Serviceorientierte Architekturen: Vom Konzept zum Einsatz in der Praxis

Roger Heutschi, Christine Legner

Bericht Nr.: BE HSG/ CC BN3 / 2
Lehrstuhl: Prof. Dr. H. Österle
Version: 1.0
Datum: 21. September 2005

Universität St. Gallen - Hochschule für Wirtschafts-, Rechts- und Sozialwissenschaften (HSG)

Institut für Wirtschaftsinformatik
Müller-Friedberg-Strasse 8
CH-9000 St. Gallen
Tel.: ++41 / 71 / 224 2420
Fax: ++41 / 71 / 224 2777

Prof. Dr. A. Back
Prof. Dr. W. Brenner (geschäftsführend)
Prof. Dr. H. Österle
Prof. Dr. R. Winter

Inhaltsverzeichnis

1 Einleitung 3

2 Serviceorientierte Architekturen 4

2.1 Definition 4

2.2 Umsetzung Serviceorientierter Architekturen 4

3 Fallstudien zum Einsatz von SOA in der Praxis 6

3.1 Beispiel Sparkassen Informatik 7

3.2 Beispiel Interessengemeinschaft Haustechnik (IGH) 7

3.3 Beispiel Telekommunikationsunternehmen 8

3.4 Beispiel Deutsche Post Brief 9

3.5 Beispiel Retailbank 9

3.6 Beispiel Bankenplattform 10

4 Einsatzbereiche Serviceorientierter Architekturen 11

4.1 SOA als standardisierte Integrationsinfrastruktur 14

4.2 SOA zur Entflechtung der Applikationslandschaft 15

4.3 Services als vermarktbare IT-Produkte 18

5 Zusammenfassung und Ausblick 20

Literatur 21

© HSG / IWI / CC BN3 / 2

1 Einleitung

Bei der Umsetzung neuer Geschäftsanforderungen und Geschäftsmodelle stoßen Unternehmen immer häufiger an die Grenzen der gewachsenen Systemlandschaft: Angesichts der Vielzahl und Heterogenität der eingesetzten Informationssysteme stellen sie fest, dass Anpassungen nur mit hohem Integrationsaufwand zu bewältigen sind und jede zusätzliche Integrationsbeziehung die Wartbarkeit und Beherrschbarkeit der Gesamtlandschaft beeinträchtigt [QuWi03, 51]. Von Serviceorientierten Architekturen (SOA) erwarten sie, dass Anwendungslogik künftig als plattformunabhängiger Service (bzw. Dienst) bereitgestellt wird und sich so beliebig verteilen und dynamisch zu Geschäftsprozessen verknüpfen lässt [CKMT03, 30]. Dieser Vision stehen bisher nur beschränkt Erfahrungen mit der Umsetzung des SOA-Konzeptes und dessen produktiven Einsatz gegenüber.

Ziel dieses Arbeitsberichtes ist es, den Stand der Umsetzung von SOA in der Praxis aufzuzeigen und typische Einsatzbereiche abzuleiten. Grundlage der Analyse bilden Unternehmen, die eine Serviceorientierte Zielarchitektur entwickelt und bereits Erfahrungen in der Umsetzung gesammelt haben. Angesichts der geringen Anzahl dokumentierter SOA-Projekte wurden die sechs Fallbeispiele so gewählt, dass sie die Breite der in der Praxis verfolgten Ziele und Architekturentscheidungen aufzeigen. Die Fallbeispiele wurden entweder von den Autoren erhoben oder sind in der Literatur ausführlich dokumentiert.

Der Arbeitsbericht ist wie folgt aufgebaut: Abschnitt 2 definiert SOA anhand von vier Charakteristika und zeigt verschiedene Umsetzungsvarianten verteilter Anwendungsarchitekturen. Abschnitt 3 gibt einen Überblick über die SOA-Projekte in den sechs Fallbeispielen. Abschnitt 4 zeigt, dass die wenigsten Projekte sämtliche Aspekte einer SOA gleichermaßen verfolgen. Ausgehend von den Projekttreibern, -zielen und Architekturentscheidungen leitet er drei typische Einsatzbereiche ab. Der Arbeitsbericht schließt mit Zusammenfassung und Ausblick.

© HSG / IWI / CC BN3 / 2

2 Serviceorientierte Architekturen

2.1 Definition

Es gibt bisher keine allgemein anerkannte Definition des Begriffes SOA. Die Literatur [vgl. W3C04; PaGe03; BCHP05; NeLo04, 72f.] kennzeichnet SOA als verteilte Anwendungsarchitektur mit folgenden Charakteristika:

- *Interoperabilität*: Sowohl zur Servicebeschreibung wie auch für Interaktion, Kommunikation und Nutzung der Services werden einheitliche, breit akzeptierte Standards verwendet. Erst die Abstützung auf offene, plattformunabhängige und verbreitete Standards ermöglicht die Interoperabilität eines Dienstes und seine Verwendung in unterschiedlichem Kontext.
- *Schnittstellenorientierung*: Ein Service stellt eine zeitlich stabile Schnittstelle dar, die von den Implementierungsdetails, z.B. der verwendeten Programmiersprache und den Software-Komponenten, abstrahiert. Die Schnittstelle ist selbstbeschreibend, d.h. der Service ist über Metadaten technisch und fachlich vollständig spezifiziert. SOA trennen damit die Beschreibung der Funktionalität in Form einer Schnittstelle von ihrer Implementierung.
- *Modularisierung*: Eine SOA strukturiert die Applikationsarchitektur in eine überschaubare Menge teilautonomer Subsysteme (Domänen und Services). Nach bekannten Prinzipien der Komponentenbildung werden dabei Funktionalität oder Ressourcen mit grosser Abhängigkeit (Kohäsion) untereinander so zusammengefasst, dass sie gegenüber den anderen Subsystemen eine möglichst geringe Abhängigkeit (lose Kopplung) aufweisen.
- *Anwendungsorientierung*: Services bieten Funktionalität auf einer groben, an Geschäftsaktivitäten orientierten Granularitätsebene und kapseln fachlich abgrenzbare, aus geschäftlicher Sicht sinnvolle und wertschöpfende Leistungen.

2.2 Umsetzung Serviceorientierter Architekturen

Während Unternehmen bereits in der Vergangenheit Services auf Basis verteilter objekt- und komponentenbasierter Technologien implementierten, gelten mittlerweile Web Services als viel versprechender Ansatz für die SOA-Umsetzung [Bagh05, 152]. Web Services bieten mit WSDL einen Standard für die Beschreibung von Serviceschnittstellen, mit SOAP ein

© HSG / IWI / CC BN3 / 2

Protokoll für den Nachrichtenaustausch zwischen Services und mit UDDI eine Spezifikation für einen Verzeichnisdienst zur Publikation und Veröffentlichung von Servicebeschreibungen. Sie basieren auf verbreiteten plattform- und implementierungsneutralen Standards wie http und XML und gelten als viel versprechende Kandidaten für das Wrapping (engl. für ‚Umhüllen') bestehender Anwendungen und die Realisierung einer implementierungsunabhängigen Serviceschicht [Bake02, 628].

