

# Erfahrung bei der Einführung einer SOA für den Senat für Bildung, Jugend und Sport des Landes Berlin

E. Ulrich Kriegel, Holger Kurrek

Abteilung Sichere Business IT Infrastrukturen  
Fraunhofer Institut für Software- und Systemtechnik  
Mollstraße 1  
D-10178 Berlin  
ulrich.kriegel@isst.fraunhofer.de  
holger.kurrek@isst.fraunhofer.de

**Abstract:** Im Rahmen des Projektes „Integrierte Software Berliner Jugendhilfe“ wurde eine diensteorientierte und komponentenbasierte Architektur für die Senatsverwaltung für Bildung, Jugend und Sport von Berlin realisiert. Dabei zeigte sich, dass die Einführung und Realisierung moderner Architekturkonzepte keine hinreichende Bedingung für den Erfolg eines Projektes sind. Gerade in Projekten, in denen mehrere unabhängige Hersteller Realisierungsarbeiten durchführen, ist die Sicherung der konzeptionellen Integrität sowohl im Entwurf als auch später im Betrieb relevant.

## 1 Einleitung

Seit 2002 führt die Senatsverwaltung für Bildung, Jugend und Sport von Berlin das mehrstufige Projekt „Integrierte Berliner Jugendverwaltung (ISBJ)“ durch, in dem bestehende IT-Fachverfahren (Kinder in Tagesbetreuung, Wirtschaftliche Jugendhilfe, Zentrale Vormundschafts- und Unterhaltsvorschusskasse) modernisiert werden. Hauptziele des Projektes sind ein fachübergreifendes Controlling, die Verbesserung der Arbeitsprozesse der Sachbearbeiter, die Auslagerung von Arbeitsprozessen als Self-Services an externe Partner, z. B. an die Betreiber von Kinderbetreuungseinrichtungen und das Anbieten innovativer Dienstleistungen im Rahmen der E-Government-Initiative für die Bürger von Berlin. Die einzelnen Teilprojekte werden entweder ausgeschrieben oder im Rahmen bestehender Dienstleistungsverträge entwickelt, so dass unterschiedliche Hersteller an den Lösungen arbeiten.

Das Fraunhofer Institut für Software- und Systemtechnik begleitet das Projekt seit Ende 2002 als Qualitätssicherer [KNS04]. Qualitätssicherung beinhaltet hierbei sowohl die analytische Qualitätssicherung der gelieferten Artefakte als auch die Sicherung der konzeptionellen Integrität der entwickelten Softwaresysteme. Zusätzlich werden auch die Anforderungen der IT-Sicherheit und der wirtschaftlichen Betriebsführung geprüft.

Während der Laufzeit des Projekts zeigte sich, dass die Spezifikation und anschließende Realisierung einer diensteezentrierten Architektur zwar eine notwendige, aber keine hinreichende Bedingung dafür ist, dass die realisierten Fachverfahren erfolgreich in den produktiven Betrieb überführt werden können. Gerade in Projekten, in denen mehrere Anbieter Komponenten und Dienste realisieren, die dann in ein fertiges Gesamtsystem integriert werden müssen, ist die strikte Einhaltung von Prozessen in Entwicklung und Betrieb unumgänglich. Aber selbst dann existieren mit heutigen Technologien noch Probleme, für die es derzeit keine befriedigende Lösung gibt.

## **2 Das Projekt ISBJ**

Von der Ausgangslage her ist das Projekt „ISBJ“ ein typisches Projekt für die öffentliche Verwaltung: Die erstellten Lösungen müssen konform zu den gesetzlichen Vorgaben sein. Diese sind über einen gewissen Zeitraum stabil, nach einer Änderung im Rahmen des Gesetzgebungsprozesses muss die Lösung jedoch sehr schnell an die neuen gesetzlichen Regelungen angepasst werden. Wartbarkeit und Modifikationsfähigkeit sind deshalb primäre Anforderungen an die Softwaresysteme der öffentlichen Verwaltung.

Hinsichtlich der IT-Sicherheit sind die Vorgaben des BSI für E-Government-Anwendungen [EGHB] zu berücksichtigen, die auch den IT-Grundschutz beinhalten. Hinzu kommen noch Vorgaben des Landes Berlin sowie des Betreibers bzw. zentralen Dienstleisters der IT-Infrastruktur im Land Berlin. Der Betrieb der im Rahmen des Projekts entwickelten Anwendungen ist nach den Best Practices aus der IT-Infrastructure Library [ITIL] zu organisieren und in die bestehenden Betriebsprozesse der vorhandenen IT-Infrastruktur einzugliedern.

