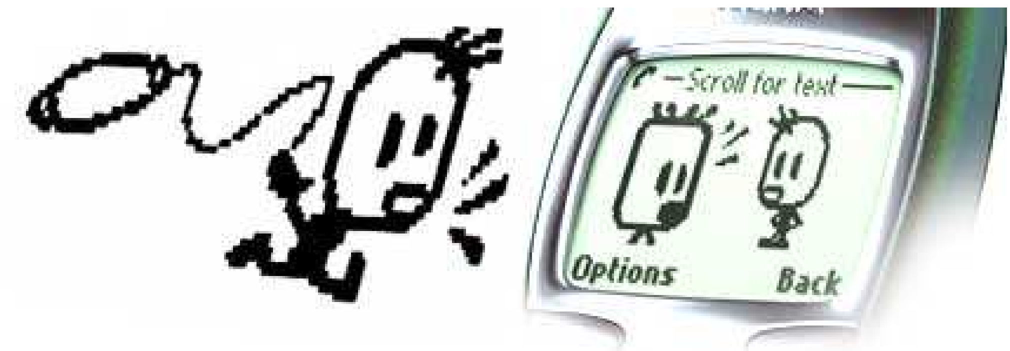
TECHNISCHE GRUNDLAGEN DES MOBILE COMMERCE

18. Juli 2001



Dirk Müller
Universität Kaiserslautern
Sommersemester 2001

Inhaltsverzeichnis

1 Einführung

1.1 Anforderungen an drahtlose Technologien

Electronic Commerce (E-Commerce) auf mobilen Endgeräten, dem *Mobile Commerce* (M-Commerce*)*, unterliegt ganz speziellen Anforderungen. Zum Einen entstehen diese Anforderungen aus Beschränkungen der verwendeten Übertragungs- und Endgerätetechnik, zum Anderen aber auch aus der unterschiedlichen Benutzergruppe dieser beiden Felder. Gegen die Benutzung des klassischen elektronischen Handels, des E-Commerce, sprechen unter anderem die Kosten für die benötigte, nur stationär verwendbare Computertechnik und die Anforderung an die Nutzer, zumindest Grundkentnisse in der Funktionsweise der Hard- und Software zu besitzen. Die Bedienung von Computerprogrammen ist immernoch für Technikunerfahrene schwierig und kompliziert, zudem herrscht manchmal eine, eventuell auch unbegründete, Abneigung gegen die klassische Computertechnik vor.

Der M-Commerce versucht diesen Anforderungen gerecht zu werden, in dem es den elektronischen Handel auch für Technikunerfahrene mobil und leicht benutzbar macht.

Dis Beschränkungen aufgrund der aktuell verbreiteten Technik der verfügbaren Endgeräte sind:

- schwache CPU Leistung. Gängige *Digital Wireless User Agents (WUA)*, wie z.B. Handy's oder *Personal Digital Assistants* (PDAs), verwenden aus Platz- und Stromverbrauchsgründen langsame, häufig RISC-basierte Mikroprozessoren.
- wenig ROM / RAM Speicher verfügbar. Die zu verwendende Software muss klein im Codeumfang und im Speicherverbrauch sein. Dies begrenzt die Funktionalität und Flexibilität der eingesetzten Applikationen.
- unzuverlässige, schwache Stromversorgung. Daraus folgt, daß das Endgerät für diese Situation optimiert sein sollte, d.h. Stromsparmodi besitzen sollte und mit unerwarteten Stromversorgungsausfällen zurecht kommen muss.
- kleine und geringauflösende Displays. Für Mobiltelefone sind z.Zt. schwarz/weiss LCD Displays mit Auflösungen kleiner als 160x100 Pixel üblich.
- beschränkte, häufig umständlich zu verwendende Eingabemöglichkeiten. Mobile Geräte müssen ohne PC-übliche Eingabegeräte auskommen. Das Gerät muss daher effektive und einfache Texteingabe- und Bedienmöglichkeiten bieten. Für Mobilfunktelefone ist z.B. das Texteingabesystem T9 [19] verbreitet, mit dem mit wenigen Eingaben über die Zifferntasten des Mobilfunktelefons aus einem eingebauten Wörterbuch „passende“ Wörter ausgewählt werden können.

Aufgrund dessen ist zumindest zur Zeit die Darstellung und Benutzung der klassischen WWW-Internetseiten und Webapplikationen des E-Commerce auf einem digitalen WUA nicht möglich. Zusätzlich existieren aber auch Einschränkungen im Übertragungsweg:

- geringe Bandbreite, häufig orts- und zeitbezogen stark schwankend. Verbreitete Mobilfunknetze wie das *Global System for Mobile communication* (GSM) [20] unterstützen lediglich 9600 Bps (bei optimalen Empfangsbedingungen).
- häufige Störung der Verbindungen (Bündelfehler) durch Fremdgeräte oder Unterbrechungen der Datenübertragung bedingt durch Umbuchen in andere Funkzellen, oder kurzzeitige Überlastung des Kanals.
- hohe Latenzzeiten (im Sekundenbereich).

Es ist daher einsichtig, daß die im Internet verwendeten Standardprotokolle (z.B. TCP, UDP, IP) [21] nicht bis zum WUA verwendet werden können. Als Beispiel ist das *Transmission Control Protocol over IP* (TCP/IP) zu nennen, dessen Algorithmen auf eine faire Lastverteilung über eine konstant bleibende Gesamtbandbreite optimiert sind und in einern

Umgebung mit stark schwankenden Latenz- und Bandbreiten völlig versagen. Ebenso ist das im Web vorherrschende *Hyper Text Transfer Protocol* (HTTP)[6] für mobile Bedürfnisse ungeeignet, da der Overhead zu groß ist und es keinen *Session Support*, d.h. die Zuordnung der Dokumentanfragen zu einer Browsersitzung, zum Beispiel für die Verwaltung eines Warenkorbs, bietet (Kapitel 2.3.1).

Auch für die M- Commerce Applikationen selbst gelten spezielle Anforderungen:

- Möglichst einfache Nutzung, d.h. sie sollten völlig ohne Sachkenntnis bedienbar und ablenkungsfrei benutzbar sein, z.B. während des Telefonierens oder der Arbeit.
- Optimiert für die technischen Grenzen der derzeit gängigen Endgeräte und Übertragungswege. Dabei gilt es eine Vielzahl verschiedenster Endgeräte mit unterschiedlichen Hardwarelimitierungen und Features (wie z.B. Graustufen- oder hochauflösende Farbdisplays) optimal zu unterstützen.
- Kooperation und Integration der parallelen Dienste, die WUA üblicherweise unterstützen, wie Telefonieren, Versenden von Kurzmitteilungen (*Short Message Service*) oder der *Voice-Mailbox*.

Aufgrund dieser Anforderungen haben sich die letzten Jahre eine fast unüberschaubare Vielfalt von Umsetzungen für M-Commerce Applikationen hervorgetan, deren wichtigsten Vertreter kurz in Kapitel 1.3 angesprochen werden.

1.2 Übersicht

Im Folgenden wird sich dieses Seminar auf die Betrachtung von interessanten M-Commerce Techniken konzentrieren. Es ist dabei unmöglich, einen umfassenden Überblick zu gewähren, ohne den gegebenen Rahmen nicht völlig zu sprengen. Deswegen werden wir die wichtigsten Vorläufer und Konkurrenzentwicklungen zu dem *Wireless Application Protocol* (WAP) [4] vorstellen und in Kapitel 2 das Protokoll selbst zu beschreiben. In Kapitel 3 betrachten wir *Bluetooth*[16], eine Funktechnik für kurze Übertragungsstrecken (weniger als 10 Meter).

1.3 Ein Blick über den Tellerrand

1.3.1 i-Mode

i-Mode ist eine in Japan mit etwa 60% Marktanteil[1] (Abbildung 1) sehr verbreitete Architektur[1], die es ermöglicht, praktisch unangepasste Standard-Internetinhalte auf mobilen Endgeräten darzustellen. Die Technik geht auf eine proprietäre Entwicklung der japanischen Firma NTT DoCoMo [9] auf Basis der paketorientierten Datenübertragung mittels des *Pacific Digital Cellular System* (PDC) (etwa mit dem *Global System for Mobile communication* (GSM)[20] zu vergleichen) zurück. Interessant ist dabei, dass auf eine Untermenge von HTML3/4 (sogar inklusive Frames, Tabellen und Formulare), genannt *compact HTML (*cHTML)[22] als Grundlage für die Seitendarstellung aufgebaut wird. Als Grafikformate werden JPEG und (animated) GIF zugelassen.

Der Hauptbestandteil dieser Architektur ist das *I-Mode-Center*, das als Gateway zwischen dem Mobilfunknetz und dem Internet sowie diverser direkter Standleitungen zu z.B. Banken fungiert.

Als Hauptvorteil dieses Systems ist zu sehen, dass Standard-Internetseiten verwendet werden können und deshalb nicht wie bei WML (WAP) die Seiten neu erstellt und parallel gewartet werden müssen, man zudem auch nicht auf mobile Nutzung vorbereitete Webseiten besuchen kann.

