

SOA Implementierung in der Luftfahrtindustrie am Beispiel der Lufthansa

Rolf Kubli, Michael Burkard, Elina Rantakallio, Annette Kreczy

EDS Information Business GmbH

Schaffhauserstrasse 550

CH-8052 Zürich

{rolf.kubli | michael.burkard | elina.rantakallio | annette.kreczy}@eds.com

Abstract: Dieser Beitrag zeigt die Lösung und die Erfahrungen eines wegweisenden SOA Projektes der Luftfahrtindustrie auf. Ausgehend von der EDS „Agile Enterprise Plattform“ und „Designed for RunTM“ SOA Strategie, wird eine Implementierung der „EDS Airline SOA Plattform“ für die einheitliche Einbindung (Middleware, Security) der Lufthansa Kernapplikationen in die „Common IT Plattform“ der Star Alliance dargestellt. Die Diskussion der Herausforderungen und Erfahrungen der Projektentwicklung, Einführung und Produktion bestätigt den erwarteten Nutzen eines SOA-Ansatzes, zeigt aber auch die Bedeutung des technischen Designs, der Testplanung, der Managementlösung und der Governance auf.

1 Einleitung

Ein Passagier der Austrian Airlines ruft im Call Center der Lufthansa an und möchte sein Business Ticket Wien über Frankfurt nach New York auf First Class ändern (upgrade) und mit seinem Meilenguthaben bezahlen.

Können Sie sich vorstellen, was ein solches Szenario-Beispiel für die IT-Systeme der betroffenen Fluggesellschaften bedeutet? Nun, es stellt grosse Anforderungen an die Integration aller wichtigen Applikationen (Reservation, Inventar, Check-in, Ticketing und Vielflieger Programm) der verschiedenen eigenständigen, aber als Mitglieder der Star Alliance zusammenarbeitenden Unternehmen. Es geht dabei darum, Transaktionen über unterschiedliche Datenformate, Protokolle und Generationen von Systemen schnell und zuverlässig abzuwickeln. Dies lässt erahnen, dass eine Service-orientierte Architektur ein vielversprechender Lösungsansatz für ein grundsätzliches, typisches Problem der Zusammenarbeit in der Transportbranche sein könnte und warum das Thema auf immer grösseres Interesse bei vielen Fluggesellschaften stösst.

Nach der häufig zitierten Definition von OASIS ist eine Service-orientierte Architektur (SOA) ein Paradigma für die Strukturierung und Nutzung verteilter Funktionalität, die von unterschiedlichen Besitzern verantwortet wird [Wi06]. Es gibt jedoch eine Reihe anderer Beschreibungen für eine SOA, so etwa auch die folgende. Eine Service-orientierte Architektur (SOA) meint ein allgemeines Modell der Datenverarbeitung geprägt von lose gekoppelten Elementen (Dienste, Funktionen, Komponenten)

und als Service exponierten Applikationen („wrapped“ legacy systems), welche über ein intelligentes Netzwerk publiziert, konsumiert und dynamisch kombiniert werden.

2 Herausforderungen in der Transportindustrie

Die Unternehmen der Transportindustrie sehen sich einer Reihe von Herausforderungen gegenüber, von der Globalisierung, über neue Konkurrenten, z.B. Billigflieger, bis zu ständig ändernden und anspruchsvolleren Kundenbedürfnissen. Ein Kunde erwartet heute ein nahtloses Zusammenspiel von Reisebüros, Fluggesellschaften, Hotels, Schifffahrtsgesellschaften und anderen Transportunternehmen bei seinen Geschäfts- oder Ferienreisen.

Angesichts der sehr heterogenen IT-Landschaften in den meisten Unternehmen, nämlich einer Mischung aus Jahrzehnte alten Legacy-Systemen und modernsten Internetanwendungen, dem extremen Wettbewerb zwischen den Reiseanbietern und immer neuen regulatorischen Auflagen ergeben sich hohe Anforderungen an die Reaktionsfähigkeit der Informatiksysteme. Sie müssen laufend den Änderungen der Geschäftsmodelle und der Geschäftsprozesse angepasst werden, zum Beispiel bei der Bildung von Allianzen oder der Lancierung neuer Angebote – und das mit möglichst geringen Kosten.

Vor diesem Hintergrund kann das Paradigma der Service-orientierten Architektur eine entscheidende Grundlage für die Modernisierung bestehender Software und den Bau neuer Applikationen werden und damit die Wettbewerbsfähigkeit der Unternehmen wirkungsvoll unterstützen. Siehe auch [Bu07].

EDS hat als führender IT Service Provider für die Transportindustrie vor einigen Jahren begonnen eine entsprechende Strategie für ihre Kunden zu entwickeln und in ein umfassendes Angebot investiert, die „EDS Airline SOA“. Dabei wurde von der Konzernstrategie der „Agile Enterprise Plattform“ ausgegangen und die Erfahrungen mit SOA aus anderen Branchen, z.B. der Finanzindustrie einbezogen. [KN07]

3 Die EDS Airline SOA

3.1 Agile Enterprise Plattform

Die „EDS Agile Enterprise Plattform“ ist eine Netzwerk-basierte Plattform, welche eine flexible Technologie-Basis schafft, die den Unternehmen hilft rasch und kosteneffizient auf Marktanforderungen zu reagieren und damit ihre Wettbewerbsfähigkeit zu steigern. Dabei ist Service-Orientierung eines der wichtigen Entwurfsprinzipien für Agilität, wie in Abbildung 1 dargestellt.

