FAKULTÄT FÜR !NFORMATIK

Faculty of Informatics

A Cooperative Resource Scheduling Tool With Automated Planning

DIPLOMARBEIT

zur Erlangung des akademischen Grades

Diplom-Ingenieur

im Rahmen des Studiums

Wirtschaftsingenieurwesen Informatik

eingereicht von

Thomas Schneider
Matrikelnummer 0326062

an der
Fakultät für Informatik der Technischen Universität Wien

Betreuung
Betreuerin: ao. Univ.-Prof. Mag. Dr. Silvia Miksch
Mitwirkung: DI Dr. Wolfgang Aigner

Wien, 25.10.2011

(Unterschrift Verfasser)

(Unterschrift Betreuerin)

Erklärung zur Verfassung der Arbeit

Thomas Schneider
Kasernenstraße 7a/12
7000 Eisenstadt

‘Hiermit erkläre ich, dass ich diese Arbeit selbständig verfasst habe, dass ich die verwendeten Quellen und Hilfsmittel vollständig angegeben habe und dass ich die Stellen der Arbeit — einschließlich Tabellen, Karten und Abbildungen —, die anderen Werken oder dem Internet im Wortlaut oder dem Sinn nach entnommen sind, auf jeden Fall unter Angabe der Quelle als Entlehnung kenntlich gemacht habe.’

Eisenstadt, 25.10.2011

(Unterschrift Verfasser)

Abstract

Abstract

Assignment of staff to work tasks is a complex problem that involves a large number of factors and requires a lot of expertise. Long term as well as short term requirements need to be met which demands flexible solutions. Software tools can aid planners in reaching optimal dispatching plans but currently available solutions provide only incomplete support. This thesis describes the design, prototypical implementation, and evaluation of a tool called A-Plan for semiautomated assignment planning.

First, established procedures in a selected company were observed and investigated. During this phase, vulnerabilities of the current process and the software systems in use were identified. Applying a user-centered development process, a new tool has been designed and a prototype called A-Plan was implemented. Finally, the prototype has been evaluated by potential users and IT professionals.

In A-Plan, the assignments of staff to customers are displayed visually and can be modified via direct manipulation. This makes required information quickly available to users. Also, a user friendly and efficient search method is implemented. A-Plan provides awareness and communication features for collaboration of multiple planners and simplifies data input. The semi-automated planning feature supports complex scheduling tasks in computing optimal routes for service technicians in a geographic area using the savings algorithm.

A major contribution of this work is that it shows how the fields of information visualization, collaboration and optimization can be successfully combined for supporting human dispatchers in their complex resource scheduling tasks. The feedback of users in our evaluation was generally positive, especially compared to the system currently in use. The implemented prototype was considered to be more efficient and easier to use. This work also shows the benefits of early involvement of potential users in the development process.

Kurzfassung

Einsatzplanung von Mitarbeitern zu deren Arbeit ist eine komplexe Aufgabe mit einer großen Anzahl an Einflußfaktoren und erfordert viele Fachkenntnisse. Langfristige und kurzfristige Anforderungen müssen berücksichtigt werden, dafür ist eine flexible Lösung notwendig. Planungssoftware kann Benutzer dabei unterstützen, optimale Einsatzpläne anzufertigen, jedoch sind derzeit verfügbare Werkzeuge nur bedingt brauchbar. Diese Arbeit beschreibt das Design, die prototypische Implementierung und die Evaluierung des Werkzeugs A-Plan für automatisationsgestützte Einsatzplanung.

Zunächst wurden die Abläufe einer realen Firma beobachtet und analysiert. Während dieser Phase wurden Schwächen der bestehenden Prozesse und des im Einsatz befindlichen Softwaresystems identifiziert. Mittels eines benutzerzentrierten Entwicklungsprozesses wurde ein neues Tool designed und daraufhin der Prototyp A-Plan entwickelt. Als Abschluss wurde der Prototype durch potentielle Benutzer sowie IT Experten evaluiert.

In der Applikation A-Plan werden Einsätze von Mitarbeitern bei Kunden visuell dargestellt und können mittels direkter Manipulation modifizert werden. Dadurch können die benötigten Informationen den Benutzern rasch geliefert werden. Außerdem wurde eine benutzerfreundliche und schnelle Suchfunktion implementiert. A-Plan berücksichtigt für die Zusammenarbeit mehrer Benutzer 'awareness', bietet Kommunikationsfunktionen sowie vereinfachte Dateneingabe. Die automatisationsgestützte Planungsfuntkion unterstützt die komplexe Planungsaufgabe dadurch, dass optimale Routen für Servicetechniker berechnet werden.

Ein wichtiger Beitrag dieser Arbeit ist, wie die Bereiche Informationsvisualisierung, Zusammenarbeit und Optimierung erfolgreich zur Unterstützung menschlicher Disponenten in ihrer komplexen Ressourcenplanung kombiniert werden können. Das Feedback der Nutzer in unserer Evaluierung war allgemein positiv, vor allem im Vergleich zu dem im Einsatz befindlichen System. Der implementierte Protoyp ist demnach effizienter und einfacher zu bedienen. Diese Arbeit zeigt auch die Vorteile der frühzeitigen Einbeziehung von potentiellen Benutzer während des Entwicklungsprozesses.

Contents

List of Figures

List of Tables

CHAPTER 1

Introduction

In an economy time is money. Proper resource utilization (e.g., staff, machines, rooms, vehicles) is one of the major cost factors. Let´s look at three different application examples:

- An airline maintains its fleet in regulated maintenance intervals. A central disposition dispatches service technicians for scheduled service appointments. Emergency defects, however have to be repaired immediately. The service dispatcher plans the maintenance schedule on a long term basis, and must be able to dispatch technicians in emergency situations immediately.
- A mobile nursing care company has to dispatch its staff to the patients depending on their condition. The employees have different qualifications: some are only allowed to give injections, but everyone is able to clean etc. The planning has to take all qualifications, regulations as well as limitations into account. It is also very important to minimize the distances between the assignments, because the travel between patients causes additional costs for the company.
- A gas device (heater, stoves, et cetera) maintenance provider has to dispatch its staff to the customers. Customers with a maintenance contract get their gas devices serviced in specific intervals. Failure of a gas devices can be reported to a call center and the call center agent can promptly dispatch a service technician. If the technician is not available (e.g., in case of sickness), the dispatcher has to rearrange the service technician to the assignments.

The given examples illustrate the relevance of this problem and the need for tool support. Better scheduling can reduce costs and increase the quality of service for the customers while reducing the administrative work as well as travel and vacancy times. Particularly, a Visual Analytics (VA) approach that integrates automatic methods and supports humans via interactive visual interfaces [Thomas and Cook, 2005] seems to be a perfect fit for this kind of problem complexity. In this thesis we present the design, implementation, and evaluation of a VA prototype that combines automatic optimization and interactive visual interfaces to aid employees of a gas device maintenance provider to schedule their service technicians. The three main challenges in this context are to (1) design interactive visualization methods; (2) integrate automated planning functionality; and (3) support synchronous collaboration when handling customer requests. Our main aim is to aid personnel in their complex planning and scheduling tasks while keeping an optimal balance between automatic algorithms and user interaction.

To evaluate the tool in real life, the project will be implemented for a gas device maintenance provider. User requirements will be inquired at the beginning of the project and a prototype will be tested by actual users.

CHAPTER 2

Problem Analysis

The basis for the qualitative research of this thesis was thoroughly analyzed. Qualitative research helps according to Alan Cooper [Cooper et al., 2007, page 50] to understand :

- Behavior, attitudes, and aptitudes of potential product users
- Technical, business, and environmental contexts - the domain - of the product to be designed
- Vocabulary and other social aspects of the domain in question
- How existing products are used

In our work we used the particular methods of task demonstration, ethnographic observation and interviews for the analysis as presented in [Kulyk et al., 2007]:

- *Ethnographic observation* is a method where the designer observes the users working environment in practice. Observation is very useful but also has to deal with severals problems: The observations can be misinterpreted (when the observer does not know the context of the actions). An observation can also disturb the work of the observed group, and the observed work can differ from normal work situations.
- *Interviews:* In interviews subjects will be asked about their work and the use of the software and artifacts. Bartlett and Toms [Bartlett and Toms, 2005] notice that *'a drawback to the interview approach is that it relies on recall, rather than directly capturing the activity of interest, and is thus vulnerable to missing details that were either forgotten, or not considered relevant.'* In [Cooper et al., 2007] different groups of interview partners are described:
  - *Stakeholders* are people with authority and/or responsibility for the product. The goal is to gather information pertaining to budgetary, scheduling and

technical constraints. Stakeholders may have important insights on users that will help to formulate the research plan.

  - *Subject matter experts* are often expert users. Similar to stakeholders they can provide valuable perspectives on a product and its users.
  - *Customers* are those people who make the decision to purchase a product. Customers and user often coincide.
  - *Users* are the people that actually work with the system and should be the main focus of the analysis.

- *Task demonstration:* This method is similar to observation, however the observer may ask questions, and the demonstrator may explain certain tasks in a more detail manner. A disadvantage is that the task is described by the user, so the feedback may be very limited and problems may not become visible, since most experienced users are not aware of these problems [Shneiderman, 2005].

We started with a task demonstration, so we got an overview about the application and the process. In the ethnographic observation we found out how the work is really done (section 2.4). The interview process a possibility to get more information about the users and the problems with the current application. Further methods for task analysis are questionnaires and participatory workshops [Kulyk et al., 2007]. Questionnaires have compared to interviews two disadvantages. The user doesn't have the possibility to ask additional questions, which may leave the person misunderstood. In a participatory workshop a focus group discuss the requirements of the system, here one member of the group may dominate the discussion, in particular, when the participant has an higher status. In our focus group one person has an higher status, since he is longer in the company than the other participants. So we decided to omit a participatory workshop.

2.1 Initial Situation

Resource scheduling without computer support can be very difficult. To illustrate the problems, all tasks and the currently used software in our company will be analyzed. The company's primary field of business is supplying their customers with energy. Furthermore, the company operates biomass power plants, and offers a variety of services, such as thermography. One area, customer service, offers service and repair of gas appliances, the activities of this area are examined in this thesis. The company has 50.000 customers and 200 employees. The customer service department supports 20.000 customers. 35 employees of customer service are dispatched by 7 employees in the office.

The IT infrastructure is state of the art: the PCs are about four years old and were equipped with mouse, keyboard and 24 inch TFT displays. Microsoft Windows Vista is used as operating system, furthermore Microsoft Office is used for mailing and working with documents. The companies standard software is supplied by SAP. Two different SAP systems are used in the company: SAP ERP IS/U CCS for the whole company and supports finance, human resources, material management, controlling and the functions needed for energy customers (e.g., accounting, billing, dunning etc.). The customer service uses SAP CRM customer service, which consists of two parts: A mobile client for all technicians which allows them to support themselves during their work, as well as an solution for scheduling and billing.

The interesting part is everything surrounding scheduling in customer service. Three different situations for dispatching assignments to customers can be distinguished:

2.1.1 Regular Assignment for Customers with Maintenance Contract

The service technician is scheduled according to the customers maintenance contract, normally on a yearly basis. The existing software system delivers a report with scheduled maintenance contracts. The dispatcher assigns the contracts accordingly.

The dispatcher assigns the appointments manually, using an existing resource scheduling tool. Automated planning is not possible with this system, furthermore the usage is complicated.

2.1.2 Failure of gas devices

Customers report repair requests through the company's call center. The call agent tries to solve the problem via telephone. If the problem cannot be solved, she makes an entry in the failure list. Afterwards she tries to contact a service technician near the customer to solve the problem: If she finds a free technician near to the customer, she makes an entry in the printed form of the assignment plan. Short term planning is always done manually on the printed because entering the data into the existing software system is complicated and would take to long.

Once the problem has been solved, the entry's status is changed to "done".

2.1.3 Refusal or Deferral of Assignment for Maintenance

When customers want to cancel or change the date of their appointment, they contact the company's call center. The dispatcher corrects the date in the scheduled maintenance plan.

2.2 Currently Used Software

The company we investigated uses the software SAP CRM customer service. The cases mentioned above (refusal or deferral of assignments, failure and assignment planning) will be presented in reference to the currently used software. Afterwards we show how the work is done, and which tools are in use.

The most important entities in SAP CRM are:

- customer: Represents a customer of the company. All data relating to customers are located here, e. g. the address, the name, birth date etc.

- installation: Is the location where a gas device is installed. A customer may have more than one installation, e.g. a business customer may have multiple branch offices. Every installation location is individually saved.
- component: Represents an individual gas device. A component is installed in an installation and an installation can have more then one component. Relevant information concerning the manufacturer and technical specifications are saved on the component.
- contract: Contains relevant information pertaining to the customer's maintenance contract.
- assignment: Is the planned appointment carried out by a service technician.

In SAP, the 'transactions' are comparable to Windows applications. The following transactions are used for processing:

- CIC - customer interaction center: In the CIC all master and transactions data are managed.
- Service list: The service list takes the data from the CIC into the resource planning tool.
- Resource planning tool: This tool offers the assignment planning function.

In the following section we discuss the processes in detail:

2.2.1 Appointment Request

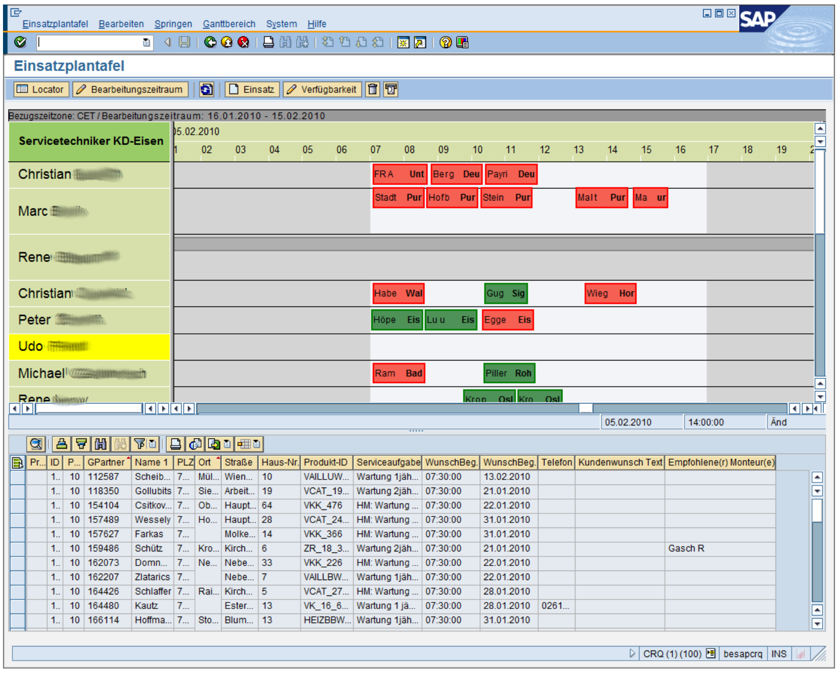
The customer requests an appointment by telephone. Two different cases can be distinguished:

1. No assignment was previously appointed.
2. The planning was already done.

Appointment Request without Prior Assignment

To use the requested date for an assignment, this date must be set as the next "due date" of the contract. To do this, specific steps must be taken:

- The customer's record must be opened in the CIC. Here the requested maintenance date must be inserted as the next "due date".
- Once this has been done, the service list is opened. Here the appropriate customer care center is chosen. This depends on the customer's installation location.



Pr...	ID	P...	GPartner	Name 1	PLZ	Ort	Straße	Haus-Nr.	Produkt-ID	Serviceaufgabe	WunschBeg.	WunschBeg.	Telefon	Kundenwunsch Text	Empfohlene(r) Monteur(e)
	1..	10	112587	Scheib...	7...	Mül...	Wien...	10	VAILLUW...	Wartung 1jäh...	07:30:00	13.02.2010			
	1..	10	118350	Gollubits	7...	Sie...	Arbeit...	19	VCAT_19...	Wartung 2jäh...	07:30:00	21.01.2010			
	1..	10	154104	Csitkov...	7...	Ob...	Haupt...	64	VKK_476	HM: Wartung ...	07:30:00	22.01.2010			
	1..	10	157489	Wessely	7...	Ho...	Haupt...	28	VCAT_24...	HM: Wartung ...	07:30:00	31.01.2010			
	1..	10	157627	Farkas	7...		Molke...	14	VKK_366	HM: Wartung ...	07:30:00	31.01.2010			
	1..	10	159486	Schütz	7...	Kro...	Kirch...	6	ZR_18_3...	Wartung 2jäh...	07:30:00	21.01.2010			Gasch R
	1..	10	162073	Domn...	7...	Ne...	Nebe...	33	VKK_226	HM: Wartung ...	07:30:00	22.01.2010			
	1..	10	162207	Zlatarics	7...		Nebe...	7	VAILLBW...	Wartung 1jäh...	07:30:00	22.01.2010			
	1..	10	164426	Schlaffer	7...	Rai...	Kirch...	5	VCAT_27...	HM: Wartung ...	07:30:00	28.01.2010			
	1..	10	164480	Kautz	7...		Ester...	13	VK_16_6...	Wartung 1 jä...	07:30:00	28.01.2010	0261...		
	1..	10	166114	Hoffma...	7...	Sto...	Blum...	13	HEIZBBW...	Wartung 1jäh...	07:30:00	31.01.2010			

Figure 2.1: Currently used resource planning tool in SAP

- Finally, the maintenance list appears. The user selects the desired maintenance entries and confirm them for resource planning.

The actual planning takes place in the resource planning tool as seen in figure 2.1. The screen is divided into two separate areas. In the upper area, the technicians assignments are displayed. The color of the box in any given row indicates the status of the assignment. In the lower area a list of open assignments is displayed. To plan an appointment for a specific date, the corresponding line in the lower area of the screen must be selected and inserted in the technician's time frame.

In the upper area the assignments of the technicians along the timeline are presented. The assignments are displayed as boxes, different colors indicate the corresponding status of the assignment. In the lower area the customer objects that have to be scheduled are represented. These are the objects that have been selected in the service list in the previous step. To plan an assignment for the requested appointment of the customer, the corresponding line in the lower area has to be selected and inserted at the requested point in time at a technician.

After this has been done, the assignment's status can be changed to "fixed". This status indicates that the technician has to be on time at the customer.

Appointment Request with Preexisting Assignment

In this case an existing appointment has been made in the resource planning tool. To the assignment date, the dispatcher has two possibilities to choose from:

- The assignment can be moved via drag and drop into appropriate are of the display.
- The dispatcher can open a detailed view of the assignment and change the date of the assignment.

The dispatcher can also set the status of the assignment to ‘fixed’, when the technician has to be at the customer at the specified time.

2.2.2 Planning of Due Maintenance

Every maintenance contract in the affected period must be assigned. The due date is determined by the last maintenance appointment, and the intervals defined in the contract. The maintenance intervals can be either annual or biennial.

The process is similar to that of an appointment request without assignment, however it is not necessary to insert a specific date. In this case, the user selects the maintenance period from the service list, as well as the affected rows. In the resource planning tool, the maintenance appointments are assigned to the appropriate field technician.

It is the dispatcher’s responsibility to assign the appropriate service technician, because of the different types of devices serviced by the company. The dispatchers background knowledge of the available field technician is a key factor in planning activities, as there is no automatic assignment. In the resource planning tool, it is only possible to enter absences such as vacation or sick leave.

2.2.3 Cancellation of an Appointment

If a customer needs to cancel an appointment, the dispatcher deletes the appointment in the resource planning tool. The dispatcher writes the abbreviation “AG”, which means “abgesagt” (canceled), into the assignment description.

2.2.4 Gas Device Failures

In the settlement of failures, two types are distinguished:

- Recording of a failure without allocation
- Assigning with allocation to a technician

Recording of a Failure without Allocation

Generally, device failures are directly reported to the companies call center. Using the CIC platfrom, the dispatcher opens the customers data and records a new failure. It is important that all of the device data is correctly noted, because some older devices can no longer be repaired. The dispatcher notes a brief failure description, as well as the customer's telephone number. In this short consultation an estimated repair date is set.

The dispatcher starts a report showing a list of open activities and contact a suitable technician. As soon as the dispatcher finds an available technician, necessary details are noted, and the open activities status is changed to "transmitted".

Assigning a Failure with Allocation to a Technician

In order to record a device failure of this type, it is necessary to insert a business transaction into the CIC. In this entry, estimated data is inserted, such as the number of hours needed, etc. After all of the necessary positions have been filled in, the status of the transaction is changed to "in progress" and finally to "released".

When the business transaction is successfully saved, it can be assigned in the resource scheduling tool.

2.3 Criticism of the Currently Used Software

The existing system runs stable and offers a lot more functionality than actually needed in the investigated company. In SAP CRM virtually everything is configurable. However, the biggest problem with this software is not the functionality, rather the elaborate processing that cause usability difficulties. The user related problems will be discussed in a more detailed manner:

2.3.1 Resource Planning Tool

Abbreviation of Information at Assignments

The information in assignments is kept very short and the abbreviation is usually not longer than four letters (figure 2.2). The user cannot locate a single assignment, because it is necessary to open every item via double-click. The detail view of an assignment is shown by means of a modal dialog box. This box allows only one assignment to be analyzed at a time, and the response time is very long.

No Consistent Drag and Drop Behavior

Direct manipulation is an advantage of the resource planning tool, however the long response time is definitely a disadvantage. It would be better for the user when the application responded immediately, once the desired assignment was chosen. Direct manipulation is only possible for existing assignments, thus creating further disadvantages for the user. To insert a new assignment, the dispatcher marks the customer record in

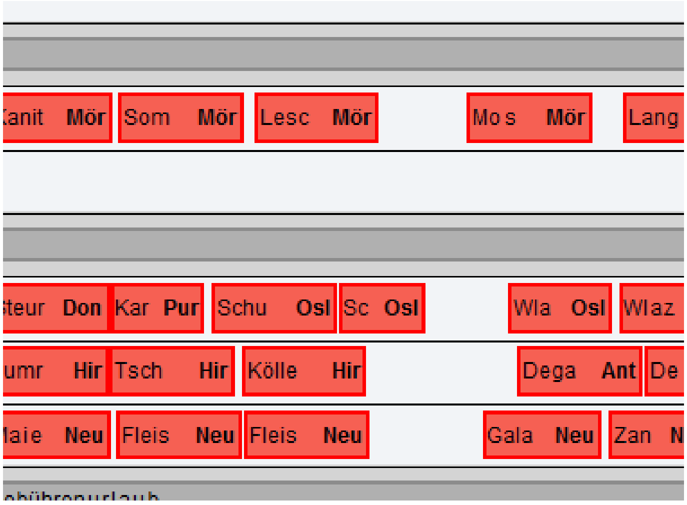


Figure 2.2: Abbreviation at the resource planning tool

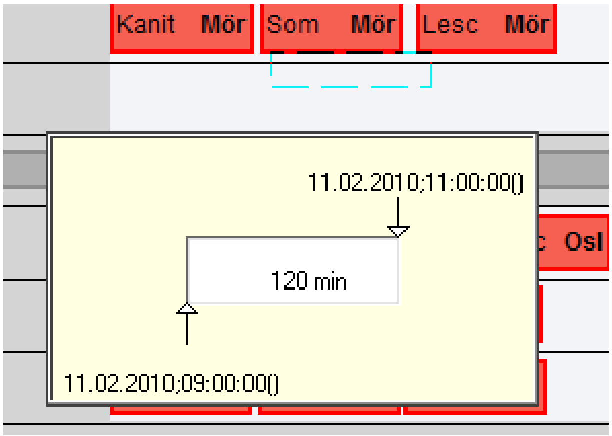


Figure 2.3: Insertion of an new assignment

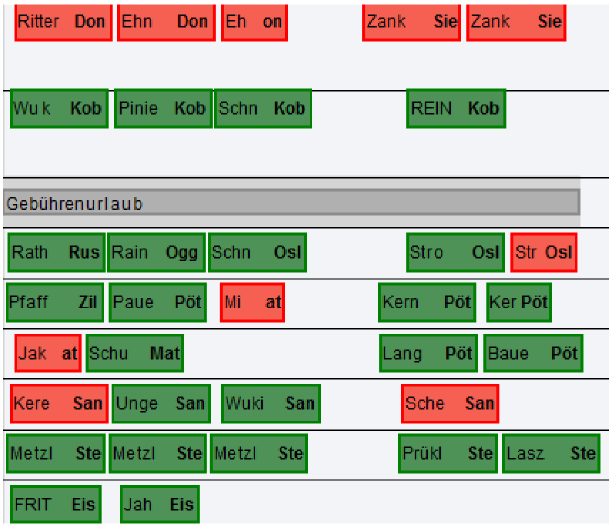


Figure 2.4: Confusing color marking

the lower screen area. In the planning area, the field where the new assignment is to be inserted is marked (seen in figure 2.3).

It would be easier and more efficient when the system had a consistent drag and drop behavior for existing and new assignments alike.

Partially Confusing Color Marking of Assignments

The use of colors (figure 2.4) to differentiate between states of the appointments is considered quite reasonable. However, it is irritating that the red color code can have different definitions: 'hand generated' and 'not processed'. There should be a cleaner color code system for the assignment status.

Withdrawal of Changes not Possible

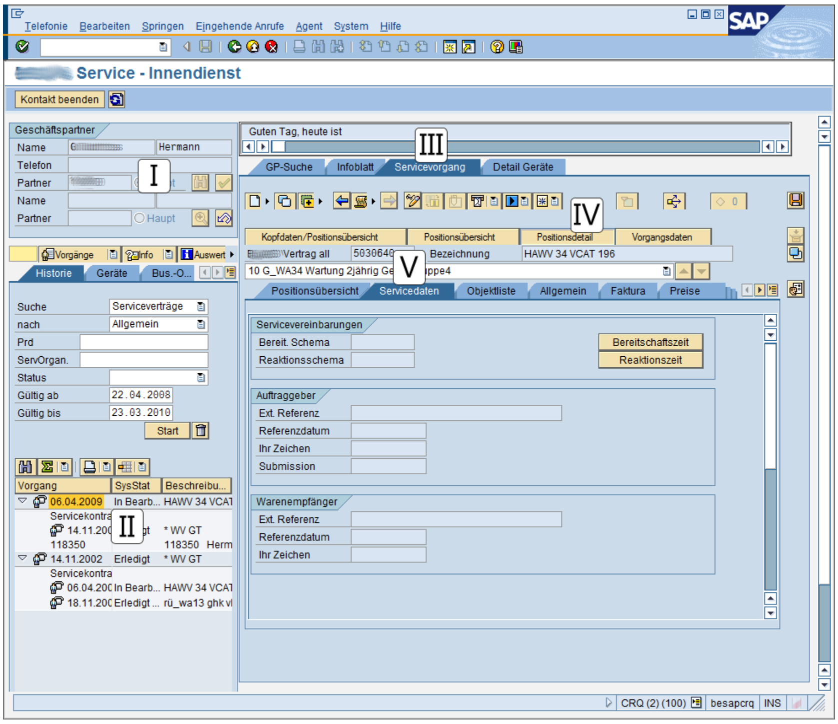
If an appointment status has already set to 'completed', it cannot be changed. The system doesn't have an 'undo' function, which is another disadvantage for the user.

Complex Handling of Three Applications

The user has to handle three different applications for assignments processing. It would be better when all of the necessary functions were facilitated in a single resource planning tool.

Missing Technician Information

The dispatcher has a huge responsibility when it comes to allocating the appropriate technician to an assignment. Because the system doesn't supply this sort of information,



Name [redacted] Hermann
Partner [redacted]

Suche Serviceverträge
nach Allgemein
Gültig ab 22.04.2008
Gültig bis 23.03.2010

Vorgang	SysStat	Beschreibu...
06.04.2009	In Bearb...	HAWV 34 VCAT
Servicekontra		
14.11.200	gt	* WV GT
118350		118350 Herm
14.11.2002	Erledigt	* WV GT
Servicekontra		
06.04.200	In Bearb...	HAWV 34 VCAT
18.11.200	Erledigt ...	rü_wa13 ghk vl

[redacted] Vertrag all 5030640 Bezeichnung HAWV 34 VCAT 196
10 G_WA34 Wartung 2jährig Ge ppe4

CRQ (2) (100) besapcrq INS

Figure 2.5: customer interaction center

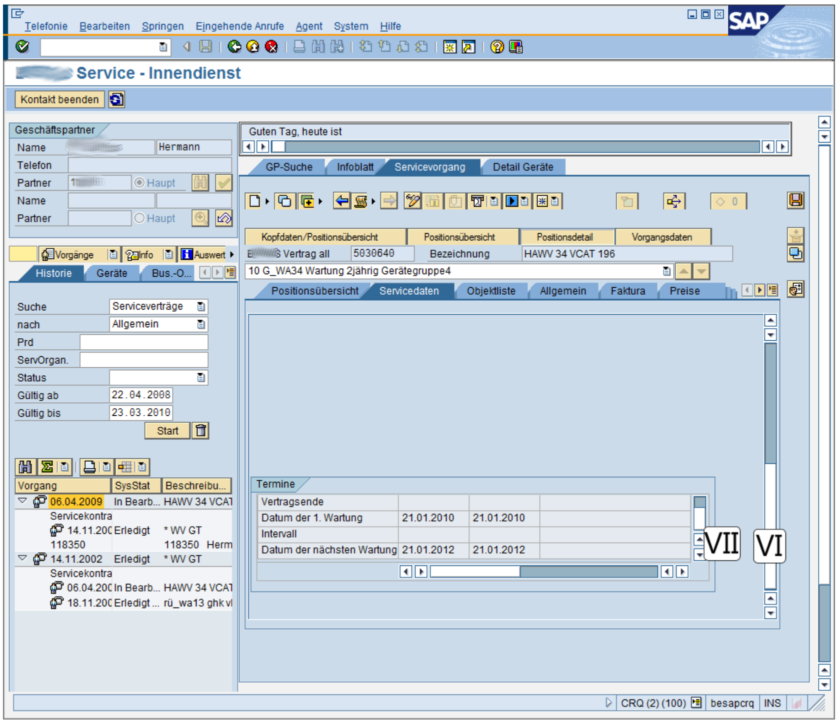
the dispatcher has to rely on recalled knowledge about every individual technician. The system only allows basic information to be inserted, such as sick leave, vacation and long term absences. Other information like the place of residence or skills of the technicians are not available.

