

Buzzwords: Wenn Consulting anfängt, confusing zu sein

Wenige Branchen sind so anfällig für Schlagworte wie die IT. Häufig vermittelt die Debatte über das Potenzial neuer Technologien allen Beteiligten nur noch eine vage Vorstellung von deren Substanz. Überstrapazierte Begriffe wie „Cloud“, „SOA“ und „ESB“ verlangen nach einer Rückbesinnung auf den ursprünglichen Begriffskern und einer Versachlichung der Diskussion.

Keywords

SOA, cloud, service bus, architecture consulting, innovation management

Stichworte

SOA, Cloud, Service Bus, Architektur-Beratung, Innovations-Management

1. Der Hype Cycle als Sinnbild der IT-Branche

Was haben Business Process Management und Web 2.0 mit Virtual Environments, Green IT und Cloud Computing gemeinsam? All diese Begriffe standen seit 2005 aufeinanderfolgend auf dem „Gipfel der überzogenen Erwartungen“ im „Gartner Hype Cycle of Emerging Technologies“ [1]. Mit dieser Fieberkurve haben die Analysten der Gartner Group wohl die eingängigste Visualisierung der raschen Abfolge von Schlagworten entwickelt, die die Fachpresse und das Marketing von Produktherstellern und Dienstleistern dominieren (vgl. Abbildung 1). Der seit 1995 jährlich veröffentlichten Experten-Einschätzung liegt die Hypothese zugrunde, dass die Auseinandersetzung von Herstellern, Anwendern und Branchen-Beobachtern mit Innovationen in der IT in den meisten Fällen den gleichen Verlauf nimmt: Auf anfängliche Begeisterung über das Potenzial einer Idee folgt regelmäßig große Ernüchterung angesichts von Hürden in der praktischen Umsetzung; eine dem tatsächlichen Potenzial angemessene Beurteilung findet erst

sehr viel später statt. Dies erscheint intuitiv plausibel, auch wenn die Darstellung (zumindest bis 2009) nicht von quantitativen Referenzen – etwa in Form von Verkaufszahlen, Anwenderbefragungen oder Begriffs-Häufigkeiten in Fachbeiträgen – begleitet wird. Viele (66 von 123 seit 2004 gelisteten) Themen tauchen erstmalig auf der Kurve auf, wenn sie bereits Hype geworden, die Erwartungen also bereits übersteigert sind. Für die meisten dieser Themen, die direkt auf dem Höhepunkt der allgemeinen Aufmerksamkeit einsteigen, nimmt Gartner eine Zeitstrecke von zwei bis fünf Jahren an, bis ein gewisser Reifezustand erreicht ist. In der Standard-Ansicht des Hype Cycle ist diese zeitliche Komponente über die Farbgebung der Positionsmarke für ein Thema angedeutet. Legt man jedoch wie in Abbildung 2 die Zyklen verschiedener Skalierungen auf der Zeitachse übereinander, erhält man eine recht augenfällige Illustration der Verwirrung, die viele IT-Verantwortliche gegenüber der Vielfalt an Trends und Schlagworten empfinden.

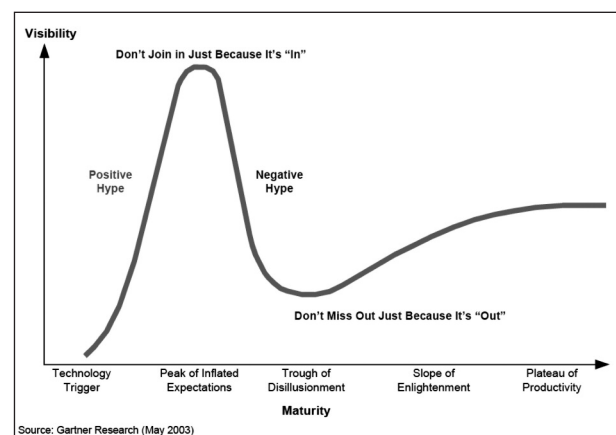


Abbildung 1: Phasen des Gartner Hype Cycle [10]

