



GMD Report 107

GMD –
Forschungszentrum
Informationstechnik
GmbH

Jens Budach
Andreas Wombacher

Konzeption und Entwicklung exemplarischer Informationsdienste für mobile Endgeräte

August 2000

© GMD 2000

GMD –
Forschungszentrum Informationstechnik GmbH
Schloß Birlinghoven
D-53754 Sankt Augustin
Germany
Telefon +49 -2241 -14 -0
Telefax +49 -2241 -14 -2618
<http://www.gmd.de>

In der Reihe GMD Report werden Forschungs- und Entwicklungsergebnisse aus der GMD zum wissenschaftlichen, nichtkommerziellen Gebrauch veröffentlicht. Jegliche Inhaltsänderung des Dokuments sowie die entgeltliche Weitergabe sind verboten.

The purpose of the GMD Report is the dissemination of research work for scientific non-commercial use. The commercial distribution of this document is prohibited, as is any modification of its content.

Anschriften der Verfasser/Addresses of the authors:

Jens Budach
Zwillingstraße 48
D-64546 Mörfelden-Walldorf
E-mail: Jens@budach.de

Andreas Wombacher
Institut für Integrierte Publikations- und Informationssystem
GMD – Forschungszentrum Informationstechnik GmbH
Dolivostraße 15
D-64293 Darmstadt
E-mail: Andreas.Wombacher@gmd.de

ISSN 1435-2702

Kurzfassung

Mobilität gewinnt zunehmend an Bedeutung in unserer Gesellschaft. Dies gilt insbesondere für die Verfügbarkeit von Informationen und Diensten auf mobilen Endgeräten. Ein neuer, stark wachsender Markt ist die Bereitstellung von Informationen und Diensten mittels der WAP Technologie, wobei aktuell meist Handys als Endgeräte eingesetzt werden. Die zu meist kleinen Displays der Endgeräte erfordern Konzepte für die Visualisierung der unterschiedlich strukturierten Inhalte und deren Interaktionen. Im Rahmen der Studie werden potentielle Inhalte und deren Interaktionen kategorisiert und Lösungsvorschläge für die Visualisierung der einzelnen Problembereiche erarbeitet.

Schlagwörter: WAP, WML, WMLScript, Informationsdesign, Designkonzept, mobiles Endgerät, mobiles Internet, XML, XSL

Abstract

Mobility is a major issue in today's business society. In particular, this applies to the availability of information and different services for wireless devices. A new, increasing market is based on the WAP technology providing information and services to wireless devices, which are currently mobiles. The small display of these devices requires special visualization concepts depending on the different content structure and interaction behavior. Within this study potential content and interactions are categorized and adequate visualization solutions are provided, respectively.

Keywords: WAP, WML, WMLScript, information design, design solution, wireless device, wireless Internet, XML, XSL

Inhaltsverzeichnis

1	EINLEITUNG	7
2	SPRACHEN DES MOBILEN INTERNET – EINE ANALYSE	8
2.1	WML – WIRELESS MARKUP LANGUAGE	8
2.1.1	WML – eine Einführung.....	8
2.1.1.1	Darstellung von Text und Inhalten	9
2.1.1.2	Struktur von WML.....	10
2.1.1.3	Navigation in WML	10
2.1.1.4	Zustandsmanagement in WML.....	11
2.1.1.5	Erstes WML-Beispiel	11
2.1.2	WML ist XML.....	12
2.1.2.1	Elemente	14
2.1.2.2	Groß- und Kleinschreibung.....	15
2.1.2.3	Attribute	15
2.1.2.4	Entity-Referenzen.....	16
2.1.2.5	Kommentare	17
2.1.3	WML 1.1.....	17
2.1.3.1	Dokumenten Prolog.....	18
2.1.3.2	Standardelemente	18
2.1.3.3	Benutzereingaben	20
2.1.3.4	Formatierung und Grafiken.....	24
2.1.3.5	Ereignismodell in WML.....	28
2.1.3.6	Zustandsmodell / Variablen	33
2.1.4	WML 1.2.....	34
2.1.5	Was WML nicht kann	34
2.2	WMLSCRIPT.....	36
2.2.1	WMLScript 1.1.....	36
2.2.1.1	Kommentare	37
2.2.1.2	Variablen	38
2.2.1.3	Datentypen.....	38
2.2.1.4	Funktionen.....	40
2.2.1.5	Bibliotheken.....	41
3	INFORMATIONSPORTEN FÜR MOBILE ENDGERÄTE	45
3.1	ANWENDUNGSBEREICHE.....	45
3.1.1	Allgemein	45
3.1.2	Konkret.....	47

3.1.2.1	Informationendienste	47
3.1.2.2	Auskunftssysteme	48
3.1.2.3	Interaktive Dienste	49
3.1.2.4	Portale	50
3.2	ENTWICKLUNG VON KLASSIFIKATIONSMERKMALEN	52
3.2.1	<i>Klassifikationsmerkmal Medialer Typ</i>	53
3.2.1.1	Medialer Typ Text	53
3.2.1.2	Medialer Typ Grafik	53
3.2.1.3	Mediale Typen Sound und Video	54
3.2.1.4	Multimedia	54
3.2.2	<i>Klassifikationsmerkmal Struktur</i>	55
3.2.3	<i>Klassifikationsmerkmal User Interaktion</i>	56
3.3	ANWENDUNG DER KLASSIFIKATION	56
3.3.1	<i>Beispiel Überweisung</i>	58
3.3.2	<i>Beispiel Nachrichten</i>	59
3.3.3	<i>Beispiel Zugfahrplan</i>	60
3.3.4	<i>Beispiel Portal</i>	62
3.3.5	<i>Beispiel Aktienkurse</i>	63
4	ENTWICKLUNG VON WML-SEITEN	64
4.1	RAHMENBEDINGUNGEN	64
4.1.1	<i>Mobile Endgeräte</i>	64
4.1.1.1	Allgemeine Merkmale	65
4.1.1.2	Spezielle Merkmale	67
4.1.2	<i>Mobilfunknetz</i>	68
4.1.3	<i>Endnutzer</i>	69
4.2	LÖSUNGSKONZEPTE FÜR RELEVANTE WAP-BEREICHE	70
4.2.1	<i>Navigation</i>	70
4.2.1.1	Navigation über Aktionen	71
4.2.1.2	Navigation in der Anwendung	72
4.2.2	<i>Cards und Decks</i>	75
4.2.3	<i>Grafiken</i>	78
4.2.4	<i>Lösungskonzept Überweisung</i>	79
4.2.5	<i>Lösungskonzept Nachrichten</i>	86
4.2.6	<i>Lösungskonzept Zugfahrplan</i>	92

4.2.7	<i>Lösungskonzept Portal</i>	97
4.2.8	<i>Lösungskonzept Aktienkurse</i>	101
5	REALISIERUNG VON WML-SEITEN MIT XSL(T)	105
5.1	XSL UND XSLT	105
5.2	BEISPIEL ONLINE-AUKTION	106
5.2.1	<i>Entwicklung der XML-Quelle</i>	106
5.2.2	<i>Entwicklung der XSLT-Stylesheets</i>	107
5.2.2.1	<i>Stylesheet des Startdeck</i>	108
5.2.2.2	<i>Stylesheet der Ebene 2</i>	111
5.2.2.3	<i>Stylesheets der Ebene 3</i>	111
5.2.3	<i>Generierung der WML-Seiten</i>	113
6	AUSBLICK	115
6.1	MOMENTANE HINDERNISSE	115
6.1.1	<i>Endgeräte</i>	116
6.1.2	<i>Kosten</i>	116
6.1.3	<i>Entwicklungsreife</i>	116
6.2	ZUKÜNFTIGE ENTWICKLUNGEN	117
6.2.1	<i>Regionale Dienste</i>	117
6.2.2	<i>Kontextdienste</i>	118
6.2.3	<i>Personalisierung</i>	118
6.2.4	<i>Sprachsteuerung</i>	118
6.2.5	<i>Zahlungsmittel mobiles Endgerät</i>	119
6.2.6	<i>Neue Übertragungstechniken</i>	119
6.2.7	<i>Multimedia</i>	120
6.3	FAZIT UND RESÜMEE	121
7	LITERATURVERZEICHNIS	122
8	ANHANG	128
8.1	ABBILDUNGSVERZEICHNIS	128
8.2	TABELLENVERZEICHNIS	130
8.3	BEISPIELVERZEICHNIS	130

1 Einleitung

Das Streben nach Mobilität gewinnt in der heutigen Gesellschaft immer mehr Bedeutung. Auch deshalb steigt der Bedarf, von unterwegs auf Informationen zu zugreifen, stetig. Der Boom der Mobiltelefone insbesondere in Europa ist für diese Entwicklung nur ein Beleg. Aufgrund der neuen Entwicklungen im mobilen Bereich werden Anwendungen, die den Zugriff auf Informationen mit Hilfe mobiler Endgeräte ermöglichen, in nächster Zukunft ein rasantes Wachstum erfahren.

Forrester Research prognostiziert, dass im Jahre 2004 etwa 30 Prozent der Europäer ein Mobiltelefon nutzen werden, um auf das Internet zuzugreifen. Nokia erwartet für das Jahr 2002 weltweit eine Milliarde Handybesitzer. Dies sind nur zwei der zahlreichen Prognosen, welche die oben stehenden Thesen belegen helfen.

Der Zugriff auf Informationen mit Hilfe mobiler Endgeräte gestaltet sich bis dato eher umständlich. Bisher stellte der Short Message Service (SMS) die einzige Möglichkeit dar, Informationen über ein Mobiltelefon zu erhalten. Einen verbesserten Zugang zu Informationen soll nun das Wireless Application Protocol (WAP) ermöglichen, welches einen weltweiten, herstellertunabhängigen Standard für die Übertragung von Daten zu mobilen Endgeräten darstellt.

2 Sprachen des mobilen Internet – eine Analyse

2.1 WML – Wireless Markup Language

Zunächst wird in den folgenden Abschnitten die Wireless Markup Language (WML) in der Version 1.1 analysiert und beschrieben. Zwar hat das WAP-Forum Ende 1999 die Version 1.2 verabschiedet, aber weder ist diese Version in den heute verfügbaren Microbrowsern implementiert (Stand: 29.05.2000), noch hat das W3C bisher die WML-Version 1.2 beurteilt und verabschiedet.¹ Aus diesem Grund wird das Hauptaugenmerk auf die WML-Version 1.1 gelegt. Wenn von einer anderen Version als WML 1.1 gesprochen, wird darauf ausdrücklich hingewiesen. Es werden dabei die wichtigsten Funktionalitäten und Merkmale herausgearbeitet und Möglichkeiten bzw. Beschränkungen aufgezeigt. Zusätzlich wird auf die Überschneidungen bzw. Unterschiede von WML und der HyperText Markup Language (HTML) des Internet eingegangen.

2.1.1 WML – eine Einführung

WML wird derzeit in der Literatur häufig als "Handy-HTML"² beschrieben und trifft recht genau die Aufgabe von WML. WML ist die Auszeichnungssprache, in der man Dokumente strukturiert, die über ein mobiles Endgerät (z.B. Handy) abgerufen werden sollen. Die Bezeichnung "Handy-HTML" ist allerdings genau genommen nicht ganz korrekt, da WML und HTML nur Verwandte zweiten Grades sind (siehe Kapitel 2.1.2).

Die Mobilfunknetze und die mobilen Endgeräte besitzen gewisse Charakteristika, die sie von Festnetzen bzw. den stationären Endgeräten unterscheiden. Ein Mobilfunknetz zeichnet sich insbesondere durch geringe Bandbreite, hohe Durchlaufzeiten, instabile Verbindungen und geringere Verfügbarkeit aus. So erreicht beispielsweise der in Europa verwendete Mobilfunkstandard Global System for Mobile communication (GSM) derzeit nur eine Übertragungsrate von 9,6 kBit/s.³ Ein mobiles Endgerät hat mit Beschränkungen, wie geringer CPU-Leistung, kleinem Datenspeicher, kleinen Displays, beschränktem Stromspeicher und den

¹ Vgl. WAP FORUM (1999e) und RÖWEKAMP, Lars (2000), S.104

² Vgl. BAGGER, Jo (1999), S.126 und RÖWEKAMP, Lars (2000b), S.52

³ Vgl. ŽIVADINOVIĆ, Dusan (1999), S.122

Sprachen des mobilen Internet – eine Analyse

unterschiedlichen Eingabegeräten (Sprache, Touchpad, Zahlentastatur) zu kämpfen (Näheres zu den Rahmenbedingungen in Kapitel 4.1).

Bei der Entwicklung von WML wurde versucht, die oben genannten Rahmenbedingungen zu berücksichtigen und WML für diese Verhältnisse zu optimieren.⁴ So wird WML beispielsweise vor der Übertragung über drahtlose Netze in Bytecode umgesetzt, der wesentlich weniger Bandbreite benötigt als Dateien, die im Klartext übertragen werden.

In Abbildung 1 ist beispielhaft das Verhältnis der Übertragung von HTML und WML dargestellt.

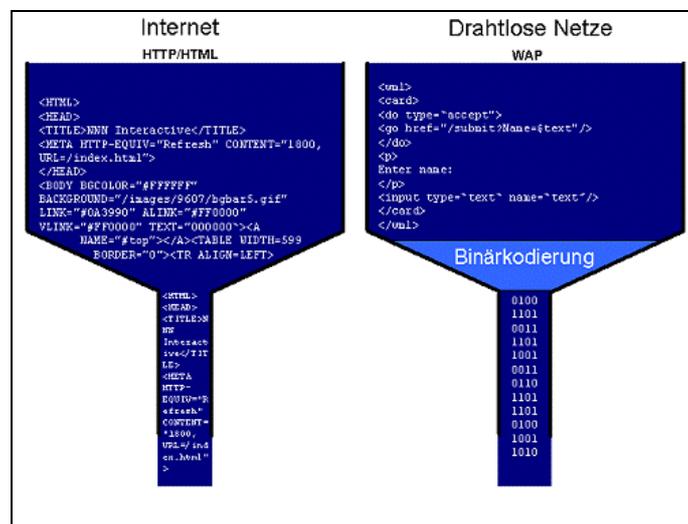


Abbildung 1: Vergleich der Übertragung von HTML und WML⁵

Die Funktionen und Merkmale von WML lassen sich nun grob in vier Gruppen einteilen, die im Folgenden näher erläutert werden.⁶

2.1.1.1 Darstellung von Text und Inhalten

WML unterstützt die Verwendung von Text und Grafiken. Zusätzlich bestehen verschiedene Möglichkeiten, Text zu formatieren.

⁴ Vgl. BUDACH, Jens (2000), S.6-8

⁵ Vgl. FLAHERTY, Natasha (1999), S. 9

⁶ Vgl. WAP FORUM (1999), S.5

Sprachen des mobilen Internet – eine Analyse

2.1.1.2 Struktur von WML

Eine WML-Datei besteht aus einer oder mehreren selbstständigen Informationseinheiten, den sogenannten Karten (engl. cards). Eine Informationseinheit kann zum Beispiel ein Auswahlmenü, einen Bildschirm mit Text oder eine Eingabeaufforderung enthalten. Die WML-Datei repräsentiert dabei den Kartenstapel (engl. deck). Lädt ein mobiles Endgerät eine WML-Datei, wird zunächst nur die erste bzw. angegebene Card angezeigt. Je nach Anwendung kann der Nutzer dann durch eine Reihe weiterer Cards innerhalb des Deck navigieren, ohne dass ein erneuter Serverzugriff erfolgt. Das Deck ist das Pendant zur HTML-Datei, wird ebenfalls per URL identifiziert und ist die kleinste Einheit, die an das mobile Endgerät gesendet werden kann. Die Abbildung 2 verdeutlicht diesen Sachverhalt nochmals.

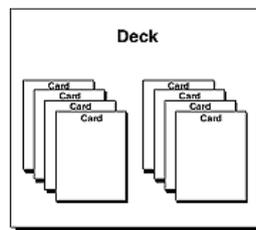


Abbildung 2: Struktur von WML

2.1.1.3 Navigation in WML

WML unterstützt die Navigation zwischen Cards innerhalb eines Deck und zwischen verschiedenen Decks. Zusätzlich bietet WML ein Ereignismodell, welches für die Navigation oder das Ausführen von Skripten genutzt werden kann. Einzelne Cards eines Deck werden mit sogenannten Ankern (engl. anchor) angesprochen. Dieses Verfahren wird auch bei HTML 4.0⁷ verwendet. Ferner hat WML eine sogenannte Historyfunktion implementiert, die es ermöglicht zu bereits besuchten Cards und Decks zurückzukehren. Um die Navigation zu erleichtern, unterstützt WML Soft-Buttons oder Soft-Keys. Dies sind spezielle Benutzerschnittstellen des mobilen Endgeräts, denen je nach Kontext, unterschiedliche Funktionen zugewiesen werden können. Ein Beispiel für so einen Soft-Key ist eine graphisch erzeugte Taste, die es ermöglicht, zur vorhergehenden Card zurückzukehren.

⁷ RAGGET, Dave [u.a.] (1997)

2.1.1.4 Zustandsmanagement in WML

Das Zustandsmodell von WML erlaubt die Definition von Variablen, mit denen benutzerangepasste Sichten auf Informationen dargestellt werden können. Trägt ein Nutzer beispielsweise seinen Wohnort in ein Eingabefeld ein, wird dieser in einer Variablen mit einem bestimmten Namen gespeichert. Auf diese Variable kann dann mit Hilfe des Variablennamens zugegriffen werden. Diese Anpassung an bestimmte Zustände ermöglicht eine effiziente Nutzung der vorhandenen Netzwerkkapazitäten.

2.1.1.5 Erstes WML-Beispiel

Die Struktur und der Aufbau eines WML-Dokuments lässt sich am besten anhand eines einfachen Beispiels darstellen.

Beispiel 1: Einfaches WML-Beispiel

Nokia WAP Toolkit 1.2	WML
 <p>Abbildung 3: 1.Karte</p>  <p>Abbildung 4: 2.Karte</p>	<pre><?xml version="1.0"?> <!DOCTYPE wml PUBLIC "-//WAPFORUM//DTD WML 1.1//EN" "http://www.wapforum.org/DTD/wml_1.1.xml"> <wml> <card id="card1" title="1. Karte"> <do type="accept" label="weiter"> <go href="#card2"/> </do> <p> Hallo Welt!
 (1. Karte) </p> </card> <card id="card2" title="2. Karte"> <p> Nochmal Hallo!
 2.Karte </p> </card> </wml></pre>

Das obige Beispiel enthält ein Deck, also eine WML-Datei, mit zwei Cards. Die ersten vier Zeilen werden im Abschnitt 2.1.3.1 erläutert und sind für jedes WML-Dokument obligatorisch. Der eigentliche Inhalt beginnt mit dem Tag `<wml>`, der am Ende des Dokuments mit `</wml>` wieder geschlossen wird. Die einzelnen Cards werden durch die Tags `<card>` und `</card>` eingeschlossen. Die erste Card besitzt die Attribute `id` und `title`. Über den Wert "card1" des Attributs `id`, kann die erste Card per URL, nach dem Schema `.../datei.wml#card1`, angesprochen werden. Der Inhalt des Attributs `title`, gibt dem Browser im mobilen Endgerät

Sprachen des mobilen Internet – eine Analyse

die Möglichkeit, einen Titel anzuzeigen. Wie oben beschrieben, besitzt WML ein Ereignismodell, das es ermöglicht, auf verschiedene Ereignisse zu reagieren. Das Element `do` stellt dabei einen generellen Mechanismus zur Ausführung von Aktionen dar.⁸ Das Element `do` wird von den Attributen `type` und `label` näher erläutert. Diese Zusatzinformationen werden von den Browsern der mobilen Endgeräte unterschiedlich eingesetzt. Welche Aktion ausgeführt werden soll, definiert man innerhalb des Elements `do`. Mit der Anweisung `go` legt man einen Sprungverweis zur angegebenen URL, hier `#card2`, fest. Innerhalb des aus HTML bekannten Elements `p`, steht der eigentliche Text der Card. Die zweite Card erreicht man, indem die Funktion `Options` des mobilen Endgerätes aus Abbildung 3 aufgerufen wird. Diese zeigt ein Menü mit allen definierten Aktionen an. Da die Darstellung von Gerät zu Gerät variiert, ist das Menü hier nicht explizit abgebildet. Die zweite Card entspricht dem Aufbau der ersten Card.

2.1.2 WML ist XML

WML basiert auf der Extensible Markup Language (XML)⁹ Version 1.0 des World Wide Web Consortium (W3C). Genauer gesagt ist WML eine Anwendung von XML, ebenso wie HTML eine Anwendung der Standard Generalized Markup Language (SGML) ist. "Die Standard Generalized Markup Language ist der internationale Standard für die Definition, Identifikation und Benutzung der Struktur und des Inhalts von Dokumenten."¹⁰ Beide Anwendungen, WML und HTML, sind sogenannte Document Type Definitions (DTD) der jeweiligen Sprachen XML und SGML.

XML und SGML werden Metabeschreibungssprachen genannt, da sie keine Auszeichnungssprachen an sich sind. Sie enthalten die Definitionen, wie Auszeichnungssprachen für Dokumente aussehen müssen. XML ist dabei eine Teilmenge von SGML. In der Literatur findet sich auch die Kurzbeschreibung "XML ist SGML fit gemacht für das Web". Das bedeutet, dass XML die Komplexität von SGML verloren, aber deren Möglichkeiten weitestgehend beibehalten hat.¹¹

⁸ Vgl. RÖWEKAMP, Lars (2000b), S.56

⁹ Vgl. BRAY, Tim [u.a.] (1998)

¹⁰ Vgl. BAGER, Jo (1997), S.298

¹¹ Vgl. BEHME, Henning (1999), S.37

Sprachen des mobilen Internet – eine Analyse

Da in XML die Zukunft für Internetdokumente gesehen wird, wurde am 26.01.2000 vom W3C eine neue Anwendung (DTDs) von XML, die Extensible HyperText Markup Language (XHTML), verabschiedet. XHTML ist der Nachfolger von HTML. Dabei handelt es sich um eine Neuformulierung von HTML 4 in XML 1.0¹²

Die Abbildung 5 verdeutlicht den Zusammenhang der einzelnen Auszeichnungssprachen, die in diesem Kontext auftauchen.

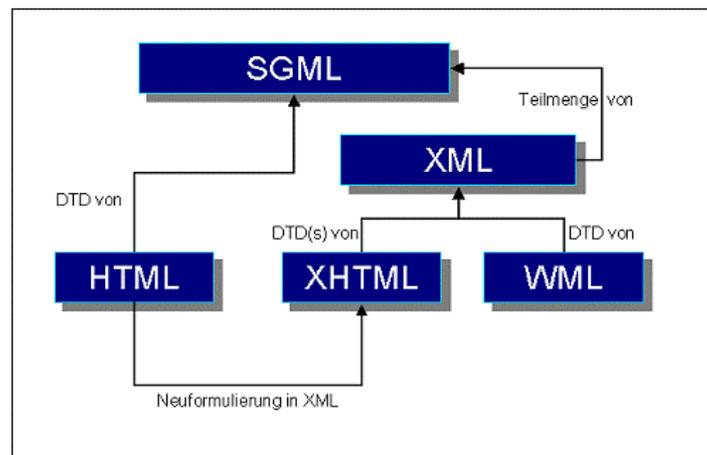


Abbildung 5: Beziehungsverhältnisse von SGML, XML und deren Anwendungen¹³

Diese Arbeit beschränkt sich auf WML, die im Wireless Application Protocol (WAP) als Dokumentenauszeichnungssprache für mobile Endgeräte definiert wurde.¹⁴ Wie schon erwähnt, ist WML eine Anwendung von XML, das bedeutet, dass WML auch allen Vorschriften und Beschränkungen von XML unterliegt.

Die Document Type Definition von WML enthält alle erlaubten Elemente, Attribute und Entitäten. Da WML bewusst an HTML angelehnt wurde, kann sie relativ leicht von jemandem erlernt werden, der auch mit HTML vertraut ist. XML ist im Allgemeinen wesentlich strenger gehalten in Bezug auf die Einhaltung bestimmter Schreibweisen als SGML bzw. HTML. Wenn im Folgenden von Restriktionen in XML gesprochen wird, beziehen sich die Angaben auch auf WML.

¹² Vgl. RAGETT, Dave [u.a.] (2000)

¹³ Vgl. BEHME, Henning; MINTERT, Stefan (1998), S.17

¹⁴ Vgl. WAP FORUM (1999)

Sprachen des mobilen Internet – eine Analyse

2.1.2.1 Elemente

XML erfordert, dass alle geöffneten Elemente auch wieder geschlossen werden. In HTML kann beispielsweise ein Tag, wie `<p>` ohne ein schließendes `</p>` verwendet werden, ohne dass dies falsch wäre. In WML ist es zwingend erforderlich, das Tag `<p>` auch mit dem korrespondierenden Tag `</p>` zu schließen, sonst entspricht das Dokument nicht den Konventionen von XML.

Beispiel 2: HTML vs. WML – Elemente schließen

HTML	WML
<pre><body> <p>Diese Art ist in HTML komplett in Ordnung. <p>Der nächste Paragraph wird dann mit einem erneuten Tag gesetzt </body></pre>	<pre><card> <p>In WML müssen alle Elemente geschlossen werden</p> </card></pre>

Da alle Elemente in XML wieder geschlossen werden müssen, sind davon auch die sogenannten leeren Elemente betroffen. Definiert man in HTML einen Zeilenumbruch mit `
`, muss das leere Element für den Zeilenumbruch in WML mit `
` definiert werden. Demzufolge werden alle leeren Elemente in XML mit `/>` abgeschlossen. In XML ist allerdings auch erlaubt, leere Elemente mit einem zusätzlichen schließenden Tag zu versehen (`
</br>`). Diese Ausnahme gilt allerdings nicht für WML.

Beispiel 3: HTML vs. WML – leere Elemente schließen

HTML	WML
<pre><body> <p>Um in HTML einen Zeilenumbruch zu erreichen, schreiben Sie:
 </body></pre>	<pre><card> <p>In WML werden leere Elemente, wie der Zeilenumbruch, mit einem Schrägstrich in dem leeren Element abgeschlossen.
 Hier ist die neue Zeile.</p> </card></pre>

Elemente in XML müssen so ineinander geschachtelt werden, dass sie sich nicht überlappen. Ein Element kann dabei andere Elemente enthalten oder selbst in anderen Elementen enthalten sein.

Sprachen des mobilen Internet – eine Analyse

"Ausgehend von einem Root-Element müssen alle weiteren Elemente ineinander geschachtelt sein: `<root><a><c></c></root>` ..." ¹⁵ Werden diese Konventionen in XML-Dokumenten eingehalten, werden die Dokumente "wellformed" oder "wohlgeformt" genannt.

2.1.2.2 Groß- und Kleinschreibung

XML ist case-sensitiv, d.h., dass XML zwischen Groß- und Kleinschreibung unterscheidet. Definiert man mit XML beispielsweise ein Element der Form `<element></element>` und eins der Form `<Element></Element>`, sind das für XML zwei unterschiedliche Elemente. In WML wurde definiert, dass alle Tags klein geschrieben werden müssen.

Beispiel 4: HTML vs. WML – Groß- und Kleinschreibung beachten

HTML	WML
<pre><BODY> <P>In HTML wird nicht zwischen Groß- und Kleinschreibung unter- schieden </body></pre>	<pre><card> <p>Alle Tags in WML müssen klein geschrieben werden.</p> </card></pre>

2.1.2.3 Attribute

Attribute sind entweder nähere Erläuterungen einzelner Elemente oder Metadaten zu einzelnen Elementen. So kann man in HTML beispielsweise die Höhe und Breite eines Bildes durch die Attribute `height` und `width` bestimmen. In XML gibt es die Konvention, dass Werte von definierten Attributen in Anführungszeichen eingeschlossen werden müssen. Zusätzlich müssen Attributnamen in WML klein geschrieben werden. In HTML macht es keinen Unterschied, ob man den Attributwert in Anführungszeichen setzt und Groß- und Kleinschreibung beachtet. Der HTML-Browser interpretiert beide Versionen korrekt.

¹⁵ Vgl. BEHME, Henning (1999), S.39

Sprachen des mobilen Internet – eine Analyse

Beispiel 5: HTML vs. WML – Attribute in Anführungszeichen

HTML	WML
<pre><BODY bgcolor=silver> <P align=right>Die Attri- butwerte können in Anfüh- rungszeichen stehen, müs- sen aber nicht. </BODY></pre>	<pre><card id="card1" title="Willkommen"> <p>Alle Attributwerte müssen von Anführungszeichen umschlossen sein.</p> </card></pre>

2.1.2.4 Entity-Referenzen

In XML können Abkürzungen definiert werden, die als Platzhalter für beliebige Texte stehen. Damit hat man die Möglichkeit, oft wiederkehrende Textteile in gekürzter Form einzubetten, die dann vom verarbeitenden Programm durch den kompletten Textteil ersetzt werden. Eine weitere Anwendung finden Entity-Referenzen bei Zeichen, denen in XML eine besondere Bedeutung zukommen, wie etwa spitze Klammern, Anführungszeichen, Apostroph und das et-Zeichen (&). Sollen diese Zeichen direkt im Text vorkommen, müssen sie kodiert werden. Jedes dieser Zeichen hat dabei einen Namen, der von & und ; eingeschlossen wird.

Die folgende Tabelle zeigt die wichtigsten Entity-Referenzen und deren Kodierung in WML¹⁶:

Tabelle 1: Sonderzeichen in WML

Entity-Referenz	Zeichen	Beschreibung
&;	&	et-Zeichen (Ampersand)
<;	<	kleiner als
>;	>	größer als
';	'	Apostroph
";	"	Anführungszeichen
 ;		Non-breaking Space ¹⁷
­;		Soft-hyphen ¹⁸

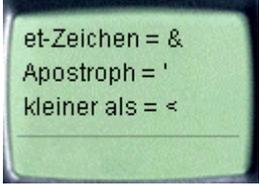
¹⁶ Vgl. WAP FORUM (1999), S.12

¹⁷ Setzt ein Leerzeichen, an dem kein Zeilenumbruch stattfinden soll.

¹⁸ Zeigt dem Microbrowser an, dass hier eine sinnvolle Stelle für einen möglichen Zeilenumbruch ist.

Sprachen des mobilen Internet – eine Analyse

Beispiel 6: Sonderzeichen in WML

Nokia WAP Toolkit 1.2	WML
 <p>Abbildung 6: Sonderzeichen</p>	<pre>... <card> <p> et-Zeichen = &amp;
 Apostroph = &apos;
 kleiner als = &lt; </p> </card> ...</pre>

In WML ist explizit definiert, dass die Sonderzeichen "kleiner als" (<) und das et-Zeichen (&) kodiert werden müssen, wenn sie innerhalb des eigentlichen Textes vorkommen sollen. Es ist aber ratsam, alle Sonderzeichen aus der obigen Tabelle zu kodieren.

2.1.2.5 Kommentare

Kommentare sind in WML dort erlaubt, wo auch normaler Text stehen darf. Kommentare gehören nicht zum Dokument und werden ignoriert. Außerdem werden sie vor der Umsetzung in Bytecode aus dem Dokument entfernt. Es gilt dieselbe Struktur wie in XML.

```
<!-- Dies ist ein WML-Kommentar -->
```

Obwohl die oben genannten Restriktionen nicht für HTML gelten, gehört es in Webdesignerkreisen zum guten Ton, die Voraussetzungen von XML auch auf HTML anzuwenden.¹⁹ Eine Ausnahme bildet dabei die Behandlung von leeren Elementen, da man nicht davon ausgehen kann, dass jeder HTML-Browser versteht, was etwa mit `
` gemeint ist.

2.1.3 WML 1.1

Im Juni 1999 verabschiedete das WAP Forum im Einklang mit dem W3C den WAP Standard 1.1, der auch WML 1.1 umfasst.²⁰ Im Folgenden werden nun die wichtigsten Merkmale von WML aufgezeigt.

¹⁹ Vgl. BEHME, Henning; MINTERT, Stefan (1998), S.26

²⁰ Vgl. http://www.wapforum.org/what/technical_1_1.htm (29.05.2000)

Sprachen des mobilen Internet – eine Analyse

2.1.3.1 Dokumenten Prolog

Da WML eine Sprache von XML ist, muss jedes WML-Dokument mit einer sogenannten XML-Deklaration, gefolgt von einer Dokumenttyp-Deklaration, beginnen. Die XML-Deklaration gibt an, dass es sich bei dem Dokument um ein XML-Dokument handelt. Die Deklaration sieht folgendermaßen aus.

```
<?xml version="1.0"?>
```

In der Dokumenttyp-Deklaration wird auf die dem Dokument zugrunde liegende DTD verwiesen. Diese enthält alle erlaubten Elemente, Attribute usw., die in dem Dokument verwendet werden dürfen. In diesem Fall wird auf die DTD von WML 1.1 verwiesen. Die Dokumenttyp-Deklaration sieht wie folgt aus.

```
<!DOCTYPE wml PUBLIC "-//WAPFORUM//DTD WML 1.1//EN"
"http://www.wapforum.org/DTD/wml_1.1.xml">
```

Besitzt ein XML-Dokument die XML- und Dokumenttyp-Deklaration, werden sie "gültig" genannt.²¹ In der WML-Spezifikation wird explizit darauf hingewiesen, dass es nicht erlaubt ist, die beiden Deklarationen wegzulassen.²² Aus Gründen der Übersichtlichkeit, wird in den nachfolgenden WML-Beispielen auf die XML- und Dokumenttypdeklaration verzichtet.

2.1.3.2 Standardelemente

In diesem Kapitel werden die wichtigsten Elemente der WML-Spezifikation 1.1 herausgearbeitet und mit Beispielen verdeutlicht. Für eine komplette Referenz wird auf die WML-Spezifikation des WAP-Forums verwiesen.²³

Das Element `wml` ist das Wurzelement, das alle weiteren Elemente einer WML-Datei hierarchisch geschachtelt enthält. Auch das Element `card` wurde schon angesprochen. Es ist das Basiselement in WML. Zu den wichtigsten Attributen zählt `id`, das eine eindeutige Benennung ermöglicht und so eine Grundvoraussetzung für die Navigation zwischen Cards darstellt. Weiterhin wichtig ist das Attribut `title`, dessen Wert beim Anzeigen der Card als Überschrift eingeblendet werden kann. Steht der Wert des Attributs `newcontext` auf "true" kann man beim Laden der Card erreichen, dass ein Reset, also ein Zurücksetzen der Histo-

²¹ Vgl. BEHME, Henning; MINTERT, Stefan (1998), S.62

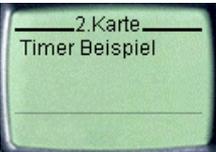
²² Vgl WAP FORUM (1999), S.31

²³ WAP FORUM (1999)

Sprachen des mobilen Internet – eine Analyse

ryfunktion und der Variablen erfolgt. Lässt man das Attribut weg, ist standardmäßig der Wert "false" eingestellt. Das Attribut `ontimer` reagiert, wenn ein zu definierender Timer abläuft. Nach Ablauf des Timers wird zu einer angegebenen URL gesprungen.

Beispiel 7: Timer in WML

Nokia WAP Toolkit 1.2	WML
	<pre><wml> <card id="kartel" title="Timer_Bsp" ontimer="#karte2" newcontext="true"> <timer value="20"/> <p> 1. Karte </p> </card> <card id="karte2" title="2.Karte"> <p> Timer Beispiel </p> </card> </wml></pre>
	

Das Element `<timer value="20">` setzt einen Countdown von 20 Millisekunden. Nach Ablauf der Zeit, wird über das Attribut `ontimer` die Card mit der `id="karte2"` aufgerufen.

WML bietet die Möglichkeit, Aktionen global zu definieren, so dass sie für jede Card innerhalb eines Deck gelten. Diese Aktionen repräsentieren Navigationsoptionen. Damit kann eine wiederholte Definition der Optionen innerhalb jeder Card entfallen. Dieses erreicht man mit dem Element `template`, das direkt nach dem Tag `<wml>` steht.

Sprachen des mobilen Internet – eine Analyse

Beispiel 8: Template in WML

Nokia WAP Toolkit 1.2	WML
 <p>Abbildung 9: Template 1</p>  <p>Abbildung 10: Template 2</p>	<pre data-bbox="576 389 1145 920"><wml> <template> <do type="prev" label="Zurück"> <prev/> </do> </template> <card id="kartel" title="Willkommen"> <do type="accept" label="weiter"> <go href="#karte2"/> </do> <p> Template Beispiel </p> </card> <card id="karte2" title="Willkommen"> <p> 2. Karte </p> </card> </wml></pre>

Jede Card, die innerhalb des Deck definiert wird, erhält automatisch die Navigationsoption `Zurück`. Das heißt, von jeder Card aus, kann die Historyfunktion aufgerufen werden, die dann die zuletzt betrachtete Card wieder anzeigt. In diesem Beispiel wird automatisch eine "programmierbare" Taste (Softkey) des mobilen Endgeräts mit der Funktion `Back` belegt. Über die Funktion `Options` in Abbildung 9 können die weiteren definierten Aktionen abgerufen werden, was in dem Beispiel die Aktion `weiter` ist. Diese Aktion ruft die 2.Karte, `karte2`, des Deck auf. Die Aktionen innerhalb von `template` können durch Aktionen innerhalb der Cards überschrieben werden. So kann beispielsweise eine Aktion `do` mit dem Attribut `name="aktion"` im Template mit einer Aktion `do` mit demselben Attribut `name="aktion"` in der Card überschrieben werden.

