

Ubiquitous Computing (Ubiquitäre Informationstechnologien)

Vorlesung im WS 00/01

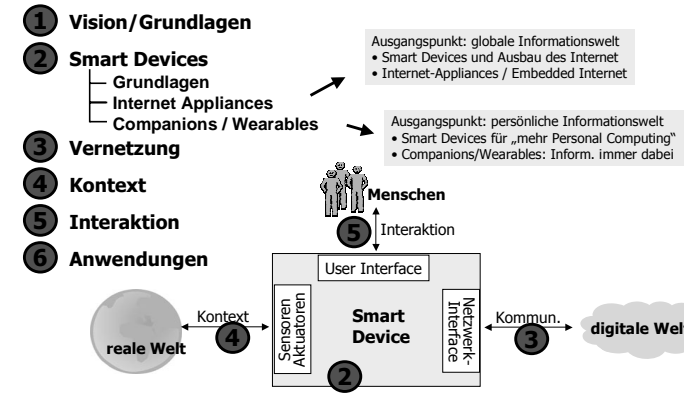
Hans-Werner Gellersen



Universität Karlsruhe
Institut für Telematik
Telecooperation Office

HWG 5-1

Ubiquitous Computing



HWG 5-2

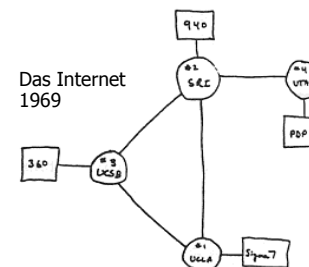
Smart Devices: Internet-Appliances

- Einführung
- Mobiler Informationszugriff
- WAP: Wireless Application Protocol
- WebSIM: Web Server im Mobiltelefon
- CoolTown: "Serving the Real World"

