

Fachhochschule Wedel
University of Applied Sciences Wedel

Seminararbeit
in der Fachrichtung Wirtschaftsinformatik

WAP

Boris von Chrzanowski
wi5496
6. Fachsemester

November 2004

Inhaltsverzeichnis

1	Vorwort	3
2	WAP Gateway	4
3	Architektur	6
3.1	Trägerdienste	6
3.2	Transportschicht	7
3.3	Sicherheitsschicht	7
3.4	Transaktionsschicht	7
3.5	Sitzungsschicht	8
3.6	Anwendungsschicht	8
4	WAP Services	9
4.1	WAP PUSH	9
4.2	User Agent Profile / UAProf	10
4.3	Wireless Telephony Application	11
5	Content-Authoring-Sprachen	13
5.1	WML	13
5.2	WMLScript	14
5.3	XHTMLMP	14
6	Ausblick	16

1 Vorwort

Zum momentanen Zeitpunkt macht das Wireless Application Protocol, kurz WAP genannt, einen Spagat zwischen den Versionen WAP 1.x und WAP 2.0. Zwar ist der Standard 2.0 Mitte des Jahres 2002 verabschiedet worden, doch beginnt die Unterstützung auf Clientseite erst allmählich. Nur wenige Mobiltelefone unterstützten die neuste Spezifikation vollständig. Zwar wird häufig mit der Nutzbarkeit von WAP 2.0 geworben, doch wird vielfach nur die neue Seitenbeschreibungssprache (siehe Kapitel 5.1) interpretieren und nicht die neue Übertragungstechnik eingesetzt.

WAP im allgemeinen ist eine Sammlung von Technologien und Protokollen, um Datenkommunikation auf schmalbandigen Clients verfügbar zu machen. Dies ist bis heute wichtig und wird erst mit schnelleren Übertragungstechniken an Gewicht verlieren. Der erste WAP Standard wurde 1998 von einer Vereinigung von Hardwareherstellern, dem WapForum, mit der Release 1.0 geschaffen. Es folgten die Versionen 1.1 und 1.2, bis im Jahr 2002 die aktuelle Version 2.0 verabschiedet wurde. Das WapForum benannte sich im selben Jahr in die open mobile alliance um und hat aktuell ca. 200 Mitglieder.

Die vorliegende Arbeit bezieht sich auf WAP 2.0. Allerdings ist diese Version vollständig abwärts kompatibel und deshalb wird ebenfalls ausführlich auf die WAP V1, hauptsächlich Release 1.2, eingegangen. Dies ist sinnvoll, da ein schneller Wechsel von WAP V1 auf 2.0, zwar absehbar ist, aber noch nicht begonnen hat.

In den folgenden Kapiteln wird dem interessierten und informationstechnisch vorgebildeten Leser ein Überblick über die WAP Technologie gegeben. Die ersten Kapitel beschreiben dabei hauptsächlich die notwendige Hard- und Softwarearchitektur. Diese Punkte legen den Grundstein für das Verständnis der letzten Kapitel, in denen es um die Anwendung von WAP geht. Sinn dieser Arbeit ist es nicht, die Content-Authoring-Sprachen zu beschreiben, würde dies den Rahmen sprengen, sondern vielmehr dem Leser Sprachen und Funktionen der höheren WAP Layer näher zubringen, so dass er sich gezielt bei Bedarf weiterbilden kann.

2 WAP Gateway

Mobile WAP Netze entfalten ihre gesamte Funktionalität über die mobilen Endgeräte und über das WAP Gateway. Anders als bei der bekannten WWW/HTML Internetkommunikation, bei der der Client direkt auf den Server zugreift, wird in der WAP Kommunikation ein Gateway zwischen WAP Client und Server geschaltet. In WAP V1 ist dieses meist zwingend erforderlich, da Internetserver weder den WAP Protokollstack implementierten, noch die unterschiedlichen WAP Trägerdienste unterstützen. Das Gateway verbindet somit das mobile Netz mit dem Internet, indem es Anfragen aus dem mobilen Netz mit dem WAP Protokollstack annimmt und per Internetprotokoll an den Web-Server weiterleitet. Die Antwort nimmt dann den umgekehrten Weg. In WAP 2.0 ist das WAP Gateway für normale Anfragen obsolet, da die neuste Version die HTML/1.1 Kommunikation beherrscht und somit die Protokollstacks von Webserver und mobilem Endgerät kompatibel sind. Ein WAP Gateway in Version 2.0 ist nur noch für WAP Push und andere Enhanced Features nötig.

