

# Alte Instrumente, neues Orchester: SOA und ihre Anforderungen an das User-centered Design

**Henning Brau**  
German UPA  
Postfach 80 06 46  
70506 Stuttgart  
hb@germanupa.de

**Julia Maly**  
User Interface Design GmbH  
Martin-Luther-Straße 57-59  
71636 Ludwigsburg  
julia.maly@uid.com

**Bernd Watzal**  
German UPA  
Postfach 80 06 46  
70506 Stuttgart

## Abstract

Die Serviceorientierte Architektur (SOA) stellt einen Paradigmenwechseln in der System-Entwicklung dar. Sie stellt nicht die Entwicklung eines geschlossenen Systems, sondern die prozessbezogene Unterstützung der gesamten Organisation durch IT-Systeme in den Mittelpunkt. Das „Orchester“ aus bestehenden Systemen wird dabei nicht um weitere

Instrumente (Systeme) erweitert, sondern durch eine geschickte „Umbesetzung“ verschlankt und rationalisiert. Der Usability Professional wird bei diesem Vorgehen zum Knoten in einem umfassenden Netz aus Stakeholder- und IT-Interessen. Zugleich bleibt er Anwalt der Anforderungen der Nutzer.

Der Beitrag stellt den SOA-Ansatz vor und zeigt auf, wie sich die Rolle des Usability Professionals durch ihn verändert. Chancen und Risiken des Vorgehens werden aufgezeigt.

## Keywords

SOA, Usability Professional, Dienst, Geschäftsprozess, Eclipse, Framework

## 1.0 Einleitung

Die frühen IT-Systeme wurden in Organisationen zunächst meist als Mittel zur Rationalisierung auf der Handlungsebene eingesetzt. Dabei lösten sie lediglich bestehende Werkzeuge ohne übergreifenden Bezug zum Geschäftsprozess einer Organisation ab, z.B. die Tabellenkalkulation als Ersatz für Taschenrechner und Matrizen.

Mit der zunehmenden Funktionsvielfalt und Workflow-Orientierung ab den 90er Jahren verlangte die Neueinführung von IT-Systemen aber bald eine Anpassung der bestehenden Geschäftsprozesse (z.B. Design, Entwicklung, Produktion, Vertrieb) an die vorgegebenen Dienste des jeweiligen Systems. „Dienst“ bezeichnet hierbei eine Bündelung von Funktionen in einem in sich geschlossenen Workflow, z.B. die Verwaltung von Kundendaten. Die Dienste eines Systems sind inklusive Rollen und Schnittstellen für bestimmte „Use Cases“ vordefiniert. Soll ein Dienst innerhalb eines Use Cases ausgeführt werden, gibt es für den Nutzer meist Freiheitsgrade lediglich dahingehend, wie

etwas ausgeführt wird (zum Beispiel Drag'n'Drop, Tastenkombinationen) nicht ob und nur bedingt wann. Auch die Weiterverwendung von Daten und Diensten in den später folgenden Prozessabschnitten ist nur über Schnittstellen vorgesehen: Anderer Prozessabschnitt, andere Systeme.

In Organisationen gibt es heute innerhalb verschiedener Prozessabschnitte eine Vielfalt unabhängiger Systeme mit festen Use Cases, Workflows und Diensten. Für den Anwender ist die Bearbeitung der Aufgaben in „seinem“ Prozess meist mit einem „System-Hopping“ verbunden. Er muss den Abschnitt und die Systeme mit ihren Diensten also kennen und wissen, wann er welches System mit welchem Dienst anzuwenden hat. Es ist daher weiterhin der Nutzer, der sich an das System anpassen muss.

