

# Business Technology

Architektur, Innovation & Strategie



Norm Judah:  
„Zukunft als  
Gegenwart“

## ROBUST und *flexibel*



Baukasten  
für Fachleute

Agiles Architektur-  
management

Technische  
Schulden

## Business-SOA und technische Integration

# Migrationsschritte zur Servicelandschaft

---

Die atemberaubende Dynamik des Wettbewerbs diktiert dem Business höchste Agilität. Die IT-Architektur muss sich schnell, adaptiv, kosteneffizient, nachhaltig und risikofrei an neuen strategischen Zielen ausrichten. Transparenz über fachliche Strukturen wie Business Capabilities und deren vertikalisierte Wechselbeziehung mit autonomen Services spielen in der hier vorgestellten Transformationsmethodik eine Schlüsselrolle.

---

AUTOR: KORNELIUS FUHRER

Während ich mich in meinem Artikel im Business Technology Magazin 1.2013 [1] mit Architekturmanagement und den Methoden und Instrumenten zur strategischen Steuerung der SOA-Transformation auf der Basis des Enterprise Architecture Managements (EAM) auseinandergesetzt habe, richte ich meinen Fokus in diesem Beitrag auf die businessorientierte Gestaltung und die technische Integration.

### VON DER STRATEGIE ZUR BUSINESS-SOA

Unternehmen agieren innerhalb einer Wettbewerbsumgebung, die aus Trends, Kunden, Lieferanten, Wettbewerbern, Branchenentwicklungen etc. besteht. Unter Berücksichtigung dieser Faktoren werden Grundsatzentscheidungen getroffen, abgestimmt und umgesetzt. Die Koordination des Verhaltens unterschiedlicher Organisationseinheiten zur Erreichung gemeinsamer Ziele wird als Strategie bezeichnet. Das Businessmodell beschreibt Organisationseinheiten einer Domäne und ihre ordentliche Geschäftstätigkeit. Das Businessmodell ist somit

eine Zeitpunkt Betrachtung. Die Strategie beschreibt das koordinierte Vorgehen zur Erreichung von gesteckten Zielen zu einem zukünftigen Zeitpunkt. Sie beschreibt die Transformation des Businessmodells vom Istzustand in einen Sollzustand und liefert somit eine Zeitraum Betrachtung.

**Abbildung 1** detailliert diesen Sachverhalt. Großunternehmen und Konzerne unterteilen sich in mehrere Domänen, die jeweils eigene Businessmodelle umsetzen. Sie lassen sich weiter horizontal in Business Capabilities unterteilen, die durch eine definierte Anzahl von Services durch die IT unterstützt werden. Daraus geht hervor, dass die SOA-Transformationsstrategie unmittelbar an die Transformationsstrategie des Businessmodells gekoppelt sein muss.

Ausgerichtet an der Strategiearchitektur betrachtet die Businessarchitektur die Fachlichkeit der Organisation. Wie bei Businessarchitektur und IT-Architektur muss auch zwischen Business- und IT-Strategie eine Integration stattfinden. Alle Architekturelemente der

Businessarchitektur, wie Domänen mit ihren lokalen Businessmodellen und -produkten, die Geschäftsprozesse mit Akteuren und unterschiedlichen Rollen, oder Business Capabilities müssen mit strategischen Zielen der Strategiearchitektur verknüpft sein. Eine mangelhafte Integration führt zu nicht abgestimmten, oftmals kontraproduktiven Aktivitäten, welche die Umsetzbarkeit der Strategien am Ende verhindern. Dieser Punkt unterstreicht, wie wichtig die strategische Ausrichtung ist und wo eine SOA-Transformation organisatorisch aufgehoben werden muss, damit sie noch koordinierbar und umsetzbar bleibt.

