H. Wandke, S. Kain & D. Struve (Hrsg.): Mensch & Computer 2009: Grenzenlos frei!?
München: Oldenbourg Verlag, 2009, S. 353-362

# Konsistenz und Effizienz durch User Interface Pattern – ein Praxisbericht

Sonja Auer, Anke Richter

Siemens AG

**Zusammenfassung**

Mit User Interface (UI) Pattern lassen sich bewährte User Interface Lösungen systematisch beschreiben und zugänglich machen. Eine spezifische Lösung für ein Produkt wird abstrahiert und für andere Produkte verfügbar gemacht. Der vorliegende Artikel beschreibt die Anwendung von UI Pattern in der Energieautomatisierung. Hier zeichnen sich Software-Applikationen durch eine hohe Komplexität und Heterogenität aus. Es wird gezeigt wie sich UI Pattern in den Produkt-Entwicklungsprozess integrieren. Eine Strukturierung der UI Pattern Bibliothek nach User Intentions ermöglicht die Zuordnung von abstrakten Benutzeraufgaben zur konkreten Darstellung auf dem User Interface. Bei der UI Konzeption kann somit ein schneller Überblick über relevante UI Pattern gewonnen werden. Die Wiederverwendung von UI Pattern steigert die Effizienz in der Konzeptionsphase und ermöglicht für die späteren Endanwender von Automatisierungssoftware eine hohe Konsistenz in der Bedienung.

## 1 UI Pattern in einer globalen Produktentwicklung

Wiederverwendbare Bausteine – sogenannte UI Pattern – leisten einen wichtigen Beitrag zur Verbesserung der Prozesse einer globalen Produktentwicklung. Bedienkonzepte können effizienter erstellt werden, weil auf vorhandenem Wissen aufgebaut wird. Die Bedienkonzepte verschiedener Produktfamilien können nach und nach vereinheitlicht werden. Somit kann eine hohe Konsistenz in der Bedienung erreicht werden. UI Pattern sind kein Selbstzweck. Die Kosten, die durch die Beschreibung und Pflege von UI Pattern entstehen, müssen in der Organisation zu einem sichtbaren Mehrwert führen. Dieser Praxisbericht gibt einen Überblick darüber, wie UI Pattern in der Softwareentwicklung bei Siemens Energy eingesetzt werden. Das konkrete Anwendungsfeld ist die Energieautomatisierung. Die Erkenntnisse lassen sich auch auf andere Anwendungsfelder übertragen.

Die Energieautomatisierung beschäftigt sich damit, Stromausfälle zu verhindern und die Versorgungssicherheit zu gewährleisten. Die Aufgaben reichen von der Kraftwerkseinsatzplanung und der Überwachung des Netzzustands bis zur Analyse von Störfällen. In diesem Anwendungsfeld ergeben sich verschiedene Herausforderungen an die Softwareentwicklung:

- Die Anwender haben vielfältige Aufgaben in einem hochkomplexen Umfeld zu erledigen; die Ausführung dieser Aufgaben findet in unterschiedlichen Kontexten statt z.B. in zentralen Leitwarten, im Back Office, in Call Centern oder direkt vor Ort im Feld bei den Umspannwerken;
- Es existiert eine große Anzahl und Heterogenität von Produkten, die teilweise über Jahrzehnte gewachsen sind und unterschiedliche Technologien verwenden (z.B. Web Applikationen, WinCC, qt, .Net);
- Die zunehmende Globalisierung erfordert, dass Entwickler in unterschiedlichen Zeitzonen organisiert werden müssen; Missverständnisse und Inkonsistenzen werden dadurch unvermeidbar; die Entwicklung benötigt Hilfsmittel, um Missverständnissen vorzubeugen.

Bereits erfolgreich umgesetzt wurde ein standardisierter Entwicklungsprozess, der Projekten klare Phasen und Meilensteine vorgibt. Hierdurch wird eine Grundqualität gesichert, jedoch wird noch keine konsistente Bedienung über verschiedene Produkte hinweg gewährleistet. An dieser Stelle ergibt sich der Ansatzpunkt für UI Pattern.

