

# **Seminarband zur Projektgruppe Kooperatives Wissensmanagement**

Heinz Nixdorf Institut  
Universität Paderborn

Informatik und Gesellschaft  
Prof. Dr. Ing. Keil-Slawik\*

15.6 - 16.6.2002

\*Leitung: Ludger Merkens

# Inhaltsverzeichnis

<b>1</b>	<b>Virtuelle Zusammenarbeit - ein Zukunftsausblick</b>	<b>3</b>
1.1	Formen der virtuellen Zusammenarbeit . . . . .	3
1.1.1	Virtuelle Zusammenarbeit mit CVW . . . . .	3
1.1.2	Beispiel Mobile Computing . . . . .	5
1.2	Anforderungen an mobile Devices . . . . .	6
1.2.1	unsichtbare Computer und anpassungsfähige Oberflächen . . . . .	7
1.2.2	Aufgabenorientiertheit . . . . .	7
1.2.3	intuitive Bedienbarkeit und Wartbarkeit . . . . .	8
1.2.4	Portabilität . . . . .	10
1.3	Anforderungen an das Gesamtsystem . . . . .	11
1.3.1	Modularisierung von Services . . . . .	11
1.3.2	spezielle Anforderungen durch nur zeitweilig bestehende Netzwerkverbindungen . . . . .	12
1.3.3	Datenzentriertheit . . . . .	13
1.3.4	Zugriffsschutz . . . . .	13
1.3.5	Schnittstellen zwischen Services . . . . .	14
1.4	Fazit . . . . .	14
1.5	Literatur . . . . .	15

# 1 Virtuelle Zusammenarbeit - ein Zukunftsausblick

## 1.1 Formen der virtuellen Zusammenarbeit

Sich treffen, Dokumente austauschen, miteinander reden und gemeinsam arbeiten ohne sich jemals tatsächlich zu sehen, sich sogar von einem beliebigen Ort auf der Welt online in einen Arbeitsprozeß einklinken zu können, das ist das Ziel von sogenannten kooperativen virtuellen Arbeitsplätzen und mobilen Geräten. Diese Seminararbeit behandelt die Möglichkeiten, die sich durch solche Systeme ergeben sowie die Anforderungen, die sie an Hardware und Software stellen.

### 1.1.1 Virtuelle Zusammenarbeit mit CVW

Ein Beispiel für virtuelle Zusammenarbeit ist das Cooperative Virtual Workspace-System, das als Open Source Projekt von der Firma MITRE entwickelt wurde.

Im CVW-System treten computerbasierte soziale Interaktionen an die Stelle von realen Zusammentreffen. Die verschiedenen Benutzer eines CVW-Systems sind räumlich getrennt, sitzen beispielsweise in verschiedenen Büros, und treffen sich in einem gemeinsamen, virtuellen Raum. Hier können sie sich unterhalten und gemeinsam Dokumente bearbeiten, ohne daß sie wissen müssen, wo sich ihre Gesprächspartner überhaupt befinden. Genau wie in der Realität ist es in CVW möglich, Objekte mitzunehmen und woanders wieder abzulegen oder sie einer anderen Person zu übergeben.

Anders als herkömmliche Kommunikationswerkzeuge im Netz, wie beispielsweise Netmeeting oder ICQ ist CVW nicht sessionzentriert sondern dokumentenzentriert. Zwischen diesen beiden Formen von Systemen gibt es wesentliche Unterschiede: Bei einem sessionzentrierten Kommunikationsprogramm ist es zwar genau wie bei CVW möglich mit mehreren Personen zusammenzuarbeiten, allerdings ist, nach Beendigung der Kommunikation bei einem erneuten Einloggen nichts mehr von der vorherigen Arbeit zu sehen.

Bei einem dokumentenzentrierten Werkzeug wie CVW hingegen, werden im Laufe einer Zusammenarbeit Dokumente erzeugt, die für spätere Zugriffe persistent im System abgelegt werden und bei einem erneuten Einloggen wieder verfügbar sind.

Trotzdem gehen natürlich durch die reine Schriftform insbesondere die nonverbalen Anteile der Kommunikation, also Gestik und Mimik verloren. Um das zu kompensieren, werden in CVW Audio und Video als zusätzliche Medien eingesetzt. Außerdem können Dokumente und Bilder erzeugt und den Gesprächspartnern gezeigt oder übergeben werden,