### BUSINESSORIENTIERUNG DURCH VERTIKALISIERUNG DER IT-UNTERSTÜTZUNG

In den vergangenen Jahren wurden immer neue Methoden zur Beschreibung des „Was“ im Business entwickelt. Diese beruhen auf der Annahme, dass alleine die Businessprozesse und Business Use Cases einzelner Geschäftseinheiten keinen ganzheitlichen Einblick in die unternehmensweite Serviceidentifikation gewährleisten und folglich nicht das richtige Mittel für die businessorientierte Gestaltung bei einer SOA-Transformation sind. Stattdessen favorisieren Businessarchitekten eine Sicht auf das Unternehmen, die auf äußerst grobgranularen Businessfunktionalitäten basiert, so genannten Business Capabilities, die von den jeweiligen Businessprozessen sowie Informations-, Anwendungs- und organisatorischen Silos (Produkte, Kunden etc.) unabhängig sind. Anhand von Business Capability Maps [3], als zusätzliche Architekturmanagementsicht zu den im ersten Teil vorgestellten Bebauungsplänen und Roadmaps können Unternehmen grobgranulare Funktionalitäten identifizieren und sich auf diejenigen bei einer SOA-Transformation konzentrieren, die beispielsweise geschäftskritisch oder für den Geschäftserfolg entscheidend sind.

Ausgerichtet an den Architekturelementen der Strategiearchitektur wird das Businessmodell in Business Capabilities über einige Stufen von Sub Business Capabilities bis auf Business-

funktionen vertikal dekomponiert (Abb. 2). Services implementieren das durch Businessfunktionen abgegrenzte fachliche Segment einer Business Capability – deren Businessfunktionen lassen sich in der Praxis direkt auf Serviceoperationen abbilden. An der Schnittfläche zwischen Business- und Servicearchitektur werden technische Strukturen an fachlichen ausgerichtet und bilden den Kern der Businessorientierung.

Analog zu Business Capabilities lassen sich deren zugeordneten Businessobjekte über Informationsobjekte bis auf physikalische Entitäten dekomponieren. Besonderes Augenmerk liegt hier auf den Informationsobjekten, die einen disjunkten Ausschnitt der Businessobjekte darstellen und als Vorlage für die Datenhoheit von Services sowie zur Implementierung kanonischer Datenmodelle dienen. Lesen Sie mehr hierzu im Blog des Autors unter Objektmodelle [2].

Inhaltlich sind Services und Business Capabilities korrespondierende Artefakte, die eine in sich abgeschlossene und eigenständige Geschäftstätigkeit aus unterschiedlichen Perspektiven (technisch, fachlich) abbilden. Betriebswirtschaftlich findet an diesem Punkt eine Substitution von menschlicher Arbeit durch Kapital (hier: IT) statt. Für die Business Capability ist es daher unerheblich, ob eine Substitution von Arbeit durch IT bereits vorgenommen wurde. Bei der Servicebetrachtung ist eine solche Substitution bereits erfolgt. Beide Konstrukte unterscheiden sich durch den Grad der IT-Unterstützung, sind ansonsten jedoch verhaltensähnlich