2.3.2 Customer Interaction Center CIC

Overloaded Interface

In the customer interaction center (figure 2.5) all of the customers data are shown. The interface is structured using tabs. Every single tab is structured in tabs again. Some information is redundant, most is useless and important information is hard to find. In figure 2.5 and 2.6 you can see where the information of the next maintenance date is positioned:

- start by selecting the customer (figure 2.5 I)
- select the contract (figure 2.5 II)
- select the tab ‘Servicevorgang’ (figure 2.5 III)
- select the button ’Positionsdetail’ (figure 2.5 IV)



[redacted] Service - Innendienst

Geschäftspartner

Name	[redacted]	Hermann
Telefon		
Partner	[redacted]	● Haupt
Name		
Partner		○ Haupt

Suche	Serviceverträge
nach	Allgemein
Prd	
ServOrgan.	
Status	
Gültig ab	22.04.2008
Gültig bis	23.03.2010

Vorgang	SysStat	Beschreibu...
06.04.2009	In Bearb...	HAWV 34 VCAT
Servicekontra		
14.11.200	Erledigt	* WV GT
118350		118350 Herm
14.11.2002	Erledigt	* WV GT
Servicekontra		
06.04.200	In Bearb...	HAWV 34 VCAT
18.11.200	Erledigt ...	rü_wa13 ghk vl

Guten Tag, heute ist

[redacted] Vertrag all	5030640	Bezeichnung	HAWV 34 VCAT 196

10 G_WA34 Wartung 2jährig Gerätegruppe4

Termine

Vertragsende			
Datum der 1. Wartung	21.01.2010	21.01.2010	
Intervall			
Datum der nächsten Wartung	21.01.2012	21.01.2012	

CRQ (2) (100) besapcrq INS

Figure 2.6: Placement of information "next service due" in CIC

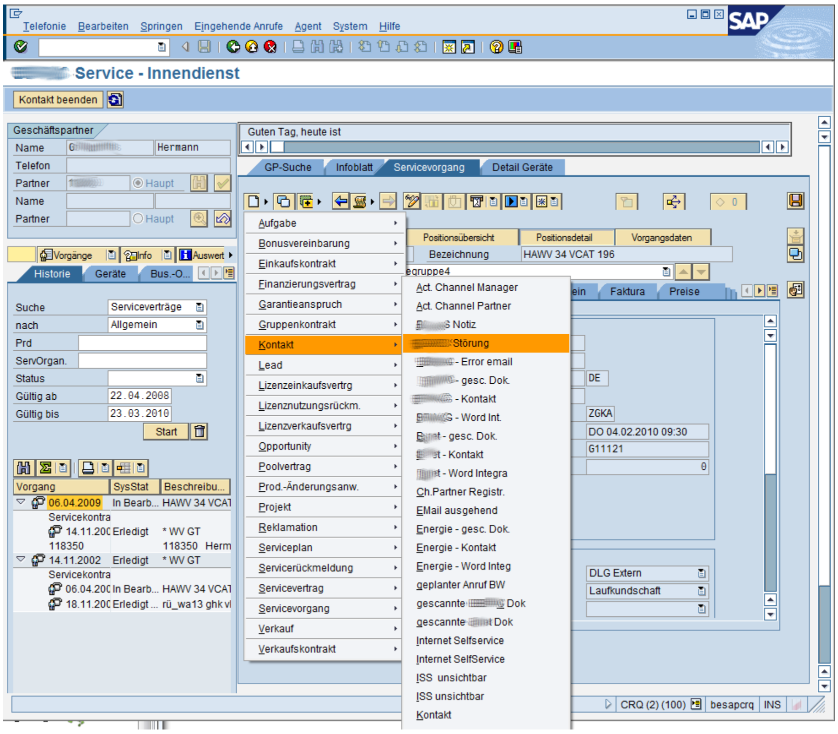
- select the tab 'Servicedaten' (figure 2.5 V)
- scroll down to the end of the mask (figure 2.6 VI)
- scroll down the table to the last row (figure 2.6 VII)

Another example of the overloaded interface can be seen in figure 2.7. As shown in figure 2.7, the context menus contains 22 items, and each item has more than 20 sub items. A reduction of the menu items would be advisable, as not all of the items are actually in use.

Hidden Error Messages

When a business transaction can't be saved due of missing data, the CIC only displays a small red dot in the upper right hand corner, as shown in figure 2.8. For further information, the user must click on this button to show the error message (figure 2.9).

Alan Cooper describes in About Face [Cooper et al., 2007, pages 534 to 547] opportunities for feedback in case of errors: *Needless to say, this means that subtle status information is simply never communicated to users at all.* Furthermore he describes rich visual modeless feedback (RVMF) as a way to provide information in case of incomplete or inconsistent data input. RVMF provides the information directly next to the data



Name [redacted] Hermann

Partner [redacted]

Suche Serviceverträge

nach Allgemein

Gültig ab 22.04.2008

Gültig bis 23.03.2010

Vorgang	SysStat	Beschreibu...
06.04.2009	In Bearb...	HAWV 34 VCAT
Servicekontra		
14.11.200	Erledigt	* WV GT
118350		118350 Herm
14.11.2002	Erledigt	* WV GT
Servicekontra		
06.04.200	In Bearb...	HAWV 34 VCAT
18.11.200	Erledigt ...	rü_wa13 ghk vi

Bezeichnung HAWV 34 VCAT 196

egruppe4

DE

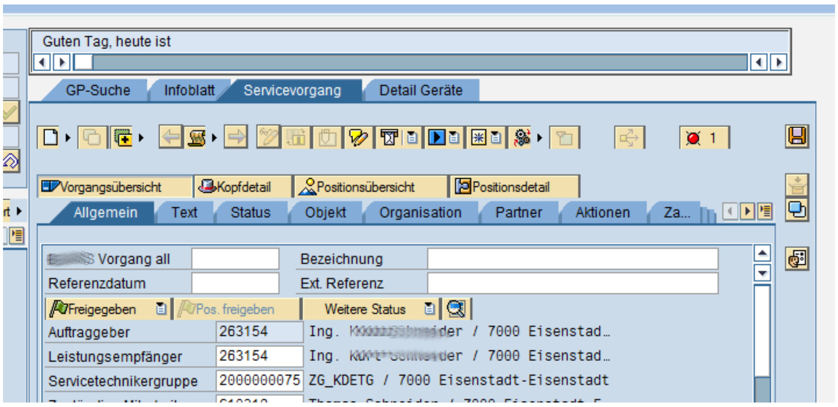
ZGKA

DO 04.02.2010 09:30

G11121

0

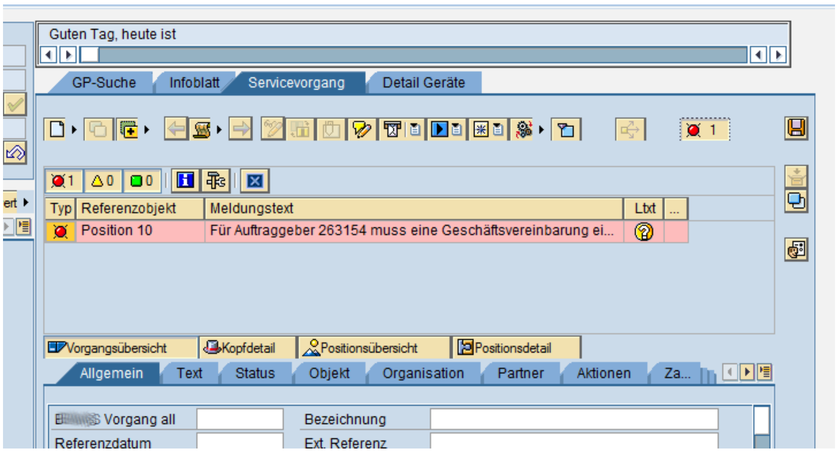
Figure 2.7: Context menu when clicking on new button



[redacted] Vorgang all		Bezeichnung	
Referenzdatum		Ext. Referenz	
Auftraggeber	263154	Ing. [redacted] / 7000 Eisenstad…	
Leistungsempfänger	263154	Ing. [redacted] / 7000 Eisenstad…	
Servicetechnikergruppe	2000000075	ZG_KDETG / 7000 Eisenstadt-Eisenstadt	

Figure 2.8: Error when business transaction can't be saved

input field and is modeless, so the user can continue his work. To provide clear feedback in the application, RVMF would improve the usability of the existing application. In difference to RVMF the error message in CIC prevents saving the entered data, so the user has to correct the data and save afterwards.



Typ	Referenzobjekt	Meldungstext	Ltxt	...
●	Position 10	Für Auftraggeber 263154 muss eine Geschäftsvereinbarung ei...	?	

Figure 2.9: To receive the error message, the list of error messages has to be opened

Safe Exploration

When a user wants to create a failure without assignment a new entry is needed. Confusing is the arrangement of the input fields in this place: the to-date is left of the from-date. This leads to many missentries. The even bigger problem here is the reaction of the program to the incorrect entry. The user gets an error message that the from-date has to be less than the to-date. Instead of allowing a correction, the program is terminated without saving.

This is entirely unreasonable: One of the most important aspects in user interface design is safe exploration. The principle can be summarized as "Let me explore without getting lost or getting into trouble."[Tidwell, 2005] A wrong input should not end in dire consequences for the user, in the mentioned case all data is lost. If the customer has already hang up the telephone, there is no possibility to recover the data.

2.4 How the Work Is Done

In a user observation it was discovered out that in many cases the dispatcher does not use the software. The dispatcher answers questions and confirms customer data on the telephone within seconds, which is not possible with the existing software. Therefore, the users of the software have created a bypass that makes a rapid telephone answer possible. Grudin describe this form of user behavior in his law:

> ‘Grudins law: when those who benefit are not those who do the work, then the technology is likely to fail, or at least be subverted’ [Norman, 1993]

The company decided to purchase the software, without reference to users. For the users, the existing software is too slow and awkward to use in real world situations. Rather than retrieving information from the system, users use two artifacts:

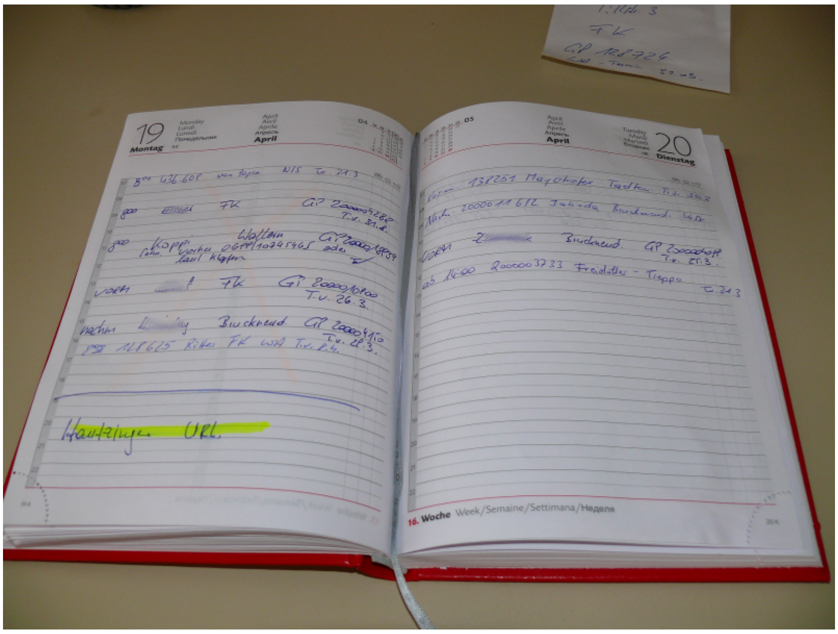


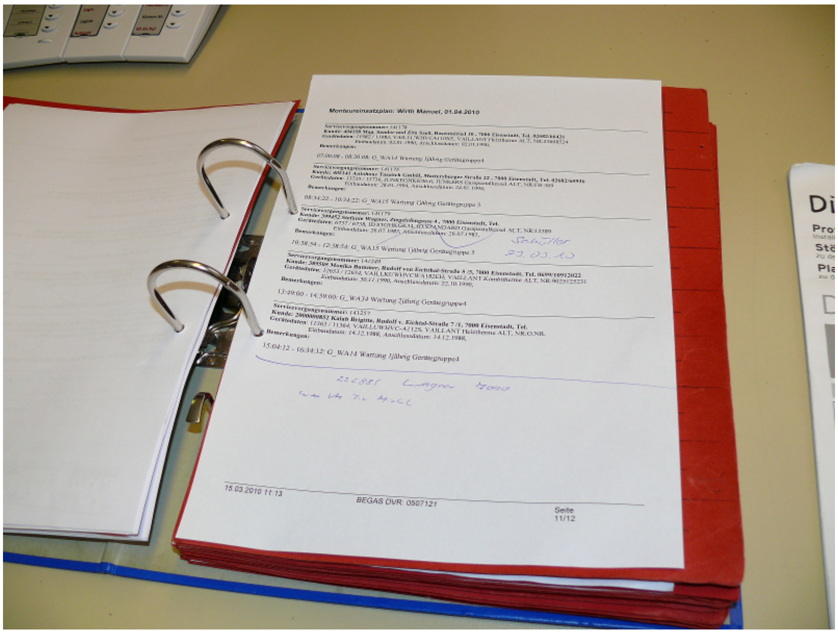
Figure 2.10: A book calendar is used to write down all data from the customers during phone calls

- *A Book Calendar* (figure 2.10) includes all appointment requests from customers, so the customer's data can quickly be covered during the phone call. In the evening all data is recorded in the system, thus the long waiting times of the service list will occur only once for all assignments of the day.

- *A folder* (figure 2.11) with separator sheets for each technician as well as a list of the technicians assignments. The lists are always printed from the SAP system some days in advance. When a customer calls the dispatcher and reports short-notice cancellations or schedule changes, the information will be recorded in this folder. Later the information is transferred into the system. The reason for using the folder as a tool is that the resource scheduling tool only delivers limited information of the assignments, and therefore no quick answers on the phone are possible. The information is sorted by region. Every region has an alphabetical listing of available technicians, making allocation of assignments easier.

2.4.1 Consequences

Calendar books and folders are rarely used by multiple user, thus creating a further disadvantage: Only one dispatcher can work with the book or the calendar folder at a time. Nevertheless, in order to be able to serve multiple customers simultaneously, the technicians are divided into regions and every dispatcher only gets calls of his/her region.



Monteureinsatzplan: Wirth Manuel, 01.04.2010

Servicevorgangsnummer: 141176
Kunde: 404158 Mag. [illegible], 7000 Eisenstadt, Tel. 02682/64421
Gerätedaten: 11982 / 11983, VAILLI/WHVCAT10VE, VAILLANT Heizitherme ALT, NR:23668524
Einbaudatum: 02.01.1990, Anschlussdatum: 02.01.1990,
Bemerkungen:

07:00:08 - 08:20:08: G_WA14 Wartung 1jährig Gerätegruppe4

Servicevorgangsnummer: 141173
Kunde: 402143 Autohaus Tauzisch GmbH, Mattersburger Straße 22 , 7000 Eisenstadt, Tel. 02682/68936
Gerätedaten: 13735 / 13736, JUNKERKKW6, JUNKERS Gasspeicherkessel ALT, NR:FD 909
Einbaudatum: 29.01.1994, Anschlussdatum: 29.01.1994,
Bemerkungen:

08:34:22 - 10:34:22: G_WA15 Wartung 1jährig Gerätegruppe 5

Servicevorgangsnummer: 141179
Kunde: 399452 Stefanie Wagner, Ziegelofengasse 4 , 7000 Eisenstadt, Tel.
Gerätedaten: 6737 / 6738, IDSTOTKGK14, IDSTANDARD Gaspeicherkessel ALT, NR:12389
Einbaudatum: 26.07.1985, Anschlussdatum: 26.07.1985,
Bemerkungen:

10:38:54 - 12:38:54: G_WA15 Wartung 1jährig Gerätegruppe 5

Schüller
23.03.10

Servicevorgangsnummer: 141249
Kunde: 389505 Monika Baumer, Rudolf von Eichthal-Straße 8 /5, 7000 Eisenstadt, Tel. 0699/10912022
Gerätedaten: 12653 / 12654, VAILLKE/WHVCWA182EH, VAILLANT Kombitherme ALT, NR:9025125231
Einbaudatum: 30.11.1990, Anschlussdatum: 22.10.1990,
Bemerkungen:

13:49:00 - 14:59:00: G_WA34 Wartung 1jährig Gerätegruppe4

Servicevorgangsnummer: 141257
Kunde: 2000000851 Kaleb Brigitte, Rudolf v. Eichtal-Straße 7 /1, 7000 Eisenstadt, Tel.
Gerätedaten: 11363 / 11364, VAILLI/WHVC-A7125, VAILLANT Heiztherme ALT, NR:O.NR.
Einbaudatum: 14.12.1988, Anschlussdatum: 14.12.1988,
Bemerkungen:

15:04:12 - 16:34:12: G_WA14 Wartung 1jährig Gerätegruppe4

226885 Lagner 7000
[illegible]

15.03.2010 11:13 BEGAS DVR: 0507121 Seite 11/12

Figure 2.11: A folder with one sheet per technician is used for answering phone calls

Again, it is the dispatcher's responsibility to maintain an overview of the regional technicians. To keep all sites of 35 technicians in mind is almost impossible. However, when numerous customer call from one region, the employees can only handle one incoming call at one time. The consequence is a very long waiting time, at a higher incoming call rate. Furthermore the dispatcher has to search manually for an ideal date.

The call could be answered much quicker if the dispatcher had a software tool that supports this task. Thus, the weak software design in the illustrated example, directly affects the company's customers. Another consequence is that the subsequent capture in the system caused an additional effort. This situation would be more efficient, when the planning would be made directly into a software system.

CHAPTER 3

State of the Art

In the following, we will discuss related research work relevant to different aspects of our work. The research was done in digital libraries, in literature and the internet. A main aspect for a tool is the design of the user interface and interaction with the system. In the example service provider a software with a lot of functionality exists, but it is not being used because of a far too complex interface. In this chapter state of the art for visualization and interaction will be shown.

Furthermore, the application is used by more than one person at a time, therefore aspects of collaboration will be analyzed. In our opinion a proposal for tour planning would bring a considerable improvement at the work of the users, therefore algorithms for tour planning will be shown.

3.1 Interface and Interaction

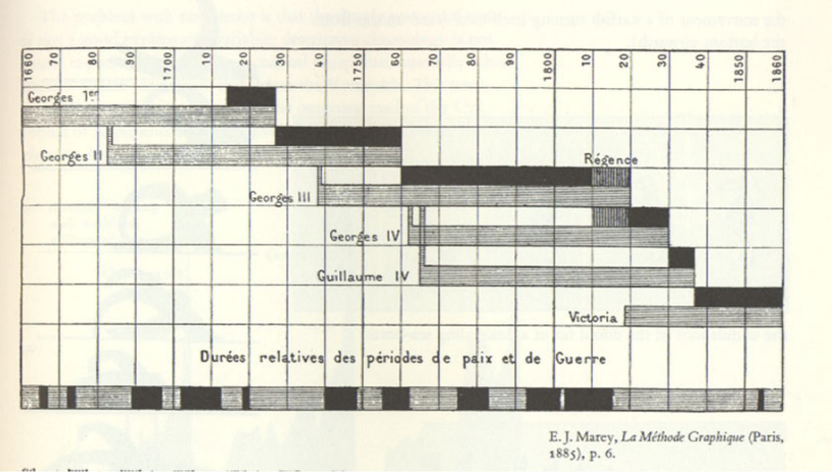
A lot of literature about 'interface and interaction' exists. In the following section techniques that appeared most suitable and interesting in the context of scheduling will be examined.

3.1.1 Visualization of time oriented data

At scheduling tasks the aspect how to visualize the time oriented data is important for the usability of the application. In this section different visualization techniques will be shown.

Timelines

Timelines are a simple visualization of events or time intervals. Karam describes timelines in [Karam, 1994] as linear graphical visualization of events over time. In general the time axis is presented in one axis direction (mostly x-axis) and category on the other



E. J. Marey, *La Méthode Graphique* (Paris, 1885), p. 6.

Figure 3.1: Timeline from E.J. Marey, La Methode Graphique (Paris 1885), page 6 [Tufte, 1983]

axis. Events are shown as bars where the length shows how long the event is active. In [Tufte, 1983] examples for timelines are shown, as in figure 3.1.

Timelines are often used in scheduling scenarios, as in our example mentioned in the previous chapter (figure 2.1).

LifeLines

In [Plaisant et al., 1996] LifeLines are introduced (figure 3.2) as a further development of timelines. In LifeLines the thickness of the bars shows the importance of the cases. Also information is encoded as colors. Plaisant also shows some possibilities for overview and detail, and zooming in her work. This techniques are discussed later.

GANTT

GANTT charts are a special form of timelines invented by Henry L. Gantt. It was published in the same year he died (1919), a long time before it was used in several project management products. In his book [Gantt, 1919] some example for his timelines are shown (figure 3.3). In the early GANTT charts users used progressively filling the bars to show how much of the task was complete.[1] Modern project software as in figure 3.4 also show milestones and hierarchies of tasks.

Time Mosaics

The disadvantage of the mentioned visualization techniques is the high space consumption. Luz and Masoodian try to decrease the space consumption in their temporal mosaic

[1] An evolution of the GANTT chart can be seen in http://www.ganttchart.com/Evolution.html, accessed on 19th August 2010.

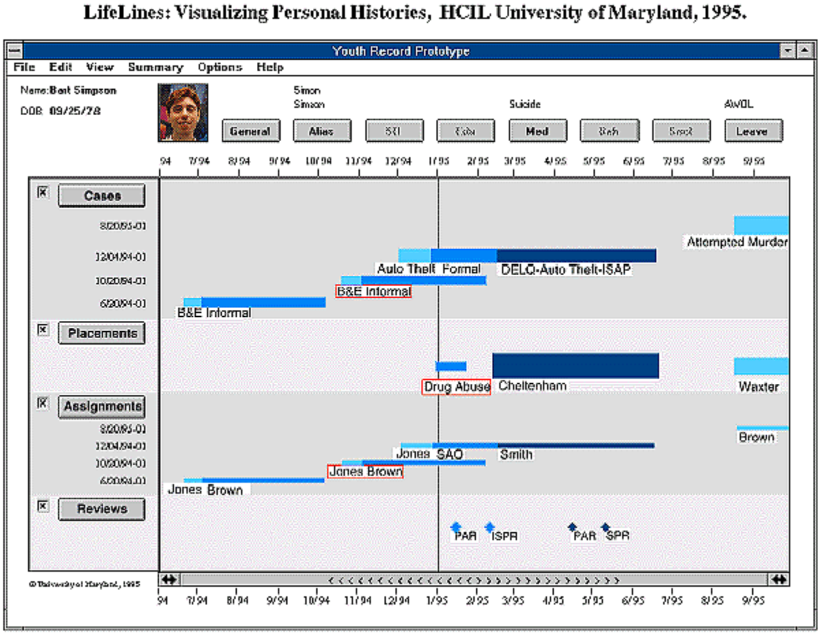


Figure 3.2: Example for LifeLine Visualization from [Plaisant et al., 1996] where LifeLines for Juvenile Justice records are shown. This simple youth record shows cases, placements, assigned workers and available reviews. A click on the 'drug abuse' highlights the labels of the associated breaking and entering case, and the workers who where assigned to the case.

technique[Luz and Masoodian, 2007]. timelines tend to require much space since inactive intervals need to be represented as blank areas. Therefore, it is difficult to get an overview of many categories on whether they run concurrently or not. Temporal Mosaics have been developed to avoid this disadvantage.

Figure 3.5 shows an example of a multimedia application as timeline. In Figure 3.6 the same data is visualized as Temporal Mosaic. In the Temporal Mosaic view all bars from one group (for example voice) are shown in one compound rectangle. If more than one voice category is active, the corresponding color bars in the group share the space of the group. If only one category is active, the whole height has the color of the category. White space signals that no category is active.

The distinction between categories is done using by different colors (A, B, C and D in figure 3.6). Hence a distinction between more than five categories becomes difficult. Furthermore movement of assignments in temporal mosaics via direct manipulation is difficult. Because of these facts Temporal Mosaics appear inappropriate for scheduling tasks.

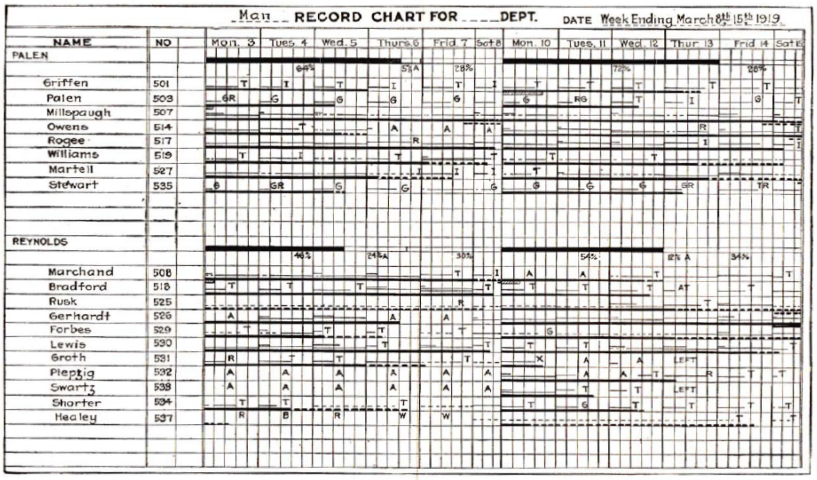


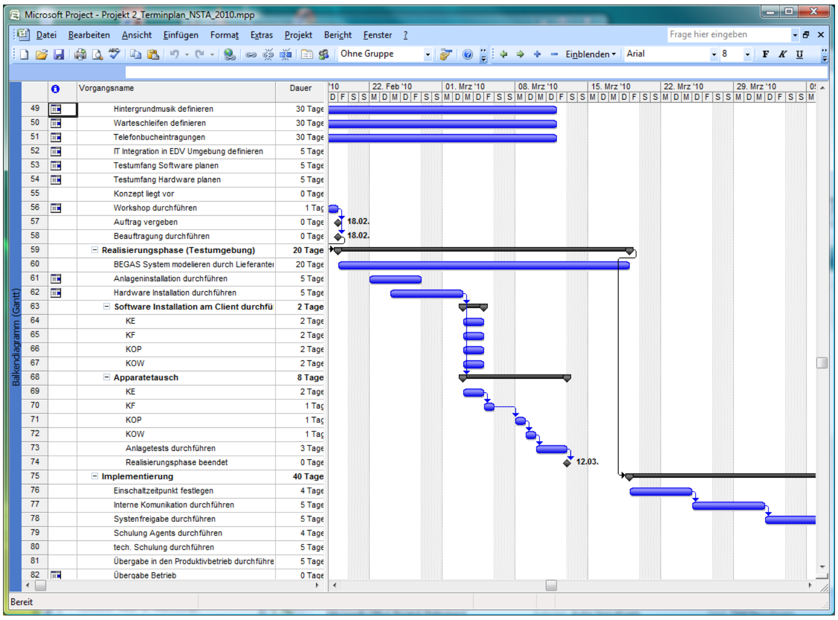
Fig. 8.—Key for Man Record Chart

The daily space represents the amount of work a man should have done in a day, and also the time taken to do the work.
Estimated time for work done.
Time on job for which we have no estimates.
Solid line = cumulative estimated time for work done. Broken line = total time used on work not estimated.

