

Bewerten und Entdecken von Patterns in E-Business Anwendungen

Jutta Eckstein
jeckstein@acm.org
www.jeckstein.com

Mit dem Web haben sich E-Business Applikationen durchgesetzt. Diese Applikationen kombinieren die Navigation durch einen elektronisch vorliegenden Katalog mit Operationen auf diesem Katalog. In diesem Sinne sind E-Business Applikationen lediglich eine bestimmte Art von Web Applikationen mit ähnlichen Anforderungen: Gute Navigierbarkeit, ansprechende und akzeptable Benutzerschnittstelle, ein klares Geschäftsmodell, etc.

Dabei stellen aber E-Business Applikationen eine neue Herausforderung an die Designer: Sie müssen nicht nur den Benutzer dahingehend unterstützen, dass er findet was er sucht (ein Produkt, das er kaufen wird), sondern zusätzlich noch den Einkaufsprozess einfach gestalten. Zum Beispiel soll der Benutzer über Neuerscheinungen informiert werden und ausserdem möglichst lange im Shop gehalten werden.

Ablauf

- Einführung und Vorstellung einiger Beispiel Patterns (30 min)
- Pattern Mining (45 min)
 - Brain Storming über Probleme in E-Business Anwendungen
 - Andenken von Lösungsmöglichkeiten
 - Skizzieren einzelner Patterns
- Zusammenfassung (15 min)
 - Vorstellen einzelner Patterns
 - Weiterentwicklung der Patterns

Ziel dieser Veranstaltung ist einerseits das Kennenlernen von Mustern im Bereich von E-Business Anwendungen, und andererseits durch aktive Teilnahme Kenntnisse darüber zu erlangen, wie Muster entstehen.

Inhalt

- Patterns
- Bereiche und Ziele von E-Business Patterns
- Beispiel Patterns
- State-of-the-Art
- Pattern Mining (Workshop)
- Zusammenfassung und Referenzen

Patterns: Sinn und Zweck

- Stellen Expertenwissen zur Verfügung
- Kommunikationsmittel
 - Vokabular von Lösungen
 - Erfahrungsaustausch
- Lösung zu wiederkehrenden Problemen
- wägen das Für und Wider ab um ein Gleichgewicht herzustellen
- können mit anderen Patterns kombiniert werden

Grundsätzlich dienen Patterns als Kommunikationsmittel: Einerseits stellen sie ein Vokabular für Experten zur Verfügung und andererseits beinhalten sie Expertenwissen das zum Erfahrungsaustausch verwendet werden kann, vor allem für „Neulinge“. Alle Patterns sind ähnlich aufgebaut: So bestehen sie immer aus einer Lösung zu einem wiederkehrenden Problem.

Missverständnisse zu Patterns

- Machen Software o.a. billiger
- Machen Software schneller
- Neu für Experten
- Erfindung des Autors
- Tricks
- GoF
- OOP

Viel wurde über Patterns geschrieben und vieles wurde ihnen auch angedichtet, deshalb erst mal der Versuch klarzustellen, wozu Patterns *nicht* beitragen: Sie machen Software weder billiger noch schneller. Die meisten der bekannten Design Patterns [5] verlangsamen Software sogar eher. Weiterhin sind Patterns niemals neu für Experten – das ist oft die größte Frustration, wenn man anfängt sich für Patterns zu interessieren. Man entdeckt so gut wie nichts neues – vorausgesetzt man ist in der Domäne ein Experte. Ein Pattern ist auch niemals von dem entsprechenden Autor erfunden. Zum einen ist ein Pattern meist eine Gemeinschaftsarbeit von mehreren Autoren und andererseits entstehen Patterns aus dem Erfahrungsschatz und können von daher nicht neu erfunden sein – ansonsten wären es keine Patterns.

Patterns stehen auch nicht notwendigerweise mit objekt-orientierter Programmierung in Zusammenhang. Außerdem sind Patterns auch nicht auf Design Patterns beschränkt. Neben Design Patterns gibt es noch eine ganze Menge anderer Patterns, die sich auf andere Domänen außerhalb des Design-Bereichs konzentrieren, zum Beispiel: Architektur, Technische Dokumentation, Analyse, oder auch konkreter: Telekommunikation, Pädagogik, etc.

Definition

- “Each pattern describes a problem which occurs over and over again in our environment, and then describes the core of the solution to that problem, in such a way that you can use this solution a million times over, without ever doing it the same way twice.”

Christopher Alexander

Jedoch muss die Lösung eher als Lösungsweg betrachtet werden, da sie unterschiedlich implementiert werden kann. Dies ist auch die eigentliche Intention eines Patterns, d.h. ein Pattern kann hundertmal eingesetzt werden, ohne dass dabei auch nur einmal exakt die gleiche Lösung verwendet wird.

