Die approbierte Originalversion dieser Diplom-/Masterarbeit ist an der Hauptbibliothek der Technischen Universität Wien aufgestellt (http://www.ub.tuwien.ac.at).

The approved original version of this diploma or master thesis is available at the main library of the Vienna University of Technology (http://www.ub.tuwien.ac.at/englweb/).

FAKULTÄT FÜR !NFORMATIK

Real Time Kollaborativer Personal Information Manager mit XVSM

DIPLOMARBEIT

zur Erlangung des akademischen Grades

Diplom-Ingenieur/in

im Rahmen des Studiums

Software Engineering & Internet Computing

eingereicht von

Dominik Röck
Matrikelnummer 0126080

am:
Institut für Computersprachen der Technischen Universität Wien

Betreuung:
Betreuerin: A.o. Univ. Prof. Dr. Dipl.-Ing. eva Kühn

Wien, 09.09.2008

(Unterschrift Verfasser/in) (Unterschrift Betreuer/in)

Danksagung

An dieser Stelle möchte ich all jenen danken, die durch ihre persönliche und fachliche Unterstützung zum Gelingen dieser Diplomarbeit beigetragen haben.

Besonderer Dank gebührt meinen Eltern, die mir dieses Studium durch ihre Unterstützung ermöglicht haben.

Weiters bedanke ich mich bei A.o. Univ. Prof. Dr. Dipl.-Ing. eva Kühn für die Betreuung meiner Diplomarbeit und ihre wertvollen Hinweise und Anregungen.

Inhaltsverzeichnis

ABBILDUNGSVERZEICHNIS 5
TABELLENVERZEICHNIS 7
ABKÜRZUNGSVERZEICHNIS 8
KURZFASSUNG 9
ABSTRACT 10
EINLEITUNG 11

A. Theoretische Grundlagen 13
1. Aktuelle Forschungen und Softwaresysteme 14
1.1. Momentaufnahme und einfache Systeme 15
1.2. Spezial-Systeme 18
1.3. Flexible Systeme 30
1.4. Verteilte Aufgabenmanager 33
2. Psychologische Hintergründe 37
2.1. Tagebuchstudie 37
2.2. Fragebogenstudie 42
3. PIM Anforderungen 46
3.1. Aufgaben- und Informationsmanagement 46
3.2. Daten- und Workflow-Management 50
3.3. Kooperatives Arbeiten 51
3.4. Kommunikationen mit anderen Applikationen 53
3.5. Spezialfunktionen 54
3.6. Zielgruppe 56
3.7. Abgrenzung und Diskussion 56
4. Technische Anforderungen 60
4.1. Zeit und Raum 60
4.2. Skalierbarkeit 61
4.3. Plattformunabhängigkeit 61
4.4. Flexibilität 62
4.5. Kompatibilität 62
4.6. Datenpersistenz 63

4.7. Sicherheit 63
4.8. Usability 63
4.9. Offline Betrieb 63
4.10. Konfliktlösung 64
5. Zusammenfassung 65

B. Umsetzung in der Praxis 67
1. Eingesetzte Technologien 68
1.1. Java 68
1.2. XVSM – MozartSpaces (www.mozartspaces.org) 68
2. System Design und Implementierung 75
2.1. Registry 75
2.2. User 79
2.3. TaskGroup 86
2.4. Task 90
3. Anwendung 96
3.1. Mogi 96
3.2. Mogi 2.0 97
3.3. Walkthrough 100
4. Bewertung 108
4.1. Zeit und Raum 108
4.2. Skalierbarkeit 108
4.3. Flexibilität 110
4.4. Plattformunabhängigkeit 110
4.5. Offline Betrieb 111
4.6. Datenpersistenz 111
4.7. Konfliktlösung 111
4.8. Sicherheit 111
4.9. Interaktion mit anderen Systemen 112
5. Ausblick 113
5.1. Integration von Dokumenten 113
5.2. Erweiterte Eingabe-/Ausgabemöglichkeiten 113
5.3. Zusammenspiel mit Applikationen 114
5.4. Positionserkennung 114

5.5. Optimierung des Registry Konzeptes 115
ZUSAMMENFASSUNG 116
LITERATUR 118

Abbildungsverzeichnis

Abbildung 1: Versuch mit Digital Notes; Bildschirmausschnitt 18
Abbildung 2: Hauptfenster von Taskmaster (Belotti et al. 2003) 21
Abbildung 3: Von Pepys (Lamming und Newman 1992) generiertes Tagebuch 23
Abbildung 4: Ergebnisse der Testreihe; ohne Hilfe, mit Hilfe von Pepys und mit Hilfe des Videotagebuches (Lamming und Newmann 1992) 24
Abbildung 5: Nahaufnahme von SensCam v. 2.3. und vorgesehne Trageweise. (http://research.microsoft.com/sendev/projects/sensecam/) 25
Abbildung 6: Hauptfenster von Swift To-Do List von Dextronet (http://www.dextronet.com/swift-to-do-list.php) 31
Abbildung 7: Userinterface des Prototyps von Kreifelts, Hinrichs und Woetzel (1993) 33
Abbildung 8: Hauptfenster von iGoogle To-Do Liste sowie Detailfenster für die Freigabe (www.igoogle.com) 34
Abbildung 9: Durchschnittliche Häufigkeiten der Gedächtnisprobleme (eigene Auswertung, Quelle: Eldridge, Sellen und Bekerian 1992) 43
Abbildung 10: Einteilung von Computer Supported Cooperative Work Applikationen (CSCW) nach R. Johansen (basierend auf Teich 2006) 60
Abbildung 11: Datenstruktur Container, Tuple, AtomicEntry in XVSM (Wittmann 2008) 70
Abbildung 12: Kommunikation zwischen verteilten Applikationen über Entries im Space (Wittmann 2008) 71
Abbildung 13: Klassenstruktur der Registry 77
Abbildung 14: Klassenstruktur des Users 84
Abbildung 15: Klassenstruktur der TaskGroup 88
Abbildung 16: Klassenstruktur des Tasks 93
Abbildung 17: Hauptfenster von Mogi 96
Abbildung 18: Hauptfenster von MOGI 2.0 99
Abbildung 19: Hauptfenster Benutzer Dominik: Hinzufügen eines Tasks 101

Abbildung 20: Hauptfenster Benutzerin Stefanie: der neue Task wird eingeblendet 102
Abbildung 21: Detailansicht - mittleres Hauptfenster 102
Abbildung 22: Hauptfenster Benutzerin Stefanie 104
Abbildung 23: Hauptfenster Benutzerin Stefanie: Anwendungsbeispiel private und öffentliche Priorität 105

Tabellenverzeichnis

Tabelle 1: Anzahl an Ressourcen und aktiven To-Dos pro Person aus der Kurzzeitstudie von Bellotti et al. (2004) 16
Tabelle 2: Durchschnittliche Häufigkeitsraten und durchschnittliche Anzahl von schweren Problemen in den Problemklassen (eigene Auswertung, Quelle: Eldridge, Sellen und Bekerian 1992) 43
Tabelle 3: Struktur der Registry 76
Tabelle 4: Struktur des XVSMPeers 76
Tabelle 5: Struktur der Inbox 77
Tabelle 6: Struktur von PrivateUser 81
Tabelle 7: Struktur von PublicUser 81
Tabelle 8: Struktur von UserGroup 82
Tabelle 9: Struktur von PrivateLocation 82
Tabelle 10: Struktur von PuplicLocation 83
Tabelle 11: Struktur von PrivateTaskGroup 87
Tabelle 12: Struktur von PublicTaskGroup 87
Tabelle 13: Struktur von Message 89
Tabelle 14: Struktur von PrivateTask 91
Tabelle 15: Struktur von PublicTask 92
Tabelle 16: Struktur von Note 92
Tabelle 17: Vergleichsmatrix der vorgestellten Systeme 107

Abkürzungsverzeichnis

BSCW	Basic Support for Cooperative Work
bzw.	beziehungsweise
CORSO	Coordinated Shared Objects Spaces
CSCW	computer supported cooperative work
CSS	Cascading Style Sheets
dh.	das heißt
etc.	et cetera
EuroCoOp	European Community of Consumer Cooperatives
GPS	Global Positioning System
GUI	Graphical User Interface
IDE	integrierte Entwicklungsumgebung
Mogi	Management Organiser
oä.	oder Ähnliche
PDA	persönlicher digitaler Assistent
PIM	Personal Information Manager
PR_deep	Passive replication with a deep object tree
TLM	Task List Manager
URL	Uniform Resource Locator
usw.	und so weiter
vgl.	vergleiche
WFMS	Workflow-Management-Systeme
XML	Extensible Markup Language
XVSM	eXtensible Virtual Shared Memory
zB.	zum Beispiel
zz.	zur Zeit

Kurzfassung

Aktuelle Recherchen zeigen, dass bereits eine Vielzahl an Aufgabenmanagement Systemen existiert. Während einige dieser Systeme eine Verteilung per E-Mail unterstützen, bietet der Großteil der Anwendungen weder eine gemeinsame Nutzung der Inhalte noch eine Synchronisierung dieser.

Aufbauend auf diesen Erkenntnissen und unter Berücksichtigung der menschlichen Arbeits- und Denkweise soll diese Diplomarbeit einen Ansatz bieten, der die bestehenden Nachteile beseitigt und ein benutzerfreundliches System zur Verwaltung von Aufgaben und Informationen liefert.

Dabei soll der Mensch in allen Situationen des täglichen Lebens unterstützt werden. Sei es, dass er beispielsweise an Aufgaben erinnert wird, um Ad-hoc-Abläufe abzubilden, oder um lediglich wichtige Daten permanent zu speichern. Insbesondere werden in dieser Arbeit die menschlichen Verknüpfungen des Gehirns berücksichtigt und genauer analysiert, um eine möglichst präzise Vorstellung von deren Einprägungs- sowie Erinnerungsverhalten zu bekommen. Diese Erkenntnisse werden dann im Aufgaben- und Informationsmanagement System anwenderfreundlich abgebildet, um die Auffindbarkeit der Daten zu erleichtern und das Arbeiten mit dem System so benutzerfreundlich wie möglich zu gestalten.

Abstract

Evidence indicates that there exist various different task management systems. However, while some assist with an allocation of information through e-mail support, they do not encourage the distribution, share and synchronization of these management assignments.

Referring to these findings, this Master Thesis constitutes an approach to amplify the coherent status quo and develop a revised task management tool in order to overcome these shortfalls.

This new task management tool should assist and support all situations of every day's life. Thus, it should allow for functionalities such as task reminder management, the planning of ad hoc workflows as well as for relevant data memory and storage. However, to fulfil these requirements, this Master Thesis not only explicitly considers human brain interconnections, but also exhibits a deeper analysis of those in order to gain an even more precise picture of imprinting and memory behaviour. In a next step those findings will then be incorporated into the task information and management system with the main aim of a user-friendly interface. Hence, the outcome should allow for efficient and simplified work with the task management system.

EINLEITUNG

Die zunehmende Globalisierung und die damit einhergehende Dynamik am Arbeitsplatz fordern den Menschen laufend. Die Arbeit in teils geographisch verteilten Teams ist dabei ein wichtiger Aspekt. Um diesen Herausforderungen gerecht werden zu können, bieten flexible Management-Systeme bereits eine Unterstützung in vielen Bereichen. Dazu zählen neben Konferenz-Tools, auch Work-Flow-Management Systeme und Group-Ware-Systeme.

Auch im privaten Leben, so zeigen Studien, benötigt der Mensch Unterstützung in seinen Tätigkeiten. Die häufigsten Probleme sind hier Unkonzentriertheit, das Vergessen von gewohnten oder geplanten Aktivitäten oder das Vergessen auf etwas zu antworten. Auch Fakten, Namen oder andere Information, die einem wichtig erscheinen, gehen oft verloren.

All diese Probleme deuten darauf hin, dass das menschliche Gehirn mit seinen Aufgaben überfordert ist. Ziel ist Arbeit ist es deshalb, ein Werkzeug zur Bewältigung von Aufgaben im täglichen Leben sowie eine Gedächtnisstütze zu schaffen. Besonderes Augenmerk wird darauf gelegt, den Menschen in seinen Aktivitäten möglichst umfassend zu unterstützen.

Zu diesem Zweck werden existierende Softwarelösungen analysiert, psychologische Hintergründe für Gedächtnisprobleme ermittelt und der Versuch gestartet, ein neues, sozial-kompetentes, technisch versiertes Tool zu entwickeln, das den Anforderungen einer Gedächtnisstütze entspricht.

Der Aufbau dieser Arbeit gliedert sich in einen theoretischen und in einen praktischen Teil. In dem theoretischen Teil werden, nachdem der Status quo an Softwarelösungen und die psychologischen Hintergründe betrachtet wurden, die Anforderungen an dieses Werkzeug ermittelt. In dem Praxisteil der Arbeit wird beschrieben, wie die gesammelten Ergebnisse bei der Entwicklung des Werkzeugs implementiert und Probleme bewältigt wurden. Mit einem Ausblick

auf Erweiterungen und Optimierungsmöglichkeiten wird die Arbeit abgeschlossen.

A. Theoretische Grundlagen

Die Entwicklung einer Softwarelösung für die umfassende Unterstützung des menschlichen Gedächtnisses erfordert neben der praktischen Entwicklung auch theoretische Überlegungen. Auf diese theoretischen Aspekte soll an dieser Stelle eingegangen werden.

Zu diesem Zweck wird der Status quo im Umgang mit Merkhilfen erfasst. Weiterhin werden gängige Softwarepraktiken analysiert, dh. es werden aktuelle Forschungen und Softwarelösungen begutachtet und bewertet.

Zur exakteren Bestimmung der Anwendungsbereiche und Features eines Aufgabenmanagers werden anschließend menschliche Gedächtnisprobleme durchleuchtet und ihr Einflussbereich erfasst. Dazu werden psychologische Forschungen und Studien betrachtet und ausgewertet.

Danach folgt eine Aufstellung der Anforderungen an Gedächtnisstützen im Allgemeinen. Dabei sollten möglichst alle Bereiche des täglichen Lebens abgedeckt werden. Zusätzlich fließen die Erkenntnisse über bestehende Forschungsarbeiten und Softwarelösungen in den Anforderungskatalog mit ein.

Abschließend werden die technischen Anforderungen an das System und vorausblickend mögliche Problembereiche ermittelt. Dazu werden die verschiedenen Gesichtspunkte der Software analysiert und ihre Ansprüche festgelegt.

1. Aktuelle Forschungen und Softwaresysteme

Personal Information Management umfasst im weiteren Sinne alle Hilfsmittel, die der Mensch zur Verwaltung und Wiederauffindung von Informationen jeder Art verwendet. Persönliche Vorlieben, das Umfeld, der Zugang zu Informationstechnologien sowie persönliche Verhaltensweisen prägen dabei die Auswahl und Anzahl an verwendeten Personal Information Management Werkzeugen (Kelly 2006). Im engeren Sinn wird unter einem „Personal Information Manager“ (PIM) eine Software verstanden, die persönliche Daten wie Kontakte, Aufgaben, Termine, Notizen und im weiteren Sinne auch Dokumente oä. verwaltet (http://de.wikipedia.org/wiki/Personal_Information_Manager).[1] Es gibt eine beachtliche Anzahl an Forschungsarbeiten in den Bereichen Koordinierungswerkzeuge und PIM. Neben reinen Forschungsarbeiten sind auch viele kommerzielle Produkte für PIM am Markt zu finden (Elsweiler & Ruthven 2007, Kelly 2006).

Sowohl kommerzielle Produkte als auch Forschungsarbeiten decken ein breit gefächertes Spektrum an verschiedenen Anwendungsbereichen und Lösungsansätzen ab. Einerseits wird versucht, spezifische Probleme zu lösen bzw. konkrete Anforderungen zu erfüllen. Zu diesem Bereich zählen unter anderem Terminkalender und Projekt Management Software für den Einzelanwender. Andere Lösungsansätze haben das Ziel, ein allumfassendes und in sich geschlossenes Produkt zur Bewältigung des Arbeitslebens zu schaffen. Dazu zählen Gruppen-Software und Groupware Lösungen, welche die Zusammenarbeit von Gruppen über zeitliche oder räumliche Distanz hinweg unterstützen (http://de.wikipedia.org/wiki/Groupware).

Vertreter der Gruppen-Software sind Virtuelle Office Systeme wie BSCW oder Microsoft Sharepoint. Dabei handelt es sich um Computersysteme, die es ermöglichen, dass sich BenutzerInnen unabhängig von ihrem Standort in einem

[1] Im weitern Teil dieser Arbeit bezieht sich die Abkürzung PIM auf die engere Definition eines „Personal Information Managers“.

System anmelden und zB. ihre E-Mails, Dokumente oder Kalender abrufen und benutzen können (http://en.wikipedia.org/wiki/Virtual_office). Diese Produkte sind für den Einsatz von Telearbeit geeignet, da sie die Arbeit an gemeinsamen Dokumenten ermöglichen und die Kommunikation zwischen den TeilnehmerInnen fördern. Es werden dabei schwach strukturierte Arbeitsabläufe von der Software unterstützt. Der Nachteil von Gruppen-Software ist, dass konkreten Geschäftsprozesse oder Abläufe nicht abgebildet werden können, wie es beispielsweise von Projektmanagementsoftware angeboten wird (Braun 2005).

Im Folgenden werden einige kommerzielle Applikationen und Forschungsergebnisse genauer beschrieben und analysiert. Die Produkte wurden hierfür nach Anwendungsgebieten gruppiert und ihre repräsentativen Funktionen genauer betrachtet. Da an dieser Stelle nicht alle angeführten Produkte im Detail beschrieben werden können, wird auf die jeweilige Literatur oder Produkthomepages verwiesen.

1.1. Momentaufnahme und einfache Systeme

In einem ersten Schritt wird auf einfache Lösungen, die der Verwaltung von persönlichen Informationen dienen, eingegangen, um in einem nächsten Schritt allgemein gängige Datenmanagement-Methoden zu betrachten.

Eine Momentanaufnahme von Bellotti et al. aus dem Jahre 2004 zeigt, wie umfangreich die Ressourcen sind, mit denen Personen ihre Aufgaben, auch To-Dos genannt, aufzeichnen bzw. festhalten. Im Rahmen einer Studie mit acht Testpersonen wurden alle aktiven Aufgaben der TeilnehmerInnen erfasst und die Ressourcen, mit deren Hilfe die Aktivitäten festgehalten wurden, aufgezeichnet. Insgesamt wurden 595 Aufgaben gesammelt. In Tabelle 1 sind die Ressourcen und ihre durchschnittliche Anzahl an To-Dos aufgelistet.

Besonders auffallend ist die bereits erwähnte hohe Anzahl an Ressourcen, die zur Verwaltung der Aufgaben verwendet wurden. So wurden im Durchschnitt

pro Person 11,25 verschiedene Hilfsmittel eingesetzt. Bei einer durchschnittlichen Anzahl von 74,4 Aufgaben pro Person ergaben sich so im Mittel 6,6 Aufgaben pro Ressource. Auch die Langzeitstudie von Bellotti aus dem Jahre 2004 bestätigt dieses Ergebnis und lässt erkennen, dass die TeilnehmerInnen ihre täglichen Aufgaben über zu viele verschiedene Ressourcen verwalten. Das Aufgabenmanagement wird durch das Fehlen von Einheitlichkeit und Übersichtlichkeit erschwert. Auf weitere Ergebnisse aus der Studie wird später noch einmal eingegangen.

Ressource	**Durchschnittliche aktive To-Dos pro Person**	**Aufteilung in % vom Gesamtwert der To-Dos**
E-Mail	26,6	35,8
Online Kalender	8,6	11,6
Papierform oder Notizzettel	7,2	9,7
Online Ordner	3,5	4,7
Online To-Do-Listen	3	4,0
PDA Kalender	2	2,7
Tagesplaner/ Notebook-Planer	1,6	2,2
...		

Tabelle 1: Anzahl an Ressourcen und aktiven To-Dos pro Person aus der Kurzzeitstudie von Bellotti et al. (2004)

Digital Notes

Ein Vertreter für eine einfache und praktische Softwarelösung sind die Digital Notes von dem Post-it Hersteller 3M (http://www.3m.com/us/office/postit/digital/digital_notes.html). Dabei handelt es sich nach eigenen Angaben um ein „Nachrichtenbrett mit Gehirn“, mit dessen Hilfe beispielsweise Listen erstellt und organisiert werden können. Tatsächlich handelt es sich um ein mit einem Texteditor vergleichbares Werkzeug. Das Tool unterstützt in der neuesten Version jedoch bereits das Sortieren der Post-its, das Einfügen von Bildern, das Setzen von Terminen und die Suche nach Stichworten. Weiterhin können Farben und Schriftarten der

Post-its frei gewählt werden, wodurch auch eine Kategorisierung ermöglicht bzw. simuliert wird.

Der Vorteil des Programms liegt in der einfachen Handhabung. Das Programm ist intuitiv bzw. in wenigen Minuten zu erlernen. Optisch ist es an die bereits bekannten Post-its angelehnt und daher den BenutzerInnen vertraut. Leider wurde die Umsetzung von Papier Post-its auf elektronische Post-its so konsequent vorgenommen, dass auch die Nachteile mit übernommen wurden. So bleibt beispielsweise die Kennzeichnung von bereits erledigten Tätigkeiten der Improvisationsfähigkeit der BenutzerInnen überlassen. Auch eine Priorisierung von Aufgaben ist nicht möglich. Die Software bietet hierfür keinerlei Unterstützung. Das wohl größte Manko der Software wird sogar von den Post-its in Papierformat besser gelöst. Denn wird ein Post-it beispielsweise auf dem Kühlschrank platziert, so sind die Nachrichten, Listen oder Ähnliches von allen Personen im Haushalt ersichtlich. Die Software erlaubt dies jedoch nur Personen, die unter demselben Account im System angemeldet sind.

Ein Versuch zeigt sehr schnell, dass sich auch das altbekannte Problem der Ansammlung von herkömmlichen Papier-Post-its am Bildschirmrand durch den Einsatz von Digital Notes nicht lösen lässt. So kann es passieren, dass der Bildschirm zu einer so genannten Pinnwand – wie in Abbildung 1 – umgestaltet wird. Ein zusätzliches Defizit ist die ausschließliche Verwendbarkeit der Applikation für WindowsbenutzerInnen.

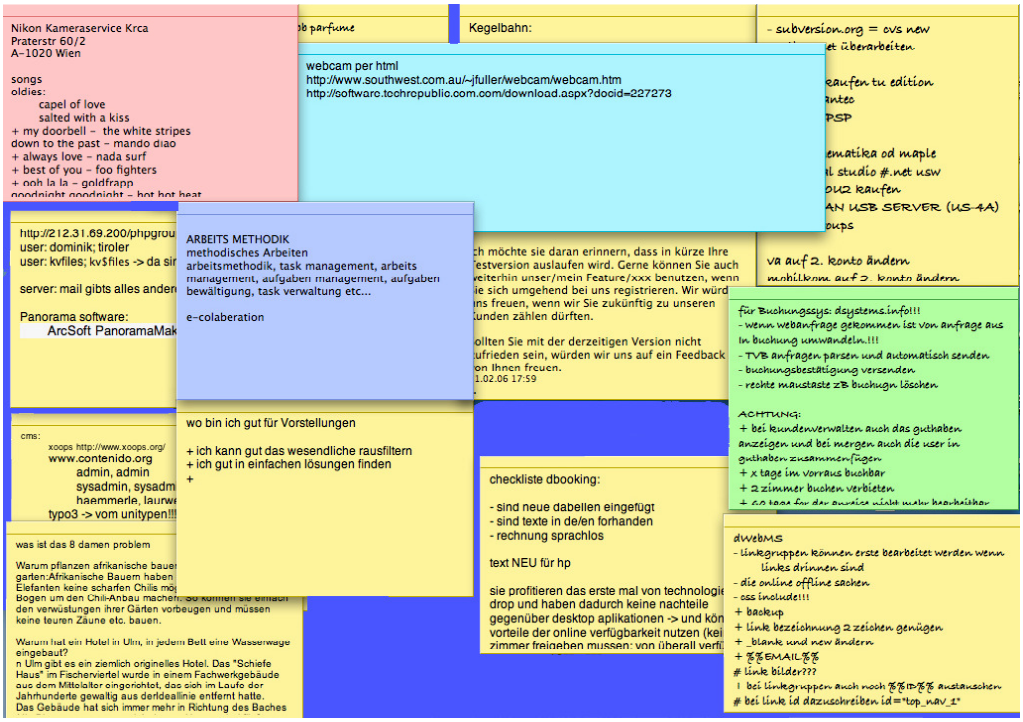


Abbildung 1: Versuch mit Digital Notes; Bildschirmausschnitt

Zusammenfassend kann festgestellt werden, dass persönliche Informationen und Aufgaben vorwiegend ineffizient verwaltet werden. Leider wird diese Situation auch durch einfache Werkzeuge nicht verbessert, da diesen oft die nötige Funktionalität fehlt. Sie bieten weder die Möglichkeit zur intelligenten Priorisierung noch zur Verteilung von Aufgaben und Information. Um die Probleme von solchen universell einsetzbaren Werkzeugen zu vermeiden und um auf die speziellen Bedürfnisse von BenutzerInnen einzugehen, wurden Speziallösungen zur Verwaltung und Organisation von Aufgaben entwickelt, die nun behandelt werden.

1.2. Spezial-Systeme

Unter den Spezialisten werden an dieser Stelle Applikationen verstanden, die durchaus für PIM geeignet sind, sich aber auf das Managen eines Spezialgebietes beschränken. Diese Gebiete beinhalten die Organisation von Daten, Faktenwissen, aber auch Aufgaben oder Erinnerungen.

Es befinden sich zahlreiche Werkzeuge zur Verwaltung von Faktenwissen auf dem Markt, einige sind beispielsweise Verwaltungssysteme für Kontakte, Passwortmanager oder Dokumentenverwaltungssysteme. Eines dieser Werkzeuge zur Verwaltung von Terminen oder Aufgaben kennen und nutzt viele, den persönlich favorisierten Kalender.

Aufgabenmanagement per E-Mail

Ein großer Teil der beruflichen und privaten Tätigkeiten wird heutzutage über den PC abgewickelt. E-Mail Clients stellen dabei zentrale Tools für den interpersonellen Kontakt dar. In zahlreichen Studien zu Personal Information Management zeigte sich bereits, dass PC AnwenderInnen die Aufgaben des täglichen Lebens größtenteils mittels E-Mails organisieren. Als PIM übernehmen E-Mail-Werkzeuge dabei drei Hauptfunktionen: Aufgabenmanagement, Archivierung von persönlichen Daten und Kontaktverwaltung. (Whittaker et al. 2006, Belotti et al. 2003, Belotti et al. 2004).

Ein kommerzielles E-Mail-Werkzeug mit erweiterter PIM Funktionalität stellt zum Beispiel Microsoft Outlook dar, das E-Mail, Aufgabenmanagement, Kontaktverwaltung und Kalender in einer Applikation vereint. Ein großer Nachteil dieses Systems ist jedoch, dass die einzelnen Werkzeuge nicht miteinander interagieren. So ist beispielsweise der Aufgabenmanager nicht mit der Inbox/Outbox verknüpft, obwohl Nachrichten und To-Dos oft ineinander greifen. Die Zusatzfunktionen unterstützen die AnwenderInnen daher auch nur unzureichend darin, die Informationsüberflutung die über das Medium E-Mail generiert wird, zu bewältigen. (Belotti et al. 2003, Whittaker et al. 2006)

Belotti, Ducheneaut, Howard und Smith entwickelten daher im Jahre 2003 einen auf E-Mail basierenden Prototyp für Aufgabenmanagement. Die Software Taskmaster kann als Add-On für Micrsoft-Outlook-Clients verwendet werden. Dabei wird ein Duplikat aller eingehenden sowie ausgehenden Nachrichten in den Taskmanager kopiert und dort weiter verarbeitet. Die Applikation

unterscheidet sich dabei in mehreren Punkten von einem gewöhnlichen E-Mail-Werkzeug.

- Hauptelement ist die Aufgabe, Thrask genannt, nicht die Nachricht. Zusammenhängende Nachrichten (Antworten etc.) werden in einem Thrask gruppiert. Datenanhänge oder Links können für die UserInnen von größerer Bedeutung sein, als die Nachricht selbst. Solche Informationen werden daher als eigenständiges Element im selben Thrask wie die Nachricht gespeichert. AnwenderInnen können auch andere Elemente, die niemals Teil der Nachricht waren, zu einem Thrask hinzufügen. Weiters ist es möglich Thrasks zu splitten.
- Thrasks können durch die UserInnen betitelt werden. Weiters können alle gesendeten und erhaltenen Nachrichten bearbeitet werden, zB. um persönliche Notizen festzuhalten.
- Jedes Element kann Metainformationen enthalten wie zB. Deadlines, Erinnerungen, Aktionen oder Farbcodes.
- Alle Elemente werden aggregiert und repräsentativ dargestellt. Informationen über Deadlines, Zusammenhänge zwischen Aktivitäten und aufgabenspezifische Kontakte sind gemeinsam mit dem Task „auf einen Blick“ ersichtlich.

In Abbildung 2 ist das Hauptfenster von Taskmaster ersichtlich. Im oberen Bereich des Fensters wird die Thraskliste dargestellt. Darunter im zentralen Bereich werden die Elemente des ausgewählten Thrasks wie Links, Attachments, usw. angezeigt. Der unterste Bereich dient zur Anzeige von Details des ausgewählten Elementes.

