

Ein Portal – Was ist das eigentlich?

Jochen Rütschlin

jochen.ruetschlin@DaimlerChrysler.com

DaimlerChrysler AG · Forschung und Technologie
Labor IT for Engineering · Abteilung Prozesskette Produktentwicklung (FT3/EK)
Postfach 23 60 · D-89013 Ulm / Donau

Zusammenfassung

In diesem Beitrag soll der Versuch unternommen werden, den vagen Begriff eines Portals genauer zu spezifizieren. Dazu werden Intention und technische Bestandteile eines Portals erörtert. Hilfreich dabei ist der Blick auf bestehende Systeme, die unter dem Begriff Portal laufen, um durch Generalisierung und Klassifikation einer Definition näher zu kommen. In diesem Zusammenhang werden auch einige Ergebnisse aus einer in unserem Hause durchgeführten Evaluierung von drei Portal-Systemen vorgestellt. Als Ausblick werden einige Anforderungen angesprochen, die mit der heutigen Portal-Technologie gar nicht oder nur sehr schwer bzw. umständlich zu realisieren sind.

Schlüsselwörter: Portal, Web, Application Server.

1. Einleitung

Laut einer Studie der Gartner Group von vor etwas über einen Jahr [1] ist der Portal-Hype schon über den Punkt der überzogenen Erwartungen (*peak of inflated expectations*) hinaus auf dem absteigenden Ast. Es wird vorhergesagt, dass der Zeitpunkt, ab dem sich diese Technologie bezahlt machen soll (*plateau of productivity*), noch binnen der nächsten zwei Jahre – vom Zeitpunkt der Prognose gerechnet – erreicht wird. Nach gut der Hälfte dieses prognostizierten Zeitraums ist vielleicht die Zeit gekommen, eine kleine Bestandsaufnahme zu machen.

Die anfängliche Euphorie über die Lösung aller Probleme durch Einsatz eines Portals, führte bei genauerem Hinschauen schnell zur Ernüchterung. Der Portal-Begriff war zwar in aller Munde, doch keiner wusste so recht was sich eigentlich dahinter verbarg bzw. verbergen sollte. Diejenigen, welche konkrete Vorstellungen hatten und sich Gedanken über den Nutzen eines Portals machten, hatten dann auch meist eine sehr spezielle Sicht der Dinge, so dass der Versuch einer Portal-Definition allgemeiner hätte nicht ausfallen können: ein Portal als

der Eintrittspunkt zu einem Informationssystem. Erschwerend kam hinzu, dass – gewissermaßen um „auf der Portal-Welle mit zu reiten“ – viele Produkte und Lösungen als Portal-Lösungen verkauft wurden, was sicherlich auch zu einer „Verwässerung“ des Begriffs führte.

In diesem Beitrag soll daher der Versuch unternommen werden, die vage Vorstellung eines Portals etwas genauer zu fassen. Zeigt sich die Schwierigkeit darin, eine aussagekräftige, knappe und allgemeingültige Definition zu geben, werden zu deren Klärung in Abschnitt 2.1 erst einmal einige Betrachtungen zur Intention eines Portals angestellt. Zusammen mit der Erläuterung einiger typischer Bestandteile technischer Natur in Abschnitt 2.2 soll dies zu einer begrifflichen Bestimmung führen.

Um ein Portal zu realisieren bietet der Markt diverse Produkte an. Jeder große, namhafte Hersteller hat mittlerweile ein Portal-Produkt in seinem Portfolio. Exemplarisch haben wir uns mal drei, aus unterschiedlichen Lagern kommende Lösungen, die allgemein hin unter dem Schlagwort Portal einzuordnen sind, angeschaut. Einige der Ergebnisse dieser Evaluation sollen in Abschnitt 3 vorgestellt werden.

Schließlich werden in Abschnitt 4 – gewissermaßen als Ausblick – ein paar Probleme aus der Praxis angesprochen. Der Beitrag schließt mit einer Zusammenfassung in Abschnitt 5.

