Universität-Gesamthochschule Paderborn
Fachbereich Wirtschaftwissenschaften
Wirtschaftsinformatik 2
Prof. Dr. L. Nastansky

Diplomarbeit

Mobile Computing und Serveranbindung

-

Entwicklung einer generischen Serverkomponente für die Anbindung mobiler Kommunikationsfrontends an Groupware Backend Server

vorgelegt bei

Prof. Dr. L. Nastansky

von:

Bernd Altmiks

Studiengang Wirtschaftswissenschaften

Matrikelnummer: 385 92 40

Triftweg 36, 33106 Paderborn

Inhaltsverzeichnis

Seite:

Seite:

Danksagung

Mein Dank gilt allen, die zum Gelingen dieser Diplomarbeit beigetragen haben.

Besonders bedanken möchte ich mich bei Herrn Prof. Nastansky und bei Dipl.-Wirt.-Ing. Riempp für ihre Unterstützung und ihre Betreuung bei der vorliegenden Arbeit.
Ebenfalls bedanken möchte ich mich bei den Studenten Nico Dirks und Dirk Sievers für die gute Zusammenarbeit im Mobile Notes Projekt, ohne die diese Arbeit überhaupt nicht möglich gewesen wäre. Mein Dank gilt auch den Mitarbeitern der Firma Pavone Paderborn, die durch ihre materielle Unterstützung zum Gelingen dieser Arbeit beigetragen haben.

Eidesstattliche Erklärung

Ich erkläre hiermit an Eides Statt, daß ich die vorliegende Diplomarbeit selbständig und nur unter Verwendung der angegebenen Hilfsmittel erstellt habe.

Diese Arbeit hat in gleicher oder ähnlicher Form noch keiner Prüfungsbehörde vorgelegen.

_____

Paderborn, den 14. März 1996

Abkürzungsverzeichnis

Abb.	Abbildung
ACL	Access Control List
AG	Aktiengesellschaft
API	Application Programmers Interface
ARQ	Automatic Retransmission Request
ASCII	American Standard Code for Information Interchange
AvALoN	Advanced Access to Lotus Notes
CBC	Cipher Block Chainging
CCITT	Comite Consultatif International Telegraphique et Telephonique)
CGA	Colour Graphics Adapter
CRC	Cyclic Redundancy Check
DEE	Datenendeinrichtung
DES	Data Encryption Standard
DÜE	Datenübertragungseinrichtung
DV	Datenverarbeitung
ECB	Electronic Code Book
E-Mail	Electronic Mail
ETB	End of Transmission Block
EWR	Europäischer Wirtschaftsraum
FEC	Forward error correction
GmbH	Gesellschaft mit beschränkter Haftung
GS	Group Seperator
Hrsg.	Herausgeber
LCD	Liquid Cristal Display
PC	Personal Computer
PCMCIA	Personal Memory Card International Association
PDA	Persönlicher Digitaler Assistent
RS	Record Seperator
RSA	Rivest; Shamir; Adleman
SOH	Start of Header
Tab.	Tabelle
Vol.	Volume

Abbildungs- und Tabellenverzeichnis

Abbildung: Seite:

Tabelle: Seite:

1. Einleitung

1.1. Szenario

"In den achtziger Jahren haben Computer und ihre Anwendungen einen starken Wandel erfahren. Die Computertechnik entwickelte sich von der zentralen hin zur dezentralen Datenverarbeitung und wurde somit für weite Personenkreise verfügbar. Durch die Personal Computer und noch mehr durch die mobilen Systeme wurden die Computer personenbezogen und für das tägliche Leben unentbehrlich."[1] Und gerade im täglichen Leben spielt Kommunikation eine entscheidende Rolle.

Auch die Gesellschaft des 20. Jahrhunderts unterzieht sich einem stetigen Wandel. Die Liberalisierung internationaler Handelsbeziehungen, die politischen und ökonomischen Veränderungen in Osteuropa oder die Schaffung des europäischen Wirtschaftsraumes (EWR) vergrößern nicht nur den Aktionsradius der in dieser Gesellschaft tätigen Unternehmen, sondern verlangen auch von den einzelnen Individuen ein steigendes Maß an Mobilität und Kommunikation. Die Entwicklung immer leistungsfähigerer mobiler Systeme und vor allem die Fortschritte und Verbreitung mobil nutzbarer Kommunikationstechnologien eröffnen Anwendern neue Nutzungsmöglichkeiten im Arbeits- und Freizeitbereich.[2] Viele der in dieser Gesellschaft tätigen Unternehmen haben erkannt, daß die Möglichkeiten des Mobile Computing einen entscheidenden Wettbewerbsvorteil bedeuten können. "Fest steht: Von zwei Konkurrenten wird sich unter sonst gleichen Bedingungen derjenige durchsetzen, dessen Datennutzung effektiver ist, manchmal mit atemberaubender Schnelligkeit"[3]

"Mobile Computing wird in den kommenden Jahren somit die treibende Kraft für den gesamten Datenverarbeitungsmarkt (DV) Markt sein und zu einer Umstrukturierung der traditionellen Büroumgebung mit Kopierern, Faxgeräten und Druckern führen."[4]

Ein weiterer bedeutender Faktor auf dem Weg zur modernen Kommunikationsgesellschaft ist das Zusammenwachsen von Computer- und Telekommunikationstechnologien sowie die Miniaturisierung der einzelnen Komponenten. So sind heute, Dank der PCMCIA-Technologie, Modemsysteme von der Größe einer Kreditkarte herstellbar, die sich wiederum an viele mobile Computersysteme anschließen lassen. Auf diese Weise bilden Computer- und Kommunikationstechniken eine Einheit, die für das Mobile Computing von entscheidender Bedeutung ist. Neben der Entwicklung der einzelnen Komponenten wie den PCMCIA-Modems gehen die Computerhersteller immer mehr dazu über, Computer- und Kommunikationstechnologien in einem Gerät zu integrieren und lassen somit die Grenzen zwischen Computer- und Kommunikationsprodukten verschmelzen.

---

[1] Kleinwächter, Rolf: (Mobile Computing und Kommunikation), S. 1
[2] vgl. Niemeier, Joachim: (Mobile Computing), S. 12-16
[3] Niemeier, Joachim: (Mobile Computing), S. 16
[4] Kleinwächter, Rolf: (Mobile Computing und Kommunikation), S. 1

Hierbei lassen sich folgende Entwicklungstrends feststellen:

- Die Endgeräte werden immer kleiner und durch neue Eingabe- und Ausgabetechniken wie z.B. Handschriftenerkennung komfortabler in ihrer Bedienung.
- Die Hilfstechnologien wie z.B. Displays oder Speichermedien werden immer leistungsfähiger. Funktionalitäten, die bisher nur stationären Computersystemen vorbehalten waren, können dadurch in mobile Systeme integriert werden und ermöglichen dem Benutzer mobiler Endgeräte ein komfortables Arbeiten.

Auf Grund dieser Entwicklungen werden mobile Computersysteme in Zukunft stationäre Desktopsysteme in vielen Bereichen ersetzen und das Leben und Arbeiten der modernen Kommunikationsgesellschaft stark beeinflussen.

Abb. 1: Geschätzte Marktentwicklung für mobile Computer weltweitvgl. Niemeier, Joachim (Mobile Computing) 1994 S. 36

1.2 Aufgabenstellung

An der Lehr- und Forschungseinheit Wirtschaftsinformatik 2 der Universität-Gesamthochschule Paderborn wurde in Zusammenarbeit mit der Firma Pavone und der Philips AG im Rahmen eines Prototypen die AvALoN Anwendung (Advanced Access to Lotus Notes) entwickelt, die es ermöglicht, Lotus Notes Datenbanken im begrenzten Umfang auf dem "Screenphone P100" der Philips AG zu nutzen.[5]
Aufbauend auf dem oben erwähnten Prototypen erarbeitet die Mobile Notes Projektgruppe bestehend aus den Studenten Nico Dirks, Dirk Sievers und dem Autor dieser Arbeit das Mobile Notes System, das aus einer Serverkomponente und einer, auf dem "Screenphone P100" basierenden, Frontendapplikation besteht.[6] Im Unterschied zum bereits existierenden AvALoN Prototypen soll nun durch die Verwendung der in diesem Projekt entwickelten generischen Serverkomponente der Zugang zu Groupware Backend Servern für beliebige, programmierbare Frontendsysteme realisiert werden. Zusätzlich soll die Mobile Notes Serverkomponente die AvALoN-Funktionalitäten erweitern und neue Sicherheitsmechanismen, die dem Schutz der Datenübertragung sowie der übertragenden Daten dienen, bieten.
Basierend auf dem Mobile Notes Projekt werden an der Lehr- und Forschungseinheit Wirtschaftsinformatik 2 der Universität-Gesamthochschule Paderborn neben der vorliegenden Arbeit die folgenden Diplomarbeiten verfaßt:

- Architekturen und Anwendungskonzepte von Groupware in der mobilen Kommunikation - Generische Entwicklung von Benutzungsschnittstelle und Prozeß-Modulen für ein intelligentes Display-Telephon. (Autor: Nico Dirks)
- Architekturen und Anwendungskonzepte von Groupware in der mobilen Kommunikation - Generische Entwicklung von Kommunikationssteuerungsmodulen und Datenrepositories (Autor: Dirk Sievers)

Thema dieser Arbeit ist die Entwicklung der generischen Serverkomponente, die, im Vergleich zum AvALoN System, erweiterte Funktionalitäten und Sicherheitsmechanismen bietet und auf mehreren Systemplattformen (Microsoft Windows NT, IBM OS/2) lauffähig ist. An einigen Stellen wird jedoch auf die oben erwähnten Diplomarbeiten verwiesen, da für die Erstellung der Mobile Notes Serveranwendung auch Komponenten verwendet werden, die Thema der Arbeiten von Nico Dirks bzw. Dirk Sievers sind und an entsprechender Stelle näher behandelt werden.

---

[5] siehe Kapitel 2.5 Screenphone P100 sowie Kapitel 2.6 AvALoN-Prototyp
[6] siehe hierzu auch die Diplomarbeiten der Autoren Nico Dirks und Dirk Sievers

1.3 Aufbau der Arbeit

Im folgenden zweiten Kapitel werden im ersten Abschnitt wichtige Begriffe der Arbeit erläutert und abgegrenzt. Weiterhin wird ein kurzer Überblick über aktuelle mobile Frontendsysteme gegeben. Im besonderen wird hier auf das "Screenphone P100" der Philips AG eingegangen, das stellvertretend für andere Kommunikationsfrontends als Beispiel für die Umsetzung und Anwendung der im Mobile Notes Projekt entwickelten Serverkomponente dient. Im zweiten Abschnitt wird der an der Lehr- und Forschungseinheit Wirtschaftsiformatik 2 der Universität-Gesamthochschule Paderborn entwickelte AvALoN-Prototyp (Advanced Access to Lotus Notes) vorgestellt.
Das anschließende dritte Kapitel gibt einen kurzen Überblick über das Mobile Notes Projekt und stellt im folgenden die Überlegungen und Konzepte, die bei der Entwicklung der Mobile Notes Serverkomponente verwendet wurden, vor. Im Vordergrund stehen hierbei das generische Übertragungsprotokoll sowie die verschiedenen Sicherheitsmechanismen, die dem Schutz der Datenübertragung sowie der Daten selbst dienen.
Im vierten Kapitel wird die entwickelte Serverkomponente beschrieben. Zu Beginn werden die Funktionalitäten, Vorteile und Einsatzmöglichkeiten der in dieser Arbeit erstellten Mobile Notes Serverkomponente erläutert und die Umsetzung der im dritten Kapitel beschriebenen Konzeptionen und Überlegungen dargestellt. Von besonderer Bedeutung sind hierbei die Sicherheitsmechanismen und das generische Übertragungsprotokoll der Mobile Notes Serverkomponente die anhand von einem praktischen Anwendungsbeispiel vorgestellt werden. Weiterhin wird die Mobile Notes Initialisierungsdatenbank, die der Steuerung der Serverapplikation dient, anhand von Screenshots vorgestellt und erläutert.
Im abschließenden fünften Kapitel werden die wichtigsten Gedanken und Entwicklungen dieser Arbeit zusammengefaßt. Es werden zukünftige Weiterentwicklungen der erstellten Mobile Notes Serverkomponente diskutiert und ein Ausblick auf die zu erwartenden Entwicklungen auf dem Computer- und Kommunikationsmarkt gegeben.

2. Grundlagen und Definitionen

2.1 Mobile Computing

Der Begriff Mobile Computing wird als Oberbegriff für Anwendungen tragbarer Computer innerhalb eines größeren Datenverbundes verwendet.[7] In Bezug auf diese Arbeit erscheint es als sinnvoll, den Begriff Mobile Computing unter dem Gesichtspunkt Mobile Kommunikation näher zu definieren. Unter Kommunikation wird im allgemeinen der Austausch von Informationen zwischen selbständigen Systemen verstanden. Der Begriff "Mobile Kommunikation" faßt sowohl die mobile Datenkommunikation, die durch den Einsatz mobiler Rechner wie Laptops, Notebooks etc. gekennzeichnet ist, als auch die Benutzung von analogen und digitalen Mobilfunksystemen zusammen.[8] Die folgende Graphik gibt einen Überblick über die Komponenten des Mobile Computing:

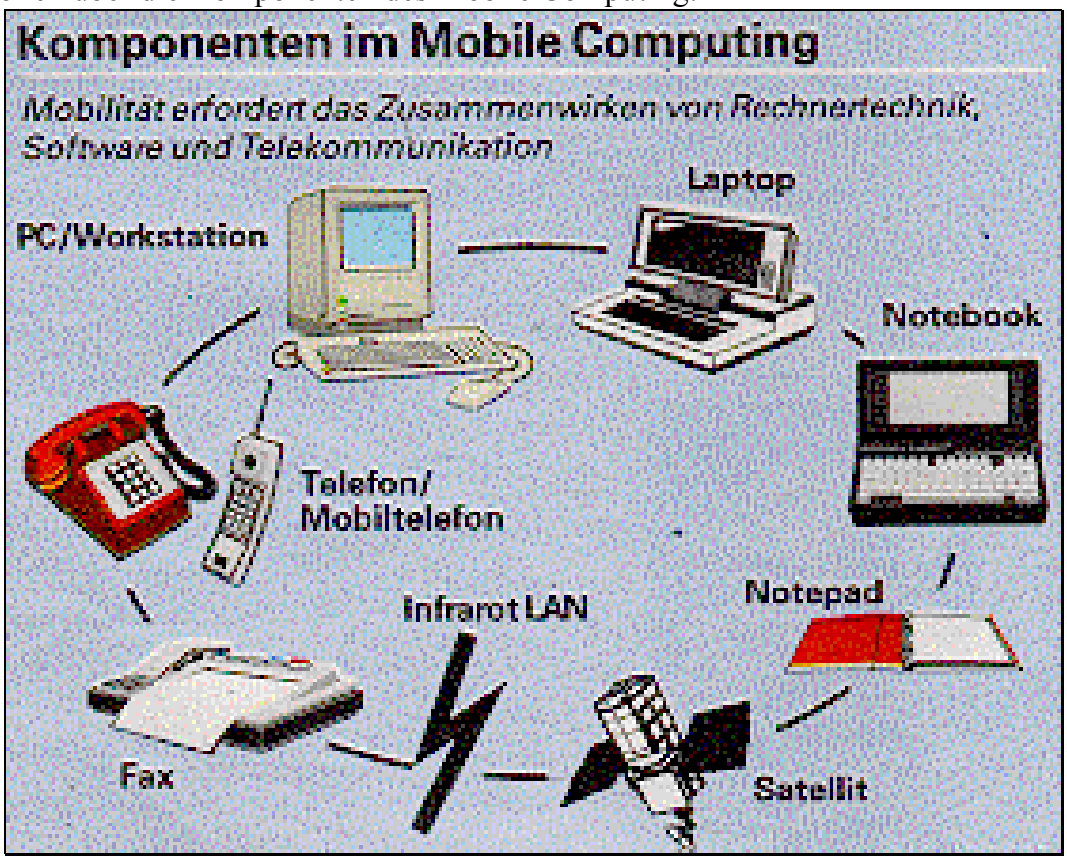

Abb. 2: Komponenten im Mobile Computing vgl. Computerzeitung spezial 14.09.95 S. 31

[7] vgl. Niemeier, Joachim: (Mobile Computing), S. 175
[8] vgl. Lexikon der PC-Fachbegriffe Band 2 Teil 6: "Mobilkommunikation"

2.1.1 Einsatzfelder

"Generell bieten sich Einsatzpotentiale für mobile Informations- und Kommunikationstechnik, wo heute Notiz- und Schreibblöcke, Terminbücher, Formulare, Lieferscheine, Produktinformationen, Preislisten etc. mobil eingesetzt werden, um Informationen aufzunehmen oder weiterzugeben, aktuell zu verarbeiten, ihre Qualität zu sichern und deren Weiterverarbeitung zu erleichtern. Damit sind insbesondere Personen und Berufsgruppen angesprochen, die beruflich viel reisen oder innerhalb des Unternehmens mobil sein müssen."[9]
Die folgende Tabelle zeigt einen Überblick über typische Einsatzfelder des Mobile Computing:

	Einsatzfeld	Berufsgruppe/Einsatzschwerpunkt
Tätigkeits-zentrierter Einsatz	Mobiler Manager-Arbeitsplatz	Fach- und Führungskräfte Freiberufler, Selbständige
Funktionsbereichs-zentrierter Einsatz	Verkaufsaußendienst	Handelsvertreter, Vertriebsbeauftragte Außendienstvertreter
	Produktkonfiguration & technischer Kundendienst	Techniker und Service-Ingenieure vom Wartungs- und Instandhaltungsunternehmen
	Lagerlogistik & Materialfluß	Lagerarbeiter, Disponenten Verkaufspersonal
	Instandhaltung & Qualitätssicherung	Instandhaltungstechniker Wartungspersonal, Prüftechniker Entwicklungsingenieure
	Schadensbegutachtung	Sachverständige, KFZ-Versicherer Polizei, Feuerwehr Not- und Pannendienste
	Forschung & Monitoring	Versuchstechniker, Umweltschutz
	Kundenmanagement	Marktforschung Ticket-Check-In/-Out Point of Information Kundenbezogene Dienste

(Fortsetzung auf der nächsten Seite)

[9] Niemeier, Joachim: (Mobile Computing), S. 20

	Einsatzfeld	Berufsgruppe/Einsatzschwerpunkt
Branchen-zentrierter Einsatz	Transportleistungen & Speditionslogistik	LKW-Fahrer Verkaufsfahrer, Lieferanten Taxifahrer, Kurierdienste
	Presse, Verlage & Sendeanstalten	Journalisten, Reporter Redakteure Auslandskorrespondenten
	Finanzdienstleistungen	Börsenmakler, Versicherungsmakler Bankkundenbetreuer
	Unternehmensberatung	Unternehmensberater Steuerberater
	Handwerk, Baugewerbe & Planungsbüros	Vermessungstechniker Architekten Energieversorgungsunternehmen Bauunternehmen Ingenieurbüros
	Krankenhaus & Pflegebereich	Mediziner, Ärzte Krankenschwestern

Tab. 1: Typische Einsatzfelder für Mobile Computing (Niemeier, Joachim 1994 S. 24)

2.1.2 Vorteile und Risiken

Aus den Möglichkeiten des Mobile Computing ergeben sich Vorteile, die im folgenden kurz dargestellt werden sollen:

- Zugriff auf aktuelle Informationen vor Ort z.B. bei einem Kunden, durch Einsatz mobiler Frontendsysteme
- Diverse Aufbereitungs- und Auswertungsmöglichkeiten der Daten unabhängig von einer stationären Computeranlage
- Sofortige Verfügbarkeit der vor Ort erfaßten Daten im Unternehmen durch einen Datenabgleich mit dem jeweiligen Server
- Vermeidung von Doppelerfassung der Daten und somit auch der damit verbundenen Fehler
- Bessere Erreichbarkeit der Anwender mobiler Systeme
- Flexiblere Arbeitszeiten, da Arbeit zum Teil auch von zu Hause oder Unterwegs erledigt werden kann
- höhere Effizienz der Mitarbeiter, die sich aus den zuvor erwähnten Vorteilen der Anwendung mobiler Computersysteme ergibt

Auch aus Sicht des Kunden einer mit Mobile Computing arbeitenden Unternehmung ergeben sich Vorteile, die nicht zu unterschätzende Wettbewerbsvorteile bedeuten können. Als Beispiele seien die schnellere Bearbeitung von Anfragen, Bestellungen etc. die wiederum kürzere Lieferzeiten nach sich ziehen, oder die durch aktuelle Informationen erhöhte Kompetenz der Ansprechpartner genannt.
Zusammenfassend kann festgestellt werden, daß Unternehmen, die die Möglichkeiten von Mobile Computing nutzen, auf lange Sicht einen Wettbewerbsvorteil erlangen, der unter dem starken Konkurrenzdruck der heutigen Wirtschaft von großer Bedeutung sein wird.[10]

Risiken des Mobile Computing

Neben den oben erwähnten Vorteilen des Mobile Computing ergeben sich jedoch auch Risiken durch die Benutzung mobiler Systeme. Da die mobilen Computer vorwiegend in wirtschaftlichen Bereichen der Gesellschaft zum Einsatz kommen, haben die in diesen Systemen bewegten und verwalteten Daten nicht nur für das jeweilige Wirtschaftsunternehmen, sondern auch für die Konkurrenz der Unternehmung einen hohen Wert. Daher kommt dem Schutz der mobilen Systeme und der in ihnen gespeicherten Daten besondere Bedeutung zu.
Hier sind erstens die mobilen Computer selbst zu erwähnen. Diese können durch Unachtsamkeit etc. des Benutzers vergessen, oder gestohlen werden, was neben dem materiellen Schaden auch unberechtigten Dritten die Möglichkeit verschaffen könnte, Zugang zu vertraulichen oder firmeninternen Daten zu erhalten. Daher ist ein Schutz dieser Systeme durch ein Paßwort oder andere Sicherheitsmechanismen ein wichtiger Punkt bei der Entwicklung mobiler Frontendsysteme.[11]
Einen weiteren Risikofaktor stellt die Kommunikation des mobilen Frontendsystems mit dem jeweiligen Backend Server dar. Der Datentransfer mobiler Systeme erfolgt in den meisten Fällen über eingebaute bzw. externe Modems die wiederum an das öffentliche Telefonnetz bzw. an die Netze der privaten Anbieter angeschlossen werden. Neben Störungen dieser Verbindungen, wie sie z.B. in Räumen, in der Nähe elektrischer Anlagen oder durch schlechte Telefonleitungen verursacht werden können, kann auch die Datenübertragung selbst Ziel von Abhörangriffen Dritter sein.
So ist neben der fehlerfreien Übertragung der Daten der Schutz des Datentransfers zwischen dem mobilen Frontendsystem und dem Backend Server vor Zugriffen bzw. Manipulationen Dritter von besonderer Bedeutung. Hierzu bietet sich die Verschlüsselung der zu übertragenden Daten an, auf die in dieser Arbeit noch ausführlich eingegangen werden wird.[12]

---

[10] vgl. Eitel, Barbara; Torabli, Kian: (Mobile Computing) , S. 31
[11] siehe hierzu die in Kapitel 1 erwähnten Diplomarbeiten der Autoren Dirk Sievers und Nico Dirks
[12] siehe hierzu Kapitel 3.3 Sicherheitsaspekte

2.1.3 Entwicklung des Mobile Computing

Wie schon in der Einleitung erwähnt, wird Mobile Computing in der Zukunft starken Einfluß auf die in dieser Gesellschaft tätigen Unternehmen und Personen nehmen. Zuversichtliche Marktprognosen wie die folgende Graphik belegen, daß die mobilen Computersysteme die stationären Computer immer mehr ersetzen und verdrängen werden. Die zunehmende Verbreitung der mobilen Kommunikation und der Ausbau der Telekommunikationsnetze steigern zusätzlich die Einsatzmöglichkeiten für das Mobile Computing und sind ein entscheidender Faktor für die Entwicklung und Bedeutung mobiler Systeme.