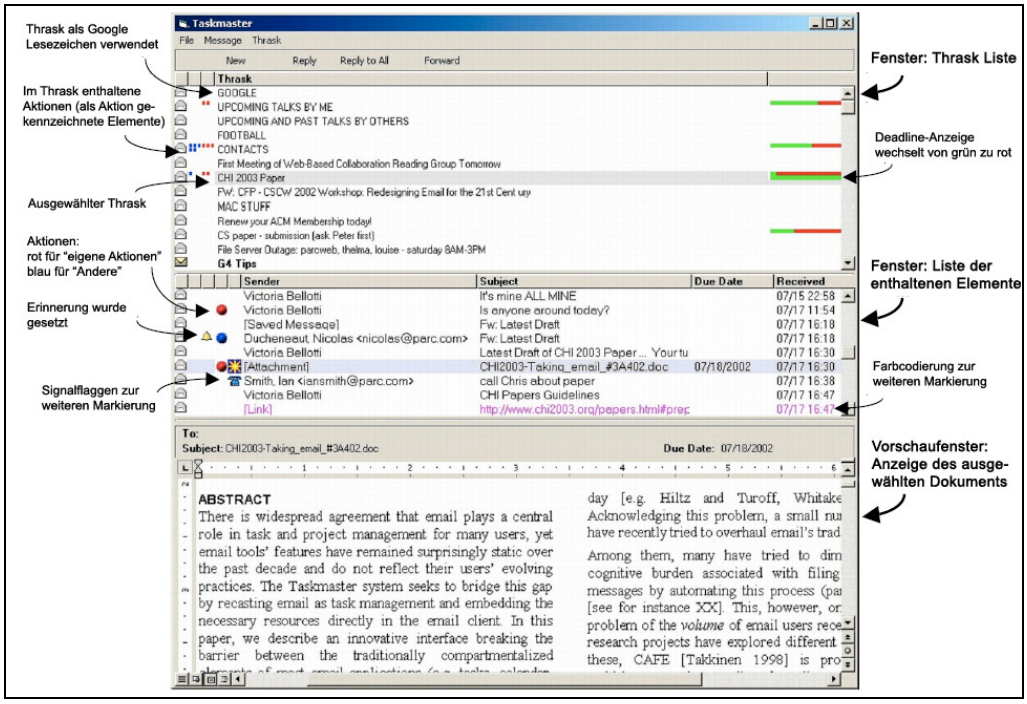


Abbildung 2: Hauptfenster von Taskmaster (Belotti et al. 2003)

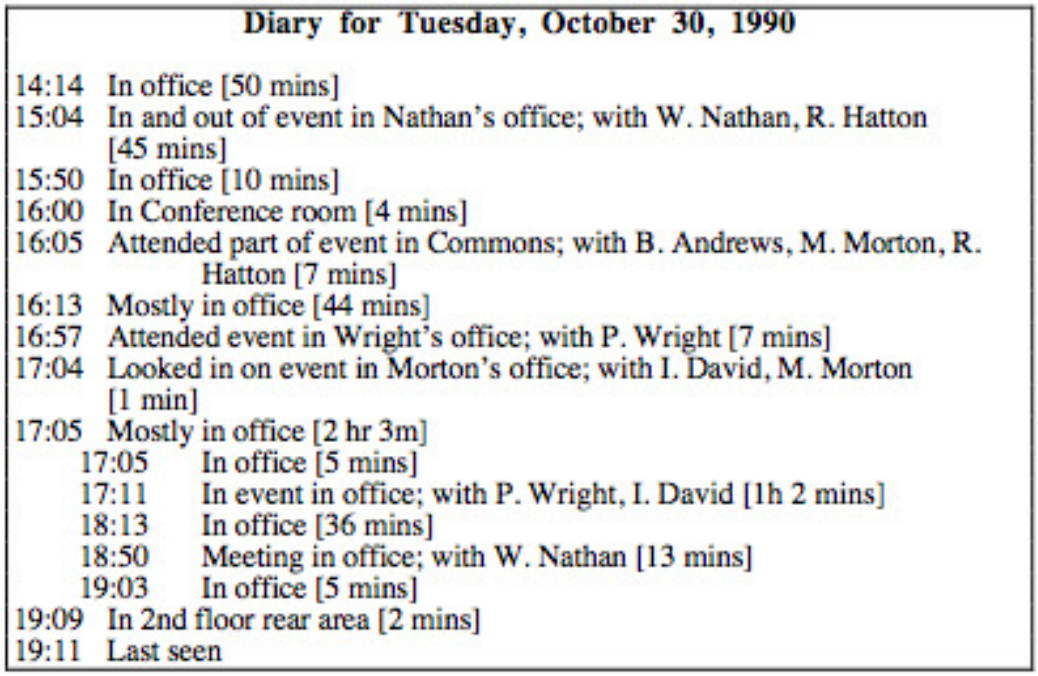
Der Prototyp wurde in einer zweiwöchigen Testreihe von 9 TeilnehmerInnen im täglichen Leben getestet. Einige TeilnehmerInnen verwalteten ihre Emails neben der Testreihe parallel weiterhin in Outlook. Der doppelte Arbeitsaufwand, der bei E-Mail-Werkzeugen aufgrund der Informationsflutung generell hoch ist, war ein großer Kritikpunkt der TeilnehmerInnen. Auch technische Einschränkungen, sowie von Outlook gewohnte, fehlende Features (zB Druckfunktion, Nachverfolgung von Emails) wurden bemängelt. Die Idee des auf E-Mail basierenden Aufgabenmanagers an sich wurde von den Teilnehmern sehr positiv bewertet. Insbesondere das Konzept des Thrask als zentrales Element, um Nachrichten, Dokumente, Links etc. zu verwalten, fand großen Anklang. Hilfreich waren die Deadline-Balken, die anzeigen welcher Thrask die dringlichsten To-Dos enthält.

Tagebuchgenerator

Um gezielter auf die Anforderungen der BenutzerInnen einzugehen, wurde von Lamming und Newman im Jahre 1992 ein weiterer Ansatz für Applikationen entwickelt. Bei dieser Gedächtnisstütze handelt es sich um so genannte Tagebuchgeneratoren. Ein aktueller Forschungsprototyp wird derzeit von

Microsoft Research Cambridge unter der Bezeichnung SenseCam entwickelt. Das Konzept der Tagebuchgeneratoren basiert auf der Erkenntnis, dass sich die AnwenderInnen zum aktuellen Zeitpunkt nicht im Klaren sind, welche Informationen in Zukunft für sie relevant sein werden. Daraus entstand die Überlegung: „Je mehr Informationen dem System bekannt sind, desto höher ist die Wahrscheinlichkeit, dass die benötigten Informationen enthalten sind." Dies erfordert, dass das System selbst die Daten aufzeichnet, die von den AnwenderInnen als nicht relevant eingestuft werden. Aufgrund dieser Notwendigkeit werden diese Systeme so entworfen, dass die Umgebung nach Daten abgetastet wird sowie diese mit einem Zeitstempel versehen und gespeichert werden.

Lamming und Newman entwickelten mehrere Prototypen. In der Folge werden zwei ihrer Tagebuchgeneratoren genauer beschrieben. Beim ersten Prototyp handelt es sich um einen automatischen Tagebuchgenerator, der „Pepys" genannt wurde. Pepys ist ein System, dem es möglich ist, mit Hilfe von Infrarotschnittstellen zu kommunizieren. Sender an Wänden und Geräten teilen dem System Aktivitäten der BenutzerInnen mit. Es werden beispielsweise Tätigkeiten wie das Eintreffen in das Büro, das Absenden eines Fax oder das Arbeiten am PC mit einem Zeitstempel versehen gespeichert. Da sich der Mensch genaue Zeiten jedoch nur sehr schwer merken kann, werden die vom System elektronisch erfassten Aktivitäten auf eine für den Menschen verständliche Ebene umgerechnet. Dh. die im System eingehenden Signale werden auf Zusammenhänge und Abhängigkeiten geprüft und daraus Tätigkeiten generiert. Anhaltspunkte für das System sind dabei Position, Telefonanrufe oder PC-Gebrauch. Die aufgezeichneten Ergebnisse werden den AnwenderInnen täglich per E-Mail zugestellt und können so permanent gespeichert werden. Ein typisches Beispiel des erzeugten Tagebuches ist in Abbildung 3 ersichtlich.



Diary for Tuesday, October 30, 1990

14:14 In office [50 mins]
15:04 In and out of event in Nathan's office; with W. Nathan, R. Hatton [45 mins]
15:50 In office [10 mins]
16:00 In Conference room [4 mins]
16:05 Attended part of event in Commons; with B. Andrews, M. Morton, R. Hatton [7 mins]
16:13 Mostly in office [44 mins]
16:57 Attended event in Wright's office; with P. Wright [7 mins]
17:04 Looked in on event in Morton's office; with I. David, M. Morton [1 min]
17:05 Mostly in office [2 hr 3m]
17:05 In office [5 mins]
17:11 In event in office; with P. Wright, I. David [1h 2 mins]
18:13 In office [36 mins]
18:50 Meeting in office; with W. Nathan [13 mins]
19:03 In office [5 mins]
19:09 In 2nd floor rear area [2 mins]
19:11 Last seen

Abbildung 3: Von Pepys (Lamming und Newman 1992) generiertes Tagebuch

Ein Vorteil von Pepys ist ohne Zweifel, dass im beobachteten Bereich keinerlei Informationen verloren gehen. Damit einhergehend lässt sich allerdings bereits ein Nachteil erkennen. Da das System auf den Arbeitsbereich der Sensoren angewiesen ist, können nur Aktivitäten im Umkreis dieser Sensoren aufgezeichnet werden. Verlässt man den Bereich, der für eine Anwendung von Pepys ausgelegt ist, erfolgt keine Aufzeichnung – alle Informationen gehen verloren. Des Weiteren können im Einzugsbereich von Pepys nur solche Aktivitäten protokolliert werden, für die Sensoren installiert wurden. Sollen zusätzliche Informationen aufgezeichnet werden, ist eine aufwendige Nachrüstung erforderlich. Es kann daher davon ausgegangen werden, dass die erhaltenen Informationen größtenteils zu ungenau sind und zu wenig Aussagekraft besitzen. Beispielsweise sagt die Information, dass ein Telefonanruf getätigt wurde, noch nichts über dessen Inhalt aus.

Videotagebuchgenerator

Bei dem zweiten System, das von Lamming und Newman (1992) entwickelt wurde, handelt es sich um ein Videotagebuch. Kameras in der gesamten Umgebung der AnwenderInnen erfassen dabei ihre Tätigkeiten. Beim Wechseln

von Positionen wird den AnwenderInnen von den Kameras automatisch verfolgt. Auch hier werden die gesammelten Daten mit einem Zeitstempel versehen gespeichert und können nach Bedarf wieder abgerufen werden.

In Abbildung 4 sind die Resultate einer Testreihe mit und ohne Hilfe der beiden Systeme dargestellt. Dazu wurden die AnwenderInnen im Rahmen von vier Sitzungen (Recall Session R1 bis R4) zu ihren Tätigkeiten befragt. Die Graphik zeigt den Anteil der abgerufenen Tätigkeiten (Proportion of Recalled Activities) je System und Recall Session. Die Ergebnisse werden gemessen an den Ergebnissen der Videotagebuchstudie (100%) dargestellt. Es geht deutlich hervor, dass sich die Versuchspersonen mit Hilfe von Pepys und des Videotagebuches an mehr Ereignisse erinnern können.

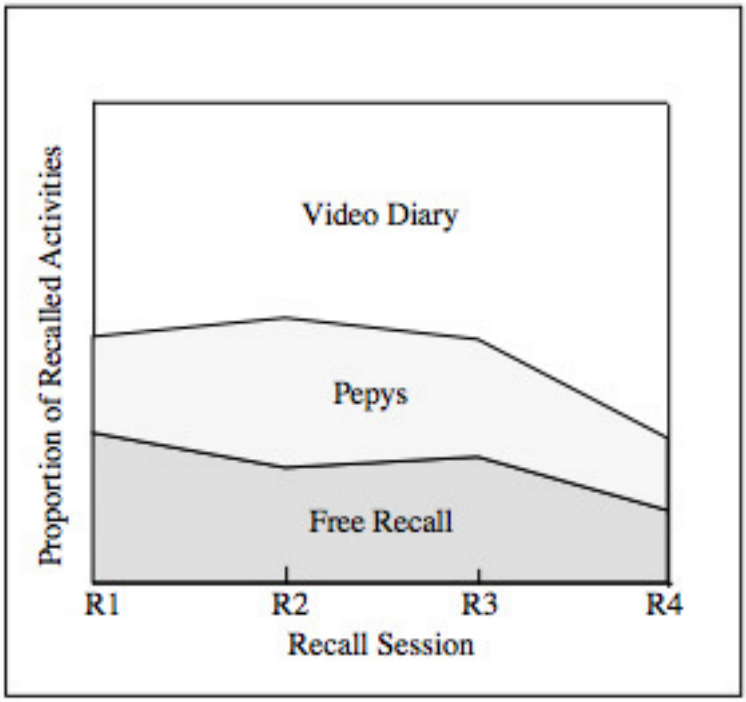


Abbildung 4: Ergebnisse der Testreihe; ohne Hilfe, mit Hilfe von Pepys und mit Hilfe des Videotagebuches (Lamming und Newmann 1992)

Ein Vorteil des Videotagebuchsystems zeigt sich in der zusätzlichen Aufzeichnung der gesprochenen Worte und Bilder, wodurch sämtliche Inhalte von Gesprächen abrufbar sind. Allerdings gestaltet sich die Suche nach Inhalten, trotz der auch hier verwendeten Zeitstempel und Verknüpfungen der Tätigkeiten, eher schwierig und zeitaufwendig. Weiterhin ist der

Speicheraufwand, trotz inzwischen günstig erhältlicher Speichermedien, kaum zu bewältigen. Wird das System in der Arbeitsumgebung eingesetzt, ist zu bedenken, dass sich die BenutzerInnen durch die Videoaufzeichnungen beobachtet und überwacht fühlen und daher die nötige Akzeptanz fehlt.

SenseCam

Einer der aktuellen Forschungsprototypen im Bereich der Tagebuchgeneratoren ist die von Microsoft Research Cambridge entwickelte SenseCam. Das Grundkonzept von SenseCam ist den oben beschriebenen Systemen von Lamming und Newmann (1992) sehr ähnlich. Eine portable Digitalkamera zeichnet automatisch Bilder auf und reagiert aufgrund integrierter Licht-, Infrarot-, und Temperatursensoren, etc. auf Veränderungen in der Umgebung der Person. Die Kamera ist mit einem Weitwinkelobjektiv ausgestattet, um ein möglichst großes Blickfeld erfassen zu können. Die Aufzeichnungen können automatisiert sowie auch manuell durch die BenutzerInnen getätigt werden. Die Ereignisse des Tages werden in VGA Qualität auf einer digitalen Speicherkarte gesichert (ca. 30.000 Bilder) und können extern gespeichert werden. Die Kamera ist in Abbildung 5 ersichtlich.
(http://research.microsoft.com/sendev/projects/sensecam/)

Abbildung 5: Nahaufnahme von SensCam v. 2.3. und vorgesehne Trageweise. (http://research.microsoft.com/sendev/projects/sensecam/)

Eine im Jahre 2007 von Sellen et al. durchgeführte Studie zur Gedächtnisunterstützung von SenseCam zeigte, dass SenseCam das Erinnerungsvermögen der Testpersonen effektiv unterstützt. Das Erinnerungsvermögen nahm jedoch mit Zeitverlauf (Dauer der Studie 3-10

Tage) ab. Weiters stellt sich heraus, dass sich die Testpersonen teils besser an Aktivitäten erinnerten, wenn diese passiv durch SenseCam erfasst wurden, als wenn die Teilnehmer die Tätigkeiten selbst festhielten.

Insgesamt kann festgehalten werden, dass das SenseCam Konzept zwei große Nachteile der zuvor beschriebenen Systeme wettmacht. Erstens sind durch die portable Kamera keine räumlichen Einschränkungen gegeben. Zweitens können die BenutzerInnen Momente, die ihnen besonders wichtig erscheinen, aktiv festhalten. Auch das Problem der hohen Speicherkapazität ist aufgrund der fortgeschrittenen Technik nicht mehr so gravierend. Ein Defizit ist jedoch die Handhabung des Systems, da die BenutzerInnen die SenseCam mit Sicht auf das Umfeld getragen müssen. Durch die automatisierte Aufzeichnung können auch rechtliche Probleme im Sinne des Datenschutzes etc. auftreten.

Computer Supported Cooperative Work

Abschließend soll an dieser Stelle eine Gruppe von Applikationen erwähnt werden, die zur Unterstützung von kooperativer Arbeit – Computer Supported Cooperative Work (CSCW) – eingesetzt werden. Es handelt sich dabei um umfangreiche Softwareprodukte für Datenaustausch, wie Basic Support for Cooperative Work (BSCW) von OrbiTeam und Microsoft Sharepoint Services sowie Softwareprodukte für kooperative Synchronisierung wie Microsoft Office Groove und Live Mesh.

Bei BSCW handelt es sich um eine Webanwendung, die den BenutzerInnen Zusammenarbeit über das Internet ermöglicht. Die AnwenderInnen können dabei mit Hilfe des Web-Browsers auf das System und die Daten zugreifen. Es werden unter anderen, folgende Funktionalitäten angeboten:

- Erstellen und Verwalten von gemeinsamen Arbeitsbereichen
- Ablegen, Verteilen und Verwalten von Dateien (Dokumente, Bilder etc.) inklusive Versionisierung und Benachrichtigungsservice für Änderungen
- Managen von Terminen (Erinnerungen), Kontakten, Aufgaben und Notizen

Überblick über Aktivitäten der Teammitglieder

Erstellen und Veröffentlichen von Blogs

Erstellen von Umfragen

Die Informationen stehen allen Berechtigten jederzeit und ortsunabhängig zur Verfügung und komplexe Arbeitsabläufe können koordinieren werden. Weiters wird die Kommunikation zwischen den Beteiligten in Form von E-Mails, Chat, Diskussionsforen und Konferenzschaltungen unterstützt. Zur Bearbeitung der verwalteten Dokumente werden diese auf den lokalen Rechner geladen. (www.bscw.de, http://de.wikipedia.org/wiki/BSCW)

Der Vorteil des Systems ist, dass es sowohl für Windows- als auch für UNIX-Betriebsysteme angeboten wird. Als Nachteil kann genannt werden, dass das System sehr komplex ist und etwas Einarbeitungszeit in Anspruch nimmt, um effektiv arbeiten zu können.

Auch bei Microsoft Sharepoint Services handelt es sich um eine Webanwendung, zu der die BenutzerInnen per Web-Browser Zugriff erhalten. Sharepoint Serves stellt folgende Funktionen zur Verfügung:

Erstellen von Arbeitsbereichen für Gruppen

Zentrale Datenablage mit Benachrichtigungsservice bei Änderungen

Managen von Terminen (Erinnerungen), Ankündigungen, Kontakten, Aufgaben und Notizen

Überblick über Aktivitäten der Teammitglieder

Erstellen von Blogs und Wikis

Integration in Microsoft Office Produkte

Integration von Workflows

Neben Dokumentenmanagement bietet Microsoft Sharepoint Services auch die Möglichkeit für Erweiterungen durch die Integration bzw Erstellung von Webapplikationen. (http://office.microsoft.com/de-de/sharepointtechnology, http://de.wikipedia.org/wiki/Windows_SharePoint_Services)

Im Vergleich zu BCSW ist die Integration in die Officeprodukte von Microsoft ein großer Vorteil von Sharepoint Services. Damit wird ein einfaches und schnelles Veröffentlichen bzw Bearbeiten von Inhalten ermöglicht. Negativ zu

bewerten ist die Vorraussetzung von Microsoft Server als Betriebssystem für die Installation der Webanwendung.

Im Gegensatz zu den beiden bereits erwähnten Systemen, handelt es sich bei Groove um keine Webanwendung, sondern um ein System, das Daten zwischen Computern über Peer-To-Peer Verbindungen austauscht. Für die Verwendung des Systems ist demzufolge eine Clientsoftware nötig, die in die Microsoft Office integriert ist. Die gebotene Funktionalität umfasst:

Erstellen von Arbeitsgruppen
Gemeinsame Nutzung von Dateien
Konfliktlösung bei gleichzeitiger Bearbeitung von Dokumenten
Informationen über Aktivitäten der Teammitglieder
Synchronisation mit Sharepoint Services
Chat, Voicemail

Der Vorteil des Systems ist, dass die Daten lokal gespeichert und damit jederzeit zur Verfügung stehen. Das System steht jedoch, wie bereits Sharepoint Services, nur für WindowsanwenderInnen zur Verfügung. Ein weiter Nachteil ist die Verwendung eines zentralen Servers als Relay zwischen den Clients; damit findet keine reine Peer-To-Peer Kommunikation statt. (http://office.microsoft.com/de-de/groove, http://de.wikipedia.org/wiki/Microsoft_Groove)

Auch bei Live Mesh handelt es sich um eine Plattform, in der die Synchronisation zwischen den verschiedenen Geräten des Anwenders im Vordergrund steht. Es werden Clients für Windows, sowie in Zukunft auch für Windows Mobil und Macintosh Computern zur Verfügung gestellt. Jedem einzelnen Geräten kann mitgeteilt werden, welche Daten zu synchronisieren sind. Auch Gruppenarbeit wird unterstützt, indem Inhalte an andere BenutzerInnen freigegeben und synchronisiert werden. Ein Webinterface bietet dem Anwender zudem die Möglichkeit von Unterwegs (zB. in Internetcafes) auf Daten zuzugreifen. Um die Übersicht zu behalten, ist je Element ersichtlich, welche Kontakte eine Zugriffsberechtigung besitzen und ob jemand an den Daten arbeitet. Für die Kommunikation zwischen den Kontakten ist Microsoft

Messenger in Life Mesh integriert. (www.mesh.com, www.chip.de/artikel/Live-Mesh-Microsoft-bastelt-am-Netz-im-Netz_31718355.html)

Der Vorteil des Systems liegt in der einfachen Handhabung. Eine aufrechte Internetverbindung reicht aus, um die Daten auf den aktuellen Stand zu bringen. Ein Nachteil des Systems könnte sein, dass BenutzerInnen nicht bereit sind sensible Daten auf Fremdserver zu speichern.

Workflow-Management-Systeme

Im Gegensatz zu CSCW werden in Workflow-Management-Systemen (WFMS) Unternehmensprozesse abgebildet. Dabei übernimmt das System sowohl den Kontrollfluss der Kommunikation zwischen den beteiligten MitarbeiterInnen als auch die Steuerung der benötigten Anwendungen. Ein Nachteil dieser Systeme ist, dass die Abläufe im Vorfeld modelliert werden müssen und nicht flexibel oder ad hoc durch die BenutzerInnen angepasst werden können. (Braun 2005)

Die hier erwähnten Produkte und Forschungsarbeiten eigenen sich für spezielle Anwendungsbereiche. Groupware Systeme zielen insbesondere auf die Koordination und Zusammenarbeit von Teams ab, die Verwaltung von persönlichen Informationen steht dabei nicht im Vordergund. Workflow Systeme sind durch ihre Auslegung für Unternehmensprozesse zu komplex, um einfache Projekte damit abzubilden. Tagebuchgeneratoren helfen bei der Erinnerung an geschehende Ereignisse, bieten jedoch keine Unterstützung für zukünftige Aufgaben.

Die vorgestellten Systeme zeichnen sich im jeweiligen Einsatzgebiet durch eine effektive Unterstützung aus. Eine Kombination grundlegender Features ist bei der Konzipierung eines PIMs daher sinnvoll. Die Tools sollten jedoch für den persönlichen Gebrauch im Alltag konzipiert sein und eine einfache, schnelle und unkomplizierte Handhaben erlauben.

1.3. Flexible Systeme

In diesem Abschnitt werden Werkzeuge analysiert, die ein möglichst breites Anwendungsgebiet erlauben.

Menschliche Gedächtnisstützen

Lamming et al. führten im Jahre 1994 die gemeinsamen Ideen von Newman und ihm selbst aus dem Jahre 1992 weiter und entwickelten einen Leitfaden für Systeme, die als „menschliche Gedächtnisstützen“ bezeichnet werden. Ziel dieser Systeme ist es, Faktenwissen zu speichern und wieder abrufbar zu machen. Dabei soll sowohl das Auffinden von Dateien, Dokumenten oder Notizen als auch die Erinnerung an Namen, Personen und Plätze unterstützt werden. Auch diese Systeme sollen Wissen zur Verfügung stellen, bei dem sich die AnwenderInnen zum aktuellen Zeitpunkt nicht bewusst sind, es jemals wieder zu benötigen. Auch hier gilt der Grundsatz – je mehr Daten das System speichert, desto höher ist das Potential für eine Unterstützung. Wie in den Systemen zuvor erfolgt die Datenerfassung automatisch. Die Systeme von Lamming und Newman aus dem Jahre 1992 werden dabei erweitert, sodass zukünftige Ereignisse bzw. Aufgaben eingegeben werden können und Erinnerungen möglich sind.

Noch weiter geht Brown im Jahre 1995 und erweitert das System von Lamming et al. dadurch, dass Informationen über den Aufenthaltsort der BenutzerInnen, beispielsweise aus den GPS Daten des Eingabegerätes, erfasst werden. Bei der Eingabe von Aufgaben werden daher die Lokalitäten der AnwenderInnen mitgespeichert. Kommen die BenutzerInnen zu einem späteren Zeitpunkt wieder in diese Umgebung, wird die Aufgabe automatisch im System angezeigt. Ähnliches wird auch durch Personen im Umfeld der BenutzerInnen ausgelöst.

Task List Manager

Ein anderer Ansatz für flexible Gedächtnisunterstützung sind Aufgabenmanager. Sie bieten keine automatische Tagebuchfunktionalität, dennoch sind sie universell einsetzbar. Task List Manager sind wie oben

erwähnt bereits in vielen Systemen mit PIM-Funktionalität integriert, wie beispielsweise Microsoft Outlook. Im Internet werden zahlreiche Produkte wie OrchestrateHQ (www.orchestratehq.com), Ta-da Lists (www.tadalist.com) und Remember the Milk (www.rememberthemilk.com) angeboten. Stellvertretend wird hier die Software „Swift To-Do List“ von Dextronet beschrieben. In Abbildung 6 ist das Hauptfenster der Applikation dargestellt. Dabei handelt es sich um eine Applikation, in welcher Aufgaben in Listenform organisiert werden können, dem so genanten Task List Manager (TLM). Die erstellten Aufgaben können dabei durch Priorität, Kategorie, Datum und Erinnerung ergänzt werden. Auch eine Sortierung kann nach diesen Attributen erfolgen. Weiterhin ermöglicht das System, Notizen zu den Einträgen hinzuzufügen. Dadurch wird auch das Festhalten von Faktenwissen ermöglicht. (http://www.dextronet.com/swift-to-do-list.php)

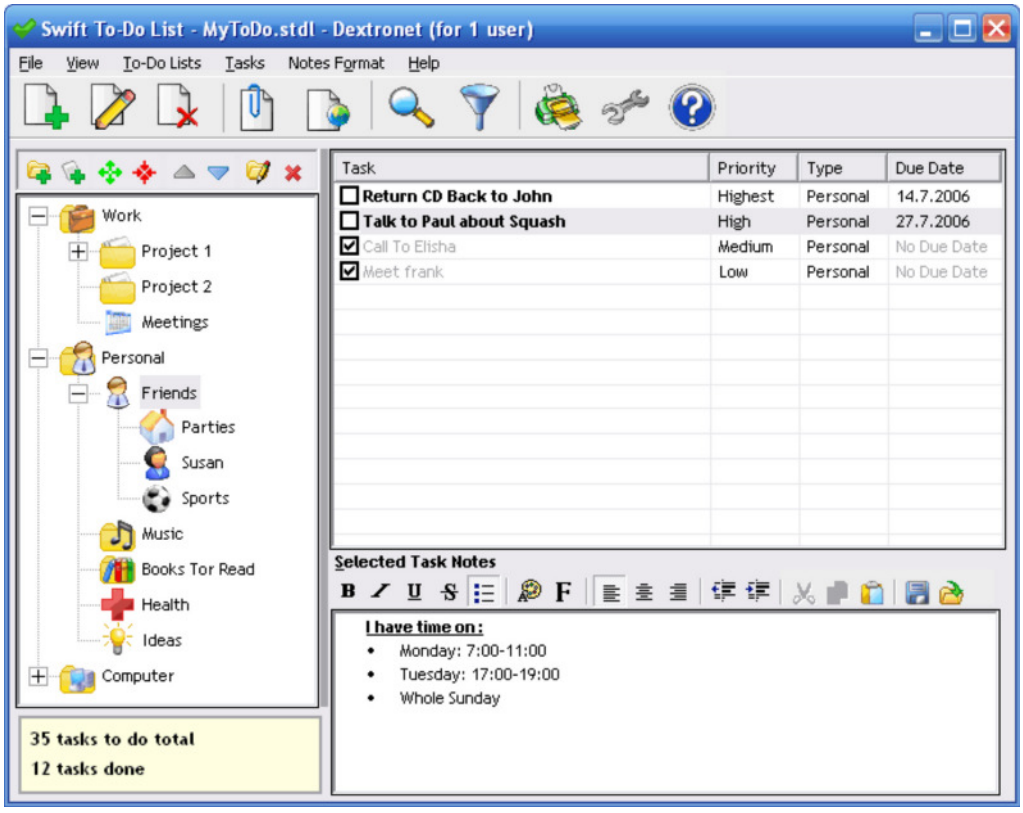


Abbildung 6: Hauptfenster von Swift To-Do List von Dextronet (http://www.dextronet.com/swift-to-do-list.php)

Bellotti et al. führten im Jahre 2004 nach intensiver Studie des Umgangs mit Aufgaben und deren Management ebenso einen TLM ein. Es wurde dabei vor

allem darauf Wert gelegt, den Aufgaben zusätzliche Eigenschaften zuordnen zu können. So ist es möglich, eine für den Task zuständige Person zu deklarieren bzw. die Örtlichkeit, in der die Aufgabe stattfindet, anzugeben. Auch Beziehungen zwischen den Aufgaben werden in Form einer Baumstruktur unterstützt.

