

Universität Konstanz
FB Informatik und Informationswissenschaft
Bachelor-Studiengang Information Engineering

Bachelorarbeit

Mobiles Navigationssystem für MedioVis

*zur Erlangung des akademischen Grades eines
Bachelor of Science (B.Sc.)*

Studienfach: Information Engineering
Schwerpunkt: Mensch Computer Interaktion
Themengebiet: Angewandte Informatik

von

Christian Harr

(Matr.-Nr.: 01/499768)

Erstgutachter: *Prof. Dr. Harald Reiterer*
Zweitgutachter: *Prof. Dr. Rainer Kuhlen*
Betreuer: *Thorsten Buering*
Einreichung: *23. Juni 2006*

Zusammenfassung

Schlüsselworte: *Navigation, Navigationssystem, Ortung, Ortungsverfahren, Positionsbestimmung, kabellose Datenübertragung, ortsbezogene Dienste, PDA, 3D-Visualisierung, Medio Vis.*

Waren Navigationssysteme bis vor kurzem hauptsächlich in der Luft- und Seefahrt vertreten und bekannt, in Folge dessen auch den wenigsten Menschen im Gebrauch vertraut, so hat sich dies in den letzten Jahren stark verändert. Grund dafür ist die rasante Entwicklung im Bereich der KFZ-Navigation, welche inzwischen auch in immer mehr Autos zu finden ist, in naher Zukunft vermutlich sogar kaum noch wegzudenken sein wird. Die Präsenz in den Medien hat sicherlich dazu beigetragen, dass das Wissen über diese Systeme enorm gestiegen ist, weshalb die meisten Menschen inzwischen auch wissen, wie ein solches GPS-basiertes Navigationssystem grob aussieht und funktioniert. Der nächste Schritt in dieser Entwicklung wäre und ist nun, Navigationssysteme auch in Gebäuden einzusetzen.

In dieser Arbeit werden zunächst einige bereits bestehende Navigationsysteme vorgestellt, um dann im Folgenden ausführlich auf die benötigten technischen Grundlagen einzugehen. Da die Realisierung einer ansprechenden 3D-Visualisierung auf einem PDA Ziel dieses Projektes ist, befasst sich ein weiteres Kapitel auch mit den hierfür erforderlichen Designgrundlagen. Des Weiteren ist die Akzeptanz durch den Benutzer bei einem neu eingeführten System enorm wichtig, weshalb anschließend in einem benutzerorientierten Ansatz auf dessen Bedürfnisse eingegangen wird, gefolgt von der Vorstellung des zu diesem Projekt entwickelten Prototypen in Flash und SWIFT und der Anleitung zur vollständigen Umsetzung eines solchen Systems. Den Abschluss bildet dann noch ein kleiner Ausblick.

Abstract

Keywords: *navigation, navigation system, location tracking, location-based services, positioning, position finding, PDA, wireless data transfer, 3D-visualization, MedioVis.*

Although navigation systems were until very recently mostly known and used in the realm of air and sea transport and therefore only rarely trusted to public use, a change in this trend can be observed in recent years. The main reason for this is the rapid developments in the field of car navigation, which in the meantime is to be found in growing numbers in cars and will most probably become a regular fixture in the near future. The presence of the media has most probably contributed to the rise of knowledge about these systems, and to the fact that most individuals now know what GPS-based navigation systems look like and how they work. The next step in development would be to implement navigation systems in buildings.

In this paper, some navigation systems in existence to this day will first be introduced in order to then give an in-depth picture of the technical basics. A following chapter will be dedicated to the necessary design basics, since the realization of a 3D visualization on a PDA is the final objective of this project. As the acceptance of such a new system by the user is essential, a further consumer-oriented chapter will discuss particular needs and requirements by users. Next, the prototype developed for this project using Flash and SWIFT will be presented along with the instructions for the full implementation of such a system. The conclusion then gives a small glance into the future.

Inhaltsverzeichnis

1	Einleitung	8
1.1	Begriffserklärungen	9
1.2	Aufgabenstellung / Inhalt der Arbeit	11
2	Bestehende Systeme	13
2.1	Tracking	13
2.1.1	MagicMap	14
2.1.2	Ekahau Positioning Engine	17
2.2	Location-Based Services	21
2.2.1	HIPS	21
2.2.2	SAiMotion	22
2.3	Gesamtsysteme	23
2.3.1	ActiveCampus	23
2.3.2	SmartLibrary	25
3	Technische Grundlagen (Backend)	29
3.1	Kabellose Datenübertragung	29
3.1.1	Infrarot	29
3.1.2	Bluetooth	31
3.1.3	GPS	31
3.1.4	Ultraschall	32
3.1.5	WLAN	33
3.1.6	Fazit (Anwendbarkeitsanalyse)	34
3.2	Bestimmung der Benutzerposition (Tracking)	36
3.3	Location-Based Services	38
3.4	Programmiersprachen (Server)	40
3.5	Routenberechnung	41
4	Visualisierung und Designgrundlagen (Frontend)	44
4.1	Visualisierungstechniken	44
4.1.1	OpenGL	44
4.1.2	Flash / Swift 3D	45
4.1.3	VRML (JavaScript/XML)	45
4.2	Designgrundlagen	47
4.2.1	Landmarks	47
4.2.2	Detailierungsgrad	48
4.2.3	Sichtweise / Perspektive	49
4.2.4	Blickfeld	51
4.2.5	Ausrichtung der Karte	52

5 Benutzerorientierter Ansatz	54
5.1 Benutzerprofile	55
5.2 Benutzbarkeits-Kriterien	56
5.3 Benutzer-Erfahrungen	57
5.4 Konzeptionelles Modell	57
5.5 Anforderungen und Werkzeuge	58
5.6 Anwendungs-Szenario	60
6 Prototyp	62
6.1 Steuerungsprogramm	62
6.2 Programmierung	63
6.3 PDA-Handhabung	65
6.4 Anzeigen	65
6.5 Werkzeugleiste	67
6.5.1 Optionen	67
6.5.2 Suchliste	68
6.5.3 Auswahl der Sichtweise	68
6.5.4 Gesamtansicht	68
6.5.5 Informationen	68
6.6 Darstellung des Benutzers	69
6.7 Navigationselemente	69
6.8 Ablauf einer Suche	70
6.9 Erweiterungen	71
7 Vorgehensweise zur vollständigen Implementation	73
7.1 Hardware	73
7.2 Daten	74
7.3 Software	74
7.3.1 Client-Programme	74
7.3.2 Server-Programme	75
7.3.3 Verwaltungsprogramm	76
7.4 Kosten	77
8 Ausblick	78
Literaturliste	80
Internetquellen	82
Abbildungsverzeichnis	86
Abkürzungen	87
Anhang	88

Stichwortverzeichnis

94

1 Einleitung

Navigationssysteme: Die ersten ihrer Art wurden während des Zweiten Weltkriegs entwickelt und dienten als Leitsystem für Kampfflugzeuge (*LORAN*¹) und zur Unterstützung der maritimen Navigation (*Decca*²).

Heutzutage beruhen die meisten Systeme auf dem amerikanischen Satellitennavigationssystem GPS, welches seit Anfang der 90er Jahre auch in Kraftfahrzeugen verwendet wird und somit auch zu Land Einzug gehalten hat. Um aber von den Amerikanern und deren GPS unabhängig zu sein, hat die ESA (European Space Agency) zusammen mit China, Indien, Kanada und Israel ein eigenes Satellitensystem namens *Galileo* entwickelt, welches im Jahr 2010 voll einsatzbereit sein soll. Beide Systeme sind allerdings aufgrund der großen Abschattung und der ungenügenden Genauigkeit (für den nicht-militärischen Bereich) nur für den Außenbereich anwendbar.

Indoor-Navigationssysteme und ortsbezogene Dienste sind im Gegensatz dazu noch kaum verbreitet, doch ist ihre Entwicklung stark im Kommen. Ansätze auf diesem Gebiet sind bereits vorhanden, die meisten im universitären Bereich. Jedoch handelt es sich hierbei um kaum vollendet implementierte Systeme. Viele Techniken erwiesen sich in den anfänglichen Testphasen als zu teuer oder zu ungenau. Dank der stetigen Weiterentwicklungen werden sich die Möglichkeiten auf diesem Gebiet in den nächsten Jahren aber immer weiter verbessern. Dazu tragen auch die in diesem Bereich fallenden Marktpreise und die größeren zur Verfügung stehenden Bandbreiten zur Datenübertragung bei. In Folge dessen werden auch die grafischen Visualisierungsmöglichkeiten auf kleinen Displays wie einem PDA immer größer. Denn waren bis dato nur äußerst einfach aufgebaute

¹Navigationssystem der U.S. Army während des zweiten Weltkrieges [LORAN]

²Navigationssystem der Britischen Marine während des zweiten Weltkrieges [Seefunk]

2D-Visualisierungen realisierbar, so können Dank dieser technischen Innovationen in naher Zukunft auch anspruchsvollere Grafiken zum Einsatz kommen.

Diese Arbeit wird sich nun mit dem Thema des computergestützten Navigierens innerhalb geschlossener Gebäude befassen und die eben erwähnten, verbesserten Visualisierungsmöglichkeiten in Form von 3D-Grafiken umsetzen.

1.1 Begriffserklärungen

Folgende Begriffe bilden die Grundlage für diese Arbeit und geben einen kleinen Einblick über die darin behandelten Themen. Sie wurden dem Online-Lexikon Wikipedia [Wikipedia] entnommen.

Navigation

Navigation ist die "Steuermannskunst" zu Meer (Nautik), zu Land und in der Luft. Allgemeiner bezeichnet sie das sich Zurechtfinden in einem geografischen Raum, um einen bestimmten Ort zu erreichen. Die Tätigkeit des Navigierens (von lat. navigare; sanskrit navgathi) besteht aus drei Teilbereichen:

(1) Bestimmen der geografischen Position durch Ortung nach verschiedensten Methoden, (2) Berechnen des Weges zum Ziel und (3) Führung des Fahrzeugs zu diesem Ziel, also vor allem das Halten des optimalen Kurses.³

Ein Navigationssystem besteht ebenso aus diesen drei Teilen und entspricht bei der Vorgehensweise immer noch der urtümlich historischen Navigation aus der Seefahrt. Allgemein ist es ein elektronisches Gerät, das der Positionsbestimmung dient und gegebenenfalls beim Erreichen eines gewünschten Zieles behilflich ist.

³Erklärung des Begriffes "Navigation" [Wikipedia]

Der erste Punkt bei der Navigation ist die Orts- oder Positionsbestimmung, wobei der Standpunkt in einem globalen oder regionalen Koordinatensystem ermittelt wird. Auch der Mensch, beispielsweise als Fußgänger, ortet seine Position. Dies erfolgt meist unbewusst durch Gleichgewichtssinn und Sichtnavigation. Um durch Sichtnavigation die Raumlage zu bestimmen ist gutes Kartenmaterial eine große Hilfe. Um einiges komplizierter ist diese Bestimmung jedoch bei bewegten Objekten:

Tracking

Der Begriff Tracking (dt. Nachführung) umfasst alle Bearbeitungsschritte, die der Verfolgung von (bewegten) Objekten dienen. Ziel dieser Verfolgung ist zum einen die Extraktion von Informationen über den Verlauf der Bewegung und die Lage eines Objektes und zum anderen die Verminderung von negativen Einflüssen, herrührend von zumeist zufälligen Messfehlern (Messrauschen). Die extrahierten Informationen können die Geschwindigkeit der Bewegung, die Beschleunigung sowie Informationen bezüglich der Lage zu einem bestimmten, oft in der Zukunft liegenden, Zeitpunkt sein.⁴

Es gibt mehrere Techniken und dabei mehrere Methoden, um ein möglichst exaktes *Location Tracking* zu erreichen, welche im Laufe der Arbeit noch genauer vorgestellt werden.

Hat man die Position und die Bewegung eines Objekts, in diesem Fall den Benutzer des Navigationssystems, erst einmal festgestellt, gibt es die Möglichkeit ihm für seine spezifische Situation besondere Dienste anzubieten. Diese ortsbezogenen Dienste sind gerade erst im Kommen und werden bisher erst in einigen

⁴Erklärung des Begriffes "Tracking" [Wikipedia]

Handy-Netzen eingesetzt. Solche Angebote und Informationen werden aber immer wichtiger in unserer schnell wachsenden IT-Welt, weshalb sich diese Technik in den nächsten Jahren sicherlich durchsetzen wird.

Location-Based Services

Standortbezogene Dienste (engl. Location Based Services (LBS), auch: Location Dependent Services (LDS)) sind über ein Netzwerk erbrachte mobile Dienste, die unter Zuhilfenahme von positions-, zeit- und personenabhängigen Daten dem Endbenutzer selektive Informationen bereitstellen oder Dienste anderer Art erbringen.⁵

Dazu zählen auch die für dieses Projekt benötigten Richtungsangaben, um die Navigationsziele zu erreichen, ebenso wie weitere Einsatzmöglichkeiten, die noch erläutert werden.

1.2 Aufgabenstellung / Inhalt der Arbeit

MedioVis ist eine visuelle Benutzerschnittstelle zur Suche und Exploration multimedialer Bibliotheken und wurde von der Arbeitsgruppe Mensch-Computer-Interaktion der Universität Konstanz entwickelt. *Mobiles MedioVis* ist eine auf allen gängigen PDAs lauffähige Version dieses Programms.

In diesem Projekt soll ein *Indoor*-Navigationssystem entwickelt werden, das den Benutzer via PDA zu den im *MedioVis* gesuchten Medien führt.

Nachdem ein oder mehrere Medienobjekte (Video, Buch, etc ...) ausgewählt wurden, soll dieses System die optimale Strecke berechnen und den Benutzer zu den physischen Lagerorten dieser Medien in der Bibliothek leiten. Zusätzlich sollen dem Anwender weitere nützliche Informationen über seine Umgebung geboten

⁵Erklärung des Begriffes "Location Based Services" [Wikipedia]

werden. Ziel ist es, ein möglichst einfaches und benutzerfreundliches Interface zu entwickeln, welches dem Benutzer auch Spaß bei der Ausführung seiner Aufgaben macht und einen zeitlichen Gewinn zur herkömmlichen Suche gewährleistet.

Im anschließenden Kapitel dieser Bachelorarbeit werde ich einige bestehende Systeme und den momentanen Entwicklungsstand auf diesen Gebieten aufzeigen. Das darauf folgende dritte Kapitel beinhaltet die technischen Grundlagen. Wie bereits festgestellt besteht der Navigationsvorgang aus drei Teilen. Zwei dieser Teile, das Tracking der Benutzerposition und die Berechnung der optimalen "Rundreise", sind in diesem dritten Kapitel thematisiert. Des Weiteren beinhaltet es die unterschiedlichen Techniken der kabellosen Datenübertragung und der serverseitigen Verarbeitung dieser Daten. Der dritte Navigationsteil, wie der Benutzer zu seinem Ziel geleitet wird, ist Thema des vierten Kapitels, welches die möglichen Visualisierungen und die Designgrundlagen behandelt. Da der Benutzer im Mittelpunkt stehen soll, gehe ich im fünften Kapitel auf seine Anforderungen und die Benutzbarkeit solch eines Programms ein. Das sechste Kapitel beschreibt den zu diesem Projekt entwickelten Prototypen und Kapitel sieben beinhaltet eine theoretische Vorgehensweise zur vollständigen Implementation dieses Prototypen. Kapitel acht schliesst diese Arbeit mit einem Ausblick ab.

2 Bestehende Systeme

Bei der Recherche nach bestehenden Systemen stellt man schnell fest, dass die Entwicklung von Navigation und Tracking in Gebäuden noch in den Kinderschuhen steckt. In diesem Kapitel werde ich einige Systeme vorstellen, die für den Indoor-Bereich konzipiert wurden und beispielhaft für die jeweilige Teilgebiete sind.

Wer sich darüber hinaus ein Beispiel für kommerzielle KFZ-Navigationssysteme ansehen möchte, dem empfehle ich die Seite von *TomTom International BV*⁶, welche eins der bekanntesten Modelle zeigt.

2.1 Tracking

Im universitären Bereich existieren bereits einige implementierte Trackingkonzepte, die jedoch bei der Positionsbestimmung noch viel zu ungenau sind.⁷ Als Beispiel lassen sich hierzu *PlaceLab*⁸, *HORUS*⁹, *RADAR*¹⁰, *LEASE*¹¹ nennen. Das Einzige kostenfreie System, das eine Genauigkeit von unter zwei Metern erreicht, ist das Projekt MagicMap der HU Berlin, welches im Anschluss vorgestellt wird.

Die bekanntesten Programme im kommerziellen Sektor sind *Ekahau*, *Aeroscout*¹² und *Wherenet/Siemens*¹³. Alle basieren auf der WLAN-Technik und haben eine

⁶TomTom KFZ-Navigationssystem [TomTom]. Screenshot im Anhang.

⁷Im Anhang eine Auflistung mit den mittleren Ungenauigkeiten der Systeme.

⁸Intel / UC Berkeley

⁹University of Maryland

¹⁰Microsoft Research

¹¹Avaya Labs Research / Rutgers University

¹²Webseite der Firma Aeroscout: <http://www.aeroscout.com/> (Stand: 16.6.2006)

¹³Webseite der Firma Wherenet: <http://www.wherenet.com/> (Stand: 16.6.2006)

mittlere Positionierungsabweichung von nur einem Meter. Auf ein weniger kostenintensives, aber dennoch ausgereiftes System, die *Ekahau Positioning Engine* und ihre Tools, wird direkt nach MagicMap eingegangen.

2.1.1 MagicMap

Das Project MagicMap wurde an der Humboldt-Universität Berlin entwickelt und ist eine Open-Source-Software. Geschrieben ist sie in Java und läuft auf den Betriebssystemen Windows (XP, Pocket PC) und Linux. Die Messung der Signalstärke läuft über NetStumbler bzw. die Linux Wireless Tools, daher ist es nicht ohne weiteres plattformunabhängig.

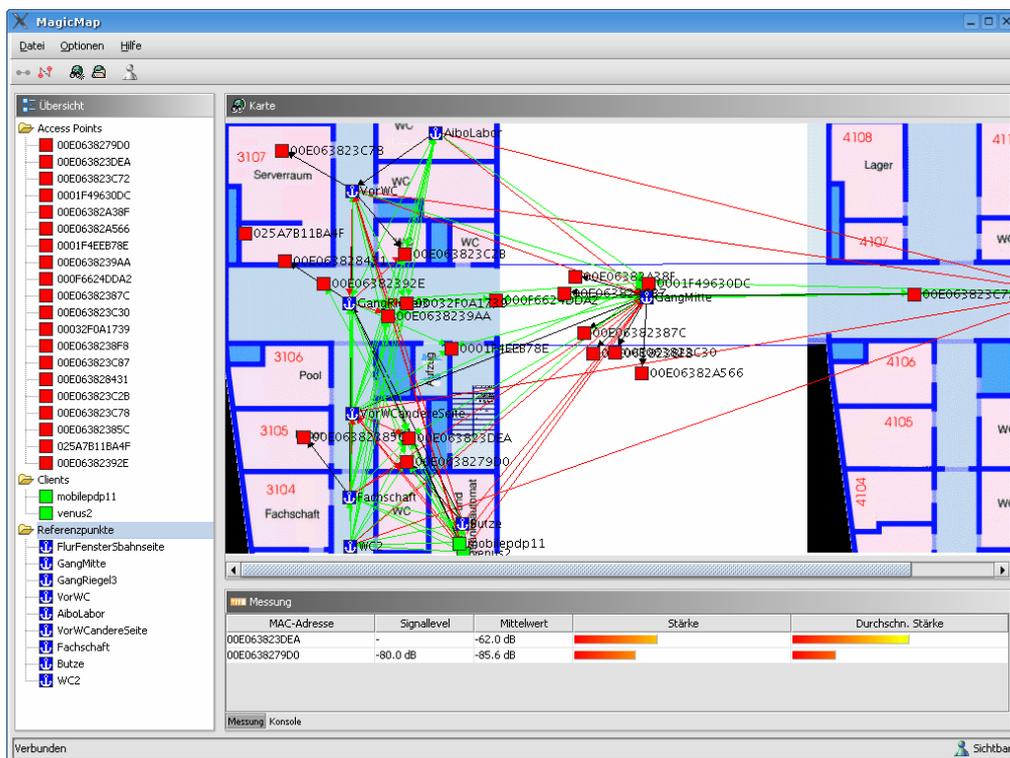


Abbildung 1: Screenshot MagicMap [MagicMap]

Die Stärke dieses Ansatzes ist die ad-hoc Fähigkeit. Mit dieser erreicht man auch ohne Kenntnisse der Infrastruktur, also ohne die Position umgebender WLAN Access Points zu kennen, einfach über die Kooperation und den Austausch der Messdaten, sehr schnell eine einigermaßen akkurate Ortung.

”MagicMap ist eine reine Softwarelösung, die bei den mobilen Systemen außer einer konventionellen WLAN-Ausstattung keine weitere Hardware erfordert. Die WLAN Access Points können beliebig verteilt sein und es sind weder Eingriffe an der AP-Hardware noch an der Software nötig.”¹⁴

Es ist ein verteiltes Softwaresystem und besteht aus mehreren Modulen [MagicMap]:

GUI, zeigt die Position der Knoten auf Karten an, siehe Screenshots. Das GUI gibt es betriebssystemunabhängig in Java für ”ausgewachsene” Rechner, sowie in .Net/C# für Pocket PCs mit Windows 2003/2005.

