

Nils Tegtmeier/Bernhard Gehra/Heinz Möllenkamp/Christoph Künne

Serviceorientierte IT-Architekturen – Kritische Bestandsaufnahme und Herausforderungen für das Controlling

Autoren



Nils Tegtmeier

ist Chief IT Architect in der Praxisgruppe Insurance und Topic Expert für Enterprise Architecture Management bei Platinion, einer 100 %igen Tochter der Boston Consulting Group.



Dr. Bernhard Gehra

ist Projektleiter bei der Boston Consulting Group und Mitglied der Praxisgruppen Financial Institutions und Information Technology.



Heinz Möllenkamp

ist Partner und Managing Director bei der Boston Consulting Group und Mitglied der Praxisgruppen Financial Institutions und Information Technology. Er leitet europaweit den Bereich „IT/Operations in Banking“.



Christoph Künne

ist Berater bei der Boston Consulting Group und Mitglied der Praxisgruppen Technology, Media & Telecommunications und Information Technology.

Motivation

Die flexible und schnelle Anpassung der Informationssystemarchitektur an sich ändernde Geschäftsanforderungen ist seit Jahren eine wesentliche Herausforderung für das IT-Management in Unternehmen. Das Konzept der so genannten serviceorientierten Architekturen (SOA) bietet IT-Verantwortlichen und Fachbereichen zahlreiche Ansatzpunkte, diese Herausforderungen zu meistern. Daraus ergeben sich auch für den Controller neue Möglichkeiten und Nutzenpotenziale, z. B. die effiziente Steuerung von Investitionen in eine SOA-Einführung oder die Etablierung einer Prozesskostenrechnung.

Nach einem anfänglichen „SOA-Hype“ zur Jahrtausendwende wurden die in SOA gesetzten Erwartungen in den folgenden Jahren teilweise erfüllt, aber auch vielfältig überdehnt. Angesichts der Schlagwortfülle der mit SOA oft verbundenen Konzepte und Technologien herrscht bei vielen Entscheidern Unsicherheit, wofür SOA ursächlich steht – und was das tatsächlich Neue daran ist. Groß angelegte SOA-Programme scheinen nicht das erhoffte „business alignment“ zu erzielen. Andere Pilot-Projekte weisen darauf hin, dass der Aufwand für die Einführung der Serviceorientierung in der IT-Architektur häufig höher ist als erwartet. Das zeigen auch die Diskussionen mit den CIOs im Rahmen der IT-Kosten-Benchmarking-Studien, die die Boston Consulting Group (BCG) regelmäßig in Industrien wie Banken, Versicherungen und Pharma durchführt (vgl. Möllenkamp/Freudenstein 2009, Heydorn/Rejda 2009, Cummins 2005).

Aktuelle Marktanalysen zeigen, dass die Begeisterung über serviceorientierte Architekturen deutlich abflaut. Der Gartner Group zufolge (vgl. Sholler 2008) planen nur 25 % aller 2008 befragten CIOs den Einstieg in SOA (53 % in 2007). 16 % aller Befragten erklären, bewusst keine SOA-

initiativen zu verfolgen (7 % in 2007). Auch bei zahlreichen anderen technologischen Konzepten und Technologien, z. B. Web 2.0, Digital Paper oder RFID, ließen sich in der Vergangenheit ähnliche Entwicklungs-

- Management und Controlling im Unternehmen sind häufig mit neuen Technologien zur Unterstützung von Aufgaben und Abläufen konfrontiert. Eine der Unterstützungsmöglichkeiten sind serviceorientierte Architekturen (SOA). Dieser Beitrag soll das Verständnis von SOA schärfen und z. B. anhand einer Checkliste Praxisbezug für den Controller herstellen.
- SOA bezeichnet ein Architekturparadigma im Bereich von verteilten Systemen. Es vereint existierende Paradigmen, neuere technologische Standards und Technologien.
- Das Konzept der Serviceorientierung bei der Gestaltung der IT-Anwendungslandschaft ist nicht neu. Jedoch ermöglichen erst technische Entwicklungen der letzten Jahrzehnte die Durchsetzbarkeit der SOA-Prinzipien. Nach einem „SOA-Hype“ findet jetzt ein sachlicher Umgang mit dem Thema statt.
- Die Umsetzung von SOA-Implementierungen erfolgt idealerweise in klar abgegrenzten Pilot-Projekten in enger Abstimmung zwischen Fach- und IT-Bereich.
- Auch für den Controller entstehen durch eine SOA-Einführung klare Nutzenpotenziale. Mit einer soliden Kosten-Nutzen-Analyse und einem fundierten Kennzahlenwerk lassen sich langfristig die Erfolge und Investitionen steuern.
- Die Umsetzung der SOA-Prinzipien über z. B. einen Enterprise Service Bus (ESB) ist weitaus mehr als eine einfache Softwareeinführung. Der Erfolg einer SOA-Initiative steht und fällt mit der Einbindung in die übergreifende IT-Strategie. SOA wird in Zukunft weiterhin ein wichtiges und ökonomisch relevantes Thema für das Management bleiben.

muster erkennen. Nach einer anfänglichen Phase überhöhter Begeisterung und übertriebener Erwartungen fand relativ schnell eine sachliche Eingrenzung auf ein realistisches Nutzenpotenzial statt. Oftmals ist das die notwendige Grundlage und der Startpunkt für die Entwicklung nachhaltiger wirtschaftlicher Anwendungsmöglichkeiten.

Vor diesem Hintergrund soll der vorliegende Beitrag den aktuellen Status von SOA beleuchten. Ergänzt durch Fallbeispiele und Implementierungserfahrung aus der Beratungspraxis werden weitergehende Nutzenpotenziale für Controlling und Management diskutiert. Im Folgenden wird zunächst eine Definition und Beschreibung der Prinzipien serviceorientierter Architektur gegeben, um SOA abzugrenzen und einzuordnen. Daraufhin werden Praxisbeispiele vorgestellt und Nutzenpotenziale fürs Controlling aufgezeigt.

Definition und Erläuterung

Definition

SOA beschreibt ein Architekturparadigma im Bereich von verteilten Systemen. Es stellt Regeln für die Gestaltung einer modularen IT-Architektur auf, um die Auf-

gaben – technisch auch ‚Dienste‘ genannt – von Mitarbeitern und Abteilungen besser zu strukturieren und zu nutzen.

Historie

Der Gedanke der Serviceorientierung ist seit der ersten Benennung dienstorientierter Architekturen nicht neu. SOA entwickelt vielmehr existierende Paradigmen (z. B. Enterprise Application Integration) durch Kombination neuer technologischer Standards (z. B. Web Services) und Technologien (z. B. XML) weiter (Abbildung 1). Für eine SOA-Implementierung gibt es keine typische Blaupause. Eine serviceorientierte IT-Architektur muss immer mittels einer Auswahl aus der Fülle bestehender technologischer Bausteine auf den spezifischen Anforderungskontext eines Unternehmens zugeschnitten werden.