Die Studien von Bellotti et al. haben ergeben, dass ein Großteil der eingetragenen Aufgaben nur für einen gewissen Zeitraum relevant ist. Die Versuchspersonen verwenden für die Bezeichnung der notierten Aufgaben in To-Do Listen oder in ihren Notizen häufig nur für sie verständliche Abkürzungen oder Begriffe. Die Begriffe sind für die BenutzerInnen während des Zeitraumes der Tätigkeit verständlich, nach gewisser Zeit jedoch wird meist vergessen, wofür der Eintrag vorgesehen war. Weiterhin zeigt die Studie, dass die BenutzerInnen zu keinen unnötigen Eingaben gezwungen werden wollen. Diesen Anforderungen wird dadurch entsprochen, dass alle Eingaben optional und dennoch jederzeit leicht zu ergänzen sind. Beispielsweise ist es für die AnwenderInnen nicht erforderlich anzugeben, ob es sich bei der Aufgabe um eine schnell zu erledigende Tätigkeit oder um ein komplexes Projekt handelt. Solche Angaben können aber zu einem späteren Zeitpunkt weiterhin erfolgen. Außerdem besitzt der Prototyp von Bellotti et al. ein flexibles Filtersystem sowie verschiedene Sichten auf die Aufgaben. Damit können Aufgaben schnell und einfach nach einem bzw. mehreren Attributen sortiert/gefiltert werden. Allerdings unterstützt das TLM von Bellotti et al. die gemeinsame Verwendung von Aufgaben noch nicht.

Zusammenfassend kann festgehalten werden, dass die beschriebenen Anwendungen flexible und universelle Einsatzmöglichkeiten zur Bewältigung von Aufgaben im täglichen Leben bieten. Lediglich der Aspekt der verteilten und gemeinsamen Bearbeitung von Aufgaben wurde noch nicht berücksichtigt. Dazu wird nun eine Forschungsarbeit, die dies in einem Prototyp verwirklicht, betrachtet.

1.4. Verteilte Aufgabenmanager

Kreifelts, Hinrichs und Woetzel entwickelten im Jahre 1993 in Zusammenarbeit mit der European Community of Consumer Cooperatives (EuroCoOp) ein Prototyp Software System. Ausgiebige Untersuchungen hatten ergeben, dass die Probleme von bisherigen Systemen vor allem in der Starrheit und Isolation zu anderen Applikationen liegen. Der entwickelte Prototyp ermöglicht es, kooperative Aktivitäten zu spezifizieren und zu managen. Im Detail wird den BenutzerInnen ermöglicht, Aufgaben zu organisieren, Prozesse zu verfolgen, Dokumente gemeinsam zu nutzen und Nachrichten auszutauschen. In Abbildung 7 ist das Interface des Prototyps ersichtlich.

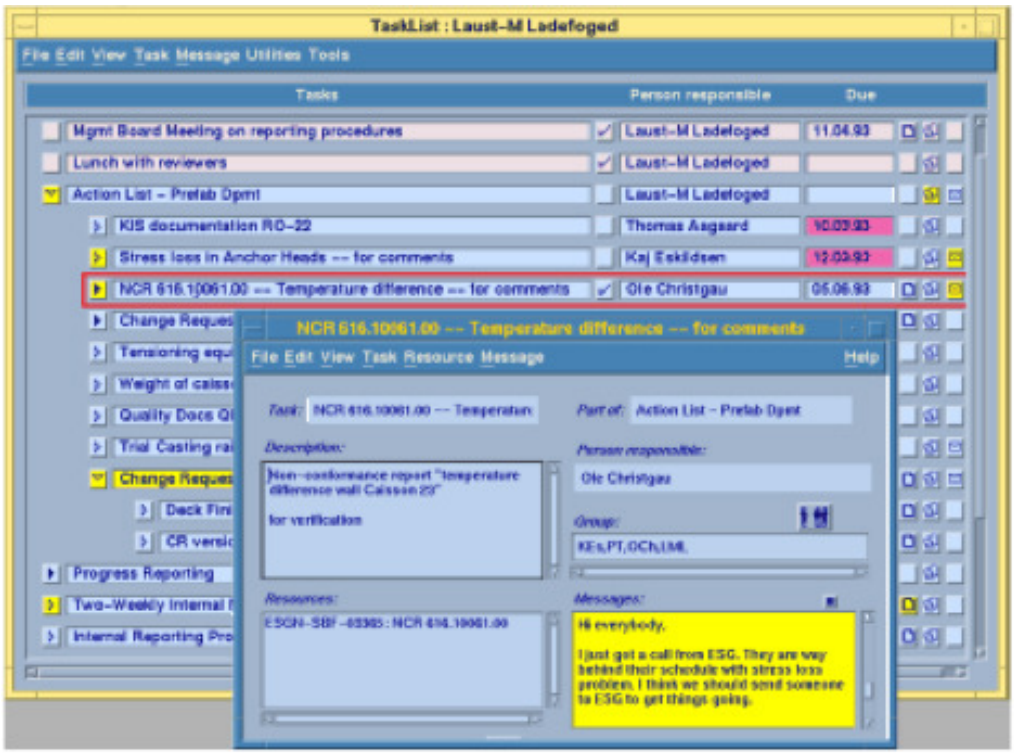


Abbildung 7: Userinterface des Prototyps von Kreifelts, Hinrichs und Woetzel (1993)

Durch die gebotene Funktionalität des verteilten Aufgabenmanagers ergeben sich folgende Einsatzgebiete:

- Persönliche To-Do Listen
- Brainstorming
- Vorbereitungen von Meetings

- Projektplanung und Verfolgung
- Projektdurchführung
- Wiederkehrende Aufgaben
- und als Mitteilungsbrett

(Kreifelts, Hinrichs und Woetzel 1993)

Aktuell werden gemeinsam verwendete Task-Manager vielfach über Internet genutzt. Die zuvor genannten Systeme wie Ta-da Lists bieten großteils die Möglichkeit, die Listen via E-Mail oder Online zu sharen. Auch iGoogle (www.igoogle.com) bietet ein Gadget, das es ermöglicht To-Do Listen zu erstellen und mit Freunden und Bekannten zu sharen, um diese gemeinsam bearbeiten zu können. Der TLM bietet die Möglichkeit mehrere Listen zu erstellen. Die Listen können mehrere Tasks enthalten, jedoch keine weiteren Sub-Tasks. Jeder Liste kann mit Fälligkeitsdatum, Priorität und Lokalität versehen werden. Freunde können per E-Mail über die To-Do Liste informiert werden, und per Link auf diese zugreifen. Dabei kann ausgewählt werden, ob die Person die Inhalte nur lesen oder auch bearbeiten kann. Ein Nachteil des Gadgets ist jedoch, dass es nicht möglich ist, einzelne Listen/Tasks zu sharen; es werden immer alle Inhalte des Task-Managers freigegeben. Durch die Hinzufügung weiter Gadgets wie Kalender, oder Notizfunktion zu iGoogle, kann das System flexibel erweitert werden. Wobei die einzelnen Tools leider nicht miteinander interagieren. Das Hauptfenster von iGoogle To-Do Liste ist in Abbildung 8: Hauptfenster von iGoogle To-Do Liste sowie Detailfenster für die Freigabe (www.igoogle.com)ersichtlich.

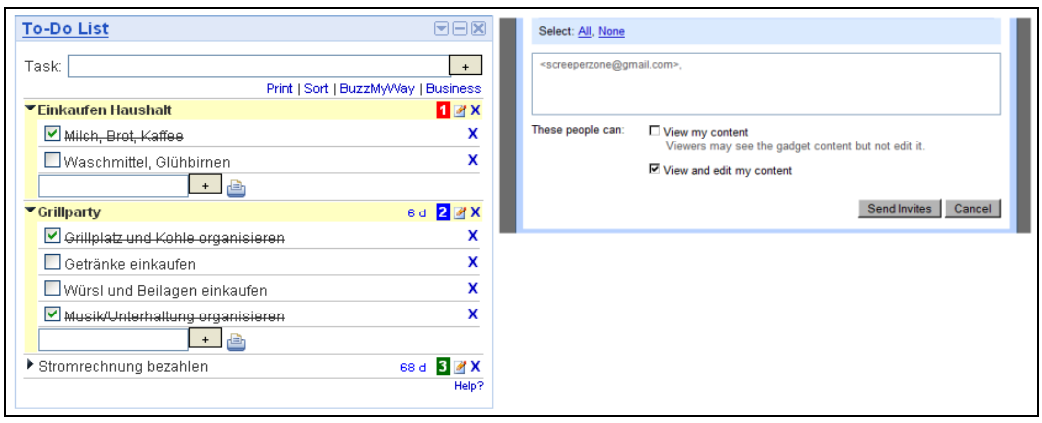


Abbildung 8: Hauptfenster von iGoogle To-Do Liste sowie Detailfenster für die Freigabe (www.igoogle.com)

Im Vergleich zu den zuvor betrachteten Lösungen zeichnen sich diese Aufgabenmanager vor allem durch die Möglichkeit der gemeinsamen Verwendung und Bearbeitung von Aufgaben mit anderen Personen aus. Eine mobile Anbindung für den Einsatz im Alltag wäre hier noch vorteilhaft, damit auch von unterwegs auf die Daten zugegriffen werden kann und die Daten immer auf dem aktuellen Stand sind.

In diesem Kapitel wurden Forschungsergebnisse und daraus entstandene Prototypen sowie kommerzielle Produkte von Systemen betrachtet, die sich für den Einsatz als Aufgabenmanager eignen. Hierzu wurden sie nach ihren Anwendungsgebieten gruppiert und miteinander verglichen.

Zusammenfassend kann festgehalten werden, dass begleitend zu den akademischen Forschungsarbeiten zahlreiche Prototypen und Systeme entwickelt wurden mit dem Ziel, Informations- und Aufgabenmanagement im privaten sowie beruflichen Alltag anzubieten. Jedoch konnte sich noch kein System etablieren. Bereits die Studie von Belotti (2004) zeigte, dass Personen durchschnittlich 11 verschiedene PIM-Tools verwenden. Den Systemen fehlt es teilweise an Funktionalität und bequemer Handhabung. Die Literatur sieht die Gründe eben auch in den schier unendlichen Variationen von Personal Information Management, das von der Umgebung, persönlichen Vorlieben, Kontext und dem Zugang zu Informationstechnologien abhängig ist. Zwar verbindet die Menschen auf höchster Ebene dasselbe Personal Information Management Verhalten, wie zB organisieren, aufzeichnen von Informationen zur späteren Wiederverwendung. Auf operativer Ebene ist Personal Information Management jedoch einzigartig und spezifisch. Auch die Bewertung von PIM-Tools ist daher schwierig und erfolgt oft Ad-Hoc. Diese Evaluierungen bringen aufgrund des Umfangs und der Komplexität der Thematik jedoch oft nicht die gewünschten Ergebnisse. Zuverlässige und gültige Metriken zur Bewertung von PIM Verhalten und Tools gibt bis dato nicht. (Kelly 2006, Elsweiler & Ruthven 2007)

An dieser Stelle ist hervorzuheben, dass sich nur sehr wenige Werkzeuge für den Einsatz im privaten Alltag eignen. Da die Anwendungen größtenteils für die Arbeit im betrieblichen Umfeld ausgelegt sind, sind diese oft zu komplex und aufwendig in der Bedienung, um auch kleine Aufgaben im privaten Leben organisieren oder private Informationen im System festhalten zu können.

Die Anwendungen sind meist auf spezielle Bedürfnisse zugeschnitten und nur bedingt für eine umfassende Unterstützung des Menschen geeignet. Beispielsweise sind Task-List-Manager sehr gut zur Abarbeitung von Projekten geeignet. Die Archivierung und Organisation von sonstigen Informationen (Adressen, etc.) wird jedoch nur unzureichend unterstützt. Um bei der Entwicklung des PIM-Systems möglichst genau auf die Ansprüche der Menschen eingehen zu können, sollen nun die psychologischen Aspekte des menschlichen Gedächtnisses genauer betrachtet werden.

2. Psychologische Hintergründe

Wie Lamming et al. in ihrem Report aus dem Jahre 1993 ausführen, ist es wichtig, die Arbeitsweise des menschlichen Gedächtnisses zu verstehen, um eine nützliche und effektive Anwendung zur Unterstützung von Aufgaben, Erinnerungen und Arbeitsabläufen entwickeln zu können. Daher ist es erforderlich festzustellen, welche Probleme bei der Bewältigung dieser Tätigkeiten auftreten können und wie sich diese in unserem Alltag auswirken bzw. bemerkbar machen.

Von Eldridge, Sellen und Bekerian aus der Forschungsgruppe Rank Xerox Research Center Cambridge wurden 1992 zwei Studien durchgeführt, um festzustellen, wie sich Gedächtnisprobleme auf die Arbeit auswirken. Im ersten Schritt wurde dazu mittels einer Tagebuchstudie die Spannweite der Probleme erfasst. Anschließend wurde im Rahmen einer Befragung die Auftrittshäufigkeit der einzelnen Probleme ermittelt.

2.1. Tagebuchstudie

Es soll nun genauer auf die Ergebnisse der Tagebuchstudie von Eldridge, Sellen und Bekerian (1992) eingegangen werden. Dabei werden die auftretenden Gedächtnisprobleme des Menschen analysiert und erläutert.

Die im Rahmen der Studie erfassten Probleme können in drei Gruppen eingeteilt werden: retrospektive Gedächtnisprobleme, prospektive Gedächtnisprobleme und Action Slips.

Als retrospektive Gedächtnisprobleme werden Erinnerungslücken bezeichnet, die mit vergangenen Events oder Informationen aus der Vergangenheit in Zusammenhang stehen. Dazu gehören: das Vergessen, wo etwas abgelegt wurde, das Vergessen eines Namens oder das Vergessen eines Details eine vergangene Tätigkeit betreffend.

Zu den prospektiven Gedächtnisproblemen zählen all jene, wo vergessen wird, eine geplante Aktion auszuführen wie unter anderem einen Brief aufzugeben oder einen wichtigen Anruf zu tätigen.

Probleme in der Gruppe Action Slips werden durch das Kurzzeitgedächtnis verursacht. Man versteht darunter alle Probleme, wo bereits begonnene Tätigkeiten nicht zu Ende geführt werden, nachdem sie aus irgendwelchen Gründen unterbrochen wurden. Dazu zählt zum Beispiel, dass man sich nicht mehr an die Tätigkeit vor dem Anruf des Abteilungsleiters erinnern kann oder dass man vergessen hat, weshalb man einen Raum betreten hat.

In Summe wurden während der Studie 182 Probleme aufgezeigt. Davon konnte etwas weniger als die Hälfte den retrospektiven Problemen zugeordnet werden. Wobei der Testfall, dass sich jemand nicht mehr an den Namen einer kürzlich kennen gelernten Person erinnern konnte, am häufigsten angeführt wurde. Die prospektiven Probleme traten mit knapp 30 % am zweithäufigsten auf. Mit etwas mehr als 20 % bildeten die Action Slips die kleinste Gruppe an Gedächtnisproblemen.

Retrospektives Gedächtnis

Die Kategorie mit den meisten Erinnerungslücken, die retrospektiven Gedächtnisprobleme, wurde bisher auch am häufigsten untersucht. Die bekannteste Untersuchung von Tulving aus dem Jahre 1983 (zitiert in Lamming et al. 1993) unterscheidet das episodische und das semantische Gedächtnis. Das episodische Gedächtnis, auch autobiographisches Gedächtnis genannt, bezeichnet die Erinnerung an rein persönliche Erfahrungen und Fakten. Das semantische Gedächtnis hingegen enthält das persönliche Wissen über die Welt und deren Zusammenhänge, unabhängig von der eigenen Person. Im Rahmen von Personal Information Management ist insbesondere das episodische Gedächtnis von Interesse.

Das episodische Gedächtnis wird durch drei Merkmale definiert. Das erste Merkmal besagt: „the basic unit of information in episodic memory is an episode“ (Lamming et al. 1993). Wobei Tulving unter einer Episode ein Event versteht, das deutlich abgrenzbar, aber dennoch Teil einer größeren Serie von Events ist. So kann das Leben einer Person aus einer Serie von Episoden verstanden werden.

Das zweite Merkmal sagt aus, dass sich der Mensch die genauen Zeitangaben einer Episode nur sehr schlecht merken kann, hingegen temporäre Beziehungen zwischen Episoden für ihn einfach zu merken sind. So fällt einem beispielsweise nicht mehr der genaue Zeitpunkt eines Treffens ein, man kann sich jedoch häufig daran erinnern, dass das Treffen nach einem Arztbesuch und vor dem Urlaubsantritt stattgefunden hat.

Das dritte Merkmal des episodischen Gedächtnisses besagt, dass Teile der Information einer Episode es erleichtern, sich an weitere Teile dieser Episode zu erinnern. So ist es möglich, dass die Namen von TeilnehmerInnen eines Meetings erst wieder aus dem Gedächtnis abrufbar sind, nachdem man an die anderen TeilnehmerInnen oder an den Ort des Treffens erinnert wird.

Bei der Entwicklung von Software gilt es, diese Klassen und ihre Merkmale des menschlichen Gedächtnisses zu berücksichtigen, um eine möglichst optimale Unterstützung des Menschen zu gewährleisten. Zusammenfassend kann festhalten werden, eine Software zur Unterstützung des Gedächtnisses sollte die Information in einer Form repräsentieren, die es dem Anwender erlaubt, seine Episoden aus der Vergangenheit abzurufen sowie deren Beziehungen zueinander und ihre Charakteristiken einzusehen. Wobei es auf der Suche nach einer Episode ausreichen sollte, eines der Merkmale Zeitraum, Beziehung zu anderen Episoden oder bestimmte Attribute der Episode zu kennen, um sie im System zu finden. Mit einer solchen Software ist es nicht erforderlich, sich die Information selbst zu merken; dies wird von dem PIM-System für die Benutzerinnen übernommen.

Nach dieser Studie zum retrospektiven Gedächtnis werden nun Ereignisse, die sich in der Zukunft abspielen, betrachtet und damit auf die zweitgrößte Gruppe der Gedächtnisprobleme eingegangen.

Prospektives Gedächtnis

Die zweithäufigste Kategorie an Gedächtnisproblemen, die aus der Tagebuchstudie von Eldridge, Sellen und Bekerian (1992) hervorging, ist noch nahezu unerforscht. Es handelt sich dabei, wie bereits erwähnt, um das Vergessen von zukünftigen Ereignissen. Die Studie bestätigt, dass es dem Menschen schwer fällt, ohne Helfer wie Listen, Terminkalender oder Benachrichtigungen bestimmte Ereignisse in der Zukunft zum erforderlichen Zeitpunkt in Erinnerung zu rufen. (Lamming et al. 1993)

An das prospektive Gedächtnis werden verschiedene Anforderungen gestellt. Man unterscheidet zwischen

Erinnerungen zu einem bestimmten Zeitpunkt
Erinnerungen in einem Zeitraum
Erinnerungen, die eventabhängig sind

(Lamming et al. 1993)

Ein Beispiel für eine zeitpunktbasierende Tätigkeit ist es, die Schwiegereltern vom Flughafen abzuholen. Eine Tätigkeit, die innerhalb eines vorgegeben Zeitraums erledigt werden muss, ist die Buchung des Urlaubs. Als Beispiel für eine eventbasierende Tätigkeit kann angeführt werden, dass eine Nachricht erst dann weitergegeben werden kann, wenn diese eingetroffen ist.

Es kann daher festgehalten werden, eine Software zur Unterstützung des prospektiven Gedächtnisses sollte sich nicht nur auf die Erinnerung an Termine beschränken. Ebenso sollten Tasks unterstützt werden, die keinen bestimmten Zeitpunkt für eine Ausführung erfordern oder wo der Zeitpunkt noch nicht feststeht und er daher von anderen Events abhängig ist. Nur so kann flexibel auf die Anforderungen der BenutzerInnen eingegangen werden.

Action Slips

Die kleinste Klasse an Problemen, die Eldridge, Sellen und Bekerian im Report aus dem Jahr 1992 beschreiben, sind die Action Slips. Dabei geht der „rote Faden“ während einer Tätigkeit verloren; dies ist vorwiegend auf zwei Ursachen zurückzuführen. Die Erinnerungslücken treten einerseits aufgrund von Unaufmerksamkeit oder durch Unterbrechungen von außen auf. Dies kann zu kleinen Verzögerungen bis hin zum Abbruch der Tätigkeit führen. Zumeist ist allerdings nur ein kleiner Hinweis nötig, damit die Tätigkeit wieder aufgenommen wird. Die zweite Ursache für ein Auftreten von Action Slips kann in der schlechten Planung der Tätigkeit liegen. Diese Probleme führen meist zu erheblichen Verzögerungen.

Auch diese, in den Softwarelösungen oft vernachlässigte Klasse an Problemen, erfordert eine geeignete Unterstützung. Eine zweite, von Eldridge, Sellen und Bekerian (1992) unabhängige, Tagebuchstudie von Czerwinski, Horvitz und Wilhite aus dem Jahre 2004 wurde durchgeführt. Dabei wurde die Häufigkeit untersucht, mit der Unterbrechungen von Tätigkeiten auftraten bzw. mit der soeben noch durchgeführte Tätigkeiten gewechselt wurden.

Für diesen Zweck protokollierten 11 Testpersonen vier Tage lang ihre Tätigkeiten. Die Auswertungen der Studie führten zu dem Ergebnis, dass die Aktivitäten in 70 % der Fälle unterbrochen wurden. Die Störungen erfolgten in 40 % vom Ausführenden selbst, zu 19 % kam es aufgrund der nächsten Tätigkeit auf der „zu erledigen Liste“ zu einem Wechsel, 14 % der Unterbrechungen wurden durch Telefonanrufe verursacht, 10 % durch Meetings oder Termine ausgelöst und an letzter Stelle wurden 3 % durch Deadlines oder Notfälle hervorgerufen. Als Auswirkung der Unterbrechungen bzw. Tätigkeitswechsel zeigte sich, dass die Wiederaufnahme von kritischen oder umfangreichen Tätigkeiten die doppelte Zeit erfordert als der erneute Einstieg in alltägliche Aktivitäten. In Summe nehmen Unterbrechungen und Tätigkeitswechsel daher ca. 11 % der Arbeitszeit ein.

Die Tagebuchstudie von Eldridge, Sellen und Bekerian (1992) war bei der Klassifizierung der Erinnerungslücken des menschlichen Gedächtnisses und für eine Festlegung ihrer Charakteristiken sehr erfolgreich. Für Rückschlüsse auf die Häufigkeit von Gedächtnislücken sind sie weniger geeignet, da die Aufzeichnungen wieder im Einflussbereich des retrospektiven Gedächtnisses liegen. Grund dafür ist, dass Probleme nicht immer unmittelbar aufgezeichnet werden. Das führt dazu, dass Tätigkeiten mit geringerer Brisanz, die man gegenüber Aktivitäten mit hoher Bedeutung schneller vergisst, zu niedrig repräsentiert werden. Dagegen werden Aktivitäten mit hoher Auffälligkeit und mit größerer Auswirkung sehr konsequent aufgezeichnet und sind dadurch im Verhältnis zu häufig vertretet.

2.2. Fragebogenstudie

Die zweite Studie zu Gedächtnislücken während der Arbeit von Eldridge, Sellen und Bekerian (1992) wurde mittels Fragebogen durchgeführt. Ziel der Befragung war die Ermittlung von quantitativen Informationen bezüglich der Häufigkeit und der Gewichtung der verschiedenen Typen von Gedächtnisproblemen, die im Laufe der Tagebuchstudie hervortraten.

Der Fragebogen bestand aus 21 Fragen und konnte in ca 10 - 15 Minuten vollständig ausgefüllt werden. Die Fragen zielten darauf ab, wie häufig ein bestimmtes Problem aufgetreten war und konnten auf einer 5-Stufen-Skala von „täglich“ bis „nie“ bewertet werden. Zusätzlich konnte jede Problemstellung durch Kennzeichnung mit einem Stern als schwerwiegend gewichtet werden. 118 Fragebögen wurden für die Analyse ausgewertet.

Das Ergebnis

Für die Auswertung wurden die Fragen den Problemklassen Retrospektiv, Prospektiv sowie Action Slips zugeordnet und innerhalb jeder Klasse die Auftrittshäufigkeit addiert. Durch die Division mit der Anzahl der Fragen in den Klassen ergeben sich die in Abbildung 9 dargestellten durchschnittlichen Häufigkeiten.

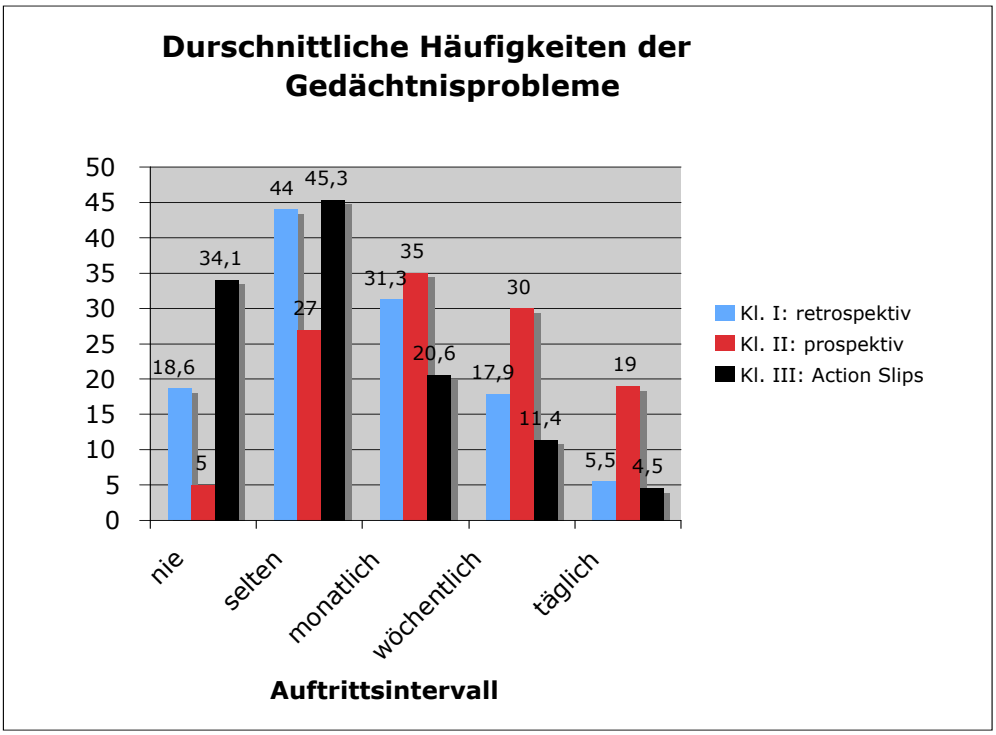


Abbildung 9: Durchschnittliche Häufigkeiten der Gedächtnisprobleme (eigene Auswertung, Quelle: Eldridge, Sellen und Bekerian 1992)

Werden weiterhin die durchschnittlichen Häufigkeiten gewichtet (nie = 0, selten = 1, monatlich = 2, wöchentlich = 3 und täglich = 4), ergeben sich die durchschnittlichen Häufigkeitsraten per Problemklasse aus Tabelle 2. Die durchschnittliche Anzahl der als schwerwiegend eingestuften Gedächtnisprobleme sind in der letzten Spalte ersichtlich.

Klasse der Gedächtnisprobleme	**Durchschnittliche Häufigkeitsrate**	**Durchschnittliche Gewichtung**
I Retrospektiv	1,55	17,6
II Prospektiv	2,27	41
III Action Slips	1,20	11,8

Tabelle 2: Durchschnittliche Häufigkeitsraten und durchschnittliche Anzahl von schweren Problemen in den Problemklassen (eigene Auswertung, Quelle: Eldridge, Sellen und Bekerian 1992)

Erstaunlicherweise ergibt sich durch die Gewichtung der Häufigkeiten, dass die prospektiven Gedächtnisprobleme Spitzenreiter sind. Dennoch treten diese durchschnittlich lediglich mit einer Häufigkeitsrate von 2,27 (ca. ein mal monatlich). Dieses Ergebnis ist darauf zurückzuführen, dass die Testpersonen unterschiedliche Stärken und Schwächen aufweisen, denn immerhin wurde beispielsweise 19 Mal angeführt, dass täglich vergessen wurde, Einträge aus Checklisten abzuarbeiten. Es gibt daher Gruppen von Menschen, die mit einem Problem sehr gut umgehen können, wohingegen andere dort Schwierigkeiten haben. (Eldridge, Sellen und Bekerian 1992)

Auffallendes Ergebnis der Studie war auch, dass die Korrelation zwischen der Häufigkeitsrate der einzelnen Probleme und der Kennzeichnung als schwerwiegend sehr hoch ist. Das heißt, jene Fragestellungen, die als häufig eingestuft wurden, wurden mit hoher Wahrscheinlichkeit auch als schwerwiegend deklariert. Dieser Zusammenhang kann jedoch dadurch erklärt werden, dass sich Menschen von Problemen, die häufig auftreten, beeinflussen lassen und diese daher als wichtig erachten. (Eldridge, Sellen und Bekerian 1992)

Gemäß Lamming et al. (1993) sollte zur Kenntnis genommen werden, dass auch Fragebögen methodischen Problemen unterliegen. Die Schätzungen über die Häufigkeiten und die Gewichtung der Probleme müssen daher als von den Testpersonen „wahrgenommene Werte“ gesehen werden. Es besteht jedoch kein Grund anzunehmen, dass die Fragebogenstudie nicht die relativen Häufigkeiten wiedergibt. Insbesondere sollten die verstärkt wahrgenommenen Probleme an erster Stelle für die Unterstützung durch die Software stehen.

Die Tagebuchstudie war hilfreich zur Erfassung der gesamten Breite an Gedächtnisproblemen, mit denen der Mensch im Alltag konfrontiert wird. Sie konnte jedoch kein genaues, quantitatives Resultat liefern. Zur Ergänzung der Ergebnisse wurde daher eine Befragung über die gefundenen Problemklassen durchgeführt, welche zu teils erstaunlichen Resultaten führte und damit die

Notwendigkeit einer technologischen Unterstützung von Gedächtnisproblemen bestätigt.

Betrachtet man beide Studien gemeinsam und versucht die technologische Unterstützung für eine breite Masse umzusetzen, ist deutlich erkennbar, dass eine Spezialisierung auf einzelne Kategorien von Gedächtnisproblemen nur wenig Sinn macht. So geht aus den Ergebnissen der Befragung hervor, dass die Stärken und Schwächen von AnwenderInnen sehr unterschiedlich sind. Daraus kann geschlossen werden, dass nur eine umfassende Unterstützung aller Problemklassen den gewünschten Erfolg bringt.