Stumbler, misst sensorische Daten der Umgebung, z.B. WLAN-Signalstärkeinformationen. Es gibt verschiedene Stumbler für WLAN, Bluetooth, RFID und ZigBee, jeweils für unterschiedliche Hardware und Betriebssysteme.

Server, sorgt für die Verteilung der Daten zwischen den Knoten (Signalstärke-Messwerte, Positionsberechnungen bzw. Referenzmessungen).

P2P-Kommunikation, erlaubt den Austausch von Daten auch in infrastrukturlosen Umgebungen, also auch ohne Zugang zu einem Server bzw. zu Access

¹⁴Zitat aus [MagicMap].

Points.

Positioning Engine, berechnet die Positionen. Die Aufteilung der Berechnung kann unterschiedlich konfiguriert werden: Berechnung auf den Clients, auf dem Server oder verteilt. Im "Auto-Modus" wird die Berechnung entsprechend der Leistung der verfügbaren Rechner ad-hoc verteilt.

Tracker, beobachtet die Positionsdaten und triggert bei bestimmten Mustern voreinstellbare Aktionen.

MagicMap ist eine voll Client-basierte Ortung und ist mit einem zusätzlichen Manipulationsschutz ausgestattet. Da es sich aber um ein Forschungsvorhaben einer Hochschule handelt, kann keine Gewähr für einen erfolgreichen Einsatz in einem laufenden Betrieb gegeben werden, weshalb eine Unterstützung durch die Universität Berlin leider auch nicht gegeben ist.

Sicherlich ist es interessant sich diese Software zu Probezwecken einmal herunterzuladen und auszuprobieren. Allerdings besteht bei problematischen Bedingungen, wie der wechselnden Positionen oder Verfügbarkeit der Access Points, immer noch Forschungsbedarf.

In der aktuellen Version 1.0 ist leider erst eine Genauigkeit von 3-5m erreichbar. Ab der Version 2.0 aber soll, wenn die Position von drei Signalquellen (AP) und ausgiebige Referenzmessungen vorhanden sind, eine Genauigkeit von unter zwei Metern erreicht werden. Wann allerdings die Version 2.0 herauskommt, ist noch nicht bekannt.

2.1.2 Ekahau Positioning Engine

Das *Ekahau Positioning Engine Software Concept* der Firma Ekahau Inc [Ekahau] beinhaltet ein klassisches RTLS (Real Time Locating System). Es ist eine *end-to-end* Lösung, welche die vorhandene Infrastruktur erkennt und analysiert. Laut Hersteller kann die Position tausender Objekte exakt bestimmt und in Echtzeit von anderen Programmen ausgelesen werden. Diese können die Informationen dann sowohl verarbeiten und dem Benutzer aufgearbeitet zur Verfügung gestellt werden, als auch weitere spezifische Umgebungsdaten anzeigen. Das System ist aufgebaut auf der Standard 802.11 WLAN-Technik und arbeitet mit allen handelsüblichen Basisstationen und Netzwerkkarten. Keine weitere Hardware oder Infrastruktur ist erforderlich.

Das ganze System besteht aus vier Komponenten:

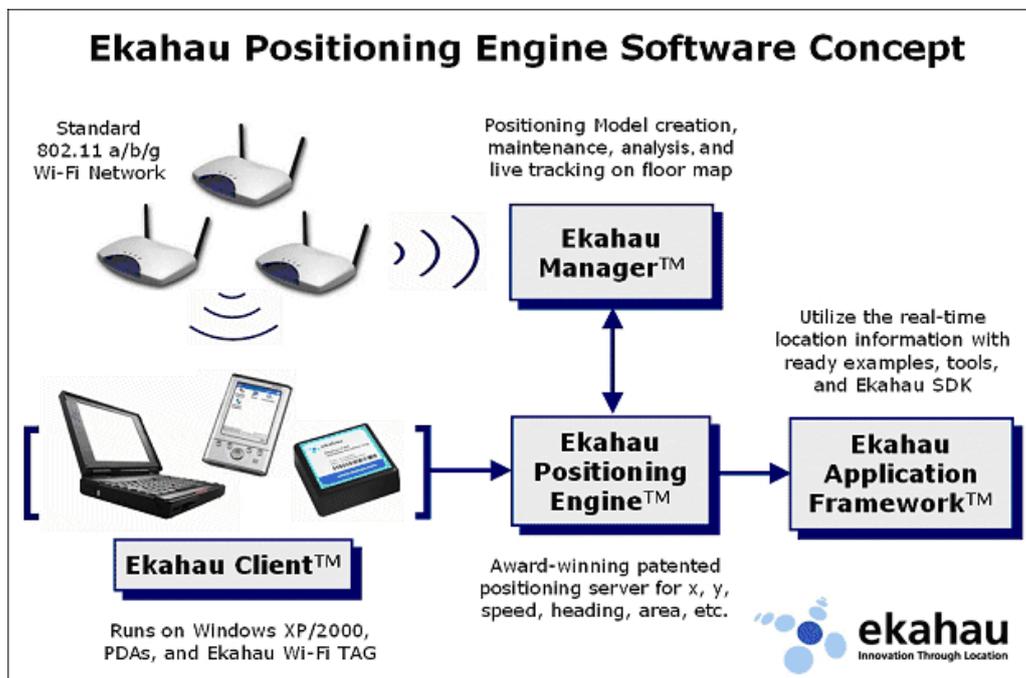


Abbildung 2: Ekahau Komplettsystem [Ekahau]

Ekahau Client™, ein kleines Programm, das auf dem Endgerät läuft (PC Laptop, PDAs, Wi-Fi Tag, etc.) und die aktuell empfangenen Signalstärken an die *Ekahau Positioning Engine™* schickt.

Ekahau Positioning Engine™ ist ein Dienst, der auf einem Server oder Desktop PC läuft und die Position des jeweiligen Clients berechnet (x, y - Koordinaten, Ebene, Blickrichtung, Geschwindigkeit, Gebietsname, etc).

Ekahau Manager™ ist eine Anwendung, um die Umgebungsdaten aufzuzeichnen mit denen das Positionierungsmodell erstellt wird. Ein Teil davon ist die *Ekahau Site Calibration™* mit der die Endgeräte auf einer Karte angezeigt werden und die Genauigkeit der Positionierung analysiert werden kann.¹⁵

Ekahau Application Framework™ and SDK, dies ist ein Set von hilfreichen Tools und einem einfachen Interface zur Programmierung eigener Anwendungen, z.B. um Umgebungsinformationen an den Client zu schicken.

Das komplette Paket ist vielseitig einsetzbar und kostet ca. 5000 US\$. Ein möglicher Anwendungsfall wird mit dem Projekt *SmartLibrary* am Ende dieses Kapitels ausführlich beschrieben. Ein weiteres Beispiel ist folgendes:

”Das RTLS-System von Ekahau bietet eine Personenortungslösung, die sich in das bereits in den meisten Krankenhäusern und Gesundheitszentren in Betrieb befindliche Wi-Fi-Netzwerk integrieren lässt. Krankenhausverwaltung und Sicherheitspersonal können dank der im RTLS-System integrierten EPE-Software (Ekahau Positioning Engine) Version 3.1 und der T201 WLAN-Tags von Ekahau den

¹⁵Screenshot im Anhang

Aufenthaltort von Ärzten und Krankenschwestern über eine Intranet-Website auf jedem Internet-fähigen Gerät feststellen. Im Notfall kann ein mit dem Ekahau-Tag ausgestatteter Krankenhausmitarbeiter den Alarmknopf drücken, um Hilfe herbeizurufen. Das Alarmsignal kann auf die Monitore des Sicherheitspersonals sowie per SMS Textbotschaft auf ihre Mobiltelefone weitergeleitet werden.”¹⁶

Ein weiteres hilfreiches Tool der Firma *Ekahau Inc.* ist die kommerzielle Software *Ekahau Site Survey*. Sie bietet die Möglichkeit Access Points zu verwalten, deren Standorte zu ermitteln und so genannte Rogue Access Points¹⁷ zu finden. Zur Planungshilfe kann auch anhand der Messergebnisse der optimale Standort für einen weiteren Access Point ermittelt werden [Golem 2004]. So kann eine optimale Signalabdeckung des gesamten Bereiches und eine maximale Abweichung der gemessenen Objektposition von nur einem Meter erreicht werden, sowie eine genaue Aussage über die Installations- und Netzwerkqualität des lokalen WLAN-Netzes gemacht werden.¹⁸

”Die Software soll sowohl den Signal-Rauschabstand als auch die Interferenzen zwischen zu dicht beieinander platzierten bzw. auf gleichen Frequenzen funkenden Access Points anzeigen. Darüber hinaus könne durch Veränderung der Kanäle der Access Point die Auswirkung auf die Messergebnisse simuliert werden, um ein optimiertes WLAN zu erreichen.”¹⁹

¹⁶Anwendungsbeispiel aus [MTB 2006]

¹⁷*Rogue Access Points* werden von Angreifern installiert um Netzwerke auszuspionieren. Sie gehören nicht zum eigentlichen Netzwerk.

¹⁸Screenshot im Anhang

¹⁹Zitat aus [Golem 2004]

Die Software wird auf einem gängigen Laptop oder Tablet-PC installiert mit welchem die Wege abgeschritten werden. Konstant werden dabei die Signalstärken der einzelnen Access Points automatisch gemessen und gespeichert. Die Grundrisse der Gebäude können als Bilder in das Programm geladen werden, wodurch die Feldstärken und die Messpunkte direkt auf dem Gebäudeplan angezeigt werden können. Die abgelaufene Strecke wird somit, wie in folgender Abbildung ersichtlich, nachgezeichnet und die zusätzlich verfügbare Datenrate visualisiert.

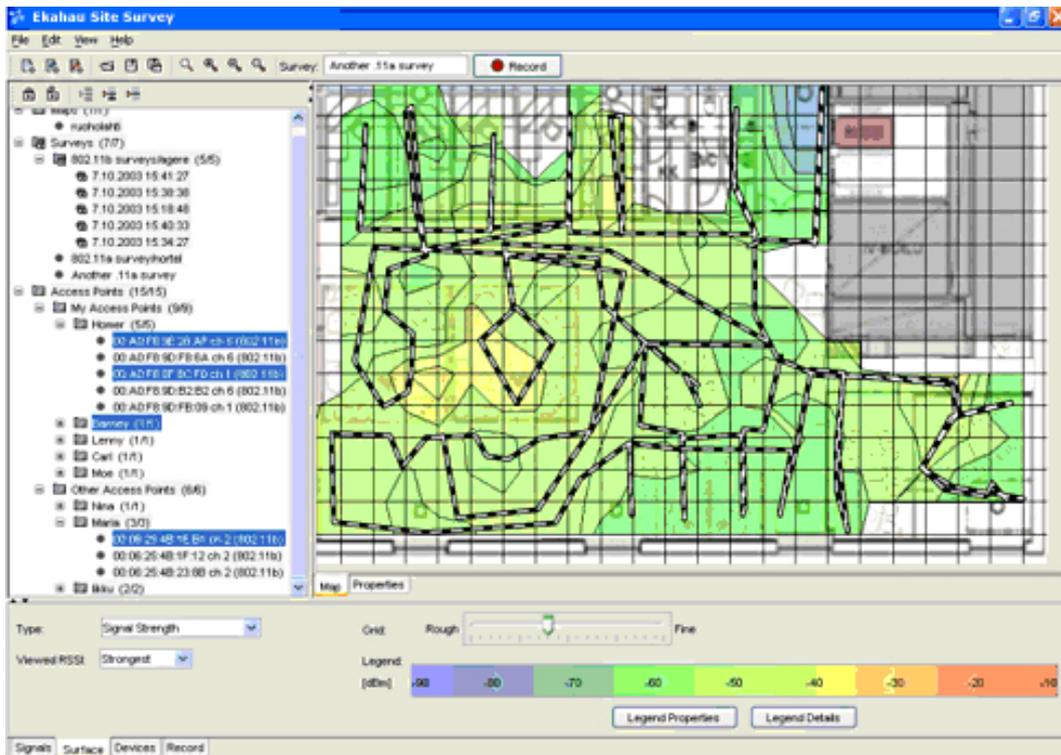


Abbildung 3: Ekahau Site Survey - Messpunkte entlang der Wegstrecke [Ekahau]

Dieses Software-Tool ist aktuell in der Version 2.2 vorhanden und kann bisher nur über den deutschen Distributor *Psiber Data* aus den USA bestellt werden. Es kostet je nach Paketgröße zwischen 1500 € und 3500 €.

2.2 Location-Based Services

2.2.1 HIPS

HIPS ist ein laufendes Projekt des Fraunhofer Instituts für Angewandte Informationstechnik, das die Benutzerposition verwendet, um dem Benutzer standortgebundene Dienste anzubieten.

”Das Projekt HIPS hat ein neuartiges Informationssystem für mobile Rechner (PDA) entwickelt und evaluiert, bei dem auch die Route und der aktuelle Standort des Benutzers beeinflussen, welche Informationen ihm angeboten werden. Außerdem passt das System seine Informationsangebote adaptiv an die Interessen und Kenntnisse des jeweiligen Benutzers an. [...]”

Im Museum benutzen die Benutzer des HIPS-Systems einen PDA mit Kopfhörer. Aber nicht vorgegebene Erläuterungen entlang einer festen Route - wie bei einer Führung von Tonband-Kassette - diktieren ihren Weg durch das Museum, sondern das, was sie in der Vorbereitung für sich als anschauenswert markiert haben, ihr spontan gewählter Weg und ihre besonderen aktuellen Interessen bestimmen den Inhalt und Abfolge der Führung - im Kopfhörer und ergänzend auf dem Bildschirm ihres PDA.”²⁰

HIPS bietet also eine Benutzer-adaptive und Standort-adaptive Museumsführung. Der Benutzer kann durch dieses Informationssystem seinen Museumsbesuch zu Hause vor- und nachbereiten. Dieses System basiert auf der Infrarot-Technik. Da bei dieser Technik für eine gute Abdeckung viele Sender notwendig sind, wurde bei diesem Projekt nur über jedem Gemälde und jeweils in den Eingangsbereichen zu den Räumen ein Infrarotsender angebracht.

²⁰Zitat aus [HIPS]

2.2.2 SAiMotion

Ein weiteres, in diesem Zusammenhang erwähnenswertes Projekt des Fraunhofer Instituts ist SAiMotion.

”SAiMotion (Situation Awareness in Motion) entwickelt und evaluiert ein nomadisches Informationssystem, das den mobilen Besucher in einer komplexen Umgebung personalisierte und situativ angepasste Informationen bereitstellt. Dazu wertet das System neben Ortsmerkmalen auch Benutzer-, Aufgaben- und Umgebungsmerkmale der aktuellen Situation aus und leitet daraus proaktiv geeignete Informations- und Dienstangebote ab.”²¹

Eingesetzt werden kann dieses System für jede Art von location-based services. Exemplarisch unterstützt es bereits auf Messen die Vorbereitung, Durchführung und Auswertung von Messebesuchen. Hierbei können die Besucher zur Vorplanung die für sie interessanten Aussteller auf einem Geländeplan fokussieren. Während des Besuchs der Messe wird der Besuchsplan dann vom System fortlaufend unterstützt, macht den Benutzer auf Änderungen aufmerksam und teilt ihm Neuerungen mit. Auf dem Messegelände lassen sich dann beispielsweise die Gesprächspartner von ihrem PDA termingerecht zu ihrem Treffpunkt leiten, bei weniger Zeitdruck auch zu interessanten Exponaten, die auf dem Weg liegen. Zur Nachbereitung des Messebesuchs kann das SAiMotion-System Routen, Adressen und Gesprächsnotizen auswerten und in die Datenbasis des Besuchers integrieren. In umfangreichen Feldtests wird evaluiert, in welcher Konstellation die Nutzung als effektiv und komfortabel bewertet wird. [SAiMotion]

²¹Zitat aus [SAiMotion]

2.3 Gesamtsysteme

Auf dem nicht-kommerziellen Sektor existieren im Indoor-Bereich so gut wie keine vollständig implementierten Komplettsysteme. Einige kommerzielle Systeme werden bei der Logistik von Unternehmen eingesetzt, allerdings noch keine auf dem Gebiet der Fussgänger-Navigation.

Ich werde im Folgenden zwei Systeme vorstellen, die auf dem Campus einer Universität eingesetzt werden, was auch Ziel dieses Projekts ist. Jedoch sind auch diese noch in der Testphase und wurden bisher nur teilweise umgesetzt.

2.3.1 ActiveCampus

Das Projekt ActiveCampus läuft an der Universität von Kalifornien in San Diego (US). Die Hauptsponsoren dabei sind Hewlett Packard (*Mobile Technology Solutions program*) und Microsoft (*Microsoft Research University Relations*). Location-based Services werden dabei eingesetzt, um den Unterricht zu unterstützen und den Studenten ihr tägliches Leben auf dem Campus zu erleichtern. Das Teilprojekt ActiveClass wiederum unterstützt und fördert die Zusammenarbeit der Studenten mit ihren Professoren.²²

Der Vorgang sieht wie folgt aus: Ein Server lokalisiert die einzelnen Personen im WLAN-Netz und zeichnet sie zusammen mit ihren momentanen Aktivitäten auf einer Karte ein. Das Tracking erfolgt dabei über die Messung der Signalstärke der auf dem Campus verteilten Access Points im WLAN-Netzwerk der Universität. Zusätzlich sind auf der erstellten Karte alle wichtigen Plätze des Campusgeländes eingetragen, wie zum Beispiel der Buchladen oder die Cafeteria.²³ Je nach aktueller Position bekommt man spezifische Informationen über seine Umgebung

²²Informationen aus [UCSD 2006]

²³Weitere Screenshots siehe Anhang

(Ansprachen, Kunstausstellungen, Kollegen, Labore, Institute, Bibliotheken, usw ...) angezeigt. Ebenso wurden alle gängigen Chat-Programme als weiterer Dienst eingebunden. Auf diese Weise können die Studenten mit ihren Freunden, die sie ebenfalls auf der Campus-Karte sehen, kommunizieren.

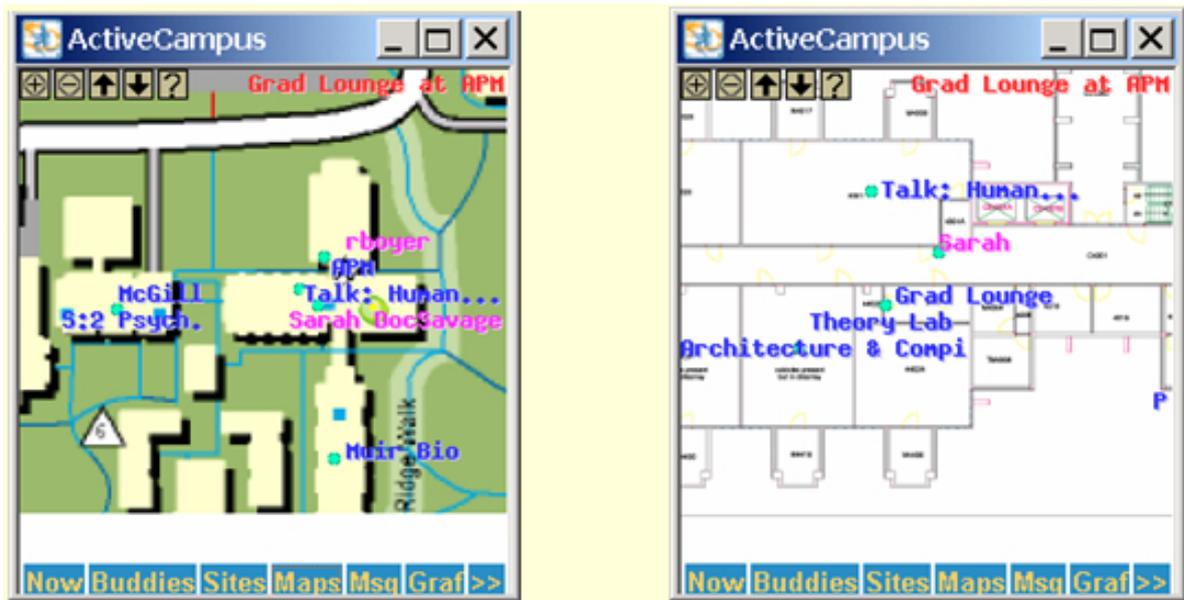


Abbildung 4: Screenshot des ActiveCampus Browsers. (*links*) Campus-Übersicht, (*rechts*) Gebäudeansicht. [UCSD 2006]

Sehr bemerkenswert ist zudem, wie mit dem Thema Datenschutz umgegangen wird. Es musste ein Mittelweg gefunden werden zwischen einem offenen Zugang und dem Schutz der Privatsphäre. Gelöst wurde das Problem, indem man, bevor man sich im Netzwerk angemeldet hat, die Möglichkeit hat, seinen Modus nach den eigenen Bedürfnissen einzustellen.²⁴ Ziel ist es, die Studienkollegen zwar untereinander zu verbinden, Missbrauch (stalking) soll hingegen verhindert werden, indem nur Freunde die aktuelle Position und Aktivitäten einsehen können.