Die Service-Orientierung unterstützt als Integrationsparadigma auch direkt die Vorstellung, die mit dem Begriff „Digital Nervous System“ verbunden ist. Dabei wird die Informationsverarbeitung im Unternehmen mit der Funktionsweise des Nervensystems des Menschen verglichen. Diese Analogie wurde unter anderem von Bill Gates vor etwa 10 Jahren populär gemacht. Genau so wie ein lebender Organismus ununterbrochen sensorische und andere Informationen erfasst, verarbeitet und damit Aktionen auslöst, soll entscheidungsorientierte Informationsverarbeitung im Unternehmen nahtlos integriert funktionieren.

EDS Agile Enterprise Platform

SOA & Event Driven “DNS” for the Airline Industry

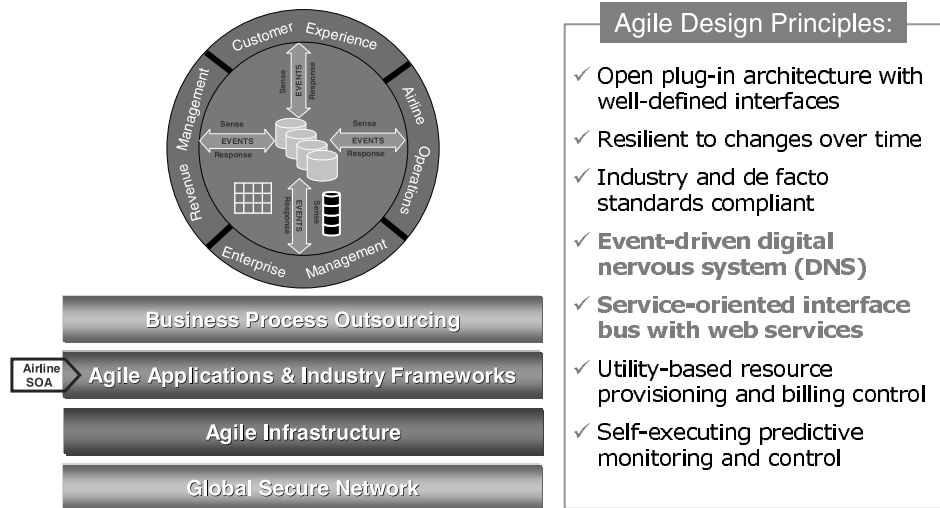


Abbildung 1: EDS Agile Enterprise Plattform

Die EDS SOA Strategie ist Teil des umfassenden Angebots der „Agile Enterprise Plattform“. Besonderes Augenmerk liegt dabei auf den Gesamtkosten (TCO Total Cost of Ownership) des SOA Vorhabens, zusammengefasst bezeichnet als “Designed for Run™”. Gemeint ist die Situation, dass üblicherweise das Augenmerk häufig nur auf der Entwicklungsphase liegt, statt alle Kosten einer voll funktionsfähigen Lösung im voraus zu berücksichtigen. Heutige Architekturen verlangen beispielsweise robuste Testumgebungen, welche den ausfallfreien 24-Stunden Betrieb realistisch nachbilden. Die vielen Elemente einer anspruchsvollen SOA Architektur müssen einwandfrei zusammenarbeiten, sonst ist der unterstützte Geschäftsprozess gefährdet.

Eine der ersten Gelegenheiten, die ganze Bandbreite der Erfahrungen einer vollständigen geschäftskritischen Lösung (Hardware, Netzwerk und Applikationen) zu machen, bot sich EDS in der Luftfahrtindustrie. [WS08]

Warum hilft eine SOA einem Unternehmen bzw. der IT des Unternehmens agiler zu werden? Weil die Interoperabilität und die Wiederverwendung von Standard-Services die Flexibilität erhöht und die Integrationsaufgaben erleichtert. Gleichzeitig werden Investitionen in bewährte, robuste Legacy-Systeme geschützt. Modernisierung dieser Legacy-Systeme ist deshalb ein Schlüssel der Transformation der IT Landschaft, um diese Funktionalität als Service im Rahmen der SOA nutzen zu können.

3.2 **Airline SOA Strategie**

Treibende Kräfte für den Wandel der IT in Richtung SOA sind im Fall von Fluggesellschaften typischerweise Prozesse für Interline Electronic Ticketing, Frequent Flyer Bonus Programme, Allianzen und Zusammenarbeitsabkommen aller Art zwischen Reise-dienstleistern, neue gesetzliche Vorgaben, etwa im Sicherheitsbereich, neue Vertriebskanäle, z.B. Buchung über Mobilgeräte und umfassende, präzise Informations-systeme für den Kunden. Verglichen mit der vollständigen Ablösung aller Systeme durch eine neue Generation in einem Zug, stellt der SOA Ansatz eine interessante Alternative für eine schrittweise Erneuerung derselben Systeme dar.

Wie geht man nun bei der Entwicklung einer SOA Strategie konkret vor?

