Universität Karlsruhe (TH)
Institut für theoretische Informatik

Prof. Dr. Alex Waibel
Fakultät für Informatik
Universität Karlsruhe (TH)

MyConnector - Design and Implementation of an Adaptive Context-aware Communication Service

Diplomarbeit
Tobias Kluge

Advisors:
Dipl. Inform. Maria Danninger
Dr. Rainer Stiefelhagen
Prof. Dr. Alex Waibel

31. März 2006

Ich erkläre hiermit, dass ich die vorliegende Arbeit selbständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel verwendet habe.

Karlsruhe, den 31. März 2006

Abstract

With the advance of network technologies, new communication channels and mobile devices, the way of communication has changed tremendously over the last 10 years. In this thriving world of mobile communications, the difficulty of communication is no longer contacting someone (the receiver), but rather contacting them in a socially appropriate manner. Ideally, senders should have some understanding of the receiver's availability to get in contact at the right time, in the right contexts, and with the optimal communication medium. The Connector is an adaptive and context-aware service designed to facilitate efficient and socially appropriate communication.

This thesis describes the design and implementation of MyConnector, an extension of the Connector service that leverages machine-learning techniques to model contextual knowledge about the user to infer the user's availability for communication. Input comes from automatically detected context cues such as calendar information, PC activity and the current activity in the office, as well as self-reported user feedback. A set of user studies were conducted to investigate and optimize the design of the MyConnector service and detect the best context cues to predict people's availability. The level of information granularity being displayed is user-defined in the owner's privacy settings. MyConnector clients run as WinXP application and as web interface.

Acknowledgements

Thank goes to my advisors Maria Danninger, Rainer Stiefelhagen and Alex Waibel, who gave valuable feedback and ideas for this work; also to Christian Wojek (for the provided office activity data) and the participants of the user studies and the survey.

Contents

List of Figures

List of Tables

1. Introduction

1.1 Motivation

The way of communication has changed tremendously over the last ten years. Today one can choose between more and more communication channels like email, instant messaging, mobile phones and VoIP. With mobile communication devices such as cell phones, smart phones and Blackberries as well as Wifi networks all over the place, people are reachable nonstop - via voice calls, emails, short messages or the like. Nowadays the problem is not anymore to contact somebody - but *how* to contact somebody *in a socially appropriate manner*!

These new communication technologies bring considerable advantages, as well as burdens, to both the sender and the receiver in a communication. The sender seems to be favored over the target of communication, the receiver. The sender calls when the situation is conducive to communication, but does so with little knowledge of the receiver's situation. This problem is further exacerbated with the advance of mobile communication, which decouples location from situation and thus decreasing the capacity for a sender to make informed decisions about the person they are calling. The sender has no information about the current situation of the receiver - which could be a meeting, swimming or during public transportation. Ideally, the sender should have some understanding of the availability of the receiver to contact him at the right time, in the right context and with the optimal communication media.

A system that acts like a proxy between the parties in a communication and knows the availability would improve the communication. But to be accepted by its users, such a system has to adapt automatically to its users - by learning their availability and behave as they would - for example automatically accept a communication request or reject it in an inconvenient situation.

1.2 Project CHIL and the Connector service

The framework of the **CHIL project - Computers in the Human Interaction Loop -** intends to develop context-aware, proactive computer services that assist people during daily interactions with others [CHIL]. Rather than expecting people spend their time attending to technology, CHIL's goal is to develop computer services that are sensitive in attending to human activities, interactions, and intentions. To act in a proactive yet implicit way, services should be able to identify and possibly even understand human activities.

The Connector is a CHIL service designed to intelligently connect people at the right place, the right time, and with the best possible medium for socially appropriate communication. It maintains an awareness of its users' activities, preoccupations, and social relationships to mediate a proper moment and medium of connection between particular people. In this system, personal agents act as virtual administrative assistants, who know how to selectively facilitate some calls while blocking others.

1.3 Scope of this work

This thesis describes the design and implementation of *MyConnector*, a modern communication-supporting service and an extension of the Connector service. MyConnector leverages machine-learning techniques to learn the availability of its users. By broadcasting the receivers availability, it allows senders to make more informed decisions and thus supports users to communicate in a more socially appropriate manner. MyConnector provides the opportunity to specify personal privacy settings to infer what information should be accessible to whom and when.

The system learns user availability from automatically collected context cues such as calendar information, PC activity and the current activity in offices - as well self-reported user input, such as the current activity, but also more subjective measures like mental and physical engagement in the current activity. A pilot study comparing the results from different classifiers to the self-reported availability level was conducted to detect the best context cues to predict availability.

1.4 Organization of this thesis

Related work is described in Chapter 2. Design and Implementation of the MyConnector prototype is introduced in Chapter 3. Learning the user's availability is discussed in Chapter 4, and related privacy issues are described in Chapter 5. Conclusion and an outlook into future work can be found in Chapter 6.

2. Related work

Papers of different research areas have influenced this work. Section 2.1 presents papers that investigate interruption and availability. The development of the My-Connector prototype is inspired by systems described in Section 2.2. Related work on privacy - an important aspect of such a system - is described in Section 2.3.

2.1 Interruption and availability

Different groups have analyzed the effect of interruptions on activities. Brian Bailey investigated in [BaKC00] the disruptive effect of interruptions on the performance of the user's task. The results are that

> (i) a user performs slower on an interrupted task than a non-interrupted task, (ii) the disruptive effect of an interruption differs as a function of task category, and (iii) different interruption tasks cause similar disruptive effects on task performance.

This means that the interruption of the user - e.g. for a communication request - should be depending on what the user is currently doing. In the tasks of his paper (e.g. calculating, reading, registration) the completion time for the interrupted task was significant longer than for the uninterrupted task. Bailey suggests that

> an application should wait for an opportune moment such as when the user reaches a task boundary or during a period of low interaction

when it is necessary to interrupt the user. He believes that future systems should

> recognize some basic set of tasks performed by a user, [...] observe or predict when the user is switching between tasks [...] [or should integrate] other parameters of an interruption such as urgency or task relevance into the calculation of an "'opportune moment"'.

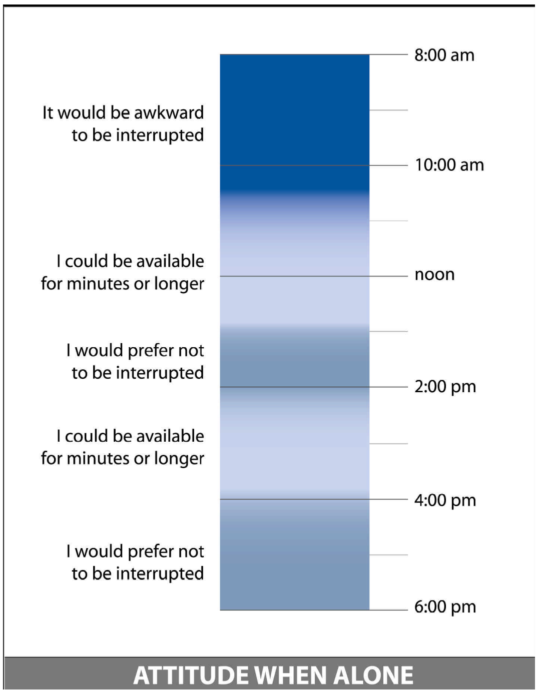

Figure 2.1: Interruption over time for one day, taken from [HCKE02]

Similar results have been made by Piotr Adamczyk and Brian Bailey in a later study ([AdBa04]) where they investigated the impact of different interruption moments on the user's emotional state. Their system predicted the best points for an interruption which resulted in less annoyance, frustration and time pressure of the user. To gain good results they broke down the tasks that the user had to perform into smaller subtasks and choose the best point between these subtasks to interrupt the user.

Hudson et all describe in [HCKE02] the availability of corporate IBM research managers where they analyzed the daily schedule of these people. The participants weared a RIM Blackberry and had to fill a short survey 10 times a day. Figure 2.1 shows a summery of the interruptibility for one day where the lighter color indicates a better availability. The attitudes toward interruption based on the activity of the subjects are visualized in figure 2.2.

2.2 Predicting availability

The group around **Eric Horvitz** and Paul Koch from Microsoft Research developed several systems that help to notify users about information and communication requests:

- *Priorities* - a email prioritization and message relay system ([HoJH99])
- *Notification Platform* - a general notification framework (also [HoJH99])
- *Bestcom* - a communication software that contacts sender and receiver depending on context ([HKKJ02])
- *Coordinate* - a system that forecasts presence and availability ([HKKJ02])
- *BusyBody* - a supervised learning and inference system ([HoKA04])

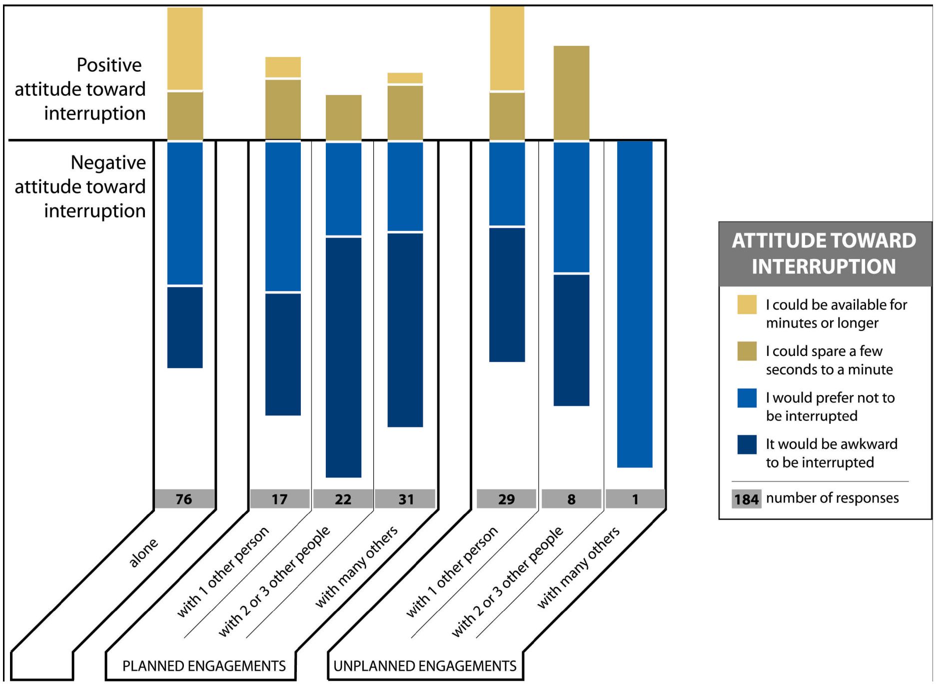

Figure 2.2: Interruptibility depending on activity, taken from [HCKE02]

Especially the systems Bestcom and Coordinate are relevant. Coordinate does a forecasting of the availability of a user - e.g. when the user is out-of-office and the time he will come back. Different sensors as calendar information, computer vision analysis, ambient acoustical analysis and the position based on wifi strength and GPS are feed into Bayes networks to learn and predict the users availability. Coordinate can - for example - predict the probability of the attendance of participants of a meeting or forecast when the user will be in the office again. The expected costs of an interruption can also be computed, which allows the system to decide what interruption should go through to the user.

Bestcom, a communication service, uses this functionality to inform the sender automatically about the best communication media depending on the goals and context of the sender and the receiver. It serves as a proxy and maintains the privacy about the receivers' state. The functionality offered to the sender are

> establishing a real-time connection on the same channel, shifting to another channel, taking a message, and providing the contactor[sender] with better times for communicating with the contactee[receiver], coupled with services to schedule and manage the future communication.

James Fogarty, working for the Human Computer Interaction Institute at Carnegie Mellon University, published different papers related to the field of availability. In [FKAG+05] he describes the steps that have been made to implement a system that takes the task engagement as a additional input value for a system that learns and understands the availability of the user. The subjects of the study had to perform programming tasks which have been monitored and analyzed. Six different groups of (simulated) sensors for the programming task have been evaluated: reading code, coding, doing code navigation, doing interface navigation, doing task switching

group	performance
Manager	87.7%
Researcher	81.1%
Internal (shared office)	80.1%

Table 2.1: Performance of different groups, source: [FoHL04]

and doing debugging. The seven most predictive features are strongly related to the programming task and contains for example control resize events, tree events and switching between different editor windows. For learning the availability also bayesian networks and sensor-based statistical models of human interruptibility have been used.

In another study [FoHL04] Fogarty investigated the interruptibility in office environments. Different sensors as microphones, magnetic sensors (to detect if a door activity was opening or closing, another to detect if the phone was physically off the hook), motion sensors next to the door and the subject's desk, and software installed on the subject's PCs that logged the number of keyboard events, mouse moves, mouse clicks, application information (title, type, executable name of each application). The subjects have been asked via speaker to provide their availability level from 1 (highly interruptible) to 5 (highly non-interruptible) and they had to answer in the following 10 seconds - orally, recorded by a microphone. Naive Bayes classifiers have been used to learn and predict the availability. Best results have been made with the following sensors:

- Phone activity in the last 15 seconds
- User is talking
- Computer activity
- Motion detector (in offices with more than one worker)

Different groups of subjects have participated, the best results for each of them can be seen in table 2.1.

Fogarty developed a learning toolkit called *AmIBusy* (see [Foga04]) that deals with input from different sensors especially for context-aware applications. Doing that it

> seeks to abstract *what* information was collected [by providing] a high-level abstraction that can be estimated from whatever sensors are available in an environment.

It automatically adapts the internal statistical model for new or disappearing sensors.

Fogarty also describes in [FoLC04] a system called *MyVine* that shows users of the system - similar to MyConnector - the availability of their contacts for different communication media. It is based on speech detection, location information (based on the network connection), computer activity and calendar information. 26 people used the system for four weeks. Email, phone and instant messaging have been used as communication channels. Sadly, no detailed information about the data and the performance of the system are available.

Social Relationship	General Relationship
Family	Social
Friends	Social
Acquaintances	Social (weak)
Strangers	Social (weak), Legal
Charities	Social (weak)
Employer	Market, Legal, Social (weak to moderate)
Co-workers	Social (weak to moderate), Legal
Companies	Market, Legal

Table 2.2: Some example relationships and their categorization, source: [HNLL04]

2.3 Privacy in ubiquitous computing environments

Privacy is especially important to users when it comes to data that allows somebody to trace them. For example the current instant messaging tools allows other users to see if their contacts are online or offline.