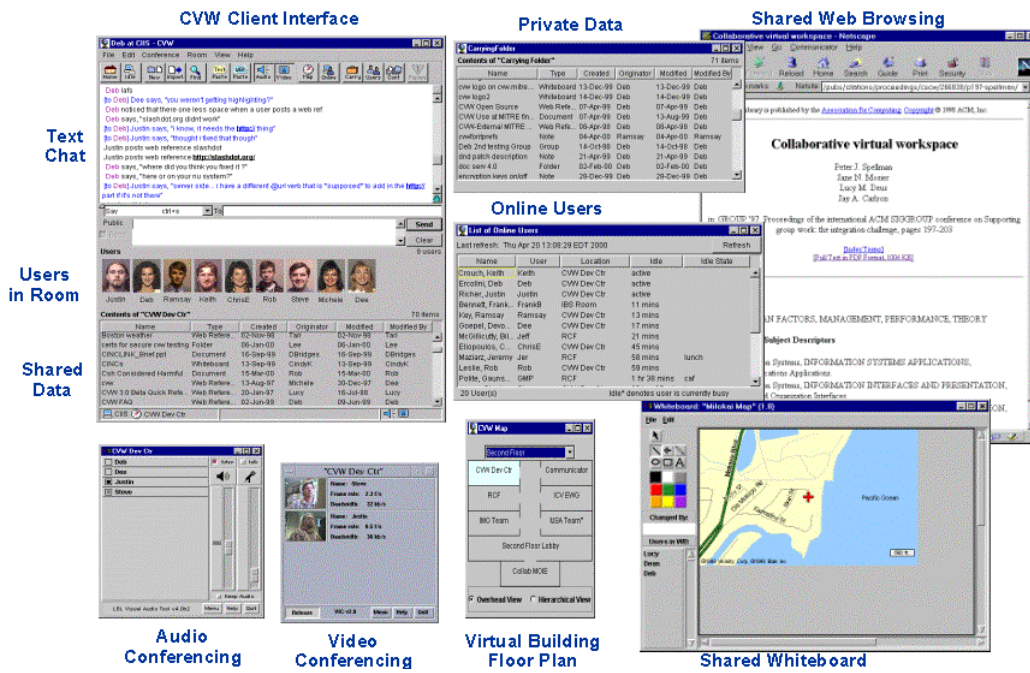


Abbildung 1.1: Grafische Oberfläche von CVW

um Sachverhalte zu visualisieren. Insgesamt besteht CVW aus Audio- und Videoconferencing-Tools, einem Chat, einem Whiteboard und einem Dokumentmanagementsystem. Diese Tools sind unter einer Grafischen Oberfläche, wie sie in Abbildung 1.1 zu sehen ist, integriert.

Die Vielzahl verschiedener Kommunikationswerkzeuge ist auch deswegen von Vorteil, weil zum Kennenlernen der Benutzer untereinander ganz unterschiedliche Tools nützlich sind, als für die produktive gemeinsame Arbeit an einem Dokument. So bietet sich zum Kennenlernen eher eine Kommunikation in Echtzeit an, wie Audio- und Videoconferencing, während zum Zusammenarbeiten Whiteboard und Dokumentmanagement aber auch eMails besonders von Bedeutung sind.

Die Grundidee des CVW-Systems basiert im Wesentlichen auf dem Konzept der MUDs und MOOs. MUD bedeutet Multi User Dungeon und bezeichnet textbasierte Multiplayer-Rollenspiele im Internet. MOOs sind objektorientierte MUDs.

CVW besitzt genau wie ein MUD verschiedene Räume, die durch Gänge verbunden sind. Innerhalb der Räume können sich Personen aufhalten und außerdem können sich hier verschiedene Objekte befinden. Objekte sind beispielsweise Texte, Zeichnungen oder Audiodateien. Hierbei ist wichtig, daß Objekte, genau wie in der Realität persistent sind. Das bedeutet, ein Objekt befindet sich zu einem Zeitpunkt immer in genau einem Raum und verschwindet auch dann nicht, wenn sich kein Benutzer im gleichen Raum befindet.

Außerdem verwaltet CVW Rechte für Räume und Objekte, die festlegen, welche Benutzer Zugriff darauf haben. Diese Rechte kann der Administrator für das gesamte System festlegen. Zusätzlich kann jeder, der einen Raum oder ein Objekt erzeugt, bestimmen,

welche Benutzer Zugriff darauf haben sollen, welche von ihnen sich ein Objekt ansehen und welche auch selbst etwas verändern dürfen.

Um die Zugriffsmöglichkeiten flexibler zu gestalten, können Objekte importiert und exportiert werden, um sie lokal auf dem eigenen Rechner zu bearbeiten. Hierfür ist es notwendig, Objekte beim wieder-Importieren mit dem bereits im System vorhandenen Objekt abzugleichen, also in erster Linie zu überprüfen, ob ein Dokument zwischen dem Export und dem Import der überarbeiteten Version von einem anderen Benutzer verändert wurde und gegebenenfalls dafür zu sorgen, daß die Versionen sinnvoll zusammengefügt werden.

Um dies zu ermöglichen, werden Dokumente in CVW auf einem separaten Server, dem Document Server gespeichert und mit Hilfe von Links in das System eingebunden.

Eine Zusammenarbeit in CVW könnte beispielsweise so ablaufen: Betty und John arbeiten gerade in Johns Arbeitsraum an einem gemeinsamen Paper für einen Vortrag. Hierbei spielt es keine Rolle, daß John gerade in seinem Büro in London sitzt, während Betty sich auf keinem Kongress in New York befindet. Sie diskutieren via Videoconferencing, welche Themen sie im Einzelnen vorstellen möchten, während John gleichzeitig ein Dokument anlegt, in dem die wichtigsten Inhalte ihres Gesprächs festgehalten werden. Bob betritt den Raum, sieht sich Johns Entwurf an, schreibt einige Annotationen an den Text und gibt Betty die Dokumentation eines Projekts aus dem letzten Jahr, die die beiden in ihre weitere Planung mit einbeziehen können. Das alles ist möglich, ohne daß Betty, John und Bob sich jemals real begegnet sein müssen.

### 1.1.2 Beispiel Mobile Computing

Noch weiter als CVW soll in Zukunft das Konzept des Mobile Computing gehen. Beim Mobile Computing sind die Benutzer des Systems nicht mehr an ihr Büro und ihren Rechner gebunden sondern besitzen eine Reihe auf bestimmte Aufgaben spezialisierter, mobiler Devices: zum Beispiel einen Terminplaner, der automatisch über ein Netzwerk Daten mit den Terminplanern der Kollegen abgleicht, eine Digitalkamera, die Bilder automatisch auf einem Server im Internet zur Verfügung stellt, und Handgeräte mit einem Display, ähnlich wie unsere heutigen PDAs, über die der Benutzer mobil im Internet surfen kann.