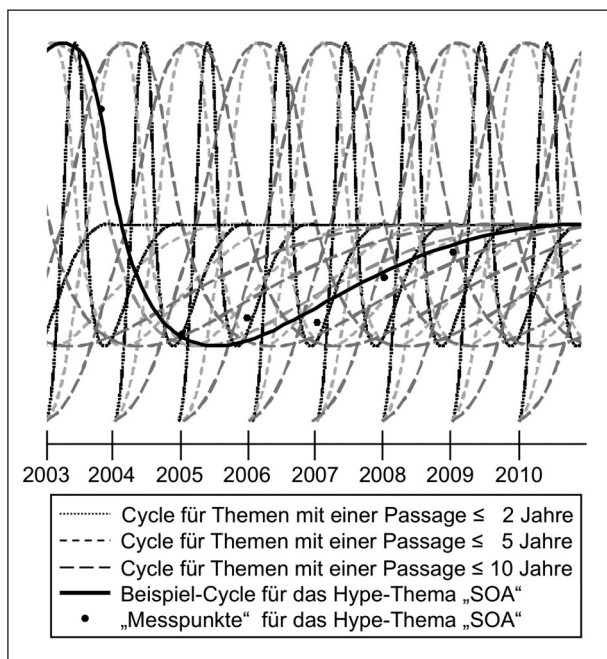


Abbildung 2: Der Hype Cycle – Sinnbild der Begriffsverwirrung

2. SOA und Cloud Computing: alter Wein in neuen Schläuchen?

Dass es auch für Branchen-Insider schwer sein kann, hier den Überblick zu behalten, zeigt das aktuelle Top-Thema „Cloud Computing“. Das Bemühen, Differenzen, Begriffseinschlüsse und Querbezüge zu vorangegangenen Konzepten wie „Software as a Service“ oder „Grid-Computing“ aufzuarbeiten, ist bereits Gegenstand zahlreicher Artikel [vgl. z. B. 2, 3]. Ein vermeintlich eleganter Ausweg aus dieser Definitionsnot ist, Unterschiede im Begriff lediglich als einen Wechsel der Perspektive zu beschreiben: Den Endanwender interessiert im Grunde genommen nicht, wie die Ressourcen-Allokation technisch funktioniert; was von außen also als „Cloud“ daherkomme, könne sich für den Systemadministrator von innen eben als „Grid“ darstellen. Mit einer solchen Argumentation können die Anbieter von Lösungen für die Server- oder Datenbank-Virtualisierung ihre Produkte als die Eckpfeiler einer „Private Cloud“ bewerben. Wenn aber die Wolke nur noch eine Metapher dafür ist, dass sich der Endanwender nicht mehr um Details der technischen Realisierung kümmern muss, mutieren auch Online-Spiele und Freemail-Angebote schnell zu Angeboten aus der Cloud [4]. Als Orientierungshilfe für den CIO taugt der Begriff spätestens hier nicht mehr, denn seine technische Bedeutungssubstanz hat er mit dieser Auslegung zweifelsfrei verloren. Ein vergleichbares Schicksal hat die „serviceorientierte Architektur“ (SOA) bereits hinter sich. Zu Beginn der 2000er Jahre konzentrierte sich die Diskussion über Serviceorientierung auf den Einsatz eines sehr spezifischen Technologie-Stacks, nämlich SOAP als Nutzdaten-Hüllformat über http als Transportprotokoll mit WSDL als Definitionsschema für den Schnittstellen-Kontrakt. Dieses Gespann erwies sich als Treiber für eine bis dahin selten er-

reichte Interoperabilität und lose Koppelung von Systemen – gegen den Aufpreis zusätzlicher Abstraktion und damit verbundenen Verarbeitungs-Overheads. Schlechte Erfahrungen mit Durchsatz und Wartbarkeit der ersten entstandenen Service-Landschaften führten zur unvermeidlichen Geringschätzung des Konzeptes, gepaart mit der ebenso wahren wie banalen Erkenntnis, dass eine neue Technologie allein kein sauberes, funktionales Design ersetzt.

Hier ist nun ein deutlicher Wandel in vielen Darstellungen des Konzeptes SOA zu beobachten: Die „richtige“ funktionale Gliederung eines Systems wird zum entscheidenden Kriterium für die Serviceorientierung, während die technische Umsetzung beinahe zur Nebensache verkommt. Auf diese Weise lassen sich auch CORBA-basierte Integrationslösungen als Fallbeispiele für eine gelungene SOA auffassen [5]. Gleichzeitig rücken die „richtige“ Governance und das Zusammenarbeits-Modell von Fach- und IT-Seite bei der Service-Definition in den Vordergrund der Debatte [6].

