

~~Editorial: Was machen die Schweizer IT-Provider falsch?~~

~~Neuerungen: V2.2 Amazonas Workflow~~

~~SOA (Serviceorientierte Architektur)~~

~~Business Processing Ebene~~

~~Service Provider Ebene~~

~~Enterprise Integration Ebene~~

~~XPDL vs. BPEL~~

~~Auflösung letzter Wettbewerb~~

~~Neuer Brückenwettbewerb~~

~~Info Bridge kostenlos abonnieren / Feedback~~

Editorial: Was machen die Schweizer IT-Provider falsch?



Liebe Leserin, lieber Leser



Personalfreizügigkeit, off-shoring und near-shoring sind in der letzten Zeit auch im IT-Umfeld ein wiederkehrendes Thema. Da ist die Managersicht, aus welcher es gilt, Kosten um alles in der Welt zu minimieren. Alles was zuviel kostet (und was tut dies nicht) wird ausgelagert, erfahrene Arbeitskräfte ab einem bestimmten Alter ausrangiert, Saläre beschnitten. Das ist verständlich, wenn es darum geht, die restlichen Arbeitsplätze zu erhalten, weniger verständlich hingegen, wenn es damit den Unternehmensgewinn für die Aktionärsicht zu verschönern gilt. Off-shoring ist dann natürlich eine willkommene Möglichkeit, um zu "Bruchteilen" der Kosten ein paar Tausend Kilometer weit weg programmieren zu lassen. Ob in dieser Rechnung immer alle Aufwendungen und Nebenkosten miteinbezogen werden, das möchte ich bezweifeln. Man muss auch die zurückkommenden Erzeugnisse genau unter die Lupe nehmen und untersuchen, ob diese genau den gesetzten Erwartungen entsprechen. Ein Unterfangen, das schon schwierig genug ist, wenn man lokal bleibt. Sicher hängt der Erfolg davon ab, wie gut ein solches Projekt vorbereitet

(Anforderungsspezifikation!!) und dokumentiert ist, wie gute Verbindungsleute man vor Ort hat und welchen Umfang das Projekt hat. Ein gut durchdachter und eingespielter Prozess wird hier erfolgsrelevant. Man kann es auch so ansehen, dass bei einem Viertel der hiesigen Ansätze dreimal soviel Fehler gemacht werden "dürfen" und man dabei immer noch günstiger fährt. Ob diese Rechnung aufgeht? Im Speziellen wenn mit der Zeit die Kunden vom Offshorer "abhängig" sind, und der Wirtschaftsboom auch dort eingezogen ist und entsprechend die Preise sich nach oben orientieren werden. Schlussendlich werden die Offshorer als intelligente Leute verkauft und dies sind sie auch, aber auch seit Jahrhunderten als clevere Händler bekannt.

Ein nächster Faktor ist die Reaktionszeit, die es braucht, wenn zusätzliche, vom Markt diktierte, neue Anforderungen an

das zu entwickelte oder gewartete System gestellt werden. Und das alles beim heute stark interaktiven und iterativen Entwicklungsprozess. Über soziale Verantwortung und den Langzeiteffekt für unseren Schweizerarbeitsmarkt kann sich jeder seine Gedanken selbst machen.

Wenn Sie nun denken, ich würde mich für unser "Vrenelis-Gärtli" wehren, dann möchte ich doch anmerken, dass wir, d.h. die inländische IT-Branche sich selber an der Nase nehmen muss. Es gilt die Frage aufzuwerfen, ob die geforderten Arbeitsergebnisse nicht in halber Zeit in doppelter Qualität und Professionalität geliefert werden könnten? Gerade hier sollten wir in der Schweiz alles daran setzen, diesen rohstoffarmen Marktbereich zu erhalten. Bei einer alle Kosten miteinbeziehenden Rechnung von Anforderungen bis Wartung, mit eingespielten Technologien und fitten, effizienten und gut ausgebildeten Teams, bin ich überzeugt, dass wir auch in der Schweiz zu konkurrenzfähigen Preisen produzieren könnten. Jede IT-Organisation hat heute ein Verbesserungspotential, das mit Wille und Konsequenz mittelfristig auszuschöpfen ist. Leerläufe und Politikum sind dabei als Erstes auszumerzen, gefolgt vom Sicherstellen von klaren Anforderungen, Qualität und Effizienz im Entwicklungsprozess. Der Kunde wird sicher nichts dagegen haben, wenn er sich mit dem Lieferanten, der das Umfeld kennt, in seiner eigenen Sprache verständigen kann. Ich würde sogar den Beweis antreten, dass wir in einem Einpersonenjahr-Projekt zu Fixpreis, alle Kosten einberechnend, gleich teuer oder günstiger produzieren können, als ein Offshorer. Frech ich weiss, aber wer nimmt die Herausforderung an?