Allerdings ist eine effiziente Nutzung allein aufgrund des Datenaufkommens und der schlechten Navigierbarkeit der üblicherweise recht großen Seiten auf den PDAs wohl in

[1]neben „J-Phone“ [23] und „KDDI“ [24], die vergleichbare Techniken auch für ihre Mobilfunknetze anbieten.

der Praxis kaum gegeben, da die üblichen Webapplikationen für die Darstellung in einem Browserfenster mit mindestens 800x600 Pixeln entworfen werden und daher die Schriften und Bilder für die niedrigauflösenden PDAs viel zu gross sind und eine entsprechende Skalierung erfordern. Da allerdings immer mehr Internetseiten „pixelgenau" spezifiziert werden (d.h. die Größen der Textspalten in Pixelangaben fest vom Webdesigner vorgeschrieben werden), führt dies zu einem schlechteren Benutzbarkeit der Seiten auf den PDAs. Deswegen entstehen auch hier vermehrt Internetseiten, die doch ganz speziell auf die Bedürfnisse von iMode-Webbrowsern zugeschnitten werden [10].

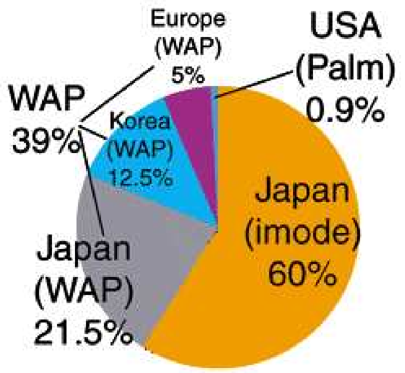


Abbildung 1: Mobile Commerce Techniken im Vergleich

Bemerkenswert ist, daß bereits über 25 Millionen (Juli 2001)[9] Nutzer diesen Dienst verwenden, was im Vergleich zu Europa oder den USA eine ausserordentlich hohe Zahl darstellt. Ein Grund stellt mit Sicherheit die volumenorientierte Kostenabrechnung dar. Dabei ist die Geschwindigkeit des PDC Netzes ist dabei mit 9600 Bps gleich dem des GSM Netzes.

Obwohl es sich um ein nicht-Standardisiertes System handelt, wird seitens des Mobilfunkunternehmen versucht, mit Erweiterungen wie z.B. kJava (eine ausserordentlich kleine Implementierung einer Java Virtual Machine für mobile Endgeräte, siehe auch 2.3.2 auf Seite 12) eine weitgehende Plattformunabhängigkeit zu erreichen.

1.3.2 LEAP

Das *Lightweight and Efficient Application Protocol* (LEAP)[25] wurde ganz bewußt als Gegenentwuf zu WAP entwickelt, da dieses nach Ansicht der LEAP-Entwickler mit nicht-tolerablen Schwächen und Design-Fehlern behaftet ist (Patentisierung, Umgehung des RFC-Schemas für die Etablierung Internet-relevanter Protokolle, etc.) [2]. Anstatt der äquivalenten Umsetzung der üblichen Internetprotokolle für das Mobilfunknetz, wie dies bei WAP/WAE zu finden ist, hat man hier einen effizienten Transportmechanismus auf Basis von IPs, dem *Efficient Short Remote Operations* (ESRO) entworfen, der als zuverlässiger, aber verbindungsloser Transportmechanismus fungiert und damit ein Mittelweg zwischen dem zu umfangreichen und für mobile Anwendungen unangepassten TCP und dem unzuverlässigen *User Datagram Protocol* (UDP) darstellt. Auf Basis dieses Transportlayers werden verschiedene Dienste, wie *Efficient Mail Submission and Delivery* (EMSD) und *Efficient Hyper Text Delivery* (EHTD) definiert. Der Schwerpunkt liegt vor allem im Bereich Messaging, d.h. Verwaltung und Versenden von Email- oder ähnlichen Nachrichten, da dies nach Ansicht der Entwickler im Gegensatz zum Web-Browsing ein sinnvollerer Dienst darstellt. Es ist allerdings noch keine praktische Umsetzung dieser Protokolle bekannt.

Aufgrund der fehlenden praktischen Umsetzung und unfertigen Spezifikation des Protokolls muss jedoch die Tauglichkeit dieser Architektur kritisch betrachtet werden.

1.3.3 HDML, TTML, ITTP, VoiceXML

Handheld Device Markup Language (HDML) und *Intelligent Terminal Transfer Protocol* (ITTP) stellen chronologische Vorgänger des WAE/WAP Standards, entwickelt von Unwired Planet (mittlerweile umbenannt in Phone.com) [26]und Ericsson[27] dar. Die Entwicklung dieser beiden Architekturen wurden zugunsten von WAP eingestellt. Es sind jedoch von Phone.com immernoch Browser für diese Architekturen erhältlich; es ist also denkbar, dass bei fehlendem Erfolg von WAP die Entwicklung dieser alternativen Protokolle und Spezifikationen neuen Aufschwung erhalten. Die Idee und die Syntax von HDML ist dabei beinahe identisch zu WAP/WML (Abbildung 2):

```
<HDML VERSION=3.0>
  <DISPLAY NAME=jan>
     Willkommen auf meiner HDML Seite..<BR>
    <ACTION TASK=GO TYPE=ACCEPT
   LABEL=Click TASK=GO DEST=#eis>
  </DISPLAY>
  <CHOICE NAME=eis KEY=Auswahl>
    <ACTION TYPE=ACCEPT TASK=GO DEST=#kommentar>
      Welches Eis magst du?
    <CE VALUE="Erdbeere">Erdbeere
    <CE VALUE="Schoko">Schokolade
    <CE VALUE="Vanille">Vanille
  </CHOICE>
  <DISPLAY>
    Wie, du magst auch $Auswahl? So ein Zufall!<BR>
  </DISPLAY>
</HDML>
```

Abbildung 2: HDML Beispiel

Die Bedeutung der Tags wird nach Betrachtung der WAP Syntax (siehe 2.3.2 auf Seite 12) leicht verständlich.

Tagged Text Markup Language (TTML) stellt auf Basis des 1997 von Nokia vorgestelltem Smart-Messaging Konzept (heute *Short Message Service*, SMS) eine vornehmlich von Nokia spezifizierte Markierungssprache dar, die für das Versenden von Internetinhalten an GSM-Mobiltelefone gedacht war. Die gängigen SMS-Diensten verwenden dieses Format allerdings nicht mehr, da sie ohne die Nutzung des Internets ihre Informationen bereitstellen.

Voice Extensible Markup Language (VoiceXML)[28] stellt eine XML-basierte Sprachspezifikation der Firma Motorola dar, die zum sprachgesteuerten Abruf von Internetinhalten über Mobiltelefone gedacht ist. Dabei soll mit Hilfe einfacher Sprachkommandos der Abruf von Internetseiten gesteuert werden, die im VoiceXML Format vorliegen müssen. Der Inhalt der Seiten wird dann von einem Sprachsynthetisierer in einem Gateway dem eingewählten Mobiltelefon „vorgelesen“, gedacht für Kontoauszüge, Banktransaktionen und ähnliches. Ein kleines Beispiel für VoiceXML [15] ist in Abbildung 3 zu finden.

Der Sprachsynthesizer liest dabei den <*prompt*>-Block vor, wobei der <*grammar*>-Block spezifiziert, welche Antworten vom Benutzer in welcher Form zu akzeptieren sind. Beim Akzeptieren der Benutzerantwort durch die Grammatik wird die Antwort an den Server via dies <*submit*> -Tags übermittelt, der daraufhin mit dem Dialog fortfahren kann.

Die Entwicklungsumgebung von Motorola stellt auch Gateway-Funktionalität für HTML und WML Inhalte zu Verfügung. Man hofft, daß man mit der Prägung des Begriffs *Voice-Commerce* (V-Commerce) und der Weiterentwicklung der Spracherzeugungstechnologie zusammen mit VoiceXML einen Industriestandard setzen kann. Dabei ist die Benutzung dieser Technologie wohlgemerkt nicht auf mobile Endgeräte beschränkt, sondern auch für stationäres Telefonbanking, sprachgesteuerte Museumsführer etc denkbar.

1.3.4 MExE, J2ME

Das *Mobile Station Application Execution Environment* (MExE)[29] ist ein Protokoll für Funkübertragung und für GSM- und Universal Mobile Telecommunication Systems (UMTS)[30]-basierte Wireless User Agents gedacht. Während die bisherigen Techniken

```
<?xml version="1.0"?>
<vxml version="1.0">
<form>
  <field name="drink">
    <prompt>Would you like coffee, tea,
     milk, or nothing?</prompt>
    <grammar src="drink.gram"
      type="application/x-jsgf"/>
  </field>
  <block>
    <submit next=
     "http://www.drink.example/drink2.asp"/>
  </block>
</form>
</vxml>
```

Abbildung 3: VoiceXML Beispiel

wie auf WAP vornehmlich für die Darstellungen von Informationen durch Übertragung von generierten Seiten gedacht ist, ist MExE in der Lage, vollständige Programme auf den mobilen Endgeräten auszuführen sowie die Unterstützung weiterer Mensch-Maschine Schnittstellen, wie zum Beispiel Spracherkennung.

Dazu ist ein vollständiges und sehr strenges Sicherheitskonzept notwendig, das in den gängigen mobilen Endgeräten noch nicht integriert werden kann, da z.B. für die *Java 2 Platform Micro Edition* (J2ME)[31] von Sun Microsystems die meisten aktuellen Endgeräten noch nicht genügend Ressourcen bereitstellen können.

Das MExE Protokoll unterstützt sogenannte „Classmarks“, mit denen das Endgerät seine Möglichkeiten und Einschränkungen kategorisieren kann, um dem MExE Server die Auswahl für das Endgerät geeigneter Applikationen zu ermöglichen.

