

Architektur eines Web Portals für mobile Endgeräte

Jochen Müller, Torsten Lenhart, Markus Hillenbrand, Aline Verney

Integrated Communications Systems Lab,
Department of Computer Science,
University of Kaiserslautern,
67653 Kaiserslautern, Germany
{jmueller, t_lenhart, hillenbr, verney}
@informatik.uni-kl.de

Abstract. Die Weiterentwicklung mobiler Endgeräte hat zu einer Zunahme der mobilen Nutzung des Internets und seiner Webangebote geführt. Mit den dadurch entstehenden erweiterten Möglichkeiten gehen aber auch Probleme einher, vor allem weil sich die neue Generation von Endgeräten in vielen Eigenschaften von herkömmlichen Desktop-Computern unterscheidet. Dies hat zur Folge, dass vor allem im HTTP/HTML-basierten World Wide Web neue Wege beschritten werden müssen. Diese Arbeit beschäftigt sich mit der Fragestellung, wie die Unterstützung heterogener Endgeräte im Rahmen von Web-Applikationen zu erreichen ist und stellt eine Architektur für ein solches Web Portal vor.

1 Einleitung

In den letzten Jahren hat sich das Internet zum festen Bestandteil unserer Gesellschaft entwickelt. Laut einer repräsentativen Umfrage ist die Zahl der Bundesbürger, die unmittelbar über einen Zugang zum Internet verfügen, vom 1.Quartal 2003 zum 1.Quartal 2004 um 4 Prozentpunkte auf nunmehr 57% gestiegen [FW04].

Zusätzlich ist das Angebot an mobilen Endgeräten, welche mit einem Browser oder Email-Client ausgestattet sind signifikant gestiegen. In diesem Zusammenhang wird von Herstellern und Netzbetreibern das Schlagwort Mobile Internet propagiert: Dem Kunden wird „anytime and anywhere“ ein Zugang zu verschiedenen Internetdiensten angeboten.

Neben den Entwicklungen der Netze und Endgeräte, verändert sich auch die inhaltliche Organisation des Internets. Insbesondere das Konzept des „Web Portals“ hat sich dabei durchgesetzt. Durch eine thematische und/oder regionale Gruppierung der Inhalte auf Web-Portale und zusätzliche Personalisierungsmöglichkeiten wird versucht die Informationsmenge für den User einfacher zugänglich zu machen.

Aufgrund dieser Entwicklung erscheint es sinnvoll bei der Entwicklung von Web-Applikationen und insbesondere bei Portalen auch die Zielgruppe der mobilen Nutzer einzubeziehen.

2 Adaptionprozess

„Anpassungsfähigkeit unterscheidet die Gewinner von den Verlierern.“
Charles Darwin, englischer Wissenschaftler und Begründer der Evolutionstheorie

Eine Analyse der Charakteristika mobiler Endgeräte hat gezeigt, dass es oftmals nötig ist, Web Applikationen an die jeweiligen Geräteeigenschaften anzupassen. Die Vorgehensweisen, um eine solche Anpassung zu erreichen, werden im Folgenden als Adaptionprozess bezeichnet. Der Adaptionprozess muss in der Lage sein, für möglichst viele Geräte genau auf ihre Bedürfnisse zugeschnittene Daten mit minimalen Aufwand zu erzeugen [Ca04].

Es können zwei voneinander abgrenzbare Ansätze der Adaption identifiziert werden, die sich im Wesentlichen darin unterscheiden, ob die entsprechende Web-Applikation am Adaptionsvorgang beteiligt ist oder nicht [WDI04b]. Im ersten Fall spricht man von individuell kontrollierter Adaption. Die Anpassung von Inhalten an die jeweiligen Eigenschaften des anfragenden Endgeräts ist hier als wesentliche Funktion in die Applikation eingebettet. Die Adaption kann also vom Betreiber der Web-Applikation konkret beeinflusst und gesteuert werden. Von einer individuell unkontrollierten, universellen Adaption spricht man, wenn die Anpassungsvorgänge ohne Beteiligung der angesprochenen Webseite oder Web-Applikation ablaufen. Es werden hier die gesendeten Daten mit Hilfe von generischen Vorgehensweisen an ganz spezielle Bedürfnisse angepasst. Für die Adaption ist allein der Browser verantwortlich, beispielsweise das Small-Screen Rendering.

Im Bereich der individuell unkontrollierten Adaption wurden in den letzten Jahren große Fortschritte gemacht und neue, leistungsfähige Algorithmen entwickelt. Trotzdem kann diese Art der Adaption manchmal keine zufriedenstellende Ergebnisse liefern. Aus diesem Grund werden kurz nur Verfahren der individuell kontrollierten Adaption vorgestellt. Diese kann man allgemein in die Klassen, des single, multiple und flexible Authoring unterteilen [DIW2004a]. Beim Multiple Authoring werden jeweils separate Webseiten für die verschiedenen zu unterstützenden Geräte oder Geräteklassen erstellt. Im Gegensatz dazu, wird beim single Authoring nur eine einzige Version der Seite erstellt und gleichzeitig viel Zeit in die Ausgestaltung des Adaptionprozesses investiert. Das flexible Authoring kombiniert die beiden Ansätze.

