

DIE IMPLEMENTIERUNG VON SOA – GRAFISCH, PRAKTISCH, GUT

Das Thema serviceorientierte Architekturen (SOA) beherrscht die Medien und die Planungspapiere der IT-Stabsabteilungen. Laut Wikipedia.com ist SOA „an evolution of distributed computing and modular programming, SOA builds applications out of software services“. Das Konzept ist im Prinzip simpel, allerdings kann die reale Umsetzung sehr komplex werden. Das zugrunde liegende Architekturprinzip sieht eine große Menge von atomaren Diensten vor, die zu anspruchsvollen verteilten Anwendungen kombiniert werden. Es gibt wohl kaum eine bestehende IT-Landschaft, die sich so ohne weiteres nach diesem Prinzip transformieren ließe.

Die Modellgetriebene Entwicklung bietet eine visuelle, auf Industriestandards und etablierten Sprachen aufsetzende Methode für Spezifikation und Management von Service Architekturen. Die Modellierung ermöglicht den Anwendern in einer SOA basierten Umgebung, auf einer höheren Abstraktionsebene zu arbeiten und – dank grafischer Analyse und automatisierter kontinuierlicher Überprüfung des Modells – projektübergreifende Techniken der Qualitätssicherung zu nutzen. Der höhere Abstraktionsgrad macht Details unsichtbar und ermöglicht rollenspezifische Ansichten der beabsichtigten Use Cases und der Funktionalität. Nutzer können die Richtigkeit und Effizienz des angestrebten Designs per Simulation analysieren und testen, um die Konformität zu den Anforderungen zu verifizieren.

Ein Modellgetriebener Ansatz zur SOA Entwicklung versetzt Anwender in die Lage:

- Durch kontinuierliche Überprüfung auf Fehler und Inkonsistenzen in jedem Stadium der Entwicklung eine akzeptable Qualität sicherzustellen und die Kosten zu senken.
- Durch besseres Verständnis der Komponenten und ihrer Schnittstellen die Effizienz verteilter, netzbasierter Anwendungen zu maximieren.
- Durch Reverse Engineering existierender Software, Ergänzung vorhandener Services mit SOA Schnittstellen sowie verbesserte Orchestrierung der existierenden Komponenten den Wiederverwendungsgrad von Services zu erhöhen.
- Durch Simulation die Korrektheit von Änderungen gegenüber früheren Versionen einer Anwendung vor der Freischaltung zu verifizieren.

Geschäft und IT im Gleichklang

Mit der Zunahme an Menge, Größe und Komplexität von Informationssystemen entsteht oft eine Lücke zwischen der Unternehmensstrategie und der Fähigkeit der IT, die Anwendungen zur Implementierung dieser Strategie zu liefern. Die Unternehmensplanung fokussiert sich auf eine umfassende Architektur und ebensolche Geschäftsprozesse, aber die entwickelten Services sind in der Regel isoliert voneinander individuell auf eine spezifische Menge von Anforderungen hin konzipiert und entwickelt worden. Bemühungen, vorhandene Anwendungssoftware für neue oder veränderte operative Anforderungen wieder zu verwenden, waren entweder a priori zu teuer oder lieferten eher zufällige Ergebnisse, die selten den Erwartungen genügten.

Ein anderes Problem ist in dem kontinuierlichen Wandel begründet, dem sich Unternehmen und ihre IT Bereiche heute stellen müssen. Die Zeiten, in denen eine Anwendung für eine Lebensdauer von drei, fünf oder mehr Jahren geplant und realisiert wurde, sind längst vorbei. Änderungen der Unternehmensstrategie, der Anwenderbasis oder der Organisationsstrukturen können dazu führen, dass neue Anwendungen bereits Tage oder Wochen nach ihrer Einführung veraltet sind. Aufkauf oder Zusammenschluss von Unternehmen führt ebenfalls zu Komplikationen, weil überlappende oder widersprüchliche Anwendungen kombiniert oder ersetzt werden müssen – meist mit relativ geringer Vorwarnzeit.

SOA fördert die Wiederverwendung von Software und erhöht deren Relevanz für das Geschäft. SOA stellt sicher, dass die Services



Scott McKorkle

ist Director Product Marketing für die Telelogic Tau Produktlinie. Er verfügt über mehr als 20 Jahre Erfahrung im Markt für Realtime und Embedded Systeme, wo er sich lange Zeit auf Vertrieb und Produktmarketing konzentrierte. In den letzten Jahren hat er sich mit der Entwicklung von Software für Embedded Systeme und der Einführung neuer Technologien im Umfeld integrierter Entwicklungsumgebungen und Realtime-Betriebssysteme befasst. Er studierte Computer Science mit Abschluss Bachelor an der University of Illinois.