Softwaresysteme für die öffentliche Verwaltung werden z. Zt. meist als Individualsysteme in einem Nischenmarkt realisiert, der von wenigen kleinen Anbietern dominiert wird. Die am Markt angebotenen Softwaresysteme entsprechen meist technologisch nicht dem aktuellen Stand der Softwaretechnik, auf Grund der benötigten Kapazität sind nicht alle angebotenen Lösungen in Großstädten wie Berlin einsetzbar. Aus den damit verbundenen Problemen resultiert die zusätzliche Anforderung, möglichst Software auf einer skalierbaren „Standardplattform“ zu realisieren, so dass die Abhängigkeit von bestimmten Herstellern reduziert und ggf. eine Migration zu einem anderen Hersteller, d. h. ein flexibles Sourcing, erleichtert wird.

Neben diesen speziellen Rahmenbedingungen gelten all die Rahmenbedingungen bzgl. Leistungsfähigkeit, Ausfallsicherheit und Wartbarkeit, wie sie auch für Softwaresysteme in Unternehmen gelten: Von den Mengengerüsten her sind die nichtfunktionalen Anforderungen der einzelnen Fachverfahren mit denen kleinerer Finanzdienstleister vergleichbar. So muss das Teilprojekt „KiTa (Kinder in Tagesbetreuung)“ Informationen zur Betreuung von jährlich etwa 150 000 Kindern verwalten und den verteilten Zugriff von etwa 300 Sachbearbeitern aus 12 Bezirksverwaltungen und aus der Senatsverwaltung über das Intranet und von etwa 1000 Trägern von Kindertagesstätten über das Internet ermöglichen.

Die Mitarbeiter der öffentlichen Verwaltung besitzen eine hohe fachliche Kompetenz. Erfahrungen auf dem Gebiet der Softwaretechnik sind jedoch nur begrenzt vorhanden, so dass teilweise die Auswirkungen bestimmter fachlicher Anforderungen nicht abgeschätzt werden können.

Für den Erfolg großer IT-Projekte in der öffentlichen Verwaltung ist deshalb eine herstellerunabhängige Unterstützung in Form einer externen Qualitätssicherung essentiell, wie sie z. B. von öffentlichen Forschungseinrichtungen erbracht werden kann. Neben den üblichen Aufgaben der analytischen Qualitätssicherung besteht eine wesentliche Aufgabe der Qualitätssicherung in der Sicherung der konzeptionellen Integrität der gesamten Entwicklung, in der Abschätzung der Auswirkungen fachlicher Anforderungen auf die Komplexität der Realisierung, in der Gewährleistung der IT-Sicherheit sowie in der Definition der Rahmenbedingungen für einen sicheren Betrieb.

## **2.1 Sicherung der konzeptionellen Integrität**

Projekte der Größenordnung von ISBJ werden in mehreren Phasen realisiert, so dass unterschiedliche Anbieter mit der Realisierung von Teilprojekten beauftragt werden können.

Um die konzeptionelle Integrität der Arbeiten der einzelnen Teilprojekte zu sichern, wurde seitens der Qualitätssicherung für das Projekt ein verantwortlicher Software- und Systemarchitekt benannt, der den Auftraggeber in allen architekturelevanten Fragen berät. Er ist für alle Fragen zuständig, die die Plattform, Software- und Systemarchitektur betreffen. Soweit nicht globale Vorgaben betroffen sind, liegt die Realisierung der einzelnen Komponenten in der Hoheit der sie erzeugenden Auftragnehmer. Regelmäßige Absprachen zwischen den Anbietern und dem Architekten haben sich jedoch als günstig erwiesen, da hierdurch auch der Aufwand in der analytischen Qualitätssicherung reduziert werden konnte.

Zu Beginn des Projektes wurde die unten näher erläuterte Referenzarchitektur spezifiziert und später erweitert, in die sich alle Realisierungen einzuordnen haben. Die Arbeiten gingen dann in die Spezifikation der Systemarchitektur über, die in enger Zusammenarbeit mit den für den Betrieb Verantwortlichen erfolgte.

Um eine bessere Wartbarkeit und Wiederverwendung der realisierten Komponenten zu ermöglichen, wurden verbindliche Programmierkonventionen vorgegeben, an die sich alle Realisierungen zu halten haben.

## **2.2 Abschätzung der Auswirkungen fachlicher Anforderungen**

Ein Großteil der fachlichen Anforderungen können auf unterschiedliche Art und Weise realisiert werden. Eine wesentliche Aufgabe der Qualitätssicherung besteht deshalb darin, bei der Formulierung fachlicher Anforderungen durch den Auftraggeber auf die Konsequenzen für die Realisierung sowie für die Geschäftsprozesse hinzuweisen.

Im Teilprojekt „KiTa“ wurde deshalb ein Jour Fixe etabliert, an dem neben den fachlich Verantwortlichen und der realisierenden Firma auch der Architekt teilnimmt. In diesem Kreis werden alle fachlichen Anforderungen diskutiert und einer Lösung zugeführt.

### **2.3 Gewährleistung der IT-Sicherheit**

Im Projekt „ISBJ“ wird auch mit personenbezogenen Daten gearbeitet, deren Vertraulichkeit unbedingt gesichert werden muss. Weiterhin sind die in den Zahlungen eingesetzten Mittel von mehr als einer Milliarde Euro im Jahr sicher zu verwahren. Daher wurde die Bedeutung der IT-Sicherheit von Anfang an auch durch die Projektverantwortlichen hoch eingeschätzt.

