

# **Die integrierte Client-Entwicklung für PC und Mobile Endgeräte - am Beispiel der Portaltechnologie -**

Gabriel Vögler, Thomas Flor

DaimlerChrysler AG, Research and Technology,  
Software Architecture (RIC/SA),  
Postfach 23 60, 89013 Ulm  
{gabriel.voegler|thomas.flor}@daimlerchrysler.com

**Abstract:** Zur Vermeidung einer redundanten Entwicklung sowie Wartung von Informationen und Anwendungen für verschiedene Clients muß eine Architektur für den Zugriff auf Unternehmensinformationen mit mobilen Endgeräten in der Praxis als Erweiterung der bestehenden Infrastruktur für den bereits existierenden Desktop-Zugriff realisiert werden. Mit der Portaltechnologie stellt dieser Artikel eine praxisorientierte Strategie und Lösung für Pervasive Computing Applikationen vor.

## **1 Einleitung**

Die zunehmende Verbreitung von internetfähigen mobilen Endgeräten, wie z.B. PDA's oder Smartphones, ermöglicht es, die bisher rein PC-gestützten Unternehmensprozesse, durch eine ausgewählte "Mobilisierung" effizienter zu gestalten. In vielen Großunternehmen existiert ein konzernweites Intranet, was den Mitarbeitern auf Basis von Internettechnologien den Zugriff auf unternehmensspezifische Informationen (z.B. Konzernnachrichten), operative Systeme (z.B. funktionspezifische Datenbankanwendungen) und diversen Tools für die Organisation des Arbeitsalltags (z.B. Reisemanagement) erlaubt. Für viele dieser Informationen ist auch ein mobiler Zugriff sinnvoll, der in einem mobilen Kontext hilfreich ist oder den Mitarbeitern die Möglichkeit gibt, ihre Arbeitszeit effizienter zu nutzen (indem z.B. die Wartezeit bei Flugreisen für die Erstellung der Reisekostenabrechnung genutzt wird).

Unternehmen stehen somit vor der Herausforderung, die existierende IT-Architektur um Möglichkeiten des mobilen Zugriffs auf Unternehmensanwendungen zu erweitern. Im Falle des Intranets bedeutet dies z.B. die existierende Infrastruktur für die Bereitstellung des PC-User-Interface so zu erweitern bzw. zu modifizieren, dass unterschiedliche Client-spezifische Darstellungen unterstützt werden. Dabei soll die redundante Pflege von Inhalten und Anwendungen für PC und mobile Endgeräte vermieden werden. Vielmehr wird nach einer Lösung gesucht, die neben den nötigen Technologien für Pervasive Computing (z.B. spezielle Transcoding-, Synchronisations- oder Sicherheitsdienste), auch die Möglichkeiten der integrierten Pflege der Intranet-Applikationen für PC und mobile Endgeräte bietet. Außerdem muss auf technische Neuerungen auf dem Markt für mobile Endgeräten flexibel reagiert werden können. In dem folgenden Artikel wird erläutert, wie diesen Anforderungen mit der Portaltechnologie begegnet werden kann.

## 2 Anforderungen und Lösungsansätze für Pervasive Computing

Die Anforderungen, die sich aus den technischen Merkmalen der Endgeräte ergeben, sind vielfach publiziert worden [Bu01]. Die zu berücksichtigenden Unterschiede finden sich sowohl auf der Hard- als auch auf Softwareebene. Auf der Hardwareebene sind diese Unterschiede sowohl quantitativ (Bildschirmgröße, Farbtiefe, Speicherkapazität, etc.) als auch qualitativ bezogen auf den Aufbau des Geräts (Tastatur, Eingabestift, Telefonastastefeld, etc.). Auch auf der Softwareebene muss einem sehr heterogenen Umfeld begegnet werden. So ist die Unterstützung von Markup-Sprachen (verschiedene Versionen von HTML, cHTML, WML) und weiteren Standards für die Visualisierung (z.B. JavaScript, CSS) sehr unterschiedlich.

