Schliesslich geht es ja darum, mit einer SOA eine klare Ordnung zu schaffen und gleichzeitig eine hohe Flexibilität zu ermöglichen. Denn der SOA-Ansatz bedeutet ja gerade nicht, dass mit einem Schlag alles geändert wird, sondern dass man zeitlich gestaffelt bestimmte Funktionalitäten als Services konzipieren und zur Verfügung stellen kann. Somit amortisieren sich die Kosten für die aufwendige Vorarbeit relativ schnell.

In welchem Zeitraum konnten die Kosten für Ihr Projekt im Zusammenhang mit WincoLink amortisiert werden?

Eindeutig ist das nicht zu beantworten, weil die SOA-Reise immer noch weitergeht. Für ein einziges Projekt würde sich die Investition auch kaum rechnen. Das tut sie erst dann, wenn man sie auf einen grösseren Teil der IT-Landschaft ausdehnt. Ich schätze aber, dass sich die Investition in eine SOA-Infrastruktur nach vier, fünf Jahren rechnet. Und Sie dürfen nicht vergessen, dass wir 1998 vor der Entscheidung standen, entweder all die umfangreichen und teuren Legacy-Applikationen für viel Geld durch heutige Lösungen zu ersetzen oder eben mittels Kapselung und einer SOA-Schicht Service-tauglich zu machen. Natürlich werden diese alten Kernapplikationen nicht ewig in Betrieb bleiben. Mit dem SOA-Ansatz haben wir aber den Vorteil, dies – und auch die abgestufte Konsolidierung der Hardwaresysteme – sowohl technisch als auch Business-mässig kontrolliert ablaufen zu lassen.

Für welche Geschäftsfelder haben Sie bislang Ihre Applikationen SOA-fähig gemacht?

Zuerst passten wir wie erwähnt die Applikationen für das Leben-Geschäft mit Grosskunden an. Zudem laufen auch die Partnerverwaltung, die Nichtleben-Schadenerfassung und -abwicklung und die Vertragsverwaltung in der SOA-Umgebung. Gegenwärtig sind wir daran, auch die Einzel-Leben-Systeme Service-tauglich zu machen.

Wenn Sie im Rahmen Ihrer SOA-Strategie alle Abläufe sauber definieren, standardisieren und dokumentieren, könnten Sie dann die Funktionalitäten nicht ebensogut auslagern und Utility-artig nutzen?

Ich sehe diesen Trend sehr wohl im Bereich des Infrastrukturbetriebs – etwa um zu vermeiden, dass Investitionen in Systeme an Peak-Belastungen ausgerichtet werden müssen, die nur ein- oder zweimal im Jahr auftreten. Aber selbst hier ist hohe handwerkliche Konzentration notwendig, um den versprochenen Nutzen zu haben. Grundsätzlich bin ich jedoch nicht dafür, die Verantwortung für Kernanwendungen auszulagern. Ausserdem dürfte es sich mit dem Utility- oder On-Demand-Computing ähnlich verhalten wie mit SOA – dass nämlich nicht alles auf einmal umgestellt werden muss, sondern dass man eine Strategie verfolgt, mit der gezielt bestimmte Leistungen ausgelagert werden. Dies dort, wo der Nutzen hinreichend hoch ist. ■

Über Martin Frick

Martin Frick ist Leiter IT der Winterthur Group, hat Physik und Mathematik studiert und 1990 an der Ruprecht-Karls-Universität in Heidelberg doktoriert. Danach forschte er in Computational Physics an der Universität Groningen, bevor er bei Thinking Machines in Cambridge, USA, als Application Engineer tätig war. Von 1994 bis 1999 arbeitete er als Manager bei Andersen Consulting (heute Accenture) in München, und von 1999 bis 2002 amtete er als Chief Operating Officer bei der virtuellen Bank Enba in Dublin. Seit 2002 ist Frick Chief Information Officer bei der Winterthur Group.