Wenn aber die Serviceorientierung nur ein Synonym für gutes Design und die richtige Organisationsform ist, wie können dann Plattform-Anbieter ohne Gesichtsverlust erklären, warum bestimmte Schwierigkeiten erst beim Einsatz neuer Produkte aufgetreten sind – oder welchen Anlass es überhaupt gab, ein neues Produkt einzusetzen?

3. ESB: Architektur-Stil oder Produkt?

Mitunter ist es nicht einmal wirklich einfach festzustellen, ob und seit wann man überhaupt neue Plattformen einsetzt. Ein schönes Beispiel hierfür stammt ebenfalls aus dem Umfeld der Serviceorientierung: Der Enterprise Service Bus (ESB), (Abbildung 3). Dieses Konzept begann seine „Karriere“ im Jahr 2002 als Beschreibung des EAI-Herstellers Sonic für seine neue Middleware-Komponente SonicXQ. Frühzeitig gelang es Sonic, Analysten für den Bus zu gewinnen; Sonics „Chief Technology Evangelist“ David Chappell machte keinen Hehl aus seinen erfolgreichen Bemühungen, den Begriff zu platzieren [7]. 2004 veröffentlichte Chappell bei O'Reilly eine ausführliche Beschreibung des Ansatzes [8]. Wer allerdings Chappells Ausführungen mit den zu diesem Zeitpunkt als ESB vermarkteten Komponenten vergleicht, stellt vergleichsweise wenige Ähnlichkeiten fest.

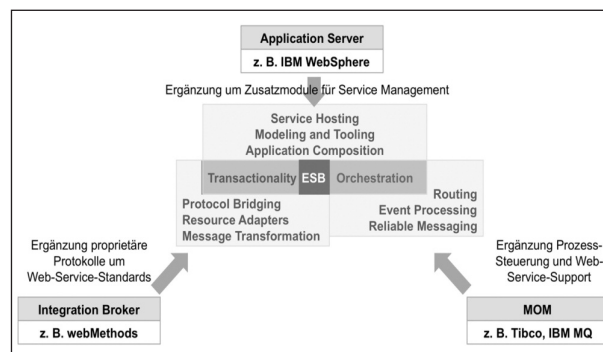


Abbildung 3: ESB – Marktgrenzung inzwischen äußerst schwierig

Chappells ESB ist in erster Linie ein Design-Konstrukt und keine einzelne Komponente: Mehrere Komponenten in der IT-Landschaft können Aufgaben übernehmen, die im Ganzen genommen einen Service Bus darstellen. Die dezentrale Realisierung des Bus-Konzeptes ist für Chappell von entscheidendem Vorteil für die Zuverlässigkeit, Robustheit und Performance der Integrationsarchitektur. Die meisten kommerziell verfügbaren ESB-Implementierungen erinnern dagegen eher an das Hub-And-Spoke-Prinzip traditioneller EAI-Architekturen mit den typischen Risiken für Durchsatz und Verfügbarkeit.

Diese Diskrepanz ist insofern nicht verwunderlich, als in den meisten Fällen lediglich ein Rebranding bestehender Produktportfolien stattgefunden hat. Abhängig vom Erbe der jeweiligen Komponente haben die betreffenden Hersteller dabei dem Akronym ESB verschiedene Schwerpunkte gegeben: Anbieter klassischer Integration Broker wie webMethods betonen Aspekte wie Protocol Bridging. MOM-Spezialisten wie Tibco stellen die Themen Routing, Event Processing und Reliable Messaging in den Vordergrund. Wer wie IBM einen starken JEE-Application-Server im Produktportfolio hat, betont Fähigkeiten wie Application Composition und das zugehörige Toolset. Angesichts dieser Fülle von Konnotationen fällt es seit mehreren Jahren schwer, den Begriff und einen entsprechenden Markt überhaupt sauber abzugrenzen [9].

4. Gegen den Hype hilft methodische Konsequenz, nicht grundlegende Skepsis

Die drei Beispiele SOA, ESB und Cloud stehen exemplarisch für die Unschärfe der Begriffe, die die IT-Branche vielfach dominieren. Doch welche Schlussfolgerungen sollten IT-Verantwortliche daraus für die Zukunft ziehen, und welche Position kann der IT-Berater gegenüber dieser Gemengelage einnehmen?