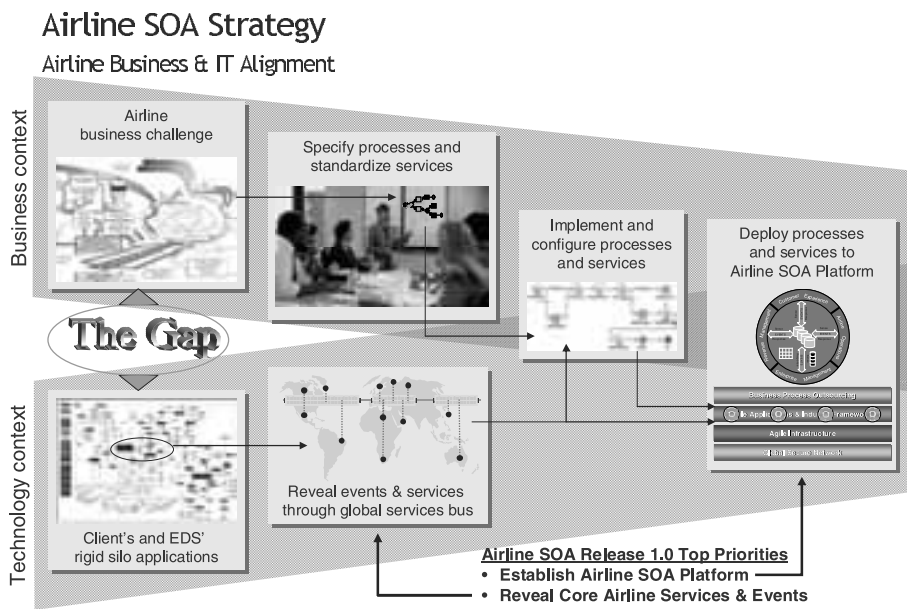


Abbildung 2: Airline SOA Strategie

Grundsätzlich geht man bei der Entwicklung einer SOA Lösung von den kritischen Geschäftsprozessen der Fluggesellschaft einerseits (Abbildung 2 oben), und den bestehenden Applikationen und Daten (Abbildung 2 unten) andererseits aus. Aufgrund einer Analyse der neuen, zukünftigen Anforderungen an die Geschäftsprozesse wird ein Plan entwickelt, die Lücken schrittweise zu schliessen (im Bild „The Gap“, Abstimmung IT und Geschäft [Fe07]). Normalerweise besteht die Kernaufgabe darin, die Integrationsschicht zu definieren (Enterprise Service Bus) und die bestehenden Applikationen, soweit sie zukunftsfähig sind, als standardisierte Services zu exponieren.

Welche Anforderungen und Herausforderungen nun im konkreten Falle der Lufthansa zu bewältigen waren, wird im Folgenden dargestellt.

4 Ausgangslage: Lufthansas Common IT Platform

Der Auslöser für die Umsetzung des SOA Projektes ist auf Lufthansas Entscheid zurückzuführen, auf ein neues von der Firma Amadeus entwickeltes Passenger Service System (PSS) zu wechseln. In einem solchen System werden nicht nur sämtliche Flugbuchungen und deren Änderungen verwaltet (Reservierungssystem), sondern auch die Sitzplatzverfügbarkeiten und die Gesamtauslastung eines Fluges (Inventarsystem), sowie die Passagier- und Gepäckabfertigung (Check-In System). Wie man daraus leicht erkennen kann fliessen eine grosse Anzahl an Inputparametern, wie z.B. Passagier- und Frequent Traveller Daten, Preise, Buchungsstände und Sitzplatzauslastungen, aber auch Informationen zu Flugplanänderungen und Anschlussflügen, um nur einige zu nennen, in einem solchen System zusammen. Diese Daten werden durch unterschiedliche externe Anwendungen gesteuert und müssen mit diesen kontinuierlich ausgetauscht werden. In einer ersten Phase sind etwa 30 Lufthansa Anwendungen betroffen, welche Daten mit dem Amadeus PSS System (und umgekehrt) austauschen. Viele dieser Anwendungen benutzen unterschiedliche Datenformate und Transportmethoden, was schnell dazu führen kann, dass viele Konvertierungsschritte von Nöten sind, um wieder miteinander kommunizieren zu können.

Aus dieser Situation heraus hat Lufthansa, als eines der ersten Star Allianz Mitglieder den bedeutenden Strategieentscheid gefällt, eine Middleware Plattform zwischen der Amadeus „Common IT Platform“ (CITP, siehe Abbildung 3) und den Lufthansa Anwendungen aufzubauen. Dabei sollte diese Plattform bereits im ersten Schritt so ausgelegt werden, dass sie später auch den anderen Star Allianz Mitgliedern, ohne grossen Zusatzaufwand, Zugang zu den gemeinsam genutzten Systemen gewähren kann. Damit war der Grundstein für die Middleware gelegt und das Lufthansa SOA (LH SOA) Projekt geboren.

Aber was sind denn nun konkret die grossen Vorteile einer solchen Plattform?