HWG 5-3

Einführung

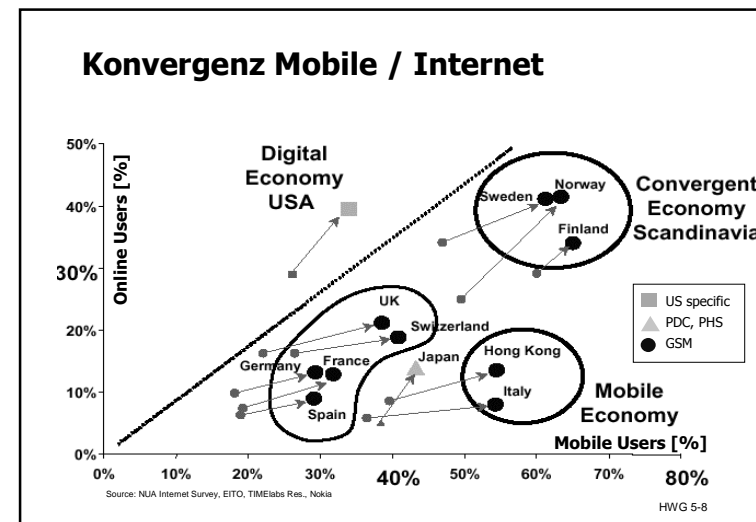
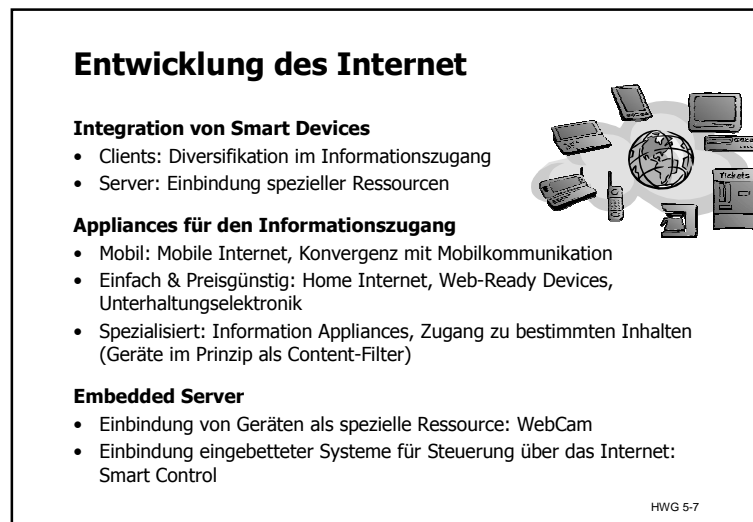
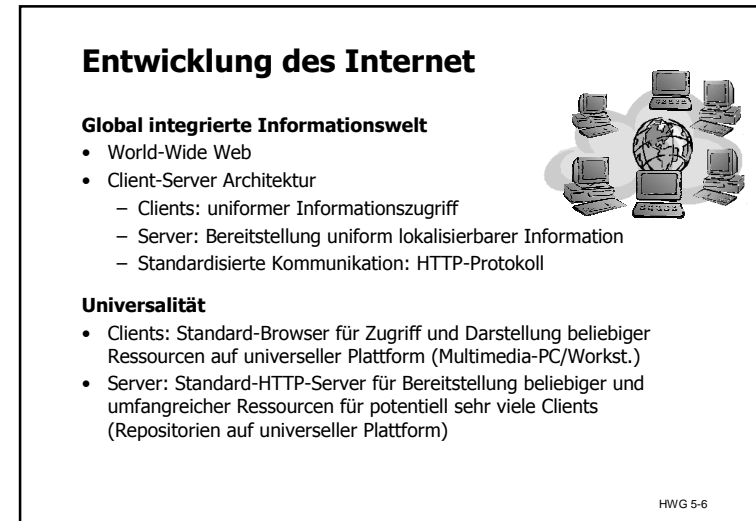
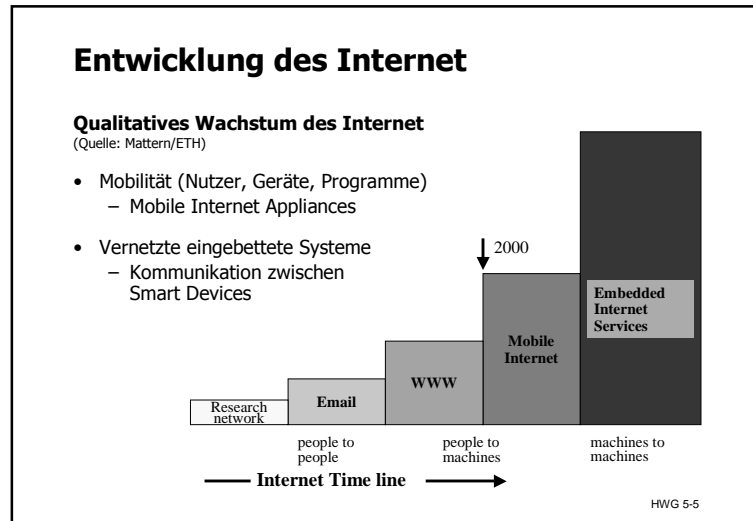
Entwicklung des Internet



- 70er Jahre
- Pionierarbeiten
- 80er Jahre
- Forschungsnetz
- 90er Jahre
- Kommerzialisierung und Popularisierung
 - Vernetzung aller Computer

- weitere Entwicklung
- Vernetzung aller Geräte ?
 - Vernetzung aller Dinge ?

HWG 5-4



Appliances als Content-Filter

Beispiel 1: Magic Medicine Cabinet (Digitale Hausapotheke)

(Video, Andersen Consulting)

- Filter für Gesundheits-Information
 - USA: 2/3 der User benutzen das Internet zur Suche nach Gesundheits-Information
 - „Situated Portal“: Tor zur Gesundheitsinformation, aber an Ort und Situation gebunden
- Appliance-Gedanke: Informationsdienst eingebettet in Aktivität
 - Informationszugriff bei alltäglicher Gesundheitspflege
 - „Integration of physical and informational healthcare“

Beispiel 2: iM Tuner

- Empfangsgerät für Internet Radio
- Implementiert als Fernsteuerung für PC & Hifi



Smart Devices: Internet-Appliances

- Einführung
- **Mobiler Informationszugriff**
- **WAP: Wireless Application Protocol**
- **WebSIM: Web Server im Mobiltelefon**
- **CoolTown: "Serving the Real World"**

HWG 5-10

Mobiler Informationszugriff

Information auf mobilen Geräte

- begrenzte Rechenleistung und Speicherkapazität schränkt Verarbeitung von Information ein
- begrenzte Darstellungsressourcen: kleine Bildschirmfläche, Anzahl Pixel, und Farbtiefe
- begrenzte Interaktionsressourcen: z.B. für Auswahl, für Navigation, für Texteingabe
- begrenzte Bandbreite
- fehleranfällige Kommunikation

