Vorlesung am 14. Juli 2009

Serviceorientiertes E-Government

Marktgängige Produkte und Ansätze
Fazit zur Vorlesung

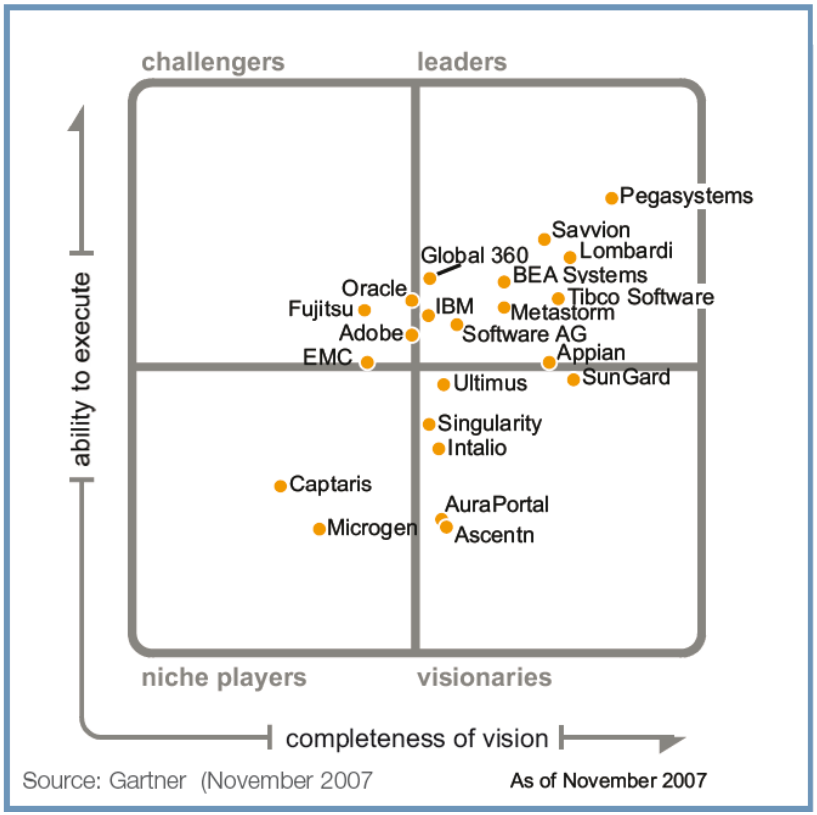
Dr. Frank Sarre
Lehrbeauftragter der LMU München

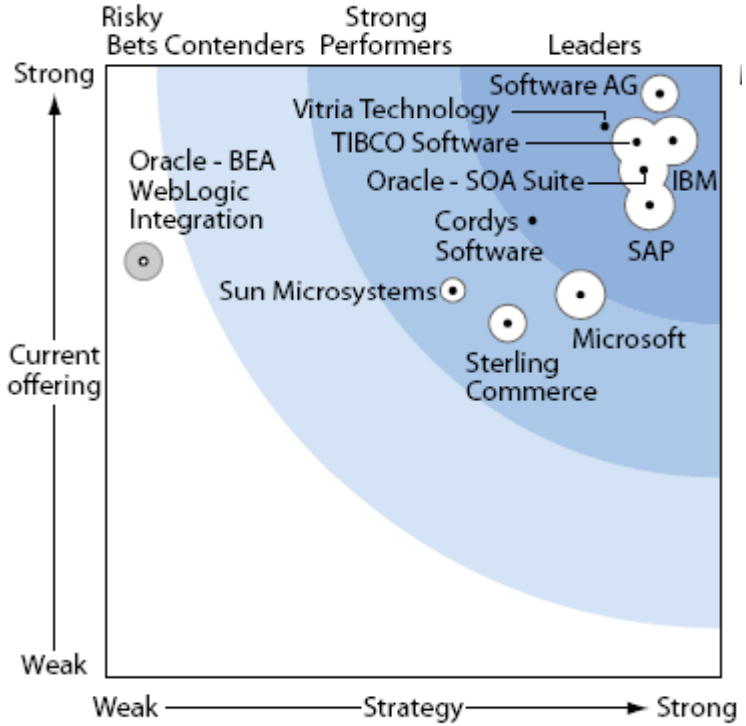
Hauptproduktbereiche von „SOA-Stacks“

- Enterprise Service Bus
- Verzeichnisdienste / Repository
- Business Process Management
- Monitoring / Betrieb
- Identity- und Accessmanagement

Studien von Gartner und Forrester [1]

Business Process Management



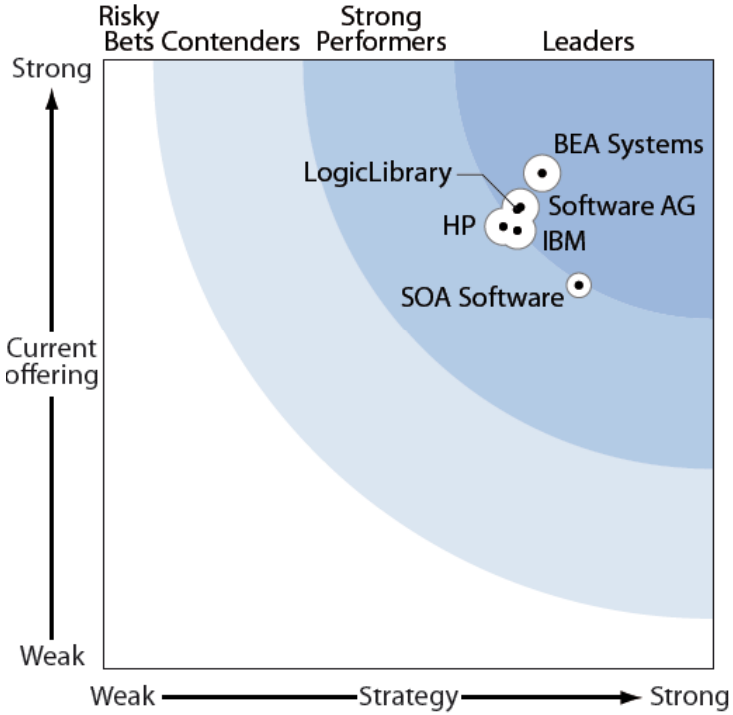


Quelle: Forrester Wave, Q4 2008

Studien von Gartner und Forrester [2]

Integrated SOA Governance

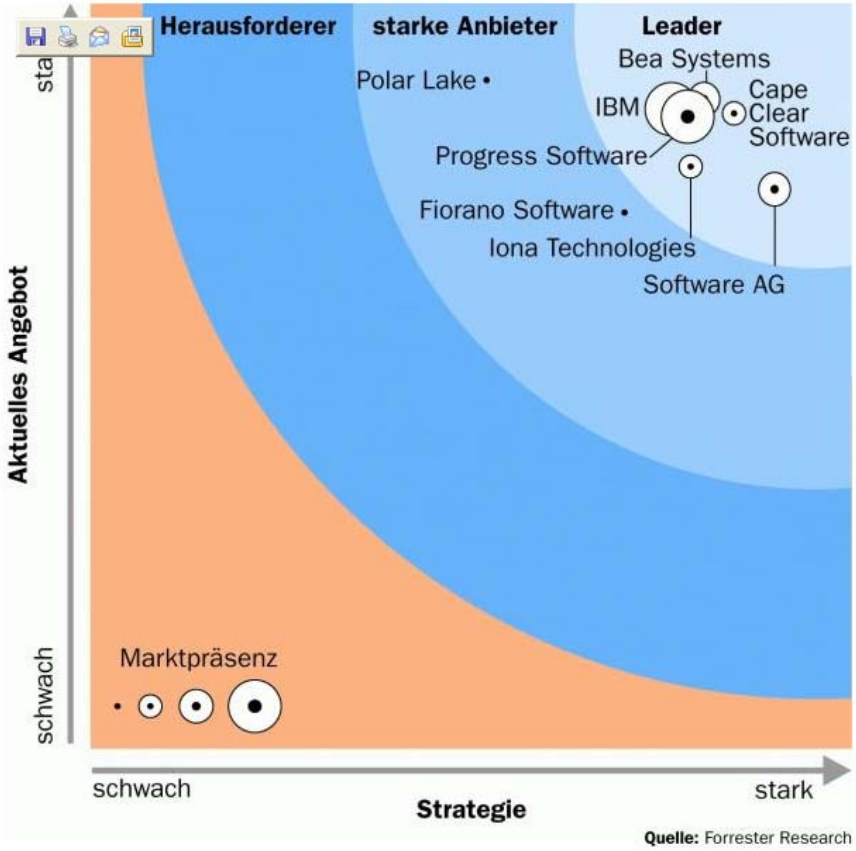
SOA Service Lifecycle Management



Quelle: Forrester Wave-Studie, Q1 2008

Studien von Gartner und Forrester [3]

Enterprise Service Bus



Marktteilnehmer [1]

Erkenntnisse:

- Die Studienergebnisse zeigen einige Anbieter, die in fast allen Bereichen unter den „**Leadern**“ sind

  → Angebot einer SOA-Lösung „aus einer Hand“ möglich

  → Jedoch: Nicht in allen Bereichen die beste Lösung

- In einzelnen Aufgabenbereichen gibt es Spezialisten, die eine „Best-of-Breed“-Lösung liefern könnten

  → Aufgrund von offenen Standards sollte die Integration solcher Lösungen möglich sein

- Neben den Herstellern treten auch Softwareberatungshäuser als Marktteilnehmer auf, die teils herstellerabhängig, teils herstellerunabhängig sind

Marktteilnehmer [2]

- Anbieter, die das komplette Spektrum oder einen großen Teil aller Funktionalitäten anbieten:
  - IBM
  - ORACLE
  - Software AG
  - SAP
  - TIBCO
  - Microsoft
- Beispiele spezialisierter Anbieter:
  - BPM: Pega
  - ESB: IONA, Progress
  - Repository / Governance: Amberpoint, SOA Software, HP

SOA Suiten: IBM [1]

IBM

- Umfangreiches Portfolio von Anwendungen aller Art bereits seit langem vorhanden
  - Modellierung
  - Implementierung
  - Betrieb
- Unterschiedliche Produktfamilien werden kombiniert bzw. werden konsolidiert
  - Rational
  - Websphere
  - Tivoli
- Angebot eines Produktportfolios von SOA-Produkten unter dem Label „**SOA Foundation**“

SOA Suiten: IBM [2]

IBM SOA Foundation

Modellierungsphase

- WebSphere Business Modeler
- Rational Software Architect

„Zusammenstellungsphase“ (Begriff von IBM)

- WebSphere Integration Developer
- Rational Application Developer
- Lotus Component Designer
- Lotus Forms
- Lotus Domino Designer
- WebSphere Portlet Factory
- Rational Tester for SOA Quality

SOA Suiten: IBM [3]

IBM SOA Foundation (Fortsetzung)

Implementierungsphase

- WebSphere DataPower SOA Appliances
- WebSphere Process Server
- WebSphere ESB
- WebSphere Message Broker
- WebSphere Partner Gateway
- WebSphere Adapters
- WebSphere Portal
- InfoSphere Information Server
- WebSphere Application Server
- WebSphere Extended Deployment
- WebSphere Business Services Fabric
- Lotus Expeditor
- FileNet P8 Platform

SOA Suiten: IBM [4]

IBM SOA-Foundation (Fortsetzung)

Betrieb

- WebSphere Business Monitor
- Tivoli Composite Application Manager for SOA
- Tivoli Composite Application Manager for WebSphere
- Tivoli Identity Manager
- Tivoli Access Manager
- Tivoli Federated Identity Manager
- Tivoli Provisioning Manager

Governance und Best Practices

- WebSphere Service Registry and Repository

SOA Suiten: ORACLE [1]

ORACLE

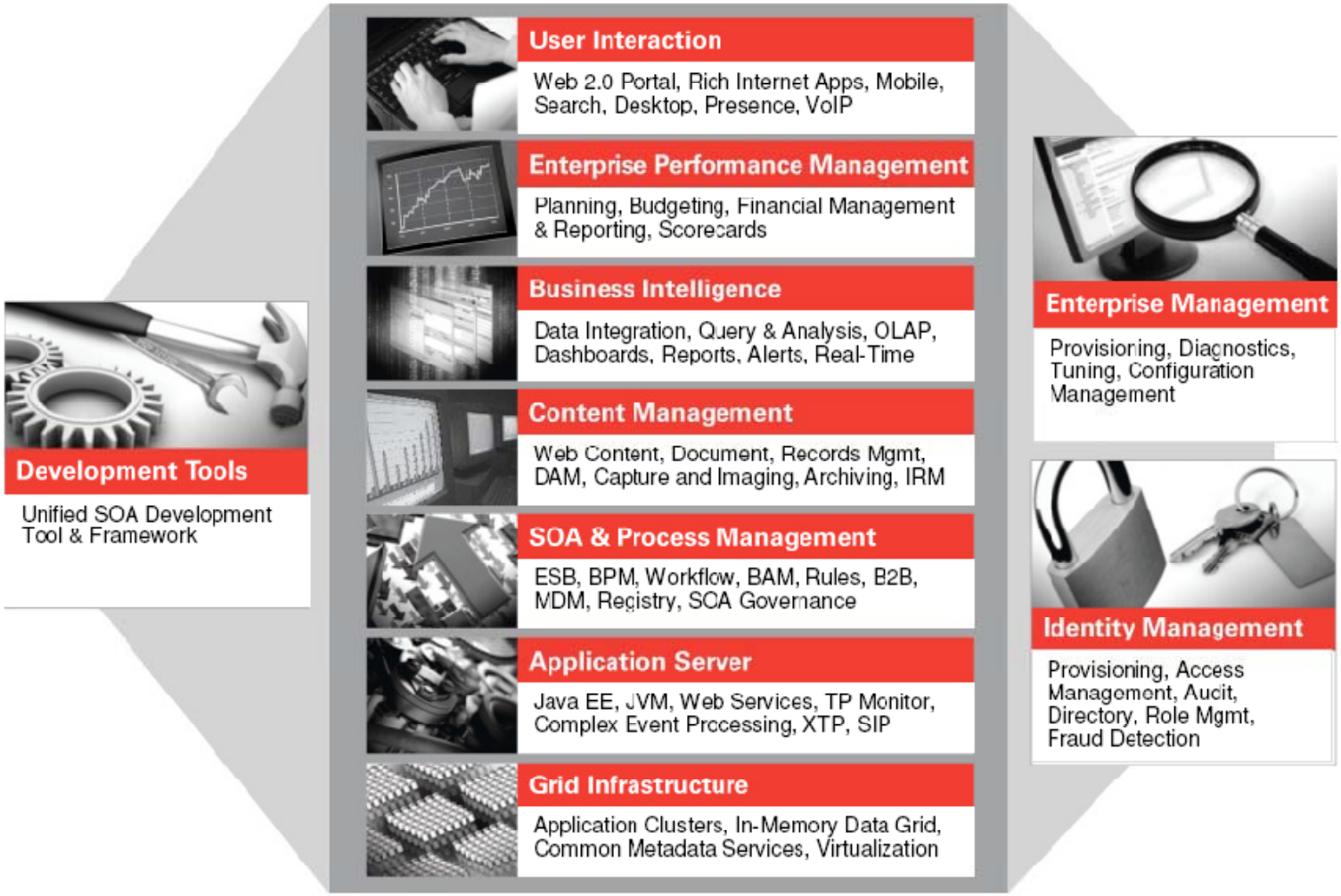
- ORACLE hat die eigene Marktposition im Segment „SOA-Technologien“ mit dem Zukauf von BEA wesentlich verstärkt

