Television meets Facebook: Social Networking Through Consumer Electronics

Mariana Cristina Baca Del Rosario
S. B. Computer Science, Massachusetts Institute of Technology, 2006
S. B. Mathematics, Massachusetts Institute of Technology, 2006

Submitted to the Program in Media Arts and Sciences,
School of Architecture and Planning
in partial fulfillment of the requirements for the degree of
Master of Science
at the
Massachusetts Institute of Technology
September 2008

Author
Program in Media Arts and Sciences,
School of Architecture and Planning
May 09, 2008

Certified by
Henry Holtzman
Chief Knowledge Officer, Research Scientist, Media Laboratory
Thesis Supervisor

Accepted by
Deb Roy
Chairman, Academic Program in Media Arts and Sciences

Television meets Facebook: Social Networking Through Consumer Electronics

Mariana Cristina Baca Del Rosario

Submitted to the Program in Media Arts and Sciences,
School of Architecture and Planning
on May 09, 2008, in partial fulfillment of the
requirements for the degree of

Master of Science

Abstract

This thesis explores how the merging of ubiquitous consumer electronics and the sociable web improve the user experience of these devices, increase the functionality of both, and help distribute content in a more sociable way. The project will consist of one well fleshed out principal application: a digital video recorder (manufactured by Motorola, running OCAP) connected to the Facebook social network. By connecting these two technologies, the user can now automatically watch the shows her friends like and are willing to report to the social network; in return, the user transmits her viewing data back to the social network.

Thesis Supervisor: Henry Holtzman
Title: Chief Knowledge Officer, Research Scientist, Media Laboratory

Television meets Facebook: Social Networking Through Consumer Electronics

Mariana Cristina Baca Del Rosario

Thesis Advisor
Henry Holtzman
Chief Knowledge Officer, Research Scientist
Program in Media Arts and Sciences,
School of Architecture and Planning

Television meets Facebook: Social Networking Through Consumer Electronics

Mariana Cristina Baca Del Rosario

Thesis Reader
Judith Donath
Associate Professor of Media Arts and Sciences
Program in Media Arts and Sciences,
School of Architecture and Planning

Television meets Facebook: Social Networking Through Consumer Electronics

Mariana Cristina Baca Del Rosario

Thesis Reader
John Maeda
E. Rudge and Nancy Allen Professor of Media Arts and Sciences
Program in Media Arts and Sciences,
School of Architecture and Planning

Television meets Facebook: Social Networking Through Consumer Electronics

Mariana Cristina Baca Del Rosario

Thesis Reader
David P. Reed
Adjunct Professor
Program in Media Arts and Sciences,
School of Architecture and Planning

Acknowledgments

Henry Holtzman: For being a great advisor, always willing to help, brainstorm, assist, and motivate my work. You have been an amazing help the last couple of years.

John Maeda: For initially attracting me to the Media Lab, for being a great UROP advisor while I was an undergraduate, for ordering great food at the Royal East, and always encouraging us to work harder.

Judith Donath and David Reed: Thank you for all your insights, suggestions and motivation over the course of the thesis. You have been great help.

Marie-Jose Monpetit and everyone at Motorola: Thank you for the sponsorship, your technical expertise and access to hardware. Your help was essential to the completion of this project.

Sanghoon Lee and Matt Hirsch: Thanks for the random technical help and general brainstorming ideas. You were great groupmates.

Takashi Okamoto, Kyle Buza, Luis Blackaller, Kate Hollenbach and Nozomi Kakiuchi: Thanks for always being there to help keep the PLW amusing and the providing company for drinks at the Thristy Ear. And as always, a willing ear for late night rantings. Good luck with your future endeavors.

Diego Baca: Thanks for always being there to chat and being a great cheerleader for my project. And for being on all around great brother.

Jose Chaparro: Your dedication to radio and television knows no bounds.

Mike Brown: Sorry the cat is so fat and stupid these days. Thanks for listening to my late night gripings.

Matt Lockhart: For reminding me that some people don't use Facebook. Thank you for putting up with me when I was horribly busy and sleepy deprived. You light up my day. Thank you.

Mom and Dad: Sorry my phone was never charged, at least this kept me at lab often enough for you to call me there. Without you, I would never have made it this far. Thank you, I love you.

for my father and my mother: searching for justice since february 2003

Contents

List of Figures

Chapter 1

Introduction

> *The most merciful thing in the world, I think, is the inability of the human mind to correlate all its contents.*
>
> *- H.P. Lovecraft, The Call of Cthulhu*

Consumer electronics with their increasing computing power and connectivity are lacking good ways of integrating themselves with social networking applications and the flexibility of user-focused innovation characteristic of the web. At the same time, tons of personal user data, especially through RRS feeds–continuous, not immediately useful data—is flowing out of social networking applications at a rate faster than the average user can consume it, much less transmit it to his daily life. Social networking applications do very well in augmenting many forms of social interaction, but they are also an untapped resource in many ways, because they fail at transmitting this personal peer data to the user in a meaningful way.

This thesis aims at uniting these two domains through one exemplary embodiment of this concept: Facebook meets Television—by uniting a DVR to a social network, this thesis aims at trying to make consumer electronics easier to use and more connected, use up this personal user data in ways that enhance peer recommendation and the overall feel of a social networking application, and add new venues of media transmission in the process, so that the user can find new media, not at random, but in relation to the feel of the social network and his peers. This project will be evaluated for its success in these three areas once complete.

1.1 Motivation

This thesis originated from three important research concepts: a focus on the next generation of consumer electronics (CE 2.0), an eye towards using user-generated personal data from social networking applications in a simple yet productive way, and looking at new ways in which the ideas of viral marketing and word of mouth can be applied with new technologies. This project was executed in collaboration with Motorola, so the particular focus in using television, as opposed to other forms of media for the exemplary embodiment of this thesis came from their collaboration and support. The idea evolved from a desire to incorporate everyday consumer electronics into the Web 2.0 idea of Mashups. "Mashups are an exciting genre of interactive Web applications that draw upon content retrieved from external data sources to create entirely new and innovative services." [18] From that idea, the concept of joining ethernet-enabled consumer electronics transformed into Facebook meets Television.

1.1.1 Consumer Electronics 2.0

CE 2.0 is an initiative started at the Media Lab in order to bring several Web 2.0 design concepts into the consumer electronics world. [11] Some of the main design concepts of CE 2.0 are: awareness, simplicity, connectivity, interoperability and privacy. This project tries to explore some, if not all, of these themes as a use case of the CE 2.0 concepts.

- In terms of awareness, this project enables the television set top box to be aware and discover your friend's favorite shows and learn them.

- In terms of simplicity, there are three areas covered by CE 2.0, and this project touches on all of them: presentation, structure, and information. In terms of presentation, the project is simple in providing most of the information the user needs through the social networking application and only very specific data through the set-top box. In terms of structure, most of the setup and veto process happens in the Facebook application, and only the viewing and rating of shows happens on the television environment. In terms of information, the specificity of information is tailored to the user's specific needs and filtered through the social network, not the television.

* In terms of privacy, the system operates strictly on an opt-in basis, using only data explicitly provided by users when sharing and mining data, although it uses private user data to enhance personalized results.

* In terms of interconnectivity and interoperability it is providing a platform for connecting everyday devices like a television to existing social networking sites. It does not attempt to recreate existing structures or new devices, but plays nicely with current TV set top boxes and social networks. Set up is also simple and happens exclusively outside the television.

1.1.2 Tacit Data

Every day, more people are joining social networking applications and linking themselves with other friends, acquaintances and colleagues. As a result of subsequent postings, rss feeds, and uploads, each user is bombarded with an exponential torrent of data. This information, which I term in the thesis *tacit data* rushes past the average user, neither read, recorded, or collected in any meaningful way, except intermittently. Tacit data is data that is understood to be part of the implicit knowledge of the social network, but it is not easily catalogued or accessible to any individual user. Examples of this sort of data is Twitter feeds, which inform users of their networks daily thoughts and activities, user profiles on Facebook, even rss feeds from personal blogs and social networks. This data, although it increases the knowledge of the individuals in the social network, is not often used in an explicit fashion by members of the social network. It mostly exists as implicit knowledge, that a user is a aware that their friends have profiles and personal daily feeds, but often the users do not have a direct way of using and managing this large body of data.

On the other hand, users are also becoming more strongly connected, not just through increasing links in social networking sites, but through pervasive consumer electronics that are connected to the internet, yet provide very little in the way of social interaction in a passive sense. The result is that users are not augmenting their direct interpersonal relationships with this data with more passive, network oriented interactions, but with a tide of unsorted, unfiltered waves of tacit media and data. Although the social network application as a whole is valuable to the user, this particular aspect of the social network—feeds of personal user data—is not often being used in a tangible, explicit way by the user, just consumed implicitly. What is needed is some way of sorting and filtering the data from social networking applications into a condensed form and delivering it straight to their relevant consumer electronics, allowing the individual more time to follow up on meaningful data, not waste time with absent social interaction, and provide a direct connection between what is happening in his social network and his everyday life.

Some examples of this problem can be easily illustrated in the case of cellphones, directions, and television. One common phenomenon happens when friends post or change cellphone numbers on a social networking site. The new number might be emailed to you, or even appear on your friend's page, but will not migrate into your cellphone when you are stranded somewhere. Similar things happen on the way to parties or meetings, despite having mapping software on handheld devices and cars, the addresses are rarely transported without advanced planning. Finally, in the area of personal entertainment, your friend's latest update on their favorite television shows or favorite new movies have no way of making it from their profile page to your automatic video recorder without the user manually entering each one. For people with online acquaintances in the hundreds, this task can be herculean, and probably not worth it if it is attempted by hand. As a result, that data sits unused (except by advertisers, of course).

1.1.3 Media Sharing

There are two main motivations behind the concept of media sharing in this project: how does information spread virally and how do people react knowing they might be a source of media information. Increasingly, because of the veritable flood of advertising, users are relying more and more on direct peer recommendations instead of traditional advertising. Because of that, new ways of spreading peer recommendations must be found to ease the time burden on consumers. Learning about these patterns of information transmission, tracking the path of peer recommendations, frequency of similar posts could be a great help to both users seeking new media as well as content producers and distributers seeking to find out both the potential impact of a new product as well as how well received it has been. This system is designed to aid in the analysis of both these patterns of behavior.

The other aspect of media sharing that needs to be considered is privacy and user patterns in selective data sharing. Many data mining initiatives leave users feeling exposed and reluctant to opt-in. The motivation of this project is to create a system that enables sharing but at the same time anonymizes individual responses, allows users to opt in rather than have to opt out by default, and only automatically mines the users private data to enhance their own profiles, not anyone else's. All these features provide a direction in media sharing without sacrificing privacy.

1.2 Evaluation

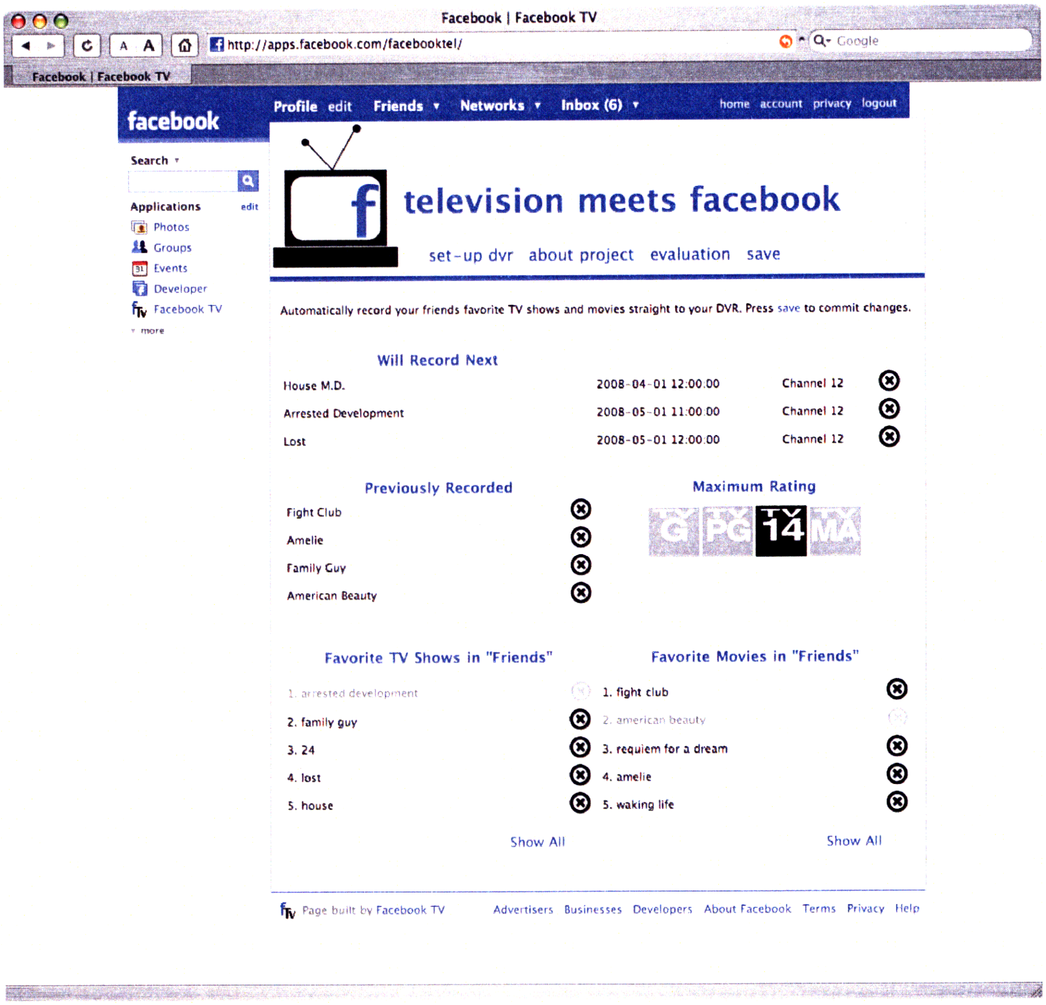
The success of this thesis will be evaluated based on three main goals. The first goal is technical implementation: whether the idea presented can be applied as a physical demonstration product and whether test users can interact with the object. The second goal of the project is to demonstrate the use of this type of project to disseminate media in an automated fashion through social networking applications. The last goal is to show how such a product would enhance—not replace—social interactions formed by social networking applications by reducing work and improving social communications between its members. Additionally, this thesis is meant to be educational, highlighting how the concepts presented can be applied in future directions of technological innovation while applying the conclusions of this thesis.

As a technological object, the success of the project can be measured in two domains: does it work sufficiently well as to be able to test the product with real users and does the implementation follow Consumer Electronic 2.0 principles. In achieving both those goals, the project can demonstrate the viability of the CE 2.0 idea by being an embodiment of the idea that can be tested, and it allows for user testing to determine whether the other goals were satisfied.

The second goal is to demonstrate the viability of such a product to disseminate media in an automated fashion. The goal is not simply to allow users to consume more media; the goal is to be able to automate the process of peer recommendations in a way that is helpful to users. Users are flooded with data from all sources, and it is difficult for an individual user to sort through the noise in order to grab what data he needs, for whatever purpose. By automating data collection, the goal is to create something useful in the realm of media dissemination. This can be evaluated by seeing how much initial volume is narrowed down, and whether the results collected generate valuable information to a user.

The third goal is to enhance social interaction using this system. Not just any type of media is collected, specifically television and film recommendations are collated. The reason is because much of modern social cues and interaction relate to how people in a social network respond in conjunction to these social cues and media information. However, due to the overwhelming amount of data coming from large social networks—especially through social networking applications—tools which help the user narrow down this data will improve the value of his interactions within a social networking application and outside of it. The value of distilling this realm of social interaction will be evaluated through a qualitative interview of users using the system for a longer period to time. Through the impressions of the users, it can be determined whether the thesis is of social relevance to an end user.

As a didactic tool, this project endeavors to show that connecting social networking applications to consumer electronics is a viable concept technically, socially, and


Facebook | Facebook TV
http://apps.facebook.com/facebooktel/
Google
Facebook | Facebook TV
facebook
Profile edit Friends Networks Inbox (6)
home account privacy logout
Search
Applications edit
Photos
Groups
Events
Developer
Facebook TV
more
television meets facebook
set-up dvr about project evaluation save
G PG 14 MA
Page built by Facebook TV
Advertisers Businesses Developers About Facebook Terms Privacy Help

Automatically record your friends favorite TV shows and movies straight to your DVR. Press save to commit changes.