Anbindung an globale Informationswelt

- erfordert Anpassung von Information
 - wo? (im Client, beim Server, oder dazwischen)
 - wie? (Filterung, Transcoding, Umstrukturierung)

HWG 5-11

Anpassung von Information

Filterung/Anpassung im Client

- transparenter Zugriff, aber ineffizient (z.B. Übertragung von mehr Information, als dargestellt werden kann)

Anpassung im Server: Content Negotiation

- Client gibt im HTTP Header an, was er verarbeiten kann (Header Fileds wie `Accept`, `Accept-Encoding`,...)
- Server nutzt Kenntnis des Browsers/Client-Systems (z.B. `User Agent: Pocketweb/2.3b ...`)

Anpassung in einer Zwischeninstanz

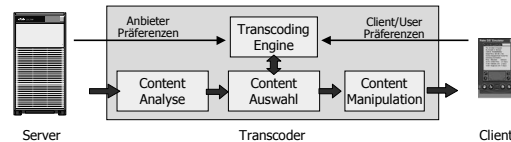
- Gateway zwischen Mobilnetz und Internet: universelles Transcoding für alle Clients im Mobilnetz
- Proxy: Stellvertreter-Instanz eines mobilen Clients

HWG 5-12

Anpassung von Information

Transcoding

- Automatische Überführung von Information in ein anderes Format / eine andere Darstellung



Dimensionen der Anpassung

- Modalität: Überführung von einer Darstellungsmodalität in eine andere (z.B. video-to-image, text-to-speech, table-to-list, image-to-url...)
- Qualität: innerhalb einer Modalität Skalierung der Qualität (z.B. Quantis. von Frequenzanteilen, image-to-bitmap, text-to-keywords, ...)

HWG 5-13

Anpassung von Information

Beispiel PocketWeb

- erster Handheld Web Browser, 1994
 - Architektur: Client - Proxy - Server
 - Client-Proxy-Komm.: HTTP über GSM
- Transcoding im Proxy
 - Filterung nicht-darstellbarer Elemente
 - Umformatierung/Skalierung von Bildern
 - Ersetzung von Tabellen, Frames



Weitergehende Anpassung von Inhalten

- Bewertungsabhängige Anpassung
z.B. Ignorieren „unwichtiger“ Bilder (z.B. grafische Bullets)
- Anpassung der Dokumenten-Struktur
z.B. Aufteilung großer Dokumente in kleinere, verknüpft durch Links
- Einfügen von Elementen für einfachere Navigatiomm
z.B. IVZ am Dokum.-Anfang: Liste mit Links zu Header-Elementen

HWG 5-14

Anpassung von Information

Beispiel WEST

- „Web Browser for Small Terminals“, Viktoria Institute Göteborg, 1999
- Architektur: Client - Proxy - Server

Inhaltsaufbereitung im Proxy

- „Chunking“: Aufteilung von Web-Seiten in Karten (max. 7 pro Ebene, ggf. hierarchisch)
- Erzeugung von Thumbnails für Karten
- Extraktion von Links, Keywords

Flip Zooming Navigation auf Client

- Fokus+Context Visualisierung von Web-Seiten: eine Karte im Fokus, die anderen als Kontext
- Flip... : Lineare Navigation durch Karten
- ...Zooming : Auswahl von Karte in Fokus für vollst. Anzeige



HWG 5-15

Smart Devices: Internet-Appliances

- Einführung
- Mobiler Informationszugriff
- WAP: Wireless Application Protocol
- WebSIM: Web Server im Mobiltelefon
- CoolTown: "Serving the Real World"

HWG 5-16

Wireless Application Protocol

Mobiltelefone

- Klein, portabel, persönlich
- Autonom und allgegenwärtig vernetzt
- multimodal (Sprache, Text, Grafik)
 - ➔ ideale Plattform für mobile Informationsdienste ?