## 2.0 Der SOA-Ansatz

Die Serviceorientierte Architektur (SOA) sieht hingegen vor, benötigte Dienste an den Geschäftsprozessen zu orientieren. Dazu werden zunächst

die bestehenden Prozesse und IT-Systeme analysiert. Danach werden die Prozesse rationell restrukturiert und beschrieben (Business Process Re-engineering; BPR). Der entstehenden Prozesslandkarte, welche die einzelnen Geschäftsprozesse und ihre Abhängigkeiten über die ganze Organisation hinweg darstellt, werden nun die jeweils benötigten Dienste zugeordnet: In welchem Geschäftsabschnitt wird welcher der zur Verfügung stehenden Dienste wann benötigt, unabhängig von den zugrunde liegenden Systemen? Welche Komponenten (Datenbanken, Server und User Interfaces etc.) aus verschiedenen Systemen können zusammen neue Dienste generieren?

Eine hohe Wiederverwendbarkeit von Komponenten verschiedener Systeme ist dabei maßgebliches Ziel: Sie werden insgesamt so in Dienste gefasst und koordiniert („orchestriert“), dass ihre Leistungen für andere Rollen in weiteren Prozessabschnitten weiter genutzt werden können. Einfacher ausgedrückt erhält der Nutzer im Idealfall in seiner jeweiligen Rolle zur rechten Zeit am rechten Ort immer nur die Funktionen zur

Verfügung gestellt, die er aktuell benötigt; im Idealfall, ohne dass dafür überhaupt ein neues System entwickelt werden müsste.

Wir beziehen uns in diesem Artikel auf eine vollständige, also eine „One-Client“ SOA. Bei ihr wirken alle Komponenten der Systeme, welche für einen Dienst aktuell benötigt werden, in einen universellen Client als grafisches User-Interface hinein, wie es das Eclipse RCP Framework ermöglicht.

Eclipse ist ein Open-Source-Framework zur Entwicklung von Software nahezu aller Art, das sich u.a. zur Umsetzung von Rich Clients zur Bündelung von SOA-Diensten durch so genannte Sichten, Editoren und Perspektiven eignet. Eclipse RCP steht dabei für die „Eclipse Rich Client Platform“. Diese Plattform setzt sich zunehmend auch als Entwicklungsumgebung für SOA-Vorhaben durch.

Das User-Interface ist bei einer solchen One-Client SOA also nicht in festen Masken organisiert, sondern ändert sich je nach Rolle und Prozessabschnitt, ohne dass der Nutzer das System wechseln muss. Dabei ist es unmaßgeblich, ob es sich bei dem User-Interface um einen webbasierten „Thin Client“ oder um einen lauffähigen „Rich Client“ handelt. SOA ist also keine Webtechnologie und sieht auch nicht zwingend eine echte Neuentwicklung von Systemen vor.

### 3.0 Rolle der Usability Professionals

Laut Definition des „Arbeitskreises Berufsfeld“ der German UPA (GIJPA AK Berufsfeld, 2009) ist ein Usability Professional (UP; der Begriff wird stets geschlechterneutral verwendet) eine Person, die qualifiziert und methodisch Anforderungen an die Usability interaktiver Systeme herleitet, umsetzt oder deren Umsetzung überprüft. Dabei ist sie spezialisiert auf die Analyse, Gestaltung,

Prüfung und Bewertung von interaktiven Systemen sowie auf die Durchführung von Nutzertests. Nach ISO/DIS 9241-210 (2008) ist die Mensch-zentrierte Gestaltung ein Ansatz, welcher zum Ziel hat, ein nutzbares System zu erschaffen, indem Nutzungskontext, Ergonomie und Usability-Wissen beachtet werden. Der UP erhebt Nutzer- und Nutzungsanforderungen und vermittelt diese u.a. in Form von Anforderungslisten, Personas und Wireframes an die technischen Entwickler.

Doch der SOA-Ansatz erfordert eine Abwendung vom alleinigen Blick auf die Anwender: Die Organisationsentwicklung und das Zusammenfassen von verschiedenen Systemen stellen den UP vor veränderte und erweiterte Herausforderungen.