Ich wünsche Ihnen beim Lesen der InfoBridge 6 viel Vergnügen.

Mit freundlichen Grüssen

Andres Koch
Dipl. El. Ing. HTL, M. Math
Geschäftsführer
Object Engineering GmbH

Wie immer freuen wir uns auf Ihr Feedback auf die vorliegenden Beiträge und beantworten Ihnen auch gerne Ihre Fragen. [Kontaktformular](#)

Neuerungen: V2.2 Amazonas Workflow



Die Amazonas Process Engine ist im vergangenen halben Jahr kräftig weiterentwickelt worden. Im April wurde die Version 2.1.6 freigegeben. Der nächste Release ist die Version 2.2.

Engineering	Produkte	Komponenten	Support
-------------	----------	-------------	---------

Bitte füllen Sie **alle Felder** dieses Formulars aus und senden Sie es ab. Ihre Anfrage wird umgehend beantwortet.

Firma	<input type="text" value="DCA Steuerungen AG"/>
Abteilung	<input type="text" value="Entwicklung"/>
Vorname	<input type="text" value="Huggeler"/>
Name	<input type="text" value="Paul"/>
Telefon	<input type="text" value="062 765 43 34"/>
E-Mail Adresse	<input type="text" value="phuggeler@dca.com"/>
Support-Bereich	<input type="text" value="Steuerung 2"/>
Benutzer-Name	<input type="text" value="phu-dca"/>
Problem-Beschreibung	<input type="text" value="Trotz Restart beim Transfer hängen geblieben"/>

Start eines Prozesses direkt aus dem Formular

Bald täglich füllen wir Web-Formulare aus und senden sie ab. Wenn es sich nicht um einen Einkauf über einen Webshop handelt, so erfolgt noch oft eine Verarbeitung per E-Mail oder "manueller" Art. Dass Reaktion oder Antwort ausbleiben, haben wir alle schon erfahren - die Meldung mag verloren gegangen sein.

Dem AWF Designer (Tool zur Gestaltung von Prozessen) ist eine Formular-Generierungs-Funktion hinzugefügt worden, so dass beim Gestalten des Prozesses ein **Web-Formular generiert** wird. Diese generierte Seite ist eine Rohseite und kann vom Web-Designer auf das aktuelle CI angepasst werden.

Mit dem **Startup-Servlet des AWF Client** wird also mit dem Ausfüllen und Absenden des Formulars **direkt ein Prozess instanziiert** (ausgelöst) und dieser Prozess durchläuft alle Aktivitäten **bis zur vollständigen Erledigung**. Beispiele dafür sind Support-, Service-, Kundendienst- oder Hotline-Prozesse.

Diese können sehr **einfach** ausgeprägt sein, z.B. das Backoffice nimmt die Meldungen entgegen und leitet sie an die verschiedenen Kundenbetreuungsstellen weiter, erledigt nach Abschluss das wenige Verbleibende in Sachen Administration und hat natürlich den Überblick über die Pendenzen.