Internet-Anbindung

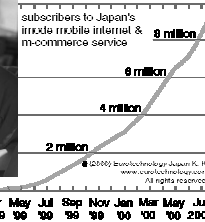
- niedrige Datenrate (9.6 kpbs), kleine Displays (~ 96 x 65 Pixel), begrenzte Eingabemöglichkeiten
- HTTP/HTML nicht optimal: HTML-Seiten zu groß, schwierig navigierbar, komplexer Browser, zu langsam über GSM

HWG 5-17

Wireless Application Protocol

Compact-HTML (cHTML)

- reduzierter HTML-Umfang
- einfache Migration
- für einfache, kleine Seiten; Browser mit kleinem Footprint
- NTT DoCoMo's I-Mode (de facto Standard in Japan)



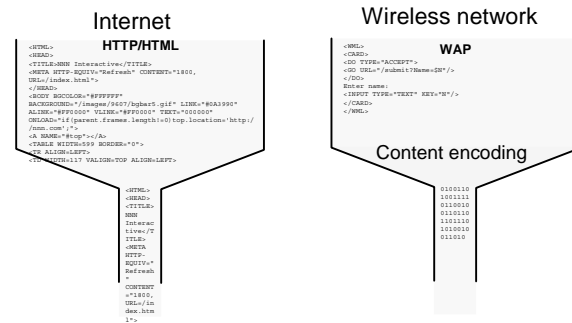
WAP und WML

- auf WWW-Standards aufbauend aber optimiert für mobilen Informationszugriff
- WAP für drahtlose Datenübertragung: Inhalte in kompaktem Binärformat, Scripts in Bytecode
- Statt reduziertem HTML dediziertes Markup für Handy User Interface: WML (Wireless Markup Language)

HWG 5-18

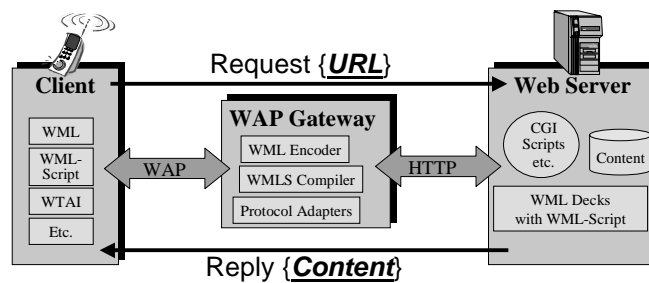
HTTP/HTML vs. WAP/WML

„Big Pipe - Small Pipe Syndrome“

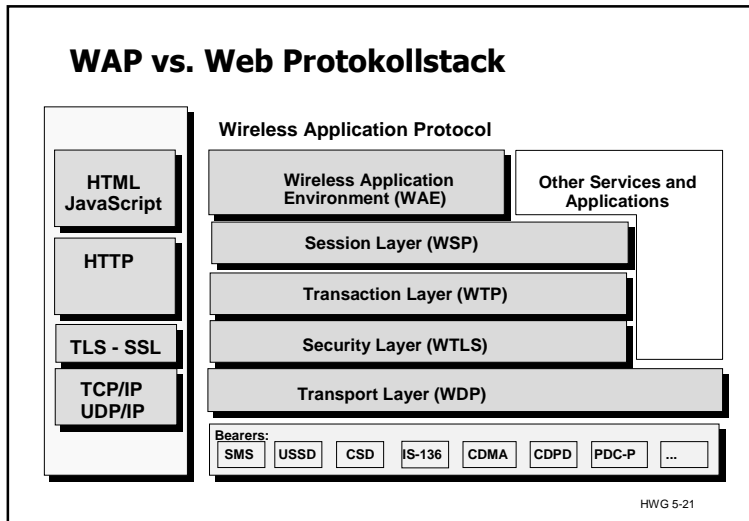


HWG 5-19

WAP Architektur



HWG 5-20



Wireless Application Environment

WAE User Agent: MicroBrowser

- Benutzer-Auswahl von Anwendungen oder externen Links resultiert in WSP-Request
- WSP-Reply
 - WML-Deck (i.e. Dokument), im Gateway in WAP Binary XML (WBXML) codiert
 - WMLScript in Bytecode (compiliert im Gateway)
 - Content in bekannten Formaten, z.B. vCard, vCalendar

WSP: Wireless Session Protocol

- kompakter als HTTP
- Content Negotiation basierend auf User Agent Profilen
- ab WAP 1.2 Push (Benachrichtigung, heute SMS)

HWG 5-22

Wireless Markup Language

Tag-basierte Markup Sprache (XML-DTD)

- Display-Management (Text, Bilder)
- Dateneingabe (Text, Auswahl aus Listen)
- Navigation (Knopfbelegungen, Hyperlinks)

