

Achim Bode

## Rich Clients – Die Zukunft des Internet

*Ursprünglich hatte das Internet reinen Informationscharakter. Heute bieten Websites längst mehr als das. Vor allem Portale warten mit einem hohen Grad an Personalisierung und Interaktion auf, während jedoch ihre Technik diesen Ansprüchen kaum noch gerecht wird. Abhilfe können Rich Clients schaffen: Sie geben Internetseiten die interaktive Leistungsfähigkeit von Office-Anwendungen und erfüllen damit die stetig wachsenden Anforderungen.*

Schon lange stellt sich die Frage, warum sich Internetnutzer mit so wenig zufrieden geben: Während sich das Design von Internetseiten ständig verbessert, verlangen die meisten Seiten dem User bei der Nutzung einiges an Geduld ab. HTML kann man seine häufig beklagten Usability-Probleme nur schwerlich zum Vorwurf machen: Für die heutigen Formen der Internet-Nutzung war es nie gedacht.

### **Von der Doktorarbeit zur Anwendung**

In den Anfangszeiten enthielt das Internet vorwiegend wissenschaftliche Texte, die durch Hyperlinks aufeinander Bezug nahmen. Im Laufe der Jahre besiedelten nach und nach kommerzielle Firmenprofile das Web, wodurch sich das Internet vor allem zu einem Marketing-Instrument entwickelte. Produktpräsentationen verbinden Marken mit »Customer Experience«, einer Erlebniswelt für den Kunden. Portale sind so umfangreich geworden, dass sie die Komplexität von Programmen erreicht haben: angefangen von Shops

und Auktionen, über Bankapplikationen bis hin zu B2B-Plattformen; Weblogs, Foren und Web-Email-Services bieten komplette Texteditoren.

Die Vorteile des Internet werden auch von Software-Anbietern immer häufiger erkannt: Es ist keine Installation erforderlich, neue Versionen sind für alle Nutzer sofort verfügbar und – last but not least – die Anwendung ist überall und jederzeit nutzbar. Immer mehr Hersteller bieten daher mittlerweile über das Internet abrufbare Versionen ihrer Software an, so genannte ASP-Anwendungen.

### **Nur der Server denkt**

HTML ist mit dem Anspruch angetreten, Informationen im Netz verfügbar zu machen, die untereinander verlinkt sind. Niemand hatte erwartet, dass die heute üblichen Interaktionen und Designs damit möglich sind. Die Entwicklung von verlinkten Texten hin zu kompletten Anwendungen fand so langsam statt, dass es kaum jemandem auffiel. Dadurch wurde die verwendete Technik Stück für Stück mühsam den neuen Herausforderungen angepasst. Der Punkt, an dem man über einen Technologiewechsel nachdenken sollte, wurde verpasst.

Stylesheets haben die grafischen Fähigkeiten erweitert, Formulare brachten mehr Funktion und JavaScript sogar ein Mindestmaß an Client-Intelligenz. Im Wesentlichen können sich HTML-Seiten jedoch nicht verändern, wenn der Nutzer eine Aktion durchgeführt hat. Die Entscheidung, welche Seite unter welchen Bedingungen auf eine Seite folgt



**Joachim Bode**

Jahrgang 1969, arbeitete nach seinem Architekturstudium bei der Pixelpark AG als Konzepter und Projektmanager. Neben Kunden aus der Automobilindustrie und großen Finanzinstitute war er für die europäischen Webauftritte des Sportartikelherstellers adidas zuständig. Er war Mitgründer des Internet-Portals *seniorplace.de* und arbeitete als Flash-Developer und IT-Consultant. Heute ist er freiberuflicher Konzepter und Programmierer für Rich Clients in Berlin.

E-Mail: [jo@achimbo.de](mailto:jo@achimbo.de)

und wie diese aussieht – man spricht hier von der (Programm-)Logik –, liegt auf dem Server. Sie folgt einem seriellen Grundprinzip: Jeder Mausclick lädt eine neue Seite.

Die Server-Logik der Seitenabfolge wurde zweckentfremdet, um mit starren Webseiten interaktives Verhalten vorzugaukeln. Das funktioniert – aber ist es wirklich ideal?

### **Bedienerfreundlichkeit als Nebenprodukt**

Das Bedürfnis nach Erlebnis auf Marketing-Sites führte zum verstärkten Einsatz eines Browser-Plugins, das sich vom Zeichenprogramm zum Quasi-Standard für »reichere« Websites gemausert hat: Macromedia Flash. Der Grund war vor allem das enorme Ausmaß an Möglichkeiten in der Animation. Bilderreichtum, Bewegung, Überblendungen, Sound und schließlich sogar Videos deckten die Bedürfnisse der Marketing-Branche ideal ab.

