Fachhochschule Gießen - Friedberg

Fachbereich Mathematik, Naturwissenschaften und Informatik

Diplomarbeit

Corporate email and calendaring solutions embedded in a
strategy for Computer Supported Cooperative Work

Zur Erlangung des akademischen Grades
Diplom-Informatiker (FH)

Diplomand
Timotheus Pokorra, Matr.-Nr. 625252

Betreuer
Prof. Dr. Berthold Franzen (FH Gießen - Friedberg)
Dipl. Inf. (BA) Andrea Wachter (Operation Mobilisation)

ERKLÄRUNG ZUR DIPLOMARBEIT

Gemäß § 5 Abs. 8 der Prüfungsordnung des Fachbereichs MNI, der Fachhochschule Gießen – Friedberg

Hiermit versichere ich, die vorliegende Diplomarbeit selbständig verfasst und unter ausschließlicher Verwendung der angegebenen Literatur und Hilfsmittel erstellt zu haben.
Gleichzeitig versichere ich, diese Diplomarbeit in gleicher oder ähnlicher Form weder veröffentlicht noch einer anderen Prüfungsbehörde vorgelegt zu haben.

Siegen, 15.11.2002

Timotheus Pokorra

ACKNOWLEDGMENTS

TABLE OF CONTENTS

FIGURES

TABLES

1 ABOUT THIS DIPLOMA THESIS

This diploma thesis should give an overview of the current situation of the group calendaring market. Group calendaring is a Groupware application, and Groupware is the implementation of the studies of the field "Computer Supported Cooperative Work". Therefore this environment will also be investigated to show that there is not only group calendaring, but even more applications that help groups working together.

This diploma thesis will show some of the standards involved in Groupware, and the fight for survival of proprietary solutions and the creation of the Internet Calendaring standards are investigated.

A closer look to one open source groupware solution will show that still a lot of work has to be done. The reader will realise how difficult it is to build an application with a clean structure and readiness for future extensions when several people are voluntarily working on a project.

1.1 The Environment of this Diploma Thesis

Operation Mobilisation (OM) is a christian mission organisation that has about 40 offices in different countries. These offices are responsible for the activities of OM in their country or neighbouring countries. Especially during the summer months the number of involved people increases due to special outreaches. The whole administration and organisation is done by those offices. The office of the **International Coordination Team** (ICT) in Carlisle provides, beside other help and tools, the OM Standard Linux Server. The Linux team develops and maintains this configured distribution of Redhat Linux. The goal of this standardised server is to have one solution for all offices with the same configuration. That simplifies the support by the Carlisle office and means that not every office has to figure out the best suitable Linux configuration.

One problem is the very different situations of the offices. Some offices are based in poor countries with a minimum of technical infrastructure, e.g. there is only an expensive dialup connection to the Internet. There are 2 ocean-going ships as well, with each 300 staff and 50 workstations on board. Other offices are based in Europe and the United States and can use a really good technical infrastructure. Smaller offices often have different needs than huge offices.

The aim of the computers in OM is to support the people working together, and to give them more time for doing the real job which is described by OM with these words: "[...] bring a message of hope to the peoples of the world. [...] we love Jesus and we want others to have the opportunity to hear about Him. Operation Mobilisation works in more than 80 countries, motivating and equipping people to share God's love with people all over the world. OM seeks to help plant and strengthen churches, especially in areas of the world where Christ is least known." [OM]

1.2 The Task

The task was to collect the requirements concerning an email and calendaring solution with support of Palm synchronisation to provide groupware functionality for the existing OM Standard Linux Server. Furthermore a general survey of existing solutions on the market should be given. Using this survey, the group of solutions that best fit the requirements should be specified, and finally one chosen solution needs to be integrated into the server.

1.3 The Structure of this Diploma Thesis

The first chapter contains this introduction.

The second chapter investigates the term "Computer Supported Cooperative Work". It refers to the possible applications of CSCW, and describes the underlying technologies involved. Also the relation to Multimedia and Pervasive Computing is mentioned. The new term "enterprise application integration" also is remotely connected with CSCW. This chapter describes the ideas behind this diploma thesis.

In the third chapter the main focus is on the functionality that is expected from email, group calendaring and synchronisation. The standards used to realise the ideas are named and described.

The fourth chapter consists of two parts: First there is the description of how the analysis was realised, and in the second part an outline of the current market situation is given. So this chapter contains information about the implementations

of the ideas of chapter 2 by using the standards described in chapter 3.

The fifth chapter covers the specific requirements and solutions for OM, and contains an outlook for future computer support for OM.

A summary of the diploma thesis is in chapter 6.

The appendix holds some tables and installation instructions that were created during the work on the topic.

All information can also be found on the enclosed CD-ROM. The diploma thesis document and the results of the analysis are originally in XML format and were converted by PHP programs to well structured and linked HTML documents.
All texts taken from the Internet are also saved on the CD-ROM, and they can be easily accessed via the links in the HTML version of the diploma thesis.

2 COMPUTER SUPPORTED COOPERATIVE WORK

This chapter describes some of the ideas that build the basis and the background of groupware products.

The terms "Computer Supported Cooperative Work" and "Groupware" are explained during this chapter, but also other terms like "Pervasive Computing" and "Enterprise Application Integration" are mentioned, which influence groupware applications.

The main focus is put on scheduling and on calendaring because this was the field of CSCW that was most required by OM.

2.1 Definition of Computer Supported Cooperative Work and Groupware

In 1988, Greif described **Computer Supported Cooperative Work** (CSCW) in this way: "CSCW is an identifiable research field focused on the role of the computer in group work". [GREIF]

Wilson wrote in 1991: "CSCW is a generic term which combines the understanding of the way people work in groups with the enabling technologies of computer networking, and associated hardware, software, services and techniques." [WILSON]

What is then the relationship between CSCW and groupware? In the words of Borghoff and Schlichter, "CSCW includes the universal scientific research field, groupware deals with the respective practical system solutions of collaborative work" [BORGHOFF, p. 88]. And further on they say: "Groupware refers to software systems supporting teamwork and integrating theoretical foundations achieved by CSCW research" [BORGHOFF, p. 92].

Johansen states that "Groupware can involve software, hardware, services and/or group process support" [JOHANSEN].

Therefore groupware is the name for computer systems whose functionality is based on the results of the CSCW research.

In the following parts of this second chapter often the term "groupware" is mentioned, although the title of the chapter is "CSCW". But that is just the right way: In fact the topic of CSCW is to define the term "groupware" and to fill it with appropriate meaning.

2.2 Groups being supported by groupware

The goal of groupware is to support groups or teams in their work. It is assumed that these groups have a "**common task (or goal)**" [ELLIS et al.] that they try to achieve by working together.

Borghoff and Schlichter write about different types of groups: They call groups that only communicate via computer "electronic groups", in contrast to "electronically supported groups" which have both personal contacts and electronic communication [BORGHOFF, p. 93]. The goal of CSCW is to "[make] distributed communications as efficient as face-to-face communication." [BORGHOFF, p. 94].

The groupware system is supposed to "provide an interface to a **shared environment**" [ELLIS et al.]. That means that the members of the group are accessing the same objects in the environment at the same or different times. That is especially a requirement when these objects have to be changed by different persons until the objects are in the aimed state.

There are different degrees of a shared environment and a common task. That means there are a lot of very different kinds of groupware applications, which are investigated in a further chapter.

Therefore, all groups of people that work towards a common goal and want to use electronic communication to deal with a shared environment, need groupware applications in order to work and communicate fast and efficiently.

2.3 Classification of Groupware

Grudin's extended groupware classification according to time and space is shown in the following table [GRUDIN]. Grudin extended the classic classification by the factor predictability.

Space/time	Same time (synchronous)	Different times (asynchronous)	
		predictable	unpredictable
Same place	face-to-face meeting	shift-work	blackboard
Different places (predictable)	Video conference	email	joint editing of documents
Different places (unpredictable)	mobile phone conference	non-real-time computer conference	workflow management

table 2.1 Groupware Classification

Here is just one example how to read this table: Email is a groupware application

that assumes different predictable places, which are the different mail servers that normally are not in the same place or on the same machine, but their position is predictable because they don't move. If you want to send someone an email, you just send it to his email address which is associated with his email server. The sender and the receiver have access to an specific email at different times: The sender writes it, then he sends it and looses control over the email. He only can read it again if he has a copy of the mail, but cannot change the instance of the email anymore that he sent away. While the time difference between the sending and the receiving is not a guaranteed fixed time, the asynchronous access to the email is predictable: When the mail has left the email server of the sender, he cannot access it anymore, and as soon as the email arrives at the email server of the receiver, he has complete control over the mail. So it is predictable that both would never access the mail at the same time and that the receiver would never be able to read the mail before the sender has sent it.

What about calendaring? It would fit in the grid at "predictable different places" and "unpredictable different times". The argument for the place is the same as with the email, because we are again dealing with a server that manages the calendar. The access to a calendar item is asynchronous because all users can read and write to a calendar independently, according to their rights on that specific calendar. One user can add a new appointment, two hours later another user reads it, and again 30 minutes later the first user cancels it. You cannot predict the chronological order of different people accessing the calendar or one of its items.

2.4 Applications of Groupware

This chapter gives a short description of some application areas of CSCW. It should help answer the question: How can CSCW be used in the daily working life?

2.4.1 Email / Message systems

Email is one of the most used groupware applications, but it is only a loose communication link between the group of people using it: People are not forced to answer to requests, and the information contained in different emails is not easy to be organised. Some emails only contain small parts of a discussion, other messages are holding important decisions or profound drafts. Anyway, email

systems provide a fast and direct communication channel for text messages and other attached media files.

Groups can use email communication by either sending their mails to a **mailing list** which distributes them to all members of the group that have subscribed to the list. An archive of a mailing list provides the history of past discussions and communication, which helps new members to join the group and to catch up the information that they otherwise would miss.

The **usenet** with its **newsgroups** is a different way of communication. Its main goal is to support distributed discussions. So one of the advantages of newsgroups is that all news or message items are saved only once on a server, and the users can download those items they are interested in. This overcomes the heavy traffic caused by mailing lists, where all incoming emails are delivered to all subscribers. And naturally it provides an archive of written news. There is a clear hierarchy of the messages, but this is also provided with simple mailing lists, when a message has reference tags to previous mails in the thread.

The disadvantage of the usenet is that a newsgroup has to be checked regularly if you are expecting answers to your question or comments to your statement, and you need to remember where in the hierarchy your article was placed.

A **forum** is also an application for supporting discussions like the usenet, but it is represented in HTML and is not distributed across the network but runs centralised on one server.

Newsgroups, mailing lists and forums can be moderated, and the names and email addresses of involved users can be hidden.

2.4.2 Conferencing

Conferences and meetings are an essential part of team work, because only in a meeting the group can discuss the different views about the goal, talk about how far the group have got to reach that goal, and what actions have to be taken in the coming weeks or months to come even more closer to the solution of the given task.

Borghoff and Schlichter give some more "benefits of group meetings" [BORGHOFF, p. 374]: Sharing of ideas, modification and synergy of ideas, different people can find problems more easily with an idea than the person that

had the idea, encouragement by knowing that you belong to a team, and people can learn from each other.

But there are also problems with meetings described in [BORGHOFF, pp. 374-375], e.g. how to partition the time and right of speaking between the participants, how to come up with new ideas when still listening to other's ideas, having not enough self-confidence to speak out new ideas, or some members of the group are dominating the meeting and the others are just passively participating.

Borghoff and Schlichter describe different kinds of conferencing which can be supported by groupware applications in different ways. The following information is taken from [BORGHOFF, pp. 95-100] if not another source is mentioned.

There are **face-to-face meetings**, i.e. all attendees are physically in one meeting room. The use of computers can be helpful in preparing and showing a presentation that helps all people to visualise what the speaker is talking about. A groupware application could also help record the meeting minutes or even more detailed information about the discussion. If necessary, all attendees could be provided with a personal computer so that they can contribute to the collection of information on e.g. an electronic whiteboard at the front of the conference room.

Another type of conference is called **distributed electronic meeting**. Such a meeting is between people that are not in the same room but can be in another building across the road or on the other side of the world. Borghoff and Schlichter divide this type of meeting in 4 categories:

- Asynchronous computer conferencing is normally realised by the use of email. It is not required that the people take part in the conference at the same time. This is also called "non-real-time computer conferencing" [BORGHOFF, p. 122]. Such conferences take a lot of time till they are finished with a usable result, but they don't appear on any schedule. So the danger is to underestimate the time that is needed for such conferences.

- Shared screen and audio or video connection: The participants can see the same screen and are able to talk about the subject. The identical screen is realised by a data link, and the applications in use should support the principle **What You See Is What I See** (WYSIWIS). This principle enables the attendees of such a meeting to work on the same document at the same time.

- There needs to be an additional link between the participants that carries the audio or video information. If the video images are not displayed on an extra screen but integrated into a window beside the window of the shared application on the screen of each participant, this type of conference is called "desktop conferencing" [BORGHOFF, p. 122].

- There is the possibility that there are not only human attendees, but also pieces of software called **agents** that act in a programmed manner and can, for example, support the meeting members by searching information about mentioned keywords, or by recording the minutes on their own. These agents could also participate in a face-to-face meeting, but then someone would be required to control the agent which means to type with the spoken information.

- In an **Virtual Reality conference** (VR conference) all participants are represented in the virtual space by avatars. Those avatars are virtual characters that are controlled by the hardware that is monitoring the belonging participant. The hardware can be e.g. a microphone for the audio information, or a data glove to visualise the movements of the attendee.

2.4.3 Calendaring and Scheduling

Calendaring and Scheduling is necessary to enable a group to organise meetings. Calendaring is the main subject of this diploma thesis, so this topic gets more attention than the other applications of groupware.

This chapter should give an idea about the terms involved in calendaring and scheduling which are used by most applications in the same way. A detailed example of a calendaring standard is described in the chapter 3.3, which is about the Internet calendaring standards.

Calendaring and sociological implications

Palen describes in her dissertation the history and the background of calendaring. She states that schedules make "it possible to dependably coordinate with others" [PALEN, p. 40]. She says there are public schedules, that determine e.g. when shops are open, or when there is a public holiday. Our public schedules are also influenced by nature, which makes the difference of day and night or summer and

winter. Personal schedules allow yourself to plan your own time, and they enable the people around you to have access to you on agreed times in order to coordinate action. These schedules don't need a physical representation, they often are just maintained with speech acts.

Palen also dicusses some sociological aspects of calendars and time in general. She says that time is a possession of every person, and people can show each other respect or low regard by either giving time or letting someone else wait [PALEN, pp. 27-28].

Another more practical problem is that if management introduces the use of shared electronic calendars, they are the ones who mainly get the benefit from it because it is easier to organise meetings [PALEN, p. 10]. But it is up to the normal group member to enter all his appointments into the shared calendar which often takes more time than just to maintain a paper calendar [PALEN, p. 11]. There is a "critical mass" [PALEN, p. 113] of people needed that maintains their calendars so that group calendaring can be used effectively in an organisation.

Another challenge of group calendaring applications is that some people already use their own personal calendar application with special advantages, e.g. portability if it runs on a **Personal Digital Assistant** (PDA), or it supports a special to-do list. Those people have to manage the synchronisation of the different calendars [PALEN, p. 111]. It is necessary to convince those people how important it is to have shared calendars so that they are willing to take on the additional effort of synchronising.

But calendaring can also be a great help to remind people of recurring personal dates, e.g. birthdays of old friends. That helps to stay in contact, and this is a good social component [PALEN, p. 112].

Personal Calendaring and Scheduling

Reekes, Vice President of MeetingMaker, a company involved in calendaring software, describes the evolution of electronic calendaring: It started with personal calendars, was extended to group and shared calendars, and was finally developed to collaborative scheduling. The following lines are a summary of Reekes' text [REEKES].

A personal calendar is in Reekes' definition a simulation "of paper-based day

planners" [REEKES, p. 1]. It helps the user to "track appointments and manage daily tasks". A personal calendar fits together with the idea of **Personal Information Management** (PIM). PIM applications started to exist on the desktop of personal computers, and then got their own hardware with the invention of the **Personal Digital Assistant** (PDA).

A group and shared calendar is an improvement to the personal calendar because it gives the opportunity to see the calendars of other people in the organisation. So the calendar is not isolated from its environment, but shared inside the organisation. The user of a shared calendar knows about the availability of his colleagues and can therefore better coordinate his own time or the time of the group. The use of group calendaring became possible when client/server technology was becoming cheaper and more common.

Collaborative scheduling allows the user to interact with the other users through the calendar software, and to arrange meetings synchronously and in real time. That means he can invite other persons to a meeting, and immediately get the response from their calendaring software whether they are free at that time or not, and then he can book the time on their calendar so that later attempts of arranging a meeting would realise that there could be already another meeting at that time.

Another source from Alt-N Technologies states that scheduling depends on "current and accurate" [ALT-N T] personal calendars, otherwise the booking of meetings would not be reliable. It has to be assumed that the free time on the calendar is not already booked by appointments not listed there.

Meetings, Appointments, Events

The Microsoft Outlook 2000 Online Help defines these terms in this way [MSOUTLOOKHELP, see Using calendar; About appointments, meetings and events]:

"**Appointments** are activities that you schedule in your Calendar that do not involve inviting other people or reserving resources."

"A **meeting** is an appointment you invite people to or reserve resources for."

"An **event** is an activity that lasts 24 hours or longer."

So appointments are all incidents in which only the user is involved. A personal calendar would only contain these types of entries. Meetings affect at least 2 or

more calendars, and often need to be confirmed by the other attendees. An example for an event could be a vacation or a seminar. Sometimes there are extra calendars for the details of a seminar, but the whole happening as one entity is called an event.

Other Items of a Calendar

There are also other components of a calendar beside meetings, appointments and events. People also want to save notes or journals, to-do entries, and alarms in their calendar. These entries can often be linked to real calendaring incidents. Notes or journals help to extend the calendar to be like the personal paper notebook you carried with you, where you wanted to have a notice beside an appointment describing your personal thoughts about it. To-do items describe activities that are not bound to a special date or time, but have to be done during a given period of time. Alarms are useful to be reminded about an event. For example, an alarm can either remember you a day before the event to prepare something for the event, or it can help you 5 minutes before to stand up from the desk and walk to the meeting room.

Privacy

There are different levels of privacy that a calendar can allow to its users. All calendar solutions at least allow the user to mark a single event as private. That means that other people just see that there is an appointment, but cannot see the details. Some programs have all calendars strictly private by default, and allow the user to invite other people and give them different rights to see or change his calendar, or just to show them the time blocks when the user is available.

Time Organisation

Depending on the level of privacy, people can see if you are available or not available during a period of time, which is also called a time block. A time block can be marked with 4 different states: **free**, **busy**, **tentative** and **out of office**. "Free" means that you are available at that time, and "busy" means the opposite. "Out of office" also gives the other people the information that you are not available in the given period of time. "Tentative" is something in between. These time blocks can either be shown to the other people as if you were available, or in

a different style so that they know you might be available. This kind of property of a time block is used whenever an appointment has no fixed point in time up to which it has to be done, e.g. you just book it for the morning, but if another more important request comes in for a meeting, you can move your personal appointment to the afternoon.

Organising a Meeting

Group calendars allow the organiser of a meeting to select the desired attendees/resources and check at which time they are all available. He should see the free and busy times of all attendees/resources and therefore know when people are available or unavailable. The invitation is either sent via email or is directly inserted into the calendar of the attendee. At that point in time, the meeting is in the state "no response". Invited persons can accept the invitation or decline it or set its state to tentative. That means they might attend the meeting when nothing else is more important.

Booking Resources

Meetings don't only need to be attended by the right people, but also a room needs to be booked, and perhaps some equipment is required, e.g. a projector.

That means that every resource should have its own calendar. There can be a manager for a resource who decides in conflict situations which group or user should get the resource.

If there is a group of the same resources available, the calendaring application should assign a free resource to the request of the organiser.

Different Roles in Calendaring

The **organiser**, also sometimes called host, is the one person that invites other people to a meeting. He is allowed to change details of the meeting proposal.

The invited persons that join a meeting are called **attendees** or **participants**.

If someone has been invited to a meeting but is not able to attend the meeting, he can ask one of his colleagues or employees to replace himself at the conference. The one who delegates the participation is called the **delegator**, and the person that is sent to the meeting is the **delegate**. Another equal name of this role is

delegatee.

Another role is the one an assistant takes when he is acting on behalf of his superior. The assistant is then called the **designate**. He is allowed to manage the calendar of his chief, e.g. to organise a meeting that is held by the chief, but he is too busy to deal with the details of organising the meeting.

Recurring Events

There are different levels of granularity how to define recurring events. Nearly all scheduling products allow to define daily, weekly, monthly, and yearly events. Some of them also allow to give a day of the week on which the event should occur. Another feature is to give the interval of the event, so it is possible to arrange an appointment that occurs every second Monday. It's even better to be able to combine these attributes, e.g. to say an event recurs every first Monday in a month.

Repeating events often have an end date at which they stop repeating.

The calendaring software should allow the modification of the whole series of events as well as changes to a single event on a special day.

Realtime Calendaring versus Message Based Calendaring

There are always two different ways to realise calendaring: A calendaring solution can depend on electronic mail services, and use them to exchange invitations and responses.

In contrast to that, realtime calendaring uses an extra calendar server that holds all information of the calendars of the users, and invitations can be inserted into the calendar immediately. If the user is connected to the calendar server, he sees the invitation at the same time it has entered the server.

The advantage of this solution is that if the mail server crashes, calendaring still can be used, because it is a different server program. Furthermore it supports realtime collaboration: You can be sure that the free/busy times are accurate and that there are no unread invitations that wait for the user to be added to his calendar. Unread invitations are already integrated into the calendar and given the state e.g. "no response".

Another advantage is that the calendar and the email client don't need to cooperate too close, so you are able to use programs from different producers.

The Meeting Scheduling Problem

Sen and Durfee [SEN] define the following attributes of a meeting:

- a set of participants
- the owner, also called host or organiser
- duration
- priority
- starting time preferred by the organiser; if it is not completely defined, participants are allowed to propose e.g. the time of the given day, if no special time but only the date is given
- the latest possible starting time, also called deadline
- latest possible time of scheduling the meeting: It has to be decided to this point in time, whether or not the meeting will take place, and when exactly it is scheduled.
- The actual meeting start and end time, which is chosen in the end of the decision process

A personal calendar consists of the following attributes:

- the starting date of the calendar
- the end date of the calendar
- set of time slots, which can cover e.g. an hour or 30 minutes, and the information about what appointment or meeting is planned for this time slot, or if it's free.

There are several possible methods that help in organising a meeting: The organiser can make a proposal which is also called a **request**. Potential attendees can send a **reply** which can be positive or negative, or they can request changes of the meeting, e.g. another date or time. The organiser can **add** new details to a proposed meeting, but also **cancel** attendees or a whole meeting.

It is also a great help for the organiser when he can request free busy information from the people he wants to attend the meeting. When people **publish** their free busy details, he can see what times could be suitable for all or most of them.

More details about this topic can be found in [RFC2446_ITIP] and in chapter 3.3.

Agents in Calendar Applications

Borghoff and Schlichter describe the use of agents in groupware applications, and

they also write about agents helping in scheduling and organising meetings.

At first there should be a short summary about agents:

Agents are autonomous, they are able to act without intervention of the user [BORGHOFF, p. 419]. They know about their own internal state and about their available actions. An agent knows about the "problem domain" and is able to understand changes in its environment and how to "react appropiately". That means that the agent has its own behaviour.

A special language called **Agent Communication Language** (ACL) was developed at the **Advanced Research Projects Agency** (ARPA). This language consists of a vocabulary, a **knowledge interchange format** (KIF) and a **knowledge query and manipulation language** (KQML) [BORGHOFF, p. 416].

Agents can be used for **distributed problem solving** (DPS) [BORGHOFF, p. 425]. That means that several agents have "to cooperate in order to solve complex problems" [BORGHOFF, p. 425]. They have to share "their knowledge, goals, skills and execution plans". The complex problem is subdivided into several subproblems, that can be solved by one agent in coordination with the other agents.

The **Contract Net Protocol** [BORGHOFF, p. 438] helps to find out which agents can solve what subproblem, and it allows agents to negotiate how to exchange information in order to collaborate. The primary message types are: Task announcement, task bid and task assignment. There are also request, acknowledgement, termination and report messages. There is a manager that announces the problem, and agents can bid their available resources, and the manager decides what is the best bid. Based on that decision, he assigns the appropriate tasks to the bidders [BORGHOFF, p. 439].

Now follows the description about how agents are able to solve meeting scheduling problems:

It is assumed that every personal calendar is managed by an agent. The agent of the person that wants to schedule a meeting takes on the role of the manager in the contract net protocol, and sends his announcement with the attributes of the proposed meeting to the other agents, which are responsible for the several personal calendars of the participants. These agents match the proposed times with their set of time slots, and either reply with yes or no or "suggest a new time

interval" [BORGHOFF, p. 473]. The manager collects all the bids and decides whether a solution was already found, or if he has to reannounce the meeting with different time intervals, that better fit to the received bids from the other agents. If the final interval time is sent around, the agent of a personal calendar has to recheck if the time slot still is free, and then he reserves it for the meeting.

2.4.4 Shared Workspaces

This category of groupware application is also called "shared information spaces". Appelt writes in [APPELT, p. 1] that a shared workspace is a "repository of shared information". He states that a system with support for shared workspaces allows the users to upload and download files. It also enables a user to give other users different levels of access to his files. There should also be a version management system that helps to cope with different versions of the same document, which is especially a problem when several people are working on the same file.

Borghoff and Schlichter say there is "separate responsibility" [BORGHOFF, p. 123], i.e. it is clear which user is responsible for what parts of a document.

They also compare "mutual exclusive access" against "synchronous access" [BORGHOFF, p. 123]. With mutual exclusive access, only one person at the time has write access to a document. That is realised e.g. by locks. Synchronous access allows several people to work at the same time together on one document. Here again the WYSIWIS principle applies.

Another type of information sharing is in between those two types of access, it is working with "alternate versions" [BORGHOFF, p. 123]. That means that several people are working on local copies of the same document, and these different versions of the document have to be synchronised, which often requires manual adjustment to resolve conflicts.

2.4.5 Group Editors

The **What You See Is What I See** (WYSIWIS) principle was already mentioned several times, and group editors are those applications that realise this principle. Group editors enable several people to work on the same document at the same time. Each one of them sees the same version of the document. If someone changes the document, notification messages are sent to the other users and their screen is updated [BORGHOFF, p. 386].

The group editor helps users to be aware of the other persons that are working on the same document, and shows which parts of it are edited by whom. It has to avoid that people are working on the same segment of the document, because then it would not be clear which changes should be applied to the document. So only one user has write access to a segment of the document, but still the other users are allowed to read this segment [BORGHOFF, p. 121].

According to Borghoff and Schlichter, the architecture of a group editor consists of the data layer and the operation layer [BORGHOFF, pp. 397-399].

The data layer holds the content information with the content of the document including the formatting information. The structural information, which holds information about other users' activities, also belongs to the data layer. The data layer is completed with history information about previous document versions and user information, e.g. the email address.

The operation layer provides the functionality for editing the content and for changing the structure of the document. This layer is responsible for the coordination of updates, in order to have everywhere the same information, and for the coordination of locks that can be set by the user or enforced by the editor application. It furthermore notifies the user about the activities of other users working on that document [BORGHOFF, p. 399].

2.4.6 Workflow Management

When people are working together to achieve a common goal, they have different activities that need to be coordinated. Some activities can only be carried out asynchronously, e.g. when two activities on the same object cannot take place at the same time, because either one depends on the others' completion, or at least one of them needs unrestricted control over the object.

There are some different approaches described by Borghoff and Schlichter [BORGHOFF, p. 124]:

The "form-oriented systems" concentrate on the document that is forwarded from one member of the group to another, and each of them performs his task on the document. For example two people have both to add a chapter, the third person has to review it, and the boss gives his signature.

A "procedure-oriented system" divides the process of reaching the goal in a lot of

different procedures, which are subdivided into several steps. Each member of the group is assigned to one or more procedures. Each step in a procedure receives an information unit, processes it and returns the modified version, so that it can be the input for the next procedure.

"Conversation-oriented systems" try to model the normal interactions between human beings. There are electronic message equivalents for some speech acts, e.g. request, promise, reject, accept, cancel, etc. [BORGHOFF, p. 340].

The "communication structure-oriented systems" know about the specific roles that are assigned to the different members of the team, and are able to use this knowledge to bring some order in the processing of work.

2.5 Disciplines involved in the Development of CSCW Applications

Borghoff and Schlichter write that CSCW is "an interdisciplinary application domain" [BORGHOFF, p. 93]. The knowledge and achievements of several disciplines are required to develop and introduce a groupware system that is accepted by all its users in a company.

Computer science provides the necessary hardware and software capabilities.

Software designers need to develop applications and user interfaces that can be used in a simple and quick way, so that the users are willing to accept and use the groupware solution. People must see the benefit for themselves and the company from changing traditional ways of work to using CSCW.

Telecommunication technology connects different workstations both inside an office and also over long distances, even around the globe.

Information Management is needed to cope with the amount of data that is processed by the people involved and to provide access to the data to the right people at the right time.

Sociology was already mentioned in the chapter about calendaring, and it is an important discipline: Everyday work is changed a lot by CSCW to make work more efficient. Sociologists should ensure that people are not overrun by the increased teamwork and the awareness of others about everything they are doing, but to encourage them to enrich the team with their individual ideas and visions and to find a way how the goals of the group can be reached together without losing members of the team.