Dokumentenstruktur

- Dokument = Stapel von Karten (Deck of Cards), mind. eine Karte
- Benutzerinteraktion aufgeteilt auf einzelne Karten
- Navigation zwischen Karten (Links, UI-Events, History, Timer)

Variablen

- Zustandsverwaltung, globale Variablen im Browser

HTML Hypertext-Auszeichnung von Information, Deklaration von Navigation	WML Deklaration von Benutzer-interaktion (User Interface und Navigation/Dialog)
---	---

HWG 5-23

WML Beispiel 1 – Dokumentstruktur

Prolog

```
<?xml version="1.0"?>
<!DOCTYPE wml PUBLIC "-//WAPFORUM//DTD WML 1.1//EN"
"http://www.wapforum.org/DTD/wml_1.1.xml">
```

Navigation

```
<wml>
<card id="welcome" title="Card 1">
  <do type="ACCEPT" label="Goto Card2">
    <go href="#Card2"/>
  </do>
  <p>Text is here</p>
  <p><anchor>Provided by Tec0
    <go href="http://wap.teco.edu"/>
  </anchor></p>
</card>
```

Card

Deck

```
<card id="Card2" title="Card 2">
  <do type="ACCEPT">
    <go href="/cgi/age.pl?a=${a}"/>
  </do>
  <p>your Age
  <input format="**N" name="a" title="Age:"/></p>
</card>
</wml>
```

Variablen

Eingabe

HWG 5-24

WML Beispiel – Erläuterungen

Prolog

- WML basiert auf XML: Prolog muss Verweis zur DTD enthalten (im Gegensatz zu HTML)

Navigation / Dialogablauf

- DO-Element: bindet Abläufe an Benutzeraktionen
 - vordefinierte Aktionstypen (accept, prev, options,...)
 - Abstraktion von plattformspezifischen Benutzeraktionen („linker Knopf gedrückt“ etc.)

Eingabe

- INPUT-Element für Texteingabe, Formateinschränkung möglich
- auch: SELECT aus Listen, KEY zur Direktauswahl von Optionen

HWG 5-25

WML Beispiel - Screenshots



HWG 5-26

```
<?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="welcome" title="TV-Control">
<do type="ACCEPT" label="Remote Control">
<go href="#progCard"/>
</do>
<p>TV Remote Control</p>
<p><anchor>Use the remote control!
Use the remote control
Provided by TecO
</anchor></p>
<p><anchor>Provided by TecO
<go href="http://wap.teco.edu"/>
</anchor></p>
</card>
<card id="progCard" title="Program">
<do type="ACCEPT", label="select TV">
<go href="/cgi/tv.pl?p=$ (P)"/>
</do>
<p>Select the Program:
<select name="P">
<option value="ard">ARD</option>
<option value="zdf">ZDF</option>
<option value="pro7">Pro7</option>
<option value="sat1">Sat1</option>
<option value="3sat">3Sat</option>
<option value="rtl">RTL</option>
<option value="arte">ARTE</option>
</select></p>
</card>
</wml>
```

Beispiel 2: Fernbedienung



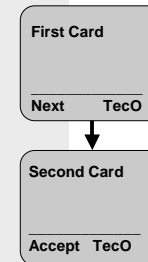
HWG 5-27

WML Templates

<TEMPLATE> Tag

- Definiert Aktionen/Events die in allen Cards vorkommen sollen

```
<wml>
<template>
<do type="OPTIONS" LABEL="Teco">
<go href="http://wap.teco.edu/" />
</do>
</template>
<card id="card1">
<do type="Accept" label="Next">
<GO URL="card2"/>
</do>
First Card
</card>
<card id="card2">
Second Card
</card>
</wml>
```



HWG 5-28

WMLScript

Script Sprache

- ähnlich JavaScript
- Prozeduren, Verzweigungen, Schleifen

Compilierung im Gateway

- Ausführung auf Virtual Machine in Client WAE
 - Verbesserte Nutzung der Bandbreite
 - Effizientere Ausführung auf dem Endgerät