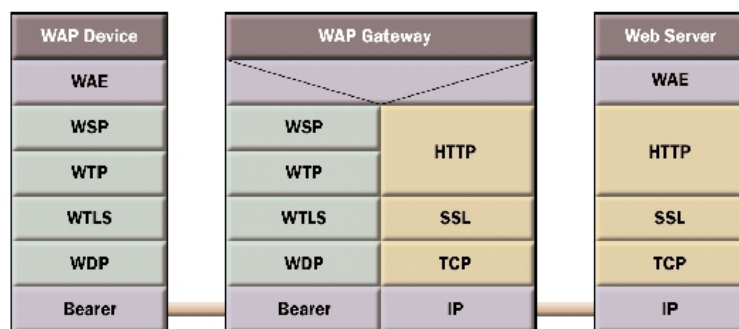


Abbildung 2.1: WAP Gateway für WAP 1.x

Das klassische WAP Gateway besteht aus zwei essenziellen Elementen: Dem Protokoll Gateway, das die direkte Umwandlung der Protokolle gewährleistet und dem Coder/Decoder (CODEC), mit dem die verschickten Daten umcodiert werden (siehe Abbildung 3.2). Mittels des CODEC wird WML und WMLScript in ein kompakteres Binärformat kompiliert.

Allerdings werden WAP Gateways nicht zwangsläufig mit WAP 2.0 verschwinden, denn sie können auch hier den Kommunikationsprozess optimieren. So können Daten zwischengespeichert werden (WAP Proxy statt WAP Gateway) und es können location, privacy und presence based services angeboten werden.

WAP Gateways werden hauptsächlich von den Mobilfunknetzbetreibern angeboten. Da der Besitzer des Gateways aber sämtliche Informationen mitprotokollieren kann, ist es besonders für sicherheitskritische Anwendungen, wie dem Onlinebanking, ratsam, ein eigenes WAP Gateway zu betreiben. Dies liegt vor allem daran, dass bei WAP V1 die Verbindung ins Internet mit dem SSL Protokoll und die Verbindung in das Mobilnetz mit WTLS kodiert wird. Dieser Transfer bricht aber die Sicherheitskette.

Neben kommerziellen WAP Gateway Produkten, wie z.B. von Nokia, gibt es auch KANNEL, eine Open Source Lösung für Linux.

3 Architektur

WAP 2.0 implementiert sowohl den HTTP Protokollstack aus dem Internet, wie auch den WAP V1 Stack. Grundlegendes Wissen über die im Internet benutzten Layer wird in dieser Arbeit vorausgesetzt. Ein schematischer Vergleich zwischen WAP V1 und Internetprotokollen ist aber in Abbildung 3.1 gegeben. Somit wird in den folgenden Erläuterungen hauptsächlich auf die sechs Schichten von WAP V1 eingegangen und nur am Rande auf die WAP 2.0 Layer.

Anwendungen müssen nicht den gesamten Protokollstapel verwenden, sondern können ihn auf die jeweils gebrauchten Schichten einschränken. So ist es z.B. möglich, die Sicherheitschicht zu überspringen, wenn die entsprechende Funktionalität nicht benötigt wird.