3. PIM Anforderungen

Nach der Betrachtung von psychologischen Hintergründen für Gedächtnisprobleme sollen in diesem Abschnitt die nötige Funktionalität sowie mögliche Anwendungsbereiche für ein PIM-System erarbeitet werden.

3.1. Aufgaben- und Informationsmanagement

Wie in dem vorangegangenen Kapitel deutlich gemacht wurde, können Applikationen zur Unterstützung des Menschen verschiedene Bereiche abdecken. Im ersten Schritt soll nun die zeitliche Funktionalität von Gedächtnisstützen bestimmt werden – dh. die Fähigkeit gegenwärtige und zukünftige Aktivitäten zu unterstützen, aber auch Tätigkeiten, die in der Vergangenheit liegen, wieder abrufbar zu machen.

To-Do Listen

Als erstes System werden persönliche Tätigkeitslisten betrachtet. Tätigkeiten, die in näherer Zukunft erledigt werden sollen, werden auch als To-Dos bezeichnet.

Da nicht alle Aufgaben von gleicher Relevanz und Aktualität sind, sollten To-Do-Listen die AnwenderInnen dabei unterstützen, Zeit- und Arbeitspläne zu erstellen. Dazu ist es erforderlich, dass Planzeiträume und Deadlines, dh. der Zeitraum der geplanten Fertigstellung der jeweiligen Aufgabe, erfasst werden. Dies erlaubt, die Tätigkeiten nach ihrer Dringlichkeit zu reihen. Bei der Eingabe von Zeiträumen sollten die BenutzerInnen jedoch nicht zu genauen Datumsangaben gedrängt werden, die später möglicherweise nicht eingehalten werden können. Eine Festlegung des Zeitrahmes oder Zeitbedarfs zB. in Stunden, Tagen, Wochen oder Monaten ist bereits für eine Reihung ausreichend.

Des Weiteren sollte beachtet werden, dass es sich bei den Aufgaben sowohl um einfache Tätigkeiten, wie zB. einen Schadensfall bei der Versicherung melden, als auch um komplexe Projekte, die aus mehreren kleineren Aufgaben zusammengesetzt sind, handeln kann. Beide Fälle sollten vom System unterstützt werden.

Erinnerungen

Ein weiterer Anwendungsbereich eines PIM ist die Verwaltung von Erinnerungen. Es handelt sich dabei ebenfalls um zukünftige Angelegenheiten, die im Vergleich zu den zuvor erwähnten To Dos zu einem exakten Zeitpunkt ausgeführt werden sollen. Wie bereits oben erwähnt, wird gefordert, dass ein Aufgaben Manager zeitkritische Ereignisse und Aufgaben unterstützt. Es sollte daher möglich sein, Einträge mit Terminen zu versehen; die BenutzerInnen werden dann zum gegebenen Zeitpunkt vom System an die Verpflichtung erinnert.

Zur Erhöhung der Benutzerfreundlichkeit des Systems könnte dieses die aktuelle Situation der AnwenderInnen bzw. deren Umfeld berücksichtigen. Wird vom System beispielsweise erkannt, dass eine Tätigkeit gerade eben ausgeführt wird bzw. bereits erledigt ist, könnte die Erinnerung daran vom System unterdrückt werden, um die AnwenderInnen nicht unnötig zu stören. Dies könnte erreicht werden, indem das System erkennt, dass sich die BenutzerInnen am Durchführungsort befinden und daraus schließt, dass die Tätigkeit ausgeübt wird. Die Möglichkeit, den aktuellen Ort anzugeben, erlaubt es den AnwenderInnen zusätzlich, Tätigkeiten und Termine nach diesem Ort zu filtern. Beispielsweise könnte die Aktivierung des Ortes „Wien“ die Erinnerung „Oper besuchen“ aufrufen. Ferner weist der PIM bei verteilten Systemen darauf hin, wenn sich eine Person aus den Kontakten am selben Ort befindet.

Wiederkehrende Aufgaben

Als Nächstes werden wiederkehrende Aufgaben betrachtet, wie beispielsweise das Bezahlen der Stromrechnung. Es handelt sich dabei um Tätigkeiten, die in

regelmäßigen Abständen durchzuführen sind. Sie werden schnell zur Gewohnheit und daher leicht vergessen. Auch solche Aufgaben soll das System unterstützen. Vor allem für kleine Aufgaben, wie das regelmäßige Blumengießen, sind die existierenden Lösungen zu aufwendig. Hier soll das System eine einfache und komfortable Möglichkeit bieten, um die eigene Vergesslichkeit in den Griff zu bekommen.

Wiederkehrende Aufgaben können auch Tätigkeiten sein, die ursprünglich als einmalige Projekte angesehen wurden, die jedoch zeitunabhängig ein weiteres Mal oder wiederholt durchzuführen sind. In solchen Fällen sollte die Möglichkeit bestehen, die bereits abgeschlossenen Projekte zu reaktivieren. Vor allem für komplexe Projekte mit hohem Planungsaufwand wäre dies von Vorteil, da auf bereits bestehendes Wissen zurückgegriffen werden kann. Ein erneutes Planen und Koordinieren der Aufgaben ist nicht mehr nötig – zuvorkommend wird auch vermieden, dass einzelne Arbeitsschritte vergessen werden.

Faktenwissen

Bisher wurden Ereignisse betrachtet, die in der Zukunft liegen. Nun soll darauf eingegangen werden, wie Faktenwissen am effizientesten festgehalten werden kann. Dieser Anwendungsbereich wird von existierenden Systemen, wie oben bereits analysiert wurde, oft vernachlässigt. Dabei kann es sich um private oder berufliche Daten aus der Vergangenheit, wie beispielsweise die abgegebenen Lottozahlen, handeln oder um Informationen mit permanenter Gültigkeit, wie die E-Mail-Adresse eines Geschäftskunden, die Abfahrtszeiten des täglichen Pendlerverkehrs oder die Verwaltung von Passwörtern.

Die Funktionalität eines PIM sollte das Speichern von jeder Information, die den BenutzerInnen wichtig erscheint, einfach ermöglichen. Als Beispiel kann der Titel des soeben im Radio gespielten Songs genannt werden. Auch ein Einsatz als Brainstorming Instrument wäre denkbar, indem das System das Erstellen von Ideensammlungen unterstützt. Zusammenfassend sollte ein PIM

als Notizbuch für Inhalte dienen, die von BenutzerInnen im Alltag aufgegriffen werden und die ihnen wichtig oder wissenswert erscheinen.

Aktuelle Tätigkeiten

Besonderes Augenmerk verlangen Tätigkeiten, in deren Ausübung sich die BenutzerInnen gerade befinden. Gedächtnisprobleme zeigen sich in diesen Fällen dadurch, dass aufgrund einer Unterbrechung vergessen wird, woran gerade gearbeitet wurde oder an welchem Punkt der Tätigkeit die Unterbrechung stattfand. Hier kann ein PIM-System die AnwenderInnen unterstützen, indem eine Funktionalität für den Verlauf von Tätigkeiten angeboten wird. Im Detail handelt es sich hier um ein Monitoring von aktuellen Aufgaben. Dabei könnten die To-Dos mit Attributen wie „Status“ oder „in Bearbeitung“ ausgestattet werden. Nach einer Unterbrechung im Arbeitsfluss könnten die BenutzerInnen alle Aufgaben, die sich in Bearbeitung befinden, herausfiltern, sie werden so möglicherweise an die unterbrochene Tätigkeit erinnert und können diese fortführen.

Dokumentation von Tätigkeiten

Ebenfalls um die aktuelle Tätigkeit, nicht aber um Unterbrechungen, geht es beim nächsten Anwendungsbereich. In besonderen Fällen kann auch eine Dokumentation der aktuellen Tätigkeit erwünscht sein. Zum Beispiel wenn die AnwenderInnen eine kritische Tätigkeit ausführen und die Arbeitsschritte oder ihre Überlegungen zu dieser Tätigkeit festhalten möchten. Hier könnte neben einer Dokumentation von Tätigkeiten in Textform durch Aufzeichnung im Audio- oder Videoformat, vergleichbar mit Diktiergeräten oder Kameras, oder durch das Anhängen von Photos oder Bildmaterial angeboten werden.

Nachdem nun Anwendungsbereiche von PIM-Systemen zur Unterstützung von vergangenen, gegenwärtigen und zukünftigen Aktivitäten behandelt wurden, sollen nun jene Aufgabengebiete ermittelt werden, die aus den Bereichen

Telearbeit bzw. Prozessmanagement stammen, aber auch im Alltag Anwendung finden.

3.2. Daten- und Workflow-Management

Noch nie zuvor waren die Menschen so mobil und dabei mit so leistungsfähiger Computertechnologie wie Laptops, PDAs oder Mobiltelefonen ausgestattet. Dieser Trend wird sich vermutlich auch weiter fortsetzen. Aus diesem Grund sollte ein PIM-System den Datenzugriff unabhängig vom Aufenthaltsort ermöglichen. Für einen PIM impliziert dies, dass die Daten auch über Mobiltelefone oder Internet Browser abrufbar sein sollten.

Funktionalitäten, die von Groupware Systemen angeboten werden, können auch für den Gebrauch in PIM-Systemen von Vorteil sein. So könnte beispielsweise das Anhängen von Dateien bei Einträgen im System unterstützt werden; dies ermöglicht es den AnwenderInnen, die Dateien auch von unterwegs abzurufen und zu bearbeiten. Einer der größten Vorteile von Groupware bzw. Virtuellen Office Systemen, nämlich dass Daten unabhängig vom Aufenthaltsort der AnwenderInnen bearbeitet werden können, könnte für einfache Anwendungsfälle auch in PIM-Systemen umgesetzt werden.

Des Weiteren sollte es den AnwenderInnen ermöglicht werden, kleinere und mittlere Projekte mit dem PIM abzuwickeln. Dabei sollte auf eine einfache und schnelle Handhabung Wert gelegt werden. Das Ziel eines PIM sollte es dennoch nicht sein, ein vollwertiges Workflow System bereitzustellen, vielmehr sollten kleine und mittlere Ad-hoc-Workflows abbildbar sein und abgewickelt werden können. Die Unterscheidung zwischen Workflow und Ad-hoc-Workflow liegt darin, dass der Ablauf von Prozessen in einem Workflow System von vornherein fest definiert wird und das System keine Abweichungen zulässt. Bei einem Ad-hoc-Workflow hingegen sind Änderungen und Erweiterungen der Prozessdefinition während der Abarbeitung der Prozesse möglich. Daher ist ein Ad-hoc-Workflow flexibler und erlaubt es, auf unvorhergesehene Ereignisse zu reagieren. (http://de.wikipedia.org/wiki/Ad-hoc-Workflow)

Für eine reibungslose Ausführung der Workflows sollte das System eine Funktionalität für das Monitoren der Prozesse bieten. Status, Verlauf, erledigte Aufgaben sowie Prozesse in Arbeit sollten abgerufen und definiert werden können. Um auf unvorhergesehene Ereignisse zu reagieren, sind Mechanismen erforderlich, die es ermöglichen, Tätigkeiten neu anzuordnen, einzufügen, zu beseitigen bzw. die Prioritäten neu zu vergeben.

Gemeinsam mit der Funktionalität von Groupware und Workflow Systemen ergibt ein PIM-System ein flexibles Werkzeug zur Abwicklung von kleinen Projekten. Nach der Analyse von PIM-Systemen aus der Sicht von Prozessmanagement soll nun auf den bisher vernachlässigten Aspekt der gemeinsamen Nutzung von Aufgaben eingegangen werden.

3.3. Kooperatives Arbeiten

Die Mobilität der Menschen sowohl im Beruf als auch in der Freizeit bedeutet häufig gemeinsame Aufgaben, Ziele oder Interessen in unterschiedlichsten Kreisen. Die beteiligten Personen können aus Teams von verschiedenen Abteilungen eines Betriebes, aus familiären Beziehungen oder aus dem Freundeskreis in der Freizeit stammen. Um gemeinsame Aufgaben und Interessen zu unterstützen, soll an dieser Stelle das PIM-System um die Multi-User-Funktionalität erweitert werden. Dazu werden im folgenden Abschnitt mögliche Anwendungsszenarien beschrieben. Des Weiteren wird die Applikation für den Gebrauch als Kommunikationsplattform erweitert.

Um das System mit Multi-User-Funktionalität auszustatten, ist es nötig eine nahezu Echtzeitkommunikation zwischen den Parteien zu ermöglichen. Ein einfacher Anwendungsbereich ist, dass Themengebiete im System veröffentlicht und Personen zB. zur Diskussion über Freizeitaktivitäten eingeladen werden. Alle eingeladenen Personen können sich zB. mittels Textnachrichten an den Themen beteiligen, wobei es keine Rolle spielt, wo sich die Person gerade befindet oder zu welchem Zeitpunkt die Nachrichten verfasst werden. Auch

weitere Personen können jederzeit zur Diskussion eingeladen werden. Im weiteren Verlauf der Arbeit werden gemeinsame Einträge im PIM-System auch shared Tasks genannt.

Durch die Verwirklichung der Kommunikation zwischen den BenutzerInnen werden neue Einsatzgebiete für PIM-Systeme geschaffen. Es dient als Mitteilungsplattform für eine Gruppe von Personen. Termine, Aushänge und Mitteilungen werden zentral verwaltet und sind von allen involvierten Personen jederzeit abrufbar.

Insbesondere werden auch alle oben erarbeiten Anwendungsgebiete gemeinschaftlich verwendet. Ein PIM dient beispielsweise als Brainstormingsystem und fördert die Ideenfindung von einer Gruppe von AnwenderInnen, da die TeilnehmerInnen jederzeit und von jedem Ort aus ihre Ideen einbringen können. Die Abarbeitung von Aufgabenlisten erfolgt gemeinsam und für Termine, die in der Gruppe vereinbart werden, fungiert das System zusätzlich als Gruppenkalender. Auch im System gespeichertes Faktenwissen oder andere Informationen sind jederzeit für weitere Personen zugänglich und von diesen ergänzbar.

Projekte sowie Ad-hoc-Workflows können in Gruppen erarbeitet und ausführt werden. Dabei ist kein zusätzlicher Aufwand nötig. Der Status der Tätigkeiten ist jederzeit ersichtlich und auf unvorhergesehene Ereignisse kann umgehend reagiert werden.

Zusammenfassend erlaubt die Nutzung von gemeinsamen Daten den AnwenderInnen eine einfache Möglichkeit, Aufgaben, Ziele oder Tätigkeiten zu organisieren. Dabei wird den AnwendernInnen nicht nur Einsicht in Daten gewährt, es wird auch ermöglicht, diese zu bearbeiten und zu ergänzen. Alle Ergänzungen und Änderungen sind für die beteiligten Personen umgehend und von überall ersichtlich.

Im nächsten Schritt wird die Möglichkeit der Zusammenarbeit mit anderen Applikationen analysiert.

3.4. Kommunikationen mit anderen Applikationen

Applikationen, wie beispielsweise Groupware-Systeme, sind in der Regel mit möglichst umfangreicher Funktionalität ausgestattet und bilden nach außen ein in sich geschlossenes Produkt (Kreifelts, Hinrichs und Woetzel 1993). Es ist allerdings nicht möglich, Kooperation auf ein Produkt zu beschränken. Ein Ziel des Systems sollte es sein, dass AnwenderInnen den Großteil ihrer Aufgaben-, Informations- und Zeitmanagements über den PIM abwickeln können. Es ist daher erforderlich, dass das PIM Schnittstellen zu gängigen Datenmanagement-Tools bietet.

Wie aus den oben vorgestellten Untersuchungen von Bellotti et al. (2004) hervorgeht, wird ein großer Anteil des persönlichen Informationsmanagements per E-Mail abgewickelt. E-Mail-Systeme werden dabei als Zeit-, Aufgaben- und Informations-Management-Werkzeuge verwendet. Die E-Mails werden hier, um zu einem späteren Zeitpunkt darauf aufmerksam zu werden, im Posteingangsordner belassen und/oder nicht als gelesen markiert. Eine effizientere Variante wäre hier, dass einzelne E-Mails per drag and drop in das PIM-System geschoben werden können und dazu automatisiert Einträge generiert werden. Damit können auch Tasks, die per E-Mail eintreffen, im PIM verwaltet werden.

Aufgrund der häufigen Verwendung im Alltag sind Schnittstellen zu gängigen Kalender- oder Adressverwaltungswerkzeugen ebenfalls von besonderem Interesse, damit die dort verwalteten Daten problemlos in das PIM-System integriert werden können.

Neben der Integration von Daten aus Applikationen von Drittherstellern sollte auch ein Datenexport von den PIM-Daten in diese Applikationen möglich sein. Daneben ist es auch unbedingt erforderlich, dass die Daten extern gesichert werden können. Zu diesem Zweck kann zB. eine Exportfunktion im XML Format vorgesehen werden. Die Daten können so in andere Applikationen

importiert, als Sicherungskopie gespeichert oder mittels CSS in eine geeignete Repräsentationsform gebracht werden.

Denkbar ist auch eine Integration des PIM-Systems in bestehende Chat Systeme, wie beispielsweise Skype (www.skype.org). Viele UserInnen verwalten mit diesen Systemen bereits ihre Kontakte. Ein großer Vorteil dieser Applikation ist, dass zB. Adressänderungen etc. unter bestimmten Voraussetzungen automatisch vom Profil des bekannten Kontaktes übernommen werden. Eine Erweiterung der Chat Systeme um die PIM Funktionalität wäre daher eine denkbare Alternative. Das PIM-System profitiert jedenfalls durch verbesserte Kommunikationsmöglichkeiten.

In der Folge sollen Funktionen eines PIM-Systems betrachtet werden, die eine bessere Bedienbarkeit ermöglichen und damit den Komfort des Systems verbessern.

3.5. Spezialfunktionen

Bei dem Prototypen Pepys von Lamming und Newman (1992) wurde bereits vorgesehen, dass das System den Ausführungsort einer Tätigkeit mitspeichert. Die Information, an welchem Ort sich BenutzerInnen befinden, bezieht das Pepys-System von Infrarotsendern an Wänden oder an Gegenständen. Im Modell von Brown (1995) wurde diese Funktion erweitert, sodass Tätigkeiten, die Anwender an einem bestimmten Ort ausgeführt haben, bei Betreten des Ortes im System wieder angezeigt werden. Diese Funktionalität könnte in geänderter Form auch für das PIM-System verwendet werden, indem bei modernen mobilen Geräten die GPS Daten für die Ermittlung der Position herangezogen werden. Bei der Erstellung eines Tasks wird dieser sodann automatisch mit dem aktuellen Aufenthaltsort verknüpft. Es ist jedoch zweckmäßig, dass die BenutzerInnen auch manuell einen Bestimmungsort zuweisen können. Betreten die AnwenderInnen dann ein solches Umfeld, werden alle Aufgaben, die mit diesem Ort verbunden sind, angezeigt.

Wird dieser Gedanke fortgeführt und ist der aktuelle Aufenthaltsort der AnwenderInnen der Applikation für die Kontakte ersichtlich, können diese, ohne ständigen Kontakt zu halten, auf den jeweiligen Aufenthaltsort der anderen reagieren. Als Beispiel soll hier eine Lebensgemeinschaft dienen. Befindet sich eine Person aus der Gemeinschaft in einer Einkaufsstraße und es fehlt im Haushalt an gewissen Dingen, können von der anderen Partei Aufgaben generiert werden, die mit dem Aufenthaltsort der Person in der Einkaufsstraße verknüpft sind. Diese wird auf die neuen Aufgaben aufmerksam gemacht und kann die Besorgungen erledigen.

Um die Eingabe von Daten bei mobilen Geräten zu erleichtern, gibt es mehrere Möglichkeiten. Handschrifterkennung wird bereits bei Handhelds erfolgreich eingesetzt und sollte auch in einem PIM-System für die mobilen Geräte zur schnelleren Eingabe angeboten werden. Weiterhin könnte eine Spracheingabe zur Verfügung gestellt werden, vergleichbar zu dem oben beschriebnen Pepys-System von Lamming und Newman. Im Gegensatz zu diesem System sollte aber keine permanente Aufnahme erfolgen. Die großen Datenmengen sind für die BenutzerInnen zu unübersichtlich und Inhalte können nur noch schwer gefunden können. Es ist daher von Vorteil, wenn die Steuerung der Aufzeichnung durch die AnwenderInnen selbst erfolgt. Im System wird ein Task mit der Audiodatei verknüpft und kann somit jederzeit abgerufen werden. Eine Ergänzung der Daten um Notizen ermöglicht zusätzlich eine verbesserte maschinelle Suche. Dieses Szenario ist auch bei Bild- und Videodokumentationen anwendbar.

Die Langzeitstudie von Bellotti (2004) über die Anwendung von Aufgabenlisten und deren Abarbeitung zeigte, je wichtiger einer Person eine Aufgabe erscheint, desto öfter wird diese irgendwo festgehalten. Um redundante Einträge zu vermeiden, sollte ein PIM-System während der Eingabe eines Tasks die bereits existierenden Einträge auf Ähnlichkeiten prüfen. Das Aufzeigen von Duplikaten ermöglicht damit eine effizientere Organisation der Daten und eine Zeitersparnis bei der Eingabe von Tasks für die AnwenderInnen.

Es ist bereits ersichtlich, mit welchen Mitteln es möglich ist, die Benutzerfreundlichkeit eines PIM-Systems zu verbessern, und damit auch die Akzeptanz des Systems zu erhöhen. Die hier aufgezeigten Features, wie beispielsweise die Verwaltung von Örtlichkeiten sowie die Aufzeichnung per Ton- und Videogerät, stellen mögliche Funktionen für ein PIM-System dar.

3.6. Zielgruppe

Da es sich bei dem im Rahmen dieser Magisterarbeit erstellten PIM um einen Prototypen handelt, wird er in erster Linie für Forschungszwecke entwickelt und vorrangig von Personen für Testzwecke verwendet. Dennoch handelt es sich bei der eigentlichen Zielgruppe des PIM um AnwenderInnen, die nach einer einfachen, elektronischen Lösung für Post-its, Informationsverwaltung und Aufgabenmanager suchen.

Es wird davon ausgegangen, dass es sich bei den AnwenderInnen um Personen aller Bildungsebenen und Berufssparten beliebigen Alters handelt. Es sollte bei der Entwicklung von verteilten Management Werkzeugen besonders dynamische Benutzergruppen bzw. AnwenderInnen mit wechselnden Beziehungen berücksichtigt werden. (Ohumukai & Takeda 2003)

Zum Vervollständigung der Analyse wird in der Folge geklärt, wohin die Entwicklung des PIM-Systems nicht führen sollte bzw. wo die Grenzen des Systems zu finden sind.

3.7. Abgrenzung und Diskussion

Das hier veranschaulichte PIM-System soll vorrangig die Organisation von Aufgaben, die Speicherung von Faktenwissen und das Zeitmanagement von AnwenderInnen vereinheitlichen. In den vorangegangen Kapiteln wurden bereits bestehende Tools mit PIM Funktionalität analysiert und beschrieben. Die Vor- und Nachteile dieser Systeme, sowie Unterschiede zu dem hier entwickelten PIM, werden hier zusammengefasst.

Die Bandbreite von Aufgabenmanagern reicht von einfachen elektronischen Notizzetteln bis zu kollaborativen Task-List-Managern. Diese Tools basieren großteils auf To-Do Listen und erleichtern damit eine zeitgerechte Abarbeitung von Aufgaben. Eine Verwaltung von persönlichen Informationen, wie zB Kontakte oder Örtlichkeiten wird grundsätzlich nicht unterstützt. Diese Informationen können zwar in den To-Do Listen festgehalten werden, das Widerauffinden der Informationen gestaltet sich jedoch durch fehlende Archivierungs-, Such-, Filterfunktionalität schwierig.

Auf E-Mail basierende Systeme haben den Vorteil, dass sie auf etablierten Arbeitsweisen aufbauen. Für die persönliche Informationsverwaltung fehlt gängigen Systemen wie Microsoft Outlook jedoch Funktionalität, insbesondere sind die angebotenen PIM-Tools wie Aufgabenmanager und Notizverwaltung nicht mit dem E-Mail-Client verknüpft. Das vorgestellte, auf E-Mail aufbauende, PIM-System Taskmaster bietet bereits bessere Unterstützung für Aufgabenmanagement. Die Verwaltung von persönlichen Informationen wird jedoch auch hier unzureichend umgesetzt, da alle Informationen einer Aufgabe zugeordnet werden müssen (zB Adressen, Notizen).

Tagebuchgeneratoren eignen sich sehr gut, für das Festhalten von vergangenen Ereignissen. Für Aufgabenmanagement oder Terminverwaltung etc. bieten sie jedoch keine Funktionalität. Auch die Fülle an Daten, die ein Tagesablauf bei Tagebuchsystemen generiert, ist aus dem Gesichtspunkt der benötigten Speicherkapazität nicht praktikabel. Vielmehr sollten die BenutzerInnen entscheiden können, welche Informationen sie für wichtig halten und die Kontrolle besitzen, die Daten im System festzuhalten.

Groupware Tools sind für die Arbeit im Team konzipiert. Als Schnittstelle unterstützen sie die Teilnehmer bei der Koordination einer gemeinsamen Aufgabe. In diesem Bereich bieten sie Basisfunktionalität, wie Dateimanagement, Gruppenkalender, Kommunikationstools etc. Die Funktionen können oft durch die Integration von Einzelapplikationen flexible erweitert werden. Groupware Tools und Personal Information Management

greifen immer mehr ineinander. Ein Hauptgrund dafür ist, dass Daten zunehmend in digitaler Form gespeichert werden und damit einfach geshared werden können. Bei der Entwicklung eines PIM ist jedoch zu berücksichtigen, dass Personal Information Management in erster Linie eine private Tätigkeit ist. Hauptaufgabe ist die Unterstützung bei der Erledigung täglicher Aufgaben und das Festhalten persönlicher Daten. Die Freigabe solcher Informationen für eine Gruppe von Personen ist daher, anders als in beruflichen Teams, delikater. (Erickson 2006)

Workflow-Management-Systeme dienen der Steuerung und Kontrolle von strukturierten Abläufen. Bei der Planung von Prozessen unterstützen sie die BenutzerInnen mit graphischer Modellierung und der Simulation von Abläufen. Für die persönliche Planung kleiner Projekte sind diese Systeme jedoch zu kompliziert und der Planungsprozess zu aufwendig. Weiters sollten die Workflows die PIM BenutzerInnen nicht zu einer starren, vorgegebenen Abarbeitung von Arbeitsschritten zwingen, sondern vielmehr die Richtung weisen und so bei der Bearbeitung unterstützend wirken. Die Umsetzung von Ad-Hoc-Workflows wäre daher denkbar, da sie unstrukturierter sind als prozessorientierte Workflows. Die Möglichkeit den Status und Befinden eines Projektes zu prüfen, könnten insbesondere als Hilfestellung bei Action Slips dienen. Auch die Wiederverwendung von bereits abgebildeten, sich wiederholenden Abläufen steigert die Effizient bei der Planung von Projekten.

Es wurden an dieser Stelle sowohl erforderliche als auch mögliche Aufgabenbereiche für ein PIM-System gesammelt. Dabei handelt es sich um eine Software, die es ermöglicht, persönliche und geschäftliche Information zu organisieren. Das System erlaubt dabei verschiedene Daten in einer Applikation zu verwalten und bietet so einen einfachen Zugang zu den Informationen.

Zusätzlich wurden zahlreiche Features und Anwendungsszenarien erörtert, die ein PIM-System zu einem nützlichen Werkzeug machen. Funktionen wie das Verwalten von Aufgaben oder Erinnerungen, das Speichern von Fakten und das Abwickeln von Projekten ermöglichen einen flexiblen Einsatz des PIM-

Systems. Durch eine gemeinsame Verwendung der Tasks wird das System außerdem zu einem interaktiven Tool zur Organisation und Verwaltung gemeinsamer Informationen und Aufgaben.

Die Anforderungen an ein PIM-System aufgrund menschlicher Verhaltensweisen und Charakteristiken wurden in diesem Kapitel umfassend ermittelt. Neben den erforderlichen und möglichen Eigenschaften zur Unterstützung für Aufgaben- und Informationsmanagement wurden auch psychologische Aspekte von Gedächtnisproblemen analysiert.

4. Technische Anforderungen

Nach der sorgfältigen Betrachtung der möglichen Anwendungsbereiche und Features für ein PIM-System wird in diesem Abschnitt das technische Anforderungsprofil erstellt.

4.1. Zeit und Raum

Ziel ist es den verteilten PIM in der Matrix von R. Johansen (1988) (siehe Abbildung 10), in der CSCW Applikationen nach ihren örtlichen und zeitlichen Anwendungsgebieten eingeteilt werden, in der synchronen/asynchronen und geographisch verteilten Kategorie zu platzieren.