# 2 User Interface Pattern

Der Pattern Ansatz ist nicht neu. Er geht auf Christopher Alexander (1979) zurück, der als Begründer des Pattern Ansatzes gilt. Ziel war es, mit Pattern den Einwohnern bei der Gestaltung ihrer Lebensräume zu mehr Selbstbestimmung zu verhelfen. Ein Pattern beschreibt eine bewährte Lösung und gibt konkrete Hinweise, wann es einzusetzen ist (use when). Das grundsätzliche Konzept, Bausteine systematisch einzusetzen und wiederzuverwenden, wurde von der Softwareentwicklung aufgegriffen (Beck & Cunningham 1987, Gamma et al 1995, Buschmann 1996) und wird seitdem bei der Programmierung und der Entwicklung von Software-Architekturen eingesetzt.

Der Schritt liegt nahe, Pattern auch bei der konzeptionellen Entwicklung von Bedienoberflächen einzusetzen. Erstmalig übertragen Norman & Draper (1986) den Pattern Ansatz von Alexander auf das HCI Gebiet. Bekannter wurde das Thema User Interface Pattern 2001 mit der Veröffentlichung von Jan Borchers (2001) und durch die Website „Common Ground“ von Jenifer Tidwell, welche eine der ersten UI Pattern Sammlungen im Internet war (Tidwell 1999). Später folgten dann weitere Sammlungen wie beispielsweise der Web-Design Katalog von Martijn van Welie (2008). Kürzlich veröffentlicht wurde die UI Pattern Sammlung von Infragistics (2009).

Im Wesentlichen können UI Pattern Bibliotheken zwei Gruppen zugeordnet werden:

*Domänenübergreifende-generische Bibliotheken*

Die eine Gruppe sind Sammlungen mit UI Pattern für unterschiedliche Domänen. Es werden keine direkten Abhängigkeiten zwischen Pattern aufgezeigt (Tidwell, Infragistics). Tidwell (2006) unterscheidet daher auch nicht in der konkreten Ausprägung eines Patterns auf dem

UI, beispielsweise fallen unter „Card Stack“ sowohl „Tab Cards“ als auch „Collapsible Panels“. Hier wird ein guter Überblick über verschiedene Darstellungsmöglichkeiten eines Patterns gegeben, aber kein Hinweis, welche Darstellung ein konkretes Problem am besten löst. Domänenübergreifende-generische Bibliotheken eignen sich, um sich Wissen über gute UI Lösungen anzueignen. Weniger gut eignen sich diese Sammlungen, wenn für eine spezifische Dömane eine bewährte Lösung gesucht wird.

*Domänenspezifische Bibliotheken*

Domänenspezifische UI Pattern Bibliotheken beschreiben UI Pattern für spezifische Anwendungsfelder. So bei Borchers (2001), Yahoo (Malone et al. 2005), SAP (Arend & Peters 2008) und Kunert (2006). Zur Lösung von komplexen Design-Herausforderungen innerhalb einer Domäne können eine Vielzahl an UI Pattern und UI Pattern-Kombinationen erforderlich sein (Deng et al. 2005). Ein systematisierter Zugang, bei dem die Zusammenhänge der einzelnen UI Pattern aufgezeigt werden, unterstützt einen effizienten und konsequenten Einsatz. Neben der hierarchischen Strukturierung und Abgrenzung einzelner UI Pattern untereinander kann der Zugang über konkrete domänespezifische Aufgaben (zum Beispiel „Patientenakte erstellen“ bei der Domäne Krankenhausinformationssystem) erleichtert werden. Einige UI Pattern Sammlungen bieten zudem Entscheidungsfilter (Greene et al. 2003), Querverweise oder verschiedene Suchzugänge als Einstieg zu den beschriebenen UI Pattern an.

Im Gegensatz zu generischen UI Pattern Bibliotheken lassen sich UI Pattern von domänenspezifischen Bibliotheken weniger gut auf andere Anwendungsfelder übertragen. Synergien, die sich durch den Austausch von UI Pattern zwischen Anwendungsdomänen ergeben, können nicht genutzt werden.