- Die ORACLE **SOA Suite** ist eine Sammlung der typischen Anwendungen im SOA-Bereich:
  - Oracle JDeveloper 10g (Implementierung)
  - Oracle Business Rules (Implementierung)
  - Oracle BPEL Process Manager (Modellierung, Implementierung)
  - Oracle Enterprise Service Bus (Integration)
  - Oracle Business Activity Monitoring (Betrieb)
  - Oracle Web Services Manager (Betrieb)

- Die Suite integriert sich in die **ORACLE Fusion Middleware**

SOA Suiten: ORACLE [2]

ORACLE Fusion Middleware



SOA Suiten: ORACLE [3]

ORACLE (Fortsetzung)

SOA-Governance-Tools

- Oracle Enterprise Repository (BEA)
- Oracle Web Services Manager
- Oracle Enterprise SOA Management Pack
- Oracle Service Registry (UDDI V3-Registry)

Offensichtliche Strategie der Firma ORACLE

- Akquisition von marktführenden Spezialprodukten, z.B. BEA

  Problem: Integration unterschiedlicher Produkte

  Integration von BEA jedoch innerhalb eines halben Jahres gut gelöst

SOA Suiten: Software AG [1]

Software AG

- Bisheriges Geschäftsfeld „Enterprise Transaction Systems“
  mit der Datenbank *Adabas* und der Programmiersprache *Natural*
- Neue Ausrichtung auf SOA mit dem Kauf der *webMethods-Suite:*
  - webMethods CAF (Content Application Framework)
    (Entwicklung von Web-Oberflächen und Portalen)
  - webMethods ESB
    (Integration und Kommunikation)
  - webMethods BPM
    (Modellierung mit BPMN, nicht BPEL)
  - webMethods BAM (Business Activity Monitoring)
    (Monitoring und Auswertung aller Systemfunktionen)

SOA Suiten: Software AG [2]

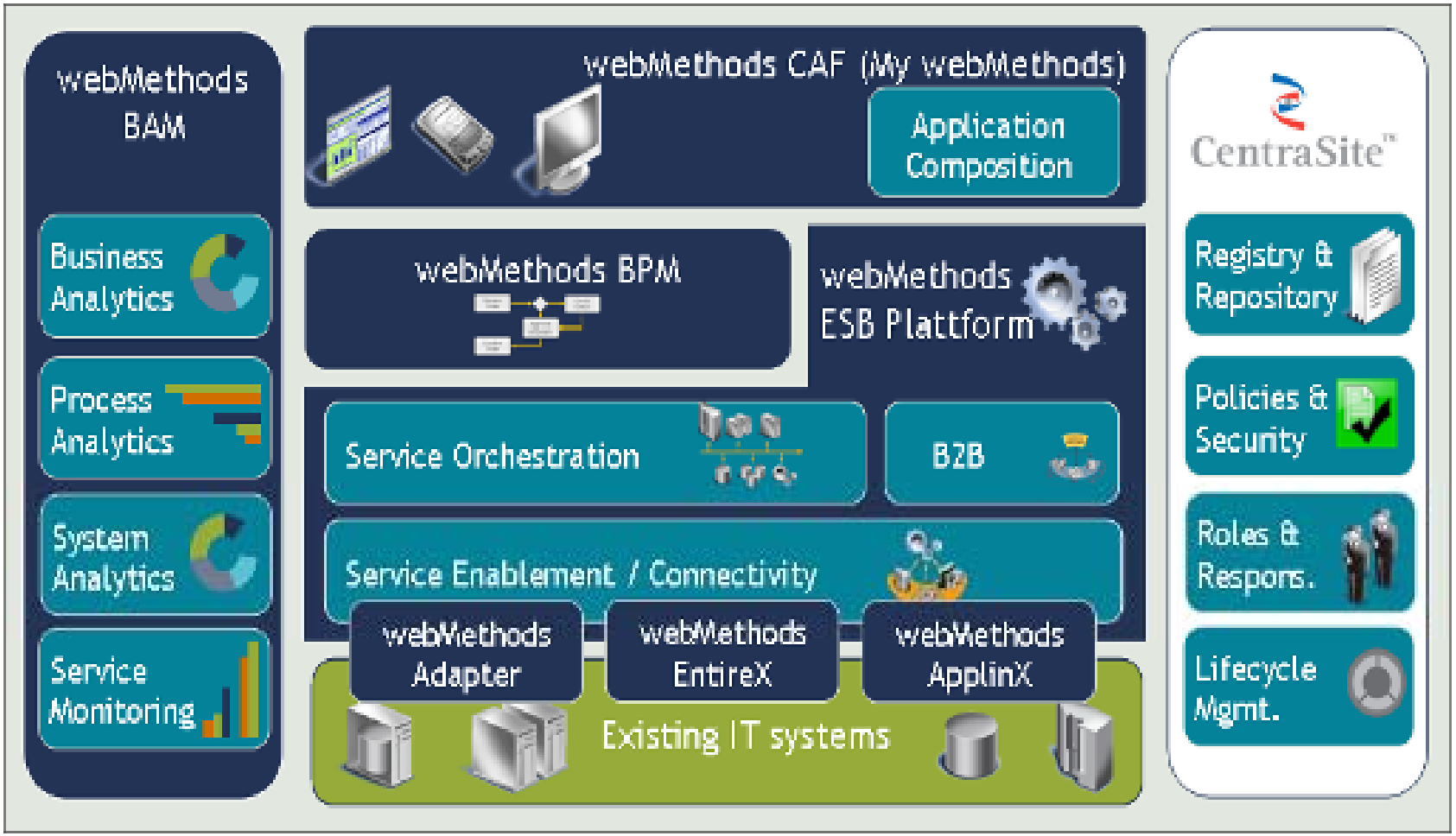
Software AG (Fortsetzung)

- Integration eigener Produkte in das SOA-Portfolio
  - EntireX
    (Kapselung bestehender Programme in Services)
  - ApplinX
    (Erstellung von Web-Oberflächen auf Basis bestehender Anwendungsoberflächen, z.B. auch Großrechneranwendungen)
  - Tamino XML-Server
    (Native XML-Datenbank)
  - **CentraSite**
    (Registry/Repository und SOA-Governance)
- Es wird vor allem die Umgestaltung vorhandener Systemlösungen in eine SOA adressiert („**Application Modernization**“)

SOA Suiten: Software AG [3]

Software AG (Fortsetzung)

Quelle: Software AG



SOA Suiten: Software AG [4]

Stärken

- Der Bereich des Business Process Management ist durch die Integration von webMethods eine besondere Stärke
- SOA-Governance ist durch flexibel erweiterbares Repository umfänglich möglich

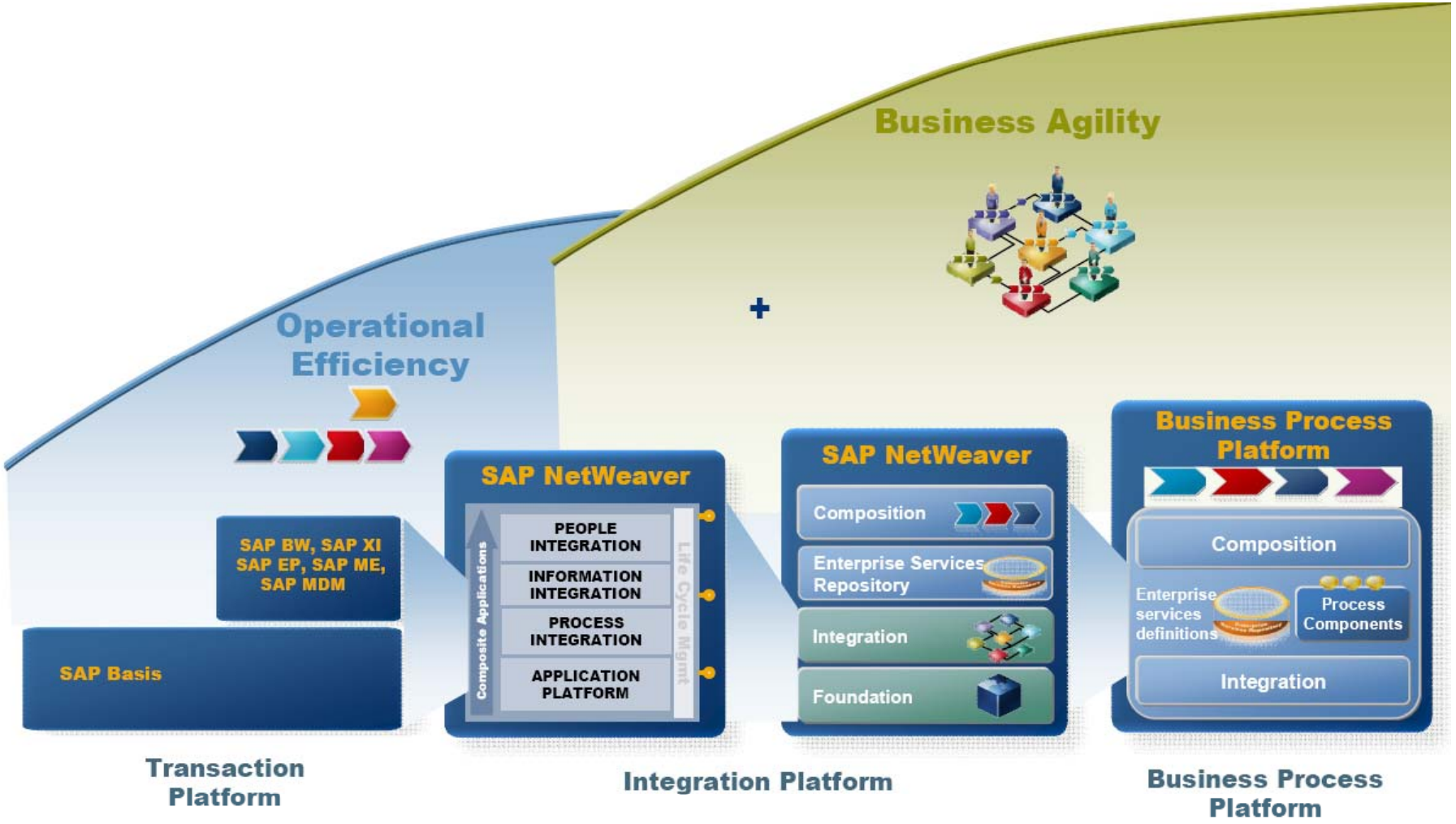
Schwächen

- Portfolio ist nicht so vollständig wie bei IBM oder ORACLE

SOA-Suites: SAP [1]

SOA-Strategie

Quelle: SAP



SOA-Suites: SAP [2]

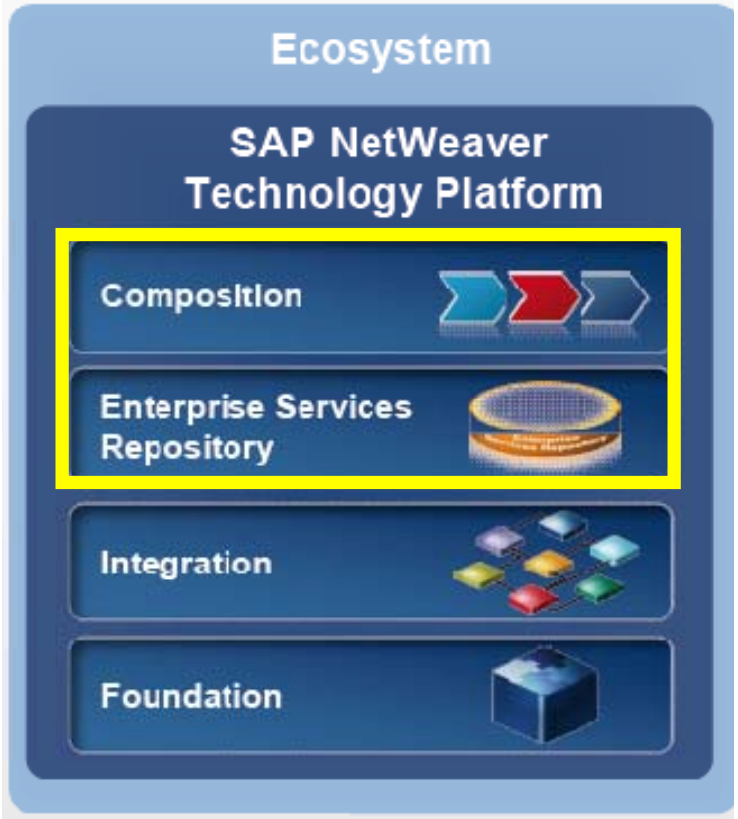
SAP Netweaver

- Die bestehenden Netweaver-Technologien *Integration* und *Foundation* wurden ergänzt:
  - Composition