Different research groups offer different methodologies to protect the personal data in ubiquitous computing. Scott Lederer et all investigated in [LeMD03] the factors *inquirer* and *situation* in relation to privacy preferences of users. The results show that the subject that requests information is a stronger indicator than the current situation of the user. He points out that individuals were more likely to apply the same privacy preferences to the *same inquirer* in *different situations* than to apply the same privacy preferences to *different inquirers* in the *same situation*.

Sameer Patil investigated in [PaKo04] the usage of instant messaging. He describes three big privacy concerns: privacy from non-contacts, privacy regarding availability, and privacy regarding the content of IM communication. The main point in his opinion is that the user wants to control how one appears to others. For the development of instant messaging application - which can also be extended for all ubiquitous applications - he means that such programs should provide modifiable policies and settings with suitable defaults and seamless interaction.

Hong et all introduced in [HNLL04] a privacy risk model that allows designer of ubiquitous computing applications to analyze their system for the risk of abuse. The analysis includes questions regarding social and organizational context and the used technology. He also provides a table with relationships and their categorization that can be seen in Table 2.2. The general relationship is based on Lessig's framework [Less99] that describes the categorization of relationships into market, social, legal and technical forces.

Employments of privacy for ubiquitous applications can for example be found in [HoLa04] (Hong describes the privacy framework *Confab* for ubiquitous computing applications and location) and [SaEN05] (Sacramento introduces CoPS, a system that allows users to control their data with complex privacy rules).

3. Design and Implementation of MyConnector

This chapter describes the design and implementation concepts of MyConnector. It starts with an online survey that has been conducted to get ideas about communication preferences and needs. Section 3.2 introduces the overall architecture of the Connector service, in which the modules developed within this thesis are embedded. Afterwards, detailed information about the developed client applications will be presented.

3.1 An introductory online survey

To get an idea what functionality of such a system would be expected and requested by potential users, an initial survey has been performed. The questions concerned demographic information, favorite communication media and patterns of communication and finally a set of questions on privacy preferences in various situations.

Invitations to participate in this survey have been sent to internal mailing lists of the Interactive Systems Labs (Karlsruhe), CHIMe Lab (Stanford), the CHIL project as well as personal contacts. 49 people from Europe and North America participated in this survey.

3.1.1 Demographic information

Demographic information about the age, the gender, the nationality and the number of co-workers have been collected.

Age

45 out of 49 users specified their age. The minimal age was 23, the maximum age 61. The average person was 34.24 years old.

Gender

45 users replied to this question. 10 of them were female, 35 were male.

Nationality	number of participants
German	21
Austria	4
Swedish	3
Greek	3
Australian	2
Italian	2
American	1
Chinese	1
French	1
Romanian	1
Swiss	1
Spanish	1
British	1
Czech	1

Table 3.1: Nationality of the online survey participants

Nationality

43 users entered their nationality, table 3.1 displays the result ordered descending by the number of subjects per nationality.

Number of co-workers

Each participant stated how many members were in the most immediate group of co-workers. The answers vary from 0 to 30 with a mean of 11.2.

3.1.2 Questions about communication

The first questions that have been asked were about the typical communication behavior, the used programs and preferred communication media.

Patterns of communication

Table 3.2 shows the result to the question *What would be relevant information for you when deciding how to contact someone?*.
Most people would find the information when the receiver would be available next time (85.7%) and his/her current location (51%) most useful. Surprisingly, people did not seem interested in what the other person's preferred communication media (4%) is.

There seems to be a trend towards information that is already implemented in currently available applications and potentially already used by the subjects. This would as well explain why the other person's preferred communication media was seen as irrelevant - maybe because this feature is not yet available.

Frequently used communication media

Answers regarding frequently used communication media are displayed in table 3.3.

Obvious is the number of email users - the survey-invitations have been sent via email. One side observation is, that only people from Europe seem to use the Skype software - when looking into the answers given by subjects claiming to use Skype.

Answer	number of participants
Next time available	42
Current location	25
Current activity	14
Importance of activity	14
Urgency of activity	14
Who the person is with	13
Preferred communication media	2

Table 3.2: Relevant information when contacting somebody

Answer	number of participants
Email	49
Cell phone	42
Office phone	39
Home phone	29
Instant messaging	23
Skype	22
VoIP	7
Phone conference	3
SMS	1
Video conference	0

Table 3.3: Frequently used communication media

Managing email

Most common software (see figure 3.1) to manage and read emails are Outlook/Outlook Express (34%) and Mozilla Thunderbird/Netscape (31%).

Interesting also seems the number of different programs used for emailing. About 72% use one tool, 22% use two tools and three participants currently even use three or more tools.

Managing contacts

Programs to manage contacts are, not surprisingly, similar to email communication (see figure 3.2). Mostly the email clients are used, but also hardware-devices as cell phones and PDAs, and the traditional paper/mind combination still exists.

Instant messaging tools

Most common tools were MSN and Skype. Multi-protocol applications that can manage more than one instant messaging protocol (e.g. Trillian) were also very common. Details in figure 3.3.

Roughly 76% of the users are using only one tool, 20% two tools and 4% three or more applications.

VoIP Audio communication

Nearly two-thirds of the people use Skype for VoIP audio communication. Figure 3.4 shows the distribution of the answers. Only 25 people replied to this question. It can be assumed, that the others do not use VoIP.

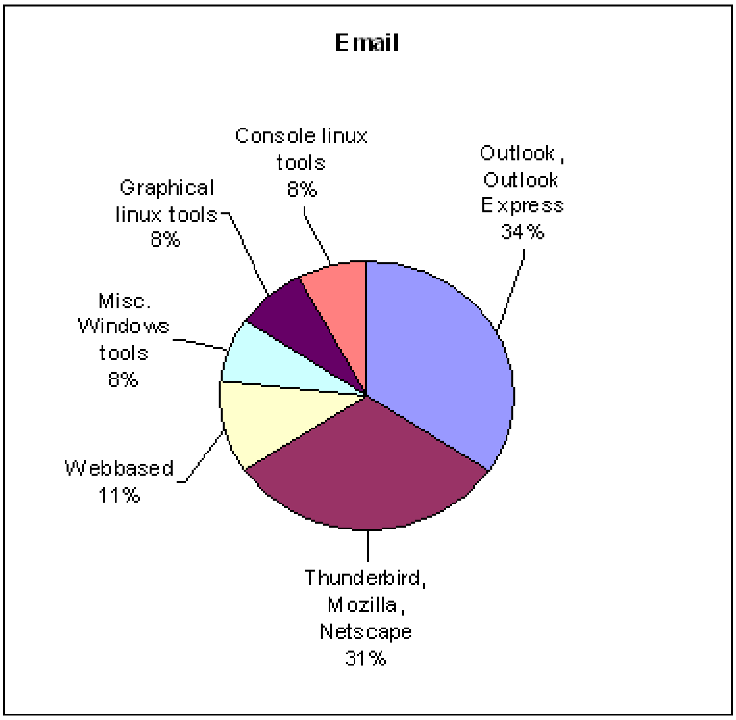

Figure 3.1: Used programs to manage email communication

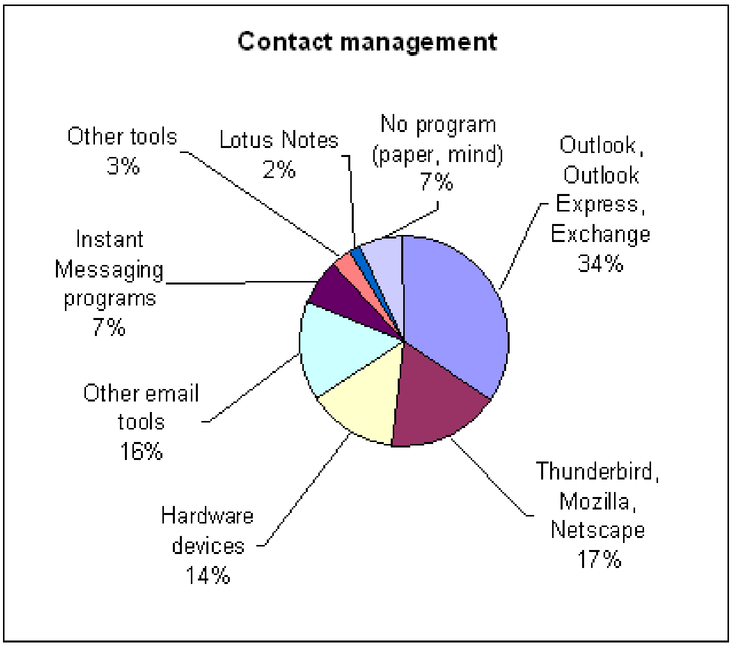

Figure 3.2: Used programs to manage contacts

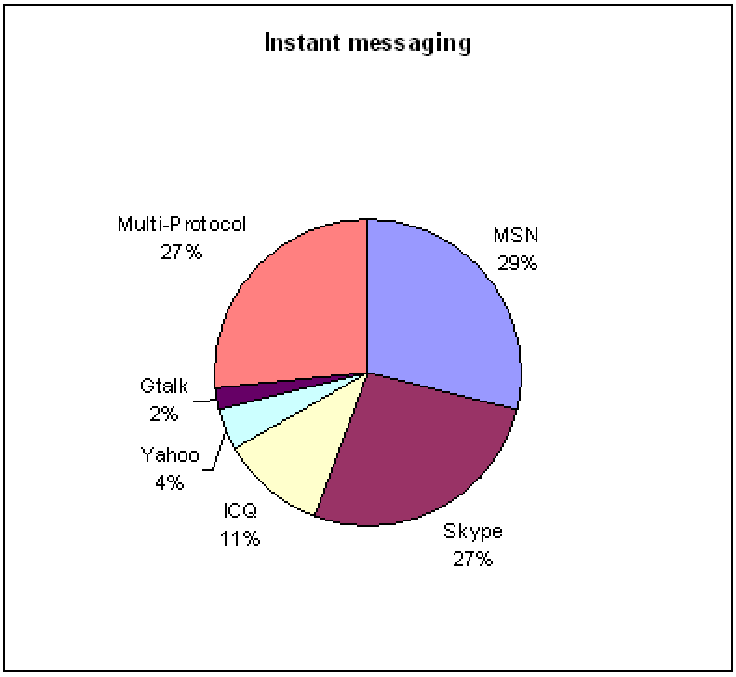

Figure 3.3: Used instant messaging applications

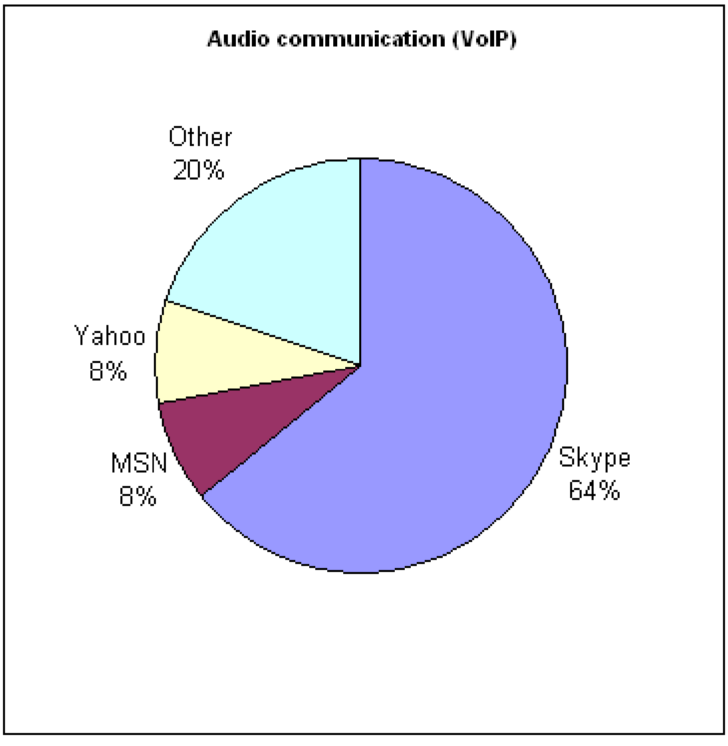

Figure 3.4: Used VoIP programs

Answer	count
Tell others how to reach you best	30
Tell you how available your contact persons are	30
Block incoming communication requests when busy or not available	30
Allow you to contact others as they wish to be contacted	30
Tell others how available you are	27
Give you the opportunity to manage your privacy settings	27
Automatically learn your communication preferences	20
Tell you where your contact persons are	19
Allow others to see your current location	9
Detailed settings what information is visible to whom	2
Distinguish between private and office	1
Time-limited notification	1

Table 3.4: Desired functionality

Would a communication management tool be useful?

Answers to the question *Would you use a tool that helps you manage your communication?* are as following: 21 people replied with yes, 9 with no and 19 with maybe.

When answering *maybe* the user had to specify the condition (multiple responses possible). 6 times the functional benefit was mentioned as the deciding factor, 5 times the ease of use, twice user-defined conditions, twice the interoperability to other communication applications, and twice that such a tool needed to support multiple operating systems.

The following answers have been mentioned only once: handling of privacy, not to expensive, not to large, only for business, it should synchronize programs and devices, full control to connections for the user (no automatic connections), the program should be extensible.

Desired functionality

Answers to the question *What should such a tool do for you? (Assume, that it will respect your privacy)* was a pre-defined list that the users could select, but it was also possible to add additional functionality. The results can be found in table 3.4.

About 61% of the persons would like to have functionality that allows them to visualize their availability and the availability of their contacts. Even the automatic blocking of incoming communication requests would be interesting for 61%. Privacy settings are also very important (55%). Users seem to be skeptic about the automatic learning of their communication preferences, only 40.8% could imagine this functionality.

Locations information was biased: 38.8% would wish to see the location of their contacts - but only 18.4% would allow others to see their current location.

3.2 MyConnector in the overall Connector architecture

The Connector was introduced as service developed at the Universität Karlsruhe within the scope of the CHIL project. MyConnector was designed as an extension of the already existing Connector architecture. **The Connector** service has been designed as a client/server application. User interact with different clients that are available to them.

The **telephony client** - in the left part of the figure - acts as a proxy for incoming phone calls, the **Smart phone client** is an application running on smart phones that provides the Connector functionality through a graphical user interface.

The final architecture can be seen in figure 3.5.