Die Gewährleistung der IT-Sicherheit wurde auf Empfehlung des Fraunhofer ISST durch ein ganzheitliches IT-Sicherheitskonzept angegangen. Entsprechend den Empfehlungen des BSI [EGHB] wurde nach dem BSI-Standard [BSIS2] vorgegangen. Wichtig für die Wirtschaftlichkeit der IT-Sicherheitsmaßnahmen war die frühzeitige Begleitung des Projekts durch das IT-Sicherheitskonzept, so dass auch schon in der Konzeptphase der jeweiligen Komponenten Anforderungen der IT-Sicherheit berücksichtigt werden konnten und damit später ein teureres Nachbessern vermieden wurde.

Zuerst wurde aus den geplanten Geschäftsprozessen eine Schutzbedarfsfeststellung abgeleitet, welche die schützenswerten Objekte erfasst und qualitativ den Schutzbedarf bestimmt, aus dem sich der Umfang der notwendigen IT-Sicherheitsmaßnahmen ableitet. Dabei konnte in Iterationen durch eine Modifikation von Geschäftsprozessen der Schutzbedarf und damit der spätere Aufwand teilweise reduziert werden.

Die begleitende Weiterentwicklung des IT-Sicherheitskonzepts unterstützt auch die nachfolgenden Design-Entscheidungen, indem Sicherheitsrisiken minimiert und damit im späteren Betrieb aufwändige Schutzmaßnahmen begrenzt bleiben bzw. Sicherheitsvorfälle vermieden werden. Dabei kann der Einsatz von IT-Sicherheitsmaßnahmen wie SSL-Verschlüsselungshardware auch zu Einsparungen in der Implementierung führen, indem dort diese Funktionen nicht eingebaut werden müssen.

Zur Gewährleistung der IT-Sicherheit werden im IT-Sicherheitskonzept Anforderungen definiert, die in den realisierten Komponenten umgesetzt und deren Wirksamkeit auch geprüft werden muss. Ebenso gelten diese Anforderungen für die Entwicklung und die Betriebsführung der Anwendungen. Daher werden entsprechende Fragestellungen in einer Arbeitsgruppe, dem IT-Sicherheitsteam, zusammen mit Anwendungsentwicklern, fachlichen Experten und den Betreibern regelmäßig diskutiert.

## 2.4 Definition der Rahmenbedingungen für einen sicheren Betrieb

Das Projekt „ISBJ“ ist das erste große Projekt, in dem komponentenbasierte Software auf der Basis einer J2EE-Plattform verfahrensübergreifend eingesetzt wird. Es existieren im Land Berlin weder Erfahrungen mit der Technologie noch mit der komponentenbasierten Software-Entwicklung. Unter Leitung des Fraunhofer ISST wurde deshalb eine Arbeitsgruppe etabliert, die sich mit Fragen des Betriebs der ISBJ-Plattform befasst. Ziel ist es, den sicheren Betrieb der Plattform und der Anwendungen durch Mitarbeiter der Senatsverwaltung zu ermöglichen. Zu Beginn des Projektes erfolgte der provisorische Betrieb durch einen externen Anbieter, Mitte 2006 konnte der Betrieb durch Mitarbeiter des Rechenzentrums von SenBJS übernommen werden. Spätestens Ende 2007 ist mit dem Umzug in ein Sicherheitsrechenzentrum der endgültige Ausbau der Systemarchitektur geplant.

Die Software wird durch verschiedene Auftragnehmer, die nach unterschiedlichen Zeitplänen arbeiten, in Form von Komponenten realisiert. Deshalb muss dafür gesorgt werden, dass die einzelnen Komponenten getestet und in Betrieb genommen werden können, ohne dass die sie nutzende Fachverfahren in Mitleidenschaft gezogen werden. Die Systemarchitektur wurde deshalb so ausgelegt, dass eine Hierarchie von Plattformen für unterschiedliche Tests zur Verfügung steht. Neben dem Produktionssystem steht ein baugleiches Abnahmetestsystem zur Verfügung. Komponenten wie die Personenstammverwaltung greifen auf andere Systeme des Landes Berlin zu und können deshalb nur im Intranet des Landes Berlin und nicht bei einem Auftragnehmer getestet werden. Deshalb wurde ein zusätzliches Testsystem etabliert, auf dem die Hersteller von Komponenten ihre Software im Zusammenspiel mit anderen Komponenten testen können. Dieses System enthält mehrere Instanzen der ISBJ-Plattform, so dass es neben den Tests auch für Schulungen genutzt werden kann.