Diese Möglichkeiten brachten absurderweise auch einen Nachteil mit sich: Sie schrien danach, genutzt zu werden. Eine ganze Generation von »Freizeit-Flashern« verausgabte sich darin, alles und jedes zu animieren. Die Technik wurde zum Selbstzweck.

Mit Flash entwickelten sich jedoch auch – ganz nebenbei – die Mittel, um bei zunehmend komplexen Interaktionen die erforderliche Usability gewährleisten zu können. Flash bietet für die neuen Aufgaben ähnliche Voraussetzungen wie Windows: Drag and Drop, sofortige Reaktion eines Balkens im Diagramm auf die Einstellung mit dem Schieberegler, hervorragende Möglichkeiten zur Visualisierung komplexer Zusammenhänge. Was man aus dem täglichen realen Umgang mit Dingen kennt, muss man nicht erlernen. Es ist »kinderleicht«.

Seit langem beweisen kleine, ausgezeichnete Anwendungen, dass man mit Flash auch die Übersichtlichkeit steigern kann. Das mittlerweile berühmte Buchungsmodul des Broadmoor Hotels<sup>1</sup> zeigt, wie es gemacht wird: Alle Schritte der Transaktion sind immer im Blick. Der Nutzer kann selbst wählen, in welcher Reihenfolge er die Schritte ausführt. Nach einer Auswahl in einem Bereich, etwa des Buchungszeitraums, sieht er direkt, welche Konsequenzen dies auf die anderen

Bereiche hat, zum Beispiel die Verfügbarkeit der Zimmer, und kann gegebenenfalls Änderungen vornehmen.

### **Intelligente Clients**

Reine Flash-Anwendungen bezeichnet man als »Rich Clients«: Bei Flash denkt nicht nur der Server, auch der Client im Browser kann seine Darstellung verändern oder Eingaben kontrollieren und reagieren, ohne den Server zu fragen. Er ist »reich« an Funktionalität und Darstellungsmöglichkeiten.

Gegen die »intelligenten« Clients wurden immer wieder die Argumente ins Feld geführt, sie seien nicht blindenlesbar und könnten nicht von Suchmaschinen gelesen werden. Moderne Rich-Client-Systeme basieren jedoch ohnehin auf Webservices und können daher mit geringem Mehraufwand »nebenbei« HTML-Seiten speziell für diesen Einsatz erzeugen. Diese simplen Text-Dokumente sind frei von Gestaltungselementen und daher für Blinden-Browser und Suchmaschinen sogar verständlicher als heutige HTML-Clients.

### **Systeme, die die Sprache der Entwickler sprechen**

Reine Flash-Anwendungen mögen für kleine Einzelanwendungen wie das Dartmoor-Buchungsmodul ausreichen. Größere Systeme arbeiten mit Redaktionssystemen und sind an Datenbanken und komplexe Programme angebunden. Komplexe Clients müssen darüber hinaus langfristig wartbar und erweiterbar sein. Für viele erfahrene Entwickler waren reine Flash-Lösungen daher indiskutabel. Sie suchten nach Lösungen, die sich besser in das Backend integrieren lassen und einen rationaleren Entwicklungsprozess ermöglichen.

Die Firma LaszloSystems in Kalifornien hat diese Technik bereits im Jahr 2000 entwickelt und mit dem »Laszlo Presentation Server« einige hervorragende Anwendungen realisiert. In Europa wurde die Technologie kaum wahrgenommen, bis 2004 der Marktführer Macromedia mit »Flex«<sup>2</sup> nachzog. Laszlo startete daraufhin die Flucht nach vorn und veröffentlichte den Programmcode unter einer Open Source-Lizenz. Seitdem ist der Server kostenlos und auch freie Softwareentwickler arbeiten an der Verbesserung der Software.