### 4.0 Herausforderungen durch das Business Process Re-Engineering (BPR)

Jede Organisation verfolgt Ziele, die innerhalb eines Kernprozesses (z.B. Produktentstehungsprozess) erzielt werden. Dabei wird ein Input (z.B. Rohstoffe) durch vordefinierte Transformationsschritte (z.B. maschinelle Bearbeitung) in einen Output (Produkt) umgewandelt. Um diesen Kernprozess herum scharen sich Supportprozesse, die ihn mit Dienstleistungen unterstützen, beispielsweise Einkauf, Marketing, Human Resources, IT etc.

Bei herkömmlichen IT-Entwicklungsprojekten wird ein gegebener Ist-Zustand einer dieser Prozesse durch ein einzelnes Neusystem rationalisiert, ohne dass davon andere Prozesse betroffen sind bzw. sein müssten. Insofern der UP an dieser Stelle auch Aufgaben der Anforderungsanalyse wahrnimmt, betrachtet er daher entsprechend der fachlichen Aufgabe

lediglich die betroffenen Prozessabschnitte und ihre Nutzungsrollen. Häufig wird er während der Definition des Rationalisierungszieles (Soll-Zustand) lediglich konsultierend hinzugezogen. Ein ganzheitlicher Überblick über die Gesamtprozesse ist für ihn daher von eher geringem Interesse.

Das BPR wird im Allgemeinen als rein betriebswirtschaftliche Aufgabe betrachtet. Meist ausgehend von einer abstrakten Prozessvisualisierung sowie –dokumentation werden Aufgaben und Rollen verändert, ohne das Kenntnisse über die Nutzungsrealität zwingend gegeben sein müssten. Für begrenzte Prozessabschnitte ist dies auch häufig unkritisch. Hier überwiegt selbst beim mensch-zentrierten Gestaltungsansatz die Betrachtung der Interaktion des Nutzers mit dem User-Interface - ohne tiefe Beachtung des Prozessbezugs (z.B. Sarodnick & Brau, 2006).

Werden aber Systemkomponenten aus verschiedenen Prozessabschnitten in höhere Dienste zusammengefasst, kann sich der UP nur noch schwer von seiner Verantwortung für die Prozessgestaltung frei machen. Denn gerade im BPR können bereits grundlegende Fehlentscheidungen getroffen werden, die das spätere Gesamtsystem in seiner Gebrauchstauglichkeit beeinträchtigen könnten. Daher muss der UP über genügend Wissen zu Prozessen und IT-Landschaft verfügen, um kritische Punkte erkennen und ihre Auswirkungen einschätzen zu können.

Allerdings kann eine intensive Auseinandersetzung mit den Prozessen auch eine Entfremdung von der Mensch-zentrierten Perspektive bewirken, wenn dem „höheren Ziel“ der Gesamtorganisation mehr Gewicht beigemessen wird, als dem jeweiligen Nutzungskontext.

## 5.0 Herausforderungen durch das gemeinsame SOA Framework

Bei Projekten zur Umsetzung von One-Client SOAs sind meist sehr unterschiedliche Systeme beteiligt, die erst in einem Client vereinigt werden müssen.

Nehmen wir an, es handelt sich um eine eingesehene Firma mit mehr als 30 Jahren Unternehmensgeschichte. Hier ist seit Mitte der 80er Jahre eine Vielzahl von Systemen bis zum heutigen Status Quo eingesetzt worden: zum einen Teil Eigen- beziehungsweise in Auftrag gegebene Neuentwicklungen, zum anderen Teil angepasste Kauflösungen. Jedes System wurde oder wird von einem Team von „Systemeignern“ gepflegt und weiter entwickelt. Jedes hat seine eigenen Benutzungsphilosophien und Design-Styleguides. Jedes hat ein festes Budget für seinen Erhalt und Weiterentwicklung.