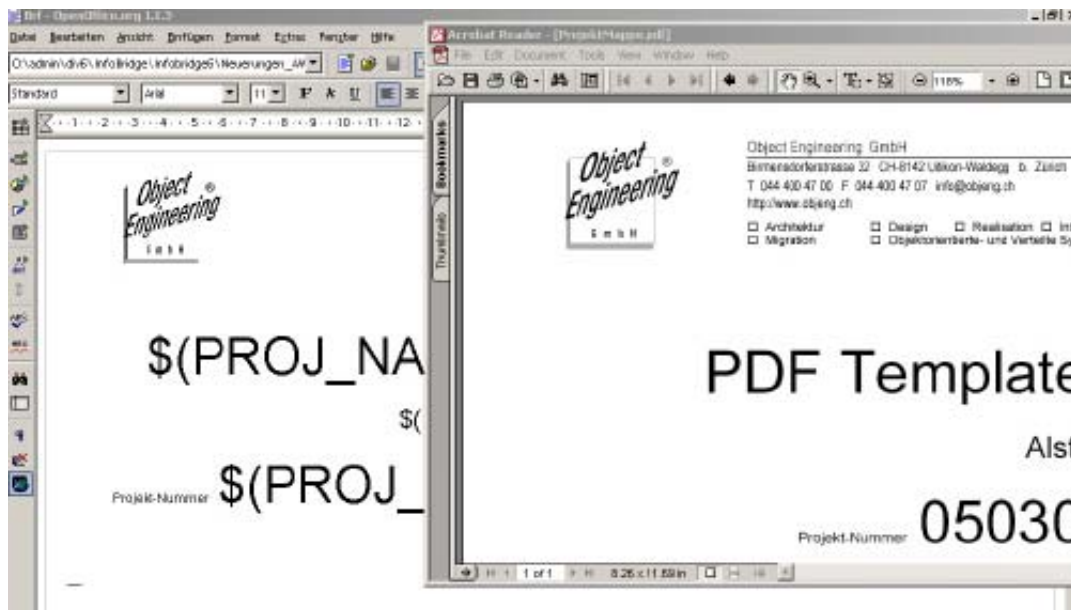
Nur schon eine solch einfache Steuerung von Abläufen mag in einem Unternehmen ein sichtbares Plus an Ordnung und ein spürbares Minus an Telefonaten bewirken.

Solche Prozesse können **schrittweise verfeinert und erweitert werden** durch:

- Automatische Identifikation des Kunden
- Einfügen von Kundendaten aus verschiedenen Stämmen (z.B. Adressdaten, Anlagedaten)
- Automatische E-Mail Benachrichtigung über den Stand der Aktivität an Kunde und andere betroffene Stellen
- Eskalierung im Verzugs- oder Problemfall
- Auslösen von Subprozessen z.B. für Rechnungsstellung

Ein besonderer Vorteil ist, dass diese Prozesse **jederzeit den Gegebenheiten entsprechend angepasst werden können** - ohne grossen Aufwand!

Für bestimmte Prozesse mag es notwendig sein, dass in einer oder mehreren Aktivitäten automatisch Dokumente oder Formulare ausgedruckt werden müssen. Mit der **Worklet Open Office-PDF-Konversion** können solche **Reports** oder **Formulare automatisch generiert** werden.



Vorlage im Open Office, Druck als PDF

Ein einfaches Beispiel ist in einem Auftragsadministrationsprozess das Ausdrucken eines Deckblattes oder einer Etikette für eine Projektablagebox. Diese Funktion ist auf der Basis des Programmes Open Office realisiert worden. Open Office (ein Open Source Tool) wird auf dem Server installiert und damit werden die Templates für die verschiedenen Formulare erstellt.

Zur **Archivierung** oder **Weiterverarbeitung** ist eine **PDF-Generierung** eingebaut worden, die sowohl **Aktivitäten**, als

auch ganze Prozesse mit den **dazugehörigen Dokumenten als PDF-Datei** anlegt und am gewünschten Ort archiviert.

Als vierte Neuerung ist eine Option geschaffen worden, die erlaubt, **ausführliche** und **komplexe** Reports zu generieren. Diese Funktion ist auf der Basis von Jasper-Reports erstellt worden (Open Source Tool) und kann PDF, CSV und HTML generieren. Die Erstellung der Vorlagen erfordert XML Kenntnisse oder ein entsprechendes Design-Tool.

Diese zusätzlichen Funktionen sind Optionen, die zur AWF Engine angeboten werden.