Grundkonzept

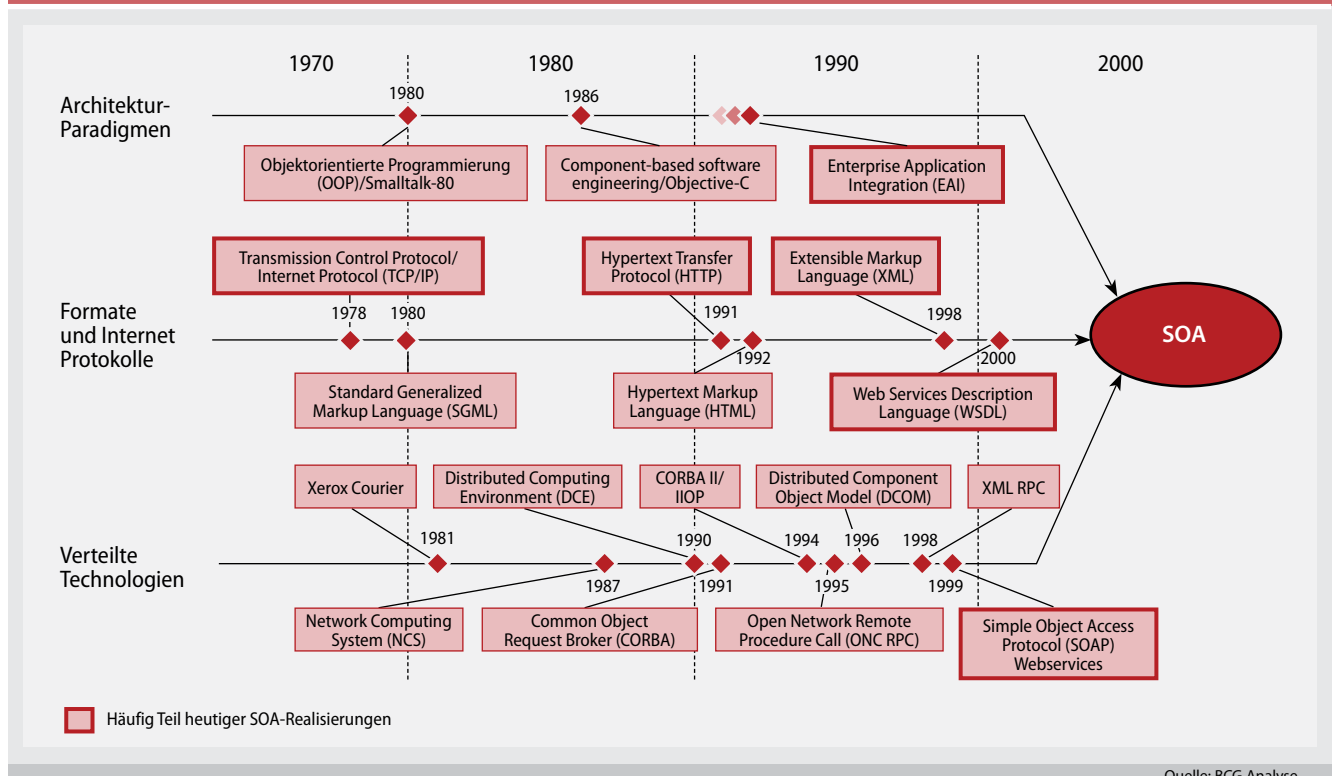
SOA fußt auf der Neuausrichtung und Zerlegung der IT-Architektur in kompatible, kleine Module – so genannte Dienste oder „Services“. Eine besondere Rolle spielt dabei die Orientierung entlang der Geschäftsprozesse des Unternehmens (vgl. Pulier/Taylor 2005). Dadurch soll die zukünftige Anwendungsentwicklung und -integration dahingehend vereinfacht wer-

den, dass die Gesamtheit aller technischen IT-Leistungen, die z. B. von Anwendungen, Datenbanken und Servern erbracht werden, hinter Diensten mit standardisierten Schnittstellen gekapselt werden. Weitergehend können durch das modulare Zusammenstellen – einer so genannten „Orchestrierung“ – verschiedener Dienste niedriger Abstraktionsebene höherwertige Dienste zur Verfügung gestellt werden. Im Rahmen eines solchen Entwicklungsprozesses verändert sich die Arbeitsweise zwischen IT- und Fachbereichen insofern, dass neue Anwendungen durch die Auswahl und Orchestrierung vorhandener Dienste erstellt werden können (Abbildung 2).

Beispiel

Ein konkretes Beispiel verdeutlicht dieses Prinzip: die Bearbeitung einer Kundenbestellung in einem Versandhaus wird üblicherweise mehrere Teilprozesse erfordern: Bestelleingang – Bonitätsprüfung Kunde – Prüfung Lagerbestand – Warenversand – Rechnungsversand – Inkasso. Jeder Teilprozess kann mit einem Service bedient werden. Einzelne Services lassen sich für andere Zwecke wiederverwenden, z. B. „Prüfung Lagerbestand“ für die Produktionsplanung oder das Marketingcontrol-

Abb. 1 | Verbindung zahlreicher technischer Konzepte durch SOA



Quelle: BCG Analyse

ling. Darüber hinaus können Services auch herausgelöst und durch externe Anbieter ersetzt werden. Für „Bonitätsprüfung Kunde“ und „Inkasso“ dürften in der Regel kompetente externe Anbieter bestehen. Umgekehrt kann aber ein Unternehmen aber auch jene eigenen Services extern am Markt anbieten, in denen es einen besonderen Wettbewerbsvorteil besitzt. Im Falle des Versandhauses kann dies der Warenversand sein. Man könnte auch sagen: das Konzept der „Deonstruktion der Wertschöpfungskette“ wird durch SOA auf den Bereich der IT-Anwendungen übertragen.

Was ist ein Service?

Ausgehend vom oben genannten Beispiel ist ein Service grundsätzlich definiert durch einen Dienstanbieter, einen Dienstnehmer, einen Dienstleistungsvertrag zwischen diesen beiden, eine vereinbarte Kommunikationsschnittstelle und eine definierte Ein- und Ausgabe. Weiterhin gibt es drei wesentliche Anforderungen, die einen Dienst ausmachen: (i) die Funktionalität, die sich ausschließlich an betrieblichen Prozessen und nicht an technischen Details ausrichtet, (ii) die Kapselung der technischen Implementierung und genutzten Technologie zur Sicherstellung einer stabilen Schnittstelle und (iii) eine Dienstgütevereinbarung

(Service Level Agreement) zur Definition von Leistungsqualität und -umfang. Das Konzept der Services ist zunächst nicht grundlegend neu. Inwiefern es tatsächlich in der Praxis umgesetzt werden kann, wird später in diesem Beitrag diskutiert.

Prinzipien

Um eine größtmögliche Flexibilität der IT-Architektur zu erreichen, folgt SOA dem Prinzip der losen Kopplung zwischen interagierenden Diensten. „Loose coupling“ bedeutet die Reduzierung künstlicher Abhängigkeiten. Gemeint sind hiermit all jene Abhängigkeiten, welche auf unterschiedlichen Standards und Anforderungen an Datenaustausch beruhen. Ein häufig genutztes Beispiel zur Nutzung eines Laptops auf Reisen kann dies bildhaft verdeutlichen: Es besteht eine unvermeidliche Abhängigkeit des Laptops vom Stromnetz. Der „Service“ Laptop bedient sich des „Service“ Stromversorgung. Probleme bereiten auf Reisen künstliche und prinzipiell vermeidbare Abhängigkeiten wie unterschiedliche Steckerformen oder Netzspannungen.