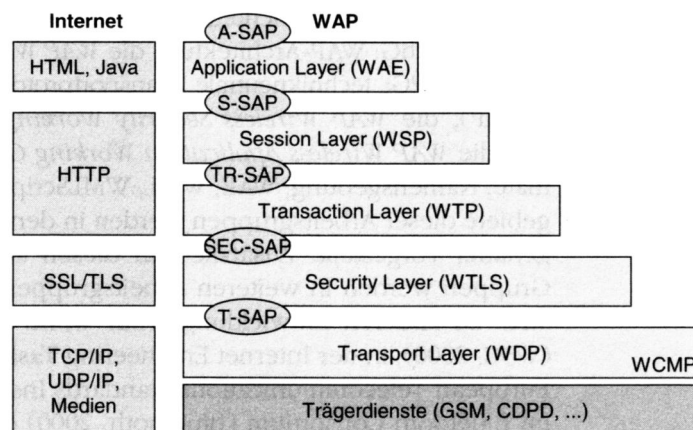


Abbildung 3.1: Die WAP 1.x Layer mit den Service Access Points (SAP).

3.1 Trägerdienste

Die unterste Schicht wird aus unterschiedlichen Trägerdiensten (Bearer Services) gebildet. Dabei liegt der Übertragung kein bestimmter Dienst zugrunde, statt dessen sind sowohl nachrichtenvermittelte Dienste (z.B. SMS) denkbar, wie auch leistungsvermittelte (z.B. HSCSD)

oder packetvermittelte (z.B. GPRS). Diese unterschiedlichen Möglichkeiten haben dazu geführt, dass keine einheitliche Schnittstelle für die unterste WAP Architektur Schicht definiert wurde. Auf die Trägerdienste setzt WAP also nur auf.

3.2 Transportschicht

Die Transportschicht stellt einen datagrammorientierten Übertragungsdienst bereit und bietet damit die erste einheitliche Schnittstelle in der WAP Architektur. WDP (Wireless Datagram Protocol) ist dabei das nutzdatentransportierende Protokoll und ist für die Adressierung via Ports, die Fragmentierung und Reassemblierung der Daten zuständig. WCMP (Wireless Control Message Protocol) ist mit ICMP zu vergleichen, wie dieses dient es der Fehlerübermittlung. WAP 2.0 führt hier UDP/IP ein.

3.3 Sicherheitsschicht

WTLS ist eine für die drahtlose Kommunikation optimierte Version von TLS (Transport Security Layer) und bietet Datenintegrität, Verschlüsselung, Authentifizierung und einen Schutz gegen Denial-of-Service-Attacks an. Allerdings sichert WTLS nur die Verbindung zwischen mobilem Endgerät und WAP Gateway. Weiterführende Kommunikation vom WAP Gateway auf Internetserver sind nicht abgedeckt, so dass in vielen Fällen eine Verschlüsselung auf Anwendungsebene sinnvoll ist. In WAP 2.0 wird diese Schicht durch TLS (Transport Layer Security) ersetzt.

3.4 Transaktionsschicht

Das WTP (Wireless Transaction Protocol) bietet einen Transaktionsdienst an, also die Möglichkeit interaktiv mit Anwendungen kommunizieren zu können. Eine Transaktion besteht dabei aus den Komponenten Anfrage und Antwort. Im WTP gibt es drei Klassen von Transaktionsdiensten, zwei zuverlässige und eine unzuverlässige. Die Dienste unterscheiden sich in der unterschiedlich häufigen Quittierung von Anfragen. In WAP 2.0 befindet sich an der Stelle von Transaktions- und Sitzungsschicht die WP-HTML (Wireless Profiled HTML) Schicht. Sie ist mit HTML/1.1 vollkommen kompatibel, unterstützt aber Nachrichtenkompression.

3.5 Sitzungsschicht

Sitzungen stellen einen gemeinsamen Zustand zwischen Server und Client her. Damit sind Optimierungen in der Übertragung möglich, indem z.B. Fähigkeiten ausgehandelt werden. Zudem werden Funktionalitäten wie das Parken, Wiederaufnehmen und das Migrieren von Sitzungen bereitgestellt. Es gibt sowohl einen verbindungsorientierten Sitzungsdienst, der auf WTP zugreift, als auch einen verbindungslosen Sitzungsdienst, der auf WDP aufsetzt. Ein speziell auf die Web-Nutzung optimiertes Sitzungsprotokoll baut auf WSP (Wireless Session Protocol) auf und nennt sich WSP/B (Browsing). Es erweitert den allgemeinen Standard um z.B. den PUSH Datentransfer.

