event Volvo Ocean Race, or more specifically the in-port races done at specific ports along the way. These types of races involve around 5-8 ships that simultaneously sail along a track set up by 4-5 buoys out in the water

[17]. The main goal of the race is be the first to clear the track by best benefitting the wind when going from one buoy to the other while also strategically position the boat in a way that blocks the wind for the opponent boats.

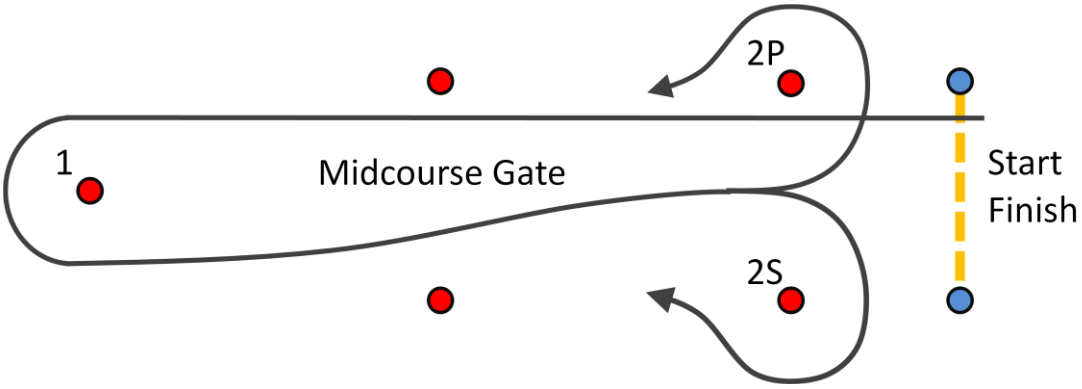


Figure 18. Overview of the VOR in-port race. (Start, Gate, 1, Gate, 2P or 2S, Gate, 1, Gate, Finish.)

One of the problems with following a race like this on regular television is that no good overview is available. Even the best of commentators have problems picking out the relative positions between the contestants solely by watching the video feed. The IPTVSports Radar feature would present a simple display of this missing information without requiring advanced knowledge and without interrupting the regular TV view.

## 4.2 Design

To create a design for the Radar one has to come up with a back-end solution for the administration and collection of data as well as a front-end GUI for the presentation. While designing both it is important to consider where the different parts of logic should be placed. The most obvious limitation of the front-end systems is the limited computational power that is available. As such it would be wise to minimize the components placed therein. However if a large part of the system runs remotely from the server then it suffers badly when scaled up in numbers of served users. Generally the amount of information that can be sent from the server to the front-end system is not as limited as the other way around so requesting large amounts of information with simple queries would be most efficient [18].

### 4.2.1 Back-end

The back-end solution for the Radar service requires basically two things. It requires positional data from some source, a source that can be different depending on the type of event that it follows. In addition to that it needs to have some server software that can collect mentioned data and package it for transport to the front-end.

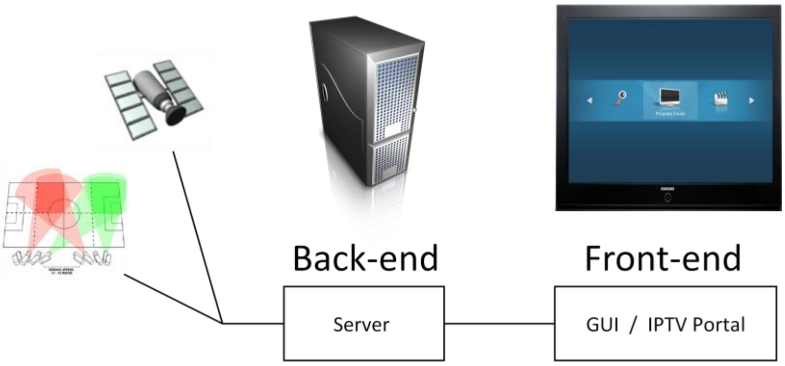


Figure 19. Radar prototype system design.

Since the positional data source can differ so much with each context this thesis will only give some examples of how it can be done in some possible sports environments or, more importantly, how it is already done today in others.

For the realm of soccer a solution to this has been in use now for some time. As mentioned in the previous project with Interactive Soccer [10], there is a Swedish company called TRACAB [19] that supply positioning of the players in soccer games from the Swedish national league. The solution involves the use of a large number of cameras that focus on different parts of the field, registering players' movements in that part. Together they can give full view information about all players as well as the position of the ball.

The context of the Radar prototype required instead the positioning of the Volvo Ocean Race sailing boats. This gives the opportunity to use GPS-data since all boats are equipped with GPS-receiver as well as a transmitting system that sends out that information to the race organization.

For other sports events other solutions could be considered. With the available technology used today GPS systems are small enough to be used in all formats from vehicle racing to marathons. If contestants follow a set and limited path cameras or short range radio tracking could be just as simple.

However, one thing that is required, be it in position tracking or in the server software, is that coordinates needs to be recorded and saved along with matched timestamps for a longer period of time. This is because when the information is needed later on by the front-end system it has to be possible to receive relevant data for what the front-end video currently shows. Even with live broadcasts the timing of the video feed can vary greatly depending on geographical location or technology with which the transmission is received resulting in the need for older position data. This is obviously required for non-live transmissions as well.

### 4.2.2 GUI

When the initial design of IPTVSports was made the radar feature was represented by the soccer version from the Interactive Soccer project. That design was based almost directly on existing video game equivalents. Due to the big difference in context with the new prototype the experience to bring from Interactive Soccer was limited.

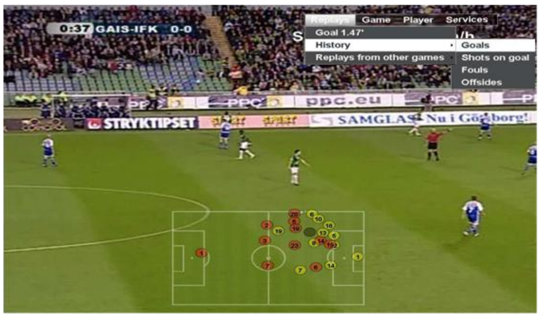


Figure 20. Screenshot of Interactive soccer thesis prototype.

As mentioned earlier Volvo Ocean Race is an ocean sailing race event, spanning over several months and large geographical areas. For the Radar it was decided that the so called in-port races would be the target to model towards. At the time the amount of information regarding exactly how such in-port races would play out was very limited. This had a quite big impact on the graphical design.

Due to the lack of knowledge the design of the Radar had to be very general. Compared to the soccer version that had its base around the field that the players were restricted to the racing boats could be sailing all over the ocean. Even if the races were using a specified track it would still be difficult to anticipate exactly how the boats would move.

The solution would be to design the radar using a different approach all together. Instead of focusing on a full overview of the event the radar would center on a set position and show only all relevant entities within the immediate area around that position. This would be a step back towards the classic view of radar as a tool to show entities in the area around the radar antenna.

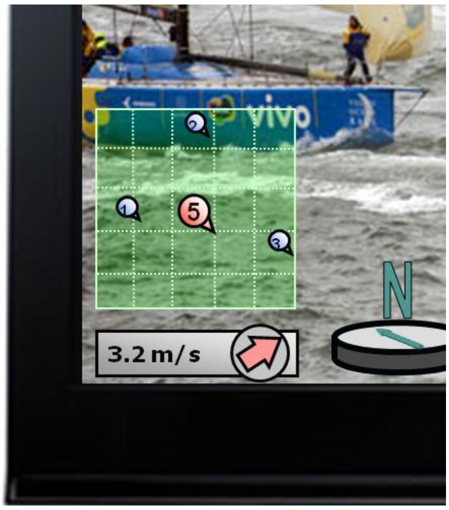


Figure 21. Early concept of the Radar in the Volvo Ocean Race context.

The graphical design that was moved from initial sketches to the Power Point concept was a square grid overlay that showed a number of different elements of interest in relation to each other. Together with the grid an information plate showing wind speed and direction as well as a compass was added. Having only limited knowledge of the sailing sport wind direction and speed seemed like reasonable pieces of information to add.

Since the focus of the radar grid was not set and would be dynamic the natural choice would be to focus it on a specific boat. The idea was inspired by existing VOR concepts with boat camera channels and having the radar follow the focus of such channels would be the logic step to take. The focused boat element was increased in size to emphasize the selection. Directional markers were also added to all boats since all movement would be relative to the center and that center was now a moving entity as well. All boat elements were shown using different colors and numbers to clarify what element would represent the actual boats.

When the design was deemed good enough a more detailed version was produced using Photoshop images. All elements were redone using a metallic blue color scheme mimicked from existing VOR concept designs.

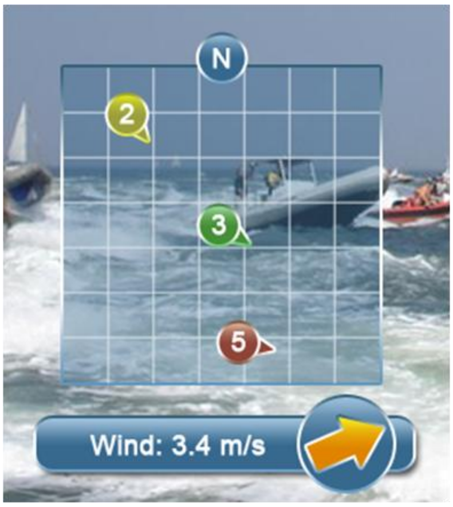


Figure 22. Themed concept of Radar for Volvo Ocean Race.

During the process of redesign some changes were made to the original concept. The compass separate to the radar grid was removed. It was originally added to give an idea of the direction the video stream was looking. This was however a bit confusing together with the radar as the grid and the boat elements would have a separate direction. Instead of the compass a small marker was added to the top of the radar grid to indicate the direction of north. Now it would be possible to take out directions using the boat directional markers together with the boat directions in the video stream.

Another design choice that changed was the different sizes of the boat elements on the radar grid. This change was motivated by additional clarity over the identity of the previous smaller boat elements. It was deemed clear enough which boat that was focused upon even with equal size.

The Photoshop design was used as a base to try out further changes during the development. Since most of the elements of the design were created using vector graphics and gradient coloring it was easy to translate the design to an SVG image format later during the production.

## 4.3 Technology Choice

While selecting the Radar as one of the use-cases to make prototypes of some research was done to find what technology could be used to implement it. Since the basics of the Radar had been done earlier but then only as a demonstrative application the focus was now to try to find a platform that was as close to the commercial IPTV solution as possible. Ideally it would be implemented on the real systems by the end of the development.

This choice was however mostly relevant for the front-end and GUI as the server hardware did not have the restrictions that the front-end suffered. Even if the back-end system choice was not a critical as the front-end it still had a major impact on how to divide the logic between the two systems and what limitations would apply on the interface between them.

### 4.3.1 Back-end

The Radar server software choice stood between creating a Java Server application or a Java Servlet [20]. Java in general was the natural choice both because of previous knowledge of the platform by the thesis worker but also by the available co-workers. It would also fit well in to the pre-existing applications used for the IPTV-solution.