Organisational theory helps the management to be able to change the structure of the company in a way that CSCW really can help to improve the effectiveness of the whole company. Sometimes teams will be composed in a different way than without CSCW. For example, it is more difficult to organise teams that change with every project. Another problem for organising a company is that now people can be involved in different teams at the same time, but the managers of the teams are based at different sites of the company.

2.6 Distributed Systems and Applications

CSCW is based on distributed systems, because members of a group who use several personal computers want to share data and applications in order to work together. This chapter describes distributed systems and the underlying technologies and approaches.

Schlichter defines "a **distributed system** as one in which hardware and software components located at networked computers communicate and coordinate their actions mainly by passing messages" [SCHLICHTER, p. 9]. This means that the members of a group use applications that are able to share information and can help each other to process the requests of the users.

2.6.1 Software Architecture

The software layers of distributed systems are shown in the following table [SCHLICHTER, p. 27]:

applications, services
Middleware
Operating system
Computer and network devices

table 2.2 Software Layers of Distributed Systems

A **distributed application** is an application that consists of different components. Each of these components has its own state and operations to change this state. Each of the components can run on a different machine, and they use the network for communication [SCHLICHTER, p. 14].

A definition for **middleware** is also given by Schlichter:

"Middleware is defined as a layer of software whose purpose is to mask heterogeneity and to provide a convenient programming model to application programmers." [SCHLICHTER, p. 27]

So it is a good help for the design and development of distributed applications because it already provides services for communication between the applications, access to distributed data and the processing of distributed transactions [SCHLICHTER, p. 27].

Middleware also provides **transparency**, i.e. it masks all the difficult problems that come up with distributed systems. For example, location transparency deals with the physical storage location of objects in the distributed system. Location transparency means that the middleware gives a logical name for the object to the application programmer. He can easily use that name without thinking where the object is located at the moment. The current location of the object is resolved by the middleware, whenever the application accesses the object through the middleware [SCHLICHTER, p. 28]. There are a lot of other types of transparency, which cannot all be covered here.

2.6.2 Communication Architectures

One very often used architecture to realise communication between CSCW applications is the **client/server** model. Some definitions should be given although the terms have already been used above. They have all been taken from [SCHLICHTER, p. 45].

- A **client** is a process that "initiates requests for service operations".
- "A **service** is a piece of software that provides a well-defined set of services."
- "A **server** is a subsystem that provides a particular service to a set of a priori unknown clients."

The clients are not known to the server before they request a service.

There is a difference between a client and a client machine, and even more between a server and a server machine: Clients normally run on client machines. Services can run on one or more server machines, i.e. a service itself can be distributed. Several servers can run on one server machine.

Client and server are often operated in **handshaking** operation: The client requests a service from the server, then it waits and depending on the implementation that can block the whole client application. When the server sends

the reply with the result of the request back to the client, the client resumes.

The **multitier** architecture introduces a new kind of application: This application is both a server and a client and is the interface between another client and another server. So there are requests sent from the client to the application in the middle, which acts as a server for this client, but itself requests services from the other server.

The example of the web applications can help to illustrate this architecture:
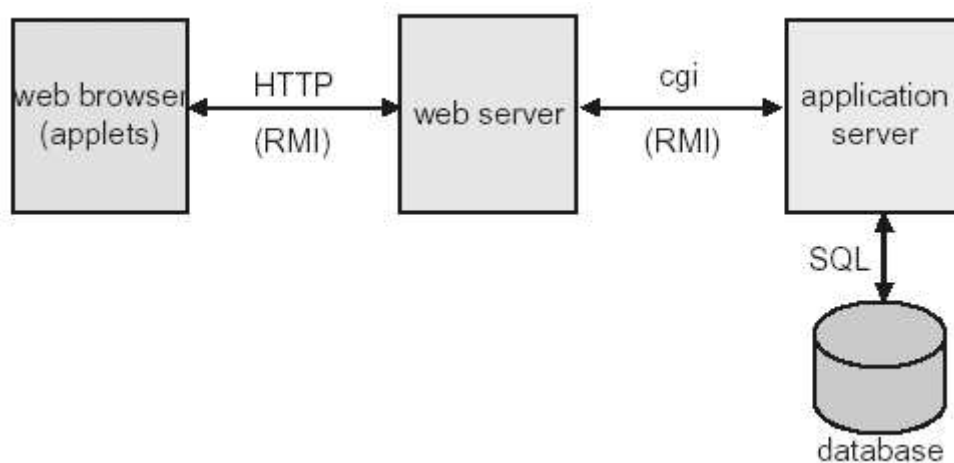
figure 2.1 Example of a multitier architecture

This figure was taken from [SCHLICHTER, p. 47].

The web browser is the client and relies on the web server. Additionally applets can run inside the web browser that have their own application code, but also require services from the server. The web browser can either communicate with the web server using the **Hyper Text Transfer Protocol** (HTTP) or the Java **Remote Method Invocation** (RMI) functionality, which will be explained below.

The web server itself is not able to handle the requests of the clients on its own. So the web server requests services from an application server. This is a server that is able to e.g. access a database and process the required services, and then sends the result to the web server, which now again acts like a server and replies to the web browser.

That example shows how the web cannot only be used for information retrieval for which it was initially designed, but also to provide all kinds of services and

therefore to support groupware applications as well. More information about web based groupware applications can be found in chapter 4.

There are a lot of application servers on the market, and it is not in the scope of this text to investigate application servers or their architectures. For further information about this topic, the reader is referred to Schlichter's description of the **JavaTM 2 Platform, Enterprise Edition** (J2EE) which basically is "a distributed application server environment" [SCHLICHTER, p. 23].

There is furthermore the **peer-to-peer** model that consists of several processes or applications which can all communicate directly with each other. There is no process that has special functionality to coordinate the other processes. That means that they all have to contain their normal application code and additionally some code that provides the coordination [SCHLICHTER, p. 42]. The advantage of this model is that no special server needs to be installed and configured. The disadvantages are bad scalability and difficult maintenance: Bad scalability means that with increasing numbers of participating applications, the traffic of exchanged messages will also increase rapidly, and that will cause poor performance. The maintenance of a huge number of applications is certainly much more expensive than the maintenance of only some servers. Also backup is easier when there is a more or less centralised solution.

2.6.3 Remote Procedure Calls

This chapter shows one aspect of middleware.

There are two ways how distributed applications can communicate with each other: Either message-based or using **Remote Procedure Call** (RPC). Java's object-oriented version of RPCs is called **Remote Method Invocation** (RMI).

The following table gives a good overview about which type of communication between distributed applications fits in which layer of the OSI model [SCHLICHTER, p. 62]:

client-server model	layer 7 application layer
RPC	layer 6 presentation layer
message exchange, e.g. request-answer protocol	layer 5 session layer
transport protocols e.g. TCP/UDP	layer 4 transport layer

table 2.3 The OSI model and Distributed Applications

The request-answer protocol is described in [SCHLICHTER, p. 62]. It provides bidirectional communication and uses timeouts and acknowledgements to detect and recover from message loss.

The advantage of RPCs is that it provides already all failure detection and correction, and normally works synchronously. So a programmer of a distributed application client just can use an RPC call to execute a remote procedure that is provided by a server, and the returned result can be used for further operations of the client.

Brokering

There is the problem that distributed applications need to know which component in the distributed system can offer which service. An RPC can only be started if the server is known that would be suitable. This problem is solved by the process of **component binding** [SCHLICHTER, p. 72]. There is static binding which means that the server address is hard-coded into the client application during its generation. Dynamic binding takes place immediately before an RPC is performed: The client needs to know a broker that can tell the current address of the server or acts as a mediator between client and server and just forwards the messages. The requirement is that the broker always needs to be informed if the server changes its address or service availability.

The **Common Object Request Broker Architecture** (CORBA) is the object-oriented approach for dynamic component binding and provides the brokerage of objects instead of procedures and services [SCHLICHTER, p. 165 ff.].

XML-RPC and SOAP

The goal is to use a standard language to form the RPC requests and replies. The currently most favoured language is the **Extensible Markup Language** (XML), because it provides well structured documents that are readable by all kinds of programs, only to mention some advantages.

XML-RPC is the easiest specification that provides RPC calls that are represented in XML documents and are transported via HTTP. For more information about XML-RPC see [XML-RPC].

The **Simple Object Access Protocol** (SOAP) is an extension of XML-RPC, and is a standard of the **World Wide Web Consortium** (W3C). It is described in [SOAP].

2.7 Enterprise Application Integration

The term **Enterprise Application Integration** (EAI) stands for a new idea how companies can use computers. This topic does not directly affect the cooperative work of single users, but allows cooperation of different applications inside an enterprise. Hansen writes the goal of EAI is about "getting heterogeneous applications to work together in support of changing business processes" [HANSEN].

A characterisation of EAI comes from Ren [REN]: He states EAI has the "ability to integrate applications within the enterprise as well as across enterprises", and it provides "infrastructure adaptability", "support for multiple integration topologies" and the "power to handle complexity".

Buyens gives this definition of EAI: "EAI is the ongoing process of putting an infrastructure in place, so that a logical environment is created that allows business people to easily deploy new or changing business processes that rely on IT." [BUYENS]

It always helps to understand a new technology or idea by looking back in history to realise what the problems are and what made this new idea necessary. Inmon [INMON] has written a text to investigate the history of integration. He mainly focuses on data integration, which is only a part of EAI, but it helps to understand the overall background of EAI. In the early beginning of computer usage in companies, the first and only requirement was to "replicate manual procedures on the computer". So there was one application for each procedure. When better

technologies came up, e.g. database technology, the existing applications where just extended, but they and their data remained seperated from each other. The problem is that there are actually data and procedures available in the companies' applications, but the applications cannot access each other's data or use the other's functionality.

An example could be a mainframe machine running programs written in Cobol, and now it is required to allow webaccess to the data managed by the established applications. That means you have to deal with different platforms and different standards.

Today, companies need to react very fast and implement solutions immediately. Competition has increased much more, because the Internet allows everyone to provide his solution to the whole world at once. The goal is to minimise the time of the development of a product until it is ready for the market. So it really helps the developers when they don't need to develop a whole application, but can combine the existing solutions and just add the required functionality.

So there are a lot of products and ideas on the market that try to bring different applications together to enable them to exchange information. They can be categorisied in different ways. One categorisation is described in the following chapter.

2.7.1 Categories of Integration

The Hurwitz Group has defined the following types of integration. They are described by Gold-Bernstein [GOLD-BERNSTEIN]. The level of abstraction increases in the following list of integration categories. "Platform integration" is at the lowest level of abstraction, then there is "data integration", "component integration", "application integration" and "process integration". "Business to business (B2B) integration" is at the highest level of abstraction.

The goal of "platform integration" is to bring together different architectures of "hardware, operating systems and application platforms" [GOLD-BERNSTEIN]. Communication can be established with messaging, **Object Request Brokers** (ORB) or **Remote Procedure Calls** (RPC).

"Data integration" provides gateways between different data stores, and also offers transformation from one data format to another.

"Component integration" deals with cooperation of applications concerning transactions and the use of the business logic represented in the procedures of only one component.

"Application integration" means that different applications can work with each other by using application adapters (also called connectors).

"Process Integration" stands for several applications that are involved in the same process. This is already a high level of abstraction, because the business manager can model workflows on this level. The workflows define how the applications have to work together and how e.g. an incoming order goes through several processes of different applications so that finally the product can be delivered.

"B2B integration" is integration not only inside the enterprise but between several companies that either work together as partners, suppliers or customers. This type of integration is also needed in mergers or acquisitions when two companies have to find the best way how to share existing data and procedures.

There are also other categorisations of integration, but this is not the place for further investigation into this topic.

2.7.2 EAI Architectures

Kang [KANG] gives a good overview over different methods of integration and their physical integration architectures. He compares "point-to-point" and "middleware" integration, and concludes, that point-to-point integration is not possible when there are too many nodes that need all to be connected with each other. Using middleware-based integration, all nodes are connected to one mediator in the middle and can communicate via this middleware.

He also describes the "message bus architecture", the "centralised architecture" and the "J2EE connector architecture" (**JavaTM 2 Platform, Enterprise Edition (J2EE)**).

Using a message bus architecture, all messages are multicasted, i.e. applications send messages to all other applications at the same time, and the addressed application reacts to the message. This causes a lot of traffic, and it is not secure because not only the addressed application can read the message.

The centralised architecture consists of one server that provides message filtering

and message delivery to the connected applications, which need each an adaptor to being able to communicate with the server.

With the J2EE connector architecture, the server provides adapters for all connected applications, i.e. the adapters are not needed anymore in each application.

There are some good diagrams about these different architectures in [KANG].

2.8 Groupware, Pervasive Computing and Synchronisation

The goal of Pervasive Computing is to enable the user to access **all** his data **anywhere** and **anytime** from **any** device. The problem is that still not all people have devices that are always connected to the Internet. One reason for that is the sensitive data which is usually shared in a groupware environment. The use of a collaboration system is normally limited to the intranet for security constraints. Another reason for not every PDA having always access to the Internet is that it is too expensive at the moment.

Another challenge to groupware applications is that people want to have access to their groupware data from any device: They want to log in on every computer on the network and use their data, and then they also need to work with their PDA. When they are travelling, they use additionally a laptop, which also should provide at least the personal calendaring information. When they are not connected to the network, they still want to insert personal appointments, and wish to later copy them to the corporate groupware system. So there is the need for synchronisation between PDAs, the main server, and client programs on laptops.

3 STANDARDS FOR EMAIL, CALENDARING AND SYNCHRONISATION

This chapter describes the standards that allow Email, shared Calendaring and synchronisation. The Calendaring chapter was divided into two chapters, because there are on the one hand the Internet calendaring standards and on the other hand a mix of other standards used for performing scheduling.

Some specifications are mentioned, but not fully paraphrased here. It is recommended to read the original specifications for further information.

The **Internet Engineering Task Force** (IETF) develops and announces the standards for the Internet. Each standard is defined in a **Request for Comments** (RFC).

3.1 The people involved

The goal of Computer Supported Cooperative Work is to support people in accomplishing their daily tasks. The question is how these persons can be represented in the system, and how they are identified by the system and by other users inside and outside the system.

Identification by the system is needed for the login process as well as for assigning the associated data to the correct user. An example for such associated data would be a mailbox that belongs to a specific person.

Identification by other users can be compared to the conventional visiting card. Other users like to have some public information about a person to being able to stay in contact and for further cooperation.

3.1.1 Directory Services

In order to deal properly with users, a system needs a directory where the users are registered. This directory can be understood like a telephone directory that enables you to communicate with any of the huge number of people in your town. In contrast to such a global directory, an address book only contains the addresses of the people you met already or had some kind of contact with them.

Every server that manages several users needs any kind of directory.

A directory needs to implement an information model and to provide a protocol for querying and manipulating the information. A directory service is used by directory

enabled applications that are able to use the information stored inside the directory.

X.500

X.500 was developed by the **Comite Consultatif International Telephonique et Telegraphique** (CCITT), which is now known as the **International Telecommunication Union - Telecommunication Standardisation Bureau** (ITU-T), and the **International Standards Organisation** (ISO). The first version of the standard was published in 1988, a second version was finished in 1992.

In his book "Understanding X.500 - The Directory" [CHADWICK] David W. Chadwick describes the history and the different parts and aspects of X.500. His book is the source for the following paragraph describing the X.500 standard.

There are the following parts of a X.500 directory model that should be mentioned:

The **Directory Information Base** (DIB) contains all the information stored in the directory. It consists of a set of entries which are instances of one or several object classes. Each object class is formed by several attributes with an attribute type and one or more attribute values. All the entries can be represented as the nodes of a **Directory Information Tree** (DIT). That means the entries stand in a hierarchical relation to each other.
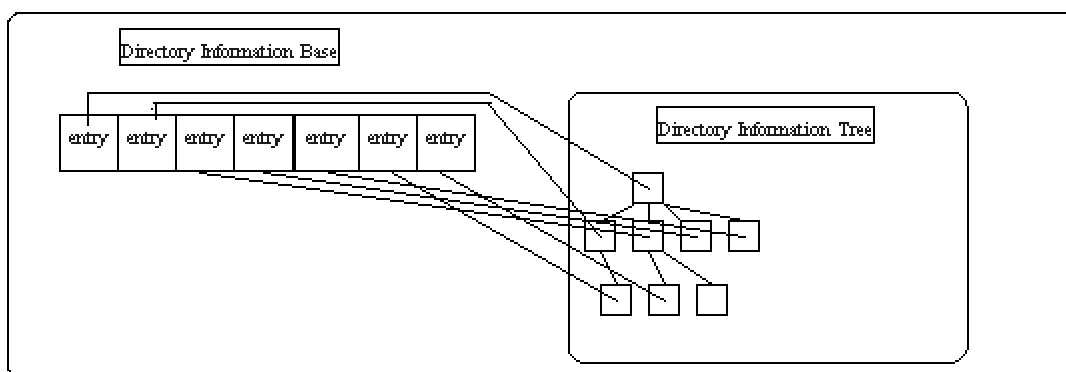
**figure 3.1 The X.500 information storage**

The X.500 Directory model does not need to be a centralised system: It supports replication and distribution of information.

The **Directory User Agent** (DUA) is the component of the system that is run by the user. The user can insert his requests and retrieve the results.

The system itself is made up by several **Directory System Agents** (DSA) which

are connected via the **Directory System Protocol** (DSP). They provide the **DSA Abstract Service** (DAS) to each other.

The DUA uses a protocol called **Directory Access Protocol** (DAP) to interact with the system. This protocol deals with the system as a black box. If a DSA cannot answer a request, a process called chaining is started: The DSA forwards the request to its connected DSAs. It waits for the results and finally returns a positive or negative answer.
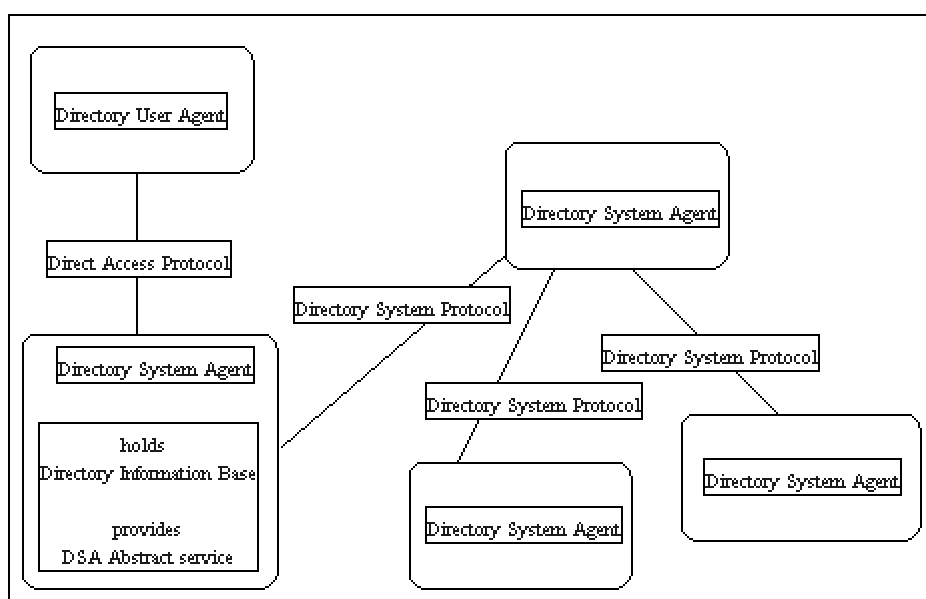
figure 3.2 The X.500 directory structure

DAP uses the **Open Systems Interconnect** (OSI) network protocol stack, i.e. DAP provides a poor performance because every request and reply goes through all 7 layers of the OSI model. Because of that dependency on the whole OSI stack and the large functionality provided, the X.500 model can be called heavy weight.

Light Weight Directory Access Protocols (LDAP)

The first RFCs for LDAP were published in 1993. A good description of LDAP and other directory services can be found in the diploma thesis of Norbert Klasen [KLASEN, p. 12]. The following information is taken from his text.

In contrast to the heavy weight X.500 model of a directory, LDAP directly uses the TCP/IP stack. That means the layers above the transport layer are bypassed, which are the session and presentation layer. Some special functionality like chaining is also missing, but because of that the LDAP system is lighter. Instead of chaining, the server without useful information redirects the questioning user to the

31

other known servers.

The goal of LDAP is to provide as much as possible the same functionality like X.500, but to be lighter. There are gateways to connect LDAP to X.500.

3.1.2 The user's representation by a vCard

The vCard, also called the Electronic Business Card, was invented by an initiative called Versit. Versit was founded by Apple, AT&T, IBM and Siemens. Since the end of 1996, the Internet Mail Consortium now has the control over vCard. The vCard specification [VCARD] describes the details of this standard.

vCard is also described in the RFC 2425 [RFC2425_MIMEDIRECTORY] and RFC 2426 [RFC2426_VCARD]. The goal is to provide the users with several ways to give their details to other users. It should be possible to send the information with email (placing the vCard in a MIME attachment), but also to send it via an infrared connection from one PDA to another. The vCard cannot only hold information about a person but also about a resource. The vCard is related to the Directory Service by representing the details of an entry of the directory. The attempt was made to map the attributes of a vCard to the attributes described in the X.520/X.521 standards [VCARD, p. 2].

There are many address book applications that support the import and export of contact details via the vCard standard.

The vCard is a text file that uses the syntax specified by the MIME specification which is described in RFC 1521 [RFC1521_MIME]. See more about encoding below.

This is an example of a vCard, created by the phpGroupWare address book:

```
BEGIN:VCARD
VERSION:2.1
X-PHPGROUPWARE-FILE-AS:phpGroupWare.org
N:Testsurname;Testfirstname;D.;Dr.
FN:Dr. Testfirstname D. Testsurname
BDAY:1970-01-01
URL:http://www.om.org
ORG:Operation Mobilisation
A.ADR;WORK:;;London Road 123;Carlisle;Cumbria;GB
LABEL;WORK;QUOTED-PRINTABLE:London Road
123=0D=0ACarlisle=0D=0ACumbria=0D=0AGB
END:VCARD
```

3.2 Email

This chapter covers several standards concerning email delivery and reception and the format of a mail.

There are two standards which describe the email transporting process and the addressing format.

2 other standards provide access to received emails for users that are not always connected to the network.

Finally there is the email message and its standardised format itself.

3.2.1 Mail Transmission Protocols

Some of the following information was extracted from [EDMONDS].

Generally, there are three types of components: The **Mail User Agent** (MUA), the **Message Store** (MS) and the **Message Transfer Agent** (MTA). The MUA is that software that helps the user to send and to receive electronic mail. The Message Store keeps messages that have been delivered earlier already. Each MUA and each Message Store need to be connected to an MTA. The MTA is the program that is responsible for routing the messages to the correct destination. More than one MTA can be involved in an email delivery, when the message is routed through several networks. That means that the MTAs always need to be online on the network.

X.400

The X.400 standards were published in cooperation by ISO and CCITT, now called ITU-T. That are several standards, all with a name in the format X.4xx. In 1984, the X.400 standards were described in a "Red Book" series. These standards can make use of the directory services provided by an X.500 implementation, which was mentioned earlier.

There are several protocols that define the communication in an X.400 Message Handling System. These protocols are just given simple names: P1 deals with message transfer between MTAs, P3 describes connections of both Message Store or MUA with the MTA, and P7 defines the rules of conversation between the Message Store and the MUA.

X.400 is a protocol that is based on the OSI model and belongs to layer 7, the

application layer.

An X.400 address consists of values for the country code (C), administration domain name (ADMD), private domain name (PRDM), given name (G), and surname (S). An example address could look like this:

`/C=UK/ADMD=OM/PRMD=ICT+Carlisle/G=Testfirstname/S=Testsurname`

Some information in this paragraph originates from [DIFFUSE].

SMTP

The **Simple Mail Transfer Protocol** (SMTP) is an alternative to X.400. In contrast to the official X.400 standard, SMTP is only a "de facto" standard. But there is the RFC 2821 [RFC2821_SMTP], that is a recommendation how to implement SMTP compliant software.

SMTP is an Internet protocol, i.e. it is directly based on TCP/IP.

An SMTP address consists of a user name or alias, often used in a format like "firstname.surname". Then there is the @ ("at") character, and it is followed by the computer name that runs the mail server of the user. This computer name consists of the domain name and the toplevel domain name, which is assigned either concerning the type of organisation which the user is working with, or the country where the user lives. An example address could look like this:

`Testfirstname.Testsurname@c.ict.om.org`

c and ict are subdomains of the domain om.org. Because OM is a worldwide operating organisation, every department has its own subdomain, in this case the department is the International Coordination Team, and because it is based at two locations, the c stands for Carlisle.

The basic structure of SMTP defines an SMTP client (that is the MUA) and an SMTP server (that is the MTA). They can communicate with each other via SMTP commands and replies. The client also can send mail data to the server.

Comparison SMTP vs X.400

The X.400 addressing format is quite complicated, but it can deal with all kinds of letters, and is therefore preferred by French speaking countries, e.g. Canada and France. SMTP addresses are easy to remember, but do not support special characters.

X.400 supports attachments, but loses the file names. SMTP needs the MIME

(see more about email encoding below) extensions for attachments.

SMTP is the most often used standard for electronic mail communication.

3.2.2 Remote Mailbox Access

Normally, the user is not always connected to the network, and is therefore not able to receive all of the emails that are addressed to him. The solution is that the mailbox of the user resides on the server, which usually also hosts the MTA. Now the user needs a way to be able to read the mails that are saved on the server.

Terry Gray [POP3IMAP] describes the three types of operation how the user can access his email:

- In offline mode, the user accesses the Mail Transfer Agent from time to time and downloads the new received messages to his personal workstation. The downloaded emails are removed from the server.

- In online mode, the user reads the messages while he is online, the messages stay on the server, and he can manipulate the mails remotely. He has no copy of them on his machine.

- In disconnected mode, the user downloads the mails from the MTA, can manipulate them locally, and when he connects again to the MTA, the local and the remote mailbox are synchronised.

Most of the following information is taken from the article written by Terry Gray [GRAY]. X.400 has its own protocol P7 to support the different operation modes, but if only Internet orientated protocols should be used, the following protocols apply.

The next paragraphs describe the two protocols POP3 and IMAP4 which both provide access to emails only. They assume the user uses SMTP for sending mails. Both use the Internet Protocol for transportation. POP3 is defined in RFC-1725 [RFC1725_POP3] and IMAP4 is defined in RFC-1730 [RFC1730_IMAP4].

POP

The **Post Office Protocol** (POP) is originally a protocol that just supports offline operation. There is the attempt to provide online operation, by allowing the user to leave the messages on the server. But this is not a real online mode. One missing feature is the support of state information about an email, e.g. answered, read, etc. This is not supported by POP. Another problem is that there is no way to handle

several remote folders, and you can't store a message on the server.

IMAP

The **Internet Message Access Protocol** (IMAP) fully supports the offline operation model, but is is also designed for online / disconnected operation. It is capable to resynchronise an offline clients to the server. It supports remote message folders that can be modified like local folders. Some functions of IMAP: "IMAP4 includes operations for creating, deleting, and renaming mailboxes; checking for new messages; permanently removing messages; setting and clearing flags; RFC 822 and MIME parsing; searching; and selective fetching of message attributes, texts, and portions thereof." [RFC1730_IMAP4]

Conclusion POP vs IMAP

The best usage of POP is the offline mode, because it was created for that type of operation.

If more functionality is required, IMAP is the best solution, because it supports all 3 operation modes. But it is more complicated, because it provides more functionality and possibilities to the user.

So it depends on the requirements of the users what protocol suites them best.

3.2.3 Email Formats

Again, X.400 has its own rules and standards to encode mail messages and deal with attachments. But there is no place here to go into details about it, because the Internet standards are more commonly used.

The **Internet Message Format** RFC [RFC2822_IMF] defines the envelope and the contents of an email, also called the header fields and the message body. SMTP handles emails in this format.

The Header Fields of an Internet Message

The following example [RFC2822_IMF, p. 44] shows some of the possible header fields in an email that is built according to the Internet Message Format.

```
From: Mary Smith <mary@example.net>
To: John Doe <jdoe@machine.example>
Reply-To: "Mary Smith: Personal Account" <smith@home.example>
Subject: Re: Saying Hello
Date: Fri, 21 Nov 1997 10:01:10 -0600
Message-ID: <3456@example.net>
In-Reply-To: <1234@local.machine.example>
References: <1234@local.machine.example>
This is a reply to your hello.
```

A description and categorisation of the header fields can be found in [RFC2822_IMF, pp. 20-28]:

There are fields in the header that show who sent the message ("originator fields") and who is the receiver ("destination address fields").

The "identification fields" help to identify the current message and its related messages, e.g. the identification string of that message that this message is the reply to.

The "informational fields" provide information only important to the user, e.g. the "subject" field.

There is the possibility to resend a received mail to another person and make it look like it was sent by the original sender. This is realised with the "resent fields". These fields ensure that the receiver still knows not only the original sender but also who was the resender.

"Trace fields" are filled by SMTP, and they provide information about which way the message took across different servers to reach its destination.

There is a way to implement additional fields to build new standards based on the Internet Mail Format. These fields are called "optional fields".

The Message Body of an Internet Mail

According to the Internet Mail Format, the content of an email is just "made up of characters in the US-ASCII range of 1 through 127" [RFC2822_IMF, p. 6]. The length of the lines is limited to 998 characters. This is enough for text content, when no special characters are used.

But sometimes there is the need to add attachments, e.g. multimedia files, images, or other files in binary format, that means files that use all 255 possible values of a

character. Also some languages use special characters, that are beyond the 7-bit representation.