Seiner Natur nach motiviert der Hype Cycle eher zu einer konservativen Haltung: Im Gegensatz zu ökonomischen Innovationstheorien werden hier die Anwender der ersten Stunde für ihren Pioniergeist alles andere als belohnt – und zwar mit erstaunlicher Regelmäßigkeit. Liegen die Analysten mit ihren Urteilen im Zeitverlauf richtig, dann erreichen die allerwenigsten Themen jemals das „Plateau of Productivity“ – in den letzten sieben Jahren ganze fünf von rund 120.

Verkauft also ein IT-Berater seinem Kunden jedes Schlagwort aufs Neue als die nächste Lösung aller Probleme, führt dies mit Sicherheit zu einem raschen Verlust seiner Glaubwürdigkeit. Sich stattdessen betont altersweise zu geben und als Antithese zu jedem Hype kategorisch zu verkünden, es gebe für den Architekten im Grunde doch nichts Neues unter der Sonne, ist in gewisser Weise ebenso fahrlässig: Diese Einstellung leistet einem Prinzip des „Weiter so“ Vorschub, das Ursache für den Innovationsstau und damit in letzter Konsequenz für die Kostenfalle ist, in der sich viele IT-Verantwortliche wiederfinden.

Eine Antwort auf die allgemeine Klage über den Hype kann ein IT-Berater jedoch geben: in Form von methodischer Konsequenz.

4.1 Erster Schritt: Bemühen um eine saubere Begriffsabgrenzung

Der erste Schritt dazu ist, den Kern eines neuen Konzeptes so klar wie möglich abzugrenzen und diese Abgrenzung in der Folge aufrechtzuerhalten. Im besonderen Fall der IT-Beratung bedeutet dies vorwiegend Konzentration auf den technischen Kern der Materie. Der Aufbau einer serviceorientierten Architektur beispielsweise ist sicher auch eine Frage von Governance und Business Alignment, aber im Grunde geht es doch um spezifische Methoden der System-zu-System-Integration.

Die Konzentration auf die technischen Aspekte eines Architektur-Stils erleichtert zunächst einmal die Abgrenzung unterschiedlicher Systementwürfe: Eine Lösung wäre demnach genau dann service-orientiert, wenn sich in ihr ganz spezifische Schnittstellen-Konstrukte wiederfinden. Sich dabei z. B. auf einzelne Protokoll-Stacks einzuschränken mag als willkürlich angreifbar erscheinen – aber immerhin ist das Kriterium trennscharf. Die Vermischung von technischen und organisatorischen Aspekten führt dagegen häufig zu einer Überdehnung von Konzepten. Die Art und Weise, wie die Entwicklungs- und Betriebsabteilung organisiert sind, mag besser oder schlechter zu einer SOA passen, aber sie ist nicht selbst als Bestandteil der Architektur zu verstehen.

Dies kann unter Umständen dazu führen, dass die Verwendung von Begriffen wie „Serviceorientierung“ vom Sprachgebrauch Dritter deutlich abweicht. Allerdings drehen sich Hypes ja nur in wenigen Fällen wirklich um definierte Standards und Spezifikationen, die ein exaktes, universelles Verständnis einfordern. IT-Berater können und sollten es sich also in gewissen Grenzen erlauben, ihr „privates“ Begriffsverständnis zu pflegen, sofern sie es mit einer nachvollziehbaren Definition hinterlegen und konsistent anwenden. So, wie er sein eigenes Begriffsverständnis offenlegen muss, muss der IT-Berater auch die Selbsteinordnungen seines Kunden hinterfragen: Was meint dieser, wenn er von „seinem SOA-Ansatz“ oder von „seinem Service Bus“ spricht? Eine wechselseitige Erläuterung der Nomenklatur sollte den Anfang jeder Zusammenarbeit bilden.

Die Fokussierung auf bestimmte, gut definierende Aspekte erleichtert sowohl die Ist-Analyse als auch die Beschreibung eines architektonischen Zielbilds. Wenn nun dabei herauskommt, dass weder die Ist- noch die Ziel-Architektur im engeren Sinne „service-orientiert“ sind, dann ist das erst einmal eine wertneutrale Feststellung. In diesem Zusammenhang ist die besondere Herausforderung für den Berater, sich und den Kunden von der Vorstellung zu befreien, dass die Auswahl eines bestimmten Ansatzes an sich bereits gut oder schlecht ist, und dass eine Klassifikation kein Gütesiegel bedeutet.