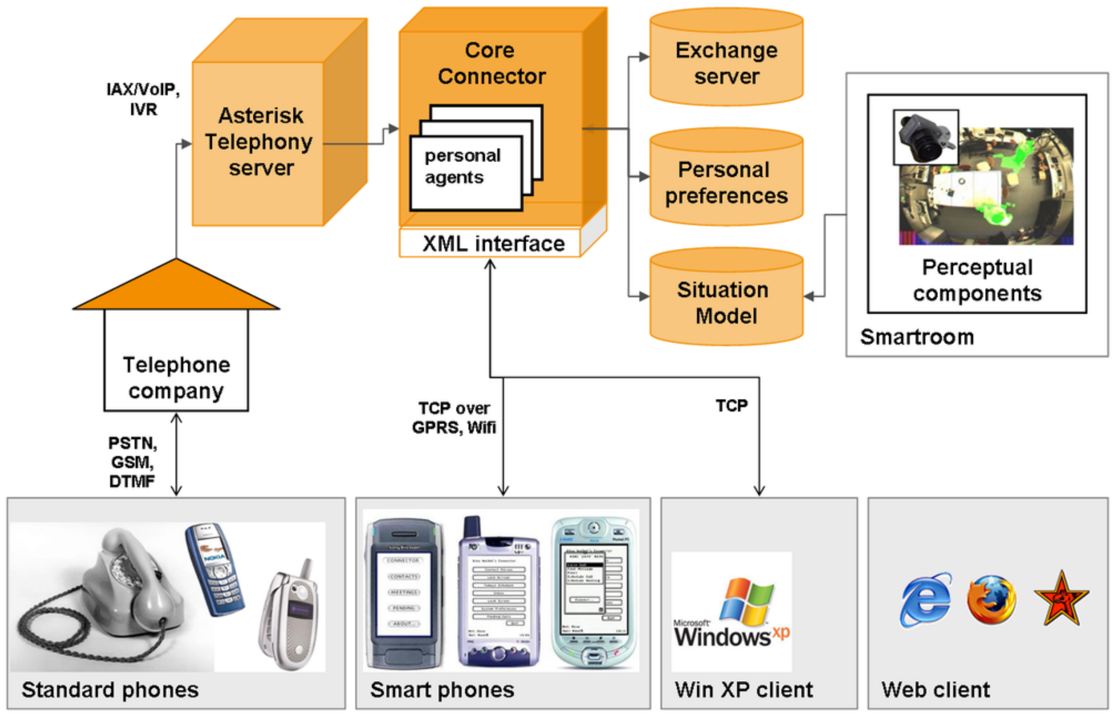

Figure 3.5: Architecture of the Connector service

Within this theses, **two more Connector clients** that bring the Connector to PCs, were implemented: a client running on Windows XP and a web client. The **Web interface** (Section 3.6) shows a user's profile and allows potential contact persons to see the user's availability and current location. An extended functionality is provided by the **WinXP client** (Section 3.5). It can be used to create and manage groups and contacts, initialize communication with available contacts and groups, modify personal settings and preferences - and can collect data, such as PC activity, used by the learning module.
The **core Connector server module** was extended to predict a person's availability and handle the new functionality.

The remainder of this Chapter will describe the details of the implementation and extension of MyConnector in the Connector framework.

3.3 Additions to the Connector server

The server code of the Connector application was refactored and now consists of the following modules:

- **Core connector**, contains common data structures and the logic of the Connector service
- **XML interface**, contains the XML communication layer that is used by the WinXP and smartphone client for exchange of the collected data and availability information
- **Learning module**, contains the code that deals with learning of the users availability

The **core connector** module contains common functions, algorithms and data structures. It connects to the deployed MySQL database and manages the data transfer between the Java code and the database. The implementation has been done with special attention to an easy extension with functionality and data structures.
A special feature: For new locations added to the database, the corresponding latitude and longitude (to display it on a map) will be automatically requested from a commercial web service and applied to the database. All this is embedded in the Java code and performed in the background.

3.3.1 The learning module

The **learning module** extends the core Connector with machine learning methods. WEKA ([WiFr05]), a well-known machine learning toolkit for Java, is integrated. This module can predict the current availability of users but also provides offline functionality to test different classifiers and attributes for their performance. It periodically writes the availability information for each MyConnector user into a shared database table (this is done by a java class that is executed by a unix cron script). The entries in this table are queried by other modules to get the person's current availability. This design is easily extensible. Other algorithms or programs predicting availability only have to add data to this one table - and the MyConnector system will automatically use this availability information.
The *offline system* allows to test the performance of different attribute combination and classifiers. It accesses the database directly and computes additional attributes, as e.g. the history of PC activity data, on the fly.

Learning of availability and powerful context cues as input for the classifier is discussed in more detail in Chapter 4.

3.3.2 The XML interface

The **XML interface** connects clients with the Connector server via a socket connection. The XML-based Connector protocol, which has been developed for prior prototypes - has been extended and enriched with the new data types and functionality introduced by MyConnector. It is being used by the WinXP and the smartphone clients.
The protocol contains functionality to receive and manage information about groups, contacts, locations and the defined rules. Also the collected sensor information and user input is sent via this protocol.

3.4 Communication rules

An extension to the availability model (see section 4.2.2) are **communication rules**. They allow MyConnector to **allow or block communication requests** depending on the person calling, the current time and the location.

Communication rules can be applied hierarchically on the **per user level** or depending on the **social relationship** (friend, coworker, ...). The conditions are the **time (work time/free time)** when this rule fires and the current **location** (home, office, etc.) of the receiver. They also allow to specify the **communication media** on what this rule should fire.

These rules can be specified as a **positive rules** - allowing communication - or **negative rules** blocking the specified communication requests, in which case a user-defined message will be returned to the sender.

For example all communication requests from the boss should reach the receiver during work time - independent of the current availability or location. A communication request from a colleague could be blocked when the user is at home or during his free time.

A user can create, change and delete communication rules. Figure 3.6 shows the edit window of a communication rule.

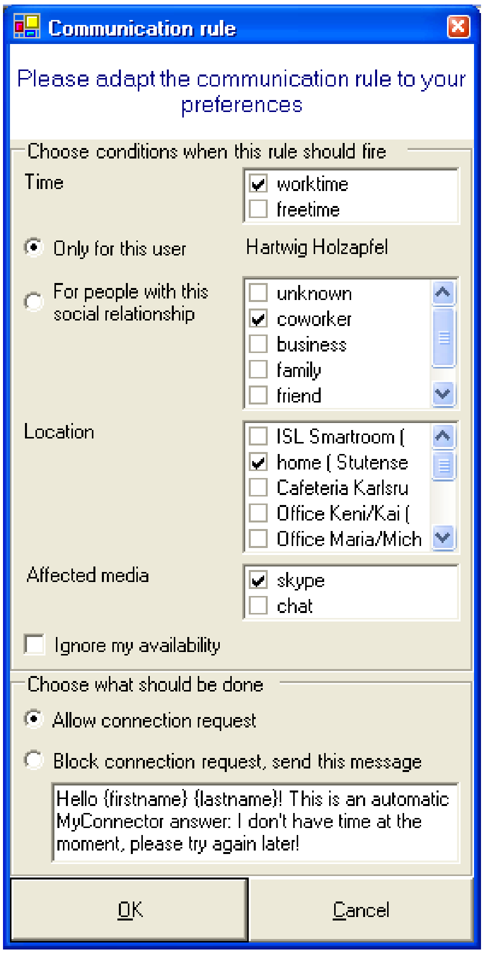

Figure 3.6: Edit window for a communication rule

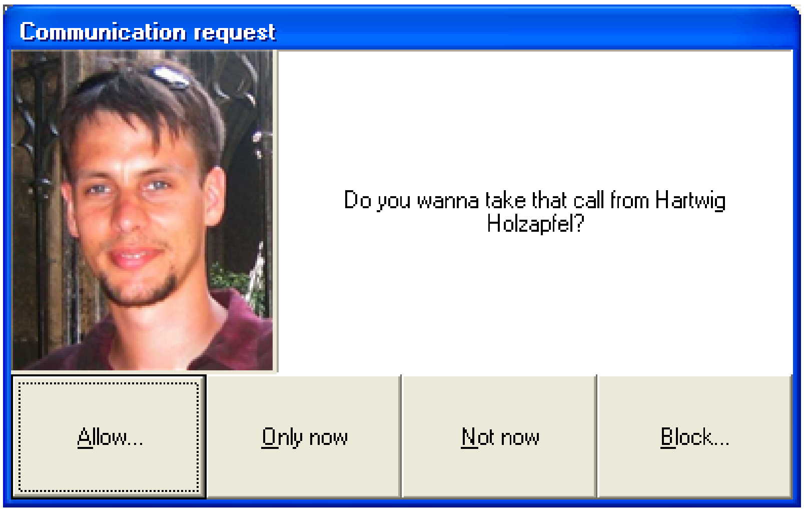

Figure 3.7: Incoming communication request

These communication rules are currently implemented for Skype calls and messages, in case the WinXP client is running. MyConnector can **allow or block Skype conversations**. If no matching rule exists a pop-up window as shown in Figure 3.7 will be displayed where the user can decide what to do with the call: take it or reject it - only for now or set up a communication rule to act permanently.

3.5 Connector client - Win XP

The Win XP client is a Windows application written in C# and runs under the .NET framework.

Figure 3.8 shows a screenshot of the user interface. It displays the current location of the user, the availability for different communication media and the global availability level (1 = not available at all, 4 = full available for communication). The Win XP client makes it easy to **manage groups and users in the address book**, and administrate a list of locations. Clicking on **My public profile** opens a browser and displays the person's public profile - as others see the availability on the web interface; the link **My calendar** opens a browser window of the personal Exchange account. **Update availability** opens a pop up window where a user can update the current availability and location.

For each contact, the available communication channels are displayed and can be executed directly. The user can click on a buddy and select an action that should be performed.

- Call the user - on the home, office or cell phone
- Send an email
- Start a Skype call, instant messaging session or video call

When selecting more than one buddy or a group, there are options for conference communication, where only the available users will be added:

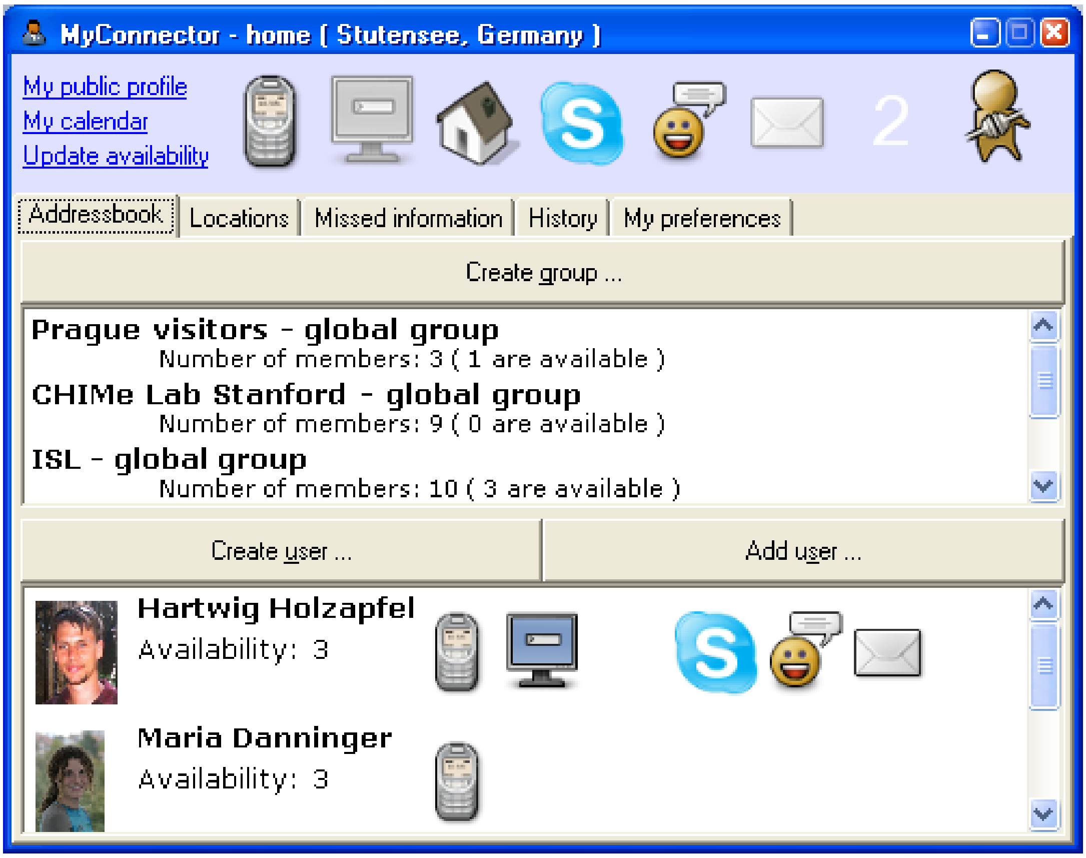

Figure 3.8: Screenshot of the Win XP client

- Email communication
- Skype conference call
- Skype group instant messaging
- Skype video conference call.

Additional to these functional benefits, this client offers the possibility to **collect context cues** that can be used to automatically learn the availability of the user: it can collect PC activity data like keyboard and mouse events, the current active application and Skype activity.

Selected features are discussed in more detail in the following paragraphs.

3.5.1 Managing contact information

The WinXP client allows to **add, modify and remove users** and **arrange them in groups**. Personal created users and groups are not visible to other users. The system also supports **global groups** that are system-wide available, but only visible for the participants of the groups.

Figure 3.9 shows the user dialog where information about contacts can be inserted. There is also a field where a URL to the photo of the contact can be inserted. This will be automatically downloaded and displayed in the address book and shown as soon as an incoming connection request of that person occurs.

