

SOA mal evolutionär, mal revolutionär

Bauplan mit viel Flexibilität



Eines der wesentlichen Merkmale einer serviceorientierten Architektur:
Sie muss flexibel an häufig wechselnde Anforderungen anpassbar sein.
Die Grundlage, um SOA-Geschäftsprozesse auch effizient steuern zu können,
sind die passenden Basistechnologien.

SOA ist beides, evolutionär und revolutionär. Die Wurzeln einer Software-Architektur, die leicht zu warten ist und sich in geordneten Bahnen an neue Gegebenheiten akklimatisieren kann, reichen weit zurück. Wer hier an Konzepte wie strukturierte und objektorientierte Programmierung denkt, liegt vollkommen richtig. Bekanntermaßen haben sich jedoch viele der damit verbundenen Vorstellungen nicht erfüllt. Mit SOA soll nun alles anders werden, und die Ausgangslage dafür ist recht gut. Im Gegensatz zu früheren Phasen stehen heute ausgereifte, plattformunabhängige Technologien beispielsweise Web-Services in vielerlei Ausprägungen, Messaging-Lösungen wie ESBs und XML-Varianten, aber auch neuere Ansätze wie AJAX (Asynchronous JavaScript and XML) und RIA (Rich Internet Application) bereit. AJAX verwendet bekannte Technologien wie HTML und JavaScript, um interaktive, Desktop-ähnliche Web-Anwendungen zu erstellen.

SOA oder nicht SOA? Um keine Missverständnisse aufkommen zu lassen: SOA ist nicht das Allheilmittel für alle nur denkbaren IT-Umgebungen und Einsatzgebiete von Software. Rein interne Applikationen in einer homogenen Landschaft, bei denen alle Applikationen von einem Anbieter kommen, sind für SOA wenig geeignet. Hier reichen die vom jeweiligen Softwarehersteller bereit gestellten Bordmittel völlig aus. Gleiches gilt beispielsweise für eng miteinander gekoppelte Batch-Programme. Während SOA vom Prinzip her auf lose miteinander verknüpfte Applikationen setzt, um so möglichst flexibel zu sein, erweist sich bei autonom ablaufenden Programmsequenzen gerade die starke Verzahnung als Vorteil. Die Konstanz und Stabilität sind hier wesentliche Voraussetzungen für eine hohe Performance.

Ähnlich verhält es sich, wenn die Applikationen, die zugrunde liegende Geschäftslogik, die Daten oder Prozesse nur selten geändert werden. Eingriffe sind nur dann zu rechtfertigen, wenn die Einbettung in eine serviceorientierte Architektur deutliche Effizienzge-

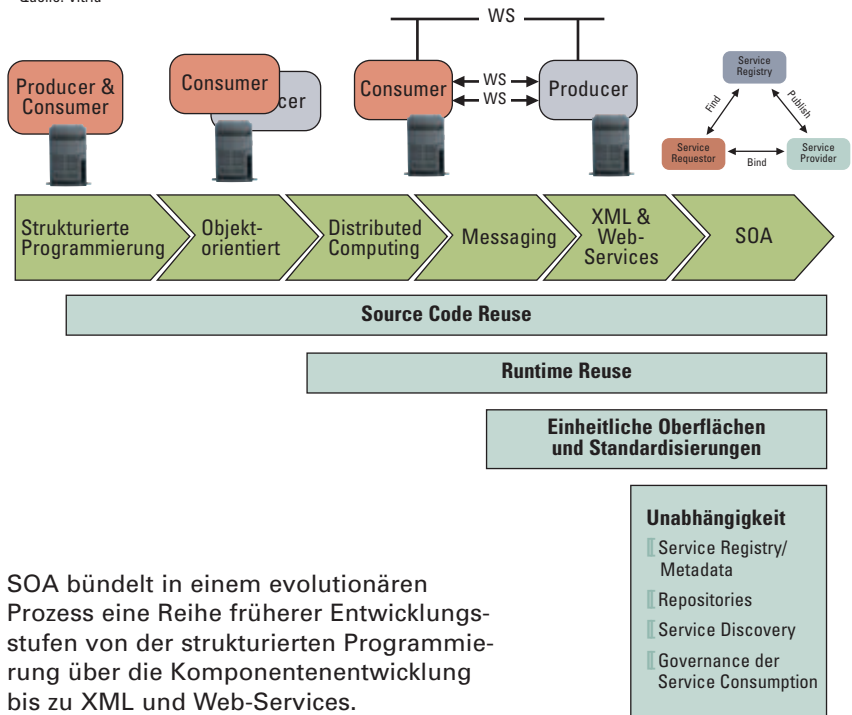
winne in den Prozessen verspricht. Verbleiben als letzte, für SOA nicht geeignete Szenarien, zusammengefasste Applikationen, in denen zeitnah – manchmal tatsächlich in Echtzeit – Daten erfasst, verarbeitet und analysiert werden. Mit einer lose gekoppelten Architektur wie sie für SOA typisch ist, lassen sich Antwortzeiten im Bereich von Millisekunden nicht erreichen. Damit ist klar: Realtime-Umgebungen und SOA sind nicht füreinander gemacht. Die hier skizzierten Einsatzgebiete, bei denen SOA eher schadet denn nützt, führen in der IT-Landschaft der Unternehmen ein Nischendasein. Sie werden