J2ME ist eine kompakte Implementierung der *Java Virtual Machine* (JVM), die für ressourcenbeschränkte Geräte gedacht ist. Sie stellt dabei nur minimale Funktionalität zur Verfügung, die durch Herunterladen von zusätzlichen, vom Benutzer auswählbaren Funktionen erweitert werden kann.

Die eigentliche Java Virtual Machine, die sogenannte *Kilobyte Java Engine* (kJava) verwendet nur etwa 128kB an Speicherressourcen, versucht jedoch eine weitgehende Kompatibilität zu der Umgebung anderer Java-Applets für herkömmliche Webbrowser bereitzustellen. Die Firma Sun sieht den Hauptvorteil ihrerer Technik in der Flexibilität und dem erweiterbaren und änderbaren Funktionsumfang, da die Microbrowser für WAP oder ähnliche Techniken statisch implementiert sind und häufig nicht ausgetauscht oder erneuert werden können.

In wie weit Anbieter auf diese Konzepte eingehen ist noch nicht abzusehen. Zumindest ist die Idee einer frei erweiterbaren und flexiblen Plattform sicherlich verlockend, wenn auch das Hauptargument bei den heutigen Produktzyklenlänge von 9 bis 12 Monaten für ein mobiles Endgerät keine entscheidende Einschränkung darstellen kann.

1.4 Warum WAP?

Betrachtet man die bereits vor dem *Wireless Application Protocol* (WAP)[4] existierenden Lösungen, so stellt man fest, daß sie allesamt inkompatibel zueinander sind und hauptsächlich auf die Bedürfnisse und Möglichkeiten der Endgeräte des jeweiligen Herstellers bzw. des verwendeten Funknetzes abgestimmt waren. Daher ist es erklärbar, dass keine der älteren Techniken, wie HDML, ITTP und TTML sich durchsetzen konnten, zudem auch keine der Spezifikationen den Umfang und die Komplettheit des WAP-Pakets erreichen. Zudem gab es auch keine Unterstützung für Scriptsprachen, wie sie bei WAP mit WMLScript Einzug gehalten hat.

WAP hat den Vorteil, dass es in seiner Spezifikation vollständig auf die Bedürfnisse von mobilen Anwendungen aufbaut und weitgehend vom eigentlichen Trägermedium, d.h. der zugrundeliegenden Funknetztechnologie abstrahiert. Das WAP-Forum sieht dies, zusammen mit der Standardisierung und herstellerseitig weitgehend akzeptierten Schnittstellen als entscheidenden Vorteil zur Entwicklung in Richtung von WAP an.

Allerdings ist zu bemerken, daß die Flexibilität von WAP bezüglich der Erweiterbarkeit recht starre Grenzen besitzt, da zum Beispiel nicht vorgesehen ist, die eingesetzten Microbrowser nach Auslieferung erweitern zu können, wie z.B. via Java-Beans in J2ME oder MExE möglich ist. Dies setzt also eine weitgehend komplette und fehlerfreie Umsetzung der Microbrowser bei der Auslieferung des Endgerätes voraus, was natürlich die Entwicklung von Anwendungen vereinfacht (da nur relativ wenige Variationen der clientseitig verfügbaren Features zu beachten ist). Die Praxis zeigte jedoch, daß der am weitesten verbreitete Microbrowser von Phone.com, den im Prinzip jedes heutzutage erhältliche Mobiltelefon verwendet, zumindest in der Anfangszeit nicht mit WAP und den spezifizierten WAP-Gateways kompatibel war sowie ganz erhebliche Fehler (Bugs) besaß. Es bleibt dabei zu bemerken, daß auch heute die Phone.com Microbrowser und Gatewaysoftware proprietäre Erweiterungen und zur WAP Spezifikation inkompatible Änderungen besitzen, es also mit der auf dem Papier spezifizierten XML/WAP Conformance in der Praxis nicht allzu gut bestellt ist. Zudem ist die Spezifikation zwar äußerst umfangreich und kaum zu überblicken, jedoch in entscheidenden Punkten nicht detailliert genug, um nur auf dieser Basis einen Microbrowser mit Umgebung zu entwickeln.

2 Wireless Application Protocol (WAP)

2.1 Übersicht und historischer Hintergrund

Die Entwicklung vom *Wireless Application Protocol* (WAP)[4], begann Mitte 1997 als Zusammenschluss von Nokia, Ericsson, Motorola und der neuen Firma Phone.com (ehemals Unwired Planet) mit dem Ziel, einen Herstellerübergreifenenden Standard für die Darstellung und Verarbeitung von Internetinhalten auf mobilen Geräten, hauptsächlich Mobiltelefonen zu definieren. Die Grundlage waren, wie bereits angedeutet, proprietäre Eigenentwicklungen der Hersteller, wie z.B. TTML von Nokia oder HDML[32] von Phone.com.

Zur Zeit ist die WAP Spezifikation Version 1.2 aktuell, die gegenüber ihren Vorgängern um einer Server-Push Funktionalität, d.h. daß der Server bei bestimmten Seiten Updates ohne vorhergehende Anfrage vom Client senden kann, und um Möglichkeiten, die Fähigkeiten des Endgerätes dem Internetserver mitzuteilen, erweitert wurden. Da alle namhaften Firmen der Endgerätehersteller und Mobilfunkbetreiber Mitglied im WAP-Forum sind, darf die Weiterentwicklung und Etablierung des Pakets als Industriestandard eigentlich als gesichert gelten.

2.2 Anforderung an WAP

WAP als Transportschicht und Umgebung und die XML basierte Seitenbeschreibungssprache WML haben eine Vielzahl von Anforderungen zu erfüllen:

- Es sollen beliebige Inhalte des Internets zugreifbar sein. Dies schliesst aus, daß es einen einzelnen Diensteanbieter gibt. Stattdessen ist es nötig, das breite bereits existierende (oder damals in der Entstehung befindliche) E-Commerce Angebot auf den Handheld zu portieren, d.h. es sollte eine Möglichkeit geben, den „Inhalt“ von konventionellen HTML Seiten zur das Mobilgerät zugreifbar zu machen. Aufgrund der limitierten Anzahl freier IP’s (IPv4) ist dabei die auf dem Desktop verbreitete

PPP/TCP/IP basierte Technik wie auch aus den limitierten Ressourcen der Mobilfunktelefonen nicht möglich.

- Kompatibel und optimiert auf die Anforderungen von mobilen Geräten, das bedeutet kleine Displays, limitierte Eingabemöglichkeiten (keine 102-Tasten DIN-Standardtastatur), eingeschränkte Bandbreite im Zugriff auf Internetseiten aufgrund der drahtlosen Übertragung, eingeschränkte Rechenleistung der Geräte sowie eine meist geringe Batterielaufzeit.
- Als Hersteller- und Länderübergreifender Standard muss es von der physikalischen Ebene, d.h. der eingesetzten Mobilfunktechnik unabhängig sein. WAP definiert nicht den physikalischen Transport der Daten, dieser kann z.B. via konventionellen SMS Diagrammen (z.B. früher in der Schweiz und den USA üblich) als auch über GSM-Modems oder digitalen Übertragungstechniken (z.B. UMTS) erfolgen. Auch Bluetooth ist als Übertragungsmedium für lokale Anwendungen denkbar.
- Aufgrund der schlechten Erfahrungen mit von Herstellern häufig durch undokumentierte Eigenentwicklungen gesetzten „Quasi-Standards“, war es ein Ziel der WAP Entwicklung, frühzeitig mit einer vollständigen und umfassenden Spezifikation die Freiheit und Herstellerunabhängigkeit zu gewährleisten. Ironischerweise wird gerade dieser praxisunerprobte Entwurf auf dem Papier häufig als einer der Hauptargumente gegen WAP ins Feld geführt.

Da zwar die Internet-Inhalte transportiert werden sollen, aber aus den dargestellten Gründen die üblichen Standards wie HTML, HTTP, TLS (ehemals SSL), TCP und IP für die schmalbandigen Funknetze zu ineffizient sind, wurden in der WAP Architektur angepasste Pendants zu den bekannten ISO Layern des Internets entwickelt, die im folgenden näher betrachtet werden.

2.3 WAP Architektur

Zentraler Bestandteil der WAP-Architektur ist das WAP-Gateway (Abbildung 4). Es bildet im Prinzip die Funktionalität eines WWW Proxy-Servers nach, fungiert aber auch als Übersetzer und Vermittler zwischen den beiden Welten, dem Mobilfunknetz mit dem mobilen Endgerät und dem Internet mit den Web-Dienste- und Inhalteanbietern. Auf der Seite des WUA wird der WAP-Protokollstack verwendet, der dann im Gateway auf die üblichen Internetprotokolle transparent umgesetzt wird, so daß der Internet Server nur die üblichen Protokolle unterstützen muss und keine spezielle Erweiterung für die WAP-Endgeräte benötigt.

Das Gateway benötigt neben den Unterstützungen für die Protokolle beider Welten auch weitergehende Management Funktionen, wie das Koordinieren von Sessions zwischen beiden Welten, dem Konvertieren der aus dem Internet in ASCII Text gelieferten WML Dateien in ein effizienteres Binärformat für die Übertragung im Mobilfunknetz, oder aus Ressourcengründen nicht im Endgerät implementierbaren Funktionalitäten, wie Bookmarklisten oder HTTP Cookies, die auch über Verbindungsabbrüche und erneute Einwahlen hinweg persistent abgespeichert werden müssen.