A Java Server application would be continuously running and manually accepting connections made by the front-end system.  A running server has the advantage of being able to keep information about for the clients connecting to it. This allows for large parts of the calculations to be run on the server as it knows everything the client has been doing. The application also has the possibility of running an open connection to the clients, pushing information when necessary without the requirement for constant requests.

Java Servlets require a web server to handle incoming requests and redirect them to the application. This imposes a rather large restriction to what can be done with the connection. On one hand it handles code that no longer has to be written for this prototype but it does not allow for open connections and the pushing of content in the same way as a Server application does. In a way one can say that it is the simple but more limited solution. The connection interface uses HTTP requests which can be made quite small and easy to use and it would still provide enough flexibility for this prototype.

With both alternatives considered the choice was made to focus on a Java Server application in first hand since it offered more flexibility in the design of the communication and the placement of the different parts of the logic.

Later on problems with the Java Script implementation used in the SVG browser Ekioh forced a change in back-end technology. The communication could not be opened directly from the browser and because of that a Java Servlet was used instead. For this prototype a larger investigation to the problem and a possible solution was considered not worth the loss of flexibility with changing to a Servlet.

### 4.3.2 Front-end

There were really only two available technologies to choose from with the front-end of the Radar service. One option was to focus on the use of HTML code together with Cascading Style Sheets (CSS), the tool used by the already existing IPTV front-end software. Using this platform would ensure that any prototype made would quite easily be inserted into the IPTV solution. The nature of the Radar would however require excessive animation with its elements and HTML/CSS does not support such technology. Dynamic updates could be possible but the end result would most probably suffer in visibility and performance. A decent solution could be found but it would probably be time consuming.

The alternative technology that was looked at was Scalable Vector Graphics (SVG), a 2D vector graphics format developed to create efficient and good looking graphics and animation using XML to describe geographic objects and Script languages to handle logic. SVG was however not yet supported by the IPTV platform and using it risked ending up with a loose prototype like the previous Interactive Soccer project. It was not available on the platform because no SVG browser software was available for the STBs that were used by the Ericsson IPTV solution. Instead SVG could at the time only be run on selected STBs, some of which Ericsson Research were working with for testing purposes.

The main reason for continuing with SVG as the main option was that the process of converting the SVG browser used for other hardware to work with the IPTV-STBs had been initiated. The conversion was scheduled to be complete around the end of the thesis project so developing a prototype on the working STBs and later moving it over was a reasonable solution. Even if this would not be available the prototype would still have a value as it would run on a STB-hardware, a considerable step forward compared to the PC-prototype from Interactive Soccer.

## 4.4 Introduction to SVG

Scalable Vector Graphics (SVG) [21] is an open standard XML specification and file format. It is used for describing two-dimensional vector-graphics that can be both static images and dynamic entities such as animations or interactive content.

Vector graphics is a way of describing content as shapes. More common image formats today such as JPEG or PNG use raster graphics (also known as bitmaps) that instead use fixed pixels to describe its content. Vector graphics allows for smaller files, unrestricted scaling, animation and rendering effects. Other popular formats of vector type today include Shockwave Flash and fixed-layout formats like PDF.

SVG has been around since 1998 and has been developed by a group of companies within the World Wide Web Consortium (W3C). Since its creation the specification has been developed into a main version including 14 different feature sets as well as some subset formats developed for specific purposes such as SVG Tiny and SVG Basic that was meant for smaller devices such as mobile phones and PDAs.

SVG can be created and edited in any text editor but most efficient for creation of advanced graphics and animations an SVG editor is recommended. Unfortunately such editors exist only in very few numbers. One of the most used editors is the free and open source application Inkscape [22]. Other commercial editors exist as well and plug-ins for saving files in SVG-format can be found for graphic editors such as Adobe Illustrator.

The SVG renderer used for this Thesis was a version of the Ekioh Web Browser [23], a web browser designed specifically for set top boxes. The browser had support for the SVG Tiny format as well as JavaScript, HTML, DOM and more.

## 4.5 Development

The development of the radar followed an iterative approach. The goal was to first try to create a version of the visual design in the SVG format. That SVG design then was fit into an existing SVG framework to be run on a PC-version of the Ekioh browser used by Ericsson Research. Here functionality could be added in an iterative process and the communication could be tested to the server. Video was not supported by this PC-version.

When the software reached a level of quality that deemed sufficient it was to be moved over to be run on a Motorola STB. On this hardware everything could be tested and adjusted for performance and graphical design could be reviewed with video playing in the background.

By the time development of the converted SVG-browser for the IPTV-STBs was done the final movement of the software to this platform could be executed and integration with existing IPTV software could be initiated.

The development of the back-end software followed the requirements from the added functionality for the front-end application.

### 4.5.1 Front-end

While developing the Radar prototype most of the work was initiated by adding new features to the front-end. Then if the added logic needed support from something that the back-end could not provide then the back-end was extended in turn.

As mentioned earlier the front-end development was also very much driven by the platform it was currently being run on. Starting with a simple graphical design with SVG moved into a connection between that design and Java Script code to create animation and interaction. Running the prototype in a PC-browser development is very effective but it certainly creates problems when it later has to be moved to hardware with much more limited capabilities. Functionality that issued no problems during development could very well be so demanding that it has to be removed later on because of performance issues or even incompatibilities. Fortunately for this thesis no strict requirements for performance existed.

#### *SVG design*

Because of the restriction to the SVG Tiny format with the Ekioh browser all SVG designs had to be created following that specification. This was a problem because the existing editors for SVG mainly support the creation of standard SVG files. There are editors that can export images to this format but none that was available during this thesis. Instead the SVG design created to follow the early concept was made editing the XML markup by hand.

Since the Radar concept was made using mainly gradients and vector shapes the SVG design could come very close to the original concept. The limitations of SVG Tiny did however slow down the process by forcing the use of simpler building blocks to create the different shapes. Circles are for example not supported in the SVG Tiny specification. Instead one has to create each circular shape by using sets of two or four Bézier curves [24] with arguments calculated using algorithms to come as close to a perfect circle as possible.

The final design included all parts from the concept created earlier.

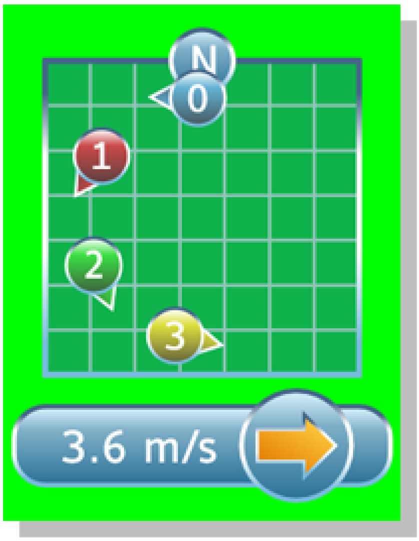


Figure 23. First SVG design of the Radar prototype.

### PC

The first interactive version of the Radar prototype was made using a PC version of the Ekioh browser (v0.1 1900 August 2008). The prototype was created within an existing widget framework that had been used by Ericsson Research to test out the capabilities of SVG. The framework was a simple widget based system to load and unload included widgets one at a time and to divert input to the one currently loaded.

The first test with series of updates initiating animation were done in a manner such that the elements clearly moved to their new position and then halted while waiting for the next update. This looked rather odd and not very natural as a representation of the real movement. If the updates would have been small enough to not be clearly visible then a solution like this could work but then the whole idea with the Radar would lose its key value. If movement of the participants in the event would be that small and slow in relation to each other then it would be easy to get clear overview using existing methods.

While instead testing continuous animation updates it became clear that the frequency of the updates is relevant to how natural and flowing the overall impression will be. When the boats were simulated to move in a circular path every 60 seconds and updates less frequent than about every second it clearly revealed the new updates.

In conclusion it was clear that the update frequency had to be high and the individual animations from the updates had to overlap with each other. Changing the frequency was as easy as changing a variable but animation introduced a new problem.

When the Radar prototype reached a near complete status following the specification it was moved over from being run on the PC-version of the SVG browser to instead run from a Motorola set top box.

### Motorola STB

The television hardware that was used for the prototype was a Motorola STB [25]. The environment was set up by connecting the STB directly to the development PC via network cable. The PC then simulated both network administration with DNS, file-server for STB boot software and web-server for hosting the Radar front-end SVG as well as back-end Servlet application.

After moving the Radar over to the STB it was revealed that the widget framework used as a base for running the Radar was not compatible with the browser version used by the STB. For this reason a new framework had to be created to house the Radar. When the new framework was developed some of the ideas created during the concept design were tested (See section 3.3). A type of widget manager was introduced to control what widget was in focus and only redirect input to that widget. The manager functioned much like a modern operative system window manager. Widgets could be opened and removed and one button was assigned to initiate a rotation of the focus among the widgets much like the functionality of the Microsoft Windows Alt-Tab window swapping [14] (see Appendix).

Due to the widget design of the Radar prototype in the previous framework no major problems were introduced by moving it over to the new framework. Some parts of the Radar SVG design was changed during this move due to feedback and testing results from the PC-platform. The Radar wind arrow feature was considered unnecessary or at least too large. Some ideas were introduced as to how to include it in some other manor but due to time constraints and lack of value it was never again included.

The wind speed label was removed in the same revision for similar reasons. When the initial design was created a limited knowledge of the sailing format led to the belief that this information was of a higher value that it really was. Later research showed that the in-port races were executed on an area small enough for wind speeds to change very little. A dynamic element to show this information was therefore not of high value. The label element was however kept for information about available interaction together with the new widget framework and window manager functionality. It highlighted the focus of the Radar by being visible only when focused upon.

Performance of the Radar on the Motorola STB showed anticipated difference due to the obvious decrease of computational power. Some parameter changes were made to smoother the animation as much as possible such as update frequency and animation overlap. It was realized that to make further improvements of the animation parts of the core animation coding would have to be profiled and redesigned. Detail reduction of the SVG elements could possibly contribute to the overall experience well.

Since the prototype preformed well enough for its purpose and that the Motorola box would hopefully be exchanged for the IPTV STBs later on any more work to finalize performance felt unnecessary. The development of the IPTVSports Radar was decided to be finished. Any further improvements or changes would be initiated when outside parties had produced the platform needed to move the Radar over to it.

### 4.5.2 Back-end

The main purpose of the back-end solution for the IPTVSports Radar was to gather positional information from some outside source and package it and present it to the front-end software depending on parameters received. The connection to real positional data never got used since the prototype never reached a field-demonstrable state but the design of the back-end was still made to allow for such connections if needed. To produce data simulated coordinates were generated throughout the development.

Most of the development of the back-end was initiated by new requirements produced from adding features to the front-end. As such no explicit back-end requirements existed.

The first back-end solution to be created consisted of a simple server application designed to accept incoming connections and respond by sending information back. Due to problems with the connection to the script engine this did not function as intended and instead the back-end was exchanged for a Java Servlet application.