Dieses Projekt ist sehr vielversprechend und scheint Zukunft zu haben, denn die

²⁴Screenshot im Anhang

Ideen wurden gut überdacht und sauber implementiert. Einen immensen Vorteil stellen natürlich die zwei großen Sponsoren dar, die mit ihrer Finanzierung das Projekt vorantreiben und umfangreiche Feldtests ermöglichen.

2.3.2 SmartLibrary

Das Projekt SmartLibrary wurde von der Universität Oulu in Finnland entwickelt und ist dort auf der Hauptebene der Bibliothek installiert. Dieses Projekt kommt von allen aktuell entwickelten Projekten dem, was in dieser Bachelorarbeit erarbeitet werden soll, am nächsten.

Das Programm SmartLibrary ist ein Service, der auf zwei bereits bestehende Systeme der Universität Oulu aufgesetzt wurde.

Die Büchereidatenbank wird mit der Software *OULA-pda* angesprochen. Dies ist eine für mobile Endgeräte (PDA, Handy) entwickelte Version des OPAC (*Online Public Access Catalogue*) der Bibliothek Oulu und beinhaltet ca. 60.000 Bücher. Das zweite System ist SmartWare, eine Prototyp-Systemarchitektur zur Unterstützung von kontextbezogenen Multimedia-Diensten. Dabei handelt es sich grafisch gesehen um den Vorläufer von SmartLibrary, wobei Programmteile von SmartWare weiter verwendet werden.

Über das Suchsystem OULA-pda kann nun ein Buch gesucht werden. Hat man dieses gefunden, wird man anschließend, per Klick auf das gewünschte Element, mit Hilfe des Java-Programms SmartWare (SmartLibrary) zum Objekt geführt. SmartWare war bereits zuvor in der Lage dynamische Positionierungen anzuzeigen, allerdings ist die Zielführung ausschließlich karten-basiert, d.h., dass es sich lediglich um eine statische Karte handelt, auf der die zur Wegfindung entscheidenden Landmarks²⁵ grün und das gesuchte Zielareal rot eingezeichnet sind. Der

²⁵Herausstechnde Merkmale der Umgebung

Nachfolger, SmartLibrary, läuft in einem XHTML-Browser auf dem mobilen Endgerät und kann nun die genaue Benutzerposition anzeigen, die auf dem Display in Form eines "Smileys" angezeigt wird. Die Symbole sind, wie bei SmartWare, grafisch markiert. Das System kann jedoch jeweils nur ein Objekt ansteuern, so dass die Berechnung einer optimalen Rundreise nicht möglich ist.

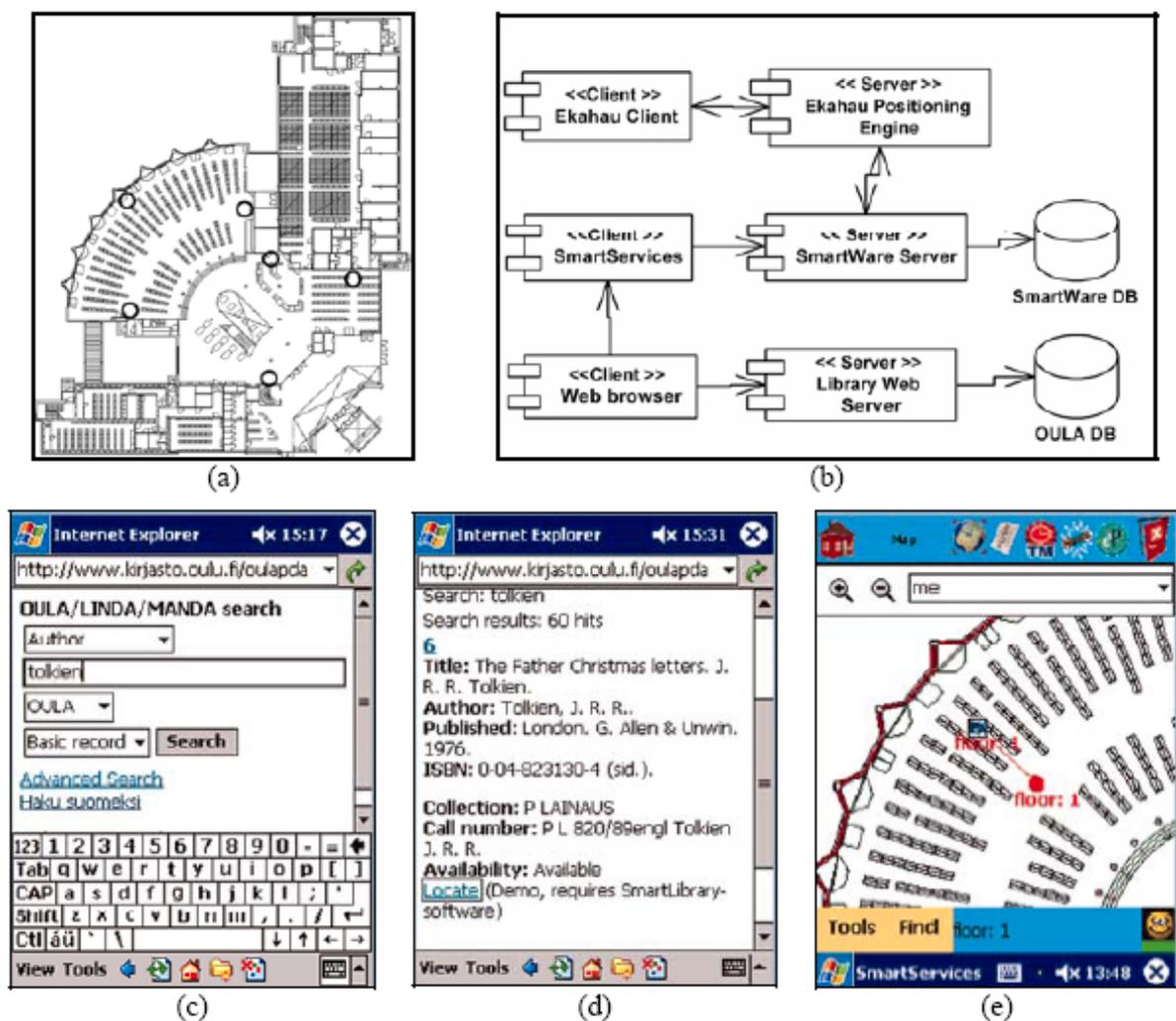


Abbildung 5: (a) Position der 6 APs (Kreise) auf der Hauptebene, (b) Darstellung der SmartLibrary Architektur, (c) Das Suchsystem OULA-pda, (d) Ergebnisliste von OULA-pda, (e) Visualisierung der Zielführung von SmartLibrary auf einem PDA. [Aittola 2003]

Im Hintergrund von SmartLibrary läuft zur Positionsbestimmung die *Ekahau Positioning Engine*, welche zuvor schon vorgestellt wurde. Diese wurde in das WLAN-Netzwerk der Universität integriert und arbeitet mit 6 Access Points, die für eine ausreichende Abdeckung der gesamten Hauptebene sorgen.

Da es allerdings weit über hundert Regale gibt und die Landmarken sich über die Zeit permanent ändern, musste ein System geschrieben werden, mit Hilfe dessen sich die Informationen zur Bibliothek verwalten lassen. Mit dem dafür entwickelten webbasierten Administrationsprogramm CPI (*Content Provider Interface*) können die Bibliotheksangestellten die zur Navigation benötigten Hintergrundbilder austauschen und die Grundrisse der Etagenpläne ändern. Regale und Landmarken werden dabei mit einem Zeichentool als Polygone gezeichnet und auf die Karten gelegt.



Abbildung 6: Web-basiertes Verwaltungssystem von SmartLibrary (CPI).
[Aittola 2004]

Um zu sehen, wie das System ankommt, wurde ein großer Benutzertest durchgeführt. Hierzu wurden 32 Bibliotheksbenutzer zufällig ausgewählt, die im Alter von 19 - 49 Jahren (Durchschnittsalter: 24 Jahre) waren. 90% von ihnen hatten schon Erfahrungen mit der OPAC Terminal-Version der Universität Oulu, wobei 25% der Probanden auch bereits Erfahrungen mit PDAs gemacht hatten. Es wurden zwei gleich große Gruppen gebildet, die in unterschiedlicher Reihenfolge jeweils eine Suche nur mit Hilfe der OPAC Terminal-Version und eine mit mobiler, navigations-gestützter Suche durchführen sollten. Das Resultat war schließlich, dass alle männlichen und 64% der weiblichen Benutzer die Suche mit mobiler Unterstützung erheblich einfacher fanden.

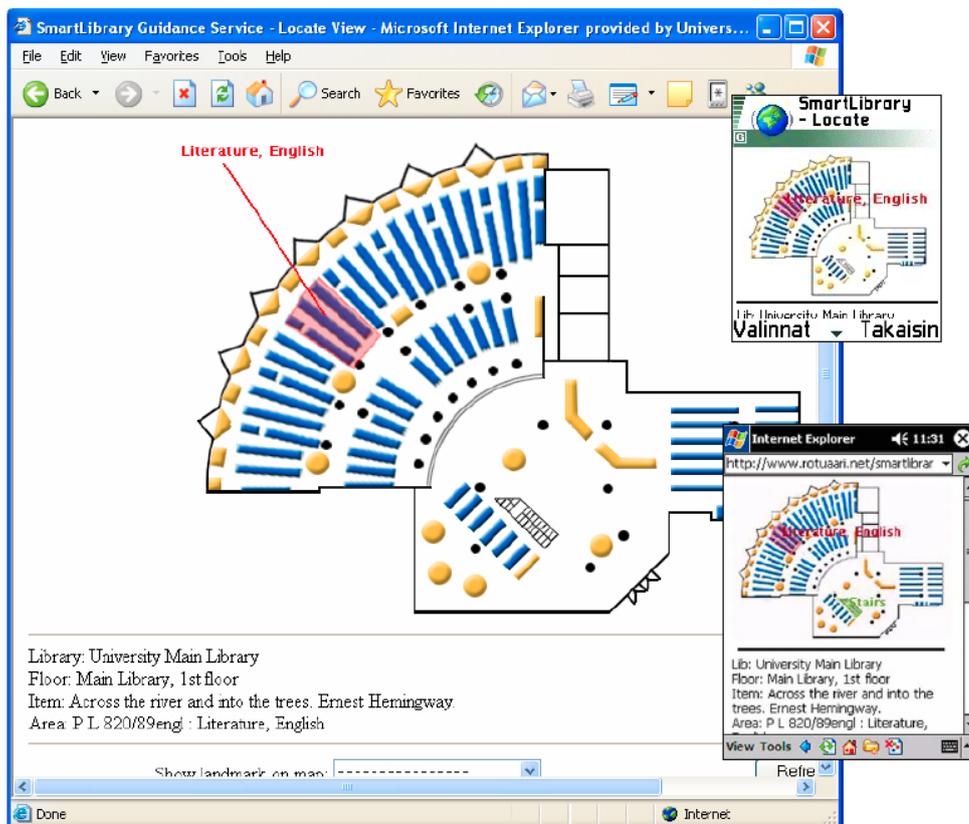


Abbildung 7: (Mitte) Desktop-Version, (Oben rechts) Handy-Version, (Unten rechts) PDA-Version. [Aittola 2004]

3 Technische Grundlagen (Backend)

Dieses Kapitel enthält einen Überblick über die technischen Grundlagen, die zur Realisierung eines mobilen Navigationssystems benötigt werden.

Zuerst werden mögliche Alternativen vorgestellt, um eine konstante, kabellose Verbindung zum Endgerät aufzubauen. Danach wird vorgestellt, wie es möglich ist mit diesen Techniken die möglichst genaue Position des Benutzers zu bestimmen. Im Speziellen wird dabei auf das WLAN-Tracking eingegangen. Die genaue Vorgehensweise der LBS-Systeme und die Berechnung der optimalen Rundreise bilden den Abschluss dieses Kapitels.

3.1 Kabellose Datenübertragung

Eine Technik auf die ich hier nicht eingehen möchte ist RFID, da diese nur mit speziellen Barcodes und Chips arbeitet und keine exakte Positionsbestimmung ermöglicht.

3.1.1 Infrarot

Infrarot arbeitet mit elektromagnetischen Wellen. Diese Wellen (Frequenzband) sind länger als Licht, aber kürzer als Radiowellen. Sie sind für das menschliche Auge nicht sichtbar und finden Anwendung in fast jedem Haushalt, bekannte Beispiele dafür sind die Fernbedienung des Fernsehers oder der Stereoanlage.

Die Datenübertragung beruht auf dem IrDA Protokoll und benötigt direkten Sichtkontakt vom Sender zum Empfänger. Die Objekte in der Umgebung können dabei die Strahlen reflektieren, was teilweise Probleme beim Tracking mit sich bringt. Ein Vorteil dieser Technik ist allerdings, dass sie nicht durch andere elektronische Geräte gestört wird.

Es gibt zwei Varianten der Infrarot-Technik. Eine ist Active Badge, wo jeder Benutzer einen kleinen Infrarotsender trägt und ständig seine aktuelle Position an einen Server zur Positionsberechnung übermittelt. Die Andere ist WIPS, bei der das Endgerät von Sendern ausgestrahlte Signale empfängt und die aktuelle Position eigenständig berechnet.

Infrarot - Technische Daten:

Datenübertragungsrate:	4 Mbps
Reichweite der Sensoren:	bis 20m
Tracking-Genauigkeit:	bis 9cm (Active Badge)

Die Verbindung zu einem einzelnen Sensor reicht zwar, aber die Empfänger und Sender haben jeweils einen Strahlenwinkel von nur 30° . Das bedeutet, wenn man mit dieser Technik ein komplettes Gebiet abdecken möchte, muss man alle ein bis zwei Meter einen Empfänger anbringen.

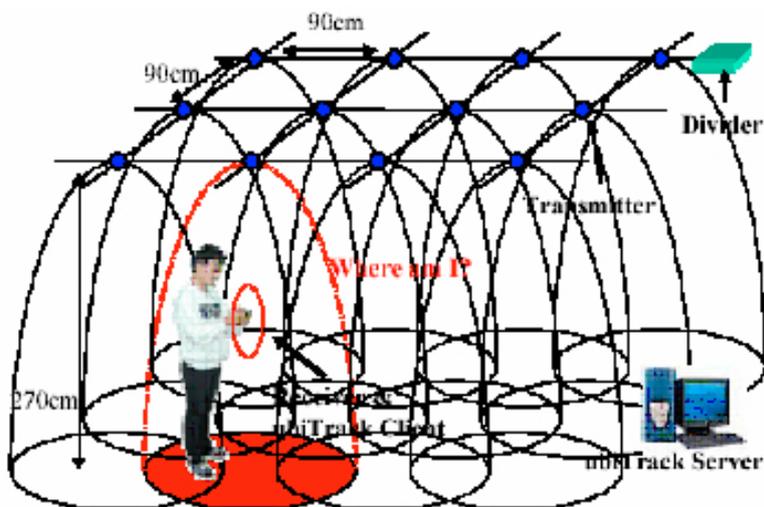


Abbildung 8: Abdeckung des ubiTrack Systemes. [Jung 2004]

3.1.2 Bluetooth

Der Bluetooth Standard ist ein Funk-Standard und arbeitet über das frei verfügbare ISM (Industrial Scientific Medical) Funknetz, welches im 2.45-GHz-Bereich angesiedelt ist. Entwickelt wurde Bluetooth für die drahtlose Übermittlung von Sprache und Daten im Nahbereich.

Bluetooth - Technische Daten:
Datenübertragungsrate: 1 Mbps
Reichweite der Sensoren: bis 10m
Tracking-Genauigkeit: bis 1,7m (Durchschn. 2-3m)

Diese Technik hat für dieses Projekt entscheidende Nachteile: Neben der geringen Reichweite der Sensoren muss auch eine sog. "Discovery-Phase" berücksichtigt werden. Sobald man in die Reichweite eines Sensors kommt, muss man sich zuerst bei diesem anmelden. Diese Phase dauert ca. 5-10 sec. und muss abgewartet werden, bevor man seine Suche fortsetzen kann. Alle zehn Meter zu warten verzögert jedoch den Arbeitsablauf stark und eine normale Suche ohne Navigationssystem wäre schneller. Hinzu kommen Probleme, die durch Funkwellen anderer Bluetooth-Geräte in der Umgebung verursacht werden.

3.1.3 GPS

Das GPS System funktioniert mit 24+ Satelliten, die mit ca. 20 200 km Abstand zur Erde kreisen. Für eine komplette Erdumrundung brauchen sie 12h und senden dabei permanent 20 W Signale aus.

Es sind mindestens 4 Satelliten notwendig, um eine gute Positionsbestimmung ausführen zu können. Dabei sollte eine möglichst direkte Verbindung vom Endgerät zum Satelliten bestehen. Da die meisten Bibliotheken, wie auch die Bibliothek

Physikalisch gesehen bezeichnet Ultraschall Schallwellen oberhalb des menschlichen Hörbereichs. Der Ultraschallbereich liegt zwischen 20.000 Hz - 1000 MHz. Fledermäuse nutzen diese zum Beispiel zur Orientierung im Dunkeln. Ultraschallwellen werden mit so genannten piezo-elektrischen Kristallen erzeugt, die unter Anlage einer entsprechenden Wechselspannung schwingen und damit die Ultraschallwellen abgeben. Die Genauigkeit liegt bei dieser Technik bei ca. 10 cm, jedoch sind die Geräte sehr teuer und die Installation zu aufwendig für dieses Projekt.

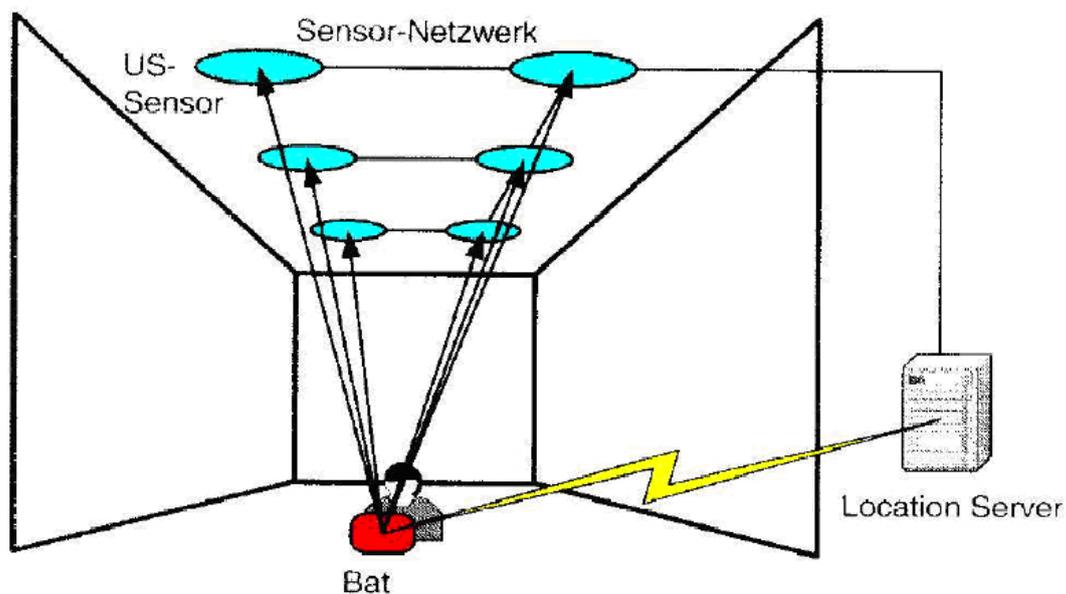


Abbildung 10: Aufbau einer Ultraschall-Umgebung

3.1.5 WLAN

WLAN (802.11b) ist die Abkürzung für "Wireless Local Area Network" und bezeichnet ein kabelloses Netzwerk, auch bekannt als RLAN. Das Funknetz ist wie Bluetooth im 2,4 GHz Frequenzband anzusiedeln und besteht aus elektromagne-

tischen Wellen.

WLAN-Netzwerke findet man z. B. in Universitäten, auf Messen, Bahnhöfen, Flug-Häfen, Museen und neuerdings auch in Zügen und Flugzeugen.

WLAN - Technische Daten:

Datenübertragungsrate:	11 Mbps
Reichweite der Sensoren:	bis 100m
Tracking-Genauigkeit:	bis 1m (je nach Testphase)

Jeder Bereich des Gebiets muss für eine genaue Positionsbestimmung von drei Access Points (AP) abgedeckt werden. Diese Verteilerknoten müssen genau aufgestellt werden, um Probleme in den Randbereichen zu vermeiden. Die Vorteile dieser Technik sind vor allem die schnelle Installation, die gute Flächenabdeckung und die, im Vergleich zu Kabelnetzwerken, geringen Installationskosten. Ein Nachteil ist die verhältnismäßig geringe Datensicherheit. Diese ist jedoch für dieses Projekt eher unwichtig.

3.1.6 Fazit (Anwendbarkeitsanalyse)

Von allen Techniken kommen eigentlich nur zwei in Frage, denn WLAN und Infrarot sind die, einzigen die relativ kostengünstig sind und die geeignete Genauigkeit bei der Positionierung haben.