Will Record Next

House M.D.	2008-04-01 12:00:00	Channel 12
Arrested Development	2008-05-01 11:00:00	Channel 12
Lost	2008-05-01 12:00:00	Channel 12

Previously Recorded

Fight Club
Amelie
Family Guy
American Beauty

Maximum Rating

Favorite TV Shows in "Friends"

1. arrested development
2. family guy
3. 24
4. lost
5. house

Show All

Favorite Movies in "Friends"

1. fight club
2. american beauty
3. requiem for a dream
4. amelie
5. waking life

Show All

Figure 1-1: Facebook Application

practically. Since it is doable, enhances social communication and is a time saving idea, it is to great use to innovators to know how to apply these concepts to future innovations. Ideas of how the conclusions arrived at by this thesis can be extended into other realms are mentioned in order to instruct others as to how to apply these ideas.

1.3 Contribution

Television meets Facebook aims to contribute in three primary ways:

- As a novel application on a social network that both aggregates different user preferences and delivers them straight to a consumer electronics product.

* As a new way to provide recommendations to users by means of mining tacit data from a social network.

* As an example of a consumer electronics device that automatically connects to a social networking application.

Although some of these aspects have been touched in partial ways by previous projects, this thesis provides a unique perspective on all of them. Later, in Chapter 2, examples of previous similar projects will be referenced and the ways in which Television meets Facebook contrasts with other work in the field will be highlighted.

1.4 Structure of the Thesis

This thesis is divided into 5 sections: introduction, background, implementation, evaluation and conclusion. The Background section is divided into two parts, detailing both the social aspects of television and social networking as a way of sharing media. The implementation section is divided into three parts: the making of the Facebook application, the motorola OCAP digital video recorder interface, and the communication protocol between the two. The Evaluation section details four different evaluation procedures: the completion of the project, the effectiveness of the UI of the Facebook application, the capabilities for data collection provided by the application, and a longer term study evaluation the whole system. It also reviews the results of the experiments and proposes modifications for the existing system as well as drawing conclusions about the effectiveness of the idea as a whole. The conclusion wraps up the scope of the thesis and proposes future directions for the idea. A previous project, Open I/O, that led to the development of this idea, is included in the appendix as a reference.

Chapter 2

Background

Like all those possessing a library, Aurelian was aware that he was guilty of not knowing his in its entirety.

- Jorge Luis Borges, The Theologians

This section will cover two main topics: studies and projects done on the social aspects of television, and how social networking applications have been used to share media and recommendations. In the realm of social television, the inherent social aspects of watching television will be explored, as well as specific attempts to add social dimensions to the television watching experience. Social Networking applications as media sharing applications have also progressed far by including intelligent, adapting methods of proposing new recommendations, as well as viewing the issues of privacy and social awareness when sharing media.

2.1 Web-Based Social Networking Media Sharing

At the core of this thesis is the idea of a social networking application as a means of gathering data. This idea at first glance seems to be no different than the general goal of the internet: to serve as a repository as well as a sieve for collective intelligence. However, in using specifically a social network as the source material of the endeavor—with social networking sites such as Facebook as the medium through which this data is gathered—one runs into three main problems. The first is the issue of privacy, and how does one protect the data one wants to share from the data one does not want to share– and to whom that data is available to. The second issue is once the user is aware of the limits of his privacy, is what does the data convey about him personally—and if that is the image he wishes to portray of himself. The third, is once the data is deemed suitable to be public for the specific audience, how can one sort through the data to select what is worthwhile and what is not? These three themes come to light within the technical work of this project, but they have also been explored at length by previous researchers in the field. This section will examine the current research in each of these three areas of concern relating to social networks and how the project offers a unique perspective on each of these ideas.

2.1.1 Privacy

Privacy when sharing data online has always been a perennial topic of online communities. As well as being one of the criteria mentioned in CE 2.0, Facebook meets Television wishes to guard a user's privacy despite sharing large amount of data among friends. The reasons for this revolve mostly around the fact that if a user feels threatened or invaded they are much less likely to use the system and share the data in the first place. The paper "Imagined Communities: Awareness, Information Sharing, and Privacy on the Facebook" by Acquisti and Gross deals with why people feel comfortable sharing personal data on social networks like Facebook despite having strong privacy concerns outside Facebook[1]. The study found no statistical difference between the concerns of privacy between members and non-members, for example. Additionally, members were willing to share highly personal information like cellphones, addresses and date of birth on Facebook, despite being in general concerned about privacy. The study links awareness of the rules of Facebook—those who were aware of who could or could not see their data were more likely to share data on Facebook. Additionally, it links the fine-tuned control Facebook gives a user over their privacy settings, saying that this control allows users to provide more personal information because of this. A surprising statistic is that data on Facebook is also more likely to be accurate than other social networks. This study was performed when Facebook was still linked to colleges and universities only, however. It would be interesting to see how this has changed since opening Facebook to the general public.

Facebook, the writers mention, is a location-based/network-based site. This increases the likelihood that the participants know each other in real life, and thus are probably more likely to give accurate information. Additionally, the networks of trust exist outside the realm of Facebook, so the user is more likely to trust their data to people based on their location.

The importance of this study is that it shows us what users find important, not only in terms of how to maintain their privacy, but in terms of what makes them more willing to share their data. Fine-tuned controls over what data gets shared makes users more likely to give that data freely out in the first place. This concept was implemented and explored in the thesis, in order to make sure all recommendations are based on user-provided data, not on automatically extracted data. Additionally, the data is more likely to be accurate, because the user trusts the data is only being used by their friends and the data is not taken from them against their will.

2.1.2 Public Image

Besides privacy, there is another aspect that a user might consider when choosing what to reveal about himself—namely, what sort of person does this or that set of social attributes say to others about who he is? One tangible example of how this scenario plays out is to observe the behaviors of users using iTunes music sharing. This feature in particular is very near to the goals of Facebook meets Television in that it delivers friend's preferences straight to one's music device. However, given the personal nature of one's music tastes, a user will be reluctant to share an accurate profile, or opt in at all, in many cases. The paper "Listening In"[31] covers many of the potential pitfalls of using such a system that make users uncomfortable. Additionally, in Donath's and Boyd's paper "Public Displays of Connection"[8], they show how one's public connections and preferences will modify the social interactions of the social network at large. In Facebook meets Television, this data might be relevant in analyzing if

Facebook, in how it displays data and connections, unduly influences what users will share about themselves with their friends.

In "Listening In," iTunes music sharing is analyzed for its good and bad qualities regarding music sharing directly onto a music playing device. iTunes share uses Apple's automatic discovery protocol Rendevous, to fine new users to share music with. This deviates from a standard social network site pattern of having to explicitly pick your friends. This leads people to be more conscious about what they share on such a network. Friends, although they might judge you on the contents of your music library, know you more intimately in other contexts, while random people in a dorm or workspace might only know each other based on their music libraries—making the importance of that first impression so much more important. At the same time, there are benefits to the iTunes system. It is not an anonymized system—unless the user gives a deliberately false name, so it works at building connections between different people who share the same taste in music. It does not filter results or give recommendations. Although similar in structure to Facebook meets Television, its goals are actually at odds. Facebook meets Television seeks to take people who are already friends, and improve that link by having them share their TV shows and movies—not in an individual fashion, but as a group be more aware of the social queues and shows that are talked about. Itunes music share is more likely to bring people who don't know each other—but share music tastes—together, because of the way it is designed. It also has a greater emphasis on the identity the user builds by what music he shares. Facebook meets Television only builds up impressions of one's friend's network—it only shows who you are by telling you what friends you keep.

In "Public Displays of Connection," the idea is presented that how one displays ones links to the community will serve as a measuring stick for the user's value and trustworthiness in the community. This is especially relevant for applications like Facebook meets Television, that rely on one's connections to others, and trusting them implicitly. Since the type of friendship is undifferentiated for the application, judging in absolute terms who one adds to one's friend's list will ultimately affect what recommendations one receives from the application. Spam in the form of recommendations is common in many social networks. To avoid that, many people are cautious about whom they add as friends—however, less linked friends might find it hard to break in without the existence of people who are more promiscuous in adding friends. These displays of connections help others establish who you are an whether you are whom you say you are—and this is ultimately how people will judge whether to add you to their list if they do not know you in real life. Given the actual consequences of adding friends that spam in an application like Facebook meets Television, these public displays of connection will become increasingly important as ways of verifying a user's identity.

2.1.3 Data Mining

The next challenge comes in once you have a social network with data that the user now wants to distill into a usable form. This project provides an outlet for the data being shared—a digital video recording device, a television. However, it also does the task of gathering this data and distilling it for the user. In doing so, one brings to light two issues regarding mining data from social networks: how does one improve the signal to noise ratio of data coming out of one's social network and how does one use that data effectively once one has it. Judith Donath covers these subjects in her paper "Signals in Social Supernets"[7], that explains the cost-benefit analysis of expanding one's social network in these supernets at the risk of getting less reliable data from them. This topic is of special importance given how closely tied Facebook meets Television is to a user's daily media consumption patterns—and a large ratio of noise or advertising manipulation on a user's data set can negatively affect a real object in the user's life. The second topic is covered in Shardanand's and Maes's paper on the Ringo Database [28], and how correlation-based algorithms can be used to greatly improve the effectiveness of peer recommendations. Facebook meets Television does not use any complicated algorithms to improve the quality of recommendations because of the scope of the project, nevertheless, the applicability to this project is mentioned for future iterations.

In "Signals in Social Supernets," the causes, pitfalls and benefits of the reliability of data in social networking sites is outlined. One of the main factors listed for less reliable connections is the ease by which one can add new friends to one's network and the relative cost of doing so. Some social networks make it relatively easy to add friends, but you don't get very much out of it. Others make it hard, but at a great incentive in private user data. Others make is easy but you either have to reveal much of your own data to get very little back, like in cases such as Livejournal, where a user can add people unilaterally, but in exchange the other user gets access to the first user's private posts—but not vice versa. Networks of trust are of value to build for a user seeking to gather data because user's acquaintances and friends are likely to have similar tastes and interests, and adding to one's network implies a certain level of trust in the information they are providing as well as what they will do with the user's data. These themes are of great importance to Facebook meets Television, since all these are the basic assumptions on which the system will be built: that users will share similar tastes, that the high barrier to entry will make it more likely that the other users are trustworthy informants, and that despite the size of the supernet, valuable data can be extracted without too much interference from bots. Since the information mining will be automatic, the reliability of the data becomes that much more important since the receiver will lose the link to its source, and will have to trust the social network as a whole.

Ringo is a database that collects user's music preferences and suggests new music to listen to based on the profiles of 2000 other users. There were a variety of different algorithms tested, but the unifying criteria for effectiveness was that whenever two users shared several opinions about different pieces of music, they were more likely to like the other recommendations they had listed. Each user needed to only provide a numerical rating for each artist or band, and yet, even with this limited data very accurate results could be obtained. In user profiles in SNSs, one is unlikely to find much more data than what movies or music one likes. Yet, with only these simple data points it would be possible to create intelligent recommendations based on this system; something that could be implemented on Facebook meets Television. Given the small scope of the project, complicated algorithms were not used. But the idea of allowing users to rate shows and improving future recommendations based on user-provided ratings was included, given how effective it has proven at culling out bad data and improving the signal to noise ratio of gathering useful recommendations from a large body of users.

2.1.4 Collaborative vs. Content Based Recommendations

Two main classes of user recommendations are being used in recommendation systems in web applications right now. One class are collaborative recommendation systems, which base recommendations on other users in the network who have similar tastes in the recommended medium. By analyzing users who rate different media similarly, the system can generate new recommendations based on that. The second type of recommendation system is content based recommendation systems, which instead looks at specific signals in a given medium, and looks for other media the user might like that is similar to previous items the user has rated highly. According to a recent paper by Adomavicius and Tuzhilin, both these methods have some similar limitations, including the new user problem—how to bootstrap the system since a new user would not have a body of recommendations to work from, overspecialization of recommendations, and a lack of contextual cues[2]. Some hybrid systems have also been proposed, as well. Sites like Amazon.com [3] use a hybrid method of content based recomendations and collaborative recommendations by looking not only at other products with similar keywords, or users who bought the same product, but also users that bought similar products and recommending based on shopping profiles as opposed to individual items[14]. These approaches have had great success in terms of new marketing but still lack many contextual clues of the social environment the user inhabits.

Adomavicius and Tuzhilin go to propose some alternate approaches that might have better success. Some mentioned include making the system less intrusive to new users by finding better ways of collecting initial information, using user demographics to improve recommendations, as well as other contextual clues, as well as feedback on the quality of recommendations given. Some of these proposed modifications are implemented in the Facebook meets Television project. For example, the system does not require the user to provide recommendations of their own initially, it only builds on the already established network of peer recommendations. It takes contextual clues into account by knowing to an extent not to take recommendations from the bulk of the social network, but only from friends. It also tries to be non-intrusive and provide feedback by allowing users to rate recommendations directly on their television after watching a particular show or movie.

2.2 Previous work in Social Television

The other main component of this thesis is consumer electronics—specifically, television. Television, despite its recent trend towards being a medium of isolation, was originally conceived as a social vehicle—like the cinema, but for the hearth. However, as the number of televisions increased, this idea of the television, as a centerpiece of conversation and of the living room, is quickly becoming a fiction. The challenges brought by the increasing devices that can be used to display television—computers, mobile phones, web feeds—cannot be ignored if we are to continue to see the television as a distinct element of social interaction. Researchers in this field have been striving to outline how is it that television used to function as a social vehicle, and how can we maintain those aspects as the technology advances. Additionally, proponents of interactive TV wonder how those elements can be transitioned into new forms of television distribution like mobile TV. Both these elements form a foundation piece on the new directions this thesis wishes to engage television viewers: by stating the ways in which dedicated television viewing is a distinct form of interaction and can be preserved as such, and by making it pervasive without compromising what makes it a television, as opposed to a different consumer electronic device.

2.2.1 Television as a Social Medium

Despite the recent trend towards thinking of television as an isolating experience, ethnographic studies have shown that television and mass media are vital forces in the forming of interpersonal relationships, and play a central role in how families and other social groups interact in a normal fashion [16]. However, nowadays circumstances conspire to make television an isolating experience—urban sprawl, travel, and time constraints make joint television watching inconvenient. Compounded by the new forms of interaction such as online computer games, which are social, location independent and interactive, television is increasingly a less social medium [25]. At the same time, this change in technology can also make many of these advances available on the television as well. With cable boxes such as TIVO integrating computing functionalities into the Digital Video Recorder, new forms of social interaction can be designed with the television in mind. Two directions can be taken in this field: the synchronous and the asynchronous route. Social TV is a new means of interaction for television that is envisioned for synchronous television watching.[20] Facebook meets Television is envisioned as an asynchronous medium. However, some of the insights and challenges of Social TV, like the type of communication TV engenders and the value of joint television watching form a basis for the work in this thesis.

Social Television is a synchronous means of attaining a joint watching experience by adding voice chat to multiple television screens. In a series of studies and surveys, this experiment revealed certain key insights into the social nature of television. Although this work is seminal on the idea of technology enhanced social television, it takes a drastically different tack than Facebook meets Television in that it does not use a social networking application at all, and it is real-time. The idea of voice-enhanced television is certainly an appealing one, but it is not without its limitations. It is limited by time constraints, just like many obstacles for joint television watching revolve around the hectic modern lifestyle. Also, voice can interfere with the enjoyment of the program and can be hard to understand when several people try to communicate at once. It is effective at conveying phatic responses—laughter, shock—and beginning the conversation about the television show, but at the same time it can be disruptive and hard to understand. Nevertheless, it provided key insights into the realm of social television through its user studies. The majority of the survey respondents mentioned large televisions as a centerpiece to what makes one want to view television as a social experience. This confirms the idea that television will always have a role in social media sharing despite the pervasiveness of it in other devices. Another is how despite television dominating the senses, users find ways of communicating with these limitations by following television queues like changes in volume and music. This showed how the interactions of those watching the show are tightly interwoven with how the show is structured—and watching shows in different mediums, such as in snippets, pausing, and changing the atmosphere of the television watching changes the communication that happens as a result of the watching of the show. They also proposed a couple of ways in which these interactions can be integrated asynchronously. One was to mine blogs and other internet postings about an episode and superimpose it with the show to add commentary to the episode. This is the sort of thing that could be integrated into a system like Facebook meets Television where user provided data on the internet is tightly coupled with a DVR. In the future, as such systems become more pervasive, such enhancements might be a key feature of social television interactions.

2.2.2 Pervasive Television

Pervasive Television both shows the obstacles and merits of transforming the fixed, living room television into a mobile, ubiquitous technology by displaying its own unique challenges. Pervasive television, especially, Mobile TV has an advantage when trying to make television social that conventional television devices do not—these devices already exist in a social medium, such as a phone, a PDA, a chat device. They do not fall into the common blunder of recreating a social network for the new application because they already contain a social networking application. Thus, the integration is far more seamless, in many ways. At the same time, things like Mobile TV suffer from the tangible deficiency that the experience of watching television is not the same around a tiny, battery-powered, public device. The media consumed is different, the enjoyment is different, and there is no way for five people to crowd around a cellphone to watch a movie. Given those two challenges, Mobile TV offers insights as to how to make conventional television again a social experience by integrating asynchronous communal watching experiences and the awareness of other watchers—even if they are not physically present—while maintaining what is good about it, like high bandwidth data and a relaxed environment.