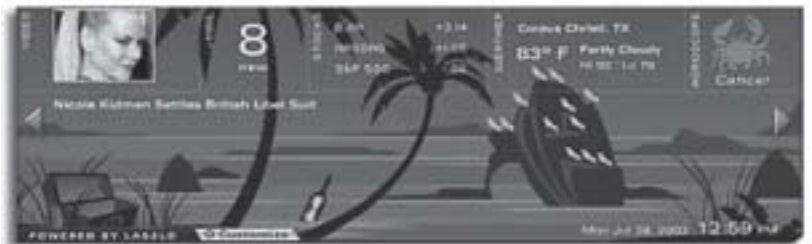
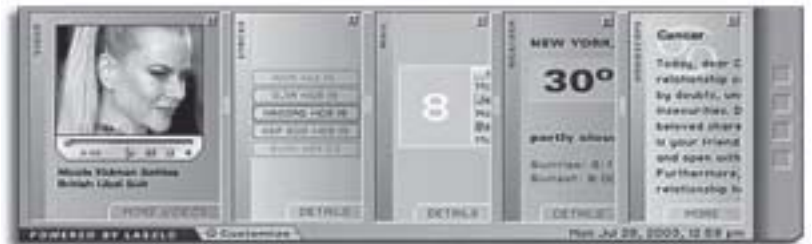
»Die Kunden sahen uns ohnehin immer fragend an, wenn wir von den Lizenzkosten für den Server redeten und sagten: ›Wir hätten gern so etwas wie diese Demo, die Sie uns eben gezeigt haben«. Dafür waren sie gern bereit, die vorgeschlagene Summe zu bezahlen. Wie viel davon auf die Lizenz des Servers entfiel, war ihnen egal. Daher konnten wir den Server auch kostenfrei anbieten – es machte keinen Unterschied für uns«, erklärt Oliver Steele, Chefentwickler von LaszloSystems, den Schritt.

Die beiden Technologien ähneln sich in vielen Dingen. Beide besitzen eine eigene Skriptsprache, die – wie HTML – auf XML basiert. Macromedia Flex bietet eine gute Integration in die neue Generation der Flash Komponenten, funktioniert daher aber auch nur mit dem Flash Player in der Version 7, während Laszlo-Anwendungen auch auf der Version 6 (97,7 Prozent Verbreitungsgrad) laufen. Als Hersteller von Flash kann Macromedia den Vorteil bieten, dass die Zusammenarbeit von Entwickler und Designer optimiert ist.

LaszloSystems ist nicht vom Verkauf des Servers abhängig. Laszlo bietet daher auch die Möglichkeit, Applikationen zu erstellen, für die der Laszlo-Server nicht auf dem Webserver installiert sein muss.

Bei Flex darf man vor allem auf die Zusammenarbeit zwischen Flash und dem weit verbreiteten PDF-Format gespannt sein. Schon vor der Übernahme von Macromedia durch den PDF-Erfinder Adobe wurden erste Ansätze einer Zusammenarbeit zwischen den beiden Formaten erkennbar. Auch die Einbindung von Video-Konferenzen über den Flash Communication Server könnte interessante neue Einsatzgebiete ermöglichen. Diese Felder erschließen sich Flex ganz natürlich, da es vom gleichen Anbieter entwickelt wird wie das Flash-Plugin selbst.

Bei Laszlo dagegen ist die Nutzung des Flash-Clients als Ausgabeformat eigentlich nur der erste Schritt: Open Laszlo war von vornherein darauf ausgelegt, auch andere Formate zu erzeugen. Erste Versuche arbeiten mit DHTML, das auf jedem Browser Plugin-frei angezeigt werden kann. Die Mozilla-Foundation denkt bereits über eine Möglichkeit nach, wie Laszlo-Code direkt im Browser dargestellt



werden kann, ohne dass zuerst eine Umwandlung in ein anderes Format stattfinden muss. Eine XAML-Ausgabe soll es zudem bald ermöglichen, den einmal für das Internet erstellten Code ohne Zusatzaufwand auch als »echtes« Windows-Programm laufen zu lassen. Entwickler wissen solche Aussichten zu schätzen.

### Die Akzeptanz steigt

Zum Glück haben gute Designer mittlerweile bewiesen, dass man mit Rich Clients übersichtliche und sehr nutzerfreundliche Seiten bauen kann, wenn man ihre Fähigkeiten nur dort einsetzt, wo sie die Usability erhöhen. Auch konservative Usability-Gurus wie Jakob Nielsen sind daher von harten Kritikern zu euphorischen Anhängern geworden.

Komplizierter war die Überzeugungsarbeit bei den Entwicklern. Als intensive Netznutzer kennen sie Flash seit seinen Anfängen und assoziieren es vorwiegend mit dem »Wegklicken« der berüchtigten Flash-Intros. Weit schwerer wog zudem ein ganz pragmatischer Aspekt: Die Backends waren zu großen,

Drei wählbare »Skins« (Darstellungen) für ein und dieselbe Anwendung: Email-Programm im Internet. Auf Wunsch zeigen Vögel die vollen Stunden an und die Mail kommt per Flaschenpost.