Wenn die Produkte sehr heterogen sind, wie bei Siemens Energy, ergibt sich zusätzlich die Schwierigkeit, dass UI Pattern, die ausschließlich anhand von typischen Aufgaben strukturiert und beschrieben werden, zu spezifisch sind und daher kaum auf andere Applikationen innerhalb der Domäne übertragen werden können. Für Siemens Energy wird deswegen eine domänenspezifische UI Pattern Bibliothek benötigt, deren Pattern-Beschreibungen abstrakt genug sind, um den Austausch zwischen Produkten zu ermöglichen, jedoch konkret genug sind, dass an spezifische Aufgaben innerhalb der Domäne angeknüpft werden kann.

# 3 User Interface Pattern in der Praxis

Seit zwei Jahren wird die UI Pattern Bibliothek für die Energieautomatisierung kontinuierlich aufgebaut und erweitert. Grundlage für die Identifikation waren acht unterschiedliche Produkte, Style Guides, Design Spezifikationen und Konzeptentwürfe. Bei der Analyse konnten verschiedene Lösungen für identische Fragestellungen identifiziert und vereinheitlicht werden. Schlussendlich wurden 135 UI Pattern in die UI Pattern Bibliothek aufgenommen, welche nach und nach beschrieben werden. Die UI Pattern Bibliothek wird über die interne Entwicklungs-Homepage für alle Kollegen aus Produktmanagement und Entwicklung zur Verfügung gestellt. Die Verwendung von UI Pattern wurde in die Prozessdokumen-

tation und Schulungsunterlagen aufgenommen, um die Bibliothek in der Organisation zu etablieren.

Bei Siemens Energy ist die UI Pattern Bibliothek der zentrale Einstiegspunkt für die Konzeption und Ausgestaltung von User Interfaces. Die Bibliothek beinhaltet domänenspezifische Lösungen für die gesamte Produktpalette. Neben Beschreibungen der einzelnen Pattern wird in der Bibliothek auf vorhandene Dokumente wie Design Spezifikationen, Style Guides oder Reuse Komponenten verlinkt (Abb. 1). In Design Spezifikationen werden alle Screens und Abläufe für eine spezifische Applikation dokumentiert. In Style Guides können sich die Entwickler über Farben und Bemaßungen der einzelnen Controls informieren. Reuse Komponenten sind wieder verwendbare Bausteine für die Implementierung. Sie beinhalten vorgefertigten Code und ermöglichen eine effiziente Realisierung von User Interfaces. Reuse Komponenten, Design Spezifikationen und Style Guides unterstützen bei der Realisierung von UI Konzepten. Die UI Pattern Bibliothek unterstützt hauptsächlich bei der Konzeption von User Interfaces.

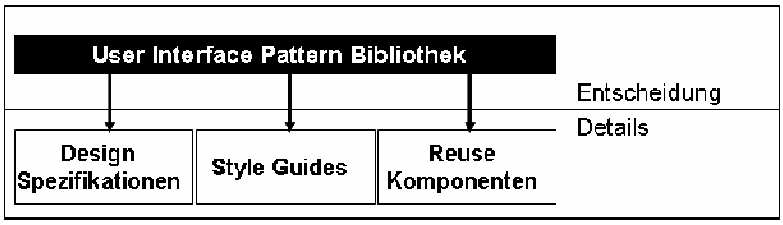


*Abbildung 1: Die UI Pattern Bibliothek als zentraler Einstiegspunkt für die Konzeption und Ausgestaltung von UIs*