Dipl.-Inform. Renate Stücka

ist als Senior Director Business Development bei der Telelogic Deutschland GmbH verantwortlich für die Umsetzung der Unternehmensstrategie in Deutschland, Österreich und der Schweiz. Dazu zählen insbesondere die Erfassung und Berücksichtigung lokaler Gegebenheiten sowie die Kooperation mit lokalen Partnern. Vor ihrem Eintritt bei Telelogic im Jahr 2000 war Frau Stücka mehr als 15 Jahre in verschiedenen Positionen verantwortlich für Marketing, Vertrieb und Entwicklung komplexer Software- und Kommunikationsprodukte. Frau Stücka hat Informatik an der Universität Dortmund studiert.

überall und jederzeit in einer Vielzahl von Kombinationen verfügbar sind – ohne die Einschränkungen früherer Architekturen, bei denen die Hürden zwischen Netzwerken oder Protokollen oft unüberwindlich sind und so die breite Verwendung und Weiterentwicklung behindern.

Service Orientierte Architektur nutzt eine Kombination von Technologien, um die Vorteile der Implementierung verteilter Dienste zu maximieren. Das Herz einer SOA ist die gemeinsame Kommunika-

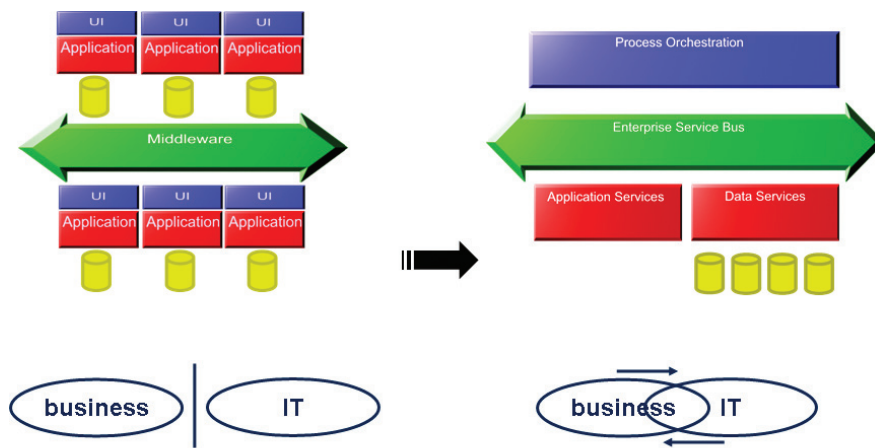


Abbildung 1: SOA bewirkt eine Neuausrichtung des Einsatzes der Service, von einem ad-hoc Ansatz mit isolierten Anwendungen zu einem zentralisierten Ansatz, der Business und IT unter einen Hut bringt.

tionsplattform, welche die Information zwischen den relevanten Anwendungen transportiert. Diese Plattform ist meist das Internet, kann aber auch ein anderes Netz sein, um segmentspezifische Anforderungen zu erfüllen. Die Kommunikationsplattform versetzt Organisationen, Unternehmen oder auch dedizierte Systeme in die Lage, ihre Dienstinfrastrukturen zu integrieren, vorhandene Anwendungen weiter zu verwenden, neue Services hinzuzufügen und die Erwartungen der Anwender adäquat zu erfüllen.

Die Komplexität von SOA ist beherrschbar

Um mit einer SOA Implementierung erfolgreich zu sein, müssen bestimmte Designziele erfüllt werden:

- Fokussierung auf ein Problem, das mit granularen Services gelöst werden kann.
- Etablierung eines Rahmenwerkes für die Umsetzung behördlicher oder anderer Konformitätsauflagen sowie für deren Nachweis, das im Rahmen der SOA Initiative berücksichtigt und mit Details vervollständigt wird.
- Verknüpfung zwischen operativen Anforderungen und den eingesetzten Applikationen, einschließlich vollständiger Traceability bis hin zur Realisierung.
- Berücksichtigung von Sicherheitsaspekten und Identitätsmanagement bei Design, Analyse und Entwicklung, da die vorhandenen Sicherheitsmechanismen oft nicht ausreichen.

- Planung für stringentes Management und Troubleshooting für die gesamte SOA Umgebung, auch wenn sie größer wird.