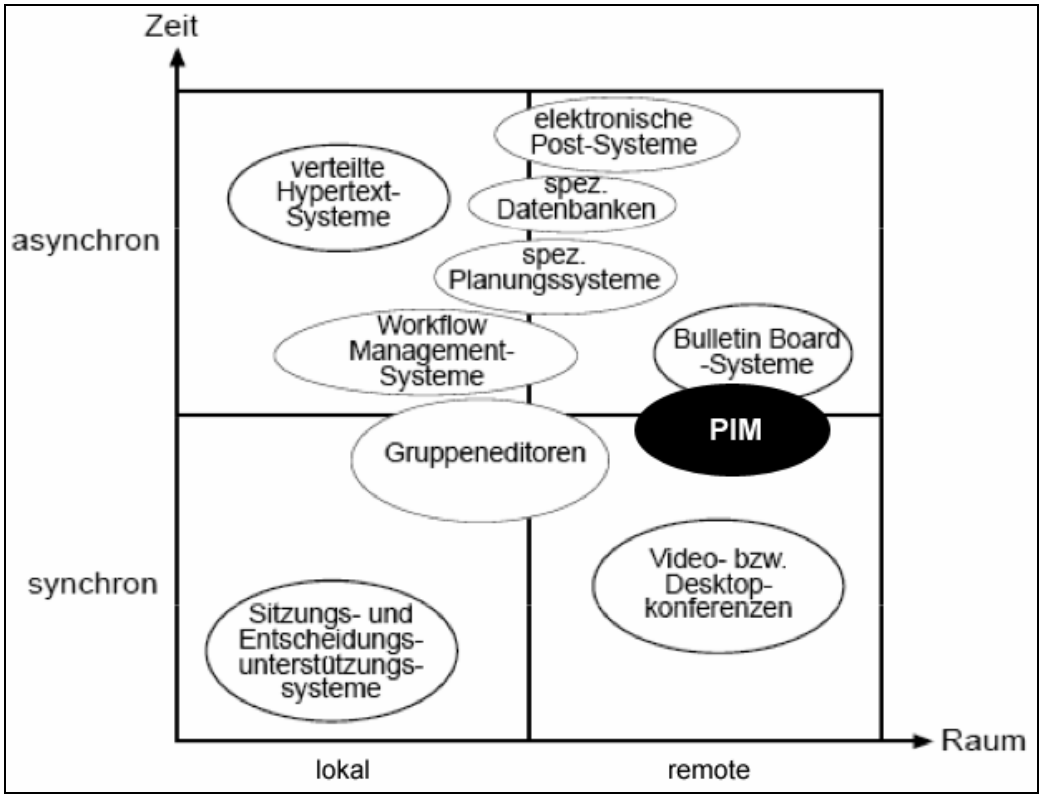


Abbildung 10: Einteilung von Computer Supported Cooperative Work Applikationen (CSCW) nach R. Johansen (basierend auf Teich 2006)

Kooperatives Arbeiten mit einem PIM soll es ermöglichen zur selben Zeit, aber auch zu verschiedenen Zeitpunkten an den Daten im System zu arbeiten. Dabei ist es nicht notwendig, dass man sich am selben Aufenthaltsort befindet.

4.2. Skalierbarkeit

Die Verwendung eines PIM-Systems ist nicht auf eine Gruppe von Personen in einem lokalen Netzwerk beschränkt. Besondere Aufmerksamkeit während der Entwicklung erfordert daher die Auslegung der Software auf eine große Anzahl von BenutzerInnen, eine hohe Anzahl an Tasks sowie eine weite geographische Verteilung.

Um den Betrieb des PIMs für viele räumlich verteilte AnwenderInnen gewährleisten zu können, ist es nötig, dass die Menge der gespeicherten Daten und die Kommunikation zwischen den AnwenderInnen natürlich skaliert. Genauer gesagt, dürfen die benötigten Ressourcen nicht mehr als linear zu den bekannten BenutzerInnen steigen. Um eine Abhängigkeit von zentralen Services zu vermeiden, und eine hohe Verfügbarkeit der Daten zu realisieren, sollten die Daten auf den Systemen der Benutzer gespeichert werden. Die Kommunikation zwischen den AnwenderInnen sollte Peer-To-Peer durchgeführt werden.

4.3. Plattformunabhängigkeit

Der PIM ist ausgelegt für den Einsatz durch Einzelanwender, in privaten Haushalten oder in Organisationen. Darüber hinaus kann der TLM über die Grenzen von Haushalten oder Organisationen hinweg gemeinsam genutzt werden. Die Plattformen der BenutzerInnen bzw. Organisationen werden dabei unterschiedlich sein. Aus diesem Grunde ist es notwendig, die Implementierung plattformunabhängig zu realisieren und damit eine volle Verfügbarkeit des TLM zu gewähren.

4.4. Flexibilität

Um die Menschen bei der Abbildung ihrer Aufgaben und Informationen im System nicht einzuschränken, muss eine flexible Datenstruktur gewählt werden, die leicht zu erweitern ist und die nötigen Freiheiten bei der Anordnung von Tasks lässt. Dazu muss die Möglichkeit bestehen die Task zu gruppieren, unterteilen und klassifizieren.

Der entwickelte PIM sollte den Menschen in möglichst allen Lebenslagen unterstützen. Auf Grund der unterschiedlichen Verhaltensweisen von Personen ist zu erwägen, dass die BenutzerInnen das System an ihre Bedürfnisse anpassen können. Beispielsweise sollte das Hinzufügen von zusätzlichen Verknüpfungen und Attributen zu Tasks angeboten werden.

Die Möglichkeit für mehrsprachige Benutzeroberflächen bzw. Anpassungen an kulturelle Eigenheiten sollte geboten werden, vor allem ist die Portierbarkeit auf neue Geräte vorzusehen. Es ist daher eine Multi-Tier-Architektur zu verwenden, die es erlaubt, das System beispielsweise auf verschiedene mobile Geräte zu portieren.

4.5. Kompatibilität

Die Verwendung des PIM-Systems in Alltag und Beruf erfordert auch eine Interaktion mit anderen Applikationen. Die oben bereits näher beschriebene Funktionalität des Importierens und Exportierens von Tasks, Kontaktdaten etc. eröffnet bereits zahlreiche Möglichkeiten, in Kooperation mit anderen Applikationen zu treten.

Eine Interaktion mit anderen Systemen ist auch gefragt, um Dateianhänge von Tasks direkt aus dem PIM-System öffnen zu können, ohne dass der Kontext zum Task verloren geht. (Kreifelts, Hinrichs und Woetzel 1993)

4.6. Datenpersistenz

Neben dem Schutz vor Datenverlusten bei System- und Netzwerkausfällen muss die Datensicherheit auch während der gemeinsamen Bearbeitung von Tasks gewährleistet sein. Im Einzelnen muss die Konsistenz und Vollständigkeit der Daten sichergestellt sein.

4.7. Sicherheit

Die im System gespeicherten Informationen sind gegen Verlust und, aufgrund des hohen Verteilungsgrades der Software, insbesondere gegen unberechtigte Kenntnisnahme durch Dritte sowie gegen Manipulation zu schützen. Weiterhin ist im System sicherzustellen, dass bei den Exportfunktionen Verschlüsselungsmechanismen angeboten werden, sodass auch hier nur berechtigte Personen auf die Daten zugreifen können.

4.8. Usability

Damit die AnwenderInnen den TLM auch in allen Bereichen und Situationen verwenden, und damit die Flexibilität und Vielseitigkeit des PIMs tatsächlich ausschöpfen, ist der Einstieg in das System einfach zu gestalten. Wie es bereits in anderen Applikationen üblich ist, sollte für den Schnellstart nicht mehr nötig sein, als die Installation auf der Plattform der BenutzerInnen und die Anmeldung im System. Die Handhabung des verteilten PIMs muss auch für AnfängerInnen leicht zu verstehen sein. (Kreifelts, Hinrichs und Woetzel 1993)

4.9. Offline Betrieb

Neben dem Einsatz als persönlicher Assistent kann der PIM auch als Gruppenwerkzeug verwendet werden. Durch den zeitlich und geographisch getrennten Einsatz des PIMs ist es nicht möglich, eine permanente Datenverbindung zwischen den einzelnen Geräten zu gewährleisten. Daher ist die Funktionalität des PIMs auch unabhängig von den Datenverbindungen zu

den beteiligten Personen zu erhalten. Es muss zumindest der Zugriff auf die Daten und das erstellen von neuen Einträgen möglich sein. Der Abgleich der Daten sollte bei wiederhergestellter Kommunikation automatisch vom System durchgeführt werden. Auftretende Konflikte und Inkonsistenzen im Datenbestand werden vom System selbst erkannt und bereinigt.

4.10.Konfliktlösung

Eine manuelle Synchronisierung der Daten durch die AnwenderInnen sollte nicht erforderlich sein. Im Falle von Datenänderungen oder bei der Wiederherstellung der Netzwerkverbindung hat daher das System die Daten automatisch abzugleichen. Eventuell auftretende Konflikte sollten vom System so weit wie möglich selbstständig gelöst werden.

5. Zusammenfassung

Im ersten Teil dieser Arbeit wurde der Status quo an Forschungsarbeiten und technologischen Hilfsmittel erhoben, die den Menschen für die Organisation ihrer täglichen Aufgaben zur Verfügung stehen. Die Reihe an Hilfsmitteln reicht dabei von einfachen elektronischen Notizzetteln bis zu komplexen, automatisierten Tagebuchgeneratoren. Hauptproblem dieser Systeme ist, dass sie nur einen eingeschränkten Anwendungsbereich abdecken und daher für ein optimales Aufgaben-, Informations- und Zeitmanagement viele verschiedene Hilfsmittel erforderlich sind.

Um ein PIM-System möglichst genau auf die Anforderungen der AnwenderInnen abstimmen zu können, wurden zunächst auch die psychologischen Hintergründe für Gedächtnisprobleme betrachtet und mit Hilfe von Studien analysiert. Es wurden drei Problemfelder, die retrospektiven und progressiven Gedächtnislücken sowie Action Slips, erfasst, welche bei der Festlegung der Features eines PIMs zu berücksichtigen sind.

Basierend auf den Ergebnissen aus den ersten beiden Abschnitten wurden in der Folge mögliche Anwendungsgebiete und die erforderliche Funktionalität, sowohl aus Sicht von Einzel- als auch GruppenanwenderInnen, betrachtet. Abgeschlossen wurde die Untersuchung mit den Anforderungen an das PIM-System aus technischer Sicht.

Basierend auf den gewonnenen Erkenntnissen kann festgehalten werden, dass der PIM ein Werkzeug sein soll, das den Menschen möglichst in allen Lebenslagen unterstützt. Da die Stärken und Schwächen von Anwendern sehr unterschiedlich sind, gilt dies auch für die Unterstützung der psychologischen Problemklassen. Dabei soll die Anzahl an derzeit verwendeten Hilfsmitteln so weit wie möglich reduziert werden, damit das PIM-System als zentrales Element für die Verwaltung von Aufgaben, Notizen oder Erinnerungen dienen kann.

Um eine solche umfassende Gedächtnisstütze realisieren zu können, sollte eine einfache Unterstützung bei der Planung von Arbeitsabläufen geboten werden. Dies ermöglicht es den AnwenderInnen, Abläufe zu überwachen und korrigierend einzugreifen. Starre Strukturen wie in Workflow Management Systemen sind jedoch nicht erwünscht.

Um die Aussagekraft von Tasks zu steigern, ist eine Kategorisierung und Priorisierung hilfreich. Insbesondere sollte auch auf flexible Filteroptionen geachtet werden, die unterschiedliche Perspektiven bei unterschiedlichen Arten von Planungen erlauben.

Diese Funktionalität soll auch im Zusammenspiel mit anderen AnwenderInnen angeboten werden. Ein weiteres Ziel des PIM-Systems ist es insbesondere, dass die Kontaktdaten von Freunden und Bekannte im PIM verwaltet werden können und mit diesen die Aufgaben, Arbeitsabläufe oder das gesammelte Wissen in nahezu Echtzeit geteilt werden. Wobei hier auf Skalierbarkeit, Performance und Unabhängigkeit von zentralen Servern zu achten ist.

B. Umsetzung in der Praxis

Im folgenden Abschnitt wird auf die praktische Umsetzung der Applikation eingegangen. Dazu wurde ein Prototyp entwickelt, der es den elektronisch vernetzten AnwenderInnen ermöglicht, Aufgaben zu managen und ihr Wissen festzuhalten. Die Ergebnisse aus obiger Analyse fließen dabei in die Entwicklung ein.

In Kapitel 1 erfolgt eine Auflistung der eingesetzten Technologien für die Entwicklung des Prototyps. Programmiersprache, Verteilungsplattform und verwendete Muster werden erklärt.

In Kapitel 2 werden die verwendeten Datenstrukturen und Konzepte beschrieben, ihre Funktionalität erläutert sowie geltende Einschränkungen ermittelt.

In Kapitel 3 erfolgt eine Präsentation des PIM-Prototyps und dessen Handhabung mit Schwerpunkt auf die Umsetzung der psychologischen Erkenntnisse aus Kapitel A.2. Die Arbeit wird abgeschlossen durch eine Bewertung des Systems in Kapitel 4 entsprechend den Metriker aus Kapitel A.4 sowie einen Ausblick auf Erweiterungen und Optimierungsmöglichkeiten in Kapitel 5.

1. Eingesetzte Technologien

Die für die Entwicklung des PIMs verwendete Programmiersprache und Middleware einschließlich Begründung für den Einsatz, deren Funktionalität und Eigenschaften werden an dieser Stelle beschrieben. Ergänzt wird dies durch die Erläuterung des zum Einsatz kommenden Design Patterns.

1.1. Java

Für die Entwicklung des PIMs kommt die objektorientierte Programmiersprache Java zur Anwendung. Der ausschlaggebende Grund dafür ist die Plattformunabhängigkeit der Sprache. Somit wird ein Einsatz ohne vorherige Anpassung auf verschiedene Betriebssysteme ermöglicht.

Weiterhin erlaubt dies eine schnelle Anpassung des PIM-Systems an mobile Geräte. Der Einsatz von Java Micro Edition (Java ME) gestattet, auf bestehenden Quellcode zurückzugreifen und so Entwicklungszeit zu reduzieren.

Als Entwicklungsumgebung wird die IDE (Integrierte Entwicklungsumgebung) von Netbeans (www.netbeans.org) in der Version 5, verfügbar für Windows, Linux, Mac OSX und Solaris, verwendet. Die Entscheidung fiel auf diese IDE, da sie neben dem bereits integrierten Graphical-User-Interface (GUI) Builder auch Java ME unterstützt und einen GUI Bilder für mobile Geräte anbietet.

1.2. XVSM – MozartSpaces (www.mozartspaces.org)

Um den AnwenderInnen Autonomie von zentralen Servern einzuräumen bzw. keine Einschränkungen in Performance oder Skalierbarkeit von Server-Client Strukturen hinnehmen zu müssen, wird der PIM-Prototyp auf der Middleware Technologie MozartSpaces aufgebaut. Die Tuple/Space basierte Middleware eignet sich sehr gut für Kollaboration. Der Grund dafür ist, dass die Mittelware sowohl Kommunikation (gemeinsame Verwendung und Austausch von Daten),

als auch Synchronisation ohne zusätzliche Mechanismen bietet. (Ducasse, Hofmann und Nierstasz 2000)

MozartSpaces (www.mozartspaces.org) ist eine Open Source Implementierung von XVSM (eXtensible Virtual Shared Memory) (Kühn et al. 2008). Dabei handelt es sich um eine Technologie basierend auf Forschungsarbeiten der Space Based Computing Group des Instituts für Computersprachen an der Technischen Universität Wien (www.complang.tuwien.ac.at/eva). Den EntwicklerInnen wird dabei ein virtueller gemeinsamer Speicherbereich zur Verfügung gestellt und damit eine hohe Abstraktion bei der Entwicklung von verteilten Applikation geboten. Dabei kommt das Konzept des Shared Data Spaces zum Einsatz. Geräte, die das XVSM Service anbieten, werden Peers genannt. Geräte, mit Applikationen die das Service nutzen, werden hier Kommunikationspartner genannt. Alle Kommunikationspartner im Netzwerk verständigen sich ausschließlich indirekt über die im Space gespeicherten Daten und sind dabei unabhängig von Betriebssystem, Programmiersprache oder Kommunikationsstandards. Die Kommunikationspartner können autonom arbeiten, bei Bedarf Anfragen bzw. Antworten in den Space schreiben und werden automatisch über Änderungen informiert. Technisch bedeutet dies, dass alle Kommunikationspartner symmetrischen Zugriff auf die Daten im Space besitzen.

Die Daten werden dabei in XVSM in so genannten Containern gespeichert, wobei jeder Peer ein Set dieser Container verwalten kann. Die Dateneinträge in den Containern werden als Entries bezeichnet. Dabei kann es sich um einen AtomicEntry, dieser ist generisch und kann durch jede Java Klasse instanziiert werden, oder ein Tupel handeln, wobei das Tupel abermals aus einem Set von Tupeln oder AtomicEntrys bestehen kann. In Abbildung 11 erkennt man einen Container, der sowohl mit drei AtomicEntrys als auch mit einem Tupel gefüllt ist.

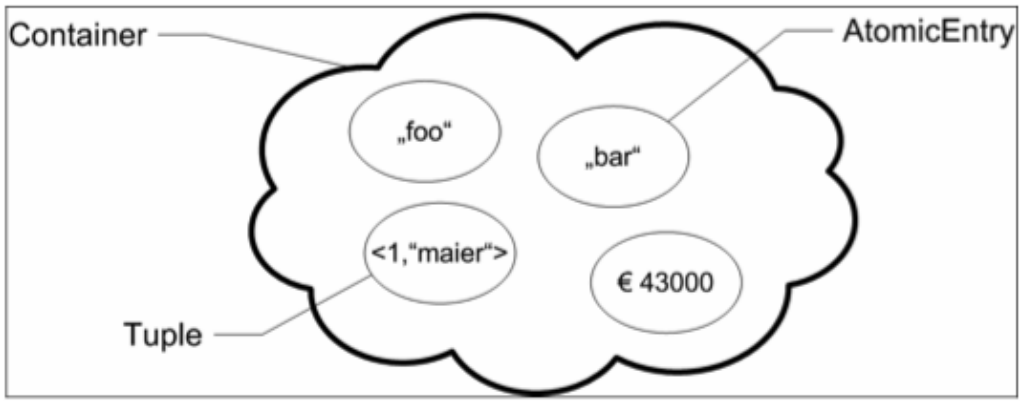


Abbildung 11: Datenstruktur Container, Tuple, AtomicEntry in XVSM (Wittmann 2008)

Für die Bearbeitung von Entries stehen folgende Funktionen zur Verfügung:

write – schreibt Entries in den Container und kann blockieren, wenn kein Platzt vorhanden ist.

shift – schreibt Entries in den Container ohne zu blockieren

read – liest Entries im Container

take – liest und entfernt Entries aus dem Container

destroy – entfernt Entries aus dem Container

Die Kommunikation zwischen den verteilten Applikationen erfolgt über die im Container gespeicherten Entries. So können unter anderem Anfragen an Kommunikationspartner im Netzwerk gestellt werden und auch deren Antwort in Form von Entries empfangen werden. In Abbildung 12 ist dargestellt, wie diese Kommunikation aussehen kann. Die Partner, die die Anfragen in den Space stellen, werden auch Consumer genannt. Die Producer, jene Applikationen, die Anfragen bearbeiten, können daraufhin die Anfragen aus dem Space lesen.

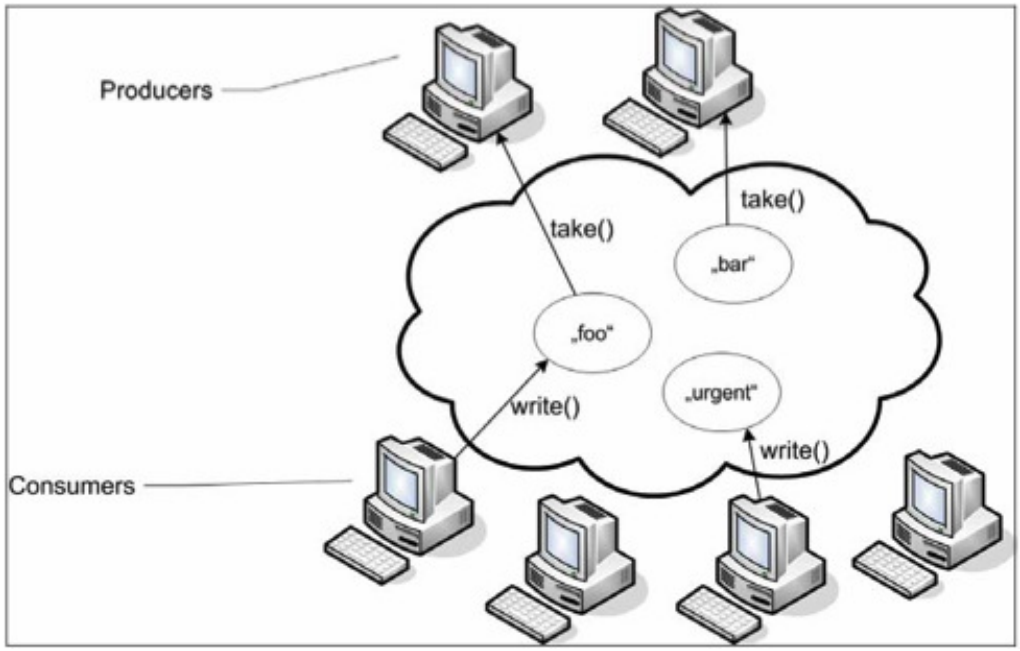


Abbildung 12: Kommunikation zwischen verteilten Applikationen über Entries im Space (Wittmann 2008)

Nachdem die Anfragen bearbeitet wurden, können die Producer die Antwort wieder über Entries dem Consumer mitteilen. Wie in der Abbildung ersichtlich, können mehrere Consumer und Producer zur selben Zeit im Space agieren.

Für die Koordination der Entries in den Containern stehen verschiedene Containertypen zur Verfügung. Dies wird mittels Koordinatoren ermöglicht, die den Containern zugewiesen werden. Die wichtigsten Koordinationstypen sind:

- FifoCoordinator – Fifo steht für „first-in-first-out", dh. die Entries werden in der Reihenfolge retourniert, in der sie geschrieben wurden.
- LifoCoordinator – Lifo steht für „last-in-first-out", dh. der letzte Entry, der in den Container geschrieben wurde, wird als Erstes gelesen.
- KeyCoordinator – Hier kann ein Schlüssl definiert werden, der gemeinsam mit dem Entry gespeichert wird. Will man den Entry lesen, wird mittels des Schlüssels der richtige Entry retourniert.
- VectorCoordinator – Die Adressierung der Entries erfolgt hier mittels des Index.

Die Verteilung der Daten zwischen den beteiligten Peers wird für den Programmierer transparent von XVSM abgewickelt. Die Verteilung wird auch als Replikation bezeichnet und in verschiedenen Strategien zur Verfügung gestellt. Diese Strategien reichen von einer unmittelbaren Verteilung auf alle Peers, bis zu einer Verteilung bei ausschließlicher Benötigung der Daten. Zurzeit befinden sich die Replikationsmechanismen für XVSM noch in der Entwicklung (Stand Mai 2008). Ein möglicher Replikationsmechanismus, der XVSM als Baustein erweitern kann, ist das PR_deep (Passive replication with a deep object tree) Verteilungs-Protokoll (Kühn 1994). Das Verteilungs-Protokoll wird für die Auslegung des Prototyps herangezogen.

Verteilungs-Protokoll PR_deep

Bei dem Replikationsmechanismus PR_deep bilden die Kommunikationsteilnehmer für jedes im Space gespeicherte Objekt einen Baum mit Replikationen des Objektes. Die Wurzel des Baumes wird als Primary Copy bezeichnet und nur der Peer, der diese besitzt, hat Schreibrechte auf das Objekt. Änderungen der Primary Copy werden für die EntwicklerInnen transparent an die Blätter, dh. Replikate der Primary Copy, verbreitet. Soll eine Änderung des Objektes durch einen Peer erfolgen, der nicht im Besitz der Primary Copy ist, muss diese zuerst an den schreibenden Peer übermittelt werden und kann erst daraufhin beschrieben werden. Dies wird als Migration der Primary Copy bezeichnet. Sie kann nur dann stattfinden, wenn alle Peers im Objektbaum, vom Wurzelobjekt bis hin zum entsprechenden Blattobjekt, online sind. Peers befinden sich im Online-Status, wenn die Kommunikation zwischen den Peers möglich ist. (Kühn 1994).

Das Verteilungs-Protokoll hat großen Einfluss auf die Offline-Fähigkeit der Applikation. Während der Besitzer der Primary Copy auch offline, also getrennt von den anderen Peers des Spaces, seine Schreibrechte behält, besteht für die Peers, die nur ein Replikat der Primary Copy besitzen, nach der Trennung vom Wurzelelement keine Möglichkeit mehr, das Objekt zu manipulieren.

Neben den beschriebenen Mechanismen wird den EntwicklerInnen folgende Funktionalität von XVSM zur Verfügung gestellt:

- Notifications
- Transaktionskontrolle
- Persistenz der Daten (noch in Entwicklung; Stand Mai 2008)
- Wiederherstellung der Daten bei Systemausfall (noch in Entwicklung; Stand Mai 2008)
- Integration von mobilen Geräten (noch in Entwicklung; Stand Mai 2008)

Request/Answer Pattern

Für die Softwareentwicklung unter Verwendung des Virtual Shared Memory Konzeptes existieren bereits zahlreiche Coordination Design Patterns. Das für die Implementierung des PIMs verwendete Muster ist das Request/Answer Pattern.

Das Pattern wird in der Applikation so verwendet, dass das Request/Answer Objekt als ein Vector geführt wird. Damit können beliebig viel Anfragen bzw. Mitteilungen gespeichert werden und die Abarbeitung dieser kann unmittelbar oder, wenn der Empfänger offline ist, zu einem späteren Zeitpunkt erfolgen. (Wernhart 1998)

Die Verwendung des Virtual Shared Memory Paradigmas begünstigt vor allem die Entwicklung von verteilten Applikation, bei denen die Anzahl und Art der Kommunikationsteilnehmer nicht bestimmt und variabel ist. Peers und Kommunikationsteilnehmer können dynamisch zum Service hinzugefügt bzw. entfernt werden. Die Kommunikation zwischen den Peers erfolgt Peer-To-Peer, wodurch Engstellen vermieden werden. Durch den indirekten Informationsaustausch, ermöglicht XVSM betriebssystemunabhängigen Zugriff auf die Daten im Space.

Im folgenden Abschnitt werden die entwickelte Datenstruktur, die im Shared Space von XVSM gebildet wird, und die Kommunikation zwischen den Applikationen der Kommunikationspartner beschrieben.

2. System Design und Implementierung

Durch den Einsatz von XVSM als Middleware wird die Entwicklung von parallelen bzw. verteilten Applikationen stark vereinfacht. Da XVSM Technologiekonzepte, wie Peer-To-Peer, für die EntwicklerInnen transparent unterstützt, wird darauf Wert gelegt, auch für den PIM eine dezentrale Struktur zu schaffen.

Daher wurde eine Datenstruktur entworfen, die es ermöglicht, ohne zentralen Server ein Netzwerk von Kommunikationsteilnehmern aufzubauen und Datenaustausch zwischen den Mitgliedern durchzuführen. In der Folge wird davon ausgegangen, dass die Möglichkeit für ein Zugriff der PIM-Applikation auf einen XVSM Peer besteht. Befindet sich auf dem Peer noch keine für das PIM-Netzwerk nötige Datenstruktur, wird diese automatisch von der Applikation angelegt.

Die zwei wichtigsten Einstiegspunkte in die Datenstruktur sind die Registry und der User, diese werden nun als Erstes beschrieben. Im Anschluss wird auf die Elemente TaskGroup und Task näher eingegangen.

2.1. Registry

Ein Einstiegspunkt für alle PIM-Applikationen in die Datenstruktur des PIM-Netzwerks ist die Registry. Es handelt sich dabei um einen Entry im Root-Container des XVSM Peers. Dieser wird pro Peer genau einmal während des Registrierens des/der ersten Benutzers/in automatisiert angelegt. Wichtigste Aufgabe der Registry ist es, die verbundenen bzw. bekannten Peers und die registrierten BenutzerInnen zu verwalten. Weiterhin dient die Registry als zentrale Nachrichtenstelle für die BenutzerInnen. Die in der Registry enthaltenen Daten sind in Tabelle 3 aufgelistet.

Name	Typ
Id	String
Peers	Vector<XVSMPeer>
RegiststeredUsers	Vector<PublicUser>
Inboxes	Vector<Inbox>

Tabelle 3: Struktur der Registry

Die Registry und im Folgenden auch alle Datenobjekten beinhalten eine ID die zur Identifizierung dient. Weiters beinhaltet die Registry einen Vector, der alle bekannten XVSM Peers speichert. Für jeden gespeicherten Peer wird dabei die URL und die Onlineinformation des Peers festgehalten. Tabelle 4 zeigt die Daten des Peer Objektes. XVSM Peers können beliebig ergänzt werden. Jene gespeicherten XVSM Peers, die online sind, bilden den virtuellen shared Memory für den Datenaustausch.

Name	Typ
Online	Integer
Host	String

Tabelle 4: Struktur des XVSMPeers

Die Registry beinhaltet weiterhin einen Vector, der für jede/n registrierte/n Benutzer/in einen Verweis (vgl. dazu Abschnitt B.2.2 User) auf diese/n enthält. Erfolgt ein Zusammenschluss von Peers, dh. sie wissen von der Existenz des jeweils anderen, werden die in den Registries gespeicherten BenutzerInnen untereinander mitgeteilt. Diese werden mit der bereits bestehenden Liste abgeglichen und noch nicht bekannte BenutzerInnen werden in die Registrierung eingefügt.

Zusätzlich ist ein Vector enthalten, der für alle registrierten BenutzerInnen eine Inbox verwaltet. Sie dient als Eingangsstelle für Nachrichten und spielt eine zentrale Rolle bei der Kommunikation mit anderen BenutzerInnen. Darauf wird später noch genauer eingegangen. Die in der Inbox enthalten Daten sind die Benutzeridentifikation, der Peer, auf dem der/die Benutzer/in registriert ist, und

ein Vector für die Nachrichten an den/die Benutzer/in. Beim Peer handelt es sich um die Adresse des Gerätes, welches das XVSM Service anbietet (vgl. Abschnitt B.1.2 XVSM-MozartSpaces). Die Benutzer/in ist der/die tatsächliche Anwender/in der PIM Applikation. In Tabelle 5 sind die Daten der Inbox zusammengefasst.

Name	**Typ**
Benutzername	String
Peer	String
Messages	VECTOR<Message>

Tabelle 5: Struktur der Inbox

Die hier beschriebene Datenstruktur der Registry im Space ist in Abbildung 13 dargestellt. Die Registry, für jeden auffindbar im Space gespeichert, verweist auf die registrierten BenutzerInnen (PublicUser), auf die Inboxen der BenutzerInnen mit den darin enthaltenen Nachrichten (Message) und die bekannten XVSMPeers.