## 3.1 User Intention Oriented Approach

Die Strukturierung der UI Pattern Bibliothek erfolgt nach sogenannten generischen User Intentions (Auer & Richter 2005). „User Intentions“ sind generische Operationen auf hohem Abstraktionsniveau (Viewing, Changing View, Editing, Selecting, Doing/Executing). Zusätzlich verschaffen Unterstrukturierungen (Sub-Intentions) eine Orientierung, und zeigen anhand von „Use when“ Kriterien auf, welche UI Pattern für die jeweils spezifische Fragestellung geeignet sind (Abb. 2). Eine dem „Viewing“ zugeordnete Sub-Intention ist zum Beispiel „Attracting Attention“. Mit weiteren Entscheidungskriterien wird dem Leser Hilfestellung gegeben, ob zum Beispiel „Attracting Attention“ die richtige Wahl für seine Fragestellung ist, oder ob nicht eine andere Sub-Intention wie zum Beispiel „Computer Status Feedback“ zutreffender ist. Das gleiche Prinzip wird auf der Lösungsebene („Thermometer“, „Quality Codes“ etc.) fortgeführt. Zusätzlich zu den „Use when“ Kriterien werden die Lösungen anhand von „Do not use when“ Kriterien voneinander abgegrenzt und auf alternative Lösungen verlinkt. Die Entscheidung, welche Lösung für einen spezifischen Anwendungsfall am besten geeignet ist, trifft der Nutzer der UI Pattern Bibliothek anhand dieser Kriterien selber.

Die Strukturierung nach generischen User Intentions ermöglicht einen problemorientierten Zugang zu den spezifischen UI Pattern. Die Anwender der Bibliothek erhalten einen schnellen Überblick über alle UI Pattern, die dasselbe Problem lösen. Der hohe Abstraktionsgrad ermöglicht die Übertragbarkeit eines UI Patterns auf andere, konkrete Aufgaben.

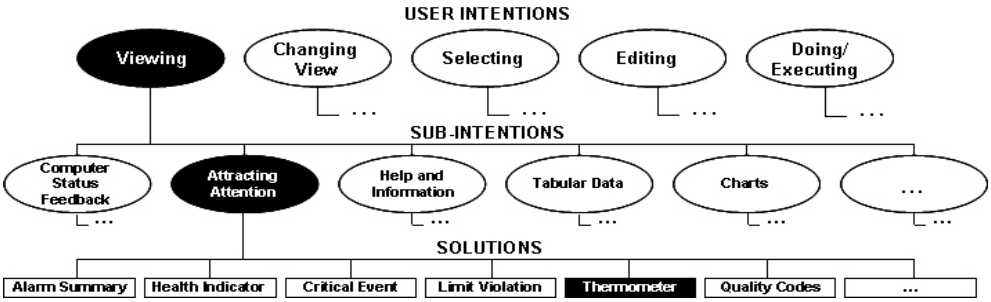


*Abbildung 2: Struktur der UI Pattern Bibliothek*

Das Beschreibungsformat basiert auf dem Format von Christopher Alexander (1979). Für den Einsatz bei Siemens Energy wurde das Format um die folgenden Punkte erweitert: Ansprechpartner („Owner“), Hyperlinks auf weiterführende Dokumente wie Style Guides oder Design Spezifikationen („Further Information“), die Anzeige, in welchen Applikationen das jeweilige UI Pattern bereits eingesetzt wird („Used in“), Beispiele aus verschiedenen Applikationen („Examples“), Usability Evaluationsergebnisse („Evaluation“), sowie eine Beurteilung darüber, wie etabliert das UI Pattern in der Organisation bereits ist („Maturity level“) (Abb.3).