## Java Servlet

With the change to a Servlet application the connection management was abstracted to a web-server and communication was changed to a simple request-answer basis. The prototype was iteratively developed by first only returning fix coordinates for entities using timestamps but was slowly enhanced with a number of different features.

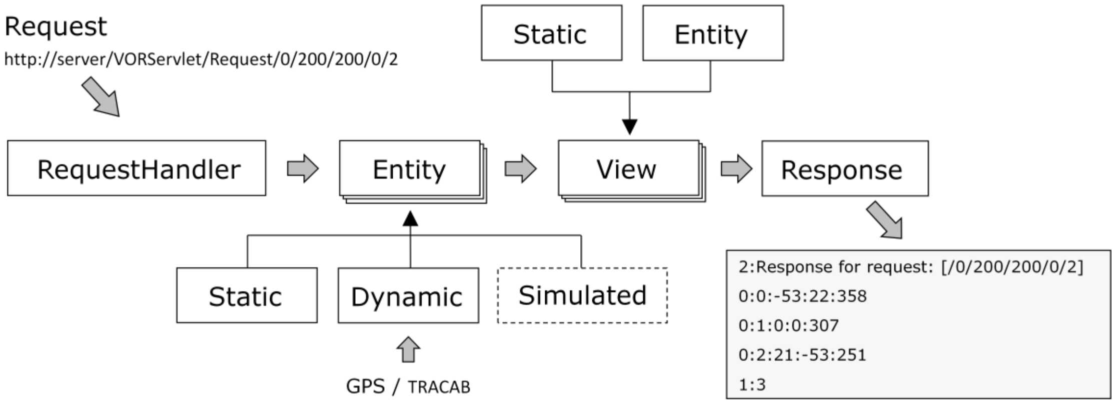
The final version of the back-end servlet was designed to be as expandable and modular as possible. The same server software would be possible to use for a number of different implementations of the Radar used in a number of different contexts.

The main part of the back-end is the ResponseHandler. It is an HttpServlet that receives any requests from the front-end and parses the request arguments. Each request requires a set of arguments some of which are mandatory and others that are optional. The arguments are passed in the request URI.

| Request type | Argument1 | Arg2 | Arg3 | Arg4 | Arg5 | Arg6 |
|---|---|---|---|---|---|---|
| 0 (View) | Window height | Window width | Time | View ID | | |
| 1 (Entity) | - | - | - | Entity ID | Scale | |
| 2 (Custom) | - | - | - | Center Lat | Center Long | Scale |

Figure 24. Request syntax for the communication between the portal and back-end system.

The rewriting of the Servlet introduced the concept of Views. A view is a center point somewhere in the geographical world that is also connected to a scale between meters and pixels. To get useful information a window size has to be set. Whenever a view is requested it is used as a window to look at the world. The requested view can be used to produce a response that gives pixel coordinates relative to the size of the window for all entities visible inside the window and information about all other entities that they are hidden.



```
2:Response for request: [/0/200/200/0/2]
0:0:-53:22:358
0:1:0:0:307
0:2:21:-53:251
1:3
```

Figure 25. Radar back-end server system overview.

Views can be static positions or they can be connected to entities like boats to follow. Views can be created before hand or by request from the front-end arguments. Coordinates can be transformed from several different systems and use Universal Transverse Mercator (UTM) as a common end system.

Entities that are available in the system can also be created freely. They can be static such as buoys with a fix geographic coordinate or they can be dynamic such as boats. Dynamic entities can be connected to an outside source for fetching of real time coordinates. In this prototype an additional type of entity was created, the simulated entity, to produce simulated positional data.

All entities and views were in this prototype coded into the Servlet software but could probably be loaded dynamically using settings to allow for easy creation of new contexts.

Due to some problems with the parsing of XML in the front-end system a custom response message format was created. This could easily be changed in the back-end by creating a new Response object type, which holds all logic concerning the format of the response message.

## 4.6 Results

The development of the Radar prototype never continued after it was finalized for the Motorola STB. The outside conversion process for SVG on Ericsson's IPTV-STBs was delayed long enough for it to get out of reach for this thesis. The Radar prototype still ended up as a very good demonstration of the type of IPTV services that could be expected in the not so far future. It might not have the polished state that was intended but every time it is demonstrated anyone that sees it can grasp the power that is available with good looking animation applications. It kick starts the imagination for all the possible services that could be developed.

### 4.6.1 Concept

The Radar as a concept is still one of the strongest in the IPTVSports package. One of the problems while creating the first version of the Radar was the lack of knowledge around the context it was to be designed for. During the work process this knowledge slowly became available through research or contacts that was not available at the beginning of the process. Assumptions that effected priorities and information were corrected and testing gave a better insight of the final product. Because of this the graphical design of the Radar was improved as a result of the development process.

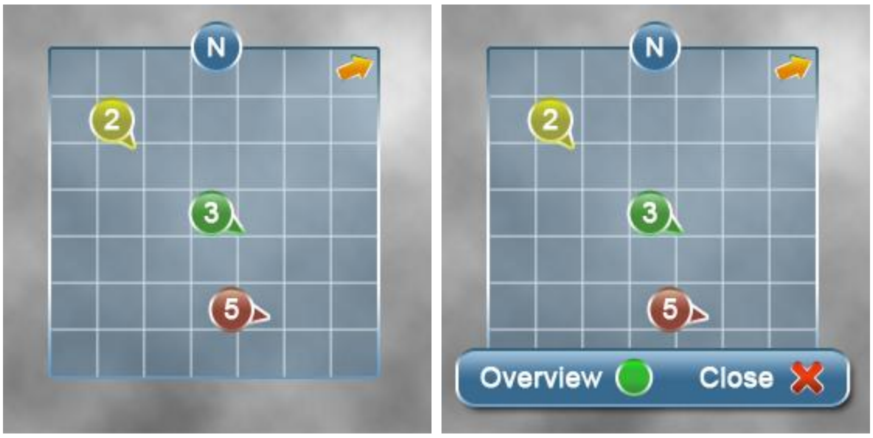


Figure 26. Left shows the default GUI and right shows enabled interaction when the widget has focus.

As with most graphical design the widget as a whole was simplified. The wind direction element was removed because in comparison to the first assumption the wind direction over the area of an in-port race does not change any considerable amount. While knowing the wind direction of the race as a whole could be interesting it is surely not as critical as to require such a large part of the design. One option would be to position a smaller version of the element inside the Radar. This was never implemented in the prototype but remain in the final design as it is still a relevant feature.

The label plate below the Radar that earlier was used to show an animated wind speed text was removed because as with the wind direction this information simply does not change on the rather small area of an in-port race. The plate and label could instead be used to show other relevant information about interaction with the widget such as changing view or closing the widget.

The size of the Radar is a variable that was not possible to get good testing data on. It was run in conjunction to a Volvo Ocean Race video when tested but it was difficult to get a good judgment on the size. One of the problems with widgets like this in general is the size and position it has so that it does not block visibility of other relevant information in the video.

All in all one of the main realizations from the development was the necessity of context know-how, something that was missing at the start of this process and slowly realized along the way. The fact that know-how is critical for making applications as connected to the context as this is nothing new. What is relevant here is that to make competitive applications and services for IPTV a connection and depth within the content context will most probably be necessary and with that the know-how.

### 4.6.2 Prototype

Due to limited availability of the hardware required to demonstrate the Radar prototype no images were taken of the final product. Even if such images existed they would show little more than the final concept images due to the heavy reliance on animation.

Even if the prototype never moved over to the real IPTV-platform its use for testing the value of SVG as the next technology to use for user interfaces on STBs still fulfilled the set expectations. The resulting prototype clearly showed an interesting application with high quality animation and graphics together with HD-video on limited hardware such as the Motorola STB in this case.

SVG delivers easy to build, good looking GUIs with a quite low performance cost. This was one of the goals when it was created. The question here is whether or not the slimmed hardware of the set top boxes could handle a more complicated application with animation and multiple elements.

The Radar software that was run on the Motorola STB was in no regard polished enough for a final product. There was clear interrupts in the animation and the intervals between new updates was highly variable due to slowed down performance. The overall impression was good but if the requirements that is set up by similar applications on PCs or other platforms really pushes the IPTV quality standard.

To develop an application similar to the IPTVSports Radar today it would not be clear from the start if a polished enough product would be possible to reach. The required hardware is on the edge of what is available today so the complexity and intensity of the application would be critical to the end result. A Radar-application created for use in a context such as Volvo Ocean Race or Formula 1 Racing could have much higher chance of success in this matter than for example a Soccer version. This is because of the number of involved elements or players and more importantly the intensity of the movement and required update frequency.

# 5 MultiviewTV use-case implementation

The MultiviewTV use-case set introduced a way for users to select from a number of different views of the same content. Today a typical sports event has everything from 20-50 different cameras following the event. Usually there is also a director team that during the transmission selects what cameras that should be shown in the stream at any given moment.

The alternative that MultiviewTV offers to its users is the possibility to choose between such cameras freely. The main stream is directed to please the general mass but it will not represent the view of every individual. This package of features could offer tools to select just the cameras that show what the user wants to see.

One of the important parts of the MultiviewTV set is the device selection. The user has clearly shown interest in some other video than the main one but where should it be shown? The simple choice would be on the television set just as everything else but another option would be to allow for the user to instead show it together with the main view. Personally selected streams could be show in smaller scale on top of the main view with technology known as Picture-in-picture (PiP). Other screens on separate devices could be used such as mobile phones or PCs of different sizes, even separate TV-sets.

## 5.1 Specification

For MultiviewTV the use-case to base the prototype on was not entirely clear from the start. The different use-cases were divided depending on where the alternative video stream could be shown and each target device required different technology. The project should be based on the Ericsson IPTV Portal platform to further emphasize that the service was developed to be realistic with the technology available to the market.

The prototype was planned to be a part of one of the ongoing projects in the office because the project could offer technology that would be of value when implementing MultiviewTV. The project also offered a lab that was set up with the latest IPTV platform which allowed for testing of the prototype.

Two main use-cases were selected to be the goal of the process. Viewing of personal cameras as a Picture-in-Picture element on the big screen TV and viewing the personal camera on a mobile phone.

The PiP solution seemed like the least demanding use-case in terms of required technology and developing it could be done in parallel to other use-cases. It would serve as a back-up solution if technology required from outside sources would not be available.

The choice of using mobile phone as an external display for the second use-case was made because one of the main use-cases in the separate project. The use-case was to move a video session from the big-screen TV to a mobile phone and as such matched perfectly with MultiviewTV.

Following the same sports context as with the previous work the MultiviewTV-prototype was designed for Volvo Ocean Race and the additional camera views that was to be available were so called Boat Cameras, views from on-board the participating boats in the in-port races.

## 5.2 Design

Since the MultiviewTV prototype was implemented using the existing platform from Ericsson's IPTV solution the prototype design was mostly focused on graphical layout and interaction. The interaction of the prototype required an alternative menu system compared to the one available in the current solution. While using the mode extensive menu concept produced with IPTVSports might have given a better result the decision was made to instead focus on creating a smaller menu just for the MultiviewTV prototype.