Ohne diese Differenzierung würde aus der Einordnung des Architektur-Stils automatisch ein Urteil, und die Qualität eines Systementwurfes ergäbe sich bereits aus dem gewählten Vorgehen. Tatsächlich lässt sich die Qualität jedoch nur danach bewerten, ob die spezifischen technischen Mittel verglichen mit dem zu lösenden Problem angemessen oder unangemessen eingesetzt werden. Die Konzentration auf die Technik erlaubt eine solche stärkere Dif-

ferenzierung zwischen Verständnis und Bewertung. Nur so ist es auch möglich, zwischen einer guten und einer schlechten SOA-Lösung zu unterscheiden. Die spezifische Art und Weise, wie etwa eine Business Process Engine genutzt werden soll, macht schließlich die Stärke eines technischen Designs aus – nicht der Einsatz der Engine an sich.

4.2 Zweiter Schritt: Transfer auf die spezifische Situation des Kunden

Der konsequente zweite Schritt, um die Angemessenheit eines Lösungsansatzes zu bewerten, ist der immer notwendige Abgleich mit der spezifischen vorgefundenen Situation. Der IT-Berater muss verstehen, welches Problem seinen Kunden bewegt und wie es sich manifestiert, bevor er mit einem Lösungsvorschlag aufwartet. Fast immer ist für eine sorgfältige Bewertung eine technische Detailanalyse erforderlich, die eine enge Zusammenarbeit mit den Architekten auf Kundenseite voraussetzt.

Bei Fragen der grundlegenden Neuorganisation von Entwicklung und Betrieb kann ein Blick auf die reine Kostenverteilung allgemeine Indikationen liefern – um etwa bei der Evaluierung eines „Platform-as-a-Service“-Szenarios die Größenordnung des erreichbaren Kostenvorteils gegenüber der hausinternen Bereitstellung abzuschätzen. Doch auch hier ist eher früher als später der Blick aufs technische Detail notwendig, um z. B. den tatsächlichen Standardisierungsgrad der bestehenden Plattformen abschätzen zu können.

Architektur-Management ist selbst noch eher Schlagwort als gelebte Praxis – deshalb ist die vorhandene Informationslage für IT-Berater stellenweise unergiebig, wenn diese ihre Kunden nicht schon seit langer Zeit kennen. Abhilfe schafft oft eine konzentrierte Vorfeld-Analyse mit Workshops und strukturierten Interviews. Beides sollte sich keinesfalls auf den Kreis der späteren Entscheider beschränken, sondern auch die operative Ebene des Kunden mit einbeziehen; der Zeitaufwand für die notwendige Konsolidierung der dabei zusammengetragenen Detailfülle ist in jedem Fall gut investiert.

Spätestens bei der Ableitung einer Empfehlung zum weiteren Vorgehen werden herstellergebundene IT-Berater stets in einem gewissen Spannungsfeld stehen – denn prinzipiell ist der Anspruch eines Produkthauses ja, dass die eigenen Komponenten grundsätzlich in der Lage sind, die Probleme des Kunden zu lösen. Wer diesen Beweis nicht antreten muss, kann seine Analyse ungezwungener angehen und erst einmal abklopfen, ob überhaupt ein geeignetes Problem für die ins Auge gefasste technische Lösung besteht. Das Konzept des Enterprise Service Bus beispielsweise spielt seine Stärken in hochgradig verteilten und heterogenen Konstellationen aus, in denen sich die durch den Transport einer Nachricht abgebildeten Prozesse ad-hoc und abhängig vom Nachrichteninhalt selbst ergeben. Sind die abzubildenden Prozesse dagegen sehr stabil mit nur wenigen Entscheidungspunkten und spielen sie sich innerhalb eines bereits weitgehend harmonisierten Technologie-Stacks ab, dann wäre eine ESB-Komponente in diesem konkreten Fall möglicherweise tatsächlich nur Ballast,

ohne dass dadurch das Konzept des ESB an sich infrage gestellt wäre. IT-Berater, die zuvor bereits vermitteln konnten, dass gerade in architektonischen Fragen eine spezifische Lösung kontextbezogen und selten vorteilhaft aus sich selbst heraus ist, tun sich jetzt leichter damit, ihren Kunden die unnötige separate Bus-Schicht beispielsweise zwischen JEE-Web-Frontend und JEE-Anwendungskern auszureden.