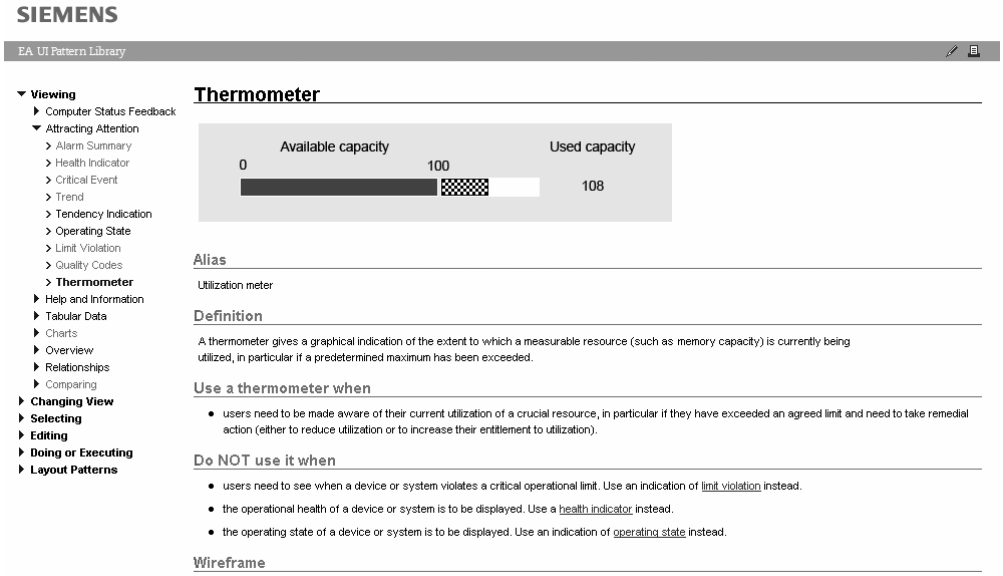


*Abbildung 3: User Interface Pattern Bibliothek, Beispiel Thermometer*

## 3.2 Integration in den Produkt-Entwicklungsprozess

Für eine erfolgreiche Umsetzung des UI Pattern Ansatzes ist es wichtig, dass die UI Pattern Aktivitäten unmittelbaren Einfluss auf den Produkt-Entwicklungsprozess und die darin stattfindenden Usability Aktivitäten haben. Wie in Abbildung 4 zu sehen ist, unterstützen die Usability Aktivitäten die Produktentwicklung bei der Energieautomatisierung in den verschiedenen Phasen.

Die Methodik orientiert sich am Usage-Centered Design Process von Constantine & Lockwood (1999). In der Phase der Anforderungsdefinition werden die für die Software relevanten Nutzergruppen und deren Hauptaufgaben identifiziert. Für jede Aufgabe wird beschrieben, was das System zur Verfügung stellen muss (System Responsibility) und welchen Zweck der Nutzer damit verfolgt (User Intentions). Die Beschreibung ist der Aufgaben ist abstrakt und lösungsfrei. Hierdurch wird die Konzentration auf die wesentlichen Bedürfnisse der Endanwender ermöglicht, die sich aus der Erfüllung der Aufgabe ergeben. Das Wissen über die Nutzergruppen und deren Aufgaben bildet die Grundlage für alle Diskussionen über den Funktionsumfang der Software. In der Phase der Anforderungsanalyse wird von den Hauptaufgaben systematisch ein Benutzungsmodell abgeleitet. Die Task Map bildet die zukünftigen Navigationskontexte für die Benutzer ab. Die Navigation Map visualisiert die Navigationswege zwischen den Navigationskontexten. Für jeden Navigationskontext wird betrachtet, welche Daten und Aktionen für die darin enthaltenen Aufgaben benötigt werden und wie diese zusammenhängen. Das Ergebnis ist ein Abstract Prototype, der die grundlegende Struktur des späteren User Interfaces zeigt. Von den Abstract Prototypes wird das

zukünftige User Interface abgeleitet und auf Papier skizziert (Visual Prototype). In der Phase der Entwicklung werden diese Skizzen weiter konkretisiert und durch Style Guides angereichert. Es entsteht ein Low-Level Prototype, der die Bedienoberfläche und Interaktion der zukünftigen Software visualisiert.