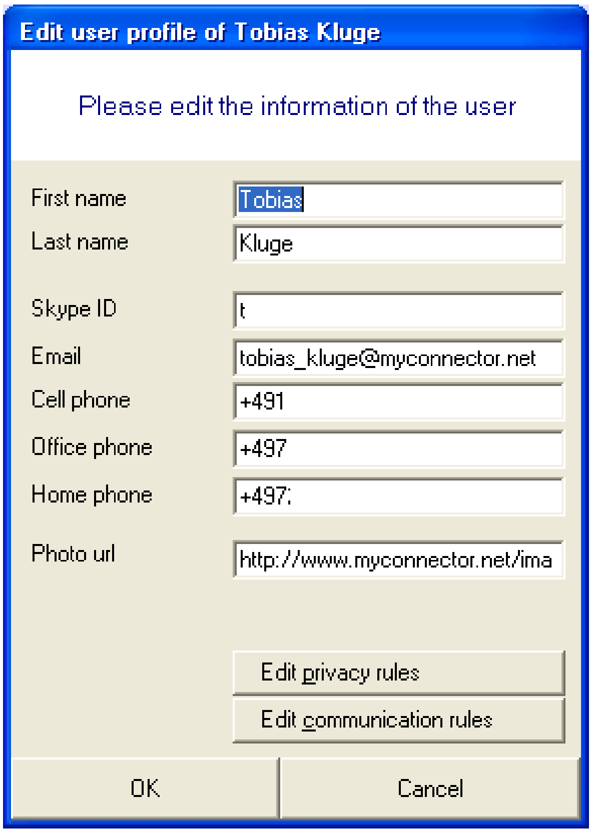

Figure 3.9: Edit user - screenshot

3.5.2 Accessing Skype

The WinXP client connects on startup to the Skype program - if installed on the local computer - and registers for Skype calls and instant messages. Every time when such an event occurs a function of the client is invoked - which e.g. counts the number of messages sent in a chat conversation or the length of a Skype call.

Skype conversations can be initiated directly from the MyConnector GUI; the full set of communication channels provided by Skype is integrated and can be used.

3.5.3 Self-reported context information

The WinXP client provides the functionality to query the user for the current availability and a set of additional context information. This is done via a pop-up window, shown in Figure 3.10. The user has to fill in the information that describe the current situation and activity. After clicking on the *OK* button this information is stored internally and send from time to time to the server.

This feature was used in the user studies described in Chapter 4.

3.5.4 Automatically collecting PC activity

As already introduced in Section 3.2 the WinXP client can automatically collect information about PC activity. To gather this information the application registers for Windows Hooks and uses functions of the Win32 API. More information about the collected data will be explained in Section 4.2.1.1.

Figure 3.10: Screenshot of the pop up window collecting self-reported context information

3.5.5 Internal implementation details

To store settings of the user (e.g. login information) the client writes some information into the **Windows registry** of the local PC. It also serializes the settings and information that have already been collected but not yet sent to the server into a file on the file system.

The WinXP client **runs very robustly** and does not lose information, even if the network connection is intermittent or the system set into standby mode. When no connection to the server can be established, the system buffers all internal communication messages (e.g. collected sensor information, newly created users, etc.). When the connection is restored, it sends this information to the server, where it is stored in the database.

3.6 Connector client - Web interface

The Web interface is very useful to get information about potential contact person's from foreign PCs, e.g. in internet cafes, or if one does not want to or cannot (Linux) install the WinXP client. The MyConnector web interface runs on a special domain - **MyConnector.net** - which hosts the user profiles.

A screenshot of the Web interface can be seen in Figure 3.11. It displays the public profile of a user containing a picture, the name, the availability level, and the physical access to different communication media and as well as the person's current location, as address and on a map.

The **level of detailedness** of information being displayed depends on the owner's privacy settings. Each MyConnector user has a public profile for unknown users.

This profile can also be accessed with mobile devices - Figure 3.12 shows the interface running on an HP iPAQ smartphone with Windows Mobile and the Opera web browser.

For each communication media the visitor can see if the media is available or not. If it is available the user can click on that icon (for email, Skype call and Skype instant messaging) and **instantly a connection is made, from any PC**. When using **email** a new web page is being shown that allows the user to write an email message - without the need of an email client. The actual email address or other contact information of the receiver is never shown!

The **Skype connection** requires a working Skype installation on the visitors PC. A special URL is used to invoke the Skype program directly and create a connection. It would even be possible to use Skype to call normal phone numbers - but in respect to privacy this method has not been implemented - because this special URL contains the Skype id of the callee or the target phone number in plain text.

The **Google Maps** service has been embedded to display the current location of the user. The map is shown only if the privacy rule allows this detailed privacy level for the location! This is especially interesting when it comes to the group web page.

Figure 3.13 shows the **profile of a group**. Functionality has been added that zooms the map automatically to the best matching area. In this example three persons are

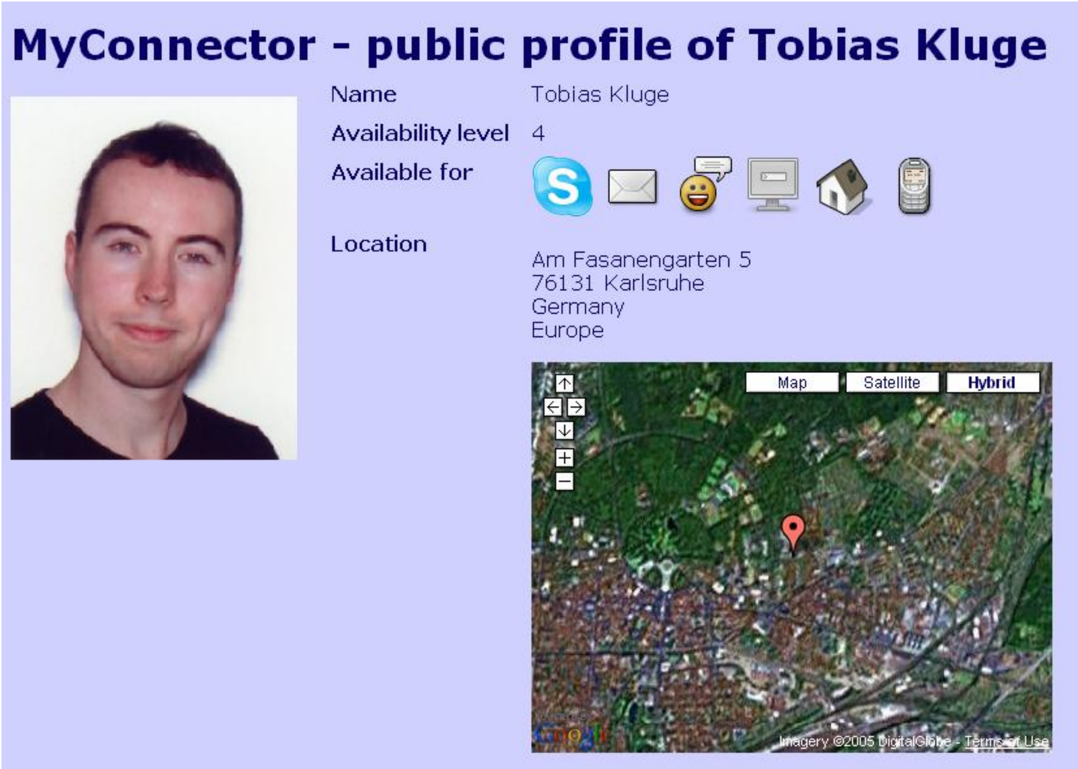

Figure 3.11: Screenshot of web interface

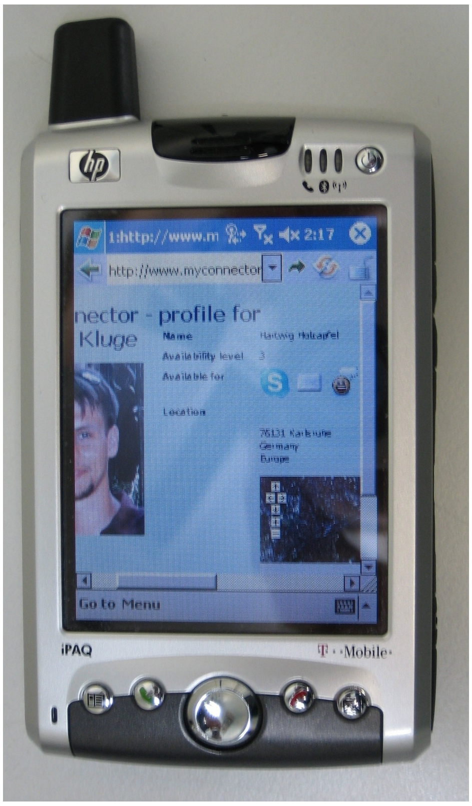

Figure 3.12: Web interface on PDA

in the group - with some users located in North America and others in Europe. It is possible to click on the user to get the availability profile with more details - and directly get in touch with him. Group communication has not been added - but is technically possible.

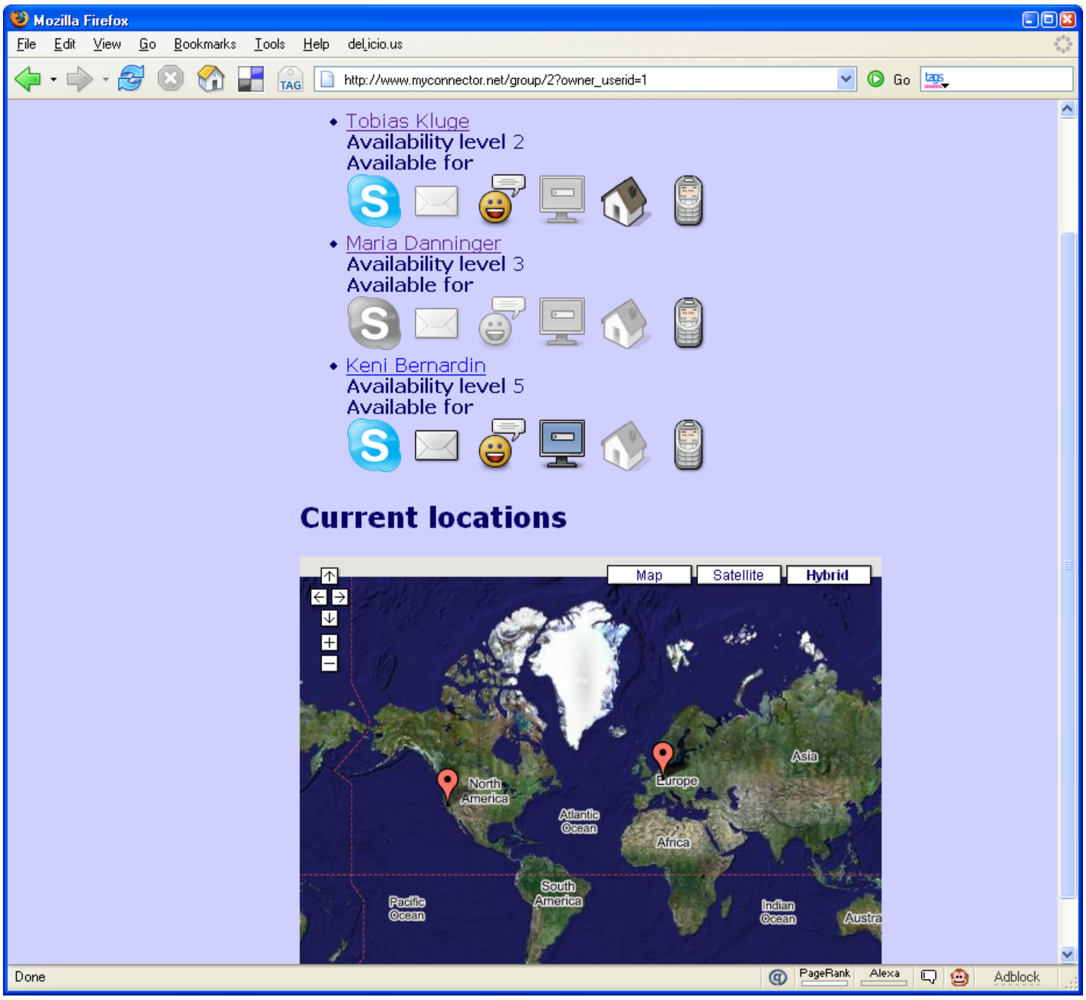

Figure 3.13: Screenshot of a group's profile.

4. Learning Availability in Mobile Contexts

MyConnector uses machine-learning techniques to model contextual knowledge about the user and to infer the user's availability for communication. Input comes from automatically detected context cues, as well as direct user feedback collected in the MyConnector WinXP application. This Chapter describes the algorithms used, and experimental results from studies designed to detect valuable context cues and their predictive power for gaging the availability.

4.1 An introduction to machine learning algorithms

Machine learning describes technologies and algorithms that make it possible for computers to act intelligent and learn as humans would do. Part of this thesis is the learning of the user's availability with machine learning algorithms. The remainder of this Section will introduce two machine learning algorithms used in this work: Bayesian Networks and Instance based classifiers.

4.1.1 Bayesian Network

The Bayesian Network classifier, also called Bayes net, is a robust machine learning algorithm based on statistical information. It is a directed acyclic graph where nodes represent attributes (and their values) and arcs represent the dependency between the attributes. Nodes can present any kind of data. The distribution of the data within a node is described with probabilities, also the transitions and the initial states have probabilities.

Figure 4.1, taken from [Char91], shows a Bayes network that consists of five attributes and probabilities for transitions and initial states.

The learning of Bayes nets performs in two stages: first the network structure and afterwards the probability distributions are learned. Different algorithms are available that help identifying the optimal network structure like hill climbing, simulated annealing and tabu search and estimate the probabilities.