The portion of the daily space through which no line is drawn shows how much the man has fallen behind what he was expected to do. The reasons for his falling behind are indicated by the following symbols:

A	Absent	I	Lack of instruction	V	Holiday
D	Defective work	M	Lack of or defective material	X	Reason not clear
G	Green operator	T	Tool troubles, or lack of tools		

Figure 3.3: Example from Organizing for work from H. Gantt [Gantt, 1919]


Microsoft Project - Projekt 2_Terminplan_NSTA_2010.mpp

		Vorgangsname	Dauer
49	▦	Hintergrundmusik definieren	30 Tage
50	▦	Warteschleifen definieren	30 Tage
51	▦	Telefonbucheintragungen	30 Tage
52	▦	IT Integration in EDV Umgebung definieren	5 Tage
53	▦	Testumfang Software planen	5 Tage
54	▦	Testumfang Hardware planen	5 Tage
55		Konzept liegt vor	0 Tage
56	▦	Workshop durchführen	1 Tag
57		Auftrag vergeben	0 Tage
58		Beauftragung durchführen	0 Tage
59		− **Realisierungsphase (Testumgebung)**	**20 Tage**
60		BEGAS System modelieren durch Lieferanter	20 Tage
61	▦	Anlageninstallation durchführen	5 Tage
62	▦	Hardware Installation durchführen	5 Tage
63		− **Software Installation am Client durchfü**	**2 Tage**
64		KE	2 Tage
65		KF	2 Tage
66		KOP	2 Tage
67		KOW	2 Tage
68		− **Apparatetausch**	**8 Tage**
69		KE	2 Tage
70		KF	1 Tag
71		KOP	1 Tag
72		KOW	1 Tag
73		Anlagetests durchführen	3 Tage
74		Realisierungsphase beendet	0 Tage
75		− **Implementierung**	**40 Tage**
76		Einschaltzeitpunkt festlegen	4 Tage
77		Interne Komunikation durchführen	5 Tage
78		Systenfreigabe durchführen	5 Tage
79		Schulung Agents durchführen	4 Tage
80		tech. Schulung durchführen	5 Tage
81		Übergabe in den Produktivbetrieb durchführe	5 Tage
82	▦	Übergabe Betrieb	0 Tage

Figure 3.4: Modern GANTT chart in Microsoft Project

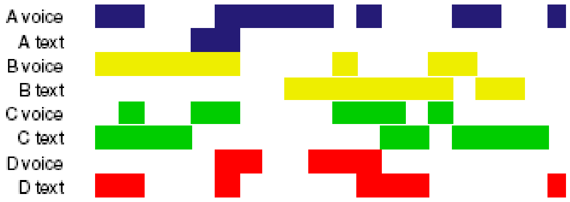


Figure 3.5: Standard timeline visualisation example [Luz and Masoodian, 2007]

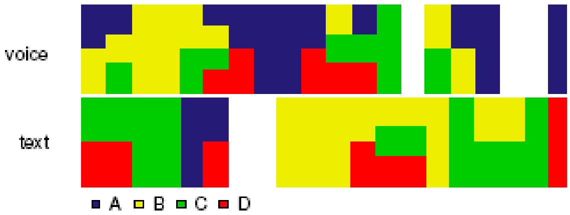


Figure 3.6: Temporal mosaic example [Luz and Masoodian, 2007]

3.1.2 Interaction

A scheduling application does not only visualize the data, the user must also have the opportunity to work with the data. If a user is working with a large amount of data, she must have the ability to determine the level of detail for herself. This approach is described Shneiderman's information seeking mantra [Shneiderman, 1996]:

> Overview first, zoom and filter, then details-on-demand.

Techniques to offer this possibility to the user are described in the following sections:

Overview+Detail

An overview+detail interface design is characterized by the simultaneous display of both an overview and a detailed view of an information space, each in a distinct presentation space [Cockburn et al., 2008]. Overview areas help the user with orientation in the content. Overview+detail interfaces are widely used for example in image editing applications, in map tools or charts.

An example for overview and detail can be seen in figure 3.7. In this example the overview also provides the possibility to pan and zoom the detail view.

Zooming

Using zoom technique the user can change between detail and overview in one view. In contrast to Overview and detail technique only one level of detail is displayed, however

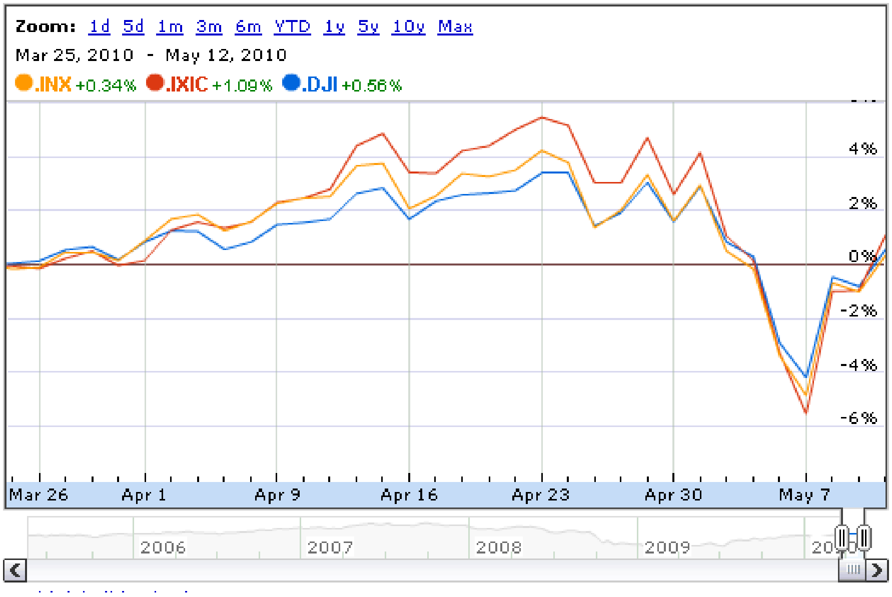

Zoom: 1d 5d 1m 3m 6m YTD 1y 5y 10y Max
Mar 25, 2010 - May 12, 2010
● .INX +0.34% ● .IXIC +1.09% ● .DJI +0.56%
4%
2%
0%
-2%
-4%
-6%
Mar 26
Apr 1
Apr 9
Apr 16
Apr 23
Apr 30
May 7
2006
2007
2008
2009
20

Figure 3.7: Example overview and detail at Google Finance where a chart of the selected time interval in the upper area and long term overview in the lower area is shown

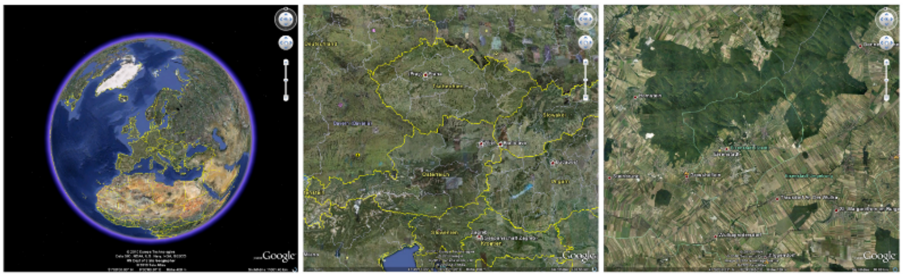


Figure 3.8: Example for three zoom levels in Google Earth

the entire available space is used for the view. Continuous and non-continuous zoom can be distinguished.

An example for an application that makes extensive use of zooming is Google Earth (see figure 3.8). In this application the zoom can be controlled intuitively using the mouse wheel. A control in the upper right corner can also be used to zoom in or zoom out. The application uses animation to help the user understand the relationship between pre- and post-zoom. According to Tversky [Tversky et al., 2002] using animations is more effective for re-orientation in time and space.

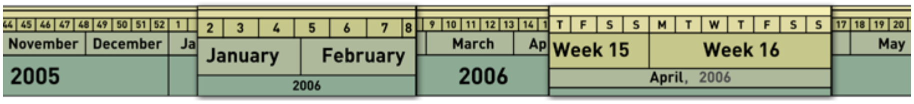


Figure 3.9: Different focused time units in TimeZoom [Dachselt and Weiland, 2006]

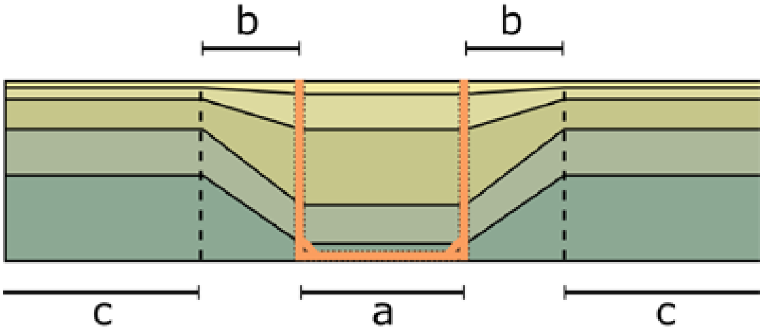


Figure 3.10: Different focus regions (a and c) and transitional zones (b) between them at TimeZoom[Dachselt and Weiland, 2006]

Focus+Context

In the focus+context technique the detail and the surrounding context is shown in a single display. An example is the fisheye distortion used in the icon panel from Mac OS X. The problem of this technique is the distortion of the information space, which might lead to misinterpretation and challenges in target acquisition [Cockburn et al., 2008].

3.1.3 Special Techniques

In this section four visualization applications are shown. They can be seen as combinations of the previously presented techniques.

TimeZoom

Dachselt and Weiland [Dachselt and Weiland, 2006] presented a new concept for a flexible timeline widget. The widget supports zooming and focusing on special intervals of time. TimeZoom uses weighted level heights for every focused time unit (figure 3.9). At TimeZoom different zones for focused regions are also possible with transitional zones similar to the fisheye distortion (figure 3.10).

Event Tunnel

Suntinger [Suntinger et al., 2008] introduces a different approach for visualization of events as metaphor considering the event stream as cylindrical tunnel. The events are plotted in temporal order in a cylinder. Two views are supported, a view from the top

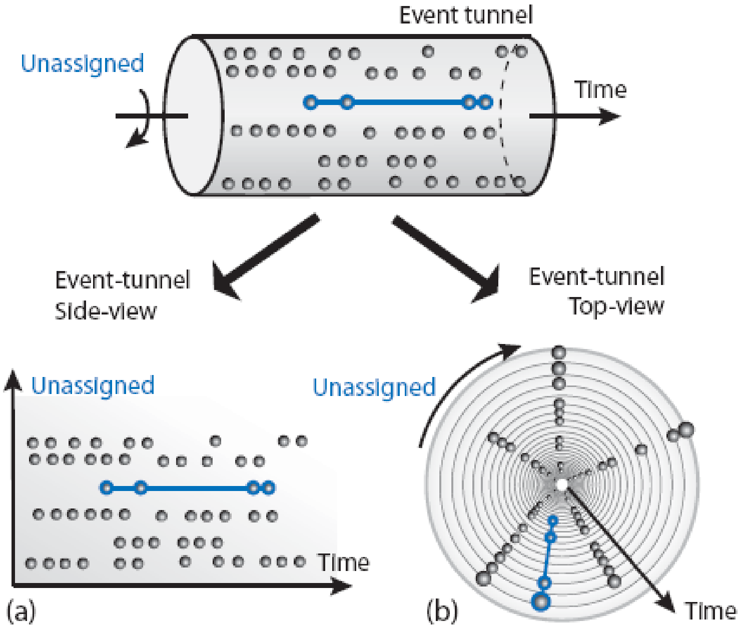


Figure 3.11: Example for the event tunnel from [Suntinger et al., 2008]

of the tunnel and a side view (figure 3.11). The events are rendered using perspective projection, so events that are far away appear smaller in the event tunnel. Event tunnels are suitable for the visualization of correlated events, for example the analysis of supply chain management in logistics.

Perspective Wall

[Mackinlay et al., 1991] show a different approach by the Persepective Wall visualization technique . In their work they use a wall metaphor for the visualization of context and detail. The details pane in the center appears linearly, the context field is distorted in perspective (figure 3.12). This is also the way the human eye perceives information, where the center area perceives details and the surrounding region only low resolution.

Date Lens

Date Lens [Bederson et al., 2004] uses fisheye distortion and zoom in a calendar control to show detail and context in one view (figure 3.13). The goal was to develop a calendar control for (mobile) devices with very small displays where only few information can be shown.

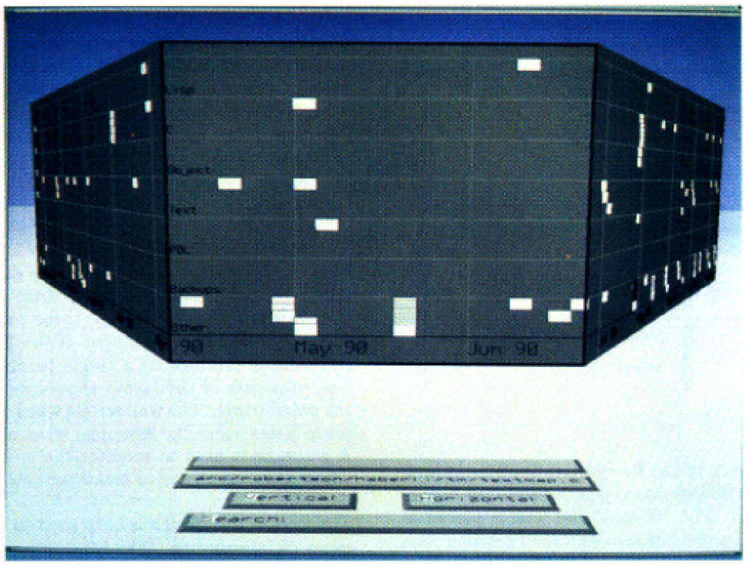


Figure 3.12: Perspective Wall from [Mackinlay et al., 1991]

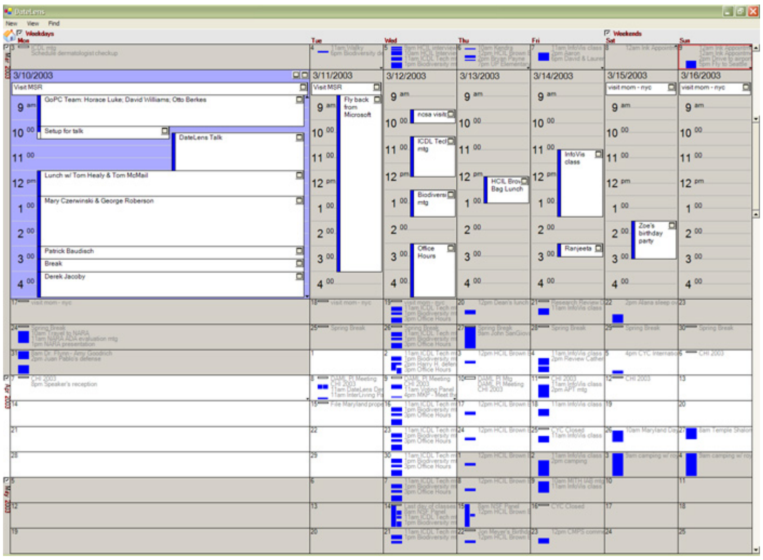


Figure 3.13: DateLens from [Bederson et al., 2004]

3.1.4 Discussion

In this section several techniques for visualization and interaction with time oriented data are shown. For our scheduling example we search for a visualization of assignments. At this data there is no dependency of the assignments and so the GANTT charts seems not to be the suitable visualization. We have up to 35 service technicians and with time mosaics the application would not be practicable as 35 color would be needed to distinguish between the technicians – no one would be able to distinguish 35 colors. The special techniques show all distortions between a central area and a less interesting peripheral area. This form of visualization may be interesting for special purposes but for our application it might be confusing for users to have a distortion in time. Therefore seems timeline and LifeLine best suited for the required tasks.

Furthermore are Overview and detail, zooming and focus and context described. In our opinion all of these techniques can be used when designing application for scheduling. Every single technique has its advantages and disadvantages. Overview and detail need more space, but always provides an full overview. Zooming use the whole space for one view, so only overview or detail are possible at one time. Focus and context may be confusing for some users, one disadvantage is that the focus area cover often other areas of the context.

Until now we only discussed visualization and interaction for a single user. In a scheduling environment normally more than one user work with the data at the same time. In the next section possibilities to support collaboration in one application will be shown.

3.2 Collaboration

Due to the fact that more than one person uses the tool at the same time aspects of Computer-Supported Cooperative Work (CSCW) systems will be examined. “These systems aim to provide suitable forms of cooperation within a group of users to perform collaborative activities” [Gomes et al., 2005]. The challenge is to implement interaction between users in the user interface. “Cooperating actors align and integrate their activities with those of their colleagues in a seemingly ‘seamless’ manner, that is, without interrupting each other, for instance by asking, suggesting, requesting, ordering, reminding, etc. others of this or that” [Schmidt, 2002]. As Schmidt describes, the awareness is an important factor in the design of cooperative applications. Dourish and Belotti [Dourish and Bellotti, 1992] describe awareness as “an understanding of the activities of others, which provides a context for your own activity”.

To provide awareness some special requirements for software systems exist. Frequently CSCW systems are distributed software systems, which require mechanisms to distribute data between many clients. Technologies to provide the distribution will be discussed in this chapter.

3.2.1 Awareness

Prinz [Prinz, 1999] suggest two types of awareness: task-oriented awareness and social awareness.

Task-oriented awareness means that the application shows "change notifications or information about the state of a certain document or a shared workspace". *Social awareness* in comparison "provide[s] notifications similar to the information received when walking along an office floor". For example social awareness is the information that the colleague is busy or the colleague is not on his place. Both types of awareness are important for a cooperative working environment.

3.2.2 Synchronous and Asynchronous

Synchronous and asynchronous cooperative work can be distinguished. An example of an asynchronous cooperative system is e-mail, where the messages are transferred at certain time intervals. Synchronous cooperative systems replicate their events as fast as possible to the clients. In case of working on the same objects, there may also be conflicts when more then one user tries to modify the object. Therefore, these systems must take such conflicts into account.

3.2.3 Architecture

Patterson [Patterson, 1995] introduced a taxonomy of architectures for synchronous Groupware applications. The aim is to provide consistent state for the system. The distinction between shared state, synchronized state and hybrid architecture is based on how often the data is replicated or how the data is exchanged between multiple clients.

In figure 3.14 the different types of *shared state* architectures are shown. Shared view architecture (figure 3.14 (a)) is when all clients work with the same view, only the displays exist more than one time. An example for this architecture is the software Team Viewer [2] with this software sharing of a desktop is possible.

Shared model means multiple displays and views on the clients accessing one model (figure 3.14 (b)). At the shared file architecture every client also has its own model logic, all clients access the same file (database). The challenge at shared model and shared file architecture is to provide mechanisms to synchronize the clients. The server has to notify all clients when changes on the data are made.

In contrast synchronized state architectures have display, view, model and the file on every client (figure 3.14 (c)). Synchronized state architectures send messages between the clients with the information that has to be synchronized. The place where the synchronization happens can be differentiated as shown in figure 3.15.

Patterson also introduces a hybrid architecture, where the model is shared and the view is synchronized (figure 3.16).

[2] The software Team Viewer is available at http://www.teamviewer.com (accessed on August 28, 2010)

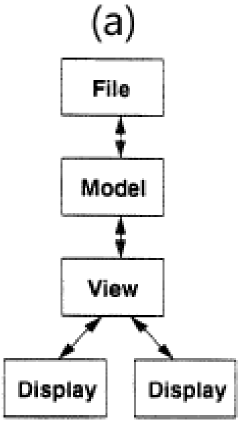


Figure 3.14: Shared state architecture [Patterson, 1995]

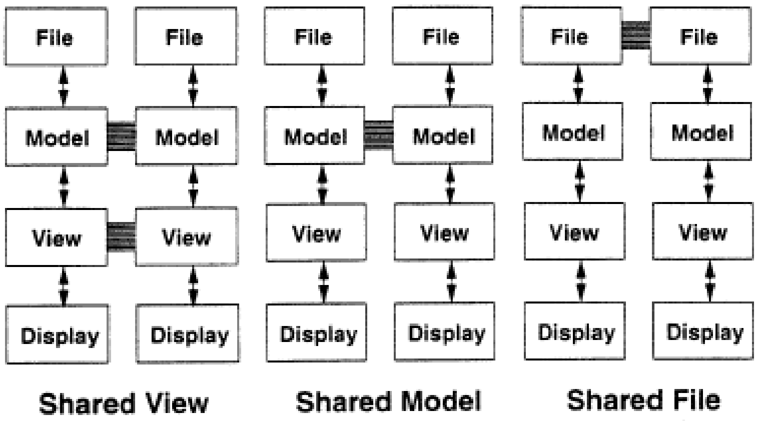


Figure 3.15: Synchronized state architecture [Patterson, 1995]

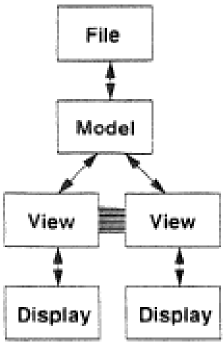


Figure 3.16: Hybrid architecture [Patterson, 1995]

3.2.4 Middleware

CSCW applications are often distributed applications, sometimes in heterogeneous environments. Middleware systems provide functionality for standardized access between servers and clients. A Middleware is a additional layer that provides better distribution transparency [Tanenbaum and Steen, 2003] and also simplifies the development process of CSCW applications. In this section some different established Middleware systems are introduced.

CORBA

CORBA stands for common object request broker architecture. The architecture of CORBA is object oriented. Interfaces are defined in the CORBA IDL (interface definition language). On the server side objects implement interfaces, on the client side proxy objects marshal request parameters to the server and unmarshal results.

CORBA supports the developer with many services, as for example the concurrency service for the handling of concurrent processes or the persistence service for saving persistent objects. CORBA implementations exist for almost any platform and programming language.

RMI

RMI is a technique supported by Java for the implementation of remote procedure calls in Java. As this technique is part of the Java environment, it is not possible to use RMI in other environments. For the definition of the interface there is no IDL like in CORBA necessary, as the interface is defined in Java too.

.NET Remoting

.NET remoting is the corresponding technology to Java RMI from Microsoft in the .NET framework. As in RMI it is possible to communicate between remote objects like between local objects. Before Microsoft developed .NET remoting, DCOM (distributed COM) was used in Microsoft environments to communicate over network. Unlike with DCOM, a remote call over the Internet is possible with .NET remoting.

Web Services

Web service are remote function calls over web standard technologies (XML, HTTP). The interface of a web service is described in WSDL, the web service definition language. Web services have the advantage, that they can be used in heterogeneous environments, for example between a SAP system and Java. Basically, they don't support object oriented programming. Only method calls are possible and no support for transaction handling exists.

Windows Communication Foundation (WCF)

Windows Communication Foundation (WCF) is also a component of the .net framework from Microsoft. WCF uses web services technology to communicate and can therefore be used in heterogeneous environments. It is even possible to mix .net remoting with WCF in one system, as WCF has the ability to communicate with .net remoting.

Discussion

The described technologies are the most important ones for distributed systems. Each technology is associated with a set of circumstances they are supposed to work best with. CORBA is often used in environments, where no other technology exist, as CORBA exists for almost every platform. For example if C++ should be used on Windows as client and Cobol on Unix as server, CORBA is to be used. Furthermore CORBA is quite old and therefore stable technology.

If Java is used as client and server, RMI will be a good choice as Middleware if calls don't need to be made over the borders of a firewall. Similar is the situation if only .NET is used for client and server with the advantage that .NET remoting can also be used in Internet scenarios behind a firewall, as .NET remoting can be used over an SOAP-HTTP channel.

Web services are widely used in internet scenarios. They can also be used for heterogeneous system environment where web service technology is available. An example is the coupling of SAP with .NET: Because of its openness a web service call is very simple. However, the performance compared to the other technologies is poor, as the call has to be translated as XML over HTTP.

Windows communication framework is a package where web service calls and .NET remoting are integrated and can even be mixed. The advantage is that the developer can simply change the used technology.

3.2.5 Groupware frameworks

In this section toolkits for handling the needs of Groupware applications are described. These frameworks hide the underlying network communication and make it easier for the developer to implement Groupware applications.

Roseman and Greenberg [Roseman and Greenberg, 1992] introduced GroupKit as toolkit for the implementation of real time Groupware applications. Three strategies for building GroupKit components are described:

- an object-oriented run-time architecture: Supports the development of the distributed process and communication.
- transparent overlays: Can be used in the user interface for some kind of shared sketching board.
- open protocols: The developer can handle policies by implementing a protocol.

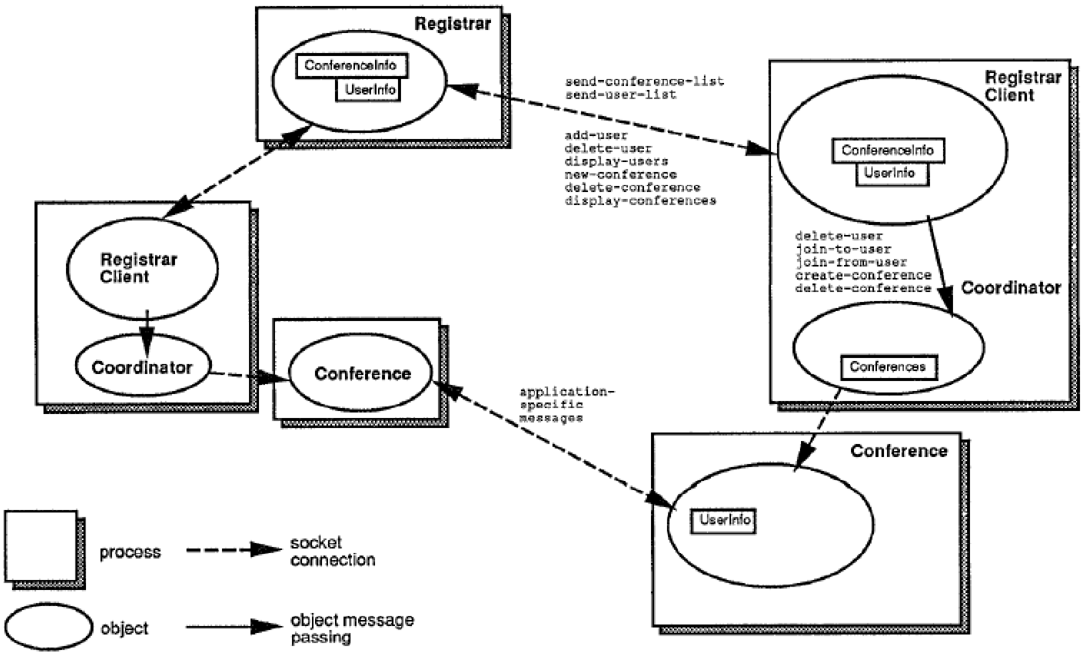


Figure 3.17: GroupKit architecture [Roseman and Greenberg, 1992]

In figure 3.17 a basic example for the use of GroupKit is shown. The registrar clients are the interfaces for the Groupware application. One central 'Registrar' manages the conferences and corresponding users. The 'Conference' object is the actual Groupware application. In GroupKit a generic conference object is implemented, the developer has to inherit a class with his own functionality. The coordinator is an intermediary between the conference and the registrar clients.

An extension to GroupKit was implemented in [De Alwis et al., 2009] in GT/SD. GT/SD stands for Groupware Toolkit / Shared Dictionary. GT/SD uses GroupKit in combination with a distributed shared dictionary, where the shared dictionary can be understood as distributed memory. The complete network handling is done by the framework. The developer can write to the dictionary like a normal dictionary. All clients are automatically notified by the framework about the updates. The same mechanisms are implemented in the library .NetworkingGT.[3]

3.3 Optimization

In our example company, the planner schedules assignments for the technicians for the next weeks manually. The challenge is to optimize the distances between the customers.

[3] The .NetworkingGT framework can be downloaded from projects website http://grouplab.cpsc.ucalgary.ca/cookbook/index.php/Toolkits/NetworkingGT, accessed on August 19, 2010

She also has to be aware of absences of the technicians. The number of assignments per day is also restricted, depending of the time of the year (in autumn bigger reserves for failure of gas devices are needed); some of the technicians have special skills for special gas devices. All these constraints have to be taken in consideration by the user and are not supported by the current software.