Studies have shown that that the majority of people favor watching television in a social setting with friends over watching it alone. However, the fast-paced lifestyle of today prevents most users from routinely watching television communally. Mobile TV uses SMS communication as a means of making the viewing of television on small mobile devices more social.[24]. The aim is to provide multiple remote users with a joint watching experience [23]. The main way this is done is by creating an awareness of other users in two ways: by providing communication with other users and by providing a common link in content. In Mobile TV, this is done in a couple of ways via SMS. One way is called JointZapping, a means by which a user, watching one TV show can tell her friends to watch the same TV show via SMS, by synchronizing the content on the two separate receivers. This ensures that two users are watching the same content, albeit it demands a great deal of synchronous watching. The second way it provides an awareness of other users is by chat channels and messaging integrated into the content-watching experience. Users can talk while the show is going on, giving the illusion of living room conversation while watching content on a mobile device. Although these different means provide good awareness of others watching the same content, they lack a degree of asynchronicity that characterizes SMS devices.

The asynchronous dimension of Mobile TV provides more insights into things that can be integrated into conventional television. The first to be mentioned is Sharelink, which unlike JointZapping is simply a link to friends for later watching of the same content. Similarly, Mobile TV provides a platform for blogging about recommendations, and being able to rate content communally online. These sorts of elements are the ones that have been exploited by Facebook meets Television—how can we integrate these activities people are already doing, like sharing links, content and ratings online and feed it directly into the television to provide the viewer with an asynchronous sense that others are watching along with him? Mobile TV provides some directions and insights into how these principles can be applied into television. However, they are dealing with a very different sort of user interface with its own challenges and advantages—ease of chat yet difficultly in viewing full length content. By applying these principles in Facebook meets Television, the project can increase the social dimension of television in a way suited for its user interface.

Chapter 3

Implementation

I cannot walk through the suburbs in the solitude of the night without thinking that the night pleases us because it suppresses idle details, just as our memory does.

- Jorge Luis Borges, Labyrinths

Facebook meets television is one implementation of a greater idea: how do we get consumer electronics to interact seamlessly with the web without having to drastically redesign each product or website every time. The ideas behind this implementation could theoretically be used to develop APIs and standardized protocols for web to consumer electronics interactions. In the future, one envisions a world where third parties can create applications both in social networks as well as on consumer electronics to unite all sorts of different domains. An application on your television might grab your friend's favorite shows, your car knows where your friends live, your friends know if you are currently home or available based on the status of your home electronics, and your coffee machine might know how your out of town visitor likes his coffee in the morning. But before this can be accomplished, there needs to be a means for connecting these devices. Increasingly, manufacturers and service providers are providing some open standards and APIs for users to play with: DirecTV allows for a remote recording API [6] Tivo allows for non-TV third party applications on their hardware [29] cellphones of all sorts allow for new applications. Soon, applications

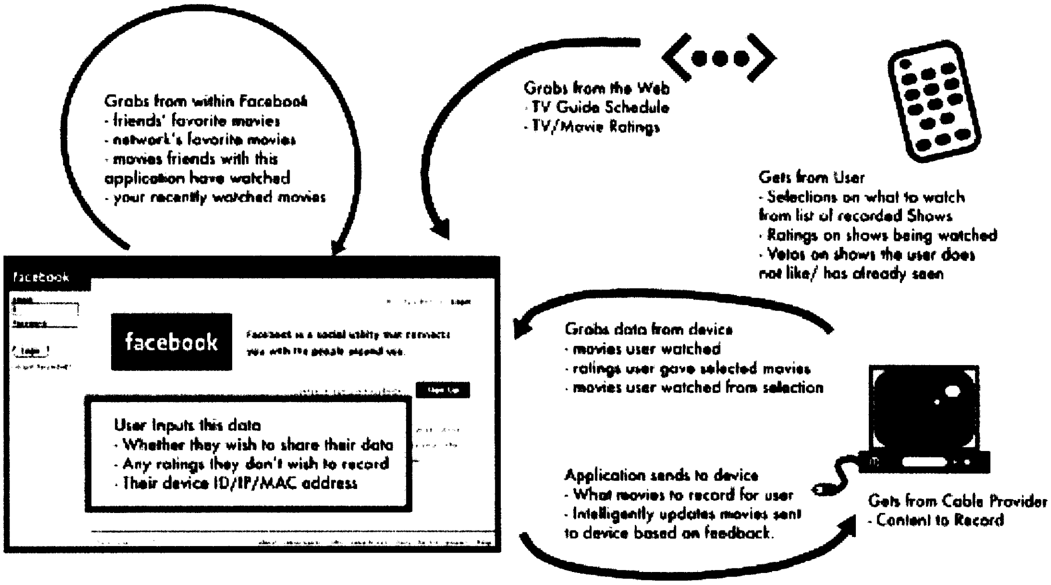

Grabs from within Facebook
- friends' favorite movies
- network's favorite movies
- movies friends with this application have watched
- your recently watched movies
facebook
facebook
User Inputs this data
- Whether they wish to share their data
- Any ratings they don't wish to record
- Their device ID/IP/MAC address
Grabs from the Web
- TV Guide Schedule
- TV/Movie Ratings
Gets from User
- Selections on what to watch from list of recorded Shows
- Ratings on shows being watched
- Vetos on shows the user does not like/ has already seen
Grabs data from device
- movies user watched
- ratings user gave selected movies
- movies user watched from selection
Application sends to device
- What movies to record for user
- Intelligently updates movies sent to device based on feedback.
Gets from Cable Provider
- Content to Record

Figure 3-1: Overview of the System

that integrate the domains of consumer electronics and social networking applications will become commonplace.

3.1 Overview

The implementation of this project consists of three basic parts: the Facebook application, the transmission of data from the TV guide into the Facebook application the out to the DVR box, and the recording and rating application on the DVR itself. The Facebook application is used for displaying three main kinds of data—setup data for the DVR, the listing of friends' favorite TV shows and movies, and listings of what will be recorded next on the DVR. This gives the user the maximum amount of control from the Facebook Platform—the user can veto shows, change ratings, and setup their DVR while reading and interacting with their social network. On the connection side of things, there are two main components: one to download data from a TV guide system, and another to send customized recommendations to the DVR and back. TV Guide data comes from XMLTV[32] and is parsed each time the user uses the Facebook application. The DVR data on the other hand, is actively requested by the DVR when it is on, in periodic intervals, when there is space available for more recordings. Finally, the DVR component is composed of a menu view that allows the user to view shows and rate them, and a back-end that grabs new shows to record and stores them for future viewing. There is no actual configuration or complicated new controls on the TV itself: the DVR is still used for viewing pre-recorded shows, but the source of the shows is the social networking application.

Overall, these three components work together to create the working application. As much as possible, open standards were used in every step of the implementation. Facebook was chosen for its open API to develop new third party applications [9]. No new social networking applications needed to be created for this particular application because an open one already existed. Similarly, TV data was gathered using XML-TV, an easily accessible format for TV guide data. Finally, in the DVR, an Open Cable digital video recorder was used to take advantage of the publicly available JAVA API.[27] OCAP is a Java-based middleware layer for cable television set-top boxes, defined by CableLabs, Motorola, Inc. makes set-top boxes that support OCAP, contain DVR, or hard-disk recording capabilities, and have an Ethernet port with an IP stack.

3.2 Facebook Application

The Facebook application component serves as a central hub to the rest of the project. It allows a user to set-up their DVR and personal settings, provides a central server for the DVR to "call home," populates the lists of favorite TV shows and movies, grabs data from the TV Guide listings, and displays what a user will watch next on the user's profile. The Facebook Application runs on a Physical Language Workshop server; is displayed in an embedded page in Facebook; makes use of several Facebook API functions to log in, send data to user profiles, and gather information from friends' lists. The application is written in PHP[21] and Javascript[12] , as those are two languages that have functions adapted to work with the Facebook API. The application is designed in a Model-View-Controller [10, p.4] fashion: the model is held by the MySql database and by the PHP functions. All the user control operations

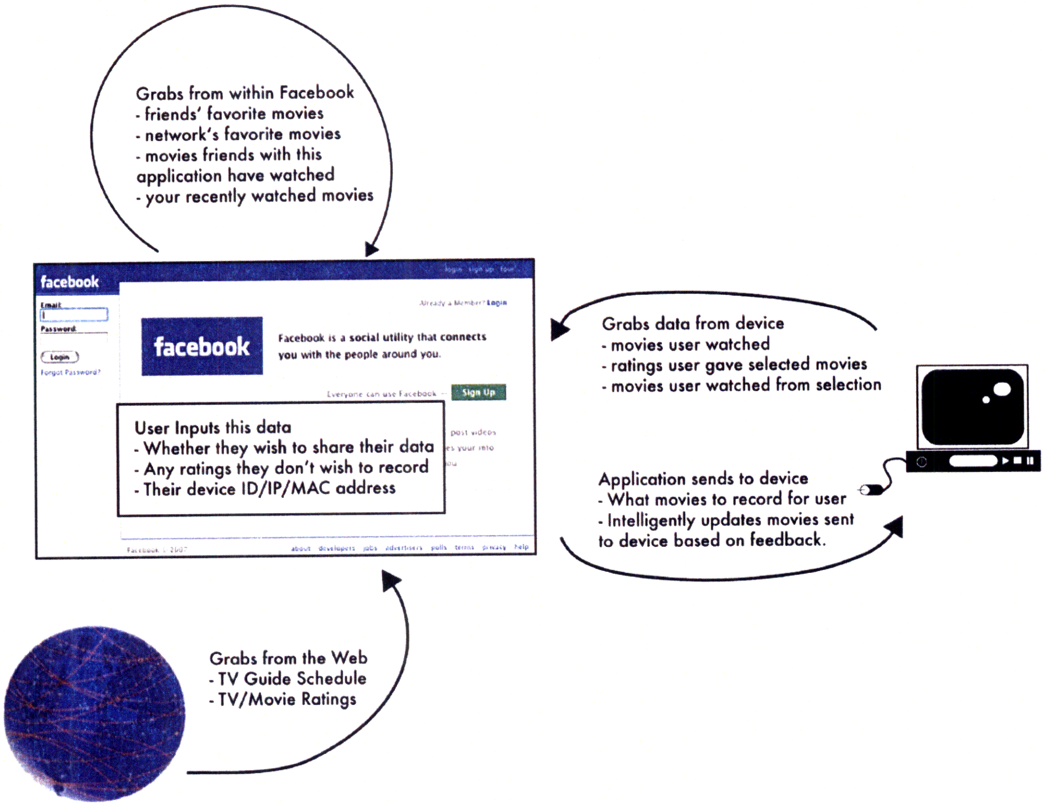
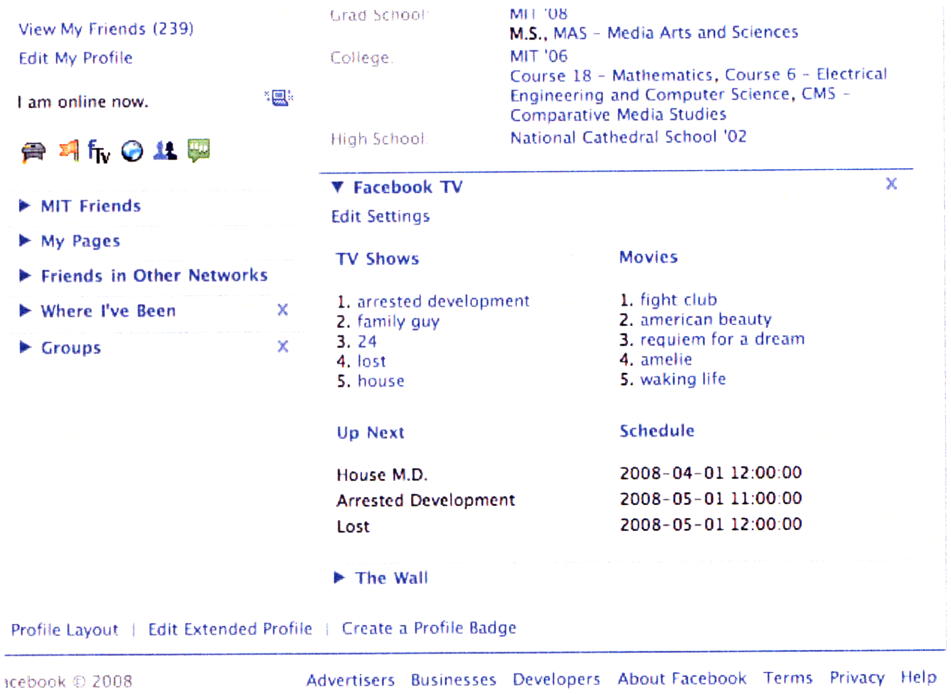

Grabs from within Facebook
- friends' favorite movies
- network's favorite movies
- movies friends with this application have watched
- your recently watched movies
facebook
facebook
Facebook is a social utility that connects you with the people around you.
Sign Up
User Inputs this data
- Whether they wish to share their data
- Any ratings they don't wish to record
- Their device ID/IP/MAC address
Grabs data from device
- movies user watched
- ratings user gave selected movies
- movies user watched from selection
Application sends to device
- What movies to record for user
- Intelligently updates movies sent to device based on feedback.
Grabs from the Web
- TV Guide Schedule
- TV/Movie Ratings

Figure 3-2: Basic Functionality of the Facebook Application


View My Friends (239)
Edit My Profile
I am online now.
▶ MIT Friends
▶ My Pages
▶ Friends in Other Networks
▶ Where I've Been
▶ Groups
Profile Layout | Edit Extended Profile | Create a Profile Badge
ıcebook © 2008
Advertisers Businesses Developers About Facebook Terms Privacy Help

Grad School: MIT '08
M.S., MAS - Media Arts and Sciences

College: MIT '06
Course 18 - Mathematics, Course 6 - Electrical Engineering and Computer Science, CMS - Comparative Media Studies

High School: National Cathedral School '02

▼ Facebook TV

Edit Settings

TV Shows	Movies
1. arrested development	1. fight club
2. family guy	2. american beauty
3. 24	3. requiem for a dream
4. lost	4. amelie
5. house	5. waking life

Up Next	Schedule
House M.D.	2008-04-01 12:00:00
Arrested Development	2008-05-01 11:00:00
Lost	2008-05-01 12:00:00

▶ The Wall

Figure 3-3: Picture of the User Profile Box

are contained in the Javascript functions, and the view is made entirely in HTML and Facebook Markup Language (FBML).

There are two main parts to the Facebook application, which will be described in the following sections. The first is the user profile box that displays application data with your friends, as well as being a quick-check window for the user. The second is the main UI of the Facebook Application, where the user can view and configure most of these settings.

3.2.1 User Profile Box

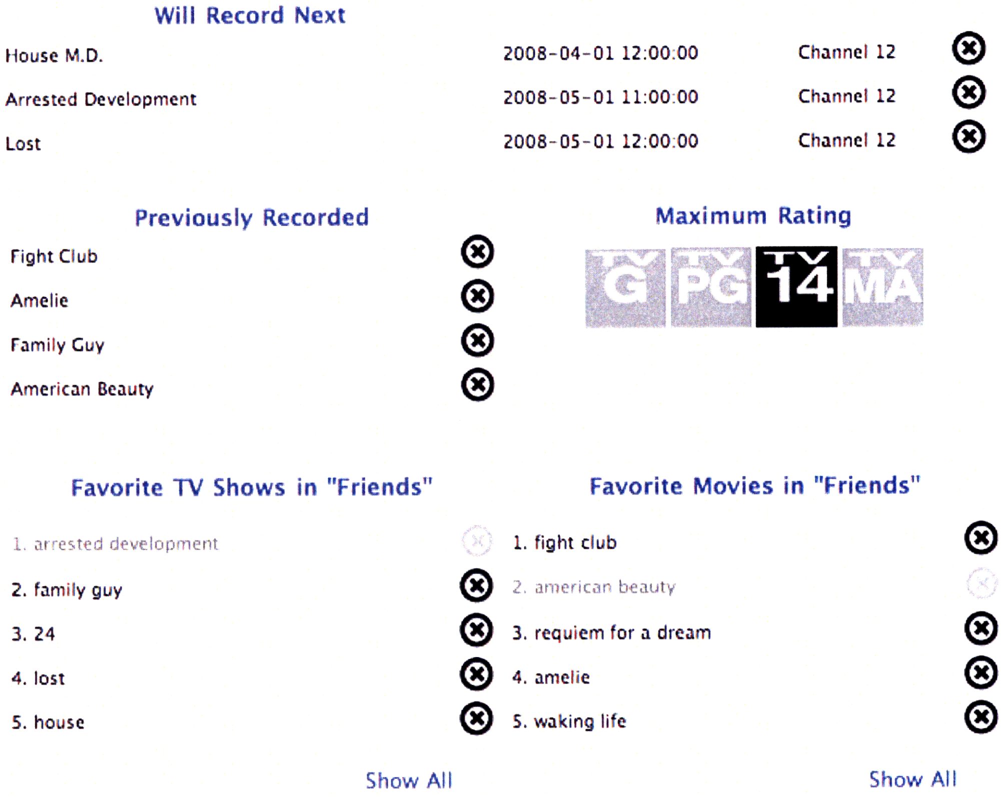
The user profile box of a Facebook application is the standard way other users can interact with a particular instance of a Facebook application. This particular profile box serves two main purposes: to inform the user in a quick fashion of what is currently popular, and to give the social network a glimpse at your friends' network and viewing habits. The profile box tends to be simply a display apparatus with minimal code; it is a portal to the rest of the application. The profile box tends to be compiled each time the user edits the app, instead of being in a constant state of change.