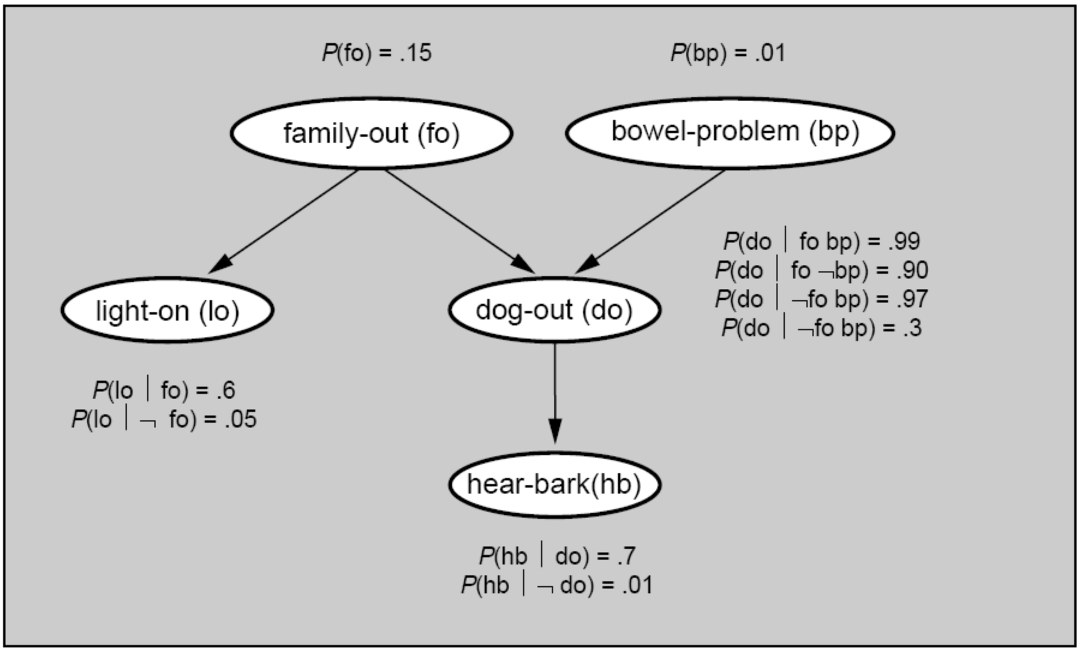

Figure 4.1: Structure of a sample Bayesian Network, taken from [Char91]

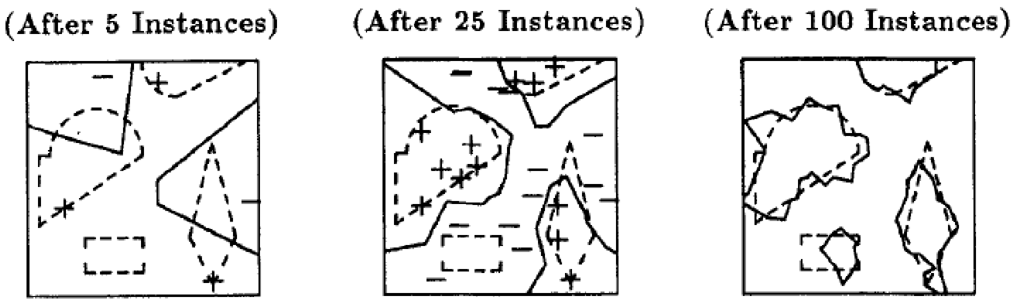

Figure 4.2: Learning of IB algorithm, taken from [AhKi91]

4.1.2 Instance based classifier

The Instance based classifier is a supervised learning algorithm that works like the k-nearest neighbor algorithm - but also normalizes the attributes' range, processes instances incrementally and tolerates missing values. The training data is stored in a multidimensional feature space with a dimension for each attribute.
To classify a data entry the distance to each training data entry is computed and the main class of the k nearest data entries gets returned. The computation of the distance increases with the number of training data.

Figure 4.2 shows how training data is learned over time. The dashed line represents the real distribution of the data, the solid lines the current approximation, + and - indicate positive and negative training instances.
The first image shows the results after learning five instances: the approximated areas - with a solid line - are improper and even partially wrong. After 25 learned instances the approximated area is more precise; after 100 learned instances the solid lines cover precise the real distribution (with the dashed lines).

Aha describes in [AhKi91] the main components that are used by instance based classifiers:

- *Similarity function*: This computes the *similarity* between a training instance i and the instance in the concept description
- *Classification function*: This receives the similarity function's result and the classification performance records of the instances in the concept description. It yields a classification for i.

4.2 Learning availability in MyConnector

This section describes the context cues used as input for the learning classifiers. A detailed description of the designed availability model follows.

4.2.1 Collected context cues

Availability for communication in mobile contexts depends on a variety of context cues: time, location, the social situation, persons around, the person calling, the current activity (it's urgency, mental and physical engagement), which media the request is coming in, if it is physically accessible after all - and probably as well the mood a person is in ... and many more.

Ideally, all context cues used to learn the availability are collected automatically and do not require direct feed-back from the user. This was impossible within the scope of this work. Input in the MyConnector system comes from a set of automatically detected context cues, as well as direct user feedback collected in the MyConnector WinXP application (as described in Section 3.5.3).

4.2.1.1 Automatically detected context cues

The following information have been collected automatically with sensors.

PC activity

The PC activity can be collected within the WinXP application. Collected and stored every minute were the following information

- the number of mouse clicks
- the number of keyboard events
- the number of switches of the active application
- the active program name

Additionally the host name, the local and remote IP address of the user's PC have been saved. The IP address is useful when the PC is used in different networks - e.g. a laptop that is used at home and at work. The local IP address that has been reported by Windows XP and the remote IP that has been reported by the MyConnector server has been stored for each data point. This is useful because some user might work behind a proxy or a local network - which cannot be distinguished easily. Also the local host name of the computer has been stored - which is useful when somebody is using multiple computers.

Figure 4.3: Active areas in an office

Skype activity

Skype activities are also tracked: Skype is running, a Skype instant messaging conversation is ongoing, a Skype call is ongoing. For each Skype communication the following information have been stored: the begin and end of the conversation, the number of participants and - for instant messaging conversations - the number of messages that have been sent and received.

Activity in offices

Within the scope of another thesis at the Universität Karlsruhe explored by Christian Wojek, offices were equipped with cameras and microphones to detect activities inside the office with audio-visual features.

Figure 4.3 shows the active areas in an office, automatically computed by Wojek's system. Depending on visual activities in these areas as well as audio and other visual features the system can detect different activities as e.g. sitting on the desk and working on the computer, or having a meeting.
Office activities were classified as: *discussion* (two person are talking together), *meeting* (more that two persons talking) and *paperwork* (person working on the desk without acoustic activity).

Calendar information

Personal calendar information were collected via an Microsoft Exchange server. This setup provides users with an easy to use web interface (see figure 4.4) to manage their appointments, and the WebDAV interface allows to programmatically access this information.

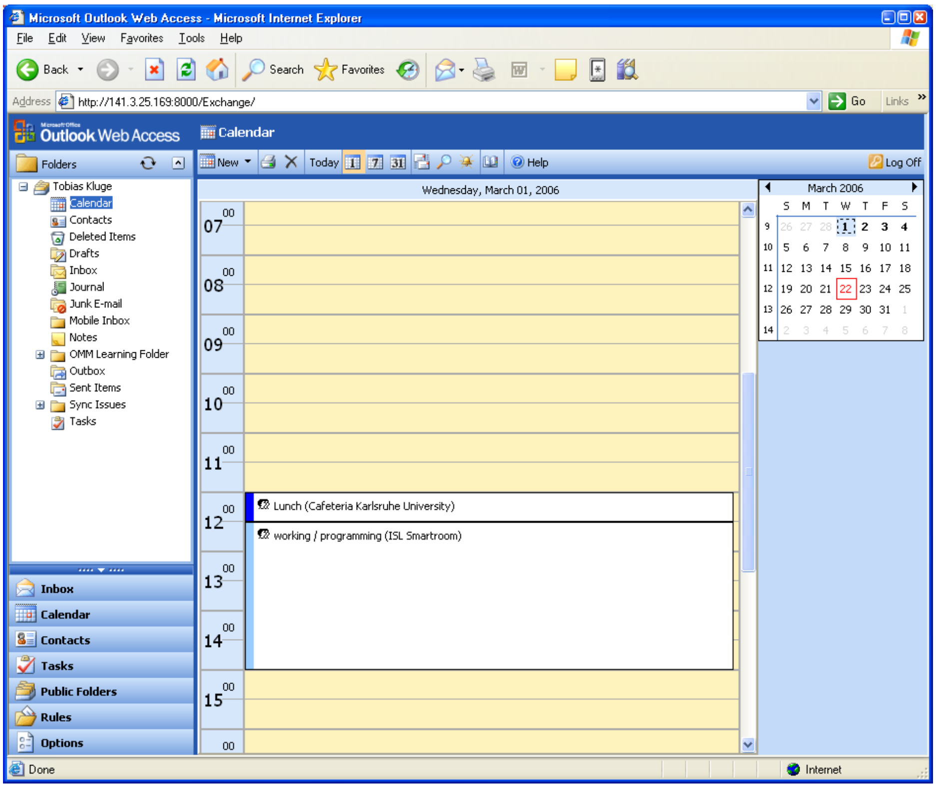

Figure 4.4: Screenshot of Microsoft Exchange web interface

Time information

Of course - time information for each data point has been collected.

4.2.1.2 Self-reported context cues

Additional to the automatically collected sensor input, feedback about more subjective factors of the current situation and activity can be manually entered by the user. Some of the data categories were taken from a large-scale mobile phone availability study conducted within the CHIL project by Interactive Systems Labs and CHIMe lab (Stanford University).

The following factors could be actively reported within the MyConnector WinXP application:

Current location

The current location can be chosen from a list of global and user-defined locations.

Social acceptability

This factor indicates *how socially acceptable it would be to take a phone call in the current situation*. Possible values are between *not acceptable at all* (value: 1) and *fully acceptable* (value: 5).

Collocation and interaction with others

People can indicate their collocation and interaction with others. Possible answers were *alone, dyad, small group (3 to 7 people), and big group (more than 7 people).* For example when working on a PC in an office shared with two co-workers, the collocation would be a small group but interaction would be alone. Vice versa - when doing a skype call while being alone, the collocation would be alone and the interaction would be dyad.

Activity category

Resulting from the former study with CHIMe lab, a list of activity categories has been developed.

- basic needs - e.g. eating, sleeping
- household needs - e.g. cooking, cleaning
- intellectual needs - e.g. doing homework, doing research, programming
- transportation needs - e.g. driving a car, doing public transportation
- communication needs - e.g. calling, writing emails
- interpersonal needs - e.g. socializing with friends
- personal needs - e.g. reading, TV

Mental and physical engagement in the activity

This values indicate the *mental and physical engagement of the user while doing the current activity.* Possible values are high, medium and low.

Importance and urgency of the activity

The importance and the urgency of the current activity are probably important when learning the availability. Possible values are high, medium and low.

Point in lifespan

This value specifies the point in the lifespan of the current activity, which can be beginning, middle or end. Probably people are more interruptible towards the beginning of an activity.

4.2.2 Availability model

To create a predictive model when using machine learning techniques, also referred to as supervised learning, it is necessary to collect data along with labels that represent ground truth about the data. In the scope of this work, this is the person's availability. To be able to learn availability, the user of the MyConnector system have to - at least for an initial training phase - specify their availability for communication.

It is obvious that the availability of a person can hardly be described with one value. The designed and used availability model uses an overall availability level that indicates how available a person is for all kinds of communication at that point in time. But since the access to different communication devices and media varies over the time - and location - additional flags for each communication media indicating the user's availability for this media (nor not-availability) have been used.

Figure 4.5: Graphical representation of the user's availability

General availability level

The *general availability level* is a numeric value from 1 to 4 that indicates the availability of a user:

1. not available at all (e.g. sleeping, swimming)
2. basically not available, but exceptions possible (e.g. meeting, driving a car)
3. busy but can be disturbed (e.g. internet browsing, preparing slides)
4. free, communication encouraged (e.g. doing public transportation, waiting for an appointment)

This level is independent of the physical access to any communication media or device.

Availability for communication media

Additionally to the user's overall availability level, a differentiation for communication medias has been designed. It indicates the physical access to a communication media that includes

- Cell phone
- Office phone
- Home phone
- Skype call
- Skype instant messaging
- Email

For each of the communication media the physical access (yes/no) is learned. For example in public transportation, one has probably physical access to a cell phone (and maybe email when using a Blackberry), but has no access to office and home phones; also Skype and instant messaging are unlikely to be available.

Figure 4.5 shows the graphical representation of the availability used in the MyConnector system.

4.2.3 Learning algorithms

Context information as input for the learning classifier are stored into a common database.

An **iterative approach** was chosen for **training and testing**. The data was sorted by the time it has been collected. Then, for each data item k, the classifier has been trained on the data of items *1* to *k-1* - and then the item k has been predicted. The match or non-match of the result has been counted and the item k has been added to the training set of the classifier.
This iterative approach has been chosen because it simulates the behavior of a real system: the user would add a first data point, the system would behave based on that - and with each additional data point predicts the availability potentially better and better.

WEKA (see [WiFr05]) - a well-known machine learning framework written in Java - is used to analyze the data, and train and test classifiers. Different classifiers were tested, but only the classifiers that performed best have been implemented in the final system (Bayes net, instance - based classifier). Learning algorithms looked at include **C4.5 decision trees, Neural networks, Bayes net classifier and the instance based classifier**. The finally used classifiers (Bayes net and instance based) have been chosen because of the computation costs (which spoke against Neural networks) and their performance (which was not good for the decision trees). There have been some extensions made to the source code of WEKA which have been reported to the developer at the University of Waikato and partially applied to the WEKA CVS - and will be part of the next release.

The following paragraphs describe the details of the implemented classifiers.

For the **Bayes net** the following algorithms have been used: the *K2 hill climbing algorithm* to learn the structure of the network and the *SimpleEstimator* to estimate the probabilities once the structure has been learned which is done directly from the provided data. These algorithms are already implemented in the WEKA learning framework.

The **Instance-based** classifier used the *normalized Euclidean distance* to measure the difference between data points in the multinomial feature space.

Experiments have been made that **compare the performance of discretizised and non-discretizised data**, which resulted in a better performance of the non-discretizised data. For the discretization of the data the built-in Discretize-class of WEKA has been used, it uses binning to discretize the data and automatically finds the number of bins.

The data used by the classifiers has been extracted directly from the database used by the MyConnector system, computed features as e.g. the history of PC activity have been added afterwards on the fly. This data has been used to train and test the classifiers.

The office activity data provided by Christian Wojek has been imported into the database and than applied to the collected data. This is necessary because his system currently is not able to deliver this data in real time.

userId	number of answers
1	341
2	73
5	57
23	38
24	23
25	59
26	57
27	36
28	63

Table 4.1: Number of questionnaires per user (big study)

The *final online system* is based on a Bayes net classifier and uses all collected context information.

This learning system was successfully deployed and tested in a longer-term user study. Within these experiments with real users, training data could be collected and used to train the classifiers.

4.3 Experimental results

Not all of the context cues used in the MyConnector system may be relevant to predict a person's availability. Experiments were conducted with the MyConnector system in order to detect which context cues, or combination of context cues, promise a strong predictive power in gaging the availability. The independent variable in these experiments was the general availability level. In two experiments a total of 1106 data points of nine users have been collected.