    (Geschäftsprozesse, Service-Orchestrierung)
  - Enterprise Service Repository

    (Service / Software-Lifecycle, SOA-Governance)

Quelle: SAP



SOA-Suites: SAP [3]

- SAP bietet alle Funktionalitäten des bestehenden ERP-Systems:
  - Fachkomponenten in Form von „Enterprise Services“
  - Identity- und Accessmanagement inkl. Single Sign On (SSO)
  - Software-Lifecycle-Management
  - Portal-Lösung
  - …
- „Out of the Box“-Geschäftsprozesse („Enterprise Service Bundles“)
- Durch die offenen Web-Service-Standards wird es möglich, eigene Anwendungen mit den SAP-Services und den Standardprozessen zu verbinden
- „Composition Environment“ (Java-Entwicklungsumgebung auf Basis von Eclipse)
- SAP setzt einen Schwerpunkt auf „Globale Datentypen“

SOA-Suites: SAP [4]

Enterprise Service Bundles (SAP Core)

Sales

- Account and Contact Management
- Activity Management
- Customer Fact Sheet
- Customer Service Execution
- Customer Quote Management
- Interactive Selling
- Opportunity Management
- Order to Cash
- Order to Cash with CRM
- Product Master Data Management
- Quote to Order for Configurable Products
- Rebate Management
- Sales Contract Management
- Sales Incentive and Commission Management
- Territory Management

Service

- Complaint Management
- Customer Service Execution
- Installed Base Management
- Service Contract Management
- Service Order Management

Marketing

- Campaign Management
- Lead Management

Channel Management

- Request for Registration Processing

Human Capital Management

- HCM Enterprise Learning
- HCM Master Data
- HCM Organizational Management
- HCM Time Management

Corporate Services

- Asset Configuration
- Easy Inspection Planning
- Environment, Health, and Safety (EH&S)
- Integration of Quality Management Systems
- Maintenance Processing
- Maintenance Service Collaboration
- Project System
- Real Estate - Room Reservation
- Travel Management

E-Commerce

- Product Catalogue Processing with CRM

Demand and Supply Planning

- Demand Planning
- Service Parts Management

Financials

- Bank Communication Management
- Credit Management
- Dispute Management
- Electronic Bill Presentment and Payment
- External Cash Desk

Procurement

- Central Contract Management
- External Requirement Processing
- Procure to Pay
- Service Procurement
- Sourcing
- Supplier Order Collaboration with SRM
- Trade and Commodity Management

Supply Network Collaboration

- Customer Collaboration for the Supply Chain
- Outsourced Manufacturing
- Supplier Collaboration for the Supply Chain

Order Fulfillment

- ATP Check
- Availability Issue Resolution and Backorder Processing

Supply Chain Visibility

- Business Event Handling for Process Tracking

Product Development and Manufacturing

- Batch Traceability and Analytics
- Integration of Manufacturing Execution Systems
- Manufacturing Work Instructions
- Technical Document Management Connectivity
- Responsive Product Development and Launch

Transportation, Warehousing

- Cross-Industry RFID Enabled Core Logistics
- Integration of External Warehouse Management Systems
- Integration of Transportation Management System
- Inventory Lookup
- Inventory Management
- Transportation Collaboration and Subcontracting
- Yard and Storage Management Processes

RFID Enablement

- Kanban Processing
- Management of Devices through Enterprise Services
- Management of Tag IDs and Tag Observations

SOA-Suites: SAP [5]

Stärken

- Der SAP-Ansatz ist gut geeignet für die SOA-Migration bestehender SAP-Installationen, da umfangreiche fachliche Komponenten zur Verfügung stehen
- Aufgrund der zahlreichen Anwendungskomponenten existiert ein großes Potential bei der Geschäftsprozessmodellierung
- Der Repository-Ansatz ist zwar erst in den Anfängen, aber ein starker Ausbau ist zu erwarten

Schwächen

- Die Einführung der Netweaver-Technologie ist in der Regeln nur in Verbindung mit dem SAP-ERP-Umfeld sinnvoll

SOA Suites: TIBCO [1]

ActiveMatrix 2.0

- Service Bus
- BusinessWorks (Business Process Management)
- Policy Manager

  (Security-Proxy-Management für Services und in Verbindung mit ESBs oder Registry / Repositorys)
- Registry
- Service Grid

  (Service-Container / Service-Kapselung für bestehende Anwendungen)
- Performance Manager

  (Monitoring)

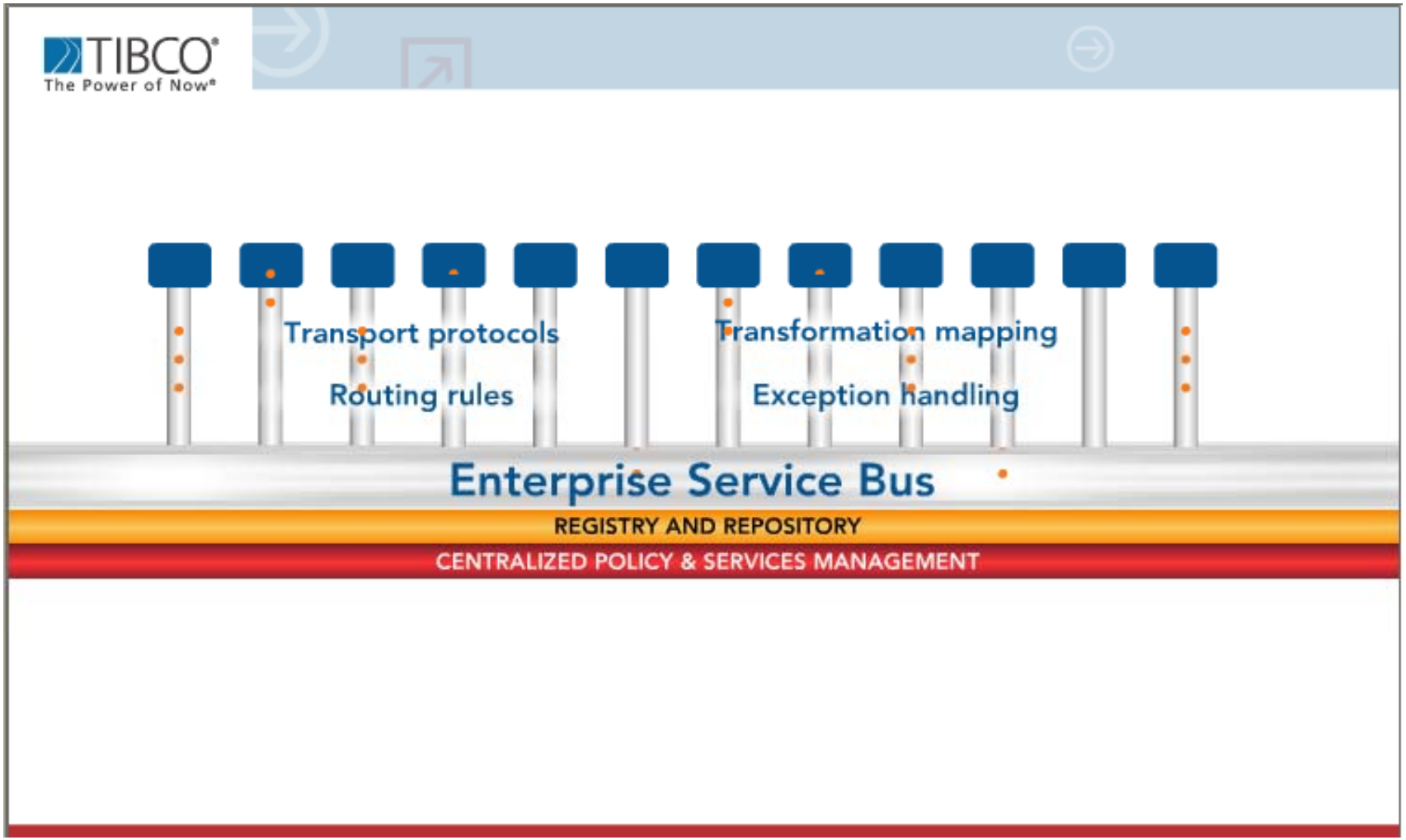
SOA Suites: TIBCO [2]

Erweitertes Portfolio

- Neben typischen SOA-Produkten existieren weitere Tools aus dem Enterprise-Software-Bereich:
  - Diverse Tools zum Business Process Management
  - Großrechneranbindung
  - Master Data Management
  - Portal Builder
  - System Monitoring und Management

SOA Suites: TIBCO [3]

SOA-Infrastruktur



SOA Suites: TIBCO [4]

Stärken

- Kommunikations- und Serviceabstraktionsschicht
- Virtualisierung von Services
- Kapselung bestehender Anwendungen in Service-Containern
- Policy-Management
  (ohne Veränderungen an bestehenden Services vornehmen zu müssen)
- Ausrichtung ähnlich wie bei der Software AG

Schwächen

- Portfolio ist nicht so vollständig wie bei IBM oder ORACLE

SOA Suites: Microsoft [1]

BizTalk

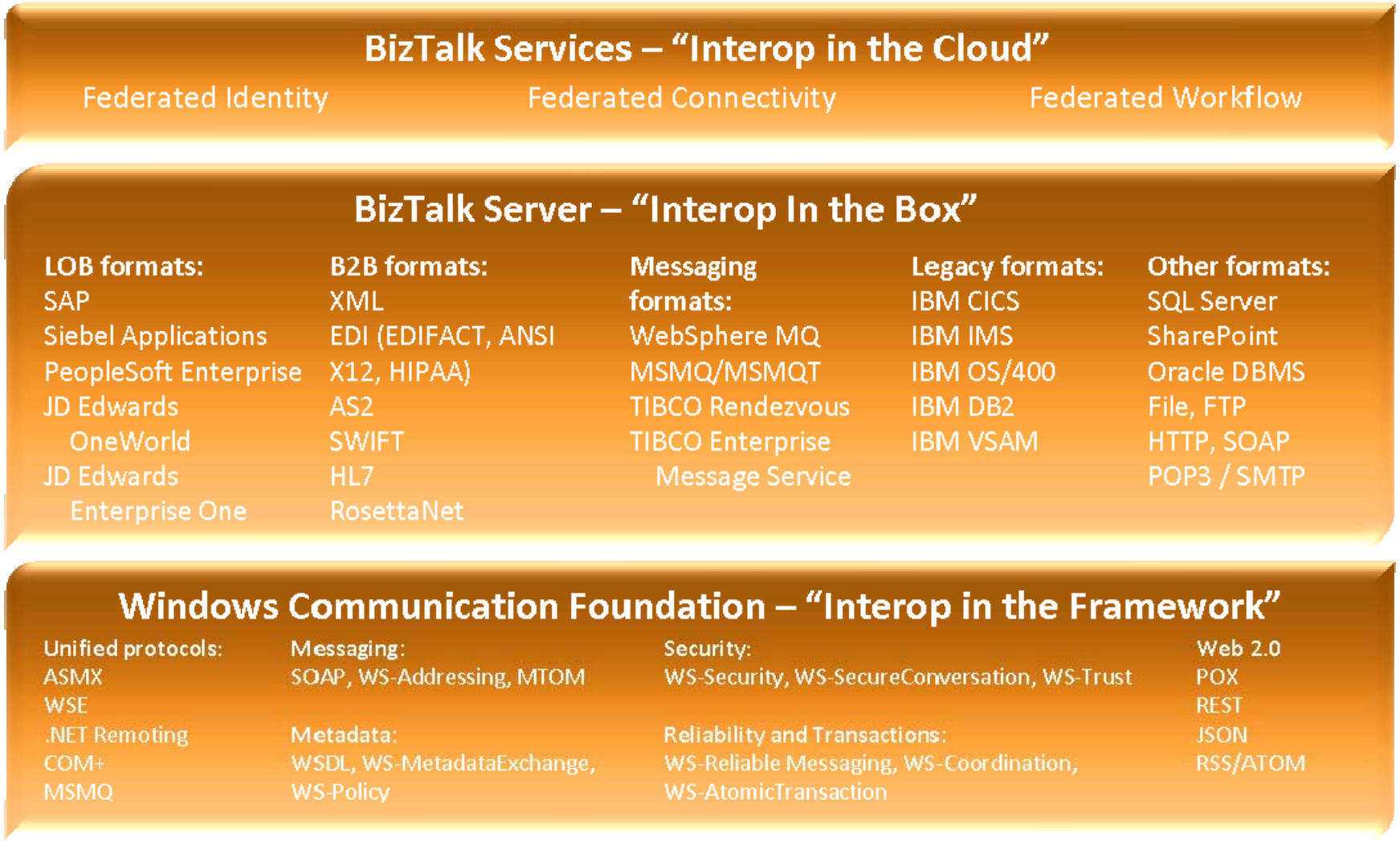
- Bestehende Produkte werden in Richtung SOA ausgerichtet
  - Service-Entwicklung: **.NET**, **Sharepoint**
  - Prozesse: **BizTalk**
  - Verzeichnisdienste: **Fremdprodukte (Amberpoint, SOA Software)**
  - Betrieb: **System Center**, **Fremdprodukte (Amberpoint, SOA Software)**
  - ESB:
    Kombination aus **BizTalk**, Windows 2003, .net-Framework und SQL-Server
  - Identitätsmanagement: Active Directory, ADFS
  - Kommunikationsbasisdienste: Windows Communication Foundation