Des Weiteren wird die Flexibilität der IT-Architektur durch die Granularität der Services beeinflusst. Die Services werden so auf Teilgeschäftsprozesse heruntergebrochen,

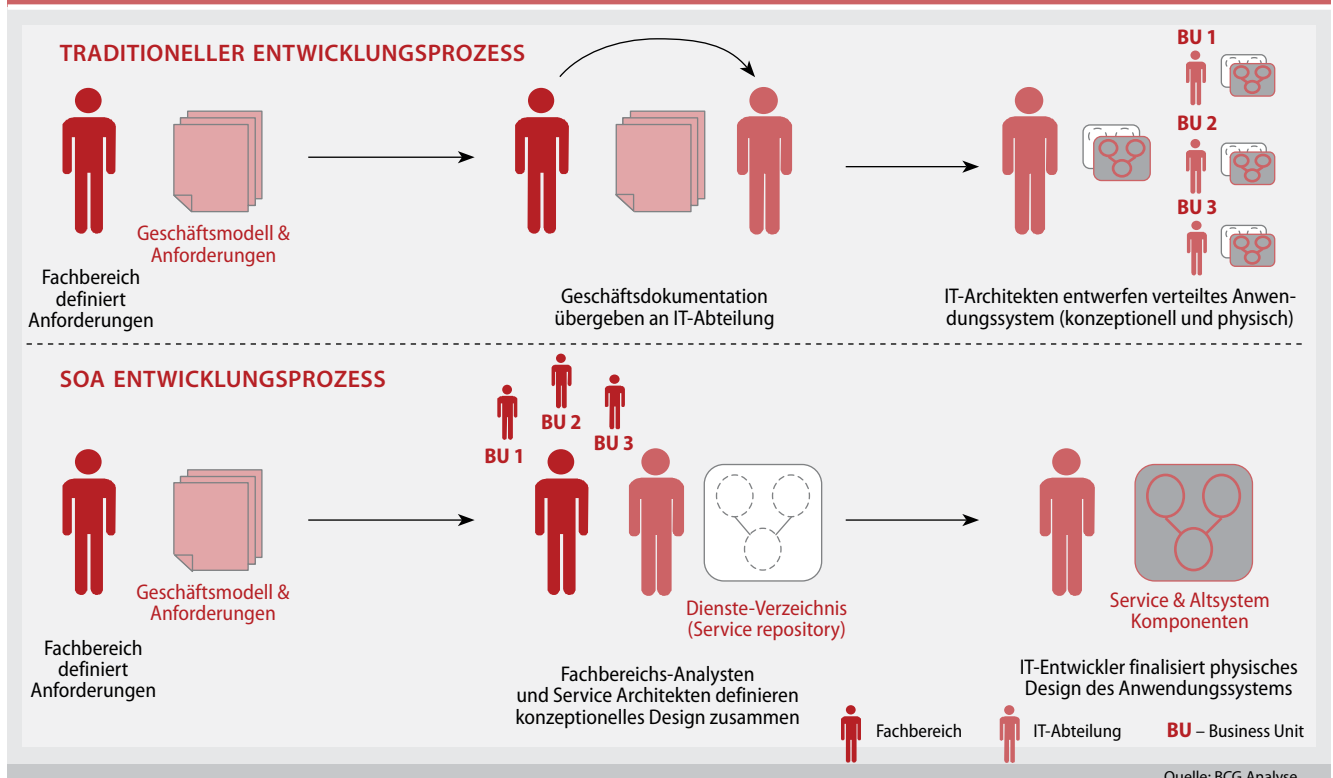
dass ein geringes Abstraktionsniveau erreicht wird. Die Wiederverwendbarkeit einzelner Services in anderen Anwendungs- oder Unternehmensteilen steigt mit der feineren Granularität. Das Prinzip der Wiederverwendung zielt darauf ab, die jeweiligen Funktionen innerhalb einer SOA nur einmal zu implementieren. Services lassen sich mit Lego-Bausteinen vergleichen, welche gerade deshalb so vielfältig kombinierbar sind, weil jeder einzelne Stein klein und von elementarer Funktion ist.

Schließlich wird durch die Modularisierung von Funktionalitäten aus zuvor oftmals monolithischen Anwendungen und durch die Interoperabilität zwischen den Diensten die Komplexität der IT-Architektur deutlich reduziert (**Abbildung 3**).

Reality Check

Je länger die Fachdiskussion andauert, desto häufiger sieht man die These vertreten, SOA sei nur eine Frage guter Architektur-Prinzipien oder der richtigen „IT-Philosophie“. Diese Position verkennt jedoch die grundlegende Bedeutung, die neuere technische Entwicklungen für die Durchsetzbarkeit dieser Prinzipien haben. Erst die Offenheit von Transportprotokollen wie HTTP (Hypertext Transfer Protocol) hat den Aufbau hinreichend robuster verteilter Systeme

Abb. 2 | Unterschiede zwischen traditionellem und SOA-Entwicklungsprozess



unter gleichberechtigten Partnern ermöglicht. Auch die weite Verbreitung von URLs (Uniform Resource Locator) zur Adressierung hat das Auffinden von Services im Netzwerk hinreichend effizient gemacht. Weiterhin ist durch Sprachen wie XML (Extensible Markup Language, Sprache zum Beschreiben und Austauschen strukturierter Daten) die Formulierung selbstbeschreibender Schnittstellen-Definitionen hinreichend komfortabel geworden. All dies sind Treiber für „loose coupling“ und Wiederverwendung. Schließlich erlauben erst moderne Integrationsplattformen dem Entwickler, sich auf das „business alignment“ statt auf technische Details der Systemkommunikation zu konzentrieren. Dementsprechend ist die Transformation einer Anwendungslandschaft Richtung SOA nie nur eine Frage von „best practices“ in der Anwendungsentwicklung, sondern immer auch mit technologischer Erneuerung verbunden. Ein sorgfältig modularisiertes Mainframe-Transaktionssystem mit Anbindung externer Partner über X.400 (Standard für elektronischen Datenaustausch) aus den frühen 80er-Jahren mag heute noch erfolgreich wart- und erweiterbar sein – serviceorientiert ist es dadurch noch nicht.

Betriebliche Anwendung

Implementierungsansätze

In unserer Beratungspraxis beobachten wir drei typische Herangehensweisen von Unternehmen an den Einsatz von SOA-Technologien.

IT-getriebene Initiativen finden ihren Anlass häufig in der Verfügbarkeit neuer Technologien, die im Rahmen regulärer Upgrades bereits eingesetzter technischer Plattformen Einzug in die Systemlandschaft halten. Aus der Motivation heraus, diese neuen Möglichkeiten für sich zu erschließen, beginnen die Systemverantwortlichen auf der IT-Seite, Fachanforderungen ohne unmittelbare Einbeziehung der Fachseite mit neuen Technologien umzusetzen. Diese explorative Nutzung folgt jedoch selten einer stringenten IT-Strategie und bleibt in der Regel Stückwerk mit hohen Wartungsfolgekosten.