Würde ein Pattern eine exakte Lösung vorgeben, so würde es sich dabei weniger um ein Pattern, sondern eher um einen Kniff oder ein Kochrezept handeln.

Bereiche einer E-Business Applikation

- Präsentation
 - Darstellung, User Interface, ...
- Navigation
 - Auffinden der Informationen, Orientierung, Berücksichtigen unterschiedlicher Benutzer, ...
- Applikation
 - Umsetzung, z.B. Session-Management, Client-Management, ...

Es gibt unterschiedliche Bereiche, die von einer E-Business Applikation abgedeckt werden müssen. Daraus folgt, dass auch E-Business Patterns diesen unterschiedlichen Bereichen zugeordnet werden können. Zunächst muss dem Bereich der *Präsentation* eine besondere Bedeutung beigemessen werden: Typisch für diese Art Anwendung ist, dass der Entwickler nicht weiß auf welcher Plattform die Anwendung später laufen wird. Man kennt weder das Betriebssystem, d.h. das verwendete Look and Feel, noch die Version des Browsers oder die Auflösung des Bildschirms. Weiterhin ist die Konkurrenz an anderen verfügbaren Anwendungen natürlich groß und der erste Eindruck entscheidet, ob der Kunde beispielsweise in dem Shop bleibt oder ob er ihn frustriert wieder verlässt. Insofern muss diesem Bereich vom gestalterischen her ein besonderer Raum eingeräumt werden.

Der von den bisher entwickelten E-Business Patterns am besten abgedeckte Bereich ist die *Navigation*. Darunter ist weniger die Verwaltung oder der Umgang mit Events gemeint, sondern vielmehr wie der Anwender die gesuchte Information findet, und wie er sich weiterhin orientieren kann. Ziel ist, dass er ständig weiß wo er sich befindet, bzw. wo er herkam und auch wie er wieder dorthin zurückkommt, außerdem wie er dieselbe Information Tage später wiederfinden kann. Dabei müssen zusätzlich die unterschiedlichen Benutzer und Benutzergruppen berücksichtigt werden.

Der dritte und letzte Bereich ist die *Applikation*: Hierbei geht es um die eigentliche Umsetzung, d.h. in diesem Bereich fallen teilweise auch die herkömmlichen Design-Patterns, teilweise finden sich hier auch spezielle Patterns wieder. Zum Beispiel, wenn es darum geht wie eine Session verwaltet werden soll, oder wie sich eine bestimmte Bandbreitengröße auf die Anwendung auswirkt, usw.

Ziele von E-Business Patterns

- **Personalisierung**
 - auf individuelle Bedürfnisse von Benutzern und Benutzergruppen eingehen, ...
- **Vereinfachen des Einkaufens**
 - der Kunde soll im Einkaufsprozess unterstützt werden, Transparenz im Zahlungsvorgang, ..
- **Umsatzsteigerung**
 - der Kunde soll möglichst lange im Shop bleiben, Hinweis auf “Schnäppchen”, ...

Orthogonal zu den Bereichen verfolgen die E-Business Patterns bestimmte Ziele: Zunächst soll sich die Anwendung an die individuellen Bedürfnisse der Benutzer und Benutzergruppen anpassen, d.h. die Anwendung soll personalisiert werden können. Wichtigere Ziele sind jedoch, dass der Einkaufsprozess vereinfacht werden soll. Dabei soll der reale Einkaufsprozess nachgestellt und der Kunde während des gesamten Einkaufsvorgangs unterstützt werden.

Das wichtigste Ziel für den Shop Anbieter ist die Umsatzsteigerung. So soll der Kunde beispielsweise möglichst lange im Shop bleiben, auch dann wenn er bereits gefunden hat was er ursprünglich suchte.

Beispiel Patterns

- Content Personalization
 - Ziel: Personalisierung
- Opportunistic Linking
 - Ziel: Umsatzsteigerung
- URL Affordance
 - Ziel: Vereinfachen des Einkaufens
- Haupt-Autoren
 - Gustavo Rossi
 - Fernando Lyardet

Als Beispiele folgen einige Patterns von Gustavo Rossi und Fernando Lyardet. Da Patterns zum weltweiten Erfahrungsaustausch dienen, werden die Patterns üblicherweise in Englisch verfasst.