- Projekt „Oslo“:
  - Microsoft Modeling Platform
  - „Oslo“-Repository

SOA Suites: Microsoft [2]

SOA Suites: Microsoft [3]

Standardprotokolle und Adapter



SOA Suites: Microsoft [4]

Stärken

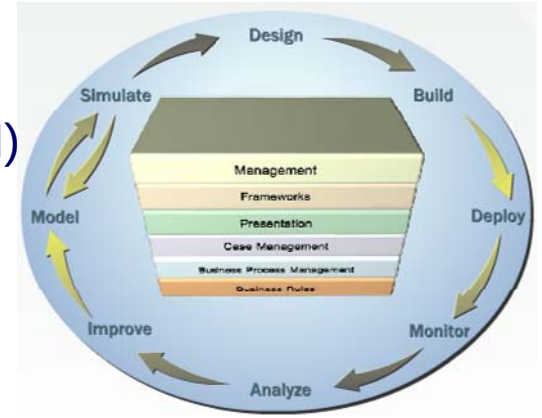
- Microsoft ist einer der Anbieter, die dieThemen **Web-Services, Verzeichnisdienste** und **WS-Standards** forciert haben
- BizTalk ist ein umfangreiches Basisprodukt für ESB, BPM und Monitoring. Es werden zahlreiche WS-Standards unterstützt.
- Visual Studio Team System ist eine umfangreiche Entwicklungs-, Test- und Deployment-Umgebung

Schwächen

- Es fehlt ein geeignetes **Repository**, das jetzt erst im Projekt „Oslo“ entwickelt wird.
- Verschiedene Technologien sind Microsoft-spezifisch

SOA-Spezialisten [1]

Pega – SmartBPM-Suite



- Laut Gartner und Forrester eines der führenden Werkzeuge für **Business Process Management** (BPM)
- Process Analyzer
  (Online-Analyse und -Auswertung)
- Process Simulator
  (Simulation von Prozessen, bevor sie tatsächlich eingesetzt werden)
- Enterprise Integration
  (Enterprise Connectors und Adapters, u.a. auch BPEL)
- Case Management
  (Out-of-the-Box Lösung für eine Fall-Management-Anwendung)
- Content Management Integration
  (Einbindung von Daten aus CMS und anderen Systemen)
- Portal Integration

SOA-Spezialisten [2]

- **IONA – Artix**
  - Artix laut Forrester einer der Market-Leader im Bereich ESB
  - Im Juni 2008 von Progress aufgekauft
  - Open Source ESB „FUSE“ auf Basis von *Apache Service Mix* (siehe Open Source Produkte)
- **Progress – SONIC / Actional**
  - **SONIC** laut Forrester auch einer der Market-Leader im Bereich ESB
  - **Actional** für Monitoring und Policy Management; integriert sich in ESB und Service Repositorys anderer Hersteller
  - Lücken im Portfolio bei Registry / Repository und BPM
  - Umfangreiche Features für Integration in Fremdprodukte

SOA-Spezialisten [3]

Amberpoint – SOA Management System

- SOA Discovery
  (Erfassung und Darstellung von Service-Abhängigkeiten)
- Runtime Management
  (Policy Management und Enforcement)
- SOA Security
- Service Level Management
  (Monitoring und Management von Leistungswerten)
- Exception Management
  (Regelbasierte Fehlerbehandlung auf allen Abstraktionsebenen)
- Transaction Tracking
  (Protokollierung für Analysezwecke)
- Metadata Federation
  (Abgleich zwischen verschiedenen Repositorys bzw. Registrys)

SOA-Spezialisten [4]

SOA Software

- **Policy Manager**
  - UDDI v3 Registry
  - Policy Definition
  - Contract Management
  - Monitoring
  - Wird von Microsoft mit BizTalk integriert
- **Repository Manager**
  - Erweiterung der Registry mit Metadaten
  - Federation mit HP Systinet, IBM WSSR oder TIBCO ActiveMatrix möglich
  - Synchronisation von Servicedefinitionen, Metadaten und Zustandsinformationen

SOA-Spezialisten [5]

SOA Software

- **Service Manager**
  - Überwachung und Enforcement von Policys in Zusammenarbeit mit dem *Policy Manager*
  - Teilfunktionen eines ESB
  - Ergänzung von Sicherheitsfunktionen
- **SOLA**
  - Kapselung von CICS-Anwendungen als Services

SOA-Spezialisten [6]

HP Systinet

- Registry / Repository
- Policy Management
  (Policy Management und Enforcement)
- Contract Management
  (Monitoring und Management von SLAs)
- Lifecycle Management
  (Erfassung und Darstellung von Service-Abhängigkeiten)

→ Wird von BEA (ORACLE) als *Aqualogic Service Registry* angeboten

Zusammenfassung (kommerzielle Produkte)

Leistungsbereiche der betrachteten SOA-Anbieter

	IBM	ORACLE / Bea	Software AG	SAP	TIBCO	Microsoft	Pega	IONA / Progress	Amberpoint	SOA Software	HP
Enterprise Service Bus (ESB)	X	X	X	X	X	X		X		O	
Verzeichnisdienste / Repository	X	X	X	X	X			X	X	X	X
Business Process Management	X	X	X	X	X	O	X	O			
Monitoring / Betrieb	X	X	X	X	X	X	X	X	X	X	X
Identity- und Accessmanagement	X	X		O		O					
Portal	X	X	X	X	X	O	X				

X = verfügbar O = verfügbar, aber mit Einschränkungen

Kriterien für den Einsatz von Open-Source-SOA [1]

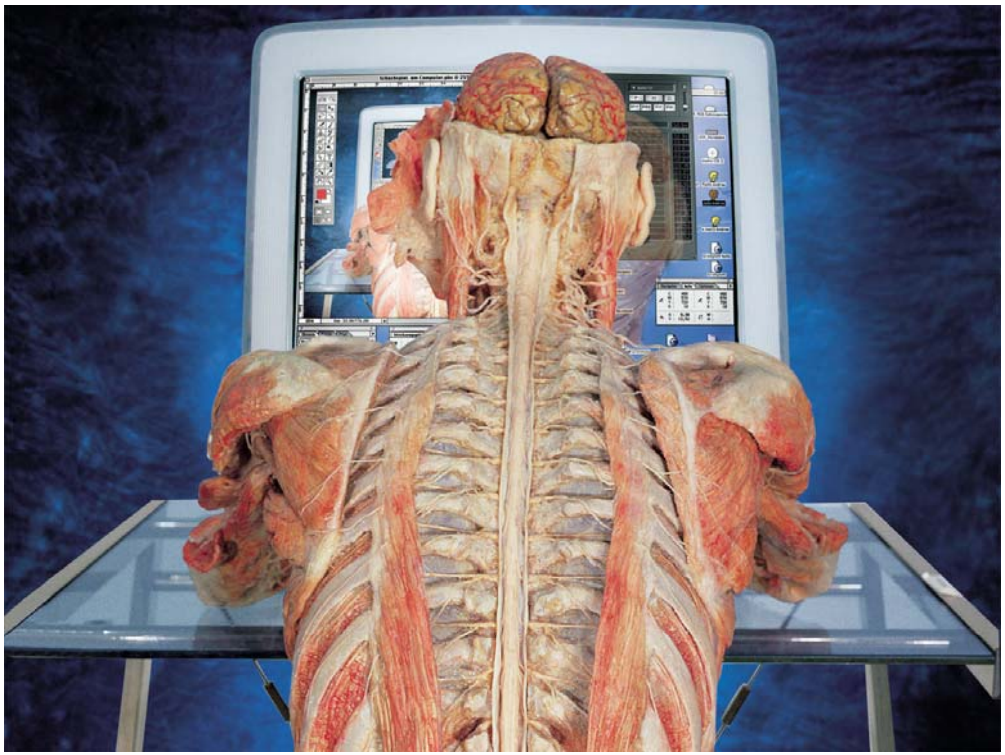
Technologisches Know-how

- **Kommerzielle Produkte** werden im Allgemeinen mit einer komfortablen **Benutzeroberfläche** und umfangreicher **Dokumentation** zur Verfügung gestellt.
  Die Komplexität der einzelnen Komponenten wird „unter der Haube“ versteckt.

- Bei **Open-Source-Produkte** gilt in aller Regel:
  - Die Produkte werden als **einzelne Funktionskomponenten** im **Binär- und Quellcode** zur Verfügung gestellt.
  - Individuelle Aufgabenstellungen und Probleme müssen entweder **mit eigenem Know-how** oder **Informationen aus der Entwicklungs-community** gelöst werden.
  - **Herstellersupport** mit vereinbarten SLAs, **ist teilweise nicht zu bekommen** oder nur gegen entsprechendes Entgelt.

Kriterien für den Einsatz von Open-Source-SOA [2]

- Beim Einsatz von Open-Source-Tools lernt man die technischen Aspekte von SOA bis in die Innereien kennen!



Gunther von Hagen: Hacker

Kriterien für den Einsatz von Open-Source-SOA [3]

Verwendung von Standards

- **Open-Source-Produkte** müssen stark auf allgemein anerkannten **offenen Standards** aufsetzen. Ob die Interoperabilität mit anderen Produkten dadurch schon gegeben ist, hängt häufig noch von „Einzelheiten“ ab.
- Im Gegensatz dazu verwenden **proprietäre Lösungen** oft „optimierte“ Standards, also veränderte Standards, die eine problemlose Interoperabilität verhindern.
- Einige Standards sind in Open-Source-Produkten abhängig vom **Entwicklungsstand**, jedoch manchmal (noch) nicht vollständig umgesetzt.

Die Nutzung von **Standards** ist eine notwenige Voraussetzung für Interoperabilität, aber keine Garantie dafür!

Kriterien für den Einsatz von Open-Source-SOA [4]

Produktentwicklung

- Im **industriellen Umfeld** ist **Relaseplanung** und **Produktsupport** von hoher Wichtigkeit.
- Die Entwicklung einzelner **Open-Source-Produkte** ist oft **nicht vorhersehbar**, da es häufig **keinen Releaseplan** gibt bzw. ein vorhandener Plan in der Regel als unverbindlich anzusehen ist.
- Bei manchen Produkten reagiert die **Community** sehr schnell auf neue Anforderungen oder aufgetretene Fehler, bei manchen jedoch nicht.
- Aufgrund der in aller Regel besseren Interoperabilität **lassen sich Open-Source-Produkte leichter austauschen** als kommerzielle Produkte.

Kriterien für den Einsatz von Open-Source-SOA [5]

Lizenztypen

- Die **unterschiedlichen Lizenztypen**, z.B. GPL, LPGL, CPL oder EPL haben unterschiedliche Konsequenzen hinsichtlich der Nutzung und der Weitergabe!
- Bei reiner **Nutzung einer SOA-Infrastruktur** sind die Unterschiede zwar auch gegeben, jedoch nicht so gravierend.
- Hingegen gilt es signifikante Unterschiede bei den Lizenzen zu beachten, wenn SOA-Lösungen **modifiziert und vertrieben** werden!
- Bei bestimmten Lizenztypen werden **eigene Entwicklungen** automatisch ebenfalls zu Open-Source, bei anderen dagegen nicht!

Open-Source ist kein rechtsfreier Raum!