Infrarot bringt einen relativ großen Installationsaufwand mit sich, da sehr viele Empfänger montiert werden müssen. Auch stellt sich die Frage, ob zwischen den engen Bibliotheksregalen immer eine direkte Sicht auf die Empfänger gewährleistet ist.

WLAN hingegen ist sehr schnell montiert, da nur einige wenige Access Points aufgestellt werden müssen. Ein weiterer Vorteil ist, dass sich das Netz auch an-

derweitig nutzen lässt, wie z.B. zum surfen im Internet, und stellt somit noch einen Mehrwert dar. In der folgenden Tabelle sind noch einmal die Vor- und Nachteile der einzelnen Techniken aufgelistet.

	PRO	CON
Infrared	<ul style="list-style-type: none"> • Sehr genau • rel. Kostengünstig 	<ul style="list-style-type: none"> • Viele Beacons nötig • Direkte Sicht nötig • Aufwändige Installation
Bluetooth		<ul style="list-style-type: none"> • Lange Verbindungszeiten • Geringe Reichweite • Andere Geräte stören
GPS		<ul style="list-style-type: none"> • 9m Ungenauigkeit nicht akzeptabel für diesen Zweck!
WLAN	<ul style="list-style-type: none"> • Hohe Geschwindigkeit • Einfache Installation • Infrastruktur anderweitig nutzbar • rel. Kostengünstig 	<ul style="list-style-type: none"> • Datenunsicherheit • Access Points müssen sehr genau platziert werden.
Ultraschall	<ul style="list-style-type: none"> • sehr genau 	<ul style="list-style-type: none"> • Hohe Kosten • Aufwändige Installation

Abbildung 11: Vor- und Nachteile der einzelnen Techniken

Wegen der hohen Geschwindigkeit und der leichten Anschaffung der benötigten Geräte, wie auch der in gleichen Szenario laufenden, erfolgreichen Realisierung von SmartLibrary der Universität Oulu, ist WLAN die für dieses Projekt beste Technik zur Positionsbestimmung.

3.2 Bestimmung der Benutzerposition (Tracking)

Wie man in der Einleitung schon erfahren hat, besteht der Vorgang bei der Navigation aus drei Teilen. Der erste Teil, die Bestimmung der geografischen Position durch Ortungsverfahren wird nun behandelt. Da jedoch in diesem Anwendungsfall nur WLAN zur kabellosen Datenübertragung in Frage kommt, werde ich hier auch nur auf die Positionsbestimmung durch diese Technik eingehen. WLAN Tracking-Systeme können auf vorhandene Infrastrukturen einfach aufgesetzt werden und die Daten und Informationen können an jeden Server geschickt werden, der ans Netzwerk angeschlossen ist.

Möchte man ein WLAN-Positionierungssystem in geschlossenen Gebäuden einrichten, so muss zuerst eine Trainingsphase durchlaufen werden. In dieser Phase werden an möglichst vielen Wegpunkten Messungen durchgeführt und die Signalstärken der Basisstationen (AP) zusammen mit den aktuellen Koordinaten in eine Liste eingetragen. Die Genauigkeit der Positionierung hängt also auch von der

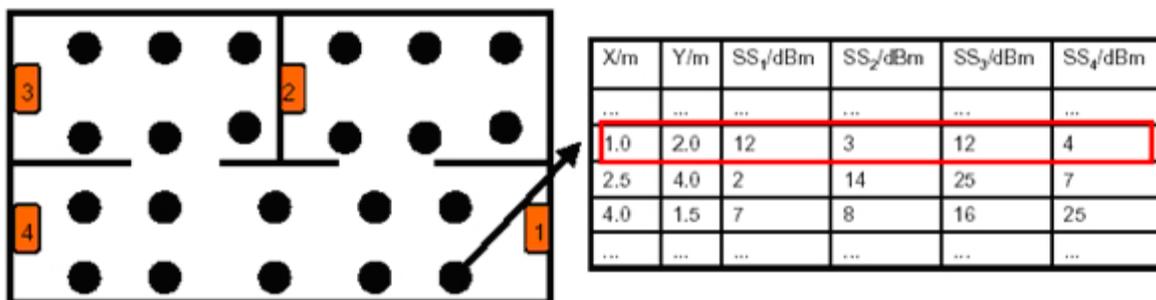


Abbildung 12: WLAN Testmessungen

Anzahl der in der Trainingsphase verwendeten Wegpunkte ab. Nach Änderungen in der Infrastruktur (z.B. Neupositionierung der Basisstationen) muss eine neue Trainingsphase durchgeführt werden.

Die Position wird meist anhand der *Triangulations*-Methode bestimmt. Dabei wird die Signalstärke der APs ausgewertet, die vom Endgerät an den Server geschickt werden. Dieses Verfahren wird auch als *Lateration* bezeichnet, durch welches die Entfernungen zu bekannten Bezugspunkten bestimmt werden. Unter Berücksichtigung der Positionen der Bezugspunkte lässt sich so der eigene Standort errechnen.

”Triangulierungsverfahren auf der Basis von Kreisen berechnen die Position anhand der Schnittpunkte von Kreisen, die sich aus den Abständen der einzelnen Sendestationen zur mobilen Station ergeben. Der Ausgangspunkt beim Triangulierungsverfahren sind drei Kreisgleichungen im 2-dimensionalen bzw. vier Kugelgleichungen im 3-dimensionalen.”²⁶

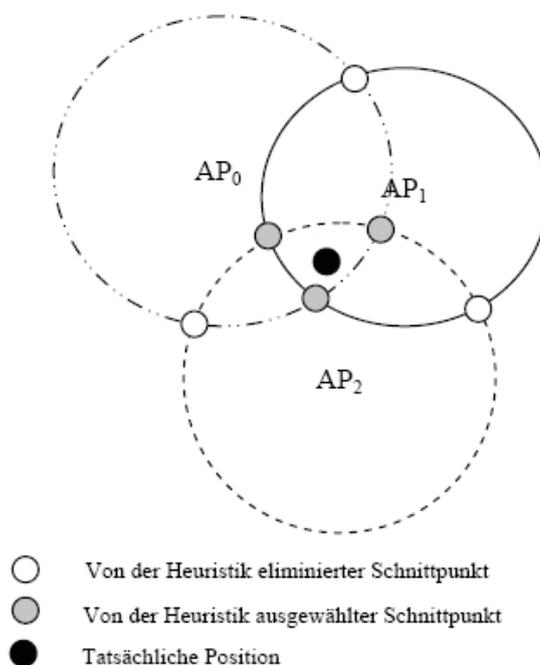


Abbildung 13: Triangulationsverfahren [Dornbusch]

²⁶Zitat aus [Dornbusch] Kapitel 3.3 Die Positionsbestimmung

Ein weiteres mögliches Verfahren ist die *Angulation*. Dabei werden die Winkel zwischen verschiedenen Bezugspunkten und dem Endgerät gemessen. Daraus lässt sich dann, ähnlich wie bei der Lateration, unter Berücksichtigung der Positionen der Bezugspunkte der eigene Standort ermitteln.

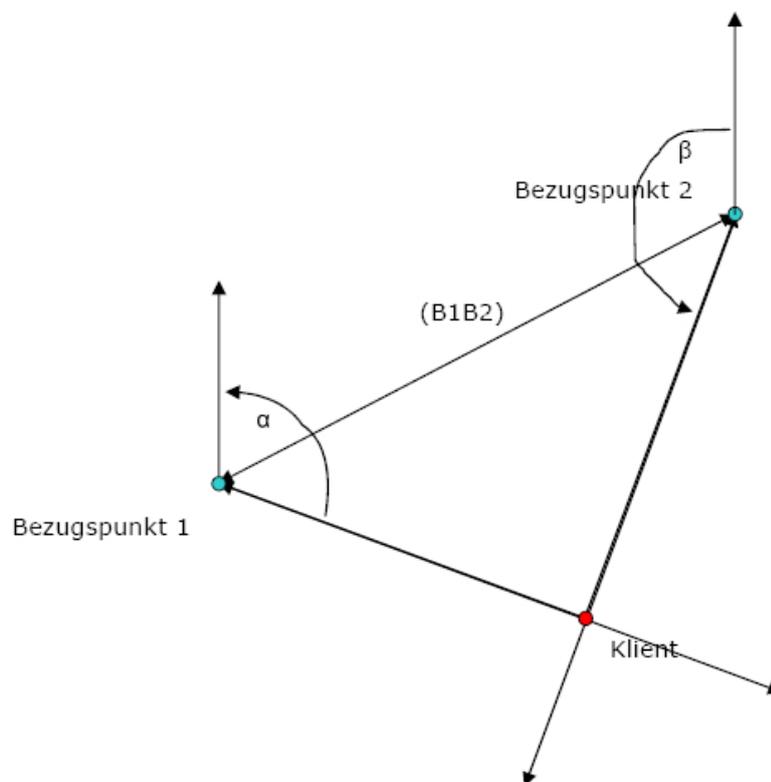


Abbildung 14: Angulationsverfahren

3.3 Location-Based Services

Diese ortsbezogenen Dienste lassen sich in zwei Klassen einteilen.

Zum einen in die der *reaktiven Dienste*, welche vom Benutzer explizit angefordert werden müssen. Sie erlauben etwa, dass ein Reisender ein Hotel, eine Tankstelle oder eine Apotheke in der näheren Umgebung findet.

Zum anderen in die der *proaktiven Dienste*, die auf bestimmte Ereignisse reagieren, wie zum Beispiel das Betreten einer bestimmten Zone. Sie werden in Navigationssystemen verwendet, um dem Benutzer bestimmte Informationen zur Routenfindung anzuzeigen, wie z.B. den aktuellen Standort, die Entfernung zum Ziel und die Richtung, in die er sich aus seiner aktuellen Lage zu bewegen hat. Ein Navigationsdienst könnte beispielsweise auf *OpenLS* vom *Open Geospatial Consortium* (OGC) basieren, bei dem es sich um eine Spezifikation für ortsbasierte Dienste handelt.

Beliebtes Testgelände für location-based services sind Messen. Dort können sich die Messebesucher aktuelle Informationen zu den Ständen in ihrer Umgebung direkt auf dem PDA anzeigen lassen.

Viele Techniken versagten dort allerdings. Zum einen aufgrund der hohen Anzahl an Geräten, die für das Tracking auf dem großen Gelände benötigt wurden und der damit verbundenen Kosten, zum anderen aufgrund der fehlenden Signalabdeckung für das gesamte Areal.

Im Outdoor-Bereich wurden diese Systeme erfolgreicher getestet und umgesetzt. Meist wurde dabei das GSM- oder UMTS-Netz der Mobilfunkbetreiber genutzt. Diese Netze sind in mehrere Cluster eingeteilt, wobei man die Position des Endgeräts jeweils immer in eines dieser Cluster einordnen kann. Dementsprechend wird dann, z.B. bei der Suche nach Dienstleistungsunternehmen, die Datenbank nach Unternehmen in diesem und umliegenden Clustern abgefragt.

Man erwartet für die Zukunft, nicht zuletzt durch die Einführung von UMTS, ein großes Potenzial ortsbezogener Anwendungen.

3.4 Programmiersprachen (Server)

Wenn man nicht auf bestehende Tracking-Systeme zurückgreifen möchte oder kann, dann gibt es die Möglichkeit eine der *Location APIs* der meisten Programmiersprachen (Java, C, C++, etc..) zu benutzen. Diese unterstützen gängige Funktechniken und haben bereits Methoden zur Positionsbestimmung und zur Umsetzung *location-based services* integriert. Die *JSR 179 Location API for J2ME™* wird im Folgenden exemplarisch für die Pakete anderer Sprachen vorgestellt.

Es handelt sich bei dieser API um ein optionales Paket für die aktuelle Java Version J2ME der Firma *SUN Microsystems*²⁷. Diese J2ME ermöglicht Entwicklern Anwendungen zu programmieren, die auf leistungsschwache, mobile Endgeräte, wie PDAs und Handys, genau zugeschnitten sind.

Die in der *Location API* eingesetzte Technik zur Positionsbestimmung ist je nach eingesetzter Funktechnik verschieden. Bei WLAN zum Beispiel müssen zur Berechnung der Benutzerposition die genauen Daten (Position, Reichweite, Stärke, etc...) der einzelnen Access Points angegeben werden. Das Tracking läuft dann intern über die jeweiligen Signalstärken und nimmt dem Entwickler somit diese Arbeit ab. Zusätzlich zur genauen Position kann auch die Geschwindigkeit und die Richtung der Endgeräte berechnet werden.

Diese generische *Location API* kann auch anderen Anwendungen Informationen zur aktuellen Position der angeschlossenen Geräte zur Verfügung stellen. Zwei Möglichkeiten hat man dabei zur Auswahl:

²⁷Firmenwebseite von SUN Microsystems: <http://de.sun.com/homepage/index.html>

1. Ein Server verarbeitet die Daten und Informationen und leitet die Ergebnisse ans Endgerät oder an andere Programme weiter.
2. Die Daten werden direkt an das mobile Gerät oder an andere Programme geschickt und dort weiterverarbeitet.

Klassische LBS werden ebenfalls mit nützlichen Funktionen und Methoden unterstützt. So können beispielsweise *Landmarks* direkt als Objekte angelegt werden. Es ist nun möglich diese zu gruppieren und sie mit anderen J2ME Anwendungen zu teilen. Zur Sicherheit und zum Schutz der Privatsphäre über die gesammelten Benutzerdaten wird zudem empfohlen das Sicherheits-Framework der MIDP 2.0 zu benutzen.

Diese Variante ist kostenfrei, hinkt aber dafür bei der Genauigkeit den kommerziellen *Tracking-Engines* etwas hinterher und die benötigten Daten über die APs müssen von Hand ausgemessen werden.

3.5 Routenberechnung

Kommen wir nun zum zweiten Teil der Navigation: Der Berechnung der Route zum angestrebten Ziel. Da in einer Bibliothek mehrere Ziele ausgewählt werden können, muss eine optimale Rundreise berechnet werden, die alle Objekte auf dem kürzesten Wege ansteuert.

Zuerst muss der gesamte Grundriss in einzelne Wegabschnitte eingeteilt werden. Danach werden die Kosten für jeden dieser Abschnitte einzeln berechnet und zusammen mit Landmarks und weiteren Informationen in einer Datenbank oder

anderweitig gespeichert.

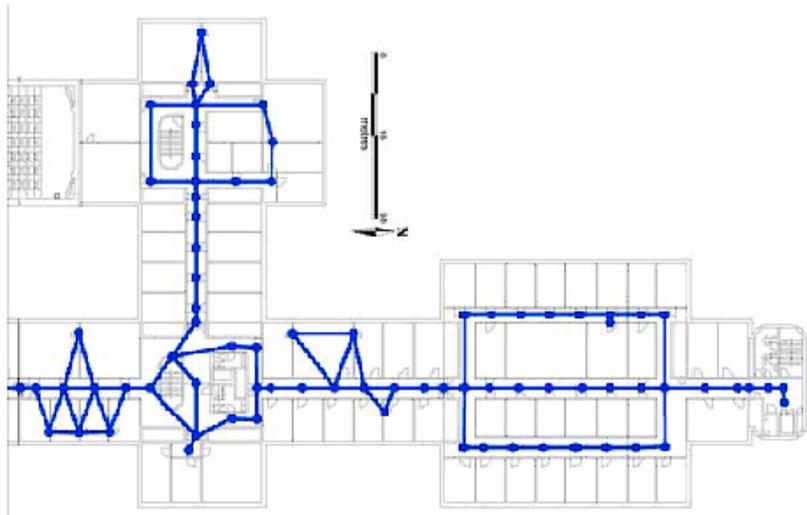


Abbildung 15: Einteilung in Wegabschnitte

Nun muss dieses Wegenetz in eine für den Computer verständliche Darstellung gebracht werden. Eine Datenstruktur, die dies unterstützt, ist ein aus Kanten und Knoten bestehender *gewichteter Graph*.

Die Knoten entsprechen dabei den Kreuzungen und stellen die Entscheidungspunkte dar. Die Kanten sind die möglichen Wege, die der Bibliotheksbesucher nutzen kann. Die Gewichtung der Kanten kann beispielsweise über die benötigte Zeit zum Ablaufen der Strecke oder über die Distanz vorgenommen werden. Zusätzlich können auch Zugangsberechtigungen und behindertenfreundliche Wege extra markiert werden. Die Kosten für Aufzüge und Treppen werden meist nach Durchschnittsreisezeit berechnet.

Für das Projekt dieser Bachelorarbeit muss man neben den Kreuzungen auch vor jedes Regal der Bibliothek einen solchen Wegpunkt setzen und die dort plat-

zierten Medien diesem zuordnen.

Die Entfernung zwischen zwei Wegpunkten berechnet man am besten indem man ein Gitternetz über den Grundriss legt und dann beispielsweise die *Euklidische Distanz* auf die Koordinaten dieser Punkte anwendet.

Über einen weiteren Algorithmus, wie dem *Point-to-curve Algorithmus*, kann die berechnete Benutzerposition einem Punkt auf der nächstliegenden Weglinie zugeordnet werden. Dadurch wird auch die Visualisierung einfacher und die Benutzerposition muss nicht 100% genau geortet werden, da man den Benutzer immer auf die vordefinierten Wege setzen kann.

Euklidische Distanz

$$\sqrt{(x_1 - y_1)^2 + (x_2 - y_2)^2 + \dots + (x_n - y_n)^2} = \sqrt{\sum_{i=1}^n (x_i - y_i)^2} \quad (2)$$

Point-to-curve Algorithmus

$$c = \sqrt{\frac{\left((y_1 - y_2)^2 x_p + (x_2 - x_1)^2 y_p + (x_1 y_2 - x_2 y_1) \right)^2}{(y_1 - y_2)^2 + (x_2 - x_1)^2}} \quad (3)$$

Abbildung 16: Empfohlene Algorithmen

Um die eigentliche Rundreise zu berechnen, verwendet man die vorher gewonnenen Kosten und implementiert einen für diese Zwecke schon bestehenden Algorithmus, wie zum Beispiel den *Traveling-Salesman-Algorithmus*. Dieser berechnet die optimale Reise von einem Startpunkt über mehrere Zwischenstationen und wieder zurück zum Ausgangspunkt.

4 Visualisierung und Designgrundlagen (Frontend)

4.1 Visualisierungstechniken

Folgende Techniken können verwendet werden, um das Interface des Clients für dieses Navigationssystem zu implementieren. Die Programmiersprachen bzw. Software bieten die Möglichkeit ansprechende 3D-Visualisierungen zu realisieren und erlauben es zudem, empfangene Daten selbstständig zu verarbeiten.

4.1.1 OpenGL

OpenGL ist eine von Silicon Graphics²⁸ (SGI) entwickelte hardware- und plattform-unabhängige Programmierschnittstelle zur Entwicklung von 3D-Applikationen im Profi-Bereich, welche heute vor allem im Spielbereich weit verbreitet ist. Sie unterstützt 2D- und 3D-Grafiken und ist für die meisten gängigen Programmiersprachen verfügbar. Um zwei davon zu nennen:

C / C++

Die typische OpenGL-Implementierung besteht aus einer dynamischen Programmbibliothek und einer C-Headerdatei. Somit ist die Nutzung in der Programmiersprache C und C++ problemlos möglich. Für die Anwendung auf dem PDA ist die Version OpenGL ES verfügbar.

Java

Java lehnt sich stark an die C-übliche Syntax an und unterstützt schon seit

²⁸Firmenwebseite von Silicon Graphics: www.sgi.com

langem 2D-Grafiken. Mit der für mobile Endgeräte zugeschnittenen Version *JSR 184 Mobile 3D Graphics API*²⁹ sind nun endlich auch 3D-Berechnungen möglich.

4.1.2 Flash / Swift 3D

Hierbei handelt es sich um ein von Macromedia³⁰ entwickeltes und inzwischen zum Standard avanciertes Animationsprogramm, mit dem interaktive Anwendungen erstellt werden können. Flash arbeitet vorrangig vektororientiert, wodurch eine hohe Komprimierung und damit ein schneller Datenaustausch erreicht wird.

Mit Flash lassen sich einfach und ohne viel Zeitaufwand 2D Grafiken und Interaktionselemente erstellen, die grafisch sehr ansprechend aussehen und sich durch die eingebundene Scriptsprache ActionScript interaktiv steuern lassen.

Mit dem dafür entwickelten Zusatzprogramm Swift 3D lassen sich Animationen von dreidimensionalen Objekten in Flash einbinden.

Ein weiteres Feature ist ein Plug-In für Swift mit dem vormodellierte Objekte aus dem Programm *3D Studio Max*³¹ eingebunden werden können. Mit diesem kann man noch detaillierter modellieren und wird daher im Profi-Bereich häufig verwendet.