Web Services haben in letzter Zeit aufgrund ihrer Unterstützung durch die grossen Softwarehersteller Aufschwung erhalten [z.B. Wood04; EAAC04, 40]. Die breit akzeptierten Web Service-Standards haben bei der Spezifikation von Diensten für eine Serviceorientierte Integrationsplattform allerdings noch nicht den Reifegrad bestehender komponentenbasierter Ansätze wie z.B. CORBA oder J2EE erreicht [vgl. KoLe04, 18; HaSc02, 16]. Web Services können deshalb in grösseren Integrationsprojekten traditionelle Middleware- und EAI-Dienste nicht ersetzen, sondern bauen auf diesen auf.

© HSG / IWI / CC BN3 / 2

3 Fallstudien zum Einsatz von SOA in der Praxis

Der folgende Abschnitt schildert für die in Tabelle 3-1 aufgeführten Unternehmen Zielsetzung und Umfang der SOA-Projekte sowie den Stand der Umsetzung.

Fallbeispiel	Quelle	Überblick über das SOA-Projekt		
		Anwendungsbereich	Status	Umsetzung
Sparkassen Informatik	[ZMCO04]	Überbetrieblich: Verbindung Backoffice (Shared Service) und Frontoffice; mehrere Geschäftsprozesse	Produktiv	Integrationslösung auf Basis IBM-WebSphere / J2EE und Web Services, mehr als 500 Services
IGH	[Heut03]	Überbetrieblich: Hersteller, Händler, Installateure; Angebots- und Bestellabwicklung	Produktiv	.Net-Implementierung von Web Services, Installation je einer Serviceinstanz bei Unternehmen
Telekommunikationsunternehmen	Eigene Erhebung	Unternehmensintern: Vertriebs- und Produktionsprozesse	In Entwicklung	Enterprise Service Bus auf Basis J2EE und Web Services im Aufbau
Deutsche Post Brief	[HeBK04] [HeSc05]	Unternehmensintern: Kerngeschäftsprozesse	Produktiv	Eigenentwickelter Service Backbone auf Basis J2EE, 20-30 Services
Retailbank	Eigene Erhebung	Unternehmensintern: Kundenberatungsprozess	Produktiv	SAP Composite Application mit ABAP Objects u. J2EE, 20-30 Services
Bankenplattform	Eigene Erhebung	Überbetrieblich: mehrere Geschäftsprozesse für den Bankbetrieb	Teilweise produktiv	J2EE-Plattform auf Basis SAP Netweaver und Web Services

Tabelle 3-1: Übersicht über die Fallstudien

Die Projekte unterscheiden sich erheblich bezüglich der eingesetzten Technologien (J2EE, Microsoft .Net) und des Projektumfangs. Einige Unternehmen setzen SOA-Konzepte zunächst in einem ausgewählten Geschäftsprozess oder Unternehmensbereich um, während andere einen unternehmensweiten Ansatz wählen. SOA findet sowohl im innerbetrieblichen Bereich (Fokus des Telekommunikationsunternehmens, der Deutschen Post Brief und der Retailbank) wie auch in der überbetrieblichen Zusammenarbeit (Fokus der Sparkassen

© HSG / IWI / CC BN3 / 2

Informatik, der IGH und des Bankenplattformanbieters) Anwendung. Da einige der aufgeführten Fallbeispiele noch in Entwicklung bzw. erst seit kurzem produktiv sind, sind die Nutzenaussagen großteils hypothetisch.

3.1 Beispiel Sparkassen Informatik

Der IT-Dienstleister Sparkassen Informatik betreibt die zentralen Backoffice-Anwendungssysteme für mehr als 200 Sparkassen [ZMCO04]. Die Sparkassen selbst nutzen verschiedenste Frontoffice-Applikationen, die sich hinsichtlich Plattform (z.B. J2EE, Microsoft .Net), Programmiersprache (u.a. Java, C#, Perl) und Schnittstellentechnologien (z.B. Java RMI, proprietäre RPC-Mechanismen) unterscheiden. Um diese mit der zentralen Backoffice-Anwendung zu verbinden, implementierte Sparkassen Informatik in der Vergangenheit für jede Frontoffice-Applikation eine individuelle Integrationslösung. Die Folge war eine Vielzahl eingesetzter Werkzeuge und Integrationstechnologien mit verschiedenen Konzepten für die Beschreibung, Namensgebung und Abgrenzung von Services.

Um zahlreichen Schnittstellentechnologien und Integrationslösungen zu reduzieren, den Entwicklungsaufwand auf Kundenseite gering zu halten und gleichzeitig eine große Menge Services (jährlich ca. 100 neu entwickelte Services) zu unterstützen, konzipierte das Unternehmen eine Integrationsschicht auf Basis der Standards WSDL und SOAP. Diese enthält momentan rund 500 Web Services mit gekapselten Geschäftsfunktionen und soll mittelfristig die individuellen Integrationslösungen ersetzen.

3.2 Beispiel Interessengemeinschaft Haustechnik (IGH)

Die rund 70 Hersteller und Händler der Haustechnikbranche, die sich in der Interessengemeinschaft Haustechnik (IGH) zusammengeschlossen haben, verfolgen das Ziel, den standardisierten elektronischen Datenaustausch zu fördern [Heut03]. Anfragen, Bestellungen und Lieferscheine wurden früher zwischen Herstellern, Händlern und Installateurbetrieben per Telefon oder Fax übermittelt. Dies führte zu manuellem Aufwand bei allen Beteiligten und zu zahlreichen Fehlern.

Um einen standardisierten Austausch von Geschäftsdokumenten zu unterstützen, entwickelte die IGH ein XML-basiertes Katalog- und Geschäftsdokumentenformat sowie die Spezifikation eines Web Service. Diesen stellt sie interessierten Parteien auch als Server- (für

© HSG / IWI / CC BN3 / 2

Hersteller und Händler) oder als Client-Implementierung (für Händler und Installateure) auf Basis von Microsoft .Net zur Verfügung. Der Web Service übernimmt die standardisierte Übermittlung und syntaktische Validierung der Geschäftsnachrichten, die anschließend in den Warenwirtschaftssystemen verarbeitet werden. Auf diese Weise erfolgt die Integration zwischen den ERP-Systemen der Haustechnik-Produzenten bzw. -Händler und den spezialisierten Branchenlösungen auf Seiten der Installateurbetriebe.