Content Personalization

- Problem:
 - Provide the user with personalized contents in nodes.
- Forces:
 - Web applications deal with thousands of objects which again have many attributes
 - We want to show different values of the same attribute to different users (e.g. different price policies)

Aus: Rossi, Gustavo, Schwabe, Daniel, Danculovic, Juan, Miaton, Leonardo. *Patterns for Personalized Web Applications*. Proceedings of EuroPLoP 2001.

Content Personalization

- **Solution:**
 - Define personalized contents in nodes by varying the attributes according to the user. This means decoupling the attribute value from it's object and coupling it with the user
- **Consequences:**
 - Improve personalization, since the same object may look different when accessed by different users.
 - There is an increased content authoring effect, since differentiated content must be generated for each possible combination

Content Personalization

- Known Uses:
 - Individual Pricing, e.g. www.half.com
 - Presenting different text for the same item according to competences, e.g. ATL (a mobile phone company in Rio de Janeiro)

Bewerten und Entdecken von Patterns in E-Business Anwendungen

The screenshot shows the Half.com website's shopping cart. The cart contains the following items:

Item	Item Price	Media Price
Eternity: The Fading Stone (Music) CD, Release Year: 1997 Condition: Like New - Discs: UNABRIDGED QUALITY USED, CD/DVD CASE, IS-9-488494888, 9-481 548-541	\$16.99	\$2.00
LIZ: RUTH AND IAN: A MOVIE DVD, Release Year: 1997 Condition: Like New - Discs: UNABRIDGED QUALITY USED, CD/DVD CASE, IS-9-488494888, 9-481 548-541	\$23.99	\$2.00
Subtotal	\$40.98	
Other discounts (\$4.00) Total discounts applied (\$4.00)		
TOTAL: \$81.74		

Below the cart, there is a section for "Get the Backlist Complete" with a text input field and a "Backlist" button. The text reads: "Redeeming your Half.com Gift Certificate is easy. Just enter your claim code in the box to the right and click 'Redeem'. If you do not have a Half.com account you will be prompted to create one after entering your claim code." Below this is a note: "Place the Backlist Complete. Your savings have been applied."

Jutta Eckstein, Objects in Action

OOP 2002 13

Der Discount über 5 \$ ist personalisiert insofern, dass er auf Informationen über die Kundenhistorie basiert.

Opportunistic Linking

- Problem:
 - Keep the user interested in the site. Seduce him to navigate even when he has already found what he was looking for.
- Forces:
 - We want to keep the user navigating in our store even after he bought something
 - We don't want to compromise the structure of the store by adding not meaningful links

Aus: Rossi, Gustavo, Lyardet, Fernando, Schwabe, Daniel. *Patterns for e-commerce applications*. Proceedings der EuroPLoP 2000. UKV Verlag Konstanz. 2001.

Meist wird eine Website mit einem bestimmten Ziel besucht, z.B. ein bestimmtes Produkt zu kaufen.

Der Kunde soll in der Site auch dann interessiert sein, wenn er bereits gefunden hat was er suchte.

Opportunistic Linking

- **Solution:**
 - Improve the linking topology by suggesting new products to explore from a given one.
 - Use relationships with strong semantics to make the user feel comfortable.
 - Take into account that many of these links may change from day to day.
- **Known Uses:**
 - www.amazon.com. Once the user has chosen a book, he receives a suggestion of another (related) book.

Verbessern der Link-Topologie, durch generieren von semantischen Beziehungen.

Ausgehend von einem bestimmten Produkt (dem gekauften) werden weitere Produkte vorgeschlagen die eine starke Affinität zum Ausgangsprodukt haben, so dass der Kunde sich persönlich angesprochen fühlt.

Diese neuen Links unterliegen ständigen Änderungen.

Bewerten und Entdecken von Patterns in E-Business Anwendungen



Jutta Eckstein, Objects in Action

OOP 2002 16

Typisches Beispiel von Amazon. Sobald man gefunden hat was man sucht erhält man direkt Empfehlungen für weitere Produkte, die mit dem gewählten Produkt in Zusammenhang stehen.

URL Affordance

- Problem:
 - How to design a useful URL.
- Forces:
 - URLs are an important part of the current web user interface because they are the mean to specify what we want to see and where.
 - Since domains are being assigned at an impressive rate, finding a 'good name' is a time-consuming task.
 - Furthermore, many websites rely on the dynamic generation of contents and they tend to underestimate the use of URLs by the users to access the information they need.

Aus: Lyardet, Fernando und Rossi, Gustavo *Web Usability Patterns*. Proceedings der EuroPLoP 2001.

Web besteht aus Links und Nodes. Die Links = URLs beschreiben die Adresse eines Objektes. Dabei sagt die URL aus wo und wie das Objekt gefunden und aktiviert werden kann. Über die URL wird das Objekt/das Dokument bestimmt, d.h. URLs beschreiben den Zugriff auf das Objekt und sollten von daher verständlich sein.