Common IT Platform will enable consistent, seamless service across Star Alliance

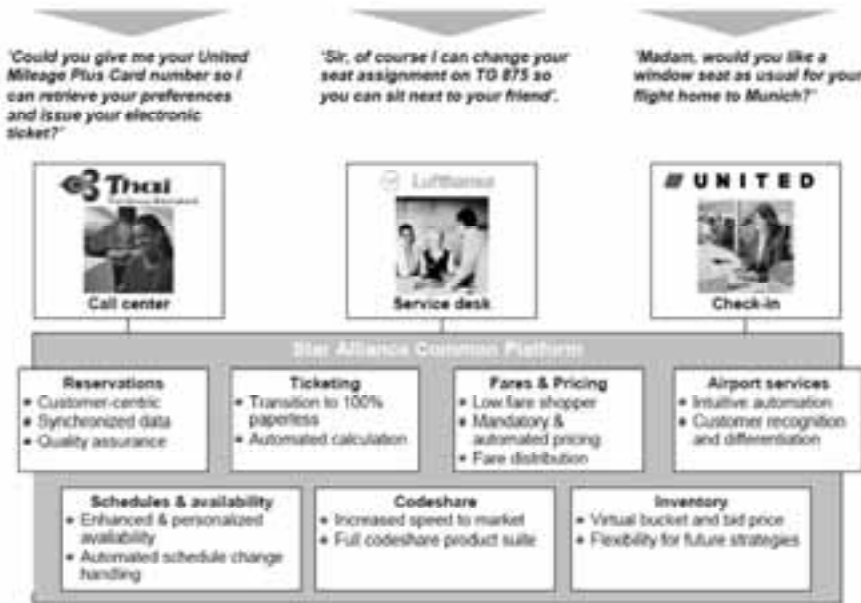


Abbildung 3: STAR Alliance Common IT Platform
(mit freundlicher Genehmigung von Lufthansa)

Mit der neuen gemeinsamen Plattform sind alle daran teilnehmenden Star Carrier (Fluggesellschaften) in der Lage harmonisierte und integrierte Services anzubieten. Dies führt dazu, dass die Carrier die Prozesse ihrer Hauptanwendungen gemeinsam nutzen können, gleichzeitig jedoch weiterhin ihre separaten Prozesse für spezifische Anwendungen beibehalten können. So gesehen verbessert CITP die Business Situation der gesamten Star Allianz wesentlich, was zu einem Wettbewerbsvorteil gegenüber den Konkurrenten führt.

Die zahlreichen Vorteile liegen auf der Hand:

- *Kostenreduktion:* Vor allem die IT Kosten in den wichtigen Bereichen wie Betrieb, Entwicklung und Wartung werden dadurch signifikant reduziert. Durch den Synergie-Effekt der gemeinsamen Nutzung kommt es zu weiteren Kosteneinsparungen für alle Beteiligten. [Mc06]
- *Vereinheitlichung der Services:* Durch die gemeinsame Nutzung der CITP ist es möglich Prozesse und Produkte der Star Carrier zu vereinheitlichen und so dem Kunden ein harmonisiertes Produkt anzubieten. Zudem werden Geschäftsprozesse wie beispielsweise das Durchchecken von Passagieren und Gepäck über mehrere Flugsegmente und Fluggesellschaften erleichtert.

- *Schnellere Markteinführung (time to market)*: Dank der CITP ist es möglich schneller neue Funktionalitäten und Produkte einzuführen. Diese können dann von allen teilnehmenden Carriern gemeinsam benutzt werden.

Basierend auf moderner Technologie und einer State-of-the-Art Implementierung unterstützt die CITP, dank der neu zur Verfügung stehenden Flexibilität bei reduzierten Kosten, die zukunftssträchtige IT Strategie der Lufthansa.

5 Zielsetzung des LH SOA Projektes und seine nicht-funktionalen Anforderungen

Ausgehend von der bestehenden EDS Airline SOA Plattform hat EDS für Lufthansa eine Middleware Plattform entwickelt, über welche die Anbindung verschiedener Lufthansa Backend-Systeme ermöglicht wird und welche somit zum zentralen Bestandteil der CITP wurde.

Um die Lufthansa Applikationen mit Altea, der Amadeus PSS Lösung, zu verbinden, wurden drei verschiedene Kommunikationskanäle (synchron, asynchron und Batch) zur Verfügung gestellt, deren Implementierung im nächsten Kapitel näher beschrieben wird.

Im Folgenden werden kurz einige der Wichtigsten nicht-funktionalen Anforderungen an die SOA Plattform aufgeführt:

Synchrone Kommunikation:

- Volumen von 800 Millionen Transaktionen pro Monat
- Spitzenwerte von 360 zustandslosen (stateless) Serviceaufrufen pro Sekunde
- Skalierbarkeit auf Spitzenwerte von über 720 zustandslosen Serviceaufrufen pro Sekunde ohne Architekturanpassungen

Asynchrone Kommunikation:

- Volumen von 230 Millionen Meldungen pro Monat
- Spitzenwerte von 300 Meldungen pro Sekunde
- Skalierbarkeit auf Spitzenwerte von über 540 Meldungen pro Sekunde ohne Architekturanpassungen

Weitere Kennzahlen:

- Middleware Verfügbarkeit von 99.95% pro Monat
- 100%-ige Redundanz (keine „single points of failure“)
- Unterstützung von 10,000 gleichzeitigen Nutzern
- Latenzzeit von max. 30ms (hin und zurück) für 90% der typischen synchronen Serviceaufrufe

und Rules, sowie die Versions- und Änderungskontrolle. SST liefert Logs und Reports über die Serviceaktivität sowohl auf Betriebs- als auch auf Geschäftsebene. Dies ermöglicht ein durchgängiges Service Level Management und Überwachung der verschiedenen Aktivitäten.