4.3 Dritter Schritt: Konzeption einer individuellen Erfolgsmessung

Der dritte notwendige Schritt für die Auseinandersetzung mit einem neuen Ansatz besteht in der frühzeitigen Definition und dem Nachhalten individueller Erfolgskriterien. Wer sich von Web Services größere Flexibilität bei der Einbindung neuer externer Partner verspricht, muss wissen, wie er diesen Zugewinn an Flexibilität später messen will. Wer mit einem Service Bus die Schnittstellen-Komplexität in den Griff bekommen möchte, muss wissen, woran genau er diese Komplexität festmacht. Wer über Cloud Computing die Bereitstellung von Ressourcen optimieren will, muss eine Vorstellung von der Dynamik des Ressourcenbedarfs haben, den die Lösung abbilden soll. Für jeden Ansatz muss der IT-Verantwortliche wissen, gegen welche Ausgangsleistung er die erwartete Verbesserung rechnet und zu welchen ausgewählten Zeitpunkten er Bilanz über die Nutzenentwicklung ziehen will. Nur selten wirkt ein neues Konzept so fundamental, dass es einen separat messbaren Wert schafft, der zuvor überhaupt nicht darstellbar war. Meist wird die Veränderung gradueller Natur sein. Für die IT als Service-Funktion im Unternehmen bedeutet dies oft schlicht eine Verbesserung in der Kostenstruktur.

Hier liegt die vermutlich größte Herausforderung für IT-Berater. Die Mechanismen, nach denen Kunden ihre Aufwände ermitteln und verrechnen, unterscheiden sich von Fall zu Fall. Daher wird es meist notwendig sein, ein individuelles Modell für die Erfolgsmessung zu erstellen. Für die initiale Motivation einer Untersuchung ist es zulässig, die Erfolge anderer Unternehmen mit augenscheinlich ähnlichen Rahmenbedingungen ins Feld zu führen; langfristig will der Kunde jedoch den eigenen Erfolg nach eigenen Maßstäben bewerten können.

In aller Regel ist es dafür notwendig, eingespielte Controlling-Verfahren zu verändern, um die relevanten Größen – seien es Auslastungsgrade von Hardware oder spezifischer Design-Aufwand für Schnittstellen – für eine Erfolgsmessung überhaupt erst erheben zu können. Die damit verbundenen Änderungs- und Erhebungskosten müssen im Übrigen dem neuen Ansatz zugerechnet werden, um eine vollständige Erfolgsbilanz ziehen zu können. Zur Erfolgskontrolle gehört ein klarer Fahrplan mit Meilensteinen für die Messung von Umsetzungsgrad und Nutzeneffekten. Vorteil und Aufgabe für den IT-Berater ist dabei, durch den Vergleich mit Anforderungsprofilen und Herangehensweisen anderer Kunden realistische Erwartungen hinsichtlich des Erreichbaren zu vermitteln. Auch hier zahlt sich die trennscharfe Verwendung von Begriffen aus: Einem Auftraggeber ist mit der Aussage, ein anderer habe für die Umsetzung seiner Cloud-Strategie zwei Jahre

benötigt, nicht gedient, wenn er selbst darunter den externen Bezug von Plattform-Infrastruktur versteht, der angeführte Referenzkunde aber stattdessen Querschnittsanwendungen wie die Groupware-Lösung ausgelagert hat.

Gerade die Nutzenbewertung reiner Architektur-Fragen gilt gemeinhin als schwierig bis unmöglich; notwendig und herstellbar ist sie dennoch. Sinnvoll ist hier, sich auf Mittlergrößen zu verständigen. Höhere Flexibilität beispielsweise sollte sich in niedrigerem Anpassungsaufwand niederschlagen, größere Robustheit im Rückgang der Anzahl kritischer Fehler im Produktivbetrieb. Auch verbessertes „Business Alignment“ muss sich letzten Endes in irgendeiner messbaren Größe niederschlagen – im Zweifelsfall in der Allokation von Ressourcen im Entwicklungsprozess. Demnach ist die neue SOA ein Erfolg, wenn sie Design- und Implementierungs-Aufwand messbar reduziert – und zwar in jenen Teilen der Anwendungslandschaft, in denen sie bereits greift.