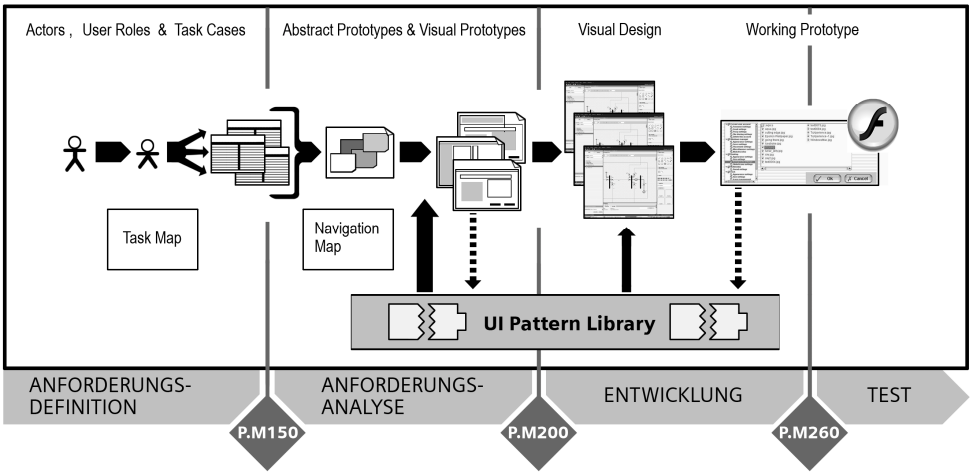


*Abbildung 4: Integration in den Produkt-Entwicklungsprozess*

Diese systematische Methodik bietet drei unmittelbare Anknüpfungspunkte für UI Pattern:

1. Zum einen wird der Schritt vom Abstract Prototype zum Visual Prototype (Wireframe) unterstützt, in dem eine erste Skizze des User Interfaces entsteht. Für diesen Schritt ist viel Erfahrung und Wissen über UI Konzepte notwendig sowie Kreativität gefragt. Die zentrale UI Pattern Bibliothek unterstützt diesen Schritt, indem sie Aufschluss über bereits vorhandene Konzepte gibt. Durch die abstrakte und lösungsfreie Beschreibung der Aufgaben in User Intentions (Aktionen) und System Responsibilities (Daten) kann von dem konkreten Anwendungskontext der Software (z.B. Fehleranalyse) abstrahiert werden auf allgemeingültige User Intentions (z.B. Information auswählen, Details anzeigen). Da sich die Struktur der UI Pattern Bibliothek ebenfalls an User Intentions orientiert, lassen sich die so identifizierten User Intentions aus der konkreten Aufgabe einfach mit vorhandenen UI Pattern abgleichen. Die Nutzer der UI Pattern Bibliothek erhalten einen direkten Zugang zu möglichen UI Lösungen. Anstatt mühsamer Recherchen durch Dokumentationen von anderen Projekten sind die relevanten Lösungen sofort zugänglich. Die Wireframes aus der UI Pattern Beschreibung lassen sich 1:1 auf die Papierskizzen der Visual Prototypes übertragen. Hierdurch ergibt sich ein großes Potential an Effizienzgewinn in der Konzeptionsphase und Konsistenz für die späteren UIs.

2. Ein weiterer Ansatzpunkt ergibt sich in der Umsetzung der Navigation Map. Die Hauptnavigationswege zwischen den einzelnen Screens (Clusters in der Task Map) müssen auf

dem User Interface adäquat umgesetzt werden. Auch hierfür bietet die UI Pattern Bibliothek Unterstützung an. Unter der Intention „Changing View“ sind beispielsweise alle verfügbaren UI Pattern zum Thema Navigation strukturiert aufbereitet. Auch hier entscheiden die „Use when“ Kriterien der einzelnen UI Pattern darüber, welches UI Pattern für den konkreten Anwendungsfall am besten geeignet ist.

3. Ein dritter Anknüpfungspunkt für die UI Pattern Bibliothek ergibt sich bei der Detaillierung der Bedienkonzeptskizzen. Wenn das Konzept schrittweise mit Details angereichert wird, muss entschieden werden, wie (d.h. mit welchen UI Pattern) die jeweiligen Details auf dem User Interface dargestellt werden sollen. Beispielsweise stellt sich im Zuge der Detaillierung oft heraus, dass Informationen nicht mehr übersichtlich auf einem Screen dargestellt werden können. Die UI Pattern Bibliothek bietet hier den passenden Überblick über verfügbare Lösungen zur Strukturierung der Informationen (z. B. durch Tab Cards oder Filter). Zusätzlich werden die UI Pattern von den Entwicklern als Referenz in der Umsetzung genutzt, da auf alle relevanten Informationen zu einer Lösung (z.B. auf vorhandene Style Guides) verlinkt wird.