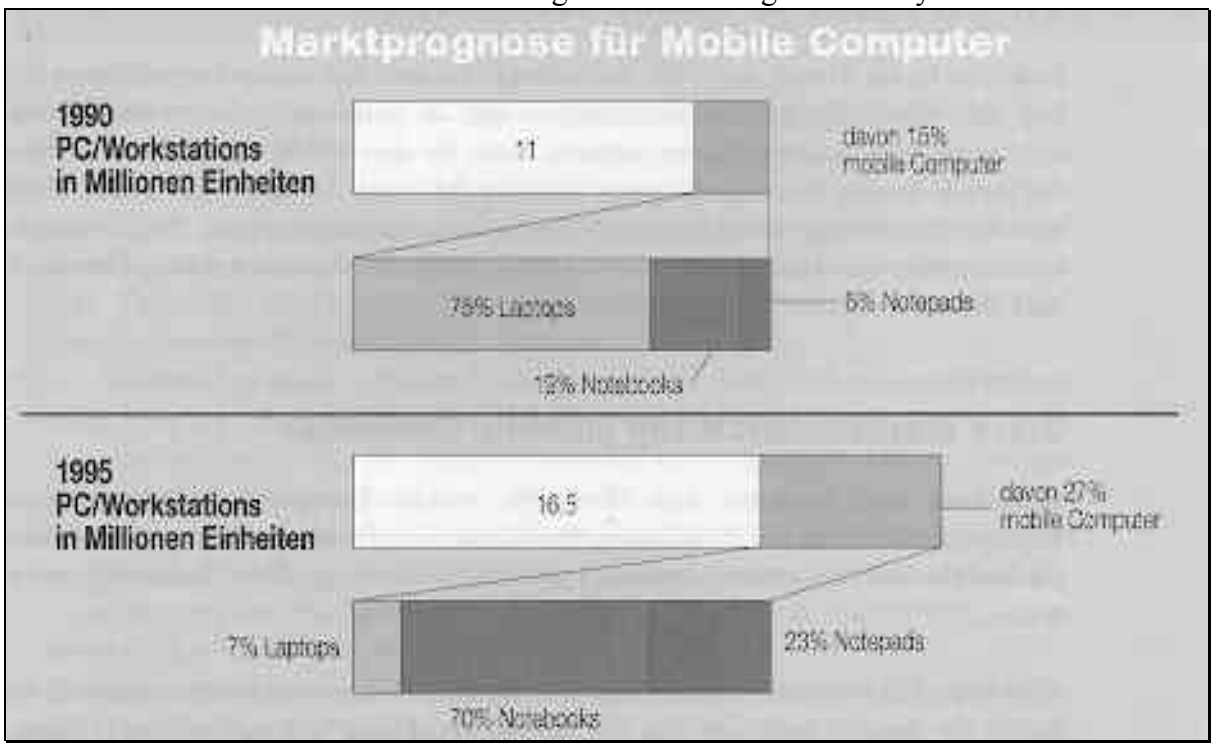

Abb. 3: Marktprognose für Mobile Computer vgl. Niemeier, Joachim (Mobile Computing) 1994, S. 36

2.2 Groupware

In der wissenschaftlichen Welt wird der Begriff Groupware aus verschiedenen Standpunkten gesehen und definiert. Eine eindeutig gültige Definition ist daher nicht möglich. Stellvertretend für die Vielzahl von Definitionen von Groupware sei an dieser Stelle folgende genannt: "Groupware stellt computergestützte Konzepte für die Teamarbeit bereit. Damit erscheint Groupware so griffig wie 1982 die PC-gestützte Tabellenkalkulation - nämlich kaum."[13] "Semantisch ergibt sich eine einfache pragmatische Definition von Groupware als Software zur Unterstützung der Gruppenarbeit."[14] Hierzu gehören beispielsweise Programme, die Daten nach vorgegebenen Kriterien aus verschiedenen Quellen wie z.B. Datenbanken oder von verschiedenen Bearbeitern zusammenfassen und den weiteren Arbeitsablauf bzw. den nächsten Bearbeiter anhand von Entscheidungskriterien automatisch bestimmen. Werden die nötigen Voraussetzungen wie z.B. die Neugestaltung veralteter Strukturen im Büro- und Verwaltungsbereich geschaffen, können mit dem Einsatz von Groupwareprodukten erhebliche Produktivitätssteigerungen im Büro- und Verwaltungsbereich von Unternehmen erzielt werden. Da Groupware nicht das Kernthema dieser Arbeit darstellt, sei an dieser Stelle auf die entsprechende Literatur verwiesen.[15]

2.3 Lotus Notes

Die Plattform Lotus Notes, ist ein System zur Verwaltung verteilter Datenbanken in Client-Server-Umgebungen. In ihrer Grundfunktionalität ist sie eine leistungsfähige Datenbank- und Kommunikationsumgebung zur Unterstützung von Arbeitsgruppen, weshalb sie in die Kategorie Groupware eingeordnet werden kann.[16] Lotus Notes ist für zahlreiche Betriebssysteme lieferbar (IBM OS/2, Microsoft Windows, Microsoft Windows NT, Apple Macintosh etc.). und bietet so eine einmalige Integrationsplattform für unternehmensweite Kommunikationssysteme und Groupwareanwendungen. "Lotus Notes ist derzeit als die einzige umfassende Entwicklungs- und Anwendungsplattform für verteiltes betriebliches Informationsmanagement anzusehen, in welcher das Replikationsprinzip für die verteilt auf Client-Arbeitsplätzen und Servern gehaltenen Dokumentendatenbanken die entscheidende Architekturkomponente darstellt."[17]

---

[13] Nastansky,Ludwig: (Gruppenarbeit - Workgroup Computing); S.: 6ff.
[14] Behrens, Olav Max: (Hypermediakonzepte in Groupwareapplikationen), S. 31
[15] siehe hierzu Herget, Josef: (Wirtschaftlichkeit der Bürokommunikation); Nastansky, Ludwig: (Workflow Management - Endlich Paradigmenwechsel im Büro?)
[16] vgl. Riempp, Gerold: (Modellentwurf für Workflow Management), S. 15
[17] Nastansky, Ludwig. ("Büroinformationssysteme"), S. 273-373

2.4 Frontendsysteme

Die Entwicklung des Mobile Computing ist verbunden mit dem Fortschritt in der Computer- und Kommunikationsindustrie. Vor allem neue Produkte, die die Kommunikation zu einem wichtigen Bestandteil mobiler Computer machten, führten dazu, daß Mobile Computing zu einem Schlagwort der modernen Kommunikationsgesellschaft geworden ist. Die folgende Tabelle gibt einen Überblick über die aktuellen, am Markt befindlichen mobilen Computersysteme:

Klasse	Gewicht/Größe	Merkmale
Netzabhängiger Portable	6 - 11 kg	Gebrauch von Standard / Add-On-Karten
Batteriebetriebener Laptop	4 - 9 kg	Spezielle Add-On-Karten, begrenzte Portabilität
Notebook	< 3,5 kg / 30 x 25 x 5 cm	Tastatur, optimale Portabilität, erfordert Unterlage
Sub-Notebook	1 - 2 kg / 21 x 16 x 3 cm	kleine Tastatur, optimale Portabilität, geringere Leistung als Notebooks
Palmtop	0,3 - 0,6 kg / 20 x 9 x 2,5 cm	kleine Tastatur, kleine LCD-Anzeige, geringe Leistung
Notepad Tablet, Pentop	< 2,5 kg / 30 x 25 x 2,5 cm	Handschrifteneingabe, Zeige-funktionalitäten ("Point, Tap and Drag"), kann in der Hand gehalten werden
Palmpad	0,5 kg	Datensammlung
Personal Digital Assistant (PDA)	0,5 kg	Handtellerformat, fokussierte Funktionen, optimale Mobilität, drahtlose Datenübertragung

Tab. 2: Klassifizierung der mobilen Systeme vgl. Niemeier, Joachim (Mobile Computing) 1994, S. 90

2.5 Screenphone P100

Bei dem "Screenphone P 100" der Philips AG handelt es sich um ein Telefon mit einem integrierten Computer. Es besitzt einen 8088-kompatiblen NEC-Processor sowie ein 40 x 16 Zeichen CGA-LCD Display. Für die manuelle Dateneingabe befindet sich unterhalb der Telefontasten eine ausziehbare US-Tastatur. Weiterhin verfügt das Screenphone über ein eingebautes Modem, das eine Datenübertragung mit 2400 Baud ermöglicht. "The basic concept of the Philips P100 modem-handler is to emulate a Standard Hayes modem. The vaste majority of modem-manufactures use this standard, and so it can be called the de facto industry-standard for modems."[18] Auf der Rückseite des Screenphones befinden sich ein PCMCIA Typ 1 Slot, ein Smartcardslot sowie eine Anschlußmöglichkeit für eine externe Tastatur. Der Preis des P 100 liegt bei ca. 1000 US $ und wurde ursprünglich für die amerikanische Citybank entwickelt (Homebanking). Ende 1995 wurden ca 50.000 Exemplare in den USA verkauft.

Screenephone P100

Abb. 4: Screenphone P100

[18] Philips (P100 System Software: Modem interface Specification Version 2.1) S. 5

2.6 AvALoN-Prototyp (Advanced Access to Lotus Notes)

Der im Rahmen einer Seminararbeit von den Studenten Nico Dirks, Dirk Sievers und dem Autor dieser Arbeit in Zusammenarbeit mit der Firma Pavone und der Philips AG an der Lehr- und Forschungseinheit Wirtschaftsinformatik 2 der Universität Paderborn entwickelte AvALoN-Prototyp kann als Grundlage dieser Arbeit angesehen werden. Im folgenden werden Aufbau und Funktionalitäten des Systems beschrieben.

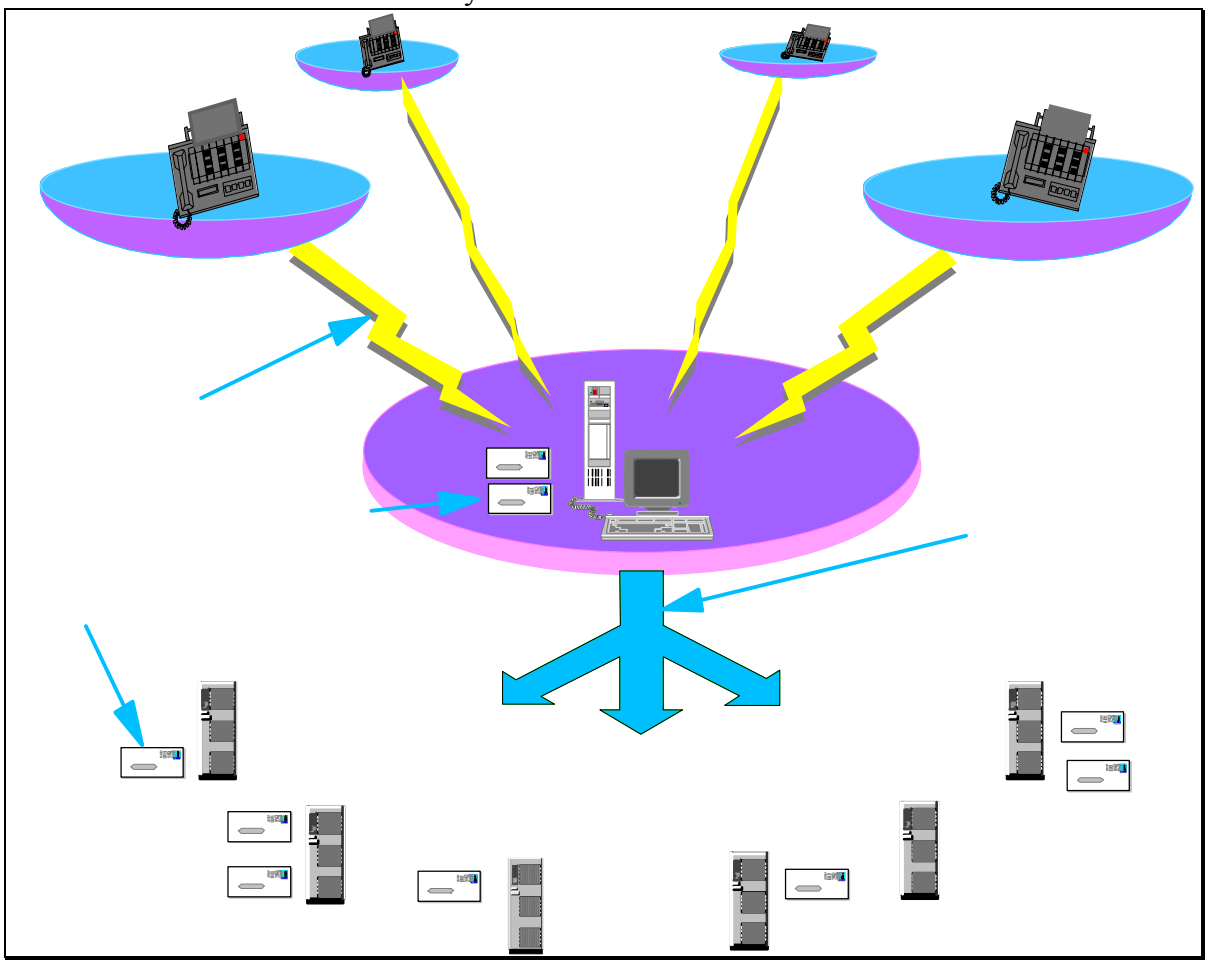

Abb. 5: Skizze des AvALoN-Projektes

2.6.1 Aufgabenstellung

Ziel des Seminararbeitsprojektes AvALoN war die Anbindung des "Screenphones P100" der Philips AG an die Lotus Notes Datenbankumgebung. Hierzu sollten im Rahmen eines Prototypen und unter Wahrung von Style-Guide Vorgaben der Philips AG folgende Funktionalitäten für das "Screenphone P100" implementiert werden:

- "Nur lesen"-Zugriff auf die in der AvALoN Initialisierungsdatenbank definierten Lotus Notes Datenbanken
- Nutzung der Memorycard zur Dokumentenablage, um das Lesen von Dokumenten auch ohne Online-Verbindung zu ermöglichen
- Möglichkeit der Erstellung von Dokumenten speziell für die Maildatenbank eines Users
- Verwendung von Sicherheitsmechanismen (Paßwortabfrage), um den Zugang zum System vor nicht berechtigten Personen zu schützen

2.6.2 Umsetzung

Die Umsetzung der Aufgabenstellung wurde durch eine dreiteilige Applikationsstruktur erreicht: Auf der Serverseite verarbeitet eine Visual Basic Applikation die ankommenden Anrufe des Screenphones. Aus Sicherheitsgründen, und um einen ID-Wechsel zu vermeiden, ist ein eigener AvALoN-Server für die Serverkomponente erforderlich, auf dem alle dem System zugänglichen Datenbanken lokal vorhanden sind. Weiterhin wird eine speziell entwickelte Notes Datenbank benötigt (im folgenden AvALoN-Initialisierungsdatenbank genannt), die sich ebenfalls auf dem Avalon-Server befinden muß, und die Benutzerdaten sowie Datenbankdefinitionen enthält.
Die für das Screenphone entwickelte Anwendung besteht aus einem Softwaremodul, das Benutzerinteraktionen gemäß den Style-Guides zuläßt und verarbeitet. Der Zugang zu Lotus Notes wird durch die an der Lehr- und Forschungseinheit Wirtschaftsinformatik 2 der Universität Paderborn entwickelte Macroware.dll, die wiederum auf dem Lotus API (Application Programming Interface) basiert, realisiert.

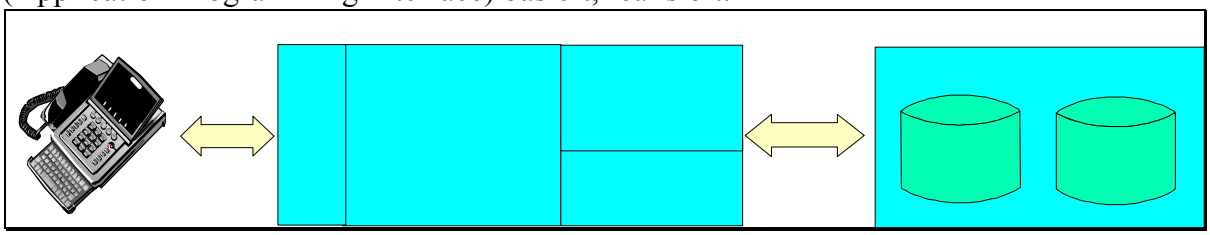

Abb. 6: Komponenten des AvALoN-Projektes

2.6.3 AvALoN-Initialisierungsdatenbank

Die AvALoN-Initialisierungsdatenbank ist eine speziell für die Servervapplikation des AvALoN-Projektes entwickelte Datenbank, die folgende Dokumente enthält:

- Datenbankdokumente: Für jede Datenbank, die dem System zugänglich sein soll, muß ein Initialisierungsdokument erstellt werden, das den Pfad, Namen und den zu verwendenden View der Datenbank enthält. Außerdem werden in diesem Dokument bis maximal 10 Felder des Dokumententyps definiert, der von dem Client gelesen werden soll.
- Personendokumente: Für jeden User des Systems muß ein Personendokument angelegt sein, das den Namen, das Paßwort und die Maildatenbank des Users enthält. Zusätzlich wird in diesem Dokument die Maildatenbank des Users initialisiert. Hierbei kann zusätzlich zu der oben bereits beschriebenen Dokumentendefinition noch eine Datenbankform angegeben werden, die für die Erstellung von Dokumenten in der Maildatenbank des Users verwendet wird.

Die Daten der AvALoN-Initialisierungsdatenbank sind für den "normalen" Benutzer des Systems gesperrt. Das Einrichten von Datenbanken sowie die Anmeldung neuer Nutzer erfolgt durch den Systemadministrator.

2.6.4 Sicherheitsmechanismen

Um einen unkontrollierten Zugriff auf Notes Datenbanken zu vermeiden, werden im Benutzerprofil eines AvALoN-Users, dessen Name sowie sein spezielles AvALoN-Paßwort gespeichert. Zu Beginn der Kommunikation zwischen dem Screenphone und der Serveranwendung ist die Übertragung des Usernamens sowie dessen Paßwort notwendig. Die AvALoN-Serverapplikation vergleicht die vom Screenphone gesendeten Daten mit denen in der AvALoN Intialisierungsdatenbank. Wird eine Übereinstimmung gefunden, so ist der Anrufer als Systemnutzer identifiziert und hat Zugriff auf die dem System zugänglichen Datenbanken. Sollte jedoch das Paßwort dreimal falsch eingegeben worden sein, wird die Verbindung durch den AvALoN-Server abgebrochen. Im Rahmen eines Prototypen war dieser Sicherheitsmechanismus ausreichend, aber dieses Vorgehen beinhaltet diverse Schwachpunkte, die im folgenden kurz aufgezeigt werden sollen:

- Die Datenübertragung zwischen Server und Client erfolgt unverschlüsselt, wodurch wichtige Daten wie z.B. Name und Paßwort abgehört werden könnten.
- Mehrmalige Fehlversuche beim Anmelden des Clients werden nicht mitprotokolliert und es erfolgt kein Warnhinweis an den User bzw. an den Systemadministrator, daß evtl. ein nicht berechtigter Dritter sich Zugang zum System verschaffen wollte.

- Die Datenübertragung wird nicht auf mögliche Fehlübertragungen abgeprüft, so daß Fehler, wie sie z.B. durch eine schlechte Telefonverbindung verursacht werden können, nicht erkannt werden und die übermittelten Daten verfälschen könnten.

Da das AvALoN-Projekt jedoch nur der Demonstration des Zugangs zu Lotus Notes Datenbanken im Rahmen eines Prototypen diente, war die Verwendung spezieller Sicherheitsmechanismen wie z.B. verschlüsselte Datenübertragung etc. nicht notwendig. Vielmehr war es Gegenstand der Seminararbeit mögliche Anwendungen von "Low Level" Frontendsystemen in Verbindung mit Backendservern aufzuzeigen.

3. Mobile Notes - Konzepte und Aufbau

Das folgende Kapitel gibt einen Überblick über den Aufbau des Mobile Notes Gesamtprojektes sowie speziell über die in dieser Arbeit zu behandelnde Mobile Notes Serverkomponente. Weiterhin werden die in dieser Arbeit verwendeten Konzepte und Überlegungen vorgestellt.

3.1 Überblick über das Mobile Notes Projekt

Das Mobile Notes Projekt ist in zwei Hauptapplikationen aufgeteilt. Die Frontendapplikation, die auf dem Screenphone der Philips AG basiert und von den Studenten Nico Dirks und Dirk Sievers erarbeitet wird, stellt hierbei die praktische Nutzung der in dieser Arbeit entwickelten Mobile Notes Serverapplikation dar.[19] Die Kommunikation der Serveranwendung mit den Frontendsystemen erfolgt über ein handelsübliches Modem, das den Hayes-Befehlssatz beherrscht und an das öffentliche Telefonnetz angeschlossen werden kann. Der Heyes-Befehlssatz bezeichnet einen Befehlsstandard für die Steuerung von Modems und wird von der Mobile Notes Serverkomponente verwendet, um das angeschlossene Modem zu initialisieren und zu steuern. Die Entwicklung der Frontendapplikation auf Basis des "Screenphones P100" der Philips AG ist Bestandteil der oben erwähnten Diplomarbeiten der Studenten Dirk Sievers und Nico Dirks und wird in der vorliegenden Arbeit nicht weiter behandelt werden.

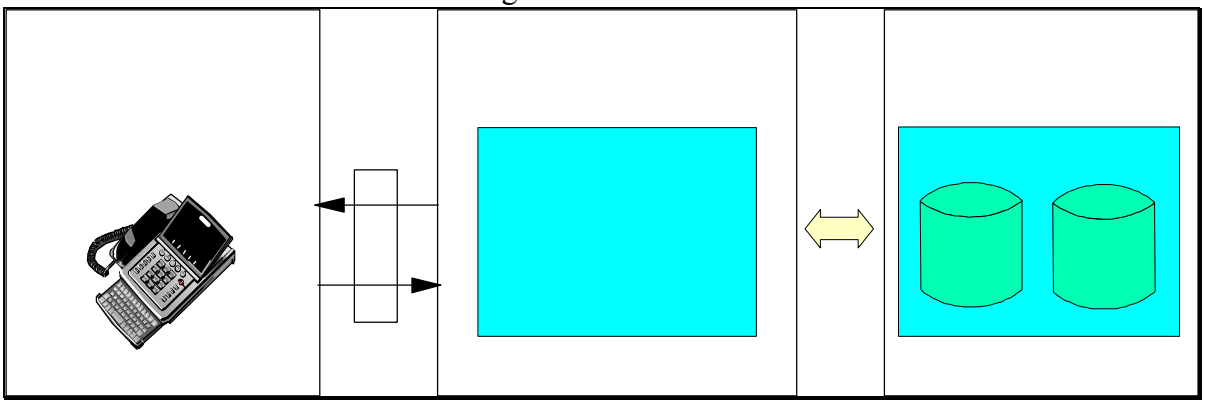

Abb. 7: Überblick über das Mobile Notes Projekt

[19] siehe auch die Diplomarbeiten der Autoren Dirk Sievers und Nico Dirks

3.2 Die generische Serverkomponente

Die Entwicklung einer generischen Serverkomponente, die es diversen Kommunikations-frontends ermöglicht auf Groupware Backend Server zuzugreifen, steht im Mittelpunkt dieser Diplomarbeit. Im folgenden Teil dieses Kapitels werden die Überlegungen und Sicherheitskonzepte, die zu der Entwicklung der generischen Serverkomponente führten, dargelegt und erläutert.

Kommunikationskomponenten

Da die Kommunikation des Mobile Notes Servers mit den Frontendsystemen über das öffentliche Telefonnetz erfolgt, muß sowohl auf der Server- als auch auf der Clientseite für die Datenübertragung ein Modem verwendet werden. Um möglichst vielen Frontendsystemen den Zugang zum Mobile Notes Server zu ermöglichen, sollte auf der Serverseite ein Modem verwendet werden, das die Standards der CCITT (Comite Consultatif International Telegraphique et Telephonique) beherrscht, die von nahezu sämtlichen Modemherstellern unterstützt werden und die verschiedenen Übertragungsmöglichkeiten normiert.[20]
Neben dem verwendetem Modem ist das Datenübertragungsprotokoll, welches sämtliche Kommunikationsbefehle zwischen dem Client und dem Mobile Notes Server umfaßt, der entscheidende Bestandteil der Kommunikation zwischen Mobile Notes Server und Client. Die Aufgaben eines Protokolls in der Datenkommunikation bestehen neben der eigentlichen Übertragung der Daten, aus der Sicherung der Übertragung gegenüber Fehlern, aus einer eventuellen Fehlerkorrektur und aus der Steuerung der Datenübertragung zwischen den verschiedenen Kommunikationspartnern.
Das in dieser Arbeit entwickelte Datenübertragungsprotokoll kann als Grundlage für verschiedene Frontendapplikationen, die auf Lotus Notes Backend Server zugreifen möchten, dienen. Hierzu sollten folgende Anforderungen erfüllen werden:

- Modularer Aufbau der Serverkomponente, um eine problemlose Anpassung des Übertragungsprotokolls an zukünftige Weiterentwicklungen (z.B. Notes 4.0) durch Austausch einzelner Module zu ermöglichen
- Leichte Ausbaufähigkeit um zusätzliche Funktionen (z.B. Sprachübermittlung etc.), da durch die kontinuierliche Weiterentwicklung Mobile Computersysteme zukünftige Anforderungen an das Mobile Notes Datenübertragungsprotokoll zunehmen werden
- Optionale Kontrollmechanismen für die Überprüfung der Datenübertragung, um auch "Low Level" Frontendsystemen, die eventuell diese Mechanismen nicht verwenden können, den Zugang zum Mobile Notes System zu ermöglichen

[20] Die CCITT ist ein internationales Gremium das allgemeinverbindliche Normen für die Datenübermittlung festlegt.

- Verwendung des ASCII-Code (American Standard Code for Information) für den Austausch von Daten
- Standardaufbau der Protokollbefehle, um eine unkomplizierte Anwendung der Befehle zu gewährleisten

3.3 Sicherheitsaspekte

Datenschutz und Sicherheit werden vor dem Hintergrund der zunehmenden Bedeutung der Telekommunikation im Businessbereich immer wichtiger. Bei der Sicherung der Datenübertragung sind zwei Hauptprobleme zu lösen: Einerseits können Daten durch fehlerhafte oder schlechte Telefonverbindungen falsch übertragen werden. Andererseits müssen die übertragenen Daten vor unerlaubten Zugriffen wie z.B. durch Abhören der Telefonleitungen geschützt werden. Ein weiterer Aspekt in Bezug auf die Sicherheit des Mobile Notes Systems ist die Sicherstellung der Identität des Benutzers sowie die Sicherung des Mobile Notes Servers und des verwendeten Clients.

3.3.1 Schutz der Datenübertragung

Um eine Fehlübertragung von Daten möglichst auszuschließen, müssen die übertragenen Daten auf Übertragungsfehler überprüft werden. "Übertragungsfehler werden in der Regel in der DÜE (Datenübertragungseinrichtung) nicht erkannt und demnach auch nicht der Endeinrichtung angezeigt. Es ist deshalb Aufgabe der DEE (Datenendeinrichtung), unter den gegebenen übertragungstechnischen Randbedingungen für eine für die Anwendung ausreichende Fehlersicherheit zu sorgen.... Jede Fehlerbehandlung setzt voraus, daß die Fehler erkannt und korrigiert werden können, oder, wenn letzteres nicht möglich ist, daß Maßnahmen getroffen werden, die die Auswirkung der fehlerhaften Daten begrenzen."[21] Fehlerkorrektur-verfahren können in der Hardware (Modem) integriert sein oder durch die spezielle Software, die für die Datenübertragung zuständig ist, durchgeführt werden.

Grundsätzlich werden zwei verschiedene Verfahren bei der Fehlerkorrektur unterschieden, welche sowohl in der Hardware als auch in der Datenübertragungssoftware integriert werden können:

- automatic retransmission request (ARQ): Hierbei werden die zu übertragenden Daten in gleich große Blöcke aufgeteilt, durch Prüfsummen ergänzt und anschließend übertragen. Der Empfänger der Datenblöcke ist nun in der Lage, anhand der Prüfsumme fehlerhafte Blöcke festzustellen und eine erneute Übertragung eines fehlerhaften Datenblocks veranlassen.