4.1.3 VRML (JavaScript/XML)

VRML ist eine Abkürzung für *Virtual Reality Modeling Language* (früher *Virtual Reality Markup Language*). Hierbei wird das Objekt (2D oder 3D) mitsamt seiner Bewegungen beschrieben. Es ist ein Dateiformat für die Beschreibung von

²⁹Final Release: <http://www.jcp.org/aboutJava/communityprocess/final/jsr184/>

³⁰Seid 2005 würde Macromedia von der Firma Adobe übernommen. URL: <http://www.adobe.com/>

³¹3D Grafik- und Animationsprogramm der Firma Autodesk. URL: <http://www.autodesk.de>

interaktiven, dreidimensionalen Welten, welche dann über JavaScript individuell gesteuert werden können. Diese Technik lässt sich sehr gut nutzen, um 3D Szenen mit intelligenten Objekten oder Guided Tours mit interaktivem Eingreifen zu realisieren. Viele 3D-Programme ermöglichen den Im- und Export von VRML-Dateien, wodurch sich das Dateiformat auch als ein Austauschformat von 3D-Modellen etabliert hat.

Zwar brauchen die damit entwickelten Programme viel Speicher, die Darstellung kann dafür aber fast fotorealistisch werden. Dabei berechnet der Computer die "virtuelle Realität" in Echtzeit anhand den vorhandenen Geometriedaten, sowie dem Verhalten und den Bewegungen des Besuchers.



Abbildung 17: Anwendungsbeispiel

Für die Wiedergabe der Szene benötigt man einen geeigneten Browser oder ein PlugIn für einen der Standardbrowser. Aktuell ist die Version 2.0 dieser Sprache.

Der Nachfolger heißt X3D, wobei VRML mit XML-Syntax eingesetzt wird. Entwickelt wurde er vom Web3D Konsortium³², das sich mit Web 3D Technologien beschäftigt und mehrere Arbeitsgruppen, wie z.B. die X3D Working Group, beinhaltet.

³²Webseite des Web3D Konsortiums: <http://www.web3d.org/>

4.2 Designgrundlagen

In diesem Abschnitt der Arbeit wird nun der dritte Teil der Navigation behandelt: die Visualisierung der Benutzerposition, der Routenführung und die Darstellung der Umgebung auf dem Display. Das Frontend ist besonders wichtig, da es für den Benutzer, der von den Vorgängen im Hintergrund nichts mitbekommt, das "eigentliche" Navigationssystem darstellt. Außerdem soll die Visualisierung das eigentliche Novum zu den bisherigen vorhandenen Systemen darstellen.

Die Entscheidung, nicht eine klassische, zweidimensionale Visualisierung, sondern das Umgebungsmodell in 3D darzustellen, fiel einerseits aufgrund der Tatsache, dass nicht die bestehenden Systeme nachgeahmt werden sollen, andererseits, weil eine 2D-Ansicht im mobilen Kontext nicht optimal ist. Die Erweiterung um eine Dimension kann einen erheblichen Informationsgewinn darstellen, da der Benutzer die Objekte um ihn herum viel leichter erkennt, als in einer einfacher 2D-Obenansicht mit symbolhafter Darstellung der Gegenstände im Raum. Dies zeigen auch Erfahrungen aus virtuellen Welten auf dem Sektor der Computerspiele.

4.2.1 Landmarks

Landmarks auf deutsch Landmarken, sind hervorstechende Merkmale in einer Umgebung. Dies ist hilfreich für den Benutzer, um seinen eigenen Standort besser bestimmen zu können. Mögliche Landmarken im Innenbereich einer Bibliothek wären zum Beispiel: Informationsstand, Treppe, Aufzug, PC-Arbeitsplatz, Türen, Flur-Kreuzungen, Kopierer, grosse Uhr bzw. Telefon an der Wand, usw... Diese müssen in der Darstellung der digitalen Welt deutlich denen in der echten Umgebung entsprechen, sie müssen vom Anwender sofort wiedererkannt werden.

Die englische Universität Loughborough hat umfangreiche Benutzertests zu die-

sem Thema durchgeführt.

Sie teilten die Probanden in zwei gleich große Gruppen ein, wobei alle Altersgruppen vertreten waren. Eine Gruppe bekam eine Wegbeschreibung mit eingezeichneten Landmarken, die andere Gruppe musste das Ziel ohne Hilfe dieser finden [Ross 2004]. Das Ergebnis dieser Untersuchung war, dass bei der Gruppe mit Landmarken:

- die Unsicherheit geringer war.
- die Navigations-Performance verbessert wurde (schneller).
- weniger Navigationsfehler begangen wurden.

4.2.2 Detailierungsgrad

Wie soeben gehört, ist es wichtig zur Positions- und Wegfindung, dass der Benutzer seine Umgebung auf dem Display des Endgerätes wiedererkennt. Dazu muss man bei der Detaillierung der Objekte im System beachten, dass sie möglichst dem Original entsprechen, oder zumindest als dieses identifiziert werden können. Natürlich wäre hierbei eine fotorealistische Darstellung wünschenswert, allein schon aus dem Grund, weil viele Anwender ein System anhand der grafischen Oberfläche bewerten. Dennoch muss man hier den Spagat zwischen einer möglichst detaillierten Darstellung und den Fähigkeiten der Endgeräte, wie einem PDA, schaffen. Diese sind momentan noch beschränkt durch ihre relativ kleinen Arbeitsspeicher, was dem Designer nicht die Freiheiten lässt, die er auf einem Desktop-PC hätte. Deshalb muss man genau überdenken, an welche Stellen man hochwertige Texturen und an welchen man eher einfache Objekte benutzt.

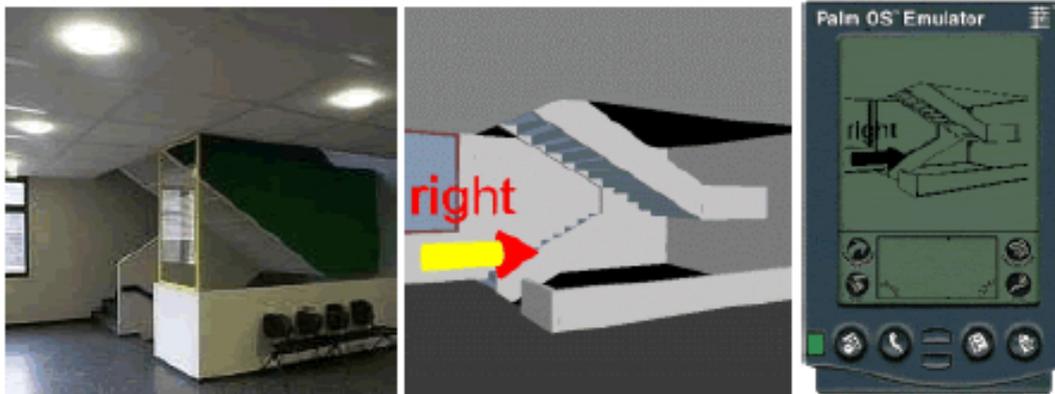


Abbildung 18: Detaillierungsgrad [Butz 2001]

4.2.3 Sichtweise / Perspektive

Hat man das 3D-Modell, die Visualisierung der Umgebung also, abgeschlossen, geht es nun darum, die Sichtweise auf die darzustellende Benutzerposition und die zu gehende Wegstrecke festzulegen.

Da die Literatur und die Praxis in Fragen der Perspektive viele unterschiedliche Meinungen vertreten, sollen dem Benutzer bei diesem Projekt mehrere Sichten angeboten werden, aus denen er die von ihm favorisierte selbst auswählen und diese auch während dem Betrieb jederzeit wechseln kann.

Drei Sichtweisen sind in der Praxis immer wieder zu finden. Die erste ist die 90°- Obenansicht, wie bei der klassischen 2D-Visualisierung, allerdings sieht diese im dreidimensionalen Modell um einiges ansprechender aus, hervorgerufen durch den plastischen Eindruck der virtuellen Umgebung. Die zweite ist eine schräge Draufsicht, die in den meisten KFZ-Navigationssystemen als "2 1/2" umgesetzt ist. Dabei ist die zu Grunde liegende Karte als Ebene dargestellt und die wichtigen Elemente treten zur besseren Erkennung aus dieser heraus. Gut zu sehen in

der nachfolgenden Grafik:



Abbildung 19: Sony Navigationssystem - Beispiel 2 1/2 D [Sony 2004]

Die dritte Sichtweise die ich hier nennen möchte ist die Ego-Perspektive, welche bisher erst in den wenigsten KFZ-Navigationssystemen zu finden ist.

Das neue Navigationssystem von Sony hat diese in ein Hybrid-Modell eingebaut um jeweils die entscheidenden Navigationspunkte deutlicher darzustellen. Diese werden dabei in einem kleinen Fenster ins obere Eck des Screen auf das herkömmlichen Navigationssystem gelegt.

Hybrid-Modelle³³ sind eigentlich sehr sinnvoll, ist man wie bei einem PDA jedoch vom Platz sehr eingeschränkt, kann man jeweils immer nur eine Sicht einstellen.

³³Weiteres Hybrid-Modell im Anhang



Abbildung 20: Sony Navigationssystem - Hybrid Modell [Sony 2004]

Wobei sich in Gebäuden die Ego-Perspektive anbietet, da sie dem Benutzer genau das zeigt, dass er momentan mit seinen eigenen Augen auch sieht. Somit kann er sich noch leichter in der Umgebung zurechtfinden. Ein geeignetes Beispiel hierfür sind die sog. "Ego-Shooter" aus dem Bereich der Computerspiele.

4.2.4 Blickfeld

Um eine gute Ego-Perspektive zu simulieren muss man das genaue Blickfeld des menschlichen Auges kennen. Mit 200°horizontal und 130°vertikal entspricht es einer Kamera-Linsenlänge von 30-35mm, welche in dem verwendeten 3D-Grafikprogramm vorher eingestellt werden muss.

Zudem darf die Kamera nicht genau auf der Benutzerposition platziert werden, sondern soweit zurückgezogen werden muss, als ob man dem virtuellen Benutzer über die Schulter schauen würde. Genaue Richtlinien hierfür gibt es jedoch nicht, dies muss der Designer nach Gefühl entscheiden. In diesem Fall habe ich mich

entschieden die Kamera in aufs Modell übertragenen 1m hinter der Benutzerposition und auf Höhe von 1,60m anzubringen. Die Regale hatten dabei eine Höhe von zwei Metern.



Abbildung 21: Prototyp - Kamera-Linsenlänge 30mm

4.2.5 Ausrichtung der Karte

Bei 2D bzw. 3D-Obenansichten gibt es zwei Möglichkeiten die Benutzerposition auf der Karte zu bewegen. Die Erste Variante benutzt eine festorientierte "genordete" Karte, welche immer den gleichen Anblick mit gleicher Ausrichtung hat. Die Einschätzung der Topologie ist dadurch besser, jedoch fällt die Orientierung um einiges schwerer als bei der zweiten Variante, der dynamischen Ansicht.

Dabei dreht sich die Karte dynamisch mit der Benutzerbewegung. "Links" auf der Karte entspricht dabei auch "links" aus Sicht des Anwenders in der realen

Umgebung. Dieser findet sich auf solch einer Ansicht viel schneller und leichter zurecht, da er nicht erst umdenken muss.

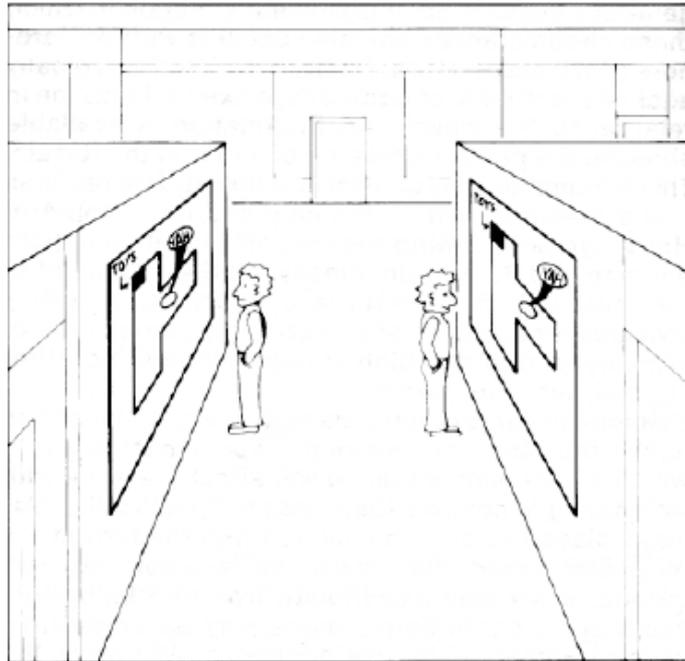


Abbildung 22: Festorientierte "genordete" Karte

5 Benutzerorientierter Ansatz

Möchte man neue Techniken als Hilfe in Arbeitsbereiche einführen, die bis dato ohne solch technische Unterstützung ausgekommen sind, so ist die Akzeptanz der Innovation durch den Benutzer von enormer Wichtigkeit. Dieser muss im Hauptfokus stehen. Dabei ist das primäre Ziel, ihn durch die Neuerung soweit zu unterstützen, dass er gerne wieder auf diese zurückgreift.

Bei einer navigationsgestützten Suche erreicht man dies nur, wenn der Benutzer mit besagter Hilfe die von ihm gesuchten Objekte auch wirklich schneller findet, als auf herkömmliche Art und Weise. Sind die hierbei gelieferten Anweisungen und Informationen für den Anwender klar und unmissverständlich, so findet er mit technischer Unterstützung den für ihn optimalen und schnellsten Weg zu seinem Ziel, was wiederum heißt, dass eine zeitaufwendige Suche entfällt.

Um dies zu erreichen, muss die Benutzbarkeit des neuen Systems den Anforderungen und Bedürfnissen der Zielgruppen genau angepasst werden. Sobald diese feststellen, dass ihnen das System die Ausführung bestimmter Aufgaben erleichtert und sie zusätzlich hilfreiche Dienste angeboten bekommen, hat die erfolgreiche Etablierung dieser neuen Technik beste Chancen. Allerdings gilt es zu beachten, besagte, zusätzlich angebotene Funktionen sorgsam auszuwählen, da der Benutzer durch eine Überhäufung von Möglichkeiten von seinem eigentlichen Ziel nicht abgelenkt werden soll. Das System muss folglich schlicht und zielgerichtet sein, um eine möglichst einfache und verständliche Nutzung zu gewährleisten und somit die angestrebte Zeiteinsparung bei einer Suche zu erzielen.

Angesichts dieser Problematik geht das folgende Kapitel auf die Anforderungen

der Benutzer ein und zeigt, was diese bei der Durchführung einer navigationsgestützten Suche benötigen.

5.1 Benutzerprofile

Bei den Anwendern der navigationsgestützten Suche handelt es sich vorzugsweise um zwei Gruppen von Bibliotheksbenutzern.

Eine erste Gruppe, und mit ihr der Hauptnutzer dieses Systems, sind die Studenten der Universität Konstanz. Diese gehören zu einer Generation, die in der Regel recht vertraut ist im Umgang mit modernen Informations- und Kommunikationstechnologien, wie, z.B. Internet, Handy, Computer usw... . Speziell der Umgang mit dem Internet wird gefördert und gefordert, da sich das Universitätsleben immer stärker Online organisiert. Als Beispiel hierfür lässt sich das bereits bestehende elektronische Lernsystem *moodle* oder das Konstanzer Ausleih- und Anfragesystem *Koala* aufführen. Es wird daher angenommen, dass die Aufnahmebereitschaft für ein neues System, welches das studentische Leben weiter erleichtert, groß sein wird.³⁴

Die zweite Gruppe beinhaltet die Angestellten der Universität, welche meist nicht in gleichem Umfang mit den erwähnten Informations- und Kommunikationstechnologien aufgewachsen sind, wie die Gruppe der Studenten. Zu ihnen zählen die Professoren, ihre Assistenten, die Sekretärinnen, der Verwaltungsbereich sowie alle weiteren Mitarbeiter der Universität.

Von beiden Gruppen wird angenommen, dass ihnen bereits navigationsgestützte Suchmöglichkeiten und ihre Vorteile, vornehmlich durch die Autonavigation, bekannt sind und sie sich daher ähnliche Erleichterungen bei der Anwendung im

³⁴Vgl. hierzu die bereits erwähnte Untersuchung zur Smartlibrary der Universität Oulu

Indoor-Bereich vorstellen können.

Allerdings haben bisher vermutlich die Wenigsten Erfahrungen mit Navigationssystemen gemacht, kennen aber meist Darstellungen aus den Medien und wissen daher in etwa, wie solche funktionieren. Von Technik-Laien wird im schlimmsten Fall ein kompliziertes und ungewohntes Programm erwartet, mit einer undurchschaubaren Technik. Diese Ängste müssen dem Benutzer durch ein einfach aufgebautes Programm bereits nach kürzester Bedienzeit genommen werden.

5.2 Benutzbarkeits-Kriterien

Kommen wir nun zu den Hauptkriterien, welche eine hohe Benutzbarkeit (*Usability*) dieses Produktes sichern sollen. Folgende Eigenschaften sehe ich als unabdingbar an, um dieses Ziel zu erreichen:

- **Effektivität**

Unter effektivem Arbeiten versteht man in diesem Fall, dass der Benutzer schnellen Zugang zu allen Informationen hat, die er zur Suche nach einem Medienobjekt benötigt und durch verständliche Anweisungen schlussendlich auch sein Ziel erreicht.

- **Effizienz**

Das System soll eine hohe Produktivität aufweisen, d.h. die Suche soll schnell und immer erfolgreich abgeschlossen werden.

- **Leicht zu erlernen**

Da der Anwender selten Erfahrungen auf diesem Gebiet hat und Zeit eine erhebliche Rolle spielt, muss das Programm und das Interface leicht verständlich und der Umgang mit diesem einfach zu erlernen sein. Hat er

dieses System einmal bedient, so soll er die Vorgehensweise und Funktionen beim nächsten mal sofort wiedererkennen können.

5.3 Benutzer-Erfahrungen

Folgende Erfahrungen sollte der Benutzer mit Hilfe der technischen Unterstützung des Navigationssystems machen, um dieses in positiver Erinnerung zu behalten, um somit eine weitere Nutzung zu erreichen:

- **Befriedigend**

Der Benutzer soll ein befriedigendes Gefühl über den gesamten Ablauf der Suche haben.

- **Spaß und Unterhaltung**

Es soll Spaß machen, das System zu bedienen. In gewisser Weise entspricht die Vorgehensweise einer Schnitzeljagd und kann vom Benutzer als unterhaltsam wahrgenommen werden.

- **Hilfreich**

Mit Hilfe des Navigationssystems soll der Benutzer seine Aufgabe schneller erfüllen können als auf herkömmliche Art.

- **Belohnend**

Belohnt wird er mit einer erfolgreich abgeschlossenen Suche.

5.4 Konzeptionelles Modell

Bei dem Konzept dieses Modells basiert die Steuerung des Programms auf der Bewegung des Anwenders. Da der Benutzer mit seiner eigenen Körperbewegung das Programm beeinflusst und die daraus resultierende Informationen direkt ans

Display geschickt werden, handelt es sich hier um das klassische Direct Manipulating.

Des Weiteren soll durch kontinuierliche Präsenz der Bedienungselemente und der Darstellung der Aktionen auf dem Interface ein schnelles Erlernen der Basisfunktionen für Anfänger erleichtert werden. Für unregelmäßige Besucher soll durch eben diese Präsenz ein schneller Wiedererkennungswert und somit eine Erinnerung der Bedienungsabläufe sichergestellt werden.

Da dieses Modell rasche, umkehrbare und aufzählbare Aktionen mit sofortigem Feedback über den Objekten unterstützt, sind auch Fehlermeldungen kaum nötig. Es ist sofort ersichtlich, ob die Aktionen zum gewünschten Ziel geführt haben oder nicht. Dadurch soll dem Benutzer das Gefühl vermittelt werden, alles unter Kontrolle zu haben, wobei der Umgang mit dem System vertrauter und sicherer wird.

5.5 Anforderungen und Werkzeuge

Kommen wir nun zu den funktionalen Anforderungen und Werkzeugen, die der Benutzer bei seiner Suche benötigen könnte. Hierbei muss man vorsichtig sein, um das System nicht unnötig zu verkomplizieren, da der Benutzer durch zu viele Funktionen verwirrt oder abgelenkt werden könnte. Aus diesem Grund sollte man auf verschachtelte Menüs verzichten. Alle wichtigen Funktionen werden am besten auf direkt sichtbare Buttons auf dem Display installiert. Da allerdings mit PDAs gearbeitet werden soll, sind die Möglichkeiten dabei aufgrund des Platzmangels auf dem kleinen Screen begrenzt.

Folgend die von mir für die Navigation als wichtig eingestuften Informationen und Funktionen:

- Name und Regalnummer des nächsten Ziels
- Entfernung zum nächsten Ziel
- Umschalten der Ansicht
- Obenansicht, um sich einen Überblick über die Ziele und die zurückgelegte Strecke zu machen
- Liste mit allen Zielen (Priorisierung, Elemente löschen und hinzufügen, Detailinformationen, direkter Zugang zu MedioVis)
- Einstellungen / Optionen (Route laden, Route speichern, Grafikoptionen u.U. für Farbenblinde, Text-Anweisungen ab-/anschalten, Anzeigen ein-/ausblenden, behindertengerechte Wege, Legende anzeigen, Beenden)
- Die genaue Position innerhalb des gefundenen Regals anzeigen. (Dies benötigt bei der herkömmlichen Suche die meiste Zeit)
- Weitere Informationen über die Bibliothek

Optionale Funktionen, die nicht primär benötigt werden:

- Mehrere Zoomstufen, d.h. der Benutzer hat die Möglichkeit den Ausschnitt, den er sieht, zu vergrößern bzw. zu verkleinern, mit der Möglichkeit in seine Navigationskarte herein- und herauszuzoomen.
- Vorschau über den Weg, den man zurücklegen muss. Einen virtuellen Flug zum nächsten Ziel, um so dem Benutzer den Weg zu verdeutlichen, den er gleich gehen soll.