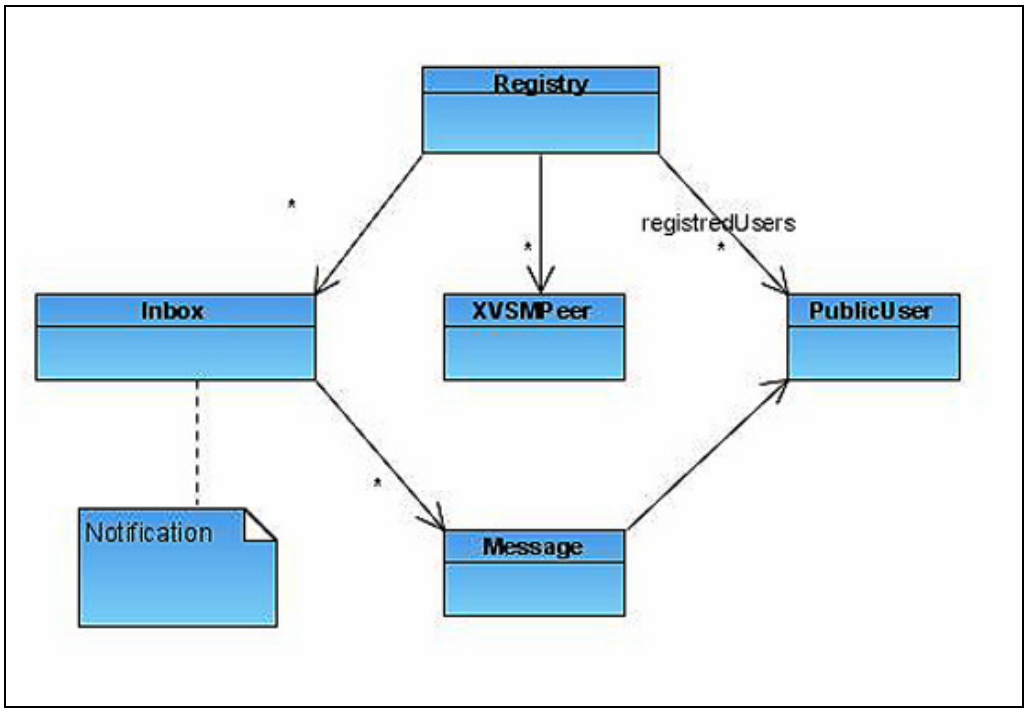


Abbildung 13: Klassenstruktur der Registry

Registrierung

Beim Anlegen eines neuen Users erfolgt im ersten Schritt die Überprüfung der Eindeutigkeit der Identifikation mit den bereits in der Registry eingetragenen BenutzerInnen. Ist der Abgleich erfolgreich, wird ein User Objekt in der Datenstruktur angelegt (vgl. dazu Abschnitt B.2.2 User). Weiters wird im Vektor der registrierten BenutzerInnen ein Verweis auf den neuen User gespeichert und eine Inbox im dafür vorgesehenen Vector der Registry erstellt.

Wie bereits erwähnt, wird pro Peer der im Space bekannt ist eine eigene Registry geführt, in der die BenutzerInnen gespeichert werden und eine Inbox zur Verfügung steht. Dabei kommt das bereits beschriebene Request/Answer Pattern (vgl. Abschnitt B.1.2 XVSM – MozartSpaces) zum Einsatz. Um auf eingehende Nachrichten aufmerksam gemacht zu werden, erstellt die Applikation der AnwenderInnen eine Notification, mit der die bekannten Inboxen der BenutzerInnen überwacht werden. Dazu werden die Inboxen des eigenen und der bekannten Peers in die Notification aufgenommen. Im laufenden Betrieb wird die Notification dynamisch an den Onlinezustand (dh. die Zugriffsmöglichkeit) (vgl. Abschnitt B.1.2 XVSM-MozartSpaces) der Peers angepasst.

Die Begründung für die autonomen Inboxen pro Peer liegt in der Replikationsstrategie von XVSM (vgl. dazu Abschnitt B.1.2 XVSM – MozartSpaces). Das Konzept besteht darin, dass Nachrichten an BenutzerInnen in deren Inboxen beim Senderpeer hinterlegt werden. Dadurch ist gewährleistet, dass die Schreibrechte für Nachrichten zu keiner Zeit verloren gehen, auch wenn der Peer der EmpfängerInnen offline ist. Beim Konsumieren der Nachrichten durch die EmpfängerInnen werden diese als gelesen markiert. Es erfolgt somit keine Migration des gesamten Nachrichtenvektors, sondern lediglich des geänderten Attributs der gelesenen Mitteilung. Die markierten Nachrichten werden erst beim nächsten Schreibvorgang eines Senders entfernt. Wird die Kommunikation zwischen zwei Peers verhindert, können die BenutzerInnen weiterhin auch Nachrichten an BenutzerInnen des jeweils getrennten Peers richten, da diese in der lokalen Registry hinterlegt werden. Der

Empfang der Nachrichten wird hingegen verzögert und erfolgt erst, wenn wieder eine Kommunikation zwischen den Peers stattfindet. Die Notification auf die Inboxen der bekannten Peers wird dabei dynamisch an die Online- bzw. Offline-Situation der Peers angepasst.

Einschränkungen im Registry-Konzept

Aufgrund der Unabhängigkeit der Peers ist es möglich und auch gewünscht, dass sich mehrere Virtuelle Shared Spaces unabhängig voneinander bilden. Bei der Registrierung von BenutzerInnen ist es daher möglich, dass die selben Benutzernamen in verschiedenen Spaces verwendet werden. Verknüfen sich die Spaces daraufhin, treten Probleme mit der Eindeutigkeit der Benutzernamen auf. Dem kann mit verschiedenen Strategien entgegengewirkt werden. Beispielsweise indem als Benutzerkennung gültige E-Mail Adressen verlangt werden, wodurch die doppelten Einträge stark eingeschränkt, aber noch nicht ausgeschlossen sind. Weiterhin kann das mehrfache Vorkommen von Benutzerkennungen durch Zusätze wie URL des Peers oder Zeitstempel ausgeschlossen werden. Die Benutzerkennung wird dann beispielsweise durch das Paar Benutzername/Peer gebildet. Im Vergleich zu Groove beispielsweise, wo die Benutzerverwaltung über einen zentralen Server realisiert wird, hat das beschriebene Konzept den Vorteil, dass es ohne zentrale Services auskommt und ein echtes Peer-To-Peer System darstellt.

Nach dieser Betrachtung der Registry aus der Datenstruktur im Shared Space und dem Registriervorgang wird nun das Konzept des Users erläutert, mit besonderem Augenmerk auf das Auffinden von Kontakten und die Anmeldung über andere Peers im Space.

2.2. User

Das zweite zentrale Element der Datenstruktur für den verteilten PIM ist der User. Im Unterscheid zur Registry dürfen auf diesen nur die BenutzerInnen, die ihn angelegt haben, Zugriff besitzen. Allerdings enthält der User Daten, die eine

öffentliche Sichtbarkeit im Space erfordern. Aus diesem Grund wurde der User in zwei Teile zerlegt. Einen privaten Teil, der PrivateUser genannt wird, und einen öffentlichen Teil, der PublicUser genannt wird. Diese Trennung von privaten und öffentlichen Teilen von Elementen setzt sich auch in untergeordneten Elementen mit sensiblen Daten fort.

Das Konzept hinter den zweigeteilten Elementen besteht darin, dass Einträge im Space nur aufgefunden werden können, wenn man im Besitz einer Referenz auf den Eintrag ist. Existiert beispielsweise im privaten Teil eines Eintrages ein Verweis auf den öffentlichen Teil, ist der öffentliche Teil vom privaten Teil aus auffindbar. Hingegen ist der private Teil für jemanden, der nur in Besitz des öffentlichen Teils ist, nicht sichtbar. Es kann daher ausgehend von einem Objekt, das zugriffgeschützt ist, hier der PrivateUser, eine private Struktur erstellt werden, auf die nur der Berechtigte Zugriff besitzt. Dabei ist es möglich, von der privaten Struktur aus Verweise zur öffentlichen Struktur zu erstellen. Es ist jedoch darauf zu achten, dass keine Verweise von der öffentlichen Struktur auf die private Struktur existieren. Durch diese Trennung der Strukturen und die rein eindirektionale Verknüpfung kann den BenutzerInnen die nötige Sicherheit vor unberechtigten Zugriffen auf die Daten geboten werden.

Der PrivateUser wird wie die Registry als Entry im Root-Container von XVSM gespeichert, wobei pro registriertem Benutzer ein eigener Eintrag erfolgt. Der Zugriffsschutz auf das Element wird von XVSM zur Verfügung gestellt[2] und daher hier nicht behandelt. Der PrivateUser enthält alle Daten, die ausschließlich für den/die Benutzer/in bestimmt sind, der die Zugriffsdaten für den Account besitzt. In Tabelle 6 sind die Daten der PrivateUser ersichtlich.

[2] Derzeit noch nicht implementiert (Stand Mai 2008), kann aber via Aspekte realisiert werden.

Name	Typ
Id	String
PublicUser	PublicUser
AllUserGroup	UserGroup
AllPrivateLocation	PrivateLocation
Settings	Settings
AllPrivateTaskGroup	privateTaskGroup
Passwort	String
AktTask	PrivateTask
AktLocation	Location
AllowInfoTo	Vector<UserInfos>
GotInfoFrom	Vector<UserInfos>

Tabelle 6: Struktur von PrivateUser

Unter anderem befindet sich im PrivateUser auch ein Verweis auf den PublicUser, in dem alle Attribute der BenutzerInnen, die für die Öffentlichkeit zugänglich sind, gespeichert werden. Es handelt sich dabei um Eigenschaften, die von BenutzerInnen eingetragen werden können, wie beispielsweise der volle Name, Geschlecht, Geburtsdatum usw. Die detaillierten Daten sind in Tabelle 7 zu sehen.

Name	Typ
Id	String
Nick	String
FullName	String
Gender	Integer
Birthday	Long
Language	String
E-Mail	String

Tabelle 7: Struktur von PublicUser

Dieses PublicUser Objekt wird in der Datenstruktur verwendet, um auf die BenutzerInnen zu verweisen, sei es im Zusammenhang mit der Registrierung

(vgl. dazu Abschnitt B.2.1 Registry) oder zur Identifizierung der EigentümerInnen von Tasks (vgl. dazu Abschnitt B.2.4 Task).

Unter den Daten des PrivateUser Objektes befindet sich weiterhin der Verweis zu der AllUserGroup. Bei der UserGroup handelt es sich um ein Objekt, das ausgehend von der Wurzel sowohl baumartig erneut UserGroups als Unterobjekte beinhalten kann als auch PublicUser Objekte. Dies ermöglicht es den BenutzerInnen Kontaktgruppen zu verwalten. Tabelle 8 stellt die Daten der UserGroup dar. Bei der UserGroup handelt es sich um rein private Daten der BenutzerInnen.

Name	**Typ**
Id	String
Bezeichnung	String
Colour	Integer
SubUserGroups	Vector<UserGroup>
PublicUsers	Vector<PublicUsers>

Tabelle 8: Struktur von UserGroup

Vergleichbar zur UserGroup, allerdings mit öffentlichem Teil, befindet sich auch das AllLocation Objekt in den Daten des PrivateUsers. Dieses ist ebenso baumartig aufgebaut und kann andere Locations enthalten. Den BenutzerInnen wird dadurch die Verwaltung von Örtlichkeiten, an denen Aktionen stattfinden, ermöglicht. Diese können den Tasks zugeordnet werden. Aufenthaltsorte bzw. Örtlichkeiten könnten damit bei der Filterung von Tasks berücksichtigt werden. Die Datenstruktur der PrivateLocation ist in Tabelle 9 ersichtlich.

Name	**Typ**
Id	String
SubLocations	Vector<PrivateLocation>
PublicLocation	PublicLocation

Tabelle 9: Struktur von PrivateLocation

Auf das in der PrivateLocation verwiesene PublicLocation Objekt haben all jene BenutzerInnen Einsicht, die Zugriff auf einen verteilten Task besitzen. Die Datenstruktur ist in Tabelle 10 ersichtlich. Wird ein Task mit mehreren BenutzerInnen gemeinsam verwendet, erfolgt eine Integration der PublicLocation in die Location-Struktur der involvierten BenutzerInnen. Dazu wird pro Benutzer/in jeweils ein eigenes PrivateLocation Objekt erstellt und in diesem auf das PublicLocation Objekt verweisen. Dadurch wird ermöglicht, dass BenutzerInnen, trotz der gemeinsamen Sicht auf das Objekt, die Location nach seinen Vorlieben in die persönliche Struktur eingliedern kann.

Name	**Typ**
Id	String
Bezeichnung	String
Image	Integer
Owner	PublicUser

Tabelle 10: Struktur von PuplicLocation

Im PrivateUser Objekt wird weiterhin auf das Settings Objekt verwiesen, wo benutzerspezifische Einstellungen vorgenommen werden. Diese sind beispielsweise das Aktivieren bzw. Deaktivieren der Erinnerungsfunktion, die Restanzeigedauer von bereits erledigten Tasks oder Darstellungsoptionen für die Filterung der Einträge.

Außerdem wird im PrivateUser Objekt für jeden aufgenommenen Kontakt festgehalten, welche Einsichtsrechte diesem gewährt werden bzw. welche Informationen bereitgestellt wurden. Dazu wird jeweils ein Vector mit UserInfos geführt, in dem PublicUser, Nachricht und logischer Zeitstempel gespeichert sind. Den Kontakten können neben den Onlineinformationen auch Informationen über den aktuellen Aufenthaltsort mitgeteilt werden.

Weiterhin wird in dem PrivateUser Objekt auf das Wurzelelement der PrivateTaskGroup verweisen (vgl. dazu Abschnitt B.2.3 TaskGroup).

Eine Übersicht der Datenstruktur für den User ist in Abbildung 14 dargestellt. Das zugriffsgeschützte PrivateUser Objekt verweist auf den eigenen öffentlichen Teil des User Objekts (PublicUser) sowie auf den privaten UserGroup Baum, in welchem die Kontakte (PublicUser) eingefügt werden. Er verweist weiterhin auf den TaskGroup Baum (PrivateTaskGroup), den Location Baum (PrivateLocation) und enthält jeweils einen Vector mit UserInfos, die die Gewährung von Einsicht in den Onlinestatus bzw. aktuellen Aufenthaltsort betreffen und einen Vector mit Informationen, die von den Kontakten bereitgestellt wurden.

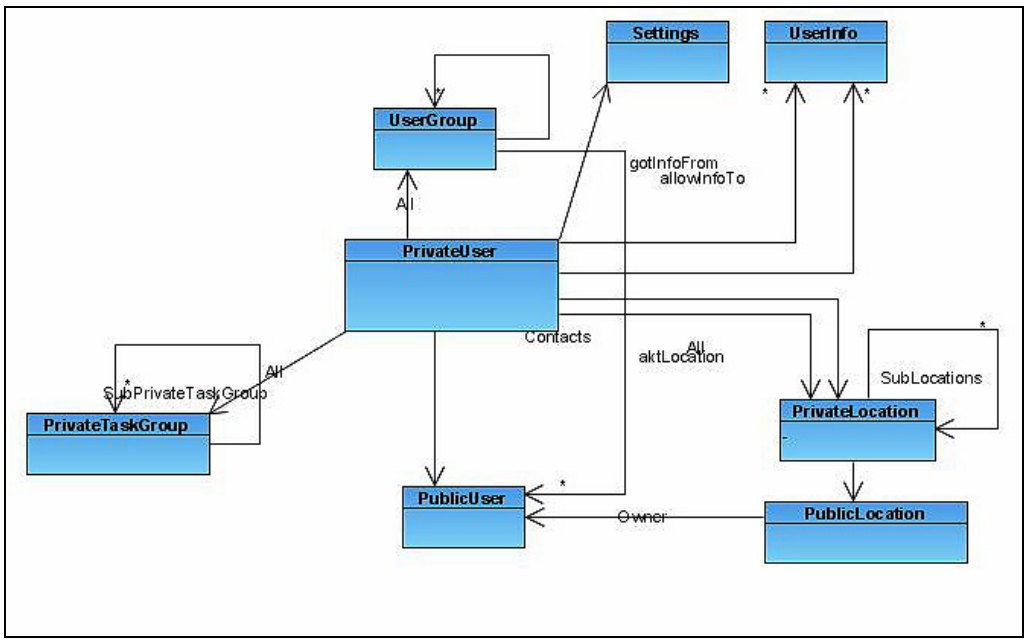


Abbildung 14: Klassenstruktur des Users

Kontakte

Wurde ein/e Benutzer/in im Space registriert (vgl. dazu Abschnitt B.2.1 Registry) und die Struktur des Users angelegt, besitzt er/sie die Möglichkeit, eine Kontaktgruppe zu verwalten. Um Kontakte zu suchen, ist es ausreichend, die lokale Registry des Peers nach den gesuchten BenutzerInnen zu durchforschen, da die Registries der verbundenen Peers die registrierten BenutzerInnen austauschen. Beim Hinzufügen eines Kontaktes in die Kontaktliste werden die Einsichtsrechte des Kontaktes in dem Vector

AllowInfoTo gespeichert. Parallel dazu wird über die Inbox eine Anfrage an den Kontakt gestellt, welche Informationen man vom diesem erhalten darf.

Im Folgenden senden BenutzerInnen während sie online sind in regelmäßigen Abständen (dem Sendeintervall) die freigegebenen Informationen an die Inboxen der Kontakte. Gleichzeitig werden die empfangenen Informationen im Vector GotInfoFrom gespeichert. Ein Kontakt wird auf offline gesetzt, wenn die Informationen über ein Vielfaches des Senderintervalls ausbleiben.

Anmeldung

Hat ein/e Benutzer/in über jenem Peer Zugriff, über den er/sie sich registriert hat (dieser Peer wird hier Home-Peer genannt), erfolgt beim Anmeldevorgang eine Prüfung der Zugangsberechtigung auf das entsprechende PrivateUser Objekt durch XVSM. Ist dies erfolgreich, hat der/die Benutzer/in Zugriff auf das PrivateUser Objekt und kann damit auf alle darin verwiesenen Daten zugreifen (vgl. dazu Abschnitt B.2.3 TaskGroup und Abschnitt B.2.4 Task).

Die BenutzerInnen können sich auch über andere, im Space bekannte, Peers (hier Replikation-Peer genannt) anmelden. Das System sucht nach dem Home-Peer der BenutzerInnen. Bei diesem zunächst reinen Lesevorgang erfolgt im Space eine Replikation der Daten vom Home-Peer auf den Replikation-Peer. Im Anschluss daran erfolgt der Anmeldevorgang wie oben beschrieben.

Einschränkungen im User-Konzept

Wird die Kommunikation der beiden Peers zu diesem Zeitpunkt verhindert, hat der/die Benutzer/in vom Replikation-Peer aus vollen Lesezugriff auf seine/ihre Daten, allerdings keine Schreibrechte, da sich die Primary Copy der Daten noch auf dem Home-Peer befinden (vgl. dazu Abschnitt B.1.2 XVSM – MozartSpaces).

Bleibt die Kommunikation jedoch aufrecht und beginnt der/die Benutzer/in Daten vom Replikation-Peer aus zu schreiben, erfolgt für alle beschriebenen

Objekte eine Migration, dh. die Schreibrechte werden auf den Replikation-Peer übertragen. Wird die Kommunikation zu diesem Zeitpunkt unterbrochen, bleiben für alle Objekte, die in der Zwischenzeit beschrieben wurden, die Schreibrechte auch weiterhin auf dem Replikation-Peer. Für jene Daten, die vom Replikation-Peer aus nur gelesen wurden, bleiben die Schreibrechte weiterhin auf dem Home-Peer.

Nach dieser Betrachtung der Datenstruktur des User Objektes im Shared Space, der Kontaktverwaltung und dem Anmeldevorgang über verteilte Peers wird nun die Struktur TaskGroups erläutert.

2.3. TaskGroup

Die TaskGroup bildet in der Datenstruktur den Einstiegspunkt zu den Tasks. Sie wird vom privaten Teil des Users referenziert. Auch die TaskGroup ist in einen privaten Teil (PrivateTaskGroup), der für alle BenutzerInnen eigens erstellt wird, und in einen öffentlichen Teil (PublicTaskGroup), der mit anderen BenutzerInnen geshared wird, geteilt.

Hat sich ein/e Benutzer/in neu registriert, wird für ihn/sie je ein Wurzelelement PrivateTaskGroup und PublicTaskGroup angelegt, von welchen aus neue Einträge generiert werden. Auch die Taskgruppen sind in einer Baumstruktur abgebildet, die beliebig verschachtelt werden kann, wobei die Verschachtelung sowohl in dem privaten Teil als auch in dem öffentlichen Teil gespeichert wird. Zusätzlich existiert in jeder Taskgruppe ein Verweis von dem privaten Teil zum öffentlichen Teil. In Tabelle 11 sind die in der PrivateTaskGroup gespeicherten Daten angeführt.

Name	Typ
Id	String
SubPrivateTaskGroup	Vector<PrivateTaskGroup>
PublicTaskGroup	PublicTaskGroup
PrivateTasks	Vector<PrivateTask>
New	Integer
Color	Integer

Tabelle 11: Struktur von PrivateTaskGroup

In der PrivateTaskGroup wird neben für den/die Benutzer/in frei definierbaren Eigenschaften auch ein Vector geführt, der die privaten Teile der enthaltenen Tasks enthält (vgl. dazu Abschnitt B.2.4 Task).

Im öffentlichen Teil der Taskgruppe befinden sich neben Eigenschaften wie zB. Bezeichnung, Zeitstempel oder Ersteller der Taskgruppe auch ein Vector mit Verweisen auf die Untergruppen sowie ein Vector mit Verweisen auf die öffentlichen Teile der enthaltenen Tasks. Zusätzlich wird ein weiterer Vector geführt, indem die involvierten BenutzerInnen gespeichert werden. In Tabelle 12 sind die in der PublicTaskGroup gespeicherten Daten übersichtlich aufgelistet.

Name	Typ
Id	String
SubPublicTaskGroups	Vector<PublicTaskGroup>
PublicTasks	Vector<PublicTask>
Owner	PublicUser
SharedUsers	Vector<PublicUser>
Bezeichnung	String
Timestamp	Long
LastModified	Long

Tabelle 12: Struktur von PublicTaskGroup

Die Datenstruktur für die TaskGroup ist in Abbildung 15 dargestellt. Der Verweis vom User (PrivateUser) auf das Wurzelelement der privaten Taskgruppenstruktur (PrivateTaskGroup) ist der Einstiegspunkt in die Nutzdaten des Systems. Die PrivateTaskGroup verweist auf ihre Unter-Taskgruppen und auf die zugehörige öffentliche Taskgruppe (PublicTaskGroup). Weiterhin verweist sie auf den privaten Teil der enthaltenen Tasks. Die öffentliche Taskgruppe verweist ebenfalls auf ihre Unter-Taskgruppen sowie auf den Ersteller der Gruppe und zusätzlich auf den öffentlichen Teil der enthaltenen Tasks.

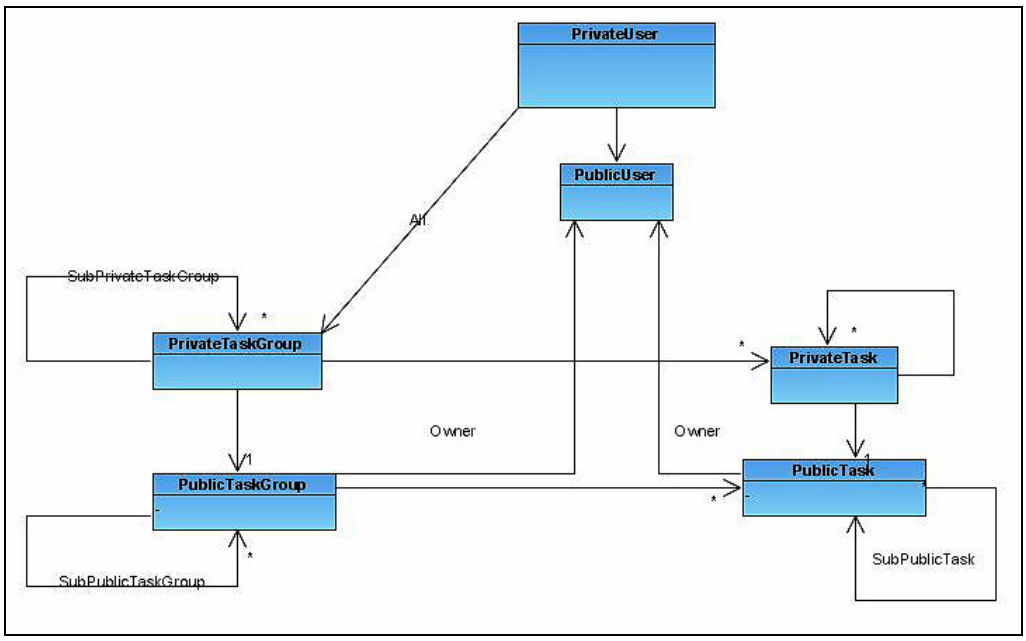


Abbildung 15: Klassenstruktur der TaskGroup

Kollaborative Taskgruppe

Bei der gemeinsamen Verwendung einer Taskgruppe mit anderen Kontakten wird diese als Nachricht in die Inbox der beteiligten BenutzerInnen gestellt (vgl. dazu Abschnitt B.2.1 Registry). Die Taskgruppe kann dabei bereits untergeordnete Elemente wie Unter-Taskgruppen oder Tasks enthalten. In der Nachricht (Message) wird der öffentliche Teil der Taskgruppe und die Kennung für das übergeordnete Element übermittelt. Eine Übersicht der gesamten Daten der Nachricht ist in Tabelle 13 dargestellt.

Name	Typ
Id	String
Sender	PublicUser
MessageTyp	Integer
Timestamp	Long
XVSMBean	XVSMBean<PublicTask or PublicTaskGroup>
Super	String
Consumed	Integer

Tabelle 13: Struktur von Message

Beim Lesen der Nachricht durch die EmpfängerInnen wird eine private Struktur nach dem Vorbild des übersendeten Elementes und dessen Unter-Elemente erstellt sowie die Verweise von den privaten Elementen zu den öffentlichen Elementen hergestellt. Danach wird geprüft, ob bei den EmpfängerInnen eine Taskgruppe mit der Kennung des übergeordneten Elements bereits besteht. Ist diese Suche erfolgreich, wird die soeben neu erstellte Struktur dieser Taskgruppe untergeordnet, ansonsten wird die Struktur im Wurzelelement eingefügt.

Werden in der Folge Änderungen an öffentlichen Elementen durch einen der beteiligten Kontakte durchgeführt, werden diese von XVSM für die AnwenderInnen transparent verbreitet.

Einschränkungen der TaskGroup

Befinden sich Kontakte, die Elemente aus der Datenstruktur sharen, auf unterschiedlichen Peers und wird die Kommunikation zwischen diesen gestört, bleiben Schreibrechte nur für jene Kontakte bestehen, auf deren Peer sich die Primary Copy befindet (vgl. dazu Abschnitt B.1.2 XVSM – MozartSpaces).

Nach der Betrachtung der Datenstruktur der Taskgruppen und der Verteilung von gemeinsam verwendeten Gruppen wird abschließend auf die Tasks näher eingegangen.

2.4. Task

Der Task dient der Speicherung der eigentlichen Nutzdaten des PIMs. Genauer gesagt, es werden Aufgaben, Erinnerungen, Faktenwissen usw. in der Datenstruktur des Space als Tasks festgehalten.

Wie die TaskGroup ist auch der Task in einen privaten und einen öffentlichen Teil getrennt und wird von dem jeweiligen Teil der Taskgruppe referenziert. Zusätzlich bildet auch der Task eine Baumstruktur, wobei diese abermals sowohl im öffentlichen als auch im privaten Teil festgehalten wird.

Dieses Konzept der Verschachtelung von Taskgruppen und Tasks erlaubt es den AnwenderInnen, Taskgruppen, dh. Kategorien zu erstellen und diese beliebig zu verfeinern bzw. zu unterteilen. In diese Kategorien können Tasks platziert und diese wiederum selbst verfeinert und unterteilt werden.

Der private Teil des Tasks enthält, neben den bereits erwähnten Verweisen auf die untergeordneten Tasks und auf den öffentlichen Teil des Tasks, die Elemente PrivatePrioStep und PrivatePrio. Diese dienen den BenutzerInnen dazu, die Tasks zu priorisieren. Es kann dabei sowohl in Stufen (PrivatePrioStep) sowie innerhalb dieser auch in kleinen Schritten (PrivatePrio) vorgegangen werden. Mit Hilfe von Severity kann der Tasks klassifiziert werden. Die zz. bestehenden Möglichkeiten sind:

- Neuer Task
- Task In Arbeit
- Klassifizierung Hoch
- Klassifizierung Mittel
- Klassifizierung Niedrig
- Wiederkehrender Task

Weitere Eigenschaften, auf die hier nicht näher eingegangen wird, sind in Tabelle 14 dargestellt.

Name	**Typ**
Id	String
SubPrivateTask	Vector<PrivateTask>
PublicTask	PublicTask
New	Integer
Color	Integer
PrivatePrioStep	Integer
PrivatePrio	Long
RememberAgain	Long
Severity	Integer
...	

Tabelle 14: Struktur von PrivateTask

Der öffentliche Teil des Tasks, die Daten sind in Tabelle 15 aufgelistet, enthält weiterhin einen Vektor mit der Referenz auf die involvierten BenutzerInnen sowie einen Vector, der Verweise auf Note Objekte enthält.

Name	Typ
Id	String
SubPublicTasks	Vector<PublicTask>
Owner	PublicUser
SharedUsers	Vector<PublicUser>
PublicLocation	PublicLocation
Notes	Vector<Notes>
Bezeichnung	String
Create	Long
Deadline	Long
Status	Integer
PublicPrio	Long
Done	Long
Timestamp	Long
LastModified	Long

Tabelle 15: Struktur von PublicTask

Das Note Element dient als Notizfeld für den Task. Die Notizen sind für alle beteiligten BenuzerInnen, die den Task sharen, ersichtlich. Antworten auf Notes können erstellt werden, indem dem Task weitere Notes angehängt werden. In jeder Note ist der/die Verfasser/in gespeichert, Änderungen können nur von diesem/dieser durchgeführt werden. Die Daten des Note Objektes sind in Tabelle 16 dargestellt.