2. Der Portal-Begriff

Wenn man über ein Portal spricht hat man meist einen von zwei Aspekten im Sinn. Bei dem einen handelt es sich um die Funktionalität, die von einem Portal angeboten werden soll bzw. eine Problemstellung die dadurch gelöst werden soll. Der andere Aspekt bezieht sich auf die Technologie, mit der diese Funktionalität realisiert werden soll, und deren Umsetzung letztendlich in einem (oder mehreren) Produkt(en) erfolgt. Gemäß dieser Ambiguität werfen wir in 2.1 erst einen Blick auf die

Intention eines Portals. Dazu haben wir einen Blick auf vorhandene, laufende Portale geworfen und versucht, sie zu klassifizieren bzw. insoweit von konkreten Ausprägungen zu abstrahieren, um Kategorien herauszubilden.

Der darauffolgende Abschnitt 2.2 beschäftigt sich dann mit typischen, technischen Bestandteilen eines Portals.

2.1. Intention

Der Grundgedanke, der einem jedem Portal zugrunde liegt, ist sicherlich einen (einzigen) Einstiegspunkt zu mehreren, u.U. verschiedenartigen Informationen bereitzustellen. Allein daraus ergeben sich schon fast von selbst die typischen Attribute, die im Zusammenhang mit einem Portal genannt werden: verschiedene Informationskanäle, Suchfunktionen und Personalisierung.

Die Personalisierung wird häufig als *der* charakterisierende Aspekt eines Portals angeführt: für den Benutzer soll eine Umgebung geschaffen werden, in der er alle relevanten Informationen auf einen Blick erfassen kann oder zumindest leichten Zugriff auf sie hat. Informationen, die nicht gleich gefunden werden können bzw. die verstreut abgelegt sind, sollen dann über die angesprochene Suchfunktion lokalisiert werden können.

Alle Informationen bzw. Anwendungen sollen quasi in einem großen Fenster – einer homogenisierten Benutzeroberfläche – ablaufen. Der angezeigte Fensterinhalt ist an die Präferenzen und die Aufgaben des jeweiligen Benutzer angepasst. Innerhalb des einzigen, großen Fensters gibt es typischerweise kleinere, virtuelle Fenster – auch *Portlets* genannt –, welche die unterschiedlichen Informationen und Anwendungen voneinander abgrenzen. Sie lassen sich konfigurieren, verkleinern oder auch wegklicken. Alles Eigenschaften, die kritisch hinterfragt, nichts neues sind, da sie schon seit Jahren von den grafischen und fensterorientierten Betriebssystemen her bekannt sind (Windows, MacOS, XWindows, KDE, etc.). Aber genau da kommt ein weiterer Charakterzug, den man mit dem Begriff eines Portals verbindet, zum Tragen: die Web-Fähigkeit. Durch die Plattformunabhängigkeit der zugrundeliegenden Web-Infrastruktur soll das Portal dem Benutzer die Möglichkeit bieten, von jedem beliebigen Punkt aus auf seine ihm gewohnte (Arbeits-) Umgebung zuzugreifen.

Während bei einer Arbeitsumgebung im B2E- (*business-to-employee*) oder Engineering-Umfeld sinnvollerweise die Portlets aus Anwendungen bestehen (oder zumindest ihren Inhalt daraus beziehen), liegt der Fokus von Konsumenten-Portalen wie Yahoo oder Excite eher auf der Darstellung von mehr oder minder statischen Informationen, gewissermaßen als ein Zugang (sprich: Portal) ins Web.

Aus dem Grund haben wir mal versucht aus unterschiedlichen Portalausprägungen und Bezeichnungen bestimmte Charakteristika abzuleiten (vgl. auch [2]).

- Der Begriff des *vertikalen Portals* bezeichnet ein Portal, welches sich mit einem bestimmten Thema beschäftigt, sozusagen ein Segment eines horizontal aufgespannten Informationsraumes betrachtet. Beispiele wären das Gesundheitsportal NetDoktor.de oder das Mobilfunkportal eplus-online.de.
- Dem gegenüber steht das *horizontale Portal*, auch *Konsumenten-Portal* oder *Megaportal* genannt. Darunter fallen Angebote offeriert von Yahoo, AOL oder Lycos. Sie wollen einer breiten Masse von Benutzern den Einstieg ins Internet erleichtern und bieten dazu eine reiche Anzahl von vortierten Themen an.
- Das Unternehmens-Portal soll dem IT-Anwender genau die Informationen und Anwendungen eröffnen, die er für seine tägliche Arbeit benötigt. Synonym dazu wird oft auch der Begriff des *Enterprise Information Portals* (EIP) verwendet.