3.3 Beispiel Telekommunikationsunternehmen

Das Telekommunikationsunternehmen hat in den letzten Jahren sein Produktangebot ständig erweitert und zusätzliche Vertriebskanäle (z.B. Internet, Call Center, Vertriebspartner) eingeführt. Neben einem hostbasierten Anwendungssystem für die Auftragsabwicklung und Produktion entstanden pro Kundensegment (z.B. Geschäftskunden, Massenmarkt, Reseller) und Zugangskanal mehrere Vertriebssysteme. Diese Systeme duplizieren die Auftragsprüf- und -abwicklungslogik und greifen über fachlich und technisch unterschiedliche Schnittstellen auf die Produktionssysteme zu. Dies schränkt die unternehmerische Agilität des Unternehmens zunehmend ein. So ist heute z.B. der ISDN-Bereitstellungsprozess je nach Vertriebskanal unterschiedlich stark automatisiert, obwohl die Fertigung identisch ist. Die Einführung neuer Produkte bzw. Produktbündel hat Anpassungen in mehreren Anwendungssystemen zur Folge und erhöht die Time-to-Market.

Das Unternehmen führt gegenwärtig ein SOA-Projekt im Bereich der Vertriebs- und Produktionsprozesse durch. Die Zielarchitektur soll Schnittstellen auf Basis eines zentralen Enterprise Service Bus (ESB) technisch standardisieren, die in den Altsystemen enthaltene fachliche Funktionalität entflechten und den Aufwand für Integrations- und Entwicklungsprojekte durch Wiederverwendung reduzieren. Ausgangspunkt des SOA-Projekts ist die Definition von Anwendungsdomänen mit unterschiedlichen fachlichen Dynamiken bzw. Lebenszyklen. Diese gruppieren logisch zusammengehörige Prozessbestandteile mit ihren unterstützenden Funktionen und Daten und bieten ihre Funktionalität über wohldefinierte Serviceschnittstellen an. Der fachliche Leistungsumfang (gekapselte Funktionalität, Datenmodell) existierender Schnittstellen wird dabei generalisiert, um Anforderungen mehrerer Prozessvarianten zu unterstützen. Neben der Domänenbildung stellt die fachliche Standardisierung von Kerngeschäftsobjekten eine weitere wichtige Maßnahme zur SOA-Umsetzung dar: Um den semantischen Integrationsaufwand zwischen Anwendungssystemen auf Datenebene, d.h. die Übersetzung des Datenmodells einer Anwendung in dasjenige der

© HSG / IWI / CC BN3 / 2

anderen, zu reduzieren, entwickelt das Unternehmen für rund 30 wichtige Geschäftsobjekte (z.B. Kunde, Vertrag, Produkt oder Rufnummer) ein für alle Services gemeinsames, applikationsübergreifendes Datenmodell.

3.4 Beispiel Deutsche Post Brief

Probleme mit der historisch gewachsenen Applikationsarchitektur waren die Auslöser für die Umsetzung einer SOA bei Deutsche Post Brief [HeSc05], dem größten Briefdienstleister Europas. Auch hier stellte die Bildung fachlicher Domänen neben der Implementierung einer standardisierten technischen Integrationsinfrastruktur einen wichtigen Schritt zur Umsetzung der SOA dar. Das Unternehmen zerlegte Kernprozesse wie „Produktentwicklung“, „Marketing- und Vertriebsplanung“ etc. in ihre betriebswirtschaftlichen funktionalen Bestandteile und bildete Applikationsdomänen für stark voneinander abhängige Geschäftsfunktionen und -daten. Für den domänenexternen Zugriff auf Funktionen und Daten wurden Serviceschnittstellen definiert, die in verschiedenen Geschäftsprozessen wieder verwendet werden können, eine aus Geschäftssicht sinnvolle Aufgabe übernehmen sowie längerfristig stabil sind. Die Kopplung von Serviceanbietern und -nutzern übernimmt der *Service Backbone (SBB)*, der aus Produkten verschiedener Anbieter in einem Best-of-Breed-Ansatz entwickelt wurde und seit Ende 2001 produktiv eingesetzt wird.

Mit der Implementierung von Services ging teilweise auch eine Harmonisierung der Applikationslandschaft einher: Zur Reduktion der Komplexität und des Wartungs- bzw. Betriebsaufwands realisiert Deutsche Post Brief insbesondere bei Neuentwicklungen die Servicefunktionalität physisch möglichst in einem System (z.B. einer zentralen Kundendatenbank). Um die Wiederverwendung und Wiederverwendbarkeit von Services zu fördern, bildete das Unternehmen einen dedizierten Bereich, der für das Architekturmanagement zuständig ist. Dieser erstellt Vorgaben für die fachliche Servicebeschreibung, formalisiert den Serviceentwicklungs- und -nutzungsprozess und prüft die Architekturkonformität neuer IT-Projekte. Aktuell weist die IT-Architektur 20-30 Services auf, die im Durchschnitt von zwei bis drei Applikationen verwendet werden.

3.5 Beispiel Retailbank

Eine Retailbank migrierte von eigenentwickelten Applikationen auf eine Standardlösung. Für die Kundenberater im Frontoffice waren Benutzeroberfläche und Prozessführung der neuen

© HSG / IWI / CC BN3 / 2

Lösung in ca. 80% der Geschäftsvorfälle zu kompliziert. Gleichzeitig deckte sie nicht alle im Beratungsprozess benötigten Funktionen und Informationen ab, da z.B. Unterschriftenmuster zur Kundenauthentifizierung oder die Verwaltung von Kontokarten in separaten Systemen enthalten waren.

Mit der Entwicklung eines Beraterarbeitsplatzes sollten die Kundenberater eine auf ihre Bedürfnisse zugeschnittene Benutzeroberfläche erhalten, die sämtliche benötigten Funktionen aus unterschiedlichen Anwendungssystemen integriert. Das Unternehmen realisierte den Beraterarbeitsplatz auf Basis des SAP Composite Application Frameworks mit einer webbasierten Benutzeroberfläche und der Benutzersteuerung über sog. Guided Procedures. Der Beraterarbeitsplatz nutzt die in den existierenden Anwendungen implementierte Funktionalität über Schnittstellen, die so zu Services zusammengefasst wurden, dass einzelne Services jeweils einen Arbeitsschritt der Guided Procedures unterstützen. Aus Zeit-, Kosten- und Performanzgründen verzichtete die Retailbank dabei auf die vollständige technische Schnittstellenstandardisierung auf Serviceebene. Da ein Grossteil der benötigten Services in der SAP-Standardlösung implementiert ist, werden diese direkt als ABAP Objects Komponenten aufgerufen, während Services weiterer Anwendungssysteme z.B. über Java-Schnittstellen eingebunden sind. Momentan sind 20-30 im Arbeitsplatz genutzte Services implementiert.