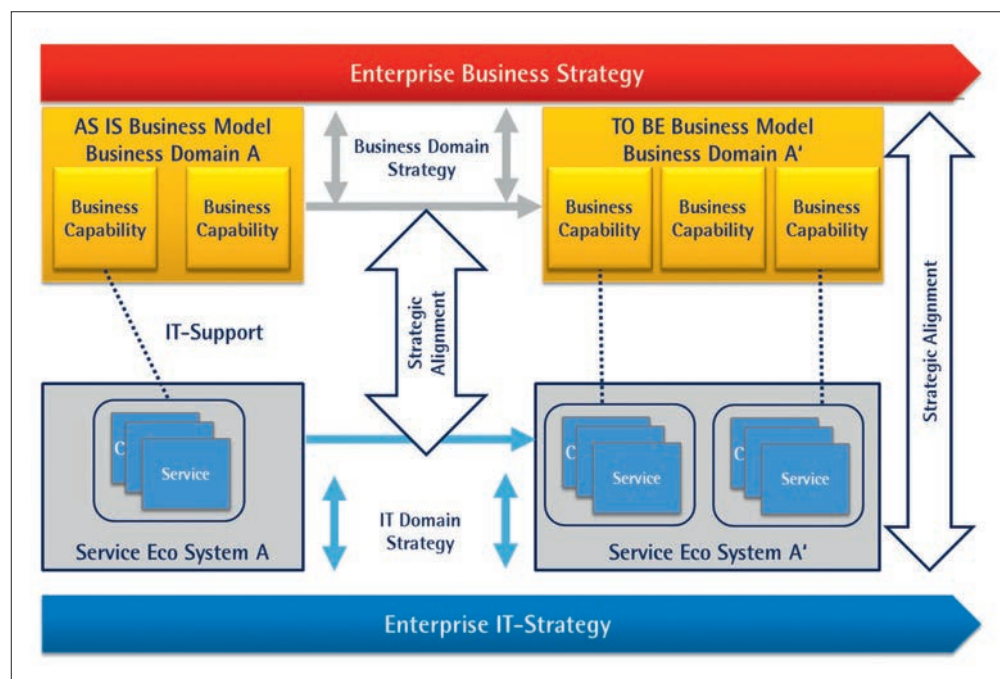


Abb. 1: Strategische Ausrichtung

und können einseitig substituier- und ergänzbar in die Businessarchitektur integriert werden.

## Top-down businessorientiert gestalten und Bottom-up über Projekte zum Leben bringen!

### TECHNISCHE INTEGRATION IN DER PRAXIS

Architekturprinzipien bilden die Grundlage für die Evolution großer Servicelandschaften. Abgeleitet von den Grundmaximen eines Unternehmens, dürfen sich diese eigentlich nie ändern. Architekturprinzipien werden über Standards operationalisiert. Richtlinien geben vor, wie Standards am besten umgesetzt werden können. Diese Prinzipien müssen knapp und verständlich sein und sollten nur die elementaren Schlüsselfragen beantworten. Architekturprinzipien helfen Architekten und Entwicklern bei ihren täglichen Servicearchitektur-Designentscheidungen bei der Transformation zur Zielarchitektur. Allgemein akzeptierte Prinzipien beschleunigen den Entscheidungsprozess und unterstützen die Konsistenz der neuen Lösung.

Neben den für die Transformation essenziellen Prinzipien, wie flexible Wiederverwendung zur Laufzeit, do-

mänenübergreifende, standardisierte Interoperabilität oder vertikalisierte Zugriffskontrolle durch Datenhoheit, ist die Serviceautonomie das zentrale Prinzip für die businessorientierte Gestaltung. Tabelle 1 konkretisiert die Kernpunkte der Serviceautonomie. Demzufolge existieren hochgradig autonome Services unabhängig von ihrer Peripherie als eigenständige Konstrukte. Indem Services exklusiv ein Segment an Businesslogik redundanzfrei kapseln, adressieren sie die fachliche Dekomposition innerhalb eines Serviceinventars.

Ausgehend von den fachlich abgeschlossenen Business Capabilities findet eine weitere Verfeinerung der Funktionsdomäne bis auf autonome Services statt. Sie bilden damit frei substituierbare Säulen durch alle Teilarchitekturen von der Strategiearchitektur bis hinunter zur technischen Architektur.

Mittels Domain Service Eco System (Tabelle 2) werden fachliche Services strukturiert sowie die Transformationsgrenzen seiteneffektfrei durch lokales Serviceinventar und lokalen Service Bus von den restlichen Domänen abgeschottet.