Die Bezeichnung der B2X-Portale setzt das Unternehmen in den Mittelpunkt der Betrachtungen.

- B2B (*business-to-business*) Portale werden im Geschäfts- und Datenverkehr zwischen zwei Unternehmen eingesetzt, z.B. zwischen dem Unternehmen und einem Zulieferer. Hier stehen Sicherheitsaspekte bzgl. Zugriffsrechte und Einsichtnahme im Vordergrund.
- B2E (*business-to-employee*) Portale stellen hohe Anforderungen an die Personalisierungskomponente, über die der Mitarbeiter für ihn spezifische, das Unternehmen betreffende Informationen angeboten bekommen soll (z.B. Interna, Stellenausschreibungen, Gleitzeitauszüge, etc.). Während sich das B2E-Portal an jeden Mitarbeiter richtet, richtet sich das EIP eher an den IT-Anwender und die Nutzung von „Produktivanwendungen“.
- B2C (*business-to-consumer*) Portale stellen schließlich den Kontakt zum Kunden her. Hier werden Kundengeschäfte abgewickelt oder Produktinformationen angeboten. Somit steht bei diesem Portal eine hohe Verfügbarkeit im Vordergrund. Durch personalisierte Angebote soll eine Kundenbindung erreicht werden.

2.2. Technologie

Technisch gesehen ist ein Portal kein isoliertes System, sondern bündelt gemäß der zuvor besprochenen Intention bereits bestehende Anwendungen oder Informationsdienste. Sein Aufbau ist daher nicht monolithisch,

sondern kann, wie in Abbildung 1 dargestellt, als eine Ansammlung verschiedener Funktionsblöcke angesehen werden, die teilweise durch unterschiedliche Systeme umgesetzt werden.

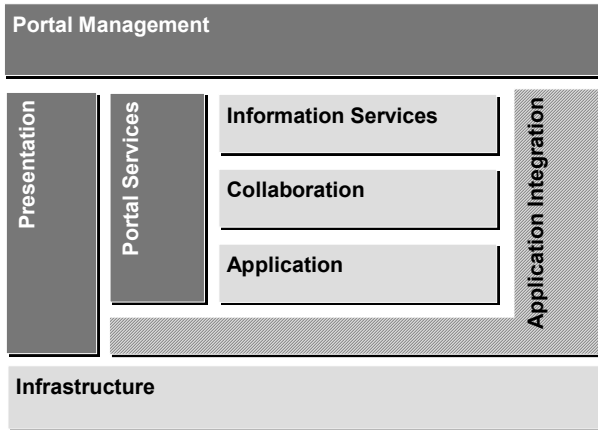


Abbildung 1: Bestandteile eines Portals.

Die in obiger Abbildung hellgrau dargestellten Blöcke repräsentieren bereits vorhandene Systeme (z.B. in einem Unternehmen), die mittels den dunkelgrau dargestellten portalspezifischen Funktionen zusammengeführt werden sollen. Die vorhandenen Systeme können dabei (in Hinblick auf die Integration in ein Portal) grob in drei Kategorien eingeteilt werden:

- *Information Services:* klassische Informationen wie Nachrichten, Dokumentationen, Info-Dienste (wie Wetter, Aktien, etc.), Web-Inhalte, allgemein hin durch ein *Content Management System* verwaltete Informationen, u.ä.
- *Collaboration:* E-Mail, Kalender, ganz allgemein Groupware, aber auch Workflow-Systeme.
- *Application:* herkömmliche (teilweise schon Webfähige) Anwendungen, die entweder selbst entwickelt worden sind oder aber fertig gekauft wurden.