Demgegenüber stehen häufig fachseitig getriebene Initiativen im Kontext eines großangelegten Programms zur Organisations- und Prozessoptimierung. Die Annahme, dass mit der konzeptionellen Neuordnung der Löwenanteil der Transformation schon geleistet sei, führt hier oft zu überambitionierten Fahrplänen für die technische Transformation der Anwendungslandschaft. Konfrontiert mit den komplexen Details von System-Abhängigkeiten und Implementierungs-Altlasten verlaufen viele dieser Programme zugunsten einer Rückkehr zum Tagesgeschäft im Sande.

Von der IT initiierte, gemeinschaftliche Initiativen mit der Fachseite ergeben sich häufig aus dem Ziel, gezielt einzelne bestehende Abläufe unter Nutzung neuer Technologien zu optimieren. Statt technologischer und organisatorischer Neuordnung um ihrer selbst willen wird hier eine stärker evolutionäre, schrittweise Transformation der Anwendungslandschaft in Gang gesetzt.

Für eine solche kontrollierte Transformation der Anwendungslandschaft hin zur Strukturierung durch Services sind Pilot-Initiativen empfehlenswert, die einen technischen Anlass mit Auswirkungen auf fachseitige Abläufe haben, die eine konkrete, messbare Zielsetzung verfolgen und deren Umfang sich klar abgrenzen lässt.

Fallbeispiel: Versicherungsgruppe Ausgangssituation und Ziel

Im nachfolgenden Fallbeispiel treffen diese drei Faktoren aufeinander. Der Kunde, Erstversicherer für Privatkunden, hatte sich im Rahmen einer Neuausrichtung der Service-Organisation für die Einführung eines Customer Relationship Management (CRM) Systems entschieden. Neben der Bereitstellung einer spartenübergreifenden 360-Grad-Sicht auf den Versicherten sollte die Plattform auch den Einstieg in die Bearbeitung von Kundenanfragen bilden. Für diesen Zweck war eine transaktionale Integration mit den nach wie vor spartenorientierten Bestandssystemen erforderlich. Zielsetzung aus Geschäftssicht war eine Steigerung der „Dunkelverarbeitungsquote“, d. h. der fallabschließenden Behandlung nach Erstkontakt im Kunden-Service-Center. Die erste Ausbaustufe beschränkte sich dabei auf die stärkere Automatisierung der Annahme von Schaden-Meldungen. Der Ablauf enthielt eine Reihe von Prüf- und Entscheidungsschritten, die das neue System unterhalb von Schwellenwerten eigenständig durchlief; oberhalb der Schwellenwerte war eine Aussteuerung zur manuellen Prüfung vorgesehen.

Abb. 3 | SOA-Hebel und Nutzenpotenziale

SOA-Hebel \ Nutzen		Wiederverwendung	Entkopplung	Reduzierte Komplexität
		Flexibilität	Einfache Rekombinier- und Anpassbarkeit existierender Funktionalität	Einfacher Austausch von bestehenden Komponenten
Time-to-market	Rückgriff auf bestehende Services	Applikationen und Services unabhängig voneinander implementierbar	Reduzierte Anzahl anzupassender Komponenten	
Effizienz	Reduzierter Entwicklungs-, Test- und Betriebsaufwand	Geringerer Abstimmungsbedarf	Weniger Komponenten in Wartung und Betrieb	
Qualität	Weniger Fehler durch Rückgriff auf erprobte Funktionen	Begünstigung von lokalen Tests und Testautomatisierung	Höhere Transparenz, weniger „Seiteneffekte“	
Governance	Schlankere Prozesse	Vereinfachtes Outsourcing	Klar abgrenzbare Zuständigkeiten und Kostentransparenz	

Quelle: Platinion Analyse

Lösung

Gegenstand des Projekts war die Verprobung einer technischen Integrations-Architektur auf Basis eines Enterprise Service Bus (ESB) und einer Business Process Engine (BPE) (Abbildung 4). Aufgabe der BPE sollte sein, den Prozess der Schaden-Erfassung und die Weitergabe statistischer Schadeninformationen an den Verband der Schadenversicherer abzubilden. Aufgabe des ESB war es, Nachrichtenformate zwischen den Systemen zu vereinheitlichen und die Infrastruktur für Service-Aufrufe mit unmittelbarer und zeitversetzter Antwort bereitzustellen. Der Ablauf blieb hinreichend einfach, um sich für die Pilotimplementierung zu eignen, enthielt jedoch mit der automatisierten Weitergabe von Daten in Echtzeit bereits einen echten Nutzungsvorteil.

Ergebnis

Das hier skizzierte Szenario realisierte und evaluierte ein Team aus vier Entwicklern innerhalb von fünf Wochen. Dabei konzentrierte es sich auf die Definition rahmengerbender Design-Prinzipien und die Ausimplementierung nur der benötigten Zugriffe. Die umgesetzte Lösung hätte nach Hochrechnungen aus Projektergebnissen auch die geschätzten Transaktionsvolumina bei Schaden-Großereignissen (rund 970.000 Messages/Stunde) mit einer schmal dimensionierten Server-Infrastruktur bewältigt.

Dennoch entschied sich der Kunde gegen eine Übernahme der Projektlösung in den Produktionsbetrieb. Wiewohl die Implementierung insgesamt als erfolgreich gelten konnte, kristallisierte sich im Projektverlauf die Erkenntnis zweier wesentlicher Hindernisse für einen kurzfristigen Ausbau der verprobten Konzepte heraus: Zum einen war die Entwicklungsmannschaft auf Kundenseite kaum mit den grundlegenden Konzepten des Architektur-Entwurfs vertraut. Zum zweiten zeigten die eingesetzten Produkte einen deutlich geringeren technischen Reifegrad als erwartet. Fehlendes internes Know-how ließ sich für den begrenzten Zeitraum des Projekts durch Einsatz externer Ressourcen ausgleichen. Ein flächendeckender Einsatz der neuen Technologien ohne vorherige Investitionen in die interne Mitarbeiterqualifikation wäre jedoch nicht empfehlenswert gewesen. Die letzten Endes nur mit größerem Aufwand behebbare oder umgehbarere technische Probleme machten insbesondere deutlich, dass es sich ohne entsprechende Kompetenz nicht lohnt, im SOA-Umfeld zu den „early adopters“ zu gehören. Mehrere andere Pro-