Mit diesem Prinzip lassen sich domänenübergreifende Harmonisierungsbestrebungen durch Unternehmensstandards und Richtlinien gestalten. Zusätzlich wird innerhalb des Eco-Systems Spielraum für eine kontrollierte Heterogenität und somit ein gewisses Maß an fachlicher Flexibilität geschaffen, der wichtige Innovationen und domänenspezifische Best-of-Breed-Ansätze

Serviceautonomie	
Vertikalisiert	Zugriff auf Businesslogik über Servicevertrag
Eigenständig	Businesslogik ist abgeschlossen und vollständig
Eindeutig	Businesslogik ist normalisiert und redundanzfrei
Unabhängig	Extrem lose Kopplung (Event-driven), hierarchiefrei, referenzfrei, kontextfrei
Laufzeitautonomie	Absolute Kontrolle über die eigene Laufzeitumgebung inkl. technischer Bausteine

Tabelle 1: Servicearchitekturprinzip: Serviceautonomie

Domain Service Eco System	
Seiteneffektfrei	Unabhängig von anderen Service- und Businessdomänen
Aggregiert	Fachlich zusammengehörende autonome Services
Fachliches Regelsystem	Grundlage ist das Businessmodell sowie dessen Capabilities
Evolution der Regeln	Regelsysteme implementieren die Vorgaben der IT- und Businessstrategie dieser Domäne
Funktionale Adaptivität	Autonome Services treten ein oder aus, erweitern oder reduzieren die Fähigkeiten des Ökosystems zur Laufzeit
Strukturelle Modifikation	Autonome Services sind dynamisch an die wechselnden Ressourcen der spezifischen Standardplattform zur Laufzeit gebunden
Lokal heterogen	Lokale Innovationsfreude und Möglichkeit für spezifische Best-of-Breed-Ansätze
Global standardisiert	Synergieeffekte durch Standards
Service Bus pro Domäne	Implementiert die technische Segmentierung

Tabelle 2: Servicearchitekturprinzip: Domain Service Eco System



zulässt und damit Wettbewerbsvorteile ermöglicht.

Diese Prinzipien unterstützen eine agile Vorgehensweise nach dem Managed-Evolution-Ansatz. Im Folgenden wird dieser Ansatz weiter über die erforderlichen Migrationsschritte detailliert.

Die Migrationsschritte 2 und 3 in **Abbildung 3** zeigen die Architektur mit Business Capabilities und Business Functions (gelb), Applikationen (blau), deren technischen Bausteine (grau) sowie Wechselbeziehungen über Informationsflüsse (hellblau). Welche Architekturelemente das Servicekonzept umfassen, wird durch einen blauen Rahmen mit dem S-Zeichen veranschaulicht.

Im linken Teil der jeweiligen Schritt-abbildung ist immer der Transformationsbereich der Servicelandschaft und im rechten Teil der Bereich der Legacy-Anwendungslandschaft dargestellt.

### Migrationsschritt I: Domänenschnitt

Der Managed-Evolution-Ansatz setzt nicht auf der grünen Wiese auf, sondern führt die SOA-Transformation während des laufenden Geschäftsbetriebs durch. So wird im ersten Schritt der Handlungsrahmen des Transformationsumfangs festgelegt.

Ist der Umfang zu klein, fehlen viele enggekoppelte Bausteine, sodass deren Abhängigkeiten nicht gemanagt werden können. In dem Fall würde die Transformation die anvisierten nachhaltigen Effekte nicht erreichen. Ist der Umfang zu groß, entsteht hingegen ein unnötiger Koordinationsaufwand zwischen Landschaftsteilen, die eigentlich sehr lose gekoppelt sind.

Es ist schwierig, den richtigen Handlungsrahmen zu finden. Das Kriterium sollte aber immer architektonisch sein und nicht organisatorisch. Daher wird aus der Legacy-Anwendungslandschaft im ersten Schritt dort ein Teil von dem Rest getrennt, wo der Kopplungsgrad am niedrigsten ist.

Nach dem Prinzip „Teile und Herrsche“ findet die Dimensionierung in Business-, Service- sowie in technischen Domänen statt.