4.3.1 Experiment design

Two experiments in form of user studies were performed by the author. Participants were researchers located in Karlsruhe and Stanford. Each of them installed the MyConnector WinXP client application on their computer (mostly laptop computers). Some participants in Karlsruhe shared offices that were equipped with cameras and microphones. Participants were asked to report their availability at work and also in their free time. Only four of the participants provided data in their free time though.

The **first study** had nine participants and a duration of five working days. Overall 689 data points were collected. Table 4.1 shows the number of answers that have been given by each user. Except user 1 no participant provided availability information for the time at night (when being not available because of sleeping). Since a person is very unlikely to be available between 1:30 to 4:30 am a data point indicating no availability has been added for each of the users at the time of 1:30, 2:30, 3:30, 4:30 am.

In a **second study** the MyConnector system has been improved. This time the focus was to get fewer dedicated users to insert more data points throughout the whole day. The questionnaire categories have been shortened, calendar data has been activated and used - and the internal learning algorithm has been tuned.

userId	number of data points
1	188
2	175
23	54

Table 4.2: Number of questionnaires per user (final study)

attributes	IB1	BayesNet
hour	66.639%	68.023%
hour, weekday	76.332%	71.044%
hour_full	67.342%	67.915%
hour_full, weekday	71.150%	69.773%

Table 4.3: Results for time data

Three participants (1,2 and 23) of the first study participated. They collected 417 data points in three days (day and night), the distribution of data points per user is visualized in table 4.2.

4.3.2 Baseline system

The **baseline system** was trained on hour, weekday and location, since these information were available for every data point and present a minimal but potential powerful set of availability information. Other research groups successfully used this feature combination (Fogarty, Horvitz).

The presented performance is a weighted sum of the performance for each user, with the fraction of data points as weight. Table 4.6 shows an example of the computation of the overall performance.
Only the results for the instance based classifier (IB1) and the Bayes net are shown.

4.3.3 The impact of *time* on availability

Time information seems to be a fairly good indicator to predict a person's availability. The availability level could be predicted with an maximal accuracy of around 76%. This performance is so good because the availability of the subjects did not change a lot during work time, and many people did not specify their availability during free time. Also - the activities of these persons were researching most of the time in their office with only a few other activities as having lunch or talking with a co-worker - which made them most of the time available. There were no scheduled meetings or other activities where they have not been available.

Analysis

Table 4.3 shows the result for the different time features that have been implemented and tested. The combination of hour (that is the hour with minutes as fraction) and weekday (the day of the week) performed best - with an accuracy of about 76.6% (IB1 - instance based classifier returning the class of only the nearest data point) / 71.0% (BayesNet). The feature *hour_full* contains only the hour, minutes have been left out.

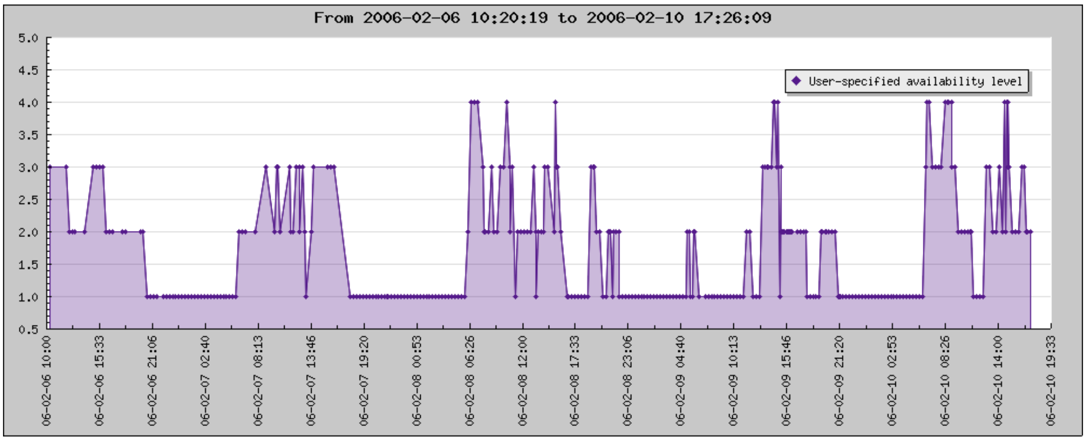

Figure 4.6: Availability profile of user 1

The **availability of a subject over five days** is visualized in figure 4.6. It displays typical daily artifacts. The x-axis shows the time, the y-axis displays the availability (1 indicates no availability, 4 indicates a high availability). The longer segments with only low availability are typical during night - with some exceptional additions on Tuesday(06-02-07) and Thursday(06-02-09).

Table 4.4 shows the availability over the course of the day, independent of the user and the day of week. The first number of the presented availability is the mean value, the second the standard deviation and in the brackets the number of data points. For example at 11 a.m. the mean availability level of all users is 2.403 with a standard deviation of 0.861, an overall of 72 data points have been reported at this time.

The core time when participants were available was between 8 and 18 o'clock for work days. The availability slows a bit down at lunch time between 11 and 14 o'clock. The deviation in the morning between 7 and 10 is high - this is because of the different activities when getting up, having breakfast, going to work and finally beginning to work - which have a highly varying availability.

Discussion

This is a promising result since users typically have daily or weekly activities that occur more or less at the same time - and this can be easily learned by the system. These results are similar to the ones made by Hudson in [HCKE02], described in Section 2.1. His focus was on availability for business times with less data points.

4.3.4 The impact of *location* on availability

Location is a very good predictor, especially in combination with time information. Results show a significant difference between user groups with heavily changing daily schedules to others with regular office hours (much stronger relation).

Analysis

Adding the location as an additional attribute to the time information, the performance increases by 2.8% (IB1) / 3.8% (Bayes net) to 79.13%/74.87%, table 4.5 shows the absolute performance.

hour	availability
1	1.555 ± 0.831 [27]
2	1.179 ± 0.383 [28]
3	1.039 ± 0.192 [26]
4	1.000 ± 0.000 [24]
5	1.000 ± 0.000 [24]
6	1.267 ± 0.442 [30]
7	2.133 ± 1.231 [30]
8	2.515 ± 1.076 [33]
9	2.205 ± 1.202 [39]
10	2.603 ± 1.016 [58]
11	2.403 ± 0.861 [72]
12	2.427 ± 0.856 [82]
13	2.531 ± 0.917 [81]
14	2.389 ± 0.840 [77]
15	2.772 ± 0.941 [79]
16	2.652 ± 0.984 [89]
17	2.564 ± 0.961 [62]
18	2.380 ± 0.998 [50]
19	2.039 ± 1.009 [51]
20	2.300 ± 0.964 [50]
21	2.206 ± 0.867 [34]
22	1.762 ± 0.717 [42]
23	1.818 ± 0.999 [33]
24	1.433 ± 0.667 [30]

Table 4.4: Availability in the course of the day

attributes	IB1	BayesNet
hour, weekday	76.332%	71.044%
hour, weekday, location	79.135%	74.868%

Table 4.5: Results for location data

Table 4.6 shows the drill-down to the results for the combination of hour, weekday and location on the user level. The overall performance is a weighted sum of the performance for each user, the weight is directly related to the number of data points provided by the user. So for example the user 25, who provided only a few data points but the classifier performed very well (97.3%) was weighted only with 0.086 - but user 1 with a lower performance of 65.3% was weighted with 0.391 because of the huge number of data points.

classifier	user								
	1	2	5	23	24	25	26	27	28
IB1	75.0%	61.9%	98.6%	83.0%	68.4%	97.3%	88.9%	80.4%	73.1%
BayesNet	65.3%	59.5%	100.0%	81.1%	65.8%	97.3%	86.1%	80.4%	74.4%
Weight	0.391	0.098	0.084	0.062	0.045	0.086	0.084	0.060	0.091

Table 4.6: Drill down to results of location data on user level

The location information is based on a list of fixed locations, provided and managed by the user. Some default locations as e.g. the offices and common rooms in Karlsruhe have been added to the list beforehand.

The results show that for common used spaces like the Karlsruhe smartroom the mean availability level is (reported by five users) 2.682. The mean availability level in the offices varies from 1.85 to 3.13, depending on the subject. But at home, the mean availability level differs a lot more: from 1.000 / 1.646 up to 3.000 and even 3.556. This might be due to the fact that very few users added availability at night - which might have lowered the mean availability but increased the deviation.

Discussion

Since the location is chosen from a list of known locations, it was sometimes hard for the users to specify other activities as e.g. doing transportation or infrequent activities as e.g. having a doctor's appointment. Some of the users added locations as e.g. *other location* or *transit*.

Apart from these drawbacks, the location seems be to a strong indicator for a person's availability - especially combined with the time features.

4.3.5 The impact of *soft factors* on availability

Soft factors - provided by the user - improve the performance of both classifiers! The performance compared to the baseline system increases to roughly 79.5% / 79.1%. The most promising parameter combination was activity category, social acceptability, importance/urgency and physical/mental engagement. Point in lifespan of the activity did not show an impact.

A detailed discussion follows.

activity category	availability
basic needs	1.723 ± 0.994 [249]
communication needs	2.612 ± 0.841 [98]
household needs	3.667 ± 0.471 [3]
intellectual needs	2.494 ± 0.914 [324]
interpersonal needs	1.615 ± 0.964 [26]
personal needs	2.500 ± 1.500 [4]
transportation needs	2.300 ± 0.971 [30]

Table 4.7: Analysis of data for activity categories

Urgency	Importance		
	low	medium	high
low	1.82653 ± 0.892 [98]	2.429 ± 0.821 [28]	2.000 ± 1.000 [6]
medium	2.44594 ± 0.681 [74]	2.579 ± 0.932 [285]	2.125 ± 0.927 [8]
high	2.56250 ± 0.609 [16]	2.333 ± 1.054 [9]	1.767 ± 1.103 [210]

Table 4.8: Analysis of data for urgency and importance of the current activity

Activity categories

Table 4.7 shows the relation between the activity category and availability. Users are normally highly available when they are doing *household needs*, but less when doing *basic needs*. *Communication needs* normally indicate a good availability, *intellectual needs* are also an indicator for availability - but with a higher deviation. *Interpersonal needs* seem to be related to a low availability, while *transportation needs* indicate a higher availability with a deviation of about one availability level.

Urgency and importance of the current activity

Table 4.8 shows the relation of the urgency and importance of the current availability to the user's availability level. The results displayed in this table indicate, that the availability goes down when the importance of the activity is high *or* the urgency of the activity is high. As expected, the availability is lowest, if both values are high and an activity thus is important *and* urgent.

Mental and physical engagement in the current activity

Table 4.9 shows the relation between the mental and physical engagement in the current activity and the person's availability. The first column with low physical engagement contains only a few values that are representative.
A trend shows that higher mental engagement results in a lower availability level; the same for an increasing physical engagement - the availability level goes down (except for the low mental engagement).

Environment

The environment attribute indicates how socially acceptable it would be to take a call or communication request in the current environment. The values ranged from *not acceptable at all* to *totally acceptable*.

Table 4.10 shows the relationship between the environment and a person's availability. It is easy to see that the more acceptable it is to communicate in an environment, the higher the availability is.

mental	physical		
	low	medium	high
low	1.000 ± 0.000 [2]	1.351 ± 0.666 [37]	2.443 ± 0.800 [122]
medium	1.500 ± 0.500 [2]	2.454 ± 1.100 [108]	2.392 ± 0.830 [204]
high	4.000 ± 0.000 [4]	2.061 ± 1.058 [49]	1.951 ± 1.135 [206]

Table 4.9: Analysis of data for mental and physical engagement

social acceptability	availability
1 - not acceptable at all	1.012 ± 0.111 [80]
2	1.621 ± 0.764 [132]
3 - tolerated	2.435 ± 0.988 [345]
4	2.714 ± 0.628 [42]
5 - full acceptable	2.785 ± 0.864 [135]

Table 4.10: Analysis of data for environment

Point in lifespan

The results of the relation between point in lifespan and availability (see table 4.11) indicate that point in lifespan does not seem to predict a person's availability very well.

All soft factors combined

Soft factors - provided by the user - improve the performance of both classifiers - as shown in table 4.12. The performance of BayesNet increases nearly to the same level as the instance based classifier - 79.486% (instance based) vs. 79.146% (BayesNet).

The **best parameter combination** was hour, weekday, location, activity category, social acceptability, importance/urgency and physical/mental engagement.

4.3.6 The impact of *PC activity* on availability

Surprisingly, the PC activity did not improve the performance of the classifier.

Analysis

This section analyzes the relation of PC activity data (keyboard (see table 4.13), mouse (see table 4.14) and window activity (see table 4.15)) to a person's availability. Data points without PC activity have been left out for this analysis.

Figure 4.7 shows an example how PC activity data changed while the availability level stayed the same over a period of about 100 minutes. The number of keyboard, mouse and window changes vary a lot.

point in lifespan	availability
begin	2.028 ± 1.013 [72]
middle	2.252 ± 1.017 [591]
end	2.085 ± 1.031 [71]

Table 4.11: Analysis of data for point in lifespan

attributes	IB1	BayesNet
hour, weekday, location	79.135%	74.868%
. . . , activity category	78.208%	75.442%
. . . , social acceptability	78.324%	74.516%
. . . , importance, urgency	79.016%	78.562%
. . . , physical/mental engagement	79.020%	74.743%
. . . , point in lifespan	76.360%	74.517%
. . . , all	79.141%	79.030%
. . . , all except: point in lifespan	79.486%	79.146%
. . . , all except: point in lifespan, social acceptability	79.367%	78.793%

Table 4.12: Results for soft factors

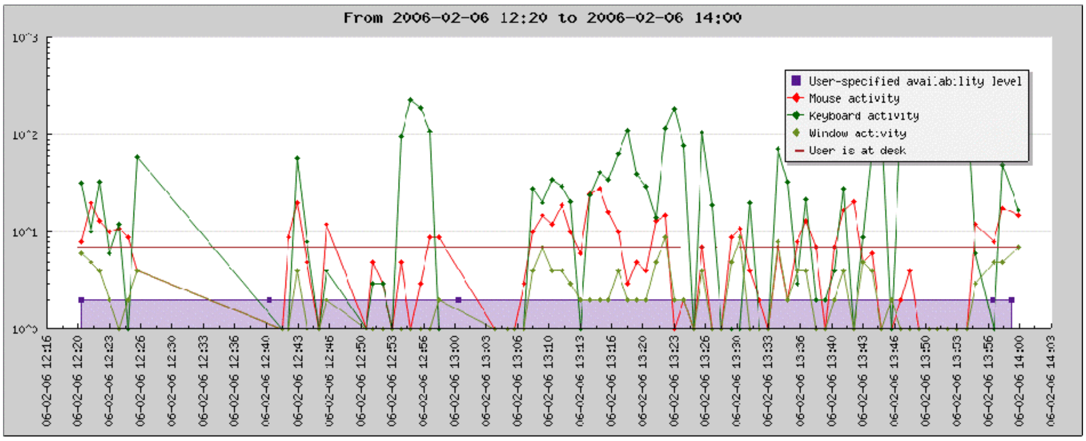

Figure 4.7: PC activity vs. availability level

Table 4.16 shows the results for classifiers trained on different attribute combinations.