3.6 Beispiel Bankenplattform

Der Anbieter einer Bankenplattform stellt seinen Kunden eine modular aufgebaute Gesamtlösung für den Bankbetrieb zur Verfügung, die verschiedene Geschäftsmodelle unterstützt. Z.B. nutzen Banken, die ein umfassendes Produktangebot selbst produzieren und vertreiben, die gesamte Funktionalität der Plattform. Sog. „Kooperationsbanken“ hingegen lagern gewisse Basisprozesse (z.B. das Wertschriftenmanagement, die Kontenbewirtschaftung etc.) an Partner aus, während „Tochterbanken“ nur Verkaufs- und Beratungsprozesse ausführen und die zentralen Management-, Unterstützungs- und Produktionsprozesse dem Stammhaus überlassen.

Damit die Kunden in der Lage sind, Wertschöpfungsaktivitäten selektiv auszulagern, hat der Plattformanbieter eine fachliche Domänenarchitektur ausgehend von den Bankprozessen und -produkten entwickelt. Auslagerungskandidaten sind vor allem Applikationsdomänen, die Produktionsprozesse unterstützen (z.B. Wertpapierabwicklung, Kontoführung,

© HSG / IWI / CC BN3 / 2

Zahlungsverkehr etc.) sowie zentrale Unterstützungsprozesse und -aktivitäten (z.B. Archivierung). Die Plattformarchitektur definiert für jede dieser Domänen Services, welche sie domänenexternen Applikationen zur Verfügung stellen. Domänenübergreifende Geschäftsprozesse nutzen Domänenfunktionalität nur über die zentrale Integrationsinfrastruktur und die Serviceschnittstellen. Das Ersetzen oder organisatorische Verlagern der Anwendungssysteme einer Domäne hat dadurch kaum Auswirkungen auf domänenexterne Prozesse und Applikationen, solange die neuen Systeme die von der Plattform definierten Services unterstützen können.

4 Einsatzbereiche Serviceorientierter Architekturen

Die Fallbeispiele zeigen, dass die Charakteristika bzw. Gestaltungsprinzipien von SOA je nach Projekt unterschiedlich priorisiert werden (s. Tabelle 4-1). In allen Projekten findet sich der Aspekt der Schnittstellenorientierung durch den Zugriff auf bestehende Applikationen über Services wieder. Unterschiede existieren dagegen bezogen auf den Grad der Interoperabilität der Serviceschnittstellen: Während die Retailbank verschiedene technische Schnittstellenstandards zulässt, setzt Deutsche Post Brief als technischen Standard auf J2EE und Sparkassen Informatik auf plattformunabhängige Web Service Standards. Eine weitergehende Standardisierung auf fachlicher Ebene nimmt das Telekommunikationsunternehmen durch Definition eines applikationsübergreifenden Datenmodells und IGH durch Entwicklung dokumentenorientierter Web Services für den zwischenbetrieblichen Datenaustausch vor. Die Modularisierung der Applikationsarchitektur ist nur für einen Teil der Unternehmen relevant. So trifft etwa Sparkassen Informatik weder besondere Maßnahmen zur Reduktion logischer Abhängigkeiten in der Applikationsarchitektur, noch nimmt das Unternehmen Anpassungen am fachlichen Leistungsumfang existierender Schnittstellen vor. Dagegen bilden z.B. Deutsche Post Brief oder die Retailbank Domänen und definieren Services auf Basis der Anforderungen unterschiedlicher Geschäfts- oder Benutzerprozesse.

Es zeichnen sich beim Vergleich der Projektschwerpunkte drei typische Einsatzbereiche für SOA ab. Diese lassen sich bezüglich der Projekttreiber, der zu treffenden Architekturentscheidungen, der Nutzenpotenzialen und Herausforderungen unterscheiden (s. Tabelle 4-2).

© HSG / IWI / CC BN3 / 2

Fallbeispiel	Ausprägung der SOA-Charakteristika bei den untersuchten Projekten			
	Interoperabilität	**Schnittstellenorientierung**	**Modularisierung**	**Anwendungsorientierung**
Sparkassen Informatik	Web Services	Implementierungsunabhängig beschriebene und in zentralem Verzeichnis registrierte Services für den Zugriff auf Backendsysteme durch Frontoffice-Applikationen	-	-
IGH	Web Services, branchenweite fachliche Standardisierung von Geschäftsdokumenten	Implementierungsunabhängig beschriebene Schnittstelle für den Austausch von Geschäftsdokumenten	-	-
Telekommunikations-unternehmen	J2EE-Standards (innerbetrieblicher Service Bus) und Web Services (für unternehmensexternen Zugriff), standardisierte Datenmodelle	Implementierungsunabhängig beschriebene und in zentralem Verzeichnis registrierte Services für den Zugriff auf Auftragssteuerungs- und Produktionssysteme	Domänen- und Servicebildung anhand unterschiedlicher fachlicher Dynamiken und fachlicher Spezifität von Logik	Kapselung von in unterschiedlichen Auftragsabwicklungsvarianten wieder verwendbaren Prozessaufgaben bzw. prozessübergreifenden Geschäftsobjekten
Deutsche Post Brief	J2EE-Standards (innerbetrieblicher Service Bus) und Web Services (für unternehmensexternen Zugriff)	Implementierungsunabhängig beschriebene und in zentralem Verzeichnis registrierte Services für den Zugriff auf verschiedene Applikationsdomänen	Domänen- und Servicebildung anhand von Abhängigkeiten zwischen Fachfunktionen	Kapselung von in unterschiedlichen Kernprozessen wieder verwendbare Prozessaufgaben bzw. prozessübergreifenden Geschäftsobjekte
Retailbank	-	In zentralem Verzeichnis registrierte Services als Zugriffspunkte auf Backoffice-Funktionalität durch gesteuerte Benutzerprozesse	Trennung von Fach- und Benutzerführungslogik	Kapselung von automatisierbaren oder interaktiven Aufgaben der Kundenberatungsprozesse
Bankenplattform	J2EE-Standards (innerbetrieblicher Servicebus)	Implementierungsunabhängig beschriebene und in zentralem Verzeichnis registrierte Services potenziell auslagerbarer Applikationsdomänen	Domänen- und Servicebildung anhand von Bankprodukten und Wertschöpfungsaktivitäten	Kapselung von wieder verwendbaren Prozessaufgaben in Produktionsprozesse einzelner Bankenprodukte bzw. produktunabhängiger Supportprozesse