3.6 Anwendungsschicht

Der Teil, den Entwickler für WAP Anwendungen nutzen, ist der WAE (Wireless Application Environment) Layer. Diese Schicht besteht aus dem WAE- und WTA-User-Agent (siehe Abbildung 3.2). Während die Aufgabe des ersten darin besteht, WML (Wireless Markup Language) und WMLScript (Wireless Markup Language Script) im Microbrowser darzustellen, erweitert der WTA User Agent WAP um Telefonfunktionen.

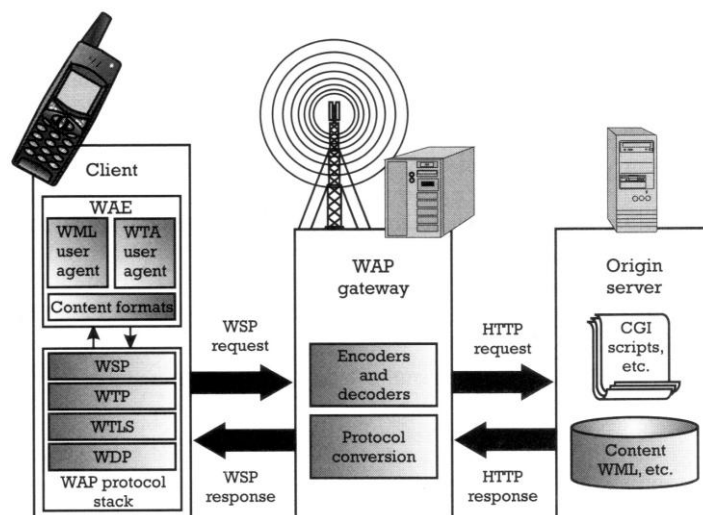


Abbildung 3.2: die beiden User Agents der Anwendungsschicht

4 WAP Services

Im folgenden sollen WAP Technologien erläutert werden. Dabei wird vor allem auf die von der open mobile alliance definierte WAP PUSH Datenkommunikation eingegangen. Zusätzlich wird eine Auswahl anderer Services kurz vorgestellt. Vielfach werden die dahinterstehenden Standards von der open mobile alliance nur angewendet und nicht definiert. Die Ratifizierung geschieht z.B. durch das W3C. Dementsprechend werden diese Techniken auch außerhalb von WAP eingesetzt und bedürfen bei weiterführendem Interesse einer eigenständigen Analyse.

4.1 WAP PUSH

Mit WAP Version 1.2 begann eine Unterstützung für PUSH Funktionalität. PUSH Übertragungen zeichnen sich dadurch aus, dass Informationen an den Benutzer versandt werden, ohne dass dieser eine explizite Anforderung stellen muss. Im bekannten Internet gibt es bisher kein verbreitetes PUSH Verfahren, statt dessen wird es durch Polling, also dem fortwährenden Nachfragen (PULL Verfahren), simuliert. Dabei kann das PUSH Verfahren in vielen Fällen die erheblich bessere Lösung darstellen, denn es ist datenreduzierend und damit ressourcenschonend. Anders, als bei der herkömmlichen Internetkommunikation, wo PUSH Verfahren an nicht ständig erreichbaren Clients scheitert, gibt es dieses Problem im WAP nicht. Hier nimmt