Die Unabhängigkeit der Systeme und ihrer Eigner wird durch den SOA-Ansatz zumindest eingeschränkt, wenn nicht ganz aufgehoben. Im Mittelpunkt steht nicht mehr die Weiterentwicklung eines weitgehend eigenständigen Einzelsystems, sondern eine Anpassung der gesamten IT-Landschaft an neu definierte Geschäftsprozesse.

Monolithische Altsysteme müssen dazu in ihre einzelnen Komponenten aufgebrochen werden. Diese Komponenten vermischen sich dann mit denen anderer Systeme zu neuen Diensten mit weitgehend dynamischen GUIs im Zielsystem. Damit eine Einheitlichkeit bei der Entwicklung dieses Zielsystems entstehen kann, muss im ersten Schritt ein gemeinsames Programmiergerüst („Framework“) definiert werden. Dieses Framework, das in der Regel später die gesamte Anwendungsarchitektur determiniert, ist danach die Blaupause der Um- und Fortentwicklung der Bestandsysteme.

Doch nicht nur die Programmierung, sondern auch die Gestaltung und das Verhalten des Zielsystems müssen als Teil des SOA-Frameworks vordefiniert werden. Eine Aufgabe, die häufig bei den UPs verortet wird.

Gemeinsame Styleguides und Bedienkonzepte verlangen aber in jedem Fall eine Anpassung eines bestehenden Systems. Je weiter das Framework eines Bestandssystems dabei von dem neuen SOA-Framework abweicht, desto aufwändiger und damit kostenintensiver wird die Umstellung für den betreibenden Bereich.

## 5.1 Geänderte Rolle für Usability Professionals

Auch hieraus ergeben sich neue Herausforderungen für den UP. Wenn schon die Zusammenarbeit mit den technischen Entwicklern eines einzelnen Systems von vielen UPs als aufreibend empfunden wird, so multipliziert sich der hier notwendige Abstimmungsaufwand mit jedem hinzukommenden System, das in dem Zielsystem untergebracht werden soll.

Die Befindlichkeiten der Systemeigner auf organisationspolitischer Ebene sind dabei eine hohe Hürde: Neben dem materiellen Aufwand für die Umsetzung von Änderungen, der je System möglichst gering gehalten werden soll, steht auch die Frage, welches Altsystem eine „Leitwolf-Funktion“ hat. Das System, das sich am wenigsten ändert, ist nämlich in Zukunft das mächtigste, weil es den anderen Systemeignern indirekt vorschreibt, was diese zu tun haben. Zugleich steckt dahinter die Aussage, dass es von Anfang an das zukunftsfähigere, bessere System gewesen ist. Hier können sich durchaus Karrierechancen nach oben oder unten verschieben.

Neben diesen eher individuellen Überlegungen liegt die Motivation zum Ab-

blocken von Änderungen natürlich in der damit einhergehenden hohen Belastung der Systemeigner. Je größer der parallel zu betreibende Aufwand der Anpassung an das SOA-Framework, desto weniger des Kernbudgets kann in die Aufrechterhaltung und Erweiterung investiert werden.

In der Folge kann der UP zum Teil von Umschmeichelung bis Anfeindung alle Facetten der nicht-fachlichen Aufmerksamkeit erleben. Ihm kommt in diesem Prozess die Aufgabe zu, an den mikropolitischen Interessen vorbei, eine für den Nutzer möglichst positive Lösung in Styleguide und Bedienkonzept zu entwickeln. Er muss daher die Systemeigner „abholen“ und zugleich die Interessen der Nutzer gegenüber dem Projekt vertreten. Durch die Präsenz einer Vielzahl von Systemeignern besteht dabei potenziell ein Übergewicht der Interessen der Software-Entwicklung gegenüber den organisations- und nutzungsbezogenen Themen.

Eigentlich also eine Umkehrung der originären Ziele des SOA-Ansatzes: Dem Geschäftsprozess und damit dem Nutzer den Vorrang vor system-zentrierten Fragestellungen zu geben.

