

Ein praxiserprobtes Rahmenwerk für die technische Anwendungsintegration

Bernhard Humm*, Marc Lohmann, Markus Voß, Johannes Willkomm

sd&m Research
sd&m AG, software design & management
Berliner Straße 76, 63065 Offenbach am Main
{Bernhard.Humm*, Marc.Lohmann, Markus.Voss, Johannes.Willkomm}@sdm.de
B. Humm zusätzlich Hochschule Darmstadt

Abstract: In diesem Artikel wird ein praxiserprobtes Rahmenwerk für die technische Anwendungsintegration vorgestellt. Es besteht aus Begriffsdefinitionen, Regeln für die Auswahl der adäquaten Integrationsebene, Referenzarchitekturen, Produktlandkarten und Hinweisen zum Vorgehen. Es hilft nachweislich, die Wartung und Weiterentwicklung von integrierten Anwendungslandschaften zu verbessern.

1. Einleitung

Globalisierung und sich ändernde Märkte fordern von Unternehmen die rasche Anpassung an neue Bedürfnisse. Die IT-Abteilungen müssen dies mit der Umgestaltung der Anwendungslandschaften zur Realisierung neuer Funktionalitäten unterstützen. IT-Anwendungslandschaften umfassen die Anwendungen, welche die Umsetzung neuer Marktanforderungen und Prozesse unterstützen oder sogar erst ermöglichen [Hes04].

Der Einsatz von Standardsoftware ist ein etabliertes Mittel, um sich dieser Herausforderung zu stellen. Zusätzlich werden in vielen Unternehmen Anwendungen zur Unterstützung von Kernprozessen individuell entwickelt. Gerade unternehmenskritische Systeme sind oftmals weit über zehn Jahre im Einsatz. Daher müssen in nahezu allen IT-Anwendungslandschaften großer Unternehmen Anwendungen unterschiedlicher Fachlichkeit, Technologie und Herkunft integriert werden.

Diesem Trend folgend wurden diverse Produkte entwickelt, die die Integration von Systemen erleichtern. Unsere Erfahrung zeigt aber, dass eine punktuelle ad-hoc Integration von Systemen unter hohem Zeit- und Kostendruck auf Dauer die ursprünglichen Ziele in das Gegenteil umkehren kann: die Kosten für die Wartung der IT-Anwendungslandschaft steigen enorm und die Integration wird extrem komplex.

Die sd&m AG hat in den letzten 5 Jahren dutzende Integrationsprojekte mit einem Gesamtvolumen von weit über 100 Mitarbeiterjahren durchgeführt. Des Weiteren berät sie seit vielen Jahren führende Unternehmen und Behörden in der Gestaltung ihrer Anwendungslandschaft. Im Forschungsvorhaben *Quasar Enterprise* kondensieren wir seit nunmehr zwei Jahren unsere Projekterfahrungen. In [HHV06] haben wir unsere wich-

tigsten Ergebnisse im Bereich serviceorientierter Architekturen (SOA) vorgestellt. In diesem Artikel stellen wir die wichtigsten Ergebnisse im Bereich der Anwendungsintegration dar: ein Rahmenwerk mit Begriffsdefinitionen, Referenzarchitekturen, Produktlandkarten und Hinweisen zum Vorgehen. Im Praxiseinsatz hat sich das Rahmenwerk als wertvoll erwiesen.

Im folgenden Kapitel 2 führen wir die wichtigsten Begriffe unseres Rahmenwerks ein. Kapitel 3 behandelt Referenzarchitekturen und Produktlandkarten. Kapitel 4 gibt Hinweise zum Vorgehen in Integrationsprojekten. Praxiserfahrungen werden in Kapitel 5 vorgestellt. Kapitel 6 rundet den Beitrag mit einem Fazit ab.

2. Integrationsarchitektur

2.1 Integrationspunkte

Für betriebliche Anwendungen hat sich die Drei-Schichten-Architektur [DK76] bewährt. Bei der Integration von Anwendungen können wir jedoch nicht davon ausgehen, dass deren Architekturen diesem Paradigma folgen. Besonders beim Einsatz kommerzieller Standardsoftware, sind die Interna der Anwendung nicht offen gelegt. Offen gelegt sind jedoch immer bestimmte Punkte an denen eine Integration ansetzen kann. Diese *Integrationspunkte erlauben* einer zweiten Anwendung, mit dieser technisch zu kommunizieren. Wir unterscheiden drei *Arten von Integrationspunkten* (vgl. auch Abbildung 1):

1. *Präsentation*: Eine Benutzerschnittstelle ist derart gestaltet, dass die zweite Anwendung diese technisch einbinden kann.
2. *Service*: Eine funktionale Schnittstelle (Application Programming Interface – API) ist offen gelegt, die von der zweiten Anwendung genutzt werden kann.
3. *Daten*: Ein Zugriff auf Daten kann z.B. über SQL oder einen Dateizugriff erfolgen.