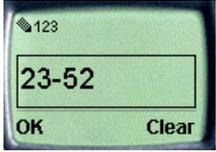
2.1.3.3 Benutzereingaben

In WML gibt es verschiedene Möglichkeiten, Benutzereingaben zu realisieren. Zum einen gibt es das Element `input`, das ein Eingabefeld definiert. Mit dem obligatorischen Attribut `name`, wird gleichzeitig eine Variable mit dem Namen des Wertes aus `name` definiert. Mit dem Attribut `value` wird der Ausgangswert dieser Variablen, die unter `name` definiert wurde, gesetzt. Das `input`-Element ermöglicht über das Attribut `format` die Definition eines einfachen Eingabemusters. Dieses legt fest, wie die Daten auszusehen haben, die eingegeben werden sollen. So sind unter anderem folgende Definitionen möglich: Nur nicht-nummerischer Großbuchstabe (`format="A"`), nur nicht-nummerischer Kleinbuchstabe (`format="a"`), nur num-

Sprachen des mobilen Internet – eine Analyse

merisches Zeichen (`format="N"`). Zusätzlich gibt es noch das Attribut `type`, dessen Wert entweder `"text"` oder `"password"` sein kann, falls es `"password"` ist, werden die eingegebenen Zeichen unkenntlich gemacht. Ferner existieren die Attribute `emptyok`, `size`, `maxlength` und `title`.

Beispiel 9: Benutzereingaben (input)

Nokia WAP Toolkit 1.2	WML
 <p>Abbildung 11: Input 1</p>	<pre><wml> <card id="kartel" title="Willkommen" newcontext="true"> <p> Bitte PIN eingeben:
 <input name="pin" format="NN\ -NN"/> </p> </card> </wml></pre>
 <p>Abbildung 12: Input 2</p>	

In diesem Beispiel wird mit dem `input`-Element eine Variable mit dem Namen `"pin"` initialisiert. Dieser wird später die Eingabe des Benutzers als Wert zugewiesen. Als Eingabemuster wurde definiert, dass nach der Eingabe von zwei numerischen Zeichen (`NN`), automatisch ein Bindestrich eingefügt wird (`\ -`), worauf die erneute Eingabe von zwei numerischen Zeichen folgt (`NN`).

Eine weitere Variante, Benutzereingaben zu realisieren, sind sogenannte Auswahllisten, die es in ähnlicher Form auch in HTML gibt. Auswahllisten werden mit dem Element `select` definiert. Dabei können Auswahllisten entweder die Auswahl mehrerer Felder oder nur die Auswahl eines Feldes zulassen. Diese Einstellung wird mit dem Attribut `multiple` vorgenommen. Ist der Wert `"true"`, ist es möglich mehrere Optionen auszuwählen, ist der Wert `"false"`, kann nur eine Option gewählt werden. Wird das Attribut nicht definiert, ist der Standardwert auf `"false"` gesetzt. Hier zunächst ein Beispiel, bevor die weiteren Attribute erläutert werden.

Sprachen des mobilen Internet – eine Analyse

Beispiel 10: Benutzereingaben (select-mehrfach)

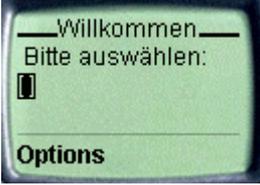
Nokia WAP Toolkit 1.2	WML
 <p>Abbildung 13: Auswahl 1</p>  <p>Abbildung 14: Auswahl 2</p>	<pre data-bbox="520 443 1150 817"><wml> <card id="kartel" title="Willkommen" newcontext="true"> <p> Bitte auswählen:
 <select iname="I" ivalue="1;3" name="var" value="spo" multiple="true"> <option value="new">News</option> <option value="spo">Sport</option> <option value="wi">Wirtschaft</option> </select> </p> </card> </wml></pre>

Die Attribute `iname` und `ivalue` dienen der Vorauswahl verfügbarer Optionen. Mit dem Attribut `iname` wird eine Variable definiert, die den Wert aus `ivalue` zugewiesen bekommt. Jede Auswahloption, die in einer Auswahlliste erscheinen soll, muss mit dem Element `option` definiert werden. In diesem Beispiel hat `ivalue` den Wert "1;3", das heißt die Optionen 1 (News) und 3 (Wirtschaft) sind vorausgewählt (siehe Abbildung 13). Das Attribut `iname` hätte dementsprechend den Wert "News; Wirtschaft". Ist `ivalue` mit "0" definiert, wird keine Vorselektion getroffen. Das Attribut `name` definiert eine Variable, in der die Attributwerte aus `value` der ausgewählten Optionen gespeichert werden. Da hier die Optionen 1 und 3 vorausgewählt sind, nimmt die Variable `var` den Wert "new;wi" an.

Bei dem Element `option` gibt es ferner die Möglichkeit anzugeben, bei Anwahl einer Option zu einer neuen URL zu springen. Dies wird mit Hilfe des Attributs `onpick` erreicht.

Sprachen des mobilen Internet – eine Analyse

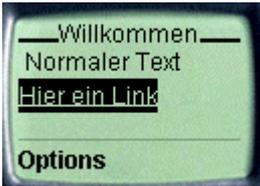
Beispiel 11: Benutzereingaben (select-einfach)

Nokia WAP Toolkit 1.2	WML
 <p style="text-align: center;">Abbildung 15: Auswahl 3</p>  <p style="text-align: center;">Abbildung 16: Auswahl 4</p>	<pre> <wml> <card id="kartel" title="Willkommen" newcontext="true"> <p> Bitte auswählen:
 <select iname="I" ivalue="0" name="var"> <option value="new" onpick="../news.wml">News</option> <option value="spo">Sport</option> <option value="wi">Wirtschaft</option> </select> </p> </card> </wml> </pre>

Bei diesem Beispiel wird die Datei `news.wml` aufgerufen, wenn die Option `News` angewählt wird. Da hier das Attribut `ivalue="0"` ist, ist keine der Optionen vorausgewählt.

Eine wichtige Funktion bei HTML übernimmt der Hypertext, die bekannte Verknüpfung von Dokumenten und Dokumentstellen untereinander. WML hat dieses Prinzip übernommen.

Beispiel 12: Verlinkung

Nokia WAP Toolkit 1.2	WML
 <p style="text-align: center;">Abbildung 17: Link</p>	<pre> <wml> <card id="kartel" title="Willkommen"> <p> Normaler Text
 Hier ein Link
 </p> </card> </wml> </pre>

Das Attribut `href` im Element `a` enthält die URL, auf die verwiesen wird, und ist damit zwingend vorgeschrieben. Der Link kann hier über den Softkey `Options` aktiviert werden. Es sei erwähnt, dass der Umgang mit integrierten Funktionen von mobilem Endgerät zu mobilem Endgerät abweicht. Das bedeutet, dass ein anderes mobiles Endgerät, eventuell eine andere Vorgehensweise für das Aktivieren eines Hyperlink bereitstellt.

Sprachen des mobilen Internet – eine Analyse

2.1.3.4 Formatierung und Grafiken

WML unterstützt den Einsatz von Grafiken. Die Grafiken müssen aufgrund der Restriktionen der Displays von mobilen Endgeräten eine Farbtiefe von einem Bit, also schwarz und weiß, aufweisen. Das Dateiformat von Grafiken in WML ist Wireless Bitmap (WBMP). Die Dateierweiterung ist entsprechend `.wbmp`. Das Einbinden der Grafiken in eine Card erfolgt wie bei HTML über das Element `img`. Das `img`-Element in WML besitzt zwei obligatorische Attribute, `src` und `alt`. Wird in HTML nur empfohlen, einen alternativen Text mit Hilfe des Attributs `alt` zu der Grafik anzugeben, ist es in WML Pflicht einen solchen Text über `alt` anzugeben.

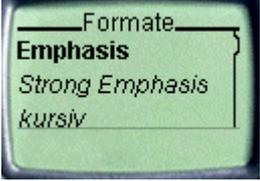
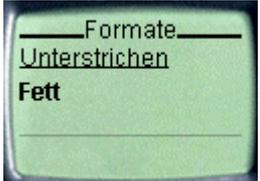
Beispiel 13: Einbinden von Grafiken

Nokia WAP Toolkit 1.2	WML
 <p data-bbox="245 1037 450 1066">Abbildung 18: Grafik</p>	<pre data-bbox="515 835 1203 1093"><wml> <card id="karte1" title="Willkommen"> <p> </p> </card> </wml></pre>

Bei diesem Beispiel sieht man, dass eine Grafik auch im Zusammenhang mit einem Link eingebettet werden kann. Konnte die Grafik nicht geladen werden, wird der Wert aus dem Attribut `alt` angezeigt. Das Element `img` hat weitere Attribute, wie `align` zur Ausrichtung der Grafik, `height` und `width` für die Angabe der Größe, `localsrc` für die Definition einer internen Grafik, die alternativ zur angegebenen Grafik verwendet werden soll. Gibt es diese alternative Grafik, hat sie Vorrang.

WML bietet in der Version 1.1 einfache Funktionen zur Textformatierung. Allerdings unterstützen nicht alle momentan verfügbaren mobilen Endgeräte diese Features. Die Elemente orientieren sich, mit zwei Ausnahmen, an den aus HTML bekannten Elementen zur Textformatierung.

Beispiel 14: Textformatierung

Nokia WAP Toolkit 1.2	WML
 <p>Abbildung 19: Textformat 1</p>	<pre><wml> <card id="kartel" title="Formate"> <p> Emphasis
 Strong Emphasis
 <i>kursiv</i> </p> </card> </wml></pre>
 <p>Abbildung 20: Textformat 2</p>	<pre><wml> <card id="kartel" title="Formate"> <p> <u>Unterstrichen</u>
 Fett
 </p> </card> </wml></pre>
 <p>Abbildung 21: Textformat 3</p>	<pre><wml> <card id="kartel" title="Formate"> <p> <small>Kleine Schrift</small>
 <big>Große Schrift</big>
 </p> </card> </wml></pre>

Die Elemente `em` und `strong` geben dem Microbrowser den Hinweis, dass der Text hervorgehoben werden soll. Wie das im Einzelnen aussieht, ist dem Microbrowser überlassen. Die Elemente `i`, `u` und `b` sind aus Textverarbeitungsprogrammen bekannt und zeigen an, dass Text entweder kursiv (`i`), unterstrichen (`u`) oder fett (`b`) dargestellt wird. Eine kleinere Schriftgröße als die Standardschrift, erreicht man durch die Tags `<small></small>`. Entsprechend erhält man eine größere Darstellung des Textes durch die Tags `<big></big>`. In der Spezifikation von WML 1.1 des WAP Forums wird empfohlen die Elemente `b`, `i` und `u` nur zu verwenden, wenn ausdrückliche Kontrolle über die Darstellung des Textes notwendig ist. Standardmäßig soll auf die Elemente `em` und `strong` zurückgegriffen werden.²⁴

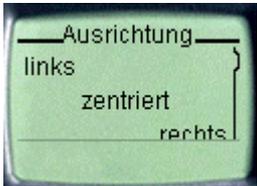
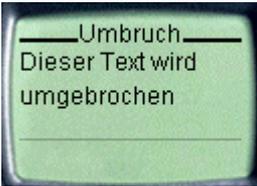
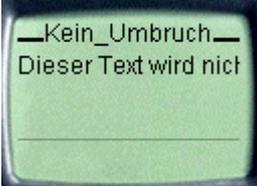
Auf die Elemente `br` und `p` wird nur kurz eingegangen, da sie in den Beispielen schon häufig verwendet wurden und aus HTML bekannt sind. Mit dem leeren Element `br` erreicht man ei-

²⁴ WAP FORUM (1999), S.48

Sprachen des mobilen Internet – eine Analyse

nen Zeilenumbruch. Innerhalb der Tags `<p>` und `</p>` muss der inhaltliche Teil eines WML-Dokuments stehen, also Text, der in irgendeiner Form ausgegeben werden soll. Das heißt im Umkehrschluss, dass außerhalb des Elements `p`, kein auszugebender Text stehen darf. Durch das Attribut `align` des Elements `p` ist es möglich, dem Microbrowser die Textausrichtung, also links, rechts oder zentriert, anzuzeigen. Das Attribut `mode` definiert, ob der Text innerhalb von `<p>` und `</p>` umbrochen werden soll, wenn eine Zeile zu lang für die Anzeige ist.

Beispiel 15: Textausrichtung

Nokia WAP Toolkit 1.2	WML
 <p data-bbox="236 981 493 1010">Abbildung 22: Ausrichtung</p>	<pre data-bbox="555 772 1208 958"> <wml> <card id="kartel" title="Ausrichtung"> <p align="left">links</p> <p align="center">zentriert</p> <p align="right">rechts</p> </card> </wml> </pre>
 <p data-bbox="236 1256 493 1285">Abbildung 23: Umbruch</p>	<pre data-bbox="555 1048 1208 1234"> <wml> <card id="kartel" title="Umbruch"> <p mode="wrap"> Dieser Text wird umgebrochen </p> </card> </wml> </pre>
 <p data-bbox="236 1536 493 1565">Abbildung 24: Kein Umbruch</p>	<pre data-bbox="555 1328 1208 1514"> <wml> <card id="kartel" title="Kein_Umbruch"> <p mode="unwrap"> Dieser Text wird nicht umgebrochen </p> </card> </wml> </pre>

Ist kein Zeilenumbruch definiert, ist es Aufgabe des Microbrowsers dafür zu sorgen, dass der komplette Text zugänglich ist.

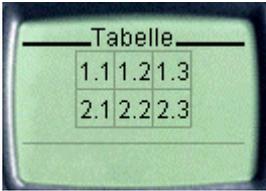
Ein sehr häufig verwendetes Werkzeug, um Layoutfunktionen in HTML zu übernehmen, sind Tabellen. WML unterstützt ebenfalls den Einsatz von Tabellen, allerdings sind die Möglichkeiten des Layouts natürlich wegen der Displaygrößen weniger umfangreich. Um eine Tabelle in WML zu erzeugen, benötigt man die Elemente `table`, `tr` und `td`. Eine Tabelle besteht aus Spalten und Zeilen. Das Element `table` ist das Basiselement. Durch das obligato-

Sprachen des mobilen Internet – eine Analyse

rische Attribut `columns` wird die Anzahl der Spalten definiert. Das Element `tr` bezeichnet jeweils eine Tabellenzeile und das `td`-Element definiert die einzelne Tabellenzelle. Das Grundgerüst einer Tabelle, die zwei Spalten und eine Zeile hat, sieht also wie folgt aus:

```
<table columns="2"><tr><td></td><td></td></tr></table>
```

Beispiel 16: Einsatz von Tabellen

Nokia WAP Toolkit 1.2	WML
 <p style="text-align: center;">Abbildung 25: Tabelle</p>	<pre><wml> <card id="kartel" title="Tabelle"> <p align="center"> <table columns="3"> <tr> <td>1.1</td> <td>1.2</td> <td>1.3</td> </tr> <tr> <td>2.1</td> <td>2.2</td> <td>2.3</td> </tr> </table> </p> </card> </wml></pre>

In diesem Beispiel ist eine Tabelle mit drei Spalten und zwei Zeilen abgebildet. Aus der Definition von `columns="3"` folgt zwingend die Darstellung von drei `td`-Elementen innerhalb jedes `tr`-Elements. Der Einsatz von Tabellen sollte gut überlegt sein, da zwar die entsprechenden Entwicklungstools Tabellen unterstützen (hier: Nokia Toolkit 1.2), die auf dem Markt befindlichen WAP-fähigen mobilen Endgeräte (hier das Nokia 7110) aber teilweise eine etwas zweifelhafte Unterstützung bieten.

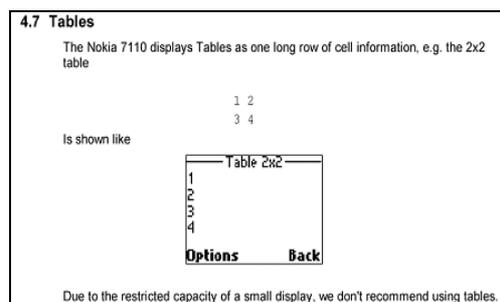


Abbildung 26: Ausschnitt aus dem Nokia7110 Developer's Guide²⁵

²⁵ Vgl. NOKIA (1999b), S.24

Sprachen des mobilen Internet – eine Analyse

2.1.3.5 Ereignismodell in WML

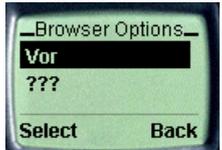
WML bietet die Möglichkeit, auf verschiedene Ereignisse zu reagieren. Ereignisse können ein ablaufender Timer, das Navigieren zu einer Card durch einen Link oder durch den Aufruf der Historyfunktion oder die Auswahl einer Option in einem Auswahlmenü sein. In dem vorangegangenen Abschnitt wurden die Ereignisse bzw. deren Behandlung schon vorweggenommen. Dieser Abschnitt fasst die Ereignisse und deren Aktionen sowie die Navigationselemente zusammen.

Zunächst ist das Element `do` zu nennen, das einen allgemeinen Rahmen zur Ausführung von Aktionen bereitstellt. Welche Aktionen ausgeführt werden, wird in eingebetteten Elementen definiert. Das `do`-Element besitzt unter anderem das obligatorische Attribut `type`, für das verschiedene reservierte Werte existieren. Diese sollen dem Microbrowser anzeigen, welchen Inhalt die Aktion repräsentiert. In der folgenden Tabelle sind die reservierten Werte aufgelistet.

Tabelle 2: Reservierte Werte für das Attribut "type" des Elements "do"

Wert	Beschreibung
accept	Positive Ausführung einer Aktion
prev	Rückwärtsnavigieren mit der Historyfunktion
help	Anfrage an Hilfestellungen
reset	Löschen oder auf Standardeinstellung zurücksetzen
options	Angebot an zusätzlichen Funktionen
delete	Löschen einer Auswahl
unknown	Keine nähere Angabe

Beispiel 17: Element "do"

Nokia WAP Toolkit 1.2	WML
 <p>Abbildung 27: Aktionen 1</p>	<pre> <wml> <card id="karte1" title="Willkommen"> <do type="accept" name="do1" label="Vor"> <go href="#karte2"/> </do> <do type="help" name="do2" label="???"> <go href="http://wap.de/help.wml"/> </do> <do type="prev" name="do3" label="Zurück"> <prev/> </do> <p> Ordentlich Aktionen </p> </card> </wml> </pre>
 <p>Abbildung 28: Aktionen 2</p>	

Sprachen des mobilen Internet – eine Analyse

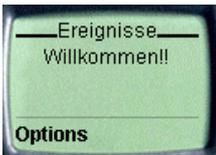
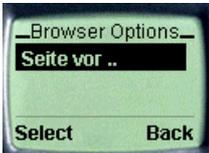
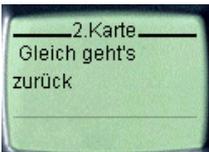
Weiterhin sind die Attribute `label` und `name` wichtig. Das Attribut `label` gibt eine Art Inhaltsangabe, die der Microbrowser verwenden kann, um ein aussagekräftiges User Interface Element zu generieren wie zum Beispiel einen Softkey mit der Aufschrift des Labels. Über das Attribut `name` kann dem Element ein eindeutiger Name gegeben werden. In dem Beispiel 17 hat sich der Microbrowser "entschieden", zwei Softkeys zu generieren, einen mit `Options` und einen mit `Back`. Unter `Options` befinden sich die zwei `do`-Elemente `Vor` und `???`. Hinter `Back` sitzt die Funktion, eine Card zurückzuspringen.

Die Definition der Reaktionen auf bestimmte Ereignisse, kann einmal als Attribut verschiedener Elemente erfolgen. Zum Beispiel kann das `card`-Element ein Attribut `ontimer` besitzen (siehe Beispiel 7).

```
<card id="Kartel" title="Hallo" ontimer="http://wap.de/1.wml">
```

Die andere Möglichkeit ist das Element `onevent`. Das Element `onevent` ist für alle auftretenden Ereignisse definierbar. Die nähere Definition zu einem bestimmten Ereignis erfolgt über das obligatorische Attribut `type`.

Beispiel 18: Ereignisse in WML

Nokia WAP Toolkit 1.2	WML
 <p>Abbildung 29: Ereignis 1</p>	<pre><wml> <card id="kartel" title="Ereignisse"> <do type="accept" label="Seite vor .."> <go href="#karte2"/> </do> <p align="center"> Willkommen!! </p> </card> <card id="karte2" title="2.Karte"> <onevent type="ontimer"> <prev/> </onevent> <timer value="30"/> <p> Gleich geht's zurück </p> </card> </wml></pre>
 <p>Abbildung 30: Ereignis 2</p>	
 <p>Abbildung 31: Ereignis 3</p>	

Sprachen des mobilen Internet – eine Analyse

Bei diesem Beispiel sieht man, dass in der zweiten Card eine Reaktion auf ein Ereignis, hier das Ablaufen eines Timers, definiert wurde. Bei Ablauf der angegebenen Zeit, wird die vorhergehende Seite aufgerufen.

Das Element `onevent` bietet eine flexiblere Verwendung, als dies mit den Attributen der Elemente `card` oder `timer` möglich ist. Flexibler deshalb, weil das Attribut nur eine URL enthalten kann, das Element `onevent` allerdings noch Aktionen wie `noop`, `prev` oder `refresh`. Wie schon erwähnt, wird über das Attribut `type` des Elements `onevent` definiert, auf welche Ereignisse mit einer Aktion reagiert werden soll. Neben der Möglichkeit, auf einen Timer zu reagieren (`ontimer`), kann `type` noch die Werte `onenterforward`, `onenterbackward` und `onpick` annehmen.

Ist der Wert `onenterforward`, tritt eine Aktion ein, wenn die entsprechende Card durch den Aufruf einer URL geladen wurde, die Card also "von vorne betreten wurde". Im Gegensatz dazu, tritt `onenterbackward` in Aktion, wenn eine Card durch den Aufruf der Historyfunktion geladen wird, also ein "Rückwärtsbetreten der Card". Mit dem Wert `onpick`, wird eine Aktion bei Auswahl einer Option aus einem Auswahlménü (siehe Beispiel 11) ausgelöst. Auf die oben genannten Ereignisse kann, bis auf `onpick`, auch mit Hilfe der gleichnamigen Attribute der Elemente `card` und `template` reagiert werden.

Zwei mögliche Aktionen auf Ereignisse, einmal der Sprung zu einer neuen Card durch eine URL und der Sprung zu einer bereits besuchten Card durch den Aufruf der Historyfunktion, wurden bereits erläutert. Es gibt noch zwei weitere Varianten, auf Ereignisse zu reagieren. Das Element `refresh` wird im Zusammenhang mit Variablen verwendet und kann diese, als Reaktion auf ein Ereignis, in einen bestimmten Zustand versetzen.

Beispiel 19: Aktion refresh

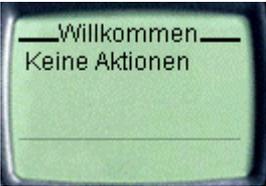
Nokia WAP Toolkit 1.2	WML
 <p>Abbildung 32: refresh</p>	<pre><wml> <card id="kartel" title="1.Karte"> <onevent type="ontimer"> <refresh> <setvar name="anzahl" value="20"/> </refresh> </onevent> <timer value="30"/> <p> Variablenwert: \$anzahl </p> </card> </wml></pre>

Sprachen des mobilen Internet – eine Analyse

Hier wird die Aktion `refresh` ausgeführt, sobald der definierte Timer abgelaufen ist. Dabei wird eine Variable mit dem Namen `"anzahl"` und dem Wert `"20"` initialisiert. Existiert die Variable `"anzahl"` vor der Aktion bereits, wird ihr der Wert `"20"` übergeben. Näheres zu Variablen in Kapitel 2.1.3.6.

Wie bereits erwähnt, kann man durch das Element `template` jeder Card eines Deck verschiedene Navigationsoptionen und Ereignisreaktionen zur Verfügung stellen. Man stelle sich nun vor, man hat sechs Cards innerhalb eines Deck und möchte jeder, außer der ersten Card die Möglichkeit geben, die Historyfunktion aufzurufen. Da das Element `template` grundsätzlich jeder Card alle Funktionen zur Verfügung stellt, gibt es die Möglichkeit innerhalb der Cards, einzelne Funktionen wieder auszuschalten bzw. zu überschreiben. Das Abschalten einzelner Templatefunktionen innerhalb der Cards, wird mit der Aktion `<noop/>` erreicht.

Beispiel 20: Ausschalten von Templatefunktionen

Nokia WAP Toolkit 1.2	WML
 <p>Abbildung 33: noop</p>	<pre><wml> <template> <do type="prev" name="do" label="Zurück"> <prev/> </do> </template> <card id="karte1" title="Willkommen"> <do type="prev" name="do"> <noop/> </do> <p> Keine Aktionen </p> </card> </wml></pre>

Im Template ist die Möglichkeit, die Historyfunktion aufzurufen, also eine Card zurückzugehen, definiert. Durch die Definition von `do` in der Card, mit demselben Attribut `name`, wird das `do` im Template überschrieben. Das heißt konkret, dass in der Card dem Benutzer keine Aktionen angeboten werden.

Näher eingegangen werden muss auf das Element `go`, das im einfachsten Fall einen Verweis auf eine andere Card oder ein anderes Deck enthält. In einer erweiterten Form können Informationen definiert werden, die bei der Aktivierung des Verweises mitübertragen werden. Eine solche erweiterte Form zeigt Beispiel 21.

Sprachen des mobilen Internet – eine Analyse

Beispiel 21: Methode "get" beim Element "go"

Nokia WAP Toolkit 1.2	WML
 <p data-bbox="260 663 509 689">Abbildung 34: method get</p>	<pre data-bbox="596 389 1206 707"><wml> <card id="kartel" title="Willkommen"> <do type="accept" label="Send"> <go href="/neu" method="get"> <postfield name="x" value="1"/> </go> </do> <p> Daten mitsenden </p> </card> </wml></pre>

Durch das Attribut `method` kann die HTTP-Übertragungsart definiert werden, mit der die Daten übertragen werden sollen. Der Wert kann entweder `get` oder `post` sein.

Hat das Attribut den Wert `get` und sind zusätzlich Daten für die Übertragung definiert, werden diese beim Ausführen der Aktion an die angeforderte URL angehängt. In Beispiel 21 wird eine Anforderung an die URL `/neu?x=1` abgesetzt.

Beispiel 22: Methode "post" beim Element "go"

Nokia WAP Toolkit 1.2	WML
 <p data-bbox="244 1485 507 1512">Abbildung 35: method post</p>	<pre data-bbox="577 1200 1209 1541"><wml> <card id="kartel" title="Willkommen"> <do type="accept" label="Send"> <go href="/neu" method="post"> <postfield name="x" value="1"/> <postfield name="y" value="2"/> </go> </do> <p> Daten mitsenden </p> </card> </wml></pre>

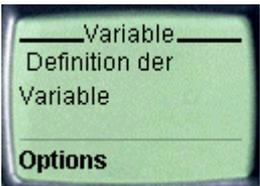
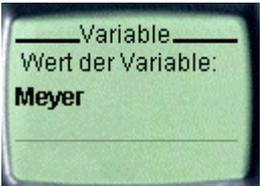
Die Methode `post` wird in der Regel nur verwendet, wenn der Einsatz der Methode `get` nicht möglich ist, weil die zu übertragenden Daten aufgrund der Menge nicht als Bestandteil der URL gesendet werden können. Mit dieser Methode wird ein gesondertes Datenfeld übertragen, was die spezifizierten Daten enthält. In diesem Fall also `x=1&y=2`. Das Element `postfield` definiert dabei die zu übertragenden Daten. Die Attribute `name` und `value` sind in diesem Zusammenhang obligatorisch.

2.1.3.6 Zustandsmodell / Variablen

Ein weiterer großer Unterschied zwischen WML und HTML ist die Möglichkeit, Variablen einzusetzen. Diese unterstützen den Austausch von Statusinformationen und Parametern zwischen unterschiedlichen Decks und Cards. Damit entsteht eine bessere Möglichkeit, auf Benutzereingaben zu reagieren. Die Variablen können im gesamten Kontext eines oder mehrerer Decks definiert und verwendet werden.

Die Definition von Variablen erfolgt unter anderem über das Element `setvar`. Dieses Element kann innerhalb der Aktionen `go`, `prev` und `refresh` eingesetzt werden. Der Name der Variable wird über das obligatorische Attribut `name` und der Wert über das ebenfalls obligatorische `value` vergeben. Zur Verdeutlichung der Einsatzmöglichkeiten soll Beispiel 23 dienen.

Beispiel 23: Definition von Variablen

Nokia WAP Toolkit 1.2	WML
 <p>Abbildung 36: Variablen 1</p>	<pre> <wml> <card id="Karte1" title="Variable"> <do type="accept" label="Var_Def"> <go href="#karte2"> <setvar name="nachname" value="Meyer"/> </go> </do> <p> Definition der Variable </p> </card> <card id="Karte2" title="Variable"> <p> Wert der Variable:
 \$(nachname) </p> </card> </wml> </pre>
 <p>Abbildung 37: Variablen 2</p>	

In diesem Beispiel wird durch den Aufruf der zweiten Card eine Variable definiert, die den Namen "Nachname" und den Wert "Meyer" besitzt. Die zweite Card enthält dabei eine Referenz auf die Variable. Das bedeutet, dass der Wert der Variable beim Anzeigen der Card für diese Referenz eingesetzt wird. Die WML-Spezifikation erlaubt verschiedene Schreibweisen, wie Referenzen auf Variablen gesetzt werden können. Gemeinsam haben die Varianten das vorangestellte Dollarzeichen. Im obigen Beispiel ist der Name der Variablen zusätzlich von Klammern eingeschlossen, die aber auch weggelassen werden können, wenn die Variable aus dem Zusammenhang klar erkannt werden kann. Demzufolge wäre hier auch `$(nachname)` erlaubt. Allerdings ist die Schreibweise mit Klammern zu bevorzugen, da sie weniger fehler-

Sprachen des mobilen Internet – eine Analyse

anfällig ist. Um ein einfaches Dollarzeichen im eigentlichen Text zu definieren, ist die Eingabe von zwei Dollarzeichen, nach dem Schema "für ein \$\$-Zeichen", nötig.

In Kapitel 2.1.3.3 wurde, bei dem Einsatz von Auswahl-, Optionslisten und Eingabefeldern, schon eine weitere Möglichkeit mit Variablen zu arbeiten, besprochen.

2.1.4 WML 1.2

Das WAP-Forum verabschiedete am 4. November 1999 die WML-Spezifikation Version 1.2.²⁶ Allerdings ist diese Version, wie schon erwähnt, noch nicht vom W3-Consortium beurteilt worden und noch nicht in den momentan verfügbaren Microbrowsern implementiert. Die Veränderungen zur Version 1.1 sind ferner minimal. Infolgedessen gibt dieses Kapitel nur einen kurzen Überblick über die wichtigsten Veränderungen zur Version 1.1.

Die Elemente `anchor`, `a` und `input` besitzen jetzt ein Attribut `accesskey`. Dieses Attribut soll eine vereinfachte Navigation bzw. Ansteuerung des Elements ermöglichen. Der Inhalt des Attributs ist ein einfaches Zeichen, das auf möglichst jedem mobilen Endgerät als Taste verfügbar sein sollte. Für Mobiltelefone wären das beispielsweise die Zahlen "1"- "9" und die Zeichen "#" und "*". Wird jetzt eines der Elemente mit einem Accesskey verbunden, kann durch Drücken der entsprechenden Taste direkt zu diesem Element gesprungen werden.

Ein komplett neues Element ist `pre`, welches anzeigt, dass der folgende Text vorformatiert ist und nicht verändert werden soll. Die Spezifikationen für das Attribut `accesskey` und für das Element `pre` sind allerdings sehr vage hinsichtlich der Darstellung im Microbrowser. In den meisten Fällen bleibt es den Herstellern der Microbrowser überlassen, wie die Funktionen angeboten werden

2.1.5 Was WML nicht kann

Die vorangegangenen Kapitel haben erläutert, welche Möglichkeiten WML bietet. Dieser Abschnitt zeigt kurz die wesentlichen Funktionen, die WML nicht abzudecken vermag.

WML bietet beispielsweise keine Möglichkeit, Benutzereingaben nach deren Richtigkeit zu überprüfen. Bis auf den einfachen Mustervergleich (siehe Beispiel 9) kann WML also nicht entscheiden, ob der Benutzer sinnvolle beziehungsweise korrekte Eingaben gemacht hat.

²⁶ Vgl. WAP FORUM (1999e) und RÖWEKAMP, Lars (2000), S.104

Sprachen des mobilen Internet – eine Analyse

WML bietet weiterhin keinen Zugriff auf Funktionen innerhalb des mobilen Endgeräts. WML kann zum Beispiel nicht in einem Mobiltelefon Anrufe initialisieren, Nachrichten senden, das Adressbuch aktualisieren oder Zugriff auf die SIM-Karte ermöglichen. Des Weiteren bietet WML auch nicht die Möglichkeit, im mobilen Endgerät Zusatzfunktionen zu implementieren oder die Geräteeinstellung zu ändern.²⁷ Daher sind auch keine Berechnungen mit WML möglich.²⁸ Schließlich kann WML keine Benachrichtigungen, Fehler- oder Bestätigungsmeldungen lokal generieren, um so einen zusätzlichen Serverzugriff zu vermeiden.

²⁷ Vgl. WAP FORUM (1999a), S.12

²⁸ Vgl. BAGER, Jo (1999), S.127

Sprachen des mobilen Internet – eine Analyse

2.2 WMLScript

Da WML nur sehr einfache interaktive Inhalte anbietet und verschiedene sinnvolle Funktionen gar nicht bereitstellt, wurde WML eine Skriptsprache an die Seite gestellt, die solche Operationen zulässt. WMLScript ist mit JavaScript aus dem HTML-Umfeld vergleichbar. Die Operationen werden dabei direkt auf dem mobilen Endgerät verarbeitet. Es muss also keine Übertragung der Daten zum Server stattfinden, der die Funktionen ausführt und das Ergebnis an das mobile Endgerät sendet. WMLScript wird wie WML als Bytecode übertragen (siehe 2.1.1). Diese Aufgabe erledigt ein WMLScript-Compiler, was dem Microbrowser eine direkte Ausführung des Codes ermöglicht.

WMLScript basiert auf ECMAScript, wurde allerdings für die Übertragung in schmalbandigen Netzen und die Verarbeitung auf mobilen Endgeräten optimiert. Viele erweiterte Funktionalitäten von ECMAScript wurden daher weggelassen, um die Sprache "leichter" zu machen, und um sie einfacher erlernen zu können.

ECMAScript wurde in der Version 1 im Juni 1997 von der europäischen Organisation ECMA als Standard verabschiedet, da die JavaScript-Versionen von Microsoft (JScript) und Netscape immer weiter von einander abzuweichen drohten. ECMAScript bildet nun die Basis für diese Skriptsprachen.

2.2.1 WMLScript 1.1

WMLScript in der Version 1.1 ist nahezu deckungsgleich mit dem sogenannten "core", also dem Kern, von ECMAScript. Der Kern enthält alle grundlegenden Typen, Variablen, Ausdrücke etc. Dieser Kern kann laut Spezifikation nahezu unverändert als WMLScript verwendet werden.²⁹ Da bis heute viele Microbrowser WMLScript noch nicht implementiert haben, wird hier nur auf grundlegende Funktionen und Möglichkeiten von WMLScript eingegangen.

Grundsätzlich wichtig ist, dass im Gegensatz zu JavaScript und HTML, der Programmcode nicht in der WML-Datei integriert sein darf. Es erfolgt immer ein Verweis in der WML-Datei auf eine externe WMLScript-Datei mit der Endung `.wmls`.³⁰

²⁹ Vgl. WAP FORUM (1999a), S.13

³⁰ Vgl. RÖWEKAMP, Lars (2000a), S.192

Sprachen des mobilen Internet – eine Analyse

Eine WMLScript-Datei besteht im einfachsten Fall aus einer Funktion, in der Regel aber aus einer Reihe von Funktionen und weiteren Angaben. Die WMLScript-Datei wird auch WMLScript Compilation Unit genannt.³¹ Die beiden Begriffe werden im weiteren synonym verwendet.

2.2.1.1 Kommentare

Kommentare erhöhen die Verständlichkeit des WMLScript-Code. WMLScript besitzt ein- und mehrzeilige Kommentare.