### Migrationsschritt II: Schnittstellentransparenz

In diesem Schritt geht es primär darum, Transparenz innerhalb des Handlungsrahmens zu schaffen und alle Schnittstellen-Wrapper-Services explizit zu gestalten.

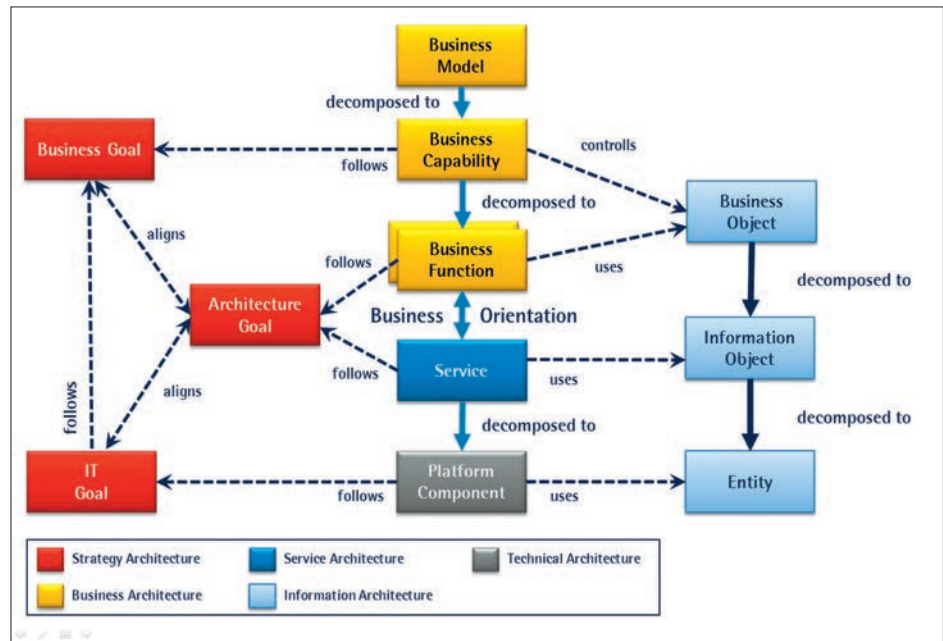


Abb. 2: Metamodellausschnitt zur Veranschaulichung der businessorientierten Gestaltung

Das bedeutet auch, dass alle Implementierungsdetails gekapselt werden müssen bzw. ausschließlich über den Servicevertrag nach außen sichtbar sind. So die Theorie. In der Praxis kommt der Architekt selten darum herum, hier Kompromisse einzugehen. Wegen eines hohen Vernetzungsgrads und vieler transitiver Abhängigkeiten ist es selten möglich, ganze Legacy-Anwendungen mit allen Komponenten und Funktionen in einem Rutsch zu transformieren und einfach abzuschalten. Daher muss hier schrittweise vorgegangen werden. Einige Komponenten werden gleich über Wrapper-Services gekapselt, andere erst später. Bis dahin benötigen diese aber weiterhin die Daten, um fehlerfrei funktionieren zu können. In solchen Fällen hilft beispielsweise die ereignisbasierte Delta-Replikation aus, die unidirektional alle entsprechenden Umsysteme „betankt“ und damit die Synchronität, Konsistenz und Funktionalität innerhalb der Anwendungslandschaft während des laufenden Geschäftsbetriebs sicherstellt.

### Migrationsschritt III: Businessorientierter Serviceschnitt

In diesem Schritt werden die Business Capabilities und Functions in Services transformiert, die wettbewerbsdifferenzierend oder besonders geschäftskritisch sind. Hier wird gemäß der vorgestellten vertikalen Dekomposition (**Abb. 2**) vorgegangen und Wrapper-Services werden durch fachlich geschnittene Businessservices abgelöst.

Auch hier lassen sich nicht alle Legacy-Anwendungen vollständig in einen Schritt isolieren und abschalten. Das Ergebnis ist daher ein Mischbetrieb aus Businessservices, Wrapper-Services und Legacy-Applikationen.