Abb. 4 | Zentrale Elemente einer SOA-Plattform-Infrastruktur

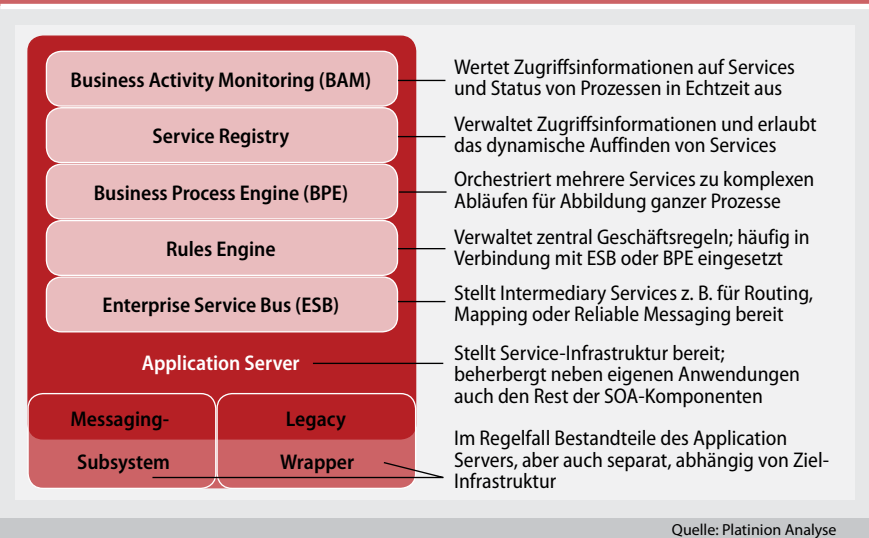
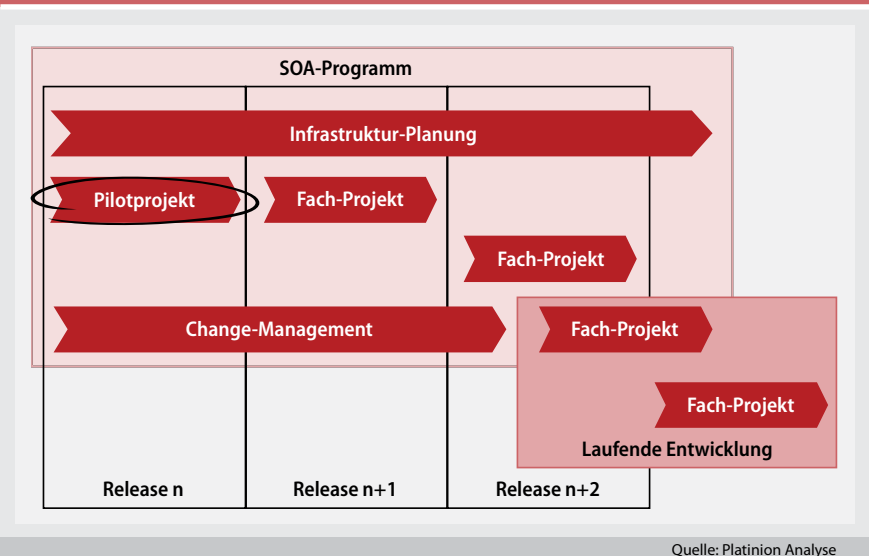


Abb. 5 | Mittelfristiger Umsetzungsfahrplan



jekte haben gezeigt, dass dies weitgehend herstellerübergreifend gilt.

Fazit

Trotz oder gerade wegen der Entscheidung gegen eine Weiterverfolgung des Ansatzes zeigt das Beispiel den Wert von Pilot-Vorhaben mit reduziertem Fokus und klaren Ausgangskriterien (Abbildung 5). Ohne die notwendige Reflektion kann auch SOA kein Bestandteil einer umfassenden IT-Strategie sein.

Fallbeispiel: Europäischer Mobilfunkbetreiber

Ein weiteres Fallbeispiel demonstriert ebenso die enge Verzahnung von technologischer Umsetzung und geschäftlicher Nut-

zenperspektive. Es zeigt einen Ansatz zur schrittweisen Transformation der Anwendungslandschaft.

Ausgangssituation

Der Kunde, ein europäischer Mobilfunkbetreiber, besitzt eine stark historisch gewachsene IT-Architektur. Diese ist gekennzeichnet durch enorme Komplexität der hohen Anzahl verschiedener Systeme und Schnittstellen, hohe Einmalkosten bei Tests, Datenmigration und Integrationsprojekten, und zuletzt eine schwache IT-Governance, die Konsolidierungsversuche nicht nachhaltig zu Ende führt. In besonderem Fokus lag der Bereich Customer Service. Kundendaten waren über mehr als

50 verschiedene Systeme fragmentiert. Der Anruf eines Mobilfunkkunden führte dazu, dass sich der jeweilige Call Center Agent in mehreren Anwendungen anmelden musste, um die verschiedenen Workflow-Schritte abzuarbeiten. Mangelndes Stammdaten-Management und unzureichende Verfügbarkeit dispositiver Informationssysteme erschwerten übergreifende Gesamtansichten bei der Bearbeitung der Kundenanfrage.

Ziel

Gegenstand des Projektes war ein 3-Jahresplan zur technologischen Entwicklung mit dem Ziel der Definition einer technischen Ausgangsplattform, die die zukünftige schrittweise Vereinfachung der Anwendungslandschaft ermöglicht. Im Bereich des Customer Service war es das Ziel, die Customer Care Transaktionen durch die Integration von Schlüsselprozessen und Kundendaten zu optimieren. Damit sollten Prozessdurchlaufzeit und Kosten signifikant gesenkt werden.

Lösung

Der Lösungsweg gestaltete sich in zwei Schritten. Zuerst wurden alle Kundentransaktionen nach der Häufigkeit ihrer Verwen-

dung analysiert. Die Liste der Top-Transaktionen (z. B. Rechnungsprüfung, Tarifwechsel, Handy-Wechsel) identifizierte sieben Anwendungen, die durch eine priorisierte Integration besondere Effizienzvorteile mit sich bringen. In einem zweiten Schritt wurde ein Enterprise Service Bus (ESB) aufgebaut, der die für die Top-Transaktionen erforderlichen Dienste von der Anwendungsschicht entkoppelt. Auf Anwenderseite wurden integrierte Übersichtsbildschirmmasken entwickelt, die den gesamten Prozessdurchlauf (z. B. durch vereinfachte Anmeldeverfahren) verkürzt und ausgewählte Prozessschritte (z. B. Kundenauthentifizierung) gänzlich automatisiert haben (**Abbildung 6**).

Der eingesetzte Enterprise Service Bus ist die geeignete Plattform für zukünftige Aufgaben. Zum einen können Dank einfacher Skalierbarkeit des ESB Services auch via Web oder interaktiver Telefonsprachsteuerung verfügbar gemacht werden und zum anderen ist über den ESB die schrittweise Integration der verbleibenden Systeme möglich, die zu einer weiteren Vereinfachung der Anwendungslandschaft führt.

Ergebnis

Die selektive Integration der am häufigsten genutzten Kundentransaktionen hat die