Eine der größten Gefahren für den UP ist dabei, sich zu sehr in die rein entwicklungsbezogenen Themen hineinziehen zu lassen. Der UP sollte zwar bereit sein, eine Moderatorenrolle zwischen den Positionen der Systemeigner einzunehmen; er muss aber zugleich die Partizipation der Nutzer vorantreiben. Eines der am häufigsten gehörten Argumente, gerade in dieser Phase auf eine Nutzereinbindung verzichten zu wollen ist, dass zuerst die Definition des gemeinsamen Frameworks abgeschlossen sein muss, das den Rahmen der Entwicklung vorgibt. Doch gerade der SOA-Ansatz mit seinem vorrangigen Ziel, ganze Organisationen umzustrukturieren, verlangt eine frühe und kontinuierliche Auseinan-

dersetzung mit den Geschäftsprozessen und ihren Abhängigkeiten zueinander.

## 6.0 Fallbeispiel

Bei einem Industrieunternehmen sollten durch eine SOA-Initiative die zentralen Clients der Produktentwicklung in ein gemeinsames User-Interface integriert werden. Die UPs machten sich an die Arbeit und stellten einen sportlichen aber nach eigener Erfahrung umsetzbaren Zeitplan für die Erstellung von Styleguides und Bedienkonzepten auf; Nutzer wurden dabei eingebunden und Aufgaben analysiert.

Die erste große Überraschung war die Entscheidung der Projektleitung, auf ein tiefgehendes BPR zu verzichten. Zuerst wollte man die Machbarkeit einer Integration der wichtigsten Systeme in der Zielentwicklungsumgebung (Eclipse RCP) nachweisen.

Noch vor dem Abschluss der Machbarkeitsstudie wurde aber festgelegt, dass die Systeme auf jeden Fall in Eclipse integriert werden sollten. Die hierfür gegebenen Begründungen bezogen sich dabei stets darauf, dass es sich um ein für die Entwicklung ideales Framework handelt. Ob man damit auch komplexe Geschäftsprozesse geeignet unterstützen könne, wurde hingegen nicht evaluiert. Die IT-Entwicklung übernahm also die Leitung des Projektes auf allen Ebenen.

Nach Vorlage der ersten Styleguides und Bedienkonzepte wurde für viele Systemeigner erst der Umfang der Anpassungen ihres eigenen Systems deutlich. Die dadurch zu erwartenden Belastungen der zur Verfügung stehenden Ressourcen wurden nicht als Chance, sondern als Hindernis der Weiterentwicklung des eigenen Systems verstanden. Die Erwartung, das eigene System habe eine Sonderstellung, zog sich

durch viele Diskussionen und Abstimmungsrunden erwiesen sich als zäh.

Dies gipfelte in einer Sitzung, in der Diskussionen über die Richtigkeit der Aussage stattfanden, eine Checkbox dürfe lediglich zwei Zustände haben. Obwohl eine Checkbox die grafische Versinnbildlichung eines booleanschen Operanden ist, der per Definition nur digital sein kann, wurde hier von den Systemeignern keine Lösung verabschiedet und das Thema blieb ein Bestandteil von mehreren ausgedehnten Sitzungen.

Darüber hinaus wurde kurzfristig entschieden, dass mehrere geplante Neusysteme auf Basis des Eclipse-Frameworks entstehen sollten. Die Integration auf technischer Ebene wurde dementsprechend mit hohem Aufwand vorangetrieben. Das Requirements Engineering, das sich eigentlich auf die Anforderung der Prozessinhaber richten sollte, wurde durch Neubesetzung aus den Reihen der technischen IT endgültig zu einer rein Technologie-orientierten Sichtweise gezwungen. Das eigentlich notwendige BPR wurde ab da vollends vernachlässigt. Zudem wurden die UPs immer mehr mit Prüfaufträgen der technischen Realisierbarkeit von Anforderungen aus den Styleguides und Bedienkonzepten gedrängt. Damit waren die UPs quasi in Vollzeit mit der originär technischen Aufgabe der Machbarkeits-Überprüfung blockiert. Der SOA-Ansatz war ab dieser Zeit inhaltlich kaum noch zu erkennen.