Tabelle 4-1: Ausprägung der SOA-Charakteristika in den Fallbeispielen

© HSG / IWI / CC BN3 / 2

	Typische Treiber	Architekturentscheidungen	Nutzenpotenziale	Herausforderungen
Standardisierte Integrationsinfrastruktur	• Paralleler Einsatz vieler verschiedener Technologien und Plattformen • Hohe Anzahl an Schnittstellen • Vielfalt an Integrationslösungen	• Technische Servicestandards: Schnittstellentechnologien, Kommunikationsprotokolle, Nachrichtenformate, Sicherheitsmechanismen etc. • Servicespezifikation: Vorgaben für die technische Servicebeschreibung • Integrationsinfrastruktur: Technologien und Produkte für die Service-Integrationsinfrastruktur	• Plattformunabhängige Nutzung existierender Funktionalität • Reduktion der Vielfalt an Schnittstellentechnologien und Integrationslösungen • Senkung der Kosten für die Systemintegration und Wartung • Bündelung des technischen Know-hows	• Zu starke Technologie- und Einzelprojektorientierung • Unveränderte Kapselung existierender, technisch orientierter Schnittstellen
Entflechtung der Applikationsarchitektur	• Mehrfache Implementierung von Funktionalität in unterschiedlichen Anwendungen • Redundante Datenhaltung • Enge Verknüpfung von Präsentations-, Ablaufsteuerungs- und Fachlogik	• Applikationsarchitektur: Anwendungsdomänen • Servicedesign: Designprinzipien für Services (z.B. bez. Wiederverwendbarkeit) • Servicespezifikation: Vorgaben für die fachliche Servicebeschreibung • Fachliche Servicestandards: Datenmodelle, Prozessmodelle etc. • Architektur-Governance: Organisationsstrukturen und Prozesse für die Serviceentwicklung und -nutzung	• Beherrschbare IT-Projekte und schnelle Umsetzung von neuen Geschäftsanforderungen • Senkung der Implementierungs- und Wartungskosten durch stärkere Wiederverwendung • Reduktion des semantischen Transformationsaufwands bei der Applikationsintegration	• Design- / Entwicklungsaufwand (Domänenbildung, wieder verwendbare Services und standardisierte Datenmodelle) • Bestimmung der geeigneten Servicegranularität und -spezialisierung • Integritätsbeziehungen bei der Modularisierung monolithischer Applikationen
Services als vermarktbare IT-Produkte	• Einführung von Marktmechanismen in der Zusammenarbeit zwischen IT-Abteilung und Fachbereichen • Nutzung von Synergien durch Bildung von Shared Service • Fremdbezug von IT-Dienstleistungen bzw. Erweiterung des Angebots von IT-Dienstleistern durch Business Process Outsourcing und Application Service Providing	• Wertschöpfungs- bzw. Geschäftsmodelle für Services • Gestaltung der Leistungsbeziehung: Organisatorische und vertragliche Gestaltung (Kompetenzen, Service Levels etc.), „Marktregeln“ bei der Servicenutzung (Bezugspflicht, Leistungsverrechnung etc.).	• Verrechenbarkeit von IT-Leistungen • Bessere Herauslösbarkeit von Teilprozessen und unterstützenden Applikationen aus der Gesamtarchitektur • Nutzung von Spezialisierungsvorteilen und Skaleneffekten in der Erbringung von IT-Dienstleistungen	• Machtstrukturen, Partikularinteressen oder zu unterschiedliche Anforderungen betroffener Organisationseinheiten • Marktrisiken des externen Servicebezugs

Tabelle 4-2: Einsatzbereiche von SOA

© HSG / IWI / CC BN3 / 2

4.1 SOA als standardisierte Integrationsinfrastruktur

Angesichts einer wachsenden Anzahl heterogener Applikationsplattformen, Betriebssysteme, Integrationsinfrastrukturen und Entwicklungswerkzeuge verfolgen Unternehmen wie Sparkassen Informatik oder IGH mit SOA das Ziel, die Integration bestehender Anwendungssysteme zu standardisieren und dazu eine plattformübergreifende Integrationsinfrastruktur aufzubauen. Diese Zielsetzung ist vergleichbar mit typischen EAI-Zielen [KoLe04]: Über eine zentrale Middlewareinfrastruktur sollen Applikationen lose gekoppelt über den Austausch von Nachrichten miteinander interagieren. Projekte in diesem Einsatzbereich konzentrieren sich stark auf die SOA-Eigenschaften der (technischen) Interoperabilität und Schnittstellenorientierung.

Nutzenpotenziale entstehen durch die Festlegung technischer Konventionen und die Verwendung verbreiteter Standards für Services:

- Services können unabhängig von internen Komponentenmodellen, Implementierungssprachen und Transportprotokollen sowie von Betriebssystemen und Hardwareplattformen der Servicenutzer oder -anbieter aufgerufen werden.
- Die Vielfalt der Schnittstellentechnologien und Integrationslösungen in der gesamten IS-Architektur wird durch die Einigung auf einen Standard reduziert.
- Standardisierte Schnittstellenbeschreibung und Werkzeugunterstützung senken den Entwicklungs- und Integrationsaufwand bei der Umsetzung neuer Anforderungen und erfordern geringeres Spezialwissen bei den IT-Mitarbeitern.

So konnte z.B. Sparkassen Informatik dank der Unterstützung von Web Service Standards durch existierende Internet-Infrastrukturen und Anwendungsplattformen eine hohe Interoperabilität mit unterschiedlichen Frontend-Technologien erreichen. Eine standardisierte Servicebeschreibung in einem zentralen Metadatenverzeichnis verringerte den Entwicklungsaufwand für Serviceanbieter und -nutzer merklich: Generatoren produzieren anhand der Beschreibungen weitgehend automatisiert WSDL-Dokumente und Code-Gerüste für die Serviceimplementierung. In den gängigen modernen Entwicklungsumgebungen können Servicenutzer mit den WSDL-Beschreibungen Proxies für den Serviceaufruf in verschiedenen Programmiersprachen generieren. Im IGH-Beispiel bilden die Web Services eine technisch und (durch einheitliche Nachrichten- und Katalogformate) semantisch

© HSG / IWI / CC BN3 / 2

standardisierte Infrastruktur, die eine m:n-Verbindung zwischen Anbietern (Herstellern und Händlern) und Kunden (Installateuren) erlaubt. Die Beispiele von Sparkassen Informatik und IGH zeigen, dass der Mehrwert von Web Services im Vergleich zu herkömmlichen Integrationstechnologien vor allem im überbetrieblichen Bereich liegt: Dank der Verwendung von Web Service-Standards lässt sich die Integration mit externen Systemen vergleichsweise einfach realisieren.