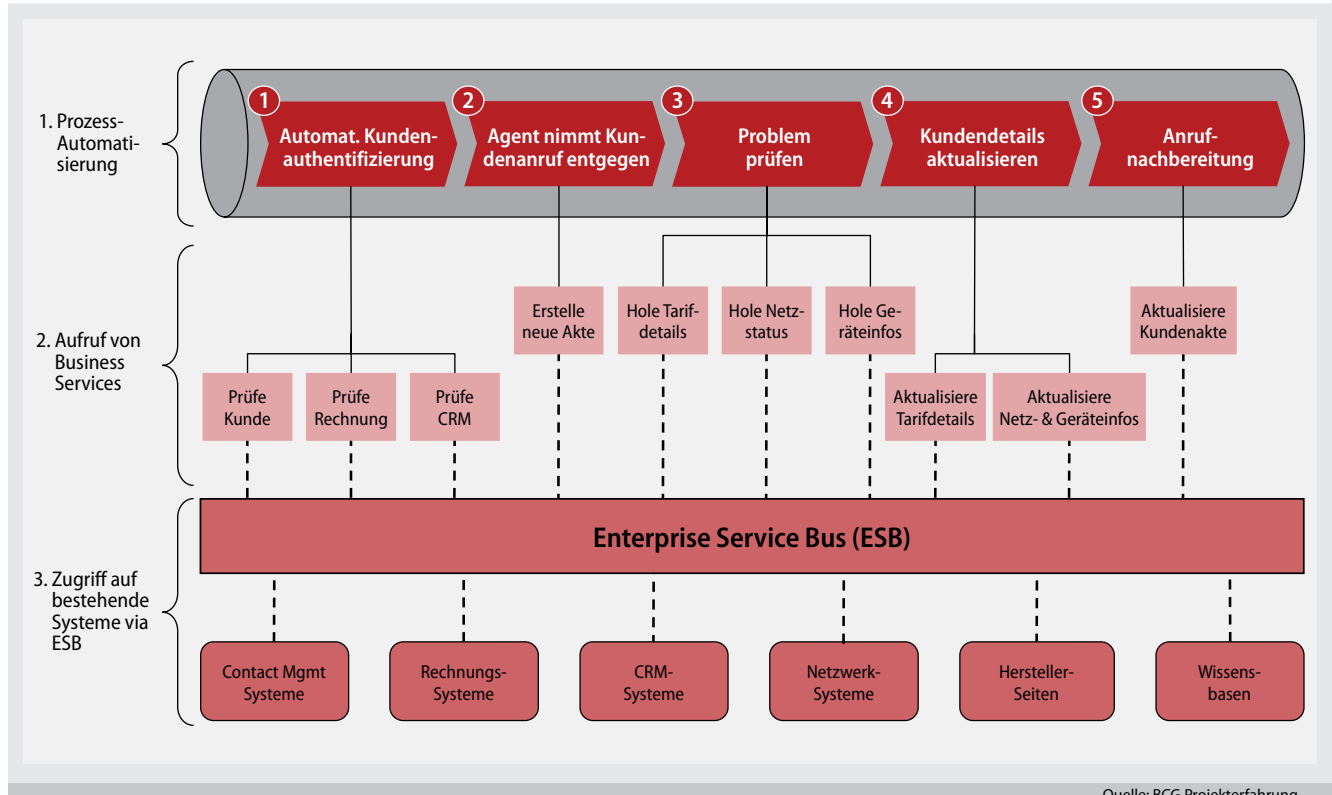
Bearbeitungseffizienz im Durchschnitt um 33 % bei allen Anrufen gesteigert. Insgesamt wurden die vorher 19 verschiedenen Bildschirmmasken auf fünf reduziert. Des Weiteren wurde z. B. die Dauer des Prozessschritts ‚Anrufnachbereitung‘ von 21 Sekunden auf 13 Sekunden verringert, was mittelfristig eine Einsparung von zirka 3 Mio. Euro ermöglicht.

Fazit

Zusammenfassend zeigt dieses Fallbeispiel gut, wie die Einführung eines ESB als zusätzliche Abstraktionsschicht zeitliche und monetäre Einsparpotenziale für den Anwender ermöglicht. Aus fachlicher Perspektive wird neuer Nutzen realisiert und Vorgänge, die wertschöpfend sind, werden schneller abgearbeitet. Aus technologischer Perspektive hat sich beim behandelten Fallbeispiel auf Ebene der transaktionalen Systeme allerdings wenig verändert.

Der Aufwand bei der Einführung eines ESB im Sinne einer vollständigen SOA-Integration kann nicht vernachlässigt werden (z. B. Anbindung erstellen, Legacy-Schnittstellen implementieren). Ein wesentlicher Punkt für das erfolgreiche Vorantreiben der Architekturuordnung ist die technologische Transformation der unterliegen-

Abb. 6 | Enterprise Service Bus am Fallbeispiel Mobilfunkanbieter



Quelle: BCG Projekterfahrung

den Systeme, die sich langfristig an das erste Aufsetzen eines Enterprise Service Bus anschließen muss.

Implikationen fürs Controlling

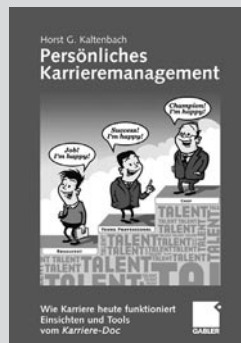
Controlling einer SOA-Einführung im Unternehmen

Ein wesentlicher Grund für die Unzufriedenheit vieler Unternehmen mit dem Erfolg ihrer SOA-Initiativen ist die konzeptionelle Schwäche ihrer Erfolgsmessung: Der Nutzen architektonischer Grundlagenarbeiten ist nur für den ohnehin Überzeugten selbst-evident; die Kosten der angeschafften Plattform dagegen schlagen unangenehm deutlich zu Buche. Für eine solide Evaluation ist es deshalb wichtig, sich bereits im Vorfeld Gedanken über ein geeignetes Kennzahlenwerk zur Erfolgsmessung zu machen. Dieses Kennzahlenwerk lässt sich in fünf Teilbereiche gliedern:

1. Der Fortschritt der SOA-Einführung bemisst sich nicht an der Anzahl der realisierten Services. Wichtigste Parameter für den wahren Erfolg der Transformation sind stattdessen die Stabilität der Service-Definitionen (d. h. abnehmende Änderungshäufigkeit) sowie der Nutzungs- bzw. Wiederverwendungsgrad. Beide dienen als Indikatoren dafür, dass der Architekt beim Service-Zuschnitt die richtige Detailebene getroffen hat. Der statische Nutzungsgrad eines Service misst die Anzahl der Systeme, die gemäß Architektur-Dokumentation auf ihn zurückgreifen. Er lässt sich nur für nicht-öffentliche Services erfassen. Der dynamische Nutzungsgrad erfasst die Entwicklung der Aufrufhäufigkeit durch verschiedene Nutzer im Zeitverlauf. Steht neben der neuen Service-Schnittstelle weiterhin eine alternative Legacy-Schnittstelle zur Verfügung, sollte sich der Nutzungsgrad zu Gunsten des Services verschieben.
2. Den unmittelbaren Beitrag der SOA zu den Unternehmenszielen messen häufig bereits existierende Indikatoren; jede SOA-Initiative sollte deshalb den positiven Effekt beziffern, den sie auf einzelne Erfolgsparameter ausüben wird. Beispiele hierfür sind Prozess-Durchlaufzeiten, Liegezeiten oder der Anteil an Ausnahmefallbehandlungen in der Sachbearbeitung. Die Quantifizierung des Nutzens und die Zurechenbarkeit von Effekten stellen dabei allerdings immer wieder eine methodische Hürde dar.

WWW.GABLER.DE

Unverblümt, direkt und treffend – alle Tools für einen erfolgreichen, authentischen Karriereweg



Horst G. Kaltenbach **Persönliches Karrieremanagement**

Wie Karriere heute funktioniert –
Einsichten und Tools vom Karriere-Doc

2009. 184 S. Br. EUR 24,90 ISBN 978-3-8349-1113-1

Der engagierte Karriere-Coach Dr. Horst G. Kaltenbach vermittelt intime Einsichten in die tatsächlichen Karrieremechanismen. Er entlarvt gängige Missverständnisse zum Thema „beruflicher Erfolg“ und sagt klipp und klar, worauf es in den typischen Stationen des Werdegangs wirklich ankommt. Konkrete Checklisten und griffige Tools befähigen zur eigenen Talent- und Standortbestimmung; anschauliche Beispiele und direkte Hinweise führen den Leser auf einen authentischen, erfolgreichen Berufsweg.