---

[21] Gabler, Hermann (Technik der Telekommunikation), S. 163

- forward error correction (FEC): Bei der FEC-Technik werden den zu übertragenden Daten zusätzliche Informationen angehängt, die eine Korrektur möglicher fehlerhafter Daten ohne die erneute Übertragung des fehlerhaften Datenblocks ermöglichen. Dieses Verfahren wird meistens dann eingesetzt, wenn eine Blockwiederholung wie bei der ARQ-Technik nicht möglich oder uneffizient ist. (z.B. bei Weltraumsonden, bei denen auf Grund der großen Entfernungen zur Erde die zu übermittelnden Daten oft mehrere Stunden unterwegs sind).

Bei beiden Verfahren werden die zu übertragenden Datenmengen erhöht und es ist keine absolute Fehlerbeseitigung möglich.
In bezug auf das Mobile Notes System ist es sinnvoll, die Fehlerkorrektur in das Übertragungsprotokoll zu integrieren und die Anwendung optional zu gestalten. Auf diese Weise ist es prinzipiell den Entwicklern einer Frontendapplikation überlassen, ob es notwendig ist, für die betreffende Anwendung die Fehlerkorrektur zu implementieren oder nicht und es müssen keine zusätzlichen Hardwarevoraussetzungen z.B. durch bestimmte vorgeschriebene Modem-Übertragungsstandards eingehalten werden, um eine Fehlerkorrektur durchzuführen.
Für die Datenübertragung des Mobile Notes Systems wird das CRC-Fehlerkorrekturverfahren verwendet, das auf der ARQ-Technik basiert und einfacher in Softwarelösungen zu implementieren ist als Verfahren der FEC-Technik. Das CRC-Verfahren wird ausführlich in der in Kapitel 1 erwähnten Diplomarbeit des Studenten Nico Dirks behandelt. Da die Fehlerkorrektur eventuell nicht auf allen Frontendtypen implementiert werden kann, ist sie für die Datenübertragung zwischen dem Frontendsystem und der Mobile Notes Serverkomponente optional, so daß das Datenübertragungsprotokoll auch ohne CRC-Fehlerkorrektur verwendet werden kann.
In ersten Tests der Mobile Notes Serverkomponente mit der im Mobile Notes Projekt entwickelten Frontendapplikation hat sich gezeigt, daß die zusätzlich zu den Daten übertragenen Prüfsummen keine spürbare Beeinträchtigung der Systemgeschwindigkeit verursachten und aufgetretene Übertragungsfehler ohne Probleme erkannt und beseitigt werden konnten.

3.3.2 Schutz der zu übertragenden Daten

Die Kommunikation der Serveranwendung mit den Frontends erfolgt über ein Modem, das an das öffentliche Telefonnetz angeschlossen ist. Standardmäßig werden die Telefonsignale im öffentlichen Netz jedoch nicht verschlüsselt, was den möglichen Mißbrauch übermittelter Daten durch Abhören der Leitungen von unberechtigten Dritten vereinfacht. Es ist deshalb sinnvoll, die Daten, die zwischen Frontend- und Serverapplikation bewegt werden, zu verschlüsseln. Trotzdem sollte der Komfort durch die Verschlüsselung nicht wesentlich beeinträchtigt werden, da lange Wartezeiten, welche durch die Dauer der Verschlüsselung verursacht werden könnten, bei der Datenübertragung außer zusätzlichen Kosten auch Unzufriedenheit beim Benutzer mobiler Frontends erzeugen. Es ist daher ein Verschlüsselungsverfahren zu wählen, das einerseits ausreichenden Schutz vor der Entschlüsselung des verschlüsselten Textes bietet und andererseits die Datenübertragungszeit nicht unnötig belastet. Da die zu übermittelnden Daten sowohl auf der Serverseite als auch auf dem Client ver- bzw. entschlüsselt werden müssen, muß außerdem darauf geachtet werden, daß ein Verschlüsselungsverfahren zum Einsatz kommt, das auch bei geringer Rechnerleistung eine ausreichende Geschwindigkeit bietet, da auf der Clientseite auch weniger leistungsstarke Systeme vorhanden sind.

3.3.2.1 Verschlüsselung

"Ziel der Verschlüsselungsverfahren ist es, Daten in einer solchen Weise einer mathematischen Transformation zu unterwerfen, daß es einem Angreifer nicht möglich ist, die Originaldaten aus den transformierten Daten zu rekonstruieren. Damit die verschlüsselten Daten für ihren legalen Benutzer noch verwendbar bleiben, muß es diesem jedoch möglich sein, durch Anwendung einer inversen Transformation aus ihnen wieder die Originaldaten zu regenerieren."[22]
Im folgenden werden die verschiedenen Verschlüsselungsverfahren kurz vorgestellt und der DES-Verschlüsselungsalgorithmus (data encryption standard), der in der aktuellen Mobile Notes Version zum Einsatz kommt, näher erklärt.

Asymmetrische Verschlüsselungsverfahren:
Asymmetrische Verschlüsselungsverfahren zeichnen sich durch einen Verschlüsselungs-algorithmus aus, der auf zwei verschiedenen Schlüsseln, einem öffentlichen und einem privaten Schlüssel, basiert. Soll eine Nachricht an eine Person verschlüsselt werden, so wird der öffentliche Schlüssel der Person, für die Verschlüsselung der Nachricht verwendet. Die verschlüsselte Nachricht kann nun nur mit dem passenden privaten, nur der Person bekannten Schlüssel entschlüsselt werden. Ein typisches asymmetrisches Verschlüsselungsverfahren ist das RSA-Verfahren, das z.B. auch bei Lotus Notes zum Einsatz kommt. Ein Vorteil

[22] Weck, G. (Datensicherheit), S. 283

asymmetrischer Verfahren ist, daß keine Schlüssel zwischen Empfänger und Absender einer Nachricht ausgetauscht werden müssen. Der für die Verschlüsselung einer Nachricht benötigte Schlüssel ist öffentlich zugänglich und nur wer im Besitz des dazugehörigen privaten Schlüssels ist, kann die Nachricht wieder entschlüsseln. Auf der anderen Seite ist das RSA-Verfahren sehr rechenintensiv und für weniger leistungsstarke Frontendsysteme, wie z.B. dem im Mobile Notes Projekt verwendeten Screenphone, kaum sinnvoll einsetzbar.

Symmetrische Verschlüsselungsverfahren:
Im Gegensatz zu asymmetrischen Verschlüsselungsverfahren werden bei den symmetrischen Verfahren für die Ver- und Entschlüsselung von Daten die gleichen Schlüssel verwendet. Der Schlüssel wird für die Entschlüsselung der Daten einer Transformation unterzogen, um mit dem selben Verschlüsselungsalgorithmus die Originaldaten wiederherzustellen. Ein Standard-verschlüsselungsverfahren, daß einen symmetrischen Verschlüsselungsalgorithmus verwendet, ist das DES-Verfahren, auf das im folgenden noch eingegangen wird. Ein Vorteil gegenüber asymmetrischen Verfahren ist die relativ hohe Geschwindigkeit mit der Daten ent- bzw. verschlüsselt werden können, wodurch es auch für weniger leistungsstarke Rechnersysteme sinnvoll einsetzbar ist. Allerdings ist es bei diesen Verfahren notwendig, daß sowohl die verschlüsselnde, als auch die entschlüsselnde Seite den verwendeten Schlüssel kennt. Daher muß zusätzlich sichergestellt sein, daß die Übermittlung des Schlüssels vom Absender der Nachricht zum Empfänger auf eine Weise erfolgt, die nicht durch unberechtigte Dritte einsehbar ist. Außerdem ist die Sicherheit vor Entschlüsselung der verschlüsselten Daten bei symmetrischen Verfahren geringer als bei asymmetrischen Verfahren.

3.3.2.2 Data Encryption Standard (DES)

Der DES bezeichnet ein von IBM in den 70ger Jahren entwickeltes Verschlüsselungsverfahren. Die Vorteile des DES liegen in seiner vergleichsweise einfachen Implementation sowie seiner hohen Geschwindigkeit bei der Ver- bzw. Entschlüsselung von Daten, weshalb er auch für weniger leistungsstarke Systeme, wie z.B. PDA's, ohne großen Komfortverlust einsetzbar ist. Die Verschlüsselung besteht im wesentlichen aus einer Iteration von Substitutionen und Permutationen. Es werden jeweils 64 Bit des Klar-(Schlüssel-)textes unter Verwendung eines für die Ver- bzw. Entschlüsselung des gesamten Textes gültigen Schlüssels von 56 Bit Länge in 64 Bit Schlüssel-(Klar-)text umgesetzt. Für die Ver- und Entschlüsselung wird der gleiche Schlüssel verwendet.[23] Da der DES keinen Text sondern nur die Symbole 0 und 1 verschlüsselt, müssen die Buchstaben eines Textes zu Beginn der Ent- bzw. Verschlüsselung in Bitfolgen dargestellt sein. Hierzu wird üblicher Weise auf den ASCII-Code zurückgegriffen, der den Zeichen einen bestimmten Wert zuordnet. Anschließend erfolgt, unter Verwendung des DES-Schlüssels, die Ver- bzw. Entschlüsselung durch 16 Iterationsschritte.

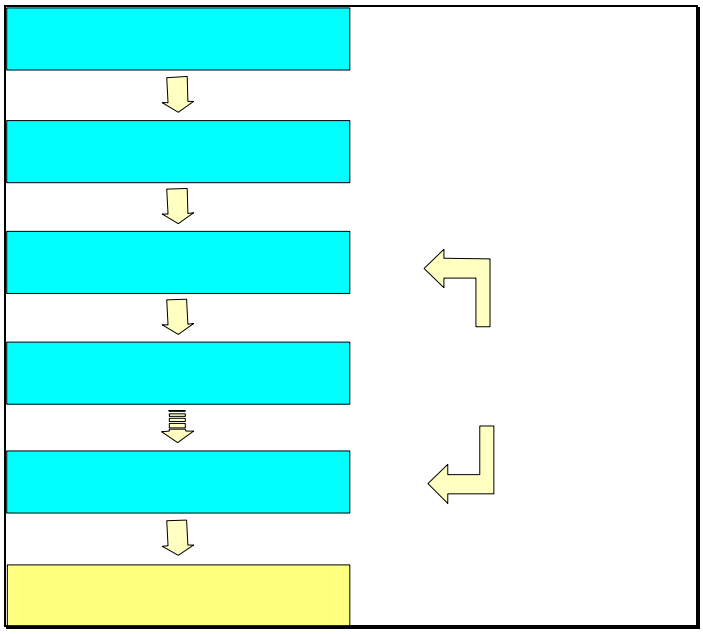

Abb. 8: Überblick über das DES-Verschlüsselungsverfahren

Die Verschlüsselung nach dem DES-Standard kann auf verschiedene Arten durchgeführt werden, die unterschiedlichen Aufwand erfordern, jedoch die Sicherheit steigern können. Bei dem in dieser Arbeit verwendeten ECB-Modus (electronic code book) wird der zu verschlüsselnde Text in 64 Bit-Blöcke zerlegt, die unabhängig voneinander jeweils mit demselben DES-Schlüssel verschlüsselt werden. Es ist der schnellste Modus, bietet aber im Vergleich zum CBC-Modus (cipher block chainging), wo eine Abhängigkeit der einzelnen zu verschlüsselnden Blöcke erzeugt wird, eine geringere Sicherheit vor der Entschlüsselung des Textes durch nicht berechtigte Dritte.

---

[23] vgl. Weck,G. (Datensicherheit), S. 290 ff.

3.3.3 Schutz vor unberechtigtem Zugriff

Neben dem Abhören der Telefonleitung ist auch der Mißbrauch des Systems durch unberechtigte Systembenutzung denkbar. Es muß sichergestellt sein, daß nur dem System bekannte, registrierte Anwender dieses benutzen können. Hierzu muß die Identifikation des Benutzers einen Authentikations-Mechanismus enthalten, der es dem System ermöglicht, die Identität des Benutzers zu überprüfen.
Bei der Mobile Notes Anwendung wird das am weitesten verbreitetste Authentikations-verfahren, die Anmeldung des Benutzers mit zusätzlicher Paßwortabfrage, verwendet, welches einfach zu realisieren und zu bedienen ist. Hierzu wird für jeden eingetragen Systembenutzer ein Paßwort vergeben. Zu Beginn einer Verbindung des Mobile Notes Clients mit dem Server werden zuerst der Name des Benutzers sowie dessen Paßwort übertragen, und erst bei Übereinstimmung von Benutzername und Paßwort ist eine weitere Benutzung der Serverapplikation möglich. Die Authentikation durch ein Paßwort bietet genügend Schutz vor unberechtigten Systembenutzern, wenn folgende Regeln eingehalten werden:

- Paßwörter sollten eine gewisse Länge nicht unterschreiten, um nicht durch bloßes "Probieren" erraten zu werden.
- Paßwörter sollten vom Benutzer jederzeit verändert werden können. Sollte ein Paßwort ausspioniert worden sein, so kann der Benutzer auf diese Weise seine Daten erneut sichern und möglichen Schaden begrenzen.
- Das Umfeld des Benutzers sollte nicht auf sein Paßwort schließen lassen. So sind z.B. der Geburtstag des Benutzers oder der Name seines Ehepartners als Paßwort nicht geeignet.

Zusätzlich werden von der Mobile Notes Serverapplikation feindliche Paßwortattacken unberechtigter Systembenutzer erschwert, indem nach einer frei definierten Anzahl falsch eingegebener Paßwörter die Serveranwendung die Verbindung zum Client abbricht und automatisch eine Nachricht an den Systemadministrator sowie den vermeindlichen Systembenutzer schickt. Auf diese Weise ist sichergestellt, daß Zugriffsversuche nicht berechtigter Personen frühzeitig entdeckt und entsprechende Gegenmaßnamen getroffen werden können.

3.3.4 Konzeption des Anmeldevorgangs

Die Anmeldung des Mobile Notes Clients beim Mobile Notes Server ist von besonderer Bedeutung, da hierbei das Paßwort des Benutzers übertragen wird. Es sollten an dieser Stelle deshalb besondere Sicherheitsmaßnahmen verwendet werden, um ein Ausspionieren des Paßwortes zu vermeiden, sowie ein Aufzeichnen des jeweiligen Antwortstrings wertlos zu machen. In der Konzeption des Mobile Notes Projektes war daher für Anmeldung des Benutzers beim Mobile Notes Server die Verwendung des asymmetrischen Verschlüsselungsverfahrens RSA angedacht, da dieses höhere Sicherheit als das symmetrische DES-Verschlüsselungsverfahren bietet.
Hierzu sollte folgender Anmeldevorgang realisiert werden:

RSA-Anmeldungskonzept:

1. Die Clientanwendung schickt den Namen des Benutzers sowie eine auf dem Client generierte Zufallszahl mit dem öffentlichen RSA-Schlüssel des Mobile Notes Servers verschlüsselt an den Mobile Notes Server. Dieser entschlüsselt die gesendeten Daten und ermittelt anhand des Namens, ob der Benutzer Zugriff auf das Mobile Notes System hat.

2. Ist der Benutzer in der Mobile Notes Initialisierungsdatenbank eingetragen und hat er somit Zugriff auf das Mobile Notes System, sendet der Mobile Notes Server die zuvor vom Mobile Notes Client übermittelte Zufallszahl sowie eine zweite vom Server selbst generierte Zahl, mit dem privaten RSA-Schlüssel verschlüsselt, zurück zum Client.

3. Die Clientanwendung ist nun in der Lage, den Mobile Notes Server zu überprüfen, da die vom Client generierte Zufallszahl nur von dem korrekten Server mit dem privaten RSA-Schlüssel entschlüsselt und zurückgeschickt werden kann. Die hierzu verwendete Zufallszahl soll dabei eine mögliche Aufzeichnung einer Serverantwort wertlos machen, da bei jedem Anmeldevorgang ein anderer Antwortstring erstellt wird. Es wird somit vermieden, daß an dieser Stelle ein zuvor aufgezeichneter Antwortstring durch einen fremden Server eingespielt werden kann. Nun überträgt der Client, wiederum mit dem öffentlichen RSA-Schlüssel verschlüsselt, das Paßwort sowie die durch den Server generierte Zufallszahl an den Server. Auch hier dient die zusätzliche Übertragung der durch den Server generierten Zufallszahl dazu, eine Aufzeichnung des gesendeten Antwortstrings wertlos zu machen, da auf diese Weise jeder Passwortübertragungsstring einmalig ist.

Wird anhand des Paßwortes der Benutzer authentisiert, erfolgt die weitere Datenübertragung aus Geschwindigkeitsgründen DES-verschlüsselt mit dem jeweiligen DES-Schlüssel des Benutzers.

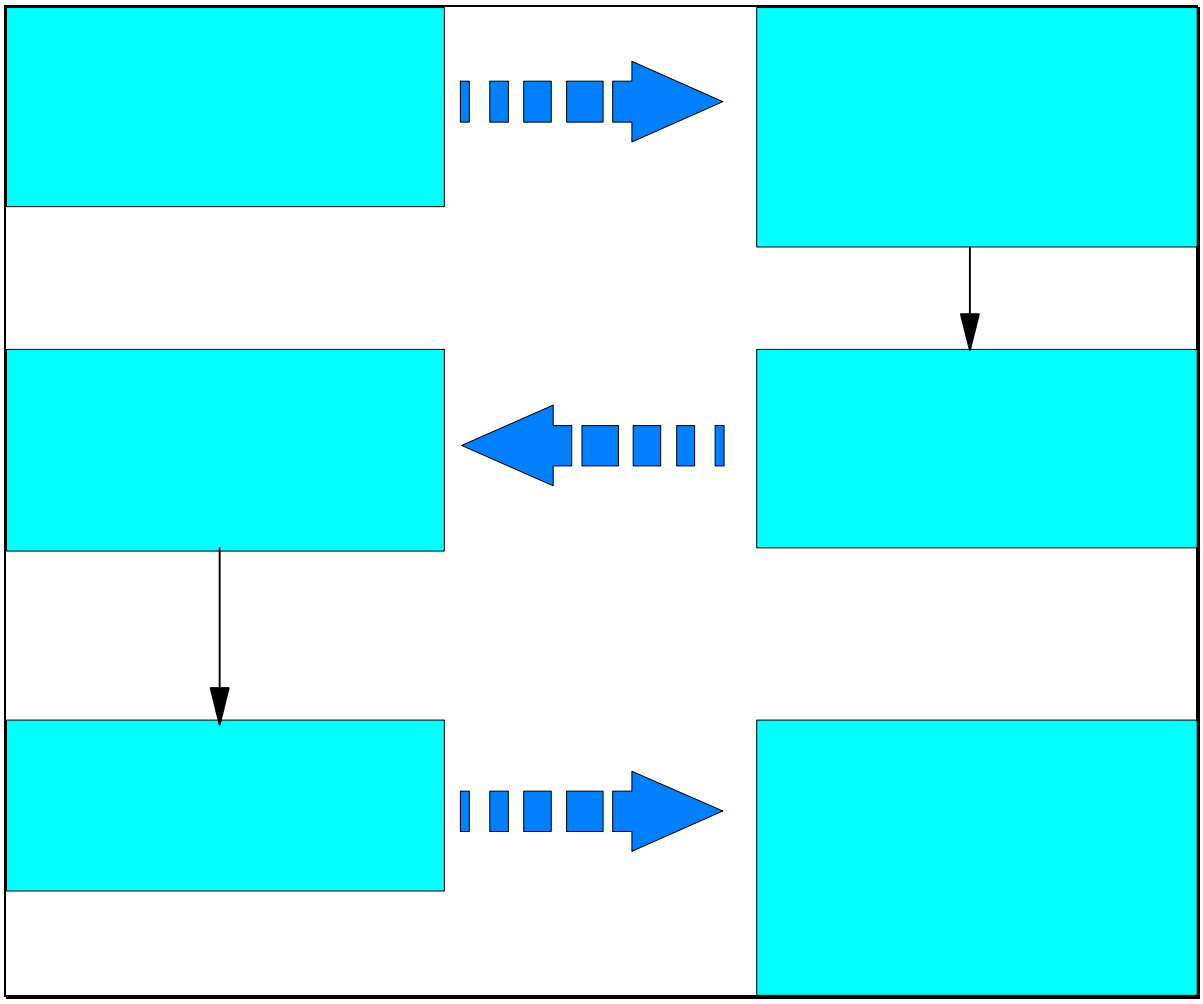

Abb. 9 RSA-Anmeldungskonzept

Auf Grund der sehr schwierigen RSA-Schlüsselgenerierung sowie fehlender RSA-Quellen ist das hier beschriebene Anmeldungskonzept noch nicht für die aktuelle Mobile Notes Version verwendet worden. Statt dessen ist das nun folgende Anmeldungskonzept implementiert, daß nur das DES-Verschlüsselungsverfahren verwendet, was jedoch im Vergleich zur oben beschriebenen Vorgehensweise, weniger Sicherheit bietet.

DES-Anmeldungskonzept:

1. Die Clientanwendung schickt den Namen des Benutzers sowie eine auf dem Client generierte Zufallszahl, welche mit dem DES-Schlüssel des Client-Anwenders verschlüsselt, an den Mobile Notes Server. Dieser kann anhand des unverschlüsselt übertragenen Benutzernamens zum einen überprüfen, ob der Benutzer überhaupt im Mobile Notes System angemeldet ist und wenn ja, auch den für die Entschlüsselung der Zufallszahl benötigten DES-Schlüssel ausfindig machen (der DES-Schlüssel befindet sich im Benutzerprofil des Benutzers auf das durch den einmalig vorhandenen Benutzernamen zugegriffen wird).

2. Ist der Benutzer in der Mobile Notes Initialisierungsdatenbank eingetragen und hat er somit Zugriff auf das Mobile Notes System, sendet der Mobile Notes Server die zuvor vom Mobile Notes Client übermittelte Zufallszahl um den Wert 1 erhöht sowie eine zweite vom

Server selbst generierte Zahl, mit dem DES-Schlüssel des Benutzers verschlüsselt, zurück zum Client. Durch die Erhöhung der Zufallszahl des Clients soll vermieden werden, daß der Antwortstring des Servers eventuell Ähnlichkeiten mit dem zuvor vom Client empfangenen String aufweist, da bei der DES-Verschlüsselung der Schlüssel für die Ver- und Entschlüsselung identisch ist und auf diese Weise bei den gleichen Klartexten ein identischer Schlüsseltext erzeugt wird.

3. Die Clientanwendung ist nun in der Lage, den Mobile Notes Server zu überprüfen, da die vom Client generierte Zufallszahl nur von dem korrekten Server mit dem jeweiligen DES-Schlüssel entschlüsselt und (um 1 erhöht) zurückgeschickt werden kann. Es wird somit vermieden, daß an dieser Stelle ein zuvor aufgezeichneter Antwortstring durch einen fremden Server eingespielt werden kann. Nun überträgt der Client, wiederum mit dem DES-Schlüssel verschlüsselt, das Paßwort sowie die durch den Server generierte Zufallszahl (um 1 erhöht) an den Server. Auch hier dient die zusätzliche Übertragung der durch den Server generierten Zufallszahl dazu, eine Aufzeichnung des gesendeten Antwortstrings wertlos zu machen, da so jeder Passwortübertragungsstring einmalig ist.

Nach der Benutzerauthentisierung, erfolgt die weitere Datenübertragung DES-verschlüsselt.

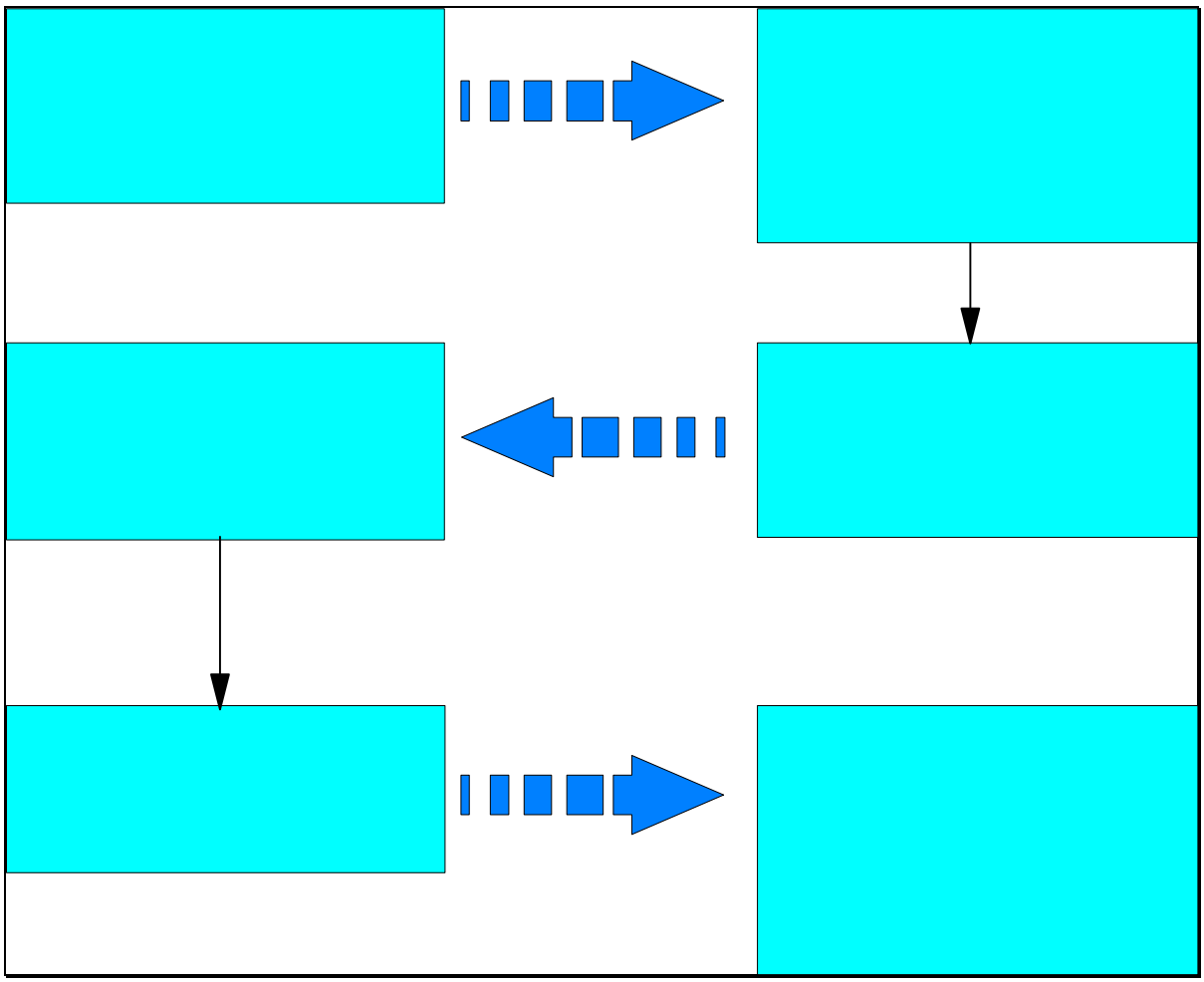

Abb. 10 DES-Anmeldungskonzept

Auf Grund der geringeren Sicherheit der hier vorgestellten Verschlüsselungsvariante der Anmeldung (unverschlüsselte Übertragung des Benutzernamens) ist das RSA-Verschlüsselungskonzept für den Anmeldevorgang vorzuziehen. Jedoch ist durch fehlende Programmquellen und Lizensierungskosten des RSA-Verfahrens, sowie die problematische Schlüsselgenerierung die Realisierung im Rahmen dieser Diplomarbeit nicht möglich. Können die hier erwähnten Probleme gelöst werden, so ist eine Erweiterung des bestehenden Anmeldevorgangs um die RSA-Verschlüsselung wegen der höheren Sicherheit sinnvoll.