ARCHITEKTUR & ENTWICKLUNG: TOBIAS MURER

JavaOne – zehn Jahre kontrollierter Darwinismus

Seit zehn Jahren gibt es Java und ebensolange die dazugehörige Entwicklerkonferenz JavaOne. Sie gilt als weltweit wichtigstes Treffen der Java Community. In letzter Zeit aber mehren sich Stimmen, die die Tiefe, Qualität, Innovationskraft und Business-Lastigkeit der Konferenzbeiträge bemängeln. Da auch die Teilnehmerzahl der diesjährigen JavaOne das Allzeit-Hoch von 1999 mit über 20'000 Besuchern wohl verfehlen wird, drängt sich die Frage auf, ob zehn Jahre JavaOne genug sind.

Abgesehen vom detailversessenen Java-Geek dürfte diese Frage aber von den meisten Besuchern verneint werden. Denn die Konferenz gilt weiterhin als treibende Inspirationsquelle und realer Zusatz für die Java Community. Sie ist der Nährboden und Ort, wo aktuelle Ideen ausgetauscht, gesät und verbreitet werden – in Form von einer Art Aktivferien für Entwickler. Denn im Gegensatz zu virtuellen Foren bietet die «reale» Austauschplattform JavaOne die Möglichkeit, über das persönliche Gespräch mit Leuten, die die Technologie vorantreiben, frühzeitig die wichtigen Tendenzen zu erkennen und zu beeinflussen.

In der von der offenen Java Community betriebenen Technologie-Entwicklung läuft ein Prozess ab, der am besten als darwinistisch zu beschreiben ist: Sobald ein bestimmter Problembereich durch bestehende Technologie ungenügend abgedeckt ist, etabliert sich in der Community eine bessere Alternative. Bei mehreren Lösungen für den gleichen Problembereich setzt sich rasch die «fitteste» Lösung durch. Bei diesem gemeinsamen Prozess besteht aber die Gefahr der Verzettelung. Für die darum notwendige Abstimmung spielt die jährliche Zusammenkunft an der JavaOne eine zentrale Rolle. Sie sorgt im Zusammenspiel mit den virtuellen Foren als Regulativ sozusagen für einen «kontrollierten Darwinismus».

Werden an der JavaOne aber auch die aktuellen technischen Herausforderungen adressiert?

Ein erster Blick auf das Programm zeigt ein breites Angebot an entsprechenden Beiträgen und stimmt positiv. Am Anfang der Java-Ära war die Innovationsrate noch eindrücklich hoch, nicht zuletzt weil Sun Microsystems unter starkem Druck stand, kontinuierlich neue Technologien zu lancieren, typischerweise mit der JavaOne als Präsentationsbühne. Dies hatte seinen Niederschlag in Kompromissen, die teilweise auch im Design der Technologien zu finden waren.

Der Preis für den Erfolg ging auf Kosten von weniger sichtbaren Technologien. Diese hätten es erlaubt, von Anfang an eine Java-Anwendungsplattform auf professionellem Niveau zu betreiben. Gemäss dem Konferenzangebot sind derzeit jedoch etliche Anstrengungen zu erkennen, diese Lücken zu schliessen. So muss insbesondere das Angebot an Technologien zur Realisierung von kleineren, einfacheren und trotzdem effizienten und qualitativ hochstehenden Anwendungen weiter optimiert werden, um im Wettbewerb mit anderen Techniken zu bestehen.

Diese Herausforderungen werden mit aktuellen Lösungen wie Java ServerFaces, Spring, Hibernate und J2EE 5.0 adressiert. Zudem ist es erfreulich zu sehen, dass der Technologie-Darwinismus in der Java Community funktioniert, da einerseits Alternativen zum J2EE-Standard aufgezeigt werden, andererseits sich solche Alternativen in künftigen J2EE-Standards wieder finden. Spannend dürfte auf jeden Fall der erste Auftritt Microsofts an der JavaOne werden. Die Community wird genau hinschauen.

Tobias Murer ist Head of Java Engineering bei der Zürcher AdNovum. Sie erreichen ihn unter tobias.murer@adnovum.ch