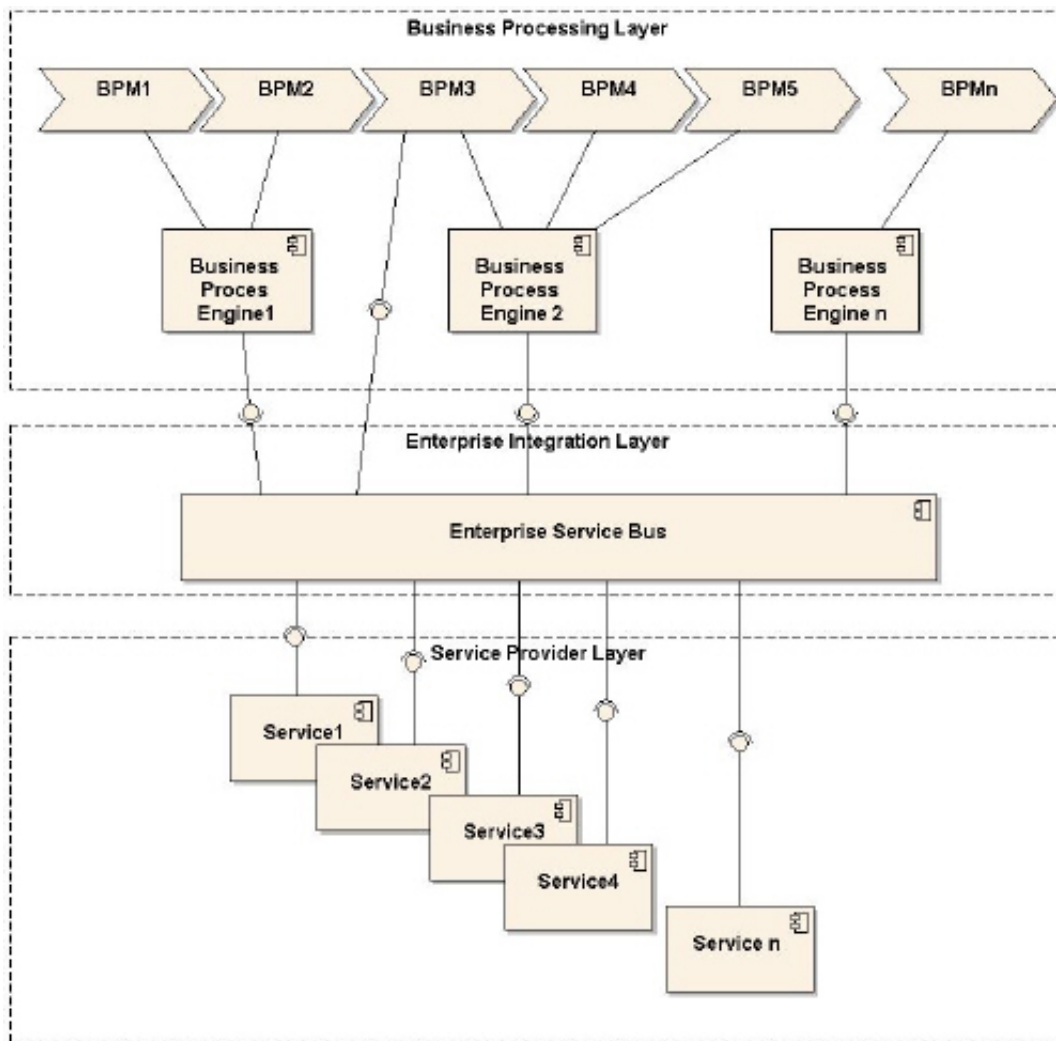
Roger Müller, Marketing & Verkauf

SOA (Serviceorientierte Architektur), die angeblich neu erfundene Lösung der ungelösten IT-Probleme?



Wie im letzten Artikel (InfoBridge 5) aufgezeigt, ist SOA nichts Neues (sarkastische Bemerkung eines Mitarbeiters SOA = **S**ame **O**ld**A**rchitecture), sondern ein bewährtes Muster konsequent umgesetzt und auf der Basis aktueller Technologien (Messaging, WebServices, Applikations-Servers) aufgebaut.