3.3.5 Sicherung des Mobile Notes Servers und der Initialisierungs-datenbank

Der Sicherung der Mobile Notes Initialisierungsdatenbank kommt eine besondere Bedeutung zu, da sich in ihr sämtliche Paßwörter sowie die speziellen DES-Schlüssel eines jeden Systembenutzers befinden. Es muß gewährleistet sein, daß diese Datenbank nur von autorisierten Personen wie dem Systemadministrator eingesehen und bearbeitet werden kann, da mit Hilfe des Paßwortes und des DES-Schlüssels eines Benutzers ein nicht berechtigter Dritter Zugriff auf alle für den Systembenutzer eingerichteten Datenbanken hat. Mögliche Schutzmaßnahmen sind z.B. die Sicherung des Raumes, in dem sich der Mobile Notes Server befindet, durch Einlaßkontrollvorrichtungen etc.. Da die Vorsichts- und Sicherheits-maßnahmen für den Mobile Notes Server jedoch im Bereich des Anwenders liegen und die Mobile Notes Serverapplikation hierauf keinen Einfluß nehmen kann, soll an dieser Stelle auf die entsprechende Literatur verwiesen werden[24].

3.3.6 Sicherung des Mobile Notes Clients

Ebenso wie der Server sollte auch der Mobile Notes Client vor unberechtigten Zugriffen Dritter geschützt werden. Ein Schutz dieser Systeme ist jedoch von der Beschaffenheit des Frontendsystems abhängig. So sind auf "Low Level" Systemen wie dem "Screenphone P100" nur geringe Sicherheitsmaßnahmen wie eine Paßwortabfrage vorstellbar. Leistungsfähigere Systeme können aber z.B. Informationen eine SmardCard auslesen, die so als Schlüssel für die Benutzung des Systems fungieren und einen wesentlich besseren und komfortableren Schutz als Paßwortabfragen bieten. Doch sei auch an diese Stelle auf die entsprechende Literatur sowie die Diplomarbeiten der am Mobile Notes Projekt beteiligten Studenten Nico Dirks und Dirk Sievers verwiesen, da eine tiefergehende Erörterung dieser Probleme den Rahmen der vorliegenden Diplomarbeit überschreiten würde.

[24] siehe z.B. Weck, G. (Datensicherheit)

3.4 Entwicklungskomponenten

Der Mobile Notes Servertask wurde in der Entwicklungsumgebung Borland C++ 4.52 unter Windows NT erstellt. Durch die Verwendung der Programmiersprache C++ ist es ohne Schwierigkeiten möglich, den Mobile Notes Servertask auch auf andere Betriebssysteme wie z.B. OS/2 zu portieren, da die Sprache nicht an ein bestimmtes Betriebssystem gebunden ist.[25] Ein weiterer Vorteil liegt in der weiten Verbreitung dieser Programmiersprache. Für viele Frontendsysteme wie z.B. das "Screenphone P 100" der Philips AG können Anwendungen in C entwickelt werden, wodurch eine einfache Anpassung und Nutzung der im Mobile Notes Projekt entwickelten Module gewährleistet ist.
Für die Verbindung der Serverapplikation zu Lotus Notes wird die im Rahmen des Mobile Notes Projektes von dem Studenten Nico Dirks mit der Borland C++ Entwicklungsumgebung erstellte AVAS_NOTE.DLL verwendet. Sie ist auf Basis der HiTest.DLL der Firma Edge Research erstellt worden, und faßt die von der Serverkomponente benötigten Navigations- und Auslesebefehle für Lotus Notes Datenbanken zusammen.[26]
Für die Entwicklung der Mobile Notes Initialisierungsdatenbank wurde die Lotus Notes 3.3 Datenbankumgebung verwendet.

---

[25] vgl. Kerninghan, Brian W.; Ritchie, Dennis M.: (Programmieren in C)

[26] weitere Erläuterungen zur AVAS_NOTE.DLL können der Diplomarabeit "Architekturen und Anwendungskonzepte von Groupware in der mobilen Kommunikation - Generische Entwicklung von Benutzungsschnittstelle und Prozeß-Modulen für ein intelligentes Display-Telephon." Autor: Nico Dirks entnommen werden.

4. Praktische Umsetzung - Die Mobile Notes Serveranwendung

4.1 Die Mobile Notes Serverkomponente

4.1.1 Funktionalitäten

Die Mobile Notes Anwendung ist ein speziell für "Low Level"-Frontends, wie z.B. dem Screenphone der Philips AG entwickelte Notes-Schnittstelle, die es den Frontendsystemen ermöglicht, Notesdatenbanken in begrenztem Umfang zu nutzen. Da die Kommunikation mit dem Mobile Notes System über ein handelsübliches, an das Telefonnetz angeschlossenes Modem erfolgt, ist nahezu jeder programmierbare Client, der ein Modem ansteuern kann, in der Lage, mit der Mobile Notes Serveranwendung auf Notesdatenbanken zuzugreifen.

Folgende Funktionalitäten sind in der aktuellen Mobile Notes Serverkomponente implementiert und können von den Clientanwendungen genutzt werden:

- Sicherung der zu übertragenden Datenpakete durch DES-Verschlüsselung. Da für die Datenübertragung das öffentliche Telefonnetz verwendet wird, werden alle Daten, die nach der namentlichen Anmeldung des Systemnutzers zwischen dem Client und dem Mobile Notes Server gesendet werden, mit dem DES (Data Encryption Standard) verschlüsselt. Hierzu wird für jeden Benutzer ein eigener DES-Schlüssel verwendet, um durch das mögliche Ausspionieren eines einzelnen Schlüssels eines Users nicht das ganze System zu gefährden. Die genaue Vorgehensweise der Datenübertragung wird im Verlauf dieses Kapitels näher behandelt.
- Optionale Überprüfung der Datenübertragung auf Übertragungsfehler mittels CRC-Prüfung (Cyclic Redundancy Check). Je nach Leistungsfähigkeit des Clients und des verwendeten Modems ist es sinnvoll, die empfangenen Datenpakete auf Fehler, die z.B. durch die Datenübertragung über das Telefonnetz verursacht werden können, zu überprüfen. Diese Funktion ist optional, da die Überprüfung der Daten je nach Leistungsfähigkeit der verwendeten Komponenten Zeit benötigt und die Systembenutzung dadurch an Komfortabilität einbüßen könnte[27].
- Zugriff auf für den Benutzer im Mobile Notes System eingerichtete und freigeschaltete Notesdatenbanken, wobei jede Datenbank nochmals durch die folgenden Zugriffsmodi näher bestimmt wird:
  - Nur lesen von Dokumenten, z.B. für Informationsdatenbanken

[27] siehe Kapitel 3.3 Sicherheitsaspekte

- Lesen und erzeugen von Dokumenten für Datenbanken die z.B. der Projektarbeit dienen
- Nur Erzeugen von Dokumenten für z.B. private Termindatenbanken, wobei Termine durch eine andere Person eingegeben aber nicht mehr gelesen werden können
- Lesen, erzeugen und ändern von Dokumenten z.B. für die private Maildatenbank eines Benutzers

Je nach definiertem Zugriffstyp stehen dem Benutzer weitere Funktionen in der Datenbank zur Verfügung, wie die folgende Tabelle dokumentiert:

Zugriffstyp Funktionen	Nur lesen	Lesen und erzeugen	Nur erzeugen	Lesen, erzeugen und ändern
View abfragen	möglich	möglich	nicht möglich	möglich
Dokomuent löschen	nicht möglich	nicht möglich	nicht möglich	möglich
Dokument erstellen	nicht möglich	möglich	möglich	möglich
Dokument ändern	nicht möglich	nicht möglich	nicht möglich	möglich
Dokument auf den Client herunterladen (Offline-Betrieb)	möglich	möglich	nicht möglich	möglich

Tab. 3: Funktionen in Abhängigkeit vom Datenbankzugriffstyp

4.1.2 Vorteile und Einsatzmöglichkeiten

Das Mobile Notes System ermöglicht es, Lotus Notes im begrenzten Umfang auch auf Systemen zu nutzen, die bisher z.B. durch ihre Prozessorleistung oder begrenzten Speicherumfang nicht dazu in der Lage waren. Es wird hierbei vor allem an die tragbaren "Westentaschencomputer" wie z.B. den Apple Newton gedacht, die ein PCMCIA-Modem ansteuern und so auf das Mobile Notes System zugreifen können. Gerade für solche Clients, die vorzugsweise als Terminkalender oder persönliche Datenbank genutzt werden, war es bisher nicht möglich, in diesem Umfang Groupware Backend-Systeme zu nutzen. Während der Entstehung der vorliegenden Arbeit ist von der Firma Ives Development das Produkt "Team Agent" speziell für den Apple Newton entwickelt worden.
"With the product, users will be able to send Notes mail from the Newton and synchronize information between the Newton and corporate Notes databases on their desktop Notes clients..."[28]

[28] unbekannter Autor (TeamAgent will synchronize Notes databases with Newtons)

Im Unterschied zu der im Rahmen dieser Arbeit entwickelten Serverkomponente, handelt es sich bei diesem Produkt um eine Anwendung, die nur für den Apple Newton ausgelegt ist. Die mit dem Mobile Notes System zur Verfügung gestellten Protokollbefehle[29] können jedoch als Grundlage für Lotus Notes Anwendungen auf jedem programmierbaren Client dienen, der die oben beschriebenen Voraussetzungen erfüllt. So wird die in dieser Arbeit erstellte Mobile Notes Komponente von den Studenten Nico Dirks und Dirk Sievers genutzt, um auf dem "Screenphone P 100" der Philips AG eine Anwendung zu programmieren, die es dem Benutzer ermöglicht, Lotus Notes Datenbanken zu nutzen.
Mit Hilfe der generischen Protokollbefehle lassen sich somit vielfältige Anwendungen realisieren, die zum einen die Vorteile der Frontendsysteme wie z.B. geringes Gewicht, kleine Maße, geringere Kosten oder hohe Mobilität mit den Nutzen der Lotus Notes Groupwareumgebung vereinen. Es können nun "maßgeschneiderte" Nischenanwendungen, die nicht den vollen Notes Funktionsumfang benötigen, realisiert werden. So sind z.B. tragbare mobile Datenerfassungssysteme denkbar, bei denen die Daten auf dem mobilen Frontend eingegeben und dann mittels Modem an den Mobile Notes Server weitergeleitet werden. Auf diese Art und Weise sind die vor Ort erfaßten Daten aktuell im Notessystem verfügbar und können entsprechend ihrer Bestimmung weitergeleitet oder verarbeitet werden, wobei voll auf die Funktionalitäten einer Groupwareumgebung zurückgegriffen werden kann.
Außerdem werden durch den Einsatz von kleineren "Low Level"-Frontends sowie des Mobile Notes Systems Lizens- und Hardwarekosten eingespart, da z.B. nur eine Notes-Lizens für den Mobile Notes Server erworben werden muß.

4.1.3 Einschränkungen der Mobile Notes Serverkomponente

Da das Mobile Notes System speziell für weniger leistungsstarke Computerfrontends ausgelegt ist, ergeben sich einige Einschränkungen in der Nutzung von Lotus Notes, auf die im folgenden eingegangen werden soll:
In der augenblicklich vorliegenden Version des Mobile Notes Systems können nur ASCII-Zeichen an die Clients übertragen werden. Die Möglichkeiten, wie sie z.B. Ritchtextfelder in Notes bieten können noch nicht genutzt werden. An dieser Stelle soll darauf hingewiesen werden, daß viele auf dem Markt befindliche mobile Frontendsysteme nicht in der Lage sind, andere Formate wie z.B. Bilder oder Videos darzustellen. Zusätzlich würde die Übertragung solcher Dateien wegen ihrer Größe relativ viel Zeit in Anspruch nehmen, wodurch die Anwendungen wenig komfortabel würden. Es ist jedoch davon auszugehen, daß die fortschreitende Entwicklung die Leistungsfähigkeit dieser Systeme zunehmend steigern wird, weshalb eine Ergänzung des Mobile Notes Systems um solche Funktionalitäten in der Zukunft durchaus sinnvoll ist.

---

[29] siehe Anhang Protokollbefehle und Konstanten

Jeder Mobile Notes Nutzer kann jeweils auf nur einen View pro Datenbank zugreifen. Die Zusammenstellung der Viewzeile kann jedoch vom Systemadministrator frei definiert werden. Für die Erstellung von Dokumenten kann nur ein Dokumententyp pro Datenbank verwendet werden. Außerdem ist die Darstellung der Dokumente einer Datenbank auf maximal 10 Felder begrenzt. Ebenso können beim Erstellen neuer Dokumente höchstens 10 Felder editiert werden.
Die Gründe für diese Begrenzungen liegen in der Einrichtung der einzelnen Datenbanken in der Mobile Notes Initialisierungsdatenbank. Eine Erweiterung der anzubietenden Views, Dokumententypen sowie darzustellender Felder ist ohne größeren Programmieraufwand zu realisieren. Allerdings zeigte der bereits existierende AvALoN Prototyp, daß die für das Mobile Notes System gewählten Darstellungsmöglichkeiten einer Notes Datenbank ausreichend sind.
Mit der augenblicklich vorliegenden Mobile Notes Serverversion kann nur ein Modem angesteuert werden, da die Notes 3.3 API, auf die der Datentransfer der Serverkomponente mit Lotus Notes basiert, nicht für multithreaded-Anwendungen ausgelegt ist. Um mehrere Modems an einem Server zu verwenden, müßte also die Serveranwendung mehrmals gestartet werden, was zu Geschwindigkeitseinbußen der Serveranwendung führt. Zukünftige API-Versionen (Notes 4.0) sollen jedoch in der Lage sein, auch multithreaded-Anwendungen zu unterstützen. Eine Erweiterung der Mobile Notes Serveranwendung um die Fähigkeit mehrere Modems anzusteuern wäre dann durchaus sinnvoll, da durch den Anschluß mehrerer Modems an den Mobile Notes Server mehrere Clients, ohne einen spürbaren Geschwindigkeitsverlust, gleichzeitig auf Lotus Notes Datenbanken zugreifen könnten.
Zusammenfassend soll noch einmal darauf hingewiesen werden, daß mit dem Mobile Notes System und dem zu benutzenden Protokoll kein kompletter Notes Client dargestellt werden kann. Vielmehr soll für weniger leistungsstarke Systeme eine Möglichkeit geschaffen werden, Lotus Notes Datenbanken im begrenzten Umfang zu nutzen.
Die aktuelle Mobile Notes Version bietet jedoch genügend Möglichkeiten um Textdatenbanken wie die Maildatenbank eines Benutzers oder Termindatenbanken etc. im ausreichenden Umfang zu bearbeiten und stellt somit eine Entwicklungsgrundlage für neue Anwendungen auf Frontendsystemen wie z.B. dem "Screenphone P100" dar.
Für die Zukunft ist eine Erweiterung des Mobile Notes Systems um zusätzliche Funktionalitäten wie z.B. der Übertragung von Bild- oder Videodateien durchaus sinnvoll, da die Leistungsfähigkeit der Hardware stetig zunehmen wird und somit auch auf mobilen Computersystemen komplexere Anwendungen erstellt werden können.

4.2 Installation der Mobile Notes Serverkomponente

Um die Mobile Notes Serverkomponente einzurichten, werden folgende Hardware- und Softwarekomponenten benötigt:

- Standard-PC (optional mit Netzwerkanschluß)
- Handelsübliches Modem, das den Heyes-Befehlsatz beherrscht (z.B. ZyXEL 1496 EG Plus) und an das Telefonnetz angeschlossen werden kann.
- Windows NT
- Lotus Notes 3.x Serverversion für Windows NT
- Mobile Notes Serverkomponente sowie die dazugehörige Mobile Notes Initialisierungs-datenbank

Die Grundvoraussetzung für die Benutzung der Mobile Notes Serverkomponente ist die Installation von Windows NT und dem Lotus Notes Server für Windows NT. (Alternativ ist auch die Verwendung von IBM OS/2 als Betriebssystem für die Mobile Notes Serverkomponente vorgesehen, jedoch wurde die Serverkomponente noch nicht mit diesem Betriebssytem getestet.) Die Mobile Notes Serverkomponente wird unter Windows NT gestartet und benötigt die Mobile Notes Intialisierungsdatenbank, welche sich im Hauptverzeichnis des Notes Servers befinden sollte. Um auf Datenbanken zugreifen zu können, muß der Server in die ACL (Access Control List) der im Mobile Notes System eingerichteten Datenbanken eingetragen sein, da die Mobile Notes Serverkomponente mit der Standard-ID des Severs arbeitet und keinen ID-Wechsel vornimmt.
Optional kann die Mobile Notes Serverkomponente mit folgenden Parametern gestartet werden:

- \l + Pfad und Name einer Log-Datei, in die sämtliche Bildschirmausgaben umgeleitet werden können, um gegebenenfalls Fehler in der Funktion der Mobile Notes Server-komponente entdecken zu können. Bsp.: mobileno\lc:\tmp\logdatei.txt
- \c + Pfad und Name der Mobile Notes Initialisierungsdatenbank, falls der Pfad bzw. der Name von den Vorgaben abweichen. Bsp.: mobileno\cc:\banken\mobileno.nsf
- \p + Serial Device an das das benützte Modem angeschlossen ist. Bsp.:mobileno\pcom1 Sollte kein explizieter serieller Anschluß angegeben worden sein, so wird die erste in der Mobile Notes Initialisierungsdatenbank definierte Schnittstelle verwendet. Diese Lösung wurde erarbeitet, da es mit dem Notes 3.x API nicht möglich ist, Multithread-Anwendungen zu erstellen und die nun vorliegende Mobile Notes Serverkomponente somit nur ein Modem ansteuern kann. Sollen mehrere Modems an einem Mobile Notes Server betrieben werden, so muß der Mobile Notes Servertask mehrmals gestartet werden. Dieses kann jedoch den Programmablauf stark verlangsamen.

Für den Pfad der Mobile Notes Initialisierungsdatenbank ist das Hauptverzeichnis des Lotus Notes NT Servers voreingestellt. Programmausgaben erfolgen standardmäßig auf dem Bildschirm. Die optionale Möglichkeit eine Logdatei anzulegen und sämtliche Ausgaben mitzuprotokollieren, dient dem Auffinden von Installationsfehlern der Mobile Notes Serverkomponente.
Ausgaben, die standardmäßig im Programmfenster der Mobile Notes Anwendung ausgegeben werden, enthalten folgende Informationen:

- Name des Benutzers, der in das Mobile Notes System einloggen möchte
- Aktionen, die der Benutzer im Mbile Notes System durchführt wie z.B.:
  - Zugriff auf Maildatenbank
  - View anfordern
  - Erstellen eines Dokumentes in der entsprechenden Datenbank etc.

Zusätzlich werden aber auch Fehlermeldungen, die z.B. durch das Senden eines falschen Paßwortes vom Mobile Notes Client an den Server verursacht werden, auf dem Bildschirm ausgegeben. Je nach der Bedeutung des entstandenen Fehlers wird auch eine Mail an den in der Mobile Notes Initialisierungsdatenbank eingetragenen Systemadministrator geschickt, um ihn z.B. darauf hinzuweisen, daß z.B. eine Datenbank falsch eingerichtet ist oder sich nicht mehr im System befindet.

4.3 Systemarchitektur der Serverkomponente

Die Serveranwendung setzt sich aus dem Mobile Notes Servertask sowie der Mobile Notes Datenbank zusammen. Der Mobile Notes Servertask ist modular aufgebaut, was den Vorteil bietet, Änderungen bzw. Anpassungen des Mobile Notes Systems durch gezielte Anpassungen bzw. durch Austausch der entsprechenden Module zu realisieren. Die vorliegende Mobile Notes Serverapplikation besteht aus den folgenden Modulen:

- AVASPROG.C: Das AVASPROG.C Modul ist das Kernmodul der Serveranwendung. Von diesem Modul ausgehend, werden die Mobile Notes Protokollbefehle verarbeitet und die Funktionen aus den folgenden Modulen aufgerufen.
- AVASNOTE.DLL: Im AVASNOTE.DLL Modul sind sämtliche Befehle, die dem Servertask den Zugriff auf Lotus Notes Datenbanken ermöglichen, zusammengefaßt. Für die Programmierung der AVASNOTE.DLL-Befehle wurde die Hitest 2.21.DLL der Firma Edge Research Inc.[30] verwendet, die wiederum auf dem Lotus Notes API basiert.
- AVASCOM.C: Alle Befehle, die die Modemkommunikation der Mobile Notes Serveranwendung betreffen, sind im AVASCOM.C Modul enthalten.
- AVABDES.C: Für die DES-Verschlüsselung wurden die in Bruce Schneider's 'Applied Cryptography' dokumentierten Funktionen verwendet.[31] Diese Funktionen sind im AVABDES.C Modul zusammengefaßt.
- AVABCRC.C: In dem AVABCRC.C Modul sind die Funktionen für die CRC-Überprüfung der zu übertragenden Daten enthalten.

[30] die Firma Edge Research ist im vollständigen Besitz der Lotus Development Corporation
[31] vgl. Bruce Schneider's (Applied Cryptgraphy) Anhang

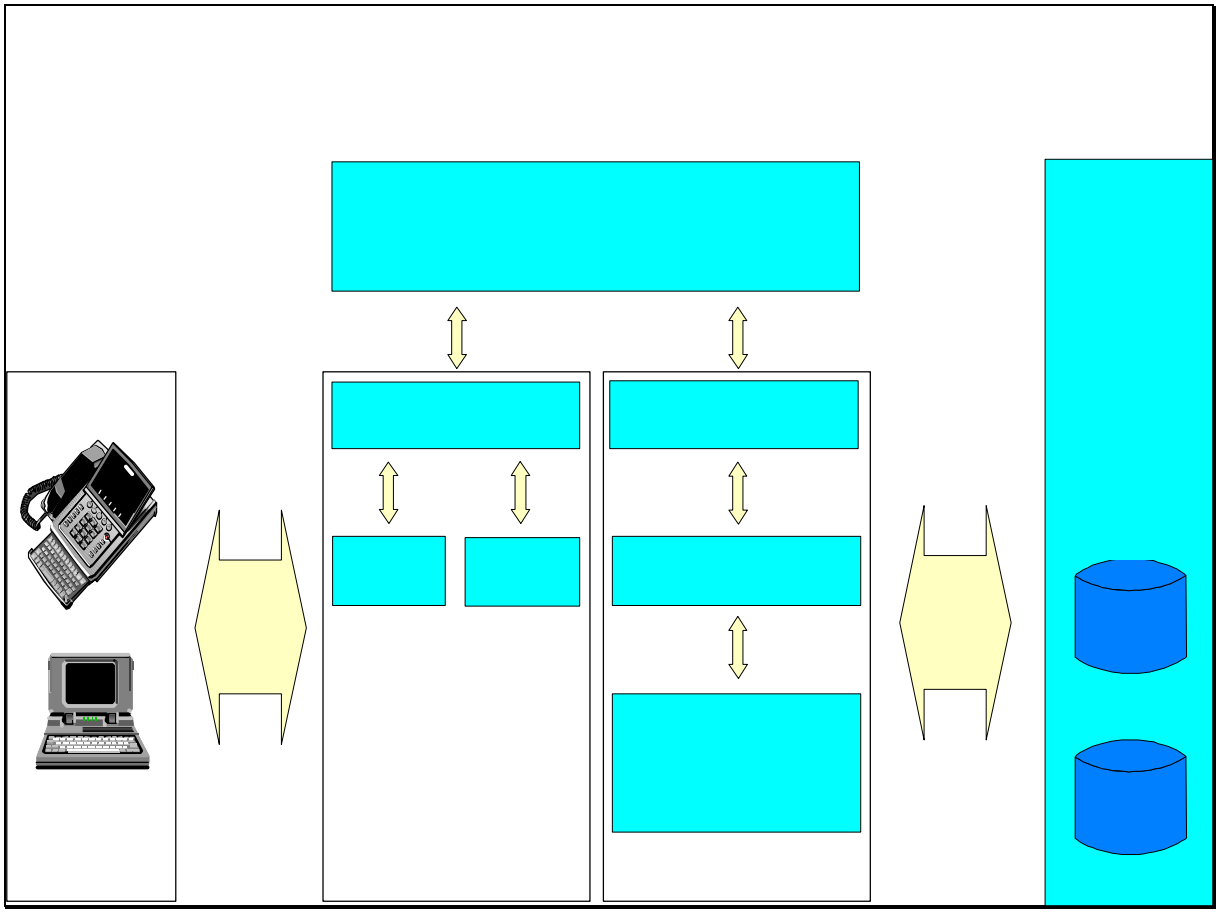

Abb. 11: Architketur der Mobile Notes Serverkomponente

4.4 Programmablauf

An dieser Stelle soll ein kurzer Überblick über den Programmablauf sowie die Hauptroutine des Mobile Notes Serverprogramms gegeben werden. Beim Start der Mobile Notes Serverkomponente werden zuerst die Startparameter des Programms ausgewertet. Sollten die eventuell angegebenen Pfade nicht vorhanden sein, stoppt das Programm nach Ausgabe einer entsprecheden Fehlermeldung. Es folgt die Auswertung der Mobile Notes Initialisierungs-datenbank und die Modemeinrichtung, indem der entsprechende Modeminitialisierungsstring an den verwendeten seriellen Anschluß geschickt wird.
Nun wird in die AVASProg_GetCall-Routine verzweigt, die bis zum Programmende nicht mehr verlassen wird. Aufgabe dieses Programmoduls ist der Aufruf der entsprechenden Programmfunktionen in Abhängigkeit des von der Clientanwendung durch die Übertragung des entsprechenden Datenstrings verlangte Aktion und dem Status des Benutzers. So stehen dem Anwender vor seiner Anmeldung im Mobile Notes System nur die Funktionen zur Verfügung, die für seinen Anmeldevorgang benötigt werden. Erst nach erfolgreicher Anmeldung kann der Benutzer die gesamten Funktionalitäten des Mobile Notes Systems nutzen, sofern ihn seine Zugriffsrechte dieses erlauben. Da für jede vom Mobile Notes Client durch die Übertragung des entsprechenden Datenübertragungsstrings veranlaßte Aktion in eine entsprechende AVAS_Prog... Funktion verzweigt wird, ist eine Erweiterung der bestehenden Mobile Notes Serverkomponente ohne Probleme, durch die Einfügung weitere Funktionsaufrufe an der entsprechenden Stelle der AVASProg_GetCall-Routine möglich.