GhostAgents run continuously on each application server on the network that hosts web services detecting each incoming and outgoing message. The agents collect and evaluate run-time statistics and implement policies. The GhostAgents do not affect the runtime message flow and clients and servers are unaware that they exist.

GhostAgents laufen auf jedem Applikationsserver im Netz, welcher Web services hostet und detektieren alle Nachrichten. Diese Agenten sammeln und evaluieren Laufzeit-Statistiken und implementieren Regeln. Sie sind für die clients und server unsichtbar

Der LookingGlass (LG) Server ist die zentrale Engine, welche Nachrichtenflüsse, die von den GhostAgents kommen korreliert und Policies an diese Agenten verteilt. LG ermöglicht einen horizontalen Einblick in den Nachrichtenfluss innerhalb der Middleware und erlaubt so einem Administrator, Netzwerk und Daten zu analysieren und systemweite Policies zu erstellen. LG ermittelt darüber hinaus Abhängigkeiten von tatsächlichen Web Service Laufzeiten und generiert Performance Statistiken. Es ermöglicht auch gezielte Ursachenforschung bei Problemen oder Systemausfällen.

TIBCO Enterprise Service Bus

Die zentralen Nachrichtenflüsse auf Unternehmensebene wurden mittels TIBCO Enterprise Message Server (EMS), Business Works (BW) und Adaptern implementiert.

EMS liefert JMS 1.1 konforme fehlertolerante und persistente Nachrichtendienste, und – wo benötigt – Sicherheitsdienste mit einer Authentifizierung gegen ein LDAP im Unternehmens. Lastbegrenzungsmechanismen werden über Nachrichtenverfall und –steuerung ermöglicht.

BW stellt Orchestrierung, Transformation und Protokollwechsel-Funktionalitäten zur Verfügung, um die verschiedener Client und Backend-Applikationen und Prozesse zu integrieren und den Datenflusses zwischen verschiedenen Applikationen und Systemen sicherzustellen.

BW Adapter sorgen auf oberster Ebene für die Integration und Interoperabilität mit bestehenden Altapplikationen wie Unisys Mainframe oder MQ Series.

TIBCO Hawk und Administrator

TIBCO Hawk ist ein Werkzeug für das Monitoring und Management verteilter Applikationen und Betriebssysteme innerhalb der LH SOA, unter Verwendung der TIBCO Messaging Infrastruktur für die Kommunikation. Hawk ist ein ereignisbasiertes System, aufgebaut auf dem Konzept eines verteilten, autonomen, intelligenten Agenten, welcher auf jeder verwalteten Maschine im Netzwerk läuft. Es wurden über 200 LH SOA-spezifische Hawk Regelwerke implementiert, um einen nahtlosen Betrieb der Plattform

sicherzustellen und die strengen Hochverfügbarkeitsanforderungen zu erfüllen. Hawk meldet alle Ereignisse und Statusinformationen an den zentralen TIBCO Administrator, welcher über eine grafische Benutzeroberfläche das Management der LH SOA Komponenten ermöglicht.

BEA WebLogic

BEA WebLogic Server ist ein Applikationsserver für den Java 2 Enterprise Edition (J2EE) Standard welcher die Informationen und Funktionalitäten der Amadeus und LH Backend-Systeme für den Service Bus als Web Service exponiert.

GlobalScape

GlobalScape Enhanced File Transfer Server (EFT) stellt sichere und nicht gesicherte FTP Dienste zwischen Amadeus und LH zur Verfügung.

Oracle RAC Database (DB) und Application Server (AS)

In der LH SOA wird die ORACLE RAC 10g Datenbank eingesetzt, um Nutzungsstatistiken und Audit-Informationen zu protokollieren und Berichte zur Verfügung zu stellen. Die in der Datenbank gespeicherten Informationen umfassen Statistiken und Audit-einträge (logs), welche von den verschiedenen Komponenten produziert werden, z.B. von SST, LG und Hawk, sowie Metadatenverzeichnisse der verschiedenen SOA Komponenten, wie z.B. von Actional, TIBCO, ORACLE Portal, sowie dem Online-Reporting.

Andere ORACLE Produkte werden für das Management Portal, das Service Verzeichnis, LDAP und die Bereitstellung von Business Intelligence Funktionen in der LH SOA genutzt.

LH SOA Nachrichtenflüsse

Die LH SOA verwendet drei Hauptkanäle für den Nachrichtenfluss, die von den verschiedenen Arten von Clients und Backend-Applikationen genutzt werden.

Synchroner Transport-Kanal

Der Synchroner Transport-Kanal (synchrone Kommunikation) liefert HTTP/S und JMS Anbindungen für alle Web Service Aufrufe – hierbei dient die SST als zentraler Web Service Proxy. Die eigentlichen Web Service Lieferanten befinden sich hinter der SST und werden üblicherweise mit BEA WebLogic oder in TIBCO Business Works implementiert. Der Einsatz der SST entkoppelt so die Clients von den Backend-Systemen.

Clients können HTTP oder JMS-basierte Transport-Protokolle nutzen, jeweils mit oder ohne Secure Service Layer (SSL), wie in der folgenden Grafik dargestellt. Die Verfahrensregeln (Policies) für Web Services werden in SST konfiguriert, z.B. für Lastverteilung und Sicherheit (Security Policies) etc.

