

Conway's Law

Carsten Sensler, Michael Heinke
ArtOfArc/ SV-Informatik
Dortmund/ Mannheim

Schlüsselworte

SOA, Contract First, Transformation, SOAfizierung, OSB, CEISeR, Service Repository, SOA Governance

Einführung - Conway's Law

Die SV Informatik ist ein Systemhaus für öffentliche Versicherungen. Sie stellt die Weiterentwicklung sowie den sichereren und performanten Betrieb für die IT-Landschaften ihrer Kunden – der SV Sparkassenversicherung, der Hamburger Feuerkasse und der Sparkassen-Versicherung Sachsen - sicher.

Die heutige GmbH ist entstanden aus einer Fusion von öffentlichen Versicherern. Die einzelnen Ursprungshäuser brachten alle ihre eigene Anwendungslandschaft in die neue Gesellschaft. In den 00ern wurde in einem mehrjährigen Migrationsprojekt diese mit redundanten Komponenten besetzte Anwendungslandschaft hochgradig konsolidiert.

Mit dieser Konsolidierung wurde eine eindeutige Verortung fachlicher Fähigkeiten auf jeweils ein System bzw. eine Systemkomponente erreicht, so dass die Anwendungslandschaft heute aus einer Reihe von Client-Server Systemen besteht, denen jeweils eindeutige Fachverantwortungen zuzuordnen sind.

Mit der einhergehenden Reduktion der Anzahl an Systemen wurde eine Reduktion von Systemabhängigkeiten erreicht. Nichtsdestotrotz führt eine solche verteilte Anwendungslandschaft zu einem höheren Schnittstellenbedarf zwischen den Systemen als in einer vergleichsweise monolithischeren Anwendungslandschaft.

Naheliegend in einer solchen Konstellation ist die SOAfizierung der Anwendungslandschaft, d.h. die Standardisierung der Kommunikationsbeziehungen über Services und die Abbildung fachlicher Prozesse durch die Orchestrierung der in der verteilten Anwendungslandschaft gehosteten Services in einer Business Process Engine (BPE).

Die SV Informatik skizzierte daher im Jahre 2009 eine Plattformstrategie, die eben dieses anstrebte: Die Bereitstellung fachlicher Services aus den jeweils für die Domäne zuständigen Systemen, die Entkoppelung der Systeme über das Routen der Kommunikation über einen zentralen Enterprise Service Bus (ESB) und die Flexibilisierung eben jener Geschäftsprozesse durch die Orchestrierung mithilfe einer BPE. Zeitgleich wurde – quasi als Untermauerung – eine Service-Richtlinie verabschiedet, die eine standardisierte und zielgerichtete Umsetzung der Strategie gewährleisten sollte.

Im weiteren Verlauf entstanden peu a peu die ersten Services und Prozesse. Eine „SOA-Keimzelle“, die das operative Umsetzen der Strategie durch einen Projektpfad unterstützen sollte wurde initiiert. Es wurden technologische Grundlagen geschaffen, organisatorische Prozesse aufgesetzt und Schulungsmaßnahmen durchgeführt.

Von einem Aufbruch des Unternehmens in Richtung einer SOA unter aktiver Einbringung der Mitarbeiter war allerdings nichts zu spüren.

Ursachenforschung wurde betrieben. Handwerklich war das Vorgehen in Ordnung. Es stellte sich allerdings heraus, dass die Organisationsform der SV Informatik im damaligen Status Quo für die Umsetzung einer SOA hinderlich war. Die Organisation ist System-bezogen aufgestellt, d.h. jede Entwicklungseinheit ist für ein oder mehrere Systeme zuständig und trägt für diese die volle Verantwortung bzgl. Stabilität, Fehleraufkommen und Entwicklungsgeschwindigkeit. Diese Verortung führt zu einer starken Einkapselung der Abteilungen, nach dem Motto „Störe meine Kreise nicht“.

Conway stellte in seinem Essay „How Do Committees Invent“ die These auf, dass Organisationen zwangsläufig Systeme (im allgemeinen Sinne) bauen, die den Kommunikationsbeziehungen der Organisation entsprechen. Die Forschung zu dieser These ist nicht eindeutig, hier allerdings traf es eindeutig zu: Kommunikation zwischen den Abteilung fand in eingeschränktem Maße statt. Lösungen, die innerhalb einer Abteilung erstellt werden konnten wurden Lösungen, die eine Abteilungsübergreifende Zusammenarbeit erforderten, vorgezogen.

Es wurde klar, dass ohne einen Change, der die Interaktion und Kommunikation zwischen den Abteilungen nicht veränderte, eine SOA nicht stattfinden würde.

In diesem Kontext sind insbesondere vier verschiedene Bereiche – PPTI - , in denen die Veränderung bedingt durch eine SOA sich zeigt. Es sind die Bereiche

- Menschen (People)
- Prozesse (Process)
- Technologie (Technology)
- Information (Informaion)

Im Folgenden verdeutlichen wir, inwiefern diese Bereiche durch die Einführung unserer SOA berührt waren/ sind.