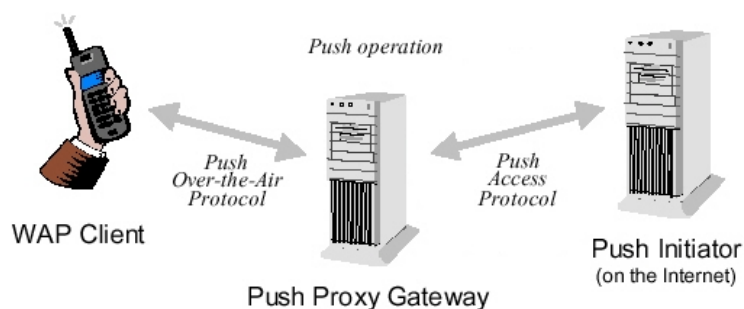


Abbildung 4.1: Push Proxy Gateway verbindet mobiles mit stationärem Internet

das Push Proxy Gateway die Daten an. Dabei beginnt der Initiator, z.B. der Web-Server, Kontakt über das Push Access Protocol (PAP) mit dem Push Proxy Gateway aufzunehmen (siehe Abbildung 4.1). Das Push Access Protocol transportiert eine XML Nachricht, die Steuerinformationen für das Push Proxy Gateway enthält, und eine Datei mit den eigentlichen Nutzdaten in WML oder XHTML.

Die XML Datei mit den Steuerungsanweisungen löst auf dem Push Proxy Gateway individuelle, vom Initiator bestimmte, Prozesse aus. So ist es z.B. möglich, die Haltbarkeit der Daten festzulegen, oder der Initiator wird über den Nachrichtenstatus informiert. Diese Statusreports spielen besonders bei Store-and-Forward Implementierungen eine Rolle, wenn das Push Proxy Gateway Nachrichten wegen Nichterreichbarkeit oder Kapazitätsengpässen des Mobilendgeräts zwischenspeichert. Die weiterführende Kommunikation zwischen Push Proxy Gateway und mobilem Endgerät erfolgt mit dem Push Over The Air Protocol, entweder verbindungslos oder -orientiert. Bei verbindungsorientierten Sitzungen muss eben eine solche Sitzung eingerichtet werden. Diese kann aber nur vom Endgerät initialisiert werden, so dass auf der Seite des Endgeräts eine Session Initiation Application (SIA) auf PUSH Verbindungen wartet, um eine Session aufzubauen. Besitzt eine Session, weißt der Dispatcher anhand der Application ID dem empfangen Content die dazu passende Anwendung zu. Eventuell hat der Client dem Push Proxy Gateway auch schon alle unterstützten Applications bekannt gegeben, so dass im Gateway nicht unterstützte PUSHs herausgefiltert werden können. Bei verbindungslosem PUSH muss der Client dies selbst tun. Besonders bei PUSH Diensten ist die Sicherheit zu beachten, denn ähnlich wie bei email und SMS drohen ungewünschte Verbindungsversuche. Dieses zu unterdrücken obliegt dem Push Proxy Gateway, das damit aber sogleich auch zum Contentfilter wird.

4.2 User Agent Profile / UAProf

Durch die hohe Diversifizierung an unterschiedlichsten mobilen Geräten ist es nicht möglich immer eine identische Antwort auf Requests zu liefern. Statt dessen müssen die Daten so an Ein- und Ausgabefunktionen des Clienten angepasst werden, dass dieser sie anzeigen und ggf. speichern kann. Dieses Dilemma wurde durch das W3C mittels des Composite Capability/Preference Profile Framework (CC/PP) gelöst. CC/PP beschreiben das entsprechende Gerät durch eine Menge von Attribute/Werte-Paaren. Ab WAP 1.2 werden diese User-Agent-Profil-Informationen auch Capability and Preference Information (CPI) genannt. Sie

werden vom Client über die verbindenden Netzpunkte bis hin zum Server mittels einer End-To-End-Verbindung übertragen. Die zwischen Client und Server liegenden Netzwerkpunkte können die Informationen noch anreichern. Die angeforderten Daten können dann, anhand der übermittelten Informationen, angepasst und versandt werden. Aus Datenschutzgründen behält letztendlich aber der User die Kontrolle über die CPI.