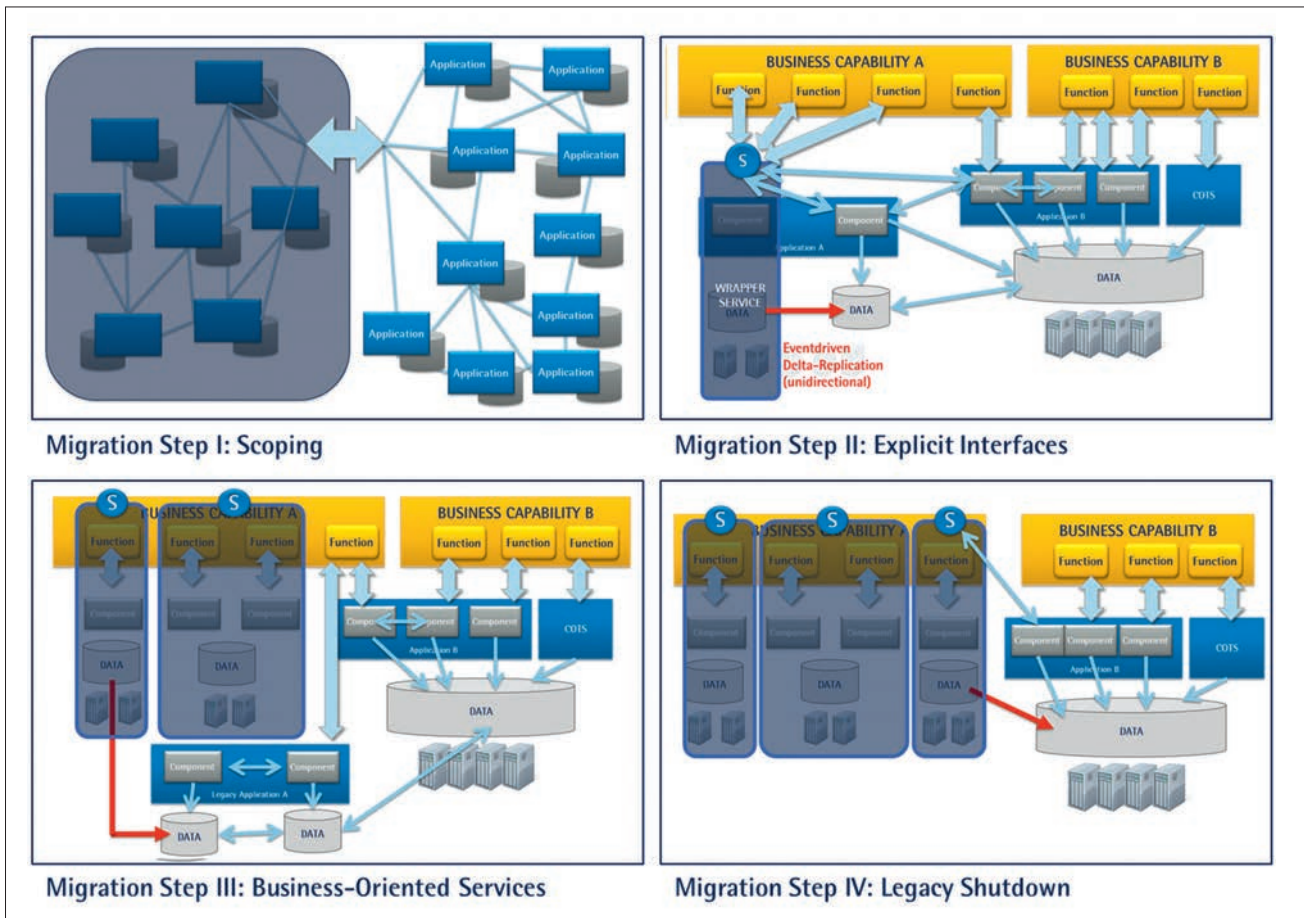


Abb. 3: Migrationsschritte zur Servicelandschaft

Da Servicekategorien wie Shared Services keine Businesslogik implementieren und damit nicht von Business Capabilities ableitbar sind, fallen sie aus dieser Betrachtung heraus. Die Servicelandskarte stellt diese durch ihre Querschnittsfunktion horizontal zu den vertikalen Businessservices dar.