The prototype was planned to utilize the technology known as Picture-in-picture (PiP) [26] to view the alternative cameras. This interface method provides multiple (usually two) different videos at once on screen. One main stream is shown over the majority of the screen while the other is displayed in a smaller format on top of it. PiP would be a core part of the MultiviewTV feature and would be required to reach its full potential though the technology has some limitations to it in terms of performance and not all hardware support it.

### *Separate Devices*

As an alternative to viewing additional video streams on the main screen with Picture-in-Picture other devices could be utilized. Many other displays are available in the modern home today and a large portion of them are small enough to be mobile. Watching the personalized videos and views on a separate device also bypasses a lot of the problems that was introduced with PiP above. It does not cover parts of the TV screen, it usually has high enough quality to not require special video views and it does not disturb any possible fellow watchers in the room by covering up the screen.

However, to watch alternative views in separate devices introduces new problems. Even if the content that is to be viewed might not need to be recorded with big visible details it still has to be decoded in smaller formats to be viewable on smaller devices. While this decoding process might not remove too much detail it still requires a separate delivery method than the TV and this means that synchronization becomes a problem. Watching the same sports event on two different devices with two different video streams will be a nightmare to synchronize, if not impossible altogether. It might very well be possible to get an acceptable result but it will require a very stabile platform and distribution first and it will probably take some time for IPTV to reach that stage.

As with PiP alternative views on different devices would best benefit from showing non-synchronized streams such as those explained in Replays&Highlights.

## 5.3 Platform

In line with the goal to produce market realistic prototypes the natural choice of platform for the MultiviewTV was the upcoming Ericsson platform. With an available lab set up to test the platform and a parallel project with some overlapping use-cases the setup was ideal.

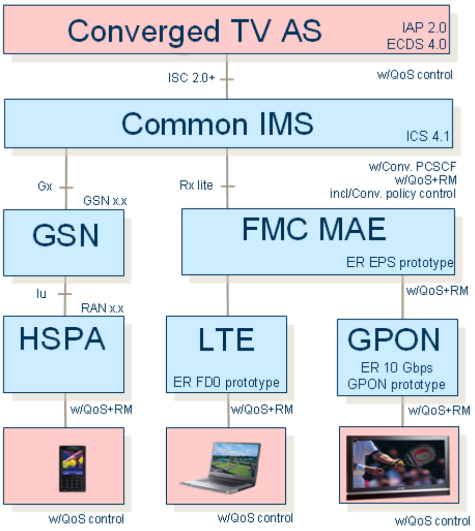


Figure 27. Converged TV overview.

Together with the general Ericsson platform the prototype was developed using functionality supplied by the parallel project. The parallel project being run mainly during the autumn 2008 focused on combining the use of Ericsson IPTV solution together with external devices and their respective TV solution such as Mobile TV. The goal was to prototype a number of use-cases to show of the sharing of sessions between these devices.

One problem with the platform was that the Picture in Picture technology that was included in the planning earlier was not available. This forced development to focus on multiple device use-cases instead of PiP.

### 5.3.1 IPTV Portal

IPTV Portal is the front-end system that is part of the Ericsson IPTV solution.  For the users the Portal serves as a window into the system by presenting a Graphical User Interface controlling the TV-channels and by offering controls to the available services. By packaging the television in this Portal it enhances the experience for the user.

The Portal also serves as a platform for new services to be developed on by supplying modules with functionality that can be accessed. Newly developed applications can then be accessed through the Portal interface to be run by the users.

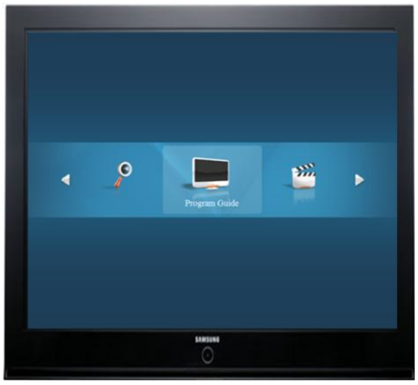


Figure 28. Ericsson IPTV Portal menu.

#### *Architecture*

The Portal is build to have the characteristics of a ‘thin client’. By this it is implied that most calculations and processing is done server side while the client keeps no critical information in memory in case the system would unexpectedly be shut down. It uses JavaScript divided into a number of different modules dynamically loaded depending on the system it is being run on and what is needed. The GUI is built up using generated HTML code together with CSS for easy branding and customization.

The server side applications are built using mainly IMS Java services to provide the Portal with relevant information to display. Video streams can be initiated and controlled using the video player build into the STBs.

#### *Coding in the Portal*

The IPTV Portal is designed to allow for easy development of new applications. To create a new application a new JavaScript module is created. As long as the new module is deployed with the rest of the Portal it will be loaded when the Portal is running.

New modules can request links to other existing modules and with them links to functionality that those modules provide much like a regular API. Such functionality allows integration of new modules into the portal and added functionality to available services.

To package new modules for deployment and to test that links to existing modules are correct new code needs to be set up following a set of rules. Then using Apache Maven it is tested and compiled into a deployable archive ready to be run.

All new applications will automatically be run as long as the requirements are met but to be accessible for the user within the Portal they need to use connections in the portal framework to add elements to menus and commands.

## 5.4 Development

The development of the MultiviewTV prototype within the Portal platform was divided into a series of parts. Some of the parts were requiring external work to be preceding them and others were doable from the start.

The original intention with the development of the MultiviewTV prototype was to follow an iterative process much like with the Radar prototype. Starting with a simple application created in the Portal framework and further functionality added with time. This did however not work out as planned.

Part of the functionality of MultiviewTV was dependant on external parts such as the functionality from the parallel team; this was known from the start. The timeframe was not entirely set however. By the time development were to begin with the prototype external work with device-interaction use-cases from the other team had not yet been initiated. After setting up the environment for the Portal framework it was decided that helping out with actually implementing one of the device use-cases would help learning the Portal and at the same time make sure that needed functionality existed.

Another unexpected source of work that emerged during development was the preparation process of the video content that was to be used for the prototype. This could most probably have been avoided or at least planned if with knowledge about video formats prior to the thesis.

### 5.4.1 Portal Application

The best way to implement a feature like MultiviewTV would be to launch relevant applications when a user simply changes to the channel that holds the content for it. To start the main IPTVSports application in our case one simply goes to the Volvo Ocean Race channel and any special options and functions associated with it becomes available. This type of application-channels is something that is supported to some level in the framework. It was however not available to this thesis both due to the somewhat outdated Portal version but also because of a lack of knowledgeable personnel on the subject.

The second best and simpler solution was to create a special application with its own video player and own interface that would be separate from the core-services of the Portal. The application would be available from the main menu in the same way as most other services. While it lost some of the connections to the regular TV content it still was considered to show a good enough view of the concept.

The Portal application was created and tested in a simulated PC environment and when all basic functionality was working such as startup and shutdown it was deployed and tested in the lab on the real hardware system. One big limitation with the PC-simulation similarly to the Radar prototype was the lack of support for video. As a major part of the basic application was to start a main background stream showing some general Volvo Ocean Race content it was top priority that is was working.

The final version of the MultiviewTV prototype ended up using Video on Demand functionality rather than live streaming. Since the application was already separate from the regular channels this solution felt just as real and was considerably easier to implement.

### *Application interaction*

When looking into the design of the interface objects of the IPTVSports Portal application there were a few points set up as goals. As mentioned earlier the menu interaction design was a critical part since the control interface that is used with a TV is often considered tiresome and difficult to use. To simplify the controls of the IPTVSports application two direct interaction options were used. One main menu was available using the same control-scheme as the rest of the Portal. The menu held options such as the shutdown of the service and future activation of other IPTVSports features. Another input, in this case the button Up, was used to initiate the MultiviewTV use-case. By doing it this way as opposed to adding it under the main menu one action was eliminated from the MultiviewTV controls. The remaining choices were left to selecting a camera view and a target device, in total 3 actions.

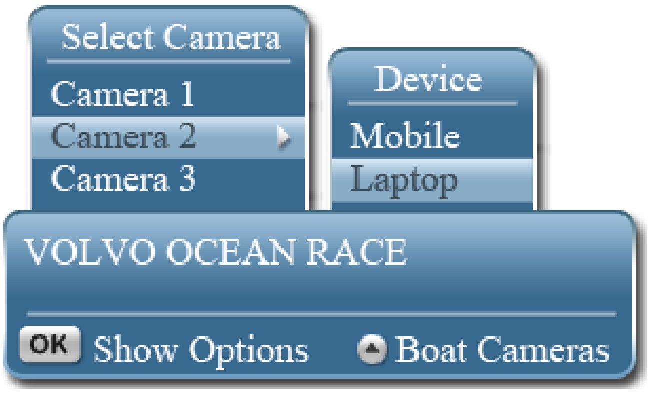


Figure 29. IPTVSports menu for MultiviewTV in the IPTV Portal prototype.

The visual design of the application would focus mainly on the menus. Again to follow the suite of the other Portal services an informational window positioned at the lower-central part of the screen was added. This window was used in other services for showing channel information or video-controls, for this application it helped notifying the user that they were in fact watching this special channel with additional functionality. To help further distinguish it from the other services the interface received an alternative look, also following the concept used by the Radar prototype.

### 5.4.2 Video Content

The video content that was used for the prototype was HD quality recordings from the Volvo Ocean Race events during 2008/2009. Ericsson owns the rights to all video content produced during the Volvo Ocean Race, this was part of the reason for selecting it as the context of the thesis prototypes. Such video content would later find its way down to the project and could be utilized.

## 5.5 Results

The time plan set up for the development of the MultiviewTV prototype was not exceptionally detailed. Because of the hope for an eventual merge of the application and the earlier Radar prototype once external development was done no set date was noted as the deadline. The only final date that was available was the date where the parallel project team would be finished and demonstrated.

The external development for SVG support on IPTV STBs got gradually delayed and soon enough the final date for the MultiviewTV prototype was in reach. Because of some unexpected work with video formats and problems with limited testing time in the crowded lab the whole prototype was threatened to not be done in time. With some great help from colleagues and some late night development the prototype finally reached a functional status, just days ahead of the final presentations. Because of the limited availability to utilize the lab for demonstrations a video was made to show the use-case in action.

### 5.5.1 Concept

The MultiviewTV concept did not change much during the development process. Part of the reason for that was that since the Picture-in-Picture implementation never got realized no feedback of it was produced.

The only real design parts present in the resulting prototype was interaction and menu elements, some of which were not present in the early concept at all. The end design consisted of a main information element acting as a reminder of the special options available and as a guide to accessing them. It connected to the custom menu that was used by the prototype use-case.