Die Abstimmung der Entwicklung und die Bereitstellung einer neuen Version der jeweiligen Komponenten mit ihren Abhängigkeiten auf den verschiedenen Systemen wurde durch die Definition von ITIL-konformen Prozessen für das Change- und Releasemanagement [ITIL] und deren technische Unterstützung durch ein Trouble Ticket System gesichert. Nur so können die unterschiedlichen Anforderungen von Entwicklern, Fachlichkeit und Betreibern erfolgreich koordiniert werden. Dabei ist die Integration der Qualitätssicherung ein Erfolgsfaktor.

Eine der größten Herausforderungen bei der Vorbereitung des Betriebes war die Etablierung eines einheitlichen Deployment-Prozesses, der auch den Anforderungen eines Produktionsbetriebes genügt. Ursprünglich verwendeten die beteiligten Unternehmen eigene Deployment-Szenarien, die z. T. dedizierte Rechner benötigten. Sensible Informationen wie Passworte oder Ressource-URLs wurden an unterschiedlichen Stellen verwaltet, so dass der Aufwand, diese zu ändern, erheblich war.

Erst durch die Vorgabe eines einheitlichen, zur verwendeten J2EE-Plattform (s. u.) konformen Deployment-Konzeptes für alle Produzenten von Komponenten und Anwendungen konnte dieses Problem gelöst werden.

## 2.5 Wissenstransfer und Schulungen

Um eine gemeinsame Kommunikationsbasis zu schaffen und um durch den Aufbau eigenen Wissens die Abhängigkeit vom Qualitätssicherer zu reduzieren, wurden parallel zu o. g. Aktivitäten Workshops mit dem Auftraggeber durchgeführt. Im Mittelpunkt stand die Abstimmung der fachlichen Zielstellung unter Berücksichtigung der technischen Möglichkeiten und der gesetzlichen Rahmenbedingungen. Als Techniken wurden Szenarien und UML-Aktivitätsdiagramme etabliert, auf deren Basis die Fachanwender ihre funktionalen Anforderungen formulieren und auch aus fachlicher Sicht validieren können. Daneben erfolgte eine Einführung in die Problematik der Software-Architekturen und in komponentenbasierte Software-Entwicklung.

Falls Auftragnehmer und Qualitätssicherer unterschiedliche Lösungskonzepte für Teilprobleme entwickelten, wurden deren Vor- und Nachteile in Form von Workshops vor dem Auftraggeber diskutiert, so dass dieser darauf aufbauend eine endgültige Entscheidung treffen konnte.

Wesentliche Grundlage für einen erfolgreichen Wissenstransfer ist eine vollständige Dokumentation, die auch durch den Auftraggeber weitergepflegt werden kann. So wurde auch in der Qualitätssicherung auf die Vollständigkeit und Korrektheit der Dokumentation zu den Lieferungen der realisierten Komponenten ein Schwerpunkt gesetzt. Außerdem wurden auch die Anforderungen und Checklisten der Qualitätssicherung dem Auftraggeber dokumentiert zur Verfügung gestellt.

Für den Betrieb wurden ITIL-konforme Prozesse wie *Incident-*, *Problem-*, *Change-* und *Release-Management* gemeinsam mit dem Auftraggeber entwickelt, abgestimmt und dokumentiert.

Als Voraussetzung für die Übernahme des Betriebes durch Mitarbeiter von SenBJS wurden gemeinsam mit dem Auftraggeber mehrere mehrtägige Schulungen zu Applikationsserver, Datenbank und Data Warehouse geplant, die dann vom Hersteller der Applikationsplattform durchgeführt wurden.

## 3 Software- und Systemarchitektur für ISBJ

Um die Anforderungen an die Unabhängigkeit von den Software-Lieferanten und an die langfristige Weiterentwicklungsfähigkeit der Anwendungen zu gewährleisten, wurde durch das Fraunhofer ISST eine komponentenbasierte und dienstzentrierte Referenzarchitektur vorgegeben: Die Zerlegung in Komponenten und Dienste ermöglicht eine gezielte Weiterentwicklung der in ihnen gekapselten Funktionalität. Komponenten können unabhängig voneinander entwickelt und dann in die Anwendung integriert werden, bei konstanten oder additiv erweiterten Schnittstellen können Komponenten für den Benutzer transparent ausgetauscht werden. Obwohl bisher weder Notwendigkeit noch die Gelegenheit dafür bestand, dass Komponenten auch durch andere als die sie erstellenden Firmen gepflegt und weiterentwickelt werden können, sollte das durch die strikten Programmiervorgaben, geforderten Realisierungsbeschreibungen und durch das Einhalten der J2EE-Standards möglich sein.

Gemeinsam mit dem Auftraggeber wurden auf der Ebene der Fachlogik die von den Anwendungen benötigten Dienste identifiziert. Dabei wurde zwischen fachspezifischen Anwendungen und domänenspezifischen bzw. Infrastruktur-Diensten unterschieden. Fachanwendungen stellen keine wieder verwendbaren Dienste bereit und werden deshalb als Ganzes von einem Auftragnehmer realisiert. Domänenspezifische Dienste besitzen das Potenzial zur Wiederverwendung in der Domäne der Senatsverwaltung für Bildung, Jugend und Sport bzw. im Land Berlin und müssen wieder verwendet werden.