4.3 Wireless Telephony Application

Mittels WTA können Telefonfunktionen über das Internet aufgerufen werden. Das Framework teilt sich in drei entscheidende Bereiche, dem WTAI (wireless telephony application interface), dem WTA User Agent und dem Man Machine Interface. Der WTA User Agent bietet einen standardisierten Zugriff auf das herstellereigene Man Machine Interface, also den Telefonfunktionen. Die WTAI sind die entsprechenden Zugriffsmechanismen, um die Telefonfunktionen nutzen zu können.

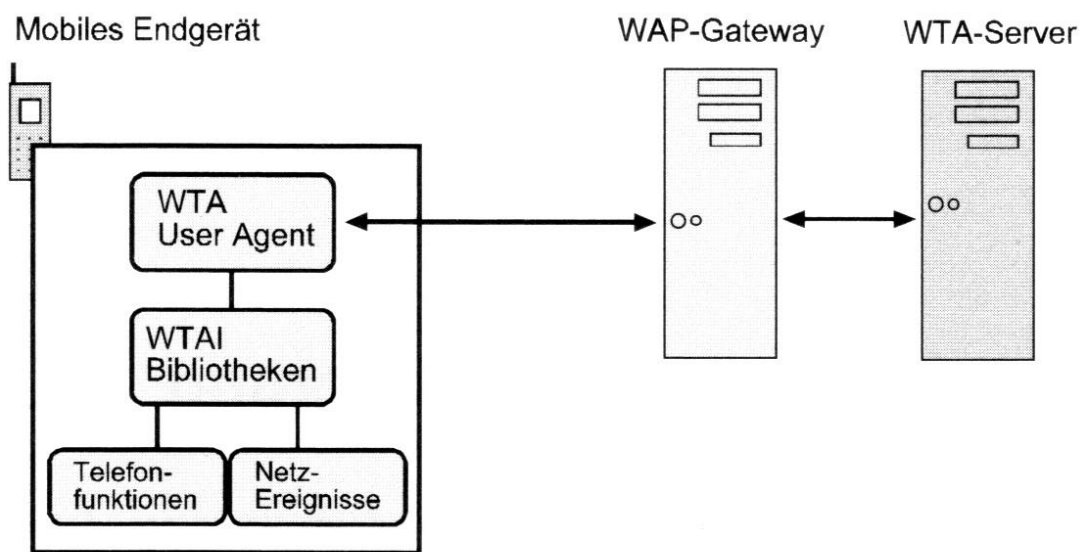


Abbildung 4.2: Eine einfache WTA Architektur

Das untenstehende Beispiel tätigt einen Anruf mittels der WTAI Public Library, wenn der Benutzer dem Link folgt. Daneben gibt es Befehle für das generelle Anrufhandling, die Mailbox, das Telefonbuch, Netzwerknachrichten und Sonstiges.

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//WAPFORUM//DTD XHTML Mobile 1.0//EN"
    "http://www.wapforum.org/DTD/xhtml-mobile10.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en">
  <head>
    <title> FH-Wedel anrufen </title>
  </head>

  <body>
    <p>
      <a href="wtai://wp/mc;0049410380480">FH-Wedel anrufen</a>
    </p>
  </body>