Einfach bestellen:
kerstin.kuchta@gwv-fachverlage.de Telefon +49(0)611. 7878-626

KOMPETENZ IN
SACHEN WIRTSCHAFT



Änderungen vorbehalten. Erhältlich im Buchhandel oder beim Verlag.

3. Positive Effekte auf das IT-Management weisen Kennzahlen des IT-Projektcontrollings aus. Dazu gehören die durchschnittliche Anzahl von einer Anforderung betroffener Systeme, die Dauer von Projekten mit Auswirkung auf Prozessabläufe oder der durchschnittliche Aufwand zur Integration neuer externer Partner. Unglücklicherweise ist eine Historisierung und Verdichtung solcher Kennzahlen über den Kontext des jeweils einzelnen Projekts hinaus in vielen Unternehmen nach wie vor ein Desiderat.
4. Erwartungen zu positiven Effekten für die IT-Organisation sollte der IT-Verantwortliche jeweils mit Bezug auf die konkrete Anwendungslandschaft formulieren. Auch diese Ziele sollten über Kennzahlen zu Beginn der Initiativen definiert werden – allein schon, um eine historische Baseline für die Erfolgsmessung zu etablieren. Typische Indikatoren sind die Kosten für den Betrieb oder die Anpassung eines bestimmten Altsystems, der Standardisierungsgrad der Plattform-Infrastruktur oder das Verhältnis von Online- zu Batch-Verarbeitung. Leider häufig aus der Kostenbetrachtung ausgeblendet werden die Verwaltungsaufwände, die die SOA über eigens eingesetzte Stabsstellen, Architektur-Boards und andere Gremien induziert.
5. Neben der Flexibilisierung von Abläufen ist ein wesentliches Erfolgsversprechen serviceorientierter Architekturen die Verbreiterung der Datenbasis über ebendiese Abläufe, da im Prinzip jeder sauber modellierte Service als Protokollstation arbeitet. Die Zeitdauer bis zur Bereitstellung von entscheidungsrelevanten Informationen ist daher ein relevanter Indikator für den Erfolg einer SOA-Transformation. Selten im Blickfeld, aber durchaus interessant ist der Nutzen, den SOA im Bereich Compliance spielen kann – durch die mögliche Protokollierung von Prozessabläufen zum einen, durch die Dokumentation von Geschäftsregeln im Zuge der Automatisierung zum anderen. Der für die Erfüllung regulatorischer Vorgaben anfallende Aufwand könnte daher ebenfalls seinen Platz im Kennzahlensystem finden.

Nutzen der SOA für den Controller

Relevant aus Controlling-Perspektive sind die Möglichkeiten, die eine voll ausgebildete SOA für die Etablierung einer Prozesskostenrechnung und die Verknüpfung

technischer mit fachlichen Service Levels erschließen könnte. Tatsächlich haben bislang die wenigsten Unternehmen mit ihren SOA-Initiativen einen Zustand erreicht, der diese Potenziale in greifbare Nähe rücken würde. Nichtsdestominder bleibt das Ziel erstrebenswert, über die Zuordnung von IT-Kosten auf einzelne Prozess-Schritte langfristig die Transparenz zu erhöhen und das Dilemma der korrekten Zuordnung von IT-Gemeinkosten zu lindern.

Business Activity Monitoring (BAM), also die Auswertung von Prozess-Statusinformationen in Echtzeit, wird mit Sicherheit für viele Unternehmen ein Schlagwort bleiben, weil trotz Hersteller-Versprechen aktuell noch hinreichend mächtige und robuste Werkzeuge fehlen. Zweifellos verbessern jedoch Auswertungen auf der Ebene von Service-Aufrufen im Sinne von Prozess-Schritten die Verfügbarkeit und Konsistenz der Datenbasis für das Controlling im Allgemeinen.

Schließlich bleibt zu hoffen, dass aus dem Controlling abgeleitete Steuerungsimpulse dank verbesserter Flexibilität der IT-Architektur schneller ihren Weg in die Systeme und damit das operative Geschehen finden.

Auch und gerade der Controller ist Stakeholder einer SOA-Initiative im Haus. Auf folgende Punkte sollte er achten:

- Der Service-Zuschnitt ist auch ein Controlling-Thema. Verursachergerechtes IT-Kostencontrolling oder Prozesskostenrechnung profitieren nur von SOA, wenn die Grenzen der Service-Definitionen eine entsprechende Abbildung auch zulassen. Eigentümerschaft und erwarteter Konsumentenkreis sind wichtige Bestandteile der Vertragsinformationen jedes Services.
- Unerlässlich ist ein Fahrplan für die langfristige Komplexitätsreduktion in der IT-Landschaft. Im Regelfall startet eine SOA-Initiative mit der Anschaffung zusätzlicher Plattformen mit hohen Lizenz- und Wartungsfolgekosten. Es muss klar sein, welche Altsysteme, Schnittstellen und älteren Integrationswerkzeuge im Gegenzug entfallen können.
- Der Fortschritt der SOA-Initiative muss sich über ein mit der IT-Abteilung vereinbartes Set an Kennzahlen überprüfen lassen. Grundlage für die Erfolgsmessung ist ein Service-Repository, also ein Verzeichnis verfügbarer Dienste und Ihrer Schnittstellendefinitionen. Halten Sie auch das Projektcontrolling im Blick: Nach einer Anlaufphase sollte sich für die bereits er-

schlossenen Gebiete der Anwendungslandschaft die vermeintliche Flexibilisierung in reduzierten Design- und Entwicklungsaufwänden niederschlagen.

- Der Ausbau der SOA-Infrastruktur sollte stets mit einem fachlichen Anlass Hand in Hand gehen. Die Kosten für architektonische Basisarbeiten sollten dagegen davon getrennt ausgewiesen werden. Das erhöht die Akzeptanz, weil niemand die SOA mit seinen Projekten subventionieren muss – und verhindert, dass die Fachbereiche statt universal verwendbarer Komponenten sich genau den individuellen Service bauen lassen, für den sie schließlich auch bezahlen.
- Allzuoft versuchen Hersteller durch Rebranding ihr in die Jahre gekommenes Produktportfolio wieder attraktiv zu machen. Erst eine technische Verprobung im eigenen Haus gibt einer IT-Mannschaft einen validen Eindruck von der technischen Beherrschbarkeit und der Validität von Versprechen zu Robustheit und Performance.

Bewertung des SOA-Nutzens

Wie oben dargelegt liegt der Hauptvorteil einer voll ausgebauten SOA in der Kombinierbarkeit von Funktionen aus verschiedenen Einzelanwendungen zu neuen Abläufen, in denen auch externe Partner geschmeidig in die Prozesskette integriert sind. Der Fokus liegt dabei klar auf einer Erhöhung des Automatisierungsgrads durch koordinierte Kommunikation zwischen den Anwendungen.

Die damit gewonnene Flexibilität und vor allem Beschleunigung bedeutet handfeste Wettbewerbsvorteile für ein Unternehmen. Ebenfalls positiv schlägt die größere Transparenz über Prozessabläufe zu Buche. Das gilt sowohl für die Nachvollziehbarkeit des Einzelfalls (wann ist eine Meldung eingegangen, wann weitergeleitet worden) als auch für die Analyse im Zeitverlauf (wie viele Meldungen erfordern eine manuelle Nachbearbeitung).