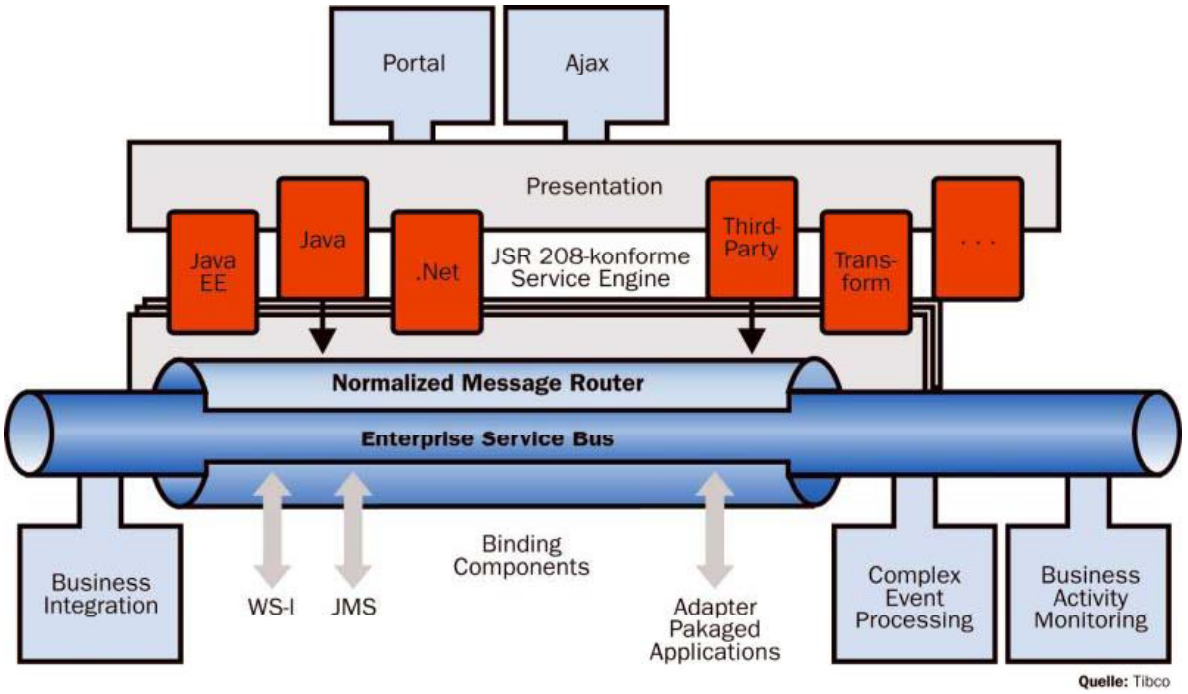
Basistechnologien [1]

- Im Open-Source-Sektor haben sich überwiegend Systeme auf Basis von **Java-Technologien** etabliert.
- Es gibt Standards für
  - Application Server (z.B. J(2)EE, …)
  - Enterprise Service Bus (z.B. JBI, JMS)
  - Business Process Management (z.B. JPDL)
  - Identity- und Access-Management (z.B. JAAS, SAML)
  - Monitoring (z.B. JMX, Mbeans)
  - Portal (z.B. Portlets, JSF)

  Die meisten Lösungen und Projekte setzen auf diesen Basis-Technologien auf

Beispiel: JBI – Java Business Integration

- Kernstück ist der **Normalized Message Router** (NMR) für Übermittlung und Transformation von Nachrichten, die zwischen Service-Containern ausgetauscht werden
- Implementierung auf dieser Basis: Apache ServiceMix, SUN OpenESB



SCA (Service Component Architecture)

- **SCA** ist ein **Programmiermodell** für die Entwicklung und Komposition von Geschäftsfunktionen in Komponenten
- Die Komponenten einer SCA laufen in einer SCA-Runtime-Umgebung.
- Für die unterschiedlichen **Bindings** der Services werden wie bei JBI ebenfalls **Service-Container** genutzt.
- SCA ist **keine (reine) Java-Technologie**, sondern unterstützt unterschiedliche Programmierumgebungen
- **SCA wird von fast allen kommerziellen Lösungen unterstützt**
- Im Open-Source-Bereich wird versucht, JBI und SCA zu unterstützen. Einige Projekte legen aber den Schwerpunkt auf eine bestimmte Technologie.
- Eine gute Einführung zu SCA ist zu finden unter http://www.davidchappell.com/articles/Introducing_SCA.pdf

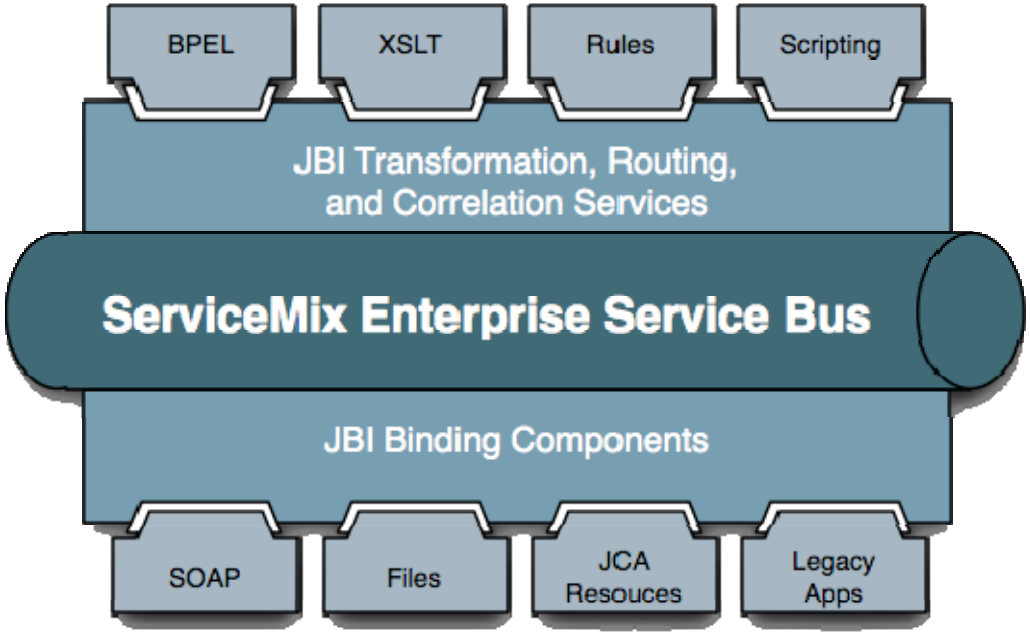
Apache Software Foundation [1]

- Komponenten mit direktem SOA-Bezug:
  - Apache ServiceMix - Enterprise Service Bus
  - Apache ActiveMQ – Messaging
  - Apache Camel – Class Library für Enterprise Design Patterns
  - Apache CXF – Entwicklungsframework für Services
  - Apache ODE (Orchestration Director Engine) – Workflow Engine
- Neben den SOA-Komponenten gibt es eine Reihe von Web-Service-Tools (http://ws.apache.org/), unter anderem:
  - AXIS – Implementierung des SOAP-Protokolls
  - jUDDI – UDDI 2.0 Registry
  - WSIF – Aufruf von Services über verschiedene Bindings
- Die Apache-Komponenten sind wiederum Teil anderer SOA-Lösungen wie z.B. FUSE oder SOPERA ASF

Apache Software Foundation [2]

Beispiel: Apache ServiceMix (http://servicemix.apache.org)

- Wird in einigen SOA-Lösungen, z.B. SOPERA ASF verwendet
- Basiert auf JBI
- Es existiert bereits eine Anzahl von JBI-Komponenten, die die Integration in den Service Bus ermöglichen



Quelle: http://servicemix.apache.org/home.html

Anbieter mit SOA-Portfolio

- Im Open-Source-Bereich gibt es nur wenige Anbieter, die den gesamten „SOA-Stack“ abdecken

  Die einschlägigsten Anbieter sind:
  - Sun Microsystems – JavaCAPS
  - Red Hat – jBoss Enterprise SOA Platform
  - SOPERA
- Beispielhafte Anbieter mit reduziertem SOA-Stack:
  - Progress – FUSE
  - MuleSource - Mule

Sun Microsystems [1]

JavaCAPS (Java Composite Application Platform Suite)

- Sun Enterprise Service Bus
- Sun Business Process Manager
- Sun Intelligent Event Processor
- Sun Master Index
- Sun Data Integrator
- Sun Adapters
- Sun GlassFish Enterprise Server
- Sun Java System Portal Server
- Sun Java System Directory Server Enterprise Edition
- Sun Java System Access Manager
- NetBeans Integrated Development Environment (IDE)

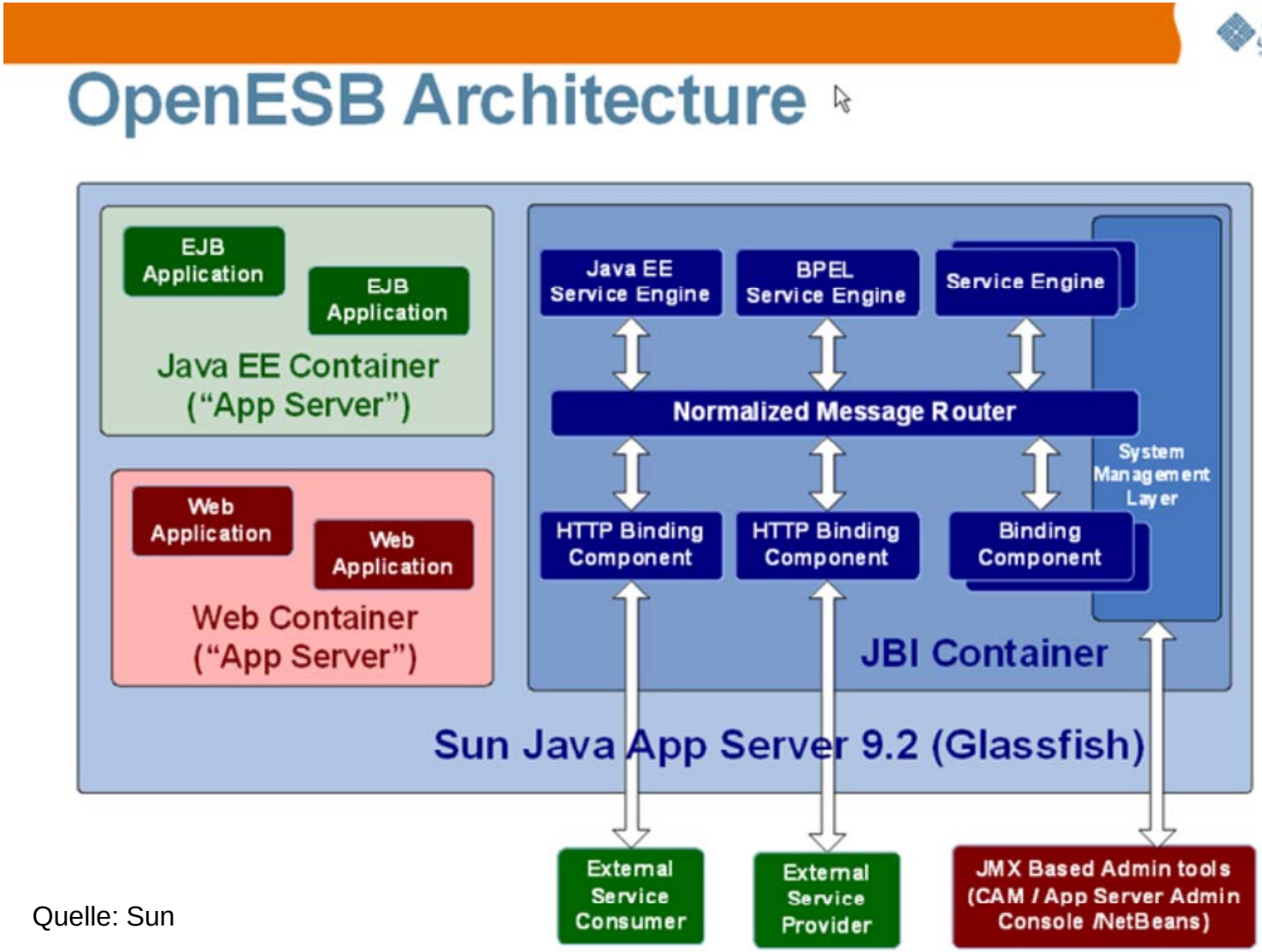
Sun Microsystems [2]

- Sun fährt eine zweigleisige Strategie:
    i. Kostenlose Open Source Software, z.B. OpenESB, Glassfish
    ii. Kostenpflichtiges JavaCAPS-Paket mit Professional Services und Support
- Sehr gute Darstellung der Funktionalität von OpenESB / Glassfish unter Verwendung von JBI und der Unterschied zu JavaCAPS unter

  http://wiki.open-esb.java.net/attach/OpenESBIntroductionScreencast/openesbIntroPreso.swf

  (Video, 38 Min.)
- JavaCAPS wird derzeit als das **vollständigste und am weitesten entwickelte Produktportfolio im Open-Source-Bereich** angesehen.
- Die Entwicklung über die für Sun typische Netbeans-Oberfläche wird von manchen Benutzern als Nachteil bewertet.

Open ESB [1]



Quelle: Sun

Open ESB [2]

OpenESB Functionality

- Service Engines
  - BPEL SE
  - XSLT SE
  - JavaEE SE
  - IEP (BAM) SE
  - ETL SE
  - SQL SE
  - Workflow SE

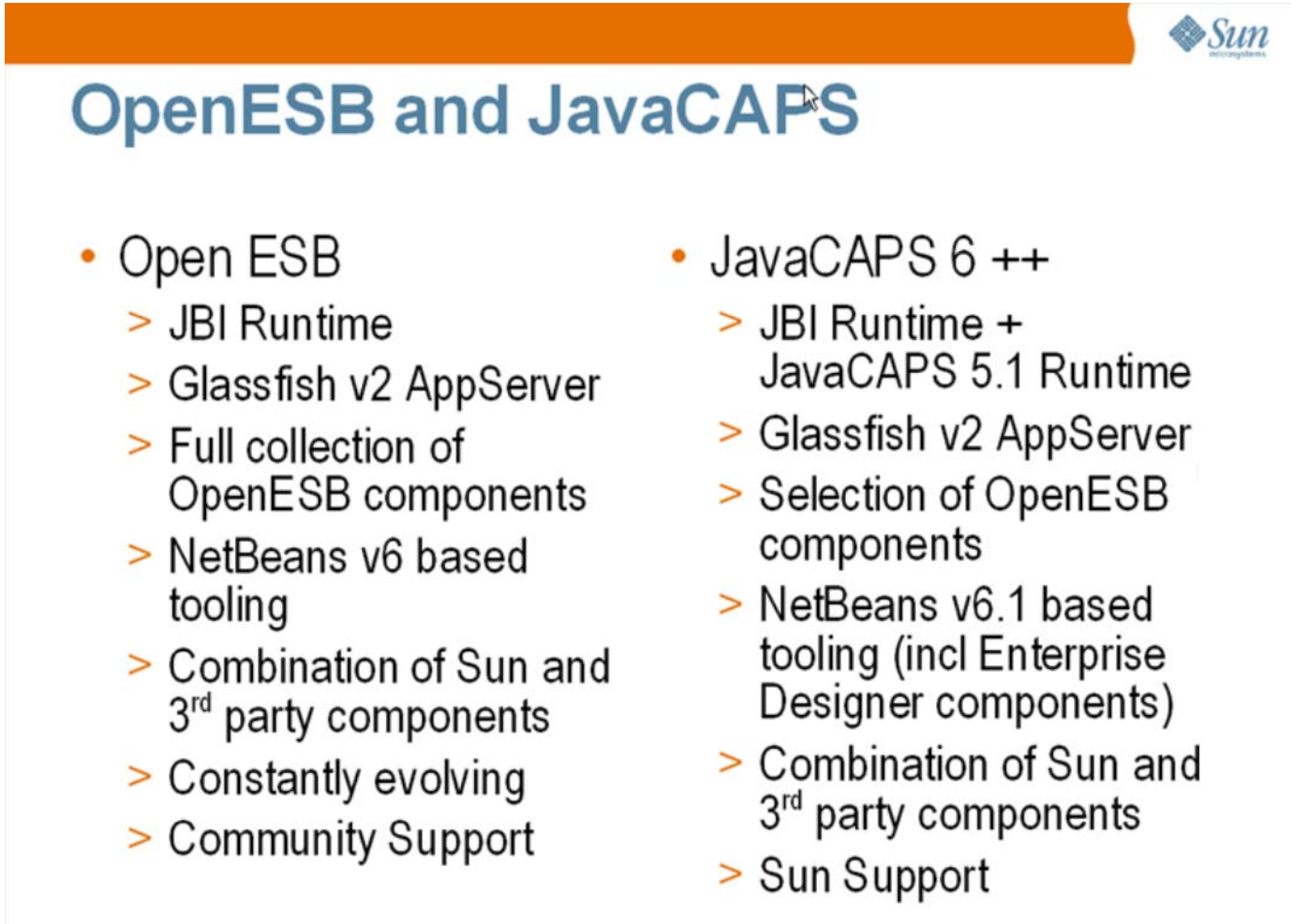
Sun to provide a supported version though JavaCAPS releases

- Binding Comps
  - MQSeries BC
  - HL7 BC
  - SAP BC
  - SMTP BC
  - HTTP BC
  - JMS BC
  - File BC
  - CICS BC
  - DCOM BC
  - CORBA BC
  - ...