3 Geräte-Identifikation

„Jede Erkenntnis ist ein Identifizieren des Nichtgleichen.“
Friedrich Nietzsche, deutscher Philosoph, Aphoristiker und Lyriker

Die Erkennung der Geräteeigenschaften stellt auf Serverseite im WWW keine einfache Aufgabe dar, weil im Rahmen des HTTP Standards keine Weitergabe dieser Informationen definiert wird. Zwar sind Angaben über das anfragende Gerät enthalten, diese reichen aber für ein genaues Bild längst nicht aus [Gi02]. Außerdem halten sich die Browser in diesen Punkt nur selten an die im Protokoll festgelegten Konventionen oder tragen gewollt falsche Informationen ein. Im Folgenden werden die wichtigsten Ansätze beschrieben, um diese Lücke im Protokoll zu schließen.

3.1 HTTP Request Header

Ein HTTP Request besteht nach der Protokollversion 1.1 aus der Adresse der angeforderten Ressource, der anzuwendenden Methode (in der Regel ist dies GET, was zum Senden der Ressource an den Client führt) und diversen Header-Feldern [Be99]. Diese Felder enthalten mitunter auch einige Angaben über verschiedene Charakteristika des anfragenden Geräts. Besonders das Feld User Agent scheint auf den ersten Blick eine viel versprechende Möglichkeit zu sein. Im Protokoll ist zwar eine vage Konvention enthalten, wie der Feldwerte aufgebaut werden sollte, allerdings wird diese von den meisten Browsern nicht eingehalten.

3.2 Composite Capabilities/Preferences Profiles (CC/PP)

Das World Wide Web Consortium (W3C) arbeitet seit einiger Zeit an dem Composite Capabilities/Preferences Profiles (CC/PP), mit dessen Hilfe die Fähigkeiten von Geräten und die Präferenzen von Benutzern detailliert beschrieben werden können. CC/PP setzt auf der XML-Serialisierung des Resource Description Frameworks (RDF) auf, welches entwickelt wurde, um den reibungslosen Austausch von Metadaten und Eigenschaftsinformationen zwischen künstlichen Agenten zu unterstützen [Iz02].

Mit CC/PP ist ein Ansatz vorhanden, mit dem auch komplexe Informationen über die Charakteristika anfragender Endgeräte für Web Server transparent gemacht werden können. Allerdings ist die Arbeit an CC/PP noch nicht abgeschlossen und einige kritische Punkte sind noch offen [WSP04] [Iz02].

3.3 User Agent Profiles (UAProf)

Bei User Agent Profiles (UAProf) handelte es sich um eine erste Implementation von CC/PP. Sie wurde von der Open Mobile Alliance (OMA) entwickelt, welche auch für die WAP Protokollfamilie verantwortlich ist. Manchmal wird UAProf auch fälschlicherweise als Vokabular für CC/PP bezeichnet, doch neben der Festlegung von Attributen und ihrer Bedeutung werden bei diesem Ansatz schon verschiedene Aspekte behandelt, die im Rahmen von CC/PP noch ungelöst sind. So wurde UAProf um in CC/PP fehlende Protokollgesichtspunkte ergänzt und erweiterte Prozessregeln für das Mischen von Informationen aus heterogenen Quellen sind im Standard enthalten. Im Zentrum von UAProf steht aber tatsächlich ein CC/PP-Vokabular, dass speziell auf mobile Endgeräte zugeschnitten ist [OMA01].

4 Architektur von SESAME

Wäre die Natur behaglich, hätten die Menschen die Architektur nicht erfunden.
Oscar Wilde, irischer Lyriker, Dramatiker und Bühnenautor

In Folgenden wird mit SESAME1 (SErver Side Adaption for Mobile Entities) ein generischer Ansatz zur Realisierung einer serverseitigen gesteuerten Adaption vorgestellt. SESAME ist als Grundkonzept bei der Erstellung einer Web-Applikation anwendbar, kann aber auch bei einer nachträglichen Erweiterung von bestehenden Applikationen um Adaptionfunktionalitäten eingesetzt werden. Einzige Voraussetzung zur Anwendung von SESAME stellt die Verwendung des Model-View-Controller-Pattern dar. Dieses Pattern entkoppelt die Anwendungs-, Darstellungs- und Interaktionslogik und ermöglicht so einfache nachträgliche Änderungen und Erweiterungen. Für weitere Informationen über MVC sei auf [Hu01] und [Tu00] verwiesen.