Nearly all performance values are lower than the performance of the attributes *hour, weekday, location*, only the BayesNet classifier performed better with the name of the active application, but only by absolute 0.68%.

Discussion

High PC activity does not seem to be related to lower availability - e.g. when browsing in the Web or writing emails. But also lower PC activity does not seem to be a sign of higher or lower availability: the user could be thinking about a

number of key pressed	availability
0	2.607 ± 0.935 [211]
1-5	2.400 ± 0.894 [25]
6-10	2.619 ± 0.999 [21]
11-20	2.263 ± 1.068 [19]
21-40	2.286 ± 0.796 [42]
41-100	2.222 ± 1.052 [45]
101-200	2.341 ± 0.900 [41]
>200	2.267 ± 1.181 [15]

Table 4.13: Analysis of data for keyboard activity

number of mouse clicks	availability
0	2.535 ± 0.921 [170]
1-5	2.365 ± 1.008 [74]
6-10	2.451 ± 1.018 [71]
11-20	2.418 ± 0.933 [67]
21-40	2.444 ± 0.956 [36]
>40	4.000 ± 0.000 [1]

Table 4.14: Analysis of data for mouse activity

number of active window switches	availability
0	2.567 ± 0.924 [194]
1-3	2.287 ± 0.991 [108]
4-8	2.524 ± 0.954 [103]
9-12	2.000 ± 1.128 [11]
>12	2.333 ± 0.471 [3]

Table 4.15: Analysis of data for window switching activity

problem, even be out of office, or calling somebody - all of which has very different availabilities.

4.3.7 The impact of ***Skype activity*** on availability

It looks like the users are more available when they are actually engaged in a Skype communication. But unfortunately, the number of the data points is not very significant.

Analysis

Five out of the nine users were regularly engaged in Skype communications during the experiments.

The following three collected binary attributes have been combined into a single value.

- skype running
- skype chat ongoing
- skype call ongoing

The combined skype activity value has been computed as following

$$skypeActivity = skypeRunning*2^0+skypeChatOngoing*2^1+skypeCallOngoing*2^2$$

Comparing Skype activity to a person's availability (see table 4.17) it looks like the users are more available when they are actually engaged in Skype communication. But unfortunately, the number of the data points - especially for Skype calls - is not significant.

Using Skype activity as additional attribute decreases the performance of both classifiers (table 4.18).

attributes	IB1	BayesNet
hour, weekday, location	79.135%	74.868%
..., keyboard, mouse, window	63.659%	73.586%
..., skype	76.362%	74.636%
..., userActivity,	72.778%	74.752%
..., userActivity, skype	70.932%	74.745%
..., userActivityHistory 10min	71.631%	73.942%
..., userActivityHistory 5min	71.744%	74.056%
..., userActivityHistory 5min, skype	69.211%	73.822%
..., activeProgramName	76.928%	75.550%
..., activeProgramName, windowActivity	70.811%	74.971%

Table 4.16: Results for PC activity data

Skype activity	Availability
Skype running	2.484 ± 0.97 [473]
Skype chat	2.690 ± 0.89 [84]
Skype call	3.000 ± 0.00 [1]
Skype chat and call	3.000 ± 0.00 [3]

Table 4.17: Analysis of data for Skype activity

Discussion

The number of data points with Skype activity is to low to have a potential positive impact on the performance of the classifiers. More data is necessary to get more valuable results.

4.3.8 The impact of *office activity* on availability

The collected office activity does not seem to help when predicting a persons activity.

Analysis

Table 4.19 shows the relation between the activity in people's offices to their availability. Six of the nine participants were in offices equipped with cameras and microphones. The distribution of these data over the whole data set is not optimal, because not all collected data had been transcribed in time - which tends to a lower performance of the classifier.

While doing paperwork people tent to be less available than when discussing with other person (availability level of 2.577 vs. 2.929/3.000). Partner in discussions and meetings are most of the time co-workers. Some of them shared rooms; all subjects have been working on the same floor - so an easy exchange was possible.

Office activity does not seem to help when predicting a persons activity. Table 4.20 shows that adding the information whether a user is in his office to time and location information does not improve the results of the classifier. The same holds true for the current situation in the room. The performance even goes down, when combining the information whether a person is in his office and the situation in the office.

attributes	IB1	BayesNet
hour, weekday, location	79.135%	74.868%
..., skype activity	76.362%	74.635%

Table 4.18: Results for Skype activity

Activity	Availability
Discussion	3.000 ± 0.632 [20]
Meeting	2.929 ± 0.257 [14]
Paperwork	2.577 ± 0.754 [111]

Table 4.19: Analysis of data for office activity

Discussion

Access to office data from all users might result in a higher performance. Other groups used similar information and had good results (Horvitz, Fogarty).

4.3.9 The impact of *collocation and interaction* on availability

Collocation and interaction with other person help to predict the availability.

Analysis

The data containing collocation and interaction has been collected only in the 2nd study. The performance therefore differs from the results previously presented.

Table 4.21 shows the coherences between the collocation and interaction with others and the person's availability. When following the values from *Alone* to *Big group* the availability level goes down from 2.66 to 1.33.
The availability data for an interaction of *Alone* and a collocation of *Alone* and *Dyad* is vice versa - this is because the data for a collocation and interaction of *Alone* contains also e.g. the data when being at home which might behave differently.

Collocation and interaction with others help to predict the availability. Using collocation and interaction with others as additional parameter, the system's performance indeed increased. The instance based classifiers performs about 4.8% better, the BayesNet by 1.7%.

Figure 4.8 shows the increase of the availability over time. The x-axis shows the data points that have been submitted by the user over time, the y-axis shows the performance of the classifier. Instance-based classifier perform in this figure better than the BayesNet classifier.

Discussion

The results are in line with the results of [HCKE02]. Hudson writes that the positive attitude toward interruption (which are close to an availability level of 3 and 4 in this work) goes down when interacting with more people. Hudson distinguishes between unplanned and planned engagements which has not been done in this work.

Note: The baseline system - trained on hour, weekday and location - performs significantly lower than for the other factors, since this data was only collected in the second study.

attributes	IB1	BayesNet
hour, weekday, location	79.135%	74.868%
..., in office	77.621%	74.750%
..., room situation	77.622%	74.280%
..., in office, room situation	77.390%	73.940%

Table 4.20: Results for office activity

	Interaction			
collocation	Alone	Dyad	Small group	Big group
Alone	2.666 ± 1.053 [183]	2.666 ± 0.943 [9]		
Dyad	2.965 ± 0.668 [29]	2.211 ± 0.449 [57]		
Small group	2.312 ± 0.916 [16]	2.000 ± 0.000 [3]	1.522 ± 0.499 [23]	
Big group	2.000 ± 0.000 [1]	2.000 ± 0.000 [1]		1.333 ± 0.471 [9]

Table 4.21: Analysis of data for collocation and interaction with others

4.3.10 The impact of *calendar data* on availability

Personal calender information does not seem to help predicting the availability.

Analysis

A relation between data retrieved from personal calendars and the availability is shown in table 4.23. Overall, one can see that the availability level goes down when there is an appointment scheduled (2.512 vs. 2.196). Taking a closer look at the single subjects (Table 4.24) it is easy to see that the existence of an appointment is not a good indicator for a lower availability for all of the persons. This is related to the different personalities of the persons.

For each data point the existence of an appointment has been added and used to train the classifiers. Table 4.25 shows that the performance does not increase, the performance of the instance based classifier even decreases.

Discussion

Similar results have been made by Fogarty in [FoLC04] - he writes that

> Calendar information is generally not considered very reliable, because not all calendar items imply unavailability and because many people have calendar items that they do not actually attend.

Fogarty has not published results for the study that compares the availability to the calendar free/busy information.

4.3.11 Summary of results

The best performance has been made when using time, location, social factors and collocation/interaction. PC activity, office activity and calendar information did not increase the performance, but this might be related to the inhomogeneous distribution of the data available for each category.

attributes	IB1	BayesNet
hour, weekday, location	65.092%	58.056%
..., collocation	68.337%	58.616%
..., interaction	69.010%	59.847%
..., collocation, interaction	69.904%	59.734%

Table 4.22: Results for collocation and interaction with others

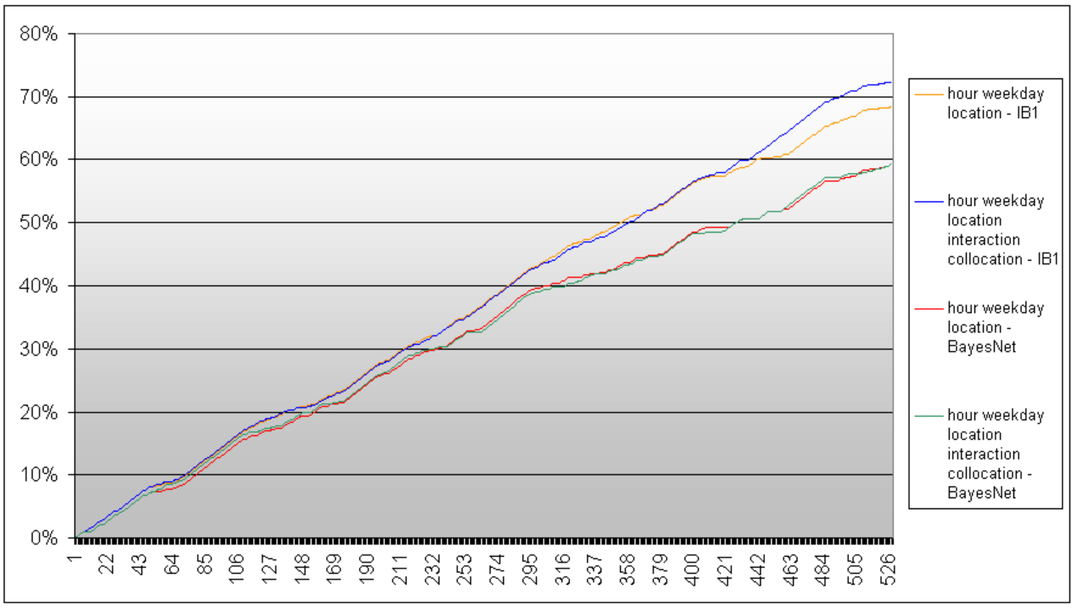

Figure 4.8: Performance of classifier over time

Unfortunately social factors are only available for the first study, and collocation/interaction only for the second study - so it is not possible to train classifiers on a combination of these promising attributes.

Problematic is the distribution of data points per user - most of the users just submitted too less data to train a good classifier.

4.3.12 User feedback

Participants of the first study have been interviewed afterwards. They mostly liked the following functionalities:

- Broadcasting of receiver's availability and the interface of the application
- the learning feature
- the opportunity to manage privacy settings.

Calendar	Availability
free	2.512 ± 0.965 [285]
busy	2.196 ± 0.924 [46]

Table 4.23: Analysis of calendar information data

	Availability		
calendar	user 1	user 2	user 23
free	2.504 ± 0.868 [127]	2.241 ± 1.026 [108]	3.120 ± 0.765 [50]
busy	2.688 ± 0.682 [16]	1.964 ± 0.944 [28]	1.500 ± 0.500 [2]

Table 4.24: Analysis of calendar information data - drill down to user level

attributes	IB1	BayesNet
hour, weekday, location	65.092%	58.056%
. . . , calendar	64.867%	58.056%

Table 4.25: Results for calendar information

Remarks have been given to improve the system. One direction was to shorten the list of self-reported factors, since this would make it possible to answer, even if busy. Another idea was to display the time when the contact persons will most likely be available again.

Two subjects pointed out that they were using two PCs - one running Windows where the MyConnector client was running and another machine with Linux where they worked on. The collected PC information are not accurate in these cases.

5. Privacy Issues in this Transparent World

Whenever personal data such as location or availability is broadcasted, privacy immediately becomes an extremely important issue to the user. This becomes obvious, as most people do not want their detailed location being shown in a Google map on the web. MyConnector provides the opportunity to specify *who* should be able to see *what* information *when*. E.g. *I want all my colleagues to see the building I am in, but only during working hours, but my family can always see where I am. The default should specifying only what continent I am on (as opposed to what country, city, street,building, or room).*

Default privacy settings were determined in the introductory survey. The design of privacy rules within the MyConnector system is described afterwards.

5.1 Default privacy settings

According to previous research it is necessary to provide appropriate default settings when it comes to privacy related data. In the introductory survey (see Section 3.1) people were asked about their privacy concerns. The should select which details about their location and current activity they would like to be broadcasted to their wife/husband, family, friends, acquaintances, coworkers and their boss, during work time and free time. The time of day seemed to be only relevant for work-related persons (co-workers, boss). As expected, less known persons (such as acquaintances) were less trusted than people in more proximate social circles (such as family and friends).

The users have been asked what level of information detail they would grant to different social relationships. The answers are described in the following paragraphs.

Privacy for location on map

The distribution of the answer of subjects to the question *Imagine you want to allow others to see your current location (on a map). What level of detail would you allow them to see?* can be found in table 5.1.

contact type	Exact position	Street	Town	Country	Continent
wife/husband	16	10	15	1	5
family	14	4	22	2	5
friends	6	4	28	4	5
co-workers	3	2	25	11	6
boss	2	0	27	9	9
acquaintances	0	1	28	10	8

Table 5.1: Privacy for location on map

Data type	possible answers
Location	Exact position, street, town, country, continent only
Availability	all communication media, selected number of communication media, no information
Current activity	Detailed (subject, begin, end), Selected (subject only), Selected(begin, end only), No information
Current participants	Detailed (including name), Number of persons, No information

Table 5.2: Privacy level for data

Answers show that trust in stronger social relationships as wife/husband and family is much higher than to work-related (boss, co-worker) and acquaintances.