- Anzeige der Geschwindigkeit
- Künstliche Landmarks setzen, wenn keine natürlichen vorhanden sind (Nummern, Farben, Schilder, etc.)
- Zusätzlich zu den Richtungspfeilen auch kurze Beschreibungen und Anweisungen in Textform (an den Landmarks orientieren), z.B.: "Den Gang vor bis zur Treppe, dann ein Stockwerk hoch.", "Vor bis zum Informationsschalter, dann links."
- Vorgefertigte Rundreise von der MedioVis Desktop-Version laden und speichern können.
- Mögliches Feature für die Desktop-Version: Berechnete Rundreise mit allen Zielen und Landmarks in einer 2D Ansicht drucken und diese ohne PDA-Unterstützung ablaufen.

5.6 Anwendungs-Szenario

Ein mögliches Szenario bei der Benutzung dieses Systems könnte in etwa so aussehen:

Ein Student hat sich im Mobilen MedioVis drei DVDs ausgesucht und möchte diese nun suchen. Es ist das erste Mal, dass er dieses System bedient. Um die Suche zu beginnen klickt er direkt im MedioVis den Button <Navigationssystem> und startet so das Programm. Übergeben wird dabei automatisch seine aktuelle Suchliste. Anhand dieser Liste berechnet das System eine Route und zeigt dem Benutzer zuerst einen Willkommenstext und einen Vorschlag zur optimalen Rundreise. Hier kann der Benutzer die Suchreihenfolge der Objekte noch einmal

ändern oder einfach mit <Start> die Suche beginnen. Er startet die Suche und läuft los. Auf dem Weg zu seinem ersten Ziel wählt er sich über die Werkzeugleiste die für ihn passende Ansicht. Hat er sich dem Regal bis auf ein, zwei Meter genähert, öffnet sich ein Pop-Up Fenster auf dem er dieses Regal in Großansicht sieht. Hervorgehoben ist farblich und blinkend der Bereich, in dem sich die DVD befindet. Der Student hat sein Objekt gefunden und schließt das Detailansicht-Fenster. Das System fragt ihn nun, ob dieses Ziel in der Liste abgehakt werden darf oder ob das eben gefundene Objekt an das Ende der Liste gesetzt und später noch einmal angesteuert werden soll. Er bestätigt den Abschluss der ersten Suche und kommt sofort zur Suche des zweiten Objektes. Er öffnet die Gesamtansicht, um zu sehen, wo er bisher schon war und wie weit er noch bis zu den anderen Zielen laufen muss. Durch die weiteren Buttons auf der Werkzeugleiste findet er Informationen über die Bibliothek, sowie die Möglichkeiten weitere Einstellungen vorzunehmen. Der Student fährt dann mit der weiteren Suche fort. Usw ...

6 Prototyp

In diesem Kapitel werde ich nun den zu diesem Projekt entwickelten Prototypen vorstellen. Es handelt sich dabei um eine Desktop-Version, bestehend aus zwei Teilen. Der größere und wichtigere Teil stellt eine mögliche Visualisierung des Navigationssystemes dar, wie es auf einem handelsüblichen PDA laufen könnte. Der zweite Teil ist ein Steuerungsprogramm, das die Benutzerposition simuliert und jetzt im Anschluss vorgestellt wird.

6.1 Steuerungsprogramm

Das Steuerungsprogramm orientiert sich an dem Verfahren zum WLAN-Tracking. Wie bei den echten Testmessungen und der Einteilung der kompletten Strecke in

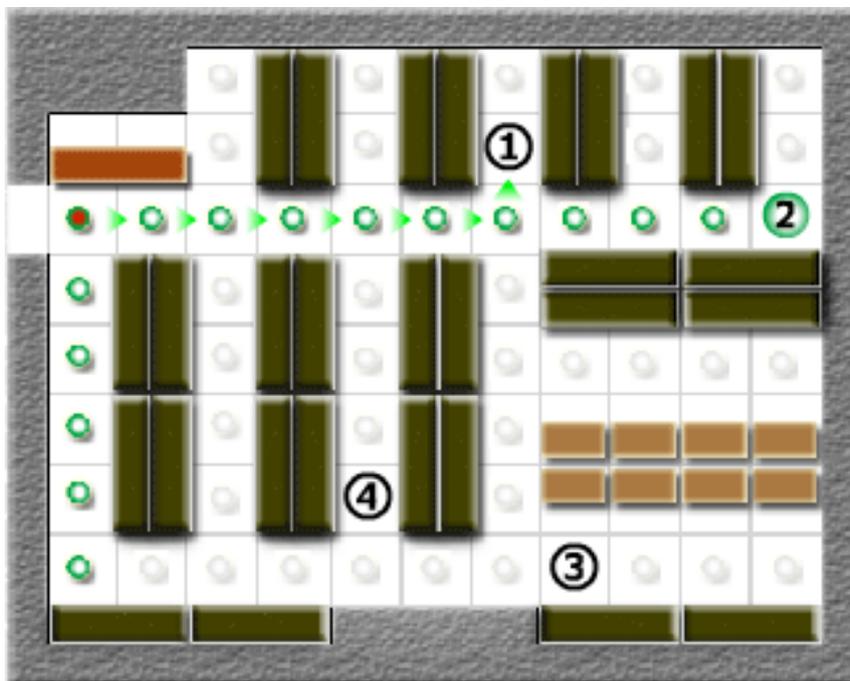


Abbildung 23: Prototyp - Steuerungsprogramm

einzelne Entscheidungspunkte zur Routenberechnung, so wurde auch bei diesem

Simulationsprogramm jeden Meter ein Wegpunkt gesetzt.

Über diese Wegpunkte kann nun der virtuelle Besucher durch die Bibliothek bewegt werden. Die Streckenabschnitte in seinem Sichtfeld, die er ansteuern könnte, sind dabei grün eingefärbt, die restlichen in grau. Der Benutzer selbst stellt den roten Punkt dar.

Die einzelnen Ziele werden in der Reihenfolge, wie sie annavigiert werden sollen, durchnummeriert.

Klickt man nun auf einen dieser grünen Wegpunkte, so bewegt sich der Benutzer in simulierter Schrittgeschwindigkeit auf diesen zu und die entsprechende Visualisierung mitsamt den Navigationspfeilen wird auf dem virtuellen PDA abgespielt.

6.2 Programmierung

Realisiert wurde der komplette Prototyp in Macromedia Flash, wobei beide Teile in eigenständigen SWF-Dateien³⁵ gespeichert und in einer Hauptdatei eingebunden wurden. In dieser wurde zu der Beschriftung der Einzelteile auch dem Steuerungsprogramm zur Erklärung noch eine Legende beigefügt. Ebenso wurde das Navigationsdisplay von dem Bild eines PDAs gerahmt, um zu verdeutlichen, wo dieser Programmteil später einmal laufen sollte.³⁶

Die einzelnen Gänge in der Visualisierung des Navigationssystems, das zu Grunde liegende 3D-Modell also, wurde in dem Programm *SWIFT 3D* der Firma *Electric Rain*³⁷ erzeugt. Diese Software kann komplexe 3D-Welten und Animationen generieren und in dem für Flash gängigen Format SWF abspeichern.

³⁵Small Web Format - abspielbare Flashanimation

³⁶Screenshot des gesamten Systemes im Anhang

³⁷Firmenwebseite von Electric Rain: <http://www.erain.com/>

Für jeden Gang wurden nun 6 unterschiedliche virtuelle Kameraflüge aufgezeichnet und in einer eigenen SWF-Datei abgelegt. Die komplette Testumgebung besteht also aus 54 Animationen. Pro Gang jeweils für die 90°-, 45°- und die Ego-Perspektive in beide Richtungen.

Dabei entsprechen 20 Frames in solch einem Kameraflug einem Meter in der ech-

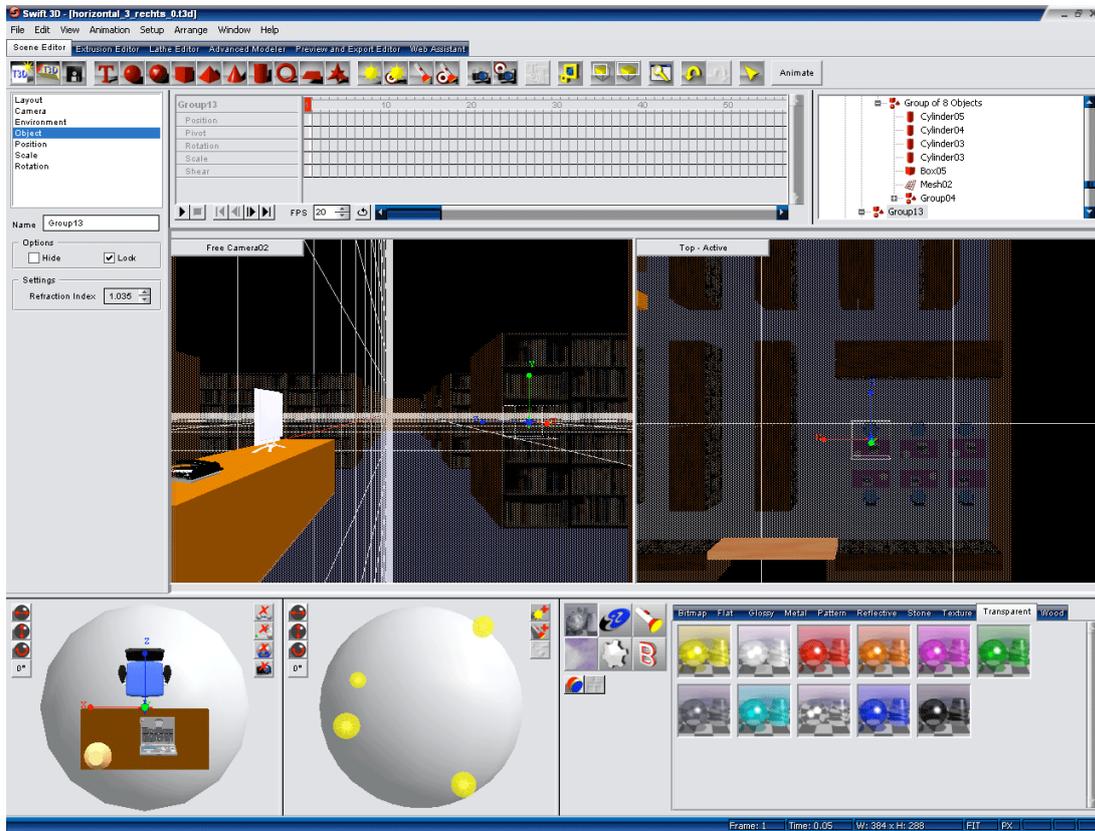


Abbildung 24: Screenshot Swift 3d

ten Umgebung ebenso, wie im Steuerungsprogramm. Angenommen wird weiter, dass der Benutzer eine Sekunde benötigt um diesen zurück zu legen.

Diese Technik ermöglicht es, dass jederzeit zwischen den Sichten gewechselt werden kann, auch während der Bewegung, da man immer die entsprechenden Frames der Animationen ansteuern kann. Dies geschieht auch bei der simulierten Bewe-

gung im Steuerungsprogramm. Bei Klick wird die entsprechende Animation auf dem "PDA" gestartet und dort das benötigte Frame angewählt.

Um einen flüssigeren Ablauf dieser zu ermöglichen werden alle Animationen beim Start der PDA-Visualisierung vorgeladen, so dass diese verfügbar sind, sobald sie benötigt werden. Darauf werden dann in Flash die Anzeigen, die Navigationselemente und die Werkzeugleiste gelegt.

6.3 PDA-Handhabung

Zur Handhabung des PDAs lässt sich sagen, dass nur bei einer von drei Sichten, nämlich der Obenansicht, es vorteilhaft wäre, den PDA im Hochformat zu halten, da man dort mehr von der zu Laufenden Strecke erkennt.

Bei den anderen zwei Sichten und für die Anordnung der restlichen Elemente, wie den Anzeigen und der Werkzeugleiste, ist es optimaler, den PDA im Querformat zu halten, weshalb sich auch diese Variante letztendlich durchgesetzt hat.

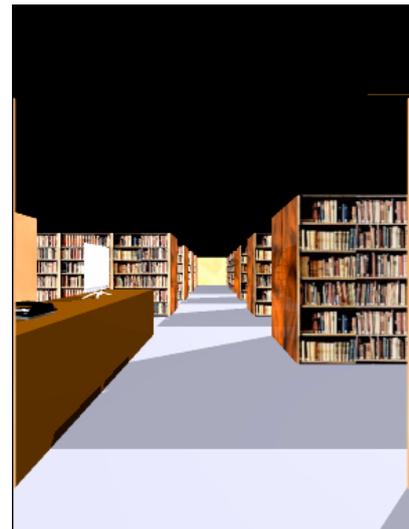


Abbildung 25: Hochformat

6.4 Anzeigen

Da es die Absicht war, das Navigationssystem möglichst schlicht aufzubauen wurden nur die zur Navigation wichtigen Informationen und Werkzeuge auf dem Display angezeigt. Zum einen, weil das Display beschränkt ist, zum anderen, weil der Benutzer nicht durch eine Vielzahl von Informationen oder Funktionen von seiner

eigentlichen Aufgabe abgelenkt werden soll.



Abbildung 26: Screenshot Prototyp

Oben links wird der Name und das Regal des momentan angesteuerten Ziels angezeigt. Anhand der forlaufenden Regalnummern, wie sie an den physischen Regalen in der Bibliothek angebracht sind kann der Besucher, ähnlich wie bei Hausnummern in einer Strasse, zusätzlich kontrollieren ob er sich auf das von ihm gesuchte Regal zubewegt.

An der rechten oberen Ecke wird die Entfernung zum nächsten Ziel angegeben, während das untere Ende des Displays die Werkzeugleiste füllt.

6.5 Werkzeugleiste

Nun die Beschreibung der Funktionen auf der Werkzeugleiste. Diese wurde in fünf Teile eingeteilt.

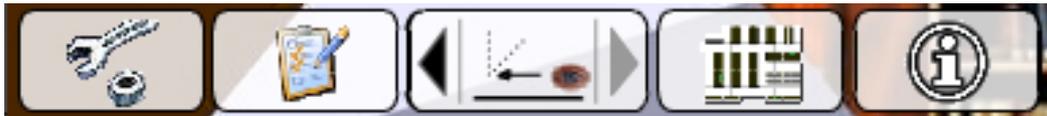


Abbildung 27: Prototyp - Werkzeugleiste

Von links nach rechts: *Optionen, Suchliste, Auswahl der Sichtweise, Gesamtsicht, Informationen.*

6.5.1 Optionen

Unter dem Punkt Optionen finden sich folgende Unterpunkte: "Einstellungen", "Suchliste laden", "Suchliste speichern" und "Programm beenden". Bei den Einstellungen hat der Benutzer zuerst einmal die Möglichkeit nicht benötigte Displayelemente an- oder abzuschalten. Des Weiteren soll es im fertigen System noch einen sehr wichtigen Punkt für körperlich eingeschränkte Menschen geben, mit dem sie die Option "Behindertengerechte Wege" aktivieren können und das System so die Routenberechnung anpasst. Beispielsweise, um Aufzüge anstelle von Treppen zu benutzen.

Bei den Punkten "Suchliste laden" und "Suchliste speichern" soll sich, ähnlich wie bei den Betriebssystemen, ein Dateibrowser öffnen, um eine Datei, die eine gespeicherte Suchliste beinhaltet, auszuwählen oder abzulegen. Über "Programm beenden" kehrt man zum Betriebssystem des PDA zurück.

6.5.2 Suchliste

Hier ist die Ergebnisliste zu finden, die dem Navigationssystem von MedioVis übergeben wurde. Diese soll der Benutzer jederzeit editieren können. Er soll zum Beispiel die Elemente der Liste löschen, neue Elemente durch einen Direktzugang zu MedioVis hinzufügen und die Reihenfolge der noch abzulaufenden Ziele beliebig ändern können. Letzteres gibt dem Anwender einen Hinweis, dass er, sobald er die Reihenfolge ändert, von der optimalen Rundreise abweicht. Dies kann im Extremfall zur Folge haben, dass er an Objekten vorbei läuft, die er später noch einsammeln möchte. Zusätzlich kann er sich bei Klick auf ein Listenelement genaue Informationen zu diesen Medien anzeigen lassen, wie beispielsweise Regisseur, Spieldauer, etc...

6.5.3 Auswahl der Sichtweise

Dieses Element besteht aus drei Teilen. In der Mitte wird anhand eines Pfeils dargestellt, welche Sichtweise gerade eingestellt ist. Mit den Buttons *links* und *rechts* kann man jeweils zur nächsten Ansicht wechseln. Ist dies nicht mehr möglich, dann wird dieser Button ausgegraut.

6.5.4 Gesamtansicht

Hier erkennt der Benutzer auf einem 2D-Grundriss, wo in der Bibliothek er schon war und wo er sich momentan befindet. Mit dieser Ansicht bekommt er einen Überblick über die Räumlichkeiten des Areals.

6.5.5 Informationen

Die wichtigsten Informationen zur Bibliothek findet der Besucher bei diesem Punkt. Primär interessant dürften ihm dabei die Öffnungszeiten der Bibliothek

und Mediothek, sowie die Ausleihinformationen zu den unterschiedlichen Objekten sein. Falls er noch Fragen zur Bedienung des Navigationssystems hat, so findet er die Antworten hier in einer kleinen Bedienungsanleitung. Für technisch Interessierte könnte man an dieser Stelle noch einen Überblick über die verwendeten Techniken geben.

6.6 Darstellung des Benutzers

Eine erste Überlegung war, den Anwender in Form eines Avatars³⁸ in die virtuelle Welt zu setzen. Allerdings haben nur die neuesten PDAs einen Kompass eingebaut, weshalb bisher noch nicht festgestellt werden kann, in welche Richtung der Benutzer gerade schaut. Man kann nur aus den vorhergehenden Bewegungen ableiten, wohin er gerade läuft. Deshalb wird in der 45°- und 90°- Ansicht eine Kugel zur Darstellung der Benutzerposition verwendet.

6.7 Navigationselemente

Grüne Richtungspfeile wurden je nach Sichtweise dezent auf den Boden vor die Benutzerposition gelegt, um möglichst wenig der Umgebung zu verdecken. Um dem allgemeinen Design treu zu bleiben, wurden diese ebenfalls leicht dreidimensional modelliert.

Nähert man sich bis auf 2m an eine Kreuzung, an der man abbiegen muss, so wird dieses Manöver durch einen abgebogenen Pfeil angekündigt. Nähert man sich der Kreuzung bis auf einen Meter, bzw. befindet sich dort, so wird mit einem direkten Linkspfeil angezeigt, in welchen Gang der Besucher einbiegen soll.

Läuft der Benutzer jedoch in die falsche Richtung, so bekommt er mit einem

³⁸Digitales menschliches Abbild

roten Pfeil angezeigt, dass er umdrehen muss und auf den vorgeschriebenen Pfad zurückkehren soll.

6.8 Ablauf einer Suche

Das Navigationssystem wird von der *Mobiles Medio Vis*-Ergebnisliste aufgerufen, sobald der Anwender seine zu suchenden Objekte ausgewählt hat und den Button <Navigationssystem starten> betätigt. Dabei wird diese Ergebnisliste als Suchliste an das Navigationssystem übergeben. Alternativ könnte er sich schon mit seiner Desktop-Version diese Liste erstellen und später dann im Navigationssystem laden.

Mit einer Begrüßungsmeldung wird der Benutzer dort empfangen und bekommt noch einmal die Suchliste angezeigt, wobei diese in der vom System berechneten optimalen Reihenfolge vorsortiert ist. Er hat zwar noch einmal die Möglichkeit, diese zu ändern, jedoch bekommt er hier auch die Meldung, dass er zusätzliche Wegstrecken in Kauf nimmt, wenn er von der optimalen Route abweicht.

Er startet nun in Richtung erstes Ziel, wobei er auch während der Bewegung jederzeit die Perspektiven wechseln und die gebotenen Funktionen auf der Werkzeugleiste nutzen kann. Passiert es nun, dass der Benutzer, während er die Werkzeuge aufgerufen hat, von der richtigen Route abkommt, werden alle anderen geöffneten Fenster geschlossen. So wird der Benutzer sofort über seinen Fehler informiert und kann diesen schnell rückgängig machen, ohne längere Zeit in die falsche Richtung zu laufen.

Hat er sein Ziel erreicht, öffnet sich ein Fenster mit allen nötigen Informationen, um das gesuchte Objekt schnellstmöglich zu finden. Dazu gehören die Regalbezeichnung, die genaue Signatur des Objekts und, was im Vergleich zur klassischen

Suche einen erheblichen Vorteil darstellt, ist die genaue Standortkennzeichnung innerhalb des Regals. Somit muss er das Zielobjekt nur noch in einem relativ kleinen Bereich suchen.