In der Praxis stellt sich vor allem die Problematik, die sehr komplexen und für große Bildschirme konzipierten Intranet-Anwendungen auf den kleinen Displays darzustellen. Durch die begrenzten Ein- und Ausgabemöglichkeiten der mobilen Endgeräte ist vor allem die Usability ein kritischer Erfolgsfaktor einer mobilen Anwendung. Der Nutzen eines mobilen Datendienstes wird bei einer komplizierten Bedienung von den Mitarbeitern erfahrungsgemäß schnell in Frage gestellt.

Grundlegende Techniken bei der Bereitstellung Client-spezifischer User Interfaces sind z.B. das Fragmentieren von Inhalten auf mehrere Seiten, die Reduzierung von Texten auf Kerninformationen, das Herunterskalieren von Bildern etc. [LL02, PS02, Gi02, Gi01]. Doch auch die Anwendung dieser Techniken führt in vielen Fällen noch zu keinem befriedigenden Ergebnis. Um die Komplexität des User Interfaces weiter zu reduzieren, muss daher in einem nächsten Optimierungsschritt differenziert werden, welche Intranet-Anwendung auf welchem Gerät sinnvoll ist. In vielen Fällen lässt sich eine akzeptable mobile Nutzung jedoch erst dann erreichen, wenn das User Interface zusätzlich personalisiert wird. Das Angebot an Intranetanwendungen wird exakt auf die Bedürfnisse eines Mitarbeiters zugeschnitten. Diese Personalisierung schließt auch ein Single-Sign-On-Konzept mit ein, was einen entscheidenden Beitrag dazu leistet, die Anzahl eingabeintensiver Dialoge zu reduzieren.

## 3 Portaltechnologie

Ein Portal ist eine integrierte und personalisierte Web-basierte Schnittstelle zu Informationen und kollaborativen Diensten [Fe00]. Ein Portal-Server generiert eine angepasste und personalisierte Homepage, die die Inhalte von verschiedenen Informationsquellen und Anwendungen aggregiert. Die Inhalte bzw. Anwendungen sind dabei in sog. Portlets organisiert. Portlets sind wiederverwendbare Komponenten, die ein Standardinterface implementieren und in einem speziellen Portlet-Container laufen, der den Portlets eine entsprechende Laufzeitumgebung zur Verfügung stellt. Die so genannte Portal-Engine stellt neben dem Portlet-Container eine Reihe von Standardservices zur Verfügung, die oft benötigte Funktionalität implementieren. Beispielhaft seien die Folgenden genannt [Fe00, Ib01]:

- **Presentation Services** aggregieren Inhalte von verschiedenen Quellen entsprechend der Präferenzen und der Zugriffsrechte des Users zu einer personalisierten Homepage. Die Homepage wird dabei so gestaltet, dass die Anordnung der Portlets übersichtlich strukturiert ist und grundlegende Navigationselemente für deren Aufruf zur Verfügung stehen. Der User kann dabei ein einheitliches Layout und Design für das gesamte Portal (inklusive aller Portlets) definieren.
- **Personalisation Services** helfen dabei, dem Nutzer eine auf seine Interessen zugeschnittene Homepage zusammenzustellen. Dies kann durch die Konfiguration mittels Administrator oder auch des Users selber erfolgen. Darüber hinaus gibt es auch Ansätze, die sich auf Rollen-basierte Regeln stützen.
- **Security Services** ermöglichen dem User nach einmaliger Anmeldung den passwortfreien Zugriff auf, durch unterschiedliche Sicherheitssysteme geschützte Bereiche (single sign-on). Daneben wird durch Security Services die Einhaltung der Zugriffsrechte überwacht (content protection).
- **Content Management Services** bieten eine einheitliche Plattform für das Verwalten und Publizieren von Textinhalten. Bei vielen Systemen können die Inhalte mit Meta-informationen ausgezeichnet werden.

Das Portlet-Konzept sowie die beschriebenen Standardservices der Portalumgebung führen insgesamt zu einer hochentkoppelten Architektur, in der es wohl definierte und abgrenzbare Komponenten gibt. Aus Implementierungssicht lassen sich zwei Ebenen unterscheiden:

- **Portal-Ebene:** enthält die nötige anwendungsneutrale Funktionalität für das Aufrufen von Portlets. Neben der Darstellung der Rahmenseite inklusive der entsprechenden Navigationselemente umfasst dies z.B. auch die Services für die Authentifizierung und die Personalisierung.
- **Portlet-Ebene:** liefert das tatsächliche User Interface für den Zugriff auf eine Anwendung. Das User Interface beschränkt sich auf die anwendungsspezifische Funktionalität.