There are some ways for translating (or also called encoding) these files into text files, and then to decode them back. Some names of encoding types are "Base-64", "Quoted-Printable", "uu-encoding", and "BinHex".

Base-64 and Quoted-Printable will be described in more detail in the next chapter.

MIME

The **Multipurpose Internet Mail Extensions** (MIME) are defined in RFC 1521 [RFC1521_MIME]. These extensions support multipart messages, i.e. a MIME encoded email can contain several files in all kind of formats. That enables MIME to be "a standard for the transfer of multimedia information via the Internet" [BORGHOFF, p. 117].

The following table shows the recursive definition of an email message in MIME format:

message	body is an entity	header (includes SMTP header information)	
		header of the entity (holds the content-type value)	
		body of the entity is one of the types on the left	text
			multipart: several body parts of a given subtype, each of them is an entity
			application
			message
			image
			audio
			video

table 3.1 Structure of a MIME Message

The iteration occurs on entity and message: The body of an entity can hold a message, and the multipart content can consist again of entities.

The "Content-Type" value of a header describes the content of the associated body, and the "Content-Transfer-Encoding" value gives the type of encoding of the body. MIME uses "quoted-printable" and "base64 encoding", or unmodified data in "7 bit", "8 bit" or "binary" format. "7 bit" stands for ASCII characters with short lines, "8 bit" data has short lines to be sent via SMTP, and binary data cannot be used with SMTP [RFC1521_MIME, pp. 14-15].

The idea of "Quoted-Printable" encoding is described in [RFC1521_MIME, p. 18].

It is mostly used for text that includes some special characters beyond the 7 bit range of value. The goal is to keep the text more or less readable by humans. Only special characters are modified according to a set of rules. That costs some bytes, because these modified characters must be different from the normal characters.

"Base-64" Content-Transfer-Encoding is not concerned about humans, and therefore encoded files "are consistently only about 33 percent larger than the unencoded data" [RFC1521_MIME, p. 21]. Basically, this encoding converts a file to a set of 6-bit characters, which means that there are 64 different values. Three bytes (8-bit) are converted to four 6-bit characters.

A "text" does not need to be an ASCII text, but can also be encoded. The charset parameter in the Content-Type field tells the character set of the text. The default is the US-ASCII character set, but also character sets for other languages with their special characters can be set here.

The binary data of some application ("application"), another email ("message"), or an "image", an "audio" or a "video" file can also be the content of an entity.

A message with several body parts ("multipart") is associated with a special subtype, which is one of "mixed", "alternative", "digest" or "parallel" [RFC1521_MIME, pp. 34-37].

"Mixed" means that there are entities with different types. If those entities do not have a header, they are just text files in US-ASCII encoding.

The subtype "alternative" allows the sender to include a document in several file formats. A "digest" is a set of email messages. "Parallel" is the same as "Mixed", but there is no significant order of the parts. The body parts are meant to be displayed at the same time.

The specification of the "Security Multiparts for MIME" [RFC1847_SECURE_MP_MIME] defines two additional types for bodyparts of a multipart entity: "Multipart/Signed" and "Multipart/Encrypted". That allows to assure who sent a message and who is able to read it.

3.3 Internet Calendaring

The "Guide to Internet Calendaring" [RFC3283_ICALGUIDE] explains the relations between the different IETF RFCs that together define the standards for Internet

calendaring.

There are some terms used by the Internet calendaring standards. They are described in [RFC3283_ICALGUIDE, p. 3], and the following information and quotations are taken from there.

Calendars are the "basic storage containers for calendaring information" of a person or resource. Calendars are made up of several **components**, which have **properties** that carry the information about the component.

A **Calendar User** (CU) is an "entity (often a human) that accesses calendar information".

A **Calendar Store** (CS) is a data store which can hold several calendars and other calendaring information.

A **Calendar Service** is a program that allows several calendar users to access their calendar stores.

The **Calendar User Agent** (CUA) is the application that runs on the user's machine and allows to manage the calendars provided by a calendar service or a local calendar store.

Calendar Access Rights define which users are allowed to perform reading or writing operations on what information on a given calendar.

The Internet calendaring standards are described in detail in the next chapters. [RFC3283_ICALGUIDE, p. 4] gives a short overview of these standards: There is a definition of a language for describing calendar objects (iCalendar), an application protocol for using the language to perform scheduling (iTIP), and a description how to transport the messages via email (iMIP). At last, there is another protocol that allows to do calendaring with a calendar service in realtime (CAP).

There is an interesting comparison of the Internet calendaring standards and the Internet message standards in [RFC3283_ICALGUIDE, p. 4]: iCalendar has the same goal as the Internet Message Format defined in [RFC2822_IMF], iTIP and iMIP can be compared with SMTP [RFC2821_SMTP], and CAP is analogous to POP3 [RFC1725_POP3] or IMAP [RFC1730_IMAP4].

New standards are in development and at least should be mentioned here: xCal [XCAL] allows to use the iCalendar language not only in the MIME environment

but also with XML. SkiCal [SKICAL] will extend the iCal language so that yellow-page directory listings can be enhanced to provide not only address information and descriptions of services, but also upcoming events and happenings.

3.3.1 iCalendar

iCalendar is defined in the "Internet Calendaring and Scheduling Core Object Specification (iCalendar)" [RFC2445_ICALENDAR]. iCalendar is a language to define the components of a calendar. It is based on the work of the vCalendar specification, that was written like the vCard specification by the Versit consortium, and is now controlled by the Internet Mail Consortium.

There are the following types of components that can fill a calendar: Event, To-Do, Journal entry, Free/Busy time information, Time Zone information, and Alarm.

Events, to-dos and journals can be recurring. Events and to-dos can contain one or several alarm components.

iCalendar defines several data types which are used to represent the properties of the components. Some of the data types will be explained with the examples.

For a complete understanding of all data types and all properties, the reader is referred to [RFC2445_ICALENDAR].

The Event Component

An event can have a start and an end date. It can be related to to-do or journal components.

This is an example from [RFC2445_ICALENDAR, p. 54]:

```
BEGIN:VEVENT
UID:19970901T130000Z-123401@host.com
DTSTAMP:19970901T1300Z
DTSTART:19970903T163000Z
DTEND:19970903T190000Z
SUMMARY:Annual Employee Review
CLASS:PRIVATE
CATEGORIES:BUSINESS,HUMAN RESOURCES
END:VEVENT
```

There is an identification string UID that allows to build relations between components of a calendar. DTSTAMP gives a timestamp to show the date and time when this message describing the event was created. Another tag, LAST-MODIFIED, describes the time of the last change to the definition of this event. The CLASS value says whether the event is private, public or confidential.

For recurring events see the extra paragraph below.

An event also can be a reminder without consuming time, e.g. to mark a special anniversary. Then the tag TRANSP with the possible values OPAQUE and TRANSPARENT should be set to TRANSPARENT, so that it is not published as busy time [RFC2445_ICALENDAR, p. 96].

The To-Do Component

A to-do component represents something that has to be done, be finished, or be prepared at a given point in time. This deadline is given in the DUE property.

A to-do component can be related to other to-do or journal components.

The following example of an to-do item is taken from [RFC2445_ICALENDAR, p. 56]:

```
BEGIN:VTODO
UID:19970901T130000Z-123404@host.com
DTSTAMP:19970901T1300Z
DTSTART:19970415T133000Z
DUE:19970416T045959Z
SUMMARY:1996 Income Tax Preparation
CLASS:CONFIDENTIAL
CATEGORIES:FAMILY,FINANCE
PRIORITY:1
STATUS:NEEDS-ACTION
END:VTODO
```

There is a priority number assigned to a to-do so that several to-dos can be ordered. The status of a to-do can be one of the values "NEEDS-ACTION", "COMPLETED", "IN-PROGRESS", or "CANCELED".

The Journal Component

A journal is supposed to hold a descriptive text. It can be used to store information about past events or to write down just incoming information, e.g. minutes of a telephone call, or to prepare an event by adding some remarks to it.

Journals can be related to each other or to events or to-dos.

The following example shows a journal item, and it is taken from [RFC2445_ICALENDAR, p. 57], but with a shorter description:

```
BEGIN:VJOURNAL
UID:19970901T130000Z-123405@host.com
DTSTAMP:19970901T1300Z
DTSTART;VALUE=DATE:19970317
SUMMARY:Staff meeting minutes
DESCRIPTION:Staff meeting: Participants include Joe\, Lisa
  and Bob. Aurora project plans were reviewed. There is currently
  no budget reserves for this project. Lisa will escalate to
  management. Next meeting on Tuesday.
END:VJOURNAL
```

There are special rules for formatting the content lines of iCalendar chunks, the

maximum length of a line should be 75 octets. The process of inserting a carriage return, a line feed and a whitespace (either blank or a tabulator) is called "folding" and is described in [RFC2445_ICALENDAR, p. 13]. The result of folding can be seen in the example with the value of the DESCRIPTION property. In fact, it could be used for all iCalendar examples shown before, but it helps the human reader to write each property in a new line.

The Free Busy Component

A Free Busy Component can hold a request for free busy times of a specified time range, or it can be the response to that request, or it can be just a publication of free busy times.

This is an example for a request for free busy times from [RFC2445_ICALENDAR, p. 59]:

```
BEGIN:VFREEBUSY
ORGANIZER:MAILTO:jane_doe@host1.com
ATTENDEE:MAILTO:john_public@host2.com
DTSTART:19971015T050000Z
DTEND:19971016T050000Z
DTSTAMP:19970901T083000Z
END:VFREEBUSY
```

The ORGANIZER is the person that is asking the ATTENDEE for his free busy information in the time range given with DTSTART and DTEND. The response would also have the same ORGANIZER and ATTENDEE properties, but additionally some FREEBUSY properties and an URL where to retrieve the most current version of the free busy times.

This is an example of a publication of free busy times [RFC2445_ICALENDAR, p. 60]:

```
BEGIN:VFREEBUSY
ORGANIZER:jsmith@host.com
DTSTART:19980313T141711Z
DTEND:19980410T141711Z
FREEBUSY:19980314T233000Z/19980315T003000Z
FREEBUSY:19980316T153000Z/19980316T163000Z
FREEBUSY:19980318T030000Z/19980318T040000Z
URL:http://www.host.com/calendar/busytime/jsmith.ifb
END:VFREEBUSY
```

In this case, the ORGANIZER is the owner of the calender whose free busy time of the given time range is published.

The Time Zone Component

The time zone component is used to define time zones with their special

arrangements about summer time (DAYLIGHT) and winter time (STANDARD) and their time difference from the **Coordinated Universal Time** (UTC).

This is not used very often, so the reader is just referred to [RFC2445_ICALENDAR, p. 66] to see an example.

The Alarm Component

An alarm only appears as part of an event or to-do component and describes what action should be taken at which time to remind the user about the event or to-do.

This example is from [RFC2445_ICALENDAR, p. 72]

```
BEGIN:VALARM
TRIGGER;VALUE=DATE-TIME:19970317T133000Z
REPEAT:4
DURATION:PT15M
ACTION:AUDIO
ATTACH;FMTTYPE=audio/basic:ftp://host.com/pub/sounds/bell-01.aud
END:VALARM
```

The TRIGGER could also contain a relative time definition to the start or end of the associated component. It defines when the alarm should go off. It can be repeated several times with a given duration of silence in between. The ACTION can have one of the values "AUDIO", "DISPLAY", "EMAIL" or "PROCEDURE". According to the ACTION value, an attached audio file referenced by the ATTACH tag is played, or a message given in the DESCRIPTION field is displayed, or an email is sent, which is assembled from the DESCRIPTION as the body, the SUMMARY as the subject, and the ATTENDEE as the recipient. The PROCEDURE action executes the file that is referenced in the ATTACH tag, but this can be dangerous in an open system due to viruses or hacker attacks.

Recurrence

There can be recurring events, to-dos and journals. Recurrences can also be used to define the summer and winter time of a time zone. There can be exceptions from recurrences, and for free busy times recurrences are split down to the single events.

The recurrence rule has a powerful syntax to produce any recurrence that could be required. It is described in all details in [RFC2445_ICALENDAR, pp. 117-125]. Normally a start date is given, and then there is a rule how often (COUNT) or till when (UNTIL) the event takes place with a given frequency (FREQ). Just some examples can be mentioned here: This example from [RFC2445_ICALENDAR, p.

118] defines that the event would recur for 10 days starting on the given start date:

```
DTSTART;TZID=US-Eastern:19970902T090000
RRULE:FREQ=DAILY;COUNT=10
```

Just one other example, from [RFC2445_ICALENDAR, p. 120], shows an event that occurs on the first friday of each month for 10 times.

```
DTSTART;TZID=US-Eastern:19970905T090000
RRULE:FREQ=MONTHLY;COUNT=10;BYDAY=1FR
```

3.3.2 iTIP

The **iCalendar Transport-Independent Interoperability Protocol** (iTIP) [RFC2446_ITIP] is the scheduling protocol that allows to exchange messages to organise meetings. The messages hold events in iCalendar format. iTIP is not bound to a specific transport protocol. iTIP just defines the application protocol for scheduling meetings and working with Internet calendaring.

All the properties are already defined in iCalendar, but iTIP now gives them a meaning and prescribes semantics for performing Internet calendaring.

iTIP has rules for events, to-dos, journals and freebusy times, which are all components of a calendar item defined in iCalendar. In the following paragraphs the iTIP semantics only for events and freebusy will be explained in more detail.

Organising Events

There are several types of messages that can be sent between Internet calendaring participants. The message types are: PUBLISH, REQUEST, REPLY, REFRESH, ADD, CANCEL, COUNTER and DECLINECOUNTER.

There are two roles defined by iTIP, Organiser and Attendee. Both of them are allowed to send some types of messages in specific situations. There are a lot of described situations, and this is not the right place to go through all of them.

The following table shows a simple example of organising a group event. The table was taken from [RFC2446_ITIP, p. 64].

Action	"Organiser"	"Attendee"
Initiate a meeting request	"A" sends a REQUEST message to "B", "C", and "D"	
Accept the meeting request		"B" sends a REPLY message to "A" with its ATTENDEE "partstat" parameter set to "accepted"
Decline the meeting request		"C" sends a REPLY message to "A" with its ATTENDEE "partstat" parameter set to "declined"
Tentatively accept the meeting request		"D" sends a REPLY message to "A" with its ATTENDEE "partstat" parameter set to "tentative"
Confirm meeting status with attendees	"A" sends a REQUEST message to "B" and "D" with updated information.	

table 3.2 Messages involved in organising an event

Just to show one example of such a message (taken from [RFC2446_ITIP, pp. 64+65]):

```
BEGIN:VCALENDAR
PRODID:-//ACME/DesktopCalendar//EN
METHOD:REQUEST
VERSION:2.0
BEGIN:VEVENT
ORGANIZER:Mailto:A@example.com
ATTENDEE;ROLE=CHAIR;PARTSTAT=ACCEPTED;CN=BIG A:Mailto:A@example.com
ATTENDEE;RSVP=TRUE;TYPE=INDIVIDUAL;CN=B:Mailto:B@example.com
ATTENDEE;RSVP=TRUE;TYPE=INDIVIDUAL;CN=C:Mailto:C@example.com
ATTENDEE;RSVP=TRUE;TYPE=INDIVIDUAL;CN=Hal:Mailto:D@example.com
ATTENDEE;RSVP=FALSE;TYPE=ROOM:conf_Big@example.com
ATTENDEE;ROLE=NON-PARTICIPANT;RSVP=FALSE:Mailto:E@example.com
DTSTAMP:19970611T190000Z
DTSTART:19970701T200000Z
DTEND:19970701T2000000Z
SUMMARY:Conference
UID:calsrv.example.com-873970198738777@example.com
SEQUENCE:0
STATUS:CONFIRMED
END:VEVENT
END:VCALENDAR
```

There are roles assigned to attendees. In this example, "A" is supposed to be the host of the meeting, "D" is not invited, but he should know about the meeting. If there is the RSVP (from French: répondez s'il vous plaît) parameter set to TRUE, then the organiser expects the attendee to send a reply. There is a room booked. E is not invited, but should know about the meeting.

Other more complicated situations are that an attendee proposes another date or time using a COUNTER message, and that the organiser replies either with a

DECLINEDCOUNTER to reject it or sends a new REQUEST to all attendees.

The way to delegate an event is described in [RFC2446_ITIP, p. 68]: The delegator sends a REPLY to the organiser, with his attendee property parameters filled with the status "delegated" and a value for "delegated-to" which holds the address of the delegate. There is also an additional attendee, the delegate, inserted in the reply, and his "delegated-from" parameter refers to the delegator. The delegator also sends a copy of the original REQUEST with the delegation information to the delegate. The delegate can react to this REQUEST as usual, i.e. decline, accept, or set to tentative.

Even situations how to replace an organiser are described in [RFC2446_ITIP, p. 75]. This can be necessary for example when the organiser has left the company, and someone needs to be responsible to apply changes for the event.

Another problem addressed by iTIP is the sequencing of messages [RFC2446_ITIP, p. 11]. Sequencing is important because every recipient of any message must know which one is the most current description of the event. The **Unique Identifier** (UID) helps to see which message belongs to which event [RFC2445_ICALENDAR, p. 111]. One event and all the following messages concerning that event have the same UID, but different SEQUENCE numbers. The SEQUENCE number is incremented with every ADD and CANCEL, and also, when information of the event is changed, with PUBLISH and REQUEST, but not with a delegating request [RFC2446_ITIP, p. 10]. A special case are recurring components that are split up: A specific instance of a recurring event is identified by the same UID as the recurring event, but has a special RECURRENCE-ID property which holds the specific date and time of the specific event [RFC2445_ICALENDAR, p. 107]. At last, the sequence of messages with the same identification can be discovered by comparing the time stamp which is given in DTSTAMP.

Exchanging Free Busy Times

Exchanging free busy times is only based on 3 methods [RFC2445_ICALENDAR, p. 31 ff.]: A calendar user can REQUEST free busy times of a given time range from another user, and this user can REPLY. Or a calendar user can just

PUBLISH his free busy times without being asked for it.

Examples of a request and a reply dealing with free busy times were given already above. It just should be mentioned that it is possible to specify several attendees in a request in order to get replies from all required people.

3.3.3 iMIP

The **iCalendar Message-Based Interoperability Protocol** (iMIP) is a transport protocol for Internet calendaring messages. It uses email standards for the transport. It is described in [RFC2447_IMIP].

The messages are wrapped in MIME encoded emails. The "content-type" value is "text/calendar". This is a very basic approach, and it is recommended to use encryption and authentication e.g. with the "Security Multiparts for MIME" [RFC1847_SMIME].

3.3.4 CAP

The **Calendar Access Protocol** (CAP) is still work in progress, and it is described in [DRAFT_IETF_CALSCH_CAP]. CAP extends iTIP by some methods that are necessary for realtime operation on a calendar service. The iTIP workflows are not changed [DRAFT_IETF_CALSCH_CAP, p. 18]. CAP sends MIME encapsulated commands, and uses the **Blocks eXtensible eXchange Protocol** (BEEP) as the transport and authentication protocol [DRAFT_IETF_CALSCH_CAP, p. 22]. BEEP is described in [RFC3080_BEEP]. It is based on TCP, and in contrast to the Internet mail protocols, it provides realtime operation.

There can be queries to search a calendar store (for examples see [DRAFT_IETF_CALSCH_CAP, p. 33 ff.]), and all other necessary commands to administrate the calendar service and the remote calendar store.

This standard is still not ready, and some bits are missing. But it will become really important for standardised calendar servers.

3.4 Other Standards for Calendaring

Because the Internet calendaring standards are not all confirmed yet and some solutions were created already before the existence of any Internet calendaring standard, there are some strange standards and ideas how to implement group calendaring.

The following solutions provide some kind of access to calendaring information in order to being able to organise a meeting.

3.4.1 MAPI

The **Messaging Application Programming Interface** (MAPI) is the Microsoft solution for applications which need to "exchange or store information in a particular format" [MICROSOFT_MAPI_F]. It consists of two parts, a messaging architecture and a client interface component [MICROSOFT_MAPI_G]. The messaging architecture defines the interactions between applications and messaging systems. The client interface component defines an interface for clients so that they can use the messaging systems through a standardised interface.

Furthermore, Microsoft uses a proprietary format for encoding MAPI messages into a serial data stream [MICROSOFT_TNEF]. It is called **Transport-Neutral Encapsulation Format** (TNEF).

Service providers are pieces of software that can be accessed by an application through the standard client interface in order to use the messaging system that the service provider is able to use. The service provider translates the general MAPI requests und functions to the specific handling of the message store, the directory and the message transport. In the rest of this text, these service providers are called connectors, because some companies call their products like this and it illustrates the functionality of this kind of software.

For example, Outlook supports MAPI service providers. That means, that any company can develop a MAPI compatible service provider, and install it with Outlook. The service provider of this company would then use the messaging system of the company to manage all data that is processed with Outlook.

An idea often used with this approach is to save calendaring data in IMAP folders. This allows to save calendaring data wrapped in an email in one IMAP folder, and all content of one IMAP folder represents a persons calendar. The calendar server is responsible for the group calendaring, and it is able to read and modify the calendar data saved in the IMAP folders.

3.4.2 FTP, HTTP, HTTPS, WebDAV

The following approach only provides very poor group calendaring: It just allows to publish and subscribe free busy times. The disadvantage is that no details of the

events that make the person busy is visible to anyone, and that everyone can only change his own free busy times. The advantage of the "only free busy times" approach is that no calendaring server is required.

Free busy times can be considered as shared information. You don't need a server, but just a shared workspace where you and your colleagues have write and read access to.

This shared workspace was first provided by a combination of the **File Transfer Protocol** (FTP) and the **Hyper Text Transfer Protocol** (HTTP) to access the directory with the free busy times on the server. The user saves his free busy times via FTP on the server whenever his calendar changes, and other users can subscribe to his free busy times and retrieve his free busy times via HTTP from the server. That helps them to organise a meeting because they can see whether the user would be available to attend the meeting or not.

FTP is not a suitable protocol concerning security constraints, because it has no encrypted password or data transmission, and a third person is able to listen to all traffic and can understand all data. HTTPS is the improved HTTP with extended security features. HTTP also provides the transfer of files in both directions, so people thought about extending HTTP with additional functionality so that it can be used for all kinds of distributed authoring and versioning. This new protocol, based on HTTP, is called **Web-based Distributed Authoring and Versioning** (WebDAV). It is defined in [RFC2518_WEBDAV] as an "extension to the HTTP/1.1 protocol that allows clients to perform remote web content authoring operations".

3.5 Synchronisation

Data synchronisation can be defined as "the process of making two sets of data look identical" [SYNCML_WP, p. 3].

There is also a good definition of the term **synchronisation protocol**:

"A data synchronization protocol defines the workflow for communication during a data synchronization session when the mobile device is connected to the network. The protocol must support naming and identification of records, common protocol commands to synchronize local and network data, and it can support identification and resolution of synchronization conflicts." [SYNCML_WP, p. 3]

At the moment there are a lot of different proprietary protocols for data

synchronisation between PDAs, mobile phones, personal computers, and servers. That makes it hard to develop applications that process data that should be available on different devices: It needs to be ensured the application can communicate and synchronise with the whole number of different existing protocols. The creators of such a proprietary protocol are often the providers of an operating system that is designed for mobile devices. They also have the problem that they want their products to be usable with as much applications as possible. One step to encourage application developers to support their protocol is to provide a synchronisation software for several platforms and operating systems, which is able to use plugins from the application developers. These plugins are also called **conduits**. They are able to access the application's data and to associate the correct data fields of the application with those of the mobile device's software.

But it is still an expensive task to provide a special conduit for each of a big number of existing products.

3.5.1 Initiatives and Alliances

In February 2000, the **SyncML** initiative was founded by some important companies that are involved in the market of wireless devices. SyncML stands for **Synchronization Markup Language**. The initiative aims to define a specification that allows devices to synchronise their data although they store it in different formats and are based on different software platforms. See more about technical details of this specification below.

In November 2000, already more than 500 companies from the same sector [SYNCML_SUPPORT] supported this initiative. A SyncML C Reference Toolkit was published under an open source licence in August 2002 [SYNCML_TOOLKIT]. There were 99 SyncML compliant products in September 2002 [SYNCML_PRODUCTS]. One problem was that many important companies joined this initiative, but Microsoft did not [STEMBERGER, p. 4].

In June 2002 the **Open Mobile Alliance** (OMA) was formed [OMA]. Its members are the SyncML initiative and other initiatives which have their own synchronisation protocol standards. Besides other 200 companies, even Microsoft

joined this alliance [MCCARTHY]. The goal of this alliance is to create open standards and specifications for mobile devices that allow interoperability between devices from different providers and based on different platforms. Hopefully they will also create good standards for synchronisation.

3.5.2 SyncML

This chapter should just give a short introduction to the SyncML specifications.

They are designed to be platform independent and transport independent. The connection can be wired or wireless. Examples for supported transport protocols are HTTP, SMTP, the **Wireless Session Protocol** (WSP), and the **Object Exchange protocol** (OBEX) [STEMBERGER, p. 2]. WSP belongs to the **Wireless Application Protocol** (WAP), and both protocols are supervised by the Open Mobile Alliance. OBEX was created by the **InfraRed Data Association** (IrDA) and is a session protocol that is often used with Bluetooth.

There are two SyncML specifications: The **SyncML Device Management** specification helps e.g. to remotely update and configure the wireless devices in the company. In this text, the main focus is on the **SyncML Data Synchronisation** specification, which consists of two protocols and a **Document Type Definition** (DTD) for the XML compliant **Synchronisation Markup Language**. The two protocols are the SyncML Sync Protocol and the SyncML Representation Protocol. The Sync Protocol describes the ways data can be synchronised between a client and a server, while the Representation Protocol defines the semantics of the individual commands and their effects.
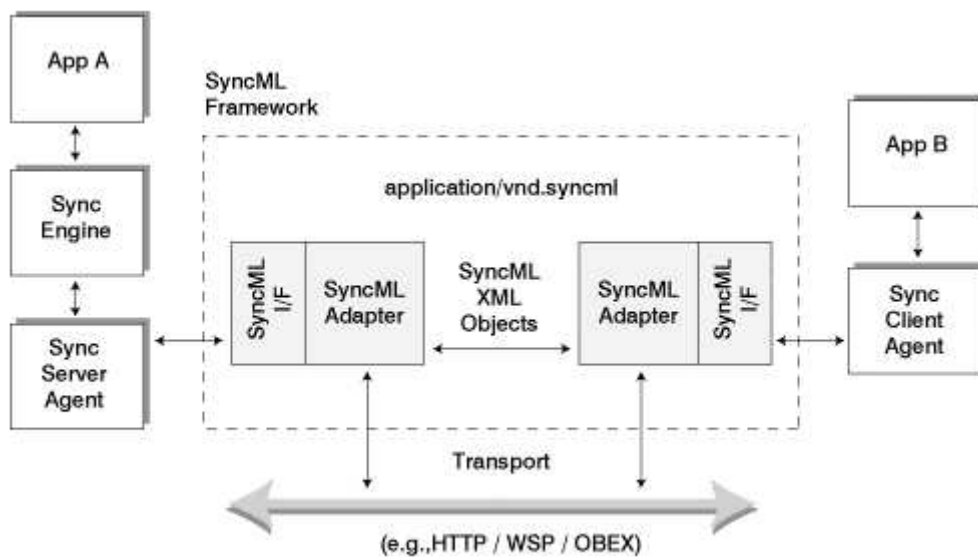
figure 3.3 SyncML Framework

The Representation Protocol describes the SyncML Framework, which is shown in fig. 7.3 [SYNCML_SYNC, p. 7]. The Sync Protocol defines the behaviour and structure of the sync engine, the server agent, and the client agent, which are all outside the framework.

The SyncML Sync Protocol

There are 7 different types of synchronisation described in [SYNCML_SYNC, pp. 8+9]. The most common type is the **two-way sync**, which only transfers the changed and new data items between the sync client and the sync server in both directions. The **slow sync** also synchronises in both directions, but transfers all data. The "**one way sync** from client only" only sends new data from the client to the server. There is also the reverse situation that the client just gets modified data from the server. Another type of synchronisation is a **refresh**, which can be used either from the client or from the server only, and means that all old data is replaced with a complete set of data from the other device. The last type, the **server alerted sync**, gives the server the opportunity to ask the client for a synchronisation.

For all types of synchronisation, it is necessary that the involved devices know about the history of their data, i.e. the **change log information**, and the history of synchronisations, which is represented with the **sync anchors**. If a device is synchronised with several other devices, it needs to keep the records of history for

each of the other devices [SYNCML_SYNC, p. 10].

[SYNCML_SYNC, p. 27] also describes the sync initialisation that is required before every type of synchronisation. This initialisation both helps to detect the capabilities of each device and to authorise the access to the data that should be synchronised.

For more information and a good introduction to the SyncML Sync Protocol, see [PABLA] or [BUCHMANN].

4 MARKET STUDY OF EXISTING GROUPWARE APPLICATIONS

The main focus of this market study is the support of group calendaring. Synchronisation with an offline client and a PDA device is also taken in account. Some assessment and comparison of the products can be based on common requirements, but the final ordering of the solutions is up to those who know the requirements of their own environment.

4.1 Analysis of products and projects

This chapter describes the way used to analyse the existing solutions for groupware needs.