Eine zu starke Konzentration auf die technischen Aspekte einer SOA birgt jedoch das Risiko, nur beschränkt wieder verwendbare Services zu definieren. Sie kann Unternehmen zu einem pragmatischen Vorgehen verleiten, in dem sie Schnittstellen durch Wrapping zwar technisch einheitlich kapseln, aber bezüglich ihrer fachlichen Funktionalität und Granularität nicht verändern [NeLo04, 171; Wilk03]. Da die Services applikationsspezifische Daten- und Funktionsmodelle verwenden, sind sie für die Wiederverwendung in einem anderen als dem ursprünglich geplanten Kontext eventuell ungeeignet. Zwar nutzen die Integrationslösungen einheitliche Schnittstellentechnologien und Kommunikationsprotokolle, diese reduzieren aber nicht den semantischen Transformationsaufwand und die fachlichen Abhängigkeiten zwischen Applikationen.

4.2 SOA zur Entflechtung der Applikationslandschaft

Die erweiterte Zielsetzung, nicht nur technisch zu standardisieren, sondern die bestehende Applikationslandschaft logisch zu entflechten und fachliche Redundanzen und Abhängigkeiten zu beseitigen, steht im Mittelpunkt der SOA-Projekte des Telekommunikationsunternehmens und bei Deutsche Post Brief. Projekttreiber sind hier Aufwand und Kosten, die bei der Realisierung applikationsübergreifender Prozesse oder dem Herauslösen einer Applikation in der bestehenden Systemlandschaft entstehen. Im Rahmen einer SOA strukturieren Unternehmen Funktionen und Daten aufgrund ihrer logischen Abhängigkeiten und nach den Anforderungen integrierter, anwendungssystemübergreifender Geschäfts- oder Benutzerprozesse [WiSc05, 52]. Dabei kapseln sie existierende Funktionen und Schnittstellen nicht einfach hinter einer Technologie wie Web Services, sondern bündeln oder erweitern sie zu Services geeigneter fachlicher Granularität und vereinheitlichen ggf. Datenmodelle oder fachliche Nachrichtentypen. Sie berücksichtigen bei der Serviceentwicklung zukünftigen Integrationsbedarf und realisieren integrationsfähige, wieder verwendbare Komponenten, um Geschwindigkeits- und Flexibilitätsvorteile bei der Entwicklung neuer Anwendungen zu erreichen [Uhl04, 317].

© HSG / IWI / CC BN3 / 2

SOA-Projekte in diesem Bereich starten in der Regel mit der Modellierung der Unternehmensarchitektur als Grundlage der Domänen- und Servicebildung. Das Telekommunikationsunternehmen trennt in seiner Zielarchitektur Systembestandteile nach ihrer fachlichen Dynamik und damit der erwarteten Änderungshäufigkeit: Kundensegmentspezifische Domänen implementieren die Endbenutzerschnittstellen („Präsentationsebene“) und davon getrennt die vertriebskanalspezifische Auftragssteuerungslogik („Zugangsebene“). Unterliegen die in der „Präsentationsebene“ enthaltenen Frontendsysteme häufigen Änderungen (z.B. aufgrund von Anpassungen der Benutzeroberflächen, Logos oder Marketinginformationen), so bleibt die auf der „Zugangsebene“ enthaltene Vertriebslogik über Monate stabil und muss allenfalls für Verkaufsoffensiven oder die Vermarktung neuer Produktbündel angepasst werden. Dagegen weisen Domänen, die allen Vertriebsprozessen gemeinsame, kundenkanalneutrale Fachlogik enthalten und die sog. „Kernservices-Ebene“ bilden (z.B. Verfügbarkeitsprüfungen), eine sehr hohe Stabilität auf. Sie werden nur in den seltenen Fällen angepasst, in denen sich die technische Abwicklung der Produkte oder das Datenmodell der Kerngeschäftsobjekte ändern. Deutsche Post Brief investiert bei der Serviceimplementierung viel Arbeit in die Bestimmung der geeigneten Servicegranularität und die Entwicklung des übergreifenden Datenmodells. Um z.B. einen Service für die Bearbeitung von Kundendaten im Zeitablauf möglichst stabil zu halten, muss sein Datenmodell sämtliche von aktuellen und künftigen Servicenutzern geforderte Sichten auf die Entität „Kunde“ abbilden. Das bedeutet, dass der Service z.B. sowohl für Marketing-Kampagnen als auch für die Auftragsabwicklung die benötigten Informationen (z.B. Kundentyp, Anschrift) in der notwendigen Granularität (z.B. Unterscheidung von Liefer- und Rechnungsanschrift) liefern muss. Die Retailbank legte ihren Fokus auf die Entflechtung der in verschiedenen Applikationen implementierten Fachlogik für die an Mitarbeiterprozessen orientierte Benutzerführungslogik. Sie passte existierende Schnittstellen an und fasste sie zu Services zusammen, die interaktive oder vollständig automatisierte Aktivitäten aus den Beraterprozessen unterstützen. So bündelt z.B. der Service „Partner sichern“ zwei Schnittstellen zum Anlegen von Partnergrunddaten und von Partnerrollen und erweitert diese um eine Überprüfung einer Risikoliste.

Wie die Beispiele zeigen, weisen die verfolgten Ziele und Designprinzipien Ähnlichkeiten zur objekt- oder komponentenorientierten Systementwicklung auf. Allerdings betrachten SOA-Projekte in der Regel die applikationsübergreifende Integration und legen den Fokus bei der Serviceimplementierung stärker auf die Wiederverwendung der in verschiedenen

© HSG / IWI / CC BN3 / 2

Anwendungssystemen enthaltenen Funktionen als auf die komplette Neuentwicklung der benötigten Logik. Nutzenpotenziale entstehen durch eine bessere Anpassungsfähigkeit der Applikationsarchitektur und eine einfachere Wiederverwendung von Komponenten:

- Projekte zur Umsetzung integrierter Geschäftsprozesse werden besser beherrschbar, da durch Schichtenbildung anwenderspezifische Benutzerführungslogik, geschäftsfallspezifische Prozesslogik und stabile generische Fachlogik stärker getrennt sind.
- Künftige Geschäftsanforderungen lassen sich in den Informationssystemen schneller abbilden, weil bestehende Services wieder verwendet werden können.
- Dank einer mit der Domänenbildung und den besser kontrollierten Schnittstellen einhergehenden Komplexitäts- und Redundanzreduktion lassen sich die Wartungs- und Betriebskosten senken.
- Eine fachlich getriebene Domänen- und Servicearchitektur erlaubt darüber hinaus eine klare Regelung der organisatorischen Verantwortung für Services.
- Der semantische Transformationsaufwand in Integrationsprojekten lässt sich durch standardisierte Datenmodelle, Nachrichtentypen etc. auf Serviceebene reduzieren.