HWG 5-29

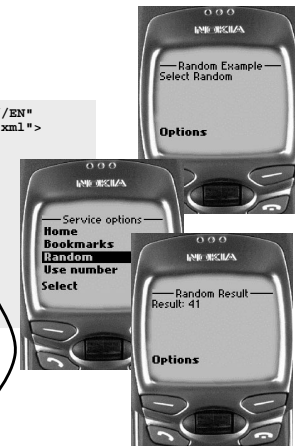
WMLScript Beispiel

WML Dokument: random.wml

```
<?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="Random Example">
    <do type="accept" label="Random">
      <go href="random.wmls#getRandom()" />
    </do>
    <p>Select Random</p>
  </card>
  <card id="card2" title="Random Result">
    <p>Result: ${RESULT}</p>
  </card>
</wml>
```

WMLScript: random.wmls

```
extern function getRandom() {
  var r=Lang.random(100);
  WMLBrowser.setVar("RESULT", r);
  WMLBrowser.go("random.wml#card2");
}
```



HWG 5-30

WTA: Wireless Telephone Architecture

Computer-Telefon-Integration (CTI)

- **WTA Browser**
 - Erweiterung des Standard WML/WMLScript Browsers
 - Zusätzliches API (WTAI)
- **WTAI umfasst Zugriff auf:**
 - Anruffunktionen
 - Text Nachrichten
 - Telefonbuch/Adressbuch
- **WTAI ist in WML und WMLScript verfügbar**

HWG 5-31

WTAI Beispiel

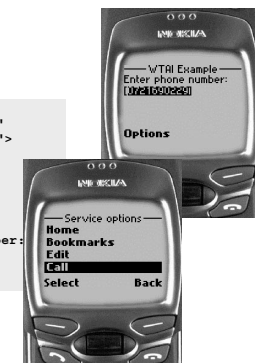
WTAI Aufruf in einer WML Seite

```
<?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="WTAI Example">
    <do type="accept" label="Call">
      <go href="wtai:cc/mc;$(N)" />
    </do>
    <p>Enter phone number: </p>
    <p><input format="*N" name="N" title="Phone Number:" />
  </card>
</wml>
```

WTAI Aufruf in WMLScript

```
function checkNum(N) {
  if (Lang.isInt(N))
    WTAI.makeCall(N);
  else
    Dialog.alert("Bad Phone Number");
}
```

Achtung: Syntax noch nicht endgültig



HWG 5-32

WAP Zusammenfassung

Anwendungsplattform

- WAP ist mehr als Zugriffstechnologie!
- WML ist mehr als Inhalts-Beschreibungssprache
- Erschließung von Mobiltelefonen für verteilte Anwendungen
- Integration Internet und Mobiltelefonie (WTAI)

Implikationen

- Anwendung nicht mehr lokal
- „Unbundeling auf dem Mobiltelefon“: Auslagerung von Anwendungen in das Netz, z.B. Telefonbuch, Spiele etc.
 - Cross-Plattform Portabilität
 - z.B. Nachbildung von Nokia-Look&Feel auf Handys anderer Hersteller ...

Anm.: wichtige Konzepte, aber aktuell noch viele Probleme (Effizienz, Kosten, Akzeptanz,...)

HWG 5-33

Smart Devices: Internet-Appliances

- Einführung
- **Mobiler Informationszugriff**
- **WAP: Wireless Application Protocol**
- **WebSIM: Web Server im Mobiltelefon**
- **CoolTown: "Serving the Real World"**

HWG 5-34

WebSIM

Integration von Mobiltelefonie und Internet

- Standardsicht: Integration von Mobiltelefonen als Client (WAP etc)
- „inverse Sicht“: Integration mobiler Geräte als Server
- Mobiltelefonie: **zwei** mobile Geräte im Tandem
 - SIM: Subscriber Identity Module, SmartCard im Handy
 - Mobiltelefon: kann als „mobiler Kartenleser“ aufgefasst werden, der zusätzlich einen I/O-Kanal für SIM-Anwendungen bietet

WebSIM (Deutsche Telekom Forschung)

- HTTP-Server in der SIM-Karte
 - Web Integration von SIM-Diensten/Anwendungen
 - insbesondere Authentisierung und Autorisierung
- „While the Web is bringing its content to the mobile phone, the mobile phone can bring its trust to the Web“

HWG 5-35

WebSIM: SIM Plattform

SIM: Subscriber Identity Module

- eingeführt in GSM-Netzen zur Authentisierung von Teilnehmern gegenüber dem Betreiber
- Smartcard mit Mikroprozessor, Speicher, serieller Schnittstelle; Betriebssystem, Dateisystem und Anwendungsprogrammen
- heute Funktionen über Authentisierung hinaus: Telefonbuch-Verwaltung, Interaktion mit SMS (Short Message Service), ...