Beispiel 24: WMLScript – Kommentare

Beschreibung	WMLScript
Einzeiliger Kommentar	// Hier ist der Kommentar
Blockkommentar	/* Kommentar über mehr als eine Zeile */

³¹ Vgl. WAP FORUM (1999a), S.63

Sprachen des mobilen Internet – eine Analyse

2.2.1.2 Variablen

WMLScript hat Zugriff auf das Zustandsmanagement und somit auch auf die Variablen von WML. Damit ist es möglich, Variablen in WML sowohl auszulesen als auch zu setzen.³² Auch WMLScript ist eine case-sensitive Sprache und unterscheidet demnach Groß- und Kleinschreibung, was bei der Definition von Variablen zu beachten ist.

Beispiel 25: WMLScript - Definition von Variablen

Beschreibung	WMLScript
Variable mit dem Namen "a" wird definiert	<code>var a;</code>
Variable mit dem Namen "meineVariable" wird definiert	<code>var meineVariable;</code>
Variable mit dem Namen "b" wird definiert und der Wert ' "Horst" ' zugewiesen	<code>var b="Horst";</code>

Bei den ersten beiden Beispielen wird eine Variable definiert, ohne ihr zunächst einen Wert zu zuweisen. Das dritte Beispiel definiert eine Variable und weist ihr gleichzeitig einen Wert zu.

2.2.1.3 Datentypen

WMLScript ist eine sogenannte "Weakly Typed Language", was bedeutet, dass der Wert einer Variablen gleichzeitig deren Typ bestimmt. Der Typ einer Variablen sagt aus, ob es sich bei dem Wert beispielsweise um eine Ganzzahl, eine Kommazahl oder einen Text handelt. Im obigen Beispiel ist die Variable `b` mit dem Wert `"Horst"` definiert worden. Der Wert `"Horst"` sagt jetzt über den Typ der Variablen aus, dass es sich um eine Zeichenkette, also um Text, handelt. In anderen Programmiersprachen muss der Typ einer Variablen bei deren Definition mit angegeben werden. Bei WMLScript ist dies nicht möglich.³³

Datentypen sind notwendig, damit die verarbeitende Instanz weiß, um welche Inhalte es sich in den Variablen handelt und wie diese weiter verarbeitet werden können bzw. müssen. So

³² Vgl. WAP FORUM (1999b), S.20

³³ Vgl. RÖWEKAMP, Lars (2000a), S.193

Sprachen des mobilen Internet – eine Analyse

macht es etwa wenig Sinn, einen Text wie "Horst" mit einer Zahl zu multiplizieren. In WMLScript können Variablen fünf verschiedene Datentypen annehmen.³⁴

Beispiel 26: WMLScript – Datentypen

Beschreibung	WMLScript
Datentyp Boolean, Wert entweder "true" oder "false"	<code>var a = true;</code>
Datentyp Integer	<code>var b = 10;</code>
Datentyp Float	<code>var c = 10.5;</code>
Datentyp String	<code>var d = "Super";</code>
Datentyp Invalid	<code>var e = invalid;</code>

Der Datentyp Boolean beschreibt eine Variable, die zwei Werte besitzen kann. Diese Werte sind "true" und "false". Dieser Datentyp wird eingesetzt, wenn Bedingungen überprüft werden sollen. Das Ergebnis kann dann entweder wahr oder falsch sein.

Der Datentyp Integer weist auf eine Variable mit einer Ganzzahl als Wert hin, während Float eine Kommazahl definiert. Um einer Variable einen Text als Wert zu übergeben, muss dieser in Hochkommata eingeschlossen werden. Damit ist für WMLScript klar, dass es sich um eine Variable mit dem Datentyp String handelt.

Ist die Variable keinem der oben genannten Typen zugeordnet, nimmt sie den Wert "invalid" an. Damit lässt sich ein Fehlerbehandlungskonzept realisieren.³⁵

³⁴ Vgl. WAP FORUM (1999a), S.14

³⁵ Vgl. RÖWEKAMP, Lars (2000a), S.193

Sprachen des mobilen Internet – eine Analyse

2.2.1.4 Funktionen

"Funktionen sind Programmteile, die zur Lösung einer bestimmten Aufgabe vorgesehen sind. [...] Vereinfacht ausgedrückt, stellt eine Funktion eine Bündelung von mehreren Befehlen dar."³⁶

WMLScript unterscheidet zwei Arten von Funktionen, externe und interne Funktionen. Der Unterschied liegt allein in der Art des Zugriffs. Auf externe Funktionen kann eine Instanz auch zugreifen, wenn sie nicht in derselben WMLScript-Datei liegen. Interne Funktionen können nur von Programmteilen angesprochen werden, die in derselben WMLScript-Datei liegen.³⁷

Beispiel 27: WMLScript – Funktionen

Beschreibung	WMLScript
Externe Funktion	<pre>extern function meineFunk() { //Funktionsinhalt }</pre>
Interne Funktion	<pre>function intFunk() { //Funktionsinhalt }</pre>

In dem ersten Beispiel wird eine externe Funktion – durch das Schlüsselwort `extern` – mit dem Namen `meineFunk` definiert. Anstelle des Kommentars, stehen dann die von der Funktion auszuführenden Befehle. Durch Weglassen des Schlüsselwortes `extern` wird eine interne Funktion definiert.

Am häufigsten wird WMLScript im Zusammenhang mit WML angewendet werden. Daher ist es sinnvoll, aus WML auf Funktionen in WMLScript zugreifen zu können. Umgekehrt kann WMLScript Werte an WML zurückliefern, die für die Präsentation von Informationen genutzt werden können.

WMLScript übernimmt, wie WML, die Verweisarchitektur von HTML und dem World Wide Web. Die WMLScript-Dateien werden über URLs angesprochen. Werden spezielle Abschnitte einer Datei aufgerufen, bedient sich WMLScript auch der oben erläuterten Anker. Mit

³⁶ Vgl. KOCH, Stefan (1999), S.43

³⁷ Vgl. RÖWEKAMP, Lars (2000a), S.194

Sprachen des mobilen Internet – eine Analyse

Ankern kann man auf einzelne externe Funktionen einer WMLScript-Datei zugreifen. Dabei wird der URL eine Raute (#) und der Abschnittsbezeichner angehängt.³⁸

Beispiel 28: WMLScript – Funktionen aufrufen

Beschreibung	WMLScript
Externe Funktion	
URL der WMLScript-Datei:	<code>http://www.weisnet.de/scr.wmls</code>
Funktion & Übergabewerte:	<code>extern testFunc('Var1', 5)</code>
Aufruf der Funktion:	<code>http://.../scr.wmls#testFunc('Var1', 5)</code>
Interne Funktion	
Funktion & Übergabewerte:	<code>xyFunc(20, 12.5)</code>
Aufruf der Funktion	<code>xyFunc(20, 12.5);</code>

Bei externen Funktionen ist der Abschnittsbezeichner der Funktionsname und die Übergabewerte der Funktion. Übergabewerte sind Werte, welche die Funktion mit ihren Befehlen verarbeiten soll.³⁹ Ein Beispiel ist die Multiplikation von zwei übergebenen Zahlen.

Da die interne Funktion nur von anderen Funktionen innerhalb derselben WMLScript-Datei aufgerufen werden kann, entfällt die Angabe der URL. Der Aufruf einer internen Funktion findet mit Hilfe des Funktionsnamens, der Übergabewerte und eines abschließenden Semikolons statt.

2.2.1.5 Bibliotheken

Eine Programmiersprache gewinnt durch Bibliotheken an Funktionsumfang und an Effizienz. Bibliotheken enthalten vordefinierte Funktionen, die bei der Programmierung erfahrungsgemäß häufig wiederkehren. Diese Funktionen sind in Unterkategorien (Bibliotheken) geordnet, die ihr ungefähres Aufgabengebiet abdecken. "Die Standardbibliotheken ersparen den Ent-

³⁸ Vgl. NOKIA (1999a), S.8

³⁹ Vgl. KOCH, Stefan (1999), S.47

Sprachen des mobilen Internet – eine Analyse

wicklern in vielen Fällen die Neuerfindung des Rades, da sie bereits einen Großteil häufig benötigter Grundfunktionen [...] zur Verfügung stellen."⁴⁰

Die WMLScript Standard Bibliothek umfasst derzeit sechs Bibliotheken. Die Bibliotheken sind `Lang`, `Float`, `String`, `URL`, `WMLBrowsers` und `Dialogs`.⁴¹

Die `Lang`-Bibliothek enthält hauptsächlich Funktionen, die den Kern von WMLScript betreffen.⁴² Eine beispielhafte Funktion ist die Ermittlung einer Zufallszahl.

Beispiel 29: WMLScript – Lang-Bibliothek

Beschreibung	WMLScript
Aufruf der Randomfunktion mit dem Übergabewert 10	<code>Lang.random(10);</code>
Ergebnis	<code>0 < Zufallszahl <= 10</code>

Bei der Nutzung von Funktionen aus Bibliotheken wird zuerst der Name der Bibliothek genannt, gefolgt von einem Punkt und dem Namen der Funktion. Im obigen Beispiel wird die Funktion `random()` aus der Bibliothek `Lang` aufgerufen. Der übergebene Wert ist 10, was in diesem Fall bedeutet, dass eine Zufallszahl ermittelt wird, die größer 0 und kleiner oder gleich 10 ist.

Die Bibliothek `Float` enthält dagegen Funktionen für typische Floating-Point Operationen.⁴³ Da die Bibliothek eine Funktion enthält, die Komma- in Ganzzahlen umwandelt, kann man die Bibliothek auch mit Ganzzahlen verwenden.⁴⁴

Beispiel 30: WMLScript – Float-Bibliothek

Beschreibung	WMLScript
Aufruf der Funktion <code>floor()</code> mit Übergabewert 10.4	<code>Float.floor(10.4);</code>
Ergebnis:	10

Diese Funktion gibt als Rückgabewert die nächstkleinere Ganzzahl, relativ zum Übergabewert, zurück. In diesem Fall ist der Rückgabewert 10.

⁴⁰ Vgl. RÖWEKAMP, Lars (2000a), S.195

⁴¹ Vgl. RÖWEKAMP, Lars (2000a), S.195

⁴² Vgl. NOKIA (1999a), S.63

⁴³ Vgl. NOKIA (1999a), S.70

⁴⁴ Vgl. RÖWEKAMP, Lars (2000a), S.195

Sprachen des mobilen Internet – eine Analyse

Bei der `String`-Bibliothek handelt es sich um Funktionen, mit denen man Textinformationen manipulieren kann.⁴⁵ Der Text ist dabei eine Folge von Zeichen, wobei jedes Zeichen durch eine Indexzahl einzeln angesprochen werden kann. Die Indexzahl richtet sich nach der Position des Zeichens in der Zeichenkette.

Beispiel 31: WMLScript – String-Bibliothek

Beschreibung	WMLScript
Aufruf der Funktion <code>length()</code> mit Übergabewert "Hase"	<code>String.length("Hase");</code>
Ergebnis:	4

Die Funktion `length` der `String`-Bibliothek ermittelt die Länge der Zeichenkette, die als Übergabewert angegeben wurde. Die Länge entspricht der Anzahl der Zeichen einer Zeichenkette.

Funktionen, die für die Verarbeitung von URLs zuständig sind, stellt die Bibliothek `URL` zur Verfügung.⁴⁶ Diese Funktionen können beispielsweise prüfen, ob eine eingegebene URL der zugrunde liegenden Syntax entspricht.

Beispiel 32: WMLScript – URL-Bibliothek

Beschreibung	WMLScript
Aufruf der Funktion <code>getHost()</code> mit Übergabewert "http://www.budach.de/wml/"	<code>URL.getHost("http://www.budach.de/wml/");</code>
Ergebnis:	"www.budach.de"

Die beispielhafte Funktion `getHost()` extrahiert den Host aus einer eingegebenen URL.

Eine wichtige Bibliothek im Zusammenhang mit WML, stellt `WMLBrowser` dar. Diese Funktionen ermöglichen das Setzen und Auslesen von Variablen in WML, das Aufrufen von Cards, Decks und der Historyfunktion.⁴⁷

⁴⁵ Vgl. NOKIA (1999a), S.73

⁴⁶ Vgl. NOKIA (1999a), S.84

⁴⁷ Vgl. NOKIA (1999a), S.91

Sprachen des mobilen Internet – eine Analyse

Beispiel 33: WMLScript – WMLBrowser-Bibliothek

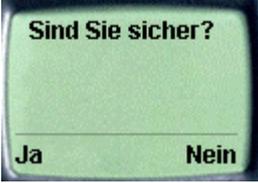
Beschreibung	WMLScript
Funktion <code>aus()</code> , die selbst Funktionen enthält	<pre>extern function aus(){ WMLBrowser.setVar("nach", "REH"); WMLBrowser.refresh(); }</pre>
 <p>Abbildung 38: WMLScript 1</p>  <p>Abbildung 39: WMLScript 2</p>	<pre><wml> <card id="card1" title="Title" newcontext="true"> <p> Vorname:<input name="vor" value="" title="Vorname" format="A*" /> Nachname:<input name="nach" value="" title="Nachname" format="A*" /> </p> <do type="accept" label="Neuer Nachname"> <go href=" ../wmls/confirm.wmls#aus()" /> </do> </card> </wml></pre>

Das Beispiel 33 zeigt die Verwendung der Funktionen `setVar()` und `refresh()` der Bibliothek `WMLBrowser`. Das WML-Beispiel enthält eine Card mit zwei Eingabefeldern. Nach Ausfüllen der Felder hat man die Möglichkeit über `Options` den Funktionsaufruf auszulösen (`<do ...> ...</do>`). Die Funktion `setVar()` setzt die Variable "nach" mit dem Wert "PAB" (Abbildung 38) auf den Wert "REH" (Abbildung 39). Die Funktion `refresh()` erneuert das Abbild im Microbrowser und ist vergleichbar mit der Aktion "Reload" oder "Aktualisieren" eines HTML-Browsers.

Eine weitere direkte Schnittstelle User Interface, stellen die Funktionen der Bibliothek `Dialogs` dar, mit deren Hilfe man Hinweise, Meldungen, Bestätigungs- und Eingabeabfragen abfassen kann.⁴⁸

⁴⁸ Vgl. NOKIA (1999a), S.94

Beispiel 34: WMLScript – Dialogs-Bibliothek

Nokia WAP Toolkit 1.2	WMLScript
 <p>Abbildung 40: Confirm</p>	<pre>extern function hinweis() { Dialogs.confirm("Sind Sie sicher?","Ja", "Nein"); }</pre>

Mit der Funktion `confirm()` kann eine Bestätigungsabfrage erstellt werden. Der erste Parameter ist der Bestätigungstext, der zweite enthält die positive, der dritte die negative Antwort. Dadurch lassen sich Reaktionen auf Benutzereingaben oder -aktionen formulieren.

3 Informationsarten für mobile Endgeräte

Die nächsten Kapitel beschreiben, welche Informationen oder Anwendungsszenarien sich grundsätzlich für die Darstellung auf mobilen Endgeräten eignen, und wie sich diese einordnen lassen. Dabei werden zunächst allgemeine Anwendungsbereiche aufgezeigt, um danach existierende WAP-Anwendungen zu analysieren und darzustellen. Auf dieser Basis werden Klassifikationsmerkmale entwickelt, durch die es möglich ist, Anwendungen und Informationseinheiten einzuordnen. Die Klassifikationsmerkmale und das Klassifikationssystem dienen als Grundlage für die spätere Entwicklung von Lösungskonzepten für relevante WAP-Bereiche.

3.1 Anwendungsbereiche

3.1.1 Allgemein

Grundsätzlich lässt sich wohl sagen, dass nahezu jede Information in irgendeiner Form auch für mobile Endgeräte verfügbar sein könnte. Ausschlaggebend ist das Verhältnis zwischen Aufwand und Nutzen. Entscheidend dabei ist die Berücksichtigung aller Beschränkungen und Besonderheiten des Mobilfunkbereiches, die in Kapitel 4.1 näher erläutert werden. Da die Mobilfunkumgebung nicht dafür geeignet ist, das im WWW übliche "Surfen" abzubilden,

Informationsarten für mobile Endgeräte

werden die Anwendungen vielmehr auf die gezielte Übermittlung von zeitabhängigen und diskreten Informationseinheiten ausgerichtet.⁴⁹ Dabei spielen vor allem die sogenannten Transaktionen eine entscheidende Rolle.

Nokia hat in Finnland seine Handykunden befragt, welche mobilen Internet-Dienste sie nutzen würden. Das Ergebnis ist in Abbildung 41 abzulesen.

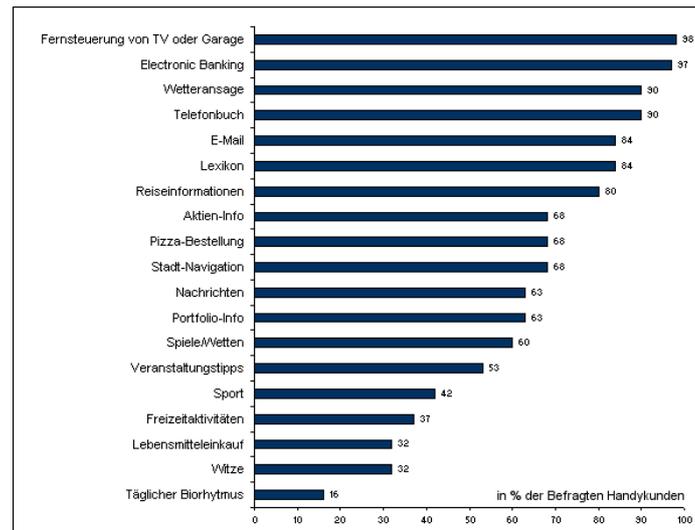


Abbildung 41: Befragung von Handykunden zu mobilen Infodiensten (Quelle: Nokia)

Besonders geeignet für den Mobilfunkbereich erweisen sich Anwendungen, die in möglichst kleine Einheiten aufgegliedert, übertragen und empfangen werden können. Eine gute Voraussetzung bieten Anwendungen, die einfach strukturiert sind, wenig Eingaben des Nutzers erfordern und bei denen die Ausgabe der Informationen textuell erfolgen kann.

Das Potenzial von WAP beschränkt sich allerdings nicht allein auf die Portierung von existierenden Anwendungen aus dem WWW. Durch Zugriffsmöglichkeiten auf Funktionen des mobilen Endgeräts, ist es möglich, neue Kategorien von Diensten anzubieten. Da die Implementierung aber bisher nicht vollständig erfolgte, wird es noch einige Zeit dauern, bis solche Mehrwertdienste verfügbar sind.⁵⁰ Ferner werden die Stichworte "Personalisierung" und "Regionalisierung" im zukünftigen Mobilfunkmarkt eine wichtige Rolle spielen.⁵¹

⁴⁹ Vgl. SIETMANN, R. (1998), S.205

⁵⁰ Vgl. AU SYSTEM RADIO (1999), S.21

⁵¹ Vgl. FLAHERTY, Natasha (2000)

3.1.2 Konkret

Die genaue Anzahl verfügbarer WAP-Sites ist schwer zu ermitteln. Ähnlich den veröffentlichten Zahlen von Websites, schwanken diese je nach Herausgeber der Zahlen. Viag Interkom meldete beispielsweise im Dezember 1999, dass bereits über 3000 WAP-Sites verfügbar seien.⁵² Die norwegische Suchmaschine "Fast" hatte nach eigenen Angaben im Februar diesen Jahres 20.000 WML-Dokumente in ihrem WAP-Bestand (<http://wap.fast.no>) und der Betreiber der Site "m-find" (<http://m-find.com/>) hatte Ende März 2000 nach seinen Angaben etwa 1100 funktionierende WAP-Sites mit über 100.000 WML-Dokumenten im Bestand.⁵³ Das steigende Interesse an WAP lässt sich auch an der Anzahl der Beiträge in Foren zu diesem Thema ablesen. So stieg die Anzahl der Beiträge im Diskussionsforum "WML and WMLScript Programmers" des Anbieters EGroups (<http://www.egroups.com>) von 287 im Monat Dezember 1999 auf 1348 im Monat März 2000.⁵⁴

Im Internet sind inzwischen einige Suchmaschinen und Kataloge für WAP-Sites verfügbar. So hat Yahoo! beispielsweise einen eigenen Bereich für die Suche nach WAP-Sites eingerichtet (<http://de.mobile.yahoo.com/>). Weitere Anlaufpunkte für die Suche nach vorhandenen WAP-Angeboten sind die Seiten von WAPJAG.COM AG (<http://www.wapjag.de/>) und Infoseek (<http://www.infoseek.de/katalog/wapkatalog/>). Allerdings findet man bei keinem dieser Dienste eine Auskunft über die Anzahl der eingetragenen Seiten.

Bei der Suche nach WAP-Anwendungen merkt man schnell, dass WAP sich noch mitten in der Entwicklungsphase befindet. Viele Anwendungen lassen sich gar nicht aufrufen, sind unvollständig implementiert oder liegen nur in einer Demoversion vor. Eine grobe Einteilung der WAP-Anwendungen kann wie folgt aussehen.

3.1.2.1 Informationsdienste

Ein großer Teil der verfügbaren Anwendungen ist auf die Verteilung von Informationen ausgerichtet. Dazu gehören Nachrichten, wie Politik, Wirtschaft, Sport, Lottozahlen, Horoskop, Wetter, Unterhaltung etc. Informationen also, die von Nachrichtensendern oder Zeitungsverlagen produziert werden.

⁵² Vgl. "WAP-Portal von Viag Interkom", Heise Online News, 15.12.1999, <http://www.heise.de/newsticker/data/dz-13.12.99-001/> (31.03.2000)

⁵³ Coombes, Philip: "Re: How many wml-sites available?", 29.03.2000, <http://www.egroups.com/group/wmlprogramming/3244.html?> (31.03.2000)

Informationsarten für mobile Endgeräte

Konkret sind das u.a. folgende Dienste:

Tabelle 3: Anwendungen im Bereich Informationsverteilung

Beschreibung	URL
Tagesschau (ARD)	http://wap.tagesschau.de
ZDF	http://zdf.msnbc.de/wap/
n-tv	http://wap.n-tv.de
NDR	http://www.ndr.de/wap/
BBC News	http://www.bbc.co.uk/mobile/mainmenu.wml
RP Online	http://rp-online.de/wap/
Süddeutsche Zeitung	http://wap.sueddeutsche.de
Handelsblatt	http://wap.handelsblatt.com

Die oben stehende Liste erhebt keinesfalls den Anspruch auf Vollständigkeit. Sie soll lediglich das Einordnen der oben genannten Informationsangebote erleichtern.

3.1.2.2 Auskunftssysteme

Die Auskunftssysteme zeichnen sich insbesondere dadurch aus, dass sie auf Benutzeranfragen reagieren und möglichst kleine und zutreffende Antwortmengen zurückliefern. Die folgende Tabelle zeigt die unterschiedlichen Auskunftssysteme mit einem konkreten Beispiel.

Tabelle 4: Anwendungen im Bereich Auskunftssysteme

Beschreibung	URL
Jobbörse	http://wap.jobpilot.de
Staumeldungen	http://wap.adac.de/
Börsenkurse	http://wap.comdirect.de/
Fahrplanauskunft	http://wap.hafas.de/
Preisinformationen	http://212.223.46.80/wap/index2.wml
Wörterbücher	http://dict.leo.org/wap
Reiseführer	http://wap.2pl.com
Routenplaner	http://wap.passo.de/cgi-bin/index.pl?portal=7277
Telefonauskunft	http://www.waptel.de/
Restaurantführer	http://wap.hamburg-highlights.de/
TV-Sendetermine	http://gaudi.dyndns.org/wap/index.wml
Kinoprogramm	http://wap.cinemaxx.de/
Gelbe Seiten	http://wap.yellowmap.de/ym/suche.miml
Flugauskunft	http://195.232.84.55/flight/flight.wml

⁵⁴ Vgl. Beschreibung der Group "WML and WMLScript Programmers" bei EGroups, <http://www.egroups.com/group/wmlprogramming/info.html> (01.04.2000)

Informationsarten für mobile Endgeräte

Notenabfrage	http://www-winfo.ku-eichstaett.de/wap/wap.wml
--------------	---

Der Hauptvorteil solcher Anwendungen ist, dass die eigentliche Rechenarbeit vom Server erledigt wird. Zum einen muss damit weniger Datenverkehr über die Mobilfunknetze fließen und zum anderen erfordert dies weniger Rechenleistung des mobilen Endgerätes.

3.1.2.3 Interaktive Dienste

Eine weitere Gruppe von WAP-Anwendungen bilden die interaktiven Dienste. Sie zeichnen sich dadurch aus, dass eine mehrmalige Interaktion zwischen Server und Client erfolgt und dass unter Umständen Daten auf dem Server durch den Client verändert werden. Weiterhin erfolgt in den meisten Fällen eine Anmeldung des Benutzers. Bei dieser Gruppe von Anwendungen stößt man noch am häufigsten auf Prototypen oder Konzepte, die noch nicht komplett umgesetzt wurden.

Die Tabelle 5 zeigt eine Auswahl interaktiver WAP-Dienste, die funktionsfähig sind oder wenigstens als Demo vorliegen.

Tabelle 5: Anwendungen im Bereich interaktive Dienste

Beschreibung	URL
Auktionen	http://wap.ricardo.de/
E-Mailverwaltung	http://wap.web.de/
Brokerage	http://wap.consors.de/
Spiele	http://webcab.de/wwe/Entertainment/Games/2.wml
Chat	http://wap.tro.de/chat/main.wml
Voting	http://www.mtvhome.de/wap/webchart/
Autovermietung	http://wap.sixt.de/
Kartenreservierung	http://wap.cinemaxx.de/
Kontaktbörse	http://www.wapme.net/
Bookmarks	http://webcab.de/be.wml
Shopping	http://wap.s24.de/
Klausurenanmeldung	http://www-winfo.ku-eichstaett.de/wap/wap.wml

Weitere interaktive Dienste sind bereits angekündigt. Unter denen, die in den nächsten Wochen und Monaten gestartet werden sollen, befinden sich beispielsweise Kontoverwaltung⁵⁵,

⁵⁵ Deutsche Bank, Nokia, <http://public.deutsche-bank.de/deuba/db/aktuell.nsf/doc/TDUR-4FRKAF?OpenDocument&NavID=AKLZ-462Q5X> (29.05.2000)

Informationsarten für mobile Endgeräte

Flugbuchungen, Flug-Check-in,⁵⁶ Mobiles Büro⁵⁷ und Logistikanwendungen wie die Aktualisierung von Lagerbeständen, Sendungsverfolgung etc.⁵⁸

3.1.2.4 Portale

Ähnlich der Entwicklung im WWW haben sich auch schon im WAP-Umfeld sogenannte Portale gebildet, welche die verschiedensten WAP-Anwendungen unter einer Oberfläche zugänglich machen. Die Portale sind entweder vertikale Portale, die ein möglichst breites Spektrum abdecken oder horizontale Portale, die sich auf ein spezielles Themengebiet beschränken und somit ein tieferes Angebot anbieten.

Da die Eingabe von URLs über ein mobiles Endgerät sehr umständlich sein kann, besitzen Portale im mobilen Bereich einen weiteren Vorteil. Durch das Portal erhält der Nutzer einen schnellen Zugriff auf viele weitere WAP-Sites, ohne neue URLs eingeben zu müssen. Schon aus diesem Grund wird eine starke Konzentration auf Portale im mobilen Bereich stattfinden.

Einen echten Mehrwert für den Nutzer bieten Portale, die das Anlegen von Profilen ermöglichen. Damit lassen sich benutzerdefinierte Ansichten auf die gewünschten Informationen erstellen und können gewährleisten, dass nur die gerade benötigten Informationen auf dem Display erscheinen. Die Konzepte der Profile und der "lernenden" Anwendungen, die aus dem Nutzerverhalten eigenständig eine Art Profil herstellen, sind eine große Chance für die Verbreitung von WAP-Anwendungen.

Alle Mobilfunknetzbetreiber in Deutschland haben inzwischen WAP-Portale entwickelt oder sind mit der Entwicklung beschäftigt. Da die Mobilfunknetzbetreiber in den meisten Fällen die Schnittstelle vom Mobilfunknetz zum Internet darstellen, haben sie einen beachtlichen Vorteil gegenüber Anbietern, die nur Inhalte anbieten können und über das Internet gefunden werden müssen.⁵⁹ Die Portale der Mobilfunkanbieter sind nur über das jeweilige WAP-Gateway des Netzbetreibers erreichbar.

Die Tabelle 6 zeigt die Mobilfunknetzbetreiber mit ihren Kooperationspartnern, die am Aufbau der Portale beteiligt sind (Stand: März 2000).

⁵⁶ Swissair, News http://www.swissair.com/about/media/ge161299_2.htm (03.04.2000)

⁵⁷ Software AG, <http://www.softwareag.com/germany/products/wap.htm> (03.04.2000)

⁵⁸ Mobile Logistics mit [logisticsofferandorder.net](http://www.loon.de) (LOON), <http://www.loon.de> (03.04.2000)

⁵⁹ Vgl. BUDACH, Jens (2000), S.19-20

Informationsarten für mobile Endgeräte

Tabelle 6: Mobilfunknetzbetreiber und ihre Portal-Kooperationspartner ⁶⁰

Mobilfunkanbieter	Kooperationspartner
T-Mobil	ZDF-heute, Focus online, Wirtschaftswoche, DM, Handelsblatt, Financial-Times-Europe, Kino News, Eurosport, SAT1-ran, TV Movie, MTV Events, Ravensburger Spiele, Lufthansa, Tegaron Verkehrsinfo, Gelbe Seiten, Lottofon, Stella Musical, eBay-Autkionen
D2privat	Yahoo!, NDR, Tagesschau, n-tv-online, SAT1, Pro7, Handelsblatt, Consors, Daily4u, ADAC, Daland Hotelkatalog, PASSO, Topware, Startrek, eBay-Auktionen
E-Plus	Spiegel online, WDR, Pro7, Reuters, Handelsblatt, GNN Telekommunikations-News, Dr. Materna, Kachelmann, AND Routenplaner, Lycos, Tomorrow, Travelchannel, Prinz, Eins Life
Viag Interkom	Pro7, ZDF-heute, Dr. Materna, Süddeutsche Zeitung, ADAC, Sixt, Hörzu, Wetter online, rp-online

Neben den Portalen der Mobilfunkanbieter gibt es Portale, die frei über das Internet zugänglich sind. Dabei handelt es sich bei den meisten Portalen um reine Linksammlungen zu weiteren Diensten. Comuserve hat dagegen eigene Anwendungen entwickelt und versucht damit eine Art mobiles Büro darzustellen. In die gleiche Richtung gehen die Konzepte von Yahoo! und Web.de.

⁶⁰ Vgl. ECKSTEIN, Pia (2000), S.28

Informationsarten für mobile Endgeräte

Tabelle 7: WAP-Portale

Beschreibung	URL
Compuserve	http://wap.compuserve.de/index.wml
Yahoo!	http://wap.yahoo.de
Web.de	http://wap.web.de/
T-Online	http://wap.t-online.de
Debitel	http://wap.debitel.net
WAPJump	http://www.wapjump.de/index.wml
Gixom WAP Katalog	http://gixom.de/i.wml
WAPJAG	http://webcab.de/be.wml http://www.wapjag.de/
Webcab	http://webcab.de/wwe/i.wml

3.2 Entwicklung von Klassifikationsmerkmalen

In den nächsten Kapiteln werden Klassifikationsmerkmale entwickelt, welche die Grundlage für die später zu entwickelnden Lösungskonzepte bilden. Die Klassifikationsmerkmale sollen dabei das Einordnen von WAP-fähigen Informationseinheiten ermöglichen.

Zunächst werden die Informationen, die auf dem mobilen Endgerät dargestellt werden können, analysiert und in ihre einzelnen Bestandteile zerlegt. Auf der Basis der Analyse lassen sich dann die ermittelten Informationsbausteine verwenden, um das gewünschte Informationsszenario abzubilden. Im weiteren Verlauf werden die erarbeiteten Klassifikationsmerkmale an Beispielen angewandt. Die Beispiele sollen in ihrer Gesamtheit ein möglichst großes Spektrum an Kombinationsmöglichkeiten der Klassifikationsmerkmale abdecken. Ziel ist es, alle relevanten Bereiche und Kombinationen der Informationsbestandteile benennen zu können und daraus abzuleiten, welche Kombinationen für derzeitige WAP-Anwendungen besonders relevant sind und welche bis dato vernachlässigbar sind.

Als Beispiele werden dabei nicht komplette Anwendungen, sondern relevante Teilbereiche betrachtet. Diese Teilbereiche sollten in WML innerhalb weniger Decks abgebildet werden können. Zum Vergleich stelle man sich beispielsweise das Verhältnis des Inhalts einer HTML-Seite zur gesamten Site vor.

Für die einzelnen Kombinationsmöglichkeiten der Klassifikationsmerkmale werden dann Lösungskonzepte erarbeitet, wie die Informationen sinnvoll in WML dargestellt werden können.

Informationsarten für mobile Endgeräte

Da es sich um Teilbereiche aus WAP-Anwendungen handelt, steht nicht das Application Design, sondern das Information Design im Vordergrund.

3.2.1 Klassifikationsmerkmal Medialer Typ

Die derzeitige Version von WAP lässt grundsätzlich zwei mediale Informationstypen zu, Text und Grafik. Da abzusehen ist, dass in Zukunft auch weitere mediale Typen Einzug in WAP halten werden, sind hier zusätzlich die Typen Audio und Video abgebildet.

Die erste Ebene der Klassifikation wird demnach durch die medialen Typen Text, Grafik, Video und Sound gebildet. Die einzelnen Kategorien werden ihrerseits durch verschiedene Merkmale näher bestimmt. Die Abbildung 42 zeigt die Übersicht der ersten Ebene.

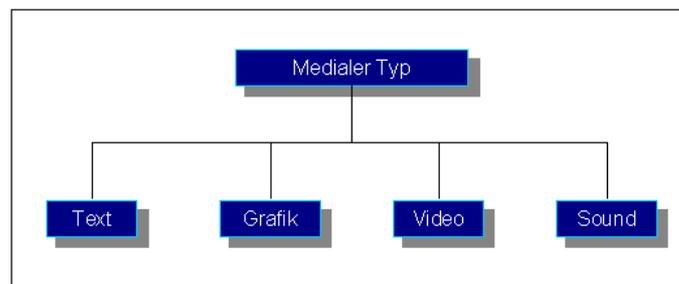


Abbildung 42: Klassifikation des medialen Typs

3.2.1.1 Medialer Typ Text

Ein Beschreibungsmerkmal des medialen Typs *Text* ist seine Länge. Bei der Textlänge handelt es sich um die Anzahl der Zeichen ohne Leerzeichen. Wichtig in diesem Zusammenhang ist das Verhältnis von Textmenge zur Menge der eingesetzten Grafiken. Ein weiteres Beschreibungsmerkmal, das Merkmal Sprache, wird in diesem Teil nicht betrachtet. Da die Sprache etwas über möglicherweise verwendete Sonderzeichen, wie beispielsweise Umlaute, aussagt, findet die Sprache bei der späteren Entwicklung der Lösungskonzepte Beachtung.

3.2.1.2 Medialer Typ Grafik

Wie schon in Kapitel 2.1.3.4 gezeigt, bietet WML die Möglichkeit, mit einfachen Grafiken zu arbeiten. Als Merkmal wird in diesem Zusammenhang die Menge der eingesetzten Grafik im Verhältnis zur Textmenge analysiert und bewertet. Bei der Entwicklung der Lösungskonzepte werden dann weitere Merkmale, wie Inhalt, Art und Größe der Grafiken hinzugezogen.

Informationsarten für mobile Endgeräte

3.2.1.3 Mediale Typen Sound und Video

Da die Integration von Sound und Video bisher in WAP noch nicht berücksichtigt wurde, werden die beiden Komponenten hier gemeinsam und nur am Rande behandelt.

Bei der Beschreibung von Sound und Video können Merkmale, wie die inhaltliche Beschreibung der Sound- und Videodaten und die jeweilige Länge verwendet werden. Ein Video kann zusätzlich nach der Anzahl der Bilder pro Sekunde und nach der Auflösung, in der es entstanden ist, kategorisiert werden. Bei Tondokumenten kann die Qualität als Merkmal angeführt werden.

3.2.1.4 Multimedia

Anhand der oben entwickelten Klassifikation, ließe sich jetzt auch sehr einfach multimedialer Inhalt als Informationsbaustein aufnehmen. Multimedia ist – wie der Name schon sagt – die Kombination verschiedener Medientypen. Im heutigen Sprachgebrauch wird mit Multimedia aber auch Interaktion bzw. Interaktivität verbunden. Bisher kann man mit WAP zwar Interaktivität aber noch keine Multimedialität erreichen. Die ersten Versuche dem abzuwehren, gibt es bereits. So hat beispielsweise Macromedia auf der CeBIT 2000 angekündigt, Macromedia Flash bald für mobile Endgeräte verfügbar zu machen.⁶¹

⁶¹ Vgl. "Flash wird mobil : Macromedia Flash für Symbian-Plattform", Golem Network News, 29.02.2000, <http://www.gnn.de/0002/6550.html> (10.04.2000)

3.2.2 Klassifikationsmerkmal Struktur

Bei der Entwicklung des Klassifikationsmerkmals *Struktur* wurden die einzelnen medialen Typen auf ihre möglichen Strukturierungen hin analysiert, um passende Merkmale daraus abzuleiten.