There are two main formats to the profile box—the narrow view and the wide view. In the narrow view, only one column of text is used, thus, the listings are shorter and less information is displayed. Since the user is placing the profile box on the left side of the application, the assumption is that the application is less important to them, and thus, less information is displayed. With each listing, a google search link is provided—this helps users and friends quickly find out about new shows they have never heard of before as well as sharing a preview of the user's friends list. Finally, the profile box also includes a short listing of what shows out of that list are on next. This gives a sense of connection between the user's DVR and those friends viewing his profile. In a tangible sense, they can be aware that the application connects to a real-life object, that is recording their preferences and transferring them into the users daily life.

The profile box is implemented in PHP in the main application. It gets rendered every time the application gets opened. This saves time on the user's side and saves database queries. Two versions get compiled: the narrow version and the wide version. An awareness of the user's behavior in a summarized form is the principal purpose of the profile box. It communicates to the user and the friends list three things: the frequency of the user's updating his application, a summary of the favorite shows on the friends list, and what will be on next. Overall, the profile box is a small look into the larger application.

3.2.2 Main Application UI

The main application allows the user to see a full listing of his friends list's preferences, veto the recording of any shows the user does not want to record, see a listing of what will be recorded next and what has previously been recorded, and set some basic DVR preferences. The UI endeavors to be clean and intuitive to use. All actions in the UI are undoable—anything crossed off can be uncrossed, and no changes are committed



Will Record Next

House M.D.	2008-04-01 12:00:00	Channel 12	⊗
Arrested Development	2008-05-01 11:00:00	Channel 12	⊗
Lost	2008-05-01 12:00:00	Channel 12	⊗

Previously Recorded

Fight Club	⊗
Amelie	⊗
Family Guy	⊗
American Beauty	⊗

Maximum Rating

Favorite TV Shows in "Friends"

1. arrested development	⊗
2. family guy	⊗
3. 24	⊗
4. lost	⊗
5. house	⊗

Show All

Favorite Movies in "Friends"

1. fight club	⊗
2. american beauty	⊗
3. requiem for a dream	⊗
4. amelie	⊗
5. waking life	⊗

Show All

Figure 3-4: The UI of the Facebook Application

until the user saves his changes. Most of the calculation of user preferences happens at this stage; through a combination of histogram data, as well as a collection of vetoes and preferences from the DVR, the application calculates what to record next.

The UI itself is based on Jacob Nielsen's "Ten Usability Heuristics,"[19, p.153]

- Visibility of system status.
- Match between system and the real world
- User control and freedom
- Consistency and standards
- Error Prevention
- Recognition rather than recall
- flexibility and efficiency of use
- Aesthetic and minimalist design
- Help users recognize, diagnose, and recover from errors
- Help and Documentation.

These heuristics were originally developed in 1990 based on an analysis of 249 common usability problems, and are used in this project as a guide to avoid UI pitfalls and in general make a better final product.

To fulfill the first heuristic, all preferences the user inputs are shown plainly on the screen. As well, the shows that will be recorded next and what has been previously recorded already are at the top of the page, so that the user is aware of the DVR's next action at all times. As for the second heuristic, the user can set the maximum rating using symbols familiar to the user. For user control and freedom, every option and every show to be recorded can be vetoed (and unvetoed) by the user, and the maximum rating that can be recorded can be changed. The design is consistent with Facebook standards and uses Facebook dialog boxes as well as familiar colors and fonts. To prevent errors, all vetoes can be over-ridden, and no changes are saved until the user presses save.

No commands are meant to require recall; all commands and options are visible to the user at all times. As for efficiency, the user can veto shows wholesale instead of just vetoing specific episodes if the user determines they do not want to record a show at all. The design is minimalist; only containing the information the user needs and nothing more. As for the last two heuristics, the system tries to remind the user in two places to save. Also, all actions can be undone, even after saving. An "About" dialog explains the purpose of the application and gives an email for the user to contact if they had any problems using the system. All ten heuristics informed the design of the main application, and how the user interacts with the application.

The application executes a number of functions behind the scenes both before and after saving. The first step the application takes is to authenticate the application and its user. Using Facebook tools from the API to do this, the application confirms that the user is a valid user and creates a new database entry for them. Once that is done, the application looks at a list of the next day's TV guide listings (see section 3.2.1) to examine what it will record next. First, the application must see what shows and movies the user's friends like. Using the Facebook API, it checks every friend in the user's profile and grabs each person's favorite shows. With those listings in hand, it uses a histogram to rate the popularity of each show. The application then compares the histogram to the past stored list of shows, and checks to see if any shows were vetoed. It also grabs the list of preferences that have been provided by the DVR (see section 3.2.2). Using that list of shows, it compares it to the TV guide listings and picks out which to record next. It has at this point compiled a list of what to record next in a table in the application.

Next, the application grabs any data about what the DVR has previously recorded—as well as other setting data—and displays it on the main view of the application. Finally, it also displays a list of the favorite shows and movies—either showing the top five or the complete list.

The application has a separate dialog box for setting the IP address of the DVR or for saving changes and preferences in the application. It uses the standard Facebook dialog for the save to keep a consistent look and feel with the rest of the website. Both these actions generate form actions that sends the status of the application back to be saved in the database. Everything that was checked on or off is saved in the database, as is the maximum rating and the DVR IP address, and the page is reloaded.

The Facebook application is written primarily in PHP and FBML/HTML, Facebook's unique markup language. Additionally, all the "controller" features, such as working checkboxes, dialog boxes and saving actions are implemented in JSFB, Facebook's version of Javascript. The model of the application is all held in a MySQL database to obey the Model View Controller paradigm. All these disparate pieces make the bulk of the Facebook application. In the following sections, I will describe how this piece of the application grabs data from XMLTV to compile the TV Listings of the day and sends them to the DVR.

3.3 Interconnectivity

There are two main connecting pieces that are needed in the implementation of this project. The first one consisted of connecting TV guide data from XMLTV to the Facebook application. The second consisted of getting the Facebook Application data to the DVR and then grabbing the user preferences from the DVR and feeding them back to the Facebook application. Both of these pieces were implemented primarily using PHP, with a bit of perl and some Java commands at either side. The main protocols used were SOAP [17], an XML transfer protocol and a simple URL request method. The methods chosen are not high in security or encryption; rather, they were chosen for their ease of use and relative small data packages as well as quick processing speeds. The goal was to connect these two pieces in a simple and platform-indpendent way, so if needed, any other set of TV Guide data from any country, or any other web-enabled DVR could connect using this very simple connection scheme.

3.3.1 TV Guide Data

Since the cable we propose to use for our experiments, MIT Cable, does not provide a TV Guide listing directly, we needed to acquire one for ourselves for this project. The source of TV Guide data we ended up using for this project was Schedules Direct[26]. In partnership with Zap2it, they provide both the lineup of channels available on MIT cable as well as access to the XMLTV data for North America. We chose this service because it has an option for using this data for personal research purposes, not simply for personal consumption.

Once the XMLTV application is running on the server, it is possible to request a listing of television data from the source. Using a simple perl script and the SOAP exchange protocol, the application grabs TV Guide data for a specified period of time. Using PHP, the thesis application then parses the XMLTV data to extract the following pieces of data:

* Channel number mapped to Station code

* Date a show is on

* Show ID

* Rating for a show

* Station a show is on

* Canonical show name mapped to Show ID.

Once the XML parser extracts all this data from the television file, these tables of data are passed on to the main Facebook application. for comparison with each user's personal preferences to extract an optimized listing of what should be recorded next. Every day and a half, the perl script refreshes the TV guide listings from XMLTV, so that the Facebook application can have the latest set of guide entries to pick new things to record.

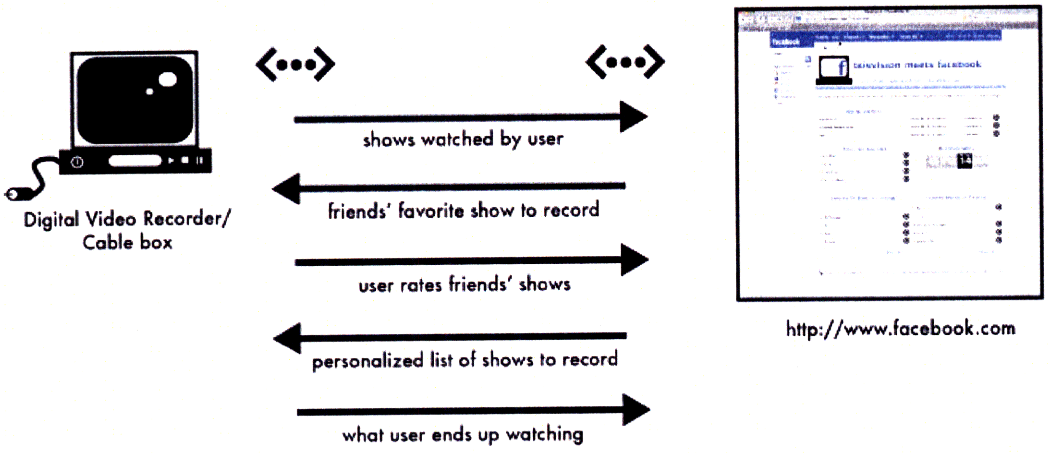

Digital Video Recorder/
Cable box
shows watched by user
friends' favorite show to record
user rates friends' shows
personalized list of shows to record
what user ends up watching
http://www.facebook.com

Figure 3-5: Connectivity between the DVR and the Facebook Application

3.3.2 Connection to DVR

The second part of the connection interface connects the DVR to the Facebook application. To do this, the method used is designed to minimize setup on the DVR itself, as well as making the server device agnostic to allow for future expansion to a wide range of DVR boxes. The connection happens through one main identifier: the device unique ID. With that identified, the device can ask for the data it needs and the Facebook application can record any incoming data from the device and associate it with the right user.

This connection is established through a HTTP GET method, initiated by the DVR box and sent to the Facebook application. This is important for two reasons: one, the DVR can be behind a local area network since only the central server needs to remain static and two, because this connection requires less setup from the user's perspective. Once Facebook knows which device belongs to which user, the device itself only needs to ask Facebook for the data, it does not need to authenticate itself, ask the user for their Facebook username and password, or anything else. Through this GET method, the DVR is able to get the latest listing of shows as an XML string as well as send ratings of currently watched shows as parameters in the request.

There are 2 main commands that can be passes as parameters to the GET method: asking for more shows to record and asking to rate a show. When the DVR requests more shows to record, it sends a parameter that specifies how many slots it has left on

Figure 3-6: The DCH6416 OCAP DVR

its harddrive. The HTTP webpage, running PHP, parses the request and grabs from the Facebook app the next batch of shows to record, and sends it to the DVR. In turn, the Facebook app removes those shows from the queue of what to record next. The second action is to rate shows. In the DVR itself, the user can rate shows they have previously watched as positive or negative (see section 3.3.1). These ratings get sent as requests with the show name and the rating as a parameter. The Facebook Application stores this data into the user's profile and uses it to determine which shows to record next in the future.

3.4 OCAP Box

OCAP is a Java-based middleware layer for cable television set-top boxes, defined by CableLabs, Motorola, Inc. makes set-top boxes that support OCAP, contain DVR, or hard-disk recording capabilities, and have an Ethernet port with an IP stack. The particular model used in this thesis is the DCH6416—an analog-digital, DVR Host set-top supporting HDTV, SDTV, and data with advanced features and ample recording capacity. This model can be programmed by means of a JAVA API on Eclipse, through an emulation console. This project leverages the Motorola provided OCAP set-top boxes so that we may concentrate on the unique aspect of our project: integrating this consumer electronics device to Facebook. The applet consists of two main parts, the UI and the recording back-end. This is the part of the application that lives on the physical device to allow communications with the Facebook Application.

3.4.1 DVR UI

The main part of the applet on the set top box is the UI for the application. As with the Facebook application, the goal is to touch on all the main UI principles and create a simple, easy to use UI framework. The UI needs to provide three critical pieces of information to the user: Device ID and password for set up of the Facebook application, a list of recorded shows and shows to be recorded, and instructions for rating and deleting shows. These three tasks were implemented in keeping with the ten usability heuristics in order to improve the user experience.

The UI gets called from the remote control or the set top box buttons themselves when the user presses the Menu button or the DVR List button. This brings up the main screen of the UI. Here, the user gets all the information they need—the setup information, the instructions for usage, and the shows that have been recorded. The title of the applet is on the top of the Menu screen, followed by the device unique ID and password—these are hardcoded into the applet. Ideally, this should use the MAC address of the device. This is all the information needed to set up the device on Facebook. The next item on the screen is a short key on how to rate shows:

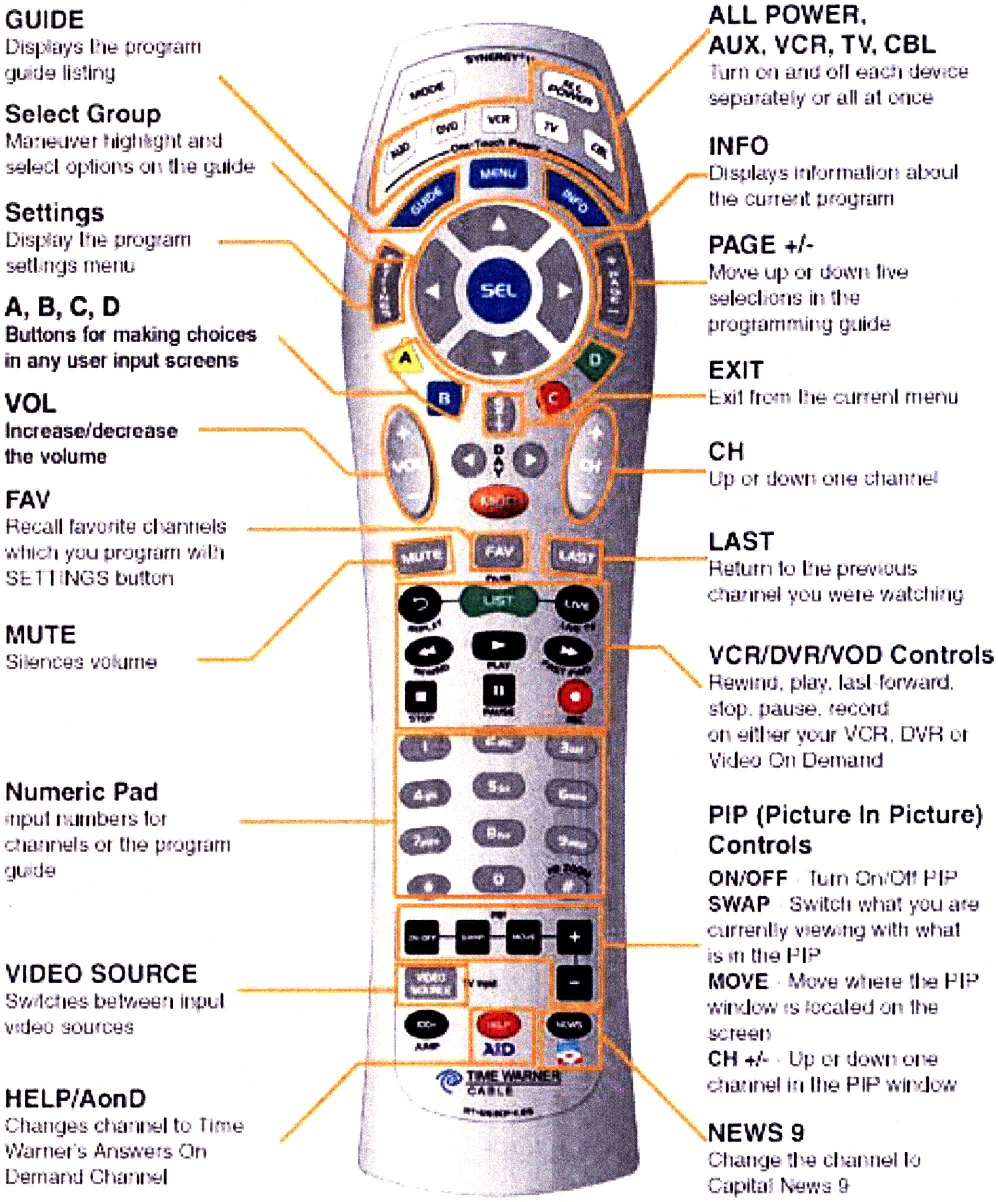
* Press the A button to mark a show as a favorite

* Press the B button to mark a show as a favorite that you do not wish to watch again.