Die Abbildung 5 gibt einen Überblick über den synchronen Nachrichtenfluss: Anfragen treffen über HTTP oder EMS ein, werden von der SST verarbeitet, welche wiederum die benötigten BEA oder BW Web Services aufruft, die dann die zugehörigen Backend JAPI Objekte ansprechen. Die Backend-Antworten werden schliesslich der anfragenden Applikation via BEA und SST, entweder über HTTP/S oder eine temporäre Queue via EMS zurückgemeldet.

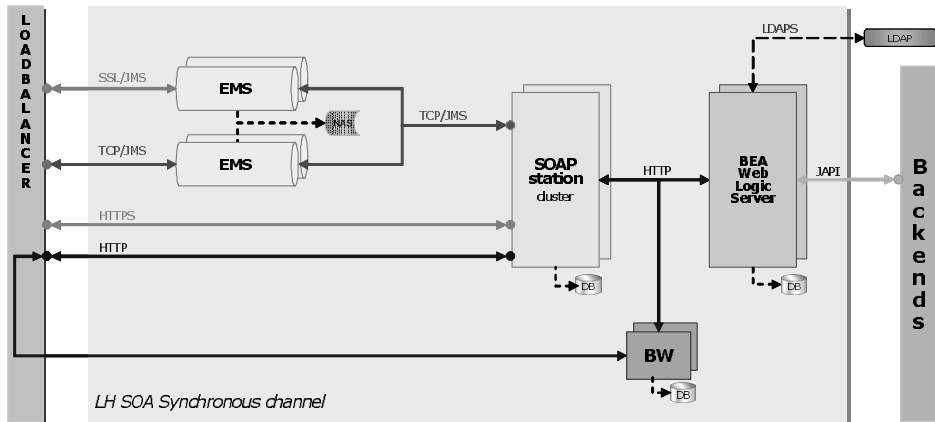


Abbildung 5: LH SOA Implementierung synchrone Kommunikation

Asynchroner Transport-Kanal

Die LH SOA unterstützt auch asynchrone Kommunikationsdienste. Eine Interaktion wird „asynchron“ genannt, wenn die den Service aufrufende Applikation nicht auf die Verfügbarkeit des Resultates der Operation wartet, oder wenn überhaupt kein Resultat erwartet wird. Der asynchrone Transport-Kanal der LH SOA unterstützt offene Standards, wie z.B. JMS, Punkt-zu-Punkt oder Publish/Subscribe Verfahren, sowie verschiedene Sicherheitsdienste. Ebenso ist er skalierbar und redundant konfiguriert, d.h. ohne „single points of failure“

Eine externe Applikation sendet eine Nachricht zu einer spezifischen Warteschlange (Queue) im Nachrichten System. Der Name dieser Warteschlange bezeichnet dabei jeweils die auszuführende Geschäfts- oder Integrationslogik. Die SOA Middleware erkennt eingehende Nachrichten in einer spezifischen Warteschlange und löst einen BW Integrationsprozess und die jeweilige Service Funktionalität aus.

Sofern der Service ein Ergebnis an die aufrufende Applikation oder an den Client zurücksenden soll, so erstellt der BW Prozess eine entsprechende Ergebnismeldung und sendet diese Nachricht an eine Ausgangswarteschlange. Der Client, der die Warteschlange „abhört“, ruft die Meldung ab, sobald sie in der Warteschlange erscheint. Der asynchrone Nachrichtenfluss ist in Abbildung 6 dargestellt.

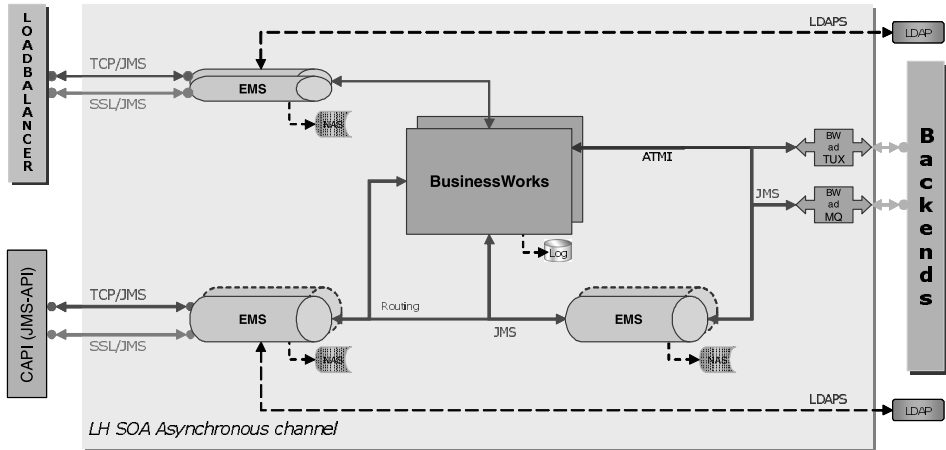


Abbildung 6: Implementierung asynchrone Kommunikation

Batch Transport-Kanal

Batch Prozesse werden über eine sichere FTP Umgebung innerhalb der LH SOA verarbeitet. Ein Client sorgt dabei für das Senden (Send) und Empfangen (Receive) von Dateien von externen Systemen. Dabei findet das Herauf- oder Herunterladen nie direkt mit dem externen System statt, sondern immer über die LH SOA. Die Middleware kümmert sich dabei um die regelbasierte Weiterleitung der Daten an die anderen Systemen. Dateien werden dabei von den externen Systemen auf dem Middleware FTP Server geladen. Die Middleware leitet sie dann zum Ziel-Client weiter oder bietet sie diesem zum Herunterladen an.