Eine Herausforderung bei der Modularisierung der Applikationslandschaft besteht im Modellierungsaufwand einer Domänen- und Servicearchitektur, die eine detaillierte Analyse der zu unterstützenden Geschäftsprozesse erfordert. Bei der Entwicklung wieder verwendbarer Services entstehen im Vergleich zu spezialisierten Einzellösungen Mehrkosten durch erhöhten Abstimmungsaufwand. Um die Anzahl bestehender Schnittstellen und Redundanzen zu reduzieren, ist häufig eine fachlich-logische Modularisierung auf Serviceebene nicht ausreichend, sondern es müssen mehrfach vorhandene Funktionen und Daten aus existierenden Systemen herausgelöst und zentralisiert werden. Solange z.B. Kundendaten über drei Anwendungssysteme verteilt sind, muss auch ein übergreifender Service für die Bearbeitung von Kundendaten individuelle Schnittstellen zu den drei Systemen implementieren. Die Modularisierung existierender Anwendungen mit eigenen internen Daten- und Prozessmodellen kann aufgrund von Integritätsbeziehungen zwischen einzelnen Transaktionen schnell sehr aufwändig werden [HaMS03, 24]. Wenn ein Unternehmen Standardsoftware einsetzt, ist es dabei stark von der Softwarelösung und damit von Softwareherstellern abhängig.

© HSG / IWI / CC BN3 / 2

4.3 Services als vermarktbare IT-Produkte

Durch die Umsetzung von SOA-Konzepten bietet der Bankenplattformanbieter seinen Kunden die Möglichkeit, einzelne Wertschöpfungsaktivitäten einfacher an externe Partner auszulagern oder innerhalb des Konzerns zu zentralisieren, und damit Spezialisierungs- und Synergieeffekte zu realisieren. Die Services bzw. durch Services gekapselte Anwendungsdomänen werden damit zu einzeln vermarktbaren Dienstleistungen bzw. Produkte und erhöhen die Flexibilität in der Aufgabenteilung zwischen internen und externen Unternehmenseinheiten [HoRW04, 34]. Damit unterstützt das SOA-Konzept die stärkere Produktorientierung bei IT-Leistungen, die das ressourcenorientierte Produktverständnis (z.B. CPU-Zeit, Bereitstellung einer Applikation) durch eine service- und prozessorientierte Leistungsdefinition (z.B. „Schadensvorfall anlegen“, „Adressvalidierung“) ersetzt [Zarn04, 44]. Dies beinhaltet die Verrechnung von IT-Leistungen über Produktpreise und die Einführung marktähnlicher Mechanismen in der Zusammenarbeit zwischen IT und Fachbereichen. Die flexible Beschaffung solcher IT-Leistungen bei externen Dienstleistern wie z.B. dem Bankenplattformanbieter ist ein logischer nächster Schritt. SOA unterstützt somit das Angebot von Shared Services, d.h. standardisierten, von mehreren Einheiten gemeinsam genutzten Ressourcen [voSc03, 78], oder Outtasking-Modellen, d.h. der gezielten Auslagerung einzelner Prozessaufgaben an Serviceanbieter [ÖsRe03, 565]. Voraussetzung sind jedoch klar definierte Kundenschnittstellen und „Marktregeln“, welche Kompetenzen und Verantwortlichkeiten in der Zusammenarbeit, die Verrechnung von Leistungen, Bezugspflichten etc. festlegen.

Nutzenpotenziale von SOA entstehen durch folgende Eigenschaften:

- Services stellen mit ihrem fachlich klar abgegrenzten, an Wertschöpfungsaktivitäten eines Unternehmens orientierten Leistungsumfang geeignete verrechen- und vermarktbare Einheiten dar.

- Sie besitzen technisch und fachlich stabil definierte, in fachlicher Terminologie beschriebene und kontrollierbare Schnittstellenvereinbarungen. Sie erleichtern damit die Identifikation, Integration und Nutzung von IT-basierten Leistungen auf einer geschäftlichen Ebene.

- Servicebasierte Prozessaktivitäten sind, sofern sie in der implementierenden Applikationsarchitektur ausreichend modular abgebildet sind, durch Verwendung

© HSG / IWI / CC BN3 / 2

von plattformunabhängigen Industriestandards physisch verteilbar. Sie können aus technischer Sicht einfach zentralisiert und unternehmensextern bezogen oder angeboten werden.

Eine Herausforderung der Vermarktung und physischen Verteilung von Services können einerseits die notwendige Modularisierung bestehender Applikationen darstellen (s. Kap. 4.2), andererseits die Machtverhältnisse und Partikularinteressen der betroffenen Organisationseinheiten. Beim Bezug von Services in einem wachsenden, aber noch immer jungen Markt entstehen Risiken aus der Unerfahrenheit bzw. Instabilität der Anbieter (z.B. durch Überverkauf von Kapazität oder unsicherem Geschäftsmodell), Schwierigkeiten bei der Vertragsgestaltung oder technischen Gefahren einer Servicenutzung über das Internet (z.B. Sicherheitslücken, mangelnde Verfügbarkeit, hohe Antwortzeiten) [Kern02, 118].

© HSG / IWI / CC BN3 / 2

5 Zusammenfassung und Ausblick

Die Analyse der sechs Fallbeispiele zeigt, dass die Umsetzung von SOA erst in den Anfängen steckt und dass Unternehmen in ihren ersten SOA-Projekten nicht sämtliche SOA-Zielsetzungen angehen. Der Beitrag identifiziert daher drei Einsatzbereiche von SOA: Der Einsatz von SOA als *standardisierte Integrationsinfrastruktur* erhöht die technische Konnektivität heterogener Applikationen, reduziert die Vielfalt der Schnittstellentechnologien und verringert damit Integrations- und Wartungskosten. Dagegen zielt der Einsatz von SOA zur *Entflechtung der Applikationsarchitektur* auf eine Reduktion von Abhängigkeiten und Redundanzen innerhalb der Applikationsarchitektur und eine stärkere Wiederverwendbarkeit implementierter Funktionalität. *Services als vermarktbare IT-Produkte* fördern die Einführung marktähnlicher Mechanismen in der Zusammenarbeit zwischen IT und Fachbereichen sowie und die Umsetzung innovativer Beschaffungsmodelle für IT-Leistungen. Für die drei skizzierten Einsatzbereiche wurden die wesentlichen Architekturentscheidungen abgeleitet, die eine Basis für einen technischen, fachlichen und organisatorischen SOA-Architekturrahmen bilden.