Die folgenden domänenspezifischen Dienste wurden im Kontext des Projekts „ISBJ“ realisiert bzw. befinden sich gerade in der Realisierungsphase:

- **Buchungskomponente**  
Das Land Berlin verwendet als Kassensystem die Software „ProFiskal“, die z. Zt. nur Dateischnittstellen zur Verfügung stellt. Die Buchungskomponente stellt im Rahmen der ISBJ-Plattform eine Schnittstelle zu „ProFiskal“ bereit, die unabhängig von der konkreten Realisierung der ProFiskal-Import-Schnittstelle ist. Damit wurde die Entwicklung der ISBJ-Fachverfahren von der Weiterentwicklung des Systems „ProFiskal“ entkoppelt.
- **Einrichtung- und Dienste**  
Die Komponente „Einrichtungs- und Dienste“ verwaltet zentrale Leistungsdaten und stellt damit in einheitlicher Weise Konfigurationsinformationen für alle Fachverfahren bereit.
- **Personenstammverwaltung**  
Die Komponente „Personenstammverwaltung“ verwaltet fachverfahrensübergreifend Personeninformationen und ordnet diesen einen eindeutigen Identifikator zu. Damit ist gesichert, dass pro Person genau ein Datensatz existiert und dass jedes Fachverfahren mit identischen Personenstammdaten arbeitet. Im Zuge des Anlegens eines neuen Personendatensatzes wird dieser über einen Abgleich mit dem Informationssystem des Landesamtes für Bürger- und Ordnungsangelegenheiten (LABO) verifiziert.
- **Regionales Bezugssystem**  
Das regionale Bezugssystem liefert Katasterinformationen und sozialraumbezogene Informationen, die für fachliche Auswertungen relevant sind. Die Informationen werden vom Statistischen Landesamt bereitgestellt und offline verteilt. Die zentrale Bereitstellung dieser Daten reduziert den Wartungs- und Pflegeaufwand. Eine spätere Nutzung eines „RBS-Dienstes“ des Statistischen Landesamtes kann für den Nutzer transparent erfolgen.

Als Infrastrukturkomponenten wurden eine zentrale Benutzerverwaltung und ein Output-Management identifiziert, das den zentralen Druck und E-Mail-Versand von Dokumenten im Portable Document Format (PDF) ermöglicht.

Bei der Analyse der Anforderungen wurde weiterhin festgestellt, dass die geforderten Auswertungen in allen Fachverfahren ähnlich sind. Deshalb wurde die Auswertungsfunktionalität aus den Fachverfahren in ein zentrales Auswertungssystem ausgelagert, das in Form eines Data Warehouses realisiert wurde.

Da in der Senatsverwaltung für Bildung, Jugend und Sport noch nie eine Dienstplattform betrieben wurde, musste sie im Rahmen des Projektes aufgebaut werden. Aus ökonomischen Gründen, und um Erfahrungen im Betrieb zu sammeln, sollte zunächst eine minimale Dienstplattform realisiert werden, die später erweitert wird.

Nach der Festlegung der Oracle-J2EE-Plattform als Realisierungsplattform wurde das Komponentenmodell für die J2EE-Plattform entsprechend präzisiert.

Sowohl bei der Realisierung von Fachanwendungen als auch beim Zugriff auf die realisierten Service-Komponenten wurde bewusst auf nachrichtenorientierte Schnittstellen und asynchrone Kommunikation verzichtet, da diese jeweils lokal *deployed* werden, und eine Typprüfung zur Kompilationszeit während der Entwicklungsphase den Nachteil einer festen Kopplung ausgleicht. Der Zugriff auf Dienste externer Anbieter, z.B. auf Einwohnermeldedaten oder das Output-Management erfolgt dagegen schon jetzt asynchron über eine nachrichtenorientierte Schnittstelle.

Auf die Einrichtung eines zentralen Service-Repositories und auf die Realisierung weiterer SOA-Features wurde verzichtet, da sie momentan keinen Mehrwert für den Auftraggeber darstellen.

Abbildung 1 gibt einen Überblick über die Systemarchitektur der ISBJ-Plattform in der endgültigen Ausbaustufe, in der der besseren Übersichtlichkeit halber die Test- und Schulungsserver nicht dargestellt sind. Auf der rechten Seite ist das Intranet-System dargestellt, das aus einem Applikationsserver besteht, auf dem die Präsentations- und die Geschäftslogik-Schicht lokalisiert sind.

Auswertungen können einen extremen Ressourcenbedarf haben, der sich negativ auf den Durchsatz der Fachanwendungen auswirken kann. Deshalb wird ein separater Applikationsserver für die Auswertungslogik betrieben. Die Datenhaltung erfolgt in einem Oracle-RAC<sup>15</sup>-System, das über drei Rechner realisiert wird, die auf ein gemeinsames Speichersubsystem zugreifen.