LH SOA Management

Um einen nahtlosen, hochverfügbaren Betrieb sicher zu stellen, ist ein effizientes Monitoring aller zentralen SOA Komponenten der LH SOA Plattform ein entscheidender Faktor. Dies wird durch LH SOA Management Werkzeuge auf verschiedenen Ebenen ermöglicht:

Horizontal: Nachrichtenflüsse werden durch Actional SOAPstation und Ghost Agents gesteuert

Vertikal: Das zentrales Management der einzelnen Komponenten wurde mit TIBCO Hawk Agents umgesetzt.

Das LH SOA Managementkonzept wird in der folgenden Abbildung 7 dargestellt:

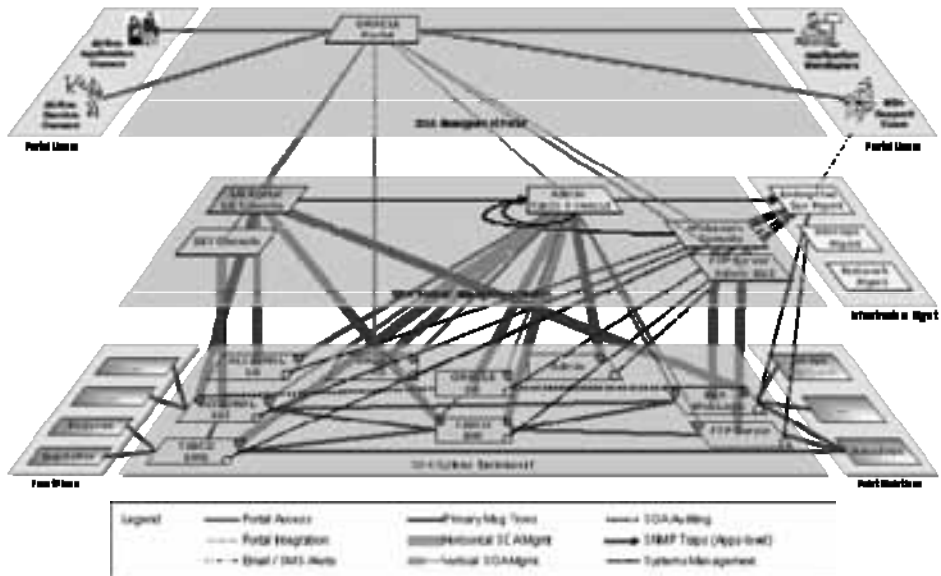


Abbildung 7: Überblick LH SOA Management

1. Die „SOA Systemumgebung“ (unterste Ebene) umfasst diejenigen Hardware und Software Komponenten, welche die SOA Runtime Plattform bilden und für den Nachrichtenfluss sorgen.
2. Das „SOA Platform Management Toolset“ (mittlere Ebene) liefert die Funktionalitäten für die horizontale SOA Nachrichtensteuerung und das vertikale SOA Komponenten Management. Darüber hinaus gibt es auf der Infrastrukturseite ebenfalls Werkzeuge für die Steuerung und Überwachung der Infrastruktur Services.
3. Das „SOA Management Portal“ (oberste Ebene) bildet das „Cockpit“ der SOA mit einer einheitlichen Benutzeroberfläche für das Monitoring und die Administration der gesamten SOA Plattform. Primäre Anwendergruppen hierfür sind die Verantwortlichen für die jeweiligen Dienste oder Applikationen, sowie Entwicklungs- und Betriebsteams.

7 Erfahrungen aus der Projekt Implementierung

Im folgenden Kapitel werden die wichtigsten Herausforderungen und Erfahrungen des LH SOA Implementierungsprojektes beschrieben.

7.1 Herausforderungen

Was das Design der Middleware angeht, so wurde auf dem bereits existierenden EDS Airline SOA Release aufgebaut, welcher jeweils mit den zusätzlichen Lufthansa Anforderungen ergänzt wurde. Trotzdem blieb ein hoher Level an technologischer Komplexität erhalten, welcher sich nicht zuletzt auch im Zusammenspiel der eingesetzten Produkte zeigte.

So ist es nicht erstaunlich, dass sich das *Provider Management* als eine der zeitaufwändigsten Aufgaben im Projekt herausstellte. Neben dem regulären Provider Tracking, welches man in jedem Projekt mit Drittanbietern wiederfindet, gab es im LH SOA Projekt vor allem technologische Herausforderungen, welche sich primär an den Schnittstellen zwischen den verschiedenen eingesetzten Produkte bemerkbar machten. Auch, dass viele der Drittanbieter Konkurrenten waren, kam erschwerend hinzu. So waren im Laufe des Projektes mehrere Task Forces von Nöten, in welchen die beteiligten Anbieter an einen Tisch zusammengebracht werden mussten, um eine konstruktive Lösung der aufgetauchten Probleme gemeinsam herbeizuführen.