Im Zusammenhang mit der Analyse von WAP-fähigen Informationen ist eine Einteilung nach dem Strukturierungsgrad der Informationen sinnvoll. Dabei entsteht eine Skala von unstrukturierter bis hin zu stark strukturierter Information.

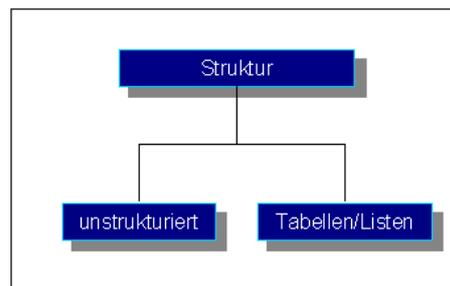


Abbildung 43: Klassifikation der Struktur

Ein einfacher Nachrichtentext kann ein Beispiel für unstrukturierte Information darstellen. Bei strukturierter Information handelt es sich um Tabellen und Listen. Dabei steigt der Grad der Strukturierung mit der Menge und Größe der Tabellen und Listen, die eingesetzt werden. Eine Unterscheidung von ungeordneten und geordneten Listen findet hier nicht statt. Stark strukturierte Tabellen sind beispielsweise Faktentabellen. Weniger stark strukturierte Tabellen sind Tabellen, die reine Layoutfunktionen übernehmen.

Eine feinere Untergliederung der strukturierten Information mit Tabellen und Listen mit ihren Merkmalen findet bei der Ausarbeitung der Lösungskonzepte statt.

Informationsarten für mobile Endgeräte

3.2.3 Klassifikationsmerkmal User Interaktion

Das Merkmal *User Interaktion* beschreibt die Menge der Benutzereingaben einer Anwendung. Benutzereingaben treten auf, wenn etwa Formulare oder Hyperlinks in der Anwendung vorkommen. Die stärkste Form der User Interaktion bilden große Formulare mit Eingabefeldern. Als weitere Möglichkeiten treten Formulare mit Options- oder Auswahlfeldern auf. Die zweite Form der User Interaktion sind Hyperlinks. Die Intensität der User Interaktion bei Hyperlinks ist generell niedriger als bei Formularen, da die Interaktion nur in der Aktivierung des entsprechenden Links besteht.

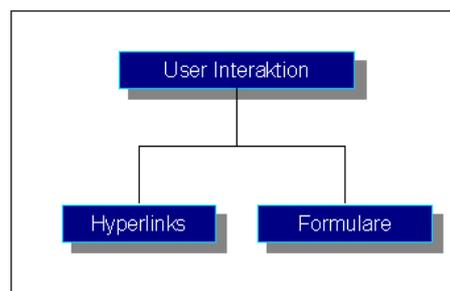


Abbildung 44: Klassifikation der User Interaktion

Die eingesetzte Menge an Hyperlinks bzw. Formularen beeinflusst demnach die Intensität der User Interaktion. Für die Entwicklung von WAP-Anwendungen ist die Kenntnis der Intensität der User Interaktion relevant, da besonders im mobilen Bereich der Weg zur Zielerreichung möglichst kurz sein muss und daher unnötige Eingaben oder Zwischenschritte zu vermeiden oder zu optimieren sind.

3.3 Anwendung der Klassifikation

Die oben erarbeitete Klassifikation wird nun auf WAP-fähige Informationseinheiten angewendet. Die Informationseinheiten stellen dabei Teilbereiche möglicher Anwendungen dar, die in einem oder wenigen Decks abgebildet werden können. Bei den Beispielen wurde darauf geachtet, dass möglichst viele Merkmalskombinationen abgedeckt werden.

Informationsarten für mobile Endgeräte

Bei der Anwendung der Klassifikation entstehen dreidimensionale Räume, deren Achsen jeweils ein Klassifikationsmerkmal repräsentieren. Abbildung 45 zeigt ein Beispiel für einen solchen Klassifikationsraum.

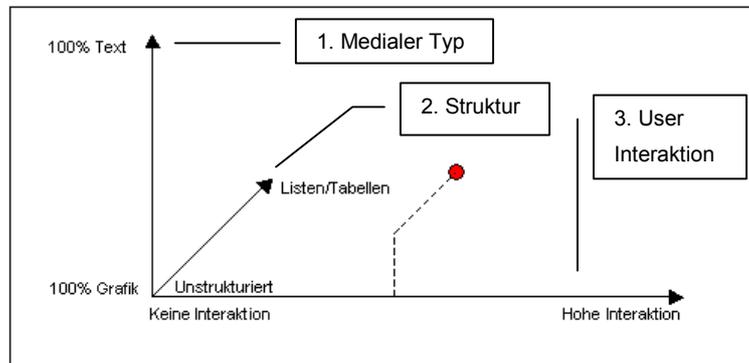


Abbildung 45: Beispiel 3D-Raum

Auf der ersten Achse befindet sich das Klassifikationsmerkmal medialer Typ. Die Skalierung entspricht dem relativen Verhältnis von Textmenge zu Grafikmenge. Enthält ein Beispiel keine Grafiken, erhält es die Zuordnung 100% Text. Da hiermit aber keine Aussage über das Verhältnis von beispielsweise viel Text zu wenig Text möglich ist, wird quasi als vierte Dimension die Größe des abgebildeten Punktes herangezogen. Die Größe des Punktes stellt die Menge des jeweilig überwiegenden medialen Typs dar. Die prozentuale Angabe beruht auf Schätzungen und kann daher nicht mathematisch berechnet werden.

Über die zweite Achse wird der Strukturierungsgrad der Information repräsentiert. Dabei entspricht die eine Seite der Achse unstrukturierter und die andere Seite stark strukturierter Information. Auf der Achse selbst können Abstufungen des Strukturierungsgrads bestehen.

Die Intensität der User Interaktion wird durch die dritte Achse abgebildet. Die Skalierung reicht von "keine Interaktion" bis "hohe Interaktion". Eine genaue Erläuterung, welche Bestandteile die User Interaktion ausmachen, wird in den jeweiligen Beispielen gegeben.

An dieser Stelle sei angemerkt, dass für eine absolut korrekte Darstellung, die Verwendung von je einer Dimension für jeden medialen Typen, jede Strukturierungsmöglichkeit und jede User Interaktion notwendig wäre. Aus Gründen der Darstellbarkeit wurde darauf in diesem Fall verzichtet.

Informationsarten für mobile Endgeräte

3.3.1 Beispiel Überweisung

The image shows a mobile banking interface for a transfer order. At the top, it displays 'Überweisungsauftrag an 508 525 53' and 'Kreissparkasse Groß-Gerau 4 3 2 4 6 2' with a red logo. The form is divided into several sections with labels and input fields:

- Empfänger:** Name, Vorname/Firma (max. 27 Stellen)
- Konto-Nr. des Empfängers:** Bankleitzahl
- bei (Kreditinstitut):** DM od. EUR*
- Beitrag:** Amount field
- Kunden-Referenznummer:** nach Verwendungszweck (max. 27 Stellen)
- nach Verwendungszweck:** (max. 27 Stellen)
- Kontoführer:** Name/Vorname, Firma, Ort (max. 37 Stellen, keine Straßen- oder Postfachangaben)
- Konto-Nr. des Kontoführers:** 20

At the bottom, there are fields for 'Datum' and 'Unterschrift'. On the right side, there is a vertical text: 'Bitte ist für Ihre Sicherheit zu beachten' and 'Bitte nicht vergessen, Ihre Karte zu deaktivieren, wenn Sie diese nicht mehr benötigen'.

Abbildung 46: Beispiel Überweisung

Das Dokument des Überweisungsauftrags eines Kreditinstituts besteht in erster Linie aus einem großen Formular mit unterschiedlichen Eingabefeldern. Demnach ist die User Interaktion sehr hoch, da viele Felder ausgefüllt werden müssen. Bei der Umsetzung auf WAP finden sich dann zusätzlich einfache Hyperlinks innerhalb des Deck wieder.

Als mediale Typen finden sich bei diesem Beispiel sowohl Text als auch Grafik, wobei der Text deutlich überwiegt. Die Textmenge an sich ist aber gering, weswegen der Punkt im Klassifikationsraum sehr klein dargestellt wird. Inhaltlich liefert der Text die Erläuterungen zu den einzelnen Eingabefeldern.

Über den Grad der Strukturierung der Überweisung lässt sich diskutieren, da keine eindeutige Aussage über die Art der Strukturierung gemacht werden kann. Entweder werden hier unsichtbare Tabellen verwendet, um die einzelnen Formularfelder auszurichten oder man betrachtet das Formular als Ganzes und ordnet es als unstrukturiert ein. Hier soll gelten, dass das Formular mit Hilfe von Tabellen strukturiert ist. Da es sich um Tabellen handelt, die Layout-Funktionen wahrnehmen, ist der Grad der Strukturierung nicht sehr stark dargestellt.

Informationsarten für mobile Endgeräte

Die Visualisierung der Analyse des Beispiels Überweisung kann wie folgt aussehen.

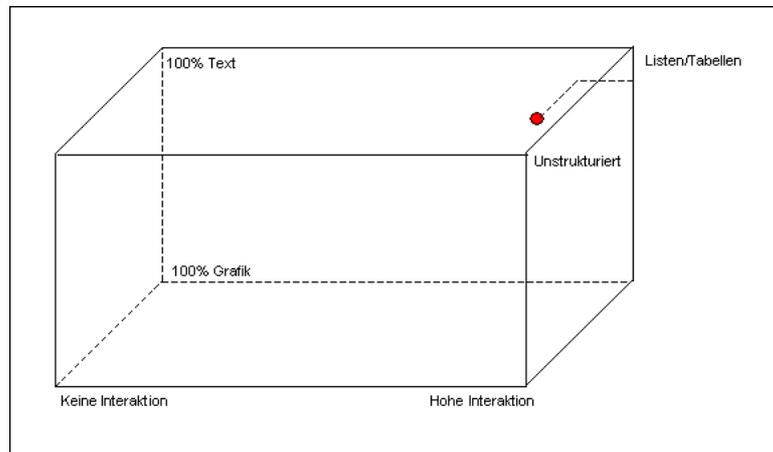


Abbildung 47: Überweisung 3D-Raum

Wie bereits oben beschrieben, lässt sich das Dokument oder die Informationseinheit mit den Merkmalen klassifizieren und visualisieren.

3.3.2 Beispiel Nachrichten

Ein Hauptanwendungsgebiet für WAP ist die Bereitstellung von Nachrichten jeglicher Art. Der Artikel einer Zeitung oder Zeitschrift besteht im einfachsten Fall aus reinem Text. Zusätzlich kann er aber auch strukturierte Informationen, wie Listen oder Tabellen enthalten, sowie mit Bildern bzw. Grafiken versehen sein.

Merkel zeigt sich angriffslustig
Neue CDU-Chefin kündigt "Klartext" gegenüber Rot-Grün an / Lob für Kohl
Von Knut Pries

Die neue CDU-Vorsitzende Angela Merkel hat der rot-grünen Bundesregierung den politischen Kampf angesagt - aus einer Position neuer Stärke, die sie beschwor. "Die Stunde unserer Gegner ist vorbei ... wir sind wieder da", sagte sie. Zuvor hatte Merkels Vorgänger Wolfgang Schäuble in seiner Abschiedsrede die Delegierten des Parteitag in Essen aufgefordert, an die Erfolge der Vergangenheit anzuknüpfen.

ESSEN, 10. April Angela Merkel betonte am Montag vor ihrer Wahl zur neuen Parteivorsitzenden, die CDU werde ihre Kräfte als Chance nutzen, weil sie "wieder genügend Raum gewonnen" habe für die politische Auseinandersetzung. "Es geht jetzt wieder zur Sache - nicht im Sinne von Schwarm drüber, sondern im Sinne von „Es wird wieder Klartext gesprochen.“ Ein verschönerndes Wort fand die bisherige Generalsekretärin für den ehemaligen Parteichef und Bundeskanzler Helmut Kohl, der erstmals seit fast 50 Jahren nicht an einem Parteitag der CDU teilnahm. Es gebe keine Alternative zur Aufklärung der Verstöße gegen das Parteigesetz, sagte Merkel. Dennoch gehe für die Bilanz der Leistungen des langjährigen Parteichefs: "Kohls Werk, Ihr Werk, lieber Helmut Kohl, bleibt historisch überragend." Merkels Rede folgte minutenlangem starker Applaus. Schäuble verabschiedete den knapp 1000 Delegierten: "Wir sind noch nicht überm Berg ... aber wir sind auf dem richtigen Weg."

Die vergangenen anderthalb Jahre hätten bewiesen: "Niederlagen werfen die Union nicht um, und Krisen auch nicht." Von der Europapolitik über das Gesundheits- und Bildungswesen bis zur Reform des Sozialstaates und dem Aufbau einer neuen sozialen Marktwirtschaft gebe es "Aufgaben in Hülle und Fülle - die Umsetzung gebietet".

Seine heftige Kritik an Machenschaften und Intrigen im eigenen Lager, die in der CDU Urmut ausgelöst hatte, ließ Schäuble in seiner Rede nur indirekt, in einem Appell zur Loyalität, anklingen: "Der Gegner heißt Abhängigkeit, Selbstverleugung, Druck oder Intrige." Der neuen Parteiführung wünsche er "am liebsten noch mehr" Vertrauen und Unterstützung, als er selbst sie erfahren habe. Dann wolle auch er weiter beitragen.

Die Delegierten, die Schäubles programmatische Ausführungen mit nur mäßigem Interesse verfolgt hatten, spendeten dem scheidenden Chef nach seinem 45-minütigen Rechenschaftsbericht stehend Beifall. Den Dank der Partei formulierte anschließend der saarländische Ministerpräsident Peter Müller: "Was Ihnen weiterführend ist, ist ungeschwiegt", erklärte Müller. Die Partei komme zudem auch in Zukunft auf Schäuble nicht verzichten: "Die Ans Wolfgang Schäuble ist in der Union nicht zu Ende."

Starken Applaus bekam auch der ebenfalls als Mitglied zurücktretende CDU-Vize Norbert Blum. Der Ex-Sozialminister forderte die CDU auf, ihr soziales Gewissen nicht einer bestimmungslosen Modernisierung und Flächendeckung zu opfern: "Die Arbeitsplätze müssen dahin, wo die Menschen ihre Heimat haben, nicht die Menschen dahin, wo die Arbeitsplätze sind."

Im Hauptteil der Debatte am ersten Tag der Konferenz, die am heutigen Dienstag zu Ende geht, beschäftigen sich die Delegierten mit der Finanzkrise im Gefolge des Spendenmandats. Schatzminister Matthias Wissmann, der ebenfalls sein Amt abgibt, präsentiert dem Parteitag einen durchgefeilten Sanierungsplan, mit dem die Partei binnen fünf Jahren entschuldet werden soll. Der scheidende Vorstand wurde allerdings für die Finanzen entlastet, soweit diese nicht vom Spendenmandat betroffen sind.

Siehe Kommentar und Hintergrund

Abbildung 48: Beispiel Nachrichten⁶²

⁶² Quelle: Frankfurter Rundschau, 11.04.2000

Informationsarten für mobile Endgeräte

Abbildung 48 zeigt einen Artikel, der nur aus Text besteht. Die Textmenge liegt bei diesem Beispiel bei 3094 Zeichen und ist im Verhältnis sehr hoch. Dieses Beispiel enthält ausschließlich unstrukturierte Informationen in Form von Text, wobei die Besonderheiten wie Überschriften und Absätze hier außer Acht gelassen werden.⁶³

Die Intensität der User Interaktion ist in diesem Fall niedrig, da sich die Interaktion bei der Umsetzung auf ein paar Hyperlinks beschränkt.

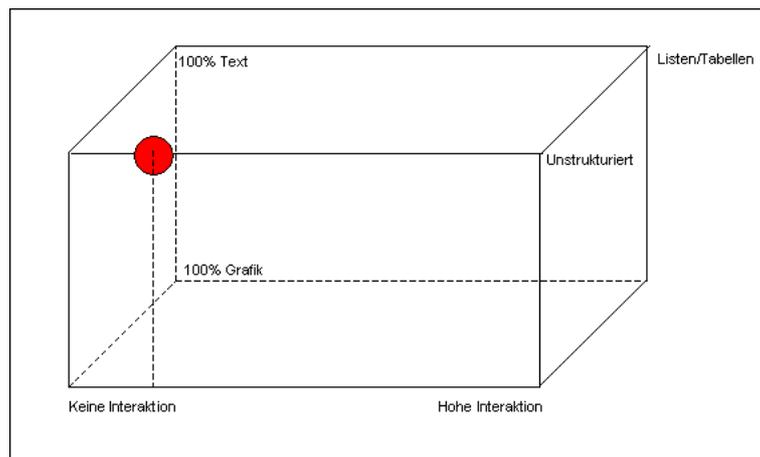


Abbildung 49: Nachrichten 3D-Raum

Nun gilt es, die vorhandenen Informationen so einzuordnen und gegebenenfalls zu konvertieren, dass daraus Informationseinheiten entstehen, die für die Nutzung auf mobilen Endgeräten geeignet sind. Im weiteren Verlauf wird gezeigt, dass die Textmengen, die später in WAP-Anwendungen verwendet werden, meist erheblich unter der Textmenge aus diesem Beispiel liegen. Das bedeutet, dass eine 1:1-Übertragung von Nachrichten aus Printmedien oder dem WWW nicht zweckmäßig und nutzerfreundlich ist.

3.3.3 Beispiel Zugfahrplan

Das Dokument des Zugfahrplans enthält in dem Beispiel in Abbildung 50 beide möglichen medialen Typen, Text und Grafik. Die zentralen Informationen werden durch Text repräsentiert. Die Grafiken stellen Ergänzungen zu den einzelnen Textinformationen dar.

⁶³ Betrachtet man Überschriften, Absätze usw. als strukturierte Information, nimmt der Grad der Strukturierung auf der Skala entsprechend zu.

Informationsarten für mobile Endgeräte

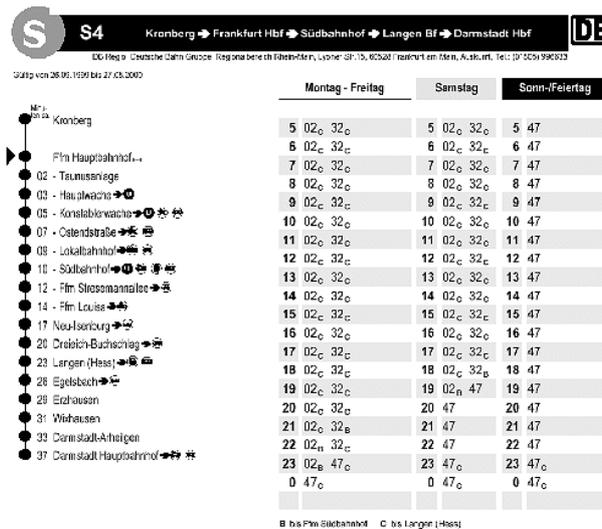


Abbildung 50: Beispiel Fahrplan⁶⁴

Die dargestellten Informationen sind sehr stark strukturiert. Die Textinformationen sind in Tabellen und Listen organisiert, die nicht nur Layoutfunktionen wahrnehmen.

Betrachtet man dieses Beispiel, so ist die User Interaktion nur schwach ausgeprägt, da keine Eingabe oder Auswahl erforderlich ist. Erst bei der späteren Umsetzung in WML kommen Eingabefelder oder Auswahllisten dazu. Daher ist die User Interaktion bei dieser graphischen Darstellung noch wenig intensiv.

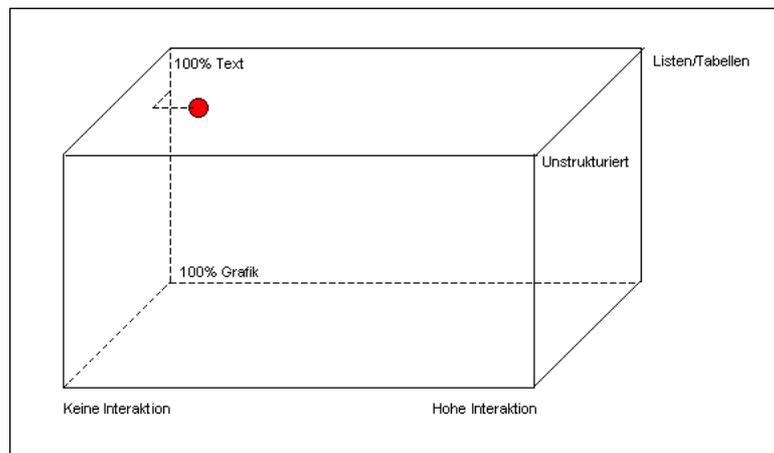


Abbildung 51: Fahrplan 3D-Raum

⁶⁴ Quelle: <http://www.rmv.de> (11.04.2000)

Informationsarten für mobile Endgeräte

3.3.4 Beispiel Portal

Ein Portal ist heute im WWW oft die erste Anlaufstation, die ein Nutzer zum Einstieg besucht. Im Prinzip sind die meisten Portale ähnlich aufgebaut. Sie bieten eine kategorisierte Auswahl an Verweisen zu anderen Webseiten. Als Beispiel wird hier das Portal von Yahoo! betrachtet.



Abbildung 52: Beispiel Portal⁶⁵

Das Beispiel in Abbildung 52 besteht im Wesentlichen aus kurzen, nicht zusammenhängenden Textabschnitten. Vereinzelt sind auch Grafiken eingestreut, die aber inhaltlich keine Bedeutung haben oder Zusatzinformationen bieten. Infolgedessen fallen sie bei der Bewertung nicht sehr stark ins Gewicht.

Die Textabschnitte sind in Tabellen und Listen organisiert, die hauptsächlich Layout-Funktionen wahrnehmen. Bei der Einordnung in die Klassifikation liegt der Grad der Strukturierung deswegen am Anfang des oberen Drittels. Da nahezu der gesamte Text mit Hyperlinks versehen ist, liegt die Intensität der User Interaktion auch im oberen Bereich.

⁶⁵ Quelle: <http://www.yahoo.com> (04.05.2000)

Informationsarten für mobile Endgeräte

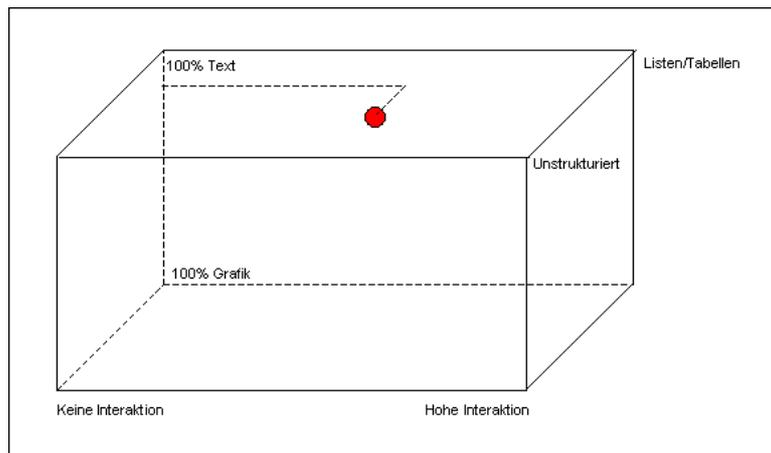


Abbildung 53: Portal 3D-Raum

3.3.5 Beispiel Aktienkurse

Das Beispiel in Abbildung 54 zeigt eine Übersicht aktueller Aktienkurse der DAX-Werte mit Zusatzinformationen wie prozentuale Differenz zum Vortag, Handelsvolumen und eine Auswahl an Charts verschiedener Aktienindizes.

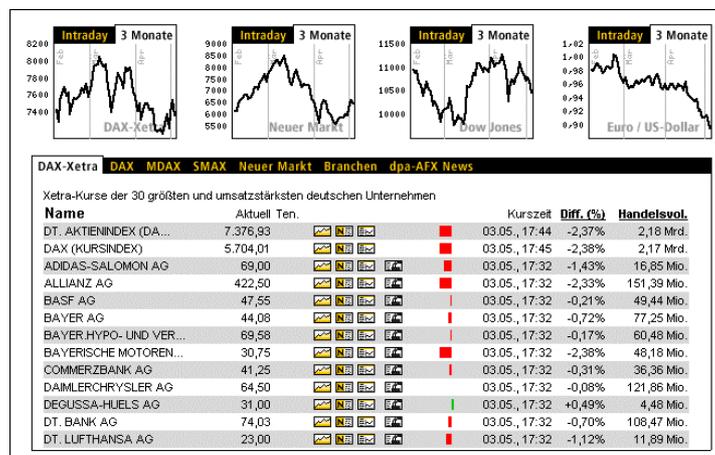


Abbildung 54: Aktienkurse⁶⁶

Die Textinformationen sind zum größten Teil in einer Tabelle untergebracht. Zusätzlich nimmt die Tabelle Grafiken auf, die per Hyperlinks auf Zusatzinformationen verweisen. Insgesamt handelt es sich hier um ein sehr stark strukturiertes Dokument. Neben den Grafiken, die auf Zusatzinformationen verweisen, bilden die Chartgrafiken das Gros des Grafikanteils.

⁶⁶ Quelle: <http://www.comdirect.de> (03.05.2000)

Informationsarten für mobile Endgeräte

Das Verhältnis von Text zu Grafik kann hier als nahezu ausgeglichen angesehen werden. Bei diesem Beispiel sind zwar zahlreiche Hyperlinks vorhanden, die aber nur auf Zusatzinformationen verweisen und für die inhaltliche Auswertung des Dokuments nicht aktiviert werden müssen. Aus diesem Grund spielt die User Interaktion hier eine untergeordnete Rolle. Die Einordnung der Informationen in das Klassifikationssystem zeigt Abbildung 55.

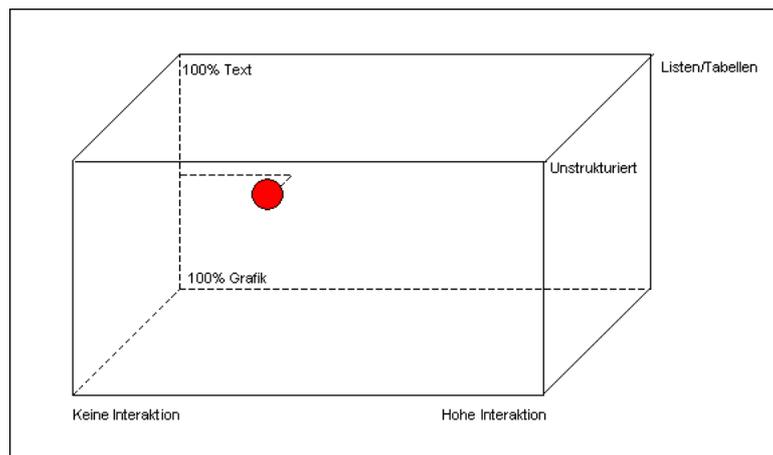


Abbildung 55: Aktien 3D-Raum

4 Entwicklung von WML-Seiten

4.1 Rahmenbedingungen

Den Mobilfunkbereich zeichnen verschiedene Rahmenbedingungen aus, die bei der Entwicklung von WAP-Anwendungen berücksichtigt werden müssen. Daher werden diese Rahmenbedingungen im Folgenden herausgearbeitet und näher erläutert. Die Rahmenbedingungen stellen, neben dem erarbeiteten Klassifikationssystem, die Basis für die Entwicklung der Lösungskonzepte dar.

4.1.1 Mobile Endgeräte

Prinzipiell lässt sich jedes mobile Gerät, das über ein Display und entsprechenden Speicherplatz für den Microbrowser verfügt, WAP-fähig machen. Neben den heute bekannten Geräten, wie Handys und PDAs, sind auch tragbare MP3-Player, Digitalkameras, Gameboys oder Uhren als WAP-Geräte vorstellbar. Auf der CeBIT 2000 wurde sogar eine WAP-Maus mit

integriertem LCD-Display präsentiert.⁶⁷ Selbst das Display muss bei zukünftigen Entwicklungen, wie Spracherkennung und Sprachausgabe, keine zwingende Voraussetzung darstellen.

Es sei erwähnt, dass in diesem Zusammenhang, Notebooks, Sub- und Mininotebooks nicht mit in die Gruppe der mobilen Endgeräte für WAP-Anwendungen genommen werden.

4.1.1.1 Allgemeine Merkmale

Die hier besprochenen mobilen Endgeräte weisen einige Gemeinsamkeiten auf, die im weiteren aufgezeigt werden. Dabei handelt es sich bei den Gemeinsamkeiten in den meisten Fällen um gemeinsame Beschränkungen im Vergleich zu stationären Geräten, wie PCs.⁶⁸

4.1.1.1.1 Leistungsfähigkeit

Die Leistungsfähigkeit bezieht sich auf unterschiedliche Bereiche der mobilen Endgeräte. Da die Geräte nicht permanent über das Stromnetz versorgt werden können, müssen Energiespeicher die Aufgabe der Energieversorgung übernehmen. Die mobilen Endgeräte sollen möglichst klein sein, was eine Beschränkung der Energiespeicherkapazität zur Folge hat. Die kleine Bauweise und der begrenzte Energievorrat sind wiederum Gründe für die geringere Rechenleistung, welche mobile Endgeräte besitzen. Ebenfalls betroffen sind die verfügbaren Datenspeicher, die auch den oben genannten Restriktionen unterliegen.

4.1.1.1.2 Eingabegeräte

Die Gemeinsamkeit bei den Eingabegeräten liegt darin, dass Benutzereingaben im Vergleich zur Tastatur-Maus-Kombination des PCs komplizierter zu handhaben sind. Das Problem, welches auf einen Entwickler von WAP-Anwendungen zusätzlich zukommt ist, dass nicht von einer einheitlichen Kombination von Eingabegeräten ausgegangen werden kann. Die Anwendung muss sich daher für alle Eingabearten ähnlich gut eignen.

⁶⁷ Vgl. "Maus mit WebCam und Internet-Anschluss", Heise Online News, 24.02.2000, <http://www.heise.de/newsticker/data/ipa-24.02.00-000/> (14.04.2000)

⁶⁸ Vgl. AU SYSTEM RADIO (1999), S.16

Entwicklung von WML-Seiten

4.1.1.1.3 Microbrowser

Trotz der jungen Geschichte von WAP, gibt es schon erhebliche Unterschiede, was die Darstellungsweise der WML-Seiten in den verschiedenen Microbrowsern betrifft. Derzeit sind Microbrowser von drei Herstellern in den verschiedenen Geräten implementiert oder verfügbar. Dabei liefert Phone.com an etwa 90 Prozent der Hersteller für mobile Endgeräte weltweit. Da Nokia und Ericsson aber nicht zu den Abnehmern gehören, sondern auf Eigenentwicklungen setzen, kann nicht davon ausgegangen werden, dass auch 90 Prozent der genutzten mobilen Geräte den Microbrowser von Phone.com verwenden.⁶⁹ Da auch im mobilen Bereich die Hersteller anfangen, proprietäre Erweiterungen in ihren Browserversionen zu implementieren, ist leider davon auszugehen, dass sich das im WWW üblich gewordene Procedere, im mobilen Bereich fortsetzt.⁷⁰

Ein intensives Testen aller Browserversionen und verfügbaren Endgeräte, ist bei der Entwicklung von WAP-Anwendungen ebenso unerlässlich wie bei der Entwicklung entsprechender Anwendungen im WWW.

4.1.1.1.4 Displays

Das Hauptproblem für die Darstellung der gewünschten Informationen stellen die Ausgabegeräte, also die Displays dar. Insgesamt sind die Displays natürlich aufgrund der geringen Abmessungen der mobilen Endgeräte auch klein.

Nicht nur die Displaygröße macht die Anwendungsentwicklung kompliziert, sondern auch die Abmessungen der Displays. Bei PC-Monitoren liegt – bis auf neuere Modelle im 16:9-Format – immer das gleiche Seitenverhältnis von 4:3 vor. Selbst die Größenunterschiede der Monitore beschränken sich auf ein paar wenige gebräuchliche Abstufungen, wie 15-, 17- und 19-Zoll Monitore. Anders verhält sich das bei den mobilen Endgeräten. Zum einen sind die Seitenverhältnisse der Displays sehr unterschiedlich und zum anderen gibt es auch keine quasi-standardisierte Abstufung der Displaygrößen.

Der Großteil der Displays arbeitet nur mit einer Farbtiefe von einem Bit, wenige können verschiedene Graustufen darstellen und ganz vereinzelt sind Farbdisplays anzutreffen. Da Gra-

⁶⁹ Vgl. MÜLLER-VEERSE, Falk (1999), S.28

⁷⁰ Microbrowser von Phone.com, <http://developer.phone.com/dev/ts/up/upsdk40.html#extensions> (29.05.2000)

fiken in WAP aber ohnehin nur eine Farbtiefe von einem Bit aufweisen dürfen, können leistungsfähigere Displays nicht ausgenutzt werden.

4.1.1.2 Spezielle Merkmale

Die heute verfügbaren Geräte werden hier in zwei Kategorien eingeteilt. Die eine Gruppe bilden die Mobiltelefone, die auch den Großteil der WAP-fähigen mobilen Endgeräte darstellen. Die zweite Gruppe sind die sogenannten Personal Digital Assistants (PDA), wobei die PDAs noch einmal in Palm-PDAs und Communicator-PDAs unterteilt sind.

4.1.1.2.1 Mobiltelefone

Die größte Gruppe der WAP-fähigen mobilen Endgeräte sind die Mobiltelefone. Gleichzeitig sind es auch die Geräte mit den stärksten Beschränkungen im Hinblick auf Energievorrat, CPU-Leistung, Speicherplatz, Eingabemöglichkeiten und Displaygröße. Bei der Entwicklung von WAP-Anwendungen ist es also sehr ratsam, sich an dieser Gruppe zu orientieren, um möglichst den kleinsten gemeinsamen Nenner zu finden.

Die Mobiltelefone besitzen in der Regel Displays von wenigen Zentimetern Größe, Platz für 4 Zeilen Text mit je 12-16 Zeichen Länge.⁷¹ Das Display hat etwa ein Größenverhältnis von 3:2. Als Eingabegerät steht typischerweise eine Telefontastatur mit einigen programmierbaren Zusatztasten, sowie Funktionstasten für die Navigation zur Verfügung.⁷²

Die Größe eines Datenblocks, die ein Mobiltelefon standardmäßig empfangen kann, liegt bei 1400 Bytes. Aus diesem Grund sollten auch die WML- und WMLScript-Dateien in kompiliertem Zustand eine Größe von 1400 Bytes nicht überschreiten. Das betrifft auch die Grafikdateien im WBMP-Format.⁷³

4.1.1.2.2 PDAs

Die PDAs zeichnen sich durch eine insgesamt höhere Leistungsfähigkeit gegenüber den Mobiltelefonen aus. Insbesondere betrifft das die Speicherkapazität und Displaygröße. Die

⁷¹ Vgl. PHONE.COM (1999), S.3

⁷² Vgl. SIETMANN, Richard (1998), S.204

⁷³ Vgl. BAGER, Jo (1999), S.130

Entwicklung von WML-Seiten

Unterschiede innerhalb der Gruppe PDAs beziehen sich im Wesentlichen auf die Eingabegeräte sowie die Dimensionen der Displays.

Palm-PDAs

Bei den Palm-PDAs handelt es sich um Geräte, deren Displays zwischen 10 und 20 Zeilen mit je 40 Zeichen anzeigen können. Das Größenverhältnis der Anzeigen liegt etwa bei 3:4. Die Eingabe erfolgt in den meisten Fällen mit Hilfe eines Stiftes über ein drucksensibles Display. Für Standardfunktionen existieren meistens einige zusätzliche Tasten.

Communicator-PDAs

Die derzeit leistungsfähige Gruppe von WAP-fähigen Endgeräten hinsichtlich Eingabemöglichkeiten und Displaygröße bilden die Communicator-PDAs. Die Anzeigegröße schwankt zwischen 20 und 30 Zeilen mit jeweils 80 bis 100 Zeichen.⁷⁴ Eine weitere Gemeinsamkeit ist das Größenverhältnis der Displays, was bei etwa 3:1 liegt. Als Eingabemöglichkeiten stehen den Communicator-PDAs nahezu vollwertige Tastaturen zur Verfügung. In einigen Fällen werden diese zusätzlich durch Touchscreens ergänzt, die mit dem Finger oder einem Stift bedient werden können.

4.1.2 Mobilfunknetz

Das Mobilfunknetz ist hinsichtlich der Leistungsfähigkeit ebenfalls Beschränkungen im Vergleich zu den Übertragungsmöglichkeiten des Festnetzes unterworfen.