SIM Toolkit

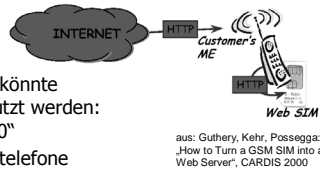
- Anwendungsplattform für sog. Toolkit-Applets
- Protokoll zum Laden solcher Applets
- Schnittstelle zum User Interface des Mobiltelefons für Benutzer-interaktion mit Toolkit-Applet
- Schnittstelle zu SMS oder anderen Kommunikationsdiensten

HWG 5-36

WebSIM Internet-Anbindung

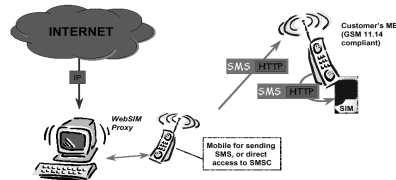
Mobiltelefon als HTTP-Gateway

- Elegante Lösung: Mobiltelefon als Gateway-Router, alle IP
- statt eigener IP-Adresse für die SIM könnte auch ein Port des Mobiltelefons genutzt werden: „Smarter Smartcards listen to Port 80“
- erfordert Erweiterung heutiger Mobiltelefone



Tunneling über SMS

- SIM-Proxy im Internet
- HTTP über SMS via Mobiltelefon zur SIM
- SMS: max. 160 Zeichen!
- Infrastruktur vorhanden



HWG 5-37

WebSIM Kommunikation

Kommunikation HTTP über SMS

- 1 Web-Client im Internet sendet HTTP-Request an SIM Proxy, z.B. [http://websim.dtrd.de/+49000000000/sign\(abcd0123...\)](http://websim.dtrd.de/+49000000000/sign(abcd0123...))
- 2 Proxy packt HTTP-Request in speziell gekennzeichnete SMS-Nachricht an die SIM
- 3 Mobiltelefon leitet Nachricht an SIM-Prozess weiter, der für den Empfang der gekennzeichneten SMS registriert ist
- 4 HTTP-Request wird ausgepackt und vom Web-Server bearbeitet
- 5 Das HTTP-Reply wird wieder in eine SMS verpackt und an den Proxy gesendet
- 6 Proxy extrahiert HTTP-Reply aus SMS-Nachricht und leitet es an den Web-Client weiter

HWG 5-38

WebSIM Implementierung

Web Server

- Applet auf der SIM Toolkit Plattform, 7k Java
- keine generischen Anfragen für Content, sondern Schnittstelle zu spezifischen Diensten

HTTP Requests

- Format: `http://<proxy-name>/<SIM-Rufnummer>/<Dienst>`
- Länge durch SMS begrenzt, max. 1 Request/SMS

Dienste

- info → antwortet mit Codes für Land, Netz und Local Area
- si=(item1, item2, ...) → SELECT ITEM, Benutzerauswahl aus Liste
- gi=(prompt) → GET INPUT, Benutzereingabe
- dt=(text) → DISPLAY TEXT
- sign=(abcd0123...) → verschlüsselt das Argument

HWG 5-39

WebSIM Anwendungen

Beispiel 1: E-Commerce

- Käufer übermittelt bei Kauf WebSIM-Adresse
- Händler fragt Bestätigung an, z.B. `http://.../.../si=(***Bookshop:,Confirm%20USD20,Cancel)`
- WebSIM bittet relativ sicheren Kommunikationskanal



Beispiel 2: Türöffner

- Haustürklingel erzeugt Anfrage, z.B. `http://.../.../si=(Open,Call Intercom,Cancel)`
- über Mobiltelefon Tür öffnen, Gegensprechanlage anrufen, oder ignorieren

HWG 5-40

WebSIM Anwendungen

Beispiel 3: Authentisierung im Internet

- A will B authentisieren, B hat WebSIM SIM_B von Netzbetreiber N
- Challenge-Response-Protokoll:
 1. B → A: +1234567890 (B's Handy-Rufnummer)
 2. A → SIM_B : [http://www.n-netz.com/+1234567890/sign\(RAND\)](http://www.n-netz.com/+1234567890/sign(RAND))
 3. SIM_B → A: $\text{hash}(\text{Key}, \text{RAND}, \text{IMSI})$ IMSI: Intl. Mobile Subscriber ID
 4. A → N: [https://www.n-netz.com/verify\(RAND,hash\(Key,RAND,IMSI\),+1234567890\)](https://www.n-netz.com/verify(RAND,hash(Key,RAND,IMSI),+1234567890))
 5. N → A: ja/nein