Als zweite wichtige Erfahrung in diesem Projekt möchten wir das Thema *Testen* herausgreifen. So wurden die Testaufwände in der ursprünglichen Planung dieses Projektes stark unterschätzt, wobei noch hinzuzufügen ist, dass EDS im Auftrag der Lufthansa auch die Abnahmetests durchgeführt hat. Die hohen Aufwände sind auf verschiedene Faktoren zurückzuführen: Zum einen wurde der Aufwand für die Erstellung der Testfälle (es gab ca. 500 davon) und Abnahmekriterien, sowie die generelle Dokumentation der Testläufe unterschätzt. Zum anderen war vor allem der Aufwand für die Durchführung der Hochverfügbarkeitstests (Failover und Redundanz) wesentlich höher als erwartet. Jeder einzelne dieser Tests brachte grosse Planungs- und Koordinationsaufwände mit sich, beginnend mit der Verfügbarkeit aller involvierten Parteien über die Testfenster der Testplattformen bis hin zur eigentlichen Durchführung der komplexen Testfälle, welche in einigen Fällen für einen einzelnen Test alleine bereits 8 Stunden beanspruchte. Anschliessend folgten die Auswertung der Testergebnisse und allenfalls Suche nach Fehlerursachen sowie deren Behebung. Dieser ganze Zyklus musste je nach Testfall zum Teil mehrmals wiederholt werden.

Zusammenfassend lässt sich sagen, dass das Testen (inkl. gewisser Aufwände für das Anforderungsmanagement, das Erstellen der Abnahmekriterien und die Fehlerkorrektur) etwa zwischen $\frac{1}{4}$ und $\frac{1}{3}$ des Gesamtaufwandes des Implementierungsprojektes ausgemacht hat.

7.2 Erfahrungen & kritische Erfolgsfaktoren

Abschliessend möchten wir einige der wichtigsten Erfolgsfaktoren zusammenfassen:

- Sorgfältige Reviews der Projektaufwände während der Planungsphase mit Fokus auf die technologische Komplexität und die zu erwartenden Testaufwände

- Architekturdesign auf dem richtigen Granularitätslevel, welches vor allem das Potential für spätere Wieder- und Weiterbenutzung berücksichtigt
- Setup einer Projektmanagement Organisation, welche in der Lage ist gleichermaßen effizient die global verteilte Stakeholder, Drittanbieter sowie On- und Off-Shore Teams zu führen
- Sicherstellung bzw. rascher Aufbau einer adäquaten technologischen Wissensbasis mit vertieftem Anwendungs-Know-how bei allen Beteiligten
- Proaktive Teilnahme an Industrie Foren
- Motivation des Projektteams über einen langen Zeitraum und insbesondere in schwierigen Projektphasen
- Zielorientierte Führungskräfte, die sich der Komplexität und technologischen Herausforderungen eines solchen Projektes bewusst sind
- Definition und Umsetzung geeigneter Führungsgremien (Governance-Boards)
- Zentrales Produktmanagement der SOA Plattform inklusive aller damit verbundenen Themenbereiche

8 Zusammenfassung und Ausblick

Das vorliegende Beispiel illustriert die erfolgreiche Realisierung und das Nutzenpotential eines SOA Ansatzes. Eine Service-orientierte Architektur hilft die Forderung nach Flexibilität und gleichzeitiger Kosteneffizienz unter einen Hut zu bringen, indem gemeinsam genutzte Services durch Integrations- und Kommunikationsmechanismen verbunden werden. Auch in diesem Projekt zeigt sich der Trend, dass der Nutzen von SOA direkt von der Leistungsfähigkeit der Integrationsebene, d.h. der Middleware, bestimmt wird und erfolgreiche SOA Ansätze stark Nachrichtenfluss-zentriert sind.

In diesem Sinne hilft eine „agile IT“ einem Unternehmen „agil“ zu werden, da sowohl Kostenoptimierung als auch eine verbesserte Reaktionsfähigkeit bei der Weiterentwicklung von Geschäftsprozessen gleichzeitig angestrebt werden können.

Das SOA Projekt mit der Lufthansa zeigt den Weg in die Zukunft. So haben bereits weitere Fluggesellschaften Interesse angemeldet, für welche EDS derzeit eine modifizierte Version der EDS Airline SOA implementiert.

9 Danksagung

Die Autoren danken dem Architekturteam unter der Leitung von Peter Schmid für die Unterstützung der Ausarbeitung dieses Beitrags.

10 Literaturverzeichnis

- [Bu07] Burger, L., SOA IN THE TRANSPORTATION INDUSTRY, EDS Viewpoint Paper, 2007.
- [Fe07] Are You Really Ready for SOA? CIO, http://www.cio.com/article/121850/Are_You_Really_Ready_for_SOA_
- [KN07] Kubli R., Nadhan E.G., BANKING ON AN SOA EVOLUTION, EDS Viewpoint Paper, 2007.
- [Mc06] Lufthansa IT Chief Looks To Cut Costs With SOA Software, InformationWeek, <http://www.informationweek.com/story/showArticle.jhtml?articleID=195300016>
- [Wi06] Wikipedia, http://de.wikipedia.org/wiki/Serviceorientierte_Architektur
- [WS08] Interview, itp, SOA Roundtable, <http://www.itpreport.com/default.asp?Mode=Show&A=1314&R=G>