Das Erreichen dieser Ziele erfordert einen strukturierten Ansatz für Design und Entwicklung, der nicht nur die angestrebte einzelne Anwendung berücksichtigt, sondern das komplette funktionierende System von Services. Dieses System muss für alle Elemente der Architektur Vorkehrungen treffen, einschließlich der beabsichtigten Nutzer und der Standard-Services, die von der Anwendung angezapft werden.

In der Luft- und Raumfahrt oder im Transportwesen ist der Nutzen der Modell-

getriebenen Entwicklung in zahlreichen Programmen und Projekten nachgewiesen, unter anderem für Satellitenkontrollsysteme, Signalisierungssysteme und sogar für Flugzeugträger. Dieser visuelle Ansatz zu systemorientiertem Design und Entwicklung ist die nahe liegende Wahl für SOA. Die Aufgabe ist durchaus vergleichbar: geht es in dem einen Fall darum, ein großes System bestehend aus vielen interagierenden Subsystemen und Komponenten zu entwerfen, so gilt es bei der SOA eine Umgebung aufzubauen, in der vorhandene oder neu aufgeteilte und zusammengestellte Dienste zu einer spezifischen Menge an synchronisierten Komponenten kombiniert werden. Diese müssen iterativ entwickelt, analysiert und getestet werden, angefangen von der abstrakten Funktionalität bis hinunter zur Implementierung im Detail.

Unter anderem muss SOA auch eine Management-Sicht bieten – einschließlich Risikomanagement. Die Abhängigkeit von wenigen Anbietern für besonders wichtige Services stellt ein potenzielles Risiko dar. Ein Weg dieses Risiko abzuschwächen, ist die möglichst allgemeine Spezifikation von Services, was jedoch andererseits die Performance beeinträchtigen kann. Analyse und Lösung des Konfliktes zwischen Risiko-Abschwächung und Performance-Zielen sind zentrales Element der Modellgetriebenen Entwicklung.

Implementierung der SOA

Soweit klingt das alles sehr gut, aber was passiert, wenn ein Unternehmen eine SOA