* Press the C button for a show that one disliked

* Press the D button to delete an entry.

All these buttons are by default on the DVR remote, so that the user need not navigate submenus to rate a show at any time. These buttons are color coded and easy to spot on the remote. Finally, there is a list of all the shows recorded and queued up for recording. Shows in white will be recorded, shows in black have been



GUIDE
Displays the program guide listing

Select Group
Maneuver highlight and select options on the guide

Settings
Display the program settings menu

A, B, C, D
Buttons for making choices in any user input screens

VOL
Increase/decrease the volume

FAV
Recall favorite channels which you program with SETTINGS button

MUTE
Silences volume

Numeric Pad
input numbers for channels or the program guide

VIDEO SOURCE
Switches between input video sources

HELP/AonD
Changes channel to Time Warner's Answers On Demand Channel

ALL POWER, AUX, VCR, TV, CBL
Turn on and off each device separately or all at once

INFO
Displays information about the current program

PAGE +/-
Move up or down five selections in the programming guide

EXIT
Exit from the current menu

CH
Up or down one channel

LAST
Return to the previous channel you were watching

VCR/DVR/VOD Controls
Rewind, play, fast forward, stop, pause, record on either your VCR, DVR or Video On Demand

PIP (Picture In Picture) Controls
ON/OFF - Turn On/Off PIP
SWAP - Switch what you are currently viewing with what is in the PIP
MOVE - Move where the PIP window is located on the screen
CH +/- - Up or down one channel in the PIP window

NEWS 9
Change the channel to Capital News 9

Figure 3-7: DVR Remote

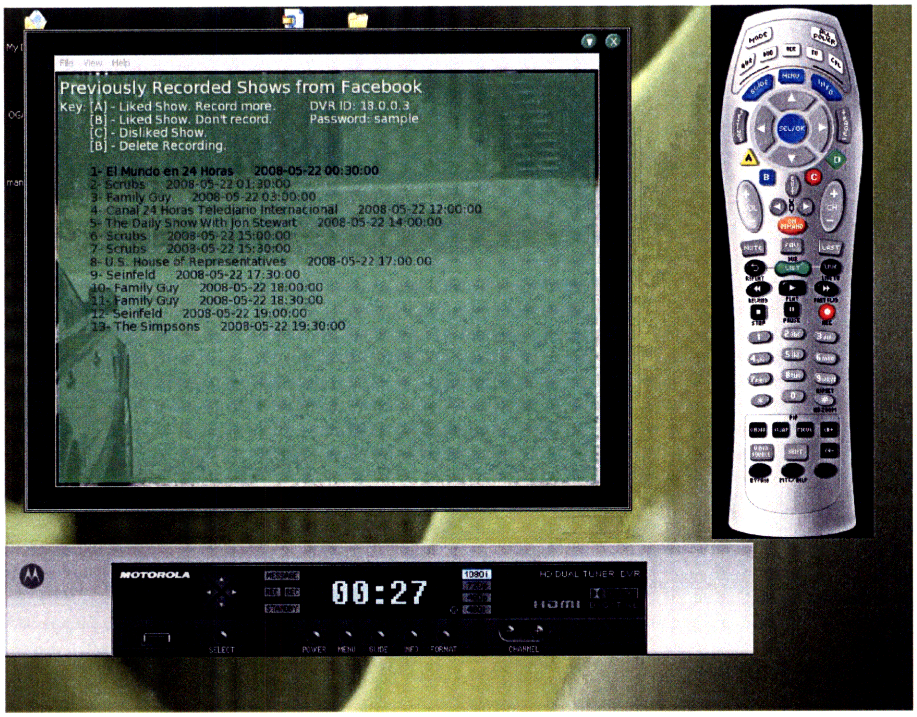
recorded already, and grey highlights a show currently selected. A user uses the arrow keys to navigate the list of shows. Pressing Play starts playback of any show that has already been recorded.

This UI fulfills the Nielsen usability heuristics quite well. For visibility of the system status, all set-up information as well as status of shows is clearly shown on one screen. The system links to the real world by using expected remote control mappings. This fulfills the consistency and standards criteria as well. The user does not have much control, for there is not much to do, but the user can definitely delete and rate shows as they please. Show deletion does not carry an undo function, however, ratings can always be overwritten by simply re-rating the show. All instructions are visible in the screen to prevent needing the users to remember anything. The system is very efficient because it puts all the user functions out at the touch of one button. The design is simple, uses contrasting colors, and matches the rest of the UI quite well. All the documentation is clearly visible on the screen.

The UI was implemented in JAVA using the OCAP JAVA API, and standard Xlet templates. It is meant to be modular so it can be easily integrated with a larger suite of applets and tools in a large PVR package.

3.4.2 Recording Mechanism

Recording on the DVR is done in a very straightforward manner using the OCAP recording API. The first step is to determine how much free space there is on the hard drive. Using API commands on resource allocation and management to determine the amount of space left on the hard drive, and using the average size file for a 30 minute show, the application calculates how many slots available there are for recording. Every time the menu option gets called on the DVR, the DVR requests a number of shows equal to the number of free slots to record on the DVR taking into account show duration from the Facebook application as specified in section 3.2.2. The results are returned as a string to the DVR. The applet then parses the string to create individual show entries for each show to be recorded. Depending on the time stamp of the show, the UI can determine whether this is a previously recorded show



```
Previously Recorded Shows from Facebook
Key: [A] - Liked Show. Record more.      DVR ID: 18.0.0.3
     [B] - Liked Show. Don't record.     Password: sample
     [C] - Disliked Show.
     [B] - Delete Recording.

     1- El Mundo en 24 Horas    2008-05-22 00:30:00
     2- Scrubs    2008-05-22 01:30:00
     3- Family Guy    2008-05-22 03:00:00
     4- Canal 24 Horas Telediario Internacional    2008-05-22 12:00:00
     5- The Daily Show With Jon Stewart    2008-05-22 14:00:00
     6- Scrubs    2008-05-22 15:00:00
     7- Scrubs    2008-05-22 15:30:00
     8- U.S. House of Representatives    2008-05-22 17:00:00
     9- Seinfeld    2008-05-22 17:30:00
     10- Family Guy    2008-05-22 18:00:00
     11- Family Guy    2008-05-22 18:30:00
     12- Seinfeld    2008-05-22 19:00:00
     13- The Simpsons    2008-05-22 19:30:00
```

Figure 3-8: UI of DVR application

or not, and display it accordingly.

Once that is done, the DVR application queries up the shows to record next in accordance to the OCAP API. The recording manager will run a separate thread to ensure recording happens when appropriate even if the user is not currently using the application. The user need not take any action to initiate a recording. Once a show is recorded, the recorded object gets linked to the show data entry retrieved from the Facebook Application. That way, once a user selects it from the menu, they can watch the show at their leisure. Due to the limited scope of the project, the user cannot watch a show while the DVR is recording, but this feature can be added in future implementations.

The system is implemented in as simpley as possible to provide a bare-bones example of the ideas presented in the thesis. The recording interface is just a sample that could instead be replaced by a professional PVR application, using the Facebook front-end to intelligently select shows for the user.

Chapter 4

Analysis

> *The human race, to which so many of my readers belong, has been playing at children's games from the beginning, and will probably do it till the end, which is a nuisance for the few people who grow up.*
>
> *- G.K. Chesterton, The Napoleon of Notting Hill*

There are several axes that can be used to measure the success of the project, both in quantitative terms as well as qualitative terms. First off, from an engineering perspective, whether the system does what it claims to do: is it a set-top box that automatically downloads a user's friend's data from Facebook, and does the Facebook application give the user the necessary control over the system. Secondly, is the UI on the Facebook application easy to use and pleasing to users, and does it pass the usability heuristics tests. Additionally, does the data generated by several users reveal interesting patterns of media consumption. Finally, in a longer study, does the system prove useful to users. Using these four types of analyses, the success of this project can be evaluated.

From the results of each of these groups we can answer part of each of the main motivations for the project. By looking at the engineering side of the project, as well as the evaluation of the UI, one can see how well this project falls into the vision of CE 2.0. By evaluating the raw data generated by the users from the Facebook application, one can see how this application aids in the media transmission and whether the application helped expose users to new media. Finally, in a longer study, both the question of media transmission and of whether tacit data proves useful to a user under a passive delivery mechanism can be answered. In each of these three realms—consumer electronics 2.0, tacit data from a social network, and new forms of media transmission—the success of this thesis be measured through these different experimental methodologies.

4.1 Engineering Perspective

This evaluation will be primarily testing to see if the system works as specified in the implementation section. This is not a user test, but a system check test. It will verify three important aspects of the implementation: the Facebook application, the connection between the DVR and the Facebook application, and the DVR software itself. As for the Facebook application, the evaluation will confirm that it downloads data from a TV guide listing and displays it accurately, if it grabs data from a user's friends list, and whether options such as vetoes, ratings, and DVR IP can be set correctly. The verification will be done by checking the database and looking at the data entries of the registered users. To verify the connection between Facebook and the DVR, a DVR will be connected to Facebook, and the database will be used to confirm that the same data being sent from the DVR is arriving to the user profile for the Facebook user. Finally, the DVR itself will be evaluated by seeing if it can record a day's worth of shows, and whether the different menu options can be used fully.

4.1.1 Results

There are three aspects that were checked in this section of the evaluation: does the application work as specified, and save the corresponding data to the database, do results get loaded from the Facebook application to the DVR, and do the different functions on the DVR work. The database had records for 22 users, and two users were tested for compatibility with the DVR. Using this data, we determined the overall functionality of the system—confirming that the system did indeed work as specified. Unfortunately, due to delays on the arrival of the testing DVR boxes provided by Motorola, the DVR recording capabilities of the machine were not fully tested, although the UI and data transfer capabilities did work as needed.

For the Facebook application, first checked was that each user had a complete entry on the database. All 22 users were able to reach the page to do this. Additionally, the page displayed data for their favorite shows and what to record next. All these entries showed up on screen as needed. The application in general loaded up, and had all the parts described in the implementation section. Also, data was properly loaded from XMLTV, and each user had a list of recommended shows. The different setting preferences also work as specified, and an user could save, change ratings, and set up a DVR. Finally, the profile box loaded in a user's profile in both columns and looked as designed.

The DVR loads its UI and can download data from Facebook from a couple of different users. Additionally, rating shows, deleting shows, and changing to the appropriate channels works. All the functions on the remote control work properly. This was individually verified by checking the database that recorded setting show preferences from the DVR. On the whole, the system works sufficiently well for testing on users and seeing if it works to fulfill the projects motivations.

4.2 Database Analysis

This section of the analysis will track data from the database for the 20 users of the Facebook application in an attempt to reveal media patterns and relevance of the data for the use in media sharing. This will be mostly a quantitative analysis. It will attempt to determine how many distinct recommendations a user on average gets, as well as how distinct these are from user to user, and how up to date the listing are with the average TV guide listings of the day—how many such shows are on, how many distinct shows are on. This data will be useful for measuring how recommendations grow as a proportion of users of the application, and how relevant the recommendations are for the average user's TV watching habits. This very quantitative analysis of the data will let us see whether this model might be a useful venue for viral data transmission or not, and how difficult or easy it would be for data bots or advertising agents to affect the pool of data, and whether it would have an effect on the user's viewing habits.

4.2.1 Results

Over twenty-two users, a large variety of shows were recommended to the users. There was some deviation between the most popular shows of users despite in general only being one degree of separation apart from each other. Additionally, over the course of a day, a large number of recordings were recommend to any particular user, increasing what any one person is likely to watch over the course of one day.

The average user had 862 different recommendations in movies and TV, making an average of 431 recommendations in each category, However, each person only had on average 142 entries with more than one respondent marking them as their favorite, which eliminates shows with uncommon spellings or improperly formatted entries. The average user had 43 TV shows with two or more respondents and 98 movies that had more than one respondent. Even if only these results get used in determining which shows to record next, the user will be exposed to a very large range of new movies and shows on TV. Another metric observed was how many people rated their most popular shows. The most popular show in any user's list got an average of 17 ratings, and the second most popular show got 13, making it quite feasible for new shows that have a certain following to quickly make a burst in popularity and make it to the top of the list. Of the top 5 most popular shows, on average, users shared about 68% of their results in common, while the number dropped to 12% for only movies. This correlation shows that movies are more likely to show a greater diversity in popularity—meaning users will be exposed to more individual distinct movies, while a greater homogeneity is shown in television—meaning friends can use this tool to record series concurrently together.

For the list of what to record next, over 22 users, 642 shows were recommended to be recorded for the next day, on different days, giving an average of 30 recordings per user per day, more than an user can reasonably watch, giving the user ample room to veto and improve his choices by using the system over a period of time. Among the results, there are many that seem like obviously mis-hits, partly due to the limitations of the searching algorithm used. This can be both an inconvenience and a blessing; one one hand, it lowers the effectiveness of the algorithm, but can also be used to expose him to shows of similar names at random. Most of the recommendations ended up being TV shows, not movies, given the lack of premium movie channels on MIT Cable, so the aspect of diversity of movies recorded did not get much testing. Nevertheless, there was a very wide range of shows being recorded, including both shows that were in the average user's top 5 list as well as more obscure television recommendations.

Overall, the system seemed to find a wide range of recommendations to give a user, and it seems, based on solely a quantitative analysis, that this system can generate data that is useful to the user in a manageable format. It both has a great range, yet manages to condense hundreds of recommendations per user into small, 10-30 show packets pet day to give the user something new to watch.

4.3 Facebook UI Evaluation

The purpose of this part of the evaluation is to determine if the UI of the Facebook application satisfies the basic usability heuristics outlined in the Implementation section and to get general feedback of the success of the Facebook application. This evaluation will be executed via user surveys and opinions primarily. By means of the Facebook application, as well as volunteers within the MIT Media Laboratory, questionnaires evaluating the UI of the Facebook application will be provided and filled out that rate the merits and faults of the application. The questions are meant to evaluate the Facebook application UI on the ten heuristic criteria presented by Jakob Nielsen. About twenty volunteers will be used in this part of the evaluation. The result of the findings will be aggregated and analyzed for a qualitative analysis on the usability of the application. The questionnaire that will be used is as follows:

* Does the application accurately inform you what it will record next and what it has already recorded?

* Does it display useful data regarding your friends' preferences?

* Do you feel like you have an intuitive understanding of how to operate the application?

* Does the system easily support undoing unwanted actions or errors?

* Do you feel that the application acts in a consistent manner?

* Do you sometimes forget to save or commit changes to the system?

* Are there operations that are too hidden or non intuitive?

* Are there functions you wish existed that would make this application faster to use?

* Are there functions you wish existed that would in general make the program more useful?

* Is there any redundant information that could be done away with in the app?

* Are portions of the application hard to read?

* Do you find the design cluttered?

* Would a help section make this app easier to use?

* Does the application allow you to recover from errors easily?

In a longer, one week study, both this part and the DVR will be evaluated as a unit in section 4.4.

4.3.1 Results

Twenty two sample users filled out the evaluation form for the application. The majority of the responses were positive about the UI of the application. Some complaints about the application remarked confusion as to its purpose lacking the DVR, not enough to do on the Facebook application itself, and errors regarding some aesthetic properties of the layout. Finally, there were complaints about not being able to set the cable provider, which is not a flaw of the application itself, but a limitation caused by the TV Guide providers.

The questionnaire the respondents filled out contained fourteen questions described in the previous section. The majority of the users responded positively to every question. Additionally, the questionnaire provided a comments section for elaboration. The main comment presented by the users was a confusion as to the purpose of the application without the DVR, but that was cleared up once the user was informed that a device is meant to be used in conjunction with the Facebook application. Along with that comment came the remark that it would be useful if the application were not limited to MIT Cable. This would be an easy change to make by simply requiring each user to setup their own Schedules Direct account to access XMLTV or if the project was developed in conjunction with a cable company. Another common complaint expressed by three users, was the need to save after making modifications. This could be solved by adding Ajax commands into the JavaScript to allow asynchronous saving. There was another bug that popped up if a user had insufficient listings and recommendations for the day; part of the sub-page might be cut off by the frame of the application. This problem was resolved by increasing the size of the background DIV. The final suggestion concerned there not being enough to do to bring a person back to the application on a regular basis. Some proposed solutions to this problem were to add other TV related functions like a TV Guide search or a manual PVR recording setting so that one application could be a hub for many TV functions. Overall, however, the responses to the application UI were positive, confirming the goals to have an intuitive and simple to use UI.