Mobile Geräte der Zukunft werden untereinander vernetzt sein und online auf gemeinsame Daten zugreifen. Im Terminplaner auf den Rechnern zu Hause und im Büro sind also beispielsweise immer die gleichen Termine eingetragen, wie auch auf dem Handgerät. Von der Digitalkamera können Bilder über die Netzwerkverbindung auf das Handy und mit dessen Hilfe ins Internet übertragen werden.

Je nachdem, welche Art von Netzwerk gerade verfügbar ist, werden die zu übertragenden Daten angepasst. Besteht eine schnelle Internetverbindung, so können große Datenmengen übertragen werden. Ist lediglich eine teure Mobilfunkverbindung verfügbar, so werden die Datenmengen entsprechend eingeschränkt.

Anders als heutige Computer, die eine Vielzahl verschiedener Funktionen ausführen können, werden mobile Devices stark auf einzelne Aufgaben, wie beispielsweise Kommunikation, Terminverwaltung, Spielen, etc. spezialisiert sein. Das wird es einfacher machen Aufga-



Abbildung 1.2: Entwurf eines Nokia-Smartphones

ben effizient, korrekt und ohne besondere Vorkenntnisse auszuführen.

Hierzu trägt auch die Gestaltung der Benutzerinterfaces bei, die nicht mehr notwendigerweise auf einem Computerbildschirm dargestellt werden. Stattdessen ist das Gerät selbst das Benutzerinterface, wie es bei eingebetteten Systemen wie beispielsweise dem von Sony entwickelten Roboterhund Aibo schon heute der Fall ist.

Eine weitere Vereinfachung für den Benutzer stellt der Einsatz von Agententechnologie dar. Sie ermöglicht es, daß der Benutzer keine expliziten Befehle mehr geben muß, sondern lediglich grobe Anweisungen an einen Agenten erteilt der dann selbständig im Auftrag seines Benutzers Aufgaben erfüllt. Agenten werden beispielsweise schon heute auf virtuellen Marktplätzen eingesetzt. Hier gibt ein Verkäufer für seine Ware einen maximalen und einen minimalen Verkaufspreis ein, und der Agent versucht unter Beachtung dieser Angaben die Waren möglichst gewinnbringend zu verkaufen.

Ein erster Schritt auf dem Weg zur Entwicklung von intelligenten, mobilen Geräten, der bereits in der Praxis umgesetzt wurde, sind die sogenannten Smartphones, eine Mischung zwischen Handy und PDA. Hier werden sich durch die Einführung von UMTS anstelle des bisher verwendeten GSM-Standards weitere neue Funktionalitäten und Möglichkeiten ergeben. In Abbildung 1.2 ist der Entwurf eines Nokia-Smartphones zu sehen.

## 1.2 Anforderungen an mobile Devices

Wie aus der Beschreibung der Grundideen von mobilen Devices bereits zu erahnen ist, unterscheiden sich diese Geräte deutlich von heutigen Computern. Hieraus ergibt sich auch ein völlig neues Anforderungsprofil an das Design von Hardware und Software.

## 1.2.1 unsichtbare Computer und anpassungsfähige Oberflächen

Damit mobile Geräte eine breite Akzeptanz finden, ist es notwendig, daß sie sich in ihrem Aufbau grundlegend von den heutigen, kompliziert zu bedienenden Computern unterscheiden und mehr Ähnlichkeit zu anderen, intuitiv bedienbaren technischen Geräten bekommen. Der Benutzer soll im Idealfall nicht merken, daß er mit einem Computer arbeitet, weil sich die Bedienung des mobilen Devices nicht wesentlich von der Bedienung eines einfachen Fernsehers oder eines Haushaltsgeräts unterscheidet

Das bedeutet beispielsweise, daß die Bedienung nicht mehr über Menüs auf einem Bildschirm geschieht, sondern daß viele Geräte vollständig auf einen Bildschirm verzichten können, und über Tasten auf dem Gehäuse, Handschrift, Gestenerkennung oder Sprache bedienbar sind.

Besondere Anforderungen an das System stellen Dienste, die von völlig unterschiedlichen Geräten abrufbar sein sollen. So möchte ein Benutzer auf seinen Kalender beispielsweise einmal sprachgesteuert über das Telefon zugreifen, einmal über einen PDA, der aufgrund seines kleinen Displays nur eingeschränkte Darstellungsfunktionen bietet, und einmal über einen großen Rechner im Büro, der eine komplexere Darstellung erlaubt. Der gleiche Inhalt soll also in völlig unterschiedlichen Formen präsentiert werden. Damit diese Anpassung des Inhalts an die verschiedenen Darstellungsmedien möglich wird, muß eine scharfe Trennung von Inhalt und Darstellung erfolgen. Ansätze hierzu finden sich beispielsweise in der Seitenbeschreibungssprache HTML, die ursprünglich dazu konzipiert war, lediglich Textauszeichnungen wie beispielsweise Überschrift oder Auflistung in den Dokumentenquelltext zu übernehmen, und dem Browser die Aufgabe zu überlassen, hieraus eine sinnvolle Bildschirmdarstellung zu generieren.