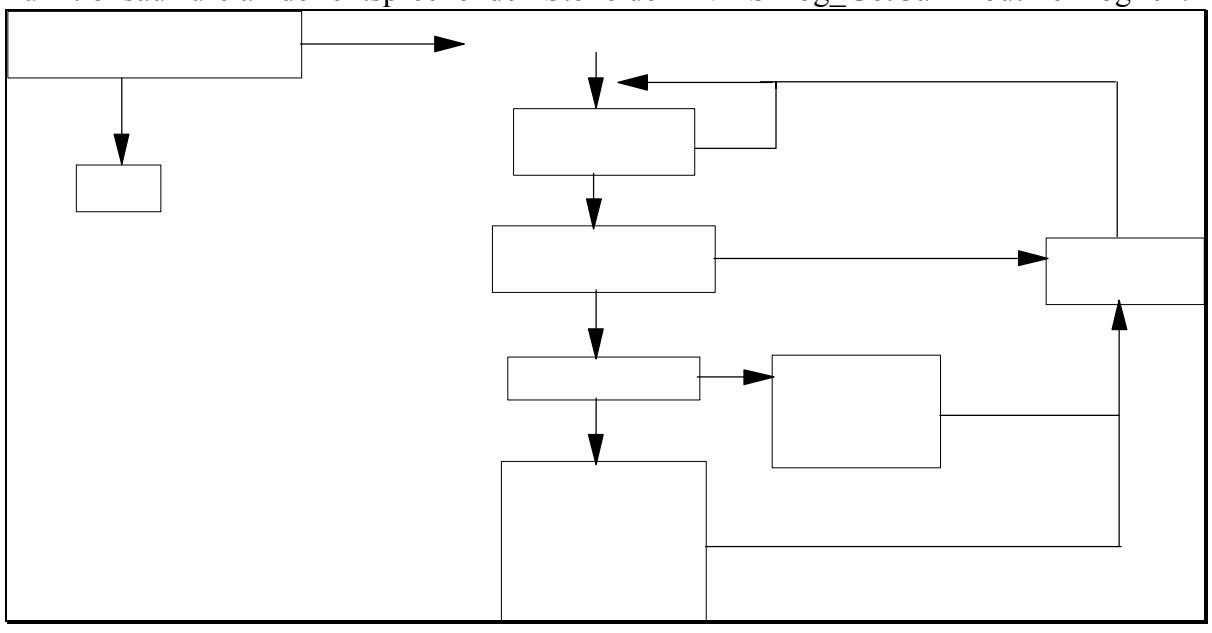

Abb. 12: Skizze: Programmablauf

4.5 Das Mobile Notes Datenübertragungsprotokoll

Das im folgenden Teil dieses Kapitels vorgestellte Datenübertragungsprotokoll stellt den Kernpunkt dieser Arbeit dar, da mit ihm die Mobile Notes Serverkomponente angesprochen und somit auch der Zugriff auf Lotus Notes Datenbanken realisiert wird.

4.5.1 Allgemeiner Aufbau

Alle Datenübertragungsbefehle werden als Textstring zwischen der Mobile Notes Serverkomponente und dem Client übertragen. Der Datenübertragungsstring ist dabei wie folgt aufgebaut:

Standardaufbau eines Datenübertragungsstrings

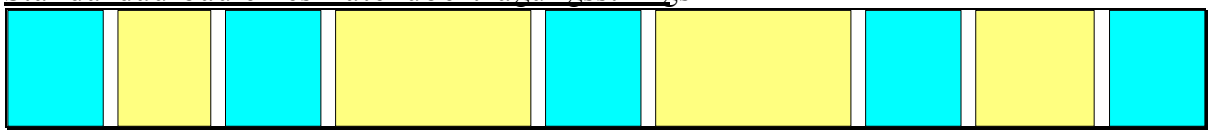

Abb. 13: Standardbefehlsaufbau

Bedeutung der Abkürzungen

SOH = ASCII-Zeichen mit dem Dezimalwert 1. Dieses Zeichen wird an den Anfang eines jeden zu übertragenden Datenblocks gestellt und zeigt an, daß ein neuer Datenblock übertragen wird.

Befehl = ASCII-Zeichen, dessen Wert einen Befehl der Datenübertragungsbefehle darstellt und die auszuführende Aktion bzw. das Senden bestimmter Daten anzeigt. Sämtliche Datenübertragungsbefehle werden als Zahlenwerte übertragen, um die Länge der des Befehlsstrings zu minimieren.[32]

GS = ASCII-Zeichen mit dem Dezimalwert 29. Dieses Zeichen dient der Trennung der Parameterlängen vom Datenteil bzw. vom Befehlskopf des zu übermittelnden Strings.

Parameterlänge = Länge in ASCII-Zeichen des x-ten Parameters im Datenteil des Strings. Je nach Befehl befindet sich eine unterschiedliche Anzahl von Parametern im Datenteil des zu übertragenden Strings. Für jeden Parameter erfolgt eine Längenangabe im Parameterteil des Strings. Jede einzelne Parameterlänge wird als unsigned short (2 Byte) übertragen, um den Datenteil entsprechend der einzelnen Parameter aufzuteilen.

[32] siehe Anhang Protokollbefehle und Konstanten

RS = ASCII-Zeichen mit dem Dezimalwert 30. Dieses Zeichen wird für die Trennung der einzelnen Parameterlängen verwendet.

Datenteil = In diesem Teil des zu übertragenden Strings befinden sich die Daten. Die Länge des Datenteils wird durch die Summe der einzelnen Parameterlängen bestimmt.

ETB = ASCII-Zeichen mit dem Dezimalwert 23. Dieses Zeichen wird an den Schluß eines jeden zu übertragenden Strings gestellt und zeigt dessen Ende an.

Jeder Datenübertragungsstring wird mit dem SOH-Zeichen gefolgt vom Befehl, der die auszuführende Aktion anzeigt bzw. die gesendeten Daten als Antwort auf eine zuvor durchgeführte Aktion kennzeichnet, eingeleitet. Im folgenden Teil werden je nach Befehl eine unterschiedliche Anzahl Parameterlängen durch ein RS-Zeichen getrennt, übertragen, die es ermöglichen, den anschließenden Datenblock wieder in die einzelnen Datensegmente zu zerlegen. Der Aufbau des Datenübertragungsstrings bietet folgende Vorteile:

- Der Datenübertragungsstring kann zusätzlichen Überprüfungen unterzogen werden kann. So muß z.B. die Länge des gesamten Datenblocks genauso groß sein, wie die Summe der einzelnen Paramterlängen.
- Mehrere Parameter bzw. Datensegmente können in einem Datenübertragungsstring zusammengefaßt werden und müssen nicht einzeln übertragen werden.
- Die Trennung des Datenteils vom Parameterlängenteil ermöglicht den Einsatz zusätzlicher Verschlüsselungsverfahren, die nur auf den Datenteil angewendet werden.

Die Verwendung der Sonderzeichen SOH, RS, GS und ETB dient außerdem der einfacheren Verarbeitung des Datenstrings sowie zusätzlichen Überprüfungen der Richtigkeit der übertragenen Daten, da ihre Position im Datenstring fest vorgeschrieben ist.
Die gesamte Kommunikation der Clientanwendung mit dem Mobile Notes Server ist dabei so aufgebaut, daß alle Befehle, die vom Client an den Server gesendet werden, (außer dem Befehl zur Beendigung der Kommunikation) entsprechend den Vorgaben des Mobile Notes Protokolls beantwortet werden. Auf diese Weise ist gewährleistet, daß der Client immer über die Durchführung bzw. aufgetretene Fehler bei der Ausführung der vom ihm angestoßenen Aktion informiert ist und kann entsprechend der erhaltenen Antwort weitere Aktionen ausführen.

Die folgende Tabelle gibt einen Überblick über die Datenübertragungsbefehle. Eine ausführliche Beschreibung der einzelnen Befehle befindet sich im Anhang dieser Arbeit.

Befehl	Parameter	Beschreibung
Allgemeine Befehle		
OP_ProtVersion	Client, Version	Übermittelt den Client und die Version zum Server
OP_Name	Name, Zufallszahl	Übermittelt den Benutzernamen zum Server
OP_Password	Passwort, Zufallszahl	Übermittelt Passwort
OP_Logoff	-	Abmelden beim Server
OP_ChangePassword	neues Passwort	Ändern des aktuellen Paßworts des Benutzers
OP_Answer	1. Anfragebefehl, 2. Antwort	
Datenbankbefehle		
OP_DBByUnid	Datenbank-ID	Positioniere den Server auf der DB mit der entsprechenden ID
OP_DBCount	-	Frage die Anzahl der für den Benutzer definierten Datenbanken ab
OP_DBGetTitle	-	Holt den Namen der aktuellen Datenbank ein
OP_DBCheckDbs	Liste von Datenbank-ID's, Last Edit Time's u . Zugriffsrecht	Liste der lokal vorhandenen DatenbankDefinitionen an den Server, zwecks Abgleich, schicken
OP_DBChangedDbs	Liste der Datenbank-ID's der geänderten Datenbanken	Antwort auf OP_DBCheckDbs, Liste der neuen, geänderten oder nicht mehr im Zugriff befindlichen Datenbanken
OP_DBGetDBDef	Datenbank-ID	Die durch die Datenbank-ID bestimmte Datenbankdefinition anfordern
OP_DBDef		Liefert Datenbankdefinition
Dokumentbefehle		
OP_DOCGetInfo	Länge einer Viewzeile	Anfordern der DokumentenID und der zugehörigen Viewzeile
OP_DOCGetView Lines	Länge einer Viewzeile Anzahl der Viewzeilen	Anfordern mehrerer Viewzeilen in einem Block ohne die dazugehörige DokumentenID um den Viewaufbau zu beschleunigen
OP_DOCFirst	-	Positioniert den Server auf dem ersten Dokument der aktuellen Datenbank
OP_DOCLast	-	Positioniert den Server auf dem letzten Dokument der aktuellen Datenbank
OP_DOCPrev	-	Positioniert den Server auf dem vorherigen Dokument der akt. Datenbank
OP_DOCNext	-	Positioniert den Server auf dem nächsten Dokument der akt. Datenbank
OP_DOCByUnid	Dokumenten-ID	Positioniert den Server auf dem Dokument mit der entsprechenden ID in der aktuellen Datenbank
OP_DOCRequest	-	Anfordern des gesamten aktuellen Dokumentes
OP_DOCContents	Felderinhalte(* Feldanzahl)	Liefert das gewünschte mit OP_DOCRequest angeforderten Dokumentes
OP_DOCUpdate	Felderinhalte(* Feldanzahl)	Modifiziert das aktuelle Dokument mit den übergebenen Feldinhalten
OP_DOCCreate	Felderinhalte(* Feldanzahl)	Erzeugt ein neues Dokument in der aktuellen Datenbank
OP_DOCDelete	-	Löscht das aktuelle Dokument

Tab. 4: Übersicht Mobile Notes Protokollbefehle

Anmerkung:

Sämtliche in dieser Tabelle aufgeführten Befehle werden durch Konstanten, die im Anhang dieser Arbeit dokumentiert werden, dargestellt.

4.5.2 Schutz vor fehlerhafter Datenübertragung

Ein wichtiger Punkt bei der Datenübertragung zwischen einer Clientanwendung und der Mobile Notes Serverkomponente ist die Überprüfung der übermittelten Daten auf bei der Datenübertragung entstandene Veränderungen des gesendeten Datenblocks.
Fehlerhaft übertragene Daten können ungewollte Verbindungsabbrüche oder Programmaktionen z.B. bei der Übertragung des Benutzernamens zu Beginn einer Kommunikation zwischen Client- und Serveranwendung verursachen. Außerdem könnten Dokumente mit fehlerhaft übertragenen Daten erstellt oder verändert werden, was z.B. im Bereich der mobilen Datenerfassung große Probleme und finanziellen Schaden erzeugen kann. Um eine fehlerhafte Datenübertragung möglichst auszuschließen, kann für die Datenübertragung des Mobile Notes Datenübertragungsprotokolls optional eine CRC-Überprüfung eingeschaltet werden, wodurch die übertragenen Daten auf eventuelle Übertragungsfeher überprüft werden.[33] Hierbei werden, wie schon im 3ten Kapitel beschrieben, zusätzlich zu den zu übertragenden Daten Prüfsummen übermittelt, die es der jeweiligen Gegenstelle ermöglichen, einen empfangenen Datenblock auf eventuell bei der Datenübermittlung entstandene Fehler zu überprüfen. Sollten nun Unregelmäßigkeiten festgestellt werden, wird eine erneute Datenübertragung von der Client- bzw. Serveranwendung veranlaßt.
Damit eine Clientanwendung die von der Mobile Notes Serverkomponente zur Verfügung gestellte CRC-Überprüfung verwenden kann, sollte für die Modemkommunikation die im Mobile Notes Projekt entwickelte Modemschnittstelle AVASCOM.C verwendet werden.
Die CRC-Überprüfung ist optional und kann vom Client zu Beginn einer Datenkommunikation mit dem Befehl OP_PROTVERSION und dem entsprechenden Parameter, der im Anhang dieser Arbeit dokumentiert ist, eingestellt werden, da bei weniger leistungsstarken Clientsystemen der Datentransfer zu sehr belastet werden könnte bzw. "Low Level"-Frontendsysteme das Verfahren nicht implementieren können.
Die CRC-Überprüfung wird bei der aktuellen Mobile Notes Version im AVASCOM.C-Modul durchgeführt. Das eigentliche Mobile Notes Hauptprogramm bearbeitet bei eingeschaltetem CRC-Modus somit nur den bereits überprüften Datenstring. Da auch eine CRC-Überpüfung keinen absoluten Schutz vor Datenübertragungsfehlern bietet, überprüft das Hauptprogramm zusätzlich anhand der Sonderzeichen SOH, GS, RS und ETB, die sich an den vorgeschriebenen Stellen des Datenübertragungsstrings befinden müssen, den korrekten Aufbau des gesendeten Datenstrings. Diese Überprüfung wird bei jedem gesendeten Datenstring durchgeführt und bietet so auch bei nicht eingeschalteten CRC-Modus eine Überprüfung des Datenstrings, wobei jedoch nur solche Fehler entdeckt werden können, die im Parameterteil bzw. an den Stellen der vorgeschriebenen Sonderzeichen aufgetreten sind. Veränderungen im Datenteil des gesendeten Datenübertragungsstrings können an dieser Stelle nicht ermittelt werden.

---

[33] Siehe hierzu Kapitel 3.3.1 "Schutz der Datenübertragung"

4.5.3 Schutz der zu übertragenden Daten

Neben der Sicherung der Datenübertragung vor Übertragungsfehlern ist der Schutz der zu übermittelnden Daten vor Spionage durch Dritte von besonderer Bedeutung. Vor allem für Wirtschaftsunternehmen, die zu den Hauptanwendern von Client-Server-Architekturen zählen, besitzen die in solchen Systemen verwalteten und übermittelten Daten einen besonderen Wert, deren Verlust hohen wirtschaftlichen Schaden für die Unternehmen bedeuten kann. Die gesendeten Daten müssen daher einer Verschlüsselung unterzogen werden, um sie für potentielle Spione, die die Telefonleitung abhören könnten, wertlos zu machen. Zu diesem Zweck wird für das Mobile Notes Datenübertragungsprotokoll die DES-Verschlüsselung verwendet. Hierbei wird der gesamte zu sendende String, bevor er durch das AVASCOM.C-Modul an das Modem weitergeleitet wird, mit dem DES-Schlüssel des jeweiligen Benutzers verschlüsselt. Eine Ausnahme bilden die Befehle OP_PROTVERSION und OP_NAME die zu Beginn eines Kommunikationsaufbaus vom Client an den Server geschickt werden müssen. Durch Befehl OP_PROTVERSION und dem entsprechenden Parameter muß neben der CRC-Überprüfung auch die Verschlüsselung eingeschaltet werden. Auf diese Weise ist gewährleistet, daß auch Clientanwendungen, die z.B. aus Geschwindigkeitsgründen keine DES-Verschlüsselung bzw. CRC-Überprüfung durchführen, die generische Mobile Notes Serverkomponente benutzen können. Der Befehl OP_NAME wiederum dient der Identifikation des Benutzers der Clientanwendung. Anhand des mit OP_NAME übermittelten Benutzernamens, der jeweils nur einmal in der Mobile Notes Initialisierungsdatenbank existieren kann, wird der für den jeweiligen Benutzer zu verwendende DES-Schlüssel, der sich in der Mobile Notes Initialisierungsdatenbank befindet, ermittelt.
Falls die DES-Verschlüsselung zuvor durch OP_PROTVERSION-Befehl eingeschaltet wurde, wird nun jeder Datenübertragungsstring mit dem DES-Schlüssel ver- bzw. entschlüsselt. Jedoch soll an dieser Stelle darauf hingewiesen werden, daß ohne Verschlüsselung sämtliche Daten, auch das Passwort eines Benutzers, als Klartext zwischen der Client und der Serverkomponente übertragen wird, was das Abhören der Übermittelten Daten sehr vereinfacht.
Die Ent- bzw. Verschlüsselung wird wie die CRC-Überprüfung in dem AVASCOM.C-Modul vorgenommen, das von dem Studenten Nico Dirks, als Komponente des Mobile Notes-Systems entwickelt wurde.[34] Das eigentliche Hauptmodul AVASPROG.C verarbeitet anschließend den vom AVASCOM.C-Modul zur Verfügung gestellten String. Auf diese Weise ist sichergestellt, daß zukünftige Erweiterungen bzw. Veränderungen in der Datenübertragung ohne großen Programmieraufwand realisiert werden können, indem das entsprechende AVASCOM.C Modul durch eine neue Komponente ersetzt wird. Die folgende Graphik gibt einen Überblick über die in der aktuellen Mobile Notes Version verwendeten Sicherheitsmechanismen:

---

[34] siehe die in Kapitel 1 erwähnte Diplomarbeit des Autors Nico Dirks

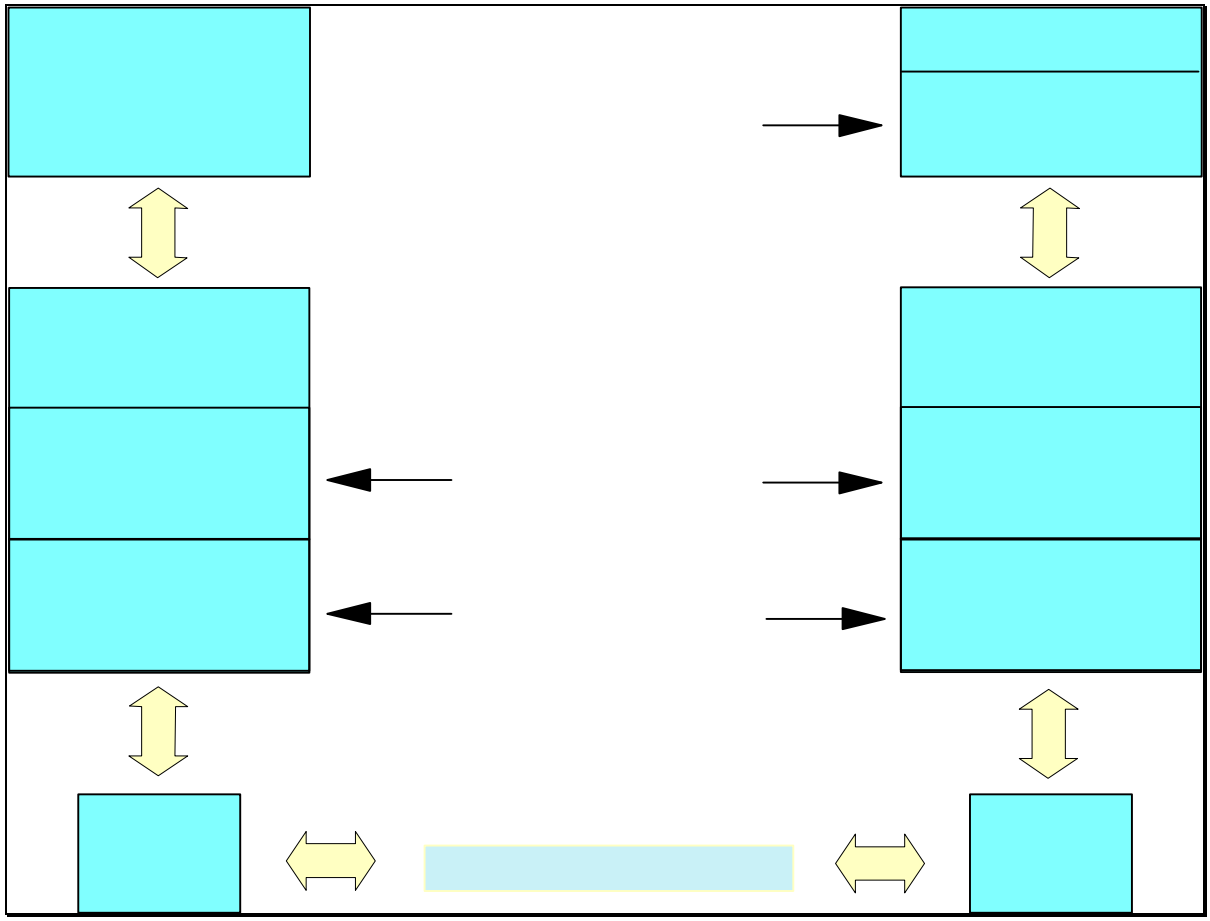

Abb. 14: Überblick über die Mobile Notes Sicherheitsmechanismen

4.6 Praktisches Beispiel

4.6.1 Der Anmeldevorgang

Der Verbindungsaufbau mit dem Mobile Notes Server erfolgt mit dem Wählen der Telefonnummer des Modems, an das der Mobile Notes Server angeschlossen ist. Ist der Kontakt zwischen dem Modem des Servers und des Clients hergestellt, muß zu Beginn der eigentlichen Kommunikation der Befehl OP_PROTVERSION durch das Frontend übertragen werden, der durch seine Parameter der Serveranwendung mitteilt, mit welchem Frontendtyp er gerade verbunden ist und mit welchen Modi die Datenübertragung erfolgen soll. Zur Zeit sind folgende Übertragungsmöglichkeiten vorgesehen:

- Standardmäßig erfolgt jede Kommunikation nach der Anmeldung des Users DES-verschlüsselt. Jedoch kann diese Verschlüsselung auch ausgeschaltet werden, was z.B. bei Entwicklung von Frontendapplikationen durchaus sinnvoll ist aber im späteren Einsatz vermieden werden sollte.
- Wahlweise kann die Datenübertragung mit bzw. ohne eine CRC-Überprüfung der zu übertragenden Daten erfolgen. Sollte die CRC-Prüfung einen Fehler bei der Datenübertragung ermitteln, so wird die Übertragung des entsprechenden Datenblocks wiederholt.

Zusätzlich zu den Übertragungsmöglichkeiten wird mit dem Befehl OP_PROTVERSION auch der Clienttyp übermittelt, um für zukünftige Weiterentwicklungen der Serverapplikation eine optimale Anpassung an das jeweilige Frontend zu ermöglichen.
Ist der Befehl OP_PROTVERSION korrekt übermittelt worden, sendet die Serveranwendung den unverschlüsselten Antwortbefehl OP_ANSWER mit dem Parameter AVA_GEN_OK an den Client zurück. Sollten der Befehl OP_PROTVERSION falsch parameterisiert oder der Befehlsstring falsch aufgebaut sein, so wird statt der Konstante AVA_GEN_OK eine entsprechende Fehlerkonstante zurückgeliefert.

Nun erfolgt die eigentliche Anmeldeprozedur des Clients beim Server:

Zuerst wird mit dem Befehl OP_NAME der Name des Benutzers, sowie eine durch den Client generierte und mit dem DES-Algorithmus verschlüsselte Zufallszahl als Parameter an den Mobile Notes Server gesendet. Dieser überprüft, ob der Name des Benutzers in der Mobile Notes Initialisierungsdatenbank eingetragen ist und ließt, wenn der Name gefunden wurde, den persönlichen DES-Schlüssel des Benutzers aus, der für die Entschlüsselung der zusätzlich übertragenen Zufallszahl dient. Der Frontendapplikation wird nun mit dem Befehl OP_ANSWER die vom Client gesendete Zufallszahl um den Wert 1 erhöht sowie eine vom Server selbst erstellte Zufallszahl übermittelt, wobei der gesamte Datenübertragungsstring mit dem DES-Algorithmus verschlüsselt ist. Sollte der Benutzer nicht in der Mobile Notes Initialisierungsdatenbank angemeldet sein, oder ist aus anderen Gründen der Name des

Benutzers falsch angegeben worden, wird statt der Zufallszahlen die Konstante AVA_GEN_FALSE unverschlüsselt übertragen und der Benutzer kann mit dem Befehl OP_NAME erneut versuchen, sich beim Mobile Notes Server anzumelden. Wird hierbei eine bestimmte Anzahl Fehlversuche, welche in der Mobile Notes Initialisierungsdatenbank definiert werden kann, überschritten, unterbricht der Mobile Notes Server die Verbindung.