---

<sup>15</sup>

RAC: Real Application Cluster

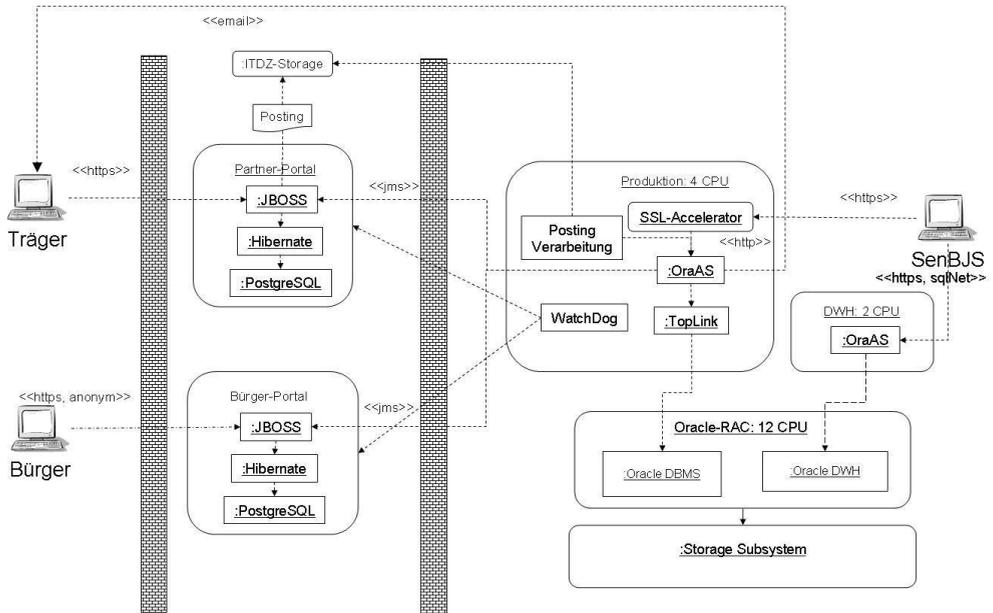


Abbildung 1: Überblick über die Systemarchitektur der ISBJ-Plattform

Für die Interaktion mit externen Partnern, z. B. Trägern von Kindertagesstätten wurde als IT-Sicherheitsarchitektur ein asynchroner Mechanismus entwickelt, der die Vorgaben des BSI für E-Government-Verfahren berücksichtigt, die einen einseitigen aus dem Intranet initiierten Verbindungsaufbau vorschreiben. Dieser Mechanismus kann später auch für die Übermittlung von Anträgen der Bürger verwendet werden. Die Träger-Arbeitsplätze werden zusätzlich durch SSL-Client-Zertifikate als geschlossene Benutzergruppe authentisiert.

Träger greifen aus dem Internet auf dedizierte Rechner in der DMZ des Rechenzentrums zu. Zur Erteilung von Auskünften besitzen diese Rechner eine separate Datenhaltung, die aus dem Intranet aktualisiert und deren Konsistenz mittels der Komponente „WatchDog“ überwacht wird.

Anträge werden in Form so genannter Postings asynchron auf einen sicheren Speicher im Intranet transferiert. Vom Produktionsrechner aus werden die Postings zyklisch abgefragt und auf ihre Konsistenz und Glaubwürdigkeit geprüft. Akzeptierte Postings werden in den Bearbeitungsprozess eingespeist. Kommt es im Zuge des Bearbeitungsprozesses zu einer Änderung des Datenbestandes, so wird diese asynchron propagiert, so dass auch der Datenbestand der Internet-Rechner aktualisiert werden kann. Um die Administration der Firewalls zu vereinfachen, wird das SMTP-Protokoll zur Übermittlung der Postings genutzt, die Aktualisierung der Rechner in der DMZ erfolgt über JMS-Nachrichten.

Die Rückmeldung durchgeführter Aktionen an die Träger erfolgt über E-Mails.

## **4 Erfahrungen aus dem Projekt**

Nach einer Projektlaufzeit von 3 Jahren und der erfolgreichen Überführung von Anwendungen in den Probeecht- bzw. Echtbetrieb können jetzt erste Aussagen über den Erfolg des Projektes getroffen werden.

Gerade wenn Ziele wie die Unabhängigkeit von Software-Herstellern erreicht werden sollen, ist eine rein analytische Qualitätssicherung nicht ausreichend. Die Qualitätssicherung muss konstruktiv sowohl bei der Gestaltung der Software- und Systemarchitektur als auch bei der Planung des Betriebes eingreifen, um die Abhängigkeit von den Software-Produzenten zu minimieren. Hier sind herstellerunabhängige Organisationen wie Forschungseinrichtungen im Vorteil, da sie auf Grund ihrer Forschungstätigkeit auch konsistente Alternativen zu Standardlösungen aufzeigen können. Außerdem sind sie nicht genötigt, eigene Produkte zu vermarkten.