Aufbau



Typische Service orientierte Architektur

Schauen wir den Aufbau einmal genauer an, dann sehen wir, dass sie in drei Hauptschichten zerlegt werden kann:

- **Business Processing Ebene**
- **Enterprise Integration Ebene**
- **Service Provider Ebene**

Diese Schichtung kann auch als konsequente Weiterentwicklung früherer Architekturen bezeichnet werden. So war die in den Neunzigerjahren aktuelle Client-Server-Architektur ein Vorgänger dazu. Aber auch diese wurde selten wirklich konsequent durchgesetzt, gab es doch oft nur eine Client-Applikation zu einem Server (Service). Damit wurden die Services nicht wirklich mehrfach eingesetzt. Die darauf folgende 3-Tier-Architektur, die oft im Client-Server-Umfeld realisiert wurde, war mit ihrer Aufteilung in Präsentations-, Business- und Persistenz-Schicht schon näher beim heutigen SOA-Gedanken. Konsequente Umsetzung war auch in diesem Fall oft an Budgetgrenzen gestossen oder von höheren Stellen von der Projektliste gestrichen worden.

Ein wichtiges Problem besteht darin, dass die Geschäftsabläufe in monolytischen Applikationen (oft Legacy-Applikationen) fest "einbetoniert" wurden. Aber gerade der Business-Prozess ist die Komponente, welche flexibel und "schnell" den aktuellen Marktanforderungen und der Geschäftsstrategie angepasst werden soll.

Das ist der Hauptaspekt, dass die Geschäftsprozesse dorthin verlagert werden, wo sie "verstanden" und gebraucht werden, auf Business-Ebene (Business Processing Layer) zum Geschäft. Mittels den Geschäftsabläufen programmiert man das Unternehmen. Mit Java, Cobol und anderen Sprachen programmiert man die Business-Komponenten auf der Service-Provider-Ebene.

Business Processing Ebene

Die Präsentations-Schicht der Client-Server-Architektur wird in der Business-Processing-Ebene "versorgt", wo sie durch Workflow-Systeme typisch auf Weboberfläche oder direkt in einer spezialisierten Benutzeroberfläche - zum Beispiel eine hochinteraktive Benutzer-Oberfläche für Callcenters - dem Benutzer präsentiert wird. Wer diese Benutzerzugänge entwickelt, ist eine Frage der Firmenkultur. Empfehlenswert ist es diese Entwicklung wie die Geschäftsprozesse selbst,

durch geschäftsnahe Entwickler (Business Engineers) definieren zu lassen, damit sie den Anforderungen der Geschäftsbenutzer optimal entsprechen.

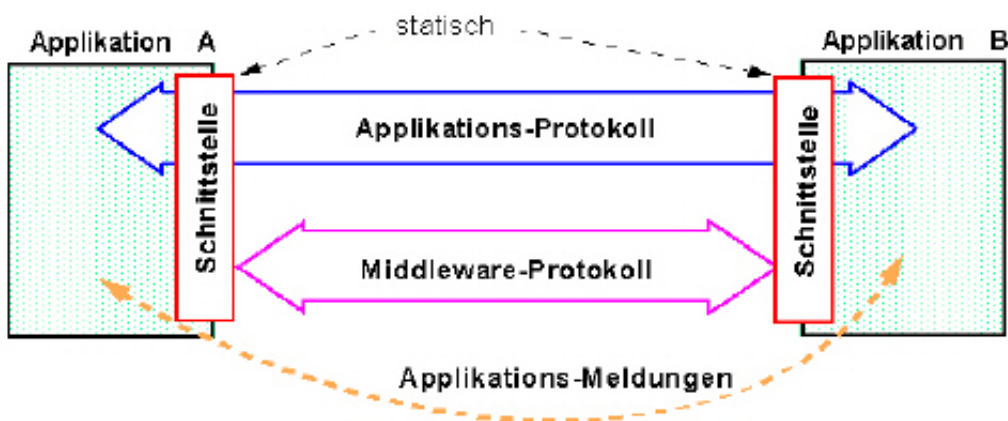
Durch das SOA-Konzept werden die übergreifenden Abläufe stärker von den Services abgekoppelt. Damit wird auch die typische Symmetrie zwischen Client (typisch Benutzerinterface) vom Server aufgelöst.