4.1.1 How to find solutions

At first there were some proposals from users who had come across some possible solutions. Other names of solutions could be found by searching on the Internet for "groupware". Another way to find further products was to read the documentation of a product, especially the references to competing products and how to import data. The discussion on the mailing lists of open source projects also gave the names of some products: People required that the coming open source product should have similar functionality to an existing product or should provide the import of data from that existing product. So while investigating the known solutions, some further solutions could be found.

4.1.2 Investigation of a product

The best way to compare the different products is to collect a group of attributes based on a list of requirements. This list contains company specific requirements as well as common requirements. Each product should be investigated concerning these attributes, so to allow the comparison of each product with the others.

4.1.3 Assessment of a product

Because the attributes are related to the companies specific requirements, it is possible to sort the found solutions using the attributes and the priority of the related requirement. So if a solution fails a requirement with the highest priority, it is marked as unsuitable. But the better a solution meets high priority requirements, it is more and more suitable. For the people who have to decide about the topic,

there is also a short summary of the pros and cons, a conclusion based on that, and finally a standardised decision statement by which the products can be grouped.

4.1.4 Common aspects of a product

Some questions should be asked when investigating any kind of product:

- Are there enough other users? What are their experiences?
- Is the providing company stable enough to support the product over a long period of time?
- Can the product be configured so that it meets the customer's specific requirements?

4.1.5 Special aspects of open source projects

There are some points in assessing a product that are different for open source projects. There will be no agreement that will ensure support like a commercial product would provide. These are the topics that have to be investigated:

- Looking back: How long does the project exist? What are the achievements of the project? Was there enough progress in relation to the time span the project exists?
- Who is leading the project? Is it a single person or a team of leaders?
- How many and what people are contributing to the project? Are there enough competent and committed people?
- Are users supported by the open source community? How busy is the mailing list? Are there competent answers to problems of the users? How many users are using the product?
- Does the license fit for the required use in the company?
- Is there commercial support by other companies? How are those companies involved in the project? Are employees working with the open source community, or are they working on their own commercial extensions? (Some licenses allow that)
- Looking ahead: Are there visions of new goals? Will future releases still meet the companies' specific requirements?

Answers to these questions can be found by visiting the projects' websites. Archives of the mailing lists provide information about past discussions. It also

helps to look at the roadmap of the project to see plans for the future. If the project is hosted at a website that provides support for open software development, there is a short standardised description of the project. Some examples for such providers are:

- Sourceforge.net (http://sourceforge.net) is a free service provided by VA Software.
- Savannah (http://savannah.gnu.org) hosts both GNU and non-GNU free software projects, and belongs to the GNU Project.
- BerliOS (http://www.berlios.de/index.php.en) is run by FOKUS, the Fraunhofer Institute for Open Communication Systems.

4.1.6 List of Attributes

The following table shows the list of attributes that was collected for this work. It is influenced by the list of requirements. In many cases, the investigation of some attributes already made clear that the product would not be suitable for OM, because a high priority requirement was not met. For there was not too much time for the project, no further investigation was conducted in those products. That is the reason why not all products are investigated concerning all attributes.

The attributes and a short description for each of them are given in the following table:

name of attribute	short description
id	Identification string
category	association to a specific category (see possible values below)
short	if the category is not exactly describing the product, some more details can be given here
title	name of the product
version	the investigated version
Internet addresses	the website of the product; sometimes other Internet links to information concerning the product
download address	the Internet link to a webpage where a test or full version of the product is provided
documentation	a link to a webpage containing help for users and description of the product, or a description where to find documentation in the downloaded version
installation help	link to a webpage containing help for administrators, or a description where to find documentation in the downloaded version
installation hints	If the program was installed for testing on the OM Standard

	Linux Server, here is a short protocol of what had to be done and how arising problems could be solved.
demo version	link to an online demo version that allows testing the product without installing it; the public demo user login name and the password are also provided.
client	short description of the requirements of the client
server	short description of the requirements of the server part of the solution
licence	Is the product under an open source licence, or is it commercial?
costs	The prices of the product given on the webpage of the producer; What other costs of required products are necessary?
marketing statement	Just one sentence that is on the webpage of the producer and helps to see what goal the company has for this product.
calendar	The functionality of the calendar that belongs to the product is described here in more detail. Some products that got a special investigation have the subtopics "invitation", "private", "resource", and/or "repeating". These give information about how other people can be invited, if there is the possibility to create private appointments, how the booking of resources is integrated into the calendar, and how repeating appointments and meetings can be created.
email	If the product contains an email client, its functionality is described here.
data exchange	This is information about how the product allows synchronisation with PDAs and offline clients. Also export/import functionality with files is mentioned if available.
modules	A short enumeration of other modules besides calendaring and email, that are included in the product.
reliability of support	short description of the company or developers community, and information about the users community
pro	The main advantages of this product. They should be already mentioned in the investigation, but are summarised here.
contra	The main disadvantages of the product which are attributes of the program that don't fulfil a requirement.
conclusion	A short summary why and how the following decision was made
decision	The decision whether or not this product is suitable for OM. There are several defined decisions that group the products again, but now according to the requirements of OM. For the list of possible decisions see below.

table 4.1 Attributes of solutions

4.2 Categorisation of Solutions

This chapter shows how groupware applications can be assigned to categories. Different categories are formed, based on different points of view. In the end, a set of categories is given by which the investigated solutions can be grouped together.

4.2.1 The Solutions and their Communication Architectures

There are several approaches to provide groupware functionality:

- **peer-to-peer model**: The clients communicate directly with each other e.g. via email. There is no central storage or processing of calendaring information. All data is stored locally. One example for this is Ximian Evolution.

- **free/busy server**: This basically provides a central storage of calendaring data. There is no processing of data on the server. Outlook and Mozilla support this basic type of group calendaring.

- **client/server model**: The user can use a client to connect to the server. The server can support clients running on different platforms. Not every client software also provides offline functionality, i.e. not all clients save the data locally.

- **multitier approach**: A web interface is used to get access to the functionality provided by the calendaring server.

Some solutions provide a server that works with both a web interface and one or more clients, e.g. for several platforms. The products that include at least one client software, normally also provide a tool for PDA synchronisation.

4.2.2 The Solutions and their Software Licences

There are a lot of different types of licences under which a software can be developed and released. This text only uses the terms commercial and open source licences. Software with a commercial licence normally is not for free, and the main characteristic is that the sourcecode is not publicly available.

There are many different types of open source licences, but here this term is used for all software which is published for free and includes the source code.

Especially with the client/server approach it is possible to mix commercial solutions with open source projects. The following list is ordered by the degree of dependability on commercial software.

- commercial client and commercial server (e.g. Exchange/Outlook, Lotus Domino/Notes, and lots more)
- commercial client with commercial client connector tool and commercial server running on Linux (e.g. Outlook & Insight Connector & Insight Server)
- open source client with commercial client connector tool and commercial server (e.g. Ximian Evolution & Ximian Connector & Exchange)
- commercial client with commercial client connector tool and open source server running on Linux (e.g. Outlook & Insight Connector & Kolab)
- open source client and open source server running on Linux (e.g. KDE-PIM & Kolab)

But also the solutions that provide a web interface can use different licence models:

- webbased interface and commercial server running on Linux (e.g. BSCW)
- webbased interface and open source server running on Linux (e.g. phpGroupWare, PHProjekt, moreGroupware)
- A service provider can give allow users to create and access their data on a server that is on the Internet. The advantage is that there is no maintenance of a server, but information is carried out of the company and does not stay inside the intranet.

Most commercial solutions are sold on a per user basis, but some of them sell the individual client software. That makes the solution even more expensive if the user wants to access his data with both e.g. a Windows client and a PDA. Some of the commercial solutions have a quite long history and changed from proprietary standards to open standards.

4.2.3 The Categories

The products investigated in this diploma thesis can be assigned to the following categories:

- **Outlook/Linux:** There is a Linux server that is able to provide Outlook with group calendaring
- **webbased & Outlook:** A server running Linux that provides a webbased interface and allows Outlook either to use this server for group calendaring or to import and export calendaring data from and to that server

- **webbased:** A Linux server just providing a webbased interface
- **webbased & client:** A Linux server providing a webbased interface and some proprietary offline client
- **client:** a client for group calendaring
- **sync:** synchronisation tools, either for synchronisation between a server and Outlook or a server and PDAs

The following categories were formed because some products should be mentioned when talking about groupware, although they are not useful for the OM Standard Linux Server:

- **no calendar:** groupware tools without group calendaring support
- **non Linux server:** server requires e.g. Solaris or Windows operating system
- **outdated:** These programs are not supported anymore.
- **service provider:** These companies just provide group calendaring on their website.

4.2.4 Investigated Products ordered by Categories

This is a table about all the products that were investigated during this diploma thesis. For more details about each solution, please refer to the appendix, or see the following subchapters that mention some of the products in more detail.

Outlook/Linux
[BILL WORKGROUP] BILL Workgroup Server; [BYNARI INSIGHT] Bynari Insight Server; [KOLAB] Kolab / Free Software Groupware Project; [CALDERA VOLUTION] Caldera Volution Messaging Server
webbased & Outlook
[ORACLE COLLABORATION] Oracle Collaboration Suite; [SUSE MAIL] Suse Mail Server; [SAMSUNG CONTACT] Samsung Contact; [DAYPOINT] FrontOffice Communications: DayPoint Enterprise; [ECAL] eCal M1; [SCHEDULEONLINE] ScheduleOnline Intranet; [WEBEVENT] WebEvent Team
webbased
[SQWEBMAIL] Courier MTA / SqWebMail; [MOREGROUPWARE] moreGroupware; [BSCW] BSCW; [PHPROJEKT] PHProjekt; [PHPGROUPWARE] phpGroupWare; [TUTOS] Tutos; [TWIG] TWIG; [SKYRIX] SKYRiX Web Groupware; [XENTRIX] intisoft Xentric groupware server; [ERIDU] eridu; [PROIEKTOR] Proiektor; [UW] UW Calendar Information Center; [GROVEWARE] Groveware CAL/RM 2000SE; [CALENDAR INTERNET] Calendar Internet; [GHC] Great Hill Corporation: Calendars for the web; [HTMLCAL] HTMLCAL; [MDAEMON] MDaemon
webbased & client

[TEAMWARE OFFICE] Teamware Office; [NOVELL GROUPWISE] Novell Groupwise; [MEETINGMAKER] Meeting Maker; [SUSE GROUPWARE] SuSE Linux Groupware Server with Lotus Domino; [LOTUS DOMINO] (IBM) Lotus Domino / Notes; [MAILSTUDIO] MailStudio Calendar Server; [CYBERSCHEDULER] Crosswind Cyberscheduler & Synchronize & CyberSync; [BROWNBEAR CALCIUM] BrownBear Calcium Web Calendar Server and TripleSync

client
[OPENOFFICE.ORG] OpenOffice.org Groupware; [MICROSOFT OUTLOOK] Microsoft Outlook; [XIMIAN EVOLUTION] GNOME Ximian Evolution; [MOZILLA] Mozilla Calendar; [KDE PIM] KDE PIM application suite (with KOrganizer); [CHANDLER] Chandler

sync
[AXISSYNC] AxisSync; [PHPORGASYNC] phpOrgaSync; [SYNCBUILDER] SyncBuilder Java Framework

no calendar
[ICQ GROUPWARE] ICQ Groupware; [PHPCOLLAB] phpcollab

non linux server
[MICROSOFT EXCHANGE] Microsoft Exchange; [SUN CALENDAR] Sun Calendar Server; [LIVELINK] Opentext Livelink; [HYPERWAVE] Hyperwave Team Workspace

outdated
[SUN STAROFFICE] Schedule and Schedule Server - StarOffice; [NETSCAPE CALENDAR] Netscape Calendar Server

service provider
[CONTACTOFFICE] ContactOffice; [APPOINTMENTQUEST] Appointment Quest; [APPOINT.NET] appoint.net; [CALENDARS.NET] Calendars.net; [CALENDARSERVER.COM] CalendarServer.com; [HUNTCAL] www.huntcal.com; [E2DO] www.e2do.com; [LOCALENDAR] www.localendar.com; [SMARTCAL] www.smartcal.com; [SUPERCALENDAR] www.supercalendar.com; [YAHOO CALENDAR] www.calendar.yahoo.com

4.3 Description of Solutions

This chapter gives a short description of some group calendaring supporting solutions.

The CAP protocol is not fully specified, so there is yet no basis for calendar servers to be built on the Internet standards. Hopefully, the soon future will bring Internet based calendaring in such a variety as there are Internet message servers and clients at the moment that are all capable of the same Internet protocols, e.g. POP3 and SMTP.

4.3.1 Integrated Client/Server Groupware Solutions

This chapter gives an overview over the history of collaboration software and the existing integrated solutions that are delivered with both a server and a client.

Just to give an impression about the size and the shares of the collaboration market, here is a quotation from [WONG]:

"In 2001, IBM led the $1.6 billion market with a 49 percent share of revenue, followed by Microsoft with 39 percent and Novell with 6 percent, according to market researcher IDC. Microsoft, however, ranked first in the number of customers. Of 210 million users worldwide last year, Microsoft captured 40 percent, followed by IBM with 35 percent and Novell with 16 percent."

The original articles from the market researchers are expensive, so it is not possible to provide the correct context of this figures. It seems that IBM/Lotus is more expensive, but enough companies are willing to pay for it anyway, so that in the end the revenue is bigger than that of Microsoft. But it seems that Microsoft will overtake IBM/Lotus by improving Exchange more and more and by having better user acceptance. For more information about this topic see [WATSON].

IBM and Lotus have the longest experience in the collaboration market with their products Notes and Domino. Lotus introduced Lotus Notes in 1989 [WATSON].

Microsoft published the first version of Exchange in 1996 [MICROSOFT_L]. It replaced the Microsoft Mail Server. Exchange provides both a mail server and groupware functionality, e.g. shared folders that can contain e.g. email, calendars and documents. Outlook is the client for Exchange.

Novell Groupwise cannot be investigated here. It did not provide a Linux server, so it was not interesting for OM anyway.

There are also smaller companies that offer group calendaring software.

One example is MeetingMaker who sell their product with the same name. They are working in this area since 1990. They concentrate on just group calendaring and don't provide any other groupware functionality.

Another example is Fujitsu/Teamware with their product Teamware Office. This is a suite that provides several groupware applications, e.g. calendar, email, and forum.

Lotus Domino/Notes

Lotus Domino and Notes were not installed during this diploma thesis, because it was obvious that these products would be too powerful for OM. They have so

many features and provide all the applications that are already covered by the existing OM solution. They would replace the OM Standard Linux with its preconfigured applications, but it would take a lot of time to prepare Lotus Domino/Notes so that it is as easy to install and maintain as the existing solution.

But nevertheless, the Lotus products at least should be mentioned because they are widely used.

Lotus Notes is the "e-mail, calendaring, group scheduling, Web access and information management client" [BRICHACEK, p. 5]. Lotus Domino is the application server for it.

The approach of Lotus is to use a powerful database that is able to manage all kinds of entities. Examples for such entities are emails, appointments, documents, and much more. The database supports replication and therefore offline or disconnected operation is possible ([LAVINE] and [BORGHOFF, p. 140]. There is the possibility to extend Domino/Notes with customer applications. Domino can work together with the IBM Websphere Application Server to provide web services [BRICHACEK].

One point that needs to be mentioned is that Lotus was able to overcome the own proprietary email and directory standards and to implement Internet standards [JOHNSTON].

Microsoft Exchange/Outlook

This chapter focuses on Outlook, which is the client for Exchange. But this will show the abilities of Exchange as well.

Microsoft Outlook is very well known because Outlook Express, which is a restricted Outlook version with only email client functionality, is given away for free together with the Microsoft Internet Explorer. And Outlook belongs to the Microsoft Office which is used by many companies.

In earlier versions up to Microsoft Outlook 2000, there were 3 modes in which Outlook could be operated. In the current version, Microsoft Outlook 2002, all modes can be used simultaneously. That means one profile can contain several email account types at the same time.

In the "No Email Mode", Outlook just works as a standalone **Personal Information Manager** (PIM) without any email functionality.

There is the "Internet Mail Only" mode, that allows using email services with POP3/IMAP/SMTP, and group Calendaring with the Internet Free/Busy feature. Internet Free/Busy uses the iCalendar standard to manage the free and the busy times of a person. The Internet Free/Busy times are saved and accessible from an HTTP server that runs the Frontpage extensions. These extensions are not only available for Microsoft's webserver, but also for the Apache webserver. For Outlook 2000, the Microsoft Web Publishing Wizard is required, but there is only a supported version for Windows NT 4.0 or Windows 98 [MICROSOFT_FB].

Invitations are sent via email. That means that no realtime calendaring is used.

In the "Corporate Workgroup" mode Outlook has a connection to a groupware server, which normally is Microsoft Exchange. This cooperation is based on MAPI. But there are some other companies that offer a plugin program, which enables Outlook to communicate with another server than Exchange. Outlook provides offline folders, that can be synchronised with the associated folder on the server. A folder can contain emails, a calendar, notes, contacts and tasks (to-do). Permissions can be granted to other users on different levels, e.g. it is possible to permit a colleague to only add items but not to edit existing items. There are also public folders, which are folders without any write or read protection.

Outlook is able to import and export appointments in the iCalendar format, but the export works only with one appointment per file. So this is no real help for exporting all events of e.g. a week. Another type of import and export works with files in the **Comma Separated Values** (CSV) text file format. The problem with this is that a lot of information, e.g. all the recurring events information, gets lost. A recurring event is replaced by one event per instance. There are no unique IDs assigned to the events, so it is difficult to avoid duplicated entries during synchronisation.

Every PDA supports synchronisation of calendar items with Outlook. That means that an Outlook conduit is delivered with the synchronisation software of the PDA.

4.3.2 Groupware Clients

This chapter gives more details about some groupware clients. Only those clients are mentioned, that either seem to establish their own standards which are widely accepted or that are using the existing Internet standards and are expected to

work with the coming Internet standards.

The Microsoft Outlook client was already investigated in detail. A connector software, e.g. Bynari Insight Connector, can be used to integrate Outlook with other servers than Exchange.

The open source community is working on some projects to provide free groupware servers and clients. On the client side, there is work in progress on Mozilla and OpenOffice.org. Mozilla is extended with a calendar application. The people developing OpenOffice.org want to integrate the Mozilla code into their groupware component. OpenOffice.org is the opensource version of StarOffice, which belongs to Sun.

These two projects, Mozilla and OpenOffice.org, are intended to run on both the Linux and the Windows operating systems. The problem is that there is no suitable free server yet available that would provide realtime calendaring.

Ximian Evolution is a client only to be used on Linux systems.

The KDE PIM application suite also requires Linux. There is a project called "Kroupware", they are developing the Kolab server (see below) and want to port the KDE PIM applications to Windows.

Mozilla

The Mozilla project is based on the source code of the Netscape browser. In March 1998 this source code was released by Netscape under an open source license. People outside the company were invited to join the project, to improve the product and to develop the future releases. There are people payed by AOL/Netscape who only work on the Mozilla project, in parallel to the developers working on the Netscape browser releases. Mozilla technology can be used for every other project that fulfills the requirements of the Mozilla or Netscape Public License.

Some detailed information about this deal is described by Eric S. Raymond [RAYMOND, pp. 27-29].

The current Netscape browsers (up from version 6.0) are based on Mozilla technology.

The Netscape Communicator 4.x contained a calendar component, but that was licensed from CS&T (Corporate Software Technologies International Inc., which is

today known as Steltor and was taken over by Oracle). So the calendar component could not brought into the Mozilla project like e.g. the email client.

Because there is no open source Calendar server available, the calendar that is in development at the moment only supports exchange of calendar items via email and free/busy times using an FTP/HTTP or WebDAV/HTTPS server. The invitation is not yet integrated with the creation of appointments.

There is no PDA synchronisation available yet. But at least it appears on the requirements list of the Mozilla calendar.

Ximian Evolution

Ximian is the same company that founded the Gnome project. Ximian is opensource, and it is sometimes called the Outlook for Linux. It can be used together with the Microsoft Exchange server, but then a commercial connector from Ximian is required. It also can be used without any calendaring server, because it supports a peer-to-peer architecture for group calendaring. This is based on the Internet calendaring standards, by communicating via emails consisting of iCalendar messages. Free/Busy times can also be sent via email.

For more information about Ximian Evolution, see [MOELLER].

KDE PIM

The KDE PIM kit also provides some collaboration clients for the Linux users, e.g. an email client (KMail), and a calendaring client (KOrganizer). KPilot is a synchronisation tool for PDAs, and there is work in progress on Kitchensync.

Work on the integration of the several applications is going on, there is the Kaplan project and the temporary Kroupware project.

The Kroupware project was initiated by the German "Bundesamt für Sicherheit in der Informationstechnik" with the goal to find and/or develop an open source group calendaring server and the fitting client software [ERFRAKON, p. 6]. One step to the final solution is to provide a Linux client [ERFRAKON, p. 36]. This project was called Kroupware, but its name will either be replaced by a better name, or the project will be integrated into the existing KDE PIM tools.

Erfrakon also plans to port these applications to Windows in some time in the future [ERFRAKON, p. 37]. That could be possible by using the QT library and Cygwin for Windows.

The solution is not ready for group calendaring at the moment: Peer-to-peer operation is not possible, because the mail and the calendaring client are not integrated yet, and there is no server available. The Kolab server, also from Erfrakon, could fill this gap. It is described below.

4.3.3 Groupware Solutions providing a Web Interface

A lot of groupware servers also give the user the opportunity to access the groupware applications using a standard web browser.

The main advantage of using a web interface is the centralised installation, nothing has to be configured on the workstations. A suitable web browser is always available [KLOECKNER, p. 6]. But there is a problem with alarms or reminders: A web interface is only able to react, but cannot act on its own. A solution for that problem is the use of a Java applet or a routine in Javascript that can poll the server to ask if there is an upcoming event. Another disadvantage when using the solution inside an intranet is that the data is not available from the outside, which would be important e.g. when people are travelling. There are often solutions to synchronise a suitable client software, often Outlook, with the server that provides the web interface.

The big problem of the open source groupware solutions is that synchronisation with an offline client and a PDA device are not yet available. Most of those solutions are based on programs written in the language **PHP Hypertext Preprocessor** (PHP). That means there would be a problem if such a solution would be used in huge enterprises, because PHP was not designed to provide scalable applications, but to provide an easy programming environment for quick development [HULL]. But that should really become a problem when a company with 200 workstations is using all applications of a PHP based groupware solution. In OM, only the calendaring application would be used, and a maximum of 50 workstations is accessing the groupware server at the same time.

A lot of calendar servers that have their own client or support Outlook as client, also provide web interfaces, e.g. Sun Calendar Server, Novell Groupwise, Samsung Contact and Suse Mail Server.

The open source solutions PHProjekt, phpGroupWare and moreGroupware are described below, TWIG and Tutos are not very different from them.

phpGroupWare is an exception from the other open source solutions because it has a really good software architecture. phpGroupWare has the potential to become a real groupware server that does not only provide a web interface. This will be described later in more detail.

The BSCW system is developed by the German National Research Center for Information Technology (GMD). BSCW stands for "Basic Support for Cooperative Work". It is a commercial product, and provides web based group calendaring and other group supporting applications. The GMD also provides free accounts on their Internet server.

One other example of a commercial server in this category is ScheduleOnline. Its advantage is that it provides Outlook and PDA synchronisation, and the server runs on Linux, without requiring a special email server software. Those are the common problems with other commercial solutions, which fail in one or two of the mentioned topics.

PHProjekt

PHProjekt is in its third version, and it seems to be an established webbased groupware application. The leader of this project is Albrecht Günther. The problem is that there are no published goals to improve it more and more, and the ongoing work is not documented. For example, there is the requirement of synchronisation with PDAs posted on the website for months, but it seems that nobody is working on it. There is no import or export functionality of calendar items at all.

There is a product called phpOrgaSync that is able to synchronise Outlook and PHProjekt. It is freeware, that means it is available for free, but the sourcecode is not under an open source licence. So there is no chance to deal with bugs: If it was a commercial software, there would be a contractor that has to ensure the usability of his sold product by providing bug fixes. If it was open source software, there is the chance for everyone in the users community to find and eliminate the bug and publish the bugfix on the Internet. But with freeware, the user has no rights besides being allowed to use the software.

phpGroupWare

This project has not yet reached the 1.0 release stage, but it is already a program that needs to be taken serious. It is led by a team of 5 people. That ensures that

the project will be pushed forward even if not all members of the leading team are working on it. The problem with open source projects is, that a lot of the developers and the team leaders are only working on the project during their leisure time. That means if there is a hard time at work or the family requires more attention, the open source project which is just a hobby will be the first to be neglected. But the propability of the project being stopped is smaller when there are several people in the team, because then responsibility is split up, and others can push the project forward.

The phpGroupWare software has a clean design, which is necessary because the project claims to provide not only a web based groupware solution, but a groupware **Application Programming Interface** (API) that will be usable also for other applications and in the future for other clients.

The phpGroupWare API is clearly divided from the applications, and supports any kind of groupware applications that can take advantage of the functionality provided by the API. Even inside the applications, the code concerning the user interface is clearly seperated from the business layer, and from the storage layer. This modularisation combined with XML RPC or SOAP functionality allows other programs in all kinds of programming languages to request services from the phpGroupWare server.

More information about the phpGroupWare API is given in chapter 5.

The goal is to port the phpGroupWare project to another language than PHP. Especially in environments where only calendaring services without a user interface are required, PHP does not provide the best scalability.

moreGroupware

This project is supported and led by a company called "Morelogs" who wanted to create their own webbased groupware solution. It is not finished yet. The advantage of an open source project being supported by a company is that there is money brought into the project, by giving people time to work on the project during their payed worktime. But the general problem with such a situation is that the communication between the developers inside the company must be public, so that other developers from the outside are able to participate in the work.

4.3.4 Servers that replace Exchange

The integrated solutions as well as the servers providing a web interface were already covered above. This chapter now shows servers that are aiming to simulate an Exchange server that is able to deal with Outlook.

Samsung Contact

Hewlett-Packard developed a product called OpenMail, which included Outlook support and group calendaring. It was bought by Samsung, renamed to Samsung Contact, and is now maintained by them. Samsung is actually using the program in their own offices. The product consists of proprietary server parts and a special MAPI connector for Outlook.

Oracle Collaboration Suite

In July 2002, Oracle also published their plans of entering the collaboration market. They bought the company Steltor with their product CorporateTime which is a group calendaring solution. Oracle is now working on the Oracle Collaboration Suite which will be based on Oracle's application server and database server. The goal of Oracle is to support Outlook so that people don't need to change the client software they are used to.

Open Source Exchange Replacements

The open source community always has the goal to replace commercial products by free products that can be improved and be used by everyone. For there is no competition but cooperation between several open source projects, it makes sense to find standards so that the products can exchange data and provide functionality to each other.

One of the first solutions to configure Linux to work with Outlook and provide Email and Calendaring was developed by Kevin Erickson ([ERICKSON] and [JOHNSON_MEAD]). He only used open source software. One disadvantage of his proposal was that he did not provide a MAPI connector tool, so he only could realise Free/Busy group calendaring and email server functionality.

Following Ericksons case study, some companies built their own packages of existing open source programs and published the scripts and installation routines either under an open source or commercial licence.

Bynari Insight

The company Bynari had the goal to develop a MAPI connector that uses standards so that their Bynari Insight Server could provide also shared folders etc. to Outlook [ADELSTEIN]. This connector is called Bynari Insight Connector. It is a MAPI compliant software and enables Outlook to communicate with the server that provides the groupware functionality. The Bynari Insight Server is a commercial server that consists mainly of open source parts, e.g. the email server and directory server and so on. But the linking parts are not published.

Bill Workgroup

The company Neuberger & Hughes also follows the same idea with its product Bill Workgroup Server. The difference is that they develop the server under an open source licence, and only sell the MAPI connector software.

Kolab

Another project in this category is the Kolab project, they also want to build a Linux based server that provides Microsoft Outlook with the same services and functionality as the Microsoft Exchange server. The difference is that Kolab is open source and should also work with a KDE client. But the Kolab server also requires the commercial Bynari Insight Connector when Outlook should be used as client. The disadvantage of Kolab is that it still uses TNEF as the format for saving messages and calendar items, because the Bynari Insight Connector does not convert the data to a standard format, e.g. iCalendar. At least, TNEF is well documented and therefore it is possible to write Linux clients for the Kolab server. The Kroupware project group modified the existing KDE PIM applications to be able to use the Kolab server.

4.4 Synchronisation

It was already mentioned with some programs' descriptions how they support synchronisation, or if they use special tools to maintain several sets of data on several clients. The best solution would allow synchronisation of the server with the PDA and some kind of offline client, that also works in disconnected mode.

Generally speaking, there are three ways to synchronise PDAs:

- Synchronisation of the PDA with the client program.
- Synchronisation of the PDA directly with the calendar server

- An extra synchronisation server manages all synchronisation between the different client tools of the user: PDA, web interface, and one or several workstation clients.

Microsoft and Palm have 2 different standards of synchronisation. Devices running Windows CE use the Microsoft ActiveSync technology, and Devices with Palm OS use the Palm HotSync technology. So it is required that a calendaring client or server provides one conduit for every PDA operating system, and it becomes even more complicated taking the other mobile devices, e.g. mobile phones, in consideration.

Palm provides Palm Network Hotsync that allows synchronisation with a remote server that can be accessed over a network [PALM_NETSYNC].

Microsoft offers the Mobile Information Server [MICROSOFT_MI], that will be integrated in the next major upgrade of Exchange [CAIN].

There is also a number of synchronisation servers, but this is again no place for investigating them.