Es ist sogar vorgesehen, daß das WAP-Gateway nur in HTML verfügbare Inhalte in binäres WML für das mobile Endgerät umsetzen kann. Inwieweit eine solche Funktionalität in der Praxis in brauchbarer Weise für eine automatische Konverierung implementiert werden kann ist eher fraglich.

2.3.1 Der WAP-Protokollstack

Die WAP-Schichten lehnen sich sehr stark dem ISO/OSI-7-Schichtenmodell an und werden im Folgenden näher erläutert (Abbildung 5). Alle Protokollspezifikationen sind in einem etwa 1000 Seiten langen Spezifikation bei [4] erhältlich.

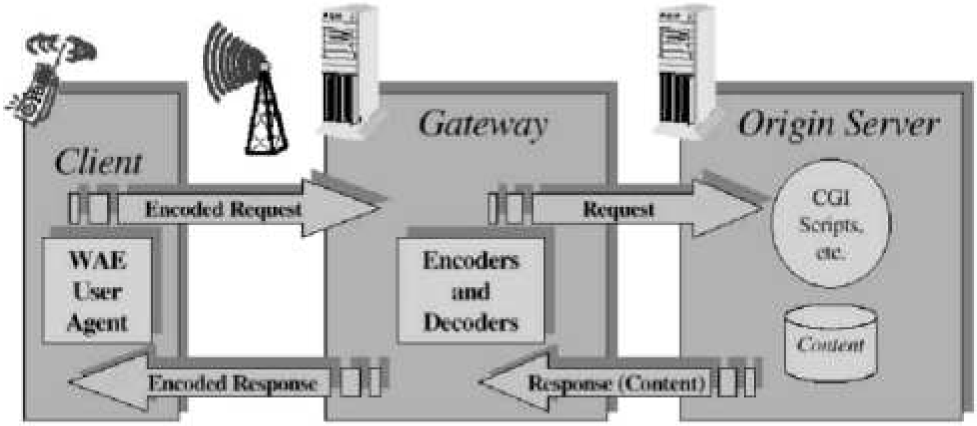


Abbildung 4: WAP Entwurfsmodell

- *Wireless Application Environment* (WAE) stellt die Applikationsschicht dar. Die Hauptaufgabe ist die Bereitstellung von Diensten, die über alle Datenstandards der mobilen Kommunikation portierbar sind. Hauptsächlicher Bestandteil ist der Microbrowser, der in WML basierten Internetinhalten mit Hilfe des WAP-Gateways navigieren kann. Es existieren jedoch keine wesentlichen Einschränkungen in der WAP-Spezifikation gegenüber der unterstützbaren Applikationen.

- *Wireless Session Protocol* (WSP) setzt die Sitzungsschicht um. Das WSP bietet dem WAE-Level zwei verschiedene Sitzungsdienste an:
  Der *Connection Mode*, der über das *Wireless Transaction Protoco*l (s.u.) dem Anfragenden und dem Diensteanbieter „Request" und „Response" Funktionalität zur Verfügung stellt, d.h. damit sollte der Anfragende unter normalen Umständen eine Bestätigung seiner Anfrage erhalten (und natürlich die Zuordnung zu der Anfrage erlauben).
  Der *Connectionless Mode*, der über das *Wireless Datagram Protocol* erfolgt, enthält keine Bestätigung für eine Nachricht, weder für die Anfragende noch für die Diensteanbieter-Seite der Kommunikation, stellt also ein unzuverlässiges Protokoll dar. Das WSP ist auf der Internetseite mit dem HTTP/1.1 Protokoll am ehesten zu vergleichen, bietet jedoch einige Erweiterungen, wie z.B. die Möglichkeit Transaktionen zu unterbrechen (suspendieren) und zu einem späteren Zeitpunkt wieder aufzunehmen, einem Support für Server-Push von Daten oder für gepipelinete asynchrone Anfragen an den Server (eine Liste von Anfragen die nicht einzeln bestätigt werden müssen).
  Getreu dem Konzept für mobile Datenübertragung implementiert WSP eine Komprimierung zu übertragender Headerinformationen. Dabei werden bekannte Header-Feldnamen durch kompakte binäre Kodierungen ersetzt sowie unveränderte Headereinträge gegenüber vorhergehenden Anfragen durch Header-Caching komplett aus den Anfragen gestrichen, was gegenüber HTTP eine Reduktion des Overheads von über 90% erreichen kann.

- Das *Wireless Transaction Protocol* (WTP) implementiert die Transaktionsschicht. Eine Transaktion ist dabei die Kommunikation zwischen einem Initiator einer Anfrage und einem Antwortenden.
  Das WTP ermöglicht verschiedene Varianten von Transaktionen: unzuverlässige Einweganfragen, die nicht vom Empfänger bestätigt werden müssen, zuverlässige Einweganfragen, die einer Bestätigung durch den Antwortenden über den Empfang bedürfen und der zuverlässigen Zweiwegeanfragen, bei der auch die Rückantwort durch den Initiator wiederrum bestätigt werden muss, bevor mit der Kommunikati-

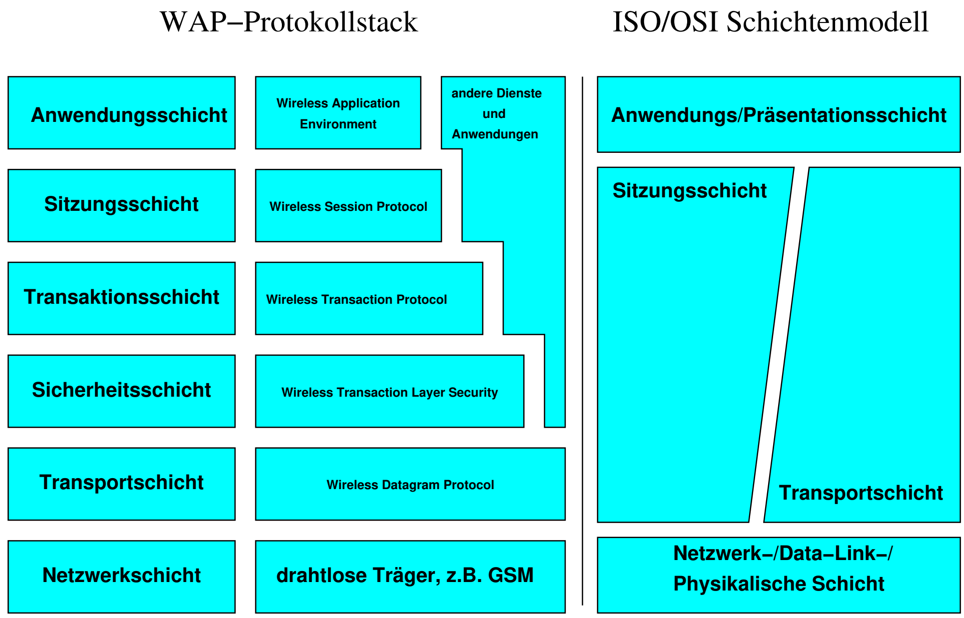


Abbildung 5: Aufbau des WAP-Stack

on fortgefahren werden kann.
Diese Schicht besitzt auch einige Optimierungen, wie zum Beispiel das Konkatenieren von Datenpaketen, um den Overhead zu minimieren oder die Möglichkeit, in Bestätigungspaketen Zusatzinformationen über die gemessene Übertragungsleistung und ähnlichem der Gegenstelle mitzuteilen.

- Das *Wireless Transport Layer Security Protoco*l (WTLS) bildet die optionale Sicherheitsschicht und ist in etwa mit der *Socket Layer Security* (SSL) bzw *Transaction Layer Security* (TLS) auf der Internetseite zu vergleichen. Die Aufgabe des Protokolls ist die Etablierung von sicherem Datenaustausch und der dementsprechend benötigten Sicherheitstechniken wie Authentifizierung, Verschlüsselung und Datenintegrität. Das Handshaking ist dabei an die speziellen Bedürfnisse der mobilen Kommunikation angepasst, basiert jedoch in grossen Teilen auf der TLS Spezifikation.
  Für die Datenübertragung wird 35 Bit DES als symmetrische Verschlüsselung verwendet, zum Handshake ein 512Bit Diffie-Hellmann Verfahren. Dies ist eine erhebliche Reduzierung der Datensicherheit, die hauptsächlich darauf zurückzuführen ist, dass jede Verschlüsselungsoperation Prozesserlast erzeugt und damit in einem erhöhten Stromverbrauch resultiert. Desweiteren wurden bei der Anpassung der TLS Spezifikation auf die Bedürfnisse der mobilen Kommunikation nach Ansicht von Kryptographieexperten Fehler gemacht, die eine Attacke auf den verschlüsselten Datenstrom besonders einfach machen. Näheres dazu findet sich in [11].

- Das *Wireless Datagram Protocol* (WDP) stellt als Basisschicht des WAP-Stacks eine Umsetzung der Transportschicht dar. Als unzuverlässiges paketorientiertes Protkoll entspricht es in etwa dem Universal Datagram Protocol (UDP over IP) auf der Internetseite, ist jedoch auf die speziellen Bedürfnisse der mobilen Kommunikation, niedriege und stark schwankende Bandbreiten und hohe Latenzzeiten, optimiert.

- Die Netzwerkschicht ist in der WAP Spezifikation nicht festgelegt, und kann völlig transparent zu den darüberliegenden Schichten ausgetauscht werden. Üblich und möglich sind gängige digitale Mobilfunkübertragungsstandards wie GSM oder gar SMS, der Short Message Service.