Abbildung 28: Prototyp - Zielfenster

6.9 Erweiterungen

Einige sinnvolle Erweiterungen sollten im Komplettsystem zusätzlich implementiert werden.

Im Bereich der LBS wäre die Anzeige der aktuellen Themengebiete (Bsp. "Englische Literatur"), in denen sich der Benutzer gerade befindet, denkbar, ebenso wie eine Beschriftung der Regale. Zudem könnte das System Vorschläge zu ähnlichen

Medien der aktuellen Suchliste anzeigen, sobald man sich in deren Nähe befindet. Diese Dienste müssten sich jedoch über die Optionen ausschalten lassen können, falls sie vom Benutzer als störend wahrgenommen werden.

In Bezug auf die mögliche Änderung der Zielreihenfolge von Seiten des Benutzers wäre hier ein Tool denkbar, das die neue Route auf dem Grundriss visualisiert und der Benutzer somit entscheiden kann, ob er dies wirklich für sinnvoll hält.

Eine weitere Möglichkeit zur Routenführung wäre ein durchgehender Richtungspfeil vom Startpunkt bis zum jeweiligen Ziel. Auf diese Weise müssten dem Benutzer die neuen Richtungsangaben nicht angekündigt werden, sondern es wäre von vorne herein ersichtlich für ihn, wohin die Route führt.

7 Vorgehensweise zur vollständigen Implementation

Da dieses Projekt aus vielen komplexen Gebieten besteht, so ist der Aufwand im Rahmen einer Bachelorarbeit für eine komplette Implementation und Installation innerhalb der Bibliothek zeitlich nicht machbar. Deswegen wurde in dieser Arbeit verstärkt die Recherche präsentiert und die neuartige Visualisierung eines Indoor-Navigationssystems fokussiert. Die Vorgehensweise zur Umsetzung des Projektes wird in Folgendem beschrieben.

7.1 Hardware

Zu Beginn ist es nötig, die entsprechende Hardware für das System zu besorgen. Dazu gehören ausreichend viele WLAN-Funkbasen, welche den kompletten Bereich der Bibliothek abdecken müssen, um den konstanten Datenverkehr und das Tracking zu ermöglichen. Weiterhin benötigt man PDAs für den Bibliotheksbenutzer, da nicht davon ausgegangen werden kann, dass die Mehrheit der Besucher im Besitz solcher Endgeräte ist. Zuletzt benötigt man noch einen einfachen Desktop-PC, welcher später als Server-Rechner fungieren soll und der ins Netzwerk integriert wird.

Anschliessend muss dafür gesorgt werden, dass jeder Punkt in den Teilen der Bibliothek, die ins System integriert werden sollen, von mindestens 3 Access Points (AP) mit ausreichender Signalstärke erreicht wird. Die Platzierung der APs und die Messung der Signalabdeckung könnte zum Beispiel mit Hilfe der beschriebenen *Ekahau Site Survey* erfolgen. Diese kann ebenfalls dazu verwendet

werden, um die Testmessungen für das WLAN-Tracking vorzunehmen, falls man dies nicht von Hand erledigen möchte.

7.2 Daten

Die ersten zu speichernden Daten erhält man durch die eben erwähnte WLAN Testmessung. Sie müssen in einer Datenbank für die *Positioning Engine* bereitgehalten werden.

Aus den daraus abgeleiteten Wegstrecken wird der gewichtete Graph konstruiert über den anschließend die Routenberechnung erfolgen kann.

Die Bibliothek Konstanz hat, wie alle Bibliotheken, eine Datenbank, in der alle Medien digital erfasst und bestimmten Regalen und Bereichen zugeordnet sind. Diese Informationen müssen nun erweitert werden, so dass jedem Regal und den in ihm stehenden Objekten ein Wegpunkt zugeteilt werden kann, um so in den gewichteten Graphen eingeordnet werden zu können.

7.3 Software

7.3.1 Client-Programme

Zwei Programme sind hierbei nötig, die auf dem Endgerät laufen müssen.

Tracking-Client

Die erste Software ist in das Tracking-System eingebunden und schickt dem Server in regelmäßigen, kurzen Abständen die aktuellen Signalstärken der umgebenden WLAN-Funkstationen, aus denen dieser die aktuelle Position des Benutzers berechnen kann. Hierbei ist zu empfehlen, dass für Client und Server die selbe

Programmiersprache gewählt wird, wie etwa Java oder C/C++ (Backend).

Visualisierungs-Client

Das zweite Programm ist das aus Sicht des Benutzers eigentliche Navigationssystem. Es ist das Interface, das dem Benutzer seine Position in der Bibliothek und die Navigationsangaben visualisiert. Dieses Interface kann, wie in Kapitel 4 beschrieben, mit OpenGL, Flash oder VRML realisiert werden. Eine beispielhafte Darstellung, wie solch ein Programm aussehen könnte, wurde im Prototypen zu diesem Projekt umgesetzt (Frontend).

7.3.2 Server-Programme

Positioning Engine

Dieses Programm bekommt vom Tracking-Client die Signalstärken geschickt und berechnet mit Hilfe der Tracking-Methoden und den Daten der Testmessungen die aktuelle Position des Endgerätes.

Routenberechnung

Mit geeigneten Algorithmen und den Informationen über die Regale berechnet dieses Modul die optimale Rundreise zu den Objekte in der Suchliste.

Visualisierungs-Server

Dieses Server-Programm steht in Kontakt mit dem Visualisierungs-Client auf dem Endgerät. Es schickt diesem entweder Anweisungen, wie die aktuelle Visualisierung auszusehen hat oder direkt Grafiken, die der Client ausgeben soll. Die nötigen Informationen dazu bekommt der Visualisierungs-Server vom Positionierungs-Server.

Die Firma Macromedia hat einen Server namens *SWIFT-Generator* entwickelt, welcher script-basiert ist und in Echtzeit SWF-Dateien für Flash generieren kann. Mit dieser Technik könnte man die im Prototypen eingesetzten Visualisierungsmethoden ebenfalls im Komplettsystem einsetzen. Zudem unterstützt dieser *SWIFT-Generator* Datenbankzugriffe, um zum Beispiel auf die Positionierungsdaten zuzugreifen. Ebenso ermöglicht dieser Server den Umgang mit personalisierten Inhalten zur individuellen Ziel- und Routenberechnung der Bibliotheksbenutzer.

7.3.3 Verwaltungsprogramm

Zuletzt benötigt man ein Programm, mit dem es möglich ist dieses System zu installieren und auf dem aktuellen Stand zu halten. Die heutigen Internet-Techniken erlauben es, solch ein Programm web-basiert aufzusetzen, so dass es den Bibliotheksangestellten von jedem Terminal aus zur Verfügung stehen kann. So wäre beispielsweise ein Java-Applet oder eine PHP-basierte Seite denkbar.

Die erste Funktion dieses Tools müsste die Möglichkeit bieten, die Infrastruktur für die Visualisierung festzulegen. Eine Überlegung hierzu ist, ein Applet zu schreiben, mit dem der Administrator vorgefertigte Elemente, wie Regale, Türen, Tische, Pflanzen, etc.. per Drag&Drop, mit Beachtung der genauen Koordinaten, in die zuvor gezeichnete Karte (Umgebung) einsetzen kann. Dabei sollte sichergestellt sein, dass hierbei die zur Navigation eingesetzten Landmarks der Bibliothek deutlich nachmodelliert werden können.

Auf Grundlage dieser digitalen Infrastruktur könnte dann das Script für den SWIFT-Generator erstellt werden.

Per Klick auf eines der Regale in der Karte können in diesem Ansatz dann die Informationen in die Datenbank eingetragen oder geändert werden. Ändert sich

etwas in der Infrastruktur der Bibliothek, kann mit dem gleichen Prinzip einfach per Drag&Drop das Element auf der digitalen Karte versetzt werden. Im Einrichtungsplan der Bibliothek können die Positionen der Elemente mit genauen Größenangaben nachgeschlagen werden.

7.4 Kosten

Die Kosten für dieses Projekt sind schwer zu schätzen. Je nach Größe des verwendeten Bibliotheksareals muss relativ dazu die entsprechende Menge an Access Points und PDAs eingekauft werden. WLAN-Funkstation sind momentan bereits ab ca. 40€ zu haben, PDAs ab ca. 300€.

Die Software kann größtenteils ohne Kosten selbst programmiert werden. Bei der Tracking-Software kommt es jedoch darauf an, ob die entsprechende Zeit und die Ressourcen zur Verfügung stehen, um diese selbst zu schreiben. Eine weitere Möglichkeit ist Open Source Projekte, wie MagicMap, zu erweitern, um eine befriedigende Positionsbestimmung zu erreichen. Als dritte Möglichkeit kann auch ein kommerzielles Programm, wie die *Ekahau Positioning Engine*, verwendet werden, das zwar mit Kosten verbunden ist, dafür aber eine ausreichende Genauigkeit garantiert. Der SWIFT-Generator zur Visualisierung ist kostenfrei, solange man bereit ist, das Firmenlogo der Firma Macromedia einzubinden. Andernfalls muss eine einmalige Gebühr in Höhe von 100€ gezahlt werden, um eine werbefreie Version zu erhalten.

8 Ausblick

Betrachtet man die gegenwärtigen Entwicklungen auf dem Gebiet der ortsbezogenen Dienste und des *Location-Trackings*, so kann und muss man davon ausgehen, dass sich diese Innovationen innerhalb der nächsten Jahre immer mehr durchsetzen werden. Aufgrund der Tatsache, dass diese Techniken im kommerziellen Sektor Anwendung finden, teilweise bereits gefunden haben, wird die Forschung und Entwicklung weiter beschleunigt.

Man wird im Alltagsleben früher oder später kaum mehr um solche Systeme herumkommen, weshalb sie zu einem immer wichtigeren Bestandteil auf dem Gebiet der Mensch-Computer-Interaktion werden. Und da Indoor-Navigationssysteme sowohl *location-based services* als auch das *Location-Tracking* beinhalten, lassen sich weitere Bemühungen auch auf diesem Gebiet legitimieren.

Zum aktuellen Zeitpunkt sind die Leistungen der benutzten Endgeräte bezüglich des benötigten Arbeitsspeichers für eine aufwendige Visualisierung noch grenzwertig, da der Informationsgehalt einer dynamischen 3D-Szene erheblich höher ist, als der von statischen oder dynamischen 2D-Anzeigen.

Dies wird sich aber in nächster Zeit sicher ändern, wenn man den raschen Fortschritt in der Computertechnik betrachtet. Zudem gibt es bereits Verfahren, um diese Datenstrukturen für leistungsschwache Geräte zu optimieren, indem man beispielsweise nur die aktuell sichtbaren Elemente im Vordergrund deutlich darstellt und den Hintergrund grafisch einfacher gestaltet.

Das Konzept der Indoor-Navigationssysteme, bei denen vom Benutzer ein oder mehrere Zielobjekte angesteuert und zudem noch hilfreiche Umgebungsinforma-

tionen angezeigt werden, kann in vielerlei Gebieten Anwendung finden.

So zum Beispiel als individueller Rundgang in einem Museum, bei dem bestimmte Kunstgegenstände aufgesucht und weitere Informationen zu den besuchten Sektionen geboten werden.

Genauso lässt es sich in allen öffentlichen Gebäuden und Ämtern, beispielsweise zur Navigation zu Bahnhofsgleisen oder Flughafengates und Terminals, nutzen. Ebenso ist die Suche von Läden und Sitzplätze in Fussballstadien oder Abteilungen in grossen Bürokomplexen und Krankenhäusern denkbar, sowie die Bereitstellung von Informationen zu diversen Örtlichkeiten oder als Hilfe beim Zurechtfinden in grossen Hotels und Einkaufszentren. Es lassen sich noch mehr Anwendungsgebiete finden und anführen, jedoch basieren alle auf den gleichen Techniken und Prinzipien, sowohl bei den technischen Grundlagen, als auch bei der Visualisierung. Nur die angebotenen Dienste und Informationen unterscheiden sich von Fall zu Fall .

Literaturliste

- [Aittola 2003] Markus Aittola, Tapio Ryhänen und Timo Ojala: *SmartLibrary - Location-Aware Mobile Library Service*, Mobile HCI 2003, LNCS 2795, pp. 411-416, 2003.
- [Aittola 2004] Markus Aittola, Pekka Parhi, Maria Vieruaho und Timo Ojala: *Comparison of Mobile and Fixed Use of SmartLibrary*, Mobile HCI 2004, LNCS 3160, pp. 383-387, 2004.
- [Belz&Hermann] Barbara Schmidt-Belz und Fabian Hermann: *User Validation of a Nomadic Exhibition Guide*, Mobile HCI 2004, LNCS 3160, pp. 86-97, 2004.
- [Butz 2001] A.Butz, J.Baus, A.Krüger und M.Lohse: *A Hybrid Indoor Navigation System*, In Proceedings of IUI 2001, Santa Fee, New Mexico, USA, 2001.
- [Ciavarella 2003] Carmine Ciavarella und Fabio Paterno: *Design Criteria for Location-Aware, Indoor, PDA Applications*, Mobile HCI 2003, LNCS 2795, pp. 131-144, Springer-Verlag Berlin Heidelberg, 2003.
- [Ciavarella 2004] Carmine Ciavarella und Fabio Paterno: *The design of a handheld, location-aware guide for indoor environments*, Pers Ubiquit Comput 2004, S.82-91, Springer Verlag London, 2004.
- [Dornbusch] Peter Dornbusch, TU München, *Realisierung von Positionierungen in WLAN*, ITG-Fachtagung "Technologie und Anwendungen für die mobile Informationsgesellschaft" 2002, Dresden, ISBN 3-8007-2723-4.
- [Gillieron 2004] P.Gillieron, D.Büchel, I.Spassov und B.Merminod: *Indoor Navigation Performance Analysis*, Swiss Federal Institute of Technology (EPFL), Geodetic Engineering Lab, Lausanne, Schweiz, 2004.
- [Heidmann 2003] Frank Heidmann und Fabian Hermann: *Benutzerzentrierte Visualisierung raumbezogener Informationen für ultraportable mobile Systeme*, Deutsche Gesellschaft für Kartographie, Beiträge des Seminars GEOVIS 2003, Hannover, S.121-131, 2003.
- [Jung 2004] Seokmin Jung and Woontack Woo: *ubiTrack: Infrared-based resident Tracking System for indoor environment*, ICAT, pp 181 - 184, 2004.

- [Komar&Ersoy] Can Komar und Cem Ersoy: *Location Tracking and Location Based Service Using IEEE 802.11 WLAN Infrastructure*, European Wireless 2004, Februar 2004.
- [Krueger 2002] Antonio Krüger, Jörg Baus und Wolfgang Wahlster: *A Resource-Adaptive Mobile Navigation System*, In Proceedings of IUI 2002, San Francisco, California, USA, 2002.
- [Krueger 2004] A.Krüger, A.Butz, C.Müller, C.Stahl, R.Wasinger, K.Steinberg und A.Dirschl: *The Connected User Interface: Realizing a Personal Situated Navigation Service*, In Proceedings of IUI 2004, Madeira, Portugal: ACM Press, 2004.
- [Kuepper 2005] Axel Küpper: *Location-based Services - Fundamentals and Operation*, John Wiley & Sons, August 2005,
- [Mabrouk 2004] M. Mabrouk, T. Bychowski, J. Williams, H. Niedzwiadek, Y. Bishr, J.Gaillet, N. Crisp, W.Wilbrink, M.Horhammer, G.Roy und S.Margoulis: *Open GIS Location Services (OpenLS): Core Services*, OGC 03-006r3, Version 1.0, Open GIS Consortium Inc., 2004.
- [Piekarski] Wayne Piekarski, Ben Avery, Bruce H. Thomas und Pierre Malbezin: *Integrated Head and Hand Tracking for Indoor and Outdoor Augmented Reality*, Wearable Computer Laboratory, University of South Australia.
- [Ross 2004] Tracy Ross, Andrew J. May und Simon Thompson: *The Use of Landmarks in Pedestrian Navigation Instructions and the Effects of Context*, Mobile HCI 2004, LNCS 3160, pp. 300-304, 2004.
- [Stahl 2001] Christoph Stahl: *V.A.I. Ein System zur intelligenten Navigation durch VRML Welten mittels graphischer Abstraktion*, Diplomarbeit von C.Stahl, Universität des Saarlandes, Saarbrücken, 2001.
- [Tomberge 2004] Tomberge, Patrick: *Navigation mittels RFID - Betrachtung der Navigationsmöglichkeiten durch RFID- Eintrittskarten bei der WM 2006*, Diplomarbeit von Patrick Tomberge, Universität Münster, 16. Dezember 2004.

Internetquellen

- [Bonsor] howstuffworks.com, Kevin Bonsor: *How Location Tracking Works*
URL: <http://science.howstuffworks.com/location-tracking.htm>
(Stand: 10.5.2006)
- [Ekahau] Firmenwebseite der Ekahau Inc., URL: <http://www.ekahau.com>
(Stand: 8.6.2006)
- [FileSaveAs] FileSaveAs, Informationswebseite, Überblick über *Handheld Computing* und die dazugehörigen Techniken. URL: <http://www.filesaveas.com> (Stand: 10.5.2006)
- [Golem 2004] Golem.de - IT-News, Artikel: Ekahau Site Survey 2.0 - WLAN-Funkabdeckung visualisieren, 2004. URL: <http://www.golem.de/0404/30964.html> (Stand: 5.3.2006)
- [GPSworld] GPS World, C.Kee, D.Jun, H.Jun, B.Parkinson, S.Pullen und T.Lagenstein: *Centimeter-Accuracy Indoor Navigation Using GPS-Like Pseudolites*, November 2001. URL: <http://www.gpsworld.com/gpsworld/article/articleDetail.jsp?id=3086>
(Stand: 17.5.2006)
- [HIPS] Fraunhofer Institut - Angewandte Informationstechnik, Projekt HIPS. URL: <http://www.fit.fraunhofer.de/projekte/hips/> (Stand: 7.6.2006)
- [IbachPPT] Peter Ibach, Institut für Informatik, Humboldt-Universität Berlin, PPT Präsentation: *MagicMap Kooperative Positionsbestimmung über WLAN*, 2004. URL: <http://www2.informatik.hu-berlin.de/~ibach/publications.html> (Stand: 16.6.2005)
- [IRREAL] Universität Saarbrücken, Artificial Intelligence Group, Projekt IRREAL. URL: http://w5.cs.uni-sb.de/website_old/irreal/ (Stand: 5.5.2006)
- [JCP 2006] Java Community Process, JSR-000179 Location API for J2ME. URL: <http://jcp.org/en/jsr/detail?id=179> (Stand 4.6.2006)
- [Karlsruhe] Universität Konstanz, Fakultät der Informatik, Informationen über die bestehenden Tracking-Varianten. URL: <http://www.tm.uka.de/itm/uploads/fohlen/11/mk08-4.pdf>
(Stand: 10.5.2006)
- [Khronos] Webseite der Khronos Group, Erklärungen zur OpenGL ES. URL: <http://www.khronos.org/opengles/> (Stand: 11.10.2005)

- [LORAN] U.S. Coast Guard Navigation Center, Informationen über LORAN-C. URL: <http://www.navcen.uscg.gov/loran/> (Stand: 12.6.2006)
- [MagicMap] Humboldt-Universität zu Berlin, Institut Für Informatik, Projekt MagicMap. URL: <http://www2.informatik.hu-berlin.de/rok/MagicMap/> (Stand: 8.6.2006)
- [Mahmoud] Webseite von SUN Microsystems, Qusay H. Mahmoud: *J2ME and Location-based Services*, März 2004. URL: <http://developers.sun.com/techttopics/mobility/apis/articles/location/> (Stand 2.6.2006)
- [MTB 2006] Medical Technology Business Europe, Ekahau ermöglicht "Panikknopf" zur Lokalisierung von Krankenhauspersonal, 2006. URL: <http://www.mtbeurope.info/deutsch/2006/d602002.htm> (Stand: 16.6.2006)
- [NSC 2006] Firmenwebseite der National Scientific Corporation. URL: <http://www.nsclocators.com> (Stand: 7.6.2006)
- [SAiMotion] Fraunhofer Institut - Angewandte Informationstechnik, Projekt SAiMotion. URL: <http://www.fit.fraunhofer.de/projekte/saimotion/> (Stand: 7.6.2006)
- [Seefunk] Seefunk und Seeschifffahrt, Private Homepage von Heinrich Busch, Informationen über *Decca*. URL: <http://www.seefunknetz.de/decca.htm> (Stand: 11.6.2006)
- [Scott] VRMLsite - 3D on the Internet, Adrian Scott: *The Marriage of Java and VRML*, September 1996. URL: <http://www.vrmlsite.com/sep96/spotlight/javavrml/javavrml.html> (Stand: 3.5.2006)
- [SUN 2006] Webseite von SUN Microsystems, Java Platform, Micro Edition (Java ME). URL: <http://java.sun.com/j2me/> (Stand: 7.6.2006)
- [Sony 2004] Golem.de - IT News, Sony stellt Linux-basierte 3D-Navigation vor. URL: <http://www.golem.de/0406/31703.html>
Screenshots: <http://scr3.golem.de/?d=0406/sony&a=31703>
(Stand: 5.6.2006)
- [Tadlys] Firmenwebseite der Tadlys Company - Wire Free Networking, Informationen über *Indoor Location Networks*. URL: http://www.tadlys.com/Pages/Product_content.asp?iGlobalId=2 (Stand: 3.6.2006)