Die Geschäftsprozesse werden nicht nur in Visio oder Powerpoint "gemalt" sondern müssen in Schritten automatisiert oder zumindest automatisch überwacht werden. Das heisst aber auch, dass "Programmieren der Unternehmung" zur präzisen Tätigkeit wird. Wenn aus Geschäftsprozessen Services (für das steht ja SOA) aufgerufen werden, dann müssen klare Definitionen vorgelegt werden und "Schwammigkeit" und Interpretationen in Varianten liegen da nicht mehr drin. Da trifft die "harte" IT-Realität den Business Analysten oder mindestens den Business Process Engineer, eine Rolle resp. Stelle, die bei der Realisation von SOA auf der Business Processing Ebene vorgesehen und vorallem akzeptiert werden muss. Nochmals: Geschäftsprozesse gerecht für ein Workflow-System zu definieren, ist eine Präzisionsarbeit und muss umso konsequenter durchgesetzt werden, je mehr automatisch ablaufen soll.

Service Provider Ebene

Lassen sie uns nach unten zur Service Provider Ebene springen. Diese Schicht wird in eine Interface- und in eine Implementations-Schicht unterteilt. Die Interface-Schicht repräsentiert die Schnittstellen, service-orientiert nach aussen. Services werden in, die Kern-Geschäftslogik abbildenden Komponenten aufgeteilt, die aber nur mit ihrer Schnittstelle in der Interface-Schicht erscheinen. In der Implementationsschicht werden entweder neue Services implementiert, Services als eingekaufte Softwarekomponenten (COTS = Commercial Off The Shelf) integriert oder in wohl den häufigsten aller Fälle, bestehende Applikations-Systeme an die SOA-Schnittstelle adaptiert. Diese Mischung wird nach bewährten aber konsequenten Software-Engineering-Muster vollzogen, worauf im Detail dann in einem anderen Artikel eingegangen werden wird.

Der Leitspruch heisst hier: Bestehendes und Neues uniform nach aussen gegenüber der Business Processing- und Business Integrations-Ebene durch eine logisch klare Schnittstelle darstellen. Das **A&O** ist Schnittstellen-Design (sprich D-E-S-I-G-N!), welches den effektiven Anforderungen (d.h. aufgrund einer Anforderungsanalyse) der Business Processing Ebene erstellt und nicht nach Gutdünken des IT-Engineers ersonnen wird. Hier müssen der Business Process Engineer und IT-Architekt oder IT-Engineer eng zusammenarbeiten.



Zusammenhang verschiedener Protokolle

Wird SOA konsequent umgesetzt, dann muss mit der Zeit in einem mittleren bis grossen Unternehmen mit hunderten von Services gerechnet werden. Wie dies zu verwalten ist, das ist die nächste Frage, um nicht zu sagen das nächste Problem. Ohne die richtigen organisatorischen Massnahmen könnte sich hier schnell ein Chaos anbahnen. Dazu gibt es mehrere Aspekte die zu organisieren sind:

- Service-Katalog: Nicht nur aktive auch inaktive Services, ihre Schnittstellen (und Formatbeschreibungen der verarbeitenden Meldungen), ihre Funktionen und das zu verwendende Protokoll müssen publiziert werden, damit sowohl der Business-Process-Engineer, der IT-Architekt und auch der IT-Entwickler diese Services verwenden können. Nichts ist schlimmer als wenn eine professionell entwickelte und getestete Komponente nicht zum Einsatz gelangt, das ist wie ein Rennpferd, das immer im Stall steht. Eine andere Gefahr ist, wenn für ein neues Projekt ein neuer Service entwickelt wird, der eigentlich in einer Form bereits vorhanden ist. Dazu kann man entweder eine spezielle Verwaltungssoftware verwenden oder ganz einfach eine interne Wiki-Page (siehe www.wiki.org). Dabei ist darauf zu achten, dass diese Service-Verwaltung auch als Service angeboten wird.
- Zur Laufzeit müssen installierte und gestartete Services auch gefunden werden. Dazu braucht es eine

Namensverzeichnis-Komponente (z.B. COS-NS, UDDI u.a.)