Mehr oder weniger orthogonal dazu verlaufen die Integrationsarchitekturen, mit denen vormals isolierte Einzelsysteme interoperabel oder aber auch einfach nur für die Nutzung im Inter-/Intranet tauglich gemacht werden. Der Block nimmt in der Abbildung eine gewisser Sonderstellung ein, da diese Aufgabe noch nicht vollständig gelöst ist und weiterhin an Umsetzungen und neuen Konzepten gearbeitet wird.

Das ganze läuft auf einer bestimmten Infrastruktur. Darunter fallen (neben dem Betriebssystem und der Hardware) Systeme wie Application Server, Web-Server, Verzeichnis- und Transaktionsdienste, aber auch bestimmte Sicherheitsdienste. Im Wesentlichen all die

Strukturdienste, die aus dem Web-Umfeld stammen bzw. über Adapter/Konnektoren mit diesem verbunden werden können.

Das typische Portal besteht aus einem Management-Teil, der Aufgaben wie z.B. die (globale) Benutzerverwaltung und das Monitoring übernimmt. Ein weiterer Block übernimmt die geräteabhängige Darstellung der Inhalte und Anwendungen beim Benutzer, und unter den *Portal Services* sind die typischen Portalfunktionen wie Personalisierung, Suche, Verwaltung von Inhalten und die Navigation zusammengefasst.

Man kann also erkennen, dass ein Portal eigentlich kein eigenständiges Produkt ist, sondern vielmehr eine Erweiterung, die auf eine bereits bestehende Infrastruktur und diversen (Informations-) Diensten und Anwendungen aufsetzt. So sei auch schon einmal vorweggenommen, dass die im folgenden Abschnitt vorgestellten Portal-Systeme (wie auch die meisten auf dem Markt befindlichen Produkte) alle auf Application Server Technologie basieren (vgl. auch [3]).

3. Drei Portal-Systeme

Im Rahmen einer bei uns durchgeführten Diplomarbeit [4] wurden drei Systeme zur Realisierung eines Portals gemäß unterschiedlicher Kriterien wie Architektur, Integration von Informationsquellen, Personalisierung und Sicherheit untersucht und – sofern dies überhaupt möglich ist – vergleichend bewertet. Bei den Systemen handelte es sich zum einen um das *Open Source* Paket Jet-speed aus der Apache-Gruppe [5], dann eine Customizing-Lösung von iPlanet mit dem Namen PAF (*portal application framework*) und – als Vertreter für kommerzielle Kauf-Software – BEAs Portal-Lösung bestehend aus dem BEA WebLogic Commerce & Personalization Server [6]. Aus Platzgründen kann hier natürlich nur entsprechend kurz auf die einzelnen Systeme eingegangen werden. Daher sollen mehr oder weniger nur die Unterschiede bzw. Charakteristika der Systeme herausgehoben werden.

Gemein ist allen drei Systemen, dass sie auf Web- bzw. Servlet/JSP (*Java server page*) Technologie basieren. Das heißt, der Benutzer ruft das Portal über eine URL in seinem Browser auf, meldet sich über eine Benutzerkennung und ein Passwort an und bekommt dann seine eigene, auf ihn personalisierte Portal-Oberfläche als HTML-Seite zurück. Diese wurde auf dem Application Server entsprechend seinem gespeicherten Profil aus unterschiedlichen Informationsquellen (unter Zuhilfenahme diverser Servlets und JSPs) zusammengebaut. Das Profil bzw. der anzuzeigende Inhalt wird dabei teilweise von dem Benutzer selbst konfiguriert oder ausge-

wählt, teilweise durch einen Operator gemäß den Befugnissen und Aufgaben des Benutzers festgelegt.

3.1. Jetspeed

Jetspeed ist ein Teil des Java *Open Source* Projektes der Apache Software Foundation und besteht fast ausschließlich aus einer Anzahl von Servlets. Als Infrastruktur wird deshalb eine Servlet-Engine und ein Web-Server benötigt, wobei die „haus eigene“ Engine Jakarta-Tomcat empfohlen wird. Das Portal lässt sich allerdings auch problemlos auf einer anderen Infrastruktur wie beispielsweise IBM WebSphere installieren.