Eine solche Aufteilung in Inhalt und Gestaltung ist zum einen deswegen sinnvoll, weil die Menge der zu übertragenden Daten geringer wird, da keine zusätzlichen Informationen zur Darstellung übermittelt werden und ermöglicht außerdem, neue Geräte mit möglicherweise völlig neuen Darstellungsformen zu entwickeln, ohne daß die Dokumente, die dargestellt werden sollen, zu modifizieren sind.

Um größere Flexibilität zu gewährleisten sind einige Erweiterungen des HTML-Konzepts sinnvoll, die in der Sprache XML umgesetzt sind. Während in HTML nur ein vorher festgelegter Befehlssatz verwendet werden kann, ist es in XML möglich, eine neue Sprache mit einem eigenen Befehlssatz zu entwickeln, die für ein spezielles Gerät besser geeignet ist, als HTML.

## 1.2.2 Aufgabenorientiertheit

Eng mit der Forderung nach unsichtbaren Computern und anpassbaren Oberflächen verknüpft ist die Forderung nach Aufgabenorientiertheit. Das Design von Hardware und Software sowie das Bedienkonzept muß auf die Aufgabe zugeschnitten sein, die das Gerät erfüllen soll. Sowohl die Hardware als auch die Software können im Rahmen der technischen Machbarkeit beliebig gewählt werden und werden vollständig anhand der Aufgabe konzipiert.

Das ist möglich, weil mobile Devices keine Universalgeräte mehr sein müssen, die in der Lage sein sollen, mit beliebigen Aufgaben umzugehen. Stattdessen werden sie auf die möglichst gute Erledigung einer sehr kleinen Zahl von Aufgaben optimiert. Ein Terminplaner oder ein PDA kann beispielsweise, um besser tragbar zu sein, ein sehr kleines Gehäuse besitzen, da er auf Geräte wie Festplatte, Lüfter, Steckkarten etc. die einen heutigen Rechner ausmachen, verzichten kann.

Unter dem Kriterium der Aufgabenorientiertheit unterscheiden sich zukünftige mobile Geräte daher von heutigen Computern. Bei Letzteren orientiert sich die Entwicklung neuer Software nicht nur an der Aufgabe sondern muß auch berücksichtigen, welche Constraints durch Hardware und bereits vorhandene Software, wie beispielsweise das Betriebssystem gegeben sind. Diese Rahmenbedingungen gehen in die Planung ein, wie und in welchem Umfang eine Aufgabe erledigt werden kann. Beim Design eines mobilen Geräts sind diese Überlegungen in der Regel nicht notwendig.

### 1.2.3 intuitive Bedienbarkeit und Wartbarkeit

Ein weiterer Punkt, der für die Akzeptanz von mobilen Geräten von herausragender Bedeutung ist, betrifft die Bedienbarkeit. Damit ein Benutzer ein Gerät annimmt, muß er in der Lage sein, ohne langwieriges Studium des Handbuchs damit umzugehen.

Das Verhalten des Gerätes ist sollte weitgehend automatisiert sein und mit minimal wenigen Eingriffen durch den Benutzer funktionieren. Zusätzlich muß einem fortgeschrittenen Benutzer natürlich die Möglichkeit gegeben werden, einzugreifen und bestimmte Automatikfunktionen abzuschalten und durch manuelle Bedienung zu ersetzen.

Eine andere Hürde für jemanden, der sich heute zum ersten Mal mit dem Computer beschäftigt, ist der ungewohnte Umgang mit Tastatur und Maus. Deswegen wird bei zukünftigen mobilen Geräten die Tastatur weitgehend durch Sprach- und Handschrifterkennung ersetzt, da die Techniken des Sprechens und Schreibens anders als die Tastaturbedienung jedem potentiellen Benutzer vertraut sind.

Mobile Geräte sollen allerdings nicht nur in der alltäglichen Benutzung einfach zu bedienen sein, sondern es ist ebenfalls wichtig, daß ein Benutzer ohne Fachkenntnisse in der Lage ist, selbständig neue Funktionen, die als Hardware- oder Softwaremodule dazugekauft werden, in das Gerät zu integrieren. Damit dies möglich wird, muß die Einbindung neuer Software und das Erkennen der Hardware weitgehend automatisiert werden, so daß der Benutzer nur über wichtige Vorfälle informiert wird und nur in Ausnahmefällen selbst eingreifen muß.

Für den Benutzer trägt es wesentlich zu einer intuitiveren Bedienung bei, wenn er Funktionen nicht über explizite Befehlsaufrufe starten und beenden muß, sondern der Computer aus dem Verhalten des Benutzers und aus Informationen über dessen Aufenthaltsort wie Position, Beleuchtung, Geräuschpegel, Netzwerkverfügbarkeit, Kommunikationskosten und -bandbreite die wahrscheinlichste Intention des Benutzers ermittelt, und daraufhin selbständig bestimmt, welche Funktionen aufzurufen sind.