#### Migrationsschritt IV: Legacy-Abschaltung

Im letzten Schritt werden alle Wrapper-Services zu Businessservices migriert sowie die entsprechenden Legacy-Anwendungen abgeschaltet.

Dabei ist es wichtig, abzuwägen, ob es kostengünstiger ist, vorerst eine Legacy-Anwendung zu entkoppeln, einen Teil der Funktionalität zu abstrahieren und über Services zu publizieren oder den Service neu zu entwickeln und die Legacy-Anwendung abzuschalten.

#### RESÜMEE

Das Architekturmanagement ist das Fundament für jede SOA-Transformation. Die hier definierten Abläufe wie das Demand-, Architektur-, Value- und Projektportfolio-Management helfen, Redundanzen und Inkonsistenzen proaktiv zu vermeiden und führen dabei zusätzlich

jedes Projekt sicher durch den Managed-Evolution-Korridor hindurch.

Ebenso definiert das Architekturmanagement den Aufbau der Architektur und bildet dabei alle Architekturelemente auf feste Strukturen ab. Deren horizontale und vertikale Wechselbeziehungen werden über das Metamodell verfeinert. Die so geschaffene Transparenz hilft nicht nur, die Architektur im Istzustand zu verstehen und zu analysieren, sondern auch, die Zielarchitektur im Sollzustand zu planen und über verschiedene projektspezifische Planarchitekturen zu operationalisieren. Dabei hilft die Verknüpfung der Strategiearchitektur mit den darunterliegenden Architekturen bei der zielorientierten Steuerung der SOA-Transformation. Sichten auf die transparente Architektur, wie beispielsweise Bbauungspläne, Roadmaps und Business Capability Maps, bilden einen Ausschnitt zielgruppenspezifisch ab und helfen, die Transformation entlang jedes Schritts zu verifizieren und projektweise in Richtung Zielarchitektur zu lenken.

Die Businessarchitektur mit ihren standardisierten symbolischen und semantischen Architekturelementen

und -prozessen bildet die übergreifende fachliche Klammer für die gesamte Unternehmens-IT und spielt deshalb eine zentrale Rolle bei der Auftragsdefinition der IT-Unterstützung. Durch die horizontale Strukturierung des Businessmodells in Business Capabilities sowie deren vertikaler Dekomposition wird ein durchgängiger fachlicher Ordnungs- und Strukturierungsrahmen sowie ein stabiles fachliches Vokabular für eine bindende Kommunikationsbasis und Zielvorgabe für den aktuellen und geplanten Geschäftsbetrieb zur Verfügung gestellt. De facto wird das Business damit befähigt, seine Anforderungen nicht auf technische, sondern auf funktional-präzise Weise zu formulieren und zu strukturieren. Das Modell macht sowohl eine zielorientierte Sicht auf Business Capabilities als auch eine businessorientierte Sicht auf die Architekturelemente der Servicelandschaft möglich. Damit fördert es das gegenseitige Verständnis zwischen Management, Business und IT und verschafft im Umfeld der SOA-Transformation die notwendige Transparenz, um die richtigen Entscheidungen konsistenter und schneller zu treffen.