Menschen:

Wir haben ein SOA-Team als virtuelles Team, querschnittlich zur Linienorganisation, etabliert. D.h. wir haben aus jeder Entwicklungsabteilung zwei Entwickler rekrutiert, die nach entsprechender Schulung Speerspitzen für SOA in den einzelnen Abteilungen sein sollten. Wir wollten keine neue reale Linieneinheit, da wir damit ja auch wieder Grenzen zu den anderen Abteilungen gezogen hätten. Wir glauben auch, dass ein SOA-Team auch ein Mindestmaß an Fachwissen der jeweiligen Domänen mitbringen muss, um in der Service- und Prozessentwicklung optimal mitzuwirken. Eine separate Einheit hätte das Domänenwissen im Laufe der Zeit verloren.

Um erfolgreich eine Transformation – und die Einführung einer SOA ist eine große Transformation – zu gestalten, hat sich bewährt, dass man neben anderen Faktoren insbesondere Unterstützer findet. Unterstützer zum einen in dem Sinn, dass aus dem Management dem Transformations-Team der Rücken gestärkt wird, wenn es einmal zu Eskalationen kommt und zum anderen in dem Sinn, dass auch genügend Atem in Form von Geduld und auch Budget bewiesen wird. Eine SOAfizierung eines Unternehmens ist keine Angelegenheit von ein paar Wochen. Wir reden hier eher von Monaten und

Jahren. Eine weitere Form von Unterstützern bzw. Unterstützung ist, dass Freunde in den Anwendungsentwicklungs-Abteilungen (AEs) gesammelt werden. Aus Erfahrung lässt sich sagen, dass es sehr hilfreich ist, wenn andere Gutes über das Transformations-Team verbreiten und somit für die Sache an sich Werbung gemacht wird. Erreichen kann man dieses, indem das Transformations-Team sich insbesondere zu Beginn der Transformation sehr stark in der Beratung und Hilfe der AEs engagiert. Dieses ging sogar soweit, dass das Transformations-Team Aktivitäten übernommen hat, die später im Regelbetrieb eigentlich die Aufgabe der AEs sein wird.

Information:

Wir leben konsequent einen „Contract First“ Ansatz, d.h. vor jeglicher Entwicklung bzgl Services müssen die Service Entwickler des Consumers und Providers einen formalen Vertrag schließen, der die Fachlichkeit des Services, deren intendierte Nutzung, SLAs, etc. festlegt. „Vertrag“ verstehen wir hier wörtlich: Es muss eine von beiden Seiten tragbare Vereinbarung getroffen werden. Nichtsdestotrotz ist der Provider eines Services offensichtlich in einer stärkeren Verhandlungsposition. Nicht nur um diese zu egalisieren, aber auch darum, setzen wir ein Common Data Model (CDM) ein. Dadurch, dass sich beide Parteien auf Strukturen in einer ihnen nicht nativen Sprache einigen müssen, müssen sie in einen intensiven Dialog gehen, der keinem einen unfairen Vorteil gibt. Darüber hinaus entkoppeln wir mit dem CDM die Schnittstellen von den Systemspezifika und fördern die Konzentration auf die Fachlichkeit eines Services.

In der Literatur finden sich oftmals skeptische Einschätzungen gegenüber einem CDM. Eine Problematik, die häufig angesprochen wird, ist die der Änderungsabhängigkeit von Services in Bezug auf die Strukturen eines CDM. Diese Problematik haben wir auch gesehen und daher uns darauf zurückbesonnen, dass es bei einem CDM im Kontext SOA nicht um ein Modell für bspw. eine Persistenz geht, sondern um eine Sprache, die das Definieren von Nachrichten unterstützt.

Für uns ist daher ein CDM nicht ein Modell im klassischen Sinne, sondern eher ein Wörterbuch, welche die fachliche Attributierungen enthält, die in der Formulierung von Nachrichten-„Sätzen“ erforderlich sind. Strukturen und Beziehungen wie man sie in einem Modell erwartet finden sich in diesem Wörterbuch kaum wieder. Dieses Wörterbuch ist gegenüber Änderungen unempfindlich: Erweiterungen des Wortschatzes führen zu neuen Ausdrucksmöglichkeiten, ändern aber keine bestehenden Sätze, sind daher im Sinne SOA kompatible Änderungen und bedingen keine Versionierung bestehender Services.

Technologie:

Der Punkt „Technologie“ ist vermeintlich der schwierigste. Allerdings hat sich gezeigt, dass man die technologischen Herausforderungen und Probleme im Allgemeinen leichter lösen kann, als viele andere Herausforderungen der SOAfizierung.