Dieser letzte Punkt ist nicht zu vernachlässigen: Die schiere Zeitdauer bis zur Umsetzung eines neuen Konzeptes wird immer zu einem parallelen Betrieb mehrerer technischer Evolutionsstufen in einer Anwendungslandschaft führen, es sei denn, deren Umfang ist ungewöhnlich überschaubar. Für diese Gleichzeitigkeit sorgen allein schon die fortschreitende Entwicklung von Produkten und Standards sowie natürlich die Notwendigkeit, das Tagesgeschäft aufrechtzuerhalten. Eine wirklich „reinrassige“ Architektur und ein zu 100 % durchgehaltenes Betriebskonzept werden einem IT-Berater als Ausgangslage in der Praxis nicht begegnen – und in den allerseltensten Fällen wird er Gelegenheit haben, eine technologische Transformation vollständig von A bis Z begleiten zu dürfen. Er kann aber zusammen mit seinem Kunden aus dieser Not eine Tugend machen und die parallel betriebenen Ansätze unter Produktionsbedingungen vergleichen. Dafür muss es jedoch möglich sein, Auswirkungen auf die Kosten oder den funktionalen Abdeckungsgrad lokal begrenzt zu messen. Ohne im Vorfeld konzipierte Messgrößen wird so ein Vergleich mehrerer Technologien oder Betriebsspielarten unter einem Dach stets Gegenstand von mitunter sehr emotional geführten Debatten sein. Es ist nicht weiter verwunderlich, dass der langjährige Leiter eines Entwicklungsteams für das Jobnetz auf dem Host skeptisch reagiert, wenn Prozesse auf Basis eines Service Bus in den Online-Betrieb überführt werden sollen. Vergleichswerte für Durchsatz, Fehlerraten und Entwicklungsaufwand können die notwendige Überzeugungsarbeit leisten. Allerdings müssen sich IT-Berater auch gegenüber ihren eigenen Umsetzungsvorschlägen unparteiisch verhalten: Zeigt ihr Konzept bei der Evaluierung nicht die erwartete Performance und können auch bei wohlwollender Prüfung keine außergewöhnlichen Umstände für eine Verzögerung des Nutzeintritts verantwortlich gemacht werden, sollten sie sich nicht scheuen, den Abbruch zu empfehlen. Dazu gehört konsequenterweise auch ein Vorschlag zum Rückbau der Lösung. Gerade die unvollendet aufgegebenen, aber beibehaltenen Renovierungsansätze verursachen die Komplexität vieler Anwendungslandschaften und tragen damit zur nachhaltigen Skepsis vieler IT-Verantwortlicher gegenüber dem nächsten großen Trend bei. Ein Ausstiegs-szenario bereits bei der Entscheidung für eine neue Technologie

mitzuformulieren ist allerdings nicht erforderlich und in der Regel sogar kontraproduktiv, weil es den Einstieg von vornherein halbherzig wirken lässt.

5. Fazit

Wird die IT-Branche auf lange Sicht mit Schlagworten operieren? Vermutlich schon: Die Differenzierung der technischen Ansätze ist mittlerweile zu groß, als dass insbesondere die Anbieter von Plattformen und Diensten ohne plakative Verkürzungen auskommen können, um überhaupt die Aufmerksamkeit von IT-Verantwortlichen auf der Entscheider-Ebene zu erlangen.

Können sich die Entscheider diesem Mechanismus durch grundlegende Skepsis entziehen? Vermutlich nicht: Sie sollten die Auseinandersetzung mit technischen Trends als Kernaufgabe begreifen, wollen sie nicht langfristig Innovationsstau verursachen.

Ist es möglich, neue Technologien bereits frühzeitig zu adaptieren, ohne „tiefe Täler“ voller Enttäuschungen durchschreiten zu müssen? Zum Glück ja: Modelle wie der Hype Cycle spiegeln kollektive Wahrnehmungen wider, nicht individuelle Erfahrungen. Vermeiden lassen sich Enttäuschungen durch eine saubere Abgrenzung der Begriffe, ein klares Verständnis von der Anwendbarkeit auf die eigene Situation und eine frühzeitige Auseinandersetzung mit der Nachweisbarkeit von Erfolgen. Wer als Berater seinem Kunden hilft, sich beizeiten über diese Fragen klar zu werden, gewinnt mit seinen Vorschlägen Glaubwürdigkeit – und tut sich dadurch im Hinblick auf die Nachhaltigkeit der Kundenbeziehung auch selbst einen Gefallen.