Zu jeder dieser Protokoll-Schichten ist in WAP eine Management-Einheit spezifiziert, die Zustandsänderungen, wie das Verlassen des Funkbereiches, dem Verlust an Akku-Leistung, einem Speicherüberlauf und ähnlichem erkennen kann und diese der zugeordneten Schicht mitteilen kann, um ein korrektes Funktionieren der jeweiligen Protokollschicht auch unter schwierigen Bedingungen erfüllen zu können.

Als Abschluss soll ein kleiner Vergleich von HTTP/TCP/IP und WSP/WTP/WDP bei einer einfachen Anfrage einer HTML/WML Seite das umgesetzte Optimierungspotential verdeutlichen:

	HTTP/TCP/IP	WSP/WTP/WDP
Anzahl übertragener Pakete	17	7
Overhead gegenüber Nutzdaten	65%	14%
Anzahl Round-Trips	10	3
Round-Trips bevor Datenübertragung beginnt	3	1

2.3.2 Wireless Markup Language

Die *Wireless Markup Language* (WML)[4] ist im Gegensatz zu ihrem Internet-Pendant HTML in XML[33] spezifiziert, erfordert dementsprechend also eine „wellformed“ Dokumentenstruktur. Neben diesen Anforderungen an die Seitenbeschreibung benötigt man ein gutes Verständnis für die Ristriktionen der Microbrowser. Es ist zu bemerken, daß das WML Dokument auf dem Gateway in eine binäre Repräsentation für die Übertragung zum Microbrowser umgewandelt wird, dem sogenannten *Wireless Binary XML* (WBXML)[34]. Auf diese Weise kann der Umfang auf etwa 20% der ASCII-Version reduziert werden. Gängige Microbrowser können binäre WML Dokumente bis zu einer Größe von etwa 1400 Bytes verarbeiten.

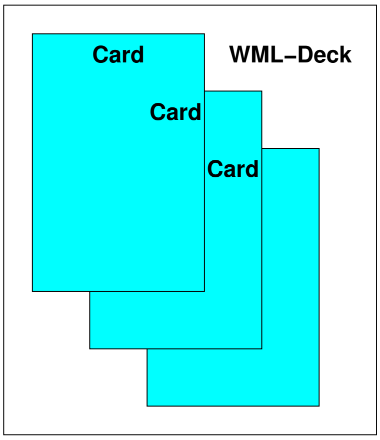


Abbildung 6: Cards und Decks in WML

Das Basiselement eines WML Dokumentes ist die sogenannte Card. Mehrere Cards können in einem Dokument zu einem Deck (also sozusagen einem Stapel von Cards) zusammengefasst werden (Abbildung 6), welches die oberste Struktureinheit eines WML Dokumentes darstellt. Der Microbrowser zeigt beim Laden das Startdeck (üblicherweise die erste Card) an. Es kann durch Aktivieren von Links oder durch andere Events, wie zum Beispiel durch Ablauf ein Timers auf andere Cards umgeschaltet werden oder das Deck durch ein anderes WML Dokument ersetzt werden.

WML bietet grundlegende Beschreibungstags für Hypertext sowie auch einige erweitere Konstrukte wie z.B. Pendants zu den Formularen bei HTML. Ein WML-Beispiel ist in Abbildung 7 zu finden.

```
<?xml version="1.0" ?>
<!DOCTYPE wml PUBLIC "-//WAPFORUM//DTD WML 1.2//EN"
  "http://www.wapforum.org/DTD/wml_1.2.xml">
<wml>
  <card id="cardOne" ontimer="#cardTwo"
   title="WML Beispiel">
    <timer value="500" />
    <p align='center'>Hallo Welt!</p>
  </card>
  <card id="cardTwo" title="Login">
  <p>
    Name: <input name="username"/><br/>
  </p>
  <do type="accept">
    <go href="#main"/>
  </do>
  </card>
  <card id="main">
    <p align='center'>
      Willkommen <b>$(username)!</b><br/>
    </p>
    <do type="accept">
      <go url="wtai:wp/mc;112" />
    </do>
  </card>
</wml>
```

Abbildung 7: WML Beispiel

Neben dem obligatorischen XML und DOCTYPE Kopf der Datei beginnt das Dokument mit dem <*wml*> Tag, in das <*card*> Blöcke geschachtelt werden dürfen. Mit dem <*timer*> Tag wird nach 500ms ein Timer-Event ausgelöst, das durch das Attribut im <*card*>-Tag auf die zweite Card verweist. Im zweiten Tag kann wie bei HTML eine Formulareingabe durchgeführt werden. Als spezielle Eigenheit sind diese Eingaben direkt über die Variablen, wie sie in dem dritten Card zu sehen sind, referenzierbar. Der <*do*> Block stellt einen allgemeinen Mechanismus zur Ausführung von Aktionen durch den Benutzer zur Verfügung; es existieren einige Elemente um in dem bisherigen Verlauf vor- oder zurückzuspringen oder auch um dem Benutzer eine Auswahl von möglichen Aktionen zur Verfügung zu stellen. Eine dieser Aktionen ist der <*go*> Tag, der in diesem Beispiel eine Verzweigung auf die dritte Card des Decks erlaubt. Im dritten Card ist als Aktion eine Funktion der *Wireless Telephony Application Interface* (WTAI)[35] verwendet worden; hier wird ein Telefonaufruf der Telefonnummer 112 initiiert.

WML erlaubt ebenfalls die Darstellung von Bildern auf den *Wireless User Agents* (WUA). Zu diesem Zweck wurde das *Wireless Bitmap* (WBMP) Format spezifiziert. Der Aufbau des Formats ist aussergewöhnlich einfach, der Header enthält neben der Identifikation mit einer Versionsnummer nur eine Bildgrößeninformation und den WBMP-Typ. Der einzige zur Zeit spezifizierte WBMP-Typ ist Typ 0. Dieser erlaubt eine 1-Bit Farbtiefe ohne jegliche Kompression als Bildtyp.

2.3.3 WMLScript

Die WAP-Spezifikation sieht auch eine Unterstützung einer leicht abgespeckten Variante von ECMAScript, also der standardisierten Fassung von Netscape's Javascript, in den Microbrowsern vor. Auch hier wird der eigentliche Skriptcode in einer binären Repräsentation, die auch im WBXML Format spezifizert ist, übertragen. Der Sprachumfang erlaubt die Validierung von Formulareingaben, Passwortabfragen, spezifische Steuerungsmöglichkeiten und einfache Interaktion mit dem Benutzer oder der WAP Session.

2.4 Bewertung von WAP

Im Folgenden werden technische und organisatorische Details der Spezifikation im Detail stichpunktartig bewertet:

- Die WAP-Spezifikation kränkelt durch lahme und fehlerbehaftete Implementierung bzw. praktischer Umsetzung der durch das WAP-Forum beschlossenenen Standards. Die Software kann noch nicht als ausgereift gelten, zudem zerstören immer wieder die nicht-standardisierten Erweiterung des de-fakto einzigen Microbrowser-Lieferanten Phone.com gegenüber der Gateway-Software die Interopabilität gegenüber Phone.com-fremden Produkten. Dies wurde in Deutschland z.B. durch das E-Plus Gate im letzten Jahr besonders bewusst, dessen Software die HOMEURL-Anfrage der Phone.com Microbrowser nicht verstand und somit die Einwahl scheiterte[17].

- Die WAP Spezifikation ist in viele dutzend Teile zersplittet, die einer getrennten Versionskontrolle unterliegen. Insofern ist es bei den recht häufigen Änderungen in Teilen der Spezifikation kaum noch die Übersicht zu behalten, wie diese in Zusammenhang bestehen. Da die Gerätehersteller bzw. Microbrowser Produzenten in der Regel nicht auch die Gatewaysoftware herstellen bietet dies ein zusätzliches Potential für Fehler und Inkompatibilitäten. Zudem spezifiziert WAP keine Konformitätsüberprüfung der umgesetzten Funktionalität, sofern die Art der Funktionalitätsumsetzung überhaupt in der Spezifikation beschrieben wurde. Aus diesen Gründen verzichten Hersteller auf die Umsetzung unklar definierter Teile der Spezifikation, besonders im WML und WMLScript Teil. Es ist zu erwarten, daß dadurch in Zukunft die auf dem Desktop bereits üblichen Inkompatibilitätsprobleme auch in verschärftem Maße für WAP gelten werden.

- Das WAP-Gateway als zentrale Komponente unterliegt den vollständigen Kontrolle der Mobilfunknetzbetreibern. Da die Daten auf dem Gateway umgesetzt werden, d.h. also auch die Transport Layer Security auf dem Gateway gewandelt wird, hat zur Folge dass alle durch die Verschlüsselung eigentlich geschützten Daten im Klartext im Gateway verfügbar sind. Es ist zu bezweifeln, ob eine Gatewaysoftware auf Dauer vor potentiellen Hackerangriffen aus dem Internet geschützt werden kann. Desweiteren besitzt das Mobilfunknetz auch ein exaktes Nutzerprofil von WAP-Surfern sowie eine vollständige Kontrolle über die zu Diensteanbietern übermittelten Daten (z.B. Onlinebanking) und den dem Consumer präsentierten Inhalten (da der Mobilfunkbetreiber mit seinem Portal und der Startseite sowohl leitenden Einfluss als auch durch Ausschliessen von Webinhalten kontrollierenden Einfluß besitzt).
  Es gibt jedoch Ansätze, die Portalseiten für den Benutzer konfiguierbar zu gestalten. Desweiteren ist es denkbar, daß Banken eigene WAP-Zugänge anbieten oder es in Zukunft freie, d.h. Mobilfunkbetreiberunabhängige Gatewaybetreiber geben könnte.