Die beiden Extremansätze zur Gestaltung von Web Applikationen, Multiple Authoring und Single Authoring, besitzen unterschiedliche Nachteile, die sich in der Praxis als schwerwiegend erweisen können [HR03]. Im Rahmen von SESAME wird daher ein Mittelweg beschritten, der die Vorteile beider Vorgehensweisen so weit möglich vereint ohne die Nachteile zu übernehmen. Dabei wird der Grundsatz „So viel Single Authoring wie möglich, so wenig Multiple Authoring wie nötig“ umgesetzt.

Es werden drei Views erstellt, die im Hinblick auf den Leistungsfähigkeit eines anfragenden Endgeräts optimiert sind. Folgende Views sind dabei vorgesehen:

¹ SESAME spielt auf das Märchen „Ali Baba und die 40 Räuber“ an. Die Architektur soll im Sinne von „Open SESAME“, den mobilen Zugang für heterogene Endgeräte zu unterschiedlichen Web Applikationen ermöglichen.

- ein Standard View für Desktops, Notebooks, große Subnotebooks
- ein Midi View für Handhelds und kleine Subnotebooks
- ein Mini View für Smartphones und Mobiltelefone

Jeder View wird mit Adaptioneninformationen, z.B. in Form von speziellen Tags, versehen. Diese zusätzlichen Informationen werden von einem Adaptionsprozessor dazu genutzt, den View an die ganz speziellen Eigenschaften des anfragenden Endgeräts anzupassen. Während durch die verschiedenen Views lediglich eine relativ grobe Adaption in Bezug auf die Displaygröße verwirklicht wird, erfolgt hier eine Anpassung an die anderen Eigenschaften (Farbfähigkeit, Netzanbindung etc.), wobei auch eine feinere Anpassung an die Displaygröße realisiert werden kann.

Die einzelnen Stufen werden von klar voneinander abgrenzbaren Teilen der Anwendung abgearbeitet. Die so identifizierbaren Module werden im Rahmen von SESAME als Prozessoren bezeichnet und als Bestandteile des Controllers realisiert. Drei Prozessoren können dabei unterschieden werden:

1. Identification Processor: Nimmt einen HTTP Request entgegen und versucht die Eigenschaften des anfragenden Endgeräts aus diesem zu ermitteln.
2. Selection Processor: Wählt einen der drei vorhandenen Views entsprechend den zuvor festgestellten Geräteeigenschaften aus.
3. Transformation Processor: Nimmt die Feinanpassung vor.

Die vom Identification Processor erkannten Geräteeigenschaften werden dabei in einem Device Capabilities Object (DeCO) abgelegt. Hier kann vom Selection und Transformation Processor aus auf sie zugegriffen werden. Beim DeCO handelt es sich um eine sessionweite Instanz, d.h. für jede HTTP Session muss der Identification Processor nur einmal zu Beginn der Session aktiv werden. In Abbildung. 1 ist die SESAME-Architektur dargestellt.

Diese Grundarchitektur erfährt durch die Einführung eines abstrakten Masterviews jedoch noch eine wesentliche Erweiterung. Beim abstrakten Masterview handelt es sich um eine Sicht, die den eigentlichen Views vorgeschaltet ist und die im Gegensatz zu diesen keinerlei Darstellungsinformationen enthält. Den verschiedenen Versionen einer Seite in den drei Views wird eine gemeinsame Seite im Masterview zugeordnet (beispielsweise den Seiten `indexstandard.jsp`, `indexmini.jsp` und `indexmidi.jsp` die abstrakte Seite `index.jsp`), welche lediglich Verweise auf die entsprechenden konkreten Seiten enthält. Dies ermöglicht eine einheitliche Verlinkung auf seinen der Web-Applikation, und für Benutzer bleiben die Bookmarks austauschbar. Die abstrakten Seiten werden dabei unter Verwendung von XML als Markup-Sprache realisiert.

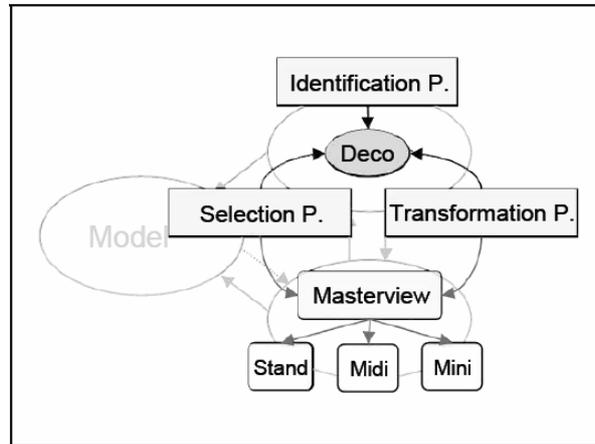


Abbildung 1: Architektur von SESAME

5 Zusammenfassung und Ausblick

Im Rahmen dieser Arbeit wurde ein Ansatz zur Anpassung von Web-Inhalten vorgestellt. Dieser soll zukünftig dazu beitragen, dem Problem der Heterogenität mobiler Endgeräte in größerem Maße Herr zu werden. Der vorgestellte Ansatz wurde auf eine MVC-basierten Applikation eingesetzt und konnte so seine Praxistauglichkeit unter Beweis stellen.