Diese Anpassungsfähigkeit an die Wünsche des Benutzers wird auch als Context Aware Computing bezeichnet. Als Kontext werden hauptsächlich die Informationen wo sich ein Benutzer befindet, mit wem er zusammen ist, und welche Ressourcen in der Nähe



verfügbar sind, aufgefasst. Context Aware Computing-Anwendungen unterteilen sich im Wesentlichen in vier Gebiete:

- *Ortsbasierte Auswahl:* Hier wird abhängig von der aktuellen Position auf einer Karte eine Auswahlliste oder eine andere Darstellung von Objekten in der Umgebung für den Benutzer so angeordnet, daß die nächsten Objekte hervorgehoben sind. Ein Beispielszenario, in dem Ortsbasierte Auswahl nützlich ist, ist folgende: Ein Dokument soll gedruckt werden. Es stehen verschiedene Drucker in verschiedenen Räumen zur Auswahl. Im Auswahldialog steht dann derjenige Drucker ganz oben, der am wenigsten weit vom Benutzer entfernt ist. Dadurch wird dem Benutzer eine Hilfestellung bei der Auswahl gegeben. Er kann dieser Hilfestellung folgen, so daß er nicht weit laufen muß, um seine Druckseiten abzuholen, kann die Hilfe aber auch ignorieren, beispielsweise weil er auf einem der Drucker kostenlos drucken kann, oder weil er Funktionen benötigt, die nicht alle Drucker besitzen.
- *Automatische Rekonfiguration:* Bei der automatischen Rekonfiguration werden Komponenten wie Hardwaretreiber, Softwaremodule, Programme oder Netzwerkverbindungen automatisch kontextabhängig hinzugefügt, gestartet oder entfernt. Eine ganz einfache Beispiel für automatische Rekonfiguration sind beispielsweise die unterschiedlichen Stromsparmodi eines Laptops. Hängt der Laptop an einer Stromversorgung, kann er permanent eingeschaltet bleiben. Arbeitet er hingegen gerade mit Batteriestrom, so schaltet er nach kurzer Zeit der Inaktivität in den Energiesparmodus, in dem das Gerät weniger Strom verbraucht. Wenn die freigegebenen Verzeichnisse von anderen Rechnern automatisch als Netzlaufwerke in das eigene Dateisystem eingebunden werden, sobald man sich ihnen nähert, so handelt es sich ebenfalls um automatische Rekonfiguration.
- *Kontextabhängige Information:* Wie der Name es schon vermuten lässt, sind kontextabhängige Informationen solche Informationen, die der Benutzer je nach Situation oder Aufenthaltsort vom System bekommt, also beispielsweise Bedienungsanleitungen in Kurzform, wenn er sich einem technischen Gerät nähert oder auch Produktinformationen, wenn er sich im Geschäft ein Gerät ansieht.
- *Kontextabhängige Befehle:* Wenn nicht nur Informationen angezeigt werden sondern auch die Ausführung von Befehlen durch die Umgebung beeinflusst wird, sind kontextabhängige Befehle im Spiel. Hier gibt es zwei verschiedene Typen - Zum einen können Befehle überhaupt erst ausführbar werden, wenn sie in einem Kontext sinnvoll sind, zum anderen gibt es Befehle, die je nach Kontext unterschiedlich arbeiten. Der erste Fall kann beispielsweise in einem Chat-Programm auftreten, wo man erst dann die Möglichkeit hat, Nachrichten an einen Benutzer zu senden, wenn dieser wirklich online ist. Die zweite Möglichkeit wäre beispielsweise ein

Befehl „Netzwerkverbindung aufbauen“, der je nach vorhandener Netzversorgung die günstigste mögliche Verbindung aufbaut

- *Kontextgetriggerte Aktionen:* Das Prinzip der kontextgetriggerten Aktionen verfolgt einen regelbasierten Grundansatz, der aus dem Bereich der wissensbasierten Systeme bekannt ist. Im Prinzip basieren kontextgetriggerte Aktionen auf einfachen IF-THEN-Statements, wie zum Beispiel IF „Benutzer betritt Wohnzimmer AND es ist dunkel THEN schalte Licht ein“ oder „IF Benutzer verlässt Küche AND Herd eingeschaltet, THEN melde Warnsignal“.

### 1.2.4 Portabilität

Eine weitere Dimension von Anforderungen ergibt sich durch den Wunsch nach portablen Geräten. Selbst einen heutigen Laptop kann man kaum in jeder Situation griffbereit haben, da er zum einen recht groß und schwer ist und zum anderen kaum länger als vier Stunden mit einer Batterieladung auskommt.

Damit ein Benutzer ein portables Gerät tatsächlich immer bei sich tragen kann, ist es daher notwendig, die Größe soweit wie es möglich und nicht der Bedienbarkeit abträglich ist, zu verringern und gleichzeitig den Stromverbrauch abzusenken. Außerdem ist die Entwicklung stärkerer und kleinerer Batterien notwendig um auch für sehr kleine Geräte eine möglichst lange Laufzeit sicherzustellen.

Neben diesen offensichtlichen Anforderungen an die Hardware muß aber auch die Software angepasst werden. Während ein Computer in der Regel nur gegen unbefugten Zugriff über ein Netzwerk oder in einem gemeinsamen Büro geschützt werden muß, kann es bei einem portablen Gerät auch passieren, daß es entwendet oder liegengelassen wird. Deswegen sollte es für jeden anderen als den Besitzer unbrauchbar sein. Neben dem klassischen Verfahren der Authentifizierung durch Username und Passwort stehen hier auch biometrische Verfahren wie die Fingerabdruck- oder Retinaerkennung zur Benutzeridentifikation in der Diskussion.

Für bestimmte Anwendungen, sogenannte Location Based Services, die eng mit Context Aware Computing verwandt sind, muß ein Gerät die Fähigkeit besitzen, seine Position im Raum oder auf einer Landkarte zu bestimmen. Die Positionsbestimmung kann beispielsweise über ein GPS-System erfolgen, aber auch über im Raum verteilte Sender/Empfänger denen sich ein Benutzer nähern und sich von ihnen entfernen kann.