Neben der Verwendung von Pattern aus der UI Pattern Bibliothek ist es wichtig, dass auch der Rückfluss in die UI Pattern Bibliothek stattfindet, damit neue Lösungen für zukünftige Projekte zur Verfügung gestellt werden. Die UI Pattern Bibliothek wächst dadurch kontinuierlich. Fertige Bedienkonzepte einer Software (in Form von Visual Prototypes und klickbaren Prototypen) werden hinsichtlich neuer UI Pattern von Usability Experten analysiert. Kriterium für die Aufnahme in die UI Pattern Bibliothek ist vor allem die Relevanz für andere Applikationen. Je mehr Applikationen von einem neuen UI Pattern profitieren können, desto schneller wird es verfügbar gemacht.

# 4 Effizienz, Konsistenz & Kosten

Durch den Einsatz der UI Pattern Bibliothek ergibt sich ein hoher Effizienzgewinn bei Siemens Energy. Vorhandenes Wissen wird strukturiert aufbereitet und für die Organisation unmittelbar nutzbar gemacht. Durch den Austausch von Konzepten zwischen Projekten wird ein hohes Synergiepotential erschlossen. Lösungen werden unkompliziert zwischen Projekten verschiedener organisatorischer Einheiten und verschiedenen Standorten ausgetauscht. Diskussionen können abgekürzt werden, da auf etablierte Konzepte verwiesen werden kann. Hierdurch verringert sich der Aufwand und es werden die Kosten für die Definition von Bedienkonzepten gesenkt.

Mit dem systematischen Austausch von UI Pattern wird zudem eine hohe Konsistenz in den Bedienoberflächen erreicht. Weltweite Entwicklungsstandorte sowie unterschiedliches Knowhow bzgl. User Interface Gestaltung und der Anwendungsdomäne Energieautomatisierung erschweren es, ein einheitliches Look & Feel für alle Applikationen umzusetzen. Statische Style Guides reichen nicht aus, um die vorhandenen Lösungen zu kommunizieren, da sie im Gegensatz zur beschriebenen UI Pattern Strukturierung keinen problemorientierten Zugang zu UI Lösungen bieten. Die strukturierte Aufbereitung der UI Pattern durch User

Intentions knüpft dagegen direkt an die Usability Ergebnisse (Nutzergruppen und deren Aufgaben) an und schafft eine Verbindlichkeit über den Einsatz existierender Lösungen. Eine hohe Konsistenz zwischen den Produkten hat sowohl für den späteren Anwender als auch für die Firma einen Mehrwert. Der Anwender kann sich schneller in einer für ihn neuen Software zurechtfinden. Damit können beispielsweise Fehleingaben verhindert werden. Die Firma Siemens erreicht eine hohe Wiedererkennung und Markenbindung beim Kunden.

Seit ihrer Einführung im Jahr 2007 hat sich die UI Pattern Bibliothek als hilfreiches Instrument bei der Produktentwicklung erwiesen. Nichtsdestoweniger gilt es verschiedene Herausforderungen zu bewältigen, um die UI Pattern in der Organisation noch effizienter nutzbar zu machen. Die kontinuierliche Beschreibung und projektübergreifende Reviews der UI Pattern sind notwendig, um ein gemeinsames Verständnis und Akzeptanz zu erzielen. Dadurch entstehen einmalig sehr hohe Kosten, weil viele interne Stakeholder in den Prozess eingebunden werden müssen. Um Kosten zu minimieren, muss dieser Prozess noch effizienter gestaltet werden.

Eine weitere Schwierigkeit liegt darin, die Verbindlichkeit in der Umsetzung der UI Pattern zu gewährleisten. Durch die Heterogenität und große Anzahl der Produkte ist es nicht möglich, jedes User Interface durch Usability Experten überprüfen zu lassen. Deshalb muss durch regelmäßige Schulungsangebote eine hohe Verbindlichkeit in der Umsetzung der UI Pattern erreicht werden. Zusätzlich müssen möglichst viele UI Pattern durch Reuse Komponenten ergänzt werden, um die Verwendung in der Entwicklung zu vereinfachen.