Gerade der Identification Processor stellt durch die Kombination von bewährten Vorgehensweisen mit der innovativen Unterstützung des erst kürzlich verabschiedeten W3C-Standards CC/PP eine Vorgehensweise dar, die vielen bestehenden Lösungen des Problems der Eigenschaftserkennung weit voraus ist. Für die Zukunft ist es notwendig, SESAME zu einem marktreifen Ansatz weiter zu entwickeln. Hierzu sollte in erster Linie die Adaptation Tag Library erweitert werden, vor allem um solche Tags, die dem Transformation Processor zugeordnet werden können. Daneben sollte die Entwicklung im Bereich der Techniken und Methoden zur Geräteerkennung und Inhaltsanpassung weiter verfolgt.

Ein Ansatzpunkt für Weiterentwicklungen wäre es, die Prozessoren als Web Services zu realisieren. Dadurch würde die Möglichkeit eröffnet, dieselben Prozessoren im Rahmen von vielen Web-Applikationen zu verwenden, die auf räumlich weit entfernten Servern ablaufen. Zwar wäre das gegenwärtig noch vorhandene Geschwindigkeitsdefizit von Web Services ein Hindernis für die praktische Anwendung, man könnte dadurch aber schon jetzt einen innovativen Weg gehen, der zukünftig hoch relevant werden könnte.

Acknowledgement

Diese Arbeit ist Teil eines von der Stiftung Rheinland-Pfalz für Innovation gefördertes Forschungsprojektes.

Literaturverzeichnis

- [FW04] Forschungsgruppe Wahlen Online: Internet-Strukturdaten: Repräsentative Umfrage - I. Quartal 2004, http://www.forschungsgruppe.de/Ergebnisse/Internet-Strukturdaten/web_I_04_1.pdf (Stand: 05.04.2004, Abruf: 26.11.2004)
- [WDI04a] 2. W3C Device Independence Working Group: Authoring Challenges for Device Independence, <http://www.w3.org/TR/acdi/> (Stand: 01.09.2003 Abruf: 26.11.2004)
- [Ca04] Canals: - EMEA Q1 2004 - highlights from the Canals research <http://www.canals.com/pr/r2004041.htm> (Stand: 20.04.2004, Abruf 26.11.2004)
- [WDI04b] W3C Device Independence Working Group: Authoring Techniques for Device Independence, <http://www.w3.org/TR/2004/NOTE-di-atdi-20040218/>, (Stand: 18.02.2004 Abruf: 26.11.2004)
- [Gi02] Gimson, Roger: W3C Device Independence Working Group (), Delivery Context Overview for Device Independence, <http://www.w3.org/TR/di-dco/>, (Stand: 13.12.2002, Abruf: 26.11.2004)
- [Be99] Berners-Lee, T.; et al.: Hypertext Transfer Protocol – HTTP/1.1, The Internet Society, 1999
- [WSP04] Web Standards Project WaSP ask the W3C: Introduction to Composite Capabilities/Preference Profiles (CC/PP) <http://www.webstandards.org/learn/askw3c/feb2004.html> (Stand: Februar 2004, Abruf: 26.11.2004)
- [Iz02] Izdepski, Eric: Ultimate Device Support with CC/PP aus: XML Journal, Ausgabe 07/2002 Montvale (New Jersey, USA), Sys-Con Media
- [OMA01] Open Mobile Alliance: User Agent Profile (UAProf) <http://www.openmobilealliance.org/tech/affiliates/wap/wap-248-uaprof-20011020-a.pdf> (Stand: 20.10.2001, Abruf: 26.11.2004)
- [Hu01] Hunt, Dr. John: You've got the Model-View-Controller; <http://www.jaydeetechnology.co.uk/planetjava/tutorials/swing/Model-View-Controller.PDF> (Stand: 29.05.2001, Abruf: 26.12.2004)
- [Tu00] Turau, Volker: Java Server Pages; 1. Auflage, Heidelberg, dpunkt Verlag, 2000
- [HR03] Hillenbrand, M. and Reuther, B.: Building Blocks for Web Applications, 7th IASTED International Conference Internet and Multimedia Systems and Applications IMSA Hawaii, USA, 8/2003, pp. 757-761.