Das europäische GSM-Netz erlaubt derzeit eine standardmäßige Übertragungsgeschwindigkeit von 9,6 Kbit/s.⁷⁵ Das bedeutet, dass eine WML- oder WMLScript-Datei mit 1400 Zeichen, durch die Umwandlung in Binärcode vor der Übertragung, etwa in einer Sekunde auf dem mobilen Endgerät erscheinen müsste. Allerdings liegen die Erfahrungswerte bei etwa drei bis fünf Sekunden für eine derartige Datei.⁷⁶ Das liegt unter anderem an den hohen Signallaufzeiten der Datenpakete, die im Festnetzbereich bei einigen Millisekunden liegen, im Mobilfunknetz aber einige Sekunden betragen können.⁷⁷

⁷⁴ Vgl. SMITH, Josh (2000)

⁷⁵ Vgl. ŽIVADINOVIĆ, Dusan (1999), S.122

⁷⁶ Vgl. KOSSEL, Axel (2000), S.194

⁷⁷ Vgl. SIETMANN, R. (1998), S.203

Weiterhin sind die Verbindungen in Mobilfunknetzen weniger stabil und müssen daher häufig wieder aufgenommen werden. Hinzu kommt, dass Mobilfunknetze eine geringere Verfügbarkeit haben und es deshalb zu Verzögerungen bei dem Verbindungsaufbau kommen kann.⁷⁸

Wichtig in diesem Zusammenhang ist also, dass die Datenmengen, die übertragen werden müssen, möglichst gering sind und dass das Mobilfunknetz so wenig wie möglich beansprucht wird.

4.1.3 Endnutzer

"Sowieso ist unklar, auf welche Zielgruppe WAP zugeschnitten wird. Weder in der Branche noch in den Medien hat man sich bis jetzt auf eine Zielgruppe geeinigt."⁷⁹

Die Zielgruppe für den Bereich WAP zu definieren, fällt keineswegs leicht. Es lässt sich allerdings sagen, dass ein Nutzer von WAP-Anwendungen in der Regel daran interessiert sein wird, schnell und unkompliziert an spezielle Informationen zu gelangen. Bei mobilen Anwendungen wird das Information Retrieval im Vordergrund stehen und nicht das Browsen oder Surfen durch Informationswelten. Ein weiterer Unterschied zum WWW liegt in der Erwartungshaltung des Anwenders. Die WAP-Dienste befinden sich in der Telekommunikationsumgebung und Anwender von Telekommunikationsgeräten und -dienstleistungen erwarten, dass diese auf Anhieb einwandfrei funktionieren. Im Bereich der Computer besitzt der Anwender inzwischen häufig eine höhere Toleranzgrenze hinsichtlich Fehlfunktionen und kleineren Hängern, da sie – leider – zum Alltag eines jeden Computernutzers gehören.⁸⁰

Insgesamt lässt sich erkennen, dass die Rahmenbedingungen für die Entwicklung von Informationsdiensten für mobile Endgeräte sehr restriktiv sind. Daher ist es notwendig, diese bei der Konzeption von WAP-Anwendungen sehr genau zu kennen und zu beachten.

⁷⁸ Vgl. WAP FORUM (1999d), S.8

⁷⁹ Vgl. BUNZ, Mercedes (2000)

⁸⁰ Vgl. HOBBS, John Maxwell (1999)

Entwicklung von WML-Seiten

4.2 Lösungskonzepte für relevante WAP-Bereiche

In den nächsten Kapiteln werden Lösungskonzepte für relevante Beispiele von WAP-fähigen Informationen erarbeitet und diskutiert. Wie bei den Anwendungsbeispielen der Klassifikation, beziehen sich die Lösungskonzepte nicht auf komplette Anwendungen, sondern auf wesentliche Teilbereiche, die in WML in einigen wenigen Cards realisiert werden können. Aufgrund des zeitlichen Rahmens und der Aufgabenstellung der Diplomarbeit werden dabei nur Lösungskonzepte für Mobiltelefone erarbeitet. Alle entwickelten Beispiele können mit einem WAP-Browser über den WAP-Server der GMD abgerufen werden. Die URL lautet <http://rom.darmstadt.gmd.de/budach/i.wml>.

Im Verlauf der Entwicklung der Lösungskonzepte werden zunächst Bereiche behandelt, die alle WAP-Anwendungen durchgängig betreffen. Im zweiten Schritt folgen Lösungsvorschläge für spezielle Informationsszenarien, die anschließend an konkreten Beispielen erläutert werden.

Die gegenwärtige Entwicklung des Internets bzw. WWWs erweckt den Eindruck, dass die eigentliche Information immer mehr in den Hintergrund tritt und stattdessen der Schwerpunkt auf das Layout und das graphische Design gelegt wird. Genau entgegengesetzt muss bei der Entwicklung einer Anwendung für den mobilen Bereich vorgegangen werden. Den zentralen Punkt der Anwendung muss die Information darstellen.⁸¹ Denn das eigentliche Problem bei der Entwicklung von WAP-Anwendungen ist nicht das Erlernen von neuen Spezifikationen und Markup-Sprachen, sondern das Herausfiltern der relevanten Informationen und deren Bereitstellung in einer Weise, die den gegebenen Rahmenbedingungen entspricht.⁸²

4.2.1 Navigation

WML bietet verschiedene Möglichkeiten der Navigation. Neben den, aus dem WWW bekannten Wegen der Navigation, wie Hypertext, Auswahllisten oder Suchfunktionen, gibt es bei WML die Möglichkeit, verschiedene Aktionen zu definieren. Diese Aktionen enthalten Navigationselemente und erscheinen nicht direkt in der Anwendung. Im Folgenden werden die einzelnen Navigationsvarianten auf ihre Einsatzfähigkeit bei verschiedenen Informationsmengen und -arten getestet und bewertet.

⁸¹ Das sollte natürlich für jeden Informationsdienst unabhängig von der Bereitstellungsform gelten.

⁸² Vgl. BAGER, Jo (1999), S.128; ZIERL, Marco (2000), S. 42 und SMITH, Josh (2000)

4.2.1.1 Navigation über Aktionen

Die allgemeine Navigation kann in WML über das Element `do` in Zusammenhang mit den Elementen `go` oder `prev` definiert werden (Siehe auch Kapitel 2.1.3.5). Ein Vorteil dieser Version ist, dass die einzelnen Navigationsoptionen nicht im Fluss der Anwendung erscheinen und so platzsparend eingesetzt werden können. Die Darstellung der Navigationselemente obliegt dabei dem jeweiligen Microbrowser. In den meisten Fällen belegt der Microbrowser zwei Softkeys mit den definierten Aktionen. Sind dabei mehr als zwei Aktionen definiert, kann über einen der Softkeys ein Menü erreicht werden, das die weiteren Aktionen enthält. Ist ein Softkey direkt mit einer Aktion belegt, kann diese sehr einfach ausgeführt werden. Das Aktionsmenü hat den Vorteil, dass alle Aktionen an einem Ort zusammengefasst sind und es daher eine gute Übersicht bietet. Zusätzlich können die Aktionen über das Element `template` für alle Cards innerhalb eines Deck definiert werden, was nicht nur Aufwand spart, sondern sich auch zu Gunsten der Dateigröße auswirkt.

Nachteil der Aktionen ist unter anderem die angesprochene uneinheitliche Darstellungsweise der Aktionen durch die Microbrowser. Für den Entwickler bedeutet dies, dass möglicherweise nicht alle Darstellungsweisen optimal für seine Anwendung passen und er dadurch unterschiedliche Versionen vorhalten muss. Für den Anwender sind die Aktionen unter Umständen gewöhnungsbedürftig, da sie nicht, wie aus dem WWW gewohnt, direkt in der Anwendung stehen. Daraus ergibt sich ein weiterer Nachteil der Aktionen. Ist die gewünschte Aktion nicht direkt über einen Softkey erreichbar, muss für die Navigation das Aktionsmenü aufgerufen werden, was ein Verlassen des Anwendungsflusses bedeutet.

Insgesamt überwiegen aber die Vorteile der Aktionen in WML. Da es für den Nutzer aufgrund der geringen graphischen Gestaltungsmöglichkeiten und der kleinen Displays schwierig ist, den Überblick über den aktuellen Kontext zu behalten, muss ihm die Möglichkeit gegeben werden, an einen ihm bekannten Punkt zurückzukehren.⁸³ Die Aktionen eignen sich daher insbesondere für das Anbieten von Standardnavigationselementen, die in jeder Card einer WAP-Anwendung verfügbar sein sollten. Dazu gehören vor allem die Navigation zur Startseite der Anwendung und das Zurückkehren zur vorhergehenden Seite. Im folgenden Kasten sind Einsatzgebiet und Lösungsvorschlag noch einmal zusammengefasst.

⁸³ Vgl. KUHLEN, Rainer (1997), S.362-363

Entwicklung von WML-Seiten

Konzept: Herstellen einer Navigationsübersicht für Anwendungen

Lösungsvorschlag:

- Definition von Navigationsmöglichkeiten zu wichtigen Stellen der Anwendung über Aktionen in Templates
- Definition der Historyfunktion in der gesamten Anwendung

4.2.1.2 Navigation in der Anwendung

Die Navigationsvarianten innerhalb der Anwendungen entsprechen im Wesentlichen den Möglichkeiten aus HTML. Die einfache Verknüpfung von Cards oder Decks kann mit Hilfe von Hypertext realisiert werden. Ein Vorteil von Hypertext ist, dass die Navigationselemente im direkten Zusammenhang zu dem Inhalt dargestellt werden können. Die Navigationsmöglichkeit erscheint somit direkt in der Anwendung und ist dadurch leicht zu erreichen. Ein weiterer Vorteil ist die allgemeine Verbreitung von Hypertext durch das WWW. Der Nutzer wird aus diesem Grund wahrscheinlich weniger Probleme mit der Navigation über Hypertext haben, da das System bekannt ist. Ferner weichen auch die Darstellungsvarianten von Links in den Microbrowsern weniger voneinander ab. Infolgedessen hat der Entwickler eine bessere Kontrolle über die Darstellungsweise seiner Anwendung.

Der Vorteil, dass die Navigation direkt in der Anwendung erscheint, stellt gleichzeitig einen Nachteil dar, da diese Form der Darstellung sehr platzintensiv werden kann. Infolgedessen ist eine Definition eines Navigationsmenüs mit Hilfe von Hypertext, das über die gesamte Anwendung verfügbar ist, nicht sinnvoll. Es besteht auch keine Möglichkeit, alle Cards eines Deck durch ein Template mit den gleichen Navigationselementen auszustatten. Das Einsatzgebiet des Hypertexts liegt vielmehr in speziellen Navigationsoptionen der einzelnen Cards.

Konzept: Navigationselemente im direkten Kontext der Anwendung

Lösungsvorschlag:

- Verwendung von Textlinks innerhalb einer Card, um cardspezifische Navigationswege zu realisieren

Handelt es sich bei den Informationen um Datenmengen, bietet WML zwei Möglichkeiten der Navigation. Zum einen kann eine Suche über Textfelder durchgeführt werden oder die Informationen werden über Listen bereitgestellt.

Entwicklung von WML-Seiten

Welche Form der Navigation konkret einzusetzen ist, hängt stark von Art und Umfang der Information ab. Sind die Datenmengen sehr umfangreich, lässt sich beispielsweise eine Navigation mit mehreren hierarchisch angeordneten Listen erstellen, die ein Navigieren vom Allgemeinen zum Speziellen zulässt. Ein weiterer Ansatz könnte eine Kombination von Textfeldern und Listen darstellen. So ist denkbar, dass Datenmengen zuerst per Eingabe der ersten beiden Buchstaben eingeschränkt werden und diese eingeschränkte Menge dann über Listen weiter durchsucht werden kann. Im Vordergrund steht dabei immer, die Benutzereingaben zu reduzieren.

Konzept: Vermeidung von Benutzereingaben

Lösungsvorschlag 1:

- Repräsentation der Daten über hierarchisch strukturierte Listen

Lösungsvorschlag 2:

- Kombination von Textsuche und Listen. Eingabe der ersten beiden Buchstaben und Darstellung des Ergebnisses als Liste

Die oben dargestellte abstrakte Beschreibung soll an einem Beispiel verdeutlicht werden. Ein Beispiel für hierarchisch strukturierte Optionslisten ist der Taschenflugplan der Deutschen Lufthansa, der als WAP-Anwendung realisiert wurde.⁸⁴ Um eine Flugstrecke angezeigt zu bekommen, müssen die Start- und Zielflughäfen ausgewählt werden. Dies geschieht über mehrere Optionslisten.



Abbildung 56: Flugplan LH 1



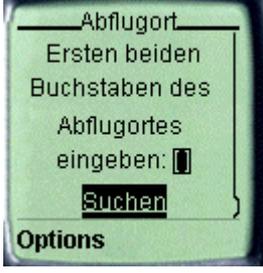
Abbildung 57: Flugplan LH 2

In der ersten Auswahl wird der Bereich des Alphabets eingegrenzt. Es bestehen elf Auswahlbereiche (Abbildung 56). Im zweiten Schritt wird der Abflugort konkret ausgewählt. Da-

Entwicklung von WML-Seiten

bei stehen im Fall "Aalborg bis Beirut" 37 Abflugorte zur Auswahl (Abbildung 57). Hier muss der Nutzer bei der Auswahl keinen Text eingeben.

Die gleiche Anwendung lässt sich mit dem kombinierten Ansatz umsetzen. Die hierarchischen Listen werden durch ein Texteingabefeld ersetzt. Um einen Abflugort zu ermitteln, gibt man die ersten beiden Buchstaben des Abflugortes ein. Die Antwortmenge wird dann als Auswahlliste dargestellt. Konkret kann dies wie folgt aussehen.

Nokia WAP Toolkit 1.2	WML (lh1.wml + lh2.wml)
	<pre data-bbox="531 701 1212 974"><wml> <card title="Abflugort"> <p align="center"> Ersten beiden Buchstaben des Abflugortes eingeben: <input name="auswahl" title="Abflugort" value="" type="text"/> Suchen </p> </card> </wml></pre>
<p>Abbildung 58: Flugplan Lösung 1</p>	
	<pre data-bbox="531 1052 1212 1444"><wml> <card title="Abflugort"> <p align="center"> Abflugort w&#xE4;hlen: <select name="abflug" title="Abflugort" value=""> <option value="1">Baku</option> <option value="2">Bandar S.B</option> [...] <option value="7">Bayreuth</option> </select> weiter </p> </card> </wml></pre>
<p>Abbildung 59: Flugplan Lösung 2</p>	

Diese Lösung hat den Vorteil, dass das Scrollen auf ein erträgliches Maß reduziert wird, da im ungünstigsten Fall nur sieben Orte nach der Suche zur Auswahl stehen. Welche Variante im Endeffekt favorisiert wird, muss von Fall zu Fall entschieden werden.

Ergänzend für alle Varianten der Navigation soll erwähnt werden, dass für die Navigation notwendige Dateinamen möglichst kurz gehalten werden. So kann aus der im WWW übli-

⁸⁴ <http://wap.is.danet.de/lh/flugplan/index.html> (29.05.2000)

chen Schreibweise wie `index.html`, bei der Umsetzung in WML `i.wml` werden. Diese Schreibweise steuert ebenfalls einen kleinen Beitrag zur Vermeidung von Benutzereingaben bei. Die Eingabelänge von URLs kann somit reduziert werden.

4.2.2 Cards und Decks

Eine WML-Datei besteht, wie in Kapitel 2.1.1.2 beschrieben, in der Regel aus einer Folge von Cards, die als eigenständige Informationseinheiten einzeln von dem mobilen Endgerät angezeigt werden. Diese Aufteilung in mehrere Cards innerhalb eines Deck ist natürlich nicht zwingend. Ein Deck kann auch nur aus einer einzelnen Card bestehen. Vor der Umsetzung der Informationen in WML muss daher überlegt werden, ob eine Darstellung in einer einzelnen Card oder in mehreren Cards eines Deck erfolgen soll.

Die Aufteilung der Informationen in mehrere Cards hat den Vorteil, dass die Übersicht über die Inhalte besser gestaltet werden kann. Der Nachteil ist, dass durch das häufige Wechseln der Cards der Anwendungsablauf immer wieder unterbrochen wird.

Bei der Darstellung der Informationen in einer einzelnen Card bleibt zwar der Ablauf bestehen, allerdings leidet die Übersichtlichkeit und der Nutzer muss sich – je nach verfügbarem Eingabegerät – sequenziell durch die Informationsseite arbeiten. Einen entscheidenden Faktor spielt hier wiederum die Art der Information, die in WML umgesetzt werden soll.

Konzept: Darstellung von zusammenhängenden Informationen

Lösungsvorschlag:

- Definition einer einzelnen Card innerhalb eines Decks, in der sequenziell navigiert werden kann

Konzept: Darstellung von einzelnen Informationseinheiten

Lösungsvorschlag:

- Verwendung einer Card pro Informationseinheit und Verlinkung der einzelnen Cards untereinander.

Entwicklung von WML-Seiten

Die oben stehende Zusammenfassung soll durch die folgenden Beispiele erläutert werden.

Nimmt man etwa den Text einer Newsmeldung, dann handelt es sich um einen zusammenhängenden Textabschnitt. Daher scheint es sinnvoll, den Text in einer Card unterzubringen. Anders verhält es sich bei Informationen, die als einzelne Einheiten betrachtet werden können. Das Suchergebnis aus einer Telefondatenbank kann beispielhaft erwähnt werden. Die gefundenen Personen mit ihren Telefonnummern und Zusatzinformationen werden jeweils in einer einzelnen Card gespeichert. Somit kann die Übersichtlichkeit der Ergebnisse gewahrt werden (Abbildung 60 - Abbildung 62).



Abbildung 60: Telefonbuch1



Abbildung 61: Telefonbuch2



Abbildung 62: Telefonbuch3

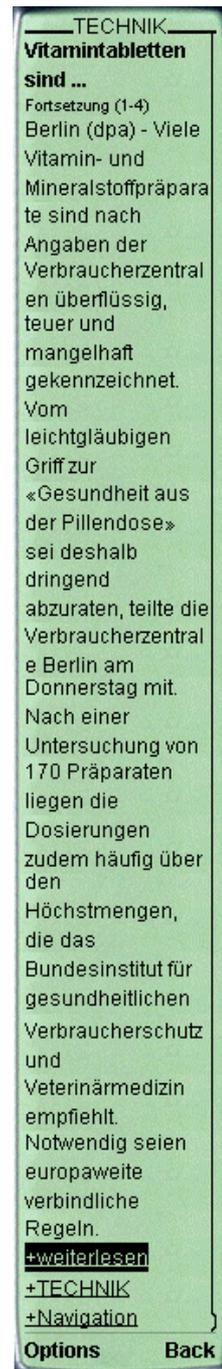


Abbildung 63: Beispiel Nachrichten

```
<wml>
  <template>
    <do type="prev" label="Zur&#xFC;ck">
      <prev/>
    </do>
  </template>
```

Entwicklung von WML-Seiten

```
<card id="Ergebnisse" title="">
  <p align="center">Budach Jens<br/>
  64546 M&#xF6;rfeiden-Walldorf<br/>06105-654242<br/>
  <a href="#c2">n&#228;chster Eintrag</a></p>
</card>
<card id="c2">
  <p align="center">
  Budach Klaus J&#xF6;rg Dr.med.<br/>
  64546 M&#xF6;rfeiden-Walldorf<br/>06105-71238<br/>
  <a href="#c3">n&#228;chster Eintrag</a></p>
</card>
<card id="c3">
  <p align="center">Budach Magdalene Dr. med.<br/>
  64546 M&#xF6;rfeiden-Walldorf<br/>06105-41078<br/>
  <a href="#c4">n&#228;chster Eintrag</a></p>
</card>
</wml>
```

Entwicklung von WML-Seiten

4.2.3 Grafiken

Die Verwendung von Grafiken in WAP-Anwendungen ist mit verschiedenen Beschränkungen verbunden. Wie bereits in Kapitel 2.1.3.4 beschrieben, können in der derzeitigen Version von WAP bzw. WML nur Schwarzweiß-Grafiken verwendet werden. Die überwiegend kleinen Displays der mobilen Endgeräte erschweren den Gebrauch von Grafiken zusätzlich. Überdies stellen die Microbrowser die Grafiken teilweise unterschiedlich dar, einige zeigen die Grafiken invertiert an. Einen interessanten Ansatz hat das WAP Forum bei der Definition von WML mit der Möglichkeit verfolgt, alternative lokale Grafiken anzugeben. Diese Grafiken sind im entsprechenden Microbrowser fest implementiert, was einen Download überflüssig macht. Der Nachteil besteht darin, dass in der WML-Spezifikation nicht definiert wurde, welche Grafiken jeder Microbrowser implementieren muss. Das bedeutet, dass eine einheitliche Darstellung lokaler Grafiken nicht gewährleistet ist.

Grundsätzlich lässt sich sagen, dass ein Dokument ab einem gewissen Grafikanteil nicht mehr sinnvoll in WML überführt werden kann. Dieser Anteil wird bei etwa 50 Prozent gesehen. Infolgedessen fallen Dokumente mit einem höheren Anteil an Grafiken nicht mehr in den hier zu betrachtenden Problemraum.

Bei Dokumenten mit Grafikanteil sollten zunächst die enthaltenen Grafiken analysiert werden. Handelt es sich dabei um inhaltstragende Grafiken, sollten diese den Rahmenbedingungen nach aufbereitet oder neu erstellt werden. Dabei ist darauf zu achten, dass nur die für die Darstellung notwendigen Teile der Grafik übernommen werden. Eine Ausgliederung der aufbereiteten Grafiken auf separate Cards, ist ebenfalls zu empfehlen. Bieten die Grafiken Zusatzinformationen zur eigentlichen Anwendung, gibt es die Möglichkeit, sie in tiefer gelegenen Ebenen anzubieten oder durch beschreibenden Text zu ersetzen. Ein Nutzer eines WAP-Dienstes wird in den wenigsten Fällen zufällig über ein Angebot stolpern und weiß aus diesem Grund in der Regel, wo er sich befindet. Deswegen sind auch Begrüßungsarien mit Logos aus Marketinggesichtspunkten nicht zwingend.

Bei der Verwendung von Grafiken ist generell darauf zu achten, dass zu jeder Grafik ein alternativer Text über das Attribut `alt` des Elements `img` angeboten wird. Damit können auch Nutzer, die mobile Endgeräte ohne Grafikerunterstützung verwenden, den Inhalt der Grafik in gewissem Maße erkennen. Nicht zu empfehlen sind verlinkte Grafiken, da unter Umständen nicht erkennbar ist, dass es sich um einen Hyperlink handelt. Eine Übersicht über die Lösungsvorschläge bietet der folgende Merkkasten.

Konzept: Verwendung von Grafiken

Lösungsvorschlag:

- Nur inhaltstragende Grafiken übernehmen oder aufbereiten
- Ausgliederung der Grafiken auf separate Cards
- Verwendung eines alternativen Texts
- Grafiken nicht mit Links hinterlegen

4.2.4 Lösungskonzept Überweisung

Die einzelnen Lösungskonzepte sollen sich nicht nur auf das konkrete Beispiel beschränken, sondern einen größeren Raum innerhalb des Klassifikationssystems um das jeweilige Beispiel herum abdecken. Damit sollen für die wahrscheinlichsten Kombinationen der Klassifikationsmerkmale Lösungsvorschläge angeboten werden.

Der in Abbildung 64 dargestellte Bereich an Kombinationsmöglichkeiten soll durch das Lösungskonzept für das Überweisungsbeispiel abgedeckt werden.

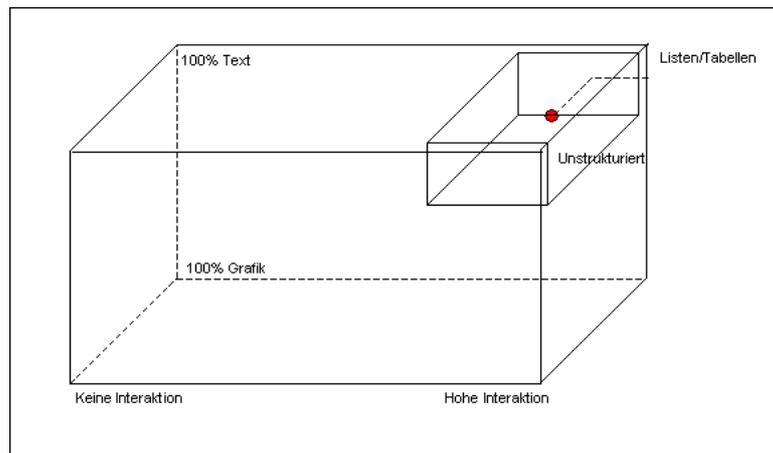


Abbildung 64: Lösungskonzept Überweisung 3D-Raum

Die hier betrachteten Informationseinheiten besitzen wenig Grafikanteil, aber eine hohe User Interaktion durch viele Eingabefelder. Da die Strukturierung bei diesem Beispiel nur dem Layout dient, findet sie bei der Umsetzung in WML keine Beachtung. Daher kann das Lösungskonzept auch für weniger strukturierte Informationen verwendet werden. Es wird davon ausgegangen, dass die hier vorliegende Merkmalskombination auch immer eine geringe Menge an Text aufweist.

Bei großen Formularen ist es sinnvoll, zusammengehörige Eingabefelder zu finden, um sie als einzelne Cards darstellen zu können. In dem Fall der Überweisung (Abbildung 46) kön-

Entwicklung von WML-Seiten

nen die Felder "*Empfänger*" und "*Konto-Nr. des Empfängers*", "*Bankleitzahl*" und "*Kreditinstitut*", sowie die Felder "*DM od. EUR*" und "*Betrag*" als zusammenhängende Einheiten betrachtet werden. Die vierte Card kann dann die Felder "*Kunden-Referenznummer*" und "*noch Verwendungszweck*" enthalten. Damit sind die Felder "*Kontoinhaber*" und "*Konto-Nr. des Kontoinhabers*" noch übrig.

In diesem Fall würde es sich um eine Anwendung handeln, bei der sich der Nutzer mit Benutzername und Kennwort anmeldet. Das bedeutet, dass verschiedene Informationen über den Nutzer bekannt sind und sich dadurch Eingabefelder des Formulars einsparen lassen. Konkret heißt das, dass die Felder "*Kontoinhaber*" und "*Konto-Nr. des Kontoinhabers*" nicht ausgefüllt werden müssen und so eine Card eingespart werden kann. Aufgrund der Tatsache, dass sich die Felder "*Bankleitzahl*" und "*Kreditinstitut*" aufeinander beziehen, kann in der Anwendung ein weiteres Feld eingespart werden, da der "*Name des Kreditinstituts*" automatisch nach Eingabe der "*Bankleitzahl*" eingetragen wird.

Die Anordnung von Eingabefeldern in WAP-Anwendungen sollte nach deren Wichtigkeit erfolgen. Felder, die unbedingt ausgefüllt werden müssen, stehen am Anfang des Dokuments. Optionale Felder dagegen werden weiter unten im Dokument oder auf einer separaten Card angeboten. Zusätzlich sollten Funktionen angeboten werden, die es ermöglichen, die Einträge des Formulars wieder zu löschen. Enthalten die Eingabefelder beschreibenden Text, sollte dieser nach Möglichkeit gekürzt werden. Angaben über Feldlängen, Eingabebeschränkungen und Ähnliches, können bei der Definition der Felder in WML angegeben werden und müssen nicht als Beschreibung auftauchen.

Die Struktur der Dokumente kann in den wenigsten Fällen 1:1 in WAP-Anwendungen übernommen werden. Solange es sich um Strukturierung für Layoutzwecke handelt, ist das wenig problematisch. Bei der Umsetzung dieses Beispiels werden keine Tabellen zur Ausrichtung der Eingabefelder verwendet werden. Über dies hinaus dürfen WML-Tabellen nur Grafiken und Text enthalten.

Grafiken sollten weggelassen werden, wenn sie keine inhaltliche Bedeutung haben. So kann in diesem Fall das Logo weggelassen werden. Zusammenfassend lassen sich folgende Lösungsvorschläge für Dokumente der obengenannten Merkmalsstruktur anführen:

Konzept: Gestaltung von großen Formularen

Lösungsvorschlag:

- Entfernen nicht notwendiger Felder
- Zusammenfassung gleichartiger Felder für die Darstellung in einzelnen Cards
- Positionierung der wichtigsten Felder an den Anfang einer Card oder eines Deck
- Übersichtsgestaltung der Eingaben mit Hilfe von Variablen
- Definition von Beschränkungen in den Feldern

Das Lösungskonzept der Überweisung besteht aus einem Deck, das sieben Cards enthält. Die ersten vier Cards bilden das eigentliche Überweisungsformular. Von dort hat man die Möglichkeit, die eingegebenen Daten abzusenden oder sich eine Übersicht über die Eingaben anzeigen zu lassen. Diese Übersicht erfolgt in der fünften Card mit Hilfe von Variablen. Es besteht dann die Möglichkeit das Formular abzusenden, es zu verändern oder zu löschen. Wird es abgesendet, erscheint eine Bestätigung und die Inhalte des Formulars werden anschließend gelöscht. Anschließend kehrt man automatisch zur ersten Card zurück. Soll das Formular abgeändert werden, kehrt man zur ersten Card zurück. Die Einträge sind weiterhin vorhanden. Die siebte Card wird aufgerufen, wenn die Einträge des Formulars gelöscht werden sollen. Während des Aufrufs werden alle Variablen, also die Einträge des Formulars, gelöscht. Dabei ist die eigentliche Card nicht zu sehen, da sie über einen Timer nur eine Zehntelsekunde angezeigt wird und dann wieder zur ersten Card verweist. Alle Cards sind über ein Template mit einem Optionsmenü ausgestattet, das es ermöglicht, zur ersten Seite zu springen, alle Eingaben zu löschen und zur vorher besuchten Card zurückzunavigieren.

Entwicklung von WML-Seiten

Nokia WAP Toolkit 1.2	WML (ue.wml#card1)
 <p data-bbox="204 600 545 663">Abbildung 65: Lösung Überweisung Card1</p>	<pre data-bbox="579 371 1190 667"> <card id="card1" title="&#xDC;berweisung"> <p align="center">Empf&#xE4;nger: <input name="empf" type="text" maxlength="27" emptyok="false"/> Empf&#xE4;nger Konto-Nr: <input name="kto empf" type="text" format="*N" maxlength="10" emptyok="false"/> Weiter </p> </card> </pre>

Bei dieser Lösung wurde bewusst auf jegliche Information verzichtet, die nicht unbedingt für den Vorgang der Überweisung notwendig ist. Der Eintrag des Eingabefeldes Empfänger wurde auf 27 Zeichen beschränkt ("maxlength='27'"). So benötigt man keinen Beschreibungstext, der auf diese Beschränkung hinweist. Ein Eintrag in dieses Feld ist optional, es kann daher auch leer gelassen werden. Das Eingabefeld der Kontonummer wurde auf zehn Zeichen beschränkt, da es derzeit keine längeren Kontonummern in Deutschland gibt und auch der Originalüberweisungsträger nur zehn Stellen für die Kontonummer vorsieht (siehe Abbildung 46). Dieses Feld darf allerdings nicht leer bleiben ("emptyok='false'").

Nokia WAP Toolkit 1.2	WML (ue.wml#card2)
 <p data-bbox="204 1487 609 1518">Abbildung 66: Lösung Überweisung Card2</p>	<pre data-bbox="636 1236 1201 1523"> <card id="card2" title="&#xDC;berweisung"> <p align="center"> BLZ: <input name="blz" type="text" format="NNNNNNNN" emptyok="false"/> Bankname wird erg&#xE4;nzt
 Weiter </p> </card> </pre>

Das Eingabefeld der zweiten Card erwartet die Eingabe der Bankleitzahl. Als Format wurde eine achtstellige Zahl festgelegt ("format='NNNNNNNN'"), da Bankleitzahlen in Deutschland achtstellige Zahlen sind.

Nokia WAP Toolkit 1.2	WML (ue.wml#card3)
	<pre> <card id="card3" title="&#xDC;berweisung"> <p align="center">W&#xE4;hrung: <select name="waeh" multiple="false" ivalue="0"> <option value="DM" onpick="ue.wmls#pf_cent('\$ (waeh) ') ">DM </option> <option value="EURO" onpick="ue.wmls#pf_cent('\$ (waeh) ') ">EURO </option> </select> Betrag \$(waeh): <input name="betr1" value="0" type="text" format="*N" emptyok="false"/> Betrag \$(pf_cent): <input name="betr2" value="00" type="text" format="NN" emptyok="false"/> Weiter </p> </card> </pre>

Abbildung 67: Lösung Überweisung Card3

Die Bestimmung der Währung, die für die Überweisung verwendet werden soll, erfolgt über ein Auswahlfeld mit zwei Optionen, DM und Euro. Je nach Auswahl der Währung, erscheint vor dem ersten Eingabefeld des Betrags die gewählte Währung ("\$(waeh)"). Dies wird mit Hilfe von Variablen realisiert. Als Eingabe sind beliebige numerische Werte zugelassen.

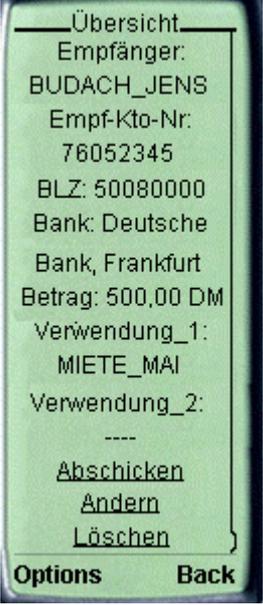
WMLScript (ue.wmls)
<pre> extern function pf_cent(variable) { var pf_cent; if(variable == "DM") { pf_cent = "Pfennig"; } else pf_cent = "Cent"; WMLBrowser.setVar("pf_cent", pf_cent); WMLBrowser.refresh(); } </pre>

Ebenso abhängig von der Auswahl der Währung, ist die Anzeige, ob es sich bei der Eingabe um einen Pfennig- oder Centbetrag handelt. Dafür wurde exemplarisch ein kleines WMLScript verwendet. Bei der Auswahl der Währung wird der Wert an das Script übergeben. Dieses liefert in Abhängigkeit der Währung den Wert "Pfennig" oder den Wert "Cent" zurück. Für die Eingabe des Pfennig- bzw. Centbetrages ist dann eine zweistellige Zahl vorgesehen. WMLScript wurde in den folgenden Lösungskonzepten nicht weiter eingesetzt, da bisher nicht gewährleistet ist, dass WMLScript von jedem verfügbaren Microbrowser korrekt interpretiert werden kann.

Entwicklung von WML-Seiten

Nokia WAP Toolkit 1.2	WML (ue.wml#card4)
 <p data-bbox="204 663 507 725">Abbildung 68: Lösung Überweisung Card4</p>	<pre data-bbox="544 371 1166 696"> <card id="card4" title="&#xDC;berweisung"> <p align="center"> Verwendungszw.1: <input name="ver1" type="text" maxlength="27" emptyok="true"/> Verwendungszw.2: <input name="ver2" type="text" maxlength="27" emptyok="true"/> &#xDC;bersicht
 Abschicken </p> </card> </pre>

Für die Eingabe des Verwendungszwecks sind jeweils Einträge mit einer Länge von 27 Zeichen möglich. Hier kann theoretisch der Abschluss der Überweisung erfolgen, indem die Angaben abgesendet werden. Der wahrscheinlichere Weg ist allerdings, dass man sich die Eingaben in einer Übersicht anzeigen lässt. Aus diesem Grund ist der Link zur Übersicht auch vor dem Hyperlink zum Abschicken des Formulars platziert.

Nokia WAP Toolkit 1.2	WML (ue.wml#card5)
 <p data-bbox="204 1771 507 1834">Abbildung 69: Lösung Überweisung Card5</p>	<pre data-bbox="544 1160 1118 1783"> <card id="card5" title="&#xDC;bersicht"> <p align="center"> Empf&#xE4;nger: \$(empf)
 Empf-Kto-Nr: \$(kto_empf)
 BLZ: \$(blz)
 Bank: Deutsche Bank, Frankfurt
 Betrag: \$(betr1),\$(betr2) \$(waeh)
 Verwendung_1: \$(ver1)
 Verwendung_2: \$(ver2)

 Abschicken
 &#xC4;ndern
 L&#xF6;schen </p> </card> </pre>

Die fünfte Card in Abbildung 69 arbeitet im Wesentlichen mit Variablen, die bei den vorhergehenden Cards bei Eingabe- und Auswahlfeldern definiert wurden. Die Variablen sind über

Entwicklung von WML-Seiten

das Attribut `name` des jeweiligen Eingabe- bzw. Auswahlfeldes initialisiert worden. Hier wurden beispielhaft Daten eingetragen, um die Funktionsweise der Variablen in WML zu demonstrieren. Allein der Bankname ist hier statisch eingetragen, da die automatische Ergänzung des Banknamens durch die Eingabe der Bankleitzahl nicht mit WML oder WMLScript zu realisieren ist, sondern den Zugriff auf eine Datenbank erfordern würde.

Nokia WAP Toolkit 1.2	WML (ue.wml#card6)
 <p>Abbildung 70: Lösung Überweisung Card6</p>	<pre> <card id="card6" title="&#xDC;berweisung" ontimer="#card1"> <onevent type="onenterforward"> <refresh> <setvar name="betr1" value=""/> [...] <setvar name="ver2" value=""/> </refresh> </onevent> <timer value="10"/> <p align="center"> Vielen Dank f&#xFC;r Ihren Auftrag! </p> </card> </pre>

Durch den Hyperlink "Abschicken" in Card 5 (Abbildung 69) gelangt man auf eine Bestätigungscard, die nach etwa einer Sekunde zur Startcard weiterleitet (`ontimer='#card1'` und `<timer value='10'/>`). Zuvor werden die Inhalte der Variablen wieder zurückgesetzt, also die Einträge der Eingabefelder gelöscht. Dies wird jeweils durch das Element `setvar` erreicht. Variableninhalte können auch mit dem Attribut `newcontext="true"` des Elements `card` gelöscht werden. Allerdings hat das den Nachteil, dass auch die Inhalte der History gelöscht werden.