The reader is referred to [BUCHMANN] to learn about the Sync4j project. This project provides frameworks to develop SyncML clients and servers, and is based on Java.

5 OM AND GROUPWARE

This chapter now deals specifically with the needs of Operation Mobilisation. The current situation is investigated, the requirements of OM are collected, and the products discovered in the previous chapter are ordered again, now concerning the requirements of OM. 5 choices for suitable solutions are given, and in the end one of these options is investigated more, and a prototype of the modifications to this program is described.

5.1 Analysis of Requirements in OM concerning Groupware

5.1.1 Current Situation

At the moment, OM are still working on standardising the use and the structure of the computer and network systems in their offices. Some procedures are not computer supported at all.

The OM Standard Linux Server, which is used by all offices except the ships, provides email functionality, access to the web, a backup system, shared diskspace on the server, and some other helpful services.

Sharing documents is possible by using the shared diskspace on the server. But there is no version control or log of modifications.

Mailing lists are mainly used globally, so a solution based on the OM Standard Linux Server is not possible, because the servers are not always connected to the Internet. At the moment, the free solution of an Internet mailing list provider is used.

There is a worldwide intranet for the whole organisation, and each office has its own local intranet as well. The local intranet is not standardised.

In the Carlisle office, the intranet provides a readable resource calendar, e.g. for laptops, mobile phones and other resources. The calendars of some important persons in the office are also available with read access. There are persons who are responsible to manage these calendars. There are also printouts of calendars for every month with the birthdays of the staff in that office and details for events during the month.

The offices on the ships use a Microsoft Windows server instead of the OM Standard Linux Server, and the people on the ships got very used to have

Microsoft Exchange for scheduling. They don't want to switch to the OM Standard Linux Server because there is no group scheduling available at the moment. But it would be better for OM to have a centralised development team of the server system, and only one system administrator in each office. The goal is to have the same linux server in each office. That allows a standardised training of system administrators and helps the Carlisle Linux team to support them when there is trouble with the system. At the moment, the ships need their own specialists for the Microsoft solution, while the servers of the other offices can be maintained by normal system administrators without special development or adaption skills.

So both the ships and the Carlisle office have their own solutions for calendaring. But there is no OM standard for group calendaring, and offices without the necessary skilled staff don't have group calendaring at all.

5.1.2 The Requirements

The list of requirements can be found in the appendix. The requirements were collected via email from the people in the offices that are responsible for installing and administrating the local computer network, and other developers working for OM. That is the origin of most of the requirements. Some other requirements came from the managers of the Linux team who are responsible for the development of the OM Standard Linux Server.

It turned out that some requirements did not belong to the task of providing OM with a groupware solution. The definition of this solution had to be made, and finally only a groupware calendar was left. All email concerning requirements were postponed, the only requirement to the calendaring solution was that it has to support the existing mail functionality of the OM Standard Linux Server. The requirement of a centralized email address book was already met by an LDAP solution.

The main requirements were long term reliability of the solution, and the price of it. For OM is a charity organisation that only exists from donations, they have to ensure that the given money is not wasted but used in a sensible way.

A clear requirement was that the solution had to be integrated in the OM Standard Linux Server. The requirement for the client side was to support Microsoft Windows because this is the OM standard operating system for the workstations.

5.2 Matching the Requirements with the Market Study

The products were ordered according to the requirements list. So products that did not support a Linux server or did not provide group calendaring at all were put to the end of the list.

There were the following categories by which all solutions were grouped:

- **suitable:** This product fulfils the requirements of OM

- **could be suitable / depends on other solution:** This product could be fine, but it requires another product which is not available yet. The two programs together would fulfil the OM requirements.

- **promising, but still under development:** The product's requirements list would probably fit the OM requirements, but the work on it still is not finished. There are already non stable testversions. There is the hope that the products in this category soon will be ready for real use.

- **need to be considered concerning the costs:** These products would fit the requirements of OM. The selling companies were asked for discounts, and OM have to decide if it makes sense for them to pay for one of these solutions.

- **even with discounts too expensive for OM:** These products would fit the requirements of OM. They first were in the category "need to be considered concerning the costs", and after the next stage (asking for discounts) they joined this category. Even with discounts these products were still too expensive for the budget of OM.

- **experimental:** The difference to the "promising" category is that these products are far away from a stable release, and it is unkown how and when they will be usable.

- **missing synchronisation support** means that there is no way to synchronise with an offline client, and/or no way to synchronise with a PDA directly or with a client that can synchronise with a PDA.

- **outdated:** The product is not available anymore.

- **not enough reliability of support:** Either the developer community is too small, or the product is free but not under an open source licence

- **not suitable for OM:** that means that either calendar functionality is missing, or the Linux server is not supported, or the Windows workstations

are not supported.

- **too difficult to integrate with current OM solution:** The products in this group provide all functionality that is expected from a mail and groupware server, and they would replace the OM Standard Linux Server. But then all the preconfiguration need to be developed again, so that the solution would have the same easy installation and maintenance than the OM Standard Linux Server. That was not an option.

5.3 Analysis Result: Ordering by Decision

This is the list of the same products listed in the end of chapter 4, but now the products are ordered by the decision which was met based on the requirements of OM. For more details about the solutions, please refer to the appendix.

suitable
none
could be suitable / depends on other solution
[PHPGROUPWARE] phpGroupWare; [MICROSOFT OUTLOOK] Microsoft Outlook
promising, but still under development
[KOLAB] Kolab / Free Software Groupware Project; [AXISSYNC] AxisSync
need to be considered concerning the costs
[SAMSUNG CONTACT] Samsung Contact; [SCHEDULEONLINE] ScheduleOnline Intranet
even with discounts too expensive for OM
[WEBEVENT] WebEvent Team
missing synchronisation support
[SQWEBMAIL] Courier MTA / SqWebMail; [MOREGROUPWARE] moreGroupware; [BSCW] BSCW; [PHPROJEKT] PHProjekt; [TUTOS] Tutos; [TWIG] TWIG; [TEAMWARE OFFICE] Teamware Office; [MOZILLA] Mozilla Calendar; [SKYRIX] SKYRiX Web Groupware; [XENTRIX] intisoft Xentric groupware server; [MAILSTUDIO] MailStudio Calendar Server; [CYBERSCHEDULER] Crosswind Cyberscheduler & Synchronize & CyberSync; [GROVEWARE] Groveware CAL/RM 2000SE; [GHC] Great Hill Corporation: Calendars for the web; [HTMLCAL] HTMLCAL
not enough reliability of support
[BILL WORKGROUP] BILL Workgroup Server; [PHPORGASYNC] phpOrgaSync; [SYNCBUILDER] SyncBuilder Java Framework; [ERIDU] eridu; [PROIEKTOR] Proiektor
experimental
[ORACLE COLLABORATION] Oracle Collaboration Suite; [OPENOFFICE.ORG] OpenOffice.org Groupware; [UW] UW Calendar Information Center; [CHANDLER] Chandler
outdated
[SUN STAROFFICE] Schedule and Schedule Server - StarOffice; [NETSCAPE CALENDAR] Netscape Calendar Server

not suitable: no linux server support
[NOVELL GROUPWISE] Novell Groupwise; [MICROSOFT EXCHANGE] Microsoft Exchange; [SUN CALENDAR] Sun Calendar Server; [LIVELINK] Opentext Livelink; [HYPERWAVE] Hyperwave Team Workspace; [CONTACTOFFICE] ContactOffice; [CALDERA VOLUTION] Caldera Volution Messaging Server; [APPOINTMENTQUEST] Appointment Quest; [APPOINT.NET] appoint.net; [CALENDAR INTERNET] Calendar Internet; [CALENDARS.NET] Calendars.net; [CALENDARSERVER.COM] CalendarServer.com; [DAYPOINT] FrontOffice Communications: DayPoint Enterprise; [ECAL] eCal M1; [HUNTCAL] www.huntcal.com; [E2DO] www.e2do.com; [LOCALENDAR] www.localendar.com; [SMARTCAL] www.smartcal.com; [SUPERCALENDAR] www.supercalendar.com; [YAHOO CALENDAR] www.calendar.yahoo.com; [MDAEMON] MDaemon

not suitable: no windows workstation support
[XIMIAN EVOLUTION] GNOME Ximian Evolution; [KDE PIM] KDE PIM application suite (with KOrganizer)

not suitable: no group calendar included
[ICQ GROUPWARE] ICQ Groupware; [PHPCOLLAB] phpcollab; [BROWNBEAR CALCIUM] BrownBear Calcium Web Calendar Server and TripleSync

too difficult to integrate with current OM solution
[BYNARI INSIGHT] Bynari Insight Server; [MEETINGMAKER] Meeting Maker; [SUSE MAIL] Suse Mail Server; [SUSE GROUPWARE] SuSE Linux Groupware Server with Lotus Domino; [LOTUS DOMINO] (IBM) Lotus Domino / Notes

5.4 The Options for OM

There are several possibilities:

Option 1: phpGroupWare & AxisSync

At the moment, phpGroupware has no support for data exchange and synchronisation with an offline client. But in contrast to the other free web based solutions, it is not too hard to add new functionality to the program. There are already import and export functions for address book items. It seems to be one of the most promising projects that will have a future. So it is useful to invest some effort into the project.

The functionality that could be added and was prototyped for this diploma thesis is the import and export of calendaring data from and to files in the **Comma Seperated Values** (CSV) format, that can be imported and exported by Outlook as well.

A prototype for Palm synchronisation also exists with the tool AxisSync, but it is not stable yet. The export of calendaring data from phpGroupware to a Palm device is working already, but the other direction is still missing. But work is going on with the development of this tool, and in the end of October there was a new release.

The solution is not ready at the moment, but approximately in autumn 2003, this product will be the first choice of free web based group calendaring solutions. But the synchronisation with an offline client won't be very comfortable.

Option 2: Kolab

This server is free, but the Bynari Insight Connector need to be bought. But it provides the full Outlook functionality, and therefore also PDA synchronisation is available.

At the moment the Kolab server is in the testing and bug-fixing phase, so it is not ready yet. It should be stable in spring 2003, and it will provide a cheap but comfortable solution.

Option 3: Schedule Online

This product is commercial and provides web based calendaring with additional software for synchronisation with Outlook and PDAs.

It is expensive, and cannot provide the conveniences of a solution that fully integrates Outlook.

Option 4: Samsung Contact

This product is also commercial and expensive, but it provides both a web interface and full Outlook integration. The PDA synchronisation is provided by Outlook.

Option 5: Free Internet Calendaring server

The best solution in the future would be a free calendar server in combination with a free calendaring client that both support Internet calendaring. The calendar server could be an extension of the current phpGroupWare API, or a program based on a future version of the libical library, which is an implementation of the Internet calendaring standards. The client could be the Mozilla Calendar.

It is hard to estimate when this solution will be ready, because the Internet calendaring standards are not confirmed yet. But in 2004 there could already have been started a new era of calendaring with some standards based free calendaring servers and clients.

If there had been more time, it would have been interesting to investigate these products even more and to compare their functionality in more detail.

The Kolab server was not installed during this diploma thesis, because it was still in Alpha testing.

There was also no time for the installation of Schedule Online, but the functionality could be checked in the online demo.

The installation of Samsung Contact was successful, and Outlook could be used with all its features. There were problems with installing the web interface which is written in Perl, so that would need further investigation. It was also not fully tested if there could be problems with the email server of the OM Standard Linux Server.

phpGroupWare got more attention, because the code was very interesting. But also here, the functionality was not investigated in all detail. However, some information can be found in the appendix with the description of the solutions.

5.5 phpGroupWare

This chapter describes the phpGroupWare project in more detail. A technical insight into phpGroupWare and its applications is given, by investigating the calendar application and explaining the prototype that was developed in this diploma thesis.

5.5.1 The Structure of phpGroupWare

phpGroupWare is organised in a 3 layer architecture. That means that the user interface, the business layer with the logic and the data storage are seperated from each other. That allows the phpGroupWare project to have the goal of becoming a groupware server that not only provides a web interface. There could be a specialised WAP interface that allows also clients with small screens, e.g. mobile phones, to make use of phpGroupWare. XML-RPCs or SOAP could be used by client applications to access the phpGroupWare server and request services from it.

There is the idea for the future to port the phpGroupWare project to a real application server and use XML for the representation of data and queries [HULL].

To understand the modularisation of phpGroupWare, it is helpful to investigate the structure of the directories and the names of the files in which the source code is represented:

In the root directory of a phpGroupWare installation, there is one directory for each application. Furthermore, there is a directory called "phpgwapi" which holds the classes that provide all necessary functionality for phpGroupWare to be a framework for the development of groupware applications. It provides interfaces to e.g. the formatting of data for specific browsers, the access of data in different databases, account management, file access, and much more. Just look in this directory and read through the different files.

There are several other special directories that don't just hold normal applications, e.g. xmlrpc, soap, syncml-server, and the two directories with the setup tool and the preferences application.

All other functionality of phpGroupWare is provided by several applications.

Each application has the following subfolders:

- doc
- help
    - EN
    - FI
    - ...
- inc
- setup
- templates
    - default
        - images
    - idsociety
        - images
    ...

figure 5.1 Subfolders of a phpGroupWare Application

The "doc" folder should provide some documentation for other developers, while "help" contains information for the users, with subdirectories for several languages.

The "inc" directory is the most important folder of an application, because the classes with the actual source code are placed here. There are different types of files, according to their different tasks. There are the classes that are responsible for the user interface, their name is in the form "class.uiAPPLICATION.inc.php", while "APPLICATION" stands for the name of the application or an important part of the application, that is coded in this file. The files with a name like "class.boAPPLICATION.inc.php" provide the procedures with the business logic,

and bo stands for business objects. The "so" classes are responsible for the access to the data storage, which is normally the SQL database.

The "setup" directory holds all information for the installation and upgrade of the application. There are the language files with translations for all strings shown in the user interface of the application. The information how to create or upgrade the appropriate tables in the database is also provided in this directory. Initial data, e.g. public holidays for the selected country, can also be given here.

The "templates" folder contains one default template layout for the application. There should be one extra subdirectory for each available template that provides a different screen layout.

The calendar application has two additional subfolders in the "inc" directory, which are "import" and "export". These folders hold files for the import and export of calendaring information into and from the several file formats.

5.5.2 A Prototype for CSV Import/Export

The goal was to have at least synchronisation of phpGroupWare with Outlook, because that would enable people to synchronise their PDAs indirectly with phpGroupWare by using Outlook as mediator. There is the hope that the AxisSync tool earlier mentioned will become available and stable soon, that would make synchronisation with PDAs much more easier.

It would already be a little help to allow synchronisation with Outlook via CSV files. The following diagram shows the normal composition of an application with its classes for the user interface, the business logic and the storage parts. Furthermore the modifications that were necessary to support CSV files can be seen in this diagram.
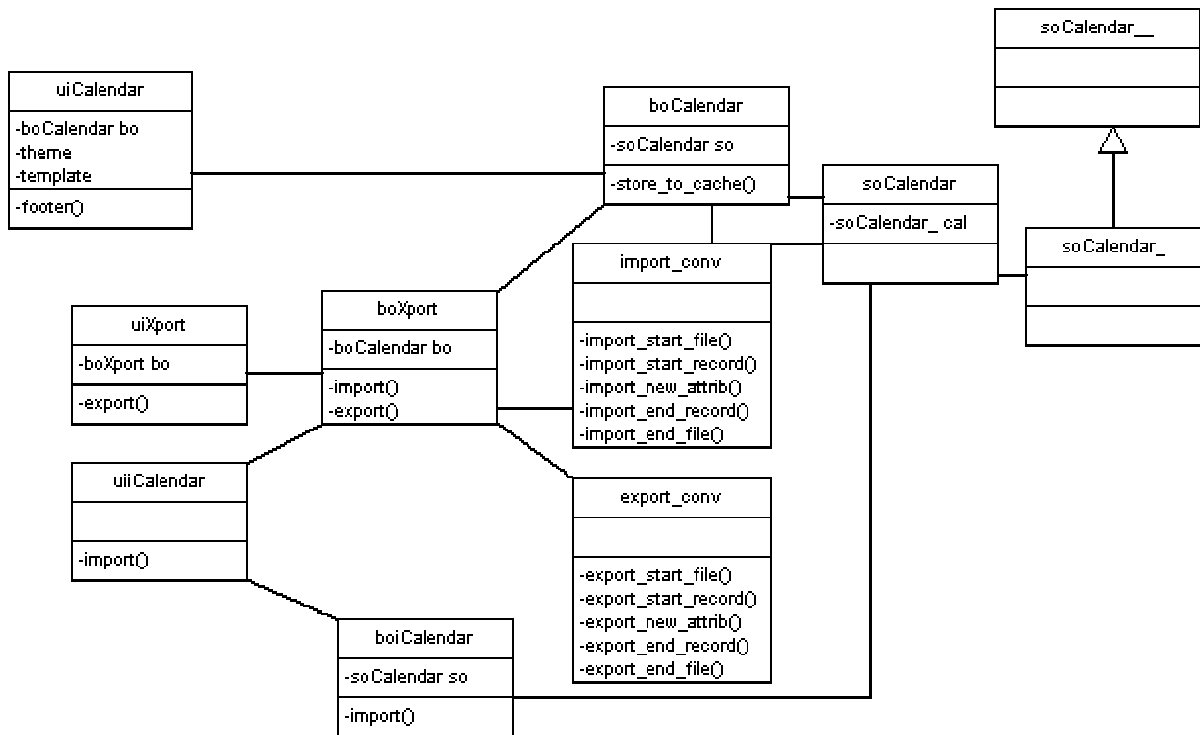
figure 5.2 Class Diagram of the Calendar Import/Export

At the moment, the socalendar_ class is based on SQL storage.

The uiXport, boXport, import_conv and export_conv classes were added to create the prototype for the CSV support. The phpGroupWare address book application already provides CSV functionality, and so the names and the structure of the classes were taken from there. Also the algorithms for storing and loading CSV data with using a buffer were taken from there.

There was already a button for the import of iCalendar files. For a good design it would be necessary to change the structure of the classes, i.e. the functionality of the uiiCalendar class would be integrated into uiXport. But because there was only the goal to provide a prototype, the structure of the classes was not changed.

CSV Export

The uiCalendar class displays the calendar application, and uses the colors and styles of the given theme and the layout of the current template. The uiCalendar class creates an business object, which is an instance of boCalendar. There is the function "uiCalendar.footer()" that needs to be modified so that beside the existing "import" button another button with the name "export" is displayed at the bottom of the calendar application. The template of this function is saved in the file
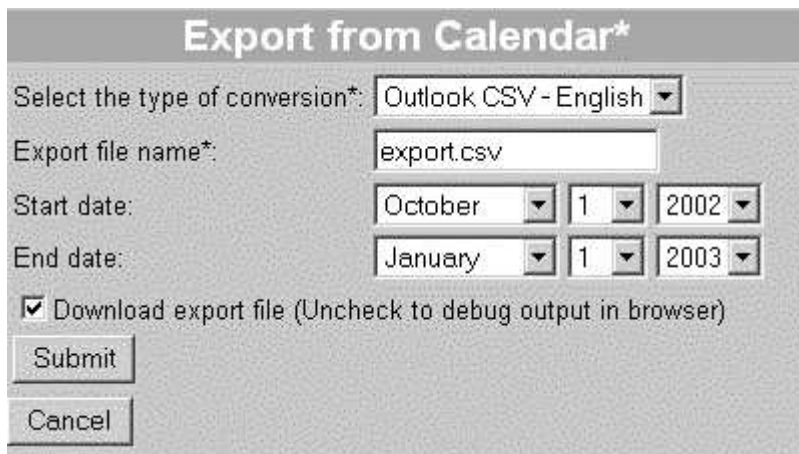
"templates/default/footer.tpl". Another variable needs to be inserted there for the second button:

```
<!-- BEGIN blank_row -->
        <td valign="top" width="10%">
          {b_rowImport}
        </td>
        <td valign="top" width="10%">
          {b_rowExport}
        </td>
<!-- END blank_row -->
```

The button is inserted in the user interface in "uiCalendar.footer()" with these lines:

```
$var = Array(
        'submit_button'            => lang('Submit'),
        'action_url_button'        => $GLOBALS['phpgw']-
>link('/index.php','menuaction=calendar.uiXport.export'),
        'action_text_button'       => lang('Export'),
        'action_confirm_button' => '',
        'action_extra_field'     => ''
);
$this->output_template_array($p,'b_rowExport','form_button',$var);
```

If the button is clicked, the associated action ("menuaction") displays the user interface that is defined in the function "uiXport.export()" and shown in the following figure.

figure 5.3 The Calendar Export Screen

The user can choose from what period of time the calendar items should be exported. He can specify the start date and the end date of this period.

The export() function looks in the "calendar/inc/export" directory for any files, and expects that each of them holds a class called "export_conv". This idea helps to provide export functions for several file formats, e.g. different language versions of Outlook CSV files. The user selects the file format of the exported calendar data by selecting one of the options.

The class uiXport has an object of the class boXport, so when the user clicks the

"Submit" button, the function "boXport.export()" is called. In boXport there is also an object of the class boCalendar, and the function "boCalendar.store_to_cache($period)" returns all calendar items in the given period in an array. Now for every event and every attribute of an event, the details have to be given to the current export_conv class. This class adds all information in the correct format to a buffer. In the end, "uiXport.export()" returns this buffer as a file to the browser of the user.

The following sequence diagram illustrates this process:
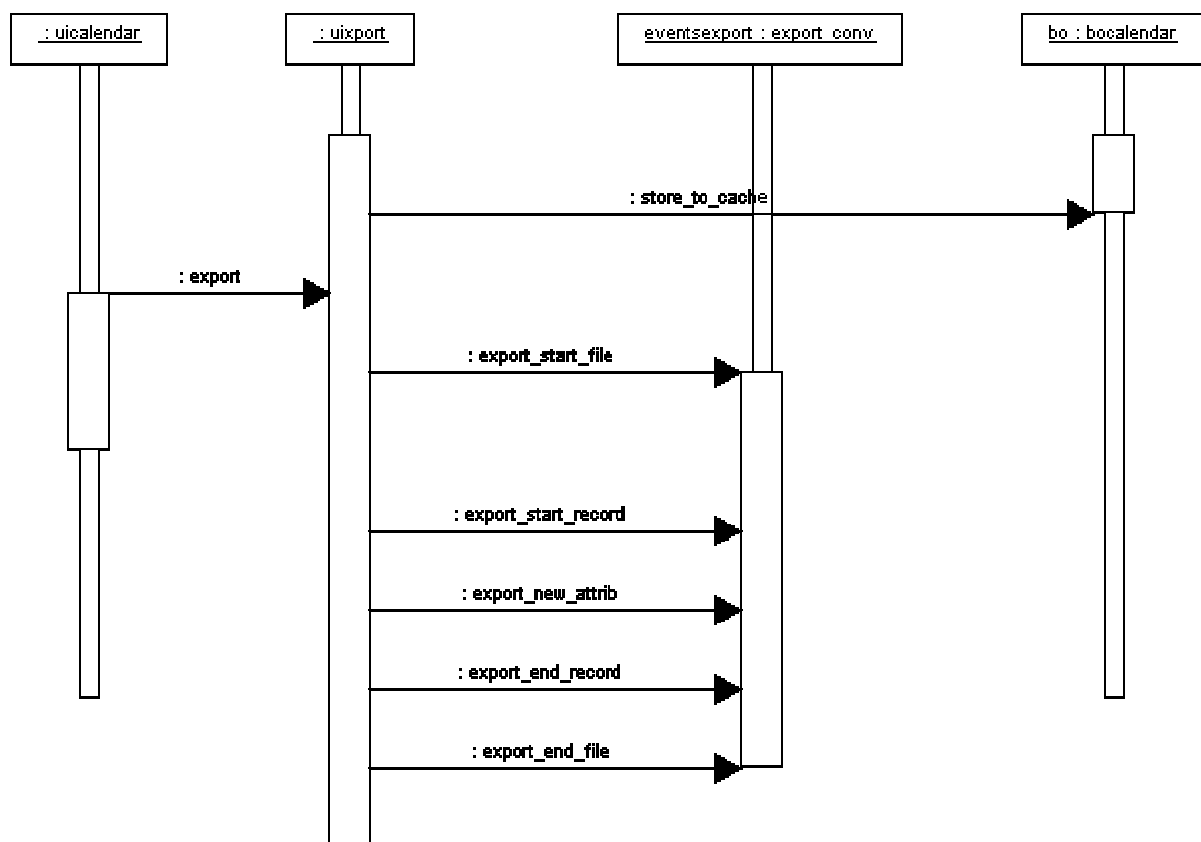
figure 5.4 Sequence Diagram for the Calendar Export

The prototype is able to export the following attributes of an event: The title or subject, the start and the end date, the name of the owner or organiser, the name of the participants, the description, the location, the priority and finally the information whether this event is private or not.

CSV Import

The import functionality was not written in the uiXport class, because there was already an import button for iCalendar files on the front screen of the calendar. As

85

already mentioned before, some functionality was inserted into the uiiCalendar class in order to change the existing classes as less as possible.

In the following screenshot the coexistence of the two possibilities of import is shown.

figure 5.5 The Calendar Import Screen

The way of using different import file formats is the same as with the export process described above. At the moment, an event on the same day and with the same title like the imported event is replaced with the imported event. For more about this topic see below.

The prototype imports the same fields of an event as it exports, but the owner is not yet resolved. The current user becomes the owner of the imported event. Participants are not imported at all at the moment.

Known Problems and Future Improvements

There are some topics that are not too difficult and would improve the import/export extension:

The export functionality should be extended by also exporting the names of the participants of a meeting and the category to which this meeting belongs to.

The classes involved in the user interface of the import functionality should be redesigned. Perhaps the people of the leading team of phpGroupWare should be asked first. The several file formats should be better integrated as well, e.g. should the icalendar format also appear in the drop down box with the Outlook CSV file formats.

At the moment the categories, the participants and the owner are not resolved in

the import process. That should be realised, but it requires that Outlook uses only user names that have accounts in phpGroupWare.

There are also problems that limitate the import and export functionality with CSV files: One problem is that recurring events are split up by the CSV export and import because there is no way to save information about recurring events in the Outlook CSV file format. Another problem are the missing identification numbers of events in the Outlook CSV file format. That means that the import algorithm can only check for attributes, e.g. the subject, in order to find out if an event already exists in the calendar. But because the subject is not unique there can be different events with the same title. And if the subject is changed, there would be 2 events. A solution would be to rely on two or more attributes, e.g. start date and title, and if at least one of them is fitting to an existing event, this event is replaced by the imported event. But that can still always cause trouble.

5.5.3 Administration of phpGroupWare via XML-RPC

This is just a proposal how to integrate phpGroupWare into the OM Standard Linux Server. It should also show a bit of the XML-RPC functionality of phpGroupWare.

The OM Standard Linux Server provides an application based on text menus that helps system administrators to add and edit users. This application is called "sysadm", and it is programmed with Perl scripts. It would be useful, if the creation of a user in phpGroupWare could be integrated with the creation of a Linux user in "sysadm".

There is an XML-RPC implementation for Perl called Frontier::RPC which was written by Ken MacLeod. The "phpgwapi/doc/xmlrpc" directory also contains a file "perl.txt" which describes how to log into the phpGroupWare server with Frontier::RPC.

Only to get a quick overview of the opportunities of XML-RPC in phpGroupWare, the following example shows how to access phpGroupWare from an external PHP script, using the functions of "XML-RPC for PHP" from Edd Dumbill.

This function logs into the phpGroupWare server given in the "domain" variable, and uses the XML-RPC client given in the "client" variable. The client has already

got the IP address of the server machine before the execution of the login function.

```
function login(&$client, $domain, $username, $password)
{
        $sessionid = -1;
        $kp3 = -1;
        $loginparams=new xmlrpcval(
                        array("domain" => new xmlrpcval($domain),
                                "username" => new xmlrpcval($username),
                                "password" => new xmlrpcval($password)
                        ), "struct");
        $msg=new xmlrpcmsg("system.login", array($loginparams));
        $response = $client->send($msg);
        $value = $response->value();
        if (!$response->faultCode())
        {
                $value =  $value->scalarval();
                if ($value['sessionid'])
                {
                        $sessionid = $value['sessionid']->scalarval();
                }
                if ($value['kp3'])
                {
                        $kp3 = $value['kp3']->scalarval();
                }
                $client->setCredentials($sessionid, $kp3);
        }
        return $sessionid != -1;
}
```

"system.login" is the method which is called on the server, and the parameters for the username and password are also delivered.

This is the XML message that is sent by this function:

```
<methodCall>
<methodName>system.login</methodName>
<params>
<param>
<value><struct>
<member><name>domain</name>
<value><string>localhost</string></value>
</member>
<member><name>username</name>
<value><string>timop</string></value>
</member>
<member><name>password</name>
<value><string>mypassword</string></value>
</member>
</struct></value>
</param>
</params>
</methodCall>
```

The answer looks like this:

```
<methodResponse>
<params>
<param>
<value><struct>
<member><name>domain</name>
<value><string>c.ict.om.org</string></value>
</member>
<member><name>sessionid</name>
<value><string>16d7659129b54e81f282fc36ff839191</string></value>
</member>
```

```
<member><name>kp3</name>
<value><string>3c125da178c4b53f360451252da03997</string></value>
</member>
</struct></value>
</param>
</params>
</methodResponse>
```

At the moment, the XML-RPC functionality is not added to every class. In the future, there should be a function called "list_methods" in every business object class. This function returns the names of all the available XML-RPC or SOAP methods.