Architekturmodell zur effizienten Gestaltung komplexer, aus vielen Einzelteilen bestehender IT-Infrastrukturen. Dieser umfassendere Ansatz lässt sich an Hand einiger Kernmerkmale beschreiben:

- ▮ Von einer Softwareeinheit (einem Service) existiert jeweils nur eine Instanz.
- ▮ Services sind lose miteinander gekoppelt. Damit sie mit anderen Komponenten möglichst effizient zusammenarbeiten können, müssen sie »eigenständig« sein.
- ▮ Der Nachrichtenaustausch zwischen Services sollte so einfach wie

Das SOA-Paket

Quelle: Vitria



SOA bündelt in einem evolutionären Prozess eine Reihe früherer Entwicklungsstufen von der strukturierten Programmierung über die Komponentenentwicklung bis zu XML und Web-Services.

zur Lösung von Spezialaufgaben benötigt, bilden jedoch eher eine Minderheit im Applikationsportfolio bei Banken, Versicherungen und Fertigungsunternehmen mit einem weit verzweigten Netz von Lieferanten und Partnern, wie es für die Automobilindustrie oder den Maschinenbau charakteristisch ist.

Charakteristische SOA-Eigenschaften. Die Abgrenzung ist das Eine, die Stärken von SOA das Andere. Dabei ist SOA keine Technologie, sondern ein

möglich sein und XML-Dokumente und XML-Schemata nutzen. XML ist deutlich flexibler als herkömmliche Programmaufrufe via Remote Procedure Calls.

- ▮ Services sollten sowohl synchrones als auch asynchrones Messaging beherrschen, um auch lang laufende Geschäftsprozesse zu unterstützen.

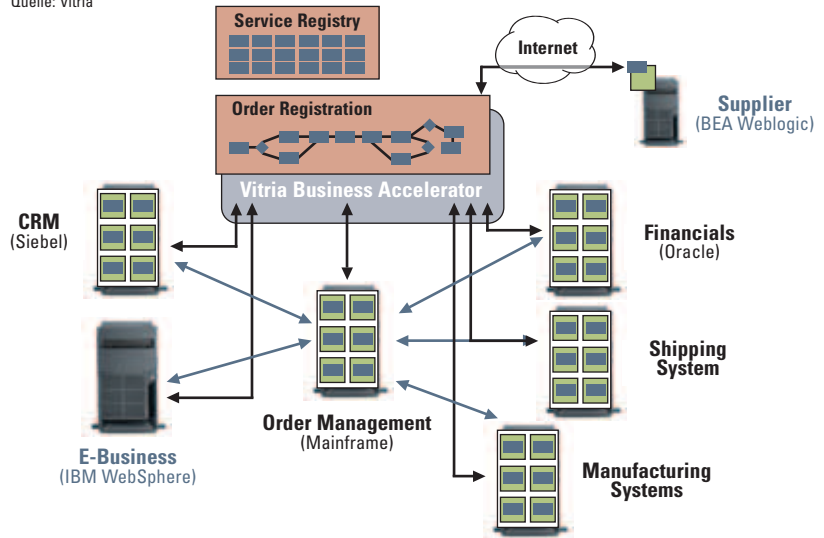
Die Stärken von SOA liefern zugleich die Kriterien, um zu entscheiden, für welche Einsatzgebiete sich SOA eignet.