Nokia WAP Toolkit 1.2	WML (ue.wml)
 <p>Abbildung 71: Lösung Überweisung Template</p>	<pre> <template> <do type="accept" label="Startcard"> <go href="#card1"/> </do> <do type="prev" label="Zur&#xFC;ck"> <prev/> </do> <do type="reset" label="Alles L&#xF6;schen"> <refresh> <setvar name="betr1" value=""/> [...] <setvar name="ver2" value=""/> </refresh> </do> </template> </pre>

Über das Template kann jede Card auf Standardnavigation und Funktionen zugreifen. Hier

Entwicklung von WML-Seiten

sind beispielhaft die Navigation zur Startseite sowie die Möglichkeit, alle Variableninhalte zu löschen implementiert.

Der gesamte Ablauf kann bis zum Absenden der Daten zum WAP-Server ohne Serverzugriff erfolgen und belastet die Netzressourcen nicht.

4.2.5 Lösungskonzept Nachrichten

Bei dem Beispiel Nachrichten handelt es sich, wie in Kapitel 3.3.2 erarbeitet, um Informationsbereiche, die relativ wenig strukturiert sind und überwiegend Text enthalten. Die Textmenge ist im Vergleich zu anderen Bereichen überdurchschnittlich groß. Zusätzlich liegt die Intensität der User Interaktion im unteren Bereich.

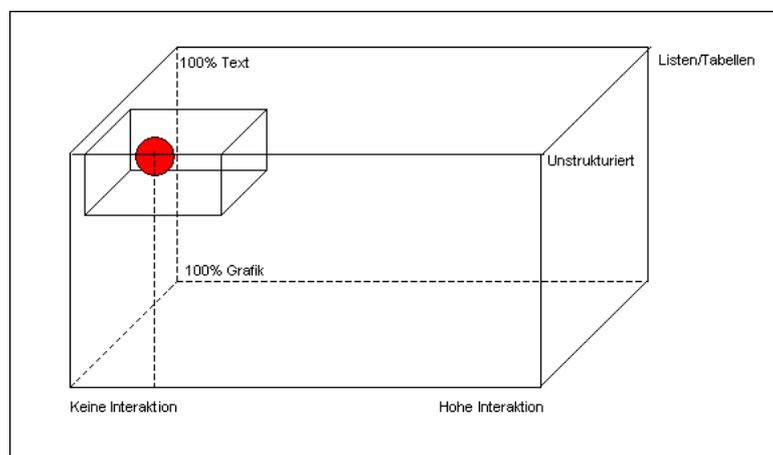


Abbildung 72: Lösungskonzept Nachrichten 3D-Raum

Abbildung 72 zeigt den Bereich, den das Lösungskonzept innerhalb des Klassifikationssystems abdecken soll.

Da der Quelltext in WML vor der Übertragung in Bytecode umgesetzt wird, ist die kompilierte Dateigröße hauptsächlich von der Textmenge abhängig, die eine WAP-Anwendung abbilden soll. In dem Überweisungsbeispiel waren die sieben Cards hauptsächlich mit WML-Elementen beschrieben, die eigentliche Textmenge blieb relativ gering. Aus diesem Grund konnte die Dateigröße bei der Kompilierung effizient reduziert werden. Die Datei im Klartext nimmt im Überweisungsbeispiel eine Größe von 4078 Bytes in Anspruch. Die kompilierte Datei ist nur noch 1187 Bytes groß. Enthalten die zu verarbeitenden Informationen viel Text, kann die Umsetzung in Bytecode nicht mehr so effizient erfolgen. Demzufolge erreicht man im Verhältnis schneller an die 1400 Bytes kompiliertes WML und muss bei der Umsetzung besonders auf die Dateigröße achten.

Entwicklung von WML-Seiten

Bei Anwendungen mit viel Text und wenig User Interaktion wird es sich in der Regel um vertikale Interaktion handeln. Das bedeutet, dass man von Grundinformationen ausgehend tiefer zu ausführlicheren Informationen navigiert. Im Journalismus gibt es das Prinzip der "invertierten Pyramide". Artikel, die nach diesem Prinzip geschrieben wurden, besitzen am Anfang alle wichtigen Informationen und Schlussfolgerungen. Im weiteren Verlauf wird dann näher auf die einzelnen Punkte eingegangen.⁸⁵ Daher ist es in diesem Zusammenhang sinnvoll, die Informationen ähnlich aufzuteilen. Der erste Teil umfasst den Grundzusammenhang in kurzer Form. Die weiteren Teile gehen ausführlicher auf die jeweiligen Informationen ein.

Aufgrund der vertikalen Navigation, kann die Anwendung schnell viele Hierarchiestufen enthalten. Es ist daher wichtig, dem Nutzer Navigationsmöglichkeiten zu geben, die ihn schnell an bekannte Punkte zurückbringen.

Der folgende Kasten fasst die erarbeiteten Lösungsvorschläge wieder grob zusammen.

Konzept: Darstellung von größeren Textmengen

Lösungsvorschlag:

- Gliederung der Informationen nach generellen und speziellen Teilen
- Vertikaler Aufbau der Navigation
- Definition von Navigationselementen für die Rückwärtsnavigation
- Besondere Beachtung der Dateigröße

Die Meldung in Kapitel 3.3.2 (Abbildung 48) enthält 3094 Zeichen und kann daher nicht komplett in einem Deck untergebracht werden. Die wenigsten Anbieter von WAP-Nachrichten bereiten die Informationen speziell für die mobile Nutzung auf. Die Nachrichten werden daher meist auf mehrere Decks verteilt. Dieses Lösungskonzept dagegen soll eine Optimierung der Nachricht für die Darstellung und Verarbeitung auf mobilen Endgeräten enthalten.

Die Nachricht kann inhaltlich in drei Teile geteilt werden, von denen zwei als eigenständige Meldungen in der WAP-Anwendungen erscheinen werden. Die Aufteilung der einzelnen

⁸⁵ Vgl. NIELSEN, Jakob (1996)

Entwicklung von WML-Seiten

Nachrichten erfolgt ebenfalls in drei Bereiche – Überschrift, Kurzform, Volltext. Die einzelnen Artikel können wie folgt aussehen.

Artikel 1	
Überschrift	Merkel zeigt sich angriffslustig
Kurzform	Die neue CDU-Vorsitzende Angela Merkel hat der rot-grünen Bundesregierung den politischen Kampf angesagt – aus einer Position neuer Stärke, die sie beschwor. "Die Stunde unserer Gegner ist vorbei ... wir sind wieder da", sagte sie. Merkel lobte anschließend das historische Werk Helmut Kohls.
Volltext	Angela Merkel betonte am Montag vor ihrer Wahl zur neuen Parteivorsitzenden, die CDU werde ihre Krise als Chance nutzen, weil sie "wieder genügend Raum gewonnen" habe für die politische Auseinandersetzung. "Es geht jetzt wieder zur Sache – nicht im Sinne von 'Schwamm drüber', sondern im Sinne von 'Es wird wieder Klartext gesprochen'." Ein versöhnliches Wort fand die bisherige Generalsekretärin für den ehemaligen Parteichef und Bundeskanzler Helmut Kohl, der erstmals seit fast 50 Jahren nicht an einem Parteitag der CDU teilnahm. Es gebe keine Alternative zur Aufklärung der Verstöße gegen das Parteiengesetz, sagte Merkel. Dennoch gelte für die Bilanz der Leistungen des langjährigen Parteichefs: "Kohls Werk, Ihr Werk, lieber Helmut Kohl, bleibt historisch überragend." Merkels Rede folgte minutenlangem starker Applaus.

Der erste Artikel ist somit in drei Textteile mit 29, 248 und 708 Zeichen aufgeteilt.

Artikel 2	
Überschrift	Schäuble fordert Anknüpfen an erfolgreiche Vergangenheit
Kurzform	Der scheidende Parteivorsitzender Schäuble hat in seiner Abschiedsrede die Delegierten des Parteitags in Essen aufgefordert, an die Erfolge der Vergangenheit anzuknüpfen. Er äußerte indirekte Kritik an Machenschaften und Intrigen in der CDU.
Volltext	Wolfgang Schäuble versicherte in seiner Abschiedsrede den knapp 1000 Delegierten: "Wir sind noch nicht über den Berg..., aber wir sind auf dem richtigen Weg." Die vergangenen anderthalb Jahre hätten bewiesen: "Niederlagen werfen die Union nicht um, und Krisen auch nicht." Von der Europapolitik über das Gesundheits- und Bildungswesen bis zur Reform des Sozialstaates und dem Aufbau einer neuen sozialen Marktwirtschaft gebe es "Aufgaben in Hülle und Fülle – die Union wird gebraucht." Seine heftige Kritik an Machenschaften und Intrigen im eigenen Lager, die in der CDU Unmut ausgelöst hatte, ließ Schäuble, in seiner Rede, nur indirekt, in einem Appell zur Loyalität, anklingen: "Der Gegensatz heißt Abhängigkeit, Seilschaften, Druck oder Intrige." Der neuen Parteiführung wünsche er "am liebsten noch mehr" Vertrauen und Unterstützung, als er selbst sie erfahren habe. Dazu wolle auch er weiter beitragen.

Der zweite Artikel ist in ebenfalls in drei Textfragmente mit 51, 210 und 773 Zeichen unterteilt.

Entwicklung von WML-Seiten

Die Umsetzung des Konzepts umfasst drei Decks. Das erste Deck enthält vier Cards, von denen die erste die Überschriften der beiden Artikel enthält und die zweite und dritte jeweils die Kurzform des Artikels. Die vierte Card enthält eine Übersicht über eine mögliche Erweiterung des Beispiels um zusätzliche Ressorts. Die beiden anderen Decks enthalten jeweils den Volltext des Artikels. In einer Übersicht kann der Verlauf folgendermaßen dargestellt werden:

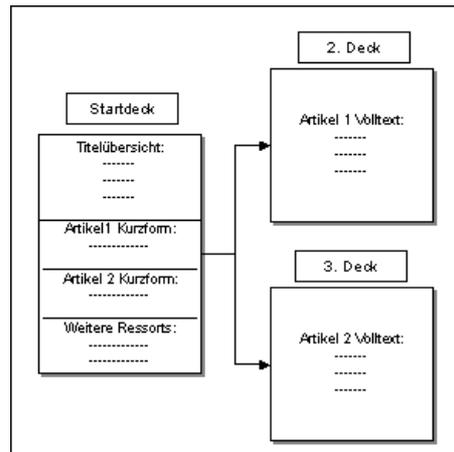
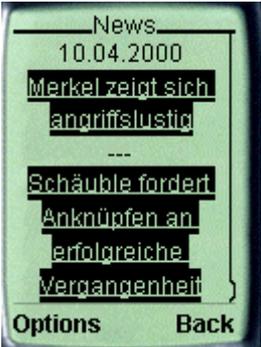


Abbildung 73: Verlauf des Lösungskonzepts Nachrichten

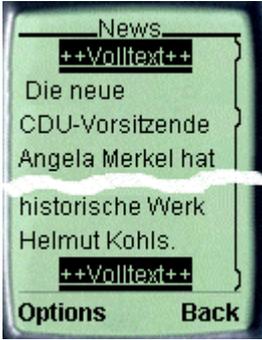
Die Umsetzung des Beispiels sieht folgendermaßen aus:

Nokia WAP Toolkit 1.2	WML (na1.wml#card1)
 <p>Abbildung 74: Lösung Nachrichten Card1</p>	<pre> <card id="card1" title="News"> <do type="accept" name="Titel"> <noop/> </do> <p align="center"> 10.04.2000
 Merkel zeigt sich angriffslustig

 Sch&#xE4;uble fordert Ankn&#xFC;pfen an erfolgreiche Vergangenheit
 </p> </card> </pre>

Über die Hyperlinks der Überschriften gelangt man zu den jeweiligen Kurzformen der Artikel. Hier wird nur ein Weg beispielhaft dargestellt, da der andere analog dazu aufgebaut ist.

Entwicklung von WML-Seiten

Nokia WAP Toolkit 1.2	WML (na1.wml#news1)
 <p data-bbox="193 712 592 741">Abbildung 75: Lösung Nachrichten Card2</p>	<pre data-bbox="616 349 1227 741"> <card id="news1" title="News"> <do type="accept" label="Volltext"> <go href="na2.wml#lang1"/> </do> <p align="center"> ++Volltext++ </p> <p mode="wrap"> Die neue CDU-Vorsitzende Angela Merkel hat [...] historische Werk Helmut Kohls. </p> <p align="center"> ++Volltext++ </p> </card> </pre>

Die Kurzform mit 248 Zeichen befindet sich im selben Deck wie die Überschriften und benötigt aus diesem Grund keinen Serverzugriff. Bei der Umsetzung wurden alle Umlaute kodiert, um Probleme bei der Darstellung durch die Microbrowser zu vermeiden. Da viele Anwender durch die Kurzform ausreichend informiert werden, ist eine schnelle und ressourcensparende Informationsversorgung möglich. Man gelangt sowohl über die Hyperlinks am Anfang und Ende der Kurzfassung sowie über das Navigationsmenü unter "Options" zum Volltext des Artikels.

Nokia WAP Toolkit 1.2	WML (na2.wml#lang1)
 <p data-bbox="193 1617 523 1677">Abbildung 76: Lösung Nachrichten Deck2_Card1</p>	<pre data-bbox="539 1276 1227 1601"> <card id="lang1" title="Volltext"> <p align="center"> ++News-Titel++ </p> <p mode="wrap"> Angela Merkel betonte am Montag vor ihrer Wahl zur [...] minutenlangem starker Applaus. </p> <p align="center"> ++News-Titel++ </p> </card> </pre>

Der Volltext ist, wie beschrieben, jeweils in einem separaten Deck untergebracht. Die Rückwärtsnavigation zu den Nachrichtentitel erfolgt entweder über die Hyperlinks am Anfang und Ende des jeweiligen Volltextes oder über das Optionsmenü des Microbrowsers. Damit ist gewährleistet, dass der Nutzer möglichst einfach zu den Ausgangspositionen zurückgelangt.

Nokia WAP Toolkit 1.2	WML (na1.wml#card_ue)
 <p>Abbildung 77: Lösung Nachrichten Card4</p>	<pre> <card id="card_ue" title="&#xDC;bersicht"> <do type="accept" name="Uebersicht"> <noop/> </do> <do type="accept" name="Titel"> <noop/> </do> <p align="center"> News
 Wirtschaft
 Sport </p> </card> </pre>

Die vierte Card des ersten Deck ist über das Optionsmenü von jeder Card innerhalb des Beispiels erreichbar. Sie zeigt eine Möglichkeit, weitere Ressorts oder Bereiche in die Anwendung einzugliedern und sie damit auszubauen.

Nokia WAP Toolkit 1.2	WML (na1.wml)
 <p>Abbildung 78: Lösung Nachrichten Template</p>	<pre> <template> <do type="accept" name="Titel" label="News-Titel"> <go href="#card1"/> </do> <do type="accept" name="Uebersicht" label="&#xDC;bersicht"> <go href="#card_ue"/> </do> <do type="prev" label="Zur&#xFC;ck"> <prev/> </do> </template> </pre>

Um die Navigation zu erleichtern, wurde ein Template definiert, das über alle Cards hinweg erreichbar ist. Zu den Standardeinträgen, wie `News-Titel` und `Übersicht`, erscheint zusätzlich der Eintrag `Volltext`, wenn man sich in der Card mit der Kurzfassung eines Artikels befindet.

Deck eins hat unkompiliert eine Größe von 2432 und kompiliert eine Größe von 1168 Bytes. Man kann erkennen, dass die Komprimierung der Datei nicht so effizient gestaltet werden kann, wie das im Beispiel der Überweisung der Fall ist. Das zweite Deck hat eine Größe von 1640 Bytes im unkompilierten und 1047 Bytes im kompilierten Zustand. Das Verhältnis bei Deck drei von unkompilierter zu kompilierter Dateigröße beträgt 1737 Bytes zu 1134 Bytes.

Entwicklung von WML-Seiten

4.2.6 Lösungskonzept Zugfahrplan

In diesem Fall geht es darum, ein Lösungskonzept für Dokumente zu erarbeiten, die ähnlich aufgebaut und strukturiert sind, wie der Zugfahrplan in Abbildung 50.

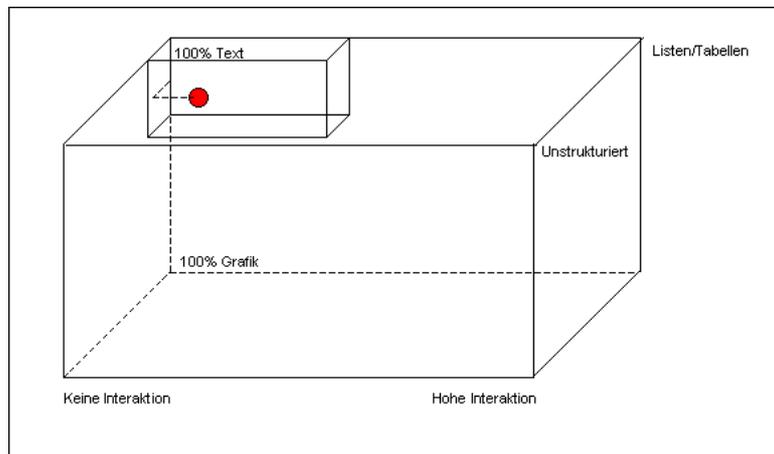


Abbildung 79: Lösungskonzept Zugfahrplan 3D-Raum

Der Bereich, den dieses Lösungskonzept einschließt, ist in Abbildung 79 dargestellt. Generell handelt es sich um Dokumente, die überwiegend Text enthalten, die in Form von Tabellen und Listen stark strukturiert sind und wenig bis mittelstarke Interaktion aufweisen.

WML bietet die Möglichkeit mit Tabellen zu arbeiten. Allerdings unterstützen noch nicht alle verfügbaren Endgeräte die Tabellen in vollem Umfang (Vgl. Abbildung 27). Nach eigenen Erfahrungen sollten Tabellen, je nach Inhalt, nicht mehr als vier Spalten aufweisen, damit sie in einer Reihe angezeigt werden können. Informationen, die in tiefen Tabellen, also Tabellen mit hoher Zeilenanzahl, organisiert sind, sollten in kleinere eigenständige Tabellen aufgeteilt werden. Damit wird gewährleistet, dass die Informationen übersichtlich auf dem Display des mobilen Endgerätes dargestellt werden können.

Ähnlich kann bei Informationen in langen Listen vorgegangen werden. Durch eine Aufgliederung der Listen in inhaltlich eigenständige Teile kann eine Informationsmenge erreicht werden, die für die Darstellung in WML geeignet ist. Allerdings wird mit jeder Untergliederung eine zusätzliche Navigationsebene in die Anwendung gebaut. Daher sollten die Kerninformationen aus der Liste schnell und einfach erreichbar sein. Die restlichen Informationen werden in weiteren Listen zusammengefasst und können als Zusatzinformationen angeboten werden, die dann über weitere Navigationsebenen gefunden werden.

Handelt es sich bei den Grafiken des Dokuments um Zusatzinformationen, die erläuternden Text einsparen können, sollten die Grafiken angemessen aufbereitet übernommen werden.

Entwicklung von WML-Seiten

Der folgende Merkkasten stellt wiederum die Lösungsvorschläge in einer Übersicht zusammen.

Konzept: Umsetzung von tiefen Tabellen/Listen in WML

Lösungsvorschlag:

- Aufteilung der Tabelle/Liste in weitere eigenständige Tabellen/Listen
- Maximal 4 Spalten pro Tabelle
- Darstellung der Kerninformationen in den oberen Ebenen
- Darstellung von Zusatzinformationen in tiefer liegenden Ebenen
- Übernahme von aufbereiteten Grafiken mit Zusatzinformationen
- Navigation über Auswahllisten

In diesem Beispiel wird die Tabelle mit den Abfahrtszeiten in einzelne, eigenständige Tabellen unterteilt. Auf der Spaltenebene werden die drei Spalten Montag-Freitag, Samstag und Sonn-/Feiertag getrennt. Weiterhin erfolgt eine Unterteilung nach den Uhrzeiten in 5-9, 9-13, 13-17, 17-21, 21-0, so dass 15 eigenständige Tabellen entstehen (siehe Abbildung 80).

Linie	START	ZIEL
S4	Kronberg	Frankfurt Hbf → Südbahnhof → Langen Bf → Darmstadt Hbf

Gültigkeit: Montag - Freitag, Samstag, Sonn-/Feiertag

Uhrzeit	Montag - Freitag	Samstag	Sonn-/Feiertag
5 02 _c 32 _c	5 02 _c 32 _c	5 02 _c 32 _c	5 47
6 02 _c 32 _c	6 02 _c 32 _c	6 02 _c 32 _c	6 47
7 02 _c 32 _c	7 02 _c 32 _c	7 02 _c 32 _c	7 47
8 02 _c 32 _c	8 02 _c 32 _c	8 02 _c 32 _c	8 47
9 02 _c 32 _c	9 02 _c 32 _c	9 02 _c 32 _c	9 47
10 02 _c 32 _c	10 02 _c 32 _c	10 02 _c 32 _c	10 47
11 02 _c 32 _c	11 02 _c 32 _c	11 02 _c 32 _c	11 47
12 02 _c 32 _c	12 02 _c 32 _c	12 02 _c 32 _c	12 47
13 02 _c 32 _c	13 02 _c 32 _c	13 02 _c 32 _c	13 47
14 02 _c 32 _c	14 02 _c 32 _c	14 02 _c 32 _c	14 47
15 02 _c 32 _c	15 02 _c 32 _c	15 02 _c 32 _c	15 47
16 02 _c 32 _c	16 02 _c 32 _c	16 02 _c 32 _c	16 47
17 02 _c 32 _c	17 02 _c 32 _c	17 02 _c 32 _c	17 47
18 02 _c 32 _c	18 02 _c 32 _c	18 02 _c 32 _c	18 47
19 02 _c 32 _c	19 02 _a 47	19 47	19 47
20 02 _c 32 _c	20 47	20 47	20 47
21 02 _c 32 _a	21 47	21 47	21 47
22 02 _c 32 _c	22 47	22 47	22 47
23 02 _a 47 _c	23 47 _c	23 47 _c	23 47 _c
0 47 _c	0 47 _c	0 47 _c	0 47 _c

Erläuterung: B bis Filz Südbahnhof, C bis Langen (Hess)

Abbildung 80: Lösung Fahrplan: Aufteilung des Dokuments

Zusatzinformation stellt die Liste der Daten des Fahrtverlaufes dar, welche die Fahrtzeit zu den einzelnen Stationen enthält. Diese wird als Menüpunkt über das Template angeboten werden. Die Liste wird in zwei Teile untergliedert, innerstädtische Haltestellen und Haltestellen außerhalb Frankfurts.

Entwicklung von WML-Seiten

Die Grafiken konnten in diesem Fall nicht adäquat aufbereitet werden, um sie in die WAP-Anwendung zu übernehmen.

Die Umsetzung des Fahrplans erfolgte mit Hilfe von vier Decks. Das erste Deck enthält sechs Cards, wobei die erste Card neben der Auswahlmöglichkeit des Wochentags und der Uhrzeit auch die Zuglinie, den Standort und die Gültigkeitsdauer umfasst. Die fünf weiteren Cards nehmen jeweils eine Tabelle der Spalte Montag-Freitag auf (siehe Abbildung 80). Das zweite Deck sollte zunächst die weiteren Tabellen der Spalte Samstag und Sonn-/Feiertag aufnehmen. Allerdings konnten die Daten nicht so aufbereitet werden, dass sie innerhalb der 1400 Bytes blieben. Die Tabellen der zwei Spalten sind nun in separaten Decks untergebracht und umfassen jeweils fünf Cards. Das vierte Deck enthält die Zusatzinformationen Fahrtverlauf und die Kontaktinformationen der Deutschen Bahn. Schließlich sorgt ein Template in jedem Deck für die Navigationsmöglichkeiten. Zur Verdeutlichung der Aufteilung in Cards und Decks soll Abbildung 81 dienen.

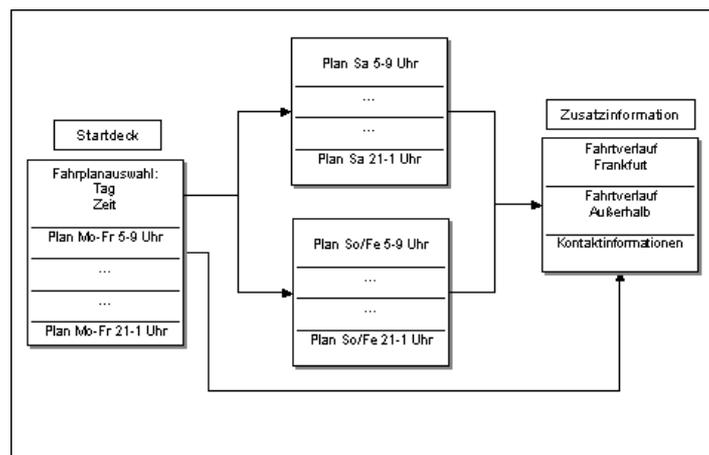
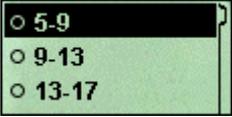


Abbildung 81: Verlauf des Lösungskonzepts Fahrplan

Die einzelnen Schritte der Umsetzung werden auf den folgenden Seiten erläutert und dargestellt.

Nokia WAP Toolkit 1.2	WML (fpM.wml#card1)
 <p>Abbildung 82: Lösung Fahrplan Deck1 Card1</p>  <p>Abbildung 83: Lösung Fahrplan Auswahl1</p>  <p>Abbildung 84: Lösung Fahrplan Auswahl2</p>	<pre> <card id="card1" title="S4 - FFM Hbf" newcontext="true"> <p align="center">S4:
Kronberg-DA Hbf
Standort: Ffm Hbf </p> <p>Tag: <select name="t" iname="i" ivalue="0" multiple="false"> <option value="M">Montag-Freitag</option> <option value="S">Samstag</option> <option value="SF">Sonn-/Feiertag</option> </select>Uhrzeit: <select name="z" iname="u" ivalue="0" multiple="false"> <option value="5-9" onpick="fp\$(t).wml#\$(t)\$ (z)">5-9</option> <option value="9-13" onpick="fp\$(t).wml#\$(t)\$ (z)">9-13</option> <option value="13-17" onpick="fp\$(t).wml#\$(t)\$ (z)">13-17</option> <option value="17-21" onpick="fp\$(t).wml#\$(t)\$ (z)">17-21</option> <option value="21-1" onpick="fp\$(t).wml#\$(t)\$ (z)">21-1</option> </select> </p> <p align="center">G&#xFC;ltig:
 26.09.99 - 27.05.00 </p> </card> </pre>

Mit Hilfe von Variablen kann hier eine Card für alle Auswahlmöglichkeiten definiert werden. Je nach getroffener Wahl wird die aufzurufende URL modifiziert. Die Kombination der beiden Selektionen führt dann direkt zu der Card, welche die Fahrplandaten enthält (`onpick='fp$(t).wml#$(t)$ (z)'`).

Entwicklung von WML-Seiten

Nokia WAP Toolkit 1.2	WML (fpM.wml#M5-9)
 <p data-bbox="204 790 507 853">Abbildung 85: Lösung Fahrplan Deck1 Card2</p>	<pre data-bbox="539 344 1204 882"> <card id="M5-9" title="Mo-Fr_5-9 Uhr"> <p align="center"> Fr&#xFC;her
 <table columns="3"> <tr> <td>5 </td> <td>02*</td> <td>32*</td> </tr> [...] <tr> <td>8 </td> <td>02*</td> <td>32*</td> </tr> </table>
 Sp&#xE4;ter </p> </card> </pre>

Beispielhaft für die Daten aus der Abfahrtszeittabelle ist in Abbildung 85 die Auswahl Montag-Freitag und 5-9 Uhr dargestellt. Über die Hyperlinks **Früher** und **Später** gelangt man direkt zu früheren oder späteren Abfahrtszeiten. Nach dem gleichen Muster sind die übrigen Tabellen der Abfahrtszeiten aufgebaut.

Nokia WAP Toolkit 1.2	WML (fpV.wml#card1)
 <p data-bbox="204 1480 639 1507">Abbildung 86: Lösung Fahrplan Deck4 Card1</p>	<pre data-bbox="675 1240 1204 1507"> <card id="card1" title="Verlauf"> <p align="center"> Haltestellen ab Ffm Hbf in:
 + Frankfurt
 + Au&#xDF;erhalb </p> </card> </pre>

Die Zusatzinformationen sind in einem separaten Deck untergebracht, um sie getrennt von den eigentlichen Informationen zu halten. Über das Template sind sie aus jedem Deck verfügbar. Der Streckenverlauf wurde zur besseren Übersicht in zwei Bereiche gegliedert.

Nokia WAP Toolkit 1.2	WML (fpV.wml#ffm)
 <p data-bbox="201 667 635 694">Abbildung 87: Lösung Fahrplan Deck4 Card2</p>	<pre data-bbox="657 344 1198 739"> <card id="ffm" title="Verlauf in Ffm"> <p mode="wrap"> Haltestelle (Min.)
 Taunusanlage (02)
 Hauptwache (03) [...] Stresem.-allee (12)
 Ffm Louisa (14)
Verlauf au&#xDF;erhalb </p> </card> </pre>

Abbildung 87 zeigt die Liste der Haltestellen innerhalb Frankfurts und die ungefähre Fahrtzeit bis dorthin. Über den Link *Verlauf außerhalb* kann die dritte Card des Deck aufgerufen werden, in der der zweite Teil des Streckenverlaufs abgebildet ist.

Die vierte Card schließlich enthält die Kontaktinformationen, die ebenfalls über das Template erreichbar sind (Abbildung 88).

Nokia WAP Toolkit 1.2	WML (fpV.wml#kon)
 <p data-bbox="204 1384 638 1411">Abbildung 88: Lösung Fahrplan Deck4 Card4</p>	<pre data-bbox="657 1149 1198 1384"> <card id="kon" title="Kontakt"> <p align="center"> DB Regio
Deutsche Bahn [...] (01805) 996633 </p> </card> </pre>

4.2.7 Lösungskonzept Portal

Das folgende Lösungskonzept soll zeigen, wie Dokumente in WML übertragen werden können, die überwiegend aus einzelnen untereinander verlinkten Textbausteinen bestehen. Die Strukturierung zeigt sich in diesem Fall durch kleine Listen und Tabellen, wobei auch weniger strukturierte Informationen denkbar sind, da die Anordnung der Informationen oft nur dem Layout dient. Die Intensität der User Interaktion ist durch die vielen verknüpften Textteile hoch. Folgende Abbildung zeigt den Bereich, den das Lösungskonzept annähernd umfassen soll.

Entwicklung von WML-Seiten

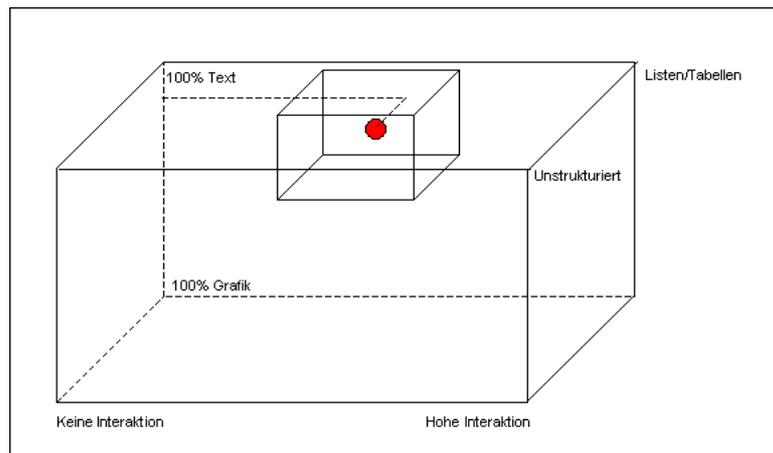


Abbildung 89: Lösungskonzept Portal 3D-Raum

Dokumente, die derartig strukturiert sind, lassen sich in WML am besten in Listen darstellen. Wie schon in Kapitel 4.2.1.2 gezeigt, kann die Suche über Datenmengen unterschiedlich erfolgen. Für dieses Beispiel wird die Verwendung der hierarchisch aufgebauten Listen gewählt. Damit dem Nutzer die Navigation nochmals erleichtert wird, sollten die beliebtesten Einträge der Liste am Anfang der Seite zu finden sein. Die Rangliste bei mobilen Anwendungen unterscheidet sich höchstwahrscheinlich von ihren Pendanten im WWW. Die Reihenfolge bei dieser Umsetzung wurde aufgrund des Umfrageergebnisses in Abbildung 41 gewählt. Zusätzlich optimiert werden können die Listen durch eine Neuformulierung der Einträge. Diese Neuformulierung kann einmal inhaltlich, zugeschnitten auf die Bedürfnisse des mobilen Bereichs geschehen, und einmal formal, indem kürzere Synonyme für die zugehörigen Einträge gefunden werden. Da es sich hier um eine beispielhafte Umsetzung handelt, wird von dieser Neuformulierung abgesehen.

Konzept: Optimierte Darstellung von kleinen Listen

Lösungsvorschlag:

- Einträge nach der Häufigkeit des Abrufs sortiert darstellen
- Umformulierung auf kürzere und prägnantere Einträge
- Hierarchisch strukturierte Listen, siehe Kapitel 4.2.1.2

In diesem Fall steht nicht im Vordergrund, alle Informationsteile des Portals aus Abbildung 52 in WML zu überführen, es sollen nur die Schlüsselbereiche definiert und umgesetzt werden. Die Kernbereiche stellen in diesem Fall die Freitextsuche und die eigentliche Katalogübersicht dar.

Entwicklung von WML-Seiten

Bei der Realisierung werden zwei Decks verwendet, um die vorhandenen Daten in WML darzustellen. Das erste Deck enthält die erste Ebene des Katalogs und eine Suchfunktion über den Kataloginhalt. Zusätzlich ist eine beispielhafte Antwortcard zu einer Suche implementiert, um zu zeigen, wie ein Suchergebnis aufbereitet werden kann. Um den Platz in den Decks vollständig auszunutzen, wird ein Teil der nächsten Ebene des Katalogs ebenfalls im ersten WML-Deck untergebracht. Die restlichen Teile des Katalogs sind über die Cards des zweiten Deck verteilt dargestellt. Eine Übersicht zeigt die Abbildung 90.

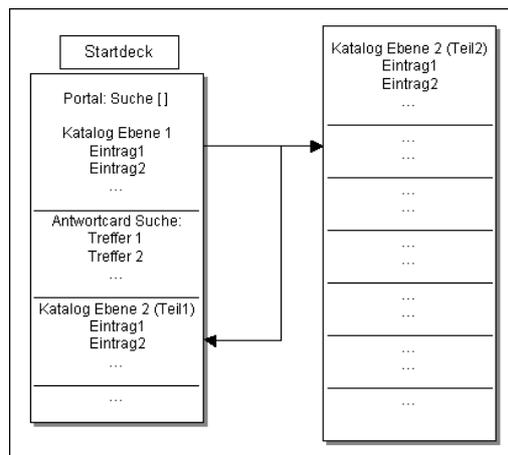


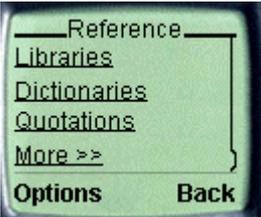
Abbildung 90: Verlauf des Lösungskonzepts Portal

Der Aufbau der einzelnen Cards und Decks wird detailliert in den folgenden Kästen beschrieben und erläutert.

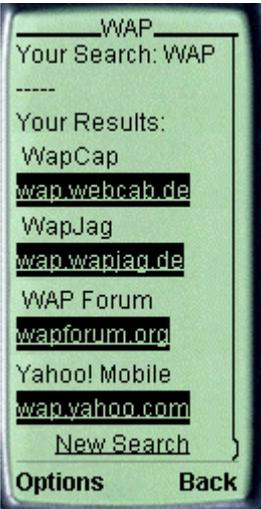
Nokia WAP Toolkit 1.2	WML (p1.wml#card1)
 <p data-bbox="220 1877 491 1937">Abbildung 91: Lösung Portal Deck1 Card1</p>	<pre data-bbox="539 1361 1216 1951"> <card id="card1" title="WAP-Catalog"> <p align="center"> Search: <input name="such" type="text" format="*M" emptyok="true"/> Start Search </p> <p mode="wrap"> -----
 Business&amp;Economy
 [...] Social Science
 Society&amp;Culture
----- </p> <p align="center"> Search: <input name="such" type="text" format="*M" emptyok="true"/> Start Search </p> </card> </pre>

Entwicklung von WML-Seiten

Zusätzlich zu der ersten Ebene des Katalogs sind am Anfang und Ende der Card Suchfunktionen implementiert. Dies soll einen einfachen Zugriff auf die Informationen ermöglichen. Die weiteren Ebenen des Katalogs sind über die Textlinks der Kategorien zu erreichen.