Die Entscheidung der SOA Laufzeit Umgebung wurde gefällt. Sie fiel auf den Oracle Service Bus (OSB). Es gab ein Team, welches manuell die notwendigen Proxy-Flows in der OEP (OSB Entwicklungsumgebung) modelliert und in die Laufzeitumgebung eingespielt hat. Durch die manuelle Tätigkeit und auch das Lernen über die Zeit hat allerdings in einer Vielzahl von unterschiedlichen Ausprägungen der Proxy-Flows geführt. Es gab auch kein zentrales Logging- und Service-Monitoring Konzept sowie fehlte es an einem Standard-Vorgehen zur Anbindung von Webservices auf dem OSB. Hinzu kam, dass es bereits generative Ansätze innerhalb mancher AEs gab, der aus PL/SQL Business-

Logik Webservice-Schnittstellen erzeugt hat. Leider folgte man hier dem Code-First Ansatz. Mit steigender Anzahl von Servicebasierter Kommunikation wurde der Wunsch und auch die Notwendigkeit eines Service-Repository Werkzeugs stärker.

Nach einer Evaluierung hat sich die SVI für das Service Repository CEISeR (Common Enterprise Integration Service Repository) von der Deutschen Telekom entschieden.

CEISeR ist prinzipiell ein Modellrepository zur Verwaltung beliebiger Metadaten. Im Kontext von SOA gab es ein entsprechendes Meta-Modell, welches die notwendigen Entitäten und deren Beziehung untereinander beschreibt und welches in CEISeR genutzt wird (beispielsweise: Ein Service ist ein WSDLPortType und hat eine Beziehung zu Komponenten einer Anwendung. Eine Anwendung wird durch ein Binding an einen Zugangspunkt einer Umgebung gebunden).

CEISeR bietet nicht nur die Möglichkeiten zur Persistierung der Informationen, sondern CEISeR bietet eine Generatoren API basierend auf OpenArchitectureWare (oAW), mit welcher auf das Objektgeflecht zugegriffen und darüber navigiert werden kann, um die Daten in einer Transformation in einer anderen Repräsentation wieder zu nutzen.

Durch die Modellierung einer Service-Architektur in CEISeR ist es dank des MDSs Ansatzes und dem oAW Generator-Framework möglich, Templates in XPAND zu entwickeln, um z.B. den ProxyFlow für den OSB generativ zu erstellen. Durch eine Möglichkeit der automatischen Einspielung der generierten OSB Artefakte wurde die Bereitstellung der Laufzeitumgebung vollumfänglich automatisiert.

Wir sind sogar einen Schritt weiter gegangen und haben nicht nur die SOA Laufzeitumgebung automatisiert mit der Service-Architektur bestückt, sondern das MDS für die Oracle BPEL Suite wurde auch aus dem zentralen Repository automatisch und generativ bereitgestellt.

Ein weiterer Anwendungsfall in diesem Kontext war die komplette Generierung der PL/SQL Service-Fassade basierend auf der modellierten Service-Architektur in CEISeR.

Dieser Ansatz eignet sich für Reports unterschiedlichster Art ebenso, wie für alle textbasierten Artefakte, die in einem Zusammenhang zur Service-Architektur mit den entsprechenden Interfaces oder Laufzeitumgebungen stehen.

Durch diesen flexiblen und äußerst agilen Ansatz war es uns möglich, nahezu auf Zuruf neue Reports oder ähnliches zu erstellen und die unterschiedlichsten Stakeholder inhaltlich zu überzeugen.

Prozesse:

Wir leben ein zyklisches Releaseverfahren mit drei Releases im Jahr. Um in diesem Zyklus möglichst frühzeitig Klarheit über die anstehenden SOA-Themen zu bekommen, findet zu Beginn der

Entwicklungsphase ein SOA-Day statt, wo in Plenum und kleinen Gruppen innerhalb des SOA Teams übergreifende Abstimmungen und zumindest Entwürfe der Contracts erstellt werden.

Der SOA-Day ist somit quasi Kickoff für die Umsetzungen des SOA-Teams im Entwicklungszyklus. Im Vorfeld des SOA-Days führen die SOA-Entwickler eine Sichtung der SOA-relevanten Anforderungen des Releases durch, um neue bzw. zu ändernde Services und Prozesse zu identifizieren. In diesem Kontext werden auch evtl. notwendige Erweiterungen des CDM identifiziert und zur Abstimmung an die Produktmanager, die dazu inhaltlich die Führung haben, gegeben.

Weitere Aspekte und Element von Transformationen können in „Business Enterprise Architecture – Praxishandbuch zur digitalen Transformation in Unternehmen“ von Carsten Sensler et al. nachgelesen werden (<http://bea-buch.de>).

Bitte fügen Sie Ihre Kontaktadresse hinzu.

Kontaktadresse:

Carsten Sensler
ArtOfArc Sensler & Karalus GbR
Höhfuhr 3
D-44309 Dortmund

Telefon: +49 (0) 171 8324185
E-Mail carsten@artofarc.com
Internet: www.artofarc.com

Michael Heinke
SV-Informatik
Gottlieb-Daimler-Straße 2
68165 Mannheim

Telefon: +49 (0) 621/454-43928
E-Mail michael.heinke@sv-informatik.de
Internet: www.sv-informatik.de