ABER: Oft werden die Inhalte einer Website generiert. Dabei wird oft die bedeutung der URL für den Benutzer unterschätzt.

Der Benutzer ist dazu gezwungen, jedes Mal wieder über die Homepage zur gewünschten Information zu navigieren.

URL Affordance

- <http://www.amazon.com/exec/obidos/subst/home/music.html/103-1306596-6860655>

versus

- <http://www.amazon.com/music>

Die URLs sind oft überwältigend und reflektieren nicht die Information
Verwenden von sprechenden und einfach zu merkenden URLs.
Die URL sollte die Struktur der Information reflektieren

URL Affordance

- Solution:
 - Design the URL as a complementary tool for the user to find and access the contents of the website.
 - A domain name that is easy to remember and easy to spell.
 - Short URLs.
 - Easy-to-type URLs.
 - Persistent URLs that don't change to keep your customer bookmarks valid.
 - Single case name (DoNoTmixTheM.com)
 - Use a spelling-checking web server to absorb common typos.
 - Design URLs that help visualize the site structure.

URL Affordance

- Consequences:
 - Easier to remember URLs
 - Reduces required amount of navigation to access a determined page.
 - Force a wide hierarchy organization, based on a breath of choices.
- Known Uses:
 - Microsoft, e.g. www.microsoft.com/sql/

Weitere E-Business Patterns

- Printer Friendly
- Subjective Access
- Explicit Process
- Secure Backtrack
- Push Communication
- ...und viele mehr

Printer Friendly:

- auch das Internet hat nicht allzu viel zum papierlosem Büro beigetragen
- direktes Ausdrucken ist oft problematisch: Font, Typefaces, Grafiken, Spalten, etc.
- => spezielle Version zum Ausdrucken zur Verfügung stellen

Subjective Access:

- E-Business hat die Entwicklung vieler neuer Services erfordert. Manche davon werden an spezialisierte Firmen weitergegeben, z.B. Bezahlung
 - Dadurch entsteht oft das Problem, dass sich der Benutzer plötzlich auf einer ganz anderen Website wiederfindet
- ⇒ es sollte eine kunden-orientierte View dargestellt werden, deren Darstellung sich danach richtet wo der Benutzer herkommt.

Explicit Process:

- Der Zahlungsprozess ist oft kompliziert und nicht durchschaubar
 - der Kunde ist disorientiert (weiss nicht was noch kommt, was er bereits getan hat...)
- ⇒ die einzelnen Schritte explizit machen

Secure Backtrack:

- ‚Back‘ bei komplexen Prozessen kann zu inkonsistenten Zuständen führen. Evt. wird dabei die gesamte Information verloren, ausserdem gibt es hier häufig Sicherheitsprobleme
- ⇒ zusätzlich ‚undo‘ und ‚edit‘ anbieten mit denen die Informationen auf definierte Weise geändert werden können ohne in inkonsistenten Zustand zu kommen

Push Communication

- Web ist typischerweise Pull-based (Stichwort Hol-/Bringschuld)
 - manchmal ist der Kunde aber grundsätzlich an bestimmten Themen interessiert und möchte darüber informiert werden (ohne umständlich dahin zu surfen)
- ⇒ z.B. per e-mail die entsprechenden links verschicken

State-of-the-art

- E-Business Patterns stecken noch in den Anfängen
 - Circa 30 Patterns sind bekannt
 - die Autoren kommen aus ca. fünf verschiedenen Ländern
 - **eine Menge Expertenwissen ist noch nicht niedergeschrieben**
- Existierende E-Business Pattern Languages
 - Design Patterns for Hypermedia Applications
 - Patterns for Designing Navigable Spaces
 - Web Usability Patterns
 - Patterns for Personalized Web Applications
 - Patterns for E-Commerce Applications

Da die Domäne E-Business noch recht jung ist, sind die Muster die das Expertenwissen auf diesem Gebiet zusammenfassen, selbst noch in der Validierungsphase.

Der Projektstart erfolgte jedoch bereits 1995, federführend von Gustavo Rossi unterstützt von Fernando Lyardet. Zwischenzeitlich gibt es circa 30 Patterns [6], die Autoren sind weltweit verstreut jedoch befindet sich das Zentrum in Südamerika, dort hauptsächlich in La Plata, Argentinien.