- Weiter benötigt man ein Metadata-Repository, worin alle in Meldungen und Schnittstellen verwendeten Attribute deklariert sind. Nur so weiss sowohl der Business Process Engineer, wie der IT-Architekt, was z.B. mit "KDNR" gemeint ist oder was "CNTRCTID" innerhalb einer XML-Meldung zu bedeuten hat. Dieses Metadaten-Repository muss sowohl maschinell (als Service), wie auch über ein Web-Interface abfragbar sein. Das sollte von einer verantwortlichen Person verwaltet und gepflegt werden, um Wildwüchse zu vermeiden. Service-Katalog und Metadata-Repository haben eine enge Verwandtschaft.

Diese drei Instrumente sind praktisch unerlässlich für den erfolgreichen SOA-Einsatz.

Enterprise Integration Ebene



Als fehlenden Teil unserer Betrachtung gilt es nun noch den Business-Processing-Layer mit dem Service-Provider-Layer zu verbinden. Wir nennen diesen Teil den Business-Integrations-Layer, welcher mit dem in der letzten InfoBridge Nr. 5 vorgestellten Enterprise Service Bus in der Regel in die Praxis umgesetzt wird. Dieser Layer hat auch die Aufgabe die "Aussenwelt" in unsere SOA zu integrieren.

Ein Beispiel wäre: eingehende Anfragen vom externen Web-Server oder einer Lieferanten-Firma an den richtigen Service oder dafür vorgesehenen Business-Prozess zu leiten.

Ein Enterprise Service Bus (ESB) bildet eine Art Telefonzentrale zwischen allen Beteiligten und leitet eine Applikationsmeldung an den richtigen Empfänger und nach dessen Bearbeitung wieder weiter an den nächsten. Das heisst, der Service muss nicht wissen wohin die Meldung, die er eben bearbeitet hat, weitergesendet wird. Er gibt diese mit den von ihm zugefügten Daten einfach an den ESB zurück, der seinerseits dann aufgrund des Inhalts der Meldung diese an den nächsten Service weiterleitet oder zur Weiterverarbeitung an den zuständigen Business-Prozess zurückgibt. Diese Weiterleitungsszenarien nennt man Orchestrierung. Die Aufgaben des ESBs sind:

- Sicherheit (Authentication/Authorisation)
- Vermittlung (Naming, Service Directory)
- Routing aufgrund des Inhalts von Meldungen an den korrekten Service
- Transformation von Meldungsformaten
- Plattformverfügbarkeit und Skalierbarkeit
- Normierte Schnittstellen resp. Verbindungs-Technologien

Erfahrung und Konzepte aus früheren EAI-Konzepten wurden mit neuen Funktionen wie Datentransformationsfunktionen, inhaltsbezogenes Routing und mit sogenannten Service-Containers (Service Adapter) ergänzt. Es gibt ganze Plattformen, die diese Funktionalitäten anbieten. Der ESB bildet praktisch die "Telefonzentrale" zwischen aussenstehenden Services (oder Client-Applikationen), Client-Applikationen auf der Business Processing Schicht und den internen Services der Service Provider Ebene. Sie ist sehr flexibel konfigurierbar, integriert sich mehr oder weniger mit allen Arten von normierten Schnittstellen und bietet eine gewisse Workflow-Funktionalität. Letztere ist aber nicht zu verwechseln mit dem Business-Workflow (wie auf der Business Processing Ebene), sondern dient zur Koordination der Integration. Die dazu geeignete Workflow-Sprache ist BPEL (Business Process Execution Language) und nicht XPDL. Wobei zu bemerken ist, dass das 'B' in BPEL wohl falsch am Platz ist (siehe Kasten XPDL vs. BPEL).

Damit ist unser "Durchstich" durch die SOA-Schichten beendet und hat ihnen hoffentlich einen besseren Einblick gegeben. SOA ist als aktuelles Schlagwort im Moment sehr beliebt. Die Umsetzung davon braucht aber eine gehörige Portion von Organisations-Talent, technischen Durchblick und das je nach Projektvorhaben angepasste Portemonnaie. Die Realisierung einer SOA kann nicht off-shore verlegt werden, denn es ist ein integrativer Prozess, der auch nicht innerhalb eines Jahres abgeschlossen ist, sondern schrittweise vor sich geht.