Eine Anwendung, die bereits jetzt verbreitet ist, sind Navigationssysteme, die einem Autofahrer anhand einer Landkarte, einer vorab berechneten Route und den aktuellen Positionsdaten Anweisungen geben, in welche Richtung er an der nächsten Kreuzung fahren soll. Eine andere Möglichkeit wird derzeit im Bereich der Werbung getestet. Hier werden Angebote, von Geschäften in der Nähe direkt per SMS auf das Handy eines Benutzers geschickt

Zusätzlich ist es für bestimmte Anwendungen auch von Interesse, die Positionen von anderen Geräten in Reichweite zu kennen und alle Geräte zu finden, die sich in Reichweite

befinden. Zu erreichbaren Geräten kann dann eine Verbindung aufgebaut werden, und die Geräte können untereinander Daten austauschen. Wenn der Terminplaner beispielsweise feststellt, daß er sich gerade in der Nähe des Zentralrechners befindet, kann er mit diesem seine Daten synchronisieren, so daß beide später auf dem aktuellen Stand sind.

## 1.3 Anforderungen an das Gesamtsystem

Die Forderungen aus den letzten Abschnitten bezogen sich im Wesentlichen darauf, was einzelne, mobile Geräte leisten müssen. Unterschiedliche Geräte können diese Leistungen auf ganz verschiedene Arten erbringen. Netzwerkverbindungen können von verschiedenen Geräten also beispielsweise genauso über Bluetooth wie über ein TCP/IP-basiertes Netz aufgebaut werden. Zusätzlich gibt es aber auch Anforderungen, die das Gesamtsystem, also das Netzwerk und die Koordination von mobilen Geräten betreffen. Sie werden in diesem Abschnitt behandelt.

### 1.3.1 Modularisierung von Services

Mobile Hardware wird, wie bereits beschrieben, stark auf bestimmte Anwendungen spezialisiert sein. Der Benutzer kauft kein Universalgerät mehr, sondern wählt stattdessen nur die Einzelgeräte aus, die er braucht.

Analog hierzu sind auch die vom Netzwerk zur Verfügung gestellten Services modular aufgebaut. Der Benutzer wählt verschiedene Module aus, um ein genau auf ihn zugeschnittenes Gesamtsystem zusammenzustellen und zahlt nur für das, was er wirklich braucht.

Hierbei sind Schnittstellen für den Datenaustausch zwischen Modulen standardisiert, so daß unterschiedliche Module auch von verschiedenen Anbietern bereitgestellt werden können. Anwendungen dieses Typs werden als horizontal aufgebaut bezeichnet. Durch die exakte Anpassung an den Benutzer sind sie flexibel und kostengünstig.

Heute dagegen sind Services in der Regel vertikal aufgebaut. Das bedeutet, der Anwender steht vor der Frage, entweder ein ganzes Paket von Anwendungen zu kaufen oder ganz auf den Kauf zu verzichten. Diese Lösung stellt sich dann als ungünstig heraus, wenn der Benutzer entweder nur einen kleinen Teil der Leistungen sinnvoll nutzen kann oder zusätzliche Services hinzunehmen möchte, die in der aktuellen Lösung nicht enthalten sind und nur von einem anderen Anbieter zur Verfügung gestellt werden. Häufig ist es nicht möglich, Services von verschiedenen Anbietern miteinander zu kombinieren.

Wenn ein Kunde beispielsweise bei einem der großen Anbieter eine Domain registrieren lassen möchte, so bekommt er diese im Regelfall zusammen mit Webspace, mehreren Mailadressen, vorgefertigten CGI-Scripts, Zugriffsstatistiken und weiteren Funktionen. In der Regel ist es nicht möglich, die Mailpostfächer zu einer Domain bei einem anderen Anbieter hosten zu lassen, und für die CGI-Scripts ebenfalls einen anderen Anbieter auszuwählen und dafür auf diese Features im Paket zu verzichten.

### **1.3.2 spezielle Anforderungen durch nur zeitweilig bestehende Netzwerkverbindungen**

Wenn ein Gerät vom Benutzer überallhin mitgenommen werden kann, bedeutet das natürlich auch, daß es in ganz unterschiedlichen Umgebungen funktionieren muß. Insbesondere die Art und Qualität der Netzwerkverbindung unterscheidet sich innerhalb verschiedener Umgebungen und auch die Art der Verbindung zwischen unterschiedlichen Geräten kann unterschiedlich sein. Ein Handy verbindet sich beispielsweise via Bluetooth mit einem PDA, und wiederum über ein Mobilfunknetz mit dem Internet. Geräte können außer Reichweite geraten, oder die Internetverbindung über das Hausnetzwerk wird unterbrochen, wenn der Benutzer das Haus verläßt. Deswegen muß ein mobiles Gerät in der Lage sein, mit spontanen Verbindungsabbrüchen umzugehen, und selbständig festzustellen, was für eine Art von Netzwerk mit welchen Endgeräten zur Verfügung steht, um Verbindungen aufzubauen.