- [TomTom] TomTom International BV, TomTom KFZ-Naviagtionssystem.
URL: <http://www.tomtom.com>
- [UCSD 2006] University of California, San Diego, Projekt ActiveCampus. URL:
<http://activecampus.ucsd.edu/> (Stand: 6.6.2006)
- [Wikipedia] Wikipedia, die freie Enzyklopädie, Online-Lexikon. URL:
<http://de.wikipedia.org> (Stand: 5.6.2006)

Abbildungsverzeichnis

1	Screenshot MagicMap [MagicMap]	14
2	Ekahau Komplettsystem [Ekahau]	17
3	Ekahau Site Survey - Messpunkte entlang der Wegstrecke [Ekahau]	20
4	Screenshot des ActiveCampus Browsers. (<i>links</i>) Campus-Übersicht, (<i>rechts</i>) Gebäudeansicht. [UCSD 2006]	24
5	(<i>a</i>) Position der 6 APs (Kreise) auf der Hauptebene, (<i>b</i>) Darstel- lung der SmartLibrary Architektur, (<i>c</i>) Das Suchsystem OULA- pda, (<i>d</i>) Ergebnisliste von OULA-pda, (<i>e</i>) Visualisierung der Ziel- führung von SmartLibrary auf einem PDA. [Aittola 2003]	26
6	Web-basiertes Verwaltungssystem von SmartLibrary (CPI). [Aittola 2004]	27
7	(<i>Mitte</i>) Desktop-Version, (<i>Oben rechts</i>) Handy-Version, (<i>Oben rechts</i>) PDA-Version. [Aittola 2004]	28
8	Abdeckung des ubiTrack Systemes. [Jung 2004]	30
9	GPS Pseudolites. [GPSworld]	32
10	Aufbau einer Ultraschall-Umgebung	33
11	Vor- und Nachteile der einzelnen Techniken	35
12	WLAN Testmessungen	36
13	Triangulationsverfahren [Dornbusch]	37
14	Angulationsverfahren	38
15	Einteilung in Wegabschnitte	42
16	Empfohlene Algorithmen	43
17	Anwendungsbeispiel	46
18	Detaillierungsgrad [Butz 2001]	49
19	Sony Navigationssystem - Beispiel 2 1/2 D [Sony 2004]	50
20	Sony Navigationssystem - Hybrid Modell [Sony 2004]	51
21	Prototyp - Kamera-Linsenlänge 30mm	52
22	Festorientierte "genordete" Karte	53
23	Prototyp - Steuerungsprogramm	62
24	Screenshot Swift 3d	64
25	Hochformat	65
26	Screenshot Prototyp	66
27	Prototyp - Werkzeugleiste	67
28	Prototyp - Zielfenster	71
29	TomTom Navigationssystem für den PDA. [TomTom]	88
30	Ekahau Manager - Grundrissmodellierung [Ekahau]	89
31	Ekahau Site Survey - Signalabdeckung vorhandener APs [Ekahau]	90
32	Screenshot des ActiveCampus Browsers. (<i>links</i>) Wichtige Plätze auf dem Campus, (<i>rechts</i>) Chat-Programm. [UCSD 2006]	90
33	Kontrollseite. [UCSD 2006]	91
34	Mittlerer Positionierungsfehler [IbachPPT]	91

35	Hybrid-Modell. Bei Klick auf ein Symbol öffnet sich eine kleine Animation des ausgewählten Areal	92
36	Kompletter Prototyp. (<i>oben</i>) Navigationssystem, (<i>unten</i>) Steuerungsprogramm.	93

Abkürzungen

AP	Access Point
API	Application Programming Interface
CPI	Content Provider Interface
EPE	Ekahau Positioning Engine
ESA	European Space Agency
GIS	Geographic Information Systems
GPS	Global Positioning System
GSM	Global System for Mobile Communications
HCI	Human Computer Interaction
IrDA	Infrared Data Association
ISM	Industrial Scientific Medical
IT	Information Technology
JSR	Java Specification Request
KOALA	Konstanzer Ausleih- und Anfrage-System
LBS	Location Based Services
LDS	Location Dependent Services
LORAN	Long Range Navigation
MIDP	Mobile Information Device Profile
NAVSTAR	Navigational Satellite Timing and Ranging
OGC	Open Geospatial Consortium
OPAC	Online Public Access Catalogues
PDA	Personal Digital Assistant
PHP	Hypertext Preprocessor
RFID	Radio Frequency Identification
SIG	Silicon Graphics
SWF	Small Web Format
UCSD	University of California, San Diego
UMTS	Universal Mobile Telecommunications System
URL	Uniform Resource Locator
US	United States
VRML	Virtual Reality Modelling Language
WLAN	Wireless Local Area Network
X3D	Extensible 3D
XML	Extensible Markup Language

Anhang

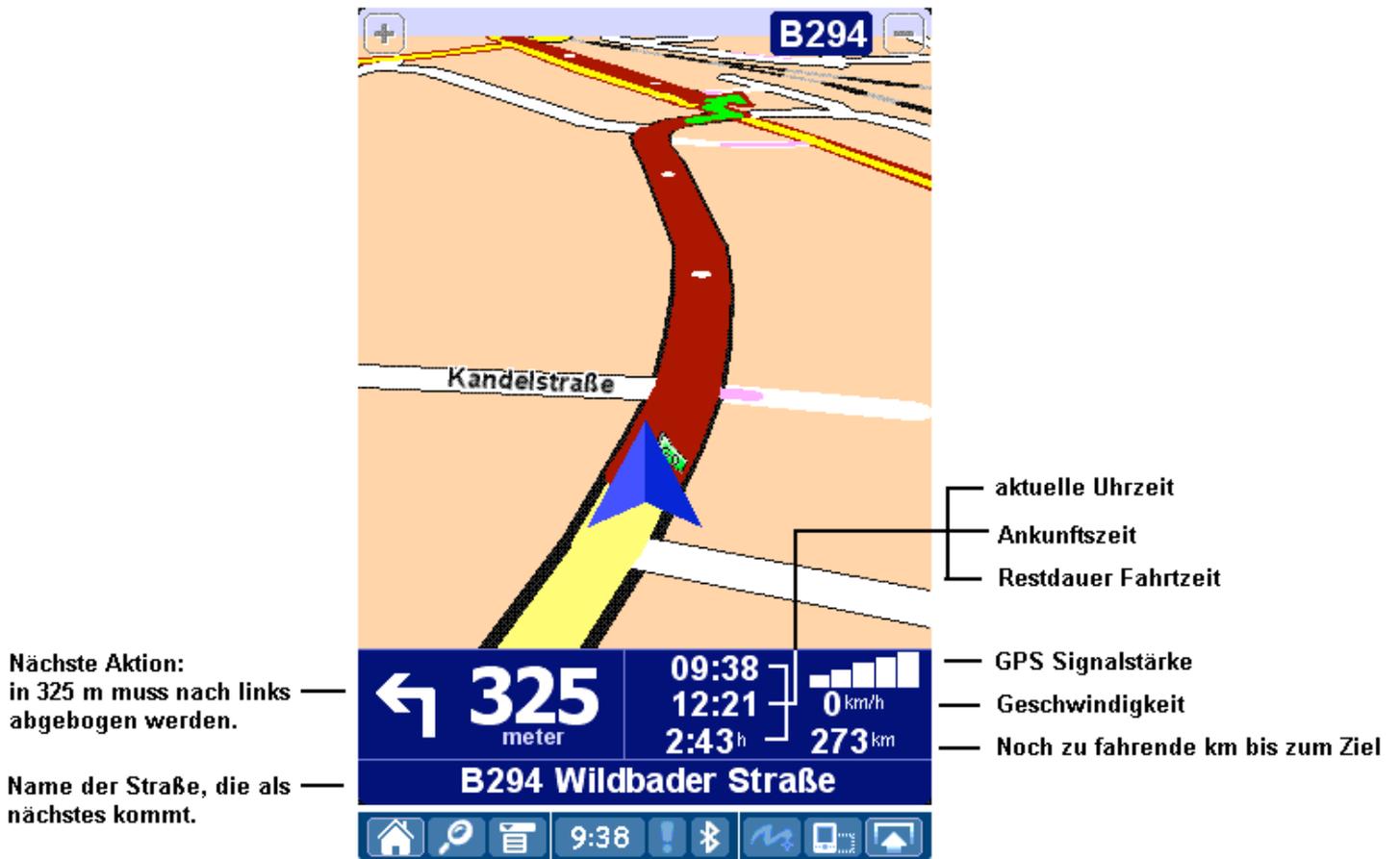


Abbildung 29: TomTom Navigationssystem für den PDA. [TomTom]

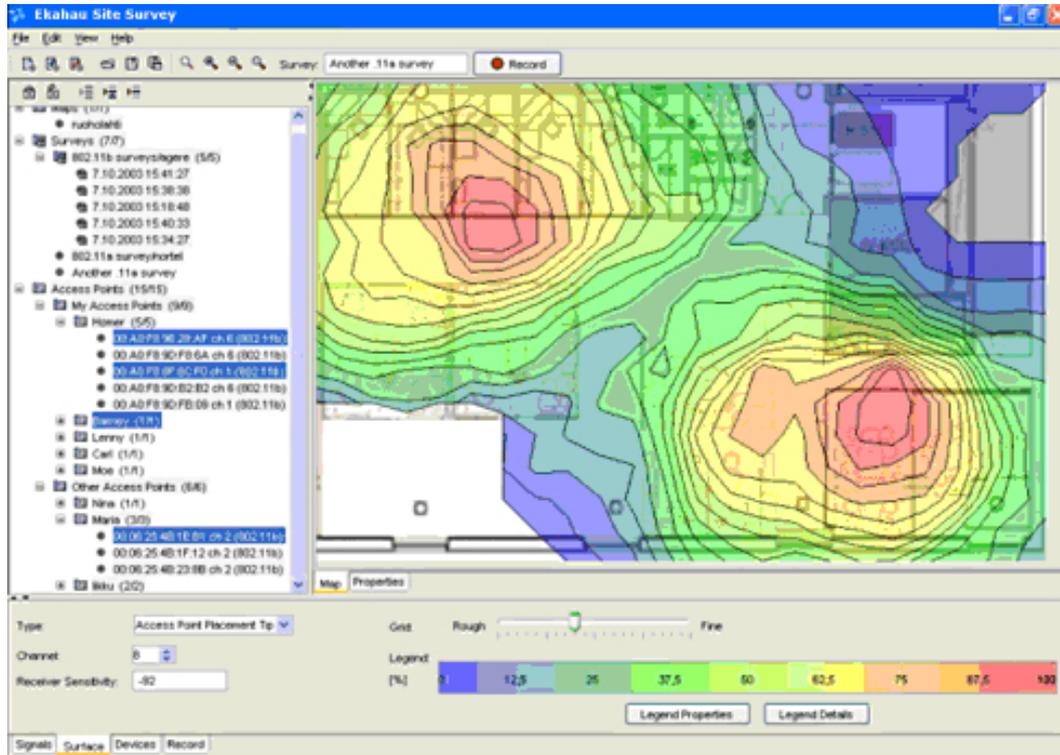


Abbildung 31: Ekahau Site Survey - Signalabdeckung vorhandener APs [Ekahau]

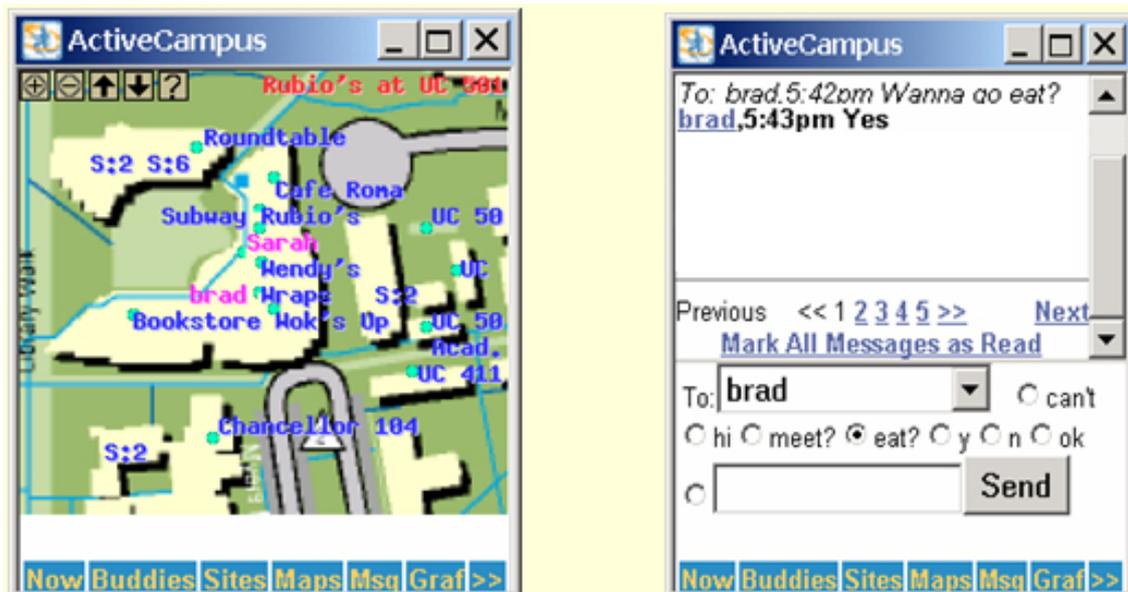


Abbildung 32: Screenshot des ActiveCampus Browsers. (*links*) Wichtige Plätze auf dem Campus, (*rechts*) Chat-Programm. [UCSD 2006]

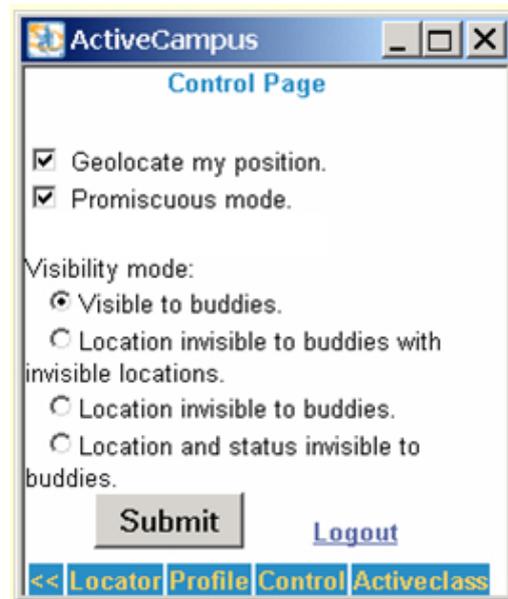


Abbildung 33: Kontrollseite. [UCSD 2006]

<u>Verfahren</u>	<u>Mittlerer Fehler</u>
RADAR (Profiling)	~3m
RADAR (Propagation)	~4.3m
HORUS (Probabilistic)	~2m
LEASE (Dynamic Profiling)	~2m
PlaceLab (Triangulation)	~10m
Ekahau (Profiling)	~1m
WhereNet, Aeroscout (TDOA)	~1m
MagicMap (Hybrid)	~10m (Triangulation) ~3m (Profiling)

Abbildung 34: Mittlerer Positionierungsfehler [IbachPPT]

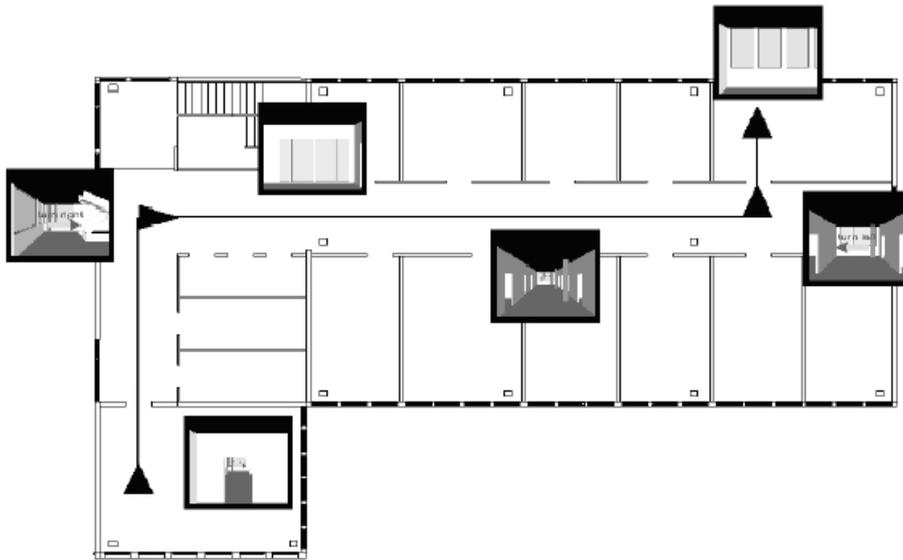


Abbildung 35: Hybrid-Modell. Bei Klick auf ein Symbol öffnet sich eine kleine Animation des ausgewählten Areals



Abbildung 36: Kompletter Prototyp. (oben) Navigationssystem, (unten) Steuerungsprogramm.

Stichwortverzeichnis

- 3D Studio Max, 45
- ActionScript, 45
- ActiveCampus, 23
- ActiveClass, 23, 90
- Aeroscout, 13
- Anforderungen, 58
- Angulation, 38
- Anwendungs-Szenario, 60
- Anzeigen, 65
- Aufgabenstellung, 11

- Backend, 29
- Beispielsuche, 60
- Benutzbarkeit, 56
- Benutzer, 54
- Blickfeld, 51
- Bluetooth, 31
 - Technische Daten, 31

- C / C++, 44

- Datenschutz, 24
- Datenübertragung, 29
- Decca, 8
- Designgrundlagen, 47
- Detaillierungsgrad, 48
- Direct Manipulating, 57

- Einleitung, 8
- Ekahau, 17, 89
- Ekahau Site Survey, 19
- Erfahrungen, 57
- Erweiterungen, 71
- Euklidische Distanz, 43

- Feldstärken, 20
- Flash, 45
- Frontend, 44
- Funktionen
 - Algorithmen, 41
 - Programmfunktionen, 58

- Galileo, 8
- Gewichteter Graph, 42
- GPS, 31
- GPS-Like Pseudolites, 32

- Hardware, 73
- Hauptfokus, 54
- HIPS, 21
- Historie, 8
- HORUS, 13
- Hybrid-Modell, 50, 92

- Infrarot, 29
 - Technische Daten, 30
- Interferenzen, 19
- IrDA, 29

- Java J2ME, 40
- JSR 184 Mobile 3D Graphics API, 45

- Kabellos, 29
- Kapitelübersicht, 12
- Kartenausrichtung, 52
- KFZ-Navigationssysteme, 13
- Komplettsysteme, 23
- Konzeptionelles Modell, 57
- Kriterien, 56

- Landmarks, 47
- Lateration, 37
- LEASE, 13
- Linux Wireless Tools, 14
- Location API, 40
- Location Dependent Services, 11
- Location Tracking, 10
- Location-Based Services
 - Beispiele, 21
 - Definition, 11
- Location-based Services
 - Beschreibung, 38
- LORAN, 8

- MagicMap, 14
- MedioVis, 11
- Messrauschen, 10
- MIDP, 41
- Mobiles MedioVis, 11

- Nachführung, 10
- Navigation
 - Definition, 9
- Navigationselemente, 69
- NetStumbler, 14
- Netzwerkqualität, 19

- Open Geospatial Consortium, 39
- OpenGL, 44
- OpenLS, 39
- Optionen, 67
- Ortungsverfahren, 29
- OULA-pda, 25

- Perspektive, 49
- PlaceLab, 13
- Point-to-curve Algorithmus, 43
- Positionsabweichung, 91
- Positionsbestimmung, 9
- Proaktive Dienste, 39
- Profile, 55
- Programmierung, 63
- Psiber Data, 20

- RADAR, 13
- Reaktive Dienste, 38
- RFID, 29
- Rogue Access Points, 19
- Routenberechnung, 41

- SAiMotion, 22
- Scenario, 60
- Server, 40
- Sichtnavigation, 10
- Sichtweise, 49
- Silicon Graphics, 44
- SmartLibrary, 25
- SmartWare, 25

- Steuerungsprogramm, 62
- Suchliste, 68
- SUN Microsystems, 40
- Swift 3D, 45, 63
- SWIFT-Generator, 76
- Systeme, 13

- TomTom, 13, 88
- Tools, 58
- Tracking
 - Beispiele, 13
 - Beschreibung, 36
 - Definition, 10
- Traveling-Salesman-Algorithmus, 43
- Triangulations-Methode, 37

- Ultraschall, 32
- Usability, 56

- Visualisierungstechniken, 44
- Vollständige Implementation, 73
- VRML, 45

- Web 3D Konsortium, 46
- Wegenetz, 42
- Werkzeuge, 58
- Werkzeugeleiste, 67
- Wherenet, 13
- WLAN, 33
 - Technische Daten, 34

- X3D, 46
- XML, 46

- Zielgruppen, 55
- Zusammenfassung, 3