Fazit

Wie im vorliegenden Artikel aufgezeigt wurde, verlagern sich Abläufe, die früher in den einzelnen Applikationen einverleibt waren, auf die Ebene der Business Processing Schicht. Dies verspricht zwar mehr Flexibilität zu gewinnen, verschiebt aber gleichzeitig die Verantwortung und den Know-How-Bedarf genau auf diese Ebene. Ob der Stakeholder das entsprechend vorgesehen und die Massnahmen dafür geplant hat, kann nur er/sie beantworten. Dieser Artikel kann nur mutmassen. Die kritischen Bereiche wurden zumindest aufgeführt, so dass der Leser darauf ein offenes Auge richten kann. SOA ist eine Herausforderung, doch mit Bedacht umgesetzt, wird dieser Ansatz eine langfristige Architektur sichern.

Andres Koch, Technische Leitung



Im Bereich Workflow sind heutzutage diverse Standards definiert und werden von verschiedenen Interessensgruppen weitergetrieben und unterstützt. Für die Definition von Prozessen geniessen zwei Sprachen weite Verbreitung: XPDL und BPEL. Während beide Sprachen XML-basiert sind, unterscheiden sie sich in deren Aufbau trotzdem beträchtlich.

- **XPDL** wurde von einigen Herstellern von Workflow Systemen innerhalb der Workflow Management Coalition als gemeinsame Prozess-Sprache entwickelt und enthält somit die Elemente, die für moderne Prozesse, ob manuell oder automatisch, notwendig sind. Modelliert nach dem Prinzip der gerichteten Graphen (Digraph), gibt es sozusagen keine Einschränkungen auf den Aufbau des Prozesses. XPDL beinhaltet das Konzept von Workflow-Teilnehmer und -Rollen, welches menschliche Akteure sowie auch Applikationen umfasst. Somit kann die Benutzerinteraktion und der Aufruf von externen Programmen innerhalb eines Prozesses genau definiert werden.
- **BPEL** hingegen kennt keine Rollen. Definiert in Zusammenarbeit von IBM und Microsoft ist diese Sprache auf die Orchestrierung von Webservices ausgelegt und ist für diesen Zweck auch besser geeignet; somit wird aber auch die Annahme gemacht, dass innerhalb des Prozesses nur Webservices aufgerufen werden. Diese Unzulänglichkeit wird jedoch durch die Unterstützung von Transaktionen und Fehlerbehandlung zumindest teils wettgemacht, während in XPDL diese Semantik fehlt und solche Funktionalität durch die Prozess-Logik explizit definiert werden muss.

Weil BPEL auf einer Block-Struktur aufsetzt, ist sie restriktiver als XPDL im Prozess-Design. Durch die Popularität von Web-Services wird BPEL von vielen kommerziellen Produkten unterstützt. Daraus sollte aber nicht der Fehlschluss gezogen werden, dass BPEL für alle Workflow-Anwendungen die sich besser eignende Technologie ist. Vorallem bei interaktiven Prozessen zeigt XPDL seine Stärken und Vorzüge und geniesst unter anderem auch im Open-Source Umfeld breite Akzeptanz.

Roman Zulauf,
Komponenten-Entwicklung

Auflösung letzter Wettbewerb:



Brückenwettbewerb Infobridge Nr. 5

Bei der in der Infobridge Nr. 5 abgebildeten Brücke handelt es sich um die Passerelle von der EXPO-02 in Biel, über welche Fussgänger den Hafen überqueren konnten.



Es gab insgesamt 10 richtige Einsendungen. Herzliche Gratulation!

Der ausgeloste Hauptgewinner ist:

Herr Urs Jutzi, Swisscom IT Services AG, Bern.

Er gewinnt eine Rundfahrt auf dem Bielersee für 2 Personen.

Neuer Brückenwettbewerb



Hoch hinaus! Die höchste Brücke der Welt wurde vor gar noch nicht so langer Zeit eröffnet.



In welchem Land steht diese Brücke? so lautet unsere Wettbewerbsfrage diesmal.

Die Antwort senden Sie uns wie immer per Fax 044 400 47 07 oder per [Kontaktformular](#). Aus den richtigen Antworten wird die Gewinnerin oder der Gewinner mit dem Los ermittelt und erhält eine passende Überraschung.

Einsendeschluss: 31. August 2005