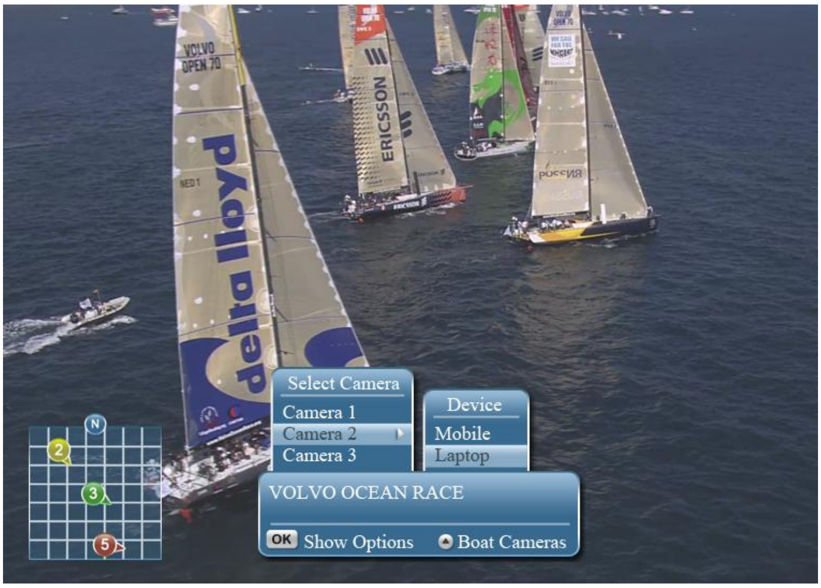


Figure 30. Final MultiviewTV prototype.

### 5.5.2 Prototype

Due to time constraints and problems with PiP the MultiviewTV prototype never received any additional features outside of the use-case used from the parallel project. The final version did however serve quite well as an example of how such use-cases could be utilized in regular services and even as a single-use case it showed a great value of IPTVSports MultiviewTV.

The most valuable result of the MultiviewTV prototype was the experience gained during the development. With the direct connection to the main research question concerning improvements for the Ericsson IPTV platform the development helped note down difficulties and problems with the frameworks and devices.

For presentation purposes a video was made showing the final prototype when performing the main use-case.

# 6 Discussion

How well is the Ericsson IPTV Solution fit to meet the requirements of future IP Television services and what can be done to improve the platform for a new way of watching television? This was basically the question that this thesis was set out to answer. The first step in doing so was to try to see into that future and find what path IPTV services would follow to bring this new technology into our homes.

The future that was found was that of IPTVSports, a service trying to enhance the living room experience for anyone that wanted to get more out of watching sports events on television. By giving easy to use information about the events as well as options to see whatever details the audience wanted to see in addition to anything they were already used to.

Part of the goal with this thesis was to find problems and difficulties with the platform while developing for someone with a small amount of experience with the system and the reasoning here reflects that.

The discussion focuses on giving examples and recommendations on the different results and problems that was found during the thesis work. The given solutions are by no means the only ones available but will give an idea to how to move forward in light of the problems encountered during the thesis.

## 6.1 Next generation IPTV services

As the first step towards improving the Ericsson IPTV platform this thesis set out to find a new IPTV service that could serve as a demonstration of what can be done as well as what would be expected. The process of finding that service ended up producing a long list of good propositions. While the quality of the suggestions might have varied it would be strange if some of them wouldn't be developed sooner or later.

The resulting ideas for services that were found could generally be categorized into a few different groups.

**IPTV as a new platform**

The service offers something already available elsewhere however; by using IPTV instead the service can utilize high definition videos to a greater extent. The fact that the service could be moved to the living room and be available to a large group of users is might add value as well. Simple examples would be services that offer instruction videos or videos that supplement a family board game.

**IPTV to enhance**

The service focuses on showing much more than just the regular TV video. Additional information such as statistics and graphics are added to the screen. Possibly additional video views are shown together with the main view. One example of this is would be the IPTVSports Radar.

**IPTV for participation**

The service focuses on giving the users a way to directly participate with the show they are watching. Instead of having to use separate systems as a means to communicate to the community or television people IPTV offers a direct channel to utilize. Users could for example use the service to vote for their favorite participant in the latest Idol series.

Services could of course end up in more than one of these categories but in general they would match at least one.

Another group of services that was present during this thesis was a group that used IPTV to allow additional devices to interact with the regular TV experience. This is a new direction for television development that Ericsson puts a high focus on. By including the question Where? In their latest campaigns on how we watch TV they want to open up this new design space and start thinking of the medium in other areas than the living room. Whether or not this will succeed is going to depend on how robust and user friendly the experience will be. It will indeed be interesting to see how well it will do.

### 6.1.1 Content Interaction

As mentioned earlier in this report it was decided quite early that a requirement for new services would be interactivity of some sort. The strategy of changing TV from a passive to a more active medium would help differentiate IPTV from the TV that we have today. By following this strategy it was found that interactivity in itself did not necessarily improve a service. Looking at existing IPTV services it was clear that efforts had been made to add this interactivity by taking interactive services from other platforms and adding them to the IPTV platform. Some examples of such services were games played on the screen using the remote as well as presence and chat systems allowing users to keep in touch discussing freely. The result was not that IPTV offered something new but rather that it included lesser versions of something that was already mastered elsewhere.

While developing services for this thesis it became clear that something else was required to make IPTV into interactive television. Within this problem was also the solution. The connection that was missing was that to the television. By simply adding something on top of the classic broadcasts the link to the TV was lost. By keeping that connection services would stay focused on TV as a medium but with added interactivity. To show the importance of the link between the TV and the services the term Content Interaction was used and during the process of finding the next generation service Content Interaction was considered mandatory.

### 6.1.2 Widgets

IPTVSports was made up of a series of smaller graphical objects and features. This type of building block-applications has been seen earlier on other platforms. The best way to describe them was to use the interface term Widget or originally window gadget. Widgets have historically described small interface details but lately they have moved to describe free floating elements, usually customizable and interactive.

This format fits very well with the television platform because of a number of reasons. To allow the connection between the content and the added features most additions to the screen should stay small enough to not cover all of the main content. If the added feature takes up all screen space it will lose the most important part of the television device which is the video content. The widget format does not automatically create content interaction but the format is good for not taking over all the focus.

### 6.1.3 Content Interactive Widgets

IPTVSports was the direct result of the search for future services but it not only showed a service with a high value, it also showed that by combining Content Interactivity with the Widget format you would get a very good template for IPTV services. Content Interactive Widgets or CIWs for short could serve as the recipe for future IPTV services by keeping the key results from this thesis.

IPTVSports showed that CIWs really work on the IPTV platform. Content Interactivity will be the most important factor when creating services to show the true power of IPTV. While such services might be more difficult and expensive to develop they supply something that sets them apart from what is available on competing platforms.

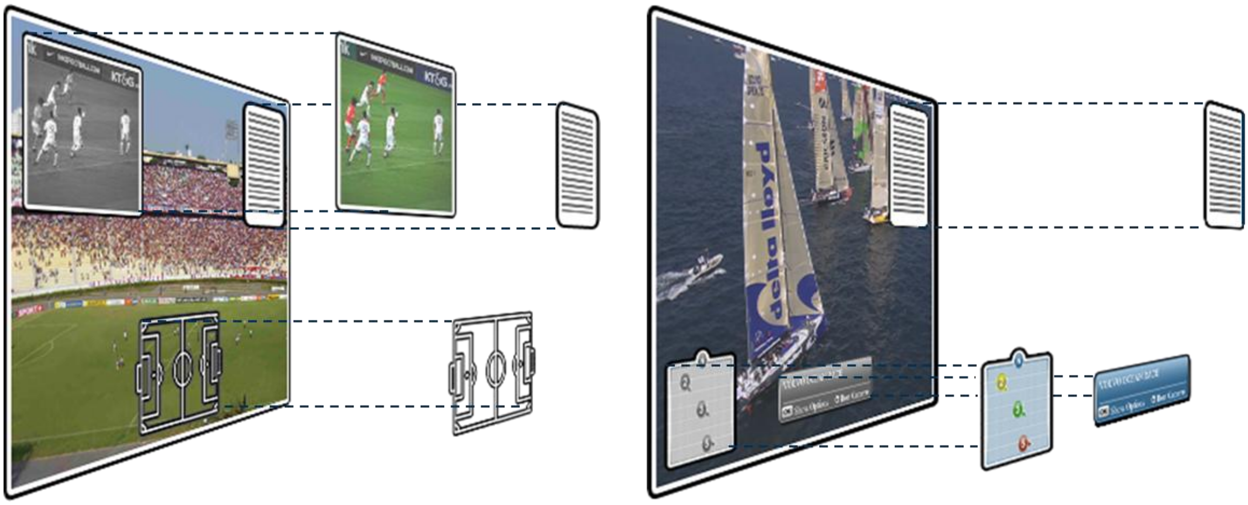


Figure 31. Content Interactive Widget concept overview.

Another valuable property of the widget format is the independence of the different elements. This would allow for a service to contain a full set of features even though not all of them could be active at the same time, both due to design but more importantly due to technical constraints. The user would not be required to own the hardware needed to use all the different parts of the service to be able to purchase it. All widgets that do not require that latest top end device can still be run on older hardware. It could even be possible to supply different versions of the same feature that each requires different hardware. This way there would be a high quality version for top end STBs and a less detailed version for the regular user. Some features could additionally be used by multiple services giving a great way to re-use technology.

Again and again during the thesis feedback pointed out that the properties of the prototypes really displayed a strong power. Just by demonstrating the prototype of the Radar any audience would see how the format could be used in different contexts giving them a richer experience. It felt necessary to explore these properties and generalize them into a format and the result of that was CIWs. The different parts and properties of CIWs each have advantages comparable to the current systems and together the potential for popular features and improved development seems rather clear.

In the search for improvements the thesis set out to find a model for future services on the IPTV platform. Using the feedback and results of the work process the concept of CIWs constitutes the goal for that search as a possible recipe for future services.

## 6.2 SVG

Moving from the more static HTML interface currently used by the IPTV solution to a dynamic graphic technology is recognized as a quite big step. The candidate process has been involved some heavy testing from Eriksson's side and the forecasted winner seems to be Scalable Vector Graphics or SVG. This thesis tried to look at both SVG in general to evaluate it as a basic user interface engine but also to see if it would be powerful enough to provide a strong platform for more advanced services in the future of IPTV.

The Radar prototype that was made using SVG managed to show opportunities but also weaknesses of the technology. Compared to its competitors SVG require small amounts of resources to run, something that fits well with IPTVs low hardware requirements. It can produce good looking animated interfaces without too much work and the vector graphic style works well with the television format. With its dynamic format it can possibly simplify customization and branding even further than the current interface.

The format is however not entirely without problems. Most of these problems are rooted in the limited resources available. In the current stage of IPTV one of the most limiting factors for new customers is the cost connected to the purchase of a new STB. Because of this IPTV is limited to relatively cheap hardware with low performance. With the interface system that has been present so far this has not been a problem. The introduction of SVG reveals this limitation.