Es drängt sich die Frage auf, ob die UPs schon früher in dem Prozess Möglichkeiten gehabt hätten, dieser Entwicklung entgegen zu wirken.

## 7.0 Empfehlungen

Die gängigen Beschreibungen der Normung geben nur zum Teil wieder,

welche außer den üblichen Aktivitäten im Mensch-zentrierten Gestaltungsprozess für einen UP in einer SOA-Initiative bestehen. Nachfolgend stellen wir einige Empfehlungen dar, die als Best Practises Hilfestellung für Kollegen in ähnlichen Vorhaben bieten sollen.

### 7.1 Sichtbarkeit für UX erhöhen

Es ist von zentraler Bedeutung, dass ein eigenständiges Arbeitspaket „User Experience“ mit eindeutigen Aufgaben, Kompetenzen und natürlich Verantwortungen definiert wird. Dies ist dann in der Projektstruktur sichtbar und bleibt nicht in einem technischen Teilprojekt verborgen. Die Projektleitung sollte für das Thema massiv sensibilisiert werden. Hierbei kann auch die Teilnahme an einer Fachkonferenz wie der „Usability Professionals“ der German UPA ein Augenöffner sein. Insgesamt sollte der UP fortlaufend Angebote zum Transfer von Usability-Wissen in das Projekt hinein machen.

### 7.2 Beteiligung am BPR

Eine Beteiligung am BPR ist eine entscheidende Chance dafür, Nutzerinteressen aus dem Kontext ihrer Anwendung heraus zu berücksichtigen. Die Expertise der UPs kann darüber hinaus die meist abstrakte betriebswirtschaftliche Vorgehensweise um Wissen um reale Prozessgestaltung ergänzen.

Dies aber verlangt eine Kenntnis über die Methoden und das Vorgehen des BPR, um in diesem Prozess analytische wie gestalterische Verantwortung übernehmen zu können, ohne die eine Beteiligung am BPR nur beschränkt sinnvoll bleiben wird. So lange diese Kenntnis nicht vorhanden ist, sind Qualifikationsmaßnahmen bezüglich der Methoden des BPR unumgänglich.

### 7.3 Developer Research

Die initiale Phase eines SOA-Projektes sollte für den UP auch darin bestehen, die Systemeigner und ihre Produkte zu analysieren. Das Sammeln und Studieren aller vorhandenen Dokumentationen zu Styleguides und Bedienkonzepten der bestehenden Systeme kann mitunter eine aufwändige Tätigkeit sein, doch können hier die größten Herausforderungen für die Systeme erkannt und schon früh konsensfähige Vorlagen erarbeitet werden, die dennoch die Interessen der Nutzer aufgreifen. Auch politische Befindlichkeiten werden in direktem Austausch erkennbar und können berücksichtigt werden.

Es kann dabei mitunter sehr hilfreich sein, im Nachhinein darzustellen, welche Definition von welchem bestehenden Styleguide oder Bedienkonzept der Altsysteme übernommen wurde. So finden sich die Systemeigner in der neuen Dokumentation wieder, was hilft, Akzeptanz für diese zu schaffen.