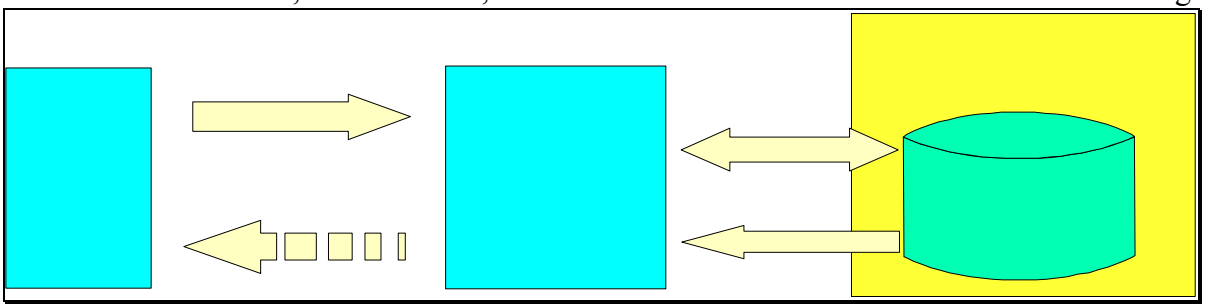

Abb. 15: Benutzeranmeldung a

Ist der Name des Benutzers von der Mobile Notes Serveranwendung akzeptiert worden, muß nun sein Paßwort mit dem Befehl OP_PASSWORD an die Serveranwendung geschickt werden. Zusätzlich wird außerdem die vom Mobile Notes Server gesendete Zufallszahl um den Wert 1 erhöht zum Server geschickt. So wird sichergestellt, daß keine Aufzeichnung eines bei einer früheren Anmeldung übertragenen Datenstrings akzeptiert wird (Playback), da durch die Zufallszahl jeder Datenübertragungsstring nahezu einmalig ist. Stellt der Mobile Notes Server fest, daß die Zufallszahl-1 mit der von ihm generierten Zahl übereinstimmt und ist das gesendete Paßwort des Benutzers mit dem in der Mobile Notes Initialisierungsdatenbank identisch, ist der Benutzer am Mobile Notes Server angemeldet, was dem Client durch den Antwortbefehl OP_ANSWER mit dem Parameter AVA_GEN_OK (DES-verschlüsselt) mitgeteilt wird. Im Fehlerfall sendet der Mobile Notes Server ein statt dem AVA_GEN_OK ein AVA_GEN_FALSE. Der Client kann nun erneut versuchen, mit dem Befehl OP_PASSWORD das korrekte Paßwort an den Server zu senden. Sollte hierbei eine in der Mobile Notes Initialisierungsdatenbank definierte Anzahl überschritten werden, unterbricht die Serveranwendung die Verbindung. Außerdem wird nun eine Warnmail an den Systemadministrator und den Benutzer selbst geschickt, da ein nicht berechtigter Dritter versucht haben könnte unter dem Namen eines angemeldeten Benutzers Zugang zum Mobile Notes System und somit zu Lotus Notes Datenbanken zu erlangen.

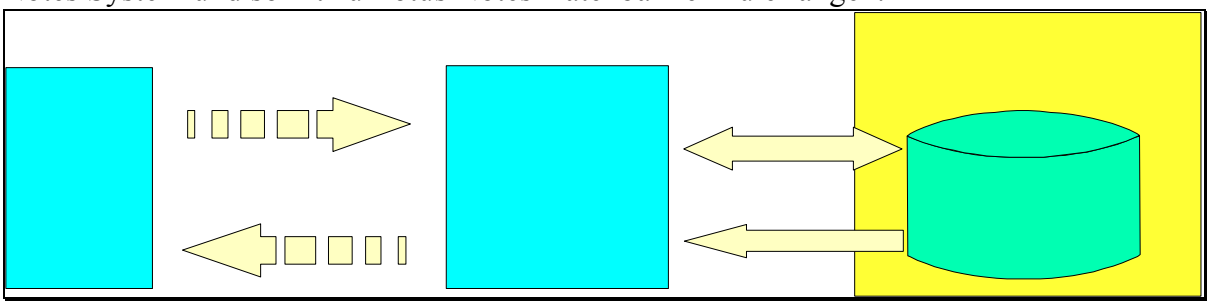

Abb. 16: Benutzeranmeldung b

4.6.2 Abgleich der Mobile Notes Datenbankdefinitionen

Um auf Datenbanken des Mobile Notes Systems zuzugreifen, sollte die Clientanwendung über die aktuelle Mobile Notes Datenbankdefinition verfügen, da die Übermittlung der Dokumente auf der aktuellen Datenbankdefiniton der Mobile Notes Serverkomponente basiert. Außerdem könnten sich Zugriffsrechte des Benutzers geändert haben oder Datenbanken nicht mehr im Zugriff des Benutzers liegen. Wird dann trotzdem versucht, auf eine Datenbank zuzugreifen, obwohl sie z.B. nicht mehr im Zugriff des Benutzers ist, liefert die Mobile Notes Serverkomponente mit dem Befehl OP_ANSWER einen entsprechenden Fehlerwert an den Client zurück. Veränderungen in einer Datenbankdefinition können außerdem zur Folge haben, daß Dokumente, die nach der alten Datenbankdefinition auf dem Client erstellt und dort gespeichert worden sind, nicht mehr korrekt in der entsprechenden Lotus Notes Datenbank gespeichert werden können. Änderungen einer Datenbankdefinition sollten daher nur nach rechtzeitiger Benachrichtigung der betroffenen User vorgenommen werden, damit wichtige Dokumente vor der Änderung in der entsprechenden Lotus Notes Datenbank gespeichert werden können.
Ist der Benutzer durch die Übertragung seines Namens und des Paßwortes erfolgreich beim Mobile Notes Server angemeldet, sollte nun zuerst überprüft werden, ob sich seit der letzten Kommunikation des Clients mit dem Server,

- Änderungen in den Datenbankdefinitionen der für den Benutzer freigeschalteten Datenbanken ergeben haben.
- neue Datenbanken für den Benutzer eingerichtet worden sind.
- Datenbanken aus dem Zugriff des Benutzer entfernt wurden.
- sich Änderungenin den Datenbankzugriffsrechten des Users ergeben haben.

Hierzu wird vom Client mit dem Befehl OP_DBCHECKDBS eine Liste aller Mobile Notes Datenbank-ID's, die sich auf dem Client befinden mit dem jeweiligen Änderungsdatum einer jeden Datenbank-ID und dem Datenbankzugriffsrecht des Users zum Mobile Notes Server geschickt. Mit der Mobile Notes Datenbank-ID wird die Unic Dokument-ID eines Datenbankinitialisierungsdokumentes bezeichnet, die in der Mobile Notes Initialisierungs-datenbank für jede eingerichtete Datenbank einmalig ist.
Sollten sich Datenbankdefinition bzw. Datenbankzugriffsrechte des Benutzers geändert haben oder sind neue Datenbanken für den User eingerichtet bzw. Datenbanken aus dem Zugriff des Users entfernt worden, werden der Clientanwendung mit dem Befehl OP_CHANGEDDBS die entsprechenden Datenbank-ID's übermittelt. Kommuniziert die Clientanwendung zum ersten mal mit der Mobile Notes Serveranwendung, so werden alle Datenbank-ID's, auf die der Benutzer Zugriff hat, dem Client übertragen.
Die Clientanwendung kann nun mit dem Befehl OP_DBGETDBDEF und der jeweiligen Mobile Notes Datenbank-ID eine aktuelle oder neue Datenbankdefinition von der Mobile Notes Serveranwendung anfordern.

Die Mobile Notes Datenbankdefinition

Die Mobile Notes Datenbankdefinition wird benötigt, um mit dem Frontendsystem die Mobile Notes Serverkomponente zu benutzen. Sie enthält alle Daten, damit auch ohne Verbindung zum Mobile Notes Server (offline) z.B. Dokumente für eine Datenbank erzeugt werden können, um sie dann später mit dem Mobile Notes Server abzugleichen oder um gespeicherte Dokumente darzustellen. Eine Mobile Notes Datenbankdefinition enthält folgende Daten:

1. Datenbank-ID: Die Datenbank-ID besteht aus der Unic Document ID des Datenbank-initialisierungsdokumentes. Sie dient der eindeutigen Identifizierung einer Datenbank.[35]
2. Kontrollzeit: Die Kontrollzeit besteht aus dem Datum und der Zeit der letzten Änderung des Datenbankinitialisierungsdokumentes (z.B. 20.08.95 11:23:17). Anhand dieser Zeit erkennt die Mobile Notes Serverkomponente, ob die auf dem Client vorhandene Datenbankdefinition noch aktuell ist.
3. Datenbankname: Der Datenbankname ist für die Ausgabe auf dem Frontend gedacht, da die Datenbank-ID keine Aussagekraft besitzt.
4. Datenbankview: Der Datenbankview ist, wie der Datenbankname, nur für die Ausgabe auf dem Client bestimmt.
5. Zugriffstyp: Durch den Zugriffstyp werden die Rechte, die ein Benutzer über eine Datenbank hat, definiert. In der jetzigen Mobile Notes Version werden die Zugriffsrechte durch folgende ASCII-Werte dargestellt: 1 = Read Access; 2 = Write Access; 3 = Read and Write Access; 4 = Read, Write and Update Access.
6. Anzahl definierter Felder: In der augenblicklich vorliegenden Mobile Notes Version können maximal 10 Felder pro Datenbank definiert werden.

Eine Felddefinition wird für jedes der eingerichteten Felder der Datenbank angelegt und bei der Anforderung einer Datenbankdefinition an den Client übertragen. Eine Felddefinition enthält folgende Daten:

Felddefinition:

7. Feldnummer: Die Feldnummer dient der Identifikation des Feldes und kann Werte zwischen 1 und 10 annehmen.
8. Aliasname des Feldes: Der Aliasname dient der Darstellung des Feldes auf dem Frontendsystem.
9. Feldtyp: In der vorliegenden Mobile Notes Version werden die Feldtypen Text, Number und Datum unterstützt.
10. Feldgröße: Die Feldgröße bezeichnet die maximale Größe des Feldes in Byte.

---

[35] siehe Kapitel 4.7.2 Datenbankdefinitionsdokument

11. Vererbungsfeldnummer: Die Vererbungsfeldnummer gibt an, aus welchem Feld bei der Erstellung eines Antwortdokumentes (z.B. Mail) Daten übernommen werden sollen.
12. ReplyMode: Der Replymode ist in der vorliegenden Mobile Notes Version noch nicht integriert. Für zukünftige Erweiterungen können hier die verschiedenen Möglichkeiten wie die Daten aus einem anderen Feld übernommen werden, angegeben werden.
13. Viewfeldnummer: Die Viewfeldnummer gibt an, ob das Feld für die Viewdarstellung eines Dokumentes verwendet wird und an welcher Position das Feld bzw. Teil im View erscheint (0 falls kein Viewfeld).
14. Anzahl Zeichen für die Viewdarstellung: Falls das Feld durch die Viewfeldnummer als ein Viewfeld definiert ist, wird hiermit die Anzahl der Zeichen, die von diesem Feld im View dargestellt werden sollen, angegeben. Ist das Feld länger, so werden nur die ersten Zeichen bis zur Anzahl der definierten Viewfeldzeichen des Feldes dargestellt. Sollte das Feld weniger Zeichen enthalten, so werden die fehlenden Zeichen durch Blanks (ASCII-Wert 0) ersetzt.

4.6.3 Navigation im Notessystem

Für die Navigation im Notessystem stellt die Mobile Notes Serverkomponente diverse Befehle zur Verfügung, die die Auswahl von Datenbanken und Dokumenten erlauben. Um auf eine Datenbank zugreifen zu können, muß diese zuvor für den Benutzer freigeschaltet werden. Die Freischaltung einer Datenbank kann auf zwei Arten erfolgen:

- Die Datenbank wird im Benutzerprofildokument der Mobile Notes Initialisierungsdatenbank eingetragen. Neben der persönlichen Maildatenbank, die nur im Benutzerprofildokument Initialisierungsdatenbank freigeschaltet werden kann, können zusätzlich für jeden Benutzer Datenbanken in eine Datenbankliste eingetragen werden[36].
- Der Benutzer ist in einer Benutzergruppe eingetragen. Für jede Benutzergruppe können in der Mobile Notes Initialisierungsdatenbank Datenbanken freigeschaltet werden. Dies ist z.B. für Projektarbeiten sinnvoll, um eine Projektdatenbank für die Dauer eines Projektes nicht in mehreren Benutzerprofildokumenten zu speichern[37].

Für bestimmte Aktionen wie z.B. die Erstellung eines Dokumentes muß der Benutzer außerdem über das entsprechende Zugriffsrecht für diese Datenbank verfügen. Im Mobile Notes System werden folgende vier Zugriffsrechte unterschieden:

- Nur lesen von Dokumenten (Read Access)
- Lesen und erzeugen von Dokumenten (Read and Write Access)
- Nur erzeugen von Dokumenten (Write Access)
- Lesen, erzeugen und ändern von Dokumenten (Read , Write and Update Access)

---

[36] siehe Kapitel 4.7.1 Das Benutzerprofildokument

[37] siehe Kapitel 4.7.3 Das Benutzergruppendokument

Der Zugriff auf Lotus Notes Datenbanken wird über die Mobile Notes Datenbank-Id realisiert. Die Clientanwendung schickt die ID der Datenbank, die der Benutzer bearbeiten möchte, mit dem Befehl OP_DBBYUNID zum Mobile Notes Server, der daraufhin versucht, die Datenbank zu öffnen. Sollte die Datenbank, z.B. weil sie sich nicht mehr im Datenbankinitialisierungsdokument angegebenen Pfad befindet oder der Benutzer keinen Zugriff mehr auf die Datenbank besitzt, nicht geöffnet werden können, wird dies dem Client mit dem Befehl OP_ANSWER und einem entsprechenden Fehlerwert gemeldet. War das Öffnen erfolgreich, wird statt einem Fehlerwert die Konstante AVA_GEN_OK zum Client geschickt.

Befindet sich der Benutzer in einer Datenbank, so kann er nun die einzelnen Dokumente dieser Datenbank auswählen wozu ihm die Befehle OP_DOCFIRST, OP_DOCLAST, OP_DOCNEXT, OP_DOCPREV und OP_DOCBYUNID zur Verfügung stehen. Weitere Erläuterungen zu diesen Befehlen können dem Anhang "Protokollbefehle und Konstanten" entnommen werden.

4.6.4 Aktionen im Notessystem

In Abhängigkeit von den Zugriffsrechten des Benutzers können nun, nach der erfolgreichen Navigation auf die entsprechende Lotus Notes Datenbank bzw. auf das entsprechende Dokument, folgende Aktionen ausgeführt:

- Abfrage des für die Datenbank definierten Views
- Anfordern eines Dokumentes aus der Datenbank, um es auf dem Client darzustellen bzw. dort zu speichern
- Erzeugen neuer Dokumente
- Verändern bereits bestehender Dokumente

Auf eine weiterführende, genauere Erklärung der hierfür zu verwendenden Befehle und der Datenübertragungsstrings wird an dieser Stelle verzichtet, weil sie den Rahmen dieser Arbeit überschreiten würde. Die für die entsprechenden Aktionen verwendeten Befehle und deren Aufbau werden im Anhang "Protokollbefehle und Konstanten" dieser Arbeit ausführlich erläutert.

Zusätzlich zu den hier aufgeführten Bearbeitungsmöglichkeiten, kann ein Benutzer, wenn er im Mobile Notes System angemeldet ist, mit dem Befehl OP_CHANGEPASSWORD jederzeit sein Paßwort ändern, was ebenfalls im Anhang "Protokollbefehle und Konstanten" dokumentiert wird.

4.7 Die Mobile Notes Initialisierungsdatenbank

Für die Steuerung der Serveranwendung ist die Mobile Notes Initialisierungsdatenbank entwickelt worden, welche Nutzerprofile, Datenbankinitialisierungen, Initialisierungsdaten für die Modemschnittstellen, mögliche Usergruppen sowie ein Administratordokument enthält. Die in der Datenbank gespeicherten Daten werden beim Start der Mobile Notes Serverkomponente bzw. bei Bedarf (Useranmeldung) ausgelesen. Es ist daher sinnvoll, daß sich die Mobile Notes Datenbank ebenfalls auf dem Mobile Notes Server befindet, um Zeitverluste durch Zugriffe auf einen anderen sich im Netz befindlichen Computer zu vermeiden.
Dadurch, daß sämtliche Initialisierungs- und Verwaltungsdaten der Mobile Notes Serverkomponente in einer Notesdatenbank gehalten werden, ergeben sich folgende Vorteile:

- Die Wartung und Einrichtung der Mobile Notes Serverkomponente kann in der bekannten Lotus Notes Umgebung erfolgen. Die Datenbank fungiert nach dem Programmstart als primäre Schnittstelle zwischen dem Systemadministrator und dem Mobile Notes System. Sowohl neue Benutzer, Datenbanken als auch Usergruppen können in der Mobile Notes Datenbank verwaltet werden.
- Die Sicherheitsmechanismen der Notes Umgebung können genutzt werden. Dies ist besonders deshalb von Bedeutung, da die Mobile Notes Initialisierungsdatenbank auch sensible Daten wie Benutzerpaßwörter oder den DES-Schlüssel eines jeden Benutzers enthält.
- Möglichkeiten der Datenbankgestaltung und Anpassung an bestehende Vorgaben (Corporate design etc.)

Im folgenden werden nun die einzelnen Dokumente, die sich in der Mobile Notes Initialisierungsdatenbank befinden, anhand von Screenshots kurz vorgestellt.

4.7.1 Benutzerprofil

Für jeden Benutzer, der das Mobile Notessystem nutzen möchte, muß ein Benutzerprofil in der Mobile Notes Initialisierungsdatenbank eingerichtet werden. Folgende Daten werden im Benutzerprofil gespeichert:

- Der Benutzername, der einmalig sein muß, da mit dem Namen das entsprechende Personendokument des Benutzers identifiziert wird. Sollte ein Benutzername schon einmal vorhanden sein, so wird dies automatisch erkannt und das Personendokument kann nicht abgespeichert werden.
- Das persönliche Passwort eines Benutzers. Durch das Passwort authentisiert sich der Benutzer gegenüber dem Mobile Notes System.

- Der persönliche DES-Schlüssel eines Benutzers. Der DES-Schlüssel dient zur Sicherung der Datenübertragung zwischen dem Mobile Notes Server und dem Mobile Notes Client. Er wird automatisch durch Lotus Notes erstellt und besteht aus acht Dezimalzahlen zwischen 0 und 255. Da Lotus Notes 3.3 keine Befehle zur Generierung von Zufallszahlen bereitstellt, wurde hierfür auf einen externen DLL-Befehl zurückgegriffen, welcher in die bestehenden Lotus Notes Befehle eingebunden werden kann. Der hier generierte DES-Schlüssel muß auch dem Benutzer der Clientanwendung mitgeteilt werden, da diese den Schlüssel ebenfalls für die Ent- und Verschlüsselung der Datenübertragungsstrings benötigt. Der DES-Schlüssel kann aus Sicherheitsgründen nicht durch die Mobile Notes Serveranwendung übertragen werden, da eine solche Übertragung in der aktuellen Mobile Notes Version nur unverschlüsselt erfolgen könnte (Die Clientanwendung besitzt noch keinen DES-Schlüssel für eine verschlüsselte Übertragung). Dem Anwender muß deshalb auf anderem Wege wie z.B. durch eine Mail oder per Post sein persönlicher DES-Schlüssel mitgeteilt werden.

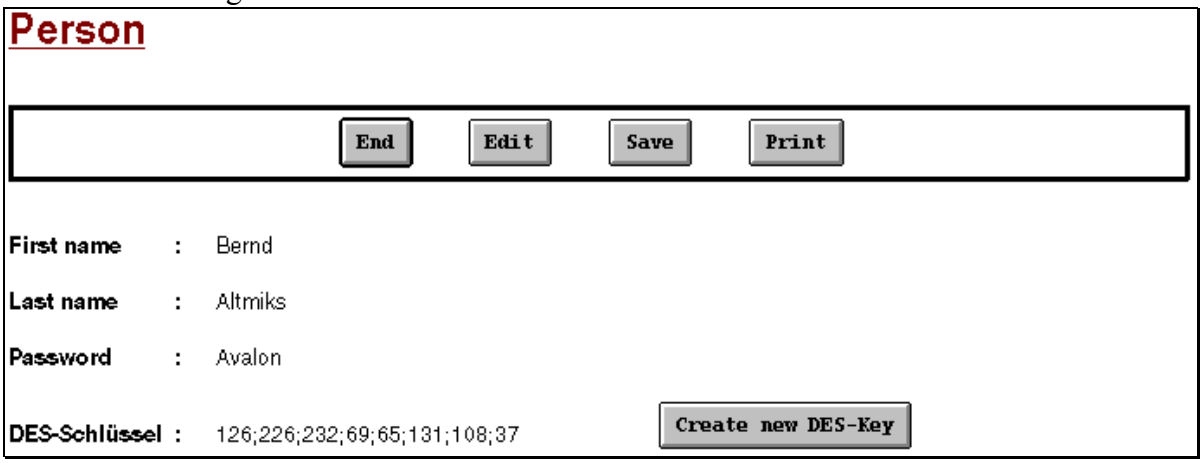

Abb. 17: Personendokument a

- Die persönliche Maildatenbank des Benutzers. Für die Einrichtung der Maildatenbank steht dem Administrator durch betätigen des "Add"-Knopfes in der Maildatenbankrubrik eine Auswahlbox zur Verfügung, in der alle Maildatenbanken, die im Mobile Notes System eingerichtet sind, erscheinen. Ist die Maildatenbank des Benutzers bereits im Mobile Notes System eingerichtet[38], so kann sie an dieser Stelle ausgewählt werden. Andernfalls muß der Systemadministrator zuerst die Maildatenbank einrichten, bevor er sie für den Mobile Notes Nutzer freischalten kann. Der Benutzer hat auf diese Datenbank immer "Read, Write and Update" Zugriff.

- Eine Liste von Datenbanken, die individuell für jeden Benutzer zusammengestellt werden kann. Zusätzlich wird für jede Datenbank das Zugriffsrecht des Benutzers definiert. Die hier eingerichteten Datenbanken müssen wie bereits die Maildatenbank eines Benutzers zuvor in der Mobile Notes Initialisierungsdatenbank definiert worden sein, um in der Auswahlbox, die durch drücken des "Add"-Knopfes in der entsprechenden Rubrik geöffnet

[38] siehe Kapitel 4.7.2 das Datenbankdefinitionsdokument

wird, zu erscheinen. Für jede Datenbank kann nur ein Zugriffsrecht vergeben werden, das mit dem "Change"-Knopf auch geändert werden kann. Mit dem "Delete"-Knopf können zuvor für den Benutzer freigeschaltete Datenbanken aus seinen Zugriff entfernt werden.

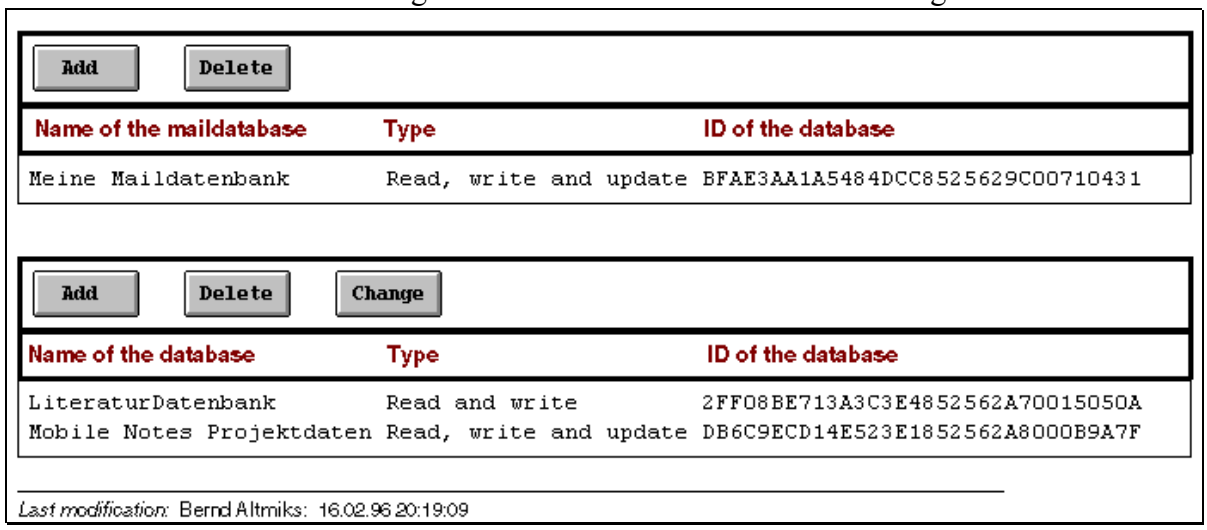

Abb. 18: Personendokument b

4.7.2 Datenbankdefinitionsdokument

Jede Datenbank, auf welche durch das Mobile Notes System zugegriffen wird, muß in der Mobile Notes Datenbank definiert werden. Für eine Datenbank können mehrere Datenbankdefinitionen eingerichtet werden, um z.B. eine englische und eine deutsche Feldbezeichnung für eine Datenbank zu realisieren. Eine Datenbankinitialisierung besteht aus folgenden Daten:

- Dem Namen der Datenbank, der nur für die Darstellung auf dem jeweiligen Frontend dient und für jede Datenbank individuell vergeben werden kann.
- Dem Pfad und Namen der Lotus Notes Datenbank, auf die zugegriffen wird.
- Die Mobile Notes Datenbank-ID, die aus der Unic Document ID des Datenbank-initialisierungsdokumentes besteht und automatisch von Lotus Notes berechnet wird. Hierdurch ist gewährleistet, daß die Mobile Notes Datenbank-ID einmalig in der Mobile Notes Initialisierungsdatenbank und somit einmalig im Mobile Notes System ist.
- Dem Typ der Datenbank. Bei einer Datenbankdefinition wird zwischen einer Maildatenbank und einer "normalen" Datenbank unterschieden. Wird eine Maildatenbank definiert, so werden der View und Felddefinitionen mit den Werten für eine Standard Notes Maildatenbank vorbelegt, um den Definitionsaufwand zu minimieren. Die Standardwerte können jedoch jederzeit abgeändert werden.
- Einem Kommentarfeld, daß für Anmerkungen des Systemadministrators vorbehalten ist.