The described case is an example for an optimization task. More precisely it is a linear programming task. Linear programming is described as the following model:

$$Min!/Max! : c_1x_1 + c_2x_2 + c_3x_3 + ... + c_nx_n$$

whereby

$$\begin{aligned}
a_{11}x_1 + a_{12}x_2 + a_{13}x_3 + ... + a_{1n}x_n &<= b_1 \\
a_{21}x_1 + a_{22}x_2 + a_{23}x_3 + ... + a_{2n}x_n &<= b_2 \\
&... \\
a_{m1}x_1 + a_{m2}x_2 + a_{m3}x_3 + ... + a_{mn}x_n &<= b_m
\end{aligned}$$

The first function is the linear function to be minimized. The other functions are constraints. $x_1 \ldots x_n$ are the variables to calculate, $c_1 \ldots c_n$ are the cost parameters (in our example the distances between the customers).

The constraints are (for example):

- Every customer should only be visited by one technician
- The number of assignments per day
- Skills of technicians for special gas devices

The aim is to bring the real world scenario into a mathematical model of this kind. There exists another constraint to the model: $x_1 \ldots x_n$ are only integer numbers. Optimization problems of this type are called combinatorial problems or integer programming. The specialty of a combinatorial problem is the exponential growth of possibilities for a bigger number of variables. Combinatorial problems base on elemental combinatorial constructs like permutation. “For n objects exists in general $n!$ possibilities to bring this objects in sequence” [Littger, 1992]. As example there are 3628800 possibilities to bring 10 objects in sequence, for 20 objects 20! is a number with 19 digits.

Because of the exponential growth of possibilities, only approximation algorithms can be used. For the special problem in this example a lot of literature can be found.

The multiple traveling salesman (M-TSP) is a model where some salesman have to travel to customers. The goal is to travel to all customers with minimal costs. An overview of M-TSP can be found in [Bektas, 2006].

The generalized formulation of the problem is the vehicle routing problem (VRP). The VRP is an M-TSP with cargo capacity restrictions. Cargo capacity can be interpreted as a restriction for the number of customers that can be visited by a technician per day. Several approaches exist for VRP. Genetic algorithms are adaptive heuristic search

	0	1	2	3	4	5	6
0	-	3.2	2.2	2.2	1.4	2.8	1.4
1	3.2	-	1	2.2	4	5.8	2.8
2	2.2	1	-	1.4	3	5	2.2
3	2.2	2.2	1.4	-	2.2	5	3
4	1.4	1.4	3	2.2	-	3.2	2.8
5	2.8	5.8	5	5	3.2	-	3.2
6	1.4	2.8	2.2	3	2.8	3.2	-

Table 3.1: Distance matrix for the example

methods based on population genetics [Bräysy, 2004]. A dynamic genetic algorithm can be found in [Hanshar, 2007], where dynamic means that new customer orders can be dispatched after the planning has completed (in our example a failure assignment). A similar solution with stochastic customers is described in [Cheung et al., 2007].

The aim is now to construct an algorithm for the example. Important is the fast calculation (nearly in realtime) for a huge number of customers (about 20000).The result of the algorithm is a suggestion for the plan that can be adapted by the user.

In the following sections two heuristics will be introduced for the described problem.

3.3.1 The Sweep Algorithm

The sweep algorithm was proposed by Gillet and Miller in [Gillett and Miller, 1974]. The algorithm assumes that the locations of customers and the headquarter are given by coordinates (x_i, y_i) and the headquarter is in the origin of the coordinate system. The distances between the locations are determined as Eucleadian distances. The customers are sorted ascending by polar angle φ (counter clockwise) and are numbered in this order as 1 to n. The first plan is calculated by taking customer $1, 2, 3, ..., i_{11}$ into the first tour, and taking customer $i_{11} + 1, i_{11} + 2, ..., i_{11} + i_{12}$ to the second tour and so on. Customers are added to a tour until the restriction (capacity or time) is reached. Afterwards, the shortest path for this route is calculated.

The second plan is obtained by starting with customer 2 for the first tour, so the first tour has customer $2, 3, 4, ...i_{21}$, the second tour has customer $i_{21} + 1, i_{21} + 2, ..., i_{21} + i_{22}$ etc. Subsequently, n tour plans are generated by starting with every customer. The best tour plan will be selected as result.

In figure 3.18 a small example is shown. The starting point is in the origin of the coordinate system, the points with the number one to six mark the customers to deliver. The delivery points are already sorted by polar coordinates. For this example a restriction to three customers per tour is given. As first step all distances between the points have to be calculated (see table 3.1).

The first tour of the first tour plan is calculated starting with point one. The next points (two and three) are added until a restriction is reached. The second tour is constructed with the points four to six. The result can be seen in figure 3.19.

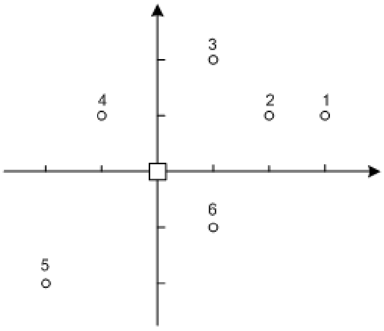


Figure 3.18: A small example with six customers and the central in the origin of the coordinate system

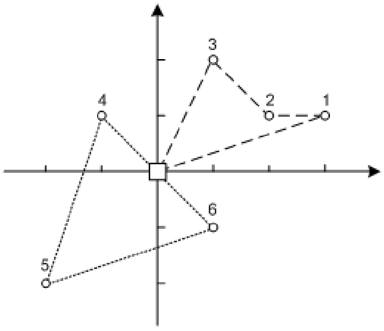


Figure 3.19: Tourplan one, generated by starting with point one, total length 18

The second tour plan starts with point two and the third tour plan starts with point three as can be seen in 3.20. The fourth to sixth tours deliver the same results, as the tours are symmetrical to the previous tour plans.

When all tour plans are generated, the shortest tour plan will be chosen (in this simple example several tour plans with the same length exist).

3.3.2 The Savings Algorithm

The savings algorithm was introduced by Clarke and Wright [Clarke and Wright, 1964] and is the most widely known heuristic for the VRP according to [Toth and Vigo, 2002]. The idea behind the savings algorithm is that savings can be obtained by joining two routes into one route. This concept is illustrated in figure 3.21, where 0 is the central depot and i and j are the delivery points. In (I) two single routes visit the two delivery points, in (II) instead the two routes are joined into one. The savings can be calculate with the following formula:

$$S_{ij} = c_{i0} + c_{0j} - c_{ij}$$

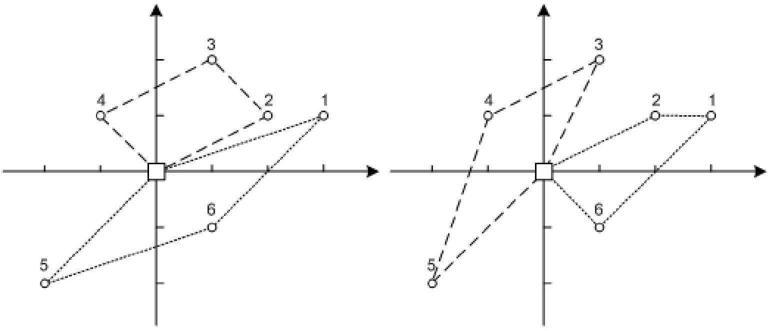


Figure 3.20: Tourplan two and three using the sweep algorithm, generated by starting with point two and three, total length 19 and 18

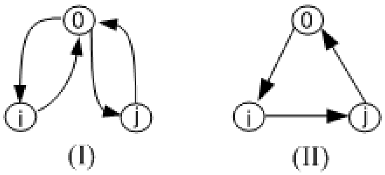


Figure 3.21: Concept of the savings algorithm

	1	2	3	4	5	6
1	-	4.4	3.2	0.6	0.2	1.8
2	4.4	-	3.1	0.6	0.1	1.4
3	3.2	3.1	-	1.4	0.1	0.6
4	0.6	0.6	1.4	-	1.1	0
5	0.2	0.1	0.1	1.1	-	1.1
6	1.8	1.4	0.6	0	1.1	-

Table 3.2: Savings matrix for the example

S_{ij} is the saving when the delivery points i and j are joined to a new tour. c_{i0} and c_{0j} are the distances from the depot to the delivery points and c_{ij} is the distance between i and j.

Two different versions of the algorithm exist. In the parallel version the algorithm constructs all routes in parallel by merging the routes with the biggest savings. In the sequential version the current route is merged with the other routes until the restriction is reached.

Let's see how the algorithm works for the example from figure 3.18: As first step the savings are calculated (see table 3.2) using the distance matrix from table 3.1. In the next step the algorithm generates a route for every delivery point as in figure 3.22.

In the parallel version the merge with the biggest savings is always taken, in our example between delivery point 1 and 2, then 2 and 3 and so on. In the sequential version

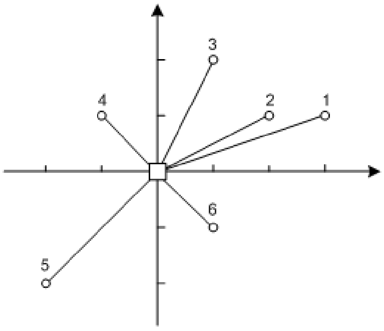


Figure 3.22: Startings point for savings algorithm

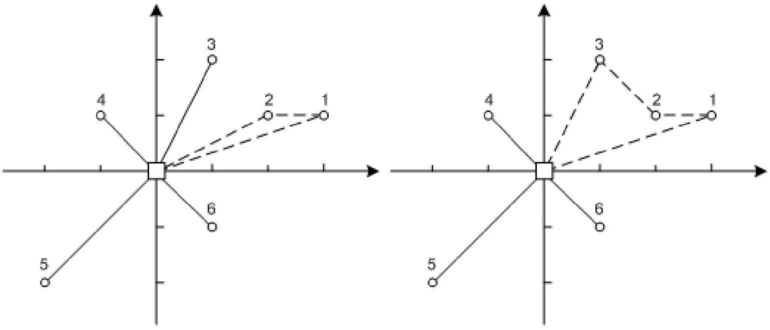


Figure 3.23: The tour plan after merging between point 1 and 2 (left) and 2 and 3 (right).

the algorithm would only merge delivery points with one route until the restriction is reached. In our simple example the result is the same.

As described by [Toth and Vigo, 2002] the parallel version of the savings calculates more optimal results than the sequential version of the algorithm.

3.3.3 Discussion

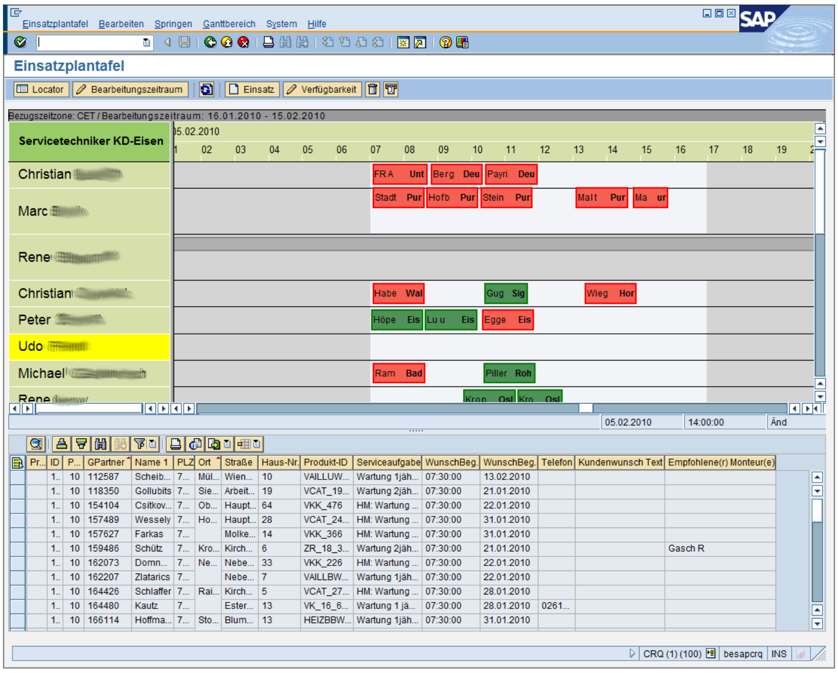
In A-Plan we want a simple heuristic for automated planing of assignments to technicians. In literature many different algorithms exists. Due to limited time an investigation of many existing algorithms is not possible in this work. In our work we decided to choose the savings algorithm, because the algorithm is the most frequently-used method for solving the VRP problem in practice [Domschke, 1990].

3.4 Dispatching Software

In the following section we will show different existing planing systems. Afterwards we will discuss the similarities and differences.

3.4.1 SAP CRM

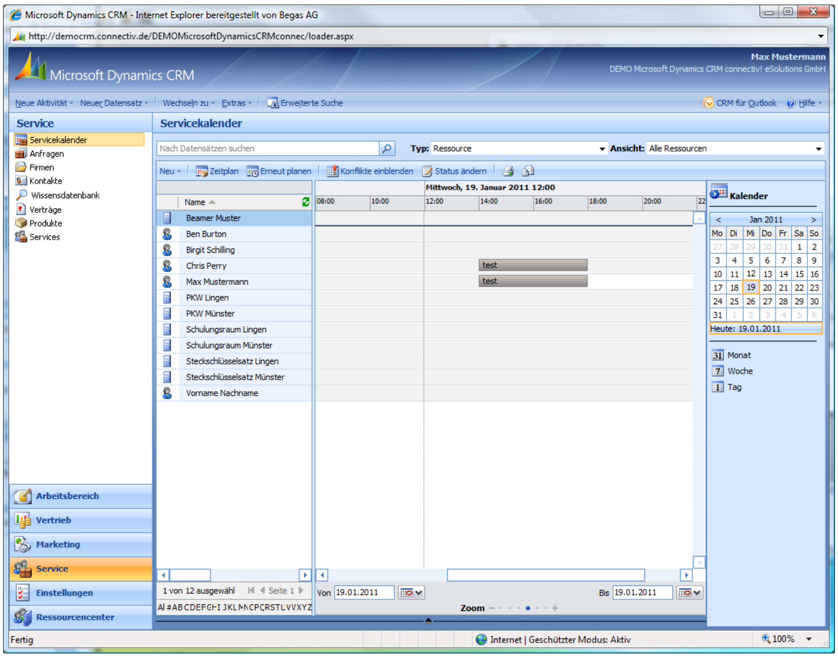
The scheduling application of SAP CRM can be seen in 3.24. This application has been discussed in section 2.2.



Pr...	ID	P...	GPartner	Name 1	PLZ	Ort	Straße	Haus-Nr.	Produkt-ID	Serviceaufgabe	WunschBeg.	WunschBeg.	Telefon	Kundenwunsch Text	Empfohlene(r) Monteur(e)
	1..	10	112587	Scheib...	7...	Mül...	Wien...	10	VAILLUW...	Wartung 1jäh...	07:30:00	13.02.2010			
	1..	10	118350	Gollubits	7...	Sie...	Arbeit...	19	VCAT_19...	Wartung 2jäh...	07:30:00	21.01.2010			
	1..	10	154104	Csitkov...	7...	Ob...	Haupt...	64	VKK_476	HM: Wartung ...	07:30:00	22.01.2010			
	1..	10	157489	Wessely	7...	Ho...	Haupt...	28	VCAT_24...	HM: Wartung ...	07:30:00	31.01.2010			
	1..	10	157627	Farkas	7...		Molke...	14	VKK_366	HM: Wartung ...	07:30:00	31.01.2010			
	1..	10	159486	Schütz	7...	Kro...	Kirch...	6	ZR_18_3...	Wartung 2jäh...	07:30:00	21.01.2010			Gasch R
	1..	10	162073	Domn...	7...	Ne...	Nebe...	33	VKK_226	HM: Wartung ...	07:30:00	22.01.2010			
	1..	10	162207	Zlatarics	7...		Nebe...	7	VAILLBW...	Wartung 1jäh...	07:30:00	22.01.2010			
	1..	10	164426	Schlaffer	7...	Rai...	Kirch...	5	VCAT_27...	HM: Wartung ...	07:30:00	28.01.2010			
	1..	10	164480	Kautz	7...		Ester...	13	VK_16_6...	Wartung 1 jä...	07:30:00	28.01.2010	0261...		
	1..	10	166114	Hoffma...	7...	Sto...	Blum...	13	HEIZBBW...	Wartung 1jäh...	07:30:00	31.01.2010			

Figure 3.24: Scheduling in SAP CRM

3.4.2 Microsoft CRM



Typ: Ressource
Ansicht: Alle Ressourcen

- Beamer Muster
- Ben Burton
- Birgit Schilling
- Chris Perry
- Max Mustermann
- PKW Lingen
- PKW Münster
- Schulungsraum Lingen
- Schulungsraum Münster
- Steckschlüsselsatz Lingen
- Steckschlüsselsatz Münster
- Vorname Nachname

Mo	Di	Mi	Do	Fr	Sa	So
27	28	29	30	31	1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30
31	1	2	3	4	5	6

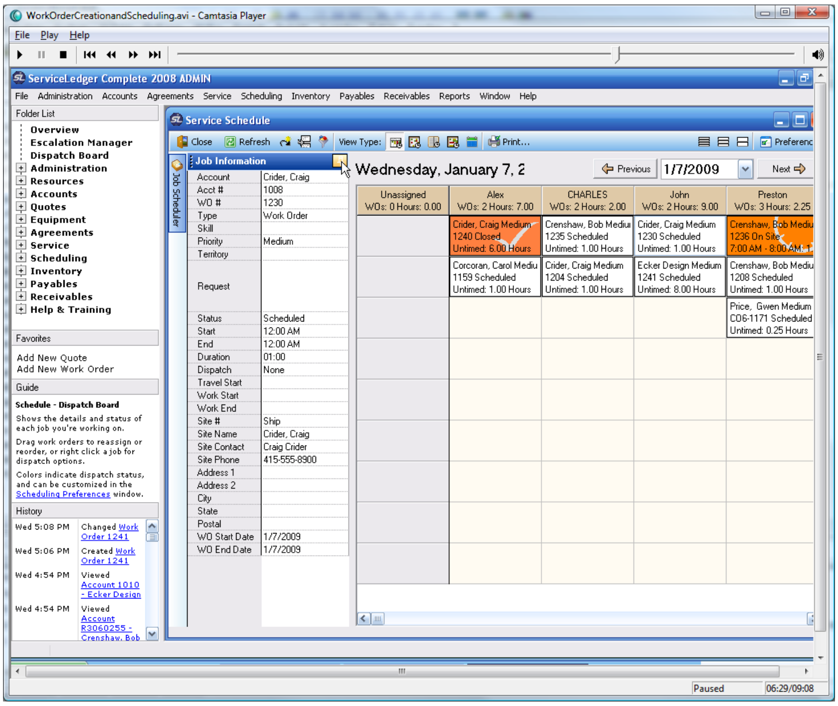
Heute: 19.01.2011

Von 19.01.2011
Bis 19.01.2011

Figure 3.25: Service planning in Microsoft CRM 4

Figure 3.25 shows the dispatching screen of Microsoft CRM 4. The application runs in the Internet browser, so the interaction possibilities are restricted (there is no direct manipulation supported, for example when moving assignments). The main focus of this application is CRM (customer relationship management), the service modules of Microsoft CRM play only a minor role in the whole system.

3.4.3 Service Ledger



Account	Crider, Craig
Acct #	1008
WO #	1230
Type	Work Order
Skill	
Priority	Medium
Territory	
Request	
Status	Scheduled
Start	12:00 AM
End	12:00 AM
Duration	01:00
Dispatch	None
Travel Start	
Work Start	
Work End	
Site #	Ship
Site Name	Crider, Craig
Site Contact	Craig Crider
Site Phone	415-555-8900
Address 1	
Address 2	
City	
State	
Postal	
WO Start Date	1/7/2009
WO End Date	1/7/2009

Wednesday, January 7, 2

1/7/2009

Unassigned WOs: 0 Hours: 0.00	Alex WOs: 2 Hours: 7.00	CHARLES WOs: 2 Hours: 2.00	John WOs: 2 Hours: 9.00	Preston WOs: 3 Hours: 2.25
	Crider, Craig Medium 1240 Closed Untimed: 6.00 Hours	Crenshaw, Bob Mediu 1235 Scheduled Untimed: 1.00 Hours	Crider, Craig Medium 1230 Scheduled Untimed: 1.00 Hours	Crenshaw, Bob Mediu 1236 On Site 7:00 AM - 8:00 AM: 1.
	Corcoran, Carol Mediu 1159 Scheduled Untimed: 1.00 Hours	Crider, Craig Medium 1204 Scheduled Untimed: 1.00 Hours	Ecker Design Medium 1241 Scheduled Untimed: 8.00 Hours	Crenshaw, Bob Mediu 1208 Scheduled Untimed: 1.00 Hours
				Price, Gwen Medium C06-1171 Scheduled Untimed: 0.25 Hours

Schedule - Dispatch Board

Shows the details and status of each job you're working on.

Drag work orders to reassign or reorder, or right click a job for dispatch options.

Colors indicate dispatch status, and can be customized in the Scheduling Preferences window.

Wed 5:08 PM	Changed Work Order 1241
Wed 5:06 PM	Created Work Order 1241
Wed 4:54 PM	Viewed Account 1010 - Ecker Design
Wed 4:54 PM	Viewed Account R3060255 - Crenshaw, Bob

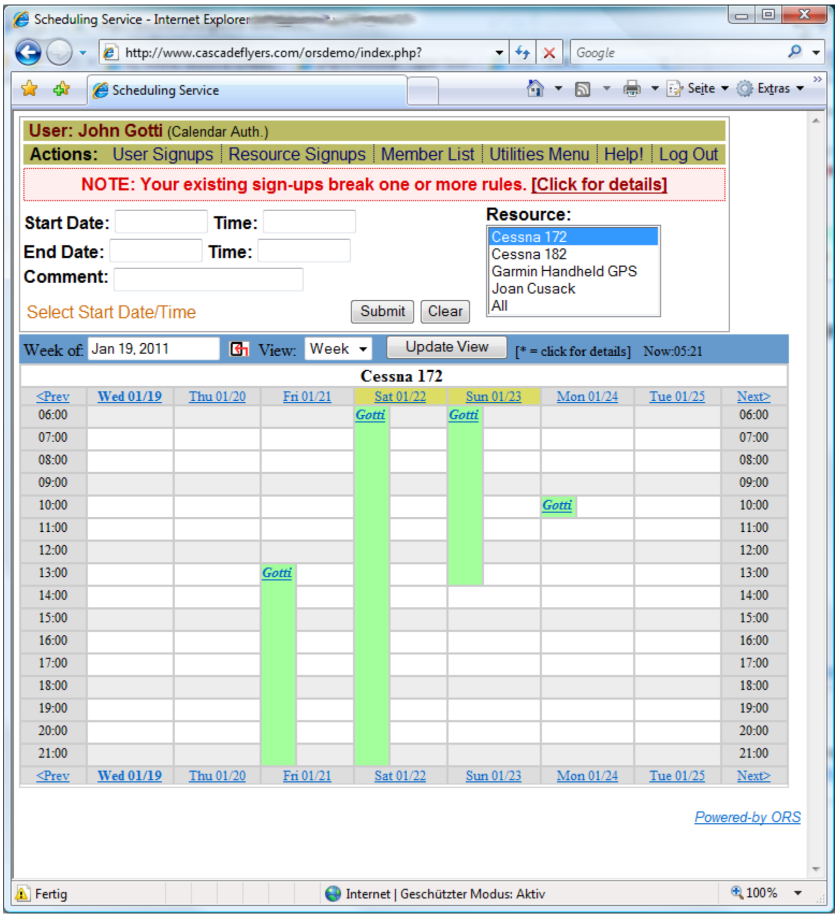
Figure 3.26: Service planning in Service Ledger

Service Ledger[4] is a software for service providers and offers many different views for dispatching, for example the dispatch board in figure 3.26. When clicking on one assignment, detailed data is shown on the left side of the window.

Service Ledger also offers the possibility to plan with a timeline as in the applications above. Furthermore a plan can be made using time slots, where a time slot is a time range when the service technician is at the customer. An interesting functionality is the MapPoint integration, this functionality shows the route on a map with the approximate travel times.

[4] Information about Servicee Ledger is available at http://www.serviceledger.com/serviceschedulingsoftware.htm (last access January 20, 2011)

3.4.4 ORS Online Resource Scheduler


Scheduling Service - Internet Explorer
http://www.cascadeflyers.com/orsdemo/index.php?
Google
Scheduling Service
Seite
Extras

User: John Gotti (Calendar Auth.)

Actions: User Signups | Resource Signups | Member List | Utilities Menu | Help! | Log Out

NOTE: Your existing sign-ups break one or more rules. [Click for details]

Start Date: **Time:**
End Date: **Time:**
Comment:

Resource:
Cessna 172
Cessna 182
Garmin Handheld GPS
Joan Cusack
All

Select Start Date/Time

Week of: Jan 19, 2011 View: Week

[* = click for details] Now:05:21

Cessna 172

<Prev	Wed 01/19	Thu 01/20	Fri 01/21	Sat 01/22	Sun 01/23	Mon 01/24	Tue 01/25	Next>
06:00				Gotti	Gotti			06:00
07:00								07:00
08:00								08:00
09:00								09:00
10:00						Gotti		10:00
11:00								11:00
12:00								12:00
13:00			Gotti					13:00
14:00								14:00
15:00								15:00
16:00								16:00
17:00								17:00
18:00								18:00
19:00								19:00
20:00								20:00
21:00								21:00
<Prev	Wed 01/19	Thu 01/20	Fri 01/21	Sat 01/22	Sun 01/23	Mon 01/24	Tue 01/25	Next>

Powered-by ORS

Figure 3.27: planning in open resource scheduler ORS

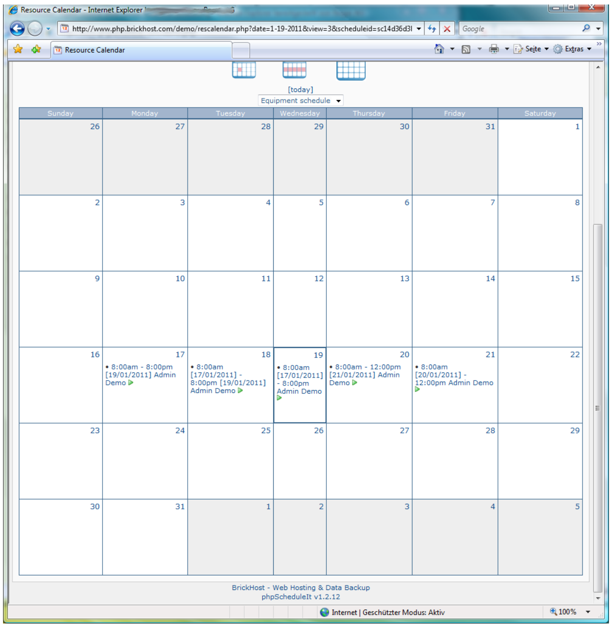
Online resource scheduler[5] is a open source tool, that offers planning functionality for planning of any resource. The application runs in a web browser, the presentation is very simple and easy.

3.4.5 phpScheduleIt

phpScheduleIt[6] is open source too, the application offers resource planing in a calendar style as in figure 3.28.

[5] available at http://ors.sourceforge.net, last access january 20, 2011

[6] available at http://www.php.brickhost.com, last access january 20, 2011



Resource Calendar - Internet Explorer

http://www.php.brickhost.com/demo/rescalendar.php?date=1-19-2011&view=3&scheduleid=sc14d36d3l

[today]

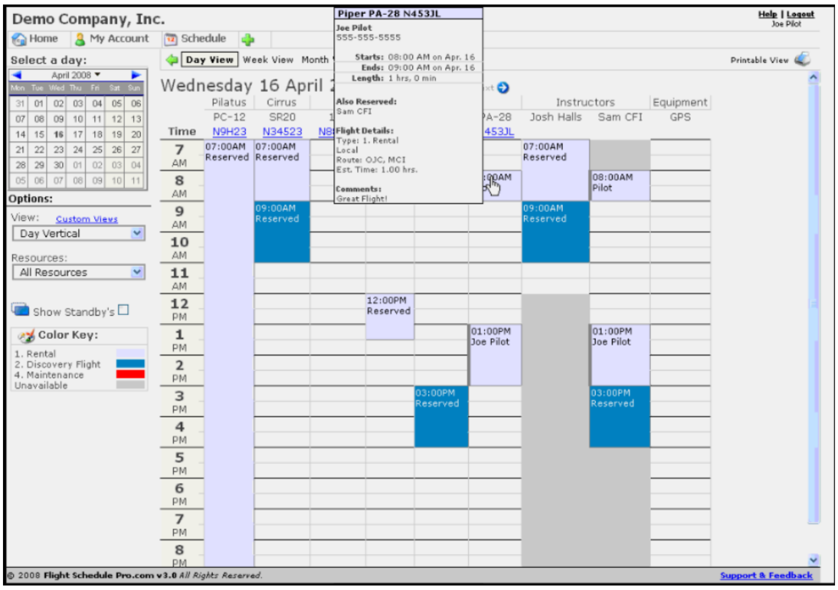
Equipment schedule

Sunday	Monday	Tuesday	Wednesday	Thursday	Friday	Saturday
26	27	28	29	30	31	1
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17 • 8:00am - 8:00pm [19/01/2011] Admin Demo	18 • 8:00am [17/01/2011] - 8:00pm [19/01/2011] Admin Demo	19 • 8:00am [17/01/2011] - 8:00pm Admin Demo	20 • 8:00am - 12:00pm [21/01/2011] Admin Demo	21 • 8:00am [20/01/2011] - 12:00pm Admin Demo	22
23	24	25	26	27	28	29
30	31	1	2	3	4	5

BrickHost - Web Hosting & Data Backup
phpScheduleIt v1.2.12

Figure 3.28: planning in phpScheduleIt

3.4.6 Flight Schedule Pro



Demo Company, Inc.