### 6.2.1 SVG Tiny

The version of SVG that is the target for the IPTV systems is a smaller subset called SVG Tiny. With only limited functionality it is better suited for smaller platforms with more limited hardware such as the IPTV STBs.

This limited version does not pose any real problems in terms of available functionality. Most of the basic features are still available and the more advanced ones would most probably be out of reach resource wise anyway. The version differences are however not limited to extended features. Some rather elementary features have been left out of SVG Tiny, often because they can be done manually with use of more basic building blocks. An example of this is the circle shape. This shape element is not present in SVG Tiny and instead one has to use Bézier curves to manually create it.

As said the limiting factors with SVG Tiny does not limit what can be created but it greatly effects how it can be created. Most available editors for SVG support only the latest full SVG specification or parts of it. Such tools cannot be used to create content for SVG Tiny because they will use functionality that may not be supported such as with the circle example above. In some cases the editor can be used with restrictive use and others require modifications after creation. There are editors capable of producing content for alternative formats but none of them were available for testing during this thesis because of price and limited availability. Worth pointing out though is that SVG is still somewhat young and more editors will most probably follow in time.

### 6.2.2 Animation Quality

Another question is if the combination of SVG and the limited resources available to it will be enough to house more elaborate applications and services. With the Radar it was visible that more advanced applications could be used but they would have to be carefully tested and polished to reach a good enough quality. Up to now performance problems has been limited to loading times when changing state or channel but with the introduction of animation the bar for what is acceptable will be even higher since any performance drops will immediately be visible.

As mentioned earlier the standards for animations of this type today have reached extreme heights. Everyday applications on devices such as mobile phones and computers use animation to richen the graphics of almost all applications. If IPTV wants to adopt similar applications it would have to follow these standards as good as possible and doing that with the limited hardware available could be a real challenge. The Radar prototype shows very well what is possible but also what is required in terms of polish.

### 6.2.3 SVG Summary

Since the most problems with SVG are connected to the power of the hardware it should be expected that most of them will go away in time as hardware improves. That does not solve the problems with introducing a powerful platform today though so saying it will be solved in time is not an option. The way to proceed is not obvious in any way.

One alternative is to provide high performance STBs to early adopters of the platform and cheaper versions to the general public, each with their own version of the SVG services. This way the applications can be balanced with the hardware in mind and allow for demonstration of power and at the same time provide acceptable services to the larger groups. Services such as IPTVSports could help differentiate IPTV from its competitors while minimizing the risk of not reaching the required quality on the applications.

Another option is to limit the use of SVG and animation to more simple interfaces such as menus and static interface elements to improve the general quality of the IPTV platform. While losing out on possible services that can help in spreading the platform is allows for a more polished overall solution.

The second option is probably a good start and when the performance of the platform has been measured lighter services can be made and possibly supplied together with a high quality version.

All in all SVG seems like a good path to take. It has the power to bring IPTV forward but it might be forced to take smaller steps to get there until the full hardware to support it is available.

## 6.3 IPTV Portal

The Portal is the main tool for development of IPTV services for Ericsson. It has just recently reached its second major version release. What weaknesses does that release have in terms of serving the needs of the next generation of IPTV services?

### 6.3.1 Outsourcing service production

Aside from regular television services CIWs and similar products will be what differentiate the IPTV platform from its competitors. By providing better tools for their creation Ericsson will ensure that there will be plenty of services to offer to its customers.

Something emphasized by the parallel thesis work of Ezequiel Lisak [3] is the need for context know-how when creating services as intimate with the context as IPTVSports, a connection that is needed to serve the important Content Interaction. As such know-how usually exists outside of the platform developer office and creation of the services is usually done by outside actors. The Ericsson IPTV Portal is however mainly built to serve in-house development. The complexity involved in creating new applications that utilizes any newer features currently requires deep knowledge about the framework, something that at this point only Ericsson has.

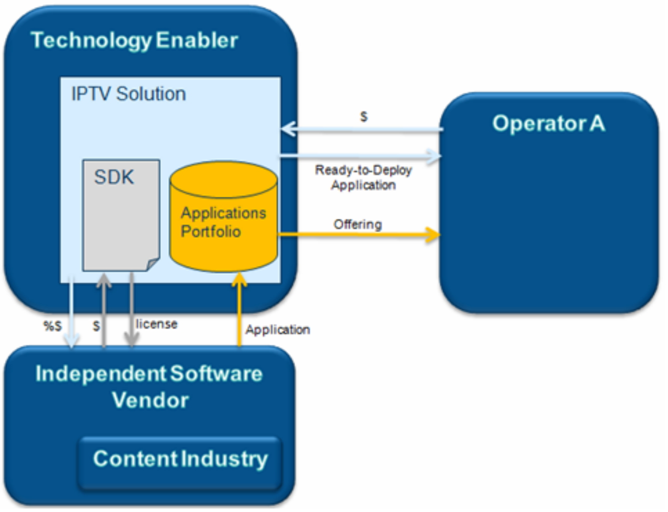


Figure 32. Proposed solution for outsourced service development.

If service development would be possible outside of Ericsson other companies could invest in the platform by developing services that Ericsson could preset as a service portfolio to the service providers. This way any service provider could create their own IPTV solution that fits their customers the best.

If Ericsson wants to continue to stay in front of the IPTV market it is critical that they support a wide range of services in the IPTV package. The first step to get there is to simplify the creation of services and possibly open up the framework for outside developers.

### 6.3.2 Service development

Effort has been made to try to make the IPTV framework as development friendly as possible with guidelines for an SDK-like interface between the modules. This has worked out fairly well compared to earlier versions but it still has not reached a level where someone could easily create and expand the Portal services with only documentations available. This is partially because the documentation for the new release not yet has been finalized. It is however not the sole reason.

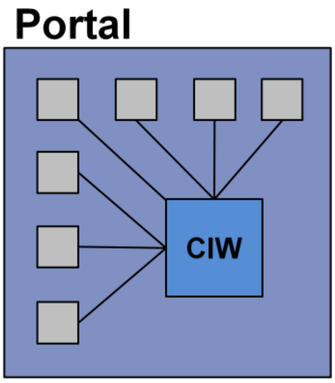


Figure 33. Current IPTV Portal design for complex services.

One of the problems with the current design of the Portal software is the module design. The design in itself has a large set of positive sides but one thing that works against it is the way it almost promotes creation of even more modules rather than grouping some together. What this means is that if logic is needed by some service the module providing that logic has to be requested and handled. If someone wants to create a more advanced module, such as the latest IPTV service one would need to request logic from a huge amount of modules. That in term requires knowledge about all those modules, something that drastically decreases the value of the SDK as you would need knowledge of most other modules anyhow. In the world of programming languages it would be like having a new library for each new function.

Together with the need for outside sources for creating more elaborate services this serves as a large obstacle for creating and serving new services on the Ericsson IPTV platform.

### 6.3.3 API module

With the latest version of the IPTV platform having just been released it would not be realistic to require thorough changes of the framework. Instead it would be interesting to see if there was a way to simplify the creation of more advanced modules using only the technology that is available.

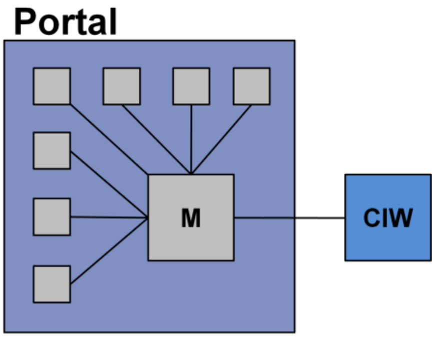


Figure 34. Proposed IPTV Portal design using an API module.

A way of solving this problem would be to create an API-module, a module that would focus on the collection and organization of useful logic. Instead of forcing new advanced services to have direct connections to all parts that serve their own piece of the logic a central module would present this logic by linking further in to the modules. This would mean that pretty much all available functionality could be found and accessed by just requiring that module.

Such a module could be created in several ways. One solution could be to have the module collect other modules implementing some API interface in a similar way that modules today implement menu interfaces to add themselves to that menu. This way the API module would not have to be updated with new functionality with added features.

A more controlled but demanding solution would be to create the new module and manually connect it to all relevant modules and provide their functionality. This would give more control to the organization of the API but would need more work with each added feature.

One of the major strengths with the module solution that is used in the framework is the way the same module can differ depending on the system it is run on. Collecting the functionality of the modules in an API could pose some problems with this design. What happens if a feature only exists on one platform? How can the API collect only features available on the system it is currently being run on?

Multiple versions of the API could be made using the same design; each providing only what can be used on the current system. One API module could be made to hold only the platform specific functionality and another for cross-platform. There could possibly even be ways to make the API module selectively require modules depending on the platform, maybe with separate interfaces for adding API functionality.

Even with limitations and possible problems a solution such as the API module could simplify the development. This would be effective for development of new services both for Ericsson and to make it possible to outsource creation of new services to parties that hold that context-specific knowledge needed to create powerful Content Interactive services.

### 6.3.4 Open SDK

The best way to design the IPTV framework to support development outside of Ericsson would of course be to have it in mind when developing the system. Instead of pasting SDK solutions on the design the design should support simple APIs from the start. Such design would allow for additional possibilities with using a better development interface.

One useful side of using an API for accessing functionality would be that the API could be accessible from several different security levels. Instead of having everything within the restricted system as it is being done today the API functions could be divided into areas, leaving secure functionality restricted to the Ericsson system.

Such separation would give service developers access to only the functionality that they need without risking problems with more delicate features like payment and account security. This would make it possible to outsource service development to parties that has the critical know-how in the context of the service.

Expanding even further on such a solution would open up for possible user created content. If the API is simple enough and secure it would be possible to let anyone create and design smaller applications. Similar services has been seen in a number of different areas but the largest one is probably the Apple iStore that offers user created applications to download or buy to their iPhones. It would not seem very unlikely that such a system could not be possible for the IPTV platform in the future.

With the current IPTV Portal one way to approach a more open development interface would be to make use of the global Javascript namespace. By selectively placing secure functionality in some API object in the document it would be possible to have services load outside of the Portal but still function using that API. Services would still have to be loaded from a service inside the portal so there would be some matter of control to it. There would of course be issues with security that would have to be looked over, for example the control over the DOM tree, but the general idea would be possible.

To enable user created services would greatly increase the power of the platform. If a system allowing such development grows in size it historically has a very good ability to keep customers for a long time. It is difficult for a new system to compete with not only a similar base system but with all added services as well. Additionally user-created services allow for the creation of content for smaller audiences instead of being forced to require a large enough potential user groups.

### 6.3.5 Widget Interface

Following the CIW format of future services it is apparent that interface elements work best in the way of widgets. When more widget services become available it will be necessary to have good control over where and how graphical elements are shown on the screen. If more than one widget would be run at the same time and both wants to occupy the same space of the screen something needs to intercept and correct. The same is true for the relationship between important elements of the video stream together with additional interface elements.

When a new service is created, and that service is made up out of a set of widgets one option for the developer is to create their own wrapper application, an application that handles all inputs, layouts and widget controls. While this solution gives all control to the developer it lacks the possibility of reacting to other widgets.