- Zu globaler Kritik an WAP und der Arbeitsweise des WAP-Forums kommen einige Stellen im Internet, welches wie bereits erwähnt zu dem LEAP Gegenentwurf geführt hat. Zu Erwähnen ist hier der nicht unerhebliche Mitgliedsbeitrag für die Mitarbeit am WAP-Standard oder dem Patentisieren der verwendeten Verfahren. Als Kritikpunkt gilt dabei, daß das WAP-Forum nicht von Anfang an mit den bekannten

Organisationen zur Standardisierung von Internetprotokollen (W3C, ETSI) zusammengearbeitet hat.

- Es ist in Frage zu stellen, ob die starke Orientierung des WAP-Protokollstacks an dem ISO/OSI-Schichtenmodell zwingend notwendig gewesen ist. Es ist vorstellbar, daß man aufgrund der speziellen Umständen auch eine schlankere, einfachere Hierarchie hätte verwenden können, um eine leichtgewichtigere Implementierung zu ermöglichen. Desweiteren ist es fraglich, ob die Unterstützung von nicht-IP-basierten Trägernetzwerken über längere Zeiträume die richtige Entscheidung gewesen ist. Dies bedingte einen erheblichen Aufwand durch Reimplementierung der IP/UDP Schicht im WAP-Stack, ebenso hätten bei einer IP-Unterstützung die oberen Layer des Stacks schlanker ausfallen können.
  Auf der anderen Seite ist dieser Entscheidung jedoch aufgrund der verfügbaren Netztechnik und der immernoch nicht generell verbreiteten Unterstützung für IPv6 durchaus mehr als verständlich.

- Die Sicherheit des WAP Sicherheitsschicht ist, wie bereits erwähnt, ungenügend. Eine Anleitung zur Entschlüsselungsattacke ist in [11] zu finden.

- Da WAP doch nur ein sehr minimalistische Benutzerschnittstelle bietet und mit der notwendigen Anpassung der Internetinhalte auf das WAP-Protokoll dem Internet-Inhalteanbieter erheblichen Mehraufwand aufbürdet, ist es fraglich, ob nicht eine Spezifikation für leistungsfähigere Geräte einfacher gewesen und bei dem Konsumenten zu größerer Akzeptanz der Technik geführt hätte.

3 Bluetooth

3.1 Übersicht

Bluetooth, eine Funktechnologie im lizenzfreien 2.4GHz Industrial, Scientific and Medical (ISM) Band, ist dazu ausgelegt, die Vielzahl von Kabelverbindungen im PC- und Mobilbereich durch eine universale funkbasierte Verbindungstechnologie zu ersetzen und ist damit als direkte Konkurrenz zu den bereits verbreiteteten *Infrared Data Association* (IrDA)[36] bzw. IEEE 802.11 Produkten zu sehen. Da sie nur auf Kurzstreckenkommunikation ausgelegt ist, d.h. auf Reichweiten von typischerweise 10m kann sie nicht als Ersatz für herkömmliche Mobilfunktechnik wirken, sie besteht vielmehr neben ihr.

Der größte Vorteil von Bluetooth gegenüber vergleichbaren Entwicklungen sind die Kosten, die mit etwa 20 DM pro Einheit nur ein Zehntel der üblichen Kosten für IrDA oder IEEE 802.11 darstellen.[2] Erkauft wurde dies mit der limitierten Sendereichweite und, noch fataler, der geringen Übertragungsgeschwindigkeit. Bluetooth unterstützt nur 721Kbps unidirectional (57.6Kpbs im Rückkanal) oder 432.6Kbps bei symmetrischen Datentransfer pro Piconet, ist also deutlich langsamer als ähnliche drahtlose Übertragungstechniken (IEEE 802.11 unterstützt 11 MBit/s). Es ist jedoch möglich, die Bandbreiten mehrere Piconets zu verwenden, um doch eine höhere (bis 6MBit/s) Bandbreite zu erreichen.

Bluetooths entscheidender Vorteil ist die einfache Verwendbarkeit, gewährleistet durch automatischen Verbindungsaufbau und automatischem Suchen nach geeigenten Bluetooth-Gerätepartnern, sowie die Möglichkeit, parallel Daten- und Sprachverbindungen zu handhaben. Daraus ergeben sich Anwendungen, wie zum Beispiel ein Headset das automatisch die Anrufe auf dem Mobilfunktelefon annimmt und das Gespräch drahtlos überträgt, oder das automatische Synchronisieren des PDAs mit dem Arbeitsrechner, sobald sich beide Geräte in Reichweite befinden. Diese Ausarbeitung wird sich jedoch hauptsächlich mit den technischen Seiten von Bluetooth beschäftigen, da die Einsatzmöglichkeiten Gegenstand eines anderen Seminars sind.

[2] Dies ist eine angestrebte Angabe der Bluetooth Group, zur Zeit liegen die Preise bei etwa 45-50DM.

Die Bluetooth Technologie ist keinesfalls revolutionär neu, sie stellt viel mehr eine Kombination der Eigenheiten von IrDA, IEEE 802.11, *Digitally Enhanced Cordless Telecommunications* (DECT) und vor allem Motorola's Piano, von dem das Konzept des Aufbaus von „Ad-hoc Personal Area Networks" entlehnt ist, dar.

Die Möglicheiten von Bluetooth sind aufgrund der Vielfältigkeit der unterstützten Endgerätetypen fast unbegrenzt. Bluetooth eignet sich zur kabellosen Verbindung aller üblichen Haushalts- und Bürogeräte und bietet in Zusammenhang mit WUAs interessante neue Möglichkeiten, z.B. die Kommunikation mit dem Warenregal im Supermarkt, um automatisch dem Konsumenten die neuesten Angebote mitzuteilen oder das Begleichen der Rechnung an der Supermarkt-Kasse via Bluetooth und einem elektronischen Zahlungsmittel.

3.2 Netzstruktur

Die Netzstruktur eines Bluetooth Netzes basiert auf dem Master/Slave Prinzip und organisiert sich in Gruppen von bis zu 8 Endgeräten in sogenannten Piconets (wobei genau ein Gerät als Master agiert und bis zu sieben weitere als Slaves reagieren). Desweiteren können die Slaves in einem *Piconet* auch zeitgleich als Slaves oder gar als Master an genau einem weiteren Piconetzes teilnehmen, also als Brückengeräte (bridge devices) agieren, wobei sie allerdings nur in maximal einem Piconetz gleichzeitig Master sein dürfen. Diese Geräte verbinden demnach die räumlich überlappenden Piconetze zu einem „Scatternet" (Abbildung 8). Zur Zeit dürfen maximal 10 Piconets sich zu einem Scatternet zusammenschliessen.

Da das verwendete Frequenzband häufig mit Signalen von Geräten wie z.B. Garagentür-Öffner, Babyphone oder Mikrowellengeräte und ähnlichem belastet ist, ändert jedes Piconets häufig die aktuelle Sendefrequenz, um den Einfluß von Störungen durch Funkeinstrahlungen zu minimieren und das Abhören zu erschweren.

Endgeräte können als aktive Slaves eine der 7 möglichen Endgeräteadressen im Piconetz erhalten und aktiv an der Kommunikation teilnehmen, oder in der unbeschränkten Menge von „geparkten" Slaves dem Piconetz Master „lauschen", ohne jedoch selbst das Recht auf Kanalzugriff (d.h. sendend oder empfangend) zu haben. Dazwischen gibt es zwei Abstufungen, die hauptsächlich darauf ausgelegt sind, den Energieverbrauch für das Verfolgen der Piconetz-Kommunikation in den Endgeräten zu minimieren.

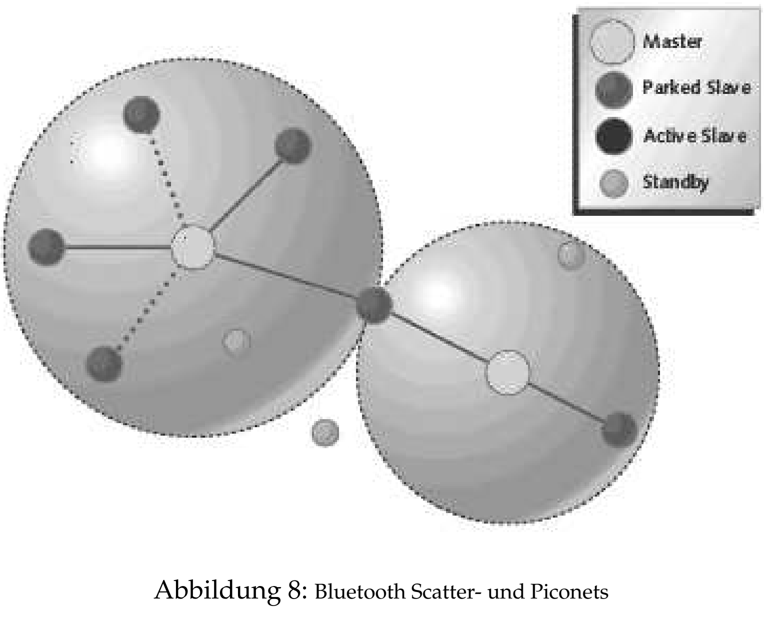


Abbildung 8: Bluetooth Scatter- und Piconets

Bluetooth verwendet zwei Verbindungstypen, *Synchronous Connection-Oriented* (SCO) und *Asynchronous Connectionless* (ACL) Links. Beide Linktypen können für Datenübertragungen verwendet werden, während Sprachverbindungen nur über SCO Verbindungen möglich sind. ACL Links werden vom Master für Broadcast-Nachrichten an die Slaves verwendet und dürfen von Slaves nur nach vorhergehender Aufforderung vom Master zur Datenübertragung verwendet werden, d.h. der Master pollt in vom jeweiligen Endgerät geforderten Abständen die Slaves und erlaubt ihnen dann für bestimmte Zeiträume eine Datenübertragung. Auf diese Weise läßt sich die von jedem Slave benötigte Bandbreite zuteilen und trotzdem sehr schnell auf verschlechterte Übertragungsbedingungen, die die verfügbare Gesamtbandbreite erheblich minimieren, entsprechend reagieren. Aufgrund der QoS-Unterstützung im LMP Protokoll sollte es in den meisten Fällen möglich sein, höher priorisierte Datenübertragungen auch unter ungünstigen Bedingungen noch aufrecht zu erhalten.