Name	Typ
Id	String
Text	String
Owner	PublicUser
Create	Long
New	Integer
LastModified	Long

Tabelle 16: Struktur von Note

Der PublicTask verweist weiterhin auf den öffentlichen Teil der Location (vgl. dazu Abschnitt B.2.1 Registry). Damit wird der Inhalt des Tasks mit einem Austragungsort verknüpft. Dies ermöglicht den AnwenderInnen eine zusätzliche Sichtweise auf die Tasks, indem diese beispielsweise nach dem aktuellen Aufenthaltsort gefiltert werden.

Um die Wichtigkeit eines Tasks auch an die involvierten Personen kommunizieren zu können, steht, neben den Priorisierungen im privaten Teil, auch eine Priorisierung im öffentlichen Teil (PublicPrio) des Tasks zur Verfügung.

Weitere Eigenschaften des PublicTasks sind Status, Deadline oder Datum der letzten Änderung. Die Datenstruktur für den Task ist in Abbildung 16 übersichtlich dargestellt.

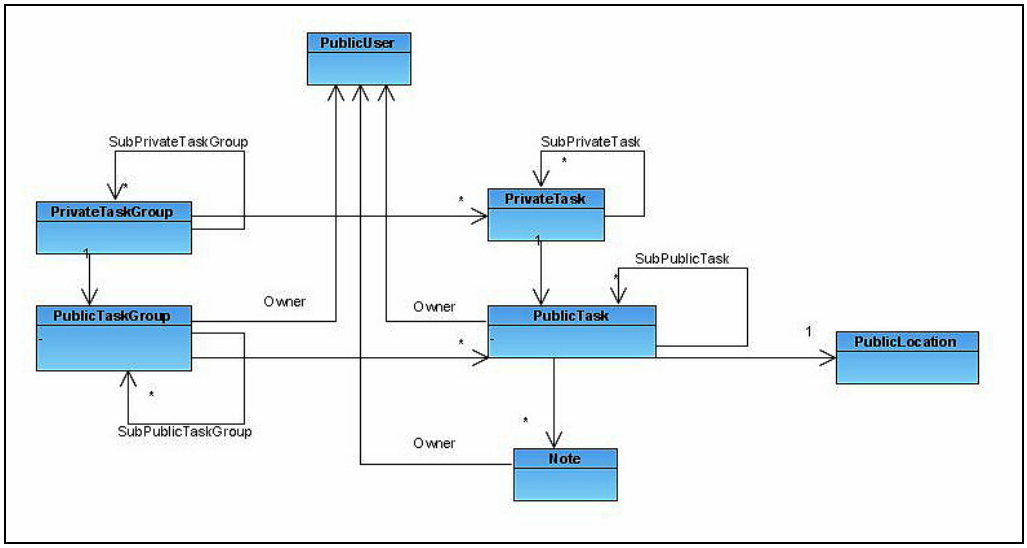


Abbildung 16: Klassenstruktur des Tasks

Kollaborativer Task

Die Verteilung von Tasks an involvierte BenutzerInnen verhält sich entsprechend der Verteilung der Taskgruppe. Es wird daher an dieser Stelle auf das Kapitel TaskGroup verwiesen.

Importieren von Tasks

Werden Daten (TaskGroups, Tasks oder Locations), die zuvor exportiert wurden, zu einem späteren Zeitpunkt wieder importiert, ist zu beachten, dass es nicht ausreicht, die Datenstruktur lediglich wiederherzustellen. Für Objekte, die nicht geshared werden, stellt dies zwar kein Problem dar. Allerdings erfordern gemeinsam verwendete Elemente eine Sicht auf dasselbe Objekt. Zudem ist es möglich, dass an den gemeinsamen Elementen inzwischen gearbeitet wurde und daher die Daten aus der Exportdatei nicht mehr aktuell sind.

Aus diesem Grund wird bei einem Import für gesharte Elemente ein Abgleich der Daten durchgeführt. Zu diesem Zweck wird von der Applikation eine Anfrage an die Inbox der involvierten BenutzerInnen gesendet, in welcher das aktuelle Element angefordert wird. Als Antwort wird die Referenz auf das Element retourniert und kann schließlich in die Datenstruktur integriert werden. Damit ist auch nach einem Datenimport wieder eine gemeinsame Sicht auf die öffentlichen Elemente gewährleistet.

Einschränkungen der Tasks

Wie auch bei den Taskgruppen (vgl. dazu Abschnitt B.2.3 TaskGroup) treten durch das Unterbrechen der Kommunikation zwischen den Peers Einschränkungen in den Schreibrechten von Elementen auf. Einschränkungen betreffen beim Task den öffentlichen Teil des Elementes. Genauer gesagt, können Änderung nur noch von jenen BenutzerInnen durchgeführt werden, die sich auf dem Peer mit der letzten Schreiboperation vor dem Kommunikationsabbruch befinden. Auch das Erstellen von Notes ist damit eingeschränkt. Da Notes aber generell nur von den ErstellerInnen bearbeitet werden können und somit eine Migration der Primay Copy in der Regel nicht stattfindet[3], sind bereits erstellte Notes jederzeit änderbar.

[3] Eine Migration erfolgt nur dann, wenn sich der Ersteller über einen anderen Peer einloggt und die Note ändert.

Wie oben bereits erwähnt, besitzt die Datenstruktur im XVSM-Space die Einstiegspunkte Registry und die User-Elemente, von diesen ausgehend wird auf die Nutzdaten verwiesen. Durch die Verwendung von Präfixen für die Einstiegspunkte in die Datenstruktur können mehrere unabhängige Datenstrukturen dh. PIM-Netzwerke gebildet werden.

Es wurde hier eine Datenstruktur und ihre Einschränkungen vorgestellt, die es ermöglicht, basierend auf dem Virtual Shared Memory Prinzip von XVSM einen flexiblen verteilten PIM zu implementieren. Der Einstiegspunkt in die Datenstruktur wird durch die dezentrale Registrierung und das User Element realisiert. Den BenutzerInnen wird die Möglichkeit gegeben, Aufgaben zu managen, diese frei zu arrangieren und zu verfeinern. Die Aufgaben können in Kategorien verwaltet und mit frei wählbaren Örtlichkeiten verknüpft werden. Zentrales Konzept bildet die Zweiteilung von Elementen in öffentliche und private Teile, wodurch die Sicherheit vor unberechtigtem Zugriff auf Daten realisiert wird.

3. Anwendung

3.1. Mogi

Das hier entwickelte PIM System Mogi 2.0 (Management Organizer) basiert auf einer Vorstudie von A.o. Univ. Prof. Dr. Dipl.-Ing. eva Kühn (www.complang.tuwien.ac.at/eva). In Abbildung 17 ist das Hauptfenster dieser Vorversion ersichtlich. Es können dabei Informationen in Form von Tasks mit Deadlines und Kommentaren listenförmig gespeichert werden. Die Tasks können dabei klassifiziert (in work, high, medium, low, ...) und priorisiert werden.

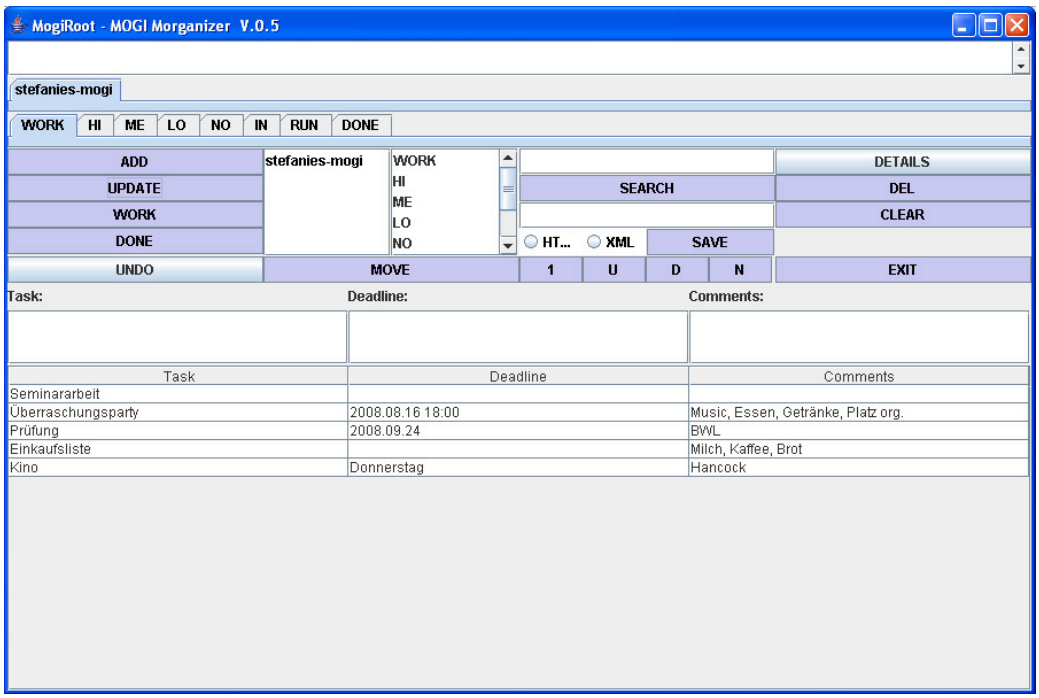


Abbildung 17: Hauptfenster von Mogi

Durch den Einsatz dieser Vorversion ergaben sich unter anderem folgende Ideen und Denkanstöße:

- Neue Tasks in der Taskliste grundsätzlich an oberster Stelle hinzufügen.
- Beim Einfügen von Tasks eine Prüfung auf Duplikate durchführen
- Kopieren von Tasks ermöglichen

Verschlüsseltes Exportieren von Tasks ermöglichen

Verbesserung des Benutzer Interfaces

Verbesserung der Such- und Filtermöglichkeiten

Drag & Drop von Tasks

Informationen über die Arbeitszeit für die Abwicklung von Tasks

Erinnerungsfunktion bei Deadlines

Ad Hoc Workflows

Trennung von privaten um öffentlichen Tasks

3.2. Mogi 2.0

Wie bereits im Kapitel A.2 „Psychologische Hintergründe“ ausgeführt wurde, sind Episoden im autobiographischen Gedächtnis der kleinste Baustein. Es handelt sich dabei um klar abgrenzbare Events, die Teil einer größeren Serie von Events sein können. Diese Erkenntnis wurde bei Mogi 2.0 in Form von Tasks umgesetzt. Genauer gesagt ist es den BenutzerInnen möglich, diese Episoden in Form von Tasks im System zu speichern, wobei Serien von Episoden einerseits durch die Gruppierung von Tasks andererseits durch die Unterteilung in Sub-Tasks abgebildet werden können.

Als weiteres Merkmal des episodischen Gedächtnisses wurde festgehalten, dass sich der Mensch genaue Zeitangaben nur schwer merken kann bzw. dass es ihm leichter fällt, sich an Dinge zu erinnern, wenn er bereits Teile der Information von einer Episode weiß, wie beispielsweise den Austragungsort. Diesem Umstand wird in Mogi 2.0 durch 6 verschiedene Filtermöglichkeiten bzw. verschiedene Zugangsmöglichkeiten zu den Tasks entsprochen.

Den BenutzerInnen werden Zugangsmöglichkeiten aus sechs verschiedenen Perspektiven geboten. Im Einzelnen handelt es sich dabei um:

- Kategorien/Taskgruppen: die BenutzerInnen haben die Möglichkeit, Tasks über die zugeordnete Kategorie aufzufinden. Über die Zuteilung von Sub-Kategorien können die Tasks schrittweise eingeschränkt werden.

- Involvierte Kontakte: dabei können die Tasks mit Hilfe der Kontakte bzw. ganzer Kontaktgruppen eingeschränkt werden. Dh. es werden nur jene Tasks angezeigt, die mit den Personen bzw. Gruppen gemeinsam verwendet werden.
- Lokalitäten: die Tasks werden nach den verknüpften Lokalitäten selektiert. Auch hier ist eine schrittweise Verfeinerung der Auswahl durch die Selektion von Unter-Lokalitäten möglich.
- Zeitangaben: erfolgt die Suche nach Tasks mittels Zeitangaben, wird in einem ersten Schritt nur ein grober Zeitraum durchsucht. Je nach gewonnenem Ergebnis kann das Zeitfenster mittels Schieberegler anschließend verfeinert werden.
- Klassifizierung: im User Interface befinden sich Buttons, die es den BenutzerInnen ermöglichen, auf schnellem Wege Tasks zu filtern. Klassen, die von Mogi 2.0 angeboten werden, sind „neu“, „erledigt“, „periodisch“, „zu erledigen binnen (Tag, Woche, Monat)“.
- Inhaltssuche: die Tasks können nach Inhalt in der Bezeichnung bzw. in den Notes durchsucht werden.

Diese Selektoren können frei variiert und kombiniert werden um die Tasks wie gewünscht zu filtern. In Abbildung 18 ist das Hauptfenster von Mogi 2.0 ersichtlich.

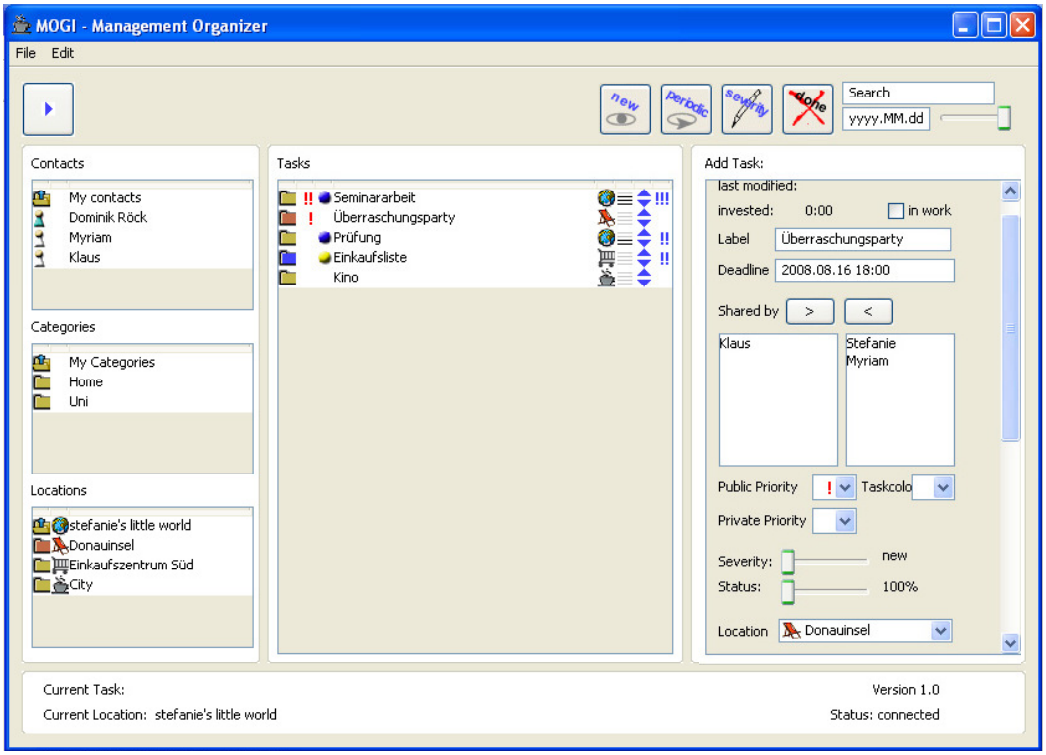


Abbildung 18: Hauptfenster von MOGI 2.0

Das Hauptfenster gliedert sich in drei Bereiche. Im Zentrum befindet sich die Liste der angezeigten Tasks. Die für die BenutzerInnen wichtigsten Informationen sind dabei bereits in der Liste erkennbar und veränderbar. Beispielsweise ist umgehend an der Farbe des Symbols erkennbar, ob der Task noch Sub-Tasks enthält. Weiterhin sind die öffentlichen und die privaten Prioritäten umgehend erkennbar und durch ein einfaches Anklicken können sie hinaufgesetzt werden. Ebenso ist der Status des Tasks ersichtlich und kann mittels Klick stufenweise erhöht werden. Die Lokalität, die dem Task zugeordnet ist, wird durch das Bildsymbol abgebildet. Zusätzlich wird zur besseren Orientierung noch die Farbe, die dem Task zugeordnet wurde, angezeigt. Wird ein Task angeklickt, werden die Eigenschaften des Tasks in der Detailansicht im rechten Bereich dargestellt und, falls der Task noch Sub-Tasks enthält, werden diese in der Taskliste angezeigt.

Im linken Bereich werden untereinander Kontaktguppen/Kontakte, Kategorien und Lokalitäten angezeigt. Dabei ist auch hier am Symbol erkennbar, ob Sub-Elemente existieren. Bei den Kontakten ist weiterhin ersichtlich, ob sich der

Kontakt im Online-Status befindet. Wie bei den Tasks werden auch hier die zugeordneten Farben bzw. bei den Lokalitäten die zugeordneten Symbolbilder dargestellt. Wird ein Element ausgewählt, wird die Taskliste im zentralen Bereich nach dem entsprechenden Element gefiltert. Haben Elemente zusätzliche Sub-Elemente, werden diese beim Anwählen auch im entsprechenden Fenster angezeigt. Dabei ist es möglich, mehrere Filter zu kombinieren.

Der rechte Bereich dient zur Darstellung von Detailinformationen bzw. der Eingabe von Daten. Je nach Tätigkeit wird in diesem Bereich der Inhalt ausgetauscht.

Wie in Abschnitt A.1.3 bereits festgestellt wurde, wollen BenutzerInnen, um schneller arbeiten zu können, keine unnötigen Daten ausfüllen. Weiterhin wird eine Aufgabe umso häufiger eingetragen, je wichtiger sie den BenutzerInnen erscheint (Bellotti et al. 2004). Dem wurde entsprochen, indem nur eine Betitelung für das Erstellen eines Tasks nötig ist. Alle weiteren Angaben sind optional und können zu einem späteren Zeitpunkt erfolgen. Um doppelte Einträge zu vermeiden, erfolgt während der Eingabe eines neuen Tasks ein Abgleich mit den bereits bestehenden. So wird den BenutzerInnen parallel eine Taskliste mit möglichen Duplikaten aufgezeigt.

3.3. Walkthrough

Hauptfester Benutzer Dominik: – Anlegen eines neuen Tasks

1) Hinzufügen eines *neuen Tasks* in der *Kategorie* „Freizeit“.
2) *Betitelung* des Task mit Überraschungsparty. Setzen eines *Fälligkeitsdatum*. *Sharen des Inhalts* mit zwei Kontakten. Setzen einer *Public Priority* (rotes !), welche für Teilnehmer mit denen der Task geshared wird, sichtbar ist. Hinzufügen der *Örtlichkeit* „Donauinsel“.
3) Der neue Task wird im mittleren Hauptfenster angezeigt.

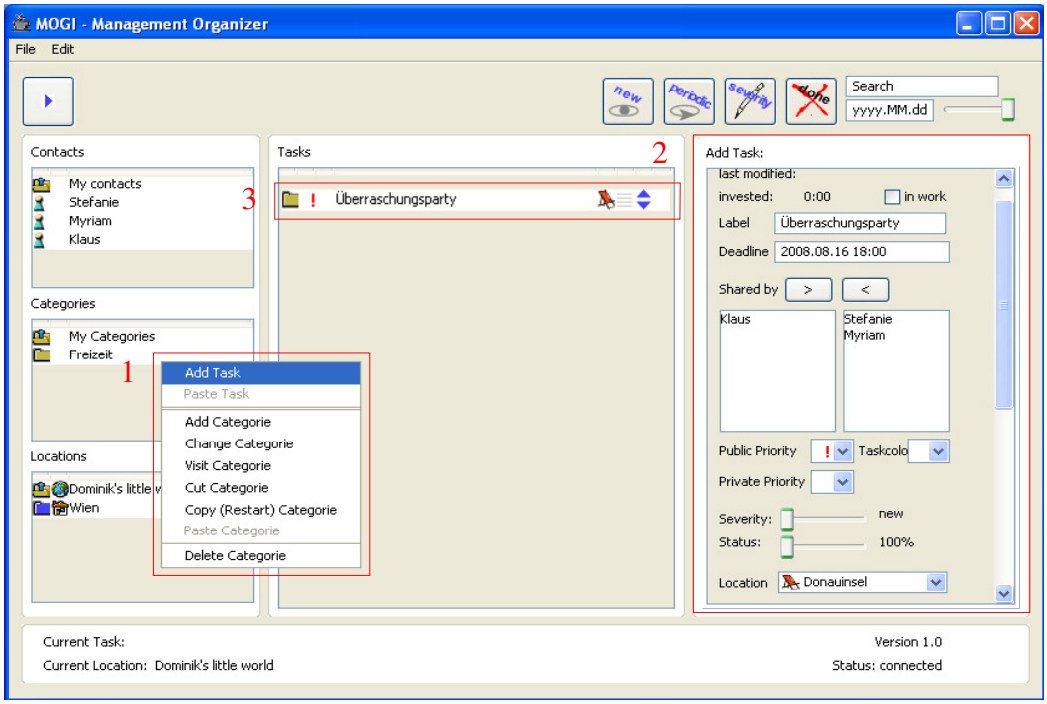


Abbildung 19: Hauptfenster Benutzer Dominik: Hinzufügen eines Tasks

Hauptfenster Benutzerin Stefanie

Der *neue Task* „Überraschungsparty“ wird bei Benutzerin Stefanie im mittleren Hauptfenster angezeigt.

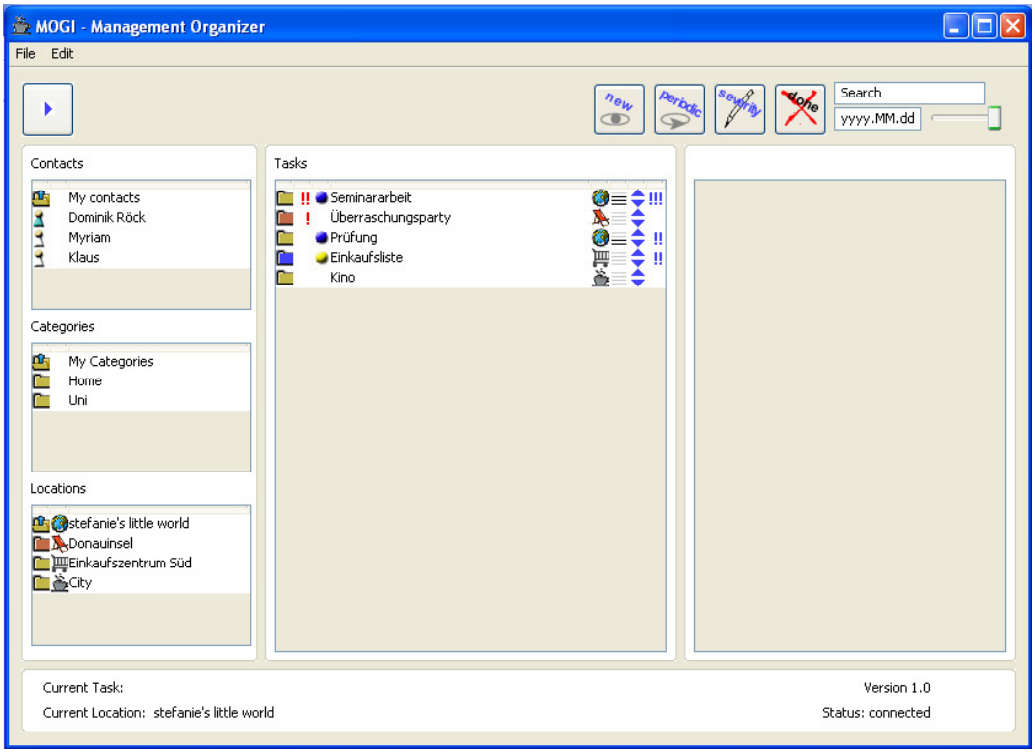


Abbildung 20: Hauptfenster Benutzerin Stefanie: der neue Task wird eingeblendet

Detailansicht - mittleres Hauptfenster

Abbildung 21: Detailansicht - mittleres Hauptfenster

(von links nach rechts):

Ordner: Blaue Ordner zeigen an, ob ein Task Unterelmente enthält (vgl. Task „Einkaufsliste“). Rote Ordner zeigen an, ob es sich um einen neuen Task handelt, der von einem Kontakt geshared wurde (vlg. Task Überraschungsparty).

Rote Rufezeichen: Zeigen die öffentliche Priorität an, die für andere Kontakte ersichtlich ist. Die Priorität kann durch einen einfachen Klick auf das Symbol erhöht werden.

Farbcodes: Tasks können für eine bessere kontextabhängige Zuordnung mit Farbcodes versehen werden. Farbcodes können frei zugeordnet werden. Im Beispiel sind alle Tasks, die mit der Universität in Zusammenhang stehen, mit einem blauen Punkt gekennzeichnet.
Titel: Pflichtfeld beim Anlegen von Tasks.
Lokalität: Die Lokalität wird hier in Form eines Icons, das der Lokalität zugeordnet wurde, angezeigt. Vgl. Liegestuhl verweist auf Donauinsel.
Status: Die vier übereinander liegenden Linien geben Auskunft über den Status des Projektes in 25% Stufen. Keine Linie = noch nicht begonnen; vier Linien = Aufgabe abgeschlossen. Der Status kann durch einen einfachen Klick auf das Symbol erhöht werden.
Pfeilsymbole: Durch Klicken auf den Pfeil nach oben bzw. unten kann die Reihenfolge der Tasks angepasst werden.
Blaue Rufezeichen: Zeigen die private Priorität an, die nur für den User ersichtlich ist. Die Priorität kann durch einen einfachen Klick auf das Symbol erhöht werden.

Hauptfenster User Stefanie: Anwendungsbeispiel Brainstorming

1) Der Task „Überraschungsparty“ enthält das *Unterelement „Ideensammlung“*. Das System wird hier für ein Brainstorming verwendet.
2) Die Benutzerin „Stefanie“ fügt der Ideensammlung ein neues Element „Einladungen“ hinzu.

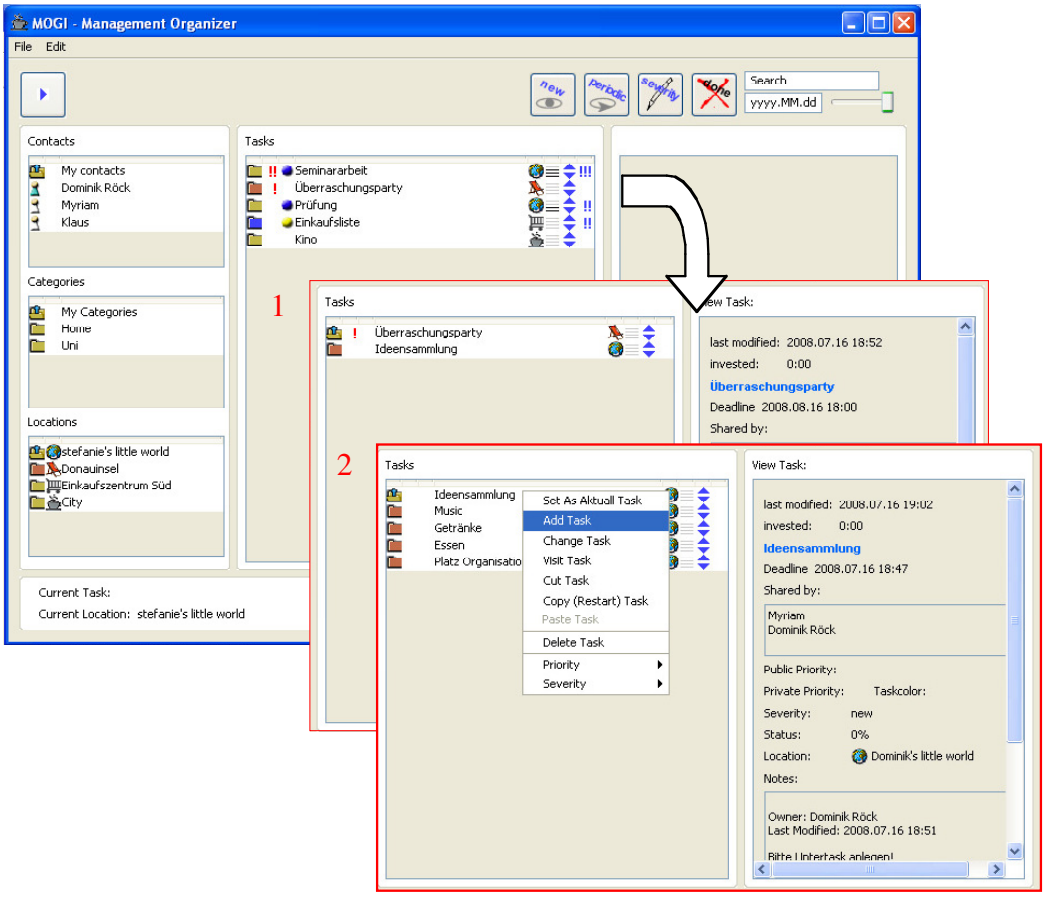


Abbildung 22: Hauptfenster Benutzerin Stefanie

Hauptfenster Benutzerin Stefanie: Anwendungsbeispiel private und öffentliche Priorität

Die *Sub-Tasks* „Einladungen“ und „Music“ sind mit einer *öffentlichen Priorität* versehen. Die öffentliche Priorität ist für alle Teilnehmer ersichtlich und bedeutet hier, dass alle Teilnehmer diesen Task zu erledigen haben.

Die *Sub-Tasks* „Einladungen“, „Music“, „Getränke“ sind mit einer *privaten Priorität* versehen. Diese Tasks fallen in den Aufgabenbereich der Benutzerin „Stefanie“ und wurden daher von ihr als wichtig gekennzeichnet.

Der *Status* der einzelnen Tasks ist über das Symbol ☰ ersichtlich.