4.4 Long-term User Study

The last part of the evaluation will be a longer term user study on a small number of participants. A couple of units would be distributed to a couple of users who would install it in their homes and use the Facebook application for a week. All participants would live in MIT dormitories and hook up the system to MIT cable. They would also be long-term users of Facebook so that they have an already established friends network. Over the course of the study, the user would observe if the presence of the DVR made them watch shows they normally would not have watched, as well as rating shows they liked or did not like to improve the recommendations of the system. All this data will be recorded by a central database for the duration of the study. Besides more tangible considerations like types and volume of media consumed, the user will also be asked regarding their satisfaction with the system, their changing perception of privacy, and how the presence of the device augmented their social interactions with regard to television watching, and how it influences the perception of their social network.

At the end of the one week trial period, the participant would fill out a questionnaire similar to the one in section 4.3, rating the effectiveness of the user interface based on usability heuristics. However, more subjective questions will also be asked in an interview for the user to evaluate the system's effectiveness in bringing the social network's data in a meaningful way into the user's life. Notable events and shows will be emphasized more than statistics in this particular study. The final goal will be to see if the system is effective at mining tacit data from Facebook and bringing it into the user's TV watching experience.

4.4.1 Results

Two users were tested for a week using the device, and their feedback was collected in an informal interview. The users were very avid TV watchers and had a large network of friends on Facebook. The general response to the system was positive, although some suggestions were made regarding how to enhance the social aspects of the system. Privacy was not a large concern of the users at all—the unfiltered user of the friends network to generate recommendations was expressed as a concern. The users were pleased by the device and application and wished that this sort of application was available in their future DVRs they purchased.

The users both found the user interface intuitive to use and useful. They were not concerned about it only doing the tasks it was designed to do as the group that only reviewed the application previously—tied to the television or DVR, they saw the usefulness of the application as it was. They did not like how some of the recommendations were hidden and only the top 5 were shown. They found that needlessly confusing.

There were several features the users were positive about, including scheduling, and the aggregation of recommendations by the system. One user in particular said that they liked the way this system scheduled shows because he liked the idea of watching things his friends might be watching instead of extrapolating simply from his tastes, because it felt more "useful" that way. He did mention, however, given he already watches many of the same shows his friends watch—during the off season, when most shows are reruns of popular shows—that the device did not record anything he had not seen already, but that it did give him a feel of connectedness. He expressed that having the device with either a database of movies, during times when new seasons were on, or with digital cable, would render the device far better at creating original recommendations, however. Although he did not find the automatic scheduler particularly useful for new recommendations, he did find the the actual listing of the favorite TV shows and Movies very useful, because he can dig up those TV shows and Movies on his own if he is hunting for things to watch. The other user said that the system should be able to have selective lists of friends instead of his entire friends list. He thinks it would be more social to share information with friends whose tastes he respects, and is likely to share comments and watch TV with in other settings. Some people in his friends list, although not bots or advertisers, are people he does not talk to, and thus the social aspect of watching their TV shows is lost. Neither does he want their input for recommendations.

The users also recommended several things for modifications, and expressed some things that were previously concerns were not concerns to them. The main feature the users expressed interest was in having a system that grabs the show recommendations straight from people's DVRs, because they felt they would get a more "honest" picture of their friends, and because they might get recommendations that although not embarrassing, are more innocuous and not the sort of thing one lists as a "favorite." Neither user expressed privacy concerns with sharing that data on a closed network like Facebook, even without options for filtering out their best friends from random friends.

4.4.2 Analysis

In response to the user comments, several conclusions can be drawn about the project and future directions. The main direction suggested by the target users was one towards more connectivity than that provided by the device, not less, and how the project partially touched on the desires of the users, but not completely. The secondary idea touched by the users was the concept of limiting one's sample data size not by specific data but by limiting the friends tapped. The users were positive about the overall idea of the device, the interface, and the automatic connectivity, but wished for a more integrated experience overall. The conclusions that can be drawn from these statements is that from the lay user's perspective, the ideas articulated in this thesis are useful concepts—users valued the summarization of their friends activities and preferences, they valued the simple interface and connectivity of their consumer electronics, and they valued the social aspects of being aware of the actions of the social network.

The users also provided unique perspective on the issue of generalized media recommendations—that even users that were not bots and advertisers might not be valuable sources of recommendations, but instead strength of social connection might be a better indicator of the value of a recommendation. Additionally, what the users found useful was the value of automation: automatically finding new shows to watch based on recommendations, automatically recording a day's worth of shows, and sending it to the television. The aggregation was useful because it lead to automation, but in of itself, without the device giving "the next step," it proved less useful. This jump in utility proves that it is the system as a whole: the Facebook network, the connection the a television guide, and the recording device, as a whole make a good system than any one part of the system working on its own. But, the users also proved that the project could have gone further in this direction towards more connectivity, mining data directly from other devices and creating new connections between consumer electronics without so much concern for privacy. That tact has the pitfalls of bootstrapping, but if implemented on a large scale through a consumer electronics manufacturer, as an optional feature, could greatly enhance how the user interacts with his television as a node of social interaction.

Chapter 5

Conclusion

The simplification of anything is always sensational.

- *G.K. Chesterton, Varied Types*

This last chapter will summarize future directions and possible applications of the ideas presented in the thesis. It will draw conclusions from the results of the evaluation regarding the effectiveness of the implementation. Additionally, it will go over the initially mentioned motivations for the project to evaluate how successful it was at fulfilling the vision for this thesis. Finally, it will analyze what original contributions the project made to its field.

5.1 Effectiveness and Contributions

The thesis made several important contributions to the three areas initially listed as the motivations of this document. Primarily, the thesis was an effective implementation of the CE 2.0 concepts outlined throughout the project. Secondarily, the project showed an effective way of using tacit data from Facebook in a useful piece of consumer electronics. Finally, the diversity of shows that an average user can acquire by using this system exemplifies how new avenues of viral media sharing can be implemented using similar systems.

The CE 2.0 vision gives a good objective measure on the success of implementation of the thesis because it deals primarily with engineering and usability concerns—categories that encapsulate the implementation challenges of the project. Going through the five main principles of CE 2.0—-awareness, simplicity, connectivity, interoperability, and privacy—the project's effectiveness can be measured in a fairly objective fashion. In terms of awareness, the main engineering challenge was to develop a system that would allow a set-top box to be aware of what your friends are watching and in turn improve the recommended viewing choices through feedback. As shown in the first evaluation section, the structure of interconnectivity used for the set-top box allows for it to connect to Facebook even from behind a local network, and allows the user to provide feedback on the shows he watches. In terms of awareness, the implementation was a success. In terms of simplicity, the evaluation in section 4.3 shows through a series of user evaluations that the Facebook application is intuitive and simple to use. It provides the necessary information to the user in a clean fashion, and segments tasks into the appropriate usage scenarios. Setup of the actual box only takes a minute within the Facebook application. As for connectivity and interoperability, a perusal of the proposed method of connectivity in section 3.2 exemplifies a system which is both device and location agnostic, providing the maximum amount of flexibility in the domain. Additionally, the working system shown in section 4.1 illustrates that all the pieces work together through different domains—consumer electronics and the web. Finally, in terms of privacy, the system operates strictly on an opt-in basis, using only data explicitly provided by users when sharing and mining data, although it uses private user data to enhance personalized results. This was an overall design decision instead of an engineering challenge, which was extensively described in section 3.1 of the implementation. Overall, this implementation proves successful and effective at advancing the goals of CE 2.0.

As for the goals regarding tacit data and data transmission, the success of these social aspects of the thesis have been attempted to be measured in both a quantitative and qualitative sense. In a quantitative sense, the effectiveness of data gathering and transmission was measured through a sample of twenty-two users using the Facebook application and measuring the breadth of data that a user would interact with while using the application. As shown in section 4.2, the network of data an average user can be exposed to using this system is much larger than what any individual user puts into the system, showing a great potential for the application to transmit new data. At the same time, given the results of the evaluations completed by users in section 4.3, the data received comes in easy to manage quantities and a simple format, so all data is easily accessible to the users. In a more qualitative sense, the one-week user study shows how a person using the completed system can use the data effectively if delivered straight into consumer electronics. The feedback provided in the long-term study in section 4.4 exemplifies how this system expands the data a user is normally exposed to as well as making it easy and manageable for the user to watch shows his social network is enjoying and watching.

Through the series of experiments described in chapter 4, the effectiveness and range of contributions of this thesis have been demonstrated. However, the goals of the thesis do not stop in this implementation. Hopefully, these ideas will be implemented more fully in a range of consumer electronics and pervasive computing applications.

5.2 Future Directions

This project is only the first implementation of a larger class of consumer electronics that can span the gamut from television applications to mapping devices. Five simple possible extrapolations of the ideas of this thesis will be presented here including:

- Cellphone Address Book
- Music Sharing Devices
- Mapping and GPS Software
- Blog aggregators into consumer electronics
- Social Awareness Applications

All of these new applications suggest new directions in which the ideas presented in the project can be taken. These new directions show how the results discovered in this project can be applied to different mediums. It provides an outlet for the educational nature of this thesis. These future directions can be pursued with the idea that the main themes proposed in this thesis will be widely applicable to these domains.

5.2.1 Cellphone Address Book

Livejournal, Facebook, and Myspace, among other social networking sites, allow users to provide location information and cellphone numbers. However, sometimes that information does not filter from the social networks into the devices themselves. When one is running late to a meeting, trying to find an elusive coffee shop your friends are meeting at, or dealing with some other spur of the moment change of plans, one does not have time to hunt down those numbers. Systems that could that could track numbers of preferred social contacts and intelligently add them to your cellphone or PDA could be of great use to the average person.

5.2.2 Music Sharing Devices

Increasingly, portable music devices come with wifi and other wireless protocols for accessing nearby music devices. Similar to iTunes music sharing on one's desktop, schemes can be implemented to stream from your friend's music devices based on the popularity of certain songs in social networks or personal preference lists like LastFM[13]. In particular, music devices can be coupled with this website, to provide music samples to your device that are recommended by others via radio channel formats or other temporary formats. Sharing music can become more organic without violating DRM standards.

5.2.3 GPS Software

Although more cars and cellphones are equipped with mapping applications, transferring location information from your email, social calendar, or contacts list to your local map still requires preplanning. Suggestions for this future direction include harvesting friends' addresses into the GPS device, adding a calendar's days worth of events into your car, as well as incorporating restaurant recommendations from your blog roll into your rental car when traveling to a new city.

5.2.4 Blogging Aggregators into Consumer Electronics

A more generic application that can be developed, once more consumer electronics devices have more open APIs, is a way to tag blog posts to specific devices in your home. A post on a great new restaurant gets sent to your car, or a fridge display for takeout. A post on a good new movie goes to your television set. Early morning meetings might change the time for your alarm clock or your coffee machine. In general, the idea is to be able to collect data from RSS feeds in the web and send atomized relevant data to specific devices.

5.2.5 Social Awareness Applications

Finally, another application that can be developed is a link between real locations and social networks. Just like Twitter[30] where people post their status during various times of the day. Instead of manually inputting status, blogs and social networks could post status based on different appliances and devices used throughout the day. Locamoda[15] is developing an application to post your location from bars and restaurants to Facebook. In a similar manner, using your coffee machine, driving to work, or calling a friend on the phone could be posted as microblog posts through the integration of consumer electronics and social networks.

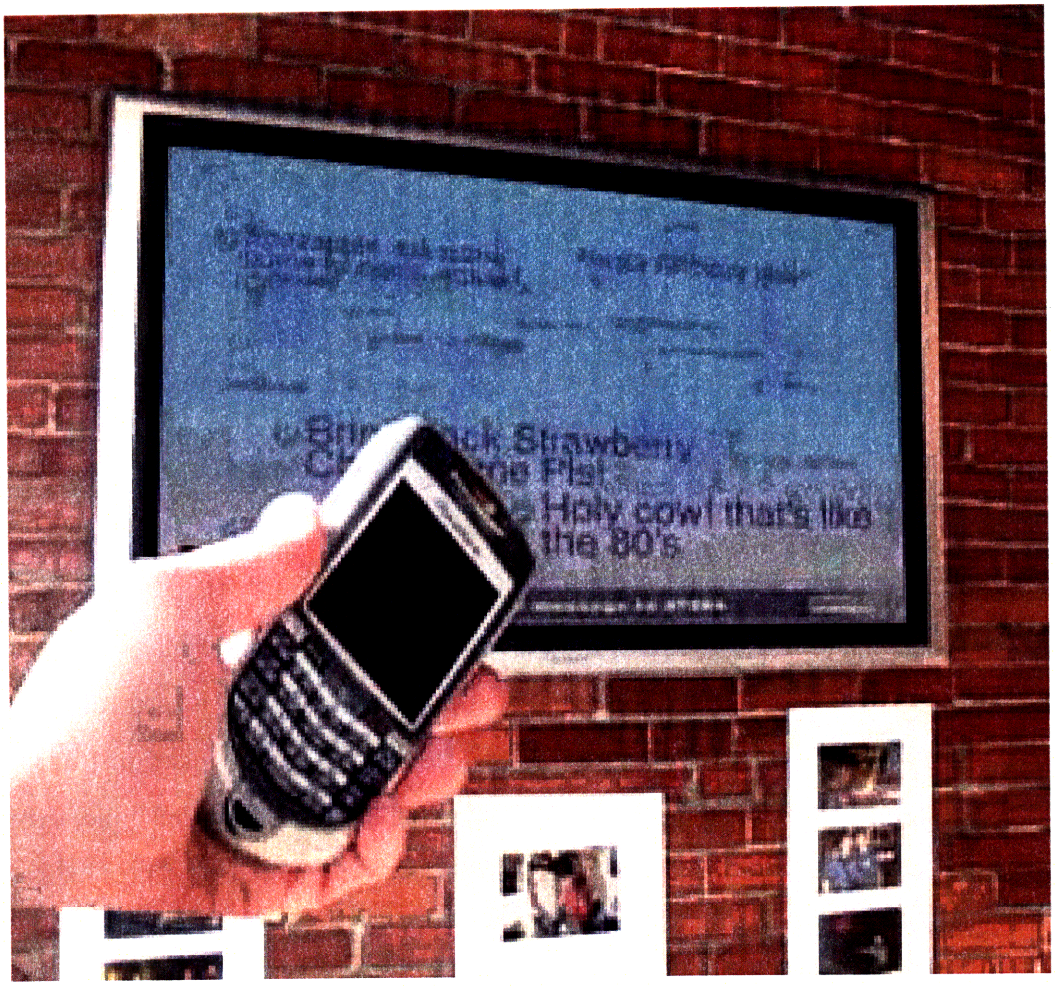


Figure 5-1: Locamoda Screen at Toscanini's

Appendix A

Open I/O

Open I/O was a project started in the Physical Language Workshop in 2005–2006 and continued into 2006–2007. In its themes of exploring remotely programmable devices, limited functionality in favor of simplicity, and control of the environment though ethernet devices, it provides a good starting point towards ideas presented in this thesis. The thesis project was designed on two pre-existing technologies: an ethernet-enabled television set-top box (the Motorola OCAP STB) and the social networking site, Facebook. Open I/O created a framework for the design of internet-enabled hardware devices and a software infrastructure to allow these devices to communicate. Open I/O was a proof-of-concept starting point to the idea of connected hardware devices could interact in a Web environment. The thesis project took it a couple of steps further and used existing internet communities and consumer electronics to exemplify the concept.