Synchrone Verbindungen erlauben den Slaves, nach festen Zeitschlitz-Zuteilungen im Piconet selbständig Datenübertragung zu initiieren und somit ohne die unkalkulierbaren Latenzzeiten beim Pollen durch den Master einen relativ gleichbleibenden Datendurchsatz zu erreichen, wie es für Sprachübertragungen entscheidend ist.

Jedes Endgerät hat die Möglichkeit, einen ACL Kanal, einen ACL und einen SCO Kanal oder maximal drei SCO Kanäle gleichzeitig zu verwenden, wobei sich jedoch der Kanaltyp einer Verbindung beinahe jederzeit ändern darf.

3.3 Funktionsweise im Detail

Es sind im Wesentlichen die folgenden Funktionsblöcke in einem Bluetooth Endgerät zu unterscheiden:

- Radio Unit, RF Sende/Empfangseinheit
- Link Control Unit für Low-Level Funktionalität in Hardware/Firmware
- Link Manager, High-Level Steuerung und Protokollstapel in Software implementiert.

Diese Komponenten werden im folgenden detaillierter betrachtet.

3.3.1 Radio Unit

Die Radio Unit realisiert den eigentlichen Radio Transceiver, der im ISM-Band operiert mit einer Sendeleistung von typischerweise 100mW, für eine typische Reichweite von bis zu 10 Metern. Es sind bis zu 100 Meter durch Erhöhen der Sendeleistung oder durch Etablieren eines Scatternets möglich.

3.3.2 Link Control Unit

Die **Link Control Unit** (Bluetooth Baseband) ist für das Senden und Empfangen der RF sowie ihrer digitalen Signalverarbeitung zuständig. Sie unterstützt den Verbinungsauf- und Abbau, asynchrone (Daten) und synchrone (Sprache) Übertragung, Fehlerkorrektur und Authentifizierung.

Bluetooth verwendet üblicherweise 79 Kanäle in dem ISM-Band in *Gaussian-Frequency-Shift Keying* (GFSK) Modulation zur Übertragung der digitalen Information. Die Frequenzmodulation macht die Verwendung eines effizienten Verstärkers möglich, welches den Stromverbrauch und die Verlustleistung senkt. Die Demodulation ist vergleichsweise einfach.

Als Übertragungsmodus wird das Time-Division-Duplex Verfahren (TDD) mit Timeslices von 625μs Länge verwendet, d.h. nach jedem übertragenen Paket in dem jeweiligen

Piconet wechseln alle beteiligten Funkmodule nach einem festgelegten und für das jeweilige Piconet spezifischen Frequenz-Hopping-Muster ihre Kanalfrequenz (d.h. maximal 1600 Frequenzhops pro Sekunde).

Bluetooth wechselt also viel häufiger als die anderen Technologien, die dieses Frequenzband mit verwenden, die Sendefrequenz. Dies erhöht zwar zum einen den Overhead an Verwaltungsdaten und zerstückelt die Nettodaten teilweise in sehr kleine Pakete, ist jedoch in stark gestörten Funkumgebungen von großem Vorteil, da dann nicht so häufig zerstörte Pakete erneut gesendet werden müssen. Es liegen zwar noch keine richtigen Praxiserfahrungen vor, aber Untersuchungen zufolge sollen die Geschwindigkeitseinbußen aufgrund von Funkstörungen unter 22 Prozent bleiben [12].

Zur Validierung der korrekten Übertragung wird das *Cyclic Redundancy Check* (CRC) und darauf aufbauend das *Automatic Repeat Request* (ARQ) Verfahren eingesetzt. Zusätzlich spezifiziert Bluetooth als optionalen Zusatz bei stark gestörten Funkumgebungen in zwei Abstufungen eine Vorwärtsfehlerkorrektur (Forward Error Correction, FEC), die jedoch durch den erheblichen Overhead den Payload-Durchsatz senkt und deswegen wohl nur als Kompensation der Latenzprobleme des ARQ-Verfahrens dienen kann.

Jedes übertragene Paket beginnt mit 16 Byte Verwaltungsinformationen und trägt variabel lange Payload mit zur Zeit maximal 343 Bytes Länge. Die Übertragung eines Paketes kann also bis zu 5 Timeslices in Anspruch nehmen. Bei synchronen Verbindungen für Sprachübermittlung wird das *Continuous Variable Slope Delta Modulation* (CVSD) Sprachcodierungsverfahren verwendet, das für seine Robustheit gegenüber fehlerhafter Datenübertragung bekannt ist. Die Übertragungsrate beträgt für Sprache 64kBit/s.

3.3.3 Link Manager und I/O Unit

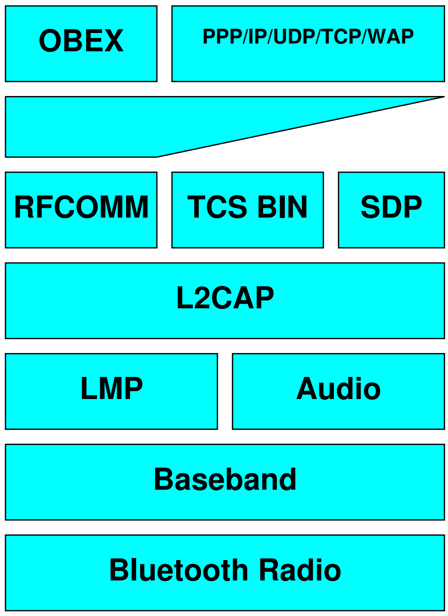


Abbildung 9: Bluetooth Protokoll Architektur

Die Link Manager und I/O Unit stellt den Bluetooth-spezifischen Software-Protokollstapel (Abbildung 9) und das *Application Programming Interface* (API) für die Anwendungssoftware dar. Sie kümmert sich ebenfalls um die High-Level Funktionalität des Bluetooth-Protokolls.

Dazu bauen zwei verschiedene Protokolle auf der Baseband Hardware auf - das *Link Manager Protocol* (LMP) und dazu teilweise parallel das *Logical Link and Control Adapti-*

on Protocol (L2CAP). LMP kümmert sich um Verbindungs- und Piconetz-Management, Sicherheitsfunktionen wie Authentifizierung, Authorisierung und Verschlüsselung und Paketgrößensteuerung sowie Quality of Service und Kontrolle der Energiesparmodi der Slaves im Piconetz, während L2CAP die eigentliche Übertragung von Daten steuert (sogenannte Asynchronous Connection Links) und zur Adaption der Protokolle der höheren Layer dient. Sprachübertragung verwenden synchrone Verbindungen, die im Baseband feste Zeitschlitze zugewiesen bekommen und somit unter Umgehung von L2CAP direkt auf dem Bluetooth Baseband operieren.

Auf L2CAP setzt wiederrum RFCOMM (Emulation eines RS-232 seriellen Ports), TCS BIN (Telephony Control Protocol als Steuerungsschicht für die Sprachübertragungen) und SDP (Service Discovery Protocol, dient zum Erkunden der von Bluetooth-Geräten angeboteten Diensten, siehe 3.3.4). RFCOMM erlaubt mit der Unterstützung von AT-Befehlen die Telefonanwahl oder gar Fax-Betrieb und eine darauf aufsetzende PPP-Schicht kann als Gateway zu TCP/IP oder WAP Diensten fungieren (Internetzugriff).

Desweiteren gibt es mit der OBEX (Object Exchange Protocol) Schicht eine Möglichkeit zum allgemeinen Objektaustausch (z.B. für Visitenkarten oder elektronischer Zahlungsverkehr) und mit IrOBEX eine Plattform für das IrDA Protokoll.

3.3.4 Auffinden von Services

Das *Service Discovery Protocol* (SDP) dient zum Auffinden von Endgeräten, die benötigte Serviceleistungen anbieten. Es ist als paketorientiertes Anfrage/Antwort-Protokoll spezifiziert, wobei es jedoch keine Möglichkeit vorgesehen ist, über das Verfügbarwerden oder den Ausfall von Serviceleistungen automatisch informiert zu werden.

Services können dabei Informationen bereitstellen (wie z.B. IP-Adressen in einem lokalen Netzwerk), Aktionen auslösen (z.B. die Annahme eines Telefonanrufes) oder eine Ressource kontrollieren (z.B. einen Netzwerkdrucker). SDP Server stellen eine Liste der verfügbaren Services via SDP bereit. Jeder Service wird durch eine Serviceklassifizierung und eine eindeutig zugeteilte ID identifiziert. Zu jedem Services in diesem Katalog können Attribute mit Parametern zugeordnet werden, die weitere Details über den jeweiligen Service spezifizieren. Die möglichen Attribute mit ihren Wertemengen hängen von der Serviceklassifizierung ab und sind von der Bluetooth Group genau spezifiziert. Zur Erweiterungen dieser Spezifizierungen existiert die Möglichkeit, eine Subklasse zu einer übergeordneten Serviceklassifizierung zu erzeugen. Ebenso besteht die Möglichkeit, ein physikalischen Service in mehrere logische Serviceklassifizierungen einzuordnen.