Help | Logout
Joe Pilot

Select a day: April 2008

View: Custom Views
Day Vertical

Resources:
All Resources

Show Standby's □

Color Key:
1. Rental
2. Discovery Flight
4. Maintenance
Unavailable

Wednesday 16 April 2

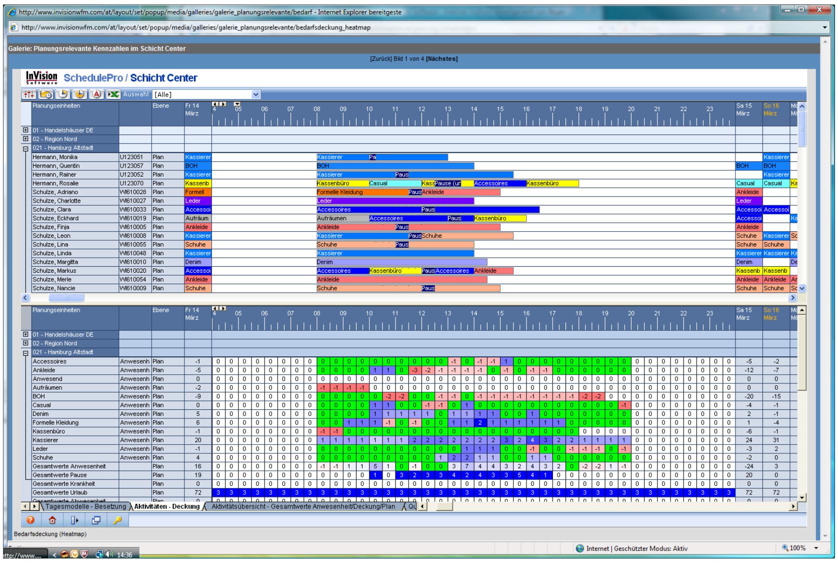
Piper PA-28 N453JL
Joe Pilot
555-555-5555
Starts: 08:00 AM on Apr. 16
Ends: 09:00 AM on Apr. 16
Length: 1 hrs, 0 min
Also Reserved:
Sam CFI
Flight Details:
Type: 1. Rental
Local
Route: OJC, MCI
Est. Time: 1.00 hrs.
Comments:
Great Flight!

© 2008 Flight Schedule Pro.com v3.0 All Rights Reserved.

Figure 3.29: planning in flight schedule pro

Flight Schedule Pro [7] is a software for scheduling of aircrafts for flight schools, universities and flying clubs. The scheduler shows reservations for every aircraft, details are shown as tooltips when pointing at one reservation as in figure 3.29.

3.4.7 Schedule Pro


http://www.invisionwfm.com/at/layout/set/popup/media/galleries/galerie_planungsrelevante/bedarf - Internet Explorer bereitgeste
http://www.invisionwfm.com/at/layout/set/popup/media/galleries/galerie_planungsrelevante/bedarfsdeckung_heatmap
Galerie: Planungsrelevante Kennzahlen im Schicht Center
[Zurück] Bild 1 von 4 [Nächstes]
InVision Software
SchedulePro / Schicht Center
Auswahl
[Alle]
Tagesmodelle - Besetzung
Aktivitäten - Deckung
Aktivitätsübersicht - Gesamtwerte Anwesenheit/Deckung/Plan
Qu
Bedarfsdeckung (Heatmap)
14:36
Internet | Geschützter Modus: Aktiv
100%

Planungseinheiten		Ebene	Fr 14 März	4 05 06 07 08 09 10 11 12 13 14 15 16 17 18 19 20 21 22 23	Sa 15 März	So 16 März
01 - Handelshäuser DE						
02 - Region Nord						
021 - Hamburg Altstadt						
Hermann, Monika	U123051	Plan	Kassierer	Kassierer Pa		Kassierer
Hermann, Quentin	U123057	Plan	BOH	BOH	BOH	BOH
Hermann, Rainer	U123052	Plan	Kassierer	Kassierer Paus		Kassierer
Hermann, Rosalie	U123070	Plan	Kassenb	Kassenbüro Casual Kass Pause (ur Accessoires Kassenbüro	Casual	Casual
Schulze, Adriano	WI610028	Plan	Formell	Formelle Kleidung Paus Ankleide	Ankleide	
Schulze, Charlotte	WI610027	Plan	Leder	Leder	Leder	
Schulze, Clara	WI610033	Plan	Accessoi	Accessoires Paus	Accessoi	Accessoi
Schulze, Eckhard	WI610019	Plan	Aufräum	Aufräumen Accessoires Paus Kassenbüro	Accessoi	
Schulze, Finja	WI610005	Plan	Ankleide	Ankleide Paus	Ankleide	
Schulze, Leon	WI610008	Plan	Kassierer	Kassierer Paus Schuhe	Schuhe	Kassierer
Schulze, Lina	WI610055	Plan	Schuhe	Schuhe Paus	Schuhe	Schuhe
Schulze, Linda	WI610048	Plan	Kassierer	Kassierer	Kassierer	Kassierer
Schulze, Margitta	WI610010	Plan	Denim	Denim	Denim	
Schulze, Markus	WI610020	Plan	Accessoi	Accessoires Kassenbüro Paus Accessoires Ankleide	Kassenb	Kassenb
Schulze, Merle	WI610054	Plan	Ankleide	Ankleide	Ankleide	Ankleide
Schulze, Nancie	WI610009	Plan	Schuhe	Schuhe Paus	Schuhe	Schuhe

Planungseinheiten		Ebene	Fr 14 März	4		05		06		07		08		09		10		11		12		13		14		15		16		17		18		19		20		21		22		23		Sa 15 März	So 16 März
01 - Handelshäuser DE																																													
02 - Region Nord																																													
021 - Hamburg Altstadt																																													
Accessoires	Anwesenh	Plan	-1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-1	0	-1	-1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-5	-2
Ankleide	Anwesenh	Plan	-5	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	-3	-2	-1	-1	-1	-1	0	-1	0	-1	-1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-12	-7
Anwesend	Anwesenh	Plan	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Aufräumen	Anwesenh	Plan	-2	0	0	0	0	0	0	0	0	-1	-1	-1	-1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
BOH	Anwesenh	Plan	-9	0	0	0	0	0	0	0	0	0	0	0	0	0	-2	-2	0	0	-1	-1	0	-1	-1	-1	-1	-1	-1	-1	-1	-2	-2	0	0	0	0	0	0	0	0	0	0	-20	-15
Casual	Anwesenh	Plan	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	-1	-1	0	1	0	0	0	0	0	0	0	0	0	0	0	-1	0	0	0	0	0	0	0	0	-4	-1
Denim	Anwesenh	Plan	5	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	0	1	1	1	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	-1
Formelle Kleidung	Anwesenh	Plan	6	0	0	0	0	0	0	0	0	0	0	1	1	1	-1	0	-1	0	0	1	1	2	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	-4
Kassenbüro	Anwesenh	Plan	-1	0	0	0	0	0	0	0	0	-1	-1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-6	-1
Kassierer	Anwesenh	Plan	20	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	2	2	2	2	2	2	2	3	2	4	3	2	2	1	1	1	1	0	0	0	0	0	0	0	0	24	31
Leder	Anwesenh	Plan	-1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	-1	0	0	-1	-1	-1	0	-1	0	0	0	0	0	0	0	0	-3	2
Schuhe	Anwesenh	Plan	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	2	2	1	1	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-2	0
Gesamtwerte Anwesenheit		Plan	16	0	0	0	0	0	0	0	0	-1	-1	1	1	5	1	0	-1	0	0	3	7	4	4	3	2	4	3	2	0	-2	-2	1	-1	0	0	0	0	0	0	0	0	-24	3
Gesamtwerte Pause		Plan	19	0	0	0	0	0	0	0	0	0	0	0	0	1	0	3	2	3	3	4	2	4	3	3	5	4	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	20	0
Gesamtwerte Krankheit		Plan	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Gesamtwerte Urlaub		Plan	72	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	72	72
Gesamtwerte Abwesenheit		Plan	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

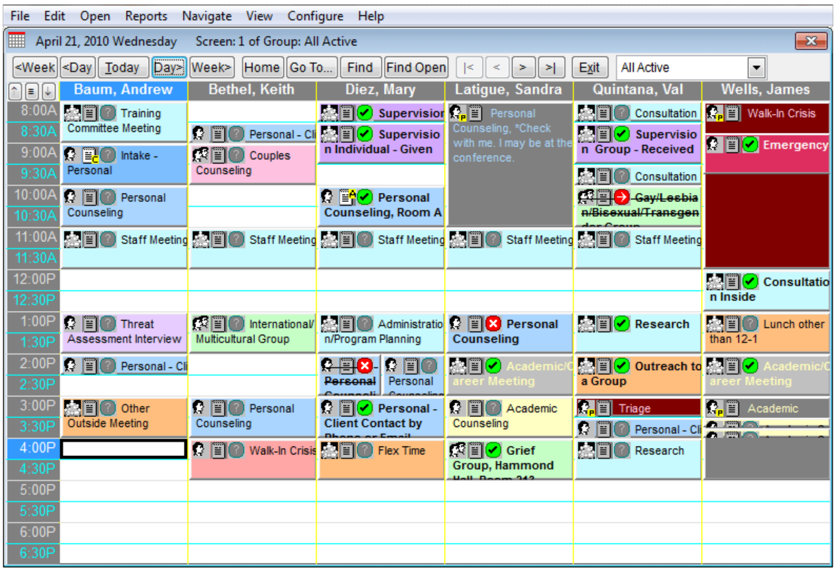
Figure 3.30: Infovision Schedule Pro

Schedule Pro [8] offers a scheduling plan, where only one day shows details and the other days are shown in a compressed form. This form of visualization is similar to DateLens from section 3.1.3. Furthermore all changes on the plan are distributed to the clients.

[7] available at http://www.flightschedulepro.com, last access january 20, 2011

[8] available at http://www.invisionwfm.com/de/produkte/personaleinsatzplanung/schedulepro, last access january 20, 2011

3.4.8 Titanium Schedule


File Edit Open Reports Navigate View Configure Help
<Week <Day Today Day> Week> Home Go To... Find Find Open |< < > >| Exit
All Active

April 21, 2010 Wednesday Screen: 1 of Group: All Active

	Baum, Andrew	Bethel, Keith	Diez, Mary	Latigue, Sandra	Quintana, Val	Wells, James
8:00A	Training Committee Meeting		Supervisior	Personal Counseling, *Check with me. I may be at the conference.	Consultation	Walk-In Crisis
8:30A		Personal - Cli	Supervision Individual - Given		Supervision Group - Received	Emergency
9:00A	Intake - Personal	Couples Counseling				
9:30A					Consultation	
10:00A	Personal Counseling		Personal Counseling, Room A		~~Gay/Lesbian/Bisexual/Transgen~~	
10:30A						
11:00A	Staff Meeting	Staff Meeting	Staff Meeting	Staff Meeting	Staff Meeting	
11:30A						
12:00P						Consultation Inside
12:30P						
1:00P	Threat Assessment Interview	International/Multicultural Group	Administration/Program Planning	Personal Counseling	Research	Lunch other than 12-1
1:30P						
2:00P	Personal - Cli		~~Personal~~ - Personal	Academic/Career Meeting	Outreach to a Group	Academic/Career Meeting
2:30P						
3:00P	Other Outside Meeting	Personal Counseling	Personal - Client Contact by	Academic Counseling	Triage	Academic
3:30P					Personal - Cli	
4:00P		Walk-In Crisis	Flex Time	Grief Group, Hammond	Research	
4:30P						
5:00P						
5:30P						
6:00P						
6:30P						

Figure 3.31: Titanium Schedule

Titanium Schedule[9] is a software for universities and counseling centers. The application also has a scheduling functionality (see figure 3.31). In the scheduling screen the user has the ability to switch between a single and multiple counselor view. In the single counselor view, the appointments of one counselor of one week is shown, in the multiple counselor view the appointments of all counselors of one day are shown.

3.4.9 Discussion

We presented some of the existing applications for planing of time depentend data. For the visualization of the data timelines and calendars are used. The multiple view from Service Ledger gives an overview in the calendar and provides details when selecting an entry. Flight Schedule Pro delivers more information when pointing on an entry (via Tooltip). We thought, that this way to provide information is useful for the user and decided to implement this functionality to A-Plan. Schedule Pro distributes all changes to the client, so all client always have the actual data — this functionality is necessary for cooperative work. Titanium Schedule uses many colors and icons on the scheduling plan to inform the user about the different assignments. Due to this fact the representation is cluttered and confusing.

[9] available at http://www.titaniumschedule.com, last access january 20, 2011

3.5 Summary

In this chapter aspects of visualization of time oriented data, interaction techniques, aspects of collaboration and an overview of optimizations for the vehicle routing problem were shown. Furthermore, different existing commercial systems were presented. For our purpose timeline visualization with overview+detail and zooming appear to be most suitable in order to provide a simple and easy-to-learn yet flexible and highly interactive representation.

Aspects of collaboration have to be taken into account, as A-Plan should be used by many people simultaneously. Therefore, the application should include mechanisms to support cooperative work.

The presented algorithms for optimization can only give an overview of this research area. In A-Plan only a simple optimization will be implemented, as for our application a fast and good suggestion is more important than finding the optimal solution.

CHAPTER 4

User Study

The application A-Plan is to be designed for a very specific user group. It therefore is necessary to take in account the capabilities and needs of these users. This human-centered design is described by Stone et al.[Stone et al., 2005, page 628]. To learn more about these groups, semistructured interviews were conducted with users of the existing system. In a semistructured interview the questions of the specific topics are defined before the interview starts. During the interview the interviewer can discuss special topics in more detail. The preparatory questions serve as support.

With the help of the interview, the following questions should be answered:

- What experience with IT systems do the people have?
- What software do they know for their work?
- What problems occur with the use of this software?
- How should an optimal system for the desired tasks look like? (visualization, collaboration, interaction)

The complete interview guideline can be seen in appendix A.

Based on the results of these interviews, prototypical types of individuals are derived. This approach corresponds to the concept of Personas, Goals and Scenarios by Alan Cooper [Cooper, 2004]. In doing so abstract personas should be elaborated which symbolizes a group of people that will use the future system. For these personas scenarios are developed - these scenarios are tasks that have to be carried out with the system to be designed. The personas and scenarios help designers to develop the prototype, as the process can be developed based on these concrete examples and not for a general user group.

For the problem analysis, task demonstration and ethnographic observations were used too. Through this deeper knowledge about the existing system and the application area can be learned. Processes and problems can be made visible and taken into account in the development of the prototype.

4.1 Method

The interview, task demonstration and ethnographic observation were carried out with the employees of the example company. For the interviews three people were available, the ethnographic observation was carried out in normal daily operation. The existing software was demonstrated by one person. The interview took about 30 minutes for each one and was done on the place of work. The task demonstration took about one hour. During the interview notes have been made.

The interview was conducted with the following persons:

- Customer care agent, female, 33 years old, in the company since 2 years
- Customer care agent, female, 26 years old, in the company since 4 years
- Customer service telephone agent and technician, male, 36 years old, in the company since 15 years

Afterwards, the answers to the interviews were compared, similarities identified and summarized. This summary is now presented as result.

4.2 Results

The respondents have considerable experience with computers. The use of SAP (or the legacy software system), Outlook and other Office applications and the Windows operating system (currently Windows Vista) are not a problem for the respondents. The user assess their knowledge with the use of computers as "good", which is also our impression.

At the remaining skills two groups can be identified: on the one hand people with better commercial practice and less know-how on the maintenance of gas appliances, on the other hand are technicians that also serviced and repaired devices. These technicians can also provide technical advice for customers on the telephone when they have troubles with their gas devices.

The staff and the management are not satisfied with the currently used software, so several alternative products from other companies were considered. The other software was better in handling than the existing product, but could not be used due the enormous costs for that software. The company tried to improve the current system by extensions and changes, but the situation still can not be seen as optimal.

The visualization and interaction techniques discussed in the previous chapter have been presented. For the respondents timelines were the most common technique, as this visualization technique is also used in the current software. LifeLines were generally not known by respondents, they seem interesting but for the required tasks as rather inappropriate. GANTT charts are familiar from Microsoft Project. Time Mosaics appeared confusing for two interviewees.

The interaction techniques zooming, overview+detail, and focus+context have been proposed. For the interviewees this is a very useful way to determine the level of detail about the information. The fisheye technique is generally unknown. Tool tips are well known from Microsoft Office applications. Overview and detail and especially zooming is well known from programs such as Google Earth or Google Maps. The existence of such functions would improve the work on the existing system, that was the opinion of all respondents.

The currently used process appears cumbersome. During a phone call the required information can not be found, the entry of a case in the currently used software takes too long. Therefore, the 'paper solution' was invented. A desired system should deliver information as quick as the 'paper solution', with the difference that all the data is stored immediately in the system. Furthermore, the new data should be displayed simultaneously on the display of other users.

4.3 Personas

Based on the interview two Personas have been derived:

- Julia Steiner: Female, 30 years old, business education, rather little technical knowledge - can hardly solve technical problems on the phone, good PC knowledge.
- Erich Gruber: Male, 50 years old, only use computer as tool for their work, is able to do everything he needs in his office but tries to avoid it, is able to solve customers problems by phone and has even the knowledge to help service technicians in the field.

4.4 Scenarios

As described by Alan Cooper [Cooper, 2004], scenarios are generated. These scenarios reflect important situations from real life of the users of the system that has to be designed. In the design process these scenarios are very helpful, as the designer can let the personas interact in the scenarios.

4.4.1 Appointment Request with Assignment

A customer (with maintenance contract) has received a postcard with an appointment. On the date of the appointment, the customer has no time and therefore he wants a service on a different date instead. She makes a phone call to the company and advise his desired date.

The phone agent opens the data of the customer. When the customer has a desired date, the phone agent tries to change the date of the assignment to the customers proposal. So the phone agent search for a free technician on the desired date. If the customer is only at home at a certain time, the telephone agent tries to take this into account - then this assignment is moved and fixed to this time. In a normal case a commitment can only be given that the technician will either come before noon or after noon, so the customer care agent move the assignment to the specified day.

When the customer does not specify a date, the old assignment has to be removed and the contract is marked for planing. The customer will be informed about the new date by post card (after the next planning).

4.4.2 Appointment Request without Assignment

A customer wants to make an appointment for service of his gas device. This can be a customer with or without service contract. In this scenario no assignment has been done so far.

The phone agent opens the customer data and look for a free technician at a desired date. When a technician is free on the desired date, the phone agent insert a new assignment on the desired date. When the customer is only at home at a certain time, the telephone agent tries to take this into account - the assignment is moved and fixed to this time.

4.4.3 Cancellation

A customer wants to cancel her appointment. She does'nt want a new appointment, so she wants no service for this year. The phone agent open the data of the customer and remove the assignment. In this case, the contract is not marked for planing — the customer is not considered at the next planning.

4.4.4 Failure

The gas unit of a client does not work and so she calls the company. The telephone agent tries to ask by telephone for example what error code on the display of the device is issued. In many cases it is an operator error. In this case the problem can be solved by phone and thus the use of a technician locally at the customer can be avoided.

If the correction of the issue is not possible by telephone, a technician is needed at the customer. The phone agent opens the data of the customer and marks the customer as 'open failure'. The telephone number of the customer is quoted, so the customer can be contacted by an technician.

The telephone agent tries to contact technicians and find out who has time to spare. If she finds a free technician, the failure is assigned to the technician who will fix the failure of the customer, as she does this by inserting the open failure from the previous step as assignment to the technician. The customer is removed from the list of open failures.

CHAPTER 5

Conceptual Design

In Chapter 2 an existing solution for the dispatching of assignments of a service provider was analyzed. Many problems with the existing solution were found:

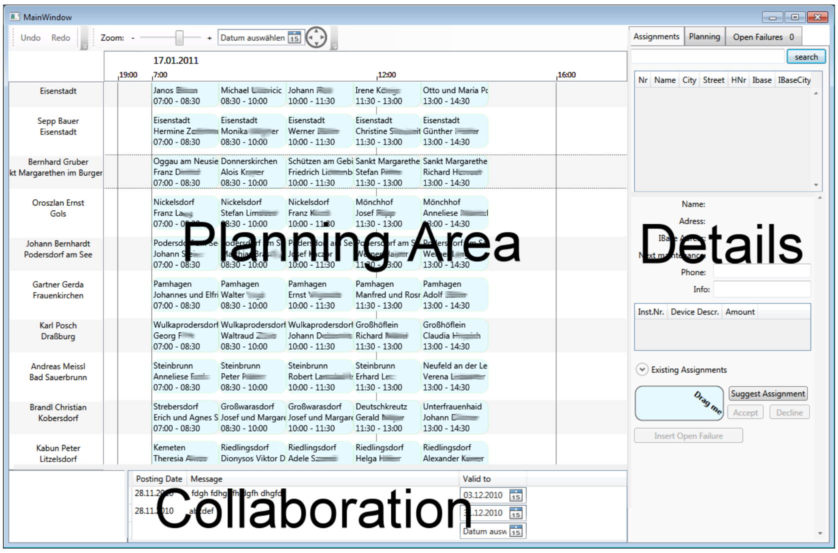
- Confusing visualization of the plan
- Inconsistend behavior
- Overloaded interface
- Complex handling of three applications
- Unclear error messages
- No safe exploration, no undo
- No consideration of collaboration

In the new application we wanted to avoid these usability issues. Furthermore, to support and ease the workflow of users, an automated planning function should be integrated into the software and the new application should allow multiple user to work simultaneously with the data.

In Chapter 3 we reported about different visualization and interaction techniques. We also discussed aspects of collaboration and techniques to implement an application that supports collaboration. In this chapter we will discuss how this techniques can be used in a real world application

5.1 Design method

Alan Cooper writes in About Face [Cooper et al., 2007] six steps to define an interaction framework:



17.01.2011

19:00 7:00 12:00 16:00

Eisenstadt	Janos [redacted] 07:00 - 08:30	Michael [redacted] 08:30 - 10:00	Johann [redacted] 10:00 - 11:30	Irene [redacted] 11:30 - 13:00	Otto und Maria Pc 13:00 - 14:30
Sepp Bauer Eisenstadt	Eisenstadt Hermine [redacted] 07:00 - 08:30	Eisenstadt Monika [redacted] 08:30 - 10:00	Eisenstadt Werner [redacted] 10:00 - 11:30	Eisenstadt Christine [redacted] 11:30 - 13:00	Eisenstadt Günther [redacted] 13:00 - 14:30
Bernhard Gruber kt Margarethen im Burger	Oggau am Neusie Franz [redacted] 07:00 - 08:30	Donnerskirchen Alois [redacted] 08:30 - 10:00	Schützen am Gebi Friedrich [redacted] 10:00 - 11:30	Sankt Margarethe Stefan [redacted] 11:30 - 13:00	Sankt Margarethe Richard [redacted] 13:00 - 14:30
Oroszlan Ernst Gols	Nickelsdorf Franz [redacted] 07:00 - 08:30	Nickelsdorf Stefan [redacted] 08:30 - 10:00	Nickelsdorf Franz [redacted] 10:00 - 11:30	Mönchhof Josef [redacted] 11:30 - 13:00	Mönchhof Anneliese [redacted] 13:00 - 14:30
Johann Bernhardt Podersdorf am See	Podersdorf am See Johann [redacted] 07:00 - 08:30	Podersdorf am S Mathias [redacted] 08:30 - 10:00	Podersdorf am Se Josef [redacted] 10:00 - 11:30	Podersdorf am S Werner [redacted] 11:30 - 13:00	Podersdorf am Se Werner [redacted] 13:00 - 14:30
Gartner Gerda Frauenkirchen	Pamhagen Johannes und Elfri 07:00 - 08:30	Pamhagen Walter [redacted] 08:30 - 10:00	Pamhagen Ernst [redacted] 10:00 - 11:30	Pamhagen Manfred und Rosr 11:30 - 13:00	Pamhagen Adolf [redacted] 13:00 - 14:30
Karl Posch Draßburg	Wulkaprodersdorf Georg [redacted] 07:00 - 08:30	Wulkaprodersdorf Waltraud [redacted] 08:30 - 10:00	Wulkaprodersdorf Johann [redacted] 10:00 - 11:30	Großhöflein Richard [redacted] 11:30 - 13:00	Großhöflein Claudia [redacted] 13:00 - 14:30
Andreas Meissl Bad Sauerbrunn	Steinbrunn Anneliese [redacted] 07:00 - 08:30	Steinbrunn Peter [redacted] 08:30 - 10:00	Steinbrunn Robert [redacted] 10:00 - 11:30	Steinbrunn Erhard [redacted] 11:30 - 13:00	Neufeld an der Le Verena [redacted] 13:00 - 14:30
Brandl Christian Kobersdorf	Strebersdorf Erich und Agnes S 07:00 - 08:30	Großwarasdorf Josef und Margare 08:30 - 10:00	Großwarasdorf Josef und Margare 10:00 - 11:30	Deutschkreutz Gerald [redacted] 11:30 - 13:00	Unterfrauenhaid Johann [redacted] 13:00 - 14:30
Kabun Peter Litzelsdorf	Kemeten Theresia [redacted]	Riedlingsdorf Dionysos Viktor D	Riedlingsdorf Adele [redacted]	Riedlingsdorf Helga [redacted]	Riedlingsdorf Alexander [redacted]

Nr	Name	City	Street	HNr	Ibase	IBaseCity

Name:
Adress:
IBase Adress:
Next maintenance:
Phone:
Info:

Inst.Nr.	Device Descr.	Amount

Posting Date	Message	Valid to
28.11.2010	fdgh fdhgfhdgfh dhgfd	03.12.2010
28.11.2010	abcdef	.12.2010
		Datum ausw

Figure 5.1: Screen layout of the application

1. Define form factor, posture, and input methods: in our case a windows desktop application.

2. Define functional and data elements: All functional and data elements will be described in the language of user interface representations. It is also important to be aware of principles and patterns.

3. Determine functional groups and hierarchy: The list of functional and data elements from the previous step will be grouped and their hierarchy will be defined.

4. Sketch the interaction framework: The first sketches have been made with a sheet of paper and a pencil. Later the design was implemented using the software Microsoft Blend, so the draft could be better discussed with an interface expert.

5. Construct Key path scenarios: The key path scenarios were discussed in section 4.4 (personas, goals and scenarios). Using these, the design was guided and evaluated during the design.

6. Check designs with validation scenarios: Here less frequent scenarios are analyzed, like edge cases, key path variant scenarios (variants of the key path scenarios) and necessary use scenarios (actions that must be performed, but not so often).

During the design process we iteratively went through this steps.

5.2 Screen layout

In the figure 5.1 you can see the layout of the application. The screen is divided into three areas:

- The Planning area is where the user can view, insert, move and delete assignments.
- Details are shown on the right side of the window. In this location the user can also start actions like searching for a customer and planning of assignments.
- In the lower area of the window the user is supported in the cooperation with other users.

The size of each area can be changed by the user, the space available will be used by the elements.

Alan Cooper recommended in About Face [Cooper et al., 2007] to dispose of save buttons and avoid using ok buttons. Instead every action should be saved automatically and an undo function should be available for the user to take back unwanted operations. The application should be designed according to this principle, every transaction should be saved immediately and should be distributed to the other clients.