Open I/O also provides a web site for building an online community and framework designed to ease the transition between hardware, software, and the Internet. It accomplishes three goals: to make networked hardware devices easy to configure with little knowledge of microelectronics, to allow users to share input device data over the Internet easily, and to allow for seamless distributed hardware applications. Open I/O solves these problems through its multi-tiered system for controlling hardware devices, implementing instantaneous discovery and programmability, and by keeping the set of operations minimal and simple. Through this architecture users can create a

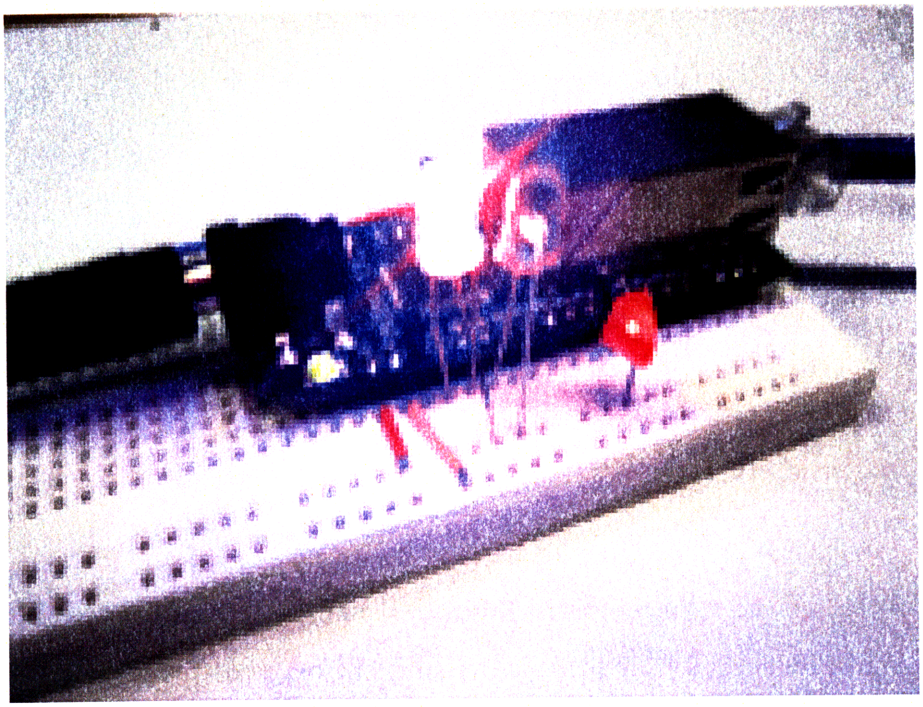


Figure A-1: A web configured Pinkie

great range of applications, ranging from simple tasks of turning devices on and off, to collecting data from wide reaching sensor networks using live data feeds transmitted through the Internet.

The backbone of the Open I/O project is a small header board called the Pinkie. It was designed by Burak Arikan and Vincent Leclerc[4], for the Open I/O Project. It is meant to be a simple development board for Open I/O users to be able to jump into networked micro-controller applications. On top of that layer, is a Dispatcher application—a lightweight JAVA webstart application designed to discover, configure, and relay information to and from the Pinkies. The top layers consist of a central database and the Open I/O API to allow users to configure their Pinkies from anywhere on the Internet as simply as by using a webpage, or as involved as by writing a standalone Java application.

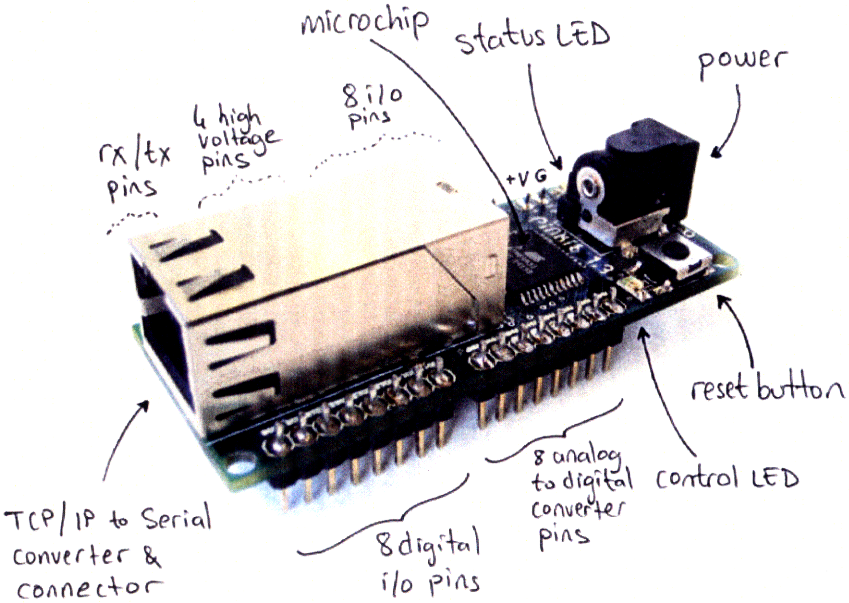


Figure A-2: The different components of the Pinkie

A.0.6 Hardware and Firmware: The Pinkie

The Pinkie is an ATmega32 [5] header board designed by Vincent Leclerc and Burak Arikan. It has a few simple components: the ATmega32 microcontroller, the Lantronix XPort ethernet to serial converter [33], 4 MOSFETs, and 3.3V voltage regulator. The ATmega32 has 8 analog-to-digital converters, and 20 I/O pins available to use(see Figure A-2). Four of those are hooked up to the MOSFETs to allow users to connect high voltage or high current devices, such as motors and relays, to the Pinkie. The other 16 I/O pins are available and each can be individually configured as either an input or output pin.

The Lantronix XPort component serves several important functions for the implementation of the Open I/O infrastructure. XPorts use DHCP to automatically obtain ethernet addresses suitable for the local network. Also, they are discoverable by other machines sharing the LAN without those machines having to know their addresses in advance, and they translate serial signals to and from network packets.

These features combine to allow the Pinkie to be fully network connected and easy to use with no need for end-user set up.

The ATmega32 microcontrollers on the Pinkies are preprogrammed with a simple C command queue. The program has an active serial connection that listens for packets of a specific format. The first character is a delimiter character, indicating the start of the packet. The next byte indicates the ID of the command; by default this number is 0. Next comes a number for the command to be executed (1 = getAN, 2 = getIO, 3 = setIO, 4 = set HV). The next byte is the pin on which the command is to be executed. Finally, the last byte is the value the pin is to be set to in the case of setHV or setIO (either 255 or 0). Each command received is parsed and entered into the queue. If the control software receives a command for a specific pin with the same ID as a command already in the queue, the old command is overwritten. The control program executes all the commands on the queue in a loop, turning pins on and off for the set I/O and set HV commands, or tri-stating and reading the pins for the get I/O and getAN commands. Also, when a "get" command is executed, the control program reads the current value on the corresponding pin, and then sends a packet on the serial line with the value of the pin it just read. The XPort relays the command to the Dispatcher, which parses the command. The outgoing packets are in the same format to the received packets.

A.0.7 Software: Dispatcher

The Dispatcher is a critical element in routing communications between the central server, Open I/O webpage, and Pinkies. It performs many functions: discovery, authentication, configuration, dispatching information from the Pinkies to other Dispatchers, applications, and webpages, and subscribing Pinkies to each other (see Figure A-3).

The Dispatcher is a JAVA webstart application that can be downloaded from the Open I/O webpage. The first thing the application does is authenticate the user as a member of Open I/O, in order to register the Dispatcher in the database. Each user can only have one Dispatcher running at a time. A Dispatcher, once logged on, will

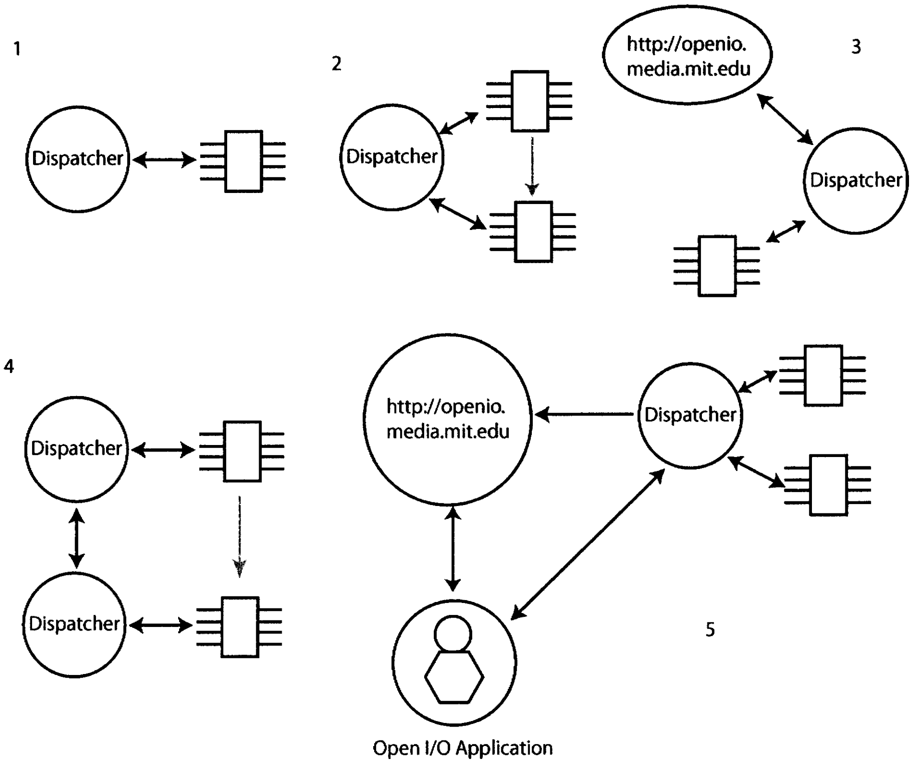
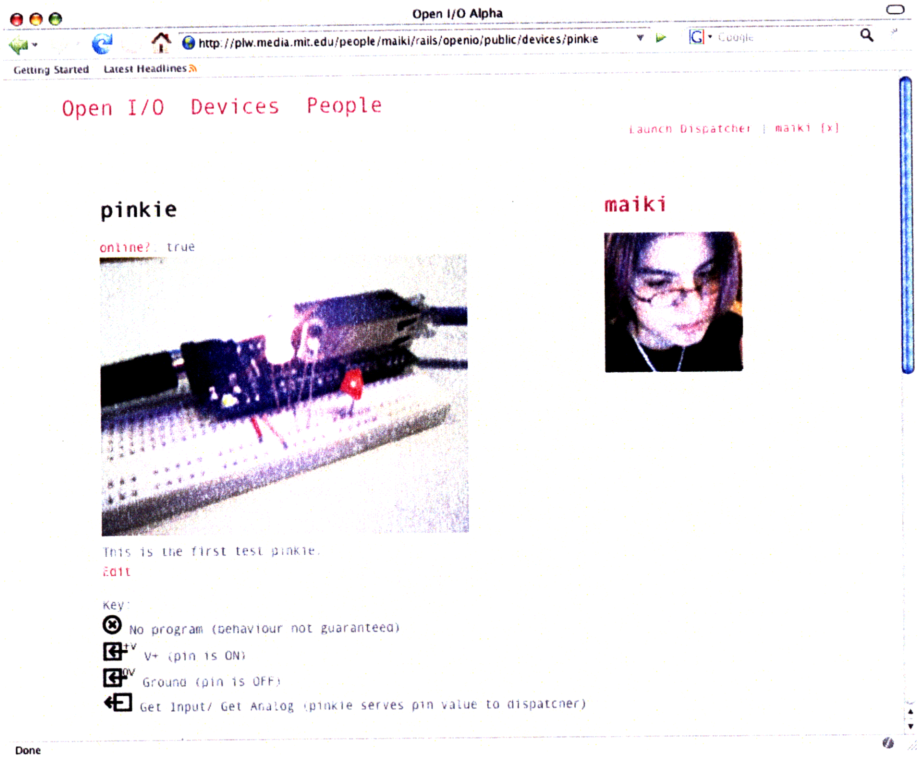


Figure A-3: Different types of interactions handled by the Dispatcher. Black lines indicate direct connections, grey lines indicate indirect connections. 1: Dispatcher configures a Pinkie, 2. Dispatcher subscribes two Pinkies it controls. 3. Dispatcher routes commands from central server, 4. Two Dispatchers subscribe two Pinkies, 5. Dispatcher routes command from a 3rd application to Pinkies.


Open I/O Alpha
http://plw.media.mit.edu/people/maiki/rails/openio/public/devices/pinkie
Google
Getting Started
Latest Headlines
Open I/O
Devices
People
Launch Dispatcher | maiki [x]
Done

pinkie

online? true

This is the first test pinkie.

Edit

Key:

⊗ No program (behaviour not guaranteed)

+V V+ (pin is ON)

0V Ground (pin is OFF)

Get Input/ Get Analog (pinkie serves pin value to dispatcher)

maiki

Figure A-4: Device Status Page.

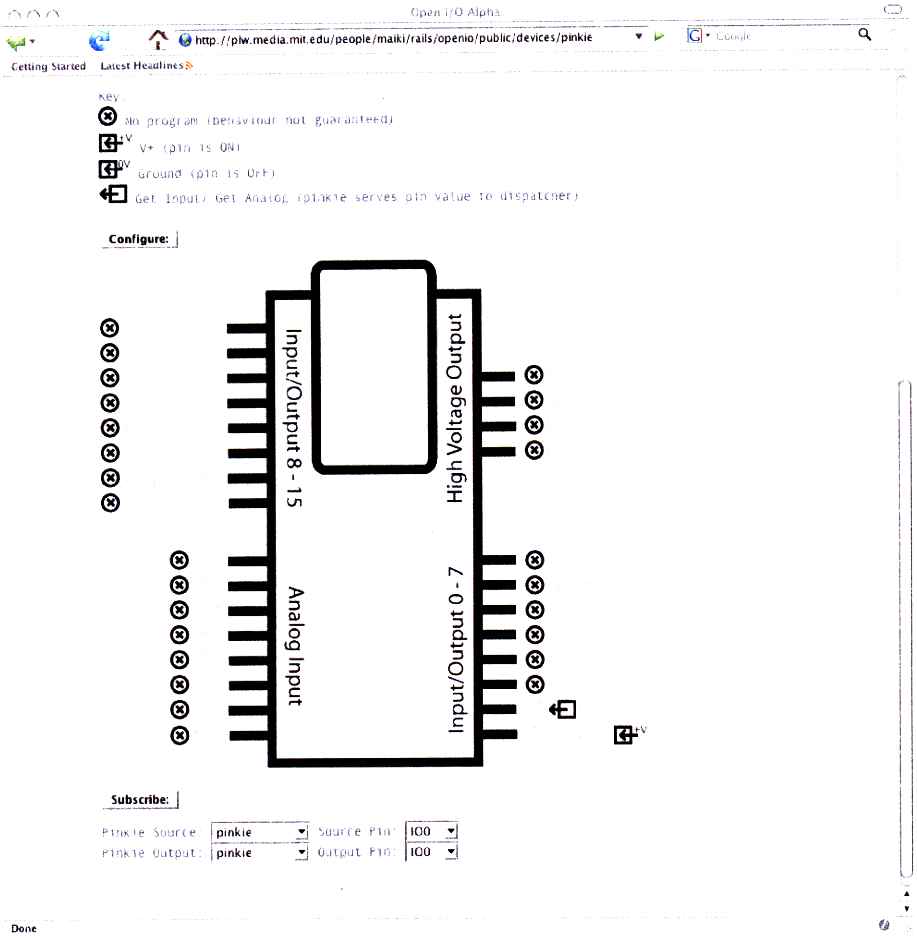
discover all Pinkies in its local network by pinging for all nearby Lantronix Xport devices. Once it discovers the devices, it checks with the Open I/O database if they are registered with another user. If they are not, those Pinkies will be controlled by this Dispatcher. The Dispatcher can run the device discovery protocol at any time, either remotely through the Open I/O API or through the Dispatcher's UI. Ownership of Pinkies is intended to be mutable and dependent on location, to allow easy transfer of ownership but prevent usage or control of Pinkies by anyone not in the proximity of them. Thus users control any Pinkies their Dispatcher finds. This also regulates that a Dispatcher running on a local computer is only routing communications for Pinkies one wishes to configure, to save memory and resources.

A.0.8 The Web: http://openio.media.mit.edu

The Dispatcher also configures, subscribes, and receives data from Pinkies. Configuring a command is done through an XML-RPC request sent from a Java Application or a server request. The Dispatcher keeps a record of all the values it sets or reads from a particular Pinkie so that it can relay the values to webpages or programs that require the status of the Pinkie. If a command is a getIO or getAN command, the Dispatcher has to set up a thread to listen for data packets sent from the Pinkie. Whenever the Pinkie sends a new packet, the Dispatcher stores a new value for that pin on that Pinkie. When a webpage or application sends a XML-RPC request for the input value of a given pin, the Dispatcher will have the latest value from the Pinkie ready to be relayed on. Finally, a Dispatcher can subscribe two Pinkies together. The source Pinkie can be any public Pinkie known to the Open I/O database, but the receiving Pinkie must belong to this Dispatcher.

A complicated situation arises when many individuals wish to run Dispatchers for their several Pinkies within one subnetwork. Open I/O's Dispatcher handles this problem by using a first come first serve method of conflict resolution. When a Dispatcher first runs its discovery method, it locates all the Pinkies in its subnet. It then asks the central server which ones already have a Dispatcher. The Dispatcher thus can only control unclaimed Pinkies. If a second person comes into the area and plugs in their Pinkie and runs the Dispatcher program, the second Dispatcher will only control the new Pinkie, since the central server will tell the Dispatcher that all the other Pinkies are being controlled by another Dispatcher. If either party wants to plug in more Pinkies, they need only call the discovery method on their Dispatcher, and the central server will register this new unregistered Pinkie to them, and the Dispatchers will retain ownership of their other Pinkies. In order to keep ownership of a Pinkie, one must always keep a Dispatcher running as long as the Pinkie is connected to the Internet. Once a Dispatcher is turned off, those Pinkies can be recaptured by a new Dispatcher if they remain on the network.