Die intensive Diskussion zwischen Auftraggeber, konstruktiver Qualitätssicherung und Lösungsanbietern half, den Realisierungsaufwand in Grenzen zu halten. Eine gemeinsame Abschätzung der Häufigkeit bestimmter Geschäftsprozesse führte zu Einsparungen, da diese nicht softwaretechnisch unterstützt sondern in Form von Organisationsanweisungen außerhalb des Systems umgesetzt wurden.

Zu Beginn der Arbeiten existierten bei allen Anbietern Defizite im Umgang mit den Kontrakten, die die J2EE-Plattform definiert. Diese konnten jedoch durch eine intensive analytische Prüfung von Quell- und Bytecode identifiziert und behoben werden.

Wesentlich für den Erfolg des Projektes ist auch die Frage, inwieweit das gewählte Architektur-Paradigma ausreicht, um einen kontinuierlichen Betrieb von Anwendungen und Ziele wie Wartbarkeit, Erweiterbarkeit und Unabhängigkeit vom Auftragnehmer über einen längeren Zeitraum zu gewährleisten und ob beim Betrieb der Anwendungen zusätzlicher organisatorischer Aufwand entsteht.

## **4.1 Einfluss der gewählten Architektur auf die Lösung**

Durch die Verwendung von Konzepten wie „Komponente“ oder „Dienst“ in Verbindung mit strikten Programmiermodellen konnte man sich den gesetzten Zielen annähern. Die gute Wartbarkeit zeigt sich u. a. darin, dass geänderte gesetzliche Rahmenbedingungen problemlos in eine Fachanwendung überführt werden konnten.

Betrachtet man die Architektur jedoch nicht aus dem Blickwinkel der Software-Architektur, sondern in Hinblick auf einen sicheren Betrieb, so trifft man auch auf Fragen, für die das gewählte Architektur-Konzept keine befriedigende Antwort liefert.

### **4.1.1 Wechselbeziehung zwischen Software- und Systemarchitektur**

Prinzipien der modellbasierten Softwareentwicklung und das Architekturmuster SOA helfen, Software zu strukturieren und Dienste-Komponenten effizient zu entwickeln. Es sind jedoch außerhalb von Forschungseinrichtungen keine Techniken bekannt, die helfen, die wechselseitige Beeinflussung von Software- und Systemarchitektur zu beherrschen, die ggf. zu starken Einschränkungen bei den Möglichkeiten des Deployments von Diensten führen kann.

Wird bei der Realisierung z.B. ein Persistenzframework verwendet, das davon ausgeht, die alleinige Datenhoheit zu besitzen, dann können Service-Komponenten nicht mehr einfach zur Lastverteilung auf mehrere Server verteilt werden. Es muss dann für eine Synchronisierung der lokalen Caches gesorgt werden, was die Komplexität der Anwendungen und damit den Aufwand bei der Betriebsführung erhöhen kann: Gegebenenfalls müssen zusätzliche Kommunikationskanäle für die Synchronisation der Caches geöffnet werden, was negative Auswirkungen auf das Sicherheitskonzept des Verfahrens bedeuten kann.

Die Entwicklung einer SOA muss deshalb immer mit der Entwicklung der Systemarchitektur koordiniert werden. Nach einer Geschäftsprozessanalyse folgt die Zieldefinition, die im Wesentlichen eine Zerlegung der Funktionalität in Dienste beinhaltet. Daran schließt sich eine Definition einer Rahmenarchitektur an, die auf einer Auswahl geeigneter Architekturstile und -muster aufbaut. Die Auswahl dieser Architekturstile und -muster berücksichtigt die nicht-funktionalen Anforderungen der Geschäftsprozesse und Dienste. Die Definition der Referenzarchitektur hat wiederum Rückwirkung auf die Definition der Dienste.

### **4.1.2 Garantierte Performance und Verfügbarkeit von Diensten**

Sollen Dienste zentral angeboten werden, so stellt der Nutzer eines Dienstes bestimmte Anforderungen an Performance und Verfügbarkeit. Durch eine lose Kopplung auf der Basis einer asynchronen nachrichtenbasierten Kommunikation zwischen Diensten können Verfügbarkeitschwankungen nivelliert werden, es gibt jedoch kein allgemeingültiges Verfahren zur Garantie einer bestimmten Performance. Eine horizontale Skalierung, d. h. das Clustering von Laufzeitumgebungen für Service-Komponenten, kann die Verfügbarkeit und auch die Performance eines Dienstes verbessern. Eine bestimmte Dienstqualität kann heute jedoch nur garantiert werden, wenn der Dienst über definierte Ressourcen verfügt. Die Möglichkeit, einen Dienst singular auf einem dedizierten Server zu betreiben, scheidet meist an ökonomischen Gründen, so dass nur noch Virtualisierungsmethoden als Lösung in Frage kommen. Virtualisierungsmethoden sind heute in einigen Serverbetriebssystemen integriert und erlauben eine exakte Ressourcenzuordnung zu Applikationen, sie sind aber heute noch nicht feingranular genug, um eine Ressourcenzuordnung auf Dienste-Ebene zu ermöglichen. Die Lösung, eine Service-Komponente in einer Virtualisierungsumgebung mit definierten Ressourcenzuordnungen wie Prozessor-Anzahl oder I/O-Kapazität zu betreiben, scheidet an einem unverhältnismäßigen Management-Overhead und bei kommerziellen Applikationsservern auch an Lizenzkosten. Deshalb werden heute Applikationsserver meist überdimensioniert. Eine Validierung der abgeschätzten Leistungsgrenzen im Produktionsbetrieb muss auf experimentellem Weg mittels Lasttests erfolgen.