Zusammenfassend lässt sich festhalten, dass die Verwendung von UI Pattern und deren systematische Aufbereitung in einer zentralen UI Pattern Bibliothek für die Organisation sehr lohnenswert ist. Es ergibt sich ein Effizienzgewinn durch die Identifizierung von Synergien zwischen Produkten. Gleichzeitig wird durch UI Pattern ein Konsens über die Gestaltung von Bedienkonzepten geschaffen, welche zu einer höheren Konsistenz in den Produkten führt. Interne Stakeholders profitieren durch die UI Pattern Bibliothek von anderen Projekten und haben die Möglichkeit, wiederum ihre Lösungen einzubringen. Dieses gemeinsame Engagement bildet die Grundlage für die Umsetzung einer einheitlichen Bedienphilosophie in allen Produkten.

## Literaturverzeichnis

Alexander, C. (1979). *The Timeless Way of Building.* New York, NY, USA: Oxford University Press.

Arend, U. & Peters, J. C. (2008). User Interface Konstrukte und Key Innovations in SAP Business ByDesign. In *Mensch und Computer 2008*. München: Oldenbourg, S. 111-116.

Auer, S. & Richter, A. (2005). The User Intention Oriented Approach to User Interface Patterns. In Workshop on User Interface Patterns. *Interact 2005*.

Beck, K. & Cunningham, W. (1987): Using Pattern Languages for Object-Oriented Programs. In Workshop on Specification and Design for Object-Oriented Programming. Technical Report CR-87-43, Tektronix, Inc. *OOPSLA'87*.

Borchers, J. (2001). *A Pattern Approach to Interaction Design*. Chichester, UK: John Wiley & Sons.

Buschmann, F., et al. (1996). *Pattern-Oriented Software Architecture Volume 1: A System of Patterns.* UK: John Wiley & Sons.

Constantine, L. & Lockwood, L. (1999). *Software for Use: A Practical Guide to the Models and Methods of Usage-Centered Design*. New York: ACM Press.

Deng, J. et al. (2005). Managing UI Pattern Collections. In *ACM International Conference Proceeding Series, Vol. 94*, New York, USA: ACM, S. 31-38.

Gamma, E. et al. (1995). *Design Patterns: Elements of Reusable Object-Oriented Software.* Amsterdam: Addison-Wesley.

Greene, S.L., et al. (2003). Tool-based Decision Support for Pattern Assisted Development. In *CHI 2003 workshop on HCI Patterns: Concept and Tools.* Fort Lauderdale,USA.

Infragistics (2009). *Infragistics Quince UX Patterns Explorer*. http://quince.infragistics.com

Kunert T. et al. (2006). Identifizierung von Interaction Design Patterns für neue Technologien. In Heinecke, A. M., Paul, H. (Hrsg.): *Mensch und Computer 2006: Mensch und Computer im Strukturwandel*. München: Oldenbourg, S.55-64.

Malone, E., et al. (2005). *Implementing a Pattern Library in the Real World: A Yahoo! Case Study*. http://www.boxesandarrows.com/view/implementing_a_pattern_library_in_the_real_world_a_yahoo_case_study

Norman, D.A. & Drapher, S.W. (1986) *User-Centered System Design: New Perspectives on Human-Computer Interaction*. Hillsdale, New Jersey: Lawrence Erlbaum Associates.

Tidwell, J. (1999). *Common Ground: A Pattern Language for Human-Computer Interface Design*. http://www.mit.edu/~jtidwell/interaction_patterns.html.

Tidwell, J. (2006). *Designing Interfaces*. Sebastopol, CA, USA: O'Reilly.

Van Welie, M. (2008). *Patterns in Interaction Design*. http://www.welie.com/patterns/

## Kontaktinformationen

Sonja Auer
Siemens AG, Energy Sector
Sonja.Auer@siemens.com

Anke Richter
Siemens AG, Corporate Technology
A.Richter@siemens.com