Insbesondere lang laufende Geschäftsprozesse, bei denen eine Vielzahl von Lieferanten und Geschäftspartnern eingebunden sind, etwa bei der Auftragsfertigung im Anlagen- und Maschinenbau, profitieren von asynchronem Messaging. Die auftragsneutrale Disposition von Bauteilen erfolgt lediglich für die Standardkomponenten, alle anderen Module werden mit den unterschiedlichsten Applikationen entwickelt, geplant und gefertigt. Charakteristisch für derartige Geschäftsprozesse sind vielfältige Applikationen, die in den verschiedensten Fachabteilungen eines Unternehmens, aber auch bei Geschäftspartnern und Lieferanten, zum Einsatz kommen. Häufig wickeln solche Applikationen ein beachtliches Transaktionsvolumen bei gleichzeitiger hoher Integrationskomplexität ab.

Trotz aller Automatisierung der internen und unternehmensübergreifenden Prozesse ist hier in Ausnahmefällen ein manuelles Eingreifen erforderlich. Knapp kalkulierte Liefer- und Durchlaufzeiten lassen sich nur einhalten, wenn rollengebunden bestimmte Mitarbeiter ereignisgesteuert und orchestrierend in den Ablauf des Order-Managements eingreifen können. Genau genommen wird an der Stelle die klassische SOA, wie sie für viele E-Commerce-Szenarien, Online-Banking, CRM-Applikationen und ähnlichem eingesetzt wird, durch neue Technologie wie Event-Processing erweitert. Derartige ereignisgesteuerte SOA-Komponen-

Die Orchestrierung

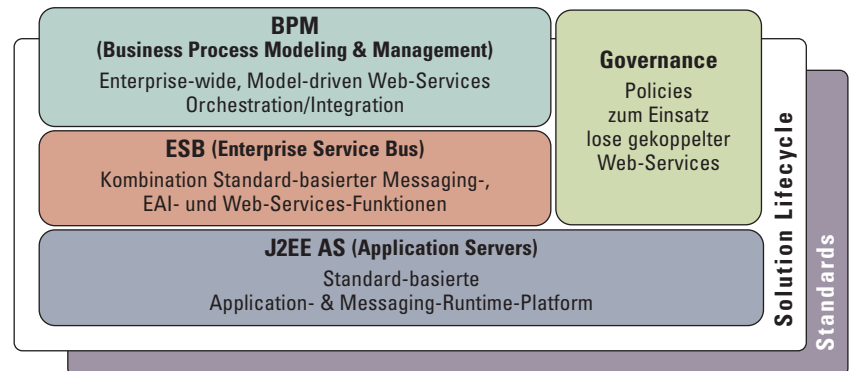
Quelle: Vitria



Die Auftragssteuerung (Order Orchestration) in einer serviceorientierten Architektur integriert Kunden, Lieferanten und Logistikdienstleister.

Die SOA-Suite

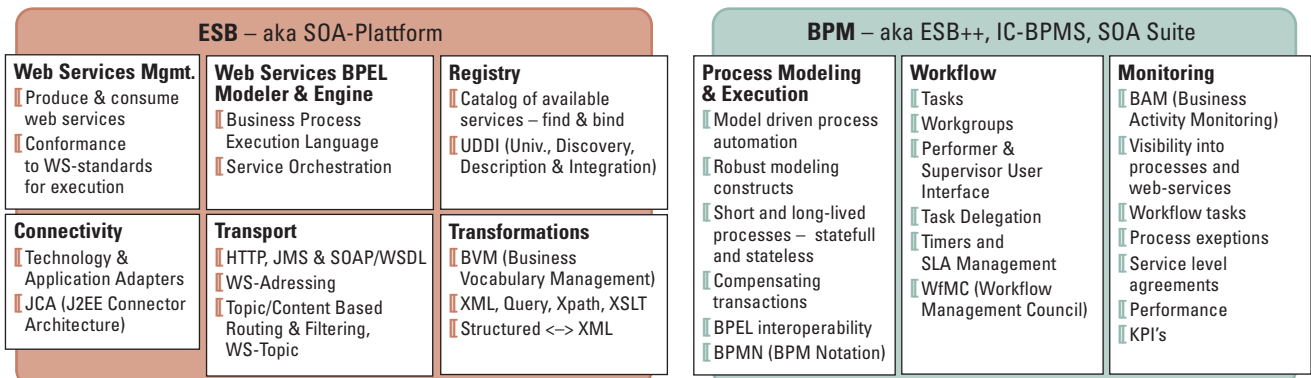
Quelle: Vitria



Auf einen Blick: Die wichtigsten Elemente einer SOA Integration Suite.