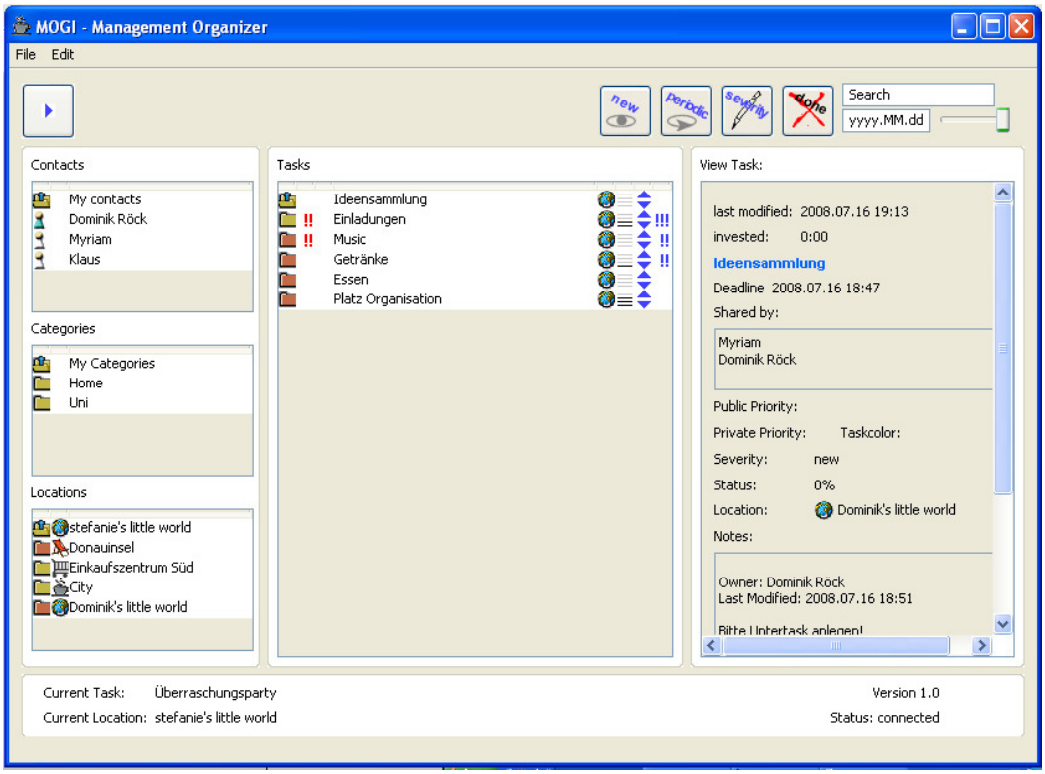


Abbildung 23: Hauptfenster Benutzerin Stefanie: Anwendungsbeispiel private und öffentliche Priorität

Als besondere Vorteile von Mogi 2.0 sind zu nennen:

Für die Inhaltssuche kann ein Schwellenwert definiert werden, sodass die Ergebnisse auf jene Tasks eingeschränkt werden, deren Übereinstimmung mindestens dem Schwellenwert entsprechen. Die Suchergebnisse werden entsprechend der Anzahl dieser Übereinstimmungen sortiert dargestellt.

Die BenutzerInnen können außerdem einstellen, ob während der Eingabe von Tasks eine Überprüfung auf Duplikate erfolgen soll.

Tasks können per Drag & Drop verschoben, kopiert oder erledigte Tasks reaktiviert werden.

Neue Task werden in der Taskliste grundsätzlich an oberster Stelle hinzugefügt. Wobei eine Reihung nach Priorisierung stattfindet. Damit sind die aktuellsten Tasks auf einen Blick ersichtlich.

Die Daten können als XML File gesichert werden, wobei auch eine verschlüsselte Speicherung möglich ist.

Tasks an denen gearbeitet wird, können als „in work“ gekennzeichnet werden. Die Summe dieser Arbeitszeiten ist für den Benutzer ersichtlich.

Neben den oben erwähnten Such- und Filtermöglichkeiten, stehen den BenutzerInnen weitere, am oberen rechten Fensterrand vordefinierte Filter, zur Verfügung. Diese sind: „neue Tasks“, „periodische Tasks“, „Sortierung nach Klassifizierung (high, medium, low, etc.)“ und „erledigte Tasks“.

Es kann angegeben werden ob und wie lange vor Ablauf einer Deadline eine Erinnerung erfolgen soll.

Der Onlinestatus und der Aufenthaltsort von Kontakten kann eingesehen werden. Um die Privatsphäre zu wahren, kann pro Kontakt definiert werden, welche Informationen bekannt gegeben werden.

Lokalitäten können bildhafte Icons zugeordnet werden. Diese werden in der Taskliste eingeblendet und erlauben eine schnelle Zuordnung von Tasks auf einen Blick.

Abschließend werden die in Kapitel A.1 – „Aktuelle Forschungen und Softwaresysteme“ vorgestellten Systeme an Hand einer Matrix (Tabelle 17) verglichen und bewertet. Die Details wurden in Kapitel A.1 – „Aktuelle Forschungen und Softwaresysteme“ sowie in Kapitel A.3.7 – „Abgrenzungen“ bereits besprochen.

	Digital Notes	Taskmaster (E-Mail)	Tagebuchgenaratoren	BSCW	Groove	IGoogle To-Do Liste	Mogi 2.0
Aufgaben / Erinnerungen	● pro Aufgabenliste	●	❍	●	❍	● pro Aufgabenliste	● auch für Sub-Task
Archivieren von Aktivitäten	❍	●	● Aufzeichnung aller Aktivitäten	●	❍	●	● mit Reaktivierungsfunktion für Aufgaben
Faktenwissen	●	●	❍	●	*	● Ad-Ons	●
Workflow	❍	❍	❍	❍	❍	❍	● Ad Hoc
Kollaboration	❍	❍ versenden; keine Synchronisierung	❍	●	*	● alles oder nichts	● pro Element und Sub-Eemelent
Priorisierung	❍	●	❍	●	*	●	● private und öffentliche
Dokumente	● Bilder	●	● im Rahmen der Aufzeichnungen	●	●	● Ad-Ons (keine Verknüfung zu den Aufgaben)	❍ in Planung
Suchen / Filter	●	●	●	●	●	❍ Filter nach Örtlichkeit und Klassifizierung	● Filter nach Örtlichkeit, Klassifizierung, Kategorien, Benutzern, Zeit
Kategorien	❍	❍	❍	❍	*	❍	● Kategorien beliebig unterteilbar
Kennzeichnen	● Farben	● Farben und Icons	❍	❍	*	❍	● Farben
Lokatilität	❍	❍	❍	❍	*	●	● beliebig unterteilbar
Sonstiges	+ einfach + geläufig - nur Microsoft	+ geläufig - Informations-überflutung - nur Aufgaben-management - Prototyp	+ gesamte Vergangenheit wird gespeichert - Informations-überflutung - nur Vergangenheit	+ Zentrale Speicherung - Daten nicht lokal verfügbar	+ lokale Speicherung + Synchronisierung - nur Microsoft	+ Synchronisation + viele Ad-Ons - keine Datenverknüpfung zwischen Ad-Ons - Daten nicht Lokal	+ freie Datenstruktur mit Gruppierung und beliebigen Sub-Elementen + 6 Filter für Informationsabruf + Synchronisation - Prototyp

Tabelle 17: Vergleichsmatrix der vorgestellten Systeme

* Groove ist ein System das Daten zwischen Geräten synchronisiert, bietet aber selbst keine PIM Funktionen

4. Bewertung

4.1. Zeit und Raum

Die entwickelte Datenstruktur im XVSM-Space ermöglicht den AnwenderInnen eine nahezu synchrone Verwendung von gemeinsamen Tasks über geographisch verteilte Gebiete. Für BenutzerInnen, die sich zu unterschiedlichen Zeitpunkten im System anmelden, wird ebenso die asynchrone Kommunikation unterstützt. Nachrichten an BenutzerInnen werden dabei im System zwischengespeichert und automatisch beim nächsten Einloggen abgearbeitet.

4.2. Skalierbarkeit

Die Daten der BenutzerInnen werden im XVSM-Space gespeichert. Dabei werden pro gespeichertem Element 2 Objekte erzeugt (privater Teil und öffentlicher Teil). Werden pro BenutzerInnen durchschnittlich x Elemente gespeichert, so ergibt sich daraus für n Benutzer in Summe $O(n)$ Objekte.

$$O(n) = n * 2x$$

Bei der gemeinsamen Verwendung von Elementen greifen BenutzerInnen, die sich am selben Peer befinden, auf dieselben öffentlichen Objekte zu. Dh. es reduzieren sich die Objekte im Space. Wird allerdings der Worst Case angenommen und alle BenutzerInnen befinden sich auf unterschiedlichen Peers, sodass die öffentlichen Teile der Objekte repliziert werden müssen und wird weiters angenommen, dass durchschnittlich y der Elemente mit durchschnittlich m BenutzerInnen gemeinsam verwendet werden, so ergibt sich:

$$O(n) = n * 2x + n * m * 2y$$

Im tatsächlichen Einsatz ergeben sich Verbesserungen dadurch, dass BenutzerInnen die sich kennen, mit hoher Wahrscheinlichkeit am selben Peer

registriert sind. Eine Analyse der Ressourcenabschätzung ergibt, dass es sich bei x und y um Durchschnittswerte handelt, sie können damit als konstanten angenommen werden. Auch die Anzahl der BenutzerInnen mit denen Tasks geshared werden, kann als konstant angesehen werden, da diese ab einer bestimmten Anzahl von BenutzerInnen im System nicht weiter ansteigt. Daher ergibt sich ein Gesamtvolumen an Objekten von:

$$O(n) = n$$

Dies entspricht einem linearen Wachstum von benötigten Ressourcen pro Benutzer/in im System.

Die Kommunikation $K(n)$ im System kann wie folgt abgeschätzt werden. Für Tasks die nicht geshared werden, ist keine Kommunikation nötig. Für gesharte Tasks erfolgt eine Nachricht an die beteiligten BenutzerInnen. Dh. es werden durchschnittlich y Tasks mit m BenutzerInnen gemeinsam verwendet.

$$K(n) = n*y*m$$

Änderungen von gesharten Task müssen an BenutzerInnen am selben Peer nicht übermittelt werden, da diese dasselbe Objekt einsehen. Angenommen alle BenutzerInnen befinden sich auf unterschiedlichen Peers und im Durchschnitt werden die Tasks k mal geändert, so ergibt sich eine Gesamtkommunikation von:

$$K(n) = n*y*m + n*y*m*k$$
$$K(n) = n*y*m*(k+1)$$

Wie oben, können y, m als Kostanten angenommen werden. Auch bei den durchschnittlichen Änderungen an einem Task handelt es sich um eine Konstante. Wobei sich auch hier Verbesserungen dadurch ergeben, dass sich die bekannten BenutzerInnen meist über denselben Peer registrieren. Daher ergibt sich eine Gesamtkommunikation von:

$$K(n) = n$$

Dies entspricht ebenfalls einem linearen Wachstum pro BenutzerInnen.

4.3. Flexibilität

Die gewählte Datenstruktur erlaubt es allen BenutzerInnen, gemeinsam verwendete Daten individuell zu arrangieren und nach ihren Vorlieben zu organisierten dh. in der eigenen Datenstruktur einzuordnen. Die Elemente können beliebig unterteilt oder gruppiert werden. Dennoch bleibt eine einheitliche Sicht auf den Inhalt der Elemente bestehen und geänderte Attribute werden mit den beteiligten Personen synchronisiert.

Derzeit besteht für die AnwenderInnen noch nicht Möglichkeit, das System durch zusätzliche Verknüpfungen bzw. Attribute an individuelle Anforderungen anzupassen. Dies ist in zukünftigen Erweiterungen des Systems zu berücksichtigen.

Aufgrund der Multi-Tier-Architektur ist es möglich, das System an verschiedene Geräte anzupassen. Durch Austausch des Front-End kann das System für Mobile Geräte angeboten werden.

4.4. Plattformunabhängigkeit

XVSM bietet die Möglichkeit, Applikationen plattformunabhängig zu verknüpfen. In Kombination mit dem in Java implementierten PIM wird damit den BenutzerInnen ein Zugang zu allen Plattformen geboten.

Eine Implementierung in anderen Sprachen ist möglich, da XVSM die Werkzeuge für die Verteilung, Kommunikation und Synchronisierung zwischen den Applikationen zur Verfügung stellt.

4.5. Offline Betrieb

Die Funktionsweise der PIMs bleibt auch bei Kommunikationsabbruch zwischen Peers erhalten. Nachrichten an BenutzerInnen von anderen Peers werden hinterlegt und nach Wiederherstellung der Kommunikation übermittelt. Änderungen an Elementen werden von XVSM beim Wechsel in den Online-Betrieb transparent verbreitet.

4.6. Datenpersistenz

Alle Informationen werden automatisch von XVSM permanent gespeichert und durch XVSM-Mechanismen wie Recoverability auch vor Systemausfällen geschützt. Bei Systemwiederherstellungen kommt es daher weder zu Datenverlusten noch zu inkonsistenten Daten. Für Schreibvorgänge auf mehrere Elemente im Space stehen Transaktionen zur Verfügung.

4.7. Konfliktlösung

Auftretende Konflikte werden im PIM so weit wie möglich automatisiert gelöst. Beispielsweise beim Importieren von verteilten Aufgaben kann es vorkommen, dass inzwischen an den Tasks gearbeitet wurde. Durch einen logischen Zeitstempel erkennt die Applikation während dem Datenabgleich die Änderung und aktualisiert den Task. Hingegen werden Änderungen im laufenden Betrieb von XVSM mit Hilfe des Verteilungsprotokolls an alle KommunikationsteilnehmerInnen weitergeleitet. Da die Daten nur vom Peer mit der Primary Copy geändert werden können, treten keine Inkonsistenzen auf.

4.8. Sicherheit

Die im Virtual Space erzeugte Datenstruktur sichert einen respektvollen Umgang mit privaten Daten. Der Zugriffsschutz auf die Einstiegspunkte in die Datenstruktur durch XVSM (vgl. dazu Abschnitt B.2.2 User) und daran anknüpfend die eindirektionale Verweisung von der privaten auf die öffentliche

Struktur der Daten erlaubt es unberechtigten BenutzerInnen nicht, Zugriff auf Daten zu erhalten.

4.9. Interaktion mit anderen Systemen

Derzeit wird ein Export und Import von Daten im XML Format angeboten, um in Kooperation mit Applikationen von Drittanbietern zu treten. Funktionalität, wie beispielsweise die Erstellung von Tasks durch verschieben von E-Mails per Drag and Drop, ist bei zukünftigen Erweiterungen zu erwägen.

5. Ausblick

Im Folgenden wird beschrieben, wie der PIM erweitert bzw. verbessert werden kann, um in der Zukunft den Schritt von einem Prototypen zu einem verkaufsfähigen Produkt zu ermöglichen.

5.1. Integration von Dokumenten

Um die Einsatzmöglichkeiten des PIMs zu erweitern, kann in folgenden Versionen eine Unterstützung für das Dokumentenmanagement angeboten werden. Denkbar ist, dass Dokumente bzw. Dateien an die Tasks angehängt werden können und mit einer Versionisierung versehen werden. Damit stehen sie für eine gemeinsame Bearbeitung durch die BenutzerInnen zur Verfügung.

5.2. Erweiterte Eingabe-/Ausgabemöglichkeiten

Um weitere Zugriffsoptionen anzubieten, kann eine Anbindung an mobile Geräte angedacht werden. Damit wird gewährleistet, dass die Hilfsmittel, die der PIM zur Verfügung stellt, von überall verwendet werden können. Zudem kommt es öfter vor, dass am Arbeitsplatz keine Software installiert werden kann. Aus diesem Grund wäre ein Webinterface für den Zugriff auf die PIM-Daten hilfreich.

Zusätzliche Eingabemöglichkeiten können realisiert werden, indem es den BenutzerInnen ermöglich wird, Tasks mit Eingabequellen zu verknüpfen, beispielsweise über die Erstellung von Audionachrichten, die dem Task angehängt werden. Die Nachricht kann dann auch zu einem späteren Zeitpunkt noch textuell kommentiert bzw. übertragen werden. Ist die Applikation auf mobilen Geräten implementiert, können die Tasks durch Bilder bzw. Videomitschnitte vom Mobiltelefon mit zusätzlicher Information ergänzt werden. Dies ermöglicht es den AnwenderInnen, auch von unterwegs zB. beim Autofahren Tasks zu erfassen.

5.3. Zusammenspiel mit Applikationen

Damit der PIM die zentrale Stellung als Aufgaben Manager ausbauen und benutzerfreundlicher werden kann, ist es möglich, die Applikation zu erweitern, indem sie mit Applikationen von anderen Herstellern kommunizieren und interagieren kann. Treffen beispielsweise Aufgaben per E-Mail bei AnwenderInnen ein, wird es ihnen ermöglicht, per Drag and Drop die E-Mail in den PIM zu ziehen und dieser generiert automatisch eine Aufgabe. Wenn vorhanden, werden aus der E-Mail bereits Informationen wie Termine oder beteiligte Personen in den Task korrekt eingefügt.

5.4. Positionserkennung

Mobile Geräte werden mit immer mehr Funktionen ausgestattet. GPS Empfänger in Mobiltelefonen sind im Aufkommen. Der PIM kann am mobilen Gerät diese Informationen nutzen und die verwalteten Lokalitäten zusätzlich mit den GPS Koordinaten versehen. Kommen die AnwenderInnen in den Umkreis dieser Lokalität, werden ihnen automatisch die an diesem Ort zu erledigenden Aufgaben bzw. die mit diesem Ort verknüpften Informationen zur Verfügung gestellt. (Brown 1995)

Für Geräte, die noch keine GPS Empfänger besitzen, könnte eine Integration des an der Universität Wien entwickelten Systems Semapedia.org helfen, die Lokalität zu erkennen. Dabei werden Aufkleber mit codierten Adressen an Lokalitäten platziert und das Mobiltelefon kann über den eingebauten Fotoapparat die Adressen entschlüsseln und zuordnen (http://www.semapedia.org).

Ist es möglich, dass der PIM den Aufenthaltsort der BenutzerInnen erkennt, wäre ein interessantes Feature, wenn AnwenderInnen im System für alle BenutzerInnen zugängliche Informationen mit Lokalitäten verknüpfen könnten. Ein interaktiver Reiseführer ist nur eine mögliche Anwendung davon.

5.5. Optimierung des Registry Konzeptes

Das für den PIM entwickelte Registry Konzept ist einfach in der Entwicklung und ermöglicht ein schnelles Auffinden von Kontakten, es ist allerdings keine sehr elegante und effiziente Lösung. Eine Verbesserung des Konzepts wäre denkbar, indem die BenutzerInnen nur dann in Registries auf anderen Peers aufgenommen werden, wenn diese BenutzerInnen auf den entsprechenden Peers auch in Kontaktlisten stehen. Die Suche nach Kontakten wird damit zwar aufwendiger, da mehrere Peers durchsucht werden müssen, allerdings skaliert das System besser.

Zusammenfassung

Zur Verwaltung von persönlichen Aufgaben und Informationen verwenden Menschen zu viele Ressourcen. Bestehende Lösungen sind meist für spezielle Anwendungsfälle konzipiert und es fehlt daher oft an Flexibilität. Kooperationen zwischen BenutzerInnen sind dabei meist nicht berücksichtigt.

Ziel dieser Arbeit war die Entwicklung eines kollaborativen Aufgaben- und Informationsmanagers unter der Verwendung von XVSM, einer Virtual Shared Memory Middleware Technologie. Bei dem Manager handelt es sich um eine räumlich getrennte und zeitlich nahezu synchrone bis asynchrone Applikation. Die Lösung zeichnet sich durch Skalierbarkeit, Plattformunabhängigkeit und Sicherheit aus.

Der Mensch wird durch die Applikation in allen Lebensumständen unterstützt. Dafür wird ihm Aufgaben-, Ad-hoc-Workflow und Datenmanagement zur Verfügung gestellt. Um flexibel agieren zu können, gestattet der Manager die Verwaltung von Kategorien mit beliebigen Unterkategorien, die Verwaltung von Kontakten mit deren Gruppierung sowie die Verwaltung von Örtlichkeiten, welche wiederum beliebig unterteilt werden können. Auch die erstellten Aufgaben können in Unteraufgaben gesplittet und mit Attributen versehen werden. Zudem ermöglicht das System Aufgaben bzw. ganze Kategorien mit anderen Personen gemeinsam zu nutzen und alle Beteiligten jederzeit über den aktuellen Status der Aufgaben zu informieren.

Um die vernetzte Denkweise der Menschen zu unterstützen, erlaubt das System verschiedene Sichtweisen auf die getätigten Einträge. So kann beispielsweise die Suche über die oben genannten Elemente eingeschränkt und über deren Unterelemente verfeinert werden. Zusätzliche Einschränkungen sind über Stichworte, Erstellungszeitraum oder Attribute der Aufgaben möglich. Dabei führt jede Eingabe zu einem unmittelbaren Resultat und in Kombination führen sie zu dem gewünschten Ergebnis.

Zuletzt können die Einträge durch Attribute wie beispielsweise „Prioritäten“, „farbliche Markierungen“ und „Status“ – um nur einige zu nennen – klassifiziert werden. Das Aufgaben- und Informationsmanagement System stellt mit jeder Herausforderung eine benutzerfreundliche und anwendergerechte Applikation in den Vordergrund.

Literatur

3M 2008, *Post-it Digital Notes*, United States, St. Paul, besucht 1. Mai 2008, <http://www.3m.com/us/office/postit/digital/digital_notes.html>.

Basic Support for Cooperative Work 2008, *BSCW*, Deutschland, Bonn, besucht 8. Juni 2008, <http://www.bscw.de >.

Bellotti, V, Dalal, B, Good, N, Flynn, P, Bobrow, D G & Ducheneaut, N 2004, 'What a to-do: studies of task management towards the design of a personal task list manager', in *Proceedings of the SIGCHI conference on Human factors in computing systems*, ACM, Wien, pp. 735-742.

Belotti, V, Ducheneaut, N, Howard, M & Smith, I 2003, 'Taking email to task: the design and evaluation of a task management centered email tool', in *Proceedings of the SIGCHI conference on Human factors in computing systems*, Florida, USA, 05-10 April 2003, pp. 345-352.

Braun, I 2005, 'Entwurf und Modellierung einer universellen Telearbeitsumgebung auf Basis einer serviceorientierten Architektur', *Dissertation*, Institut für Informatik, Technischen Universität Dresden.

Brown, P J 1995, 'The electronic Post-it note: a model for mobile computing applications', *IEEE Colloquium on Mobile Computing and its Applications*, pp. 1/1-1/3.

CHIP Xonio Online GmbH 2008, *Immer und überall synchron*, Deutschland, München, besucht 8. Juni 2008, <www.chip.de/artikel/Live-Mesh-Microsoft-bastelt-am-Netz-im-Netz_31718355.html>.

Czerwinski, M, Horvitz, E & Wilhite, S 2004, 'A Diary Study of Task Switching and Interruptions', in *Proceedings of the SIGCHI conference on Human factors in computing systems*, ACM, Wien, pp. 175-182.

Dextronet – Smart Software 2008, *Swift To-Do List*, Tschechische Republik, Brno, besucht 3. September 2007, <http://www.dextronet.com/swift-to-do-list.php>.

Ducasse, S, Hofmann, T & Nierstrasz, O 2000, 'OpenSpaces: An Object-Oriented Framework for Reconfigurable Coordination', in *Proceedings of the 4th International Conference on Coordination Languages and Models*, 11-13 September 2000, pp. 1-18.

Eldridge, M, Sellen, A & Bekerian, D 1992, 'Memory Problems at Work: Their Range, Frequency and Severity', *Technical Report EPC-1992-129*, Rank Xerox Research Centre.

Elsweiler, D & Ruthven I 2007, 'Towards Task-based Personal Information Management Evaluations', in *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*, Amsterdam, 23-27 Juli 2007, pp. 23-30.

Erickson, T (2006), 'From PIM to GIM: Personal Information Management in Group Contexts', *Communications of the ACM*, vol. 49, no. 1, pp. 74-75.

Groove Networks 2007, *Groove Virtual Office*, United States, besucht 29. Juni 2007, <http://www.groove.net>.

Google 2008, *iGoogle*, United States, besucht 14. Juli 2008, <http://www.igoogle.com>.

Johansen, R 1988, *Groupware: Computer Support for Business Teams*, The Free Press, New York.

Kelly, D 2006, 'Evaluating Personal Information Management Behaviours and Tools', *Communikations of the ACM*, vol. 49, no. 1, Jänner, pp. 84-86.

Kreifelts, Th, Hinrichs, E & Woetzel, G 1993, 'Sharing To-Do lists with a Distributed Task Manager', in *Proceedings of the 3rd European Conference on Computer Supported Cooperative Work*, Mailand, 15-17 September 1993, pp. 31-46.

Kühn, E 1994 'Fault-Tolerance for Communicating Multidatabase Transactions', in *Proceedings of the 27th Hawaii International Conference on System Sciences*, ACM, IEEE, 4-7. Jänner, Wailea, Maui, Hawaii, 1994.

Kühn, E, Riemer, J, Mordinyi, R & Lechner, L 2008, 'Integration of XVSM Spaces with the Web to Meet the Challenging Interaction Demands in Pervasive Scenarios', *Ubiquitous Computing And Communication Journal*, special issue on "Coordination in Pervasive Environments", Vol. 3, März, 2008.

Lamming, M G, Brown, P, Carter, K, Eldridge, M, Flynn, M, Louie, G, Robinson, P & Sellen, A 1993, 'The Design of a Human Memory Prosthesis', *Technical Report EPC-1993-119*, Rank Xerox Research Centre.

Lamming, M G & Newman, W M 1992, 'Activity-based Information Retrieval: Technology in Support of Personal Memory', in Vogt, F H (ed), *Personal Computers and Intelligent Systems*, Eservier North-Holland, Amsterdam.

Microsoft 2008, *Live Mesh*, United States, besucht 09. Juli 2008, <www.mesh.com>.

Microsoft Office Online 2008, *Groove*, United States, besucht 09. Juli 2008, <http://office.microsoft.com/de-de/groove>.

Microsoft Office Online 2008, *SharePoint Services*, United States, besucht 09. Juli 2008, <http://office.microsoft.com/de-de/sharepointtechnology>.

Microsoft Research 2008, United Kingdom, besucht 09. Juli 2008, <http://research.microsoft.com/sendev/projects/sensecam/>.

MozartSpaces 2008, besucht 5. Mai 2008, <www.mozartspaces.org>.

NetBeans 2006, besucht 11. November 2006, <www.netbeans.org>.

Ohmukai, I & Takeda, H 2003, 'Social Scheduler: A Proposal of Collaborative Personal Task Management', *Proceedings of the IEEE/WIC International Conference on Web Intelligence (WI'03)*, pp. 666-670.

Riemer, J 2003, 'Peer-to-Peer Groupware mit CORSO', *Diplomarbeit*, Institut für Computersprachen, Technische Universität Wien.

Sellen, A, Fogg, A, Hodges, S & Wood, K 2007, 'Do life-logging technologies support memory for the past? An experimental study using SenseCam', *Proceedings on the SIGCHI Conference on Human Factors in Computing Systems*, San Jose, USA, 28. April – 03. Mai 2007, pp. 81-90.

Skype Limited 2008, Luxembourg, besucht 13. Dezember 2007, <www.skype.org>.

Takkinen, J & Shahmehri, N 1998, 'Coordination in Message-Based Environments: Restructuring Internet E-Mail to Accomplish Tasks', in *Proceedings of the 9th International Workshop on Database and Expert Systems Applications*, IEEE Computer Society, 26-28 August 1998, pp. 566-571.

Teich, N 2006, 'Einführung in kollaborative Systeme und deren Grundlagen', *Script zum Proseminar Kollaborative Web-Anwendungen*, Fakultät für Informatik, Technische Universität Dresden, besucht 13. März 2008, <www.mmt.inf.tu-dresden.de/Lehre/Wintersemester_06_07/ Proseminar/Praesentationen/01.Teich.pdf>

Tulving, E 1983, *Elements of Episodic Memory*, Oxford University Press.

Wernhart, H 1998, 'Request/Answer Coordination Design Pattern – Coordination Design', *Technischer Bericht*, Institut für Computersprachen, Technische Universität Wien.

Whittaker, S, Bellotti, V & Gwizdka, J 2006, 'Email in Personal Information Management', *Communications of the ACM*, vol. 49, no. 1, January 2006, pp. 68-73.

Wikipedia – Die freie Enzyklopädie 2008, *Ad-hoc-Workflow*, United States, San Francisco, besucht 1. Mai 2008, <http://de.wikipedia.org/wiki/Ad-hoc-Workflow>.

Wikipedia – Die freie Enzyklopädie 2008, *BSCW*, United States, San Francisco, besucht 9. Juni 2008, <http://de.wikipedia.org/wiki/BSCW>.

Wikipedia – Die freie Enzyklopädie 2008, *Groupware*, United States, San Francisco, besucht 1. Mai 2008, <http://de.wikipedia.org/wiki/Groupware>.

Wikipedia – Die freie Enzyklopädie 2008, *Microsoft Exchange Server*, United States, San Francisco, besucht 1. Mai 2008, <http://de.wikipedia.org/wiki/Microsoft_Exchange_Server>.

Wikipedia – Die freie Enzyklopädie 2008, *Microsoft Groove*, United States, San Francisco, besucht 9. Juni 2008, <http://de.wikipedia.org/wiki/Microsoft_Groove>.

Wikipedia – Die freie Enzyklopädie 2008, *Personal Information Manager*, United States, San Francisco, besucht 1. Mai 2008, <http://de.wikipedia.org/wiki/Personal_Information_Manager>.

Wikipedia – Die freie Enzyklopädie 2008, *Virtual Office*, United States, San Francisco, besucht 1. Mai 2008, <http://en.wikipedia.org/wiki/Virtual_office>.

Wikipedia – Die freie Enzyklopädie 2008, *Microsoft Windows SharePoint Services*, United States, San Francisco, besucht 9. Juni 2008, <http://de.wikipedia.org/wiki/Windows_SharePoint_Services>.

Wittmann, M 2008, 'Application scenarios', *Dokumentation zu MozartSpaces*, Institur für Computer Sprachen, Teschnische Universität Wien, besucht 13. Juli 2008, <http://www.mozartspaces.org/1.0-alpha/docs/scenarios/MozartSpaces_Application_Scenarios.pdf>