- Other
  - Clustering
  - CASA
  - JBI Mock
  - WSIT Tech
- In Progress
  - CAM
  - Aspect SE
  - Encoding SE
  - Rules SE
  - Scripting SE

Quelle: Sun

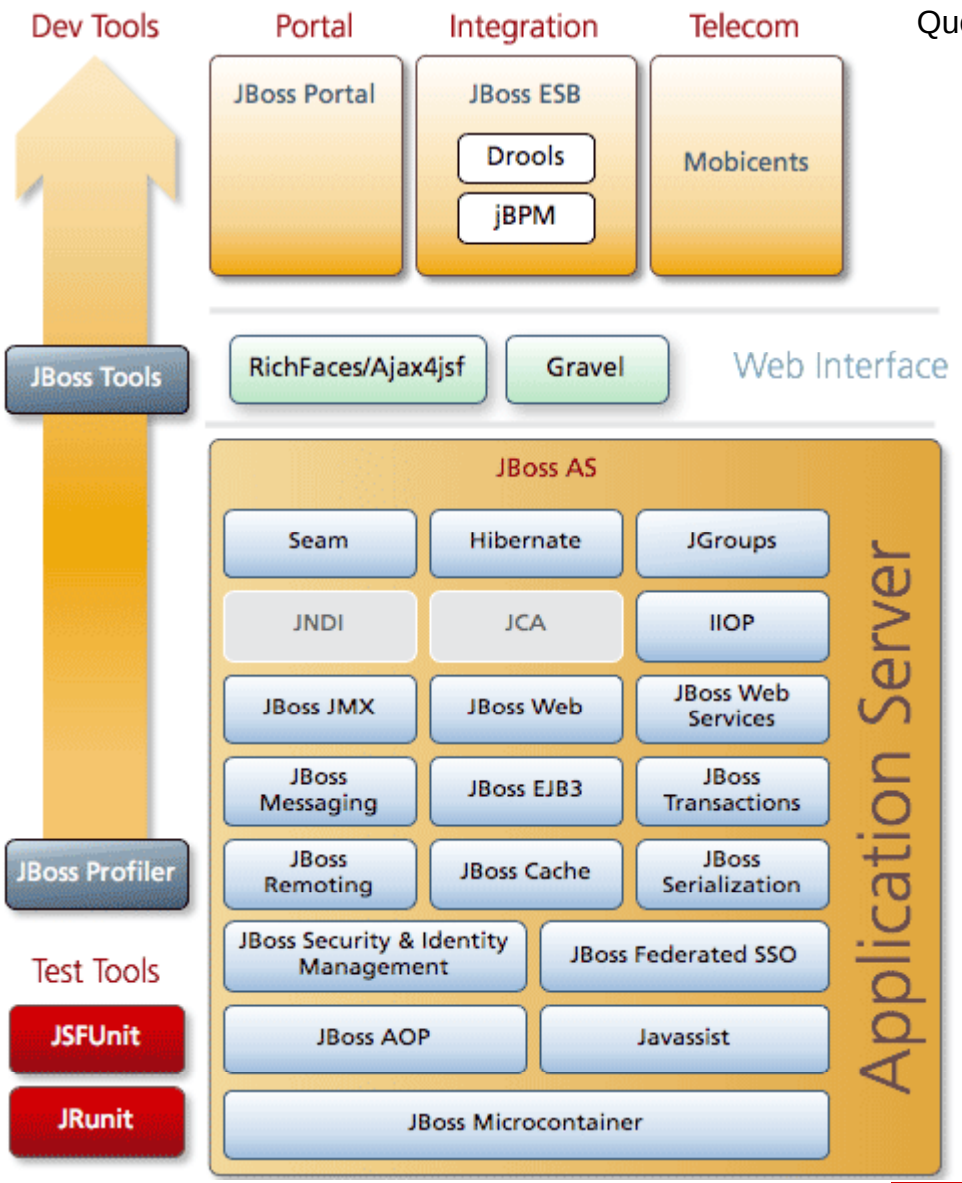
OpenESB im Vergleich zu JavaCAPS



Quelle: Sun

Red Hat [1]

jBoss-Produktfamilie



Quelle: jBoss

Red Hat [2]

jBoss Enterprise SOA Platform

- Komponenten
  - JBoss Enterprise Application Platform – JEE Application Server
  - JBoss Enterprise Service Bus – ESB und UDDI 2.0 Registry
  - JBoss jBPM – Eclipse Plugin für Modellierung bevorzugt in JPDL
  - JBoss Rules – Rules Engine

- Neben den als SOA-Plattform angebotenen Komponenten gibt es auch Lösungen für:
  - Portal
  - Security- und Identity-Management

Red Hat [3]

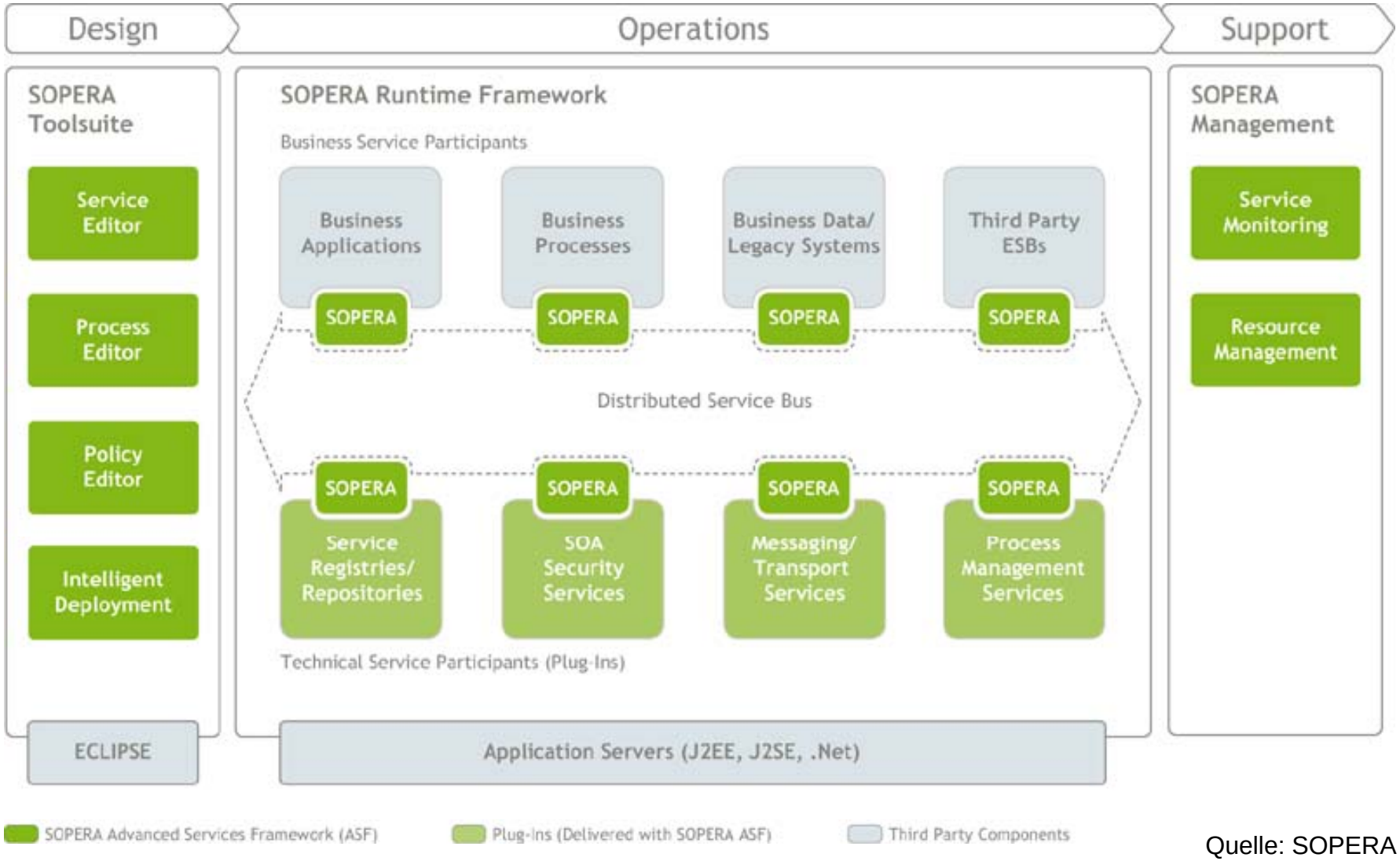
jBoss Enterprise SOA Platform

- Unterstützte Standards (laut Datenblatt):
  - SOAP 1.1
  - WS-Security
  - MTOM
  - WSDL 1.1
  - WS-Addressing 1.0
  - JAX-WS
  - UDDI 2.0
  - Java Management Extension (JMX) 1.2
  - Full J2EE 1.4 compliance

BPEL 1.1 wird nicht vollständig unterstützt
BPEL 2.0 noch nicht fertig

- JBoss hat einzelne Stärken (Application Server).
  Die gesamte SOA-Plattform ist aber **noch nicht ausgereift** und noch in der Entwicklung!

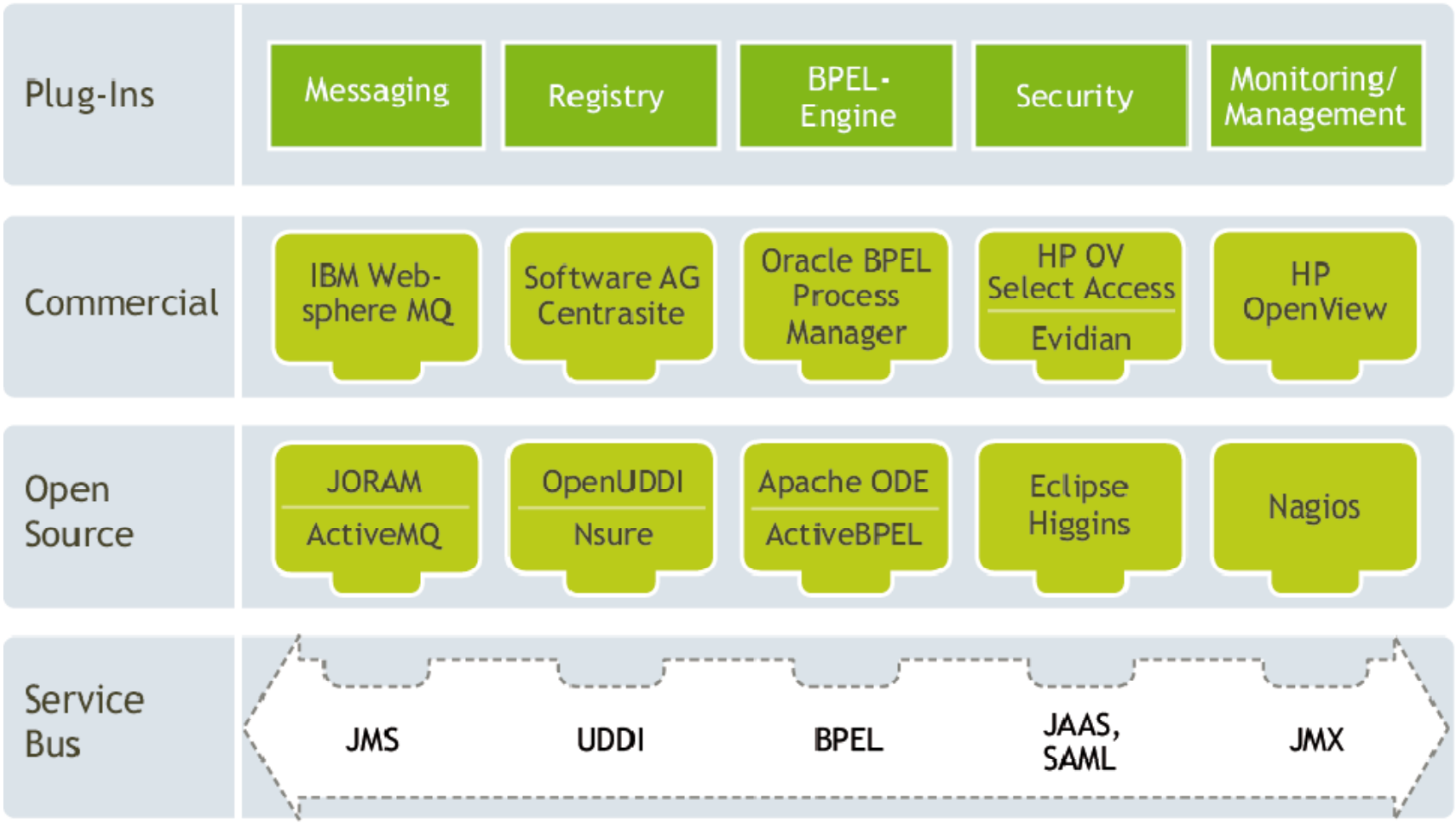
SOPERA [1]