Insbesondere wenn die Netzwerkverbindung nur eine geringe Bandbreite hat oder teuer ist, bieten sich Kommunikationsformen an, bei denen die Verbindung nur im Bedarfsfall aufgebaut wird. Solche Verbindungen stellen allerdings ganz neue Anforderungen an die Datenübertragung und das Routing im Netz:

Das Netz und die verschiedenen Geräte müssen robust gegenüber dem Ausfall von Teilnetzen sein. Für einzelne Geräte bedeutet das, daß eine Datenübertragung während einer Unterbrechung des Netzzugangs sauber unterbrochen und an der gleichen Stelle weitergeführt werden kann.

Für das gesamte Netz spielt die Forderung nach Robustheit im Wesentlichen im Bezug auf die ad-hoc-Berechnung des Routings, das mangels permanent im Netz vorhandener Hosts, für jedes Datenpaket, das von A nach B geschickt werden soll, unterschiedlich sein kann, und das Vermeiden von Schaden durch Angriffe auf das Netz eine Rolle.

Die Fähigkeit, mit nur zeitweilig bestehenden Verbindungen umzugehen, ist auch deswegen sinnvoll, weil es auf diese Art und Weise möglich ist, die Verbindungskosten zu optimieren, da so immer die günstigste gerade zur Verfügung stehende Netzwerkverbindung ausgewählt werden kann. Der Zugriff auf den Zentralrechner im Haus muß beispielsweise über eine teure Mobilfunkverbindung erfolgen, wenn sich der Benutzer gerade auf dem Weg von der Arbeit nach Hause befindet, kann aber beim Betreten des Hauses auf das Hausnetzwerk umgeleitet werden.

Damit der Benutzer möglichst nicht durch eine derartige Realisierung der Netzwerkverbindung gestört wird, muß der Verbindungsauf- und -abbau automatisch erfolgen. Es sollte nicht notwendig sein, daß der Benutzer das Gerät jedes Mal, wenn er Daten übertragen möchte, explizit zum Aufbau einer Netzwerkverbindung auffordert. Ebenso wenig sollte er explizit mitteilen müssen, welche Netzwerkverbindung gerade die Günstigste ist.

Obwohl Automatisierung in vielen Fällen hilfreich ist, sollte es dem Benutzer immer möglich bleiben, selbst einzugreifen, um beispielsweise eine Verbindung ins Netz ganz zu verhindern oder zu bestimmen, daß eine spezielle Netzwerkverbindung verwendet werden soll, also daß beispielsweise für die Übertragung großer Datenmengen nicht die günstigste Verbindung ausgewählt wird, sondern stattdessen eine möglichst Breitbandige mit einer hohen Übertragungsgeschwindigkeit.

### 1.3.3 Datenzentriertheit

Der letzte Abschnitt beinhaltete bereits die Forderung, daß das Routing in Netzwerken mit mobilen Geräten sich von dem in herkömmlichen Netzen unterscheiden muß. Ein Datenpaket weiß, wenn es abgesendet wird, möglicherweise noch gar nicht, wer genau sein Empfänger ist, sondern lediglich welche Funktionen sein Empfänger erfüllen muß. Die Datenpakete bewegen sich selbstgesteuert von Gerät zu Gerät bis sie schließlich ihren Zielort erreichen.

Um dies zu ermöglichen ist eine datenzentrierte Netzwerkarchitektur notwendig. Das bedeutet, daß Datenpakete einen aktiven Inhalt besitzen, der von Hosts im Netzwerk ausgeführt wird und der es dem Datenpaket erlaubt, selbständig die von ihm benötigten Ressourcen im Netzwerk zu verwalten und beim Erreichen eines Hosts neu zu ermitteln, welche Route es auswählt, um sein Ziel zu erreichen.

Die Fähigkeiten zum Verwalten von Ressourcen und dem Ausführen von Code erlauben einem Datenpaket, seinen Weg durch das Netz zu finden, ohne daß es dafür die Hilfe der Applikation benötigt, die es abgeschickt hat.

Um bei dieser Technik zu gewährleisten, daß herrenlose Datenpakete nicht ewig lange im Netz bleiben und weitergeleitet werden, muß jedes Paket mit einer endlichen Lebensdauer, entweder in Form eines Zeitstempels oder in Form einer maximalen Routenlänge, ausgestattet sein. Da Routen in dynamischen Netzen immer nur wahrscheinlich aber nicht sicher zum Ziel führen, da sich das Netz in der Zwischenzeit ändern kann, ist es notwendig, Pakete an bestimmten Stellen zu duplizieren und alternative Routen einschlagen zu lassen, so daß wenigstens eins an seinem Ziel ankommt.

Um Daten konsistent zu halten, obwohl von verschiedenen Geräten an verschiedenen Orten darauf zugegriffen wird, ist außerdem von überall zugänglicher gemeinsamer Speicher erforderlich. Allerdings gibt es bisher noch kein Verfahren, daß die Konsistenz von Daten in Systemen mit nur zeitweilig bestehenden Internetverbindungen und daraus resultierenden Übertragungsfehlern, sicher gewährleistet. Auf diesem Gebiet ist deswegen weitere Forschung notwendig.

### 1.3.4 Zugriffsschutz

Wenn ein mobiles Gerät für seinen Benutzer die Verwaltung von Terminen übernimmt, so ist es sicher nicht im Sinne des Benutzers, daß beliebige Personen Einträge im Terminkalender vornehmen dürfen, während es aber möglicherweise erwünscht ist, daß gemeinsame Termine automatisch mit den Kalendern von Kollegen oder Familienmitgliedern abgeglichen werden. In einem mobilen Gerät muß daher ein Mechanismus vorhanden sein, der unbefugte Zugriffe aus dem Netz auf das Gerät selbst oder auf zentrale Daten zurückweist, während er bestimmten Benutzern Zugriff auf bestimmte Daten gestattet.