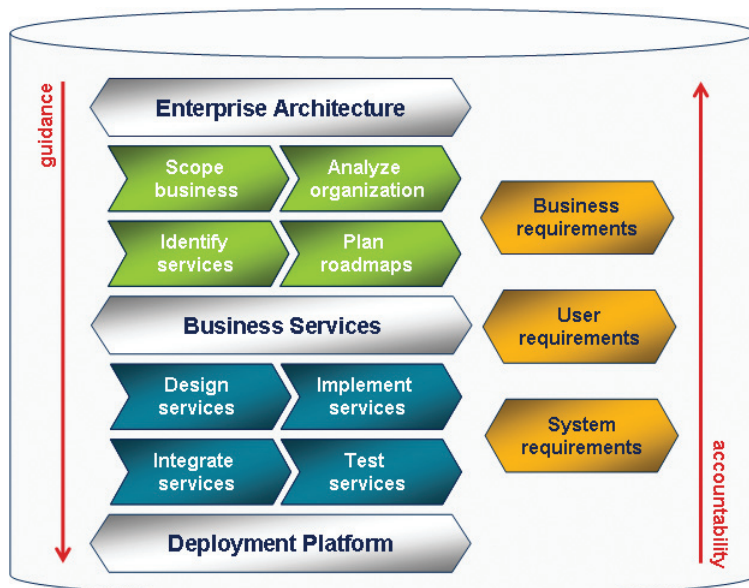


Abbildung 2: Enterprise Workflow der Modellgetriebenen SOA



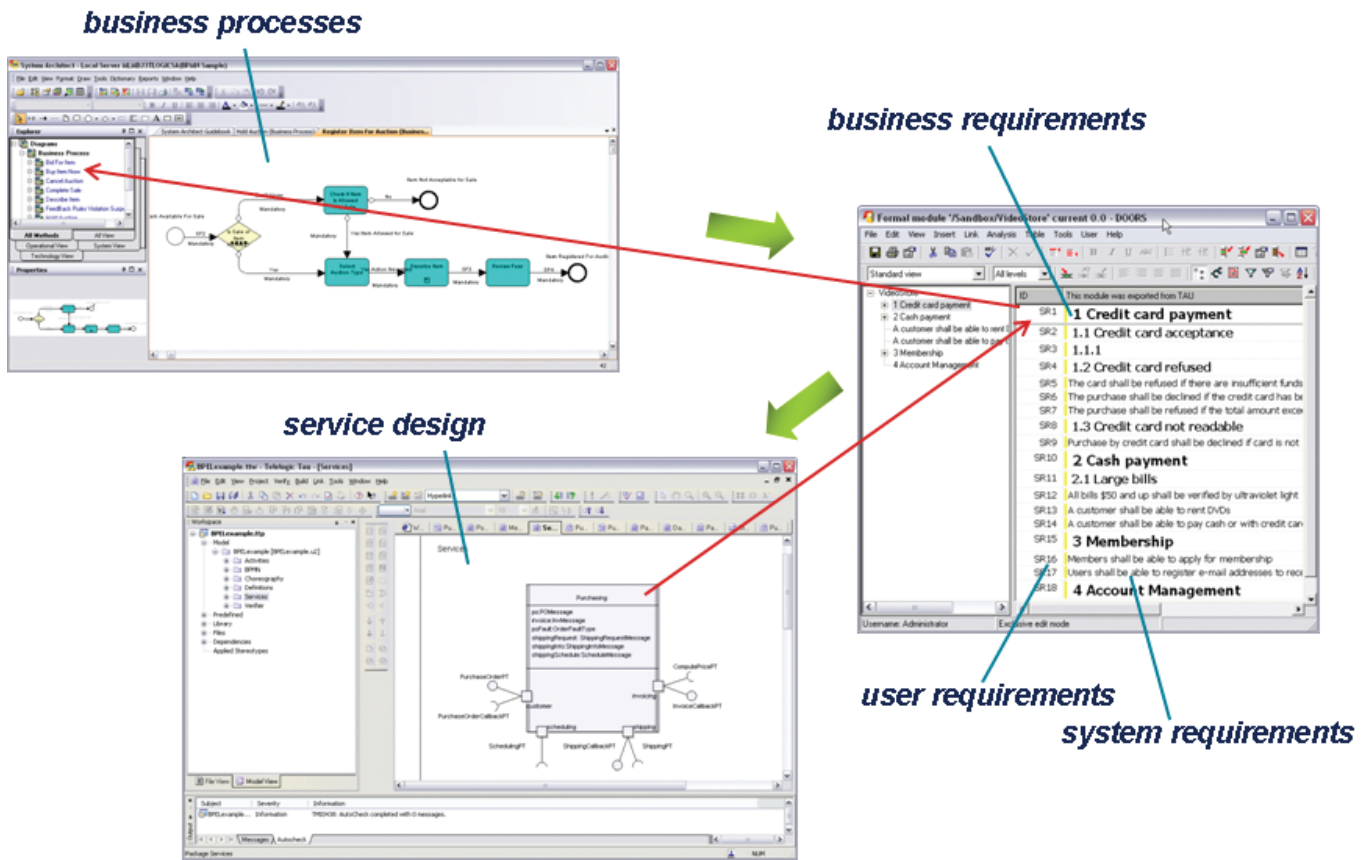


Abbildung 3: Verantwortlichkeiten können mit Hilfe des "audit trails" durch den gesamten Entwicklungszyklus verfolgt werden. Die grünen Pfeile zeigen den Workflow, die roten Pfeile die Verknüpfungen.

Umgebung einführen möchte? Die Herangehensweise kann individuell variieren; die wesentlichen Schritte sind jedoch immer die folgenden:

- Identifikation spezifischer Services für Prozesse und Funktionen
- Entwicklung und Einführung der benötigten Dienste
- Kombination der Dienste zu einer Applikation
- Administration und Pflege der Applikation

Ausgereifte Umgebungen für Modellgetriebene Entwicklung wie Telelogic Tau können SOA Initiativen unterstützen durch einen konsistenten visuellen Ansatz für Analyse, Design, Entwicklung und Wartung. Durch Integrationen mit Lösungen für Enterprise Architecture, Business Process Management und Anforderungsmanagement wie Telelogic System Architect und Telelogic DOORS kann sogar das gan-

ze Unternehmen in die SOA Initiative einbezogen werden.

Wenn die Fachabteilung eine Anwendung spezifiziert hat, wird Tau eingesetzt um die notwendigen Dienste für diese Anwendung sowie die Anwendung selbst zu konzipieren, zu integrieren und zu implementieren. Durch die Transformation textueller Anforderungen in visuelle Darstellungen können die Entwickler in Tau die durch die Fachabteilung formulierte Anwendungsbeschreibung als Basis für ein Modell nutzen, welches Funktionen und Verhalten der anzuwendenden Dienste repräsentiert. Die Ergebnisse der initialen Analyse sind in zweifacher Hinsicht hilfreich:

1. Für eine Entscheidung über die Wirtschaftlichkeit des Projektes: fortsetzen, abrechnen oder modifizieren.
2. Für die Entwicklung eines Designmodells, welches die Anforderungen berücksichtigt und eine funktionierende Lösung definiert.