Quelle: SOPERA

SOPERA [2]

SOPERA ASF im Vergleich zu kommerziellen Lösungen

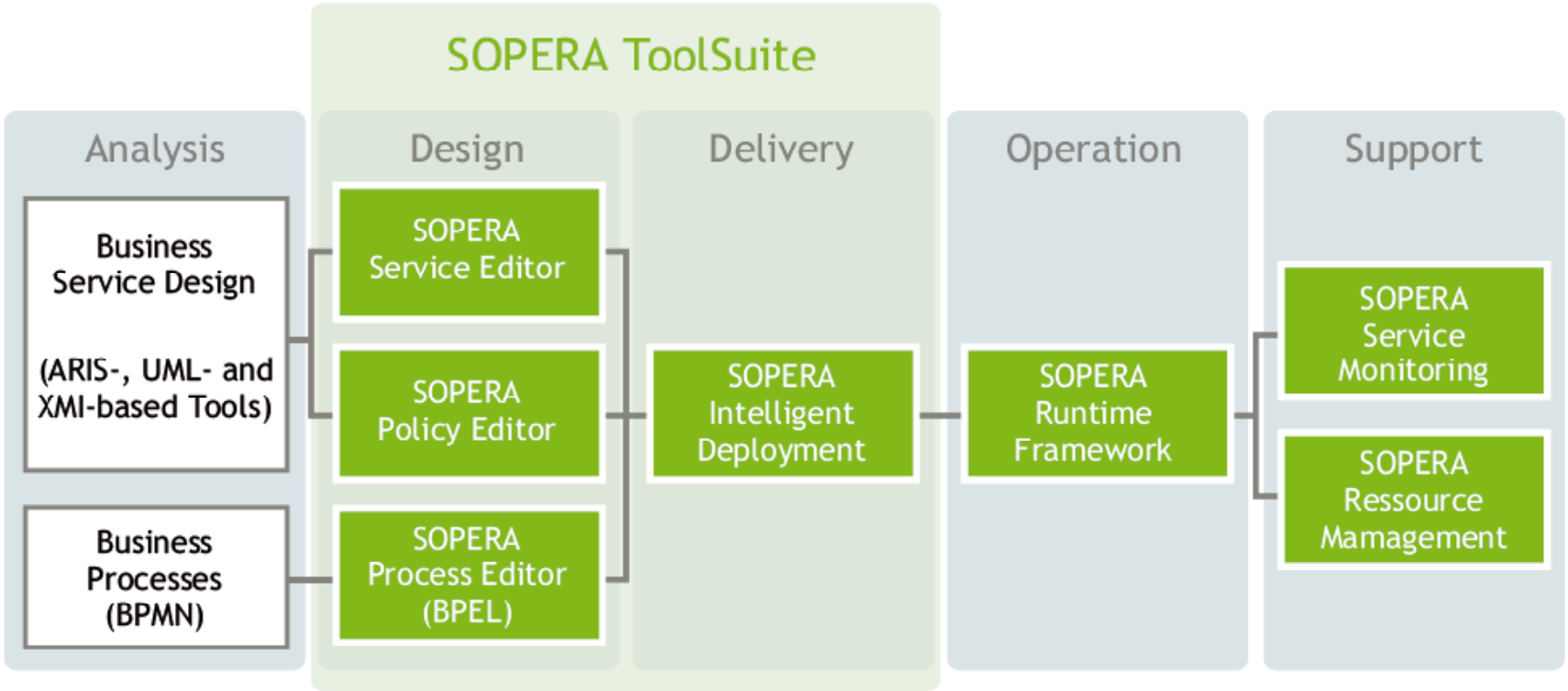


Quelle: SOPERA

SOPERA [3]

SOPERA ASF-Tools

- Die Lösung bietet Tools für einen großen Teil des Service-Lifecycles
- Keine Modellierungswerkzeuge für Geschäftsprozesse



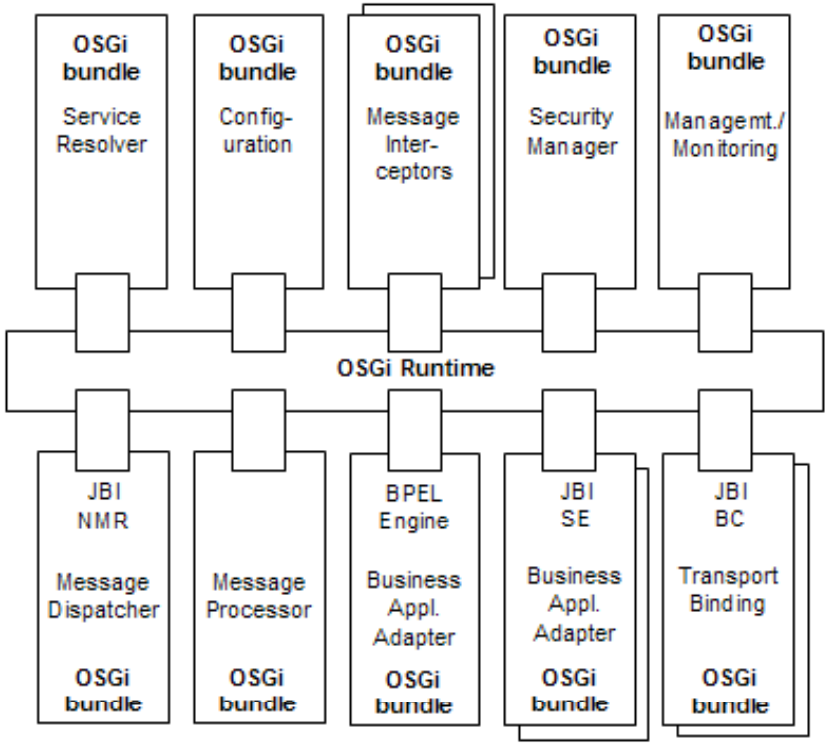
Quelle: SOPERA

SOPERA [4]

- SOPERA bietet neben der kostenlosen Edition eine kostenpflichtige Enterprise-Edition an, inklusive:
  - Hotline-Support
  - SLAs mit zugesagten Reaktionszeiten
  - Zusätzliche kommerzielle Plugins
  - .NET-Support
  - SAP-Support
  - Business Activity Monitoring
  - Diverse weitere Tools
- SOPERA bietet ein **separates Open-Source-Tool** für das Management des Apache ServiceMix ESB an
- SOPERA entwickelt im Rahmen der Eclipse Foundation das Projekt **Swordfish** federführend weiter.

Swordfish [1]

- **Swordfish** ist ein Projekt der Eclipse Foundation, das ein **Runtime-Framework** für SOA-Komponenten auf Basis von OSGi (Open Services Gateway Initiative) zur Verfügung stellt.
- Große Teile der Lösung stammen aus einem Infrastrukturprojekt der Deutschen Post.



Quelle: http://www.eclipse.org/swordfish/learn/papers.php

Swordfish [2]

- Swordfish ist **keine fertige SOA-Lösung**
- Einzelne Infrastrukturkomponenten (sog. Bundles) sind bereits verfügbar, andere müssen noch entwickelt werden
- Anwendungskomponenten werden über die Bundles eingebunden
- OSGi erweitert die Java-Umgebung um spezielle Runtime-Funktionen für die Anbindung von Services, z.B. Hot-Plugging von Services oder die Einbindung von Embedded Systems
- OSGi soll die schwergewichtigen Application Server ersetzen
- Entwicklungsumgebung ist Eclipse mit den Erweiterungen des *SOA Tools Project (STP)* und *Web Tools Project (WTP)*
- Das Projekt ist im Stadium 1.0 und es gibt noch wenig Dokumentation.

Interessante Einstiegspunkte zum Thema:

http://www.eclipse.org/swordfish/

http://searchsoa.techtarget.com/news/article/0,289142,sid26_gci1294154,00.html#

IONA / Progress

FUSE (http://fusesource.com/)

- Enterprise Service Bus auf Basis von Apache ServiceMix
- Message Broker auf Basis von Apache ActiveMQ
- Service Framework auf Basis von Apache CXF
- Mediation Router auf Basis von Apache Camel
- HQ – Monitoring auf Basis von Hyperic HQ Enterprise
- Integration Designer – Eclipse basierte Entwicklungsumgebung (jedoch noch in Vorbereitung)

Komponenten für Registry und BPM fehlen und müssen durch andere Produkte ergänzt werden.

- Interessante Video-Webinars unter http://fusesource.com/resources/video-archived-webinars/

MuleSource

MuleSource (www.mulesource.org)

- Mule ESB

  (Mule ESB ist in verschiedene andere Lösungen integriert)

- Mule Galaxy

  (ein Repository ähnlich wie das kommerzielle Produkt *Centrasite*. Es wird aber kein UDDI unterstützt und das mit Absicht („UDDI sucks“)

- Mule HQ – Monitoring

  (Das Tool ist nur in der Enterprise Edition verfügbar)

- Es gibt jeweils Enterprise Editionen, die Herstellersupport, erweiterte Qualitätssicherung und diverse Optionen für Hochverfügbarkeit und Skalierung beinhalten.

Registry / Repository

freebXML Registry

- Implementierung des OASIS ebXML Registry 3.0 Standards
- JAXR (Java API for XML-Registries)
- Registry Browser als Java-Applikation
- Registry Browser als Web-Applikation
- Diverse administrative Tools
- Keine UDDI-Unterstützung

Intalio BPMS (www.intalio.com)

- Komponenten
  - Designer – Modellierung mit BPMN auf Eclipse-Basis
    (in das SOA Tools Project (STP) eingegangen!)
  - Server – BPEL 2.0 Engine basierend auf J2EE
  - Workflow – Portalkomponente für Benutzerinteraktionen
- Editionen
  - **Community** – stark limitierte Version nur für einfache Anwendungen geeignet
  - **Enterprise** – Unterstützung der für eine SOA im Enterprise-Umfeld üblicherweise erforderlichen Funktionalitäten
- Von der Zeitschrift „Infoworld“ zum besten Open-Source-Produkt im Bereich BPM bewertet

Business Process Management [2]

ActiveBPEL (http://www.activevos.com/community-open-source.php)

- Reine BPEL-Engine
- Wird in anderen Lösungen eingebunden (z.B. SOPERA)
- Stellt kein Modellierungstool für Orchestrierung von Services zur Verfügung
- Der Hersteller bietet das kostenpflichtige Modellierungstool *ActiveVOS* an

Weitere Open-Source BPM-Tools

- Bonita – XPDL-Engine mit grafischer Modellierung
- Enhydra Shark – XPDL-Engine nach WfMC-Standard
- Apache ODE (Orchestration Director Engine) – BPEL Engine

Business Activity Monitoring (BAM)

- Es gibt zahlreiche **Monitoring Tools** auf Netzwerkebene (z.B. Nagios)
- Es gibt Standardtechnologien für die **Überwachung von Java-Applikationen** (z.B. JMX, MBeans oder NetBeans)
- Komfortable Tools für die **Überwachung der Servicenutzung** oder der aktuellen **Zustände von Business-Prozessen** gibt es im Open-Source-Bereich in der Regel nur in den kostenpflichtigen Editionen.
- Für Sun *Open ESB* gibt es ein *BAM Dashboard* auf Basis von Netbeans, JasperReports Library und iReport als Report-Designer.
- Bei den kostenfreien Tools muss **viel Aufwand** in Auswahl, Test und Konfiguration und evtl. zusätzliche Eigenentwicklung gesteckt werden.
- **Kommerzielle Tools** bzw. kostenpflichtige Open-Source-Lösungen sind in diesem Anwendungsbereich derzeit eindeutig **im Vorteil**.

Identitymanagement / Accessmanagement

- Zahlreiche Tools und Projekte
- Guter Einstiegspunkt
  http://www.iam-wiki.org/Opensource
- Nennenswerte Beispiele:
  - OpenSSO von Sun – Single Sign On für Webanwendungen
  - OpenLDAP – Verzeichnisdienst
  - OpenSAML – SAML-Implementierung in Java und C++

Liferay (www.liferay.com)

Auszug verwendete Produkte

- Apache ServiceMix
- Hibernate (Persistence Framework)
- Java J2EE/JEE
- jBPM (jBoss BPM)
- JGroups (Multicast Messaging)
- Lucene (Fulltext Search Engine)
- MuleSource ESB
- PHP
- Ruby
- Seam (jBoss Web 2.0 Technologie)
- Spring & AOP
- Struts & Tiles

Auszug verwendeter Standards

- AJAX
- JSR-168 (Portlets)
- JSR-127 (JSF)
- OpenSearch
- JSON
- REST
- RMI
- WSRP
- WebDAV

Zusammenfassung

- Für einen **ESB** gibt es ausgereifte Standardtechnologien, die in Kombination einen industriell einsetzbaren Service-Bus ergeben.
- Für **BPM** gibt es geeignete BPEL- und XPDL-Engines. Komfortable Modellierungswerkzeuge und BAM-Tools sind in aller Regel kostenpflichtig.
- Für die **Service-Entwicklung** und die **Einbindung in Portale** oder Anwendungsoberflächen gibt es zahlreiche ausgereifte Frameworks.
- Es sind diverse Technologien für **IAM** verfügbar.
- Das „**Tooling“** ist komplex, da die einzelne Produkte zwar in der Regel über Eclipse integriert werden, sich aber meist sehr unterschiedlich darstellen.
- Die **Lernkurve** kann je nach Vorkenntnissen sehr steil sein.
- **Für komplexe IT-Landschaften erforderliche Features** sind in der Regel nur in kostenpflichtigen Editionen verfügbar.