Ein solcher Authentifizierungsmechanismus ermittelt die Identität der Person, die Zugriff auf bestimmte Daten anfordert und überprüft dann, ob sie zugriffsberechtigt ist.

Derzeit wird für die Benutzerauthentifizierung im Netz häufig das Kerberos-System eingesetzt. Allerdings benötigt Kerberos statisch konfigurierte zentrale Server, die es für den Einsatz in mobilen Netzwerken ungeeignet machen.

Auch das IPSec-Protokoll, das auf einem Public Key-Verfahren basiert, ist nicht vollständig für die Verwendung in mobilen Netzwerken geeignet, da es keinen eigenen Authentifizierungsmechanismus bereitstellt.

Um Zugriffsschutz zu gewährleisten, müssen allerdings noch einige zentrale Probleme geklärt werden. Was passiert, wenn kein Authentifikationsserver erreichbar ist? Wie kann auch bei ständigem Verbindungsauf- und -abbau sichergestellt werden, daß sich kein Gerät als ein anderes beziehungsweise kein Benutzer als jemand anderes ausgibt? Diese Fragen sind zur Zeit noch Gegenstand der Forschung.

### 1.3.5 Schnittstellen zwischen Services

Wenn ganz unterschiedliche Geräte miteinander kommunizieren sollen, müssen sie sich darauf einigen, in welchem Format Daten zwischen ihnen übertragen werden. Dazu brauchen sie ein gemeinsames Interface, das genau bestimmt, welche Methoden mit welchen Parametern auf den verschiedenen Geräten ausgeführt werden können. Da ein Gerät nicht alle Schnittstellen für den Zugriff auf mögliche Kommunikationspartner kennen kann, wird ein Mechanismus gebraucht, über den ein Gerät von einem anderen dessen Schnittstellendefinition abrufen kann

Es gibt verschiedene Ansätze, dies zu realisieren. Java-Programme arbeiten zu diesem Zweck beispielsweise mit Suns proprietärem Remote Method Invocation API (RMI). Ein anderer Ansatz, der bereits in vielen verschiedenen Programmiersprachen implementiert ist, heißt CORBA. An der Uni Paderborn wurde für das Projekt sTeam ein ähnliches Protokoll mit dem Namen COAL entwickelt.

## 1.4 Fazit

Ziel der Entwicklung von mobilen Geräten und virtuellen Arbeitsräumen ist es, Die Arbeit in einem Netzwerk so optimal wie möglich an den Benutzer und die aktuelle Situation anzupassen. Virtuelle Arbeitsräume fokussieren hier insbesondere auf den Aspekt der zwischenmenschlichen Kommunikation und versuchen diese bestmöglich zu unterstützen. Mit Hilfe virtueller Arbeitsräume ergeben sich Möglichkeiten zur Telearbeit, zur Zusammenarbeit von verschiedenen Firmenstandorten aus und sogar zur Zusammenarbeit über Landesgrenzen hinweg. Zusammenarbeit wird unabhängig von einem gemeinsamen physikalischen Standort, was virtuelle Arbeitsräume insbesondere für Firmen mit weit voneinander entfernten Niederlassungen interessant macht, aber auch andere Anwendungen beispielsweise in der Lehre an einer Universität sind denkbar.

Es existieren bereits funktionsfähige Implementierungen des Konzepts der virtuellen Arbeitsräume. Eine davon ist das im ersten Abschnitt vorgestellte Programm CVW, ein anderes Programm mit einem ähnlichen Ansatz ist das an der Universität Paderborn entwickelte sTeam. Es dient dem Informationsaustausch zwischen Studenten und ihrer Kommuni-

kation untereinander. Mit Hilfe von sTeam können Studenten, die sich zu Hause oder in verschiedenen Teilen der Universität befinden, zusammen Übungsaufgaben diskutieren und bearbeiten, sich unterhalten oder gemeinsam an Dokumenten arbeiten.

Ein komplettes System von Mobilgeräten, wie hier vorgestellt, existiert bisher noch nicht. Es gibt lediglich Prototypen für einige der Einzelanwendungen wie beispielsweise des Context Aware Computing. Die wichtigsten Probleme, die bei der Entwicklung von mobilen Geräten beachtet werden müssen, sind:

- User-Interfaces, die sich nicht an expliziten Befehlsaufrufen sondern an den Intentionen des Benutzers orientiert sind. (Context Aware Computing)
- Horizontal aufgebaute, netzwerkbasierende Dienste statt „All-In-One-Lösungen“
- Datenzentrierte Netzwerke und eine Lösung der Probleme, die durch häufig unterbrochene Verbindungen entstehen

Es ist notwendig, in Zukunft weniger die Technologie selbst als viel mehr ihre Anwendungen und die späteren Benutzer ins Blickfeld zu nehmen, um Geräte aufgabenorientiert zu entwickeln und einem breiten Benutzerkreis zugänglich zu machen.

## 1.5 Literatur

Stephen Jones: The Cooperative Virtual Workspace

M. Esler, J. Hightower, T. Anderson, G. Borriello: Next Century Challenges: Data Centric Networking for Invisible Computing

P.J. Spellman, J.N. Mosier, L.M. Deus J.A. Carlson: Collaborative Virtual Workspace

Spektrum der Wissenschaft 6/2002: D.P. Anderson, J. Kubiawicz: Der Weltcomputer