Literatur

[1] Fenn, Jackie et al.: Hype Cycle for Emerging Technologies, 2009. <http://www.gartner.com/it/products/hc/hc.jsp>. 07.04.2010

[2] Christof Weinhardt et al.: Cloud-Computing. Eine Abgrenzung, Geschäftsmodelle und Forschungsgebiete, in: Wirtschaftsinformatik 51/2009, pp. 453-462

[3] Foster, Ian, There's Grid in them thar Clouds. <http://ianfoster.typepad.com/blog/2008/01/theres-grid-in.html>. 07.04.2010

[4] Ellermann, Horst: Microsoft-Vize Turner setzt voll auf die Cloud-Karte. <http://www.computerwoche.de/software/office-collaboration/1931892/>. 07.04.2010

[5] Krafzig, Dirk, et al.: Enterprise SOA. Best Practices für Serviceorientierte Architekturen – Einführung, Umsetzung, Praxis, Heidelberg (mitp-Verlag) 2007, pp. 347 ff.

[6] Herrmann, Wolfgang: Die Technik ist zweitrangig. SOA-Governance. <http://www.computerwoche.de/software/soa-bpm/1871524/>. 07.04.2010

[7] Chappell, David: Adding Clarity to the Enterprise Service Bus: An ESB Reference Architecture.

http://www.oreillynet.com/xml/blog/2005/12/adding_clarity_to_the_enterpri.html. 07.04.2010

[8] Chappell, David: Enterprise Service Bus: Theory in Practice, Sebastopol (O'Reilly) 2004

[9] Thomas Manes, Anne: Enterprise Service Bus: A Definition. http://i.i.com.com/cnwk.1d/html/itp/burton_ESB.pdf. 07.04.2010

[10] Fenn, Jackie et al.: Understanding Gartner's Hype Cycles. 2003, Gartner Research Note R-20-1971 (ohne Ortsangabe)

Autor

Nils Tegtmeier ist Chief IT Architect in der Insurance Practice Area und Topic Expert für Enterprise Architecture Management im Kölner Büro der Platinion GmbH.



Nils Tegtmeier
Chief IT Architect
Platinion GmbH
Im Mediapark 4a
50670 Köln
Tel: 0221 / 589 58 310
Fax: 0221 / 589 2051
E-Mail: tegtmeier.nils@platinion.de
Internet: www.platinion.de

Unternehmensskizze

Die Platinion GmbH, im Jahr 2000 gegründet, ist eine hundertprozentige Tochtergesellschaft der Boston Consulting Group (BCG). Das Unternehmen fokussiert seine Beratungsaktivitäten auf die Konzeption und Umsetzung geschäftskritischer IT-Lösungen und ergänzt somit die Strategiekompetenz von BCG um technologische Expertise. Die Betreuung der IT-Projekte umfasst die gesamte Wertschöpfungskette. Platinion arbeitet überwiegend für große Unternehmen und Konzerne, aber auch für bedeutende mittelständische Unternehmen in Deutschland und Europa und zunehmend auch darüber hinaus.

Glossar

Hub-and-Spoke-Prinzip: Typische Topologie von Integrationsarchitekturen mit einer zentralen Daten- oder Nachrichten-Dreh-scheibe anstelle von Punkt-zu-Punkt-Verbindungen zwischen Systemen.

MOM: Message Oriented Middleware

CORBA: Common Object Request Broker Architecture, ein in den Neunziger Jahren spezifizierter Standard für die plattformübergreifende, objektorientierte Integration

EAI: Enterprise Application Integration

SOAP: Ursprünglich als Akronym (Simple Object Access Protocol), seit Version 1.2 als stehender Begriff verwendet; Standard für den Austausch von Daten zwischen Web-Service-Endpunkten

WSDL: Web Services Description Language, Standard-XML-Schema zur Definition von Web-Services

JEE: Java Enterprise Edition

Buzzwords: When consulting gets confusing

Few industries are more prone to fall for buzzwords than IT: Often the public debate on the potential of a new technology leaves participants with only a vague notion of its essentials. Overstretched terms like "Cloud", "SOA" or "ESB" require a reflection of their original core meaning and overall a discussion of their strengths and weaknesses that's more to the point.