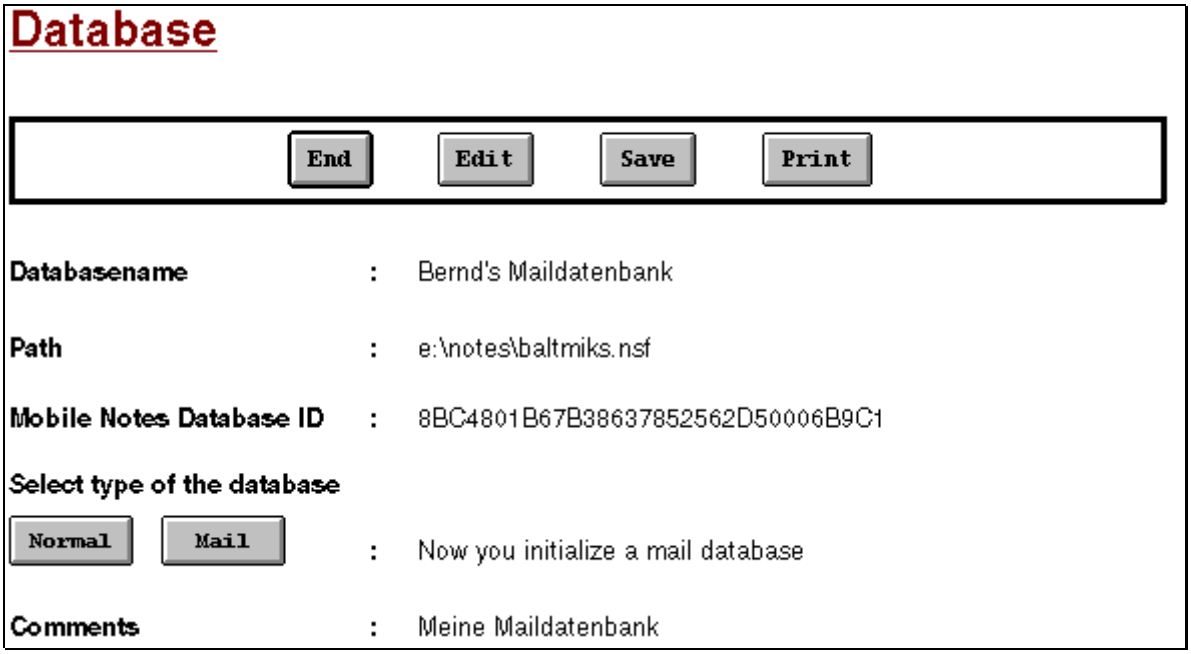

Abb. 19: Datenbankdokument a

Im Zweiten Teil eines Datenbankdefinitionsdokumentes werden der View, die für die Erstellung von Dokumenten zu verwendende Form sowie bis zu 10 Datenbankfelder definiert:

- Die Viewdefinition setzt sich aus dem Originalview der Datenbank sowie dem Namen, der für den View auf dem jeweiligen Frontend dargestellt wird, zusammen.
- Die zu verwendende Form für die Erstellung von Dokumenten in der original Notes Datenbank.
- Im letzten Abschnitt können bis zu zehn Felder der Original Notes Datenbank für das Mobile Notes System eingerichtet werden. Eine Felddefinition setzt sich aus folgenden Daten zusammen:
  - Der Feldnummer und dem Originalnamen des Feldes in der Lotus Notes Datenbank.
  - Dem anzuzeigenden Feldnamen auf dem jeweiligen Mobile Notes Frontend.
  - Der maximalen Größe des Feldes in Byte.
  - Dem Typ des Feldes, wobei in der jetzigen Mobile Notes Version die Feldtypen Time, Text und Number unterschieden werden.
  - Dem Vaterfeld des Feldes. Hiermit wird das Feld bezeichnet, aus dem bei der Erstellung eines Antwortdokumentes der Feldinhalt übernommen wird.
  - Der Viewdefinition eines Feldes, die aus der Viewposition des Originalfeldes in der durch die Mobile Notes Serverkomponente erzeugten Viewzeile und der Anzahl Zeichen, die aus dem Originalfeld in die Viewzeile übernommen werden, besteht. Sollte der Originalfeldinhalt länger sein als die angegebenen Viewzeichen, so werden nur die ersten Zeichendes Feldes übernommen.

View

Viewname in the database : All by Date

Showed client viewname : All by Date

Form

Form for creating documents: Memo

Showed Fields

Original name:	Showed name:	Bytesize:	Type:	Fa.field:	View / Pos.	Length
1. DeliveredDate	DeliveredDate	200	TIME	No fa.field	3	10
2. From	From	200	TEXT	No fa.field	1	10
3. Subject	Subject	200	TEXT	No fa.field	2	10
4.		200	TEXT	No fa.field	No View	10
5.		200	TEXT	No fa.field	No View	10
6.		200	TEXT	No fa.field	No View	10
7.		200	TEXT	No fa.field	No View	10
8.		200	TEXT	No fa.field	No View	10

Abb. 20: Datenbankdokument b

4.7.3 Benutzergruppendokument

In der Mobile Notes Initialisierungsdatenbank besteht zusätzlich die Möglichkeit, Benutzergruppen zu definieren, damit Datenbanken für mehrere Benutzer freigeschaltet werden können (z.B. Projektarbeit) und nicht jedes einzelne Benutzerprofildokument bearbeitet werden muß. Sollte eine Datenbank sowohl im persönlichen Benutzerpofil einer Person als auch durch eine Benutzergruppe in der die Person Mitglied ist, freigeschaltet werden, so wird das Zugriffsrecht, das im persönlichen Benutzerprofil der Person eingetragen ist, verwendet. Das Zugriffsrecht einer in einem Gruppendefinitionsdokument freigeschalteten Datenbank ist ansonsten für alle Gruppenmitglieder gültig. Eine Gruppendefinitionsdokument beinhaltet folgende Daten:

- Den Namen der Gruppe sowie einer näheren Beschreibung der Gruppe.
- Die Gruppenmitglieder, für die zuvor in der Mobile Notes Initialisierungsdatenbank ein persönliches Benutzerprofil eingerichtet werden muß. Diese werden in der Auswahlbox, die nach dem Drücken des "Add"-Knopfes in der entsprechenden Rubrik erscheint, aufgeführt und können für die Gruppe ausgewählt werden.
- Den Datenbanken, auf die die Gruppe den entsprechenden Zugriff erhält. Die Datenbanken müssen ebenfalls zuvor durch ein Datenbankdefinitionsdokument eingerichtet worden sein, um durch drücken des "Add"-Knopfes der Datenbankrubrik in der Auswahlbox angeboten zu werden.

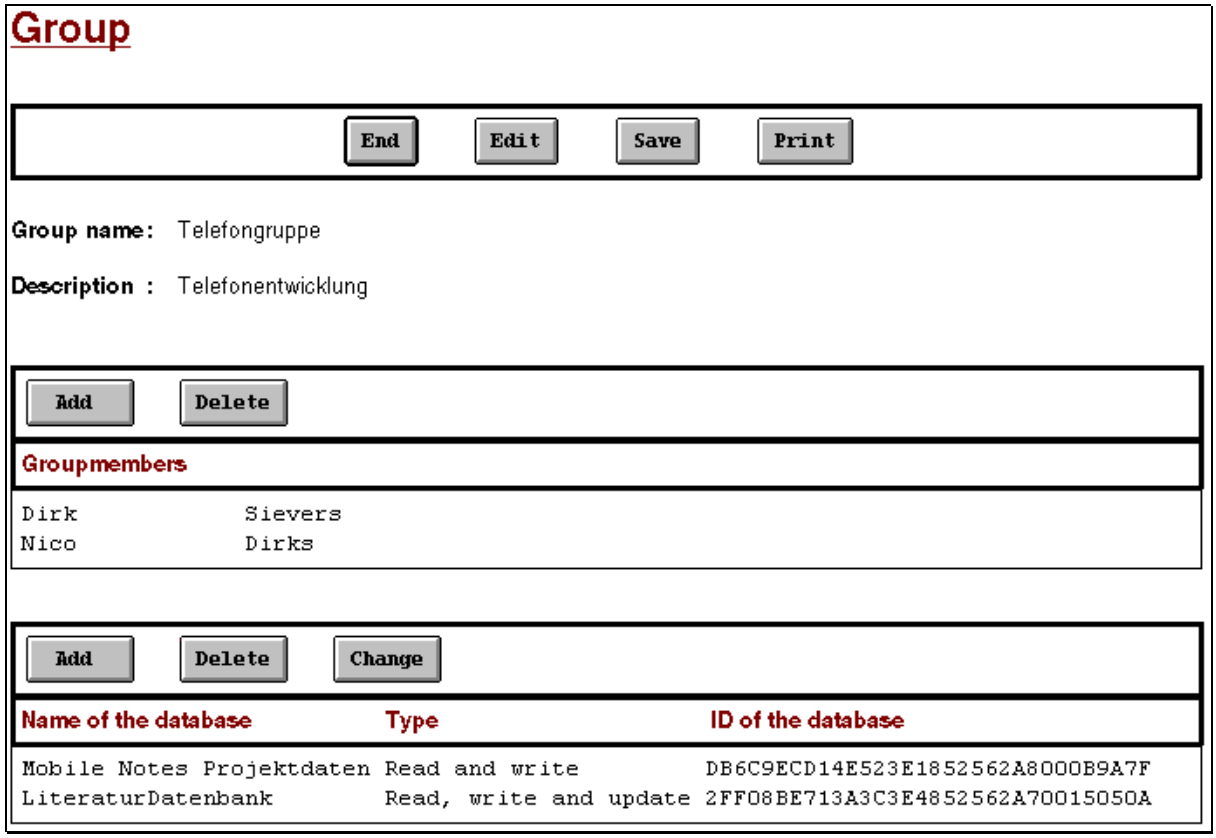

Abb. 21: Benutzergruppendokument

4.7.4 Systemadministrationsdokument

Das Systemadministrationsdokument dient der Steuerung des Mobile Notes Systems und kann nur einmal in der Mobile Notes Initialisierungsdatenbank erzeugt werden. In der vorliegenden Mobile Notes Version enthält das Systemadministrationsdokument folgende Daten:

- Den Namen des Mobile Notes Administrators
- Den Pfad und den Namen der Maildatenbank des Mobile Notes Administrators. Die Maildatenbank des Systemadministrators wird benötigt, damit eventuelle Angriffe auf das Mobile Notes System durch unberechtigte Dritte dem Administrator durch eine Mail mitgeteilt werden können.
- Der Anzahl der Versuche, den korrekten Namen eines Mobile Notes Benutzers eingeben zu können. Sollte die Anzahl überschritten werden, unterbricht die Mobile Notes Serveranwendung die Verbindung mit dem Mobile Notes Client.
- Der Anzahl der Versuche, das korrekte Paßwort des Mobile Notes Benutzers eingeben zu können. Sollte diese Anzahl überschritten werden, so wird die Verbindung mit dem Mobile Notes Client unterbrochen. Zusätzlich erfolgt eine Mail an den Systemadministrator sowie an den Benutzer selbst, da eine nicht berechtigte Person versucht haben könnte, Zugriff auf das Mobile Notes System zu erlangen.

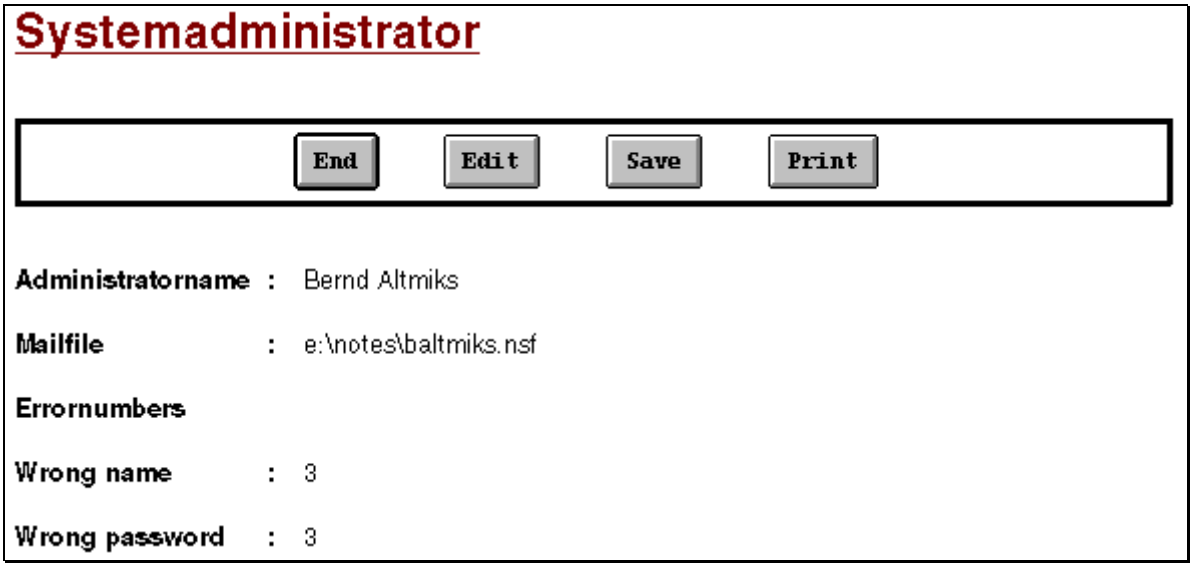

Abb. 22: Systemadministratordokument

4.7.5 Connectiondokument

Mit dem Connnectiondokument werden die seriellen Schnittstellen der Mobile Notes Serveranwendung definiert. In der vorliegenden Version der Mobile Notes Initialisierungsdatenbank können bis zu acht serielle Schnittstellen eingerichtet werden. Im Verlauf der Entwicklung der Mobile Notes Serverkomponente zeigte sich, daß die Lotus Notes 3.3 API Version nicht mulithreadingfähig ist. Dadurch ist es der Mobile Notes Serverkomponente nicht möglich, mehrere serielle Schnittstellen gleichzeitig anzusteuern. Jedoch kann durch mehrmaliges Starten der Serverkomponente mit dem entsprechenden Modemanschluß als Parameter dieses Problem umgangen werden, was aber zu Geschwindigkeitseinbußen führt.
Das Connectiondokument enthält folgenden Daten:

- Die Bezeichnung des Modemtyps, das mit dem entsprechenden seriellen Anschluß verbunden ist.
- Den Initialisierungsstring, welcher beim Start der Mobile Notes Serverkomponente an das Modem gesendet wird und dieses initialisiert.
- Die Bezeichnung der seriellen Verbindung, an der das Modem angeschlossen ist.

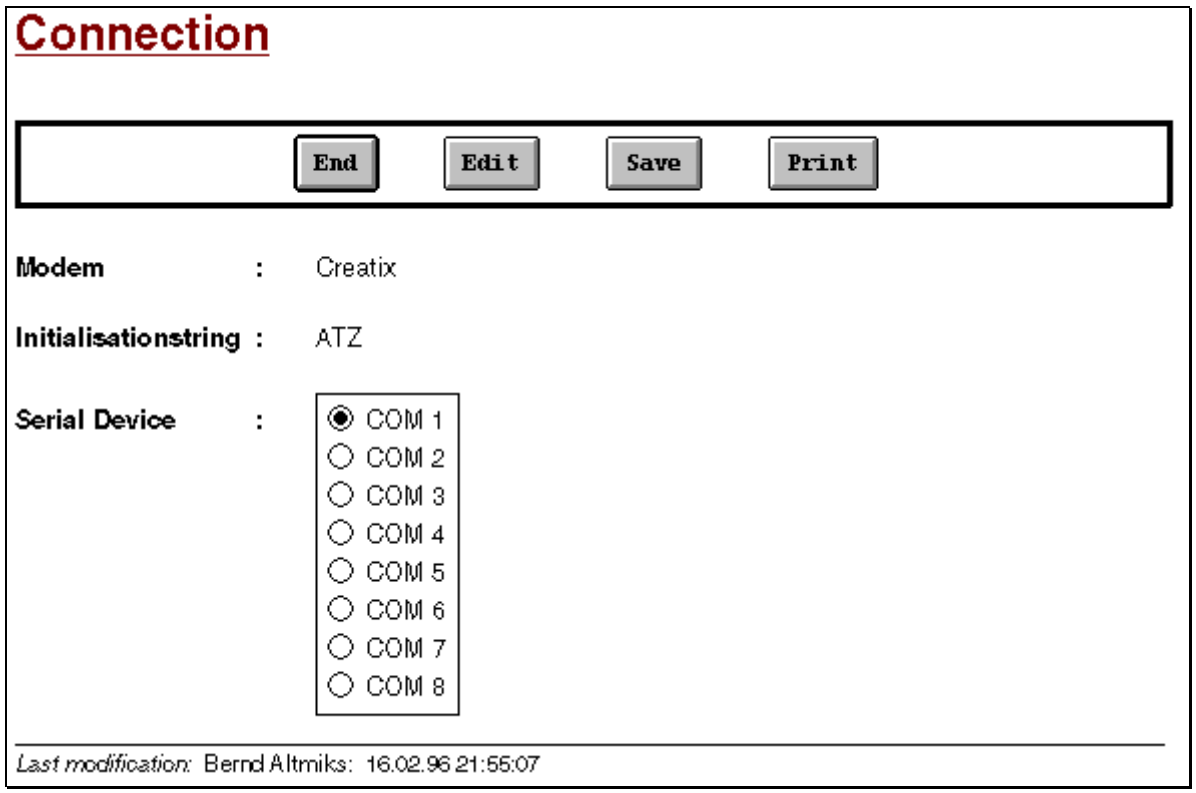

Abb. 23: Verbindungsdokument

5. Zusammenfassung und Ausblick

Die vorliegende Arbeit beschäftigt sich zunächst mit dem Thema Mobile Computing, indem Einsatzbereiche, Komponenten sowie die Vorteile und Risiken näher erläutert werden. Die hierzu dargestellten Fakten sowie die zunehmende Verbreitung mobiler Kommunikationssysteme sind die Grundlage für das Kernthema dieser Arbeit:
Die Entwicklung einer generischen Serverkomponente, die es Clientsystemen mit Modemanschluß ermöglicht, das bestehende Groupware Backend System Lotus Notes zu nutzen.
Es werden zunächst das AvALoN Projekt (Advanced Access to Lotus Notes) sowie dessen Nachfolger, das Mobile Notes Projekt vorgestellt, welches in Kooperation mit den Firmen Philips, Pavone und dem Lehrstuhl Wirtschaftsinformatik 2 der Universität-Gesamthochschule Paderborn von den Studenten Nico Dirks, Dirk Sievers und dem Autor dieser Arbeit bearbeitet wird. Die Entwicklung der Mobile Notes Serverkomponente ist ein Teil dieses Projektes. Zusätzlich wird eine Mobile Notes Applikation für das Screenphone P100 erstellt, was die praktische Anwendung der Serverkomponente demonstriert.[39]
Für die nähere Beschreibung der Serverkomponente werden zunächst die Anforderungen, die an eine solche Komponente zu stellen sind, diskutiert. Auf Grund der zunehmenden Bedeutung der in solchen Client Server Architekturen gehaltenen und übermittelten Daten, wird besonders auf die Sicherung der Daten sowie der Datenübertragung eingegangen. Es werden verschiedene Verfahren für die Fehlerkorrektur bei der Datenübertragung und für die Verschlüsselung von Daten erläutert und beschrieben. Zusätzlich wird die einfache Implementation sowie die problemlose Erweiterung durch den modularen Aufbau der Mobile Notes Serverkomponente dargestellt.
Weiterhin werden die Vorteile und Anwendungsmöglichkeiten der entwickelten Mobile Notes Serverkomponente in Verbindung mit dem Groupware Backend System Lotus Notes dargestellt. Die durch die Mobile Notes Serverkomponente geschaffene Anbindung an die Lotus Notes Groupware Umgebung bietet neben bereits existierenden Anwendungen wie dem ebenfalls am Lehrstuhl für Wirtschaftsinformatik 2 der Universität-Gesamthochschule Paderborn entwickelten Prototyp "AudioAccess"[40] der Lotus Notes Datenbanken für normale Telefone verfügbar macht, einen weitere Möglichkeit, Lotus Notes Datenbanken auf Frontendsystemen darzustellen und zu bearbeiten, die auf Grund ihrer technischen Gestaltung ursprünglich nicht hierfür konzipiert sind.
Ein entscheidender Aspekt ist dabei, daß mit der Mobile Notes Komponente neuartige Anwendungen sowohl auf der Client, als auch auf der Groupware Backend Seite, erstellt werden können, die die Vorteile einer Groupware Umgebung mit denen eines mobilen

---

[39] siehe hierzu die Diplomarbeiten der am Mobile Notes Projekt beteiligten Studenten Nico Dirks und Dirk Sievers.
[40] vgl. Soldner, Marcus (Diplomarbeit) S. 39 ff.

Frontendsystems verbinden. Bisherige Lösungen waren entweder nur für spezielle Clientsysteme gedacht, oder stellten zu hohe Anforderungen an die jeweiligen Frontendsysteme und konnten somit von vielen aktuellen mobilen System nicht genutzt werden.
Die Erklärung der praktischen Anwendung des Datenübertragungsprotokolls der Mobile Notes Serverkomponente sowie die Vorstellung Mobile Notes Initialisierungsdatenbank, welche der Steuerung des Mobile Notes Systems dient, beenden die vorliegende Diplomarbeit. Ein Überblick über alle Befehle, die das Mobile Notes Datenübertragungsprotokoll bietet, findet sich im Anhang. Zusätzliche Informationen über die Entwicklung einer Frontendapplikation, die auf dem Datenübertagungsprotokoll basiert, können den Diplomarbeiten der am Mobile Notes Projekt beteiligten Studenten Nico Dirks und Dirk Sievers entnommen werden.

Ausblick:
"Die Unterstützung mobiler Teilnehmer und deren volle Integration in existierende Informations- und Kommunikationssysteme gewinnt zunehmend an Bedeutung.... Durch den Fortschritt in der Informationstechnik wird es möglich, kleine aber dennoch leistungsfähige, tragbare Rechner herzustellen und diese durch drahtlose Kommunikationssysteme mit Festnetzen zu verbinden. Mobile Computing Anwendungen werden möglich."[41]

Aktuelle Entwicklungen auf dem Gebiet des Mobile Computing zeigen, daß Anwendungen aus diesem Bereich die moderne Kommunikationsgesellschaft stark beeinflussen werden. Als Beispiel seien an dieser Stelle die Paging-Systeme wie "Cityruf" genannt, die es in einem bestimmten Bereich (City) ermöglichen alphanumerische Daten zu übertragen. Obwohl diese Systeme einen eher geringen Leistungsumfang bieten, zeigen sie doch einen Trend in der Kommunikation der Zukunft auf: Ständige Erreichbarkeit an jedem Ort dieser Welt und die unbegrenzte Möglichkeit des Datenaustauschs mit dem jeweiligen Kommunikationspartner. Neuartige Mobiltelefone, die über ein Display für die Anzeige von Daten verfügen, sind ein weiterer Schritt in diese Richtung. An dieser Stelle setzt das in dieser Arbeit entwickelte Mobile Notes System an. Es wird nun möglich das Lotus Notes System mit den neuen Produkten der Kommunikationsindustrie auf eine einfache Weise zu verknüpfen. Durch diese Erweiterung können nun Funktionalitäten, wie sie vorher nur mit Spezialanwendungen wie dem oben erwähnten "Cityruf" möglich waren, in das bestehende Lotus Notes System integriert werden. Zusätzlich bietet das Mobile Notes System jedoch die Möglichkeit, Lotus Notes Datenbanken zu bearbeiten. Es können nun kostengünstige Systeme z.B. für die Datenerfassung erstellt werden, die die Vorteile der Lotus Notes Groupware Umgebung nutzen, ohne daß ein Lotus Notes Client verwendet werden muß.