Nokia WAP Toolkit 1.2	WML (p1.wml#ref)
 <p data-bbox="220 786 491 853">Abbildung 92: Lösung Portal Deck2 Card2</p>	<pre data-bbox="539 568 1198 864"> <card id="Ref" title="Reference"> <p mode="wrap"> Libraries
 Dictionaries
 Quotations
 More &gt;&gt; </p> </card> </pre>

Die Card in Abbildung 92 zeigt beispielhaft, wie die zweite Ebene des Katalogs in WML aufgebaut sein kann.

Nokia WAP Toolkit 1.2	WML (p1.wml#search)
 <p data-bbox="220 1646 491 1713">Abbildung 93: Lösung Portal Suchbeispiel</p>	<pre data-bbox="539 1137 1134 1727"> <card id="Search" title="\$(such)"> <p mode="wrap"> Your Search: \$(such)

 Your Results:
 WapCap
 wap.webcab.de
WapJag
 wap.wapjag.de
WAP Forum
 wapforum.org
Yahoo! Mobile
 wap.yahoo.com </p> <p align="center"> New Search </p> </card> </pre>

Abbildung 93 zeigt ein Beispiel, wie ein Suchergebnis bei einem WAP-Portal gestaltet werden kann. Dabei wird mit Hilfe der Variablen der eingegebene Suchtext am Anfang der Card eingeblendet. Danach folgt der Titel der gefundenen Seite und der Link zur Seite.

4.2.8 Lösungskonzept Aktienkurse

Das letzte Lösungskonzept, das in diesem Rahmen erarbeitet wird, basiert auf der Merkmalskombination des Beispiels Aktienkurse aus Kapitel 3.3.5. Aus Abbildung 94 wird ersichtlich, auf welchen Raum des Klassifikationssystems, das Lösungskonzept angewendet werden kann.

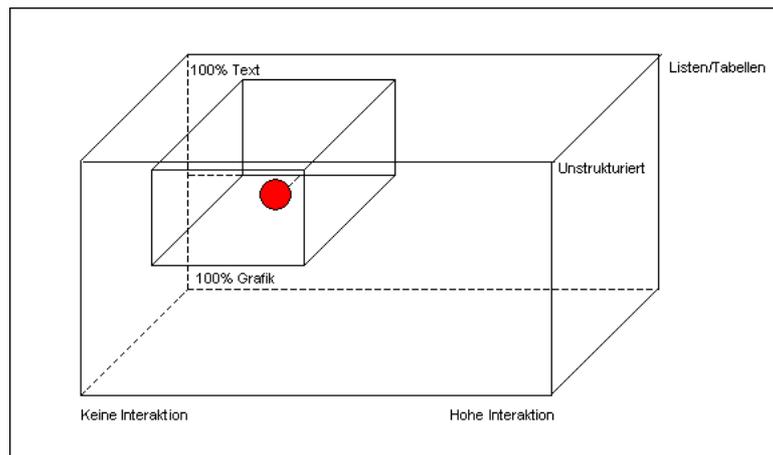


Abbildung 94: Lösungskonzept Aktienkurse 3D-Raum

Der Unterschied zu den vorangegangenen Beispiellösungen liegt beim Verhältnis von Text zu Grafik. Lag das Verhältnis bisher eindeutig zugunsten von Textelementen, ist das Verhältnis in diesem Fall nahezu ausgeglichen. Wie in Kapitel 4.2.3 dargestellt, liegt die obere Grenze des Grafikanteils von Dokumenten, die in WML übertragen werden können, bei etwa 50 Prozent. Damit ergeben sich hier auch die Maßnahmen, die in Kapitel 4.2.3 für Grafiken erarbeitet wurden.

Das hier vorliegende Dokument ist in Form einer breiten Tabelle strukturiert, die sich durch lange Zeilen auszeichnet. Aus diesem Grund wird die Tabelle auch bei der Umsetzung in WML nicht in dieser Form übernommen werden. Es wird der Vorschlag gemacht, aus den Zeilen der Tabelle eigenständige Informationseinheiten zu bilden, die dann als einzelne Cards dargestellt werden können. Daher kann dieses Lösungskonzept auch für Informationen angewendet werden, die nicht oder weniger strukturiert sind, sonst aber ähnliche Merkmale aufweisen. Zum Beispiel ließe sich eine große Liste mit ebenso langen Einträgen ähnlich aufbereiten, wie es für die Tabelle vorgeschlagen wird. Die eigentliche Navigation über die Inhalte der Tabellen oder Listen kann wiederum durch aufbereitete Listen erfolgen.

Entwicklung von WML-Seiten

Konzept: Verarbeitung von breiten Tabellen/Listen

Lösungsvorschlag:

- Darstellung der Zeilen bzw. Listeneinträge als einzelne Informationseinheiten
- Navigation über die Inhalte durch Listen

Das Beispiel der Aktienkurse in WML stützt sich, wie in Abbildung 95 dargestellt, auf fünf WML-Decks.

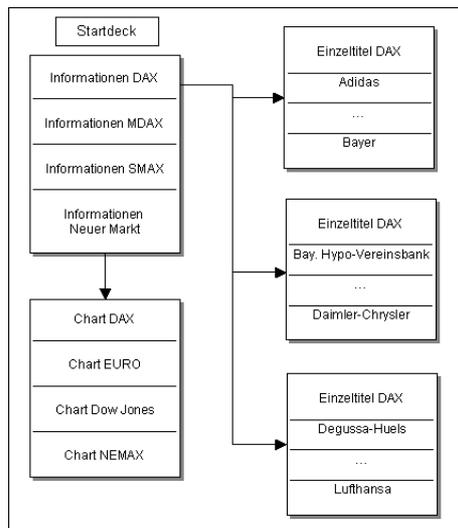


Abbildung 95: Verlauf des Lösungskonzepts Aktien

Zunächst wurden die Informationen aus der Tabelle in Abbildung 54 aufgeteilt und für jede Reihe der Tabelle eine Card erstellt. Diese Cards verteilen sich auf insgesamt vier Decks. Die kleinen Grafiken, die die Hyperlinks zu Hintergrund- und Detailinformationen repräsentieren, wurden durch Textlinks ersetzt, da ihre Bedeutung selbst in der Webanwendung nur durch eine zusätzliche Legende ersichtlich wird.

Die Chartgrafiken sind in separaten Decks untergebracht und können jeweils durch einen Hyperlink auf der entsprechenden Seite aufgerufen werden. Die in der Webanwendung vorhandenen Grafiken konnten nicht übernommen oder so optimiert werden, dass eine zufriedenere Darstellung innerhalb einer WAP-Anwendung möglich war. Dieser Umstand machte eine Neugestaltung der Grafiken notwendig.

Die Navigation über Aktionen geht in diesem Beispiel über die Standardnavigationselemente hinaus. Die Templates der zugehörigen Decks wurden mit zusätzlichen Verweisen zu den einzelnen Aktien-Indices versehen.

In einer realen Anwendung, stehen diese Informationen nicht auf statischen Cards, sondern werden aus Datenbanken dynamisch generiert.

Nokia WAP Toolkit 1.2	WML (a1.wml#xet)
<div data-bbox="240 443 507 817"> </div> <p data-bbox="204 842 544 902">Abbildung 96: Lösung Aktien Deck1 Card1</p> <div data-bbox="240 943 507 1128"> </div> <p data-bbox="204 1153 544 1214">Abbildung 97: Lösung Aktien Card1 Auswahl</p>	<pre data-bbox="576 443 1235 1088"> <card id="xet" title="DAX-Xetra_11.05." newcontext="true"> <p align="center"> DAX-Xetra 14:48 7.196,74
 Diff.(%):+1,07
 Volumen: 1,68 Mrd.
 Chart Xetra-Einzelkurse: <select name="kurs" value=" " iname="i" ivalue="0" multiple="false"> <option value="adid" onpick="a2.wml#\$(kurs)"> Adidas-Salomon</option> <option value="alli" onpick="a2.wml#\$(kurs)"> Allianz</option> [...] <option value="luft" onpick="a4.wml#\$(kurs)"> Lufthansa</option> </select> </p> </card> </pre>

Zusätzlich zu den Aktien-Indices, lassen sich zum jeweiligen Index Einzelkurse abrufen. Das geschieht mit Hilfe eines Auswahlmenüs, in dem beispielhaft einzelne DAX-Titel ausgewählt werden können. Bei Anwahl einer Optionsmöglichkeit wird die gewählte Seite direkt geladen.

Wie bereits erwähnt, sollten wichtige Elemente stets am Anfang einer Card zu finden sein. Obwohl die Einzelkursauswahl hier eine hohe Priorität besitzt, ist das Element am Ende der Card angeordnet. Dieser Umstand ergibt sich aus der Tatsache, dass manche Microbrowser Auswahllisten direkt in die Card implementieren. Das bedeutet, dass alle Elemente, die später in der Card definiert sind, zunächst nicht sichtbar sind. Erst wenn eine Auswahl erfolgt ist, sind die weiteren Elemente erreichbar. Sind diese Auswahllisten aber so gestaltet, dass bei Auswahl einer Option direkt zu einer anderen Card gesprungen wird – wie hier der Fall – bleiben die übrigen Elemente verborgen.

Entwicklung von WML-Seiten

Nokia WAP Toolkit 1.2	WML (a2.wml#baye)
 <p data-bbox="196 734 517 797">Abbildung 98: Lösung Aktien Einzelkurs</p>	<pre data-bbox="539 349 1243 828"> <card id="baye" title="11.05."> <p align="center"> Bayer 14:33 43,35 43,35
 Diff. (%) :-0,34
 Volumen: 42,80 Mio.
 Chart
 News: Bayer
 Kursdetails
 Firmenprofil </p> </card> </pre>

Exemplarisch für alle Einzelkurse, ist in Abbildung 98 eine Möglichkeit dargestellt, wie die Informationen einer Reihe der Tabelle aus Abbildung 54 in WML übertragen werden können. Dabei entsprechen die Textlinks am Ende der Card den Grafiken in der Webanwendung, die auf tiefer gehende Informationen verweisen. Da die Links auf jeder Einzelkurs-Card vorhanden sind, hätten sie auch im Template definiert werden können. Davon wurde hier allerdings abgesehen, da sonst die Übersicht des Optionsmenüs gelitten hätte.

Nokia WAP Toolkit 1.2	WML (chart.wml#xet)
 <p data-bbox="209 1512 544 1541">Abbildung 99: Lösung Aktien Chart</p>	<pre data-bbox="579 1339 1190 1473"> <card id="xet" title="DAX_3Monate"> <p align="center"> </p> </card> </pre>

Für die Grafiken wurden nur die essenziellen Informationen verwendet. In diesem Fall wurde nur die obere und untere Grenze der jeweiligen Achsenabschnitte übernommen. Der Verlauf der Kurve ist trotz der kleinen Grafik relativ deutlich zu erkennen. Den Zweck, eine Übersicht über den Verlauf eines Kurses anzubieten, kann diese Grafik auf jeden Fall erfüllen.

5 Realisierung von WML-Seiten mit XSL(T)

Mit WML wurde wieder eine neue Markup-Sprache für das Internet entwickelt. Für viele Webseiten-Entwickler bedeutet dies zusätzlichen Lernaufwand, wenn sie Internetinhalte auch für mobile Endgeräte verfügbar machen wollen. WML ist, wie bereits in Kapitel 2.1.2 dargestellt, eine Anwendung von XML. XML steht nun mit der Extensible Stylesheet Language (XSL) eine Formatierungssprache zur Seite, mit der es möglich ist, XML-Daten, sowohl in WML als auch in HTML auszugeben.

In den folgenden Abschnitten wird gezeigt, wie der kombinierte Einsatz von XML und XSL aussehen kann. Ziel dabei ist es, aus einer beispielhaften XML-Quelle mit Hilfe von XSL, WML-Dateien zu erzeugen, die dann von jedem Microbrowser angezeigt werden können.

5.1 XSL und XSLT

XSL ist die Transformierungs- und Formatierungssprache für XML. Mit Hilfe von XSL lassen sich XML-Dokumente in beliebige andere XML-Dokumente, wie WML, umwandeln. Eine Umwandlung in HTML ist ebenfalls möglich.⁸⁶ Die beiden ersten Bestandteile von XSL, XSL Transformations (XSLT) und XML Path Language (XPath), sind am 16.11.1999 vom W3C als "Recommendations" verabschiedet worden.⁸⁷ XSLT hat dabei die Aufgabe aus einem XML-Dokument, bzw. aus dessen Baumstruktur, eine neue Baumstruktur zu generieren, die dann ein neues XML-Dokument repräsentiert. Um welche Baumstruktur es sich im Ergebnis handelt, legen die Anweisungen in der jeweiligen Stylesheet-Datei fest. Bei der folgenden Vorgehensweise wird ausschließlich XSLT verwendet.

Ein entscheidender Vorteil der Kombination von XML und XSL ist die Trennung von Inhalt und Layout bzw. Formatierung. Somit hat man mit einer XML-Quelle eine Datenquelle, die noch keinerlei Angaben über die Präsentation dieser Daten enthält. Mit Hilfe eines XSL-Stylesheets lassen sich Formatierungsanweisungen festlegen. Diese können wie eine Schablone über die XML-Daten gelegt werden und erzeugen je nach Anforderung unterschiedliche Ansichten auf die XML-Daten. So lassen sich beispielsweise XML-Dateien einmal für Webbrowser und einmal für Microbrowser aufbereiten. Dafür sind dann nur zwei

⁸⁶ Vgl. BEHME, Henning (1999a), S.181

⁸⁷ Vgl. ADLER, Sharon [u.a.] (2000)

Realisierung von WML-Seiten mit XSL(T)

Stylesheets mit Formatierungsanweisungen notwendig. Auf diese Weise entfällt die redundante Haltung der Inhalte für zwei Varianten. Stellt man sich jetzt vor, dass für die optimale Darstellung der Inhalte, für jedes Ausgabemedium und dessen Software, hier die verschiedensten Browser, eine angepasste Version erzeugt werden muss, sieht man schnell die Vorteile der oben beschriebenen Vorgehensweise.

5.2 Beispiel Online-Auktion

Als Beispiel wurde eine Online-Auktion gewählt, da sie real in WML umgesetzt werden kann und eine gute Möglichkeit bietet, verschiedene Sichten auf eine XML-Quelle zu erzeugen.

5.2.1 Entwicklung der XML-Quelle

Folgende Merkmale und Eigenschaften zeichnen eine etwas vereinfachte Online-Auktion aus. Eine Online-Auktion umfasst verschiedene Auktionsobjekte, die in unterschiedliche Kategorien eingeordnet sind. Ein Auktionsobjekt besitzt dabei eine eindeutige Nummer und einen Titel. Weiterhin verfügt es über Verkäuferangaben, eine Objektbeschreibung, ein Startgebot und Enddatum und -uhrzeit.

Auszug aus dem XML-Dokument Online-Auktion (auktion.xml)

```
<auktion>
  <objekt id="1">
    <kategorie>Buch</kategorie>
    <titel>Die Paepstin</titel>
    <anbieter nr="1234">
      <vorname>Jana</vorname>
      <nachname>Mueller</nachname>
    </anbieter>
    <beschreibung>Tolles Buch ueber ...</beschreibung>
    <startgebot>1</startgebot>
    <enddatum int="20000615">15.06.2000</enddatum>
    <endzeit int="1400">14:00</endzeit>
    <gebote>
      <gebot nr="1">
        <preis>2</preis>
        <bieter nr="4321">Karl Napp</bieter>
      </gebot>
      <!--weitere Gebote-->
    </gebote>
  </objekt>
  <!--weitere Objekte-->
</auktion>
```

Das Wurzelement bildet das Element `auktion`, das die restlichen Elemente einschließt. Jedes Auktionsobjekt startet mit `<objekt id=" " >` und endet mit dem schließenden Tag `</objekt>`. Die in dem Objekt enthaltenen Elemente sind ebenso aufgebaut. Die einzelnen Gebote für die Auktionsobjekte werden jeweils mit Preis und Bieter an das Ende Gebotsliste

Realisierung von WML-Seiten mit XSL(T)

angehängt. Die einzelnen Gebote sind durch eine fortlaufende Nummer eindeutig identifizierbar.

Bei der hier entwickelten XML-Quelle, handelt es sich um ein sogenanntes wohlgeformtes XML-Dokument. Das bedeutet, dass sich das Dokument, an die Regeln der XML-Spezifikation hält.⁸⁸ Auf die Verwendung einer Document Type Definition (DTD) wurde im Rahmen dieser Realisierung dagegen verzichtet.

5.2.2 Entwicklung der XSLT-Stylesheets

Das XML-Dokument `auktion.xml` enthält alle Daten, die für eine beispielhafte Online-Auktion notwendig sind. Die vorgesehenen WML-Seiten sollen folgende Inhalte für die Online-Auktion bereitstellen.

Auf der Startseite soll eine Liste der verfügbaren Kategorien mit der jeweiligen Anzahl der enthaltenen Objekte erscheinen. Zusätzlich soll ein Link direkt auf eine Liste von Auktionsobjekten verweisen, deren Auktionszeiten demnächst ablaufen. Eine direkte Suche nach Auktionsobjekten wird exemplarisch mit auf der Startseite verfügbar sein. Bei der Navigation über die einzelnen Kategorien, folgt auf einer weiteren Seite, eine nach der Laufzeit sortierte Liste mit den Titeln der Auktionsobjekte. Der Titel eines Auktionsobjekts verweist dann auf erste Zusatzinformationen zu dem Objekt, wie Laufzeit, Mindestgebot und aktuelles Gebot. Eine Möglichkeit zur Abgabe eines neuen Gebots soll hier beispielhaft dargestellt werden. Weitere Informationen über das Objekt, wie Beschreibung und Anbieter, hält eine weitere Ebene bereit. Auch hier soll es die Möglichkeit geben, ein neues Gebot abzugeben.

Für die Aufteilung der Informationen in Cards und Decks wurde folgendes Schema gewählt. Insgesamt umfasst der Ausschnitt der Online-Auktion sechs WML-Dateien. Das Startdeck enthält dabei zwei Cards. Die erste Card enthält die angesprochene Übersicht über die Auktionskategorien und die zweite Card, die Liste der "Last-Minute-Objekte". Die zweite Ebene der Kategorie ist in der zweiten Datei untergebracht. Dabei besitzt jede Kategorie eine Card mit den Titeln der Objekte, die dieser Kategorie zugeordnet sind. Die Anzahl der Decks der dritten Ebene entspricht der Anzahl der Kategorien, in diesem Fall also drei. Jedes dieser Decks enthält die Kurz- und Zusatzinformationen der Objekte einer Kategorie. Die letzte WML-Datei wird nicht über ein Stylesheet generiert, da es nur zwei Cards umfasst, die Mel-

⁸⁸ Vgl. BEHME, Henning; MINTERT, Stefan (1998), S. 62

Realisierung von WML-Seiten mit XSL(T)

dungen zur Objektsuche und Gebotsabgabe enthalten. Eine Übersicht der Aufteilung findet sich in Abbildung 100.

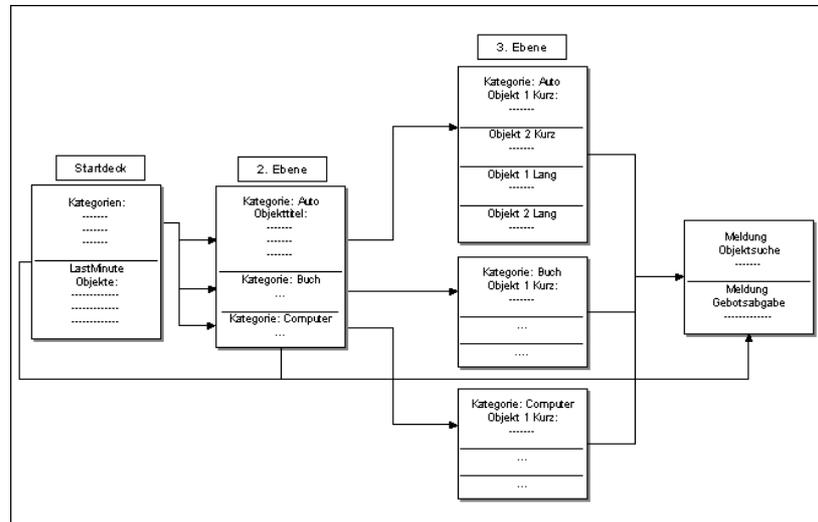


Abbildung 100: Online-Auktion Aufteilung in Decks und Cards

5.2.2.1 Stylesheet des Startdeck

Das Stylesheet des Startdeck hat die Aufgabe, die Anzahl der Objekte einer Kategorie zu ermitteln und diejenigen Objekte in einer Liste darzustellen, deren Versteigerung als nächstes geschlossen wird. Die Liste soll zusätzlich nach Enddatum und Endzeit sortiert werden.

Da XSL-Stylesheets selbst XML-Dokumente sind, enthalten sie am Anfang eine "Processing Instruction". Das Wurzelement `xsl:stylesheet` enthält die Informationen, welche XSL-Elemente und sonstigen Elemente im weiteren Verlauf verwendet werden. In diesem Fall sind die XSL-Elemente, die aus dem Namensraum von XSLT in der Version 1. Bei den Elementen ohne das Präfix `xsl:` handelt es sich um Elemente aus der WML-Spezifikation Version 1.1 des WAP-Forums.

Auszug aus dem XSL-Stylesheet des Startdecks (i.xsl)

```
<?xml version="1.0"?>
<xsl:stylesheet version="1.0" xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
xmlns="http://www.wapforum.org/DTD/wml_1.1.xml">
<xsl:output media-type="text/vnd.wap.wml" xml-declaration="yes"
doctype-public="-//WAPFORUM/DTD WML 1.1//EN"
doctype-system="http://www.wapforum.org/DTD/wml_1.1.xml"/>
<xsl:template match="/">
<xsl:variable name="kat_a">Auto</xsl:variable>
[...]
<wml>
  <card id="card1" title="Online-Auktion" newcontext="true">
    <p align="center">
      <a href="#card2"><xsl:text>Last Minute</xsl:text></a><br/>
      <xsl:text>Kategorien: </xsl:text><br/>
      <a href="Titel.wml#auto"><xsl:text>Auto</xsl:text>
      <xsl:text></xsl:text>
      <xsl:value-of select="count(//kategorie[.=$kat_a])"/>
      <xsl:text></xsl:text>
    </a><br/>
    [...]
  </p>
</card>
[...]
</xsl:template>
</xsl:stylesheet>
```

Über das Element `xsl:output` definiert man die Dokumentendeklaration, die im Zieldokument, also in der WML-Datei, vorhanden sein soll. In diesem Fall sind das die Angaben, die für WML-Dokumente notwendig sind. Darauf folgt die eigentliche Verarbeitung der Daten.

Das Element `xsl:template` definiert, welches Element oder welche Elemente in die nachfolgenden Verarbeitungsanweisungen einbezogen werden sollen. Hier ist es das Wurzelement `auktion`, das mit `/` abgekürzt wird. Anschließend folgt die Definition von Variablen, die für spätere Verarbeitungsschritte verwendet werden. Dann wird die Struktur der WML-Elemente für das spätere Zieldokument festgelegt. Text, der im Zieldokument ausgegeben werden soll, wird mit den Tags `<xsl:text>` und `</xsl:text>` eingeschlossen. Der entscheidende Teil für die erste Card ist das Element `xsl:value-of`. Bei diesem Vorgang werden die Elemente `kategorie` gezählt, die als Inhalt den zugewiesenen Variablenwert, hier also `Auto`, besitzen. Das bedeutet, dass alle Auktionsobjekte gezählt werden, die in der Kategorie `Auto` eingeordnet sind. Der Wert wird dann im WML-Dokument in Klammern hinter den Kategorienamen gesetzt. Ebenso wird mit den beiden anderen Kategorien verfahren.

Realisierung von WML-Seiten mit XSL(T)

Auszug aus dem XSL-Stylesheet des Startdecks (i.xsl)

```
<xsl:variable name="last_min">20000810</xsl:variable>
[...]
<card id="card2" title="Last-Minute">
  [...]
  <xsl:for-each select="auktion/*">
    <xsl:sort order="ascending" data-type="number" select="enddatum/@int"/>
    <xsl:sort order="ascending" data-type="number" select="endzeit/@int"/>
    <xsl:if test="enddatum/@int[.&lt;$last_min]">
      <a>
        <xsl:attribute name="href">
          <xsl:value-of select="kategorie"/>
          <xsl:text>.wml#</xsl:text>
          <xsl:value-of select="@id"/>
        </xsl:attribute>
        <xsl:value-of select="titel"/><xsl:text> (</xsl:text>
        <xsl:value-of select="enddatum"/><xsl:text>/</xsl:text>
        <xsl:value-of select="endzeit"/><xsl:text>)</xsl:text>
      </a><br/>
    </xsl:if>
  </xsl:for-each>
  [...]
</xsl:template>
</xsl:stylesheet>
```

Für die zweite Card des Startdecks sind verschiedene Verarbeitungsanweisungen im Stylesheet von Bedeutung. Zunächst wird mit dem Element `xsl:for-each` angezeigt, dass sich die folgenden Anweisungen auf jedes der angegebenen Elemente beziehen. Hier etwa werden alle Elemente unterhalb von `auktion`, also jedes `objekt` der XML-Quelle, in die Verarbeitung miteinbezogen. Es folgt eine Sortierung der Objekte über das Element `xsl:sort`, zuerst nach dem Enddatum und anschließend nach der Endzeit. Mit `xsl:if` wird geprüft, ob die Objekte ein Enddatum aufweisen, welches vor dem Datum liegt, das in der Variable `last_min` am Anfang des Stylesheets definiert wurde. Ist das der Fall, wird Titel, Enddatum und Endzeit dieser Objekte in eine Liste geschrieben. Dies geschieht jeweils mit dem Element `xsl:value-of`. Gleichzeitig werden die Titel über den Namen der Kategorie und die ID des Objekts mit den entsprechenden Kurz- und Langbeschreibungen der Objekte verknüpft.

5.2.2.2 Stylesheet der Ebene 2

Die Ebene 2 der Online-Auktion besteht, wie in Abbildung 100 dargestellt, aus Listen der Auktionsobjekte der jeweiligen Kategorie, sortiert nach dem Enddatum und der Endzeit. Das folgende Stylesheet stellt diese Ansicht auf die Informationen der XML-Quelle her.

Auszug aus dem XSL-Stylesheet der Ebene 2 (titel.xsl)

```
[...]
<xsl:template match="/">
<xsl:variable name="kat_a">Auto</xsl:variable>
[...]
  <xsl:for-each select="auktion/*">
    <xsl:sort order="ascending" data-type="number" select="enddatum/@int"/>
    <xsl:sort order="ascending" data-type="number" select="endzeit/@int"/>
    <xsl:if test="kategorie[.=$kat_a]">
      <a><xsl:attribute name='href'>
        <xsl:value-of select="kategorie"/>
        <xsl:text>.wml#</xsl:text>
        <xsl:value-of select="@id"/>
      </xsl:attribute>
      <xsl:value-of select="titel"/>
    </a>
    <xsl:if test="not(position()=last())"><br/></xsl:if>
  </xsl:if>
</xsl:for-each>
[...]
```

Über die beiden Elemente `xsl:sort` werden die Objekte zunächst wieder nach dem Enddatum und der Endzeit sortiert. Dann erfolgt die Ausgabe des Titels mit Verlinkung auf die weiteren Beschreibungen. Allerdings erfolgt die Ausgabe nur für Objekte, die in der Kategorie `Auto` eingeordnet sind. Diese Selektion findet über das Element `xsl:if` statt, wobei der Ausdruck `kategorie[.=$kat_a]` die Bedingung darstellt. Die Variable `kat_a` hat in diesem Fall auch den Wert `Auto`. Mittels `xsl:if test="not(position()=last())"` wird solange nach jedem Titel ein Zeilenumbruch eingefügt, bis das letzte Objekt, für das die Bedingung zutrifft, erreicht wurde. Damit ist die Verarbeitung für die erste Card abgeschlossen. Der gleiche Vorgang wird nun für die Kategorien `Buch` und `Computer` durchgeführt. Somit besitzt das Deck der zweiten Ebene am Ende drei Cards.

5.2.2.3 Stylesheets der Ebene 3

Die Stylesheets der Ebene 3 sind sich in der Struktur sehr ähnlich, da jeweils nur eine andere Kategorie je Stylesheet bearbeitet wird. Wäre es mit XSLT möglich, unterschiedliche Ausgabedateien direkt in XSLT zu definieren, würde sogar ein Stylesheet ausreichen. Das folgende Stylesheet zeigt die Darstellung der Informationen für die Kategorie `Auto`.

Realisierung von WML-Seiten mit XSL(T)

Auszug aus dem XSL-Stylesheet der Ebene 3 (auto.xsl)

```
[...]
<xsl:template match="/">
  [...]
  <xsl:variable name="kat">Auto</xsl:variable>
  <xsl:for-each select="auktion/*">
    <xsl:if test="kategorie[.=$kat]">
      [...]
      <xsl:text>Titel: </xsl:text><xsl:value-of select="titel"/>
      <br/>
      <xsl:text>Min.-Gebot: </xsl:text>
      <xsl:value-of select="startgebot"/>
      <xsl:text>DM</xsl:text><br/>
      <xsl:text>Akt.-Gebot: </xsl:text>
      <xsl:for-each select="gebote/*">
        <xsl:if test="(position())=last() ">
          <xsl:value-of select="preis"/>
        </xsl:if>
      </xsl:for-each>
      <xsl:text>DM</xsl:text><br/>
      [...]
    </xsl:if>
  </xsl:for-each>
[...]
```

Bei diesem Stylesheet werden wieder nur die Objekte bearbeitet, die in der Kategorie `Auto` liegen. Dabei wird eine Card erstellt, die den Titel des jeweiligen Objekts, das angegebene Mindestgebot und das aktuell höchste Gebot enthält. Der Titel und das Mindestgebot werden jeweils über die Elemente `xsl:value-of` aus dem XML-Dokument extrahiert. Da die Gebote in dem XML-Dokument sequenziell aneinander gereiht werden, ist das in der Reihenfolge letzte Gebot auch das aktuellste. Dieser Umstand kann mit der Bedingung `<xsl:if test="(position())=last() ">` ausgenutzt werden, um den Preis des jeweils letzten Gebots ausgeben zu lassen. Jedes Objekt besitzt zusätzlich einen Verweis auf entsprechende Zusatzinformationen zu dem Objekt.

Realisierung von WML-Seiten mit XSL(T)

Die Cards mit den Zusatzinformationen werden im folgenden definiert.

Auszug aus dem XSL-Stylesheet der Ebene 3 (auto.xsl)

```
[...]  
<xsl:variable name="kat">Auto</xsl:variable>  
  <xsl:for-each select="auktion/*">  
    <xsl:if test="kategorie[.=$kat]">  
      [...]  
      <xsl:text>Titel: </xsl:text><xsl:value-of select="titel"/><br/>  
      <xsl:text>Beschreibung: </xsl:text><xsl:value-of  
        select="beschreibung"/><br/>  
      <xsl:text>Anbieter: </xsl:text><xsl:value-of  
        select="anbieter/vorname"/><xsl:text> </xsl:text>  
      <xsl:value-of select="anbieter/nachname"/>  
      [...]  
    </xsl:if>  
  </xsl:for-each>  
[...]
```

Der zweite Teil des Stylesheet arbeitet nach dem gleichen Prinzip wie der erste Teil, mit dem Unterschied, dass andere Informationen aus dem XML-Dokument dargestellt werden. Mit dem Element `xsl:value-of` werden jeweils Titel, Beschreibung und der Anbieter des Objekts mit Vor- und Nachname aus dem XML-Dokument extrahiert. Für die Stylesheets der anderen Kategorien `Buch` und `Computer` ändert sich für beide Teile des Stylesheet, jeweils nur der Variablenwert der Variable `kat`.

5.2.3 Generierung der WML-Seiten

Die endgültige Generierung der WML-Seiten erfolgt über einen sogenannten XSL-Prozessor, der das Ausgangsdokument, hier das XML-Dokument `auktion.xml`, mit den verschiedenen XSL-Stylesheets zusammenführt. Für dieses Beispiel wurde das Programm `XT` von James Clark, der auch der Herausgeber der XSLT-Spezifikation beim W3C ist, in der Version "`XT packaged as a Win32 executable`" ausgewählt.⁸⁹ Der Grund für diese Wahl, ist die vollständige Implementierung der XSLT-Spezifikation und die relativ leichte Bedienung über die Kommandozeile von Windows. Mit dem Befehl `xt.exe auktion.xml stylesheet.xsl ausgabe.wml` wird aus der XML-Quelle `auktion.xml` und dem Stylesheet `stylesheet.xsl` die WML-Datei `ausgabe.wml` generiert.

⁸⁹ Quelle: <http://www.jclark.com/xml/xt.html> (29.05.2000)

Realisierung von WML-Seiten mit XSL(T)

Mit dem XSL-Prozessor XT wurden mit den Stylesheets die WML-Seiten `i.wml`, `Titel.wml`, `Auto.wml`, `Buch.wml` und `Computer.wml` generiert. Die Datei `meld.wml` wurde, wie bereits erwähnt, manuell erstellt. Eine geringfügige Nachbearbeitung der WML-Dokumente ist allerdings zu empfehlen, da die Generierung zwei Teile hinterlässt, über welche die Microbrowser stolpern könnten. Der XSL-Prozessor generiert folgenden XML-Deklaration mit dem Zusatz, um welchen Zeichensatz es sich handelt:

```
<?xml version="1.0" encoding="utf-8"?>
```

Dieser Zusatz sollte gelöscht werden, dass am Ende `<?xml version="1.0"?>` steht. Ebenso sollte beim Top-Level-Element `wml` verfahren werden, das im WML-Dokument das Attribut `xmlns`, also den Namensraum zugewiesen bekommt.

```
<wml xmlns="http://www.wapforum.org/DTD/wml_1.1.xml"> wird zu <wml>90
```

Abschließend zeigt Abbildung 101 die generierten Cards und Decks und deren Ablaufschema.

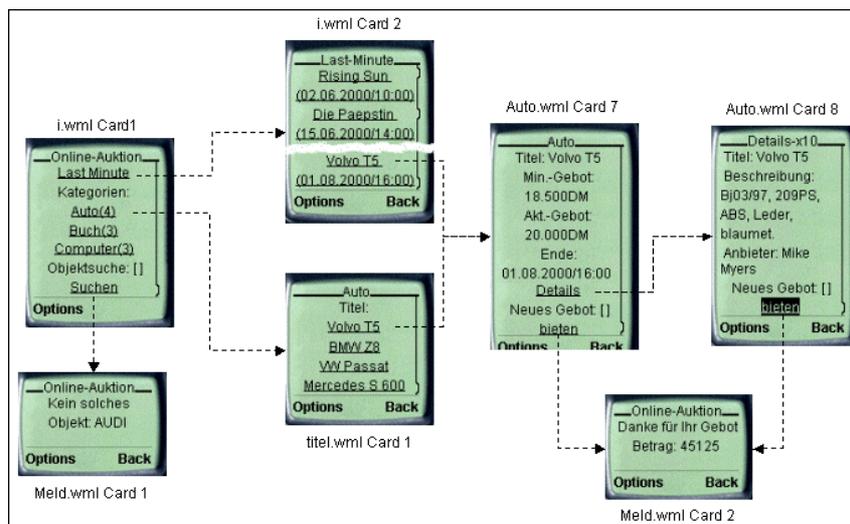


Abbildung 101: Online-Auktion WML-Seiten

⁹⁰ Vgl. BEHME, Henning (2000), S.181

Realisierung von WML-Seiten mit XSL(T)

Alle in dieser Diplomarbeit erstellten Lösungsbeispiele können über den WAP-Server der GMD abgerufen werden. Über die Indexdatei <http://rom.darmstadt.gmd.de/budach/i.wml> können alle Beispiele erreicht werden.

6 Ausblick

6.1 Momentane Hindernisse

Das Thema WAP erfährt seit etwa sechs Monaten eine starke Medienaufmerksamkeit. Die Erfolgserwartungen von WAP sind stark gestiegen. Auf der einen Seite hat es das Wort WAP schon bis in die Tagesthemen der ARD geschafft⁹¹ und auf der anderen Seite tun sich die Anbieter bisher schwer, wirkliche "Killerapplikationen" zu entwickeln, die eine starke Nutzung rechtfertigen.

Eine gute Einschätzung der Situation gibt Durlacher Research. Nach Durlacher befinden wir uns gerade am Anfang der Enttäuschungsphase nach dem WAP-Hype. Der richtige Aufschwung des Mobile Commerce tritt nach Durlacher im Jahr 2002 ein.⁹² Abbildung 102 verdeutlicht den Verlauf der einzelnen Phasen.