### 7.4 User Experience Gremium

Für die Systemeigner ist es nicht selten durchaus ungewohnt, sich mit den Vorgehensweisen und dem Eigenverständnis der anderen Systemeigner zu beschäftigen. Am Anfang sollte daher Sensibilität hierfür hergestellt werden. Es ist dabei hilfreich, Verantwortliche der Systeme für einen eintägigen Workshop an einen Tisch zu bringen. In diesem Workshop sollten noch einmal die Zielstellungen des beginnenden SOA-Vorhabens thematisiert werden. Dann sollte es eine Einführung in jedes System (Hintergrund, Geschichte, Aufgaben, Prozesskunden) durch die Systemeigner geben. Teambuilding-Maßnahmen (Arbeitsgruppen sowie abschließender sozialer Event) können

Barrieren überwinden helfen. Im Zentrum aller dieser Aktivitäten sollte das Erkennen eines gemeinsamen Zieles stehen. Dieser erste Workshop sollte dazu führen, dass ein gemeinsames Gremium zur Verbesserung der User Experience mit wöchentlichen Abstimmungsterminen eingerichtet wird.

### 7.5 Geduld am Anfang

Auch wenn es mitunter aufreibend sein kann, sich mit Details des Styleguides auseinanderzusetzen, wird es doch eine gute Idee sein, mit kleinen und vermeintlich „leichten“ Themen anzufangen. Das User Experience Gremium muss sich erst einmal finden und Ressentiments müssen teilweise in Konflikten und deren Lösung abgebaut werden. Je komplexer die anfänglichen Themen, desto schwieriger die Konfliktlösung. Es gilt also wie so oft: Keep it simple.

### 7.6 Ablehnung der Bringschuld

Der UP sollte sich auf keinen Fall in die Situation bringen lassen, dass er die technische Machbarkeit von Anforderungen aus User Experience Perspektive nachweisen muss. Die technische Realisierung bleibt stets Aufgabe der Entwicklung. Allerdings wird diese zuweilen aus politischen Interessen anführen, dass bestimmte Anforderungen im Framework nicht umgesetzt werden könnten. Es ist daher vorbeugend sinnvoll, jede Gestaltungsanforderung in einem Demonstrator abzusichern. Schon die Darstellung aller gewünschten GUI-Elemente in einem lauffähigen Mikrosystem, das bis auf die Anzeige keine weiteren Funktionen hat, kann hier Abwiegungen aufgrund angeblich mangelnder Umsetzbarkeit vorbeugen.

### 7.7 Eskalationslinien ziehen

Auch wenn die Systemeigner als solche zunächst geneigt sein mögen, alleine die Interessen „ihres“ Systems zu betrachten: Ihre Vorgesetzten sind es zumeist nicht. Sie haben den Auftrag zur SOA-Initiative erhalten und sind daher daran interessiert, dass es dann nicht ausgerechnet an dem System scheitert, das sie vertreten. Ihre Makroperspektive kann zu einem wichtigen Werkzeug für den UP werden. So empfiehlt es sich, eine Eskalationslinie auf einer höheren Managementebene zu ziehen. Wenn im UX Gremium beispielsweise keine Einigung erzielt werden kann, muss der UP die Möglichkeit haben, diesen Umstand den Vorgesetzten der Systemeigner zur Klärung zu übergeben. Die Vorgesetzten sind an einer gütlichen Einigung interessiert. Noch mehr allerdings daran, dass sie sich gar nicht erst einmischen müssen. Die Systemeigner werden sich also bemühen, nur diese Fälle eskalieren zu lassen, die wirklich nicht anders entschieden werden können. Damit ist der UP aus der Spannungsrolle und deren hoher Verantwortung heraus genommen.

### 8.0 Literatur

G|UPA (2009). Berufsfeld Usability Beschreibung der Rollen und Aufgaben von Usability Professionals im benutzerorientierten Entwicklungsprozess.

<http://germanupa.de/german-upa/berufsfeld> [01.04.2009]

ISO/DIS 9241-210 (2008). Ergonomics of human-system interaction - Part 210: Human-centred design for interactive systems. Berlin: Beuth Verlag.

Sarodnick, F.; Brau, H. (2006): Methoden der Usability-Evaluation. Bern: Hans Huber Verlag.