</html>
```

5 Content-Authoring-Sprachen

5.1 WML

Bei WML (Wireless Markup Language) handelt es sich um die Seitenbeschreibungssprache von WAP V1. In der Literatur wird WML häufig mit HTML verglichen. Dieser Vergleich ist aber nur oberflächlich stimmig, lehnt sich WML doch an XML (extensible markup language) an. Bis auf einige, z.T. gleich lautende Tags, haben HTML und WML nur wenig gemeinsam, auch wenn ab WML 1.1 die zweite Basis, auf der WML ruhte, HDML (Handheld Device Markup Language), vielfach deprecated ist. Somit ist WML auch nicht wie HTML bzw. HTML Browser kooperativ und vergebend. WML benötigt vielmehr eine DTD (Document Type Definition), ist casesensitiv und unterliegt im Allgemeinen den XML Vorschriften. So wird verlangt, dass Ausgabertext in Textabsätze gebunden ist. Zudem sind alleinstehende Tags, wie in HTML der Zeilensprung `
`, verboten. Stattdessen schreibt man in WML `
</br>`, bzw. nutzt die in diesem Fall mögliche Abkürzung `
`.

Die kleinstmögliche Einheit eines WML Dokuments, die komplett an das mobile Endgerät übermittelt werden kann, nennt sich Deck. Ein Deck wiederum besteht aus mehreren Cards, den eigentlichen WAP Seiten. Diese Konzeption soll eine frameähnliche Informationsteilung auf verschiedene Seiten gewährleisten und gleichzeitig die Zugriffsanzahl auf den Web-Server senken. Letzterer Punkt steigert die Performance, denn der Verbindungsaufbau kostet Zeit.

Neben Tags für Texte, Graphiken, Tabellen und Verlinkung, gibt es die Möglichkeit Formulare aufzubauen und so mit dem Benutzer in Interaktion zu treten. Formularelemente sind Textfelder und Auswahllisten, zudem können eingeschränkt Ereignisse und Variablen definiert werden.

Im folgenden Beispiel sind in einem Deck zwei Cards definiert. Ruft ein Benutzer dieses WML Dokument auf, so wird ihm „card1“ angezeigt. „card2“ kann direkt über die URL aufgerufen werden oder müsste aus „card1“ verlinkt sein.

```
<?xml version="1.0"?>
<DOCTYPE wml PUBLIC "-//WAPFORUM//DTD 1.2//EN"
" http://www.wapforum.org/wml12.dtd">