ESB und BPM

Quelle: Vitria



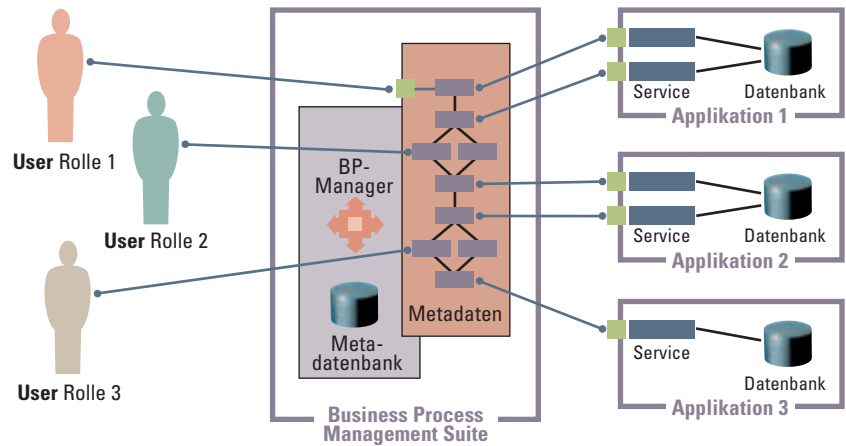
Enterprise Service Bus (ESB) und Business Process Management (BPM) unterstützen die Steuerung von Web-Services und schaffen die Basis für eine möglichst effiziente Gestaltung von Geschäftsprozessen.

ten eignen sich sehr gut zur Unterstützung lang laufender Transaktionen, bei denen im zeitlichen Ablauf von Geschäftsprozessen wie einer Auftragsbearbeitung eine Vielzahl von Events zu analysieren und miteinander zu verknüpfen sind. Als Fachbegriff hat sich an der Stelle Business Event Management, kurz BEM, etabliert.

Serviceorientierte und ereignisgesteuerte Architekturen. Spätestens an der Stelle ist deutlich, warum SOA im Vergleich zu älteren komponentenbasierten Architekturen deutlich mehr bietet. Die zugrunde liegenden Technologien – und hier speziell die neueren Entwicklungen beim Business Event Management – stellen die notwendigen Werkzeuge und Funktionen für eine flexible Abstimmung von Geschäftszielen und IT-Infrastrukturen bereit. Exemplarisch für eine derartige Lösung ist Business Accelerator von Vitria. In ihren SOA-Projekten können Anwender mit Business Accelerator

Die menschliche Komponente

Quelle: Vitria



SOA ist mehr als die reine Automation von Geschäftsprozessen. Auch manuelle Eingriffe haben darin ihren Platz.

sehr flexibel Business-Process-Integration-Software, Web-Services, Events, vorhandene unternehmenskritische Applikationen, aber auch manuelle Workflows integrieren.

Der Unterschied zu den Big-Bang-Ansätzen all umfassender Integrationsplattformen, wie sie für die technisch fokussierte Enterprise Application Integration typisch waren, ist klar er-

Glossar wichtiger Standards im SOA-Umfeld

AJAX: Asynchronous JavaScript and XML, kurz AJAX, beschreibt ein Konzept zur asynchronen Datenübertragung zwischen einem Server und dem Browser; AJAX besteht aus einer Kollektion von quelloffenen User-Interface-Elementen, mit denen sich Rich Internet Applications (RIAs) erzeugen lassen

BAM: Business Activity Monitoring

BEM: Business Event Management, gehört in die Kategorie Event Driven Applications

BPEL: Business Process Execution Language, XML-Definition für Ablaufsteuerung von Web-Services

BPM/BPMN: Business Process Management Notification, eine grafische Darstellung von Geschäftsprozessen in hoheitlicher Verantwortung des BPM Institute (www.bpmi.org)

EDA: Event Driven Architecture, ein neuer Terminus für asynchrone, Messaging-basierte Applikationen