Fazit

Warum scheitern SOA-Projekte?

Computerwoche, 3.10.2008

Finanzierung [1]

Anforderungen an eine SOA aus „Managementsicht“

- Kurzfristig Kosten senken
- Mittelfristig Kosten senken
- Langfristig Kosten senken
- Geschäftsprozesse optimieren
- Geschäftsprozesse flexibilisieren
- Erlöse optimieren
- Alle Maßnahmen sollen in möglichst kurzer Zeit Wirkung zeigen

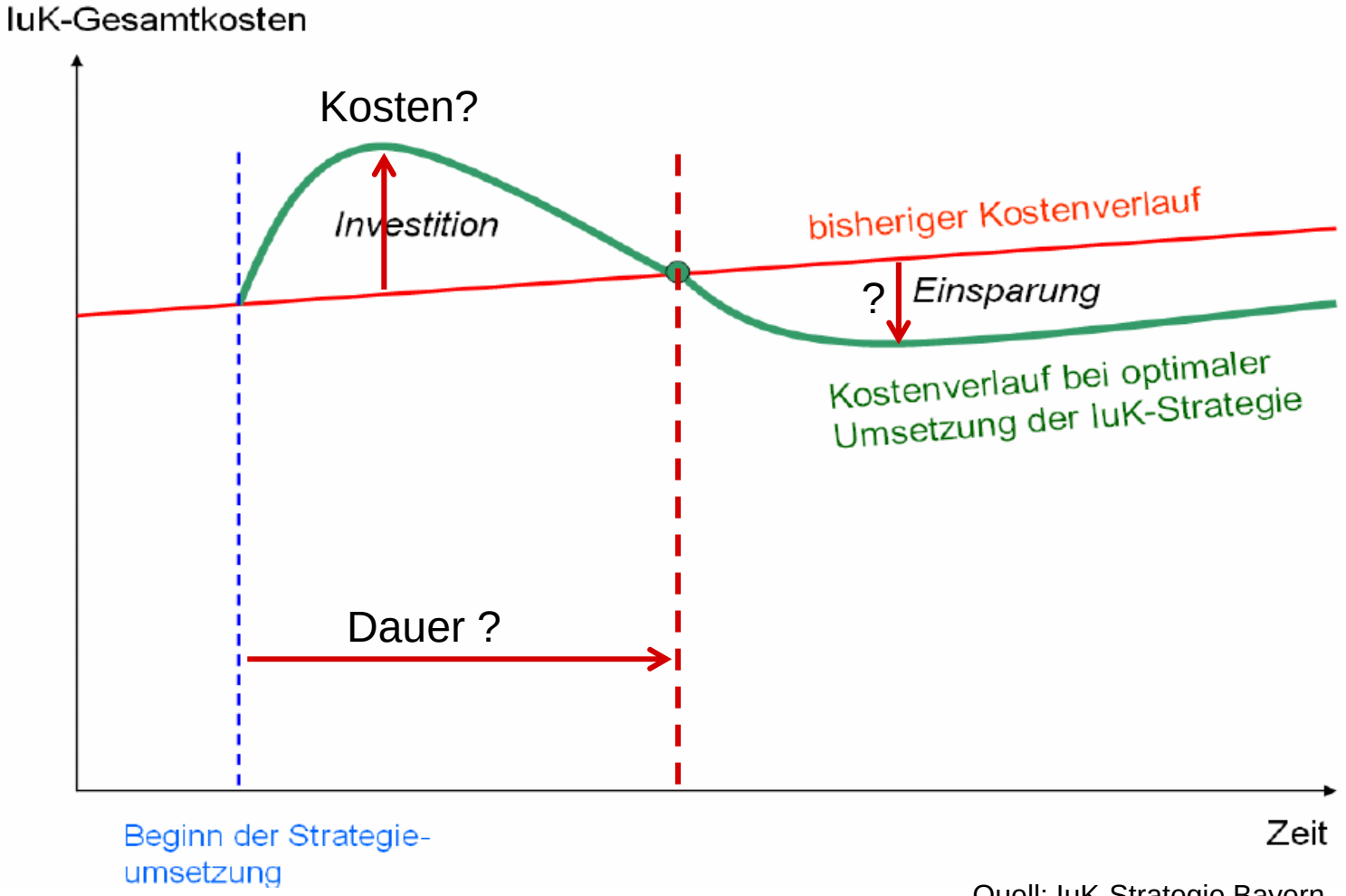
In dieser Welt kommt der Technologieaspekt nicht vor!

Eine SOA ist eine **langfristige Aufgabenstellung** und kann nur selten kurzfristigen Effekte erzielen.

Eine SOA ist in aller Regel mit einer **nicht unerheblichen Investition** verbunden.

Finanzierung [2]

Beispiel: Freistaat Bayern



Quell: IuK-Strategie Bayern

Warum scheitern SOA-Projekte?

Projekte scheitern aufgrund

- **unterbliebener Wirtschaftlichkeitsberechnungen**, weil man SOA-Projekten pauschal eine verbesserte Wirtschaftlichkeit zurechnet
  - → Überzogene Anforderungen an die Effekte einer SOA
  - → Rückzug aus SOA-Projekten, wenn kurzfristig keine Einsparungen erkennbar werden
- **zu geringer finanzieller Mittel**
  - → Notwendige Maßnahmen können nicht durchgeführt werden
  - → Der langfristige Erfolg bleibt aus und das Projekt wird ebenfalls wegen unzureichendem Effizienzgewinn beendet

Eine umfassende Wirtschaftlichkeitsberechnung ist mangels Erfahrungen und geeigneter Methoden nur schwer durchführbar

Organisation [1]

- Optimierung der Geschäftsprozesse ist im Wesentlichen eine Frage der Ablauf- und Aufbauorganisation
- Finanzielle Vorteile stellen sich ein, wenn
  - Strukturen vereinfacht
  - Abläufe optimiert
  - die Organisation auf die Geschäftsprozesse abgestimmt wird
- Wenn im Rahmen der Planung einer SOA auch die Geschäftsprozesse bearbeitet werden, kommt es zwangsläufig zu Konflikten hinsichtlich
  - Zuständigkeiten
  - Kostenverantwortung
  - Anordnungsbefugnissen

Organisation [2]

Projekte scheitern aufgrund

- falschem Projektansatz, da SOA oft als **reines Technologieprojekt** durchgeführt wird, ohne die Organisation zu berücksichtigen
- mangelnder **Einigungsfähigkeit** zwischen Abteilungen
- **Besitzstandswahrung**
- **fehlender Durchsetzungsmöglichkeiten** für Umstrukturierungen durch die SOA-Verantwortlichen
- **fehlender Unterstützung durch das Top-Level-Management**, das die notwendigen Umstrukturierungen anordnen könnte

Oft wird die **SOA-Governance**, die auch die organisatorischen Regelungen beinhaltet, unterschätzt.

Komplexität [1]

- Die Planung einer SOA ist sowohl **organisatorisch** als auch **technologisch** komplex
- Entwicklung der Architektur stellt **hohe Anforderungen** an die Fähigkeiten aller Beteiligten
  - Mitarbeiter der Fachabteilungen
  - Mitarbeiter der IT-Abteilungen
  - Projektleiter
- Insbesondere das **Projektmanagement** ist eine neue Art von Herausforderung, da viele Beteiligte für einvernehmliche Lösungen eingebunden werden müssen

Komplexität [2]

Projekte scheitern aufgrund

- unzureichendem Projektmanagement
- unterschätzter Komplexität
- unzureichender Methodenkenntnisse bei den Beteiligten
- Verzettelung in Details
- zu langer Projektdauer
- nicht realisierter Quick-Wins bzw. nicht erzielter Ergebnisse

Technologie [1]

- Eine umfassende SOA deckt eine Fülle von **technischen Aufgabenstellungen** ab, die erst einmal beherrscht werden müssen:
  - Zuverlässiges Messaging
  - Adaption von Schnittstellen
  - Abbildung von Prozessen
  - Entwicklung von (Web-) Services
  - Hinreichendes Identity- und Access-Management
  - Integration verschiedenster Anwendungen in Portalen
  - Zuverlässiger Betrieb mit definierter Qualität
  - …

Technologie [2]

Projekte scheitern aufgrund

- unzureichendem Reifegrad der am Markt verfügbaren Produkte, der für eine problemlose Integration aller Komponenten erforderlich wäre
- fehlendem Know-how der Mitarbeiter
- zu komplexer und teurer Lösung von Lieferanten

Bei fehlender SOA-Erfahrung der Beteiligten ist es schwierig, eine geeignete Lösung für eine Ausschreibung zu spezifizieren und in einem späteren Schritt einzuführen

Durchführung der Ausschreibung

- Juni 2006 — Architekturstudie der IBM im Auftrag der STK fertig
- Oktober 2006 — Beschluss für die Durchführung des Projekts
- Dezember 2006 — Finaler Projektauftrag an das LfStaD
  - Erstellung Ausschreibungsunterlagen auf Basis der Studie
  - Auswahl der Pilotanwendungen durch die STK bzw. den Ministerrat
- Februar 2008 — Auftrag an die Projective Expert Group zur Qualitätssicherung der Ausschreibungsunterlagen
- September 2008 — Neuerstellung der Leistungsbeschreibung fertig
- April 2009 — Bekanntmachung
- Juni 2009 — Anbieter geben Angebote ab

Stand der Arbeiten Anfang 2008

- Leistungsbeschreibung **nach fast zwei Jahren immer noch nicht fertig**
- **Uneinheitliche Anforderungen** hinsichtlich der beabsichtigten Systemteile der Infrastrukturlösung
- Leistungsbeschreibung basiert auf zahlreichen „Erläutern Sie“-Positionen, wobei ein **Bewertungsmaßstab fehlt**
- **Pilotanwendungen sind noch nicht spezifiziert**
- Verantwortliche Ressorts für die Pilotanwendungen haben **keine verbindlichen Zusagen** gegeben und auch **keine Manpower**
- **Technische Rahmenbedingungen** für die Schaffung geeigneter Services **ungeklärt**, da Ressorts die Anwendungen nicht selbst entwickeln, sondern über Dienstleister realisieren lassen
- **Finanzielle Kompensation** für Anpassung von Anwendungen **teilweise ungeklärt**

Schlüsselursachen für den Zustand des Projekts

- **Keine eigene Erfahrung** mit den Entwicklungsparadigmen verteilter Systeme bzw. einer SOA beim LfStaD
  - → Hersteller soll sagen, wie die SOA aussehen muss („Erläutern Sie …“)
  - → Die Erstellung der Leistungsbeschreibung hätte ausgeschrieben werden müssen → SOA-Architekt
- Kein definitives **Commitment** der beteiligten Ressorts
  - → Zuarbeiten unzureichend
  - → Aktivitäten in den einzelnen Ressorts nicht abgestimmt
- Keine institutionalisierte **SOA-Governance**
  - → Aufgrund der Ressorthoheit kann das LfStaD den einzelnen Ministerien keine verbindlichen Vorgaben machen

Problemfelder mit den Technologien [1]

Beispielhafte ungeklärte Aspekte für den Enterprise Service Bus

- Welche konkreten Sicherheitsanforderungen müssen erfüllt werden?
- Welche Adapter werden benötigt?
- Welche Art von Transaktionsfähigkeit wird benötigt?

Beispielhafte Probleme mit der Geschäftsprozessmodellierung

- Modellierungstool „Innovator“ wurde für die Fischerprüfung getestet, aber aufgrund der Tool-Komplexität dann nicht mehr verwendet.
- Die Ausbildung zur Bedienung des Tools wurde nicht hoch priorisiert.

Problemfelder mit den Technologien [2]

Beispielhafte Probleme beim Identitätsmanagement

- Kein zentrales Identitätsmanagement aus Gründen der **Ressorthoheit**
- Kein fertiges Konzept für ein **verteiltes Identitätsmanagement**

Beispielhafte Probleme bei den Pilotanwendungen

- **Zentraler E-Payment-Service nicht fertig spezifiziert** und die bisher spezifizierte Lösung ist erkennbar unzureichend.
- Für die Anwendung **E-Notar** ist nicht klar, ob die Anwender diese überhaupt brauchen bzw. in dieser Form wollen

Kontaktdaten

Dr. Frank Sarre

Anschrift: **Ludwig-Maximilians-Universität München (LMU)**
Institut für Informatik
Lehr- und Forschungseinheit für
Programmierung und Softwaretechnik (PST)
c/o Fr. M. Diem (Sekretariat von Hrn. Prof. Dr. M. Wirsing)
Oettingenstr. 67
80538 München

Telefon: **Tel. 089 / 2180 -9151 (Fr. Diem) oder**
direkt unter Tel. 089 / 18 92 37 -01

Email: **frank.sarre@pst.ifi.lmu.de**