Zum Suchen nach Services wird sowohl die Anfrage nach freien Services, die einer angegebenen Serviceklassifizierung entsprechen, als auch eine Anfrage zur Listung aller von einem speziellen Gerät angebotenen Services unterstützt.

Um das Servicewirrwarr überschaubar zu halten und eine Interopabilität mit heutigen und zukünftigen Bluetooth Geräten sicherzustellen, spezifiziert die Bluetooth SIG eine Reihe von sogenannten *Bluetooth Profilen*, die typische Geräte und ihre Serviceanbindung an das Bluetooth-Netz beschreiben. Ein Großteil der Arbeit der Bluetooth Group wird dabei in die Definierung und Wartung dieser Profile gesteckt. Zur Zeit übliche Profile werden in einer über 1400 Seiten langen separaten Spezifikation beschrieben und in einem weiteren Seminar vorgestellt und sind daher hier nicht weiter erwähnt.

3.4 Bewertung Bluetooth

3.4.1 Sicherheit

Da Funkverbindungen prinzipiell leichter abzuhören sind als Kabelverbindungen, spezifiziert Bluetooth zum Schutz vor Missbrauch optional zwei Schutzebenen, eine *Service Level Security* und eine *Link Level Security*.

Zur Authentifizierung (Link Level Security) stellen die Endgeräte fest, mit welchen Gegenüber sie kommunizieren. Dazu wird ein nur einmal gültiger *Initialisation Key* mit 128

Bit Länge zum Verbindungsaufbau aus einer PIN, einer Geräteadresse, dem Master Clock und einer Zufallszahl generiert. Über eine mit diesem Key verschlüsselte Kommunikation einigen sich beide Kommunikationsparter auf einen ebenfalls 128 Bit langen Verbindungskey, der zur Verschlüsselung der Verbindungskontrolldaten dient. Damit wird aus einem „unbekannten" Gerät ein „unsicheres" Gerät.

Die optionale *Service Level Security* ist als einfache Zugriffskontrolle auf angebotene Services für „unsichere" Kommunikationspartner zu verstehen. Service Anbieter dürfen diese je nach Art des Services bestimmten unsicheren Geräten automatisch oder erst z.B. nach Benutzernachfrage gewähren (z.B. zum Zugriff auf Terminplanerdaten zum Abgleich eines Termins). Solche Kommunikationspartner werden dann in den Stand der „vertrauenswürdigen" Geräte erhoben und der nachfolgende Datenaustausch wird über einen separaten, aus dem Link Key generierten Key mit symmetrischer Verschlüsselung, je nach lokaler Gesetzeslage mit einer Länge von 8 bis 128 Bit, durchgeführt.

All diese Sicherheitsmechanismen finden im Bluetooth Baseband statt und verlaufen gegenüber höheren Protokollebenen transparent, um zum Beispiel mit Computerviren infizierter Anwendungssoftware keine Hintertür zu bieten.

Die Bluetooth Group behält sich vor, weitere und stärkere Verschlüsselungen und Sicherheitsmechanismen auf höherer Protokollebene bei Bedarf hinzuzufügen.

3.4.2 Zukunft

Nachdem bereits mehrere Revisionen der über 1000 Seiten langen Bluetoothspezifikation mit kleineren Fehlerkorrekturen freigegeben wurden, beginnt nun die Umsetzungsphase, die über den Erfolg oder Misserfolg von Bluetooth entscheidet. Die Bluetooth Group beschäftigt sich zur Zeit hauptsächlich mit dem „Feedback" der Industrie bei der Umsetzung der Spezifikation und den wohl teilweise noch erheblichen Interopabilitätsproblemen der Bluetooth Profile.

Es wird auch bereits an der Weiterentwicklung gearbeitet, um höhere Geschwindigkeit, bessere Sicherheit und Stabilität sowie eine erhöhte Ausfallsicherheit in schwierigen Umgebungen zu erreichen. Die Hauptkritikpunkte scheinen zur Zeit die geringe Stabilität und der nur sehr dürftige Datendurchsatz bei typischen Anwendungen zu sein. Zudem ist der angestrebte Preis für die „Bluetooth Chips" zur Zeit noch nicht erreicht und wird wohl ein ganz entscheidender Faktor zugunsten oder gegen diese Technologie in naher Zukunft darstellen.

Werden diese ausstehenden Probleme zumindest zufriedenstellend gelöst werden können, so wird uns wohl der „kabelfreien" Umgebung am Arbeitsplatz, im Supermarkt oder in der Vernetzung der PCs nicht mehr viel im Wege stehen. Die Vorteile liegen auf der Hand, denn es könnte als Verbindungsdienst sowohl das lokale Netzwerk als auch den mobilen Zugang zum Internet (sofern man sich immer in 10-Meter Umkreis eines Masters mit diesem Service befindet) ermöglichen und uns in allen Lebenslagen unterstützen.

Zu erwähnen ist jedoch, dass ähnliche Möglichkeiten auch von den Bluetooth-Vorläufern und Konkurrenten, hauptsächlich DECT, IEEE 802.11 und IrDA zwar für mehr Geld aber dafür mit teilweise erheblich höheren Datenraten geboten werden.

4 Ausblick

Die vorgestellten Techniken konnten sich noch nicht vergleichbar zum Desktopbereich in der Praxis bewähren, insofern ist es noch relativ unklar, inwieweit es mit der Kundenakzeptanz für diese Techniken des Mobile Commerce in Zukunft bestellt sein wird ([18]).

Die WAP-Technologie ist eine sehr gut auf die verfügbare Hardware angepasste Architektur. Da jedoch auch hier die Entwicklung nicht stehen bleibt, wird das WAP Protokoll in Zukunft beweisen müssen, ob es auch mit besserer Hardware entsprechend in Benutzerfreundlichkeit und Qualität skalieren kann. Zur Zeit scheint der Trend eher in

Richtung von mobilen Webpads, die auf Basis unterschiedlichster Übertragungstechnologie klassische Internettechnologie und Webinhalte direkt verfügbar machen, zu gehen. Aus eigener Erfahrung ist dem Autor bekannt, daß eine solche Implementierung der benötigten Internettechniken und der Anzeigefähigkeiten in Software für solche WebPADs mit erträglichem Aufwand und akzeptabler bis durchaus angenehmer Performance schon jetzt möglich ist.

Die Bluetooth-Technologie bestaicht durch ihre Kombination verschiedenster Ideen und der auf Kosteneffizienz und einfache Wartung ausgerichteten Spezifikation und ist durchaus als sehr interessant und mit großen Potential einzustufen. Die von der Werbung bereits jetzt ausformulierten Anwendungsgebieten lassen auf eine drahtlose Zukunft hoffen. Allerdings darf auch hier nicht vergessen werden, daß man damit versucht, über die Nachteile, wie den geringen Datendurchsatz oder das anscheinend noch recht unbeherrschbare Eigenleben der dynamisch etablierten Netzstrukturen besonders in funkgestörten Umgebungen hinwegzutäuschen. Insofern ist auch hier abzuwarten, inwieweit die Spezifikation mit der Weiterentwicklung von Übertragnungstechnik und den steigenden Anforderungen mitwachsen kann.

Literatur

[1] http://www.eurotechnology.com/imode/

[2] http://www.freeprotocols.org/wapTrap/

[3] http://www.imafdi.de/m-commerce.htm

[4] http://www.wapforum.org/

[5] http://www.japon.net/imode/

[6] http://www.w3.org/Protocols/Specs.html

[7] http://www.paybox.de/

[8] http://www.12snap.com/germany/

[9] http://www.nttdocomo.com/

[10] http://www.imodelinks.com/

[11] http://www.cc.jyu.fi/~mjos/wtls.pdf

[12] http://www.wlana.org/learn/whitep.htm

[13] http://www.embedded.com/internet/0007/0007ia1.htm

[14] iX 2/2000 und iX 10/1999 des Heise Verlags

[15] http://www.w3.org/TR/2000/NOTE-voicexml-20000505/

[16] http://www.bluetooth.com/

[17] WAP-Forum der iX-Redaktion, http://www.heise.de/ix/

[18] http://cyberatlas.internet.com/markets/wireless/article/1,1323,10094_513431,00.html

[19] http://www.t9.com/

[20] http://www.gsmworld.com/

[21] http://www.rose.cc.ok.us/faculty/kdewey/park/CS365/protocol.html

[22] http://www.w3.org/TR/1998/NOTE-compactHTML-19980209/

[23] http://www.j-phone.co.jp/

[24] http://www.kddi.com/

[25] http://www.leapforum.org/

[26] http://www.phone.com/

[27] http://www.ericsson.com/

[28] http://www.voicexml.org/

[29] http://www.mobilemexe.com/

[30] http://www.umts-forum.org/

[31] http://java.sun.com/j2me/

[32] http://www.w3.org/TR/NOTE-Submission-HDML-spec.html

[33] http://www.w3.org/TR/REC-xml

[34] http://www.w3.org/TR/wbxml/

[35] http://www.wapforum.org/what/technical.htm

[36] http://www.irda.org/