Alles auf einen Blick:  
Bei Auswahl einer Raum-  
kategorie werden im  
Kalender die freien Tage  
angezeigt. Wer zuerst den  
Zeitraum wählt, sieht die  
(für diese Zeit) verfügbaren  
Zimmer.



komplexen Maschinerien angewachsen. Zum einen, weil die Anforderungen an eine interaktive Webanwendung deutlich größer sind als bei der puren Zusammenstellung von Textkomponenten; zum anderen, weil die Backends alles kompensieren mussten, was die simplen HTML-Frontends nicht leisten konnten. Die Umstellung von HTML auf ein klassisches Flash-Frontend würde für Entwickler bedeuten, dass sie mit völlig neuen Technologien arbeiten und mit ihren über Jahre aufgebauten Systemen von vorne anfangen müssten. Dynamische Rich-Client-Systeme wie Laszlo oder Flex nutzen jedoch ähnliche Technologien und die Entwickler können Teile ihrer bereits entwickelten Backends weiter verwenden.

Trotz der Scheu der Auftraggeber vor neuen Technologien bewegt sich etwas. Als wir einem Kunden vor kurzem eine kleine Demo-Applikation in Laszlo zeigten, war er skeptisch. Auch er verband Rich Clients mit bunten Filmchen ohne größeren Inhalt und Sinn. Um ihn umzustimmen, veränderten wir das Aussehen: Die Darstellung (»Skin«) dynamischer Rich Clients lässt sich mit wenig Aufwand austauschen. Laszlo bietet dazu unter ande-

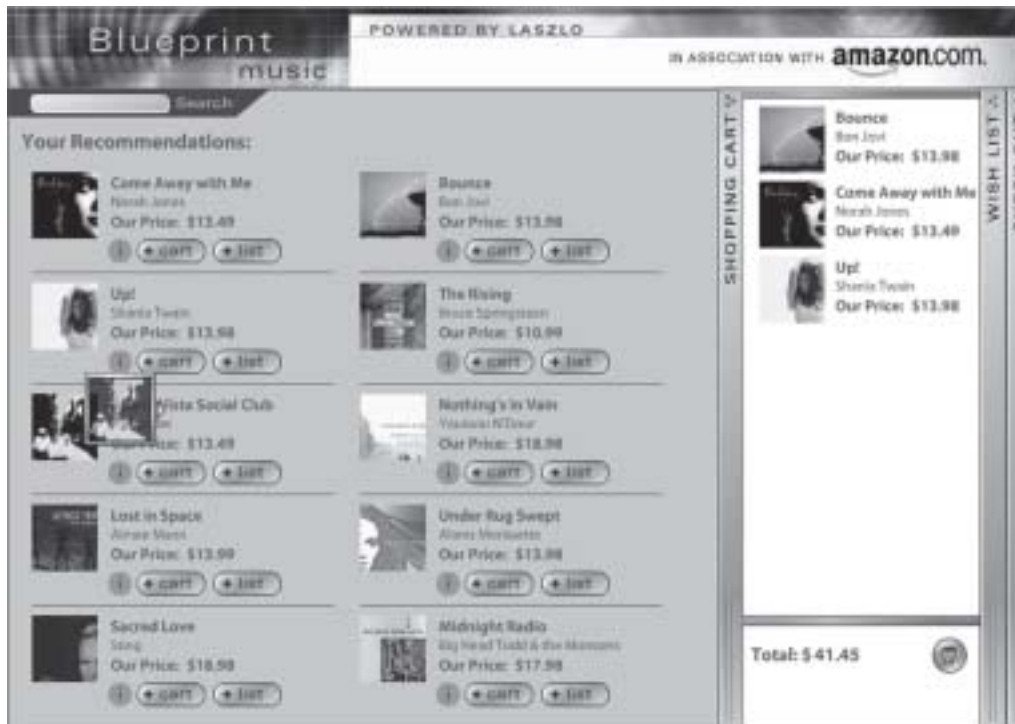
rem die so genannte Redmond-Bibliothek. Sie lässt die Oberfläche aussehen wie eine Windows-Oberfläche. Kaum hatten wir die Kunden-CI durch den Redmond-Skin ersetzt, waren sich alle Beteiligten sofort einig, dass es sich hier um »richtige« Software handelt.

### Die »Großen« steigen um

Vor allem die »Großen« gehen seit neuestem mit gutem Beispiel voran: So hat etwa der Software-Riese IBM im November 2004 ein Laszlo-Plugin für die OpenSource-Entwicklungsumgebung Eclipse herausgebracht. Dieses ist Teil des Emerging Technologies Toolkit von IBMs F&E-Bereich.