Über vertikalisierte Strukturen durch alle Architekturen hindurch erfolgt die technische Integration mittels autonomer Services, was den seiteneffektfreien Austausch von Systemteilen ermöglicht. Domain Service Eco Systems clustern diese autonomen Services fachlich, kapseln Domänen mittels eines internen Service Bus technisch ab und limitieren damit die Auswirkungen der Transformationsiterationen auf diesem Bereich. Dies ermöglicht die seiteneffektfreie, iterativ-inkrementelle Migration von Wrapper-Services bis zur Legacy-Abschaltung sowie die Ausweitung der SOA-Transformation auf weitere Domänen. Bei jedem Schritt findet in der Regel ein wichtiger Erkenntnisgewinn statt, der in die nächste Entwicklungsstufe einfließt. Wrapper-Services stellen mit der Weiterverwendung der abstrahierten Legacy-Funktionen für den langwierigen Weg der evolutionären Vervollkommnung der Servicelandschaft das nötige Zeitfenster bereit. Sie gewährleisten somit einen belastbaren Investitionsschutz.

Ein oft vernachlässigter, aber auch sehr wichtiger Faktor ist der Mensch: Mitarbeiter müssen über die neuen Methoden und Prozesse informiert sowie motiviert und geschult werden. Die Vision der jeweiligen Transformation, also: Wohin die Reise gehen soll, was damit bezweckt wird und wie wichtig die Rolle des Einzelnen ist, sollte jeder kennen. Jeder Architekt und jeder Entwickler sollte das große Ganze im Kopf haben und sich über die Konsequenzen seiner Architektur bzw. Designentscheidungen im Kontext der SOA-Transformation bewusst sein. „Think big – act local“ lautet hier die Devise.

Die vorgestellten Methoden und Prozesse werden scheitern, wenn kein Commitment des Topmanagements vorhanden ist. Die strategische Einbettung mit einer zentralen Governance ist essenziell wichtig, um die SOA-Transformation übergreifend und nachhaltig zu koordinieren sowie lokale Personen- und Projektinteressen zu übersteuern.

## AUSBLICK

In einem Zeitalter, in dem Naturkatastrophen, Kriege, Finanzkrisen und die zunehmend unübersichtlichen Märkte erhebliche Auswirkungen auf Marktakteure haben und damit eine klassische strategische Langzeitplanung obsolet machen, wird die flexible Ausrichtung auf alternative Ziele wichtiger denn je. Unternehmen werden sich auch in Zukunft mit einer zunehmenden Dynamik und Unsicherheit des Geschäfts sowie der weiter wachsenden Komplexität konfrontiert sehen. Multiple Zielarchitekturen für verschiedene Permutationen an Geschäftsausprägungen müssen simuliert, evaluiert und abrufbar gemacht werden. Die ultimative Agilität und Adaptivität einer Servicelandschaft, die es möglich macht, mit hohem Tempo von dem einen in das nächste konsistente Sollszenario zu transformieren, wird zum wettbewerbsentscheidenden – wenn nicht gar zum überlebensnotwendigen – Faktor werden.

---

## Links & Literatur

- [1] Fuhrer, Kornelius: „SOA-Transformation – Architektur Management“, in Business Technology Magazin 1.2013
- [2] Fuhrer, Kornelius: „Objektmodell Aggregation und Abstraktion“: <http://enterprise-architecture-management.blogspot.de/2013/04/objektmodelle.html>
- [3] Fuhrer, Kornelius: „Geschäftsfähigkeitslandkarte“: <http://enterprise-architecture-management.blogspot.de/2012/02/geschaeftsfahigkeitslandkarte-business.html>



### Kornelius Fuhrer

ist IT-Management-Berater bei CASSINI Consulting und unterstützt Kunden bei der strategischen IT-Planung und der businessorientierten Gestaltung. Als Mitglied von Konzernarchitekturteams sowie von SOA-Initiativen in Kundenprojekten hat er maßgeblich an der Planung und Erstellung von SOA-Transformationskonzepten aus organisatorischer und prozessualer Perspektive mitgewirkt. Als Speaker und Autor berichtet er fortlaufend zu Themenkomplexen aus EAM, Business/IT-Alignment, Governance, SOA und EAI.