For creating users using the XML-RPC interface, the file "admin/inc/class.boaccounts.inc.php" provides the function called "rpc_add_user". This function is not working yet. There was no time to find out the problem. But it should be possible to add a user with all the details, i.e. first name, last name, accountid, password, etc., as it is in the "boaccounts.add_user" function.

6 SUMMARY AND OUTLOOK

Just to finish this diploma thesis, this chapter gives a short conclusion about the current situation of the group calendaring market and shows the future functionality that will be required from groupware applications. In the end, there are some personal impressions about the past work for this diploma thesis.

6.1 The Current Situation

After investigating the existing and coming standards and the available solutions for group calendaring, the following conclusions can be made: At the moment, there are convenient and comfortable solutions, but they are commercial and not always based on open standards. Especially concerning the calendaring and synchronisation standards, this would even not be possible, because not all standards are finished and confirmed yet.

The open source community also provides several solutions, most of them are webbased. The huge advantages of a web based solution are the platform independence and the maintenance-free clients. But there is no free groupware server that is able to serve all three types of clients: a webbased interface, an offline client and PDAs. Own effort needs to be brought into each of these projects, and each company or organisation has to decide if this effort is worth the time and money that otherwise could have been spent on a commercial solution.

The other possibility, besides spending money or programming effort, is to wait for the coming times of free and standard based calendaring solutions and just take it when it is ready. A comparable situation is the huge amount of free email servers and clients that can be mixed with each other because they are based on Internet standards.

6.2 The Future of Groupware Functionality

One thing that should be considered is that it would be easier for the users if groupware functionality would be integrated in one user interface. It would help with learning how to use the system because there is not a different user interface for each application, and it would make communication via calendar, email, instant messaging or video conference even more normal.

The goal of integration is also pursued by the two companies that are the leaders in the collaboration market, IBM/Lotus and Microsoft. There are two texts which each describe the functionality expected of the coming releases: Cain writes about the future of Microsoft Exchange [CAIN], and Olsen and Hawkins give an outlook to the next releases of IBM/Lotus Notes/Domino [OLSEN]. Both companies mention the term **Contextual Collaboration** which exactly means the integration of different applications that are delivered as components.

Microsoft plans that third parties can use Exchange services and content in their applications. Microsoft also will improve the performance in two areas: The required bandwidth should be reduced, which is an increasing problem because of the growth of email traffic and storage. Furthermore the storage on the server should be moved from the current solution of a special database to the SQL server.

Lotus Domino will move closer to the application server of IBM, which is called Websphere. The goal is to develop a new server program called NextGen which provides a J2EE platform and web services. It will coexist with Domino for a time, and finally replace it. Lotus Sametime will become a real-time collaboration server. Another topic for the future is Pervasive Computing: Both servers will become more aware of pervasive devices and allow synchronisation and provide data access for PDAs, mobile phones, and other mobile devices.

6.3 Personal Impressions

It seems that there are a lot of open source projects in progress, and it was interesting to watch the ongoing work and take a snapshot of the current situation. A better understanding of how open source projects work could be achieved by realising their problems and success. It was interesting to read the mailing lists and by that to be a passive part of some open source projects and the group of people that are formulating the Internet calendar standards.

It was good to have the chance to work without a lot of constraints on an own project, with the need to organise the time and work and to be finished on a fixed date.

A huge amount of information was collected during the work for this diploma thesis: The links to Internet articles, the details of the solutions, and the text of the

diploma thesis itself needed to be managed. This was achieved by storing all the information in XML files. Some PHP scripts formatted the data and printed it to HTML documents. An Word macro was developed to help formatting the printable text of the diploma thesis. These scripts can be found on the enclosed CD-ROM, together with the online version of all information.

7 APPENDIX

7.1 Table of Requirements

category	priority	requirement: title & description
Profiles, user habits, location of mails	2	**people use different computers at different times to access their messages** Any user must be able to log into any workstation, with all personal calendar information & settings following him
	3	**configuring users settings** add or administer a users settings without using his account and password
import/export of data	2	**exchange with handhelds** Conduit to PalmOS Interfaces needed both ways: - data from Palm Pilots/Outlook to "groupware" - data from "groupware" to Palm Pilots/Outlook need to exchange at least calendaring information.
	2	**disconnected mode** / **offline client** needed for travellers In "disconnected" operation, a client connects to the server, makes a copy of all user account information, and then disconnects from the server, later to reconnect and resynchronize with the server. it mainly has to work with the personal calendar information
Calendaring / Booking	2	**Allows calendars to be shared** Viewing others Calendars, but not changing them
	3	**calendar managed by other person** permit other persons to manage personal calendar (e.g. Directors Secretary managing his calendar)
	3	**organizing a meeting** the one organizing a meeting needs to be able to ask the system to find a time when the invited persons are all free and to then book (or suggest) a meeting...
	3	**common (not personal) calendars** absences (see more below) meetings current events
	4	**Absences** annual leave booking, so system that handles - person requests leave - manager approves

	3	**Resource Booking** (special kind of common calendar) types of resources: vehicles, equipment, rooms, office equipment (e.g. projector, mobile phones etc) resources need to be set up by the administrator each resource has an assigned manager that approves the booking if there is a more urgent request this manager can cancel the previous booking, an email is send to the affected user support multiple resources of the same type
requirements from Linux Project Team	1	**long term reliability of support** The support of the solution needs to be guaranteed by either a reliable company or an active user/developer community.
	2	**costs** The costs of the solution must be affordable by OM. The costs consist of both the costs of the software licences (server and client software) and the cost of maintenance
	2	**linux server support** The server of the groupware solution has to run on the OM Standard Linux Server to provide a common solution installed on one machine.
	2	**windows workstation support** The client of the groupware solution has to run on Microsoft Windows for that is the standard operating system in OM.
	4	**POP3/SMTP support** the solution needs to support the OM Mail module (based on sendmail) for eMail Clients that use POP3/SMTP (e.g. Pegasus Mail, Outlook)
other types of communication	4	**Voting** Voting capability (Send out a survey, or invitation, and people can click a button to reply yes / no, accept/reject etc)
	4	**discussion list management** - notice board (e.g. items for sale/loan)

7.2 Table of Solutions

APPOINT.NET	**appoint.net**
Home	http://www.appoint.net
category	**service provider**, webbased service provider
decision	**not suitable: no linux server support**

APPOINTMENT QUEST	**Appointment Quest**
Home	http://www.appointmentquest.com/
category	**service provider**, webbased service provider
decision	**not suitable: no linux server support**

AXISSYNC	**AxisSync 0.5.0 Alpha**
Home	http://www.axisgroupware.org
other links	http://docs.axisgroupware.org/index.php?page=AxisSync http://savannah.gnu.org/projects/axis/
category	**sync**, synchronisation of palm and phpgroupware client: Palm Hotsync and Netsync server: phpgroupware, pgsql, coldsync
licence/costs	opensource (GNU General Public License (GPL)) costs: free
documentation	http://docs.axisgroupware.org/index.php?page=axissyncREADME
marketing statement	Axis Groupware is a groupware suite for workgroups and enterprise, built on phpGroupWare. Axis Groupware is not a fork of phpGroupWare, but rather a custom distribution of phpGroupware focusing on high quality, production deployment. AxisSync brings palm synchronisation to phpGroupware.
installation	Download: http://axisgroupware.org/axissync/axissync-0.5.0.tar.gz official installation guide: http://docs.axisgroupware.org/index.php?page=axissyncINSTALL
functionality	• *data exchange*: synchronizes palm / phpgroupware: memos, address, calendar, todos not all conduits are working yet
reliability of support	2 developers; Alpha 0.5.6 was released in the end of October; work is going on
contra	THIS IS AN ALPHA RELEASE! DO NOT USE THIS IN A PRODUCTION ENVIRONMENT!
conclusion	**Not all conduits are available yet.**
decision	**promising, but still under development**

BILL WORKGROUP	**BILL Workgroup Server 2.1.3**
Home	http://www.billworkgroup.org/billworkgroup/home
other links	http://developer.berlios.de/projects/bill http://www.n-h.com http://www.nlcom.nl/NLcom/US%20Site/
category	**Outlook/Linux**, outlook in combination with a linux server client: Outlook + MAPI Service Provider server: BILL Open Workgroup Server
licence/costs	Server software is opensource (GNU General Public License (GPL)), MAPI Service Provider is commercial, Outlook is commercial costs: Outlook belongs to MS Office; 39 EUR per MAPI Client
documentation	not available
marketing statement	N&H received more and more customer requests asking for a independent workgroup server solution running under LINUX, but using MS-Outook(TM) as the Client on the MS-Windows(TM) Desktop for use with shared calendars and task lists, etc, and not limited to e-mail. MS-Outlook(TM) uses a MAPI Service Provider to store workgroup data centrally in MS-Exchange(TM) database or other 3rd party messaging servers. Based on this, N&H decided to start an internal project that implements a storage server that can run on LINUX where different workgroup clients and other applications can exchange workgroup data. BILL Open Workgroup Server is now under the GNU public licence.
installation	Download: http://www.billworkgroup.org/billworkgroup/home/Download official installation guide: http://www.billworkgroup.org/billworkgroup/home/Doc/BillInstallLinux
functionality	• *calendar*: Outlook in workgroup mode, shared folders, user needs to give others the permission to see his folder with the calendar information • *email*: outlook functionality; perhaps problems with sendmail configuration of OM Linux • *other modules*: - • *data exchange*: palm synchronisation belongs to outlook functionality
reliability of support	N&H is a two man company, they sell the Easygate workgroup server, and support it. The opensource Bill Workgroup Server is part of the Easygate server, and is also developed by them and one other person. The Bill Server was put under GPL in February 2002.

pro	-
contra	main focus on small and medium sized workgroups: Bill has a poor performance with more than 30 accounts. no big community or company to support the product calendar functionality works just with shared folders. To see another person's calendar this person needs to give permissions to you.
conclusion	**The supporting community of the BILL Server is too small, and the Easygate Server with support is not suitable for OM, since this is a standalone server and cannot be integrated with the OM Standard Linux Server.**
decision	**not enough reliability of support**
BROWNBEAR CALCIUM	**BrownBear Calcium Web Calendar Server and TripleSync 3.7**
Home	http://www.brownbearsoftware.com/calcium/
other links	http://www.brownbearsoftware.com/calcium/TripleSync.html
category	**webbased & client** server: perl based, runs on linux (e.g. Apache)
licence/costs	commercial costs: triple sync unlimited copies: $1,500; Calcium: unlimited number of calendars, unlimited users, basic email: $395.00
documentation	see http://www.brownbearsoftware.com/calcium/docs/ and http://www.brownbearsoftware.com/calcium/FeaturesList.html
installation	Online Demo: http://www.brownbearsoftware.com/calcium/demos.html Download: http://www.brownbearsoftware.com/calcium/download.html
functionality	*calendar*: good repeating events; can only invite via email; can insert an appointment into several calendars at the same time. But events are duplicated, change of event means that every event needs to be changed: see in doc: Add to Multiple Calendars Dynamic Calendar Merging: department calendars detailed user settings for each user, watch/edit rights for each calendar no conflict management, no comparison of busy times *data exchange*: offline client available; synchronises with pda's; Outlook CSV text files import and export
pro	nice integrated solution of Handheld + Desktop + Web
contra	no real group calendaring; Desktop client has no access to several calendars
conclusion	**If it was for free, then it would be worth consideration. But without real group calendaring but only multiple calendaring this is not suitable.**
decision	**not suitable: no group calendar included**

BSCW	**BSCW 4.0.6**
Home	http://www.bscw.de/index_en.html
other links	http://bscw.fit.fraunhofer.de/
category	**webbased** client: webbrowser, javascript server: Linux / windows program, using Python and CGI
licence/costs	commercial costs: schools and universities receive royalty free licences for educational purposes; others: e.g. 1000 user: 16.850,- EUR
documentation	http://bscw.gmd.de/bscw_help-4.0/english/
marketing statement	The BSCW system supports collaboration by providing shared workspaces over the Internet. A shared workspace allows storage and retrieval of documents and sharing information within a group. This functionality is integrated with an event mechanism to provide each user with an awareness of the activities of others within the workspace. It comprises numerous features, e.g., support for threaded discussions, version management of documents, group management, search features and many more. The system is designed primarily to support self-organising groups.
installation	Online Demo: http://bscw.gmd.de/bscw/bscw.cgi User/password: get your own free account Download: http://bscw.gmd.de/InstallUnixServer.html official installation guide: http://bscw.gmd.de/InstallUnixServer.html
functionality	*calendar*: groupcalendaring works this way: search for a person, e.g. test2. insert this user into addressbook create a folder, make it shared. invite member now you can do: File New GroupCalendar but: you have to invite the members that can see/use your calendar the other person has access to the shared folder in his home window can put persons together to a group can create roles and assign to different people, for each shared folder shared calendar only with permitted people. Can build group, permission easy possible to create a public calendar, only for reading, and permit specific users for changing. *invitation*: sends email to invited people, even to people that are not members of the system no comparison of free time; every participant needs to confirm, else the appointment is marked tentative *private appointments*: not possible in shared calendar

	• *repeating events*: no repeating events
	• *resource booking*: no selection of resources
	• *email*: no integration of webmail, but uses sendmail for invitations
	• *other modules*: shared folders, shared documents, addressbook, waste basket, clipboard, discussion forums
	• *data exchange*: no open source, so there is no possibility for creating an own synchronisation tool for palm or outlook
	supports export of event in iCalendar file; no import of iCalendar files
	no export / import of vCard
reliability of support	The project was started in 1995 by FIT Fraunhofer, belonging to GMD, the German National Research Center for Infomation Technology. OrbiTeam Software GmbH was founded in summer 1998 as a spin-off company of GMD. OrbiTeam offers support, maintenance and extensions of the system. Since 1995 more than 800 BSCW Servers have gone online.
pro	a lot of good calendaring and document sharing functionality
contra	synchronisation is not provided
conclusion	**palm and outlook synchronisation is not available**
decision	**missing synchronisation support**
BYNARI INSIGHT	**Bynari Insight Server 3.5.4**
Home	http://www.bynari.net
category	**Outlook/Linux**, outlook in combination with a linux server client: Outlook+InsightConnector server: linux based server software (uses Cyrus IMAP server, Exim MTA, and OpenLDAP)
licence/costs	commercial (Outlook, InsightConnector, InsightServer) costs: Insight Server Standard Edition 25 User Base Pack (Special priced bundle package with Connector) = $600.00; 25 Additional Users License Pack = $495.00
documentation	http://www.bynari.net/UserManual35.pdf
marketing statement	Bynari's InsightServer provides a centralized messaging and collaboration system running on Linux. It allows users to migrate from installed and running Exchange servers to an open source, RFC compliant model. With this program installed on a Linux server, combined with the Bynari InsightConnector installed on client workstations, users will be able to continue to use their existing Outlook interface with very few changes.
installation	Download: http://www.bynari.net/downloads.html official installation guide: http://www.bynari.net/insightserver35x_install.pdf
functionality	• *calendar*: outlook functionality

	• *email*: outlook functionality; uses Exim MTA instead of sendmail: the configuration of exim is quite different to that of sendmail. • *other modules*: shared folders • *data exchange*: palm synchronisation belongs to outlook functionality
reliability of support	company was founded in fall 1998. InsightConnector and InsightServer were first released in 2000. a lot of detailed documents show the reliability of the company and the product
pro	cheaper than Microsoft Exchange
contra	email server causes problems with sendmail
conclusion	**That would mean for OM to change the current sendmail configuration to exim.**
decision	**too difficult to integrate with current OM solution**
CALDERA VOLUTION	**Caldera Volution Messaging Server Email Server Release 1.1**
Home	http://www.caldera.com/products/volutionmsg/
other links	http://www.caldera.com/products/volutionmsg/datasheet.html
category	**Outlook/Linux**, outlook in combination with a linux server client: outlook server: Platform: Caldera OpenLinux Server 3.1 or Open UNIX 8 Release 8.0 with Linux Kernel Personality
licence/costs	commercial
documentation	http://docsrv.caldera.com:8457/en/Navpages/MsgDoc.html
marketing statement	Volution Messaging Server is designed to be compatible with the Microsoft Outlook and other mail clients. It comes pre-configured to support mail, LDAP address book, and busy-free calendar features of this popular mail reader. Major components of Caldera's Volution Messaging Server include an advanced mail agent, proven IMAP server, support for OpenLDAP, and built-in security.
functionality	• *data exchange*: Outlook is supported concerning the Internet Free/Busy feature
conclusion	**no Redhat Linux support**
decision	**not suitable: no linux server support**
CALENDAR INTERNET	**Calendar Internet 2002**
Home	http://aufrance.com/calendar.htm
category	**webbased** server: runs on Microsoft IIS or O'Reilly WebSite
decision	**not suitable: no linux server support**
CALENDARS.NET	**Calendars.net**
Home	http://www.calendars.net/
category	**service provider**, webbased; supports Outlook

licence/costs	free for charities
marketing statement	Calendars Net currently provides over 50,000 interactive web calendars for free. Most of these calendars are for schools, churches, and other civic or social organizations, worldwide. We provide these calendars (and technical support) for free.
installation	Online Demo: http://www.calsnet.net/practice
functionality	• *data exchange*: import / export of calendar data via .csv file with Outlook; can be used with BrownBear iCal.
decision	**not suitable: no linux server support**
CALENDARSERVER.COM	**CalendarServer.com**
Home	http://www.calendarserver.com
category	**service provider**, webbased
contra	only publishing of personal calendar
decision	**not suitable: no linux server support**
CHANDLER	**Chandler**
Home	http://www.osafoundation.org/our_product_desc.htm
category	**client** client: should run on Windows, Linux, Mac
marketing statement	Our product (code-named "Chandler" after the great detective novelist Raymond Chandler), is a Personal Information Manager (PIM) intended for use in everyday information and communication tasks, such as composing and reading email, managing an appointment calendar and keeping a contact list. Because of the ease with which Chandler users can share information with others, we might call Chandler the first Interpersonal Information Manager. (The term PIM was first used in conjunction with the product Lotus Agenda in the 1980's. Chandler is the spiritual descendant of Agenda (and has a common designer in Mitch Kapor.)
reliability of support	Open Source Applications Foundation: Mitchell Kapor, Founder of Lotus, has given $5 million dollar to this project, and is leading it. Also other big names are involved, e.g. Tim O'Reilly. They want to use as much existing open source software as possible.
conclusion	**only design plans; first alpha version promised for end of 2002**
decision	**experimental**
CONTACTOFFICE	**ContactOffice**
Home	http://www.contactoffice.com/
category	**service provider**, webbased service provider
functionality	• *data exchange*: with palm / outlook synchronisation
conclusion	**no local installation of server possible**
decision	**not suitable: no linux server support**

CYBERSCHEDULER	Crosswind Cyberscheduler & Synchronize & CyberSync
Home	http://www.crosswind.com/
other links	http://www.envicon.de/cyberscheduler/cybersync/index.html
category	**webbased & client** client: webbased & clients server: linux or NT
licence/costs	commercial costs: see http://www.envicon.de/e/cyberscheduler/preise.html; no prices for Synchronize & CyberSync; 500 licenses; CyberScheduler & CyberSync 50 EUR; Synchronize: ??? DM
documentation	http://www.crosswind.com/feature3.htm
installation	Online Demo: http://www.crosswind.com/CyberSched/En_US/login.html Download: see http://www.crosswind.com/testlock.htm#synch and http://www.envicon.de/e/cyberscheduler/cybersync/indexe.htm l
functionality	• *email*: no email functionality • *data exchange*: Palm Sync supported by CyberScheduler; offline client supported by Synchronize; But no Palm Sync with Synchronize?
reliability of support	There seem to be two companies: Crosswind and envi.con; The price list of envi.con is not really helpful, still using DM; really bad integration of english translation
pro	realtime group calendaring
contra	not both offline client and palm sync
decision	**missing synchronisation support**
DAYPOINT	**FrontOffice Communications: DayPoint Enterprise**
Home	http://www.daypoint.com/products/daypointent.asp
category	**webbased & Outlook** server: Microsoft Server
licence/costs	commercial costs: Price per server, unlimited users: $995
functionality	• *data exchange*: Palm and Outlook synchronisation
decision	**not suitable: no linux server support**
E2DO	**www.e2do.com**
Home	http://www.e2do.com/
category	**service provider**
decision	**not suitable: no linux server support**
ECAL	**eCal M1**
Home	http://www.ecal.com/products.asp?action=m1
category	**webbased & Outlook** client: webbased

	server: Sun Solaris, Oracle Database
licence/costs	commercial
documentation	http://www.ecal.com/documents/m1_ss_1.pdf
functionality	• *data exchange*: Palm and Outlook support
decision	**not suitable: no linux server support**
ERIDU	**eridu**
Home	http://eridu.sourceforge.net
category	**webbased**, php
conclusion	**August 2001: Currently the Eridu project is on hold.**
decision	**not enough reliability of support**
GHC	**Great Hill Corporation: Calendars for the web 2.5.1**
Home	http://calendar.greathill.com/
category	**webbased** client: webbased server: linux/apache
licence/costs	commercial
documentation	http://calendar.greathill.com/calweb/cw_help/usermanual_frame.htm
installation	Online Demo: http://calendar.greathill.com/cgi-bin/calweb/calweb.cgi Download: http://calendar.greathill.com/dlarea.htm
functionality	• *data exchange*: import/export via text files, but no synchronisation
decision	**missing synchronisation support**
GROVEWARE	**Groveware CAL/RM 2000SE**
Home	http://www.groveware.com/products/crm2000se_overview.htm
category	**webbased**
decision	**missing synchronisation support**
HTMLCAL	**HTMLCAL**
Home	http://www.htmlcal.com
category	**webbased**, cgi client: webbased server: linux apache
licence/costs	costs: unlimited calendars: $400
installation	Online Demo: http://www.htmlcal.com/demo/
conclusion	**not a useful calendar: only formatting calendar data**
decision	**missing synchronisation support**
HUNTCAL	**www.huntcal.com**
Home	http://www.huntcal.com/
category	**service provider**
decision	**not suitable: no linux server support**
HYPERWAVE	**Hyperwave Team Workspace**

Home	http://www.hyperwave.com/e/products/tws.html
category	**non linux server**, webbased
functionality	• *data exchange*: outlook synchronisation
contra	requires Oracle 8i (8.1.7) or Microsoft SQL Server 2000
conclusion	**server doesn't run on linux**
decision	**not suitable: no linux server support**
ICQ GROUPWARE	**ICQ Groupware 1.12 Beta**
Home	http://www.icq.com/groupware/
category	**no calendar**, messaging tool (without calendar) client: ICQ groupware client server: program runs on WinNT (both server or workstation)
licence/costs	commercial
documentation	http://www.icq.com/groupware/specifications.html
marketing statement	The System is designed to support communications and collaborative work over Intranet. It comes with many internal functions such as group chats, online and offline messaging, collaborative browsing, file transfer, central message broadcasting and more. The System also supports many popular collaborating voice, video and data applications. The System is designed to support groupwork. It supports multi-user chats, multi-user collaborating browsing and multi-user messaging. The user can construct a list of members of his group and see at all times their online/offline availability.
installation	Download: http://www.icq.com/groupware/register.html
functionality	• *calendar*: no calendar available • *other modules*: chats, online and offline messaging, collaborative browsing, file transfer, central message broadcasting; collaborating voice, video and data applications
contra	no calendar beta version
conclusion	**no calendar**
decision	**not suitable: no group calendar included**
KDE PIM	**KDE PIM application suite (with KOrganizer)**
Home	http://korganizer.kde.org/
other links	http://pim.kde.org/
category	**client**, linux groupware client
documentation	http://korganizer.kde.org/workshops/KOrganizerIMIP/en/html/index.html
marketing statement	KOrganizer is KDE's calendaring and scheduling program for organizing appointments, todo lists, projects and more. It is an integral part of the KDE PIM suite, which aims to be a complete solution for organizing your personal data. KOrganizer supports the two dominant standards for storing and exchanging calendar data, vCalendar and iCalendar.

functionality	• *calendar*: The group scheduling is based on a peer-to-peer architecture using email as communication medium. • *email*: kmail • *data exchange*: see http://pim.kde.org/components/kpilot.php
decision	**not suitable: no windows workstation support**
KOLAB	**Kolab / Free Software Groupware Project**
Home	http://kroupware.kde.org/
category	**Outlook/Linux** client: Outlook and Outlook Bynari Connector or KDE PIM tools, Kroupware server: open source linux server
licence/costs	opensource, but uses outlook and bynari connector (Insight Connector 500 User Pack $11,000.00) costs: bynari connector
documentation	http://kroupware.kde.org/concept-1.0.1/index.html
installation	Download: only alpha version at the moment: ftp://master.kde.org/pub/kde/unstable/server/kolab/kolab-current/ official installation guide: http://kroupware.kde.org/howto-kolab.html
pro	should work in some time with Kroupware on windows, without outlook
conclusion	**At the moment only Outlook is supported as windows client, and it requires the Bynari Connector.** **Or wait for the KDE PIM port to windows, but the development of that has not started yet.**
decision	**promising, but still under development**
LIVELINK	**Opentext Livelink 9.1**
Home	http://www.opentext.net/livelink
category	**non linux server**, webbased server: Windows or Solaris
licence/costs	commercial
functionality	• *calendar*: optional shared scheduling;
conclusion	**no linux server support**
decision	**not suitable: no linux server support**
LOCALENDAR	**www.localendar.com**
Home	http://www.localendar.com/
category	**service provider**
decision	**not suitable: no linux server support**
LOTUS DOMINO	**(IBM) Lotus Domino / Notes**
Home	http://www.lotus.com/products/r5web.nsf/webhome/nr5noteshp

other links	http://www.lotus.com/products/r5web.nsf/webpi/Notes?opendocument&cwesite=notes http://www.lotus.com/products/r5web.nsf/webhome/nr5serverhp-new
category	**webbased & client** server: domino runs on linux, Windows, ...
documentation	see http://www-10.lotus.com/ldd/today.nsf/lookup/notes_rnext_technical_overview and http://www-10.lotus.com/ldd/today.nsf/lookup/domino_rnext_technical_overview
conclusion	**overkill of functionality**
decision	**too difficult to integrate with current OM solution**
MAILSTUDIO	**MailStudio Calendar Server**
Home	http://www.mailstudio.com/prod_cal_ov.html
category	**webbased & client** client: webbased server: unix / Linux supported, Windows 2000 support is planned
licence/costs	commercial
installation	Online Demo: http://cal.mailstudio.com/index.cgi?lang=eng&tnum=2 User/password: guest/guest
functionality	• *data exchange*: minimal outlook support: can receive invitations and react on them
conclusion	**no calendar synchronisation with offline client**
decision	**missing synchronisation support**
MDAEMON	**MDaemon**
Home	http://www.altn.com/Products/Default.asp?product_id=MDaemon
category	**webbased** server: Microsoft Windows 95 OSR2/98/ME/XP/NT/2000 operating system
decision	**not suitable: no linux server support**
MEETINGMAKER	**Meeting Maker 7.1**
Home	http://www.meetingmaker.com
category	**webbased & client** client: supports mac, win, solaris, and webinterface server: solaris, win, mac, linux (with X11)
licence/costs	commercial costs: mailto:europe@meetingmaker.com
documentation	http://www.meetingmaker.com/support/
marketing statement	Meeting Maker, Inc.'s meetingmaker is a real-time, cross-platform calendaring & scheduling solution.
installation	Download: http://www.meetingmaker.com/sales/trial

functionality	• *calendar*: support of resources, free/busy times, can check availability before inviting; real time calendaring • *invitation*: can invite people outside the system • *email*: e-mail independent system architecture; therefore OM mail is supported • *data exchange*: synchronisation with PDA over the network is provided. The system is built on a distributed, replicated database. A copy of most data is available locally, that supports the use of a laptop disconnected from the network. Synchronisation with Microsoft Outlook Calendar.
reliability of support	The company Meeting Maker Inc. has been working with real time calendaring for over 12 years.
contra	graphical interface for server administration required
conclusion	**OM Standard Linux works without graphical interface.**
decision	**too difficult to integrate with current OM solution**
MICROSOFT EXCHANGE	**Microsoft Exchange 2000**
Home	http://www.microsoft.com/exchange/
category	**non linux server**, windows server, with Outlook as client server: runs on Microsoft Windows 2000 Server
contra	no linux support
conclusion	**This server will not run on a Linux machine.**
decision	**not suitable: no linux server support**
MICROSOFT OUTLOOK	**Microsoft Outlook 2000**
Home	http://www.microsoft.com/outlook
other links	http://support.microsoft.com/default.aspx?scid=kb;en-us;Q179431 http://support.microsoft.com/default.aspx?scid=KB;EN-US;q270044 http://support.microsoft.com/default.aspx?scid=kb;en-us;Q180199
category	**client**, windows groupware client client: windows software server: not included
licence/costs	commercial costs: belongs to Microsoft Office
marketing statement	A powerful communication and scheduling program, Microsoft Outlook version 2002 offers new and enhanced functionality that will improve your efficiency and make it easier to collaborate with colleagues. Support for multiple e-mail accounts, including MSN Hotmail, simplify the task of managing your information. Features like Propose New Time

	and Group Schedules makes it a snap to coordinate team schedules.
installation	Download: belongs to Microsoft Office
functionality	• *calendar*: vCalendar files can be imported/exported supports free/busy times in Internet Only Mode; not tested, needs at least the Internet Publishing Wizard; it would work with an http server with Frontpage Extensions on the server; see installation hints in workgroup mode (using a server, e.g. Exchange) Calendars can be published • *data exchange*: synchronisation with Palm: uses Palm desktop software and Chapura PocketMirror software allows import and export of calendaring data via CSV text files
reliability of support	Microsoft is a huge company.
pro	is already installed with MS Office on all clients
conclusion	**It either could be used with an HTTP server for publishing and subscribing free busy times of other calendars. But the current OM configuration does not work. Outlook 2000 requires the Microsoft Web Publishing Wizard, which is for free, but only supported with Win98 and WinNT.** **Or it could be used as client for an Exchange server simulation. This normally requires a MAPI connector, which could be the Bynari Insight Connector, but this is too expensive for OM.** **Or it could be used with its CSV (Comma Separated Values) import/export functionality for synchronisation with another product, e.g. phpGroupWare. But programming effort is needed to import/export CSV files in phpGroupWare.**
decision	**could be suitable / depends on other solution**
MOREGROUPWARE	**moreGroupware 0.6.4pl1 Neo**
Home	http://www.moregroupware.org
other links	http://sourceforge.net/projects/moregroupware/
category	**webbased**, webbased/php client: webbased server: php, mysql
licence/costs	server: opensource (X11 license, which is GPL compatible); client: every web browser is suitable costs: nothing
documentation	no documentation yet available (just faq: http://www.moregroupware.org/faq.php)
marketing statement	Web-based groupware written in PHP4. Including modules like calendar, notes, todo, contacts, project management and others.