Als Inhalte können hauptsächlich textuelle Informationen wie HTML-Seiten, XML-Inhalte entweder roh oder in den Formaten RSS (*rich site summary*) oder RDF (*resource description framework*), sowie PHP-Skripte und Cocoon-Portlets (HTML-Fragmente, die aus XML unter Verwendung von XSL erzeugt werden) eingebunden werden. Diese Vielzahl der Möglichkeiten zur Einbindung von Inhalten ist neben der offenen Implementierung sicherlich zu den Vorteilen von Jetspeed zu zählen.

Personalisierungsfunktionalität war zum Zeitpunkt der Evaluierung zwar schon vorgesehen, allerdings noch nicht implementiert, was einen gravierenden Nachteil darstellte. Durch den *Open Source* hat hier allerdings jeder die Möglichkeit, die fehlenden Teile nachzuimplementieren. Wahrscheinlich auch ein Grund, warum die IBM Jetspeed als Grundlage für ihren WebSphere Portal-Server genommen hat.

3.2. iPlanet PAF

Der iPlanet PAF ist eine Customizing-Lösung, d.h. eine Portal-Lösung die es nicht als Produkt zu kaufen gibt, beruhend auf dem Netscape Enterprise Server und dem Netscape Application Server; letzterer nutzt den Netscape Directory Service zur Speicherung von Konfigurationsdaten (u.a. auch die Personalisierungsdaten). Obwohl auch hier die eigentliche Portalfunktionalität durch Servlets und JSPs realisiert ist, lässt sich diese nicht auf eine andere Infrastruktur übertragen.

Die Liste der unterstützten Inhaltsformate ist beim PAF nicht ganz so reichhaltig, allerdings besteht die Möglichkeit, über sogenannte *Feeder* Informationen aus beliebigen (über eine URI spezifizierten) Quellen mittels einem selbst zu implementierenden Filter auszulesen und in ein ausgabegerät-spezifisches Format zu bringen (z.B. HTML oder WML). Über diese Feeder können auch aus Datenbanken Informationen (die allerdings in einem speziellen Format vorliegen müssen) ausgelesen werden.

Die Vorteile des PAF liegen sicherlich in den Transcoding-Möglichkeiten und im Gegensatz zu Jetspeed in den etwas ausgereifteren Personalisierungskomponenten.

Nachteilig wirken sich die vielen von Hand zu tätigen Schritte, fehlende Standardkomponenten und der dadurch recht hohe Implementierungsaufwand aus, was allerdings in der Natur einer Customizing-Lösung liegt.

3.3. BEA WebLogic

Die Portallösung von BEA ist sicherlich die ausgereifteste der dreien. Sie basiert ebenfalls auf einer Application Server Architektur (WebLogic Server, WebLogic Commerce Server) mit einem zusätzlichen *Personalization Server*. Dabei besticht insbesondere dessen Möglichkeit, Inhalte und Benutzer regelbasiert zu bewerten und zu klassifizieren, wodurch eine maschinelle Auswertung der Zuordnung, welchen Inhalt welcher Benutzer zu sehen bekommt, erfolgen kann. Die Spezialisierung auf das Online-Shop Geschäft zeigt sich auch in der Bereitstellung vieler begleitender Komponenten wie die Unterstützung bei der Erstellung eines Produktkataloges, JSP-Vorlagen für Bestellungen und Benutzerverwaltung sowie die Berücksichtigung von Steuern und Kreditkartenzahlungen.

Offenheit bzgl. der Verwendung von Systemkomponenten anderer Hersteller besteht nur bei der Datenbank zur Speicherung der Konfigurationsdaten und dem HTTP-Server; in beiden Fällen werden Systeme der Marktführer unterstützt.

Mit der Integration von Informationsquellen verhält es sich ähnlich wie mit iPlanets PAF: alle über eine URI ansprechbare Quellen können eingebunden werden, sofern man einen Parser implementiert, der die Quelle untersucht und entsprechend in HTML-Code umwandelt.