Privacy in different situations

The user had to specify what level of detail they would grant to different contact types. The data types have been the location, the availability, the current activity and the current participants - the possible answers can be found in table 5.2. Contact types have been family, wife/husband, friends, circle of acquaintances, co-workers and boss.

These questions have been varied on the time: for free- and work time.

The results have been compiled into diagrams, one for each contact type. Figure 5.1 shows the results compiled into a privacy profile. The categories are displayed on the edges of the axes, the center of the diagram (0-values) indicate that all details of the data will be shown to the requester.

Comparing the privacy profile of a strong social relationship (*wife/husband*, figure 5.1) to them of less strong, more formal social relationship like *boss* (figure 5.2), the differences are interesting: the profile for *wife/husband* is independent of the time and grants a lot of details about the data; but the profile of the *boss* shows that there is a big difference between free and work time. More details would be granted when the subject is at work - partially even more than for *wife/husband* (for the attributes participants and activity). But this totally changes when work is over: the subjects granted nearly no information to the *boss*. Some subjects replied that their boss is also a friend - which might resulted in more data accessible for them.

From these results some default privacy rules have been compiled and integrated into the MyConnector system. Each subject of the user studies got four default rules dependent of the social relationship of the requester. Table 5.3 shows the final privacy rules that have been used.

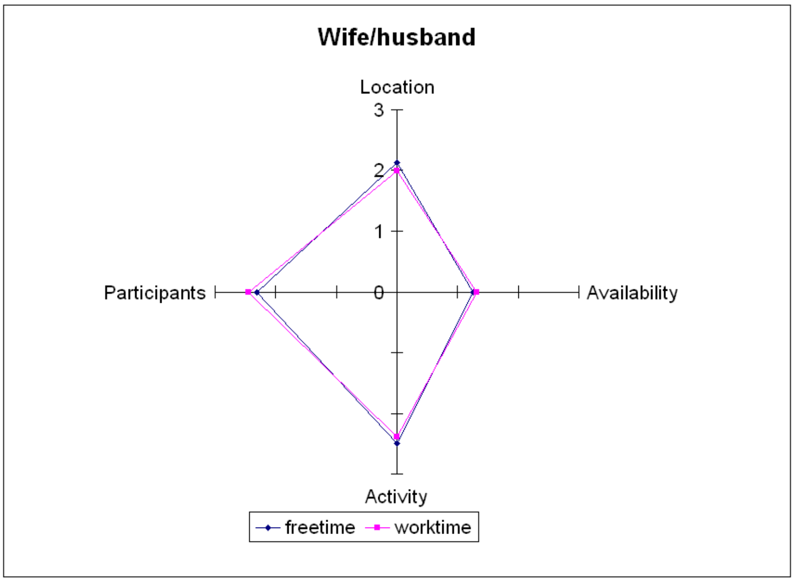

Figure 5.1: Privacy profile for wife/husband

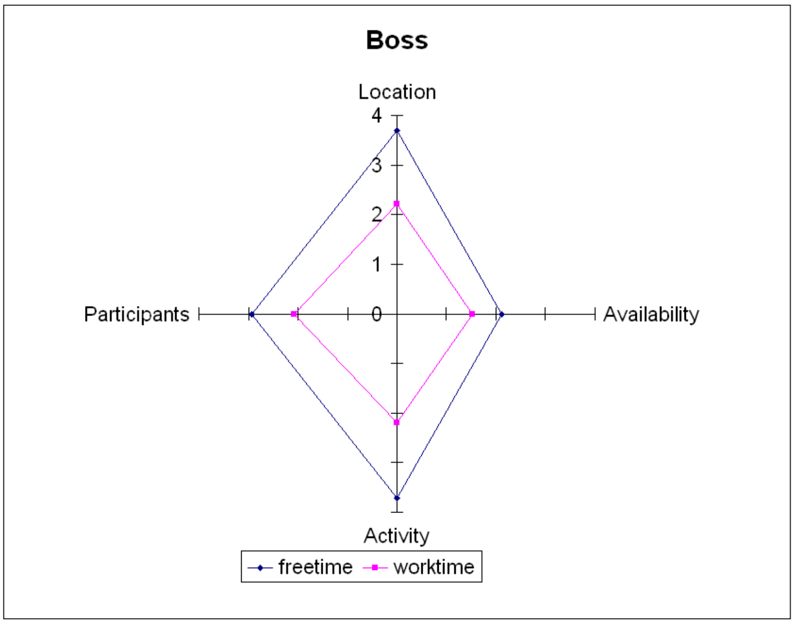

Figure 5.2: Privacy profile for boss

Social relationship	Granted data
unknown, acquaintance	availability level, availability for email
family, wife/husband, friend	all data, independent of time
boss, co-worker	in work time: all data
	in free time: no data

Table 5.3: Compiled privacy rules

5.2 Privacy rules

Hierarchical privacy rules have been implemented to specify what level of information should be visible to whom and when. The condition of the rules are the time (work time/free time), the current location of the receiver and the social relationship between the receiver and sender.

Privacy rules can be applied hierarchical for a user, for a group or in a global context (independent of user or group relationship). The algorithm that selects the privacy rules is the following:

1. Select all privacy rules on user level, check if they can be applied. If yes, do so and return result.
2. Select all privacy rules on group level, check if they can be applied. If yes, do so and return result.
3. Select all matching privacy rules in global context. Apply them and return.

Figure 5.3 shows the window to create and edit privacy rules in the MyConnector WinXP application.

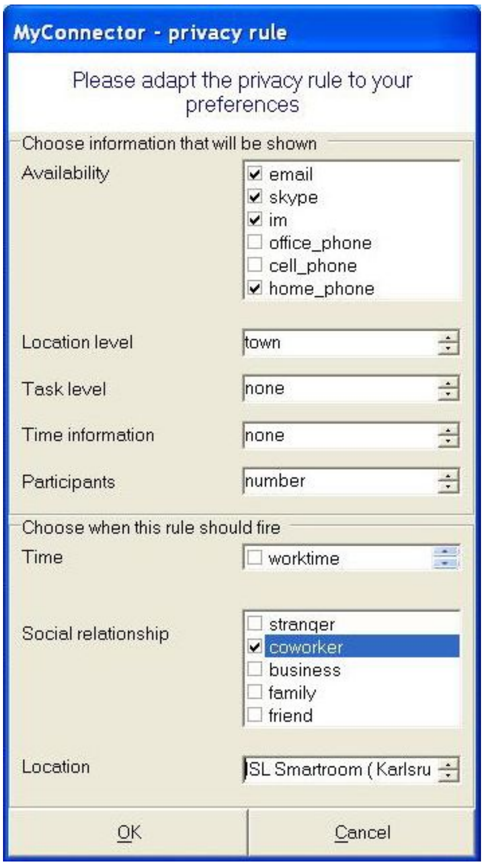

Figure 5.3: Edit window for a privacy rule

6. Conclusion and Future work

6.1 Conclusion

The outcome of this thesis is MyConnector, an extension of the Connector service that adapts to the users and predicts their availability. The Windows XP client offers an interface to set preferences and manage contact persons. Along with phone communication, it lets the user send emails and instant messages, and allows conference and video calls (via the Skype API). PC activity data can be gathered in the background as context cue to learn user availability. The new web interface shows user profiles and current availability. The level of information granularity displayed is user-defined in the owner's privacy settings, and depends on the social relationship, the time and the location. Initial default settings have been computed from the results of an online survey with 49 participants from Europe and America.

User studies were performed with nine subjects over a total time period of eight days. These studies compared the results of different classifiers to the self-reported availability level. Input for two different classifiers (Bayes network and instance-based classifier IB1) were twofold: On the one hand automatically collected context information like audio-visual activity detection in offices, calendar information, automatically gathered PC activity and the time; on the other hand a set of (partially) more subjective measures, manually entered by the users - like the location, physical accessibility for communication media, social acceptability, collocation with others, activity, mental and physical engagement, importance and urgency of the activity and the point in the activity's lifespan.
Analysis of these studies suggest that time and location alone are only good context cues to predict availability for people with regular office hours. Therefore, other factors seem to be even more important for those people who have very flexible schedules. Tracking of PC activity did not significantly improve the predictive power of the classifier, even though the active program showed to be the best indicator. Concerning calendar information it was found that the existence of an appointment is not always a good indicator for a lower availability. The analysis of the 'soft' self-reported measures indicates that these factors contribute significantly to predictions of the person's availability level. E.g. the less socially acceptable receiving a phone

call in the current environment would be, the less available a users is. Also, participants were more available when doing current activities that were judged as not urgent. Looking at people's activities in offices, it has been found that people tent to be less available when doing paperwork than when discussing with other persons.

The best achieved overall performance was about 80% accuracy.

6.2 Outlook

MyConnector will be used in the future development of the Connector service.

Location as a data category improves the performance a lot. The current implementation is based on user-reported location from a list - which is not the best solution because it requires manual updates and thus costs user attention and is not precise. An automatically sensed location from GPS or the next cell phone tower could be used to automatically detect the current location in the background.

Social factors depending on the environment have also shown a big impact on the availability. An integration of a system that can automatically predict the environment category depending on acoustic information would also improve the usability and results of the system.

The sensor data describing the current activity in offices is also worth of future work. This would make it possible to automatically detect the ongoing activity in office - which is a potential valuable indicator for the availability in offices.

To get more significant results and a more thorough understanding of these factors and their predictive power for gaging the availability, a larger sample size would need to be studied. Study participants who have regular meetings, mabye work not only in their office, and communicate a lot with each other - for example in an industrial company like SAP or Siemens - would show if there is a real usage of such a system.

Bibliography

[AdBa04] Piotr D. Adamczyk und Brian P. Bailey. If Not Now, When?: The Effects of Interruption at Different Moments Within Task Execution. In *Computer Human Interaction CHI 2004*, Band 6, Vienna, Austria, 2004. ACM, S. 271–278.

[AhKi91] D. Aha und D. Kibler. Instance-based learning algorithms. *Machine Learning* Band 6, 1991, S. 37–66.

[BaKC00] Brian P. Bailey, Joseph A. Konstan und John V. Carlis. Measuring the effects of interruptions on task performance in the user interface. In *IEEE Conference on Systems, Man, and Cybernetics 2000 (SMC 2000)*. IEEE, 2000, S. 757–762.

[Bouc05] Remco R. Bouckaert. Bayesian Network Classifiers in Weka, 2005.

[Char91] Eugene Charniak. Bayesian Networks Without Tears. *AI Magazine* Band 12, 1991, S. 50–63.

[CHIL] CHIL. Website: http://chil.server.de.

[FKAG+05] James Fogarty, Andrew J. Ko, Htet Htet Aung, Elspeth Golden, Karen P. Tang und Scott E. Hudson. Examining Task Engagement in Sensor-Based Statistical Models of Human Interruptibility. In *Computer Human Interaction CHI 2005*, Portland, Oregon, USA, 2005. ACM.

[Foga04] James Fogarty. AmIBusy: High-Level Abstraction in Ubiquitous Sensing Environments, 2004.

[FoHL04] James Fogarty, Scott E. Hudson und Jennifer Lai. Examining the Robustness of Sensor-Based Statistical Models of Human Interruptibility. In *Computer Human Interaction CHI 2004*, Vienna, Austria, 2004. ACM.

[FoLC04] James Fogarty, Jennifer Lai und Jim Christensen. Presence versus Availability: The Design and Evaluation of a Context-Aware Communication Client. *International Journal of Human-Computer Studies*, 2004.

[HCKE02] James M. Hudson, Jim Christensen, Wendy A. Kellog und Thomas Erickson. I'd Be Overwhelmed, But It's Just One More Thing to Do: Availability and Interruption in Research Management. In *Computer*

Human Interaction CHI 2002, Minneapolis, Minnesota, USA, 2002. ACM.

[HKKJ02] Eric Horvitz, Paul Koch, Carl M. Kadie und Andy Jacobs. Coordinate: Probabilistic Forecasting of Presence and Availability. In *Conference on Computer Supported Cooperative Work 2004*, Edmonton, Alberta, USA, 2002. Morgan Kaufmann, S. 224–233.

[HNLL04] Jason I. Hong, Jennifer D. Ng, Scott Lederer und James A. Landay. Privacy Risk Models for Designing Privacy-Sensitive Ubiquitous Computing Systems. In *Designing Interactive Systems: Processes, Practices, Methods, and Techniques*, Camebridge, Massachusetts, USA, 2004. ACM.

[HoJH99] Eric Horvitz, Andy Jacobs und David Hovel. Attention-Sensitive Alerting. In *Conference on Uncertainty and Artificial Intelligence*, Stockholm, Sweden, 1999. Morgan Kaufmann, S. 305–313.

[HoKA04] Eric Horvitz, Paul Koch und Johnson Apacible. BusyBody: Creating and Fielding Personalized Models of the Cost of Interruption. In *Conference on Computer Supported Cooperative Work 2004*, Chicago, Illinois, USA, 2004. ACM.

[HoLa04] Jason I. Hong und James A. Landay. An Architecture for Privacy-Sensing Ubiquitous Computing. In *MobiSYS 2004*, Boston, Massachusetts, USA, 2004. ACM.

[LeMD03] Scott Lederer, Jennifer Mankoff und Anind K. Dey. Who Wants to Know What When? Privacy Preference Determinants in Ubiquitous Computing. In *Computer Human Interaction CHI 2003*, Ft. Lauderdale, Florida, USA, 2003. ACM.

[Less99] Lawrence Lessig. *Code and Other Laws of Cyberspace.* New York NY: Basic Books. 1999.

[PaKo04] Sameer Patil und Alfred Kobsa. Instant Messaging and Privacy. In *Human Computer Interaction HCI 2004*, Leeds, UK, 2004. S. 85–88.

[SaEN05] Vagner Sacramento, Markus Endler und Fernando Ney Nascimento. A Privacy Service for Context-Aware Mobile Computing. In *IEEE Conference on Security and Privacy for Emerging Areas in Communication Networks*, Athens, Greece, 2005. IEEE, S. 182–193.

[WeBr96] Mark Weiser und John Seely Brown. The Coming Age of Calm Technogy. *PowerGrid Journal*, 1996.

[WiFr05] Ian H. Witten und Eibe Frank. *Data Mining: Practical machine learning tools and techniques.* Morgan Kaufmann, San Francisco. 2nd. Auflage, 2005.