installation	Online Demo: http://demo.moregroupware.org/ User/password: admin/admin Download: http://www.moregroupware.org/download.php official installation guide: in the distribution archive of moregroupware, see docs/INSTALL
functionality	• *calendar*: The Calendaring supports invitations of other users. display free and busy times of several people in comparison look at other's calendar, private entries possible • *email*: eMail works with existing smtp/pop3 servers. The OM sendmail functionality is supported. • *other modules*: There are a lot of other modules, but they can be disabled. There are the following modules available: calendar, contact, projects, todo, webmail, forum other modules are in development There is a very good user and group management, the admin can change the rights both per user and per group for each module. several admins are possible • *data exchange*: no palm support; no support of vCalendar or vCard
reliability of support	18 developers on Sourceforge; very active community (19 open Bugs out of 610 Bugs); supported by several companies
pro	a good shared calendar module
contra	no update functionality yet, administrator is responsible for changing the database to the new structure
conclusion	**This product is not ready yet. There is no guarantee for support of user data in future releases.** **no support of palm synchronisation**
decision	**missing synchronisation support**
MOZILLA	**Mozilla Calendar 0.8**
Home	http://www.mozilla.org/projects/calendar/
other links	http://www.mozilla.org/projects/calendar/requirements.html http://www.jsoft.com/Gary/calendar/
category	**client**, free client software, calendaring client: integrated in Mozilla server: ftp, http
licence/costs	opensource (MPL/LGPL/GPL) costs: nothing
marketing statement	Mozilla is an open-source web browser, designed for standards-compliance, performance and portability. The Mozilla Calendar project is meant to create an open source, standards based calendar solution. It will be based on existing calendar standards: iCalendar, iTIP, and iMIP. The initial codebase is being donated by OEone Corporation and

	should be thought of only as a demonstration of what can be accomplished using XUL, JS, CSS and XPCOM.
installation	Download: see installation hints
functionality	*calendar*: see requirements; publish calendar to a FTP server or webDAV enabled webserver and subscribe to other calendars using HTTP *data exchange*: palm support is in the requirements, but not yet implemented import of outlook calendar data is implemented
conclusion	**Palm support promised, but not availabe** **Free Busy Calendaring gives a error message, does not work at the moment**
decision	**missing synchronisation support**
NETSCAPE CALENDAR	**Netscape Calendar Server**
Home	http://wp.netscape.com/calendar/v3.5/index.html
other links	http://wp.netscape.com/communicator/calendar/v4.0/index.html
category	**outdated**, client/server solution client: Netscape Communicator 4 Professional Edition with Calendar server: Netscape Calendar Server
reliability of support	Netscape Calendar is not longer available: http://sillydog.org/narchive/calendar.html Netscape Calendar Server also does not seem to be available anymore. Sun and Netscape are joined in an alliance, so there are no 2 calendar servers! http://wp.netscape.com/netscape/alliance
conclusion	**outdated**
decision	**outdated**
NOVELL GROUPWISE	**Novell Groupwise 6**
Home	http://www.novell.com/products/groupwise/
other links	http://www.gwuser.org/
category	**webbased & client**, windows/netware server, client either dedicated workstation client or webbased client: runs on Windows, or web access, or palm device server: runs on Netware or Windows NT
licence/costs	commercial costs: 25 user $3,250.00
documentation	http://www.novell.com/documentation/lg/gw6/index.html
marketing statement	Organizations simply can't run without e-mail, calendaring, document management and other collaborative activities. GroupWise 6 ensures these Net services are available wherever you are. Only GroupWise has wireless support 'out of the box'. You can use any WAP-enabled device or PDA to

	check e-mail, book appointments and share documents.
installation	Download: http://download.novell.com/download.jsp?pid=136&target=sdExpLic.jsp official installation guide: http://www.novell.com/documentation/lg/gw6/pdfdoc/gw6_install/gw6_install.pdf
conclusion	**This server will not run on a Linux machine.**
decision	**not suitable: no linux server support**
OPENOFFICE.ORG	**OpenOffice.org Groupware**
Home	http://groupware.openoffice.org
other links	http://www.opencap.org/html/ http://www.openoffice.org/files/documents/81/7727/groupwarefaq.htm
category	**client**, opensource client (LGPL/SISSL) client: integrated in the openoffice.org suite server: using an opensource calendar server (not existing yet)
licence/costs	opensource costs: nothing
documentation	no product yet available
marketing statement	To establish standard-compliant groupware solutions that will: Work from within the OpenOffice.org office suite Permit an easy migration from the current StarOffice 5.2 mail, news, schedule, and schedule server system
installation	Download: no product yet available official installation guide: no product yet available
functionality	*data exchange*: There is no current plan to support palm synchronisation: See the faq link above see also http://xml.openoffice.org/xmerge/index.html
reliability of support	OpenOffice.org can't use the StarOffice 5.2 Schedule programs because they were 3rd party programs. So the people are working on integration of calendar functionality into their product, but this project is just in incubator state. They are searching what opensource programs (e.g. Mozilla Calendar) they can use, and see if they have to write an own calendaring server.
contra	The groupware project is in incubator state.
conclusion	**this project is in a too early stage**
decision	**experimental**
ORACLE COLLABORATION	**Oracle Collaboration Suite**
Home	http://www.oracle.com/ip/deploy/cs/index.html?content.html
other links	http://www.steltor.com/products/
category	**webbased & Outlook**, server in combination with windows or linux

	client: Outlook, webbased server: runs also on Linux, based on the 9i Application Server and the 9i Database
licence/costs	costs: $60 per named user
functionality	• *data exchange*: PDA Support; Exchange and Outlook compatibility and support
reliability of support	On June 21 2002, Oracle bought the time-management software company Steltor (known for their CorporateTime product). Oracle also uses the experiences from their product Oracle Office which was not successful.
contra	no fixed shipping date (perhaps end of 2002)
conclusion	**experimental**
decision	**experimental**
PHPCOLLAB	**phpcollab**
Home	http://sourceforge.net/projects/phpcollab
category	**no calendar**, webbased collaboration software, no calendar
conclusion	**no group calendar**
decision	**not suitable: no group calendar included**
PHPGROUPWARE	**phpGroupWare 0.9.14.000**
Home	http://www.phpgroupware.org
other links	http://savannah.gnu.org/projects/phpgroupware/ http://mail.gnu.org/pipermail/phpgroupware-developers/2002-August/002146.html http://docs.axisgroupware.org/index.php?page=AxisSync
category	**webbased**, php; provides API client: webbased server: php, sql (e.g. mysql)
licence/costs	server: opensource (GNU General Public License (GPL)); client: nearly every web browser is suitable costs: nothing
documentation	in the manual app
marketing statement	phpGroupWare is becoming a top intranet/groupware tool and application framework. Our goal is to provide companies/universities/individuals with a complete, secure and flexible web based groupware framework for building their own custom applications. We are focused on creating a Free Software groupware API with the powerful programability of Lotus Notes.
installation	Online Demo: not available at the moment; http://www.phpgroupware.org/demo/login.php User/password: ??? Download: http://www.phpgroupware.org/index.php?page_name=downloads official installation guide: http://docs.phpgroupware.org/12-docs/html/admin/

functionality	• *calendar*: can invite users without a problem; no invitation of external contacts conflicts are detected automatically, conflict can be ignored private appointments grant access on your calendar to a whole group, or specific users invited persons can accept or reject or mark as tentative It seems that there is already a Windows client that is able to check in given intervals if there is an alarm or change to the calendar. Alarms can also be sent by mail. • *resource booking*: no integration of resources; need to be simulated by special users that belong e.g. to a group "resources" • *email*: works with pop3/smtp • *other modules*: addressbook, admin, backup, bookmarks, calendar, contacts, chat, view cvs, ldap browser, email, filemanager, forum, ftp client, image editor, CRM, Manual, notes, polls, projects, todo, weather, ... • *data exchange*: SyncML support is under development; AxisGroupware also works on PDA sync; can import / export vcs/ics calendar files
reliability of support	5 Admins. 27 developers
pro	uses all standards (vcard, vcalendar) SyncML support clean modular software design messages about changes to ones calendar can be sent via email
contra	version 1.0 is some months in the future how to synchronise with outlook? CSV import/export would need to be programmed not designed for use on enterprise level: see http://docs.axisgroupware.org/index.php?page=TheFutureOFp hpGroupware; but if OM only uses the phpGroupWare calendar application and only a maximum of 50 workstations is logged in to the server at the same time, it should be OK.
conclusion	**depends on AxisSync (Palm synchronisation support), and on CSV implementation**
decision	**could be suitable / depends on other solution**
PHPORGASYNC	**phpOrgaSync**
Home	http://www.ds-services.de/portal/
category	**sync**, synchronisation of Outlook and PHProjekt

	client: Windows Programm (uses ODBC connection to server database) server: phprojekt
licence/costs	freeware, not opensource costs: nothing
marketing statement	This tool enables the user to synchronize his contacts and his schedule held in outlook with the open source groupware tool PHProjekt.
installation	Download: http://www.ds-services.de/portal/
functionality	• *data exchange*: synchronizes addresses & calendar
reliability of support	1 person, no opensource
contra	poor reliability
conclusion	**not enough reliability, project depends on only one developer**
decision	**not enough reliability of support**
PHPROJEKT	**PHProjekt 3.3**
Home	http://www.phprojekt.com
other links	http://sourceforge.net/projects/phprojekt/
category	**webbased**, php client: webbased (frame capable browser, javascript) server: php, sql (e.g. mysql)
licence/costs	server: opensource (GNU General Public License (GPL)); client: nearly every web browser is suitable costs: nothing
documentation	online help available
marketing statement	PHProjekt - an open source groupware suite PHProjekt is a modular application for the coordination of group activities and to share informations and document via intranet and internet. "of course - open source" - PHProjekt is free software.
installation	Online Demo: http://www.phprojekt.com/demo/index.php User/password: english/english Download: http://www.phprojekt.com/modules.php?op=modload&name=Downloads&file=index official installation guide: in the phprojekt directory, file install

functionality	• *calendar*:
	   • *invitation*: can invite one external contact can invite profiles, and can invite group members;
	   groupview possible, but outside of invitation
	   • *private appointments*: normal, private, public
	   • *repeating events*: once, daily, weekly, monthly, yearly
	   • *resource booking*: one resource can be booked

	for a meeting • *email*: eMail works with existing smtp servers. receiving email uses imap The OM sendmail functionality is supported. • *other modules*: Group calendar, project management, time card system, file management, contact manager, mail client and 9 other modules ... • *data exchange*: can import Contacts from vCard, Outlook, other list export of contacts via vCard no support of calendar data import/export synchronisation with Outlook with freeware (not opensource tool) phpOrgaSync
reliability of support	the project started in 2000, one person (Albrecht Guenther) is responsible for it. There are a lot of bugfixes. The program seems to have a big user community, it is translated in 25 languages. Several people have developed addons for the project.
pro	integrated resource booking chief status to modify someone else's calendar
contra	calendar: can only invite one person; no invitation of group members
conclusion	**depends on synchronisation of calendar information with outlook / Palm**
decision	**missing synchronisation support**
PROIEKTOR	**Proiektor**
Home	http://demo.iuveno-net.de/iuveno/Products/Proiektor
category	**webbased**, python
licence/costs	opensource
installation	Online Demo: http://imail.iuveno-net.de:8180/Proiektor/Instances/Demo User/password: demo/demodemo
functionality	• *data exchange*: no palm sync
reliability of support	there seems to be nobody is supporting it / working on it (see nearly empty mailing list archive)
decision	**not enough reliability of support**
SAMSUNG CONTACT	**Samsung Contact 7.1**
Home	http://www.samsungcontact.com/en/
category	**webbased & Outlook**, server with client: webbased and outlook client: outlook, webclient, java client server: redhat linux is supported
licence/costs	commercial costs: 100 users, $62.05 each (some clients need also extra licenses)

documentation	locally saved, register and download
installation	Download: register and download official installation guide: see local version in manual/samsung (or register and download from website)
functionality	• *data exchange*: outlook offline folder synchronisation
reliability of support	formerly HP OpenMail, was bought by Fujitsu, which sold it to Samsung. Samsung use the program in their own company. There was a new release of the MAPI connector in October.
decision	**need to be considered concerning the costs**
SCHEDULEONLINE	**ScheduleOnline Intranet**
Home	http://www.scheduleonline.com/products_intranet.php?pn_id=
category	**webbased & Outlook** client: webbased server: linux apache, windows
licence/costs	commercial costs: $44.50 per user license
installation	Online Demo: http://www.scheduleonline.com/products_intranet.php?pn_id= official installation guide: http://www.scheduleonline.com/Install.mysql.unix.txt
functionality	• *calendar*: distinguishes between meeting, appointment, resource, task, reminder, chat; resource management also integrated in meeting; allows attachments for meetings conflict management; detailed recurrence • *other modules*: forum, contacts, email, todo list, file manager, bookmarks • *data exchange*: Outlook sync; Palm sync
reliability of support	The company uses this software to provide a calendar service on the internet.
decision	**need to be considered concerning the costs**
SKYRIX	**SKYRiX Web Groupware**
Home	http://www.skyrix.com/
category	**webbased** client: webbased server: linux, requires Database: FrontBase, Oracle, Sybase
licence/costs	costs: Unlimited Edition 980 EUR
marketing statement	SKYRiX Web Groupware is web and data base based groupware solution for enterprises of every type and size. It connects classical groupware functionality with the advantages of the innovative technology of a web-based application server. A browser is sufficient in order to communicate with coworkers, customers, partners, suppliers and investors in the whole world, to work on common projects,

	to access the same documents and data, and to process and administer these platform-independently both enterprise-internally and externally.
installation	Online Demo: http://www.skyrix.com/en/demo/standard/index.xhtml
functionality	• *data exchange*: no offline client; no synchronisation with Outlook
decision	**missing synchronisation support**
SMARTCAL	**www.smartcal.com**
Home	http://www.smartcal.com
category	**service provider**
conclusion	**There is an enterprise version for intranets, but without further documentation**
decision	**not suitable: no linux server support**
SQWEBMAIL	**Courier MTA / SqWebMail 3.3.7**
Home	http://www.inter7.com/sqwebmail/
other links	http://www.courier-mta.org http://sourceforge.net/projects/courier/
category	**webbased**, webbased/CGI client: web browser server: program using CGI
licence/costs	opensource (GNU General Public License (GPL)) costs: none
documentation	see in sourcecode, pcp/README.html, some description of the calendar
marketing statement	SqWebMail is a web CGI client for sending and receiving E-mail using Maildir mailboxes. Calendaring services can be optionally enabled by the system administrator. When enabled, basic calendar management is available. NOTE: this is an experimental beta feature. This is the same webmail server that's included in the Courier mail server, but packaged independently The Courier mail transfer agent (MTA) is an integrated mail/groupware server based on open commodity protocols such as ESMTP, IMAP, POP3, LDAP, SSL, and HTTP. Courier provides ESMTP, IMAP, POP3, Webmail, calendaring, and mailing list services.
installation	Online Demo: http://webmail.inter7.com/ User/password: webmail@webmail.com/webmail Download: http://sourceforge.net/projects/courier/ http://ftp.die.net/apt/redhat/7.3/en/i386/RPMS.extra/
functionality	• *calendar*: group calendaring is experimental. • *data exchange*: no palm synchronisation; not even support of vCalendar standards

reliability of support	one person is developing it (Sam Varshavchik). average of 1 release per month (from August 2001 to August 2002) A lot of people seem to use it, the users mailing list archive on Sourceforge contains more than 13000 messages.
contra	groupware calendaring is just experimental no use of standards like vCalendar, but own Personal Calendar Protocol
conclusion	**experimental, just one developer** **no support of palm synchronisation**
decision	**missing synchronisation support**
SUN CALENDAR	**Sun Calendar Server**
Home	http://wwws.sun.com/software/products/calendar_srvr/home_calendar.html
other links	http://wwws.sun.com/software/product_categories/email_calendar_collaboration.html http://wwws.sun.com/software/products/calendar_srvr/faqs_calendar.html
category	**non linux server**, Sparc/Solaris, HP-UX, or Windows server, client webbased client: webbased server: runs on SPARC machines with Solaris, Windows NT 4.0, HP-UX 11.00
licence/costs	commercial costs: $30.00 per installed Client
installation	Download: http://wwws.sun.com/software/download/email_cal.html
functionality	• *data exchange*: no synchronisation with PDAs available at the moment. promised by Sun to be available soon. user needs to be connected to the network, because the calendor is not stored on the PC. Future: They haven't planned anything, they hope everyone will be connected to the net. Sun ONE Calendar Server can import and export calendars in iCalendar format.
reliability of support	This product was formerly called iPlanet Calendar Server.
conclusion	**This server will not run on a Linux machine.**
decision	**not suitable: no linux server support**
SUN STAROFFICE	**Schedule and Schedule Server - StarOffice 5.2**
Home	http://wwws.sun.com/software/star/staroffice/5.2/whatsnew/schedule.html
other links	http://wwws.sun.com/software/star/staroffice/5.2/details.html#7 http://groupware.openoffice.org http://wwws.sun.com/software/star/staroffice/6.0/whatsdifferent.html

category	**outdated**, client/server solution
licence/**costs**	costs: StarOffice 5.2 was free
marketing statement	Coordinate Schedules To Improve Collaboration Keep track of your time and tasks with this full-featured information manager. Store, access, and organize your scheduling and communication information quickly and easily on your desktop or your Palm OS-based personal computing device.
functionality	• *data exchange*: synchronisation with PalmOS is available
reliability of support	Sun offered StarOffice 5.2 for free download, but when they released StarOffice 6.0, this free download was stopped. You still can get StarOffice 5.2, but you have to order it on CD. StarOffice 6.0 comes without the Schedule Client and Schedule Server and has no replacement: http://wwws.sun.com/software/star/staroffice/6.0/migrationfaq.html#1q22 OpenOffice.org also can't use the Schedule programs because they were 3rd party programs. So the people are working on integration of calendar functionality into their product, but this project OpenOffice.org Groupware is just in incubator state. They are searching what opensource programs they can use, and see if they have to write an own calendaring server. They are not planning to support palm synchronisation: http://www.openoffice.org/files/documents/81/7727/groupwarefaq.htm see also http://xml.openoffice.org/xmerge/index.html
contra	Star Office 5.2 is not uptodate; StarOffice 6 doesn't contain the schedule application; OpenOffice is still working on this functionality.
conclusion	**not available any more**
decision	**outdated**
SUPERCALENDAR	**www.supercalendar.com**
Home	http://www.supercalendar.com
category	**service provider**
decision	**not suitable: no linux server support**
SUSE GROUPWARE	**SuSE Linux Groupware Server with Lotus Domino**
Home	http://www.suse.com/us/business/products/suse_business/groupware_server/index.html
category	**webbased & client**, commercial server and client software
licence/costs	costs: see http://www.suse.com/us/business/products/suse_business/gro

	upware_server/prices.html
marketing statement	SuSE Linux Groupware Server makes use of the enterprise operating system SuSE Linux Enterprise Server 7 with the new Linux kernel 2.4.18. Lotus Application Server 5.0.10 provides efficient tools for • Scheduling • Defining workflows • Workflow management • Messaging/e-mail server • Document management • Web server/web application server
functionality	• *data exchange*: EasySync Pro: Palm synchronisation software
reliability of support	see also SuSE Linux OpenExchange Server: http://www.suse.com/us/company/press/press_releases/archive02/openexchange.html; it is announced to be available in mid of November
conclusion	**overkill of functionality**
decision	**too difficult to integrate with current OM solution**
SUSE MAIL	**Suse Mail Server 3.1**
Home	http://www.suse.com/us/business/products/suse_business/email_server/index.html
other links	http://www.skyrix.com/en/products/packages/emailserver3/index.xhtml
category	**webbased & Outlook** client: webinterface based on Skyrix; Outlook supported server: linux based; Skyrix
licence/costs	commercial; Yast Lisence costs: US $999.00
installation	Online Demo: http://emailserver3.suse.de/perl/login.pl?doit=login&lang=EN
functionality	• *calendar*: Internet Free Busy times (ftp/http) • *data exchange*: palm support
decision	**too difficult to integrate with current OM solution**
SYNCBUILDER	**SyncBuilder Java Framework**
Home	http://sourceforge.net/projects/syncbuilder/
category	**sync**, framework for synchronisation with palm
licence/costs	opensource (GNU General Public License (GPL)) costs: framework is free, but development of own software costs (at least time)
marketing statement	The SyncBuilder framework allows you to develop Java applications that can connect with Palm devices. It works independently from Palm Computing, Inc.'s HotSync technology and is not bound to a specific platform (some functionality currently requires native support code, that is only available for Unix and Win32).

	Its functionality includes the ability to read and write records from/to the Palm device, and to query information on the Palm device. It also offers special support for the built-in applications (Calendar, To-Do, Address book, Mail, Expenses, MemoPad). Finally, it can read and write .PDB and .PRC database files. It supports both the normal sync through a serial cable as well as the Network HotSync technology. SyncBuilder is a framework that allows you to develop synchronisation applications between a host machine running Java 1.1 or higher, and a Palm device. It is a complete replacement for Palm's HotSync Manager.
reliability of support	no community (last available version 1999)
contra	requires development of own software no current version available on the net, nobody's working on it (using it?)
conclusion	**no community, requires own development**
decision	**not enough reliability of support**
TEAMWARE OFFICE	**Teamware Office 5.3 Ed 4 for Linux**
Home	http://www.teamware.net/Resource.phx/twoffice/index.htx
category	**webbased & client**, webbased or windows client with linux or windows server client: webbased or windows program server: linux or windows server
licence/costs	commercial costs: no information online; see http://www.teamware.net/Resource.phx/uk/index.htx for addresses
documentation	http://www.teamware.net/dman/Folder.phx/download/office-for-linux/documents
marketing statement	Teamware Office is a complete, modular set of collaborative applications. It provides a robust and scaleable messaging environment combined with active information sharing, document management, and resource management functions. Teamware Office is a secure bridge between the merging network technologies. It enables teams to share and manage information in corporate intranets, supporting mobile and virtual office concepts, full Web integration, network security, corporate and public directories and connectivity for business applications. ... Teamware Calendar is a flexible time management tool that helps people organize group meetings and book shared resources like meeting rooms or common equipment. Several calendars and calendar groups can be viewed across the enterprise in a single intuitive user interface.
installation	Online Demo: http://office.teamware.com:8008/

	User/password: John/john Download: http://www.teamware.net/Resource.phx/download/index.htx official installation guide: http://www.teamware.net/dman/Folder.phx/download/office-for-linux/documents
functionality	• *calendar*: good integration of resource management can invite people and groups; several states for invited persons to reply repeating events, possibility of using several days • *email*: should work with existing sendmail configuration no offline client support • *other modules*: mail, calendar, forum, library (document management); notifier for notices • *data exchange*: SMS Connector: built-in PDA Calendar synchronisation
reliability of support	The Teamware Company belongs to Fujitsu. TeamWare Office is not the only product of this company.
pro	pda support for calendars
conclusion	**offline client support is missing**
decision	**missing synchronisation support**
TUTOS	**Tutos 1.0**
Home	http://www.tutos.org
other links	http://sourceforge.net/projects/tutos
category	**webbased**, php client: web browser server: php/mysql
licence/costs	opensource (GNU General Public License (GPL)) costs: nothing
documentation	http://www.tutos.org/documentation/book0/index.html
marketing statement	TUTOS is the ultimate team organization software, a webbased groupware or ERP/CRM system to manage events/calendars, addresses, teams, projects, tasks, bugs, mailboxes, documents and your time spent with these things.
installation	Online Demo: http://www.tutos.org/php/mytutos.php User/password: linus/guest Download: http://www.tutos.org/homepage/download.html official installation guide: http://www.tutos.org/homepage/install.html
functionality	• *calendar*: invite persons and / or groups; invite per mail book one or more resources for an appointment participants can be allowed to change details of the appointment repeating events • *email*: works with existing smtp and pop3 server as well as directly through sendmail • *other modules*: calendar, address manager, bug

	tracking system, product/project repository (with task management, with document management, with installation management), support of different roles (manager,developer,customer etc.) support of relations to other projects (base product, subproject etc.) mailboxes (imap/pop) timetracking on projects, installations and bugs Invoices Watchlists. Stay informed on changes on projects/bugs etc via email. support for teams that are distributed over different timezones color schemes fine grained permission handling change history • *data exchange*: nothing done about palm synchronisation yet; can export iCal files, but no import no import / export of vCard
reliability of support	9 Developers, 1 Admin
pro	developed with a lot of details (e.g. everywhere are links, for example click on a participant, and you get to his details) a good documentation
contra	no palm synchronisation yet
conclusion	**palm and outlook synchronisation is not available**
decision	**missing synchronisation support**
TWIG	**TWIG 2.7.6**
Home	http://twig.screwdriver.net/about.php3
category	**webbased**, php client: web browser server: php
licence/costs	opensource (GNU General Public License (GPL)) costs: nothing
documentation	not available
marketing statement	TWIG is rapidly becoming a very useful intranet/groupware tool and application framework. It is written and implemented completely with the PHP programming language. Our intention is that it become a simple, cross-platform, fast, and browser-independent way to access or share almost any kind of information, without the complexity or costs of other intranet/groupware packages.
installation	Online Demo: http://twig.screwdriver.net/demo/ User/password: test/test Download: http://twig.screwdriver.net/download.php3

functionality	• *calendar*:
	    • *invitation*: an appointment is always assigned to a group; no possibility to invite single persons user can't specify a group to invite for a meeting
	    • *private appointments*: not possible
	    • *repeating events*: strange solution of repeating events: first save the event, then edit the data concerning the recurrence. But very detailed:
	• *email*: just IMAP support
	• *data exchange*: no palm synchronisation can import contacts from Outlook, Netscape, Pegasus no export/import of calendar data

contra	no possibility for palm / outlook synchronisation
conclusion	**no synchronisation**
decision	**missing synchronisation support**
UW	**UW Calendar Information Center 1.0.1**
Home	http://www.washington.edu/ucal/
category	**webbased**
licence/costs	opensource
documentation	http://www.washington.edu/ucal/require.html
functionality	• *calendar*: group calendaring, with realtime server, using standards • *email*: used for invitations • *data exchange*: planned, but not realised yet
reliability of support	about 6 people working on it at the University of Washington
conclusion	**under development (no group calendaring available yet, only personal calendar and public events work at the moment)**
decision	**experimental**
WEBEVENT	**WebEvent Team**
Home	http://www.webevent.com/products/team/
category	**webbased & Outlook** server: requires perl program, mysql
licence/costs	costs: $1,200 for a 10-calendar/user license including Sync; Educational institutions and 501(c)(3) non-profit organizations are eligible for a 25% discount on WebEvent licenses
installation	Online Demo: http://demo.webevent.com/public/team/cgi/webevent.cgi
functionality	• *data exchange*: outlook & Palm sync
conclusion	**there were already charity prices on the website, so there was no need to send a special request**
decision	**even with discounts too expensive for OM**
XENTRIX	**intisoft Xentric groupware server**
Home	http://www.intisoft.com/index.phtml?scp=soln_groupware

category	**webbased**, based on Xentric XML Server server: no statement about server
licence/costs	commercial
functionality	• *data exchange*: supports WAP-Browsers, other sync is planned (SyncML, Palm)
conclusion	**no offline sync, or palm sync yet**
decision	**missing synchronisation support**
XIMIAN EVOLUTION	**GNOME Ximian Evolution 1.0**
Home	http://www.ximian.com/products/ximian_evolution/index.html
category	**client**, linux groupware client client: linux client, GNOME server: either Exchange, or peer-to-peer
licence/costs	Ximian Evolution is opensource (GPL), Ximian Connector is commercial costs: Ximian Connector: 25 clients = $1449
marketing statement	Ximian Evolution is the premier personal and workgroup information management solution for Linux and UNIX. The software seamlessly integrates email, calendaring, meeting scheduling, contact management and online task lists in one powerful, fast, and easy to use application. It supports peer-to-peer calendaring with users on products that support iCalendar, including Microsoft Exchange and Lotus Notes. Ximian Connector is an optional add-in to Ximian Evolution that is now available for purchase. With Ximian Connector installed, Ximian Evolution functions as a Microsoft Exchange 2000 client, seamlessly integrated with Exchange calendaring and other mail storage and mail handling features.
installation	official installation guide: installed with Gnome

functionality	• *calendar*: It supports peer-to-peer scheduling and allows users to create and confirm group meetings online, even with Microsoft Outlook and Lotus Notes users.
	    • *invitation*: see schedule (free/busy) of others
	    • *repeating events*: even custom recurrance is supported; exceptions from the recurrance
	    • *resource booking*: can assign a type (individual, resource, room) to every attendee
	• *data exchange*: Ximian Evolution can synchronize calendar, address, and task list information with PalmOS devices.