Open I/O's backbone is a centrally located database and webpage at the address: http://openio.media.mit.edu/ . The central server stores all the existing usernames, and keeps track of active Dispatchers and registered Pinkies. It allows users to log in, check the status of their devices, and even do simple configuration tasks from a web interface (see Figure A-4). Additionally, it communicates with the Dispatcher to



Key:

⊗ No program (behaviour not guaranteed)

+V V+ (pin is ON)

0V Ground (pin is OFF)

Get Input/ Get Analog (pinkie serves pin value to dispatcher)

Configure:

Subscribe:

Pinkie Source: pinkie ▾ Source Pin: IO0 ▾

Pinkie Output: pinkie ▾ Output Pin: IO0 ▾

Figure A-5: Web configuration Interface

report if a Pinkie is already discovered by a different Dispatcher (to prevent double ownership of a device), and to validate users. In return, the Dispatcher provides the database with all code on the Pinkies and the IP address of itself and the MAC addresses and IP addresses of all of its Pinkies.

The website is programmed in Ruby on Rails. It uses the XML-RPC protocol to communicate with Dispatchers and other applications using the Open I/O Library. The database stores the users, and that person's active Dispatcher and Pinkies. It also stores any active code on a Pinkie, as well as whether it is online or not, and the current IP address associated with the Pinkie. The Pinkie's MAC address is the device's unique identifier in the Open I/O system. With this information, the central server can effectively route communications between other applications using the Open I/O Library and the Dispatchers, and ultimately, to the Pinkies.

A.0.9 JAVA Library and API

In order for the Pinkies to be fully programmable, a more sophisticated API than a simple web interface needed to be developed. In order to integrate Open I/O into most JAVA applications there is a simple Open I/O Java Library and API. This library is also compatible with the programming language Processing [22] created by Casey Reas and Ben Fry, which is popular among digital artists, and is a subset of JAVA. To run the library, the XML-RPC and commons-codec-2.0 libraries need to be installed. Both are available from the Apache website for free.

The Open I/O API consists of four main classes: an Open I/O user class, a Dispatcher class, a Public Pinkie parent class, and the Pinkie class. The User class allows an individual to log into the Open I/O database. The user can then get information regarding his Dispatcher, any Pinkies he currently has online, and any data available on publicly accessible Pinkies. Users can only configure Pinkies they have registered under their Dispatcher. However, they can get data from any Public Pinkie registered on the Open I/O database. This allows multiple users to create programs in which their Pinkies interact remotely. Once the Open I/O Library gets the Dispatcher information from the server, the application forms a direct connection

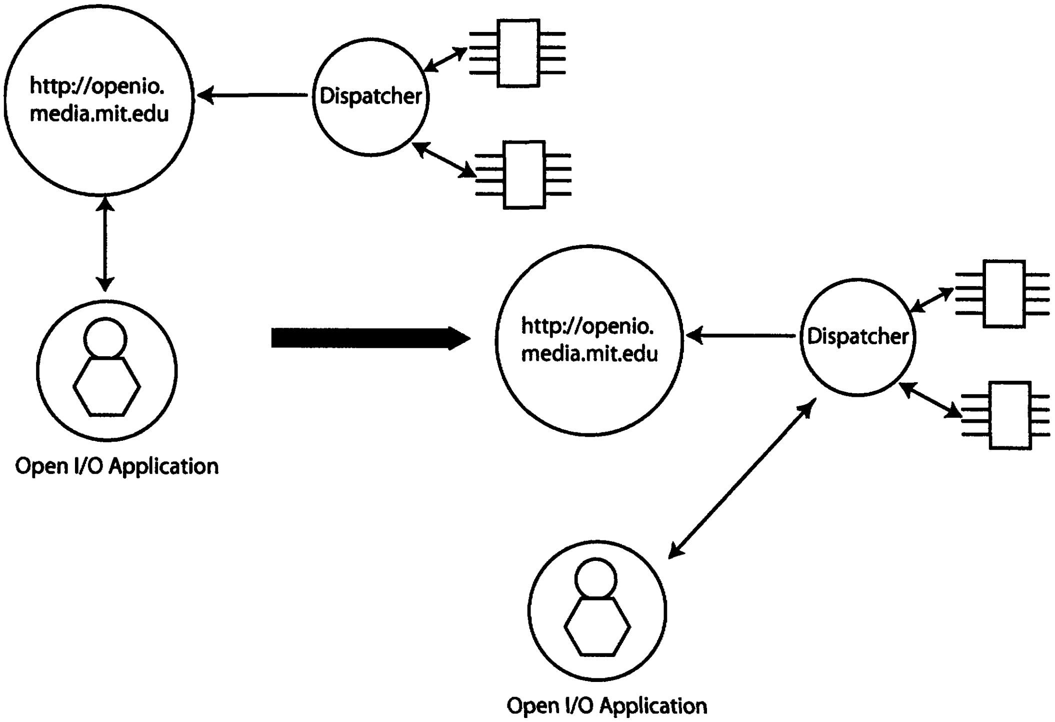
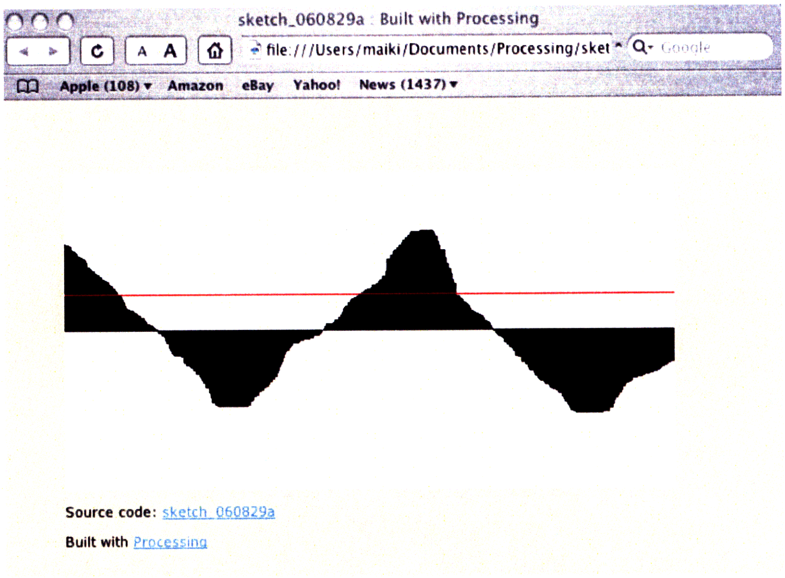


Figure A-6: How communications are routed from the central server to an application transform into a direct connection between the Dispatcher and the application.


sketch_060829a : Built with Processing
file:///Users/maiki/Documents/Processing/sket
Google
Apple (108)
Amazon
eBay
Yahoo!
News (1437)

Source code: sketch_060829a

Built with Processing

Figure A-7: A Pinkie oscilloscope programmed in Processing.

to the Dispatcher, instead of constantly going through the central database (see Figure A-6).

A.0.10 Applications

Online Oscilloscope

One simple application of Open I/O is a small oscilloscope applet (see Figure A-7). It can run on any webpage to monitor the status of a device in the real world by hooking up an output pin to one of the Pinkie's analog input pins. This particular example was written in Processing(see Figure A-8), a lightweight version of JAVA intended for digital artists.

Home Sensor Network

As a project for Joe Paradiso's Sensor Networks and Environments class at the Media Laboratory, Open I/O was used to develop a home sensor network to track a pet's daily activities while away from home. A variety of sensors including optical, vibration and capacitance sensors were connected to Pinkies in order to get a live web feed of the pets current activities and necessities.

```
import Pinkie.lib.OIOUser;
import Pinkie.lib.Pinkie;
OIOUser me;
Pinkie inputPinkie;
int[] values;
void setup() {
     size(500, 255);
     values = new int[width];
     loop();

     try {
         me = new OIOUser("maiki", ****);
         inputPinkie = me.getPinkie("00:20:4a:8d:86:2d");
         inputPinkie.configure("getAN", 7, 0);
     } catch (Exception e) {
         e.printStackTrace();
     }
}

void scope(int input){
     for(int i=1; i<width; i++) {
         values[i-1] = values[i];
     }

     values[width-1] = height/2 - input;
     stroke(0);

     for(int i=0; i<width; i++) {
         line(i, height/2, i, (height/2) - values[i]);
     }
}

void draw() {
     background(255);

     try {
         int input = inputPinkie.getAnalogValue(7);
         scope(new Integer(input).intValue());
     } catch (Exception e) {
         e.printStackTrace();
     }
}
```

Figure A-8: `oscilloscope.pde` – Processing Code for Pinkie Oscilloscope

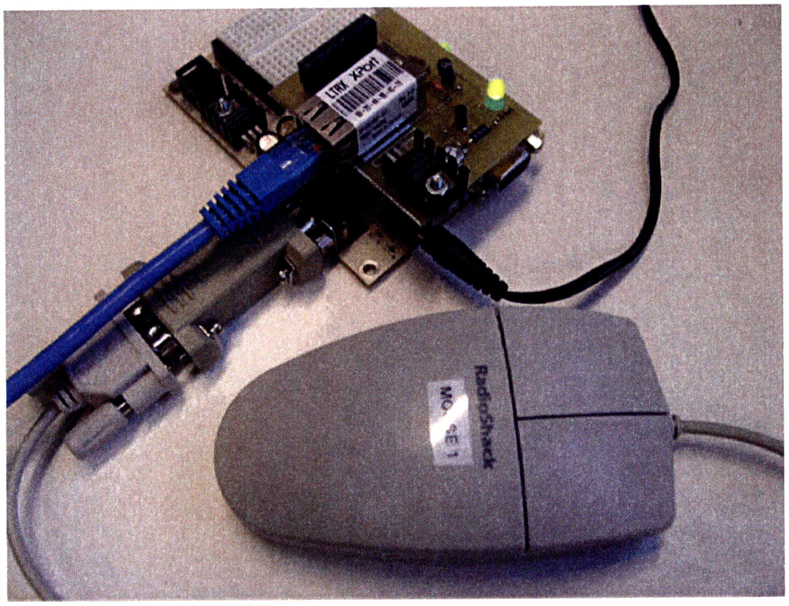


Figure A-9: Lantronix Xport Networked Mouse.

Simple Remote Device Configuration

Lights and Motors can be dynamically switched on and off through Open I/O's simple web interface. Pinkies all over the world can be controlled in this fashion. Pinkies have been used in art Installations in the US (Cambridge, MA, Oklahoma City, OK), Germany (Berlin), and Turkey (Istambul). LED's and current relays can be controlled directly through the I/O pins and motors and solenoids can be controlled though the high Voltage pins directly through the web (see Figure A-5).

Multi User Drawing Application

Multimodal applications are simple to implement using Open I/O. For example, with two sliders or knobs hooked up to two I/O pins of two different Pinkies. An two person Etch A Sketch application can be made easily with these 4 components. A previous project, the Multimouse application (see Figure A-9), hooked up serial mice to Lantronix Xports to make an Internet enabled multi-user drawing application demonstrating that Pinkies operate at a fast enough rate to allow for real-time multi-user drawing applications to work through the Internet.

Bibliography

[1] Alessandro Acquisti and Ralph Gross. *Imagined Communities: Awareness, Information Sharing, and Privacy on the Facebook.* 2006.

[2] Gediminas Adomavicius and Alexander Tuzhilin. Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *IEEE Transactions on Knowledge and Data Engineering*, 17(6):734–749, 2005.

[3] Amazon.com. Website Last Retrieved May 2008, 2006. http://www.amazon.com/.

[4] Harun Burak Arikan. Collective systems for creative expression. Master's project, MIT, Media Laboratory, 2006.

[5] 8-bit avr microcontroller with 32k bytes in-system programmable flash data sheet. Datasheet, Atmel Corporation, 2007. http://www.atmel.com/dyn/resources/prod documents/doc2503.pdf.

[6] Directv scheduler. Website Last Retrieved May 2008, 2008. http://www.directv.com/DTVAPP/customer/faqPage.jsp?assetId=P4400098.

[7] Judith Donath. Signals in social supernets. *Journal of Computer-Mediated Communication*, 13(1):231–251, 2007.

[8] Judith Donath and Danah Boyd. Public displays of connection. *BT Technology Journal*, 22(4), October 2004.

[9] Facebook factbook. Website Last Retrieved March 2008, 2008. http://www.facebook.com/press/ info.php?factsheet.

[10] Erich Gamma, Richard Helm, Ralph Johnson, and John Vlissides. *Design patterns: elements of reusable object-oriented software.* Addison-Wesley Professional, 1995.

[11] Henry Holtzman and Michael Bove. Consumer electronics 2.0 white paper. Technical report, MIT Media Laboratory, Media Laboratory, Cambridge, Massachusetts, October 2006. http://wiki.media.mit.edu/view/CE20/WebHome/.

[12] Fbjs: Facebook javascript. Website Last Retrieved May 2008, 2008. http://wiki.developers.facebook.com/index.php/FBJS.

[13] Last.fm. Website Last Retrieved May 2008, 2002. http://www.last.fm/about/.

[14] Greg Linden, Brent Smith, and Jeremy York. Amazon.com recommendations: Item-to-item collaborative filtering. *IEEE Internet Computing*, 07(1), 2003.

[15] Locamoda. Website Last Retrieved May 2008, 2008. http://www.locamoda.com/.

[16] J. Lull. Inside family viewing: Ethnographic research on television's audiences. *Canadian Journal of Communication*, 18(3), 1993.

[17] Anish Karmarkar Mark Jones, Jeff Mischkinsky, Richard Martin, Oisin Hurley, Lynne Thompson, and Hugo Haas. Soap version 1.2 specification assertions and test collection (second edition). W3c recommendation, W3C, April 2007.

[18] Duane Merrill. Mashups: The new breed of web app. http://www.ibm.com/developerworks/xml/library/x-mashups.html, August 2006.

[19] Jakob Nielsen. Enhancing the explanatory power of usability heuristics. In *CHI '94: Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 152–158, New York, NY, USA, 1994. ACM.

[20] L. OEHLBERG, N. DUCHENEAUT, J.D. THORNTON, R.J. MOORE, and E. NICKELL. Social tv: Designing for distributed, sociable television viewing. In *Proceedings of the EuroITV 2006 Conference*, pages 251–259, 2006.

[21] Facebook developers: Get to know the platform. Website Last Retrieved May 2008, 2007. http://developers.facebook.com/resources.php.

[22] Casey Reas and Ben Fry. *Processing: A Programming Handbook for Visual Designers and Artists.* MIT Press, Cambridge, Massachusetts, 2008.

[23] Anxo Cereijo Roibás, David Geerts, Elizabeth Furtado, and Licia Calvi. Investigating new user experience challenges in itv: mobility & sociability. In *CHI '06: CHI '06 extended abstracts on Human factors in computing systems*, pages 1659–1662, New York, NY, USA, 2006. ACM.

[24] R. Schatz, S. Wagner, S. Egger, and N. Jordan. Mobile tv becomes social - integrating content with communications. *Information Technology Interfaces, 2007. ITI 2007. 29th International Conference on*, pages 263–270, 25-28 June 2007.

[25] Raimund Schatz, Norbert Jordan, and Siegfried Wagner. Beyond broadcast–a hybrid testbed for mobile tv 2.0 services. In *ICN*, page 87, 2007.

[26] Schedules direct. Website Last Retrieved May 2008, 2008. http://www.schedulesdirect.org/.

[27] Mike Schwartz. Cablelabs publishes ocap middleware specifications. http://www.cablelabs.com/news/pr/2002/2002 01 03.html, January 2002.

[28] Upendra Shardanand and Patti Maes. Social information filtering: Algorithms for automating "word of mouth"'. In *Proceedings of ACM CHI'95 Conference on Human Factors in Computing Systems*, volume 1, pages 210–217, 1995.

[29] Tivo home media engine sdk. Website Last Retrieved May 2008, 2006. http://tivohme.sourceforge.net/.

[30] Twitter. Website Last Retrieved May 2008, 2006. http://twitter.com/faq.

[31] Amy Voida, Rebecca E. Grinter, Nicolas Ducheneaut, Keith K. Edwards, and Mark W. Newman. Listening in: practices surrounding itunes music sharing. In *CHI '05: Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 191–200, New York, NY, USA, 2005. ACM Press.

[32] Xmltv. Website Last Retrieved May 2008, 2007. http://xmltv.org/wiki/.

[33] Xport- embedded ethernet device server. Website Last Retrieved May 2008, 2007.