Innerhalb der IT selbst stellt der höhere Wiederverwendungsgrad in der Anwendungsentwicklung einen Vorteil dar, der sich allerdings erst langfristig auswirkt und eine gewisse Reife des Katalogs an Services voraussetzt. Grundsätzlich vorteilhaft ist die Nutzung der neuen offenen technischen Standards, die Abhängigkeiten von den Produkten und Formaten einzelner Hersteller vermeiden hilft.

Erkauft sind diese Vorteile mit einer zumindest in der Anfangsphase ansteigenden Gesamtkomplexität der IT-Landschaft und einer entsprechend steilen Lernkurve für Anwendungsentwicklung und Betrieb. Die Anpassung einer einzelnen Anwendung zieht deshalb in der Regel keinen unmittelbaren Geschwindigkeitsvorteil aus der Nutzung von SOA-Prinzipien. Die von Anfang an notwendige SOA-Plattform mit Komponenten wie dem Service Bus und der Service Registry ist eindeutig ein Kostentreiber in puncto Hardware und Lizenzen. Als Mittel zur kurzfristigen IT-Kostenreduktion dient SOA daher nicht (**Abbildung 7**).

Nicht zu unterschätzen ist auch der erhöhte Koordinationsaufwand, den die Entwicklung und Pflege eines Katalogs wieder verwendbarer Services IT und Fachseite gleichermaßen abverlangt. Das Auflösen der Grenzen zwischen einzelnen Systemen erfordert auch neue Formen der Zusammenarbeit zwischen den Systemverantwortlichen. Bei eingefahrenen Strukturen in Unternehmen mit hauseigener IT kann hierin ein erhebliches Konfliktpotenzial stecken. Nicht wenige werden die notwendige Abgabe von Entscheidungshoheit über „ihre“ Systeme im Rahmen einer neuen IT Governance zunächst vor allem als Einschränkung erleben. Das angestrebte Alignment von Business und IT ist daher eher Voraussetzung als Folge serviceorientierter Architekturen.

Zusammenfassung und Ausblick

Die Diskussion um serviceorientierte Architekturen hat gezeigt, dass SOA im Entwicklungsverlauf in einem Stadium angekommen ist, in dem das Interesse von Unternehmen und IT-Verantwortlichen geprägt ist durch eine sachliche Debatte und reale wirtschaftliche Anwendungsmöglichkeiten. Erst durch neuere technische Entwicklungen wie Offenheit von Standards oder komfortable Auszeichnungssprachen (z. B. XML) ist die Verwirklichung der SOA-Prinzipien machbar geworden. Damit lassen sich Dienste erstellen, die „loose coupling“, Wiederverwendbarkeit und geringe Komplexität ermöglichen.

Die Neuausrichtung der internen Vorgehensmodelle in der Anwendungsentwicklung kennt verschiedene Herangehensweisen. Idealtypisch, wie auch die Fallbeispiele aus unserer Beratungspraxis belegt haben, wird eine evolutionäre Transformation der Anwendungslandschaft in enger Abstimmung zwischen Fachseite und IT-Abtei-

Abb. 7 | Vorteile und Nachteile serviceorientierter Architekturen

Vorteile	Nachteile
<p>Auf der IT-Seite</p> <ul style="list-style-type: none"> • Größere Flexibilität bei der Systemintegration • Mehr Wiederverwendung in der Entwicklung • Leichtere Einbindung externer Partner • Weniger Abhängigkeit von Herstellerformaten <p>Auf der Fachseite</p> <ul style="list-style-type: none"> • Höhere Automatisierung in Kernprozessen • Verkürzung von Prozess-Durchlaufzeiten • Bessere Nachvollziehbarkeit von Prozessen • Größere Datenbasis für Prozess-Analysen 	<p>Auf der IT-Seite</p> <ul style="list-style-type: none"> • Größere Komplexität der Gesamt-Architektur • Hohe Anfangsinvestition erforderlich • Hohe laufende Kosten der SOA-Plattform • Steile Lernkurve für Entwicklung und Betrieb <p>Auf der Fachseite</p> <ul style="list-style-type: none"> • Hoher Koordinationsaufwand für Definition und systemübergreifende Pflege der Services • Zunächst keine höhere Geschwindigkeit bei der Umsetzung einzelner Fachanforderungen

lung empfohlen. Klar abgegrenzte und überschaubare Pilot-Projekte erlauben eine kontrollierte Transformation mit stetiger Rückversicherung über den erwarteten Nutzen.

Die Komplexität einer SOA-Umsetzung muss vor Beginn einer Einführungsphase vollständig reflektiert werden. Die Einführung bspw. eines Enterprise Service Bus ist eben nicht nur gleichzusetzen mit einer großangelegten Softwareeinführung. Langfristig müssen auch die vielfältigen darunterliegenden transaktionalen Systeme angepasst werden. Serviceorientierte Architekturen können unter den oben beschriebenen Prämissen einen umfassenden Beitrag zur Verwirklichung der unternehmenseigenen IT-Strategie und zur flexiblen Anpassung an sich ändernde Geschäftsanforderungen liefern (vgl. Schelp/Aier 2009). Diese Strategie darf jedoch nicht nur technologisch getrieben sein.

Angesichts des hohen initialen Änderungsaufwands bei der Einführung von SOA und entsprechend notwendiger strategischer Anschubinvestitionen ist es umso wichtiger, deren Amortisation auf lange Sicht nachzuverfolgen und über den Nutznachweis zu rechtfertigen. Damit dies gelingt, ist es sinnvoll und notwendig, dass sich das Controlling schon in den frühen Phasen einer SOA-Initiative bei der Definition eines Rahmenwerks zur Erfolgsmessung einbringt.

Bis sich der SOA-Nutzen handfest nachweisen lässt, ist nach bisherigen Projekterfahrungen auf allen Seiten ein langer Atem notwendig. Besonders wichtig ist dabei, die hinter einer etablierten Service-Fassade möglichen Rückbau-Maßnahmen in der IT-Landschaft auch konsequent in Angriff

zu nehmen. Bevor man sich also auf die nächste Generation kursierender Schlagworte wie „Event-Driven Architecture“ und „SOA 2.0“ stürzt, sollten erst einmal die Hausaufgaben zur Realisierung von „SOA 1.0“ gemacht sein.

Literatur

1. Cummins, M. et al.: BCG's benchmarking study on IT in leading pharma companies, September 2005
2. Heydorn, S., Rejdak, H., Seibold, M.: What gets measured gets done – How the collection and benchmarking of IT data can drive gains for insurers IN: BCG IT Advantage – Putting Information Technology at the core of the business, Februar 2009
3. Möllenkamp, H., Freudenstein, G., Wolter, A.: IT Performance in the European Banking Industry. BCG's sixth annual IT cost benchmarking study, Juni 2009
4. Pulier, E., Tayler, H.: Understanding Enterprise SOA, Greenwich (Manning) 2005
5. Schelp, J., Aier, S.: SOA and EA – Sustainable contributions for increasing corporate agility; Proceedings of the 42nd Hawaii International Conference on System Science, January 5-8, 2009
6. Sholler, D.: 2008 SOA User Survey: Adoption Trends and Characteristics, Gartner Group, 26 Sept 2008