Die in diesem Beitrag identifizierten Einsatzbereiche von SOA stellen erste, aus Einzelfällen abgeleitete Erkenntnisse dar und bedürfen einer weitergehenden Vertiefung und empirischen Evaluation. Während standardisierte Integrationsinfrastrukturen bereits Thema vieler Arbeiten zu Enterprise Application Integration ist, besteht Forschungsbedarf insbesondere bezüglich der fachlichen Modularisierung der Applikationslandschaft und Domänenbildung. Die Fallbeispiele zeigen, dass hier eine von den fachlichen Anforderungen ausgehende, geschäfts- und prozessorientierte Architekturentwicklung und Identifikation von Services notwendig ist, um die Zielsetzung der Wiederverwendung zu erfüllen. Gleichzeitig können Services nicht auf der grünen Wiese entworfen werden, sondern müssen die bestehende Applikationsarchitektur berücksichtigen. Ein Vorgehen zur Entwicklung von SOA sollte deshalb auf bewährten methodischen Ansätzen aus den Bereichen der Unternehmensmodellierung, dem Business Engineering, der Enterprise Application Integration und objekt- bzw. komponentenorientierte Systementwicklung aufbauen und diese erweitern.

© HSG / IWI / CC BN3 / 2

Literatur

[Bagh05] Baghdadi, Y.: A business model for deploying Web services: A data-centric approach based on factual dependencies. In: Information Systems & E-Business Management 3 (2005) 2, S. 151-173

[Bake02] Baker, S.: Web Services and CORBA. In: Meersman, R., Tari, Z. (Hrsg.): On the Move to Meaningful Internet Systems 2002. Proceedings of Confederated International Conferences CoopIS, DOA, and ODBASE 2002, 2002, S. 618-632.

[BCHP05] Baskerville, R., Cavallari, M., Hjort-Madsen, K., Pries-Heje, J., Sorrentino, M., Virili, F.: Extensible architectures: The strategic value of service-oriented architecture in banking. In: Bartmann, D., Rajola, F., Kallinikos, J., Avison, D., Winter, R., Ein-Dor, P., Becker, J., Bodendorf, F., Weinhardt, C. (Hrsg.): Proceedings of the Thirteenth European Conference on Information Systems 2005, Regensburg.

[CKMT03] Curbera, F., Khalaf, R., Mukhi, N., Tai, S., Weerawarana, S.: The next step in Web Services. In: Communications of the ACM 46 (2003) 10, S. 29-34.

[EAAC04] Endrei, M., Ang, J., Arsanjani, A., Chua, S., Comte, P., Krogdahl, P., Luo, M., Newling, T.: Patterns: Service-Oriented Architecture and Web Services. IBM Redbook, 2004.

[HaMS03] Hansen, M., Madnick, S., Siegel, M.: Data Integration Using Web Services. In: Bressan et al. (Hrsg.): Efficiency and Effectiveness of XML Tools and Techniques and Data Integration over the Web, VLDB 2002 Workshop EEXTT and CAiSE 2002. Workshop DIWeb, 2003, S. 165-182.

[HaSc02] Hars, A., Schlüter-Langdon, C.: Chancen und Risiken für verteilte Informationssysteme. In: Information Management & Consulting 17 (2002) 3, S. 13-19.

[HeBK04] Herr, M., Bath, U., Koschel, A.: Implementation of a Service Oriented Architecture at Deutsche Post MAIL. In: Zhang, L., Jeckle, M. (Hrsg.): Proceedings of the European Conference on Web Services, Springer, Berlin, Heidelberg, 2004, S. 227-238.

© HSG / IWI / CC BN3 / 2

[Heut03] Heutschi, R.: Fallstudie: Elektronischer Datenaustausch in der Schweizer Haustechnikbranche. BE HSG / BECS / 17, Institut für Wirtschaftsinformatik, Universität St. Gallen, St. Gallen 2003.

[HeSc05] Heutschi, R., Schemm, J. W.: Fallstudie: Service-orientierte Architektur bei Deutsche Post Brief. BE HSG / CCBN3 / 3, Institut für Wirtschaftsinformatik, Universität St. Gallen, St. Gallen 2005.

[HoRW04] Homann, U., Rill, M., Wimmer, A.: Flexible Value Structures in Banking. In: Communications of the ACM 47 (2004) 5, S. 34-36.

[Kern02] Kern, T., Willcocks, L. P., Lacity, M. C.: Application service provision: Risk assessment and mitigation. In: MIS Quarterly Executive 1 (2002) 2, S. 113-126.

[KoLe04] Kossmann, D., Leymann, F.: Web Services. In: Informatik-Spektrum 27 (2004) 2, S. 117-128.

[NeLo04] Newcomer, E., Lomow, G.: Understanding SOA with Web Services. Addison-Wesley, Maryland 2004.

[ÖsRe03] Österle, H., Reichmayr, C.: Out-tasking mit WebServices. In: Bullinger, H.-J., Scheer, A.-W. (Hrsg.): Service Engineering - Entwicklung und Gestaltung innovativer Dienstleistungen. Springer, Berlin 2003, S. 565-589.

[PaGe03] Papazoglou, M.; Georgakopoulos, D.: Service-Oriented Computing. In: Communications Of The ACM 46 (2003) 10, S. 25-28.

[QuWi03] Quantz, J., Wichmann, T.: Basisreport Integration mit Web Services. Berlecon Research. Berlin 2003.

[Uhl04] Uhl, J.: "Unternehmensarchitekturen" ist ein Dauerthema - aber die Ziele bzw. Motivation und damit Schwerpunkte ändern sich, vor allem mit wirtschaftlichen Randbedingungen. In: Wirtschaftsinformatik, 46 (2004) 4, S. 317

[W3C04] W3C: Web Services Architecture. W3C Working Group Note 11 February 2004, http://www.w3.org/TR/2004/NOTE-ws-arch-20040211/, 26.06.2004.

[Wilk03] Wilkes, L., SOA - Save Our Assets, in: CBDi Journal, November, 2003, http://www.cbdiforum.com/secure/interact/2003-11/soa_save_our_assets.php3, 21.04.2005.

© HSG / IWI / CC BN3 / 2

[Wood04] Woods, D.: Enterprise Service Architecture: SAPs Bauplan für Geschäftsapplikationen der nächsten Generation. Galileo Press, Bonn 2004.

[Zarn04] Zarnekow, R.: Produktorientiertes Informationsmanagement. In: Zarnekow, R., Brenner, W., Grohmann, H. (Hrsg.): Informationsmanagement - Konzepte und Strategien für die Praxis. dpunkt Verlag, Heidelberg 2004, S. 41-56.

[ZMCO04] Zimmermann, O., Milinski, S., Craes, M., Oellermann, F.: Second Generation Web Services- Oriented Architecture in Production in the Finance Industry. In: Companion to the 19th Annual ACM SIGPLAN Conference on Object-Orie Programming, Systems, Languages, and Applications, OOPSLA 2004. Vancouver, 2004, S. 283- 289

© HSG / IWI / CC BN3 / 2