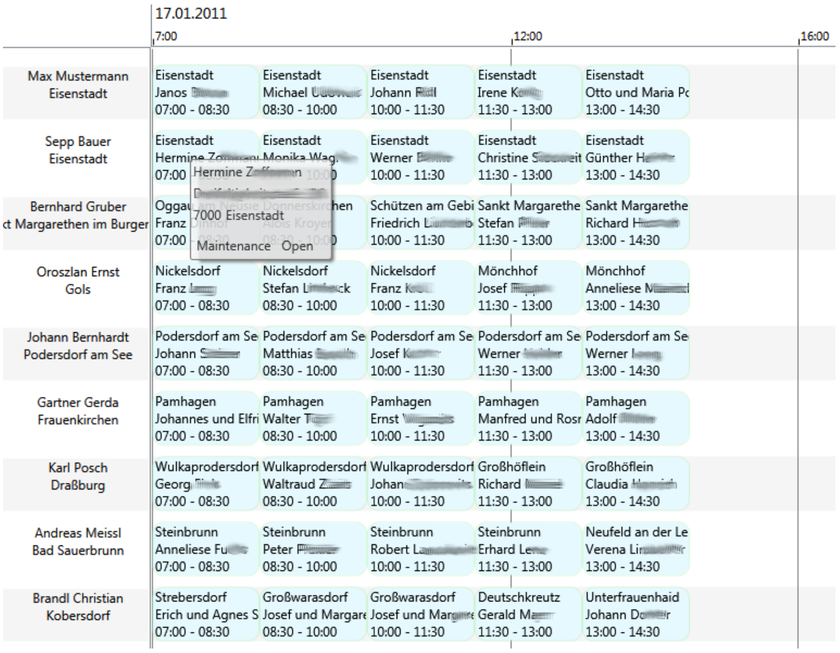
5.3 Planning

After investigating the possible visualization techniques in chapter 3 it was decided to use traditional TimeLines for the visualization of the assignment plan, because this visualization seems the best applicable for a plan that should be interactive and cooperatively used by more than one user. The assignments are displayed as boxes showing the most important information about the assignment directly as text (city, customer, time). The shape and the color of the box also gives information about the assignment: When the box has rounded corners, the time of the assignment is not fixed, when the corners are angular the time is fixed. The color of the assignment gives quick information about the type of the assignment (maintenance or failure).

Pointing at an assignment a tooltip shows more details. Clicking on an assignment all information about the customer is shown in the detail area. This resembles an overview first and detail on demand approach that should be easier to understand for the user than focus and context techniques like fisheye distortion.

The user can navigate by smooth panning and zooming as in Google Earth. The user can zoom by using the mouse wheel, a slider in the tool bar or two buttons next to the slider. Panning is done by dragging the background of the plan or by using a navigation element in the tool bar. Furthermore, the user has the option to navigate by selecting a date.

Changes in the plan are done by direct manipulation. The user can drag and drop assignments in the plan. This is also the way to insert new assignments. In this case the user can drag a mockup assignment into the plan. All changes are distributed to

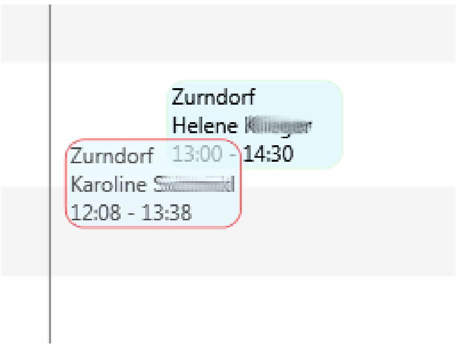


17.01.2011

Max Mustermann Eisenstadt	Eisenstadt Janos [redacted] 07:00 - 08:30	Eisenstadt Michael [redacted] 08:30 - 10:00	Eisenstadt Johann R[redacted] 10:00 - 11:30	Eisenstadt Irene K[redacted] 11:30 - 13:00	Eisenstadt Otto und Maria Pc 13:00 - 14:30
Sepp Bauer Eisenstadt	Eisenstadt Hermine Z[redacted] 07:00	Eisenstadt Monika Wag[redacted] 10:00	Eisenstadt Werner [redacted] 10:00 - 11:30	Eisenstadt Christine S[redacted]eit 11:30 - 13:00	Eisenstadt Günther H[redacted] 13:00 - 14:30
Bernhard Gruber ct Margarethen im Burger	Oggau am Neusie Franz Dinhof 07:00 - 08:30	Donnerskirchen Alois Kroyer 08:30 - 10:00	Schützen am Gebi Friedrich L[redacted] 10:00 - 11:30	Sankt Margarethe Stefan [redacted] 11:30 - 13:00	Sankt Margarethe Richard H[redacted] 13:00 - 14:30
Oroszlan Ernst Gols	Nickelsdorf Franz [redacted] 07:00 - 08:30	Nickelsdorf Stefan L[redacted]ck 08:30 - 10:00	Nickelsdorf Franz K[redacted] 10:00 - 11:30	Mönchhof Josef [redacted] 11:30 - 13:00	Mönchhof Anneliese M[redacted] 13:00 - 14:30
Johann Bernhardt Podersdorf am See	Podersdorf am Se Johann S[redacted] 07:00 - 08:30	Podersdorf am Se Matthias [redacted] 08:30 - 10:00	Podersdorf am Se Josef K[redacted] 10:00 - 11:30	Podersdorf am Se Werner [redacted] 11:30 - 13:00	Podersdorf am Se Werner [redacted] 13:00 - 14:30
Gartner Gerda Frauenkirchen	Pamhagen Johannes und Elfri 07:00 - 08:30	Pamhagen Walter T[redacted] 08:30 - 10:00	Pamhagen Ernst [redacted] 10:00 - 11:30	Pamhagen Manfred und Rosr 11:30 - 13:00	Pamhagen Adolf [redacted] 13:00 - 14:30
Karl Posch Draßburg	Wulkaprodersdorf Georg [redacted] 07:00 - 08:30	Wulkaprodersdorf Waltraud Z[redacted] 08:30 - 10:00	Wulkaprodersdorf Johan[redacted] 10:00 - 11:30	Großhöflein Richard [redacted] 11:30 - 13:00	Großhöflein Claudia [redacted] 13:00 - 14:30
Andreas Meissl Bad Sauerbrunn	Steinbrunn Anneliese Fu[redacted] 07:00 - 08:30	Steinbrunn Peter [redacted] 08:30 - 10:00	Steinbrunn Robert L[redacted] 10:00 - 11:30	Steinbrunn Erhard Le[redacted] 11:30 - 13:00	Neufeld an der Le Verena Li[redacted] 13:00 - 14:30
Brandl Christian Kobersdorf	Strebersdorf Erich und Agnes S 07:00 - 08:30	Großwarasdorf Josef und Margare 08:30 - 10:00	Großwarasdorf Josef und Mar[redacted] 10:00 - 11:30	Deutschkreutz Gerald M[redacted] 11:30 - 13:00	Unterfrauenhaid Johann D[redacted] 13:00 - 14:30

Hermine Z[redacted]
[redacted]
7000 Eisenstadt
Maintenance Open

Figure 5.2: The planning area



Zurndorf
Helene [redacted]
13:00 - 14:30

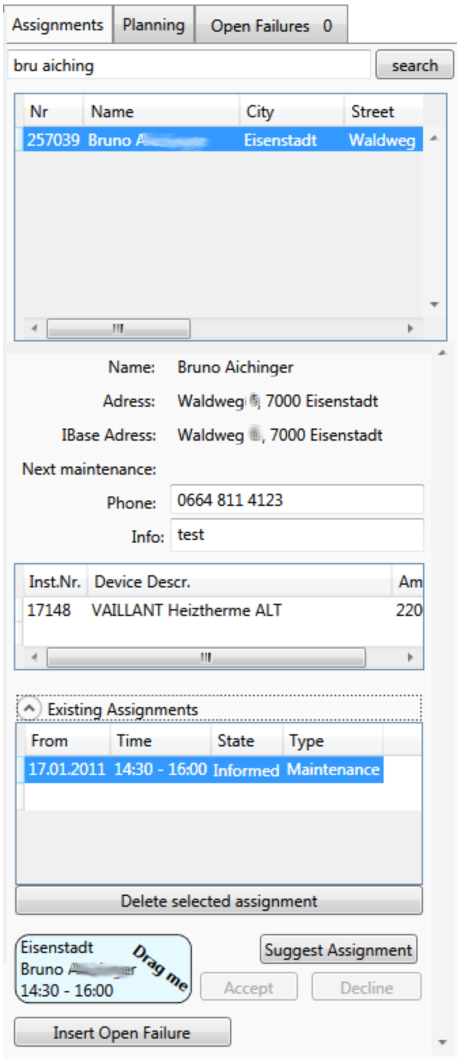
Zurndorf
Karoline S[redacted]
12:08 - 13:38

Figure 5.3: The assignment is shown transparent while dragging

all clients immediately. While one user is dragging an assignment all other users can follow the movement of the assignment. Until the assignment is dropped the rectangle is rendered transparent on the plan (see figure 5.3).

5.4 Detail

The detail area has more functionality than showing the details: the user can search for customers, can insert new assignments, insert new open failures and the user can plan customers with due maintenance contracts. This part of the interface is organized using



bru aiching

Nr	Name	City	Street
257039	Bruno A[redacted]	Eisenstadt	Waldweg

Name: Bruno Aichinger

Adress: Waldweg [redacted], 7000 Eisenstadt

IBase Adress: Waldweg [redacted], 7000 Eisenstadt

Next maintenance:

Phone: 0664 811 4123

Info: test

Inst.Nr.	Device Descr.	Am
17148	VAILLANT Heiztherme ALT	220

From	Time	State	Type
17.01.2011	14:30 - 16:00	Informed	Maintenance

Eisenstadt
Bruno A[redacted]
14:30 - 16:00

Figure 5.4: The detail and action area

three tabs:

- the assignments tab
- the planning tab
- the open failures tab

The planning area and the detail area interact, when assignments are clicked in one area, the corresponding element is also choosen and shown in the other area. Changes in one area are immeadiatly shown in the other area.

5.4.1 Assignments

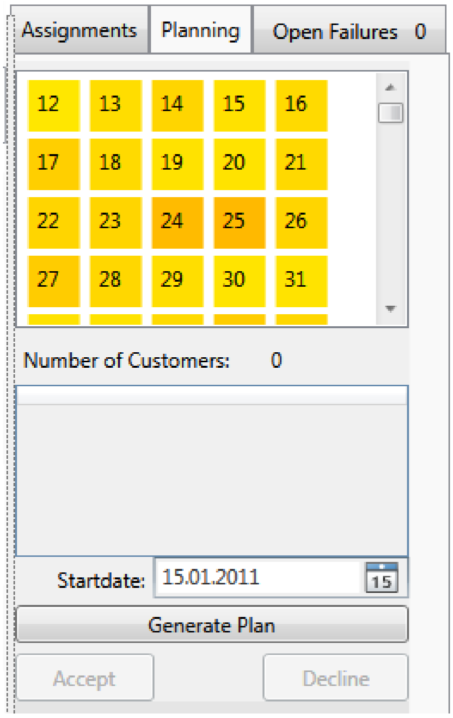
The first tab offers the following functionality:

- search: Customers can be searched by entering their data about the customers in the search box. The application always provides a full text search over all data of the customers. For example a search 'Schneider Eisenstadt' will provide all customers with 'Schneider' in the name and living in 'Eisenstadt', furthermore the result will also contain customers with 'Schneider' in the name and living in the street 'Eisenstadt Straße' for example. This way to search is similar to internet search engines, which is familiar to most users. Furthermore, it should be more user friendly than a conventional search functionality where the user has to specify the search criteria for each database field explicitly.
- information: All information about a selected assignment is shown.
- edit: Assignment data and information like the phone number can be edited in place.
- insert: For the insertion of assignments the user has two alternatives: drag´n´drop the mockup to the plan or let the system suggest a date for an assignment. Furthermore a new open failure can be inserted - the open failures are displayed in the 'Open failure' tab.

5.4.2 Planning

The planning tab is shown in figure 5.5. In our application we represented the number of customers that have to be serviced using a heat map. The heat map has been described by Wilkinson [Wilkinson, 2009]:

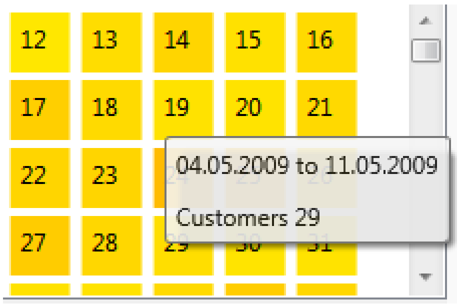
> It consists of a rectangular tiling with each tile shaded on a color scale to represent the value of the corresponding element of the data matrix.



Number of Customers: 0

Startdate: 15.01.2011

Figure 5.5: Planning of assignments of customers with due maintenance contracts



04.05.2009 to 11.05.2009

Customers 29

Figure 5.6: Visualization of open customers using a heatmap

The heat map shows the number of open customers via color intensity. Each square shows the data of one week (the number on the square tells the user which calendar week is shown). When the user points to a square on the heatmap detailinformation is shown (date and number of customers, figure 5.6). When the user clicks on the squares, the corresponding customers are selected for planning.

With the set of selected customers a plan will be generated using the savings algorithm, as this algorithm is used in many applications in practice [Domschke, 1990] and delivered good results in our tests. The calculated assignment plan is shown with transparent assignments in the plan until the user accept the proposed plan.

5.4.3 Open Failures

The list of open failures is displayed on the third tab. The layout of this tab is very similar to the assignments tab, with the only difference that open failure assignments can't be inserted automatically.

5.5 Collaboration

In the collaboration area of the window two lists are shown: the active users and common messages. In the list of active users all users working on the system are shown, as soon as they close the application they disappear on all clients. The list of common messages can be used to notify the other users about important news.

CHAPTER 6

Implementation

In the next step a prototype of the design has been implemented. Later the implemented prototype was evaluated by potential users.

For the implementation Visual Studio 2010 with .net 4.0 was used. For the user interface the WPF [1] framework was used, the communication between the server and the clients is implemented with WCF [2]. As database Microsoft SQL Server 2010 Express Edition is in use.

6.1 Model-View-Viewmodel (MVVM)

The MVVM (Model-View-Viewmodel) Pattern has been developed for WPF and Silverlight applications. It is based on the Model-View-Controller Pattern. The goal of MVVM is to keep the code as maintainable as possible. This is achieved by separating the user interface from the logic as strongly as possible. This is only possible by the so called data binding infrastructure of WPF. When binding a property of the user interface to the view model the two elements (user interface and view model) are only loosely coupled, so in the view model no code for accessing the interface is needed. Another advantage is the testability of the view models, as this can be done programmatically.

The data binding has to know about updates of the data in the view model. This is done by implementing the INotifyPropertyChange interface in the view model and after each change of the data the binding has to be informed by the PropertyChanged event. For collections the ObservableCollection<T> class already exists, when using this class the framework is informed about adding and deleting items. In this way every update is automatically displayed in the user interface.

[1] WPF = windows presentation foundation

[2] WCF = windows communication foundation

Thus the classes are divided into view (the user interface), model (those classes are responsible for the data acquisition and data storage) and for the view model (the data preparation for the view and the logic).

The user interface of WPF applications is defined in XAML (an XML language). When the MVVM pattern is consistently applied no single line of code is needed in the corresponding codebehind file. The reference to its code is done by binding to objects that implement the ICommand interface. A class is implemented for each command, the command can be bound to multiple actions in the user interface, for example to some buttons or a menu item.

The ICommand interface has two methods: execute and canExecute. In the execute method, the actual logic for the command is implemented. The method canExecute notifies WPF whether the command can be executed or not. The corresponding user interface controls are switched inactive, when the method canExecute returns false. The method canExecute is called by WPF cyclically, so the developer does not have to take care about updating the user interface.

Another advantage when using MVVM is the possibility to use Microsoft Blend. Microsoft Blend is a tool for the development of the user interface. In comparison to the possibilities of Microsoft Visual Studio, Microsoft Blend offers better functionality and support for the development of user interfaces. Blend only works with the XAML files, therefore it is necessary that no code behind exists.

6.2 MVVM light toolkit

In learning WPF and MVVM we found the MVVM toolkit from the company Galasoft[3]. It is a freely available library which facilitates working with the MVVM pattern WPF. The library provides the following functions:

- EventToCommand behavior
- RelayCommand and RelayCommand<T>
- Messenger

6.2.1 EventToCommand behavior

Normally commands can only be bound to few events of the user interface (for example clicking on a button). By EventToCommand behaviors (behaviors are special classes in WPF that can be attached to user interface elements) commands are not limited to this few specific events. Only through this functionality, it is possible to avoid code in the code behind file.

[3] the library can be accessed at http://www.galasoft.ch/mvvm/getstarted, last access on January 24, 2011
an interesting video is available at http://ecn.channel9.msdn.com/o9/mix/10/wmv/EX14.wmv, last access on January 28, 2011

6.2.2 RelayCommand and RelayCommand<T>

The normal use of ICommand implementations has the disadvantage that for each Command a separate class has to be implemented. This approach produces very much code which partly has the same content. The RelayCommand avoids this. When instantiating the class the constructor takes two parameter: the first parameter is a delegate to a method that has to be called for execute and the second parameter is a delegate for the method canExecute.

The call is made even shorter using lambda expressions. Lambda expressions are language constructs from functional programming, which can be utilized in the new versions of C#. The following example demonstrates the use of RelayCommand using lambda expressions (from the command for the search function in A-Plan):

```
public RelayCommand<string> SearchCustomers
{
   get;
   private set;
}

...

SearchCustomers = new RelayCommand<string>(
                    m => { proxy.GetCustomer(m);
                           IsBusy = true; },
                    m => !IsBusy);
```

The first parameter is the call to the execute method, the parameter of type string is also used. The second parameter is filled with the value of the boolean variable IsBusy, thus the command can not be executed as long as a search query runs (for example the button that is bound to this command, can not be clicked as long as the variable IsBusy is set to true).

6.2.3 Messenger

In the MVVM light toolkit also a messanger function exists. This is especially useful to communicate between two view models. Normally the view models don't know each other, so they have no reference to the other view model. When the messenger is used, methods can be registered to the messenger. Every time the send method is called from the messenger, all registered methods are notified. This functionality has been used in the project to communicate between the planning area and the detail area.

6.3 Middleware

In section 3.2.4 we already discussed techniques to share the data between many clients to support collaboration. Groupware frameworks offer a high abstraction level and seem

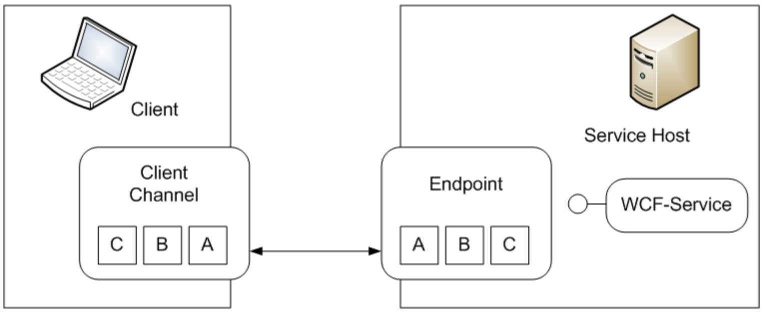


Figure 6.1: The ABC-Concept of WCF [Kuhrmann and Beneken, 2006]

to be practicable tools to develop the functionality of collaboration in A-Plan. First we learned how to use the framework and we analyzed the GroupLab.Networking library[4] in more detail. This library has the most mature development status. The other libraries[5] discussed in 3.2.4 have been developed at the university of Calgary. In our opinion the abstraction as shared dictionary is a good way to distribute updates of data without taking care about low level routines that handle the communication over the network.

In our experiments we had to find out that the elements of the shared dictionary can only be simple data types (such as string or int). Own classes, for example a class for an assignment, can not be used in these libraries. Due this disadvantage the data needs to be transferred in a sort of marshalling mechanism which makes the use of this technology very complicated.

6.3.1 Windows Communication Foundation (WCF)

Therefore we have discarded the use of groupware frameworks and decided to use WCF (we already introduced WCF in section 3.2.4). WCF is part of the .net framework and that is why we have chosen this technology. Kuhrman gives in [Kuhrmann and Beneken, 2006] a good overview of functionality and usage of WCF (figure 6.1).

The three components A-B-C have the following meaning:

- Address: specifies where the service is accessible
- Binding: describes how the service communicates, for example using TCP or HTTP
- Contract: describes what functions the service offers

In our project we used net.tcp as binding. This binding is restricted to WCF applications only, but offers better performance. The interoperability provided by other bindings was not necessary for our application.

[4] The library is available at http://grouplab.cpsc.ucalgary.ca/software/networking/, last access January 22, 2011

[5] The other libraries like Groupkit or SDG Toolkit are available at http://grouplab.cpsc.ucalgary.ca/software, last access January 22, 2011

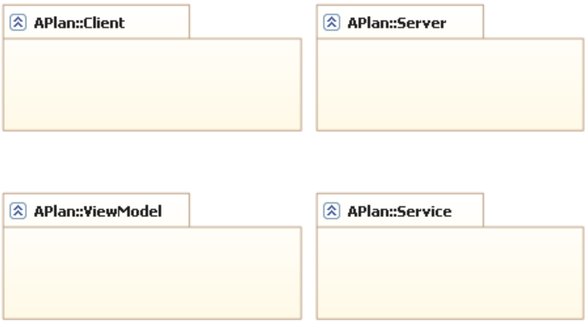


Figure 6.2: Overview of namespaces of A-Plan

The contracts are defined in the two interfaces IAPlanService and IAPlanCallback. IAPlanService defines the interface of the server and IPlanCallback defines the interface of the client which can be called by the server. This is necessary since all calls are made asynchronously. If the call would be synchronously the processing of the application would be suspended until the call of the service is terminated. These interruptions are not very user friendly, so we decided to develop the communication between server and client with asynchronous calls. At the end of each function on the server the client is called with the result as parameter.

6.4 Architecture

Figure 6.2 shows the namespaces of A-Plan. APlanClient includes all classes for the client application (controls and the main window), APlanServer has the program code to start and shut down the server, APlanViewModel includes all view model classes (shown in figure 6.4.3) according to the MVVM pattern and APlanService includes the classes for data access (figure 6.3) and the service implementation (figure 6.4).

6.4.1 Data Classes

Figure 6.3 shows the implemented classes for the access to the database, the classes correspond to the tables in the database. For the access to the database language integrated query (LINQ) is used.

The classes offer the following functionality:

- Customer stores data about the customer, like name, city, street, etc.
- City includes data about cities - for example the city name, the latitude and longitude (necessary for route calculation)
- Regiogroup divides the cities into regional areas

- IBase can be understood as premise of the customer. The address of a IBase can differ from the address of the customer, this often happens when a customer has more then one house (for example a company with many branches).

- Installation is an installed gas device.

- Assignment includes the necessary data about a single assignment.

- Message stores the information for the exchange of messages (for collaboration).

6.4.2 Service Classes

The service classes (figure 6.4) in A-Plan are:

- APlanService offers all methods that are accessible as service (via WCF).

- Routeplan is a calculated route plan using the savings algorithm.

- Route is a helper class that represents a route of one technician on a specified date.

6.4.3 Viewmodel Classes

The view model classes deliver the data as needed for the user interface. Base class for all view model classes is the ViewModelBase class from the MVVM toolkit. By inheriting from ViewModelBase all classes have the PropertyChanged event and the OnPropertyChanged method. The only thing that is necessary is to call the RaisePropertyChanged method when a property is updated.

The following view model classes have been implemented:

MainViewModel

The most important view model class communicates with the server and offers the data for the main window

EmployeeViewmodel

The EmployeeViewmodel class offers the data needed to view the employee information in the user interface.

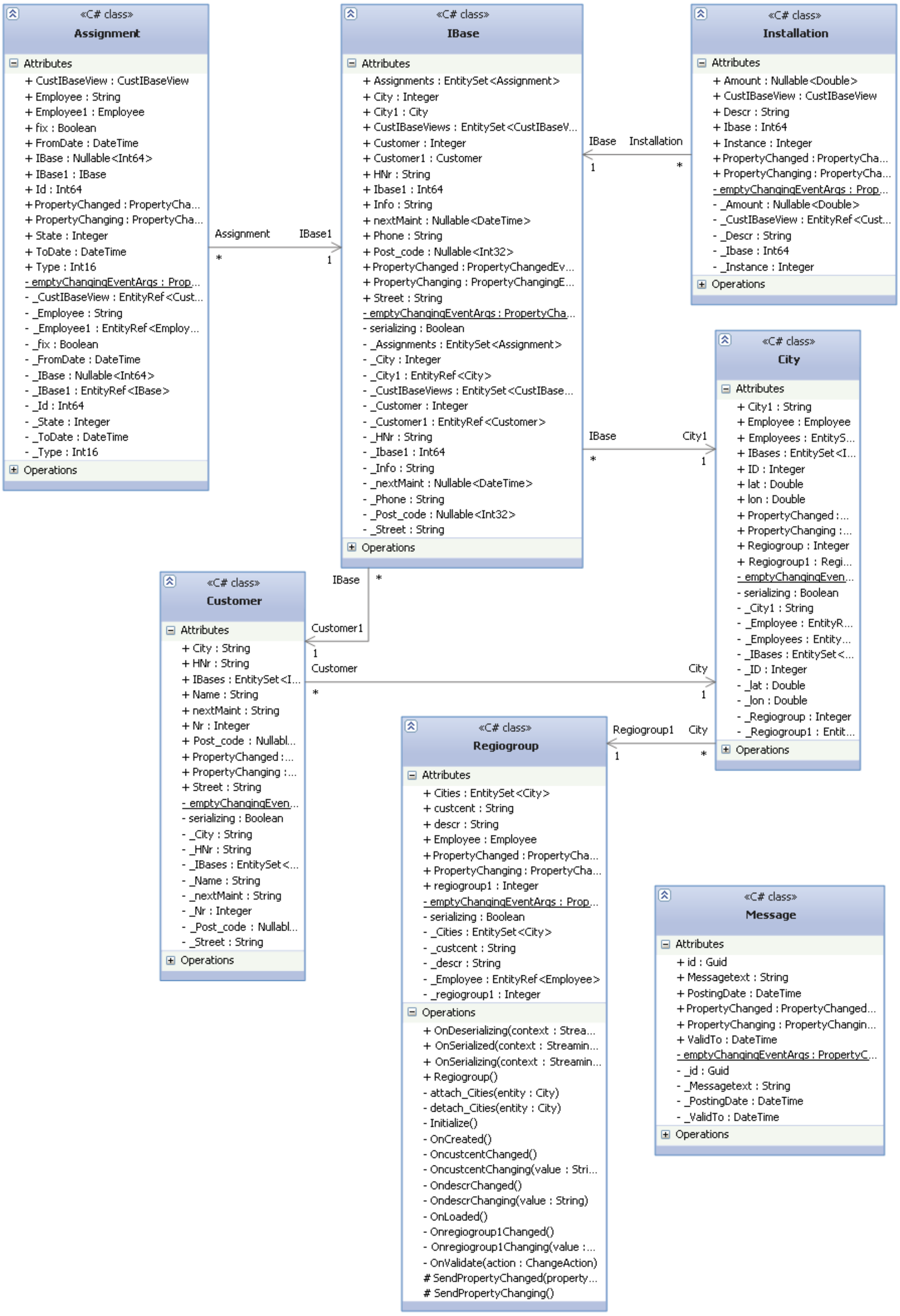


Figure 6.3: Data classes of A-Plan

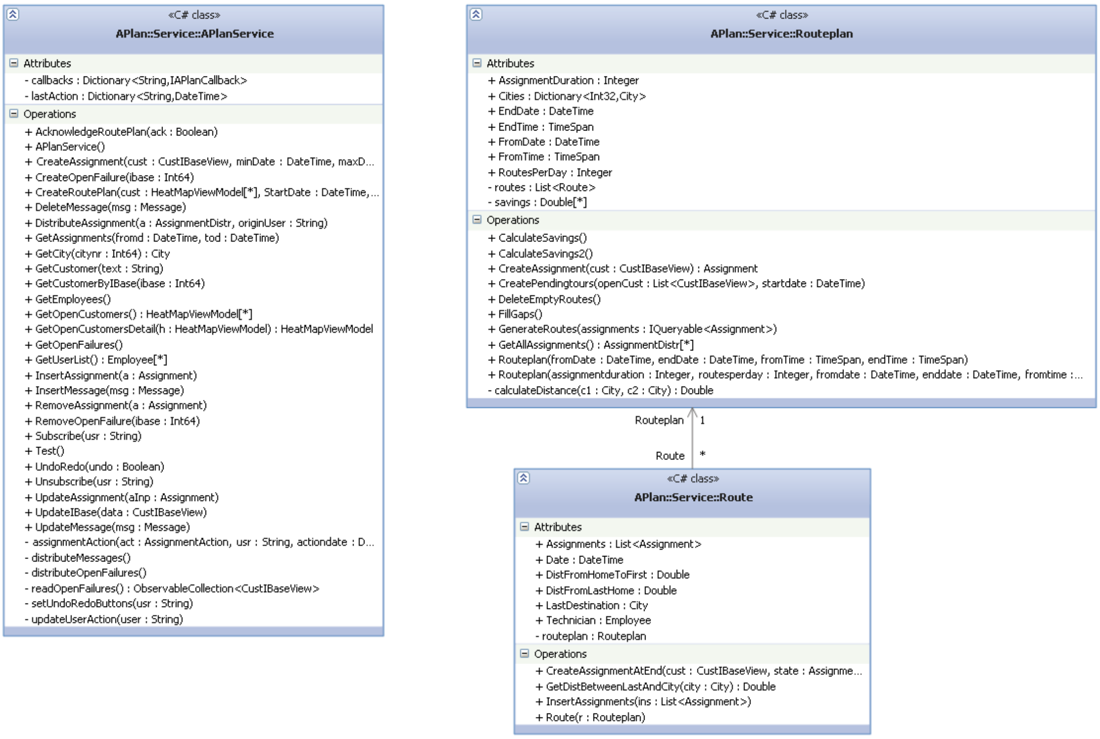

«C# class»
APlan::Service::APlanService
Attributes
- callbacks : Dictionary<String,IAPlanCallback>
- lastAction : Dictionary<String,DateTime>
Operations
+ AcknowledgeRoutePlan(ack : Boolean)
+ APlanService()
+ CreateAssignment(cust : CustIBaseView, minDate : DateTime, maxD...
+ CreateOpenFailure(ibase : Int64)
+ CreateRoutePlan(cust : HeatMapViewModel[*], StartDate : DateTime,...
+ DeleteMessage(msg : Message)
+ DistributeAssignment(a : AssignmentDistr, originUser : String)
+ GetAssignments(fromd : DateTime, tod : DateTime)
+ GetCity(citynr : Int64) : City
+ GetCustomer(text : String)
+ GetCustomerByIBase(ibase : Int64)
+ GetEmployees()
+ GetOpenCustomers() : HeatMapViewModel[*]
+ GetOpenCustomersDetail(h : HeatMapViewModel) : HeatMapViewModel
+ GetOpenFailures()
+ GetUserList() : Employee[*]
+ InsertAssignment(a : Assignment)
+ InsertMessage(msg : Message)
+ RemoveAssignment(a : Assignment)
+ RemoveOpenFailure(ibase : Int64)
+ Subscribe(usr : String)
+ Test()
+ UndoRedo(undo : Boolean)
+ Unsubscribe(usr : String)
+ UpdateAssignment(aInp : Assignment)
+ UpdateIBase(data : CustIBaseView)
+ UpdateMessage(msg : Message)
- assignmentAction(act : AssignmentAction, usr : String, actiondate : D...
- distributeMessages()
- distributeOpenFailures()
- readOpenFailures() : ObservableCollection<CustIBaseView>
- setUndoRedoButtons(usr : String)
- updateUserAction(user : String)
«C# class»
APlan::Service::Routeplan
Attributes
+ AssignmentDuration : Integer
+ Cities : Dictionary<Int32,City>
+ EndDate : DateTime
+ EndTime : TimeSpan
+ FromDate : DateTime
+ FromTime : TimeSpan
+ RoutesPerDay : Integer
- routes : List<Route>
- savings : Double[*]
Operations
+ CalculateSavings()
+ CalculateSavings2()
+ CreateAssignment(cust : CustIBaseView) : Assignment
+ CreatePendingtours(openCust : List<CustIBaseView>, startdate : DateTime)
+ DeleteEmptyRoutes()
+ FillGaps()
+ GenerateRoutes(assignments : IQueryable<Assignment>)
+ GetAllAssignments() : AssignmentDistr[*]
+ Routeplan(fromDate : DateTime, endDate : DateTime, fromTime : TimeSpan, endTime : TimeSpan)
+ Routeplan(assignmentduration : Integer, routesperday : Integer, fromdate : DateTime, enddate : DateTime, fromtime :...
- calculateDistance(c1 : City, c2 : City) : Double
Routeplan
1
Route
*
«C# class»
APlan::Service::Route
Attributes
+ Assignments : List<Assignment>
+ Date : DateTime
+ DistFromHomeToFirst : Double
+ DistFromLastHome : Double
+ LastDestination : City
+ Technician : Employee
- routeplan : Routeplan
Operations
+ CreateAssignmentAtEnd(cust : CustIBaseView, state : Assignme...
+ GetDistBetweenLastAndCity(city : City) : Double
+ InsertAssignments(ins : List<Assignment>)
+ Route(r : Routeplan)