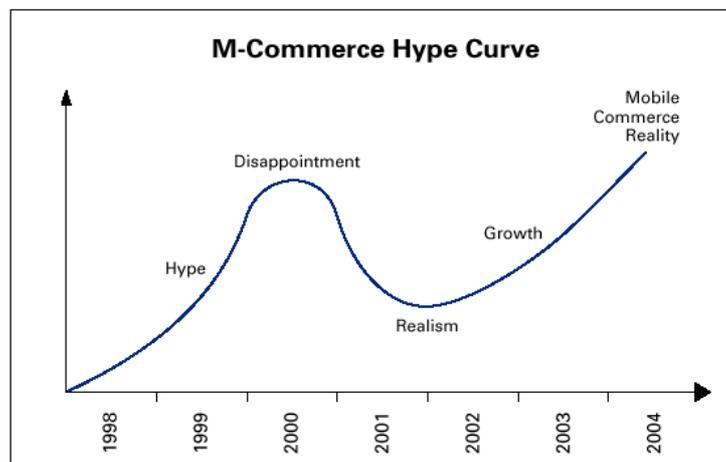


Abbildung 102: Verlauf der M-Commerce-Kurve (Quelle: Durlacher)

⁹¹ Vgl. BUNZ, Mercedes (2000)

⁹² Vgl. MÜLLER-VEERSE, Falk (1999), S.13

Ausblick

6.1.1 Endgeräte

Bei Beginn dieser Arbeit stellte das Fehlen von WAP-fähigen Endgeräten, eines der größten Probleme von WAP dar. Mit dem Abschluss dieser Arbeit hat sich die Situation deutlich verbessert. Aktuell sind fünf WAP-fähige Mobiltelefone für das GSM-Netz im freien Handel und in ausreichenden Stückzahlen vorhanden (Stand: 01.06.2000).

Vor einem halben Jahr noch war es für Entwickler und deren Unternehmen schwierig, geeignete Testgeräte zu erhalten, sodass viele Anwendungen nur auf Entwicklungsumgebungen getestet werden konnten.

6.1.2 Kosten

Da momentan meistens nur abgespeckte Internetdienste auf WAP-Basis angeboten werden, fällt es schwer einzusehen, wieso das mit einem Preis bezahlt werden soll, der um den Faktor zehn über dem eines Internetzugangs liegt.⁹³ Den Vorteil der Mobilität muss man sich also teuer erkaufen. Einen gewissen Anteil am Erfolg oder Misserfolg von WAP wird also die zukünftige Preispolitik der Mobilfunknetzbetreiber haben.

6.1.3 Entwicklungsreife

Da WAP 1.1 momentan noch die Standardimplementierung ist, können die Erweiterungen von WAP 1.2 noch nicht praxisgerecht umgesetzt werden. In WAP 1.1 sind verschiedene Teile, wie die Wireless Telephony Application (WTA) noch nicht vollständig implementiert. WTA soll Elemente enthalten, die es ermöglichen, auf Funktionen des mobilen Endgeräts zu zugreifen. Dies umfasst zum Beispiel das direkte Verwenden einer Telefonnummer, die in einer WAP-Anwendung abgelegt ist.⁹⁴ Eine weitere wichtige Funktion der WTA ist die Bereitstellung der Push-Technologie. Die Push-Technologie ermöglicht das selbstständige Senden von Daten vom Server zum mobilen Endgerät. Man kann sich zum Beispiel vorstellen, dass dadurch eingehende E-Mails angezeigt werden oder Börsenkurse aktualisiert auf dem Display erscheinen. Die Push-Technologie ist in der Version WAP 1.2 vorhanden.⁹⁵

⁹³ Basispreis von 0,39 DM/min gegenüber 0,033 DM/min (MSN.de) (06.04.2000)

⁹⁴ Vgl. WAP FORUM (1999c), S.10

⁹⁵ Vgl. BUCKINGHAM, Simon (1999)

Obwohl WMLScript schon voll in WAP 1.1 spezifiziert ist, ist die Implementierung in den Browsern der Endgeräte bisher nur zögerlich erfolgt. Erst mit den jetzt auf den Markt kommenden Geräten, kann man WMLScript uneingeschränkt nutzen. Wie schnell die Hersteller der Soft- und Hardware die Version 1.2 nun implementieren, bleibt abzuwarten.

6.2 Zukünftige Entwicklungen

6.2.1 Regionale Dienste

Derzeit arbeiten verschiedene Institutionen an Konzepten für sogenannte "location based services". Dabei steht in allen Fällen der Einbezug der geographischen Position des mobilen Endgeräts im Vordergrund. In Deutschland soll dies ermöglicht werden, indem die WAP-Anwendung auf die Koordinaten der Funkzelle zugreifen kann, in die der Benutzer mit seinem mobilen Endgerät eingebucht ist. Das W3C hat beispielsweise einen "WAP Location Workshop" abgehalten, bei dem unter anderem die möglichen Anwendungen diskutiert wurden. Unterschiedlichste Szenarien sind damit denkbar. Bei einem Notruf könnte beispielsweise gleichzeitig der Standort der hilfeschuchenden Person übermittelt werden.

In eine andere Richtung geht die Überlegung der ortsabhängigen Informationsversorgung. Man stelle sich vor, man möchte die Abfahrtszeit eines Zuges ermitteln und bekommt aufgrund seiner Position die Abfahrtszeiten der Züge des nächstgelegenen Bahnhofs auf seinem mobilen Endgerät angezeigt. Die unter Umständen umständliche Eingabe des Ausgangsbahnhofs kann damit entfallen. Ein weiteres Beispiel konnte auf der bzw. rund um die CeBIT 2000 getestet werden. Die Testversion des WAP-Dienstes listete alle nächstgelegenen Restaurants mit Adresse und Entfernung auf.⁹⁶ Spinnt man dieses Konzept gedanklich weiter, könnte man zu folgender Situation kommen. Geschäfte oder Restaurants könnten sich für spezielle Marketingdienstleistungen registrieren. Dabei würden beispielsweise Sonderangebote an das mobile Endgerät gesendet, sobald sich ein Besitzer eines WAP-fähigen mobilen Endgeräts in der Nähe eines der registrierten Unternehmen befindet.

⁹⁶ Vgl. "WAP-Handy als Restaurant-Berater", Heise Online News, 23.02.2000, <http://www.heise.de/newsticker/data/dz-23.02.00-005/> (05.04.2000)

Ausblick

6.2.2 Kontextdienste

Mit Kontextdiensten ist gemeint, dass sich die Dienste bzw. deren Präsentation an den augenblicklichen Zusammenhang anpassen können. Ein möglicher Zusammenhang ist die oben angesprochene geographische Position eines Benutzers. Andere denkbare Kontexte sind Merkmale der Umgebung, in der sich der Benutzer aufhält, wie Lichtverhältnisse, Geräuschpegel usw. Sind die Lichtverhältnisse beispielsweise schlecht, kann eventuell eine größere Schrift oder mehr Kontrast eingesetzt werden. Dies erfordert natürlich intelligente Endgeräte, welche die augenblickliche Situation an die Anwendungen übermitteln können.⁹⁷

6.2.3 Personalisierung

Die zukünftigen für den mobilen Bereich entwickelten Dienste, werden sehr speziell auf den einzelnen Nutzer zugeschnitten sein. Im Grunde dreht sich alles um die Vereinfachung der Nutzung der angebotenen Dienste. Im Mittelpunkt muss die Information stehen, die der Anwender benötigt. Im besten Fall nimmt der Nutzer die Technik gar nicht wahr.

Vor diesem Hintergrund der Personalisierung, ergibt sich auch ein enormes Diskussionspotenzial, da mit fortschreitender Personalisierung auch die Möglichkeiten der Kontrolle wachsen. Nie zuvor konnte man Personen so genau adressieren und hatte im gleichen Augenblick auch derart viele Informationen über eine Person. Gerade in Deutschland werden aufgrund der oben geschilderten Möglichkeiten, Datenschützer mit Sicherheit große Besorgnis über derartige Entwicklungen äußern.

6.2.4 Sprachsteuerung

Was bei Mobiltelefonen für das Wählen aus dem Telefonbuch zunehmend zum Funktionsumfang gehört, könnte auch einen Mehrwert für mobile Informationsdienste darstellen. Ein Beispiel ist die sprachliche Eingabe eines Textes für eine E-Mail. Ein weiteres Anwendungsgebiet betrifft die Navigation innerhalb eines WAP-Dienstes. Sie könnte die momentan umständliche Navigation ablösen. Die Aktivitäten des W3C hinsichtlich Spracherkennung und -ausgabe im WWW, lassen sich leicht auf den mobilen Bereich übertragen.⁹⁸ Auch behin-

⁹⁷ Vgl. SCHMIDT, Albrecht [u.a.] (1998)

⁹⁸ Vgl. <http://www.w3.org/Voice/> (29.05.2000)

dernten Menschen könnte die Sprachsteuerung einen verbesserten Zugang zu Informationen ermöglichen.

6.2.5 Zahlungsmittel mobiles Endgerät

Einen großen Anteil am Erfolg des Mobile Commerce könnte die Möglichkeit bieten, das mobile Endgerät als Zahlungsmittel einzusetzen. Die Akzeptanz bei den Nutzern mindestens scheint gegeben. Eine Umfrage von Mannesmann D2 ergab, dass sich 54 Prozent der Kunden ihr Handy als Kreditkartenersatz vorstellen können.

Ein mögliches Zukunftsszenario sieht folgendermaßen aus: Bei einem Supermarktbesuch fällt zukünftig das Warten an der Kasse weg. Das mobile Endgerät enthält die Möglichkeit über eine Infrarotschnittstelle beliebige Waren einzuscannen. Durch das Einscannen wird automatisch der elektronische Diebstahlschutz aufgehoben. Verlässt der Kunde den Supermarkt, werden die Waren direkt über das mobile Endgerät bezahlt.

6.2.6 Neue Übertragungstechniken

Technologisch ist High Speed Circuit Switched Data (HSCSD) der erste Schritt zur Weiterentwicklung des GSM-Netzes. HSCSD ist heute schon verfügbar, allerdings bietet bisher nur E-Plus in Deutschland dieses Verfahren an.⁹⁹ Mit HSCSD sollte eine Möglichkeit geschaffen werden, die niedrige Bandbreite der Mobilfunknetze zu erhöhen. Bei diesem Verfahren werden mehrere (zwei bis acht) Funkkanäle gebündelt. Allerdings muss jeder Kanal separat bezahlt werden, was schnell in astronomischen Minutenpreisen enden kann. Ein weiterer Nachteil ist, dass die Nutzung von HSCSD, ein entsprechendes HSCSD-fähiges Handy erfordert.

Da in naher Zukunft die Einführung des General Packet Radio Service (GPRS), einem neuen Protokoll im GSM Standard, ansteht, kann davon ausgegangen werden, dass HSCSD nur eine Übergangslösung ist. Die derzeit mögliche Übertragungsrate von 9,6 Kbit/s – ohne HSCSD – wird im zweiten Halbjahr 2000 theoretisch auf 171 Kbit/s angehoben werden können. Die Übertragungsrate wird dann zunächst bei etwa 43,2 Kbit/s liegen. Technisch ermöglicht wird dies durch eine bessere Ausnutzung der vorhandenen Funknetzkapazitäten. Ein Nachteil ist, dass sowohl auf Nutzer-, wie auf Anbieterseite neue Hardware erforderlich

⁹⁹ Vgl. <http://www2.eplus.de/cgi-bin/suche/go.pl?haupt/alleswissenswert/dienste/hscsd.htm> (01.06.2000)

Ausblick

ist, weswegen heutige mobile Endgeräte nicht in den Genuss kommen werden, mit höheren Übertragungsraten zu arbeiten. Dieser Umstand wird zunächst auch einer schnellen Flächendeckung mit GPRS im Wege stehen.¹⁰⁰ Ein großer Vorteil von GPRS dagegen ist, dass eine permanente Verbindung zum mobilen Endgerät besteht. So können beispielsweise eingegangene E-Mails direkt an das mobile Endgerät gesendet werden, ohne dass der Nutzer eine Verbindung aufbauen muss.

Der nächste Schritt wird Enhanced Data Rate for GSM Evolution (EDGE) darstellen. Diese Technologie stellt eine Weiterentwicklung von GPRS dar und wird Übertragungsraten von bis zu 384 Kbit/s erreichen. Verfügbar soll EDGE im Laufe des Jahres 2002 werden.¹⁰¹

Ab dem Jahre 2003 soll dann die Technologie der sogenannten 3. Generation eingeführt werden. In Europa wird das Universal Mobile Telephone System (UMTS) dann Übertragungsraten bis 2Mbit/s anbieten können.¹⁰² Die Versteigerung der Funkfrequenzlizenzen beginnt in Deutschland am 31. Juli dieses Jahres.¹⁰³

6.2.7 Multimedia

Bisher sind die Begriffe Multimedia und mobile Endgeräte nur schwer in Einklang zu bringen. Die "klassischen" WAP-fähigen Endgeräte können außer Text und Schwarz-Weiß Graphiken nicht viel darstellen.

Mit den oben erwähnten Neuerungen im Bereich der Mobilfunknetze wird auch die Multimedialität der mobilen Endgeräte zunehmen. Erste Ansätze in diese Richtung sind gemacht. So

¹⁰⁰ Vgl. ŽIVADINOVIĆ, Dusan (1999b), S.188

¹⁰¹ Vgl. MÜLLER-VEERSE, Falk (1999), S.20

¹⁰² Vgl. BORCHERS, Detlef (2000), S.105

¹⁰³ Vgl. "UMTS-Versteigerung offiziell mit 11 Teilnehmern", Heise Online News, 31.05.2000, <http://www.heise.de/newsticker/data/jk-31.05.00-002/> (01.06.2000)

hat beispielsweise Samsung ein Handy mit integriertem MP3-Player vorgestellt.¹⁰⁴ Theoretisch steht in naher Zukunft dann auch Videokonferenzen nichts mehr im Wege, sobald die mobilen Endgeräte mit besseren Displays ausgestattet werden.

6.3 Fazit und Resümee

Fasst man die oben stehenden Thesen und Fakten zusammen, kommt man zum Ergebnis, dass mobile Endgeräte in Zukunft leistungsfähiger und multifunktional sein werden. Die Integration von persönlichem Assistent, Handy, Informations-, Kommunikations- und Transaktionsterminal, Zahlungsmittel und Unterhaltungsmedium wird weiter voran schreiten. Die Weiterentwicklung der Mobilfunknetze ermöglicht erst die Nutzung der zukünftigen Funktionen der mobilen Endgeräte.

Mit Sicherheit kann man sagen, dass die Verbreitung mobiler Dienste in Zukunft stark zunehmen wird. In welche Richtung sich diese Dienste entwickeln, ist noch nicht endgültig abzusehen. Momentan befindet sich WAP als Teilelement des mobilen Bereichs auf jeden Fall noch in der Entwicklungs- und Erprobungsphase, die unter Umständen noch einige Zeit anhalten kann. WAP findet derzeit am ehesten als "Value-Added-Service" zu bisherigen Webdiensten und als "Special-Interest-Dienste" Anwendung.

¹⁰⁴ Vgl. "Musik aus dem Handy", Heise Online News, 24.02.2000, <http://www.heise.de/newsticker/data/dz-24.02.00-003/> (06.04.2000)

7 Literaturverzeichnis

- ADLER, Sharon [u.a.] (2000): "Extensible Stylesheet Language (XSL) : Version 1.0 : W3C Working Draft 27 March 2000", <http://www.w3.org/TR/2000/WD-xsl-20000327/> (28.05.2000)
- AU SYSTEM RADIO (1999): "WAP White Paper, February 1999", <http://www.ausys.se/servlet/PSPDownloadServlet?iListItemId=4698> (28.05.2000).
- BAGER, Jo; BLEICH, Holger (2000): "WAP-Galerie : Die nützlichsten WAP-Sites" in c't - Magazin für Computertechnik 9/2000, S.200-207.
- BAGER, Jo (1999): "Das Handy spricht Web : Eine kurze Einführung in die Wireless Markup Language WML", in c't - Magazin für Computertechnik 22/1999, S.126-130.
- BAGER, Jo (1997): "HTML++ : Style Sheets und XML – die Zukunft des Web?", in c't - Magazin für Computertechnik 8/1997, S. 298-304.
- BEHME, Henning (2000): "Wenn Markup lockt : WML: XML-Dat(ei)en aufs Handy bringen", in iX - Magazin für professionelle Informationstechnik 4/2000, S. 178-181.
- BEHME, Henning (1999): "Kunst der Stunde : Wozu die Extensible Markup Language gut ist", in iX - Magazin für professionelle Informationstechnik 2/1999, S. 36-41.
- BEHME, Henning (1999a): "Stil-Leben: XSLT : Transformation von XML-Dokumenten", in iX - Magazin für professionelle Informationstechnik 11/1999, S.181-183.
- BEHME, Henning; MINTERT, Stefan (1998): "XML in der Praxis : Professionelles Web- Publishing mit der Extensible Markup Language", Addison-Wesley Verlag, 1998.
- BORCHERS, Dettlef (2000): "Heinzelmännchen drahtlos : Generationswechsel der Handys", in c't - Magazin für Computertechnik 3/2000, S.104-109.
- BRAY, Tim [u.a.] (1998): "Extensible Markup Language (XML) 1.0 : W3C Recommendation", <http://www.w3.org/TR/1998/REC-xml-19980210> (26.05.2000).
- BROSIUS, Felix (2000): "WML. WAP-Anwendungen programmieren", Addison-Wesley Verlag, 2000 (noch nicht erschienen, 08.06.2000).
- BUCKINGHAM, Simon (1999): "Data on WAP", Mobile Lifestreams, 2000.
- BUDACH, Jens (2000): "WAP – Mobil ins Internet : Ausarbeitung zur Lehrveranstaltung 'Diplomvorbereitung' im WS 1999/2000", FH Darmstadt, FB IuD, 2000.

Literaturverzeichnis

- BUNZ, Mercedes (2000): "WAP oder : Wie das Internet laufen lernt", in Telepolis – Magazin für Netzkultur, 07.03.2000, <http://www.heise.de/tp/deutsch/inhalt/te/5873/1.html> (04.04.2000).
- CLARK, James (1999): "XSL Transformations (XSLT) : Version 1.0 : W3C Recommendation 16 November 1999", <http://www.w3.org/TR/1999/REC-xslt-19991116> (04.06.2000).
- CLARK, James; DEROSE, Steve (1999): "XML Path Language (XPath) : Version 1.0 : W3C Recommendation 16 November 1999", <http://www.w3.org/TR/1999/REC-xpath-19991116> (04.06.2000).
- COLTON, Paul (2000): "WAP Development with WML", Sams Publishing, 2000 (noch nicht erschienen, 08.06.2000).
- ECKSTEIN, Pia (2000): "WAP-M@nia", in connect 4/2000, S.26-30.
- ERICSSON (2000): "Mobile Phone R320 : Design Guidelines for WAP Services", http://www.ericsson.com/developerszone/uploadedfiles/r320_design_guidelines_b.pdf (31.05.2000).
- ERICSSON (1999): "Design Guidelines for WAP Services – Mobile Phone R380s", http://www.ericsson.com/developerszone/uploadedfiles/r380_design_guidelines_a.pdf (28.05.2000).
- ERICSSON (1999a): "Mobile Companion MC218 : Design Guidelines for WAP Services", http://www.ericsson.com/developerszone/uploadedfiles/mc218_design_guidelines_a.pdf (31.05.2000).
- FLAHERTY, Natasha (2000): "Locations for Location", Presentation at the W3C-WAP Forum workshop on Position Dependent Information Services, 28.02.2000, <http://www.w3.org/Mobile/posdep/Presentations/Noens/index.htm> (28.05.2000).
- FLAHERTY, Natasha (1999): "Wireless Application Protocol : Technical Overview“, Presentation at WirelessDeveloper '99 in Monterey, CA, USA, 6 May 1999, <http://www.wapforum.org/what/presentation.htm> (28.05.2000).
- FONTE, Paul Benedict (2000): "WAP Development with Wml and WmlScript (Sams White Book Series)", Sams Publishing, 2000 (noch nicht erschienen, 08.06.2000).
- HITZIG, Andreas (1999): "Wireless Application Protocol : Drahtlos Surfen mit Volldampf", in iX - Magazin für professionelle Informationstechnik 10/1999, S.128-132.

Literaturverzeichnis

- HOBBS, John Maxwell (1999): "Design Rules for mobile Applications", in Developer.com, 03.12.1999, http://www.developer.com/journal/techworkshop/120399_mobile.html (01.05.2000).
- HÜMMLER, Thomas (2000): "Schöne Bescherung : WAP-Geräte under Construction", in Internet Professionell Januar 2000, S.44-45.
- IRELAND, Tim (2000): "Net.Works Guide to WAP", Trafalgar Square, 2000 (noch nicht erschienen, 08.06.2000).
- JODELEIT, Bernhard (2000): "W@P's going on?", in connect 4/2000, S.17-24.
- KOCH, Stefan (1999): „JavaScript : Einführung, Programmierung und Referenz“, 2., aktualisierte und erweiterte Auflage, dpunkt-Verlag, 1999.
- KOSSEL, Axel (2000): "WAP Wahn – das Märchen vom Surfen mit dem Handy", in c't - Magazin für Computertechnik 9/2000, S.194-195.
- KUHLEN, Rainer (1997): "Hypertext", in Grundlagen der praktischen Information und Dokumentation :ein Handbuch zur Einführung in die fachliche Informationsarbeit, Saur Verlag, 1997, S.355-366.
- LINDHORST, Arno (2000): "WAP", DATA Becker, 2000 (noch nicht erschienen, 08.06.2000).
- MANN, Steve (2000): "Programming Applications with the Wireless Application Protocol : The Complete Developer's Guide", John Wiley & Sons, 2000.
- MCGRATH, Mike (2000): "WAP in Easy Steps", Computer Step, 2000 (noch nicht erschienen, 08.06.2000).
- MINTERT, Stefan (2000): "Mobil online – Ericsson MC 218 im Test", in iX - Magazin für professionelle Informationstechnik 2/2000, S.58 -59.
- "Mobile Networking with WAP", Vieweg, 2000 (noch nicht erschienen, 08.06.2000)
- MÜLLER-VEERSE, Falk (1999): "Mobile Commerce Report", Durlacher Research Ltd., 1999.
- NIELSEN, Jakob (1996): "Inverted Pyramids in Cyberspace", in useit.com, June 1996, <http://www.useit.com/alertbox/9606.html> (25.04.2000).
- NOKIA (1999): "WML Reference : Version 1.1, September 1999", <http://www.forum.nokia.com> (28.05.2000).
- NOKIA (1999a): "WMLScript Reference : Version 1.1, September 1999", <http://www.forum.nokia.com> (28.05.2000).

Literaturverzeichnis

- NOKIA (1999b): "Service Developer's Guide for the Nokia 7110, November 1999",
<http://www.forum.nokia.com> (28.05.2000).
- PHILIPPE, Jean-Richard (2000): "WAP – mobil: im Internet", Kilchberg Verlag, 2000 (noch nicht erschienen, 08.06.2000).
- PHONE.COM (1999): "Developing User Friendly Applications",
<http://updev.phone.com/dev/ts/technotes/userfriendly.pdf> (28.05.2000).
- "Practical Transformation Using XSLT and XPath (XSL Transformations and the XML Path Language)", Crane Softwrights Ltd., Seventh Edition, 2000.
- "Professional WAP", Wrox Press, 2000 (noch nicht erschienen, 08.06.2000)
- RAGGET Dave [u.a.] (2000): "Extensible HyperText Markup Language (XHTML) 1.0 : A Reformulation of HTML 4 in XML 1.0, W3C Recommendation",
<http://www.w3.org/TR/2000/REC-xhtml1-20000126> (25.05.2000).
- RAGGET Dave [u.a.] (1997): "HTML 4.0 Specification, W3C Recommendation",
<http://w3.org/TR/REC-html40> (25.05.2000).
- RENSMANN, Jörg (2000): "Geld verdienen mit WAP : From Web to WAP", in Internet Professionell Januar 2000, S.46-53
- RISCHPATER, Ray (2000): "WAP und WML – Wireless Web – Das neue Internet", Galileo Press, 2000 (noch nicht erschienen, 08.06.2000).
- RÖWEKAMP, Lars (2000): "Drahtlos die Dritte : Wireless Application Protocol 1.2 verabschiedet", in iX - Magazin für professionelle Informationstechnik 5/2000, S.104-107.
- RÖWEKAMP, Lars (2000a): "Dynamik mobil : Einführung in WMLScript", in iX - Magazin für professionelle Informationstechnik 3/2000, S.192-197.
- RÖWEKAMP, Lars (2000b): "Handy HTML : Einführung in die Wireless Markup Language", in iX - Magazin für professionelle Informationstechnik 2/2000, S.52-57.
- SCHMIDT, Albrecht [u.a.] (1998): "There is more to Context than Location", in Proceedings of the International Workshop on Interactive Applications of Mobile Computing (IMC'98), November 1998, <http://www.egd.igd.fhg.de/~imc98/> (06.04.2000).
- SIETMANN, Richard (1998): "Mobil ins Internet : Wireless Application Protocol adaptiert Mobiltelefone für das WWW", in c't - Magazin für Computertechnik 4/1998, S.202-207.

Literaturverzeichnis

- SMITH, Josh (2000): "The WAP Vision" in Webreview.com, 11.02.2000,
<http://webreview.com/pace/print/2000/02/11/feature/index2.html> (16.04.2000).
- SPALLEK, Andre M. (2000): "Das große Buch WAP", DATA Becker, 2000 (noch nicht erschienen, 08.06.2000).
- UNWIRED PLANET (Phone.com) (1999): "Enabling The Wireless Internet : Why You Should Develop Applications Using The Unwired Planet WAP Solution, February 1999",
http://www.phone.com/pub/EnablingWP_0100.pdf (29.05.2000).
- VAN DER HEIJDEN, Marcel (Hrsg.) (2000): "Understanding WAP : Wireless Applications, Devices, and Services (Artech House Telecommunications Library)", Artech House Publishers, 2000 (noch nicht erschienen, 08.06.2000).
- WAP FORUM (1999): "Wireless Application Protocol : Wireless Markup Language Specification Version 1.1, 16-Jun-1999", http://www.wapforum.org/what/technical_1_1.htm
(28.05.2000).
- WAP FORUM (1999a): "Wireless Application Protocol : WMLScript Language Specification Version 1.1, 17-Jun-1999", http://www.wapforum.org/what/technical_1_1.htm
(28.05.2000).
- WAP FORUM (1999b): "Wireless Application Protocol : Wireless Application Environment Overview, Version 16-Jun-1999", http://www.wapforum.org/what/technical_1_1.htm
(28.05.2000).
- WAP FORUM (1999c): "Wireless Application Protocol : Wireless Telephony Application Specification Version 1.1, 16-Jun-1999", http://www.wapforum.org/what/technical_1_1.htm
(28.05.2000).
- WAP FORUM (1999d): "Wireless Application Protocol : White Paper",
<http://www.wapforum.org/what/whitepapers.htm> (28.05.2000).
- WAP FORUM (1999e): "Wireless Application Protocol : Wireless Markup Language Specification Version 1.2, 4-November-1999", <http://www.wapforum.org/what/technical.htm>
(28.05.2000).
- WAP FORUM (1999f): "Wireless Application Protocol : WMLScript Language Specification Version 1.2, 4-November-1999", <http://www.wapforum.org/what/technical.htm>
(28.05.2000).

Literaturverzeichnis

"Wml and Wml Script Programmers Professional Reference", Wrox Press, 2000 (noch nicht erschienen, 08.06.2000)

WROX AUTHOR TEAM (2000): "Beginning WAP : Wml and Wmlscript", Wrox Press , 2000 (noch nicht erschienen, 08.06.2000).

ZIERL, Marco (2000): "Internet per Handgriff", in Internet Professionell Januar 2000, S.34-42.

ŽIVADINOVIĆ, Dusan (2000): "WAP inside : Soft- und Hardwarelösungen für den WAP-Zugriff", in c't - Magazin für Computertechnik 9/2000, S.196-198.

ŽIVADINOVIĆ, Dusan (1999): "Endlich gewappnet – WAP-Debütant Nokia 7110", in c't - Magazin für Computertechnik 22/1999, S.122-123.

ŽIVADINOVIĆ, Dusan (1999a): "Schmalband-Surfen : Der PC bleibt zu Hause: Erste WAP-Dienste für mobiles Internet", in c't - Magazin für Computertechnik 22/1999, S.122-123.

ŽIVADINOVIĆ, Dusan (1999b): "Drahtlos anknüpfen : GPRS: schneller mobil surfen", in c't - Magazin für Computertechnik 7/1999, S.186-193.

Anhang

8 Anhang

8.1 Abbildungsverzeichnis

Abbildung 1: Vergleich der Übertragung von HTML und WML	9
Abbildung 2: Struktur von WML	10
Abbildung 3: 1.Karte	11
Abbildung 4: 2.Karte	11
Abbildung 5: Beziehungsverhältnisse von SGML, XML und deren Anwendungen	13
Abbildung 6: Sonderzeichen	17
Abbildung 7: Timer 1	19
Abbildung 8: Timer 2	19
Abbildung 9: Template 1	20
Abbildung 10: Template 2	20
Abbildung 11: Input 1	21
Abbildung 12: Input 2	21
Abbildung 13: Auswahl 1	22
Abbildung 14: Auswahl 2	22
Abbildung 15: Auswahl 3	23
Abbildung 16: Auswahl 4	23
Abbildung 17: Link	23
Abbildung 18: Grafik	24
Abbildung 19: Textformat 1	25
Abbildung 20: Textformat 2	25
Abbildung 21: Textformat 3	25
Abbildung 22: Ausrichtung	26
Abbildung 23: Umbruch	26
Abbildung 24: Kein Umbruch	26
Abbildung 25: Tabelle	27
Abbildung 26: Ausschnitt aus dem Nokia7110 Developer's Guide	27
Abbildung 27: Aktionen 1	28
Abbildung 28: Aktionen 2	28
Abbildung 29: Ereignis 1	29
Abbildung 30: Ereignis 2	29
Abbildung 31: Ereignis 3	29
Abbildung 32: refresh	30
Abbildung 33: noop	31
Abbildung 34: method get	32
Abbildung 35: method post	32
Abbildung 36: Variablen 1	33
Abbildung 37: Variablen 2	33
Abbildung 38: WMLScript 1	44
Abbildung 39: WMLScript 2	44
Abbildung 40: Confirm	45
Abbildung 41: Befragung von Handykunden zu mobilen Infodiensten (Quelle: Nokia)	46
Abbildung 42: Klassifikation des medialen Typs	53

Anhang

Abbildung 43: Klassifikation der Struktur.....	55
Abbildung 44: Klassifikation der User Interaktion.....	56
Abbildung 45: Beispiel 3D-Raum.....	57
Abbildung 46: Beispiel Überweisung.....	58
Abbildung 47: Überweisung 3D-Raum.....	59
Abbildung 48: Beispiel Nachrichten.....	59
Abbildung 49: Nachrichten 3D-Raum.....	60
Abbildung 50: Beispiel Fahrplan.....	61
Abbildung 51: Fahrplan 3D-Raum.....	61
Abbildung 52: Beispiel Portal.....	62
Abbildung 53: Portal 3D-Raum.....	63
Abbildung 54: Aktienkurse.....	63
Abbildung 55: Aktien 3D-Raum.....	64
Abbildung 56: Flugplan LH 1.....	73
Abbildung 57: Flugplan LH 2.....	73
Abbildung 58: Flugplan Lösung 1.....	74
Abbildung 59: Flugplan Lösung 2.....	74
Abbildung 60: Telefonbuch1.....	76
Abbildung 61: Telefonbuch2.....	76
Abbildung 62: Telefonbuch3.....	76
Abbildung 63: Beispiel Nachrichten.....	76
Abbildung 64: Lösungskonzept Überweisung 3D-Raum.....	79
Abbildung 65: Lösung Überweisung Card1.....	82
Abbildung 66: Lösung Überweisung Card2.....	82
Abbildung 67: Lösung Überweisung Card3.....	83
Abbildung 68: Lösung Überweisung Card4.....	84
Abbildung 69: Lösung Überweisung Card5.....	84
Abbildung 70: Lösung Überweisung Card6.....	85
Abbildung 71: Lösung Überweisung Template.....	85
Abbildung 72: Lösungskonzept Nachrichten 3D-Raum.....	86
Abbildung 73: Verlauf des Lösungskonzepts Nachrichten.....	89
Abbildung 74: Lösung Nachrichten Card1.....	89
Abbildung 75: Lösung Nachrichten Card2.....	90
Abbildung 76: Lösung Nachrichten Deck2_Card1.....	90
Abbildung 77: Lösung Nachrichten Card4.....	91
Abbildung 78: Lösung Nachrichten Template.....	91
Abbildung 79: Lösungskonzept Zugfahrplan 3D-Raum.....	92
Abbildung 80: Lösung Fahrplan: Aufteilung des Dokuments.....	93
Abbildung 81: Verlauf des Lösungskonzepts Fahrplan.....	94
Abbildung 82: Lösung Fahrplan Deck1 Card1.....	95
Abbildung 83: Lösung Fahrplan Auswahl1.....	95
Abbildung 84: Lösung Fahrplan Auswahl2.....	95
Abbildung 85: Lösung Fahrplan Deck1 Card2.....	96
Abbildung 86: Lösung Fahrplan Deck4 Card1.....	96
Abbildung 87: Lösung Fahrplan Deck4 Card2.....	97
Abbildung 88: Lösung Fahrplan Deck4 Card4.....	97
Abbildung 89: Lösungskonzept Portal 3D-Raum.....	98
Abbildung 90: Verlauf des Lösungskonzepts Portal.....	99

Anhang

Abbildung 91: Lösung Portal Deck1 Card1	99
Abbildung 92: Lösung Portal Deck2 Card2	100
Abbildung 93: Lösung Portal Suchbeispiel	100
Abbildung 94: Lösungskonzept Aktienkurse 3D-Raum	101
Abbildung 95: Verlauf des Lösungskonzepts Aktien	102
Abbildung 96: Lösung Aktien Deck1 Card1	103
Abbildung 97: Lösung Aktien Card1 Auswahl	103
Abbildung 98: Lösung Aktien Einzelkurs	104
Abbildung 99: Lösung Aktien Chart	104
Abbildung 100: Online-Auktion Aufteilung in Decks und Cards	108
Abbildung 101: Online-Auktion WML-Seiten	114
Abbildung 102: Verlauf der M-Commerce-Kurve (Quelle: Durlacher)	115

8.2 Tabellenverzeichnis

Tabelle 1: Sonderzeichen in WML	16
Tabelle 2: Reservierte Werte für das Attribut "type" des Elements "do"	28
Tabelle 3: Anwendungen im Bereich Informationsverteilung	48
Tabelle 4: Anwendungen im Bereich Auskunftssysteme	48
Tabelle 5: Anwendungen im Bereich interaktive Dienste	49
Tabelle 6: Mobilfunknetzbetreiber und ihre Portal-Kooperationspartner	51
Tabelle 7: WAP-Portale	52

8.3 Beispielverzeichnis

Beispiel 1: Einfaches WML-Beispiel	11
Beispiel 2: HTML vs. WML – Elemente schließen	14
Beispiel 3: HTML vs. WML – leere Elemente schließen	14
Beispiel 4: HTML vs. WML – Groß- und Kleinschreibung beachten	15
Beispiel 5: HTML vs. WML – Attribute in Anführungszeichen	16
Beispiel 6: Sonderzeichen in WML	17
Beispiel 7: Timer in WML	19
Beispiel 8: Template in WML	20
Beispiel 9: Benutzereingaben (input)	21
Beispiel 10: Benutzereingaben (select-mehrfach)	22
Beispiel 11: Benutzereingaben (select-einfach)	23
Beispiel 12: Verlinkung	23
Beispiel 13: Einbinden von Grafiken	24
Beispiel 14: Textformatierung	25
Beispiel 15: Textausrichtung	26
Beispiel 16: Einsatz von Tabellen	27
Beispiel 17: Element "do"	28

Anhang

Beispiel 18: Ereignisse in WML	29
Beispiel 19: Aktion refresh	30
Beispiel 20: Ausschalten von Templatefunktionen	31
Beispiel 21: Methode "get" beim Element "go"	32
Beispiel 22: Methode "post" beim Element "go"	32
Beispiel 23: Definition von Variablen	33
Beispiel 24: WMLScript – Kommentare	37
Beispiel 25: WMLScript - Definition von Variablen	38
Beispiel 26: WMLScript – Datentypen	39
Beispiel 27: WMLScript – Funktionen	40
Beispiel 28: WMLScript – Funktionen aufrufen	41
Beispiel 29: WMLScript – Lang-Bibliothek	42
Beispiel 30: WMLScript – Float-Bibliothek	42
Beispiel 31: WMLScript – String-Bibliothek	43
Beispiel 32: WMLScript – URL-Bibliothek	43
Beispiel 33: WMLScript – WMLBrowser-Bibliothek	44
<i>Beispiel 34: WMLScript – Dialogs-Bibliothek</i>	<i>45</i>