Die Entwicklung Portal-basierter Anwendungen umfasst nur die Portlet-Ebene. Die Funktionalität der Portal-Ebene ist in dem Portal-Produkt bereits implementiert. Der Entwickler kann sich also auf die funktionalen Aspekte seiner Anwendung konzentrieren.

#### 4 Ausnutzung der Portalarchitektur für Pervasive Computing

In Kapitel 2 sind grundlegende Techniken beschrieben, wie Client-spezifische User Interfaces realisiert werden können. Die Portaltechnologie bietet für die Umsetzung dieser Ansätze eine sehr gute Ausgangsbasis. Für die Rahmenfunktionalität der Portal-Ebene können die Hersteller von Portalplattformen Standardimplementierungen für den mobilen Zugriff anbieten. Um Fragen der Anmeldung oder der Navigation zwischen Portlets muss sich nicht gekümmert werden. WebSphere Portal Server von IBM [Ib01] bietet z.B. Implementierungen für HTML, PDA ("tiny HTML") und WML.

Da die Portlets nach einem einheitlichen Programmiermodell implementiert und im System eindeutig adressierbar sind, ist es möglich, die Homepage durch separates Aufrufen einzelner Portlets in semantisch sinnvoller Weise zu fragmentieren (Abbildung 1).

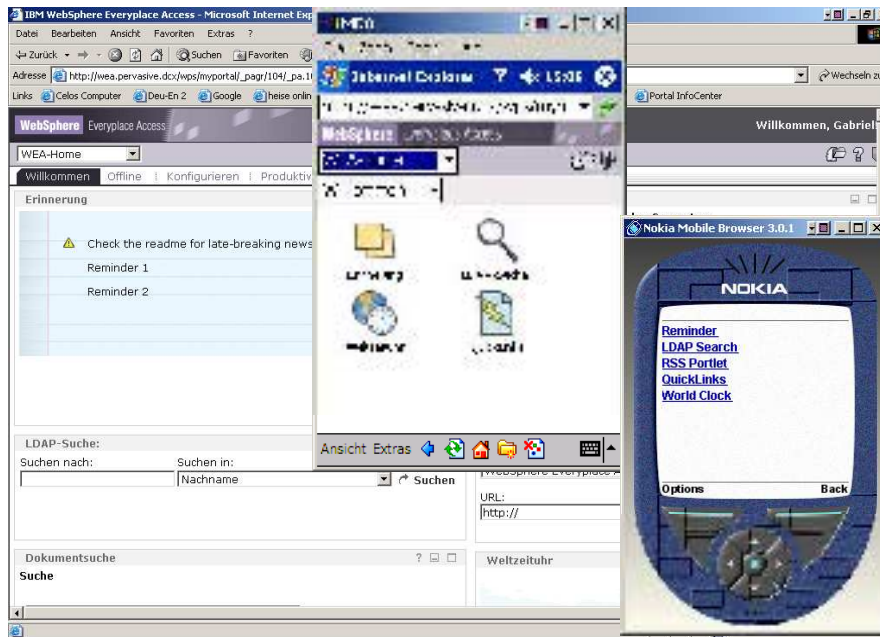


Abbildung 1: verschiedene Clients für ein Portal

Implementierungsaufwand bei der Anpassung für den mobilen Zugriff entsteht also nur auf Portlet-Ebene. Hier muss der Entwickler dafür sorgen, dass ein Portlet verschiedene Versionen seines User Interfaces zur Verfügung stellt. Da die Ausgabe eines Portlets aber rein anwendungsbezogen ist und keine sekundäre Funktionalität enthält (z.B. Login oder Suche), ist dies weniger aufwendig als bei Webseiten eines herkömmlichen Intranets. Der Forderung nach einer integrierten Entwicklung und Pflege sowohl der PC- als auch der mobilen Version kann bei Portlets z.B. durch die Anwendung eines MVC-Frameworks oder den Einsatz von Portletfiltern entsprochen werden (siehe hierzu z.B. [Ib01]).