ESB: Enterprise Service Bus, enthält bedeutende SOA-Komponenten einschließlich Connectivity, Transport, WS-Management, Registry

FLEX: RIA-Implementierung von Adobe

JAX-RPC: Java API für XML-basierte Remote Procedure Calls

JCA: J2EE Connector Architecture – Standardspezifikation zur Interoperabilität mit externen Systemen (Konnektoren und Adapter)

JMX: Java Management Extensions

ORM: Object Relational Mapping, ermöglicht die Abbildung von Objekten in relationalen Datenbanken (und umgekehrt)

RIA: Rich Internet Application, ermöglicht eine grafische Benutzeroberfläche mit Steuerelementen im Browser, RIA-Beispiele sind AJAX und Adobe Flex

SOAP: Simple Object Access Protocol, ein kompaktes XML-Protokoll, um Web-Services über HTTP aufzurufen

UDDI: Universal Description Discovery and Integration, ein Standard, mit dem Anwender unter anderem Softwarekomponenten zum Einsatz in einer serviceorientierten Architektur finden und publizieren können

WS-Addressing: Kommunikation von Adressinformationen beim Verbindungsaufbau

WS-Basic Profile: Grundlegende Handlungsanweisungen, um interoperable Infrastrukturen mit Hilfe von SOAP, WSDL, XML und HTTP zu erstellen

WS-Policy: Eine flexible, erweiterbare Grammatik, um Service Level Agreements in WS-Security zu erstellen

WS-Security: Spezifikation zur Sicherstellung der Integrität und Vertraulichkeit (einschließlich Ver- und Entschlüsselung) bei der WS-Interoperabilität

WS-Topics: Definiert die Kategorisierung und Organisation von Events, eng verwandt mit WS-Notification und WS-Eventing

WS-Transaction: Spezifikationen zur Koordination einer Vielzahl einzelner Aktionen

XSLT: eXtensible Style-sheet Transformation Language

kennbar. Kompakte SOA Integration Suites wie Business Accelerator gehen davon aus, dass die Anwender die einzelnen Bausteine wie ESB und Business Process Management zunächst für überschaubare, aber dennoch sehr wichtige SOA-Vorhaben einsetzen und dann nach der Devise »start small, grow fast«, die Reichweite der Lösung unternehmensweit ausdehnen. Die SOA Integration Suite schafft damit die Freiheit und Flexibilität, um Best-of-Breed-SOA-Lösungen zu erstellen. Skalierbarkeit, Offenheit und Infrastrukturunabhängigkeit werden so zu den bedeutenden Anforderungen, die eine SOA Integration Suite erfüllen sollte.

Offenheit und Plattformunabhängigkeit kommen beispielsweise auch darin zum Ausdruck, dass zur Model-

lierung von Geschäftsprozessen ein Open-Source-Framework wie Eclipse unterstützt wird. Die mit Business Accelerator erstellten Lösungen arbeiten mit den gängigen J2EE-Application-Servern zusammen. Dazu trägt auch das Java-Fundament von Eclipse selbst bei. Die Offenheit einer SOA Integration Suite zeigt sich schließlich darin, ob die marktgängigen JMS-Messaging-Plattformen unterstützt werden. Damit lassen sich auch vorhandene Infrastrukturen in SOA einbinden.

Die hier erwähnten Kriterien einer State-of-the-Art-Integrationslösung verdeutlichen, dass SOA weit mehr ist als Komponenten, Wiederverwendbarkeit und Web-Services. Insbesondere die Einbeziehung ereignisgesteuerter Architekturen – einschließlich BAM, Business Activity Monitoring – er-

weitert das Einsatzspektrum auf Supply-Chain- und Order-Management-Anwendungen, für die bislang mühsam manuelle Ausnahmeregelungen erstellt werden mussten. Die technische SOA-Integration ist notwendig, aber nicht hinreichend. Sie muss um Funktionen ergänzt werden, mit denen sich Geschäftsprozesse ereignisorientiert steuern lassen. Um den Transfer der Geschäftsdaten, beispielsweise durch Object Relational Mapping, und die Abläufe koordinieren und deren reibungslosen Workflow überwachen zu können, wird zusätzliche Business-Process-Logik benötigt – oder allgemeiner: Business Process Management.

Thomas Egeling

Thomas Egeling ist
Presales Consultant bei Vitria Technology
in der Geschäftsstelle München.