Eine bidirektionale Verknüpfung von Modellelementen mit den Anforderungen ermöglicht die Verfolgung der Realisierung über den ganzen Prozess hinweg, wobei auch Änderungen an Anforderungen in kontrollierter Weise eingepflegt werden.

Nachdem man ein präzises Verständnis der grundsätzlichen Herangehensweise entwickelt hat, wird das Analysemodell iterativ verfeinert bis zu einem detaillierten Designmodell. Dieses kann in jedem Stadium der Entwicklung simuliert werden, um die korrekte Funktionalität zu überprüfen. Diese steht zunächst im Vordergrund; daher repräsentiert das Designmodell die angestrebte Applikation als Ganzes. Erst nachdem man die Aufgabenstellung auf dieser Ebene komplett durchdacht und verstanden hat, geht man in der nächsten Phase daran, einzelne Service-Module (oder Komponenten) zu einer vernetzten verteilten Applikation zusammen zu stellen.

Um Wiederverwendung zu fördern und den Entwicklungsaufwand zu verringern,

kann der Entwickler verschiedene Service Kandidaten betrachten, sowohl SOA konforme als auch früher entstandene, und analysieren, ob sie für die Implementierung in Betracht kommen. Oft füllen diese Kandidaten viele Teile der Anwendung; Lücken werden dennoch bleiben. Um diese zu füllen, werden Designmodelle für die fehlenden Elemente generiert. Wenn alle erforderlichen Services zur Verfügung stehen, werden sie mit Hilfe von Tau zu einer orchestrierten und choreografierten Applikation zusammengefügt, welche konform ist zu den ursprünglich formulierten Anforderungen.

Für neue oder angepasste Services werden auf Basis der Details im Designmodell SOA-konforme WSDL und XSD Schnittstellen generiert. Erst wenn die Anwendung zufriedenstellend assembliert ist, werden die plattformspezifischen Details hinzugefügt, so dass die konkrete Einführung vorbereitet werden kann. Im Falle von neu entwickelten Service-Komponenten kann der Quellcode aus dem Designmodell generiert werden.

Bis jetzt fand die ganze Designarbeit im Modell statt. Nachdem die Implementierung abgeschlossen ist, wird die Anwendung auf dem Server oder dem Zielsystem installiert. Von diesem Zeitpunkt an wird die Anwendung über ihren ganzen Lebenszyklus überwacht und ihre Eignung und Effizienz kontinuierlich evaluiert. Laufzeitdaten werden mit den operativen Vorgaben abgeglichen und aktualisierte Business-Pläne können auf ihre Auswirkungen auf diese Applikation und deren Anwender hin untersucht werden. Zukünftige Anforderungen können ganzheitlich betrachtet werden, wobei neue oder aktualisierte Anforderungen der Fachabteilungen mit aktuellen Fähigkeiten und laufendem Wartungsbedarf abgestimmt werden.

Fazit

Nach Wikipedia.de liegt dem Prinzip SOA ein duales Begriffsmodell zugrunde: „Das Managementkonzept strebt eine an den gewünschten Geschäftsprozessen ausgerichtete Infrastruktur an, die schnell auf

veränderte Anforderungen im Geschäftsumfeld reagieren kann. Das Systemarchitekturkonzept sieht die Bereitstellung fachlicher Dienste und Funktionalitäten in Form von Services vor, die atomare Prozess-Schritte abbilden.“

Die hier vorgestellte Herangehensweise genügt beiden Dimensionen der SOA: Die Fachabteilungen definieren die Anwendungen, die sie für die Ausführung ihrer Geschäftsprozesse brauchen, und die IT bedient sich einer Entwicklungsmethodik, die sich bereits vielfach bei Design und Entwicklung komplexer Systeme von interagierenden Komponenten bewährt hat. Die Kombination und Integration von Werkzeugen wie Telelogic DOORS, Tau und System Architect sorgt für eine durchgängige Umgebung für die SOA Initiative, von der alle beteiligten Gruppen im Unternehmen gemäß ihren Aufgaben und Verantwortlichkeiten profitieren können. Praktische Beispiele werden unter anderem in einem kostenfreien Seminar bei Telelogic vorgestellt (www.telelogic.com/soa_praktisch).