3.4. Zusammenfassung

So unterschiedlich die Herkunft der einzelnen Lösungen ist, so zeigt sich auch das typische Merkmal von Portal-Produkten: es handelt sich bei ihnen um keine eigenständigen Software-Pakete sondern vielmehr um Erweiterungen, die auf andere Produkte der jeweiligen Anbieter/Hersteller aufgesetzt werden. Meist handelt es sich dabei um Application Server Technologie oder eine Servlet-Engine (mit optionalem JSP-Prozessor). Alle uns bekannten Produkte verwenden für die Realisierung der Portal-Funktionalität Servlets oder JSPs. Der Plattformunabhängigkeit von Java wegen könnte man daher meinen, dass diese Portal-Erweiterungen nun in beliebige Umgebungen, sprich Application Servern einsetzbar sind. Interessanterweise laufen aber die Portal-Servlets/-JSPs oft nur auf der vom gleichen Hersteller angebotenen (oder sogar gleich mitgelieferten) Infrastruktur bzw. sind nur mit erheblichen Aufwand in eine andere Umgebung zu portieren.

Alle drei Produkte hatten während den Tests wegen mangelnder Proxy-Unterstützung Schwierigkeiten mit dem Holen von Inhalten außerhalb des Intranets (unsere Testumgebung wurde durch eine Firewall vom Internet abgeschottet).

Eine vergleichende Bewertung von Systemen mit so unterschiedlicher Herkunft ist auch deshalb um so schwieriger, weil sie teilweise ganz unterschiedliche Ziele verfolgen. Während bei BEA der Fokus stark auf den E-Commerce über das Internet gerichtet ist, hat der PAF seine Stärken in der Transcoding-Fähigkeit, wohingegen an eine *Open Source* Implementierung sicherlich keine so hohen Ansprüche gestellt werden dürfen, wie für ein Produkt, für das sehr viel Geld bezahlt wurde. Nichtsdestotrotz weist auch Jetspeed Stärken auf, insbesondere bei den vorhandenen Möglichkeiten, eine Anzahl unterschiedlicher Informationsquellen einzubinden und der durch die Offenheit bedingten Portabilität der Portal-Komponenten.

4. Einige Probleme

In diesem Abschnitt soll noch auf einige Probleme aus der Praxis mit dem Umgang von Portalen und der Gestaltung deren Inhalte eingegangen werden. Die Probleme können als Ausblick auf zukünftig notwendige Entwicklungen angesehen werden.

4.1. Interoperabilität

Aussagen, wie „Mit einem Portal können alle für einen Benutzer und dessen Arbeit relevanten Informationen und Dienste in einem Fenster integriert werden“ beziehen sich integrationstechnisch im Hinblick auf der Interoperabilität der unterschiedlichen Portlets tatsächlich meist nur auf die gemeinsame Darstellung in einem Fenster. Software-unterstützte Aktionen zwischen den Portlets wie beispielsweise die Nutzung einer E-Mail-Adresse aus dem Adressbuch-Portlet in dem E-Mail-Portlet sind von Seiten des Portals nicht vorgesehen und müssen selbst mit Techniken der in Abbildung 1 ange deuteten *Application Integration* realisiert werden.

Ähnliches gilt auch für die Realisierung der *Single Sign-On* (SSO) Funktionalität. Zu oft werden pauschale Aussagen gemacht, dass ein SSO problemlos realisiert werden kann. Das Miteinbeziehen eines bereits vorhandenen *Corporate Directory* ist aber teilweise gar nicht möglich, so dass Benutzerinformationen (abermals) in ein neues (Portal-) Verzeichnis repliziert bzw. gänzlich neu angelegt werden müssen. Mal ganz davon abgesehen, dass bei einer SSO-Implementierung die Funktionalität der zu integrierenden Systeme eine maßgebliche Rolle spielt: Existenz eines API zur Authentifizierung

und/oder Passwortänderung; Verfall von Passwörtern; Pflege einer Passworthistorie; verwendete Methoden wie Passwörter, Token, SecurID® Card, Zertifikate, etc. Auch die Fragestellung, ob die einzubindenden Systeme weiterhin autonom verwendet werden sollen oder nur noch über das Portal anzusprechen sind, ist nicht unerheblich für die SSO-Realisierung.