Der ERP-Marktführer SAP hatte mit SAP-Portals die Integration verschiedener Datenquellen unter einer HTML-Oberfläche bereits bis zum Äußersten ausgereizt. Jetzt setzt auch SAP auf Rich Clients: Macromedia Flex wurde in die neue Version seiner großen Service-Architecture-Lösung SAP NetWeaver integriert. Begründet wurde dieser Schritt ausdrücklich mit der dringend erforderlichen Verbesserung der Usability.

Darüber hinaus wurde OpenLaszlo für die Kategorie »Best Open Source Product or



Freie Wahl der Waffen: wer das System verstanden hat, kann bequem CDs in den Einkaufskorb ziehen. Gewohnheitstiere nutzen die von »normalen Seiten« gewohnten Knöpfe.

Service« des Codie-Awards 2005 der SIIA (Software & Information Industry Association) nominiert. Frühere Nominierte sind unter anderem MySQL, PHP und viele andere, die sich mittlerweile als internationale Standards durchgesetzt haben.

Es zeichnet sich ein breiter Trend ab: Die Usability-Vorteile von Rich Clients verhelfen der Technologie zum Durchbruch. Auch wenn es noch einige Agenturen und Anwender gibt, die sich diesen Tendenzen widersetzen, wird die Marktmacht der Großen auch die letzten Skeptiker überzeugen.

### Die Macht der Gewohnheit

Moderne Rich-Client-Systeme bieten alles, was zu einer einfachen und intuitiven Nutzung nötig ist: Alle wichtigen Elemente der Seite sind gleichzeitig sichtbar, was bei HTML nicht möglich ist, weil die Serialisierung mehr Platz beansprucht.

Fokussierte Bereiche vergrößern sich, andere verkleinern sich, bleiben aber lokalisierbar. Komplexe Zusammenhänge können als Grafiken und Diagramme dargestellt werden, die sofort zusätzliche Informationen einblenden, wenn man mit dem Mauszeiger über ein

Element fährt. Diese Oberflächen bieten alles, was sich Konzepter, Designer und vor allem Nutzer wünschen.

Überraschenderweise schneiden Rich Clients bei Usability-Tests trotz ihrer sehr intuitiven Bedienbarkeit manchmal gar nicht überlegend ab. Der Grund ist die Macht der Gewohnheit: Viele Internetnutzer sind so an die unnötig umständliche »HTML-Usability« klassischer Websites gewöhnt, dass viel einfachere Möglichkeiten der Bedienung nicht mehr erkannt werden. Dabei kennen Anwender die neuen Vorgehensweisen sehr wohl: von ihrem täglichen Umgang mit der Windows- oder Mac-Oberfläche. Sie erwarten sie nur (noch!) nicht im Internet.

Eine verblüffende Reaktion auf dieses Dilemma zeigt eine Demo für eine Rich-Client-Version des Amazon-Shops.<sup>3</sup> Der Shop »imitiert« die bekannte HTML-Usability: Er bietet dieselben »+ cart«-Knöpfe (»in den Warenkorb«) an wie der Original-Shop. Gleichzeitig ist jedoch das viel intuitivere Klicken und Ziehen möglich. Nicht umsonst spricht man von einem »Warenkorb«. Haben Sie schon einmal auf einen Knopf am Supermarkt-Regal geklickt und das Produkt ist in Ihren Einkaufs-

korb gesprungen? Doch selbst bei der Imitation des HTML-Verhaltens unterstützt der Rich Client die Usability: Wenn der Warenkorb geschlossen ist, zeigt nach dem Klick eine kleine Bewegung am geschlossenen Warenkorb dezent an, dass sich hier etwas verändert hat.

### **Das Web ist »reich« an komplexen Einsatzgebieten**

HTML ist weiterhin dort sinnvoll, wo reine Texte und Bilder dargestellt und untereinander verlinkt werden sollen. Im wachsenden Bereich der Anwendungen werden sich Rich Clients jedoch durchsetzen. Nachteile wie in der Behindertengerechtigkeit und der Auffindbarkeit durch Suchmaschinen können

durch die Kombination beider Technologien ausgeglichen werden.

Informations- und Marketingsites werden durch ihren wachsenden Umfang immer mehr zu Anwendungen. Hier sind ebenso Rich Clients die Zukunft, da sie im Internet wie Windows-Programme funktionieren – so einfach benutzbar – und man sie von überall über das Netz erreichen kann.

- 1 [www.reservations.broadmoor.com](http://www.reservations.broadmoor.com)
- 2 Nach Fertigstellung des Artikels kündigte Macromedia die Testausgabe (»Alpha Release«) einer neuen Version von Flex an. Die Features dieser Version konnten hier aufgrund des Redaktionsschlusses leider nicht mehr berücksichtigt werden.
- 3 <http://www.laszlo.com/demos/>