## **4.2 Zusätzliche organisatorische Anforderungen**

Um verteilt entwickelte diensteorientierte Architekturen erfolgreich betreiben zu können, muss ein zusätzlicher organisatorischer Aufwand zur Koordination der Entwicklung betrieben werden.

### **4.2.1 Release- und Testmanagement**

Besteht eine Anwendungslandschaft weitgehend aus isolierten „monolithischen“ Verfahren, so ist auch deren Releasemanagement weitestgehend entkoppelt. Werden Dienste jedoch auf einer gemeinsamen Plattform betrieben oder benutzen Service-Komponenten gemeinsame Ressourcen, so sind die Aktivitäten zum Releasemanagement der einzelnen Entitäten nicht mehr entkoppelt und müssen in ein übergeordnetes plattformweites Releasemanagement integriert werden. Eine analoge Aussage gilt auch für das Testmanagement.

Gegenüber dem Testen isolierter Einzelanwendungen entsteht so ein erhöhter Organisations- und Koordinationsaufwand.

## 4.2.2 Koordination verteilter Entwicklung und Verfahrensverantwortung

Der Aufbau komplexer dienstorientierter Architekturen ist häufig mit der parallelen Entwicklung von Diensten verbunden. Das betrifft gleichermaßen neue Dienste wie auch die Kapselung und Adaptierung von Altanwendungen. Gerade im öffentlichen Bereich, wo diese Arbeiten in der Regel ausgeschrieben werden müssen, werden diese Entwicklungen häufig von mehreren Dienstleistern durchgeführt. Das Ziel einer funktionsfähigen, den Ansprüchen genügenden Gesamtarchitektur ist nur durch eine sorgfältige Koordination der Entwicklungsarbeiten zu gewährleisten. Grundvoraussetzung für diese Koordination ist eine IT-Gesamt-Referenzarchitektur, in der die Regeln und Standards für die konkrete IT-Gesamtarchitektur vorgegeben sind.

In der öffentlichen Verwaltung tragen sogenannte Verfahrensverantwortliche die Gesamtverantwortung für Fachverfahren. Im Rahmen der Etablierung von SOAs sind entsprechende Verantwortlichkeiten für Dienste festzulegen, die z. B. aus den Dateneigentümern abzuleiten sind. Das kann im Vergleich zu isolierten Systemen einen erhöhten Organisationsbedarf zur Folge haben, vor allem dann, wenn die Dienste verteilt betrieben werden.

## 4.3 Fazit

Zusammenfassend kann man feststellen, dass die Einführung einer SOA in einer Organisation Potentiale einer erhöhten Evolutionsfähigkeit und Reaktionsfähigkeit gegenüber sich ändernden Anforderungen birgt. Sie ist eine große Verbesserung, die notwendig aber nicht hinreichend für evolutionssichere Unternehmens-IT-Systeme ist. Die Bewältigung der vielfältigen Organisations- und Koordinationsaufgaben ist der wesentliche Erfolgsfaktor dafür, ob die mit SOA gesteckten Ziele, eine evolutionsfähige IT-Infrastruktur und die Reduzierung von Kosten, erreicht werden kann.

## Literaturverzeichnis

- [EGHB] Bundesamt für Sicherheit in der Informationstechnik (BSI): E-Government-Handbuch, Modul „Sicherer Internet-Auftritt im E-Government“, Stand 13.06.2005.
- [BSIS1] BSI-Standard 100-1: Managementsysteme für Informationssicherheit (ISMS), Version 1.0
- [BSIS2] BSI-Standard 100-2: IT-Grundschatz-Vorgehensweise, Version 1.0
- [BSIS3] BSI-Standard 100-3: Risikoanalyse auf der Basis von IT-Grundschatz, Version 2.0
- [GSK] BSI: IT-Grundschatz-Kataloge, Stand Dezember 2005
- [ITIL] IT Infrastructure Library - Service Support (ccta), Central Computer & Telecommunications Agency, The Stationery Office Books, 2000.
- [KNS04] E. U. Kriegel, L. Nentwig, E. Siegeris: "IT-Projekte in der öffentlichen Verwaltung – Analogien zum Outsourcing. Ein fallbeispiel aus der Berliner Verwaltung“, in J. Borchers, R. Kneuper (Eds): "Software Management 2004: Outsourcing und Integration", Lecture Notes in Informatics, 2004.