4.2. Transcoding

Einige Hersteller preisen ihr Portal-Produkt damit an, dass sie unterschiedliche Benutzerendgeräte (*multi-devices*; HTML-Browser, WAP-Handy, Incar-Device, Handhelds, ...) unterstützen, so dass der Zugang zum Portal von nahezu jedem beliebigen Ort zu jeder Zeit möglich ist. Ermöglicht wird dies dadurch, dass die Daten geräteunabhängig, meist in XML vorliegen und dann mittels bestimmter Transformationsregeln (XSLT, *XML style-sheet language transformation*) in das vom Endgerät benötigte Format gebracht werden (*transcoding*). Bei einfachen Textinhalten wie beispielsweise Kurznachrichten bestehend aus einer Schlagzeile, dem Untertitel und einem Textkörper funktioniert das noch ganz gut. In der Praxis zeigt sich allerdings sehr schnell, dass wenn es daran geht kleinere Web-Anwendungen einzubinden, die Umsetzung des Transformationsansatzes, wenn überhaupt, nur sehr nahe an der Datenhaltungsschicht erfolgen kann; fast immer müssen die Anwendungen jeweils für die verschiedenen Endgeräte einzeln implementiert werden. Dies lässt sich gut am Beispiel eines Routenplaners verdeutlichen, der sowohl über einen normalen Web-Browser als auch ein WAP-fähiges Mobiltelefon bedienbar sein soll. Hier treten beide, im Wesentlichen für die Transcoding-Problematik verantwortlichen Faktoren auf: unterschiedliche Programmlogik und unterschiedliche für die Anzeige zur Verfügung stehende Fläche. Beginnen wir mit der *Anzeigefläche*. Eine für das Mobiltelefon zurückgelieferte Anfahrtsskizze muss entsprechend kleiner und mit weniger Details versehen sein. (Mal ganz davon abgesehen, dass manche Ausgabegeräte unter Umständen gar keine Grafik anzeigen können.) Dies lässt sich nicht ohne weiteres aus einer entsprechenden Grafik für den Web-Browser gewinnen.¹ Daher kann eine Transformation nur sehr dicht an der Datenhaltung angesiedelt erfolgen bzw. es müssen für die Endgeräte unterschiedliche Datenformate bereitgehalten werden. Überhaupt müssen Inhalte teilweise je nach Ausgabegerät anders aufgeteilt (über mehrere Seiten verteilt) oder präsentiert werden (z.B. Vorlesen

¹ Unter der Annahme, dass der Content-Provider den Routenplaner primär für einen Browser zugeschnitten, *out-of-the-box* liefert, und das Transcoding für ein anderes Endgerät mit der Transformationsfähigkeit des Portals erzielt werden soll.

durch ein *in-car device*). Dies würde sich noch mit Transformationsregeln realisieren lassen, wenn diese Aufteilung nicht auch Auswirkungen auf die *Programmlogik* hätte (Verwendung von „Nächste Seite“ Schaltern; halten des *session context*, etc.). Ebenso müssen gerätebedingte Einschränkungen beispielsweise in der Eingabe durch zusätzliche Programmlogik ausgeglichen werden. Hinzu kommt noch, dass geräteabhängige Darstellungsformate wie HTML und WML eine enge Verflechtung von Daten, deren Darstellung und der Anwendungslogik aufweisen (man denke hier nur an in einem HTML-Strom enthaltenes JavaScript zur Überprüfung von Eingabedaten), was eine gegenseitige Überführung äußerst schwierig macht. Daher muss vielmehr die Anwendungslogik, ebenso wie die Daten selbst, geräteunabhängig formuliert werden, was derzeit noch nicht unterstützt wird.