A much better option that could help with such problems would be to have a part of the IPTV framework be designed to handle just this type of widgets. Such a Widget Interface would work as a central hub for additional widget applications that would be developed. The interface module could serve with helper functions for input controls, graphical layout and future SVG functionality. By standardizing the building blocks of future services like this development would become a faster and cheaper process. Following the train of thought from the previous section a common interface like this could make development of new services even simpler both for commercial companies as well as possible individuals.

Concerning the arrangement of interface elements and input management for a service with several widgets active at the same time this thesis has suggested a solution similar to the window management systems of modern operating systems. Input would be directed only to one widget at a time, a widget selected by circulating focus amongst all active elements. In the same manner widget size and position could be changed to fixed values using standard SVG functionality. A solution like this would fit well into a general widget interface further simplifying future service development.

## 6.4 Future work

This thesis has been quite free in its restriction to what areas to work in. Because of that freedom the project has given an insight into many of the challenges that developers and designers will face while moving forward with the IPTV medium. Naturally time has limited the results to the area around the research questions and moved the work towards the more technical aspects of IPTV.

Another area that clearly will be relevant to the future of IPTV is the advancement of the interaction design for the new services on the platform. This thesis scratched the surface of a number of issues that has to be dealt with this area. How will the user control the functions that are available? With more services available at the same time how will the buttons on the controller suffice? Will the remote control that is used today be enough for a higher level of interaction?

Compared to other devices that are designed for a more active interaction such as video games and mobile phones the quality of the buttons and the reliability of the communication with other devices have a much higher standard.

Another interesting aspect to look for is the overall design of the IPTV experience. Today the IPTV design is mostly mimicking the one-dimensional channel system that has been used since TV was invented. Is this really the best way to go forward? With all the different levels of logic and smart electronic program guides should the TV still start off with showing you channel 1 instead help you find the program that you want to watch?

Finally one area that was not really mentioned in this thesis but should hold some value anyway is the nature of television broadcasting using IP networks. Should broadcasting be limited to the large actors such as the regular TV channels that we have today or could IPTV open up the possibility for anyone to air a show without anything more than a decent internet connection? Such development could be the largest evolutionary step possible for TV and considering the popularity of similar services already available on the internet such as YouTube it would surely be a popular feature.

# 7 Conclusions

This thesis set out to evaluate the current IPTV system provided by Ericsson in the view of future services designed for the IPTV platform. It was of interest to know if the solution was powerful enough for what may come and if not then what changes would be necessary. By producing a service showcasing the requirements on the platform and by taking means as to implement it strength and weakness could be noted.

By looking at present services and by freely brainstorming on what would be possible and popular a next generation IPTV service concept was produced. It demonstrates a number of factors that are required by new services to keep them an interesting and valuable asset to IPTV.

The most important part of newer services was found to be the strong connection between the service and the content it would be presented with. Without it the services would lose out on any power and uniqueness that the TV arena had to offer.

The second major aspect found in the proposed future service was the widget-like format of the individual elements of the service. The format proved to be good at presenting additional functionality without interrupting the main content or other added service features.

Content Interactive Widgets was presented as a concept to describe the core functionality of new services for the IPTV platform. The concept was introduced as a way to combine the results from trying to answer the sub question: *How could a next generation IPTV service be designed?*

Two major user values were implemented as prototypes to evaluate the development platform but at the same time demonstrate the power of the service as a promotional tool of the possibilities of IPTV. The FurtherInfo Radar was produced to demonstrate a way of displaying complex information in a graphical format and by that increase the overall experience. MultiviewTV demonstrated the use of additional devices together with the main screen as well as new level for the user to decide more thoroughly what content that is shown.

The use of Scalable Vector Graphics for user interface elements with IPTV was evaluated. There were some clear restrictions on its use due to performance requirements and the availability of tools but the end result of the experience it would offer qualified its use. The technology was also deemed likely to improve with time.

The development of the prototypes using Ericsson's current IPTV platform revealed some important problems. The thesis present some solutions for how to solve these problems following the main research question: *What improvements can be done to the Ericsson IPTV platform?*

The design process for the new service and the research by fellow student Ezequiel Lisak showed that future services are likely to be done by outside parties rather than Ericsson. The current system does however not support this and houses complexity that requires in-house development of more advanced services. The thesis presents some ways to make the IPTV system more accessible for parties other than Ericsson.

Part of the problem with the current platform is the wide spread of source functionality in the system. The simple solution to this problem was proposed as to produce an API module that ties together functionality from other modules in an effort to present them to the developer in a more organized manner. With some different options in implementation the solution offers ways to solve the problem, however not entirely without issues.

A more thorough solution was also presented that would give additional value to the system. By redesigning larger parts of the platform to support multiple layers of development it would be possible to allow outside sources to provide production of services. Such outside sources could be anything from trusted companies to possibly even individual developers and users. With a larger cost the solution is less likely to be considered for this system but with the improved benefits it would clearly be interesting for future versions.

In addition to the requirement for outsourced development the system would need an improved support for widget-like systems. As more and more services become available a need for system supported organization of available service elements is apparent. All graphical entities will be fighting over a limited screen area and with the growing need for interaction with the different services input management become necessary.

A system for widget management could be a possible addition both for future systems and as a new element of the current IPTV system.

The future of the IPTV platform is filled with interesting problems and opportunities. Today the focus of the platforms is on providing the basic functionality and support for major features but if the system is to revolutionize TV as we see it today it will need to be improved into something more.

The future of IPTV services provided in this thesis may or may not be correct but it showcases some of the problems and requirements that are present with added interaction and complexity with the experience. By supporting development of a wide range of services Ericsson will ensure that the platform can support whatever the future brings.

## References

1. Internet Protocol Television, *[http://en.wikipedia.org/wiki/iptv]*, April 2009.

2. Ericsson - IPTV – moving into the new television era, *[http://www.ericsson.com/ericsson/press/facts_figures/doc/iptv.pdf]*, April 2009.

3. Lisak Ezequiel, *Business Development for Internet Protocol Television Services*, Chalmers University of Technology, Division of Innovation Engineering and Management, December 2008.

4. Triple play, [*http://en.wikipedia.org/wiki/Triple_play_(telecommunications)*], April 2009.

5. Digital Television, [*http://en.wikipedia.org/wiki/Digital_television*], April 2009.

6. Video on Demand, [*http://en.wikipedia.org/wiki/Video_on_demand*], April 2009.

7. Ericsson Televisionary campaign, [*http://www.ericsson.com/campaign/televisionary*], April 2009.

8. Jones, J.C., Design Methods, 2nd Edition, John Wiley & Sons, New York, 1992.

9. Ideation links, June 2008.
   Televisionary Campaign [*http://www.ericsson.com/campaign/televisionary*]
   Telco IPTV View, [http://telcotv-view.blogspot.com/]
   Microsoft Mediaroom, [http://www.microsoftmediaroom.com/]
   Experiments in Digital Television, Sanford university,[*http://graphics.stanford.edu/courses/cs448a-00-winter*
   */projects.html*]
   Investigation of IMS in an IPTV Context, [*http://www.attentec.se/documents/Investigation_of_IMS_in_an_IPTV_Context.pdf*]

10. Blomberg, Mia, *Interactive IPTV applications - Development of Interactive Applications for IPTV and making System Processes User-Centered*, Chalmers University of Technology, June 2008.

11. Volvo Ocean Race, [*http://www.volvooceanrace.org/*], April 2009.

12. Cooper, Alan & Reimann, Robert, *About Face 2.0 – The Essentials of Interaction Design,* Wiley Publishing, Inc., 2003.

13. PLAYSTATION 3, [*http://uk.playstation.com/ps3/*], April 2009.

14. Microsoft Windows Alt-Tab functionality, [*http://en.wikipedia.org/wiki/Alt-Tab*], April 2009.

15. *Ericsson Internal.*

16. Electronic Arts, Fifa 2009, Screenshot, [*http://fifa09.ea.com/images/screenshots/173651-img0029%20copy.jpg*], April 2009.

17. Volvo Ocean Race, Information on In-port races, [*http://www.volvooceanrace.org/about-the-race/inport-racing/*], April 2009.

18. Broadband internet, Asymmetric relation on download / upload rate, [*http://en.wikipedia.org/wiki/Broadband_Internet_access*], April 2009.

19. TRACAB, Image tracking solutions, [*http://www.tracab.com/*], April 2009.

20. Java Servlets, [*http://en.wikipedia.org/wiki/Java_servlet*], April 2009.

21. Scalable Vector Graphics, [*http://en.wikipedia.org/wiki/Scalable_Vector_Graphics*], April 2009.

22. Inkscape, Open source SVG editor, [*http://www.inkscape.org/*], April 2009.

23. Ekioh Browser, STB web browser with SVG support, [*http://www.ekioh.com/browser.html*], April 2009.

24. Bézier curves, [*http://en.wikipedia.org/wiki/Bezier_curve*], April 2009.

25. Motorola, [*http://www.motorola.com/*], April 2009.

26. Picture in Picture, [*http://en.wikipedia.org/wiki/Picture-in-picture*], April 2009.

*27. Ericsson Internal.*

# Appendix

The following images show concepts that were produced to illustrate some of the ideas that came up during the design process of the prototypes. Due to the fact that interaction of this magnitude is something fairly new to the television platform some changes are necessary to simplify for the users. These concepts are to be taken as suggestions for how to approach the problem.