conclusion	**could be a good program for users that don't use Windows clients.**
decision	**not suitable: no windows workstation support**
YAHOO	**www.calendar.yahoo.com**

CALENDAR	
Home	http://www.calendar.yahoo.com
category	**service provider**
decision	**not suitable: no linux server support**

7.3 Installation Hints for Groupware Applications on the OM Standard Linux Server

Installation of BILL Workgroup Server

- python 1.5.2 is already installed.
- python dev needs to be installed.
- postgresql needs to be installed:

rpm -i postgresql-libs-7.2.1-2PGDG.i386.rpm postgresql-7.2.1-2PGDG.i386.rpm
rpm -i postgresql-server
/etc/rc.d/init.d/postgresql start
sample conf file : /usr/share/pgsql/pg_hba.conf.sample

conf file: /var/lib/pgsql/data/pg_hba.conf; content see billworkgroup doc
plus: host all 127.0.0.1 255.0.0.0 trust postgres

/var/lib/pgsql/data/postgresql.conf for setting up the tcp options, and port 5432
install pygresql and mx

omniorb and omniorb-python

do something about etc/hosts localhost?
omniNames -start

mailer: there are some ^M at the end of some line in the mailer msgstore init

start everything:
/etc/rc.d/init.d/postgresql start
omniNames&
cd /usr/local/BILL-StorageServer-2.1.3
python Server.pyc
Problem: Segmentation fault

problems: Segmentation fault when trying to start the server

Installation of BrownBear Calcium Web Calendar Server and TripleSync

httpd.conf:
perl.c
perl.so
alias: path to cgi-bin
directory the same

change cgi-bin
/usr/local/Calcium-3.7
/usr/local/groupware/cgi-bin/Calcium37.pl

Installation of BSCW

extracted the bscw.tar.gz to /home/timop
run as timop:
~/bscw-4.0.6$python setup
append apache.conf to httpd.conf
modify src/config.py:
SERVER_ROOT: http://192.168.1.200/bscw
SERVER_ADMIN: timop@c.ict.om.org
SERVER_ADMINS: 'admin'
SMTP_HOST
don't think I needed have done that: create server/data/htpasswd; htpasswd -c
htpasswd username
access problems bscw
as root: chmod a+w /usr/local/groupware
~/bscw-4.0.6$python setup /usr/local/groupware/bscw
again add apache.conf
create htpasswd

http://bscw.gmd.de/faq/inst20.html
Why do I get a Python trace with "RuntimeError: cgi/bscw.cgi: No setgid"?

got an account on http://bscw.gmd.de/bscw/bscw.cgi

cd <bscw-path>
% cc -o wrapper wrapper.c
as bscw user (timop): % ./bsadmin chkconfig
this brought some error messages about group id and mode
as root: chgrp 509 cgi/bscw.cgi
as root: chmod 06755 cgi/bscw.cgi
as bscw user (timop): % ./bsadmin chkconfig

stop server: <bscw-path>/start_servers -k
start server: <bscw-path>/start_servers

configure garbage collection!

http://192.168.1.200/pub/bscw.cgi?op=rmail
problems with sending
instead of that:
as user timop:
cd bscw
./bsadmin register -r user@c.ict.om.org user test62
login with http://192.168.1.200/bscw/bscw.cgi

Installation of Bynari Insight Server

changed /etc/xinetd.d/ipop3; disable = yes
kill sendmail
setup

problem: sendmail is in bootup before insight

possible solution:
put xinetd.d/ipop3 in order again
/etc/rc.d/rc.local:
remove /usr/bin/insightserver start
create script in init.d: insightserver
with the line
/usr/bin/insightserver $1
cd /etc/rc.d/rc3.d
ln ../init.d/insightserver S79insightserver
79 because of sendmail having 80

Installation of Meeting Maker

mkdir /usr/local/mm
./mmadminstall
/usr/local/mm
mmadmin requires RedHat 7.1 and X11

see also Admin/ch4__sys/creating.html

Installation of Microsoft Outlook

see http://www.tldp.org/HOWTO/mini/Apache+SSL+PHP+fp-2.html;
see the public Outlook calendaring newsgroup:
http://groups.google.com/groups?hl=en&lr=&ie=UTF-8&oe=UTF-8&group=microsoft.public.outlook.calendaring

Web Publishing Wizard (IE 4.0 or higher) for Windows 95 & NT 4.0
see
http://www.microsoft.com/downloads/release.asp?releaseid=22658&area=search&ordinal=2
see http://download.microsoft.com/download/win95/Utility/1.52/W97/EN-US/wpie415-x86.exe
see
http://www.microsoft.com/windowsxp/home/using/productdoc/en/default.asp?url=/windowsxp/home/using/productdoc/en/webpub_overview.asp
Microsoft FrontPage 2000 Server Extensions SR1.2: Downloads for UNIX-Based
Servers: http://msdn.microsoft.com/library/default.asp?url=/library/en-us/dnservext/html/unixfpse.asp
FP2000: Installing FrontPage Server Extensions to an Apache Web Server:
http://support.microsoft.com/default.aspx?scid=KB;EN-US;Q202198&LNG=ENG&SA=ALLKB
improved mod_frontpage: http://sourceforge.net/projects/mirfak/ and
http://home.edo.uni-dortmund.de/~chripo/index.html tar xvfz fp40.linux.tar.gz
gzip -d fp40.linux.tar.gz
cd frontpage/version4.0
mv ../../fp40.linux.tar ./
./fp_install.sh

he installs the frontpage extensions to /usr/local/frontpage
server config filename: /etc/httpd/conf/httpd.conf

frontpage/version4.0/bin/fpsrvadm.exe
create a subweb
web name: freebusy
port: 80
Unix username: timop
unix group: users

./change_server
/usr/sbin/httpd
download a most recent patch from http://www.rtr.com/fpsupport/index.asp rpm -i
apache-1.3.20-16.src.rpm
cd /usr/src/redhat/SOURCES
tar xvzf apache_1.3.20.tar.gz
cd apache_1.3.20/src
patch < /home/timop/download/fp-patch-apache_1.3.22

Installation of moreGroupware

installed:
rpm -i apache-1.3.20-16.i386.rpm mm-1.1.3-1.i386.rpm
rpm -i php-4.0.6-7.i386.rpm curl-7.8-1.i386.rpm pspell-0.12.2-3.i386.rpm
libtool-libs-1.4-8.i386.rpm
rpm -i php-mysql-4.0.6-7.i386.rpm
rpm -i mysql-3.23.41-1.i386.rpm
rpm -i mysqlclient9-3.23.22-6.i386.rpm
rpm -i mysql-server-3.23.41-1.i386.rpm

/etc/rc.d/rc3.d: mv K..mysqld S99mysqld, same for httpd

changes to /etc/httpd/conf/httpd.conf:
DocumentRoot "/usr/local/moregroupware"
<Directory> "/usr/local/moregroupware"
AddType application/x-httpd-php .php
LoadModule php4_module libexec/libphp4.so
AddModule mod_php4.c

remove all other httpd / apache installations

httpd restart

see moregroupware/docs/INSTALL

/etc/rc.d/init.d/mysqld start
no password for user root at the moment
mysqladmin -u root -p create moregroupware
shell> mysql --user=root mysql

mysql> GRANT ALL PRIVILEGES ON *.* TO moregw@localhost IDENTIFIED BY 'test62' WITH GRANT OPTION;

/etc/om-firewall.d/S62.internal.local and uncomment the services you wish to use. Example for http and https:
ipchains -A int-in -j ACCEPT -p tcp --dport http

http://192.168.1.200/
run the setup
mysql: server: localhost, user: moregw, database: moregroupware

possibility to hide all other modules but the calendar from user.
Problems with calendaring:
- inviting other people does not work

Good: can see the free/busy time of invited persons;
private appointments

solved one problem with webmail, by copying the config.inc.php
from moregroupware to moregroupware/modules:
there was a warning / error in modules/wmail/sendmsg.php

changed moregroupware/modules/wmail/inc/class.html.mime.mail.inc: Line 14:
require_once('inc/mimePart.php');

try to install redhat 6.3 php:
rpm -i db3-3.3.11-6.i386.rpm
rpm --upgrade libtool-libs-1.4.2-7.i386.rpm
rpm --upgrade php-4.1.2-7.i386.rpm curl-7.9.5-2.i386.rpm php-mysql-4.1.2-7.i386.rpm

problem: only users belonging to default group can login???
passwords for the local test version: admin admin

problems: can choose participants when creating a meeting, but there are no changes to the participant's calendars; participants are not saved with the meeting
webmail interface could not be tested because of errors
the initiator of a meeting is always a participant

Installation of Mozilla Calendar

first download Mozilla 1.2, then visit the homepage of the calendar, and you can start the installation of mozilla calendar on the webpage
problems: Free Busy Calendaring gives a error message, does not work at the moment

Installation of phpGroupWare

Modifications to a Standard OM Linux Server 1.2, which is based on RedHat 7.2:

extract phpgroupware-0.9.14.000.tar.gz to /usr/local/phpgroupware:
cd /usr/local
tar xvzf phpgroupware-0.9.14.000.tar.gz

rpm -i apache-1.3.20-16.i386.rpm mm-1.1.3-1.i386.rpm
rpm -i php-4.0.6-7.i386.rpm curl-7.8-1.i386.rpm pspell-0.12.2-3.i386.rpm libtool-libs-1.4-8.i386.rpm

or better use postgresql? axissync requires postgresql
rpm -i php-mysql-4.0.6-7.i386.rpm mysql-3.23.41-1.i386.rpm mysqlclient9-3.23.22-6.i386.rpm mysql-server-3.23.41-1.i386.rpm

/etc/rc.d/rc3.d: mv K..mysqld S99mysqld, same for httpd

changes to /etc/httpd/conf/httpd.conf:
DocumentRoot "/usr/local/phpgroupware"
<Directory> "/usr/local/phpgroupware"
AddType application/x-httpd-php .php
LoadModule php4_module libexec/libphp4.so
AddModule mod_php4.c

/etc/init.d/httpd start

Database configuration:
mysql:
/etc/rc.d/init.d/mysqld start
no password for user root at the moment
mysqladmin -u root -p create phpgroupware
shell> mysql --user=root mysql
mysql> GRANT ALL PRIVILEGES ON *.* TO gw@localhost IDENTIFIED BY 'test62' WITH GRANT OPTION;

postgresql:
rpm -i postgresql-7.1.3-2.i386.rpm postgresql-libs...rpm postgresql-server...rpm php-pqsql-4.0.6-7.i386.rpm
su postgres
postmaster -D /var/lib/pgsql/data

/etc/rc.d/rc3.d: mv K..postgresql S99postgresql

(As Linux user postgres)
createuser gw --createdb --pwprompt --no-adduser
createdb phpgroupware -U gw

so that header can be written:
chmod a+w /usr/local/groupware/phpgroupware

/etc/om-firewall.d/S62.internal.local and uncomment the services you wish to use.
Example for http and https:

ipchains -A int-in -j ACCEPT -p tcp --dport http

from the client's windows explorer:
http://192.168.1.200/setup/index.php

in configuration:
192.168.1.200
use /tmp

test server admin login: admin /test62

Installation of PHProjekt

for configuration of apache and sql see the installation infos with phpgroupware

create database:
mysqladmin -u root -p create phprojekt

admin login:
http://192.168.1.200/phprojekt/setup.php
root, root

The admin has to create the resource.
there are several profiles: one to configure via contacts, the other configured in
options: group members / external

Installation of Samsung Contact

problems with / partition, it was too small
omwizard (OpenMail = OM) didn't work: wish could not be found
(as root: rpm -Uvh omwizard-7.1.0-1.noarch.rpm)

Manual installation:
as root: rpm -i contact-core-7.1.0-1.i386.rpm contact-lang-British-7.1.0-1.i386.rpm

add to /etc/profile:
PATH=$PATH:/opt/openmail/bin:/opt/openmail/adm
export PATH

ommakeom
/opt/openmail/bin/omaddmn -m test -D c.ict.om.org
omaddfb
/etc/init.d/openmail start

perl and apache:
httpd.conf:
perl.c
perl.so
alias: path to cgi-bin
directory the same

cp /opt/openmail/access/omsession.cgi /usr/local/groupware/cgi-bin
mkdir -p docsDir/omsession/admin
cp -r /opt/openmail/template/access/apps/admin/HTML docsDir/omsession/admin

docsDir is /usr/local/groupware

user is Unix user, Mailnode test:
omaddu -n "Timotheus Pokorra/test" -a "timop" -c admin -p test62 -u timop -F

F = FreeBusy

see man omaddu

uninstall firewall: rpm -e om-firewall --nodep
uptodate licences? in /var/opt/openmail/Contact-Licences
install Samsung MAPI
create profile that uses samsung
don't choose NOD Mail scanner, but only HP Openmail (with server store)
server: c.ict.om.org
user: timo pokorra (that after the -n)
password: test62
outlook: options: prompt for login name
then in openmail/General/local files: h:\openmail

it works: two accounts, create meeting, messages are sent via email
calendar, properties, permissions, add, find, add, ok

then to read another persons folder, File / open / other users folder

seeing other's free busy times also works, without specific read permissions

offline folders: create new pst file, copy e.g. calendar to this file.

Installation of Teamware Office

rpm -i TeamWAREOffice-5.3.40-10.i386.rpm
for directory other than /to: rpm -i --prefix /usr/local/to TeamWAREOffice-5.3.40-10.i386.rpm
login as user twu
to config

expert mode: NO
setup as directory server: YES
serial number
activation key
name for site: omcalendar
passwords: test6262

install everything but forum and library
to start

to status

/etc/om-firewall.d/S62.internal.local:
add lines:
Teamware office Server listen
ipchains -A int-in -j ACCEPT -p tcp --dport 25000
Teamware office client notifier listen
ipchains -A int-out -j ACCEPT -p tcp --sport 25003

start Teamware Admin on client
login settings, connection, server address: 192.168.1.200
make default
ADMIN test6262
Admin / organization directory or directly the Directory application
entry / new / user template
entry / new / user

INDEX

REFERENCES

Books

BORGHOFF: U. Borghoff, J. Schlichter (2000), Computer Supported Cooperative Work - Introduction to Distributed Applications
Springer, Berlin Heidelberg New York

ELLIS et al.: C. A. Ellis, S. J. Gibbs, G. L. Rein (1991), Groupware - Some Issues and Experiences
Communications of the ACM 34:1, 38-58
quoted in: [BORGHOFF, p. 94]

GREIF: I. Greif (Editor) (1988), Computer-Supported Cooperative Work: A Book of Readings
Morgan Kaufmann, San Mateo, CA
quoted in: [DIVITINI, page 3 slide 4]

GRUDIN: J. Grudin (1994), CSCW: History and Focus
IEEE Computer 27:5, 19-26
quoted in: [BORGHOFF, p. 119]

JOHANSEN: R. Johansen (1988), Groupware: Computer Support for Business Teams
Free Press, New York
quoted in: [BORGHOFF, p. 92]

MSOUTLOOKHELP: (), Microsoft Outlook 2000 Online Help

SEN: S. SEN, E. DURFEE (1991), A Formal Study of Distributed Meeting Scheduling: Preliminary Results
SIGOIS ACM, New York
Proc. Conf. on Organizational Computing Systems, Atlanta, GA., pp. 55-68
quoted in: [BORGHOFF, p. 471]

WILSON: P. Wilson (1991), Computer Supported Cooperative Work
Intellect Books, Oxford, UK
quoted in: [BORGHOFF, p. 92]

Internet Resources

ADELSTEIN: S. Thomas Adelstein (May 14, 2002): InsightServer and other Exchange Alternatives for Outlook
21.10.2002 http://www.bynari.net/Exchange_Replacement01bc3.pdf

ALT-N T: Alt-N Technologies, Ltd. (2002): Calendaring and Group Scheduling with MDaemon 6.0
21.10.2002
http://files.altn.com/MDaemon/White_Papers/CalendaringGroupScheduling.pdf

APPELT: Appelt, W. (2001): What Groupware Functionality Do Users Really Use? Analysis of the Usage of the BSCW System
IEEE Computer Society, Los Alamitos
21.10.2002 http://bscw.gmd.de/Papers/PDP2001/PDP2001.pdf

BRICHACEK: G. Brichacek, S. SouderTurn your Lotus applications into Web services
23.10.2002 https://www6.software.ibm.com/dw/education/ls/lsappws/lsappws-a4.pdf

BUCHMANN: David Buchmann (September 2002): SyncML and its Java Implementation sync4j
08.11.2002 http://sync4j.sourceforge.net/RandD/David%20Buchmann.pdf

BUYENS: Marc Buyens (September 1999): Enterprise Application Integration (EAI)
Xpragma
24.10.2002 http://www.xpragma.com/eai_wp.htm

CAIN: M. Cain (September 2002): Upgrading Exchange
05.11.2002
http://techupdate.zdnet.com/techupdate/stories/main/0,14179,2881896,00.html

CHADWICK: David W. Chadwick (1994, 1996): Understanding X.500 - The Directory
21.10.2002 http://www.isi.salford.ac.uk/staff/dwc/X500.htm

DIFFUSE: Electronic Mail and Newsgroup Protocols
31.10.2002 http://www.diffuse.org/email.html

DIVITINI: M. DivitiniKnowledge transfer
21.10.2002 http://www.idi.ntnu.no/emner/mnfit385/lecture/lecture9-transfer.pdf

DRAFT_IETF_CALSCH_CAP: D. Royer, G. Babics, P. Hill, S. Mansour (November 3, 2002, expires May 4, 2003): Calendar Access Protocol (CAP), Internet Draft, work in progress version 9
Network Working Group
08.11.2002 http://www.ietf.org/internet-drafts/draft-ietf-calsch-cap-09.txt

143

EDMONDS: L. Edmonds, S. Tarachandra (May 31, 1996): Electronic Mail Standards SMTP and X.400
Information Technology Services, National Library of Canada
31.10.2002 http://www.nlc-bnc.ca/9/1/p1-229-e.html

ERFRAKON: (May 2002): Workgroup Lösungen in einer gemischten Windows/Linux Umgebung
erfrakon: Erlewein, Frank, Konold & Partner
21.10.2002 http://www.erfrakon.de/whitepapers/kurzstudie-groupware.pdf

ERICKSON: Kevin Erickson (1999): Linux in Business - Case Study Implementation of Corporate E-mail Solution
21.10.2002 http://www.business-linux.at/case/case.mail.html

GOLD-BERNSTEIN: Beth Gold-BernsteinEAI Market Segmentation
21.10.2002 http://www.eaijournal.com/PDF/Gold-Bernstein.pdf

GRAY: Terry Gray (1993, 1995): Comparing Two Approaches to Remote Mailbox Access: IMAP vs. POP
21.10.2002 http://www.imap.org/imap.vs.pop.html

HANSEN: Mark Hansen (August 2001): Changing Terrain - Open middleware standards are redefining EAI and B2B integration
24.10.2002 http://www.intelligenteai.com/feature/010810/feat1_1.shtml

HULL: A. Hull, A. MayerThe Future of phpGroupware
12.11.2002
http://docs.axisgroupware.org/index.php?page=TheFutureOFphpGroupware

INMON: W. Inmon (1999): A BRIEF HISTORY OF INTEGRATION
31.10.2002 http://www.eaijournal.com/Article.asp?ArticleID=119&DepartmentId=5

JOHNSON_MEAD: C. Johnson, C. Mead (1999): Exchange Replacement HOWTO
21.10.2002 http://www.bynari.net/ExchangeReplacement.pdf

JOHNSTON: Jon Johnston (April 1999): A review of the R5 client from a messaging perspective
06.11.2002 http://www.dominopower.com/issues/issue199904/newreview001.html

KANG: A. Kang (August 2002): Enterprise application integration using J2EE
31.10.2002 http://www.javaworld.com/javaworld/jw-08-2002/jw-0809-eai.html

KLASEN: Norbert Klasen (August 2001): Directory Services for Linux in comparison with Novell NDS and Microsoft Active Directory
21.10.2002 http://www.daasi.de/staff/norbert/thesis/thesis.pdf

KLOECKNER: K. Klöckner (2000): BSCW - Educational Servers and Services on the WWW - How Shared Workspaces support Collaboration in Educational Projects
German National Research Center for Information Technology (GMD), Sankt Augustin
21.10.2002 http://bscw.gmd.de/Papers/CCCC/cccc.pdf

LAVINE: B. Lavine (July 1997): Groupware - Lotus Notes provides an exceptional collaborative work environment
05.11.2002 http://www.computerbits.com/archive/1997/0700/grpware.html

MCCARTHY: Vance McCarthy (June 2002): OMA Eyes Wireless Developers' Architecture By Year's End
01.11.2002 http://www.integrationdevelopernews.com/IntegrationNews.asp?ID=16

MICROSOFT_L: Product Lifecycle Dates - Server Product Family
Microsoft
05.11.2002 http://support.microsoft.com/default.aspx?scid=fh;en-us;LifeSrvr

MICROSOFT_MAPI_F: Platform SDK: MAPI Features
31.10.2002 http://msdn.microsoft.com/library/default.asp?url=/library/en-us/mapi/html/_mapi1book_about_mapi_features.asp

MICROSOFT_MAPI_G: Platform SDK: MAPI Glossary
31.10.2002 http://msdn.microsoft.com/library/?url=/library/en-us/mapi/html/_mapi1book_legal_information_2.asp?frame=true

MICROSOFT_MI: Microsoft Mobile Information Server Product Overview
08.11.2002 http://www.microsoft.com/miserver/evaluation/overview/default.asp

MICROSOFT_TNEF: Platform SDK: MAPI: Transport-Neutral Encapsulation Format (TNEF)
31.10.2002 http://msdn.microsoft.com/library/default.asp?url=/library/en-us/mapi/html/_mapi1book_transport_neutral_encapsulation_format_tnef_.asp

MOELLER: Erik Möller (May 2002): Evolution: Exchanging Microsoft
infoAnarchy
31.10.2002 http://www.infoanarchy.org/story/2002/5/1/64850/34022

OLSEN: E. Olsen, J. Hawkins (May 2002): After 6: What's Coming from IBM Post-Lotus Notes/Domino 6?
Open Standards Advisor
06.11.2002 http://openstandardsadvisor.com/doc/09697

OM: Operation Mobilisation - About Us
21.10.2002 http://www.om.org/us.htm

OMA: (June 2002): Press Release: New Global Organization, the Open Mobile Alliance, Formed ...
01.11.2002 http://www.openmobilealliance.org/pr2002-06-12.html

PABLA: Chandandeep Pabla (April 2002): SyncML intensive: A beginner's look at the SyncML protocol and procedures
31.10.2002 ftp://www6.software.ibm.com/software/developer/library/wi-syncml2.pdf

PALEN: L. Palen (1998): Calendars on the New Frontier: Challenges of Groupware Technology
21.10.2002 http://www.cs.colorado.edu/~palen/dissertation/LPdissertation.pdf

PALM_NETSYNC: Palm Network Hotsync technology
21.10.2002
http://www.palm.com/support/helpnotes/hotsync/networkhsinfo.html#definition

RAYMOND: E. Raymond (2000): The Cathedral and the Bazaar
21.10.2002 http://tuxedo.org/~esr/writings/cathedral-bazaar/cathedral-bazaar/cathedral-bazaar.ps

REEKES: J. Reekes (July 2002): Calendar Management: Product selection based on best practices
21.10.2002 http://www.meetingmaker.com/wp/meetingmaker-wp.pdf

REN: F. Ren (February 2002): The Marketplace of Enterprise Application Integration (EAI)
31.10.2002 http://www.geocities.com/ffren/eai.html

RFC1521_MIME: N. Borenstein, N. Freed (September 1993): MIME (Multipurpose Internet Mail Extensions) Part One; RFC 1521
21.10.2002 http://www.ietf.org/rfc/rfc1521.txt

RFC1725_POP3: J. Myers, M. Rose (November 1994): Post Office Protocol Version 3; RFC 1725
21.10.2002 http://www.ietf.org/rfc/rfc1725.txt

RFC1730_IMAP4: M. Crispin (December 1994): Internet Message Access Protocol - Version 4; RFC 1730
21.10.2002 http://www.ietf.org/rfc/rfc1730.txt

RFC1847_SECURE_MP_MIME: J. Galvin, S. Murphy, S. Crocker, N. Freed (October 1995): Security Multiparts for MIME; RFC 1847
Network Working Group
31.10.2002 http://www.ietf.org/rfc/rfc1847.txt

RFC2425_MIMEDIRECTORY: T. Howes, M. Smith, F. Dawson (September 1998): A MIME Content-Type for Directory Information; RFC 2425
21.10.2002 http://www.ietf.org/rfc/rfc2425.txt

RFC2426_VCARD: F. Dawson, T. Howes (September 1998): vCard MIME Directory Profile; RFC 2426
21.10.2002 http://www.ietf.org/rfc/rfc2426.txt

RFC2445_ICALENDAR: F. Dawson, D. Stenerson (November 1998): Internet Calendaring and Scheduling Core Object Specification (iCalendar); RFC 2445
Network Working Group
21.10.2002 http://www.ietf.org/rfc/rfc2445.txt

RFC2446_ITIP: S. Silverberg, S. Mansour, F. Dawson, R. Hopson (November 1998): iCalendar Transport-Independent Interoperability Protocol (iTIP); RFC 2446
Network Working Group
21.10.2002 http://www.ietf.org/rfc/rfc2446.txt

RFC2447_IMIP: F. Dawson, S. Mansour, S. Silverberg (November 1998): iCalendar Message-Based Interoperability Protocol (iMIP); RFC 2447
Network Working Group
21.10.2002 http://www.ietf.org/rfc/rfc2447.txt

RFC2518_WEBDAV: Y. Goland, E. Whitehead, A. Faizi, S. Carter, D. Jensen (February 1999): HTTP Extensions for Distributed Authoring -- WEBDAV; RFC 2518
21.10.2002 http://www.ietf.org/rfc/rfc2518.txt

RFC2821_SMTP: J. Klensin (April 2001): Simple Mail Transfer Protocol; RFC 2821
21.10.2002 http://www.ietf.org/rfc/rfc2821.txt

RFC2822_IMF: P. Resnick, Editor (April 2001): Internet Message Format RFC 2822
Network Working Group
25.10.2002 http://www.ietf.org/rfc/rfc2822.txt

RFC3080_BEEP: M. Rose (March 2001): The Blocks Extensible Exchange Protocol Core (BEEP); RFC 3080
Network Working Group
31.10.2002 http://www.ietf.org/rfc/rfc3080.txt

RFC3283_ICALGUIDE: B. Mahoney, G. Babics, A. Taler (June 2002): Guide to Internet Calendaring RFC 3283
21.10.2002 http://www.ietf.org/rfc/rfc3283.txt

SCHLICHTER: J. Schlichter (March 2002): Distributed Applications
TU München, Munich
21.10.2002
http://www11.in.tum.de/lehre/lectures/ss2002/va/extension/latex/va_course_student.pdf

SKICAL: G. FitzPatrick, P. Lannerö, N. Hjelm (October 25, 2002, expires April 25, 2003): SkiCal - an extension of iCalendar, Internet Draft, work in progress
CALSCH
28.10.2002 http://www.ietf.org/internet-drafts/draft-many-ical-ski-06.txt

SOAP: (May 2000): Simple Object Access Protocol (SOAP) 1.1
W3C
05.11.2002 http://www.w3.org/TR/SOAP/

STEMBERGER: Scott Stemberger (October 2001): Syncing data - An introduction to SyncML
IBM
21.10.2002 ftp://www6.software.ibm.com/software/developer/library/wi-syncml.pdf

SYNCML_PRODUCTS: SyncML compliant products
01.11.2002 http://www.syncml.org/interop/interop-compliant.html#2

SYNCML_SUPPORT: (November 2001): SyncML Press Release: SyncML Initiative Reaches the Milestone of 500 Supporter Companies ...
01.11.2002 http://www.syncml.org/press_release.asp?id=21

SYNCML_SYNC: (October 2002): SyncML Sync Protocol
21.10.2002
http://www.syncml.org/docs/syncml_sync_protocol_v111_20021002.pdf

SYNCML_TOOLKIT: (August 2002): Press Release: SYNCML INITIATIVE TO MOVE INTEROPERABILITY TOOLKIT TO OPEN SOURCE MODEL
SyncML initiative
21.10.2002 http://www.syncml.org/press_release.asp?id=36

SYNCML_WP: Building an Industry-Wide Mobile Data Synchronization Protocol (SyncML)
SyncML initiative
21.10.2002 http://www.syncml.org/download/whitepaper.pdf

VCARD: (1996): vCard version 2.1 Specification
Versit Consortium
21.10.2002 http://www.imc.org/pdi/pdiproddev.html

XCAL: F. Dawson, S. Reddy, D. Royer, E. Plamondon (July 25, 2002, expires January 23, 2003): iCalendar DTD Document (xCal), Internet Draft, work in progress
Network Working Group
28.10.2002 http://www.ietf.org/internet-drafts/draft-ietf-calsch-many-xcal-02.txt

XML-RPC: Dave Winer (1999): XML-RPC Specification
UserLand Software
05.11.2002 http://www.xml-rpc.com/spec