Serendipitous Flexibility
„unerwartete Anwendungen“
durch Vernetzung

HWG 5-41

Smart Devices: Internet-Appliances

- Einführung
- Mobiler Informationszugriff
- WAP: Wireless Application Protocol
- WebSIM: Web Server im Mobiltelefon
- CoolTown: „Serving the Real World“

HWG 5-42

CoolTown: „Serving the Real World“

CoolTown Vision/Projekt

- HP Labs Palo Alto, <http://www.cooltown.hp.com>
- Ausgangspunkt:
 - Diversifikation von Informationsprodukten
 - Offenheit und Flexibilität des WWW als Integrationsplattform

„Web Presence for the Real World“

- Web-Präsenz für alle Menschen, Plätze und Dinge

„Welcome to CoolTown ..., where people, places, and things are first class citizens of the Web“

„Interweaving the web with the context of people, places and things in the physical world will have deep implications for global and local business, institutions and individuals“

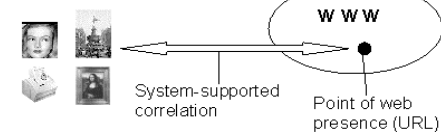
HWG 5-43

CoolTown

„People, Places, Things“

- Menschen, Plätze, Dinge: Beziehungsgeflecht in realer Welt
- Web-Präsenz von Menschen, Plätzen, Dingen: Unterstützung/Erweiterung des Beziehungsgeflechts

Person, place or thing

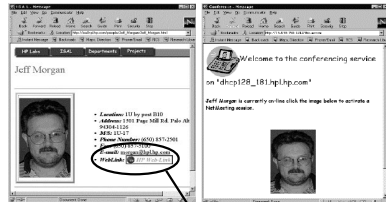


HWG 5-44

CoolTown

Web-Präsenz für Menschen, Plätze, Dinge

- Web-Präsenz von Menschen
 - Homepage (heute): statisch
 - in CoolTown: Proxy/Stellvertreter von Menschen im Web, z.B. als Kontaktpunkt für Kommunikation



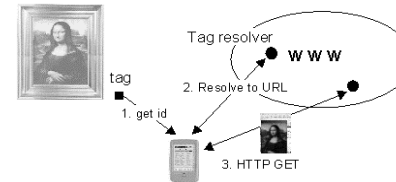
WebLink: Abstraktion von Kommunikationsdiensten

HWG 5-45

CoolTown

Web-Präsenz für Menschen, Plätze, Dinge

- Web-Präsenz von Dingen
 - Einbettung von Web-Servern in allen Geräten
 - Tagging: Dinge mit URL verknüpfen



HWG 5-46

CoolTown

Web-Präsenz für Menschen, Plätze, Dinge

- Web-Präsenz von Plätzen
 - Web-Portale von physischen Lokationen
 - Kontext für Dienstleistungen und Ressourcen
 - reflektiert Dynamik: Menschen/Dinge die kommen und gehen
 - Service/Resource Discovery: Sensoren/Tags
- Plätze im weitesten Sinne
 - z.B. Konferenzraum: Portal zu vorhandenen Ressourcen, Kontaktpunkt zu Menschen im Raum
 - z.B. Bushaltestelle: Statische und dynamische Information: wann kommt der nächste Bus, wo ist das nächste Café, wie komme ich von hier zum Bahnhof,...

HWG 5-47

CoolTown

Web-Präsenz für Menschen, Plätze, Dinge

- Reflektion der Beziehungen zwischen web-präsenten Menschen, Plätzen und Dingen
- vom Beziehungs-Kontext abhängige Präsenz

Beispiel: WebBus

- Web-Server im Bus eingebettet, GPS-Lokation integriert
- für Menschen im Bus: Portal zu ortsabhängiger Information, z.B. zu Sehenswürdigkeiten die zu sehen sind
- für Menschen an der Haltestelle: wo ist der Bus jetzt, wann wird er ankommen

HWG 5-48