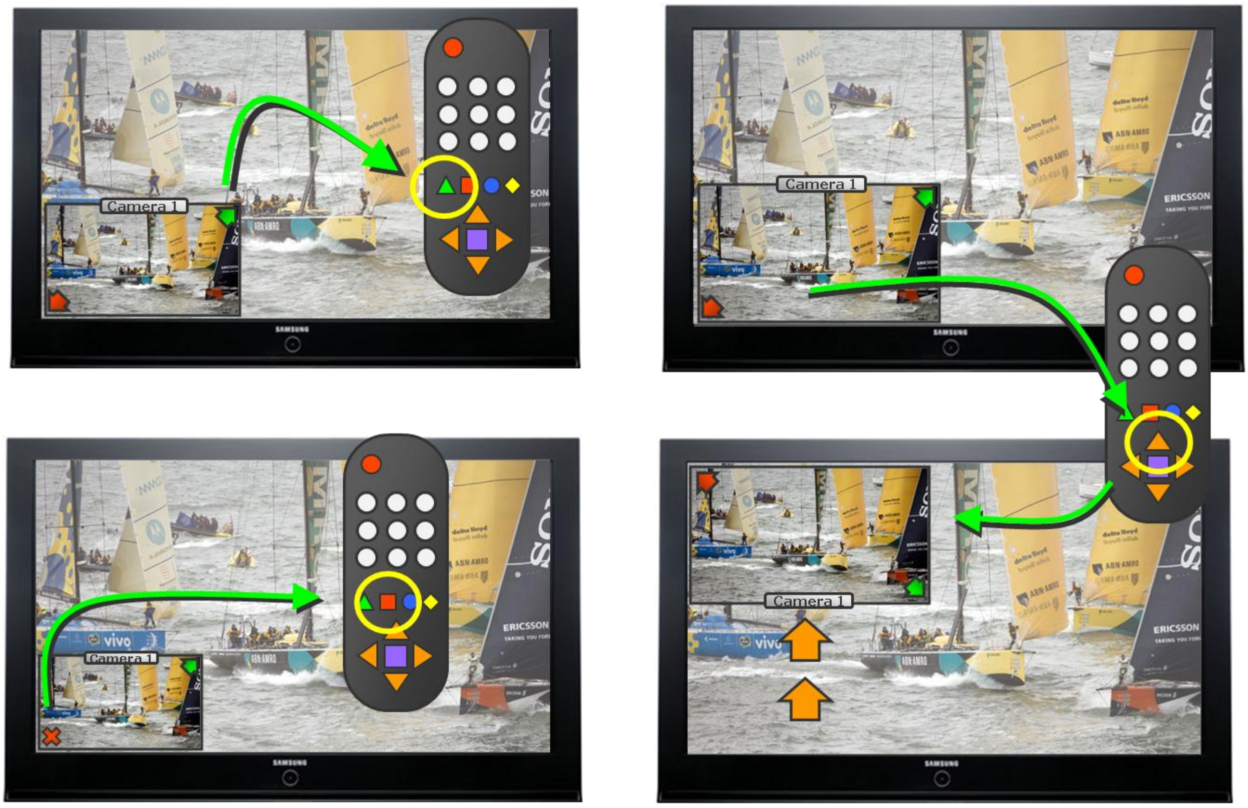


Figure 35. Picture in picture manipulation.

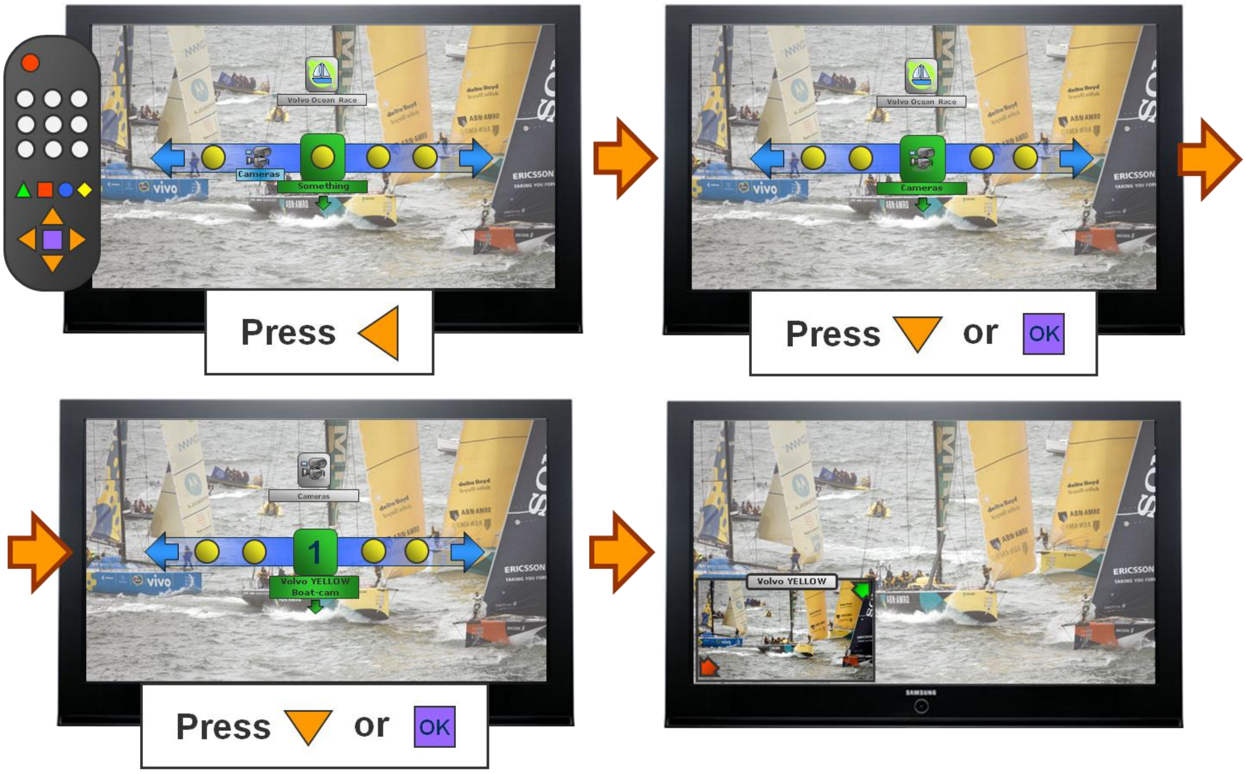
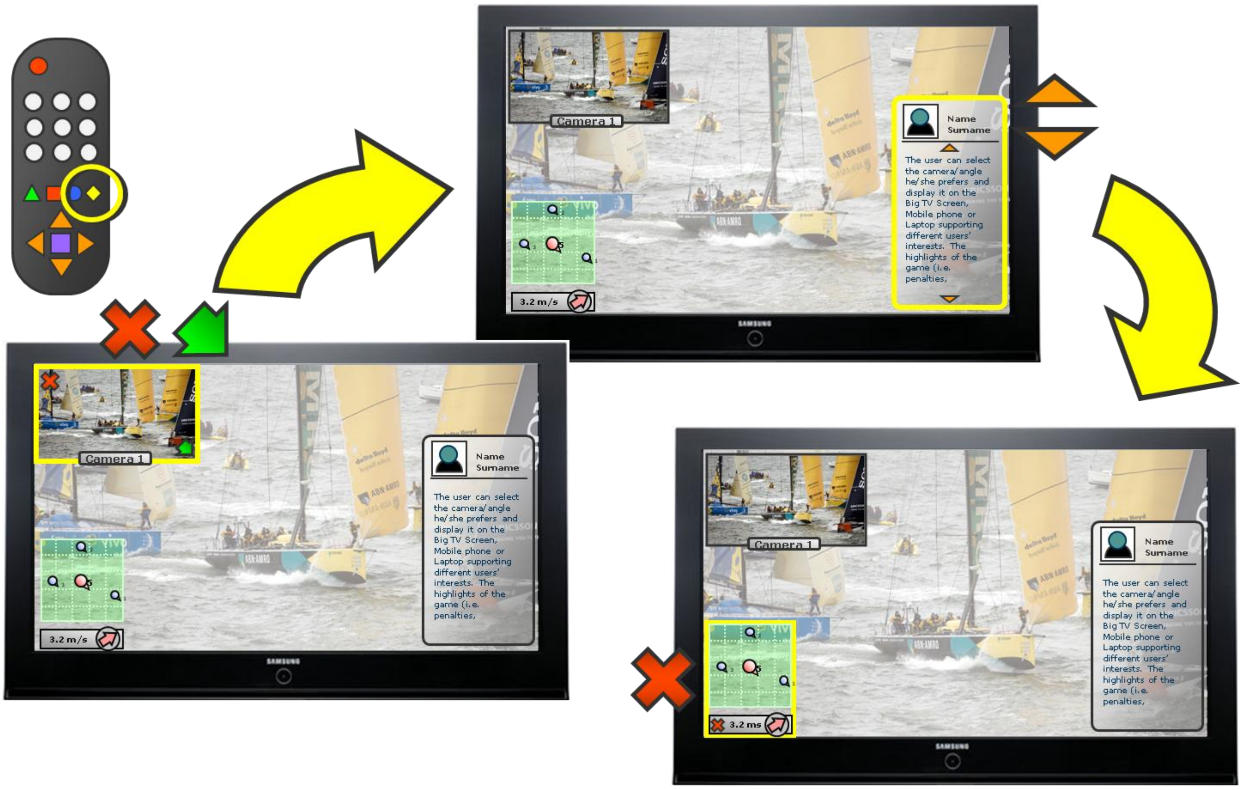


Figure 36. Menu navigation



The user can select the camera/angle he/she prefers and display it on the Big TV Screen, Mobile phone or Laptop supporting different users' interests. The highlights of the game (i.e. penalties,

Figure 37. Window focus and control

The following two images illustrate the final architecture of the two prototypes made during the thesis.

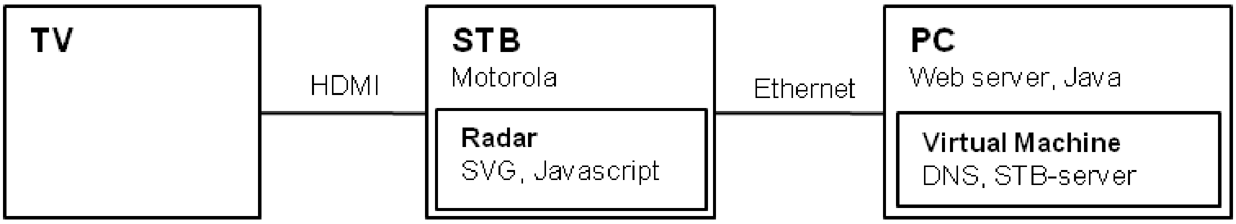


Figure 38. Radar prototype architecture

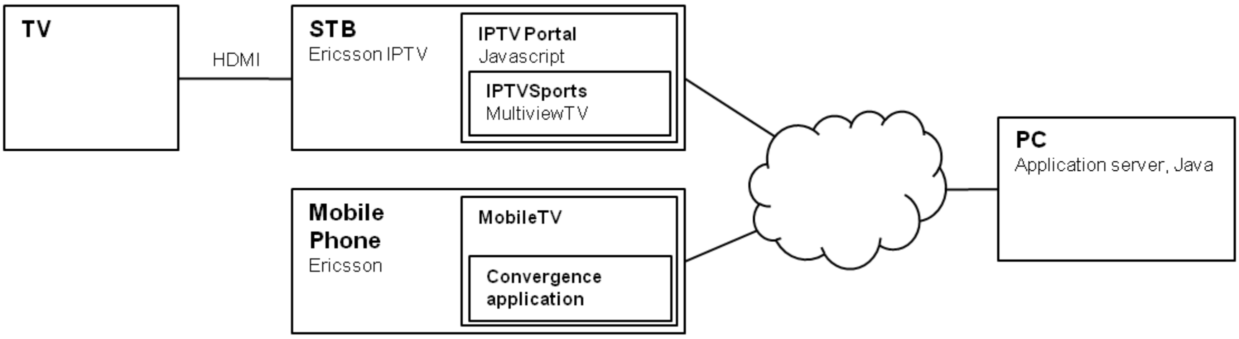


Figure 39. MultiviewTV prototype architecture

## Author

Petter A. Magnusson

M. Sc. Computer Science and Engineering

CHALMERS UNIVERSITY OF TECHNOLOGY

Contact Information:

petter.a.magnusson@gmail.com

+46 702 334 813