---

[41] Diehl, Norbert: (Mobile Computing) S. 173

Literaturverzeichnis

Behrens, Olav, Max: (Hypermediakonzepte in Groupwareapplikationen)
Konzepte, Anforderungen und Lösungsmöglichkeiten für die Integration von Hypermediakonzepten in Groupwareapplikationen); Dissertation der Hochsschule St. Gallen, 1995

Beutelsbacher, Albrecht: (Kryptologie)
Eine Einführung in die Wissenschaft vom Verschlüsseln, Verbergen und Verheimlichen, Vieweg Verlag, Braunschweig/Wiesbaden, 1994

Büllesbach, Alfred: (Informationsverarbeitung, Datenschutz und Qualitätsmanagement)
in: ITG-Fachbericht 129 Herausforderung Inforamtionstechnik, Vorträge der ITG-Fachtagung anläßlich des VDE-Kongresses'94 am 19. und 20. Oktober1994 in München, VDE Verlag, Berlin, 1994

Decker, P.; Walke, B.: (Mobile Datenkommunikation)
Mobile Datenkommunikation eine Übersicht, in: it+ti 1.05.1993, S. 12

Denning, Jens; Hartstock, Helmut; Kleppel, Michael; Kossow, Rainer; Tusche, Michael;
(Lotus Notes) Das Kompendium Einführung Arbeitsbuch Nachschlagwerk, Markt und Technk Verlag, Haar, 1995

Diehl, Norbert: (Mobile Computing)
Mobile Computing - von den Komponenten zur Systemintegration, in: ITG-Fachbericht 129 Herausforderung Inforamtionstechnik, Vorträge der ITG-Fachtagung anläßlich des VDE-Kongresses'94 am 19. und 20. Oktober1994 in München, VDE Verlag, Berlin, 1994, S. 173 - 189

Dirks, Nico: (Architekturen und Anwendungskonzepte von Groupware)
Architekturen und Anwendungskonzepte von Groupware in der mobilen Kommunikation - Generische Entwicklung von Benutzgsschnittstelle und Prozeß-Modulen für ein intelligentes Display-Telephon.) Diplomarbeit an der Universität-Gesamthochschule Paderborn, 1996

Eitel, Barbara; Kian Torabli: (Mobile Computing)
Mobilität potenziert den Nutzwert von Informationen als Produktionsfaktor. Das wirtschaftliche Potential des Mobile Computing wird längst nicht ausgeschöpft, in: Computerzeitung Spezial: Mobile Computing Nr. 37 14.09.95, S. 31

Eldib E. Osman; Minoli, Daniel: (Telecommuting)
Verlag Artech House, Inc., Boston London, 1995

Fellbaum, Klaus; Rainer, Hartlep: Lexicon der Telekommunikation
VDE Verlag, Berlin, 1984

Forman, George H.; Zahorjan, John: (Mobile Computing)
The Challenges of Mobile Computing, in: Computer 4/94 S. 39 - 46

Gabler, Hermann: (Text und Datenübermittlung)
Grundlagen der Text und Datenübermittlung, R. v. Decker's Verlag, Heidelberg, 1989

Gross, Denis; Duffy, Richard: (World without wires),
in: Communications International 6/95 S. 72 - 76

Görgen, K.; Koch, H.; Schulze, B.; Struif, K.; Truöl, K.: (Grundlagen der Kommunikations-technologie) ISO-Architektur offener Kommunikationssysteme, Springer-Verlag, Berlin u.a., 1985

Herget, Josef: (Wirtschaftlichkeit der Bürokommunikation)
Schwer zu beweisen- Wirtschaftlichkeit der Bürokommunikation, in: Business Computing 1/94, S. 42 - 46

Jagoda, A.; Villepin, M.: (Mobile Communications)
Teubner Verlag, Stuttgart, 1993

Kerningham, Brian W.; Ritchie, Dennis M.: (Programmieren in C)
Coedition der Verlage: CarHanser Verlag, München u.a., Prentice-Hall International Inc., London, 1990

Kleinwächter, Rolf: (Mobile Computing und Kommunikation)
Technologietrends in Hard- und Software Drahtlose und drahtgebundene Kommunikation, in: Electronic Selling, 1995

Lexikon der PC Fachbegriffe, Band 2, Dezember 1993 (Mobilkommunikation)

Lexikon der PC Fachbegriffe, Band 1, Dezember 1995 (Mobile Computer)

Mundt, Karl-Heinz: (Der Data Encryption Standard (DES)),
in: c't 6/1994, S. 184

Nastansky, Ludwig: ("Büroinformationssysteme"),
in : Fischer, Herold, Dangelmeier, Nastansky, Wolff, Bausteine der Wirtschaftsinformatik, S+W Verlag, S. 273-373, Hamburg, 1994

Nastansky, Ludwig: (Groupware-Anwendungen)
Nach 20 Jahren CSCW-Forschung: Durchbruch in der Praxis bei Groupware-Anwendungen in Client-Server Architekturen, in: Ludwig Nastansky (Hrsg.): Workgroup Computing - Computergestützte Teamarbeit (CSCW) in der Praxis / Neue Entwicklungen und Trends (Betriebswirtschaft aktuell, Band 12), S. 1-20; S+W Steuer und Wirtschaftsverlag, Hamburg, 1993

Nastansky, Ludwig: (Workflow Management)
Workflow Management - Endlich Paradigmenwechsel im Büro?, in: Computerwoche Extra, Ausg. Nr. 3, 18. Aug. 1995, S. 8 - 11, 25

Nastansky, Ludwig: (Gruppenarbeit - Workgroup Computing),
in: Office Management 6/91, S.: 6ff., FBO, Baden-Baden, 1991

Niemeier, Joachim; Schäfer, Martina; Engstler, Martin; Koll, Peter: (Mobile Computing)
Informationstechnologie ortsungebunden nutzen - Techniken - Einsatz - Wirtschaftlichlichkeit, Verlag Computerwoche GmbH (Hrsg.), 1994

Ohmann, Friedrich: (Kommunikations-Endgeräte)
Grundlagen, Verfahren, Bausteine, Geräte, Systeme, Springer-Verlag, Berlin u.a., 1983

Philips AG: (P100 System Software: Modem interface Specification Version 2.1)
Eindhoven, 14 April, 1994

Riempp, Gerold: (Modellentwurf für Workflow Management), Modellentwurf für Workflow Management mit verteilten Dokumenten-Datenbanken im WAN-Verbund, Diplomarbeit an der technischen Hochschule Darmstadt, 1994

Satyanarayanan, M.: (Mobile Computing)
in:Computer 9/93,S. 81 - 82

Satyanarayanan, M.; Noble, Brian; Kumar, Puneet; Price, Morgan: (Application-Aware Adaptation for Mobile Computing),
in Operating Systems Review, Januar 1995 Vol. 29 Number 1, S. 52 - 54

Schneider, Bruce: (Applied Cryptography)
Protocols, Algorithms and Source in C,Verlag John Wiley & Sons Inc., 1993

Sievers, Dirk: (Architekturen und Anwendungskonzepte von Groupware in der mobilen Kommunikation - Generische Entwicklung von Kommunikationssteuerungs-modulen und Datenrepositories) Diplomarbeit an der Gesamthochschule-Universität Paderborn, 1996

unbekannter Autor: (TeamAgent will synchronize Notes databases with Newtons)
in: Byte 12/95

Weck, Gerhard: (Datensicherheit)
Methoden, Maßnahmen und Auswirkungen des Schutzes von Informationen,
Teubner Verlag, Stuttgart, 1984

Anhang Protokollbefehle und Konstanten

Die Mobile Notes Übertragungsprotokollbefehle und deren Aufbau

Sämtliche Mobile Notes Datenübertragungsprotokollbefehle sind nach folgendem Schema aufgebaut:

SOH Operator [GS [Parameterlänge [RS Parameterlänge]* GS] Daten] ETB

wobei:

SOH	=	start of header	(ASCII-Code 01 dezimal)
GS	=	group seperator	(ASCII-Code 29 dezimal)
RS	=	record seperator	(ASCII-Code 30 dezimal)
ETB	=	end of transmission block	(ASCII-Code 23 dezimal)

und:

Operator = 1 Byte, Code für einen Befehl aus den nachfolgenden Gruppen darstellt

Parameterlänge = 2 Bytes, Länge des x.ten Parameters im Datenblock

Daten = Datenblock, indem die Parameter ohne Trennsymbol nacheinander aufgereiht sind.

Weiterhin sollen für die Übertragung die Standardsymbole aus dem ASCII-Zeichensatz genutzt werden.

Die Befehle werden in 8 Gruppen zu jeweils maximal 16 Operatoren aufgeteilt werden.

Numeriert werden diese mit denDezimalwerten 128 bis 255 um Verwechselungen mit ASCII-Symbolen auszuschließen. (siehe Abschnitt Befehlskonstanten)

Gruppe 1 / Allgemeines

Befehl: OP_PROTVERSION

Befehlsbeschreibung: Der Befehl übermittelt den Clienttypen und die zu nutzende Protokollversion (CRC verwenden, DES-Verschlüsselung etc.)

	Beschreibung	Länge	Beispiel
Parameter:	1. Clienttyp	1 Byte	AVA_PROT_P100
	2. Protokollversion	1 Byte	AVA_TVERSION

Beispiel: SOH OP_PROTVERSION GS 1 RS 1 GS AVA_PROT_P100 AVA_PROT_TVESION ETB

Antwortbefehl: OP_ANSWER

Antwortparameter:	AVA_GEN_OK	= Client und Protokollversion ok
	AVA_PROT_UNSUPPORTED_CLIENT	= Client nicht bekannt
	AVA_PROT_UNSUPPORTED_PROTOCOL	= Protokoll nicht bekannt

Befehl: OP_NAME

Beschreibung: Sicherheitscheck (Server und Client):

Client schickt den Namen des Users (Verschlüsselung abhängig von der Protokollversion), zum Server. RSA wird anfangs noch nicht implementiert (Lizenzkosten, Quellcode)

	Beschreibung	Länge	Beispiel
Parameter:	1. Vorname Nachname	beliebig	Bernd Altmiks
	2. Zufallszahl	max. 8 St.	12345678

Beispiel: SOH OP_NAME GS 13 RS 8 GS B e r n d A l t m i k s 12345678 ETB

Antwort: OP_ANSWER

Antwortparameter: 1. OP_NAME = Befehl der Antwort erwartet
2. Vorname Nachname, Zufallszahl = Daten die zuvor hochgeschickt wurden

---

Befehl: OP_PASSWORD (Einlogvorgang)

Beschreibung: Server erfährt, ob der Client korrekt ist (Passwort kann nur dem korrekten Client bekannt sein)

	Beschreibung	Länge	Beispiel
Parameter:	1. Passwort	beliebig	Avalon
	2. Zufallszahl	max. 8 St.	12345678

Beispiel: SOH OP_Password GS 6 RS 8 GS A v a l o n 12345678 ETB

Antwort: OP_ANSWER

Antwortparameter: 1. OP_PASSWORD = Befehl der Antwort erwartet
2. Statuscode = AVA_GEN_OK oder entsprechender Fehlerwert (siehe Fehlerkonstanten)

Befehl: OP_LOGOFF

Beschreibung: Der Client teilt dem Server mit, daß die Verbindung beendet wird.

	Beschreibung	Länge	Beispiel
Parameter:	-		

Beispiel: SOH OP_LOGOFF ETB

Antwort: -

Antwortparameter:

Befehl:	OP_CHANGE_PASSWORD		
Beschreibung:	Client teilt dem Server mit, daß das Passwort in der Mobileno.nsf geändert werden soll.Dieser Befehl kann erst dann gewählt werden, wenn der User erfolgreich eingelogt ist.		
	Beschreibung	Länge	Beispiel
Parameter:	1. Neues Passwort	beliebig	Neuwort
Beispiel:	SOH OP_CHANGE_PASSWORD GS 7 GS N e u w o r t ETB		
Antwort:	OP_CHANGE_PASSWORD		
Antwortparameter:	Neues Passwort	beliebig	Neuwort
Bemerkung:	Das Passwort wird vom Server zum Client zurückgeschickt und erst wenn dieser das Passwort nochmals bestätigt, (OP_ANSWER + AVA_GEN_OK) wird das neue Passwort in die Mobile Notes Initialisierungsdatenbank eingetragen		

Gruppe 2 / Status&Antwort / OP_STAT ab 144

Befehl:	OP_ANSWER		
Beschreibung:	Sendet eine Antwort auf auf die jeweilige Aktion		
	Beschreibung	Länge	Beispiel
Parameter:	1. Aufrufender Befehl	1 Byte	OP_DOC_GET_INFO
	2. Antwort	beliebig ist ...	1234567890123456Dies Die Länge des Antwortparameters ist abhängig vom aufrufenden Befehl !!
Beispiel: (ID+Viewzeile)	SOH OP_ANSWER GS 1 0 RS 28 0 GS OP_DOC_GET_INFO ETB		

Gruppe 3 / Datenbanken / OP_DB

Befehl: OP_DB_CHECK_DBS

Beschreibung: Sendet dem Server die zu überprüfenden DatenbankID's mit den dazugehörigen Kontrollzeiten und dem Zugriffsrecht.(DatenbankID = Datenbankinitalisierungs-dokumentID)

	Beschreibung	Länge	Beispiel
Parameter:	1. Anzahl der DatenbankID's	2 Byte	1
	2. Länge der gesamten Daten (ID's, Kontrollzeiten und Zugriffsrecht)	2 Byte	34

Beispiel: SOH GS OP_DB_CHECK_DBS 1 0 RS 33 0 GS (16 char UnId) (17 char Zeit) (1 char Zugriffsrecht) ETB

Antwort: SOH OP_DB_CHANGED_DBS

Antwortparameter: UnId's der Datenbanken die neu oder verändert sind

---

Befehl: OP_DB_CHANGED_DBS

Beschreibung: Sendet dem Client die veränderten (neuen) DatenbankID's (DatenbankID = DatenbankinitalisierungsdokumentID)

	Beschreibung	Länge	Beispiel
Parameter:	1. Anzahl der DatenbankID's	2 Byte	1
	2. Länge der gesamten (DatenbankID's)	x*16 Byte	16

Beispiel: SOH GS OP_DB_CHANGED_DBS GS 1 0 RS 16 0 GS 1 9 8 7 6 5 4 3 2 1 0 1 2 3

4 5 6 ETB

Befehl: OP_DB_GETDBDEF

Beschreibung: Fordert eine bestimmte Datenbankdefinition vom Server an

	Beschreibung	Länge	Beispiel
Parameter:	1. DatenbankId	16 Byte	1234567890123456
Beispiel:	SOH GS OP_DB_GETDBDEF GS 16 0 GS 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 ETB		
Antwort:	OP_DB_DEF		
Antwortparameter:	Datenbankdefinition		

Befehl: OP_DB_DEF

Beschreibung: Sendet dem Client die Datenbankdefinition

	Beschreibung	Länge	Beispiel
Parameter:	1. Länge der (Datenbankdefinition)	2 Byte	
Beispiel:	SOH GS OP_DB_DEF GS xx GS Datenbankdefinition ETB		

Aufbau einer Datenbankdefinition:

1. ID	= 16 Byte	z.B. 1234567812345678
2. Kontrollzeit	= 17 Byte	z.B. 20.08.95 11:23:17
3. DBName	= 40 Byte	z.B. Testdatenbank
4. DbView	= 40 Byte	z.B. Mein spezieller TelefonView
5. Zugriffstyp	= 1 Byte	z.B. 1; 2; 3; 4 (read, write, read a. write; read a. write a. update)
6. definierte Felder	= 1 Byte	z.B. 5 (Feldanzahl ist <= MAX-FIELDS = 10)

Die Felderdefinitionen wiederholen sich Feldanzahl mal.

Felder:

7. Feldnummer	= 1 Byte	z.B. 1
8. Aliasname	= 20 Byte	z.B. Betrifft
9. Typ	= 1 Byte	z.B. T = Text; N = Number; D = Datum
10. Größe	= 2 Byte	z.B. 100
11. Vererbungsfeldnnummer	= 1 Byte	z.B. 5 (Bekommt Daten aus Feld 5)

12. ReplyMode = 1Byte z.B. (steht noch nicht fest)
13. Viewfeld = 1 Byte Viewfeldnummer (0 falls kein Viewfeld)
und Position im View
14. Anzahl Zeichen = 1 Byte z.B. 20 (Zeichen in der Viewzeile)

Befehl:	OP_DB_BY_UNID		
Beschreibung:	Auf die Datenbank mit der UnicId springen, die in der Liste des Users steht		
	Beschreibung	Länge	Beispiel
Parameter:	UnicId	2 Byte	16
Beispiel:	SOH OP_DB_BY_UNID GS 2 0 GS 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 ETB		
Antwort:	OP_STATUS		
Antwortparameter:	AVA_GEN_OK	= Positionierung konnte durchgeführt werden	
	AVA_GEN_FALSE	= Positionierung fehlgeschlagen	

Befehl:	OP_DB_GET_TITLE	
Beschreibung:	Client erfragt den Namen der Datenbank die im Augenblick für den User im Zugriff liegt	
	Beschreibung	Länge Beispiel
Parameter:	-	
Beispiel:	SOH OP_DB_GET_TITLEETB	
Antwort:	OP_ANSWER	
Antwortparameter:	1. OP_DB_GET_TITLE	= Befehl auf den die Antwort erfolgt
	2. DbName	= Name der Datenbank (Länge beliebig)

Gruppe 4 / Dokumentenbefehle / OP_DOC

Befehl:	OP_DOC_FIRST		
Beschreibung:	Auf das erste Dokument springen, das in der Datenbank steht		
	Beschreibung	Länge	Beispiel
Parameter:			
Beispiel:	SOH OP_DOC_FIRST ETB		
Antwort:	OP_STATUS		
Antwortparameter:	AVA_GEN_OK	= Positionierung konnte durchgeführt werden	
	AVA_GEN_FALSE	= Positionierung fehlgeschlagen	

Befehl:	OP_DOC_LAST		
Beschreibung:	Auf das letzte Dokument springen, das in der Datenbank steht		
	Beschreibung	Länge	Beispiel
Parameter:			
Beispiel:	SOH OP_DOC_LAST ETB		
Antwort:	OP_STATUS		
Antwortparameter:	AVA_GEN_OK	= Positionierung konnte durchgeführt werden	
	AVA_GEN_FALSE	= Positionierung fehlgeschlagen	

Befehl:	OP_DOC_NEXT		
Beschreibung:	Auf das nächste Dokument springen, das in der Datenbank steht		
	Beschreibung	Länge	Beispiel
Parameter:			
Beispiel:	SOH OP_DOC_NEXT ETB		
Antwort:	OP_STATUS		
Antwortparameter:	AVA_GEN_OK	= Positionierung konnte durchgeführt werden	
	AVA_GEN_FALSE	= Positionierung fehlgeschlagen	

Befehl:	OP_DOC_PREV		
Beschreibung:	Auf das vorherige Dokument springen, das in der Datenbank steht		
	Beschreibung	Länge	Beispiel
Parameter:			
Beispiel:	SOH OP_DOC_PREV ETB		
Antwort:	OP_STATUS		
Antwortparameter:	AVA_GEN_OK	= Positionierung konnte durchgeführt werden	
	AVA_GEN_FALSE	= Positionierung fehlgeschlagen	

Befehl:	OP_DOC_BY_UNID		
Beschreibung:	Auf das Dokument mit der UnicId springen		
	Beschreibung	Länge	Beispiel
Parameter:	UnicId	2 Byte	16
Beispiel:	SOH OP_DOC_BY_UNID GS 2 0 GS 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 ETB		
Antwort:	OP_STATUS		
Antwortparameter:	AVA_GEN_OK	= Positionierung konnte durchgeführt werden	
	AVA_GEN_FALSE	= Positionierung fehlgeschlagen	

Befehl:	OP_DOC_GET_INFO		
Beschreibung:	Client erfragt die Viewzeile des aktuellen Dokumentes.		
	Beschreibung	Länge	Beispiel
Parameter:	-		
Beispiel:	SOH OP_DOC_GET_INFOETB		
Antwort:	OP_ANSWER		
Antwortparameter:	1. OP_DOC_GET_INFO	= Befehl auf den die Antwort erfolgt	
	2. DocId und Viewzeile	= (16 char) + Viewzeilenlänge	

Befehl:	OP_DOC_REQUEST		
Beschreibung:	Anforderung des aktuellen Dokumentes		
	Beschreibung	Länge	Beispiel
Parameter:	-		
Beispiel:	SOH OP_DOC_REQUESTETB		
Antwort:	OP_DOC_CONTENTS		
Antwortparameter:	Feldanzahl * Felder	= Dokument wird übertragen	
	AVA_GEN_FALSE	= Fehler	

Befehl:	OP_DOC_CONTENTS		
Beschreibung:	Senden eines bestimmten Dokumentes (Hierbei werden die Felder des Dokumentes übertragen)		
	Beschreibung	Länge	Beispiel
Parameter:	Länge der einzelnen Felder	2 Byte	
Beispiel:	SOH OP_DOC_CONTENTS GS 2 Byte RS...RS 2 Byte GS Feld1Inhalt Feld2Inhalt ... FeldxInhalt ETB		

Befehl:	OP_DOC_UPDATE	
Beschreibung:	Updaten des aktuellen Dokumentes	
	Beschreibung	Länge Beispiel
Parameter:	Länge der einzelnen Felder	2 Byte
Beispiel:	SOH OP_DOC_UPDATE GS 2Byte RS 2 Byte ... RS 2 Byte GS Feld1Inhalt Feld2Inhalt ...FeldxInhalt ETB	
Antwort:	OP_ANSWER	
Antwortparameter:	AVA_GEN_OK	= Positionierung konnte durchgeführt werden
	AVA_GEN_FALSE	= Positionierung fehlgeschlagen

Befehl:	OP_DOC_DELETE	
Beschreibung:	Löschen des aktuellen Dokumentes	
	Beschreibung	Länge Beispiel
Parameter:		
Beispiel:	SOH OP_DOC_DELETE ETB	
Antwort:	OP_ANSWER	
Antwortparameter:	AVA_GEN_OK	= Löschen erfolgreich
	AVA_GEN_FALSE	= Löschen fehlgeschlagen

Befehl:	OP_DOC_CREATE		
Beschreibung:	Erstellen eines neuen Dokumentes		
	Beschreibung	Länge	Beispiel
Parameter:	Länge der einzelnen Felder	2 Byte	
Beispiel:	SOH OP_DOC_CREATE GS 2Byte RS 2 Byte ... RS 2 Byte GS Feld1Inhalt Feld2Inhalt ...FeldxInhalt ETB		
Antwort:	OP_ANSWER		
Antwortparameter:	AVA_GEN_OK	= Dokument konnte erstellt werden	
	AVA_GEN_FALSE	= Dokument konnte nicht erstellt werden	

Befehl:	OP_DOC_GETVIEWLINES		
Beschreibung:	Anfordern mehrerer Viewzeilen		
	Beschreibung	Länge	Beispiel
Parameter:	1. Länge einer Viewzeile	2 Byte	30
	2. Anzahl Viewzeilen	2 Byte	5
Beispiel:	SOH OP_DOC_GETVIEWLINES GS 30 RS 5 ETB		
Antwort:	OP_ANSWER		
Antwortparameter:	Anzahl der erstellten Viewzeilen sowie die einzelnen Viewzeilen im Block		

Zusätzlich zu den in den Befehlen aufgeführten Fehlerkonstanten werden weitere Konstanten in das Übertragungsprotokoll implementiert werden, um die möglichen Fehler näher zu definieren.

Eine Überblick über die aktuellen Fehlerkonstanten gibt die folgende Liste:

Liste der aktuellen Fehlerwerte

Fehler	Dezimalwert	Beschreibung
AVA_GEN_OK	1100	Verarbeitung erfolgreich
AVA_GEN_FALSE	1101	Allg. Fehler
AVA_GEN_INTERNAL_ERROR	1102	Interner Verarbeitungfehl.
AVA_GEN_INSUFFICIENT_MEM	1103	Speichermangel
AVA_GEN_UNHANDLED_ERROR	1104	Nicht identifizierbarer Verarbeitungsfehler
AVA_GEN_SYSTEM_ERROR	1105	Nicht handhabbare System-fehlermeldung
AVA_GEN_NO_DES_KEY_SPECIFIED	1106	Kein DesKey angegeben

AVA_PAR_INVALID	1200	Ungültiger Parameter
AVA_PAR_OUT_OF_RANGE	1201	Definitionsbereich
AVA_NOTES_INVALID_DB	1400	Ungültige DB-Angabe
AVA_NOTES_INVALID_DBHDL	1401	Ungültiger DB-Handle
AVA_NOTES_OUT_OF_CONTEXT	1402	Kein weiterer Kontext
AVA_NOTES_INVALID_VIEW	1403	UngültigeViewAngabe
AVA_NOTES_VIEW_NOT_OPENED	1404	kein View geöffnet
AVA_NOTES_NO_DOC_AVAILABLE	1405	kein Dok. vorhanden
Fehler	Dezimalwert	Beschreibung
AVA_NOTES_NO_DOC_FOCUSED	1406	kein Dok. im Focus
AVA_NOTES_INVALID_FIELD	1409	Ungült. Field-Angabe
AVA_NOTES_INVALID_FORM	1410	Unguelt. Form-Angabe
AVA_NOTES_INTERNAL_ERROR	1412	interner NotesFehler
AVA_DB_NO_DB_AVAILABLE	1700	DB nicht vorhanden
AVA_DB_NO_DOC_AVAILABLE	1701	DOC nicht vorhanden
AVA_DB_NO_DB_SELECTED	1702	keine DB gewählt
AVA_DB_NO_DOC_SELECTED	1703	kein DOC gewählt
AVA_DB_FIELD_NOT_AVAILABLE	1704	Feld nicht verfügbar
AVA_DB_IS_DIFFERENT	1705	DB-Def geändert!
AVA_DB_DEF_NOT_OK	1706	DB-Def veraltet!
AVA_DB_INTERNAL_ERROR	1707	interner Fehler
AVA_DB_WRONG_DESKEY	1708	Dokument konnte nicht entschl.werden

Liste der aktuellen Protokollbefehle

Befehl	Dezimalwert	Beschreibung
OP_PROTVERSION	128	Legt das Protokoll fest
OP_NAME	130	Name + Zuf.Z. senden
OP_PASSWORD	131	Passwort + Zuf.Z. senden
OP_CHANGE_PASSWORD	132	Antrag auf PW-Änderung
OP_LOGOFF	133	Abmelden eines Users
OP_ANSWER	145	Antwort auf einen Befehl
OP_DB_CHECK_DBS	160	DB-IDs überprüfen
OP_DB_CHANGED_DBS	161	IDs der abzugl. DBs senden
OP_DB_GETDBDEF	162	best. DB-Def.anfordern

OP_DB_DEF	163	DB-Definition übertragen
OP_DB_BY_UNID	169	Db mit Id auswählen
OP_DB_COUNT	170	Anzahl Db's ermitteln
OP_DB_GET_TITLE	171	amen der akt. DB erfragen
OP_DOC_GET_INFO	176	Unid & Viewzeile anfordern
OP_DOC_FIRST	177	auf erstes Dokument gehen
OP_DOC_LAST	178	auf letztes Dokument gehen
OP_DOC_PREV	179	auf vorheriges Dok. gehen
OP_DOC_NEXT	180	auf nächstes Dokument gehen
OP_DOC_BY_UNID	181	Dokument per ID wählen
OP_DOC_REQUEST	183	Dokument anfordern
OP_DOC_CONTENTS	184	Dokument senden
OP_DOC_UPDATE	185	Dokument modifizieren
OP_DOC_CREATE	186	Neues Dokument erzeugen
OP_DOC_DELETE	187	Dokument löschen
OP_DOC_GET_VIEWLINES	189	Unid & Viewzeile mehrer Dokumente anfordern

Liste der aktuellen Protokollversionen

Protokollversion	Dezimalwert	Beschreibung
PROT_VERSION_TEST	192	TEST! ohne Des&Crc&etc.
PROT_VERSION_DES	193	Nur Des benutzen
PROT_VERSION_DES_CRC	194	Des und Crc benutzen
PROT_VERSION_CRC	195	Nur Crc benutzen
PROT_VERSION_DES_CRC_ASYM	196	Des, Crc u. Asym. Schl.
PROT_CLIENT_P100	207	Client: Philips P100