Für Textinformationen können die angesprochenen Content Management Services verwendet werden. Diese werden z.B. so erweitert, dass in den Metainformationen hinterlegt wird, in welchem Umfang ein Text auf den unterschiedlichen Geräten angezeigt werden soll. Auf diese Weise wird eine einheitliche Datenbasis benutzt, durch die eine parallele Pflege von Inhalten vermieden werden kann.

Die beschriebenen Personalisation Services können dazu genutzt werden, die in Kapitel 2 umrissene Idee eines personalisierten User Interface zu realisieren. So kann mit den erwähnten Konzepten bestimmt werden, welches Portlet einem User auf welchem Gerät zur Verfügung steht.

## 5 Schlussbemerkung

Die Portalplattform erweist sich in der Praxis als eine gute Basis für die integrierte Entwicklung von Anwendungen, die neben dem PC auch über mobile Endgeräte nutzbar sein sollen. Der hohe Abstraktionsgrad in der Architektur und die konzeptionelle Trennung zwischen Portalelementen und Portlets ermöglichen den Entwicklern eine relativ Client-neutrale Entwicklung des User Interfaces und entlasten ihn von vielen technischen Problemen. Im Gegensatz zu anderen Architekturansätzen für Pervasive Computing hat die Portalplattform ihre Einsatzfähigkeit in der Praxis schon gezeigt und kann auch die heutigen Möglichkeiten eines Desktop-Clients voll nutzen. Darüber hinaus bietet die Architektur auch die nötige Flexibilität, um auf technische Neuerungen am Endgerätemarkt reagieren zu können. Für Unternehmen, die ihre Anwendungen neben dem PC auch auf mobilen Endgeräten nutzbar machen wollen oder sich zumindest die Option dafür offen halten wollen, ist die Portalplattform daher ein geeigneter Lösungsansatz. Voraussetzung ist allerdings, dass das Portalkonzept konsequent umgesetzt wird und die besprochenen Standardservices auch genutzt werden. In der Praxis werden Portlets leider oft nur als Einstiegspunkt für eine herkömmliche Web-Anwendung verwendet, die außerhalb der Portalumgebung läuft. In diesem Fall können die hier besprochenen Vorteile dann aber nicht mehr genutzt werden.

## Literaturverzeichnis

- [Bu01] Burkhardt, J., Henn, H., Hepper, S., Rindtorff, K., Schäck, T.: Pervasive Computing. Addison-Wesley, München, 2001.
- [Fe00] Ferguson, M.: The IBM e-Business Portal Software Solution. DataBase Associates, England, November 2000.  
URL: <http://www.dbaint.com/pdf/WPSWhitePaperV5.pdf>.
- [Gi01] Gimson, R. (ed), Finkelstein, S.R., Maes, S., Suryanarayana, L.: Device Independence Principles. World Wide Web Konsortium, USA, 2001. W3C Recommendation.  
URL: <http://www.w3.org/TR/2001/WD-di-princ-20010918/>
- [Gi02] Giannetti, F. Device Independence Web Application Framework (DIWAF). HP Labs. W3C Workshop on Device Independent Authoring Techniques, 25-26 September 2002, SAP University, St. Leon-Rot, Germany.
- [Ib01] IBM Corporation: IBM WebSphere Portal Server Product Architecture V2.1.  
URL: <http://www-3.ibm.com/software/genservers/portal/pdf/WPSWhitePaperV2-1.pdf>
- [LL02] Lemlouma, T. Layada, N.: Device Independent Principles for Adapted Content Delivery. INRIA Rhône Alpes, October 2002  
URL: <http://opera.inrialpes.fr/people/Tayeb.Lemlouma/Papers/DIReport1.pdf>
- [PS02] Paternò, F., Santoro, C.: One Model, Many Interfaces. Proceedings Fourth International Conference on Computer-Aided Design of User Interfaces, pp. 143-154, Kluwer Academics Publishers, Valenciennes, May 2002.