<wml>
    <card id="card1">
        <p>card1 </p>
    </card>

    <card id="card2">
        <p>card2 </p>
    </card>
</wml>
```

5.2 WMLScript

Auch wenn WML einige eingeschränkte Interaktion mit dem Benutzer zulässt, so sind die Möglichkeiten jedoch nicht ausreichend. WMLScript erweitert an dieser Stelle den Funktionsumfang. WMLScript und JavaScript liegt der selbe Standard, in Form von ECMAScript, zugrunde. Aber anders als bei JavaScript liegt der Code für WMLScript in einzelnen Dateien und wird über das WAP Gateway getrennt angefordert, übersetzt und sodann an das mobile Endgerät geschickt. WMLScript kann man als eine nicht streng getypte Sprache ansehen. Es gibt Funktionen, Variablen, Operatoren, Kontrollstrukturen und Bibliotheken.

5.3 XHTMLMP

In der Release WAP 2.0 sollten mobiles und stationäres Internet weiter verschmelzen. Den Grundbaustein legt eine gemeinsame Seitenbeschreibungssprache, XHTML 1.0 (eXtensible

Hypertext Markup Language), die vom W3C als Nachfolger von HTML 4.01 vorgestellt wurde. XHTML bietet Konformität mit XML Vorgaben und ist auf verschiedenen Plattformen anzeigbar. Das W3C gab auch den XHTML-Basic-Dokumententyp heraus, der alle die Module einschließt, die mindestens erforderlich sind, um ein Dokumenttyp der XHTML-Muttersprache zu sein. Die XHTML Basic Unterstützung ist somit unabdingbar für jedes mobile XHTML Endgerät. Das WapForum erweiterte XHTML Basic durch einige zusätzliche Elemente und Attribute, wie interne Style Sheets. Dieses sogenannte XHTMLMP (XHTML Mobile Profile) ist die Content-Authoring-Sprache für WAP 2.0 (siehe Abbildung 5.1).

Eine Scripting Möglichkeit bietet XHTMLMP selbst nicht an und es wird auch kein direkter Aufruf einer externen Scriptsprache unterstützt. So ist es weiterhin nötig WML und WMLScript zu verwenden. Als erstes wird von einem XHTMLMP- auf ein WML Dokument verlinkt, von dem aus das WMLScript aufgerufen werden kann. Als generelles Beispiel eines XHTMLMP Dokuments möge der Quellcode auf Seite 12 dienen.

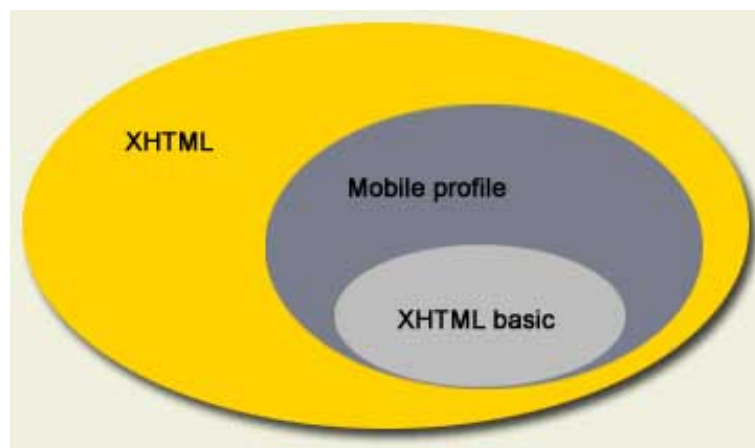


Abbildung 5.1: XHTML Mobile Profile erweitert zwar XHTML-Basic, bleibt aber eine eingeschränkte XHTML Version.

6 Ausblick

Der WAP V1 Standard ist durch einen bestimmten und zielorientierten Aufbau geprägt. Das WapForum hat viele Bausteine entwickelt, um den drei grundlegenden Problemen bei der Datenübertragung mittels des Mobilfunknetzes entgegenzutreten. Geringe Übertragungskapazität wird mit Kompression besser genutzt, lange Antwortzeiten werden vor dem Benutzer mit dem Card/Deck Konzept verborgen und WML ermöglicht eine Informationsdarstellung auf allen (simplen) Endgeräten.

Mit WAP 2.0 macht die open mobile alliance zwar den logisch richtigen Schritt, doch wird der in sich geschlossene Standard von WAP V1 damit auch durchbrochen. Dies hat zur Folge, dass das gute Zusammenspiel der einzelnen Bausteine nicht mehr gewährleistet ist. WAP 2.0 kann somit vor allem als Wegweiser dienen.

Die Wegweiser Funktion von WAP 2.0 ist aber auch sehr wichtig, weil der eigentliche Gedanke bei der Entwicklung von WAP bisher zwar technisch gelöst wurde, sich in der Praxis aber nicht durchgesetzt hat.

Internet auf Endgeräten des Mobilfunknetzes erfolgreich bereitzustellen, wird in Zukunft nur dann gelingen, wenn Internet und WAP weiter verschmelzen.

Literaturverzeichnis

- Forta, Ben:** WAP, WML und WMLScript. 1. Auflage. Muenchen: Markt + Technik Verlag, 2001
- Forum, WAP:** Wireless Application Protocol WAP 2.0 Technical White Paper. La Jolla, Jan. 2002 – Technischer Bericht
- Heijden, Marcel van der und Taylor, Marcus:** Understanding WAP: Wireless Applications, Devices, and Services. 1. Auflage. Norwood: Artech House Publisher, 2000
- kannel:** Kannel: Open Source WAP and SMS gateway. <http://www.kannel.org>, 2004
- Musciano, Chuck und Kennedy, Bill:** HTML & XHTML Das umfassende Referenzwerk. 4. Auflage. Koeln: O'Reilly, 2003
- Pauler, Herbert:** Synchronisation von Organizer und PC. Funkschau, 13 2000, 52–54
- Roewekamp, Lars:** Drahtlos die Dritte. <http://www.heise.de/ix/artikel/2000/05/104/>, Mai 2000
- Schiller, Jochen:** Mobilkommunikation Techniken fuer das allgegenwaertige Internet. 1. Auflage. Muenchen: Addison-Wesley, 2000
- Skelton, Dr. Gordon W.:** Wireless Application Development. 1. Auflage. Boston: Thomson, 2003
- Wenz, Christian und Hauser, Tobias:** WAP Architektur - Programmierung - Referenz. 1. Auflage. Muenchen: Carl Hanser Verlag, 2001