Figure 6.4: Service classes of A-Plan

AssignmentViewModel

This class provides the information for the assignments. A transformation between time information (date from and to) and spatial information (the coordinate on the assignment plan) is also performed by this class, an update on one information automatically updates the other information as the date of an assignment corresponds to certain coordinates on the timeline. The control in the user interface is bound to the X and Y properties of the object, when a user drag an assignment not only the coordinates are updated, the setter properties also set the date of the assignment.

The coordinate of the assignment can be calculated using equation 6.1. An explanation of the variables gives figure 6.5.

$$x = (Hours * 60 + Minutes) * PixelPerMinute + Days * DayWidth \tag{6.1}$$

The width of a day in pixel is calculated in equation 6.2.

$$DayWidth = (EndTimePerDay - StartTimePerDay) * PixelPerMinute + DistancePixel \tag{6.2}$$

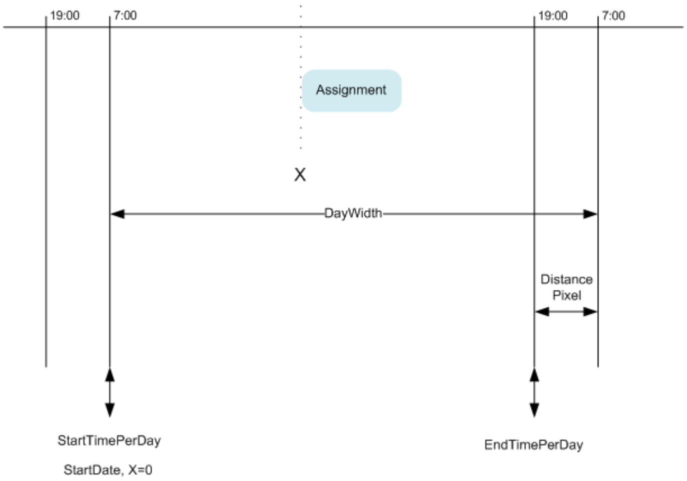


Figure 6.5: Calculation of coordinate from given date

When calculating the number of days since the startdate the application has to take care of weekends (weekends are not shown in the plan).

$$Days = NumberOfWeekdays(AssignmentDate - StartDate) \tag{6.3}$$

When the X property of the object is updated, the date of the assignment is updated and the same calculation is done in reverse.

CustIBaseViewModel

Holds the data of one IBase of a customer.

MessageViewModel

View model for the messages displayed in the lower area of the main window.

UserViewModel

Delivers the data for the list of logged in users.

ViewModelLocator

Normally the view model would be instantiated in XAML. The ViewModelLocator provides a way to access the MainViewModel globally. The second advantage is the possibility to pass a parameter to the constructor. This approach was also recommended in the use of MVVM light toolkit[6] .

[6] The video of the talk can be downloaded at http://ecn.channel9.msdn.com/o9/mix/10/wmv/EX14.wmv, last access January 28, 2011

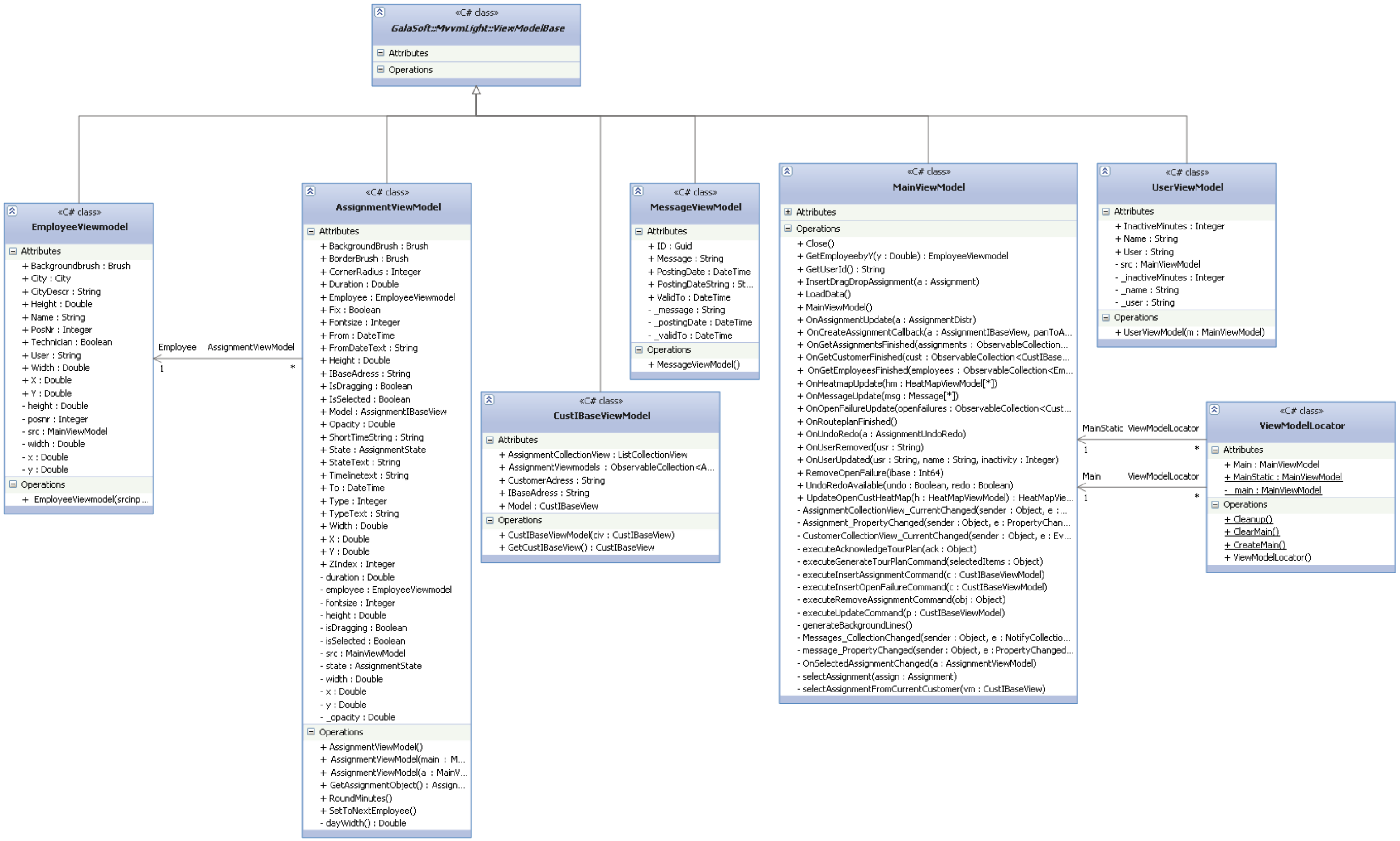

«C# class»
GalaSoft::MvvmLight::ViewModelBase
Attributes
Operations
«C# class»
EmployeeViewmodel
Attributes
+ Backgroundbrush : Brush
+ City : City
+ CityDescr : String
+ Height : Double
+ Name : String
+ PosNr : Integer
+ Technician : Boolean
+ User : String
+ Width : Double
+ X : Double
+ Y : Double
- height : Double
- posnr : Integer
- src : MainViewModel
- width : Double
- x : Double
- y : Double
Operations
+ EmployeeViewmodel(srcinp...
Employee
AssignmentViewModel
1
*
«C# class»
AssignmentViewModel
Attributes
+ BackgroundBrush : Brush
+ BorderBrush : Brush
+ CornerRadius : Integer
+ Duration : Double
+ Employee : EmployeeViewmodel
+ Fix : Boolean
+ Fontsize : Integer
+ From : DateTime
+ FromDateText : String
+ Height : Double
+ IBaseAdress : String
+ IsDragging : Boolean
+ IsSelected : Boolean
+ Model : AssignmentIBaseView
+ Opacity : Double
+ ShortTimeString : String
+ State : AssignmentState
+ StateText : String
+ Timelinetext : String
+ To : DateTime
+ Type : Integer
+ TypeText : String
+ Width : Double
+ X : Double
+ Y : Double
+ ZIndex : Integer
- duration : Double
- employee : EmployeeViewmodel
- fontsize : Integer
- height : Double
- isDragging : Boolean
- isSelected : Boolean
- src : MainViewModel
- state : AssignmentState
- width : Double
- x : Double
- y : Double
- _opacity : Double
Operations
+ AssignmentViewModel()
+ AssignmentViewModel(main : M...
+ AssignmentViewModel(a : MainV...
+ GetAssignmentObject() : Assign...
+ RoundMinutes()
+ SetToNextEmployee()
- dayWidth() : Double
«C# class»
CustIBaseViewModel
Attributes
+ AssignmentCollectionView : ListCollectionView
+ AssignmentViewmodels : ObservableCollection<A...
+ CustomerAdress : String
+ IBaseAdress : String
+ Model : CustIBaseView
Operations
+ CustIBaseViewModel(civ : CustIBaseView)
+ GetCustIBaseView() : CustIBaseView
«C# class»
MessageViewModel
Attributes
+ ID : Guid
+ Message : String
+ PostingDate : DateTime
+ PostingDateString : St...
+ ValidTo : DateTime
- _message : String
- _postingDate : DateTime
- _validTo : DateTime
Operations
+ MessageViewModel()
«C# class»
MainViewModel
Attributes
Operations
+ Close()
+ GetEmployeebyY(y : Double) : EmployeeViewmodel
+ GetUserId() : String
+ InsertDragDropAssignment(a : Assignment)
+ LoadData()
+ MainViewModel()
+ OnAssignmentUpdate(a : AssignmentDistr)
+ OnCreateAssignmentCallback(a : AssignmentIBaseView, panToA...
+ OnGetAssignmentsFinished(assignments : ObservableCollection...
+ OnGetCustomerFinished(cust : ObservableCollection<CustIBase...
+ OnGetEmployeesFinished(employees : ObservableCollection<Em...
+ OnHeatmapUpdate(hm : HeatMapViewModel[*])
+ OnMessageUpdate(msg : Message[*])
+ OnOpenFailureUpdate(openfailures : ObservableCollection<Cust...
+ OnRouteplanFinished()
+ OnUndoRedo(a : AssignmentUndoRedo)
+ OnUserRemoved(usr : String)
+ OnUserUpdated(usr : String, name : String, inactivity : Integer)
+ RemoveOpenFailure(ibase : Int64)
+ UndoRedoAvailable(undo : Boolean, redo : Boolean)
+ UpdateOpenCustHeatMap(h : HeatMapViewModel) : HeatMapVie...
- AssignmentCollectionView_CurrentChanged(sender : Object, e :...
- Assignment_PropertyChanged(sender : Object, e : PropertyChan...
- CustomerCollectionView_CurrentChanged(sender : Object, e : Ev...
- executeAcknowledgeTourPlan(ack : Object)
- executeGenerateTourPlanCommand(selectedItems : Object)
- executeInsertAssignmentCommand(c : CustIBaseViewModel)
- executeInsertOpenFailureCommand(c : CustIBaseViewModel)
- executeRemoveAssignmentCommand(obj : Object)
- executeUpdateCommand(p : CustIBaseViewModel)
- generateBackgroundLines()
- Messages_CollectionChanged(sender : Object, e : NotifyCollectio...
- message_PropertyChanged(sender : Object, e : PropertyChanged...
- OnSelectedAssignmentChanged(a : AssignmentViewModel)
- selectAssignment(assign : Assignment)
- selectAssignmentFromCurrentCustomer(vm : CustIBaseView)
«C# class»
UserViewModel
Attributes
+ InactiveMinutes : Integer
+ Name : String
+ User : String
- src : MainViewModel
- _inactiveMinutes : Integer
- _name : String
- _user : String
Operations
+ UserViewModel(m : MainViewModel)
MainStatic
ViewModelLocator
1
*
Main
ViewModelLocator
1
*
«C# class»
ViewModelLocator
Attributes
+ Main : MainViewModel
+ MainStatic : MainViewModel
- _main : MainViewModel
Operations
+ Cleanup()
+ ClearMain()
+ CreateMain()
+ ViewModelLocator()

Figure 6.6: ViewModel classes of A-Plan

6.5 Optimization

The optimization for automated generation of an assignment plan is implemented using the savings algorithm (section 3.3.2). The algorithm calculates the following steps:

- generate routeplan from existing assignments
- create pending tours (assign to employee with minimum distance to customer)
- calculate savings for the tours

The last step optimize the distances between the single tours. The processing of this part is illustrated in the following pseudo-code:

```
while(savingFound)
{
  when t1 is set
  {
    detect the earlier tour of t1 and t2
    when t1.Assignments.Count + t2.Assignment.Count <= MaximumRoutesPerDay
    {
      join t1 and t2 to earlier tour
      discard the later tour
    }
  }

  savingFound = false
  search for maximum saving in saving matrix
  when saving exists
  {
    savingFound = true;
    set t1 and t2 to the tours of the found saving
  }
}
```

6.6 Zoomable Canvas

When implementing the visualization of the plan, we encountered the problem of displaying many visual objects and permitting to pan and zoom smooth. The challenge is to render only the objects that are currently visible in the view port. Via internet search, we found a blog by Kael Rowan entitled Zoomable Canvas[7]. The Zoomable Canvas control has also the ability to make a semantic zoom to show different levels of detail at

[7] http://blogs.msdn.com/b/kaelr/archive/tags/zoomablecanvas, last access January 28, 2011

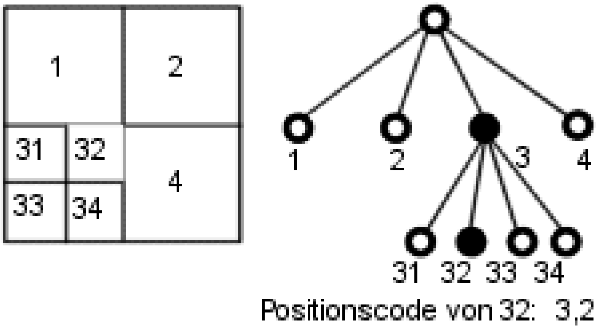


Figure 6.7: Principle of Quadtree [Geoinformatik Uni Rostock, 2001]

different levels of zoom, as in Code Canvas [Deline and Rowan, 2010]. The implementation of a semantic zoom in A-Plan would be an improvement for the visualization of the assignments, due the lack of time this feature could not be implemented.

The presented technique is the basis for the implementation of the visualization of the timeline in A-Plan. We used three Zoomable Canvas controls in our assignment view control to implement the visualization of the assignments. In XAML it is often not necessary to implement a completely new control, the better way is to customize existing controls. We implemented the visualization as ListBox, with a new template for the ItemsPanel of the ListBox. In the ItemsPanel of the ListBox the Zoomable Canvas is used with a binding to the Assignments collection of the MainViewmodel. We defined an new ItemContainerStyle to visualize the items of the collection, the position of the ListBoxItem is set using bindings to attributes of the elements of the collection. Using the MVVM pattern, all updates between the visualization (the ListBoxItem) and the corresponding view model (AssignmentViewModel) is done automatically.

The implementation of the Zoomable Canvas is based on a so called quadtree, a quadtree makes it possible to answer the question ‘which objects are in a given rectangle?’ efficiently. According to the website of the university of Rostock, department for geodesy and geoinformatics, [Geoinformatik Uni Rostock, 2001] a quadtree is a spatial index, which splits a spatial data set into homogeneous cells with uniformly decreasing size. The cells on each level are one quarter of the area of the previous cells. The process of subdividing continues until a sufficient partitioning is reached. By traversing the tree structure the data can be accessed efficiently.

CHAPTER 7

Evaluation

In order to assess the usability and utility of our design and implemented prototype, a qualitative evaluation with domain experts was conducted. The goal of this evaluation was less focused on validating the correct behavior of the prototype, but on investigating to what extent A-Plan meets the presented requirements.

7.1 Method

The evaluation was done by demonstration of the prototype, by task solving by users and by a semi-structured interviews. The guidelines for the evaluation can be found in the appendix C of this document. For the test a PC and a laptop have been used to demonstrate the possibilities of collaboration. The test subjects should solve some tasks with A-Plan, while doing this they should express their thoughts aloud. This method is called the 'think aloud method':

> The think aloud method consists of asking people to think aloud while solving a problem and analysing the resulting verbal protocols.[Someren et al., 1994]

The tasks that had to be solved by the test subjects and the details of the tasks can be seen in appendix B:

- searching for information about customers
- move, edit and delete assignments
- insert open failures
- insert assignments using the automated planing function

7.1.1 Materials, Procedure & Analysis Approach

During this process we made audio recordings and the activities of the system were recorded by a screen recorder software. In addition, written notes have been made about the activities of the testers. Through this multiple logging approach we wanted to avoid that interesting aspects will not be included in the analysis of the evaluation.

After the tasks a semi-structured interview was conducted. The questions were general questions about the application, questions about the visualization of the plan, about the detail area (assignments and open failures) and about the long term planning functionality of A-Plan. The questions are listed in appendix B.

The results of the test were collected and analyzed in tabular form. To categorize the results the proposed categories by Forsell & Johansson [Forsell and Johansson, 2010] were used. By doing this most important points for improvement have been identified. The categories of Forsell & Johansson are listed in table 7.1. The proposed heuristics of Forsell & Johansson are a new set of general heurstics for common and important usability problems that provides the widest coverage of the given problems.

We carried out the prototype evaluation with five persons: two customer care agents, one service technician and two IT experts:

- Student of informatics, female, 29 years old
- IT expert, male, 45 years old, skills in programming, system administration and SAP CRM knowledge
- customer care agent, female, 33 years old, works since two years in this field of work
- customer care agent, female, 26 years old, works since four year in this field of work
- customer service technician, male, 36 years old, works as technician for 15 years

The main focus was on the potential users of the system (the customer care agents). In section 4.3 we described two abstract personas, now in the evaluation the two customer care agents assume the role of Julia Steiner and the service technician assume the role of Erich Gruber. The reason to test the prototype with a service technician is, that in times with big crowds a service technician has to help in the back office and so we wanted to test this situation. With the two IT experts we wanted to get more critical feedback about the usability of the prototype.

In our evaluation we evaluated the scenarios from section 4.4. The users had to solve tasks that are described in more detail in appendix B.

7.2 Results

In general the feedback was very positive. When analyzing the the found problems and issues based on the heuristics of Forsell & Johansson [Forsell and Johansson, 2010], most

B3	**Spatial Organization.** Concerns user's orientation in the information space, the distribution of elements in the layout, precision and legibility, efficiency in space usage and distortion of visual elements
B5	**Information Coding.** Perception of information is directly dependent on the mapping of data elements to visual objects. This should be enhanced by using realistic characteristics/techniques or the use of additional symbols.
E7	**Minimal actions.** Concerns workload with respect to the number of actions necessary to accomplish a goal or a task.
E11	**Flexibility.** Flexibility is reflected in the number of possible ways of achieving a given goal. It refers to the means available to customization in order to take into account working strategies, habits and task requirements.
B7	**Orientation and help.** Functions like support to control levels of details, redo/undo of actions and representing additional information.
B3	**Spatial organization.** Concerns users orientation in the information space, the distribution of elements in the layout, precision and legibility, efficiency in space usage and distortion of visual elements.
E16	**Consistency.** Refers to the way design choices are maintained in similar contexts, and are different when applied to different contexts.
C6	**Recognition rather than recall.** The user should not have to memorize a lot of information to carry out tasks.
E1	**Prompting.** Refers to all means that help to know all alternatives when several actions are possible depending on the contexts.
D10	**Remove the extraneous.** Concerns whether any extra information can be a distraction and take the eye away from the seeing the data or making comparisons.
B9	**Data set reduction.** Concerns provided features for reducing a data set, their efficiency and ease of use.

Table 7.1: Categories proposed by Forsell & Johansson [Forsell and Johansson, 2010]

B5	**Information Coding.**	10
B7	**Orientation and help.**	10
E7	**Minimal actions.**	8
B3	**Spatial organization.**	3
B3	**Spatial Organization.**	3
E16	**Consistency.**	2
E11	**Flexibility.**	1

Table 7.2: Occurences per Category

of them were of the categories B7 Orientation and help and B5 Information Coding (both 10 times), followed by E7 Minimal actions (8 times) — the other occrences can be seen in table 7.2. In the following we will provide more details on the gathered data.

Three persons (the customer care agents and the service technician) gave very positive feedback about the speed in which the required tasks can be solved with A-Plan. For example the scheduling of a customer for maintenance in the current system can be estimated with four minutes effort, in A-Plan this can be done within a few seconds. The search function received much praise also, as the way to search in A-Plan is familiar to all users from search engines in the internet, this is a very easy and fast way to find the desired data. The possibility to plan customers automatically to service technicians is very interesting for the subjects, as the planning is currently completely done manually. In the actual system this is a very time consuming operation.

The screen layout was clear to all subjects. For three subjects the screen layout was a little overloaded in some areas, but they also mentioned that they have no idea how to make it better. For one customer care agent the simple way to modify the data was unfamiliar, she would prefer an additional confirmation for some operations. This is also the way how to work in the current system. One tester would prefer to use a save button instead of the automatic save with undo/redo capability.

The program crashed in some situations, which was not a problem for the testers, as we educated them about the early development and test state of the system. For a real world application bugs should be corrected.

An important thing, which is missing, is the activity history of assignments. For example if an agent deletes an assignment no one can comprehend what has been done, because the data doesn't exist anymore. This functionality has been planned in the conceptual design of the application but was not implemented due to lack of time.

7.2.1 Planning Area

The visualization of the plan was clear for all testers. The functionality to show details by tooltips has not been recognized by the users. Only after we give them a hint, they used the function and found it useful. The way to move assignments by drag-and-drop was no problem for the testers. The way to get details of assignments earned positive

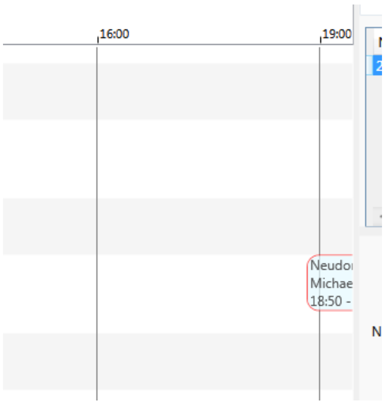


Figure 7.1: Movement of an assignment outside of the current view

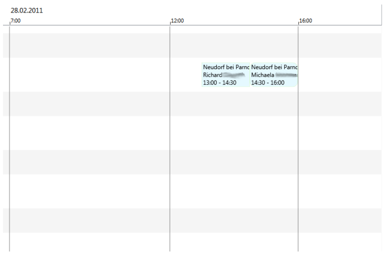


Figure 7.2: Distinction between fixed assignments and not fixed assignments

feedback too, the testers who work with the current system mentioned that the system is very fast.

Some problems occurred at several or even all testers. We want to discuss these problems that happened often in more detail:

Movement of an Assignment Outside of the Current View

The most common problem for the users during the prototype evaluation occurred on the shifting of assignments. The task was to move an assignment to the next day, all users tried to do this by dragging the assignment to the next day. The problem happened when they reached the border of the plan, they could not move the assignment anymore (figure 7.1).

Recognition of Fixed Dates

For the test persons the distinction between fixed and not fixed assignments was not strong enough (cmp. figure 7.2), they would prefer to use different colors or a border around the fixed assignments.

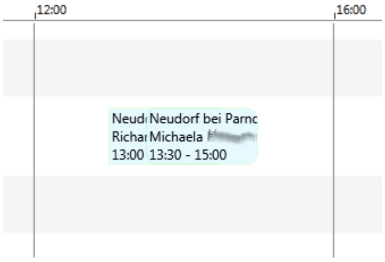


Figure 7.3: Overlapping assignments

Overlapping Assignments

In some test cases, the tester dropped an assignment over an existing assignment (figure 7.3). We didn't implement a reaction of the program (for example, a possible reaction could be that this operation is not allowed), so the second assignment is on the top of the other and the lower one could not be seen anymore.

Direct Manipulation

For three testers the way to zoom and pan was very unfamiliar, they would prefer a fixed view where no free panning and zooming is possible. One test person mentioned that the movement of an assignment by drag and drop can happen unintentionally, she would prefer to move assignments with the right mouse button.

7.2.2 Detail Area

In general there was not much criticism about the detail area, some mentioned this area is a little bit overloaded, they also mentioned that if they would work every day with the application they would get used to it.

A problem occurred two times at the insertion of an open failure: two testers clicked more than once on the button to insert an new open failure, so the application should give more feedback. It would be also possible to prevent this by disabling the button when an open failure already exists. It doesn't make sense to have more than one open failure for one customers site.

7.2.3 Planning

The planning functionality earned very positive feedback, the calculated plan was acceptable, the handling of the planning area is easy and clear. One tester had problems to understand the heat map, she would prefer a different representation instead a calendar.

7.3 Summary

Conducting the evaluation was a very interesting part of this project, testing the application with people who do their daily work in this field brought interesting inputs. Some issues arose from our opinion that the users who did the evaluation are not familiar with techniques like direct manipulation. In the current system every action has to be acknowledged, direct manipulation without the need of a save button is very unfamiliar.