4.3. Sicherheit

Ein Aspekt, der im Zusammenhang mit Portalen gerne vernachlässigt wird, ist die Sicherheit in Bezug auf die Vertraulichkeit der angezeigten Daten. Es wird geworben, dass man beim Einsatz eines Portals, von jedem beliebigen Rechner mit Internet-Zugang auf seine Unternehmensdaten (Dokumente, E-Mail, etc.) zugreifen kann, sei es aus einem Internet-Café oder gerade beim Kunden vor Ort. Gesichert werden diese Zugänge mit teilweise erheblichen Aufwand (SecurID® Card, etc.). Was hier aber gerne vergessen wird, ist dass alle angezeigten Informationen – auch SSL geschützt! – lokal auf dem Rechner in einem Puffer unverschlüsselt auf der Platte abgespeichert werden. Sie sind somit auch noch verfügbar, nachdem der Benutzer den Rechner verlassen hat, was insbesondere bei fremden Maschinen sicherheitstechnisch bedenklich ist. Der Benutzer hat zwar die Möglichkeit, über den verwendeten Browser diesen Puffer zu löschen, aber dies liegt allein im Ermessen des Benutzers und kann somit auch leicht vergessen werden (mal davon abgesehen, dass eine Anweisung, jedes mal nach Beendigung einer Sitzung den Puffer zu löschen, nicht gerade sehr benutzerfreundlich ist). Ein durch die Portal-Software angesteuerte automatische Löschung (bzw. ein gänzlich Unterbinden der Pufferung) wäre hier wünschenswert, ist aber wahrscheinlich schwierig zu realisieren, da dies einen Eingriff in den verwendeten Browser bzw. das lokale Betriebssystem bedeutet.

4.4. Tunneling

Ein weiteres Sicherheitsproblem ist das sogenannte Tunneln von verschlüsselten Informationen durch das Portal. Man stelle sich einen durch ein Portal realisierten Marktplatz vor, über den sich beispielsweise mehrere

Automobilfirmen ihren Zulieferern präsentieren. Dabei müssen die Firmen den Betreibern des Portals zwangsläufig vertrauen, da es keine Möglichkeit gibt, bei vollständiger Nutzung der Portal-Funktionalität eine gesicherte Verbindung zwischen dem Automobilhersteller und dem Zulieferer durch das Portal hindurch zu realisieren, ohne dass der Betreiber Kenntnis von dem übermittelten Inhalt bekommt.

5. Zusammenfassung

Zusammenfassend kann man sagen: *Ein Portal ist der Zugangspunkt, über den ein Nutzer eines IT-Systems alle für ihn relevanten Informationen und Anwendungen (ortsunabhängig) angeboten bekommt, um einer bestimmten Aufgabe oder einem Wunsch nachzukommen.* Eine Definition, die leider so allgemein ausfallen muss, um alle (sinnvollen) Facetten eines Portals abzudecken.

Technisch gesehen ist ein Portal die viel zitierte Spitze des Eisbergs und stellt mehr oder weniger nur die Präsentationsschicht der Informationsquellen dar. Die eigentlichen Herausforderungen der oben genannten Intention einer benutzerseitigen, umfassenden Informations- und Anwendungsintegration liegen weiterhin im Bereich der interoperablen Integration der darunter liegenden Informationssysteme. Ein Portal kann diesbezüglich eigentlich nur durch die Bereitstellung vorgefertigter Komponenten unterstützend mit einwirken.

6. Quellenverzeichnis

- [1] J. Fenn, A. Linden (Gartner Group): *2000 Hype Cycle of Emerging Technologies*. Research Note Technology, 5. Juni 2000.
- [2] M.E. Orłowska: *Integrating Web Based Applications – Challenges and Opportunities*. Keynote speak 7th International Conference on Database Systems for Advanced Applications, 19. April 2001, Hong Kong.
- [3] W. Hasselbring, A. Koschel, A. Mester: Basistechnologien für die Entwicklung von Internet-Portalen. In: Andreas Heuer, Frank Leymann und Denny Priebe (Herausgeber). *Tagungsband der 9. GI-Fachtagung »Datenbanksysteme in Büro, Technik und Wissenschaft«* (BTW 2001), Informatik Aktuell, 7.-9. März 2001, Oldenburg.
- [4] T. M. Schwarz: *Portale – Eine Diskussion zugrundeliegender Konzepte und ihre Realisierung in ausgewählten Produkten*. Diplomarbeit, Universität Ulm, März 2001.
- [5] Apache Software Foundation: *Jetspeed-Homepage*. <http://java.apache.org/jetspeed/>
- [6] BEA Systems, Inc.: *BEA WebLogic Commerce Server & BEA WebLogic Personalization Server*, Dokumentationen im Internet. <http://edocs.bea.com/wlcs/docs31/>