Pattern Mining

- Brain Storming über Probleme und Herausforderungen im definierten Themenbereich
- Jedes Problem / jede Herausforderung wird in **rot** auf ein separates Flipchart notiert
- wird bereits eine Lösung angedacht, so wird diese in **grün** auf das gleiche Flipchart notiert
 - Fokus liegt jedoch auf den Problemen

Da nahezu jeder schon einmal mit einer E-Business Anwendung umgegangen ist, ist auch jeder auf gewisse Art und Weise bereits Experte auf dem Gebiet. Über Pattern Mining soll das darin schlummernde Expertenwissen zutage gefördert werden.

Die Patterns oder Teilpatterns die dabei gefunden werden, können die Autoren entweder selbst weiterentwickeln, oder sie werden durch die Projektleiter Gustavo Rossi und Fernando Lyardet weiterentwickelt. Eine Möglichkeit selbst die Patterns weiterzuentwickeln besteht darin sie bei einer Patterns Konferenz, zum Beispiel der EuroPLoP (www.hillside.net/patterns/EuroPLoP), die in Irsee im Allgäu stattfindet, einzureichen.

Pattern Mining

- *Ausstellung*
 - alle gefundenen Probleme / Herausforderungen werden direkt ausgestellt
 - jeder geht durch die Ausstellung und notiert Kommentare und dazugehörige Probleme auf die Flipcharts
 - auch erste Lösungsansätze können notiert werden

Entwurf eines Patterns

- gibt es keine neuen zusätzlichen Ideen mehr zu einem Problem:
 - wird das Pattern skizziert
 - nach Möglichkeit werden alle Bestandteile ausgefüllt
- Wer an dem Pattern weiterarbeiten möchte hinterlässt Name und email-Adresse auf dem Flipchart

Bestandteile

- **Name** – gut zu merken, oft Hinweis auf die Lösung
- **Context** – Randbedingungen, oder: Unter welchen Umständen tritt das Problem auf?
- **Forces** – Was macht das Problem, oder die naheliegende Lösung so schwierig?
- **Problem**
- **Solution**
- **Examples / Known Uses**
- **Consequences** – Welche weiteren Probleme löst die Lösung aus? Was sind die Nachbedingungen bzw. der Post-Context?
- **Author**

Schreiben von Patterns

- Iterativ
- mit Unterstützung:
 - Durch Einreichen beispielsweise bei der EuroPLoP
 - www.hillside.net/patterns/EuroPLoP
 - einzelner (Shepherd)
 - mehrerer (Writers' Workshop)
- Rückkopplung
 - durch Anwender

Natürlich können die Patterns auch bei jeder anderen Patterns Konferenz eingereicht werden:

www.hillside.net/conferences

Zusammenfassung

- E-Business Anwendungen versuchen das *normale* Verkäufer-Käufer Verhalten auf das Internet abzubilden
- Patterns sind ein geeignetes Mittel, um das Expertenwissen über die Komplexität dieser Abbildung, zu sammeln

E-Business Anwendungen versuchen das normale Käufer-Verkäufer Verhalten auf das Web abzubilden. Die Komplexität die diese Abbildung mit sich bringt, muss von allen Beteiligten immer wieder bewältigt werden. Zur Reduzierung der Komplexität bietet sich an, das Expertenwissen zu sammeln und öffentlich zugänglich zu machen. Patterns sind ein effektives Mittel um genau dieses zu tun. Sie bieten einerseits ein Format um Erfahrungen auszutauschen und andererseits gibt es für die Entwicklung von Patterns einen definierten Prozess, der die Validierung und Veröffentlichung von Patterns unterstützt.

Erfahrungen mit E-Business Anwendungen in Form von Patterns zu erfassen, bietet einen weiteren Vorteil. Durch Kombination dieser Patterns können dann nämlich noch komplexere Probleme auf diesem Gebiet gelöst werden.

Referenzen

- Home der E-Business Patterns: www.designpattern.lu.unisi.ch/index.htm
- Lyardet, Fernando, et.al. *Web Usability Patterns*. Proceedings der EuroPLoP 2001.
- Rossi, Gustavo, et.al. *Patterns for Personalized Web Applications*. Proceedings der EuroPLoP 2001.
- Rossi, Gustavo et.al. *Patterns for e-commerce applications*. Proceedings der EuroPLoP 2000. UKV Verlag Konstanz 2001.
- Rossi, Gustavo et.al. *Patterns for Designing Navigable Spaces*. In: Harrison, Foote, Rohnert (eds), *Pattern Languages of Program Design 4*. Reading, MA: Addison-Wesley, 2000.
- Rossi, Gustavo et.al. *Design Patterns for Object-Oriented Hypermedia Applications* in: Vlissides, J., Coplien, J., Kerth N., *Pattern Languages of Program Design 2*. Reading, MA: Addison-Wesley, 1995.