The general feedback of study participants was very positive and a number of shortcomings and future improvements could be identified.

CHAPTER 8

Conclusion

In this work we implemented an application for scheduling technicians. In the beginning we investigated the field of work in such a company. It became apparent that the standard software used does not support the needed tasks well. For customer care employees, standard work routines are very cumbersome to carry out. This leads to more effort needed for processing the cases and causes longer waiting times for the customers of the company. Interestingly, paper-based workarounds have been developed by the employees of the investigated company to mitigate these problems. We were somehow astonished to encounter this quite sophisticated system of paper-based artifacts to reach a more or less smooth working environment. This provided us with very valuable insights for our developments.

Therefore we based in the development of a new system on human-centered design [Stone et al., 2005], we conducted interviews where the needs and goals of potential users were obtained. Based on the results of these interviews, certain types of individuals were derived. This approach corresponds to the concept of Personas, Goals and Scenarios by Alan Cooper [Cooper, 2004]. These Personas and Scenarios helped us to design the application.

In this thesis we have shown that the integration of potential users in the design of an application is an important factor. Functionality is not the most important factor, however it is mandatory that designers take the goals of the application, as well as the need of the potential users into consideration. The result is an application that has the needed functionality in one interface.

We also studied the related literature in the areas user interface and interaction, collaboration and optimization. We discussed some techniques to visualize time-dependent data and searched for ways to interact with this data. As our system is used by more than one person at a time, collaboration is also an important aspect. We showed how an application has to be designed when more people should cooperate in the application. For the optimization task we discussed discrete optimization problems in general and showed two heuristics to solve the task.

Based on this information we designed a mockup prototype and discussed this with an expert. Afterwards we implemented the prototype in C# with new techniques WPF and WCF. We implemented the application using the MVVM pattern, which allows a clean separation between interface and logic.

Collaboration between users is necessary in corporations, where multiple users work with the same data. We showed an implementation, where the data is distributed in real time to all clients. Aspects of awareness have also been taken into account. In most of the investigated applications, support of collaboration was not available.

A time-saving functionality is the automated creation of a plan. To find the 'optimal' result is not as important to create a good fast result. We also showed in A-Plan how to integrate the automated planing function into the application. The plan constitutes a proposal for the user. The user may review the suggested assignment and accept or deny the proposed plan. Using this human-in-the-loop approach, automatic planning and human judgement are closely coupled using interactive visual interfaces.

We evaluated the implemented prototype with three users of the current system and two IT experts. Testing the prototype with people who do their daily work in this field yielded much interesting feedback. Some issues arose from the fact that the users who did the evaluation were not familiar with techniques like direct manipulation. In the current system every action has to be acknowledged explicitly and direct manipulation without the need of a save button is therefore unfamiliar. Overall, we received encouraging feedback and were able to identify shortcomings of the design and functionality of A-Plan. All evaluation participants would prefer to use A-Plan instead of the existing system.

The application A-Plan brings big improvements in visualization, interaction, collaboration and automated planning compared to software systems in use. The use of A-Plan would simplify the work in the illustrated planning scenario. The main contribution of our work is that we have shown an effective integration of automatic methods and interactive visualizations based on a user-centric development approach. The integration of the strength of both the human and the computer enables the creation a powerful environment for a set of non-trivial and complex tasks.

CHAPTER 9

Future Work

As A-Plan is currently in the prototype stage, a number of directions for future work remain. Firstly, there are a number of software bugs that need to be fixed in order to improve the stability of the tool. Furthermore, the issues that surfaced in our evaluation (chapter 7.2) should be addressed:

- The movement of an assignment outside of the current view should be possible. The application should pan automatic, so that the assignment is in the view again.
- Fixed and not fixed assignments should be presented in different ways
- Overlapping assignments should not be possible — the application should insert the assignments after each other
- Improvements in direct manipulation: for some users the movement of an assignment has been done unintentionally, an improvement would be an additional process step.

An interesting improvement for the visualization of the plan could be a semantic zoom. In section 6.6 the Zoomable Canvas by Kael Rowan was introduced. The Zoomable Canvas control has also the ability to make a semantic zoom to show different levels of detail at different levels of zoom, as in Code Canvas [Deline and Rowan, 2010]. This functionality could be implemented in A-Plan too. For example when the user zooms out to a very high level the text on the assignments can't be read anymore — so the text could simply be omitted instead when it gets too small.

An important thing, which is missing, is the activity history of assignments. During the evaluation (section 7.2) some testers commented, that it would be very useful to have a history of all activities of an assignment. It was also planned to implement this feature, but it was not possible due the lack of time.

Future work can also be done in optimization algorithm of A-Plan. A simple savings algorithm implementation was used for the application. For testing purposes this algorithm delivered good results. A possibility for better results would be to use an improvement algorithm on the result of the savings algorithm.

If A-Plan should be used as planning instrument in the investigated company, the application has to be integrated in the currently used IT environment. All customer information (like the IBase, devices and so on) are currently maintained in the ERP system, so all data between A-Plan and the ERP system have to be replicated. This leads to many other issues in operation. Therefore, the integration of A-Plan into an existing system environment would need as much effort as the implementation of A-Plan itself.

APPENDIX A

Interview Guideline

A.1 German Version

A.1.1 Fragen

Person

- Welche Ausbildung haben Sie?
- Welche Aufgaben übernehmen Sie im Unternehmen?
- Welche Computerkenntnisse haben Sie?

Visualisierung

Möglichkeiten zur Visualisierung und Interaktion von Daten werden vorgestellt:

- Time Lines
- Life Lines
- GANTT
- Time Mosaics
- Overview and Detail
- Zooming
- Focus and Context

Welche Techniken kennen Sie bzw. welche erachten Sie als geeignet für die gestellte Aufgabe?

Bestehendes System

- Welche Tätigkeiten sind am bestehenden System umständlich durchzuführen?
- In welchen Situationen ist das Arbeiten besonders schwierig?
- Aufgrund welcher Informationen treffen Sie Entscheidungen?
- Welche Funktionen benutzen Sie im System?
- Welche Funktion wird davon am Häufigsten eingesetzt?
- Welche Aufgabe könnte am Besten im bestehenden System verbessert werden?
- Welche Tätigkeiten machen Sie als Erstes in der Früh?
- Welche Tätigkeit ist an Ihrer Arbeit besonders unangenehm?

A.2 English Version

A.2.1 Questions

Person

- What education do you have?
- What are your responsibilities in the company?
- What computer knowledge do you have?

Visualization

Possibilities for visualization and interaction are presented:

- Time Lines
- Life Lines
- GANTT
- Time Mosaics
- Overview and Detail
- Zooming
- Focus and Context

What techniques do you know or what do you consider to be suitable for the task?

Existing System

- Which tasks are complicate on the existing system?
- In which situations is working particularly hard?
- On what information do you make decisions?
- What features do you use in the system?
- What is the function which is used most frequently?
- What task could be improved best in the existing system?
- What tasks do you do first in the morning?
- What activity is particularly uncomfortable in your work?

APPENDIX B

Evaluation Guideline

B.1 German Version

B.1.1 Aufgabe

Das Ziel dieser Arbeit ist, eine Applikation für die Planung von Serviceeinsätzen von Technikern zu entwickeln. Fokus ist dabei die Interaktion, die Visualisierung, die Kooperation der Planer sowie die Möglichkeit automatisiert Routen zu berechnen.

B.1.2 Methode

Es wurde die Arbeitsweise mit einer aktuell im Einsatz befindlichen Standardsoftware (SAP CRM) analysiert. An der bestehenden Vorgangsweise wurden Schwachstellen aufgezeigt. Es wurden verschiedene aktuelle Möglichkeiten zur Visualisierung und Interaktion untersucht. Der neu zu entwickelnde Prototyp soll die Zusammenarbeit mehrer Benutzer ermöglichen. Für die automatisierte Planung konnte eine geeignete Heuristik gefunden werden.

Mit diesen Informationen wurde ein Prototyp entwickelt, der den Anforderungen gerecht werden soll. Bei der Benutzung des Prototypen kann es zu unbeabsichtigen Abstürzen und Fehlverhalten kommen, da der Prototyp noch nicht ausreichend getestet wurde. Ziel dieser Evaluierung ist jedoch nicht die Überprüfung des Prototypen auf korrektes Verhalten, sondern inwieweit das vorgestellte Konzept für die gestellten Anforderung als geeignet erscheint.

Im ersten Schritt wird der aktuelle Stand des Prototypen demonstriert. Daraufhin haben die Testpersonen mit dem Prototypen bestimmte Aufgaben zu lösen, danach folgt ein semistrukturiertes Interview über die Bedienung des Prototypen. Die Person wird daraufhin hingewiesen, dass nicht sie getestet wird, sondern der Prototyp. Außerdem wird nicht primär die Funktionalität, sondern die Bedienbarkeit des Prototypen getestet. Der Tester darf die Evaluierung jederzeit abbrechen.

B.1.3 Aufgabenstellungen

Folgende Aufgabenstellungen sind durchzuführen, der Benutzer soll dabei laut seine Gedanken sprechen:

- Suchen Sie Kunden ‘Bruno Aichinger’ aus ‘Eisenstadt’, welche Informationen können Sie ablesen
- Verschieben Sie den Termin um einen Tag und stellen Sie den Termin als ‘fixiert’ ein
- Löschen Sie den Termin von Marcel Klemenschits
- Der Kunde ‘Kurt Schneider, Waldeg 4, Eisenstadt’ meldet eine Gerätestörung, erfassen Sie diese
- Planen Sie diese offene Gerätestörung ein
- Erstellen Sie einen Plan mit ca. 40 Kunden ab 10.2.2011, sehen Sie sich die eingeplanten Einsätze an
- Planen Sie die Frank Uchner (Anlage Kulmbachstr.) für die Wartung ein

B.1.4 Leitfaden für Fragen

Generelle Fragen zur Applikation

- Was wurde Ihrer Meinung nach gut umgesetzt?
- Was wurde Ihrer Meinung nach nicht gut umgesetzt? Warum?
- Welche Aufgaben können mit dem Prototypen Ihrer Meinung nach nur schwer durchgeführt werden?
- Wie finden Sie den Bildschirmaufbau?
- Was kann verbessert werden?

Planungsbereich

- Wie beurteilen Sie die Darstellung im Planungsbereich?
- Wie beurteilen Sie die Interaktiosnmöglichkeiten im Planungsbereich?
- Wie nützlich finden Sie die Detailinformationen mittels Tooltips?
- Sind die Möglichkeiten in der Zeit zu navigieren verständlich?
- Ist die farbliche Darstellung der Termine in Ordnung?

Detailbereich und Suchfunktionalität

- Wie beurteilen Sie die Darstellung im Detailbereich?
- Wie beurteilen Sie die Suchfunktion?
- Wie finden Sie die Möglichkeiten Wartungstermine einzuteilen (Drag´n´Drop bzw. mittels Vorschlag)?
- Wie beurteilen Sie die Möglichkeit Einsätze mittels Eingabe einer Uhrzeit zu editieren?

Planung

- Wie beurteilen Sie die Darstellung der offenen Wartungen (heat map)?
- War die Bedienung verständlich?

Offene Gerätestörungen

- Ist die Aufteilung in Wartungen und Gerätestörungen Ihrer Meinung nach sinnvoll?

Zusammenarbeit

- Wie beurteilen Sie die Möglichkeit zur Zusammenarbeit?
- Wie beurteilen Sie die Möglichkeit Nachrichten an die Kollegen zu verteilen?

Abschluss

- Können Sie sich vorstellen, die Applikation für eine derartige Aufgabenstellung einzusetzen?
- Optional: Wie beurteilen Sie A-Plan im Vergleich zur bestehenden Applikation?

B.2 English Version

B.2.1 Project

The goal of this work was to implement an application for planning of service assignments of technicians. Focus was the interaction, visualization, the collaboration of the dispatcher as wells as the possibility for the automated calculation of routes.

B.2.2 Method

The currently standard software in use (SAP CRM) has been analyzed. The existing problems of the current solution have been identified. Possibilities for visualization and interaction has been examined. The prototype should allow the cooperation of several users. For automated planning we found a suitable heuristic.

With this information we developed the prototype A-Plan, this prototype should fulfill the requirements. At the operation of the prototype the application may crash or misconduct. The reason is the early development status of the prototype. Goal of this evaluation is not to test the correct behavior of the prototype, but how far is A-Plan suitable for the presented requirements.

In the first step, the current state of the prototype will be demonstrated. Subsequently, the evaluators have to solve certain tasks with the prototype, followed by a semi-structured interview about the prototype. The person is pointed out that not she will be tested, but the prototype. The tester may stop the evaluation at any time.

B.2.3 Tasks

The following tasks have to be done with the prototype, the user should speek about his experience:

- Search for the customer ‘Bruno Aichi’ from ‘Eisenstadt’. What information do you see?
- Move this assignment to the next day and set the assignment as fixed.
- Delete the assignment from Marcel Klemenschits
- The customer ‘Kurt Schn..., Eisenstadt’ reports a device failure, record this case
- Plan the device failure from the previous point
- Create a plan with 40 customer starting at ..., take a look on the planned assignments.
- Plan the maintenance of Frank ... (IBase Kulmbachstr.)

B.2.4 Guide for questions

General questions about the application

- What has been implemented well in your opinion?
- What has been implemented not good in your opinion? Why?
- What tasks can be solved difficult with the prototype?
- How do you like the screen layout?
- What can be improved?

Planning area

- How do you rate the visualization in the planning area?
- How do you rate the possibilities for interaction in the planning area?
- What do you think about showing detail informations with tooltips?
- Are the possibilities to navigate in time comprehensible?
- How do you rate the use of colors?

Detail area and search functionality

- How do you rate representation in the detail area?
- How do you rate the search functionality?
- How do you like the possibilities to insert assignments(Drag´n´Drop and suggestion)?
- How do you rate the possibility to edit date and time of assignments directly?

Planning

- How do you rate the representation of open maintenance(heat map)?
- Is the operation comprehensible?

Open failures

- Does the separation between maintenance and device failures make sense?

Collaboration

- How do you rate the possibilities for collaboration?
- How do you rate the functionality to send messages?

Completion

- Can you imagine to use the application for the requested tasks?
- Optional: How do you rate A-Plan compared to the existing system?

APPENDIX C

Prototype evaluation results

C.1 Test 1

The first test person (female) was an 29 year old student of informatics (master). She has no experience with planing software and she also does not work in the investigated company.

C.1.1 Summary of Tasks

Nr.	Time	Description	Type	Category
1	01:10	The current date can't be seen - the window is too small and the date label is outside of the view	deficit	B7
2	03:30	Selection of date: should be 'today' as default	deficit	B7
3	04:00	Movement of an assignment with drag-and-drop to the next day difficult	problem	E7
4	05:00	Bug zoomslider	problem	
5	05:00	100% zoom button is missing	deficit	E7
6	06:15	selected assignment without attracting attention	deficit	B5
7	07:20	Can't find button for insertion of new open failure	deficit	B3, B5
8	09:10	Bug zoom	problem	
9	11:10	Representation as heat map not understandable	deficit	B5
10	12:30	Person marks all rows for planning	deficit	E16
11	13:40	The empty list bemused	deficit	B5

C.1.2 Summary of the Interview

- Screen layout good, a 100% button is missing for zooming

- The tester misses a save button, in her opinion is accept/decline not necessary
- Visualization of the heat map is not clear, better would be a representation as calendar
- Selected assignments should be more remarkable
- Tooltips should not be transparent
- Insertion of an assignment with double click would be better

C.2 Test 2

The second person (male) was a 36 years old customer service technician. He works in the investigated company for 15 years.

C.2.1 Summary of Tasks

Nr.	Time	Description	Type	Category
12	00:00	The application should not pan to very old assignments	deficit	E7
13	01:50	Informations about devices are missing	problem	B7
14	03:00	Google Maps: problem at scroll/zoom	deficit	B7
15	05:50	Bug zoomslider	problem	
16	05:00	Movement of an assignment with drag-and-drop to the next day difficult	problem	E7
17	06:50	The representation of fixed assignments should be different (color, border)	deficit	B5
18	08:05	Deleted assignments are not visible anymore	problem	B5
19	11:30	Intent of info field not clear	deficit	B5
20	12:25	Info field to small	deficit	B5
21	16:45	Can't find address of IBase	deficit	B7

C.2.2 Summary of the Interview

- The tester likes automated planning, the speed, the map, the screen layout and the way information is viewed
- When searching customer with old assignments (older than one month), no panning should be done
- Some device information is missing
- The representation of fixed assignments should be different (color, border)
- A completely deletion of an assignment should not be possible, a cancellation where the assignment still exists would be better

C.3 Test 3

The third test person (female) was an 26 year old employee from customer care. She works in the investigated company for four years.

C.3.1 Summary of Tasks

Nr.	Time	Description	Type	Category
22	01:00	Search with post code leads to program crash	problem	
23	03:30	Movement of an assignment with drag-and-drop to the next day difficult	problem	E7
24	03:50	Tester move assignment out of the view	problem	B7
25	03:50	Assignment is behind another assignment	problem	B3
26	05:55	Program crash	problem	
27	07:30	Can't find Button for the insertion of an open device failure	deficit	B3, B5
28	07:50	Button device failure is pressed several times	deficit	B7
29	09:25	Insert: drag-and-drop does not work	problem	
30	09:30	Program crash	problem	
31	10:10	Program crash	problem	
32	12:20	Selection of single IBase not possible	deficit	E11
33	13:20	Program crash	problem	
34	15:00	Program crash	problem	
35	16:45	Different premises not recognized	deficit	B7
36	17:25	Insert: drag-and-drop does not work	problem	

C.3.2 Summary of the Interview

- When doing planning there should be a possibility to deselect single customers
- History of activities on assignments is missing
- The name of the customer should be bold at the assignments in the plan
- Maintenance contracts should be viewed more explicit

C.4 Test 4

The third test person (female) was also an employee from customer care. She is 33 years old, works since two years in this field of work.

C.4.1 Summary of Tasks

Nr.	Time	Description	Type	Category
37	02:20	Movement of an assignment with drag-and-drop to the next day difficult	problem	E7
38	03:05	Function for fixed assignments not found	deficit	B7
39	03:50	Before the search is operated the previous customer is still on (Clear button)	deficit	E7
40	04:45	For test is irritating that the result is automatically selected		
41	05:20	Button device failure is pressed several times	deficit	B7
42	06:20	Program crash	problem	
43	09:00	Date selection: Why is '15' as an icon when 21st is selected	deficit	E16
44	10:15	Program crash	problem	
45	12:05	Program crash	problem	

C.4.2 Summary of the Interview

- Unfamiliar, very different to the existing software
- Search function is very good
- Date should be more highlighted, the days should be better differentiated
- Instead of free zooming/panning would be a fixed view better
- The usage of the navigation symbol is not good
- Movement of assignments with drag and drop is not good, better would be to use the right mouse key
- The representation of fixed assignments should be different (color, border)
- A border around the customer data would make the detail area more clearer
- When searching for customers the open device failures should be viewed

C.5 Test 5

The last test person was male and an IT expert from the company we investigated. He works in the investigated company for 20 years. Interesting is, that this person implemented an individual software for the company. This implemented solution has been replaced by the system that is still in use by the company.

C.5.1 Summary of Tasks

Nr.	Time	Description	Type	Category
47	02:15	Movement of an assignment with drag-and-drop to the next day difficult	problem	E7
48	02:30	Move the plan using the navigation symbol: does not react	problem	
49	03:15	Assignment is behind another assignment	problem	B3
50	04:30	The representation of ‘fixed’ assignments is hard to recognize	deficit	B5
51	06:05	The procedure at device failures is not clear		
52	07:20	Can’t find address of IBase	deficit	B7

C.5.2 Summary of the Interview

- General very positive
- Device info is missing in the plan area
- In the tool tips should be different information than in the plan
- Instead of free zooming/panning would be a fixed view better
- The representation of fixed assignments should be different (color, border)
- Detail area is a little overloaded
- Heat map should use more colors
- History of activities on assignments is missing
- Support for collaboration is excellent

Bibliography

[Bartlett and Toms, 2005] Bartlett, J. C. and Toms, E. G. (2005). Developing a Protocol for Bioinformatics Analysis: An Integrated Information Behavior and Task Analysis Approach. *Journal of the American Society for Information Science and Technology*, 56(5):469–482.

[Bederson et al., 2004] Bederson, B. B., Clamage, A., Czerwinski, M. P., and Robertson, G. G. (2004). DateLens. *Interactions*, 11(4):9.

[Bektas, 2006] Bektas, T. (2006). The Multiple Traveling Salesman Problem: An Overview of Formulations and Solution Procedures. *Omega*, 34(3):209–219.

[Bräysy, 2004] Bräysy, O. (2004). Evolutionary Algorithms for the Vehicle Routing Problem with Time Windows. *Journal of Heuristics*, 10(6):587–611.

[Cheung et al., 2007] Cheung, R. K., Xu, D., and Guan, Y. (2007). A Solution Method for a Two-dispatch Delivery Problem with Stochastic Customers. *Journal of Mathematical Modelling and Algorithms*, 6(1):87–107.

[Clarke and Wright, 1964] Clarke, G. and Wright, J. V. (1964). Scheduling of Vehicles from a Central Depot to a Number of Delivery Points. *Operations Research*, 12(4):568–581.

[Cockburn et al., 2008] Cockburn, A., Karlson, A., and Bederson, B. B. (2008). A Review of Overview+Detail, Zooming, and Focus+Context Interfaces. *ACM Computing Surveys*, 41(1):1–31.

[Cooper, 2004] Cooper, A. (2004). *The Inmates Are Running The Asylum: Why High Tech Products Drive Us Crazy and How To Restore The Sanity.* Sams Publishing, Indianapolis, Ind.

[Cooper et al., 2007] Cooper, A., Reimann, R., and Cronin, D. (2007). *About Face: The Essential of Interaction Design.* Wiley Publishinc, Inc., Indianapolis, Ind., 3 edition.

[Dachselt and Weiland, 2006] Dachselt, R. and Weiland, M. (2006). TimeZoom. *CHI '06 extended abstracts on Human factors in computing systems - CHI '06*, page 682.

[De Alwis et al., 2009] De Alwis, B., Gutwin, C., and Greenberg, S. (2009). GT/SD: Performance and Simplicity in a Groupware Toolkit. In *Proceedings of the 1st ACM SIGCHI symposium on Engineering interactive computing systems*, pages 265–274, Pittsburgh, PA, USA. ACM.

[Deline and Rowan, 2010] Deline, R. and Rowan, K. (2010). Code Canvas : Zooming towards Better Development Environments. *Design*, pages 207–210.

[Domschke, 1990] Domschke, W. (1990). *Logistik, Rundreisen und Touren*. Oldenbourg Verlag GmbH, München, Wien, 3 edition.

[Dourish and Bellotti, 1992] Dourish, P. and Bellotti, V. (1992). Awareness and Coordination in Shared Workspaces. *Proceedings of the 1992 ACM conference on Computer-supported cooperative work - CSCW '92*, pages 107–114.

[Forsell and Johansson, 2010] Forsell, C. and Johansson, J. (2010). An Heuristic Set for Evaluation in Information Visualization. *Information Visualization*, pages 1–8.

[Gantt, 1919] Gantt, H. L. (1919). *Organizing for Work*. Harcourt, Brace and How, New York.

[Geoinformatik Uni Rostock, 2001] Geoinformatik Uni Rostock (2001). Geoinformatik-Service Lexikon Quadtree. http://www.geoinformatik.uni-rostock.de/einzel.asp?ID=1410, accessed on february 16, 2011.

[Gillett and Miller, 1974] Gillett, B. and Miller, L. (1974). A Heuristic Algorithm for the Vehicle-Dispatch problem. *Operations Research*, 22(2):340–349.

[Gomes et al., 2005] Gomes, R. L., Rivera, G. J. H., Courtiat, J. P., and Roche, C. (2005). Loosely-Coupled Integration of CSCW Systems. *Ifip International Federation For Information Processing*, pages 38 – 49.

[Hanshar, 2007] Hanshar, F. (2007). Dynamic Vehicle Routing Using Genetic Algorithms. *Applied Intelligence*, 27(1):89–99.

[Karam, 1994] Karam, G. M. (1994). Visualization Using Timelines. *Proceedings of the 1994 international symposium on Software testing and analysis - ISSTA '94*, pages 125–137.

[Kuhrmann and Beneken, 2006] Kuhrmann, M. and Beneken, G. (2006). *Windows Communication Framework*. Elsevier GmbH, Munich.

[Kulyk et al., 2007] Kulyk, O., Kosara, R., Urquiza, J., and Wassink, I. (2007). Human-Centered Aspects. In *Human-Centered Visualization Environments*, pages 13–75. Springer.

[Littger, 1992] Littger, K. (1992). *Optimierung*. Springer, Berlin.

[Luz and Masoodian, 2007] Luz, S. and Masoodian, M. (2007). Visualisation of Parallel Data Streams with Temporal Mosaics. *2007 11th International Conference Information Visualization (IV '07)*, pages 197–202.

[Mackinlay et al., 1991] Mackinlay, J., Robertson, G., and Card, S. (1991). The Perspective Wall: Detail and Context Smoothly Integrated. *the SIGCHI conference on*, pages 173–179.

[Norman, 1993] Norman, D. A. (1993). *Things That Make Us Smart. Defending Human Attributes in the Age of the Machine.* Addison-Wesley Pub. Co., Boston, MA, USA.

[Patterson, 1995] Patterson, J. F. (1995). A Taxonomy of Architectures for Synchronous Groupware Applications. *ACM SIGOIS Bulletin*, 15(3):27–29.

[Plaisant et al., 1996] Plaisant, C., Milash, B., Rose, A., Widoff, S., and Shneiderman, B. (1996). Lifelines: Visualizing personal histories. pages 221–227. ACM Press.

[Prinz, 1999] Prinz, W. (1999). Nessie: An awareness environment for cooperative settings. In *Proceedings of the sixth conference on European Conference on Computer Supported Cooperative Work*, ECSCW'99, pages 391–410, Norwell, MA, USA. Kluwer Academic Publishers.

[Roseman and Greenberg, 1992] Roseman, M. and Greenberg, S. (1992). Groupkit: A groupware toolkit for building real-time conferencing applications. In *Proceedings of the 1992 ACM conference on Computer-supported cooperative work*, CSCW '92, pages 43–50, New York, NY, USA. ACM.

[Schmidt, 2002] Schmidt, K. (2002). The problem with 'awareness': Introductory remarks on 'awareness in cscw'. *Comput. Supported Coop. Work*, 11:285–298.

[Shneiderman, 1996] Shneiderman, B. (1996). The eyes have it: a task by data type taxonomy for information visualizations. *Proceedings 1996 IEEE Symposium on Visual Languages*, pages 336–343.

[Shneiderman, 2005] Shneiderman, B. (2005). *Designing the user interface.* Pearson Addison Wesley, Boston, MA, USA.

[Someren et al., 1994] Someren, M., Barnard, Y., and Sandberg, J. (1994). The Think Aloud Method: A practical guide to modelling cognitive processes. *Information Processing & Management*, 31(6):906–907.

[Stone et al., 2005] Stone, D., Jarett, C., Woodroffe, M., and Minocha, S. (2005). *User Interface Design and Evaluation.* Morgan Kaufmann/Elsevier, Amsterdam.

[Suntinger et al., 2008] Suntinger, M., Schiefer, J., Obweger, H., and Groller, M. E. (2008). The Event Tunnel: Interactive Visualization of Complex Event Streams for Business Process Pattern Analysis. *2008 IEEE Pacific Visualization Symposium*, pages 111–118.

[Tanenbaum and Steen, 2003] Tanenbaum, A. and Steen, M. (2003). *Verteilte Systeme*. Pearson Studium, München.

[Thomas and Cook, 2005] Thomas, J. J. and Cook, K. A. (2005). *Illuminating the Path: The Research and Development Agenda for Visual Analytics*. IEEE, Los Alametos, CA.

[Tidwell, 2005] Tidwell, J. (2005). *Designing Interfaces*. O'Reilly, Beijing [u.a.].

[Toth and Vigo, 2002] Toth, P. and Vigo, D. (2002). *The Vehicle Routing Problem*. SIAM Monographs on Discrete Mathematics and Applications, Philadelphia, PA, USA.

[Tufte, 1983] Tufte, E. (1983). *The Visual Display of Quantitative Information*. Graphics Press, Cheshire, Connecticut.

[Tversky et al., 2002] Tversky, B., Morrison, J., and Betrancourt, M. (2002). Animation: Can It Facilitate? *International Journal of Human-Computer Studies*, 57(4):247–262.

[Wilkinson, 2009] Wilkinson, L. (2009). The History of the Cluster Heat Map. *The American Statistician*, 63(2):179–184.