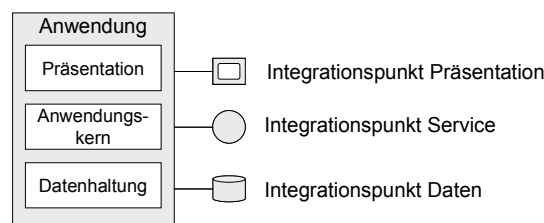


Abbildung 1: Arten von Integrationspunkten

Die Verwendung von Integrationspunkten kann aber auch deutlich komplexer sein, wie die aktuelle Technologie *Enterprise Information Integration (EII)* aus dem Bereich der Datenintegration zeigt. Hier werden Anwendungen unternehmensweit SQL-Sichten auf Kerndaten bereitgestellt, welche physisch in Datenbanksystemen unterschiedlicher Technologien, flachen Dateien oder Anwendungen zur Tabellenkalkulation abgelegt sind.

2.2 Transformation von Integrationspunkten

Oftmals erfolgt in Projekten eine Transformation zwischen den verschiedenen Arten von Integrationspunkten. So erlaubt die *Screen-Scraping-Technologie*, Daten aus der Präsentation einer Anwendung, z.B. einer HTML-Seite, zu extrahieren und anderen Anwendungen zur Verfügung zu stellen. Diese Anwendungen können somit über eine funktionale Schnittstelle Dialogkommandos absetzen. Hier wird also ein Präsentations-Integrationspunkt verwendet um einen neuen Service-Integrationspunkt zu erzeugen.

Prinzipiell sind alle Kombinationen von Transformationen zwischen den Arten von Integrationspunkten möglich. In Abbildung 2 sind in den Spalten die Arten von Integrationspunkten eingetragen, welche die zu integrierende Anwendung anbietet. In den Zeilen ist eingetragen, wie die integrierende Anwendung den Integrationspunkt verwendet. In der Tabelle sind Technologien aufgeführt, die die Transformation unterstützen. In der Praxis am häufigsten anzutreffen sind die Transformationen in der Diagonalen.

Ziel \ Quelle	Präsentation	Service	Daten
Präsentation	Portlet-Integration, Präsentations-transformation, Clipping	Service Portlets	SQL Portlets
Service	Screen Scraping	Service-Adaptoren	SQL Adaptoren
Daten			Daten-konsolidierung, -föderation und -propagation

Abbildung 2: Technologien zur Transformation von Integrationspunkten

2.3 Integrationsebenen

Bei der Integration verschiedener Anwendungen werden Integrationspunkte genutzt, um einen *Mehrwert* zu schaffen. Wir unterscheiden drei *Integrationsebenen*, benannt nach den Arten der Integrationspunkte, die nach einer möglichen Transformation zur Integration verwendet werden:

- *Präsentation*: Werden Integrationspunkte der Art Präsentation (Benutzerschnittstelle) direkt oder nach einer Transformation darauf integriert, um einen Mehrwert zu schaffen, so sprechen wir von *Präsentationsintegration*.
- *Service*: Werden Integrationspunkte der Art Service (Anwendungsfunktionen) direkt oder nach einer Transformation dorthin integriert, so sprechen wir von *Serviceintegration*.
- *Daten*: Werden Integrationspunkte der Art Daten (Datenbanken, Dateisysteme etc.) integriert, so sprechen wir von *Datenintegration*.

Abbildung 3 veranschaulicht beispielhaft die Präsentationsintegration.

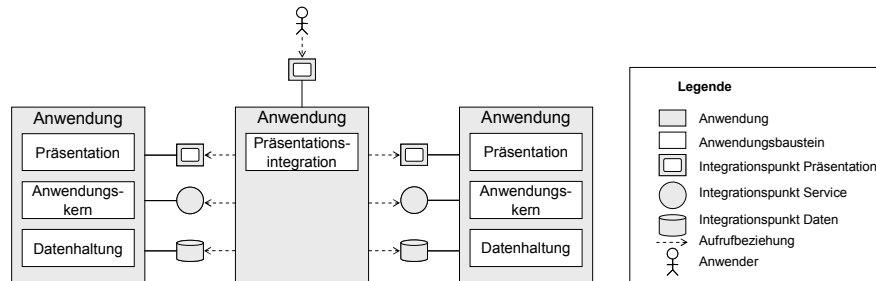


Abbildung 3: Präsentationsintegration

Die Integration auf einer Ebene ist mehr als die reine Transformation von Integrationspunkten. Häufig werden technische Dienste von Integrationsplattformen genutzt wie Workflow, Transaktionen oder Sicherheit. Die folgenden Beispiele verdeutlichen dies:

- *Präsentation*: Präsentationsintegration umfasst nicht nur das Zusammenfügen von Dialogen (statisch), sondern auch die Umsetzung der Dialogsteuerung (dynamisch).
- *Service*: Die Ablaufsteuerung bei der Serviceintegration wird auch Orchestrierung genannt. Sie kann grafisch in Kombination mit einer Codegenerierung erfolgen.
- *Daten*: Extraktion, Transformation und Laden von Data Warehouses (z.B. [HW05]) ist ein Prozess der Datenintegration. Es geht um die Extraktion aus unterschiedlichen Datenquellen, die Transformation und das Laden in einen Zieldatenbestand.

2.4 Logische Integrationsarchitektur

Unter der *logischen Integrationsarchitektur* verstehen wir die Festlegung, welche Integrationspunkte welcher Anwendungen auf welchen Ebenen integriert werden. Das Adjektiv *logisch* verwenden wir, da wir noch nicht über konkrete Plattformen sprechen.

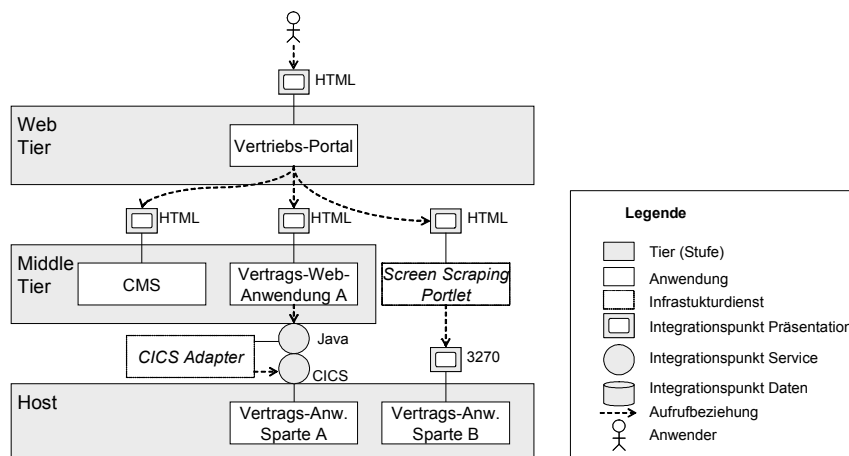


Abbildung 4: Integrationsarchitektur einer Versicherungsanwendung

Abbildung 4 zeigt ein Beispielprojekt aus der Versicherungsbranche. In diesem wurde ein Vertriebsportal für Versicherungsprodukte entwickelt. Die Vertragslogik für die verschiedenen Versicherungssparten war in Kernsystemen mit 3270-Dialogen implementiert. Für die Vertragsverwaltung einer Sparte mussten neue Dialoge entwickelt werden. Dazu wurde eine Vertrags-Web-Anwendung entwickelt, die unter Verwendung von IBM WebSphere MQ Series auf CICS-Transaktionen der Host-Anwendung zugreift – eine Serviceintegration. Die Web-Anwendung stellt HTML-Dialoge als Integrationspunkte bereit, es wird also eine Präsentationsintegration durchgeführt. Die Dialoge der hostbasierten Vertragsanwendungen anderer Sparten konnten direkt in das Vertriebsportal übernommen werden. Dazu wurden die 3270-Dialoge unter Verwendung eines Screen Scraping Portlets direkt in HTML-Dialoge transformiert. Informationen über Versicherungsprodukte wurden in einem Content Management System abgelegt. Unter Verwendung eines Portal Servers wurden die Anwendungen integriert (Präsentationsintegration). Für den Anwender stellt sich das Vertriebsportal als integrierte Anwendung dar.

	Einsatzkriterien	Erfolgsfaktoren
Präsentations-Integration	■ Existierende Dialoge unterschiedlicher Anwendungen ■ Fachlogik beschränkt auf Dialogsteuerung und Übergabe von Feldinhalten	■ Werkzeugeinsatz Portal Server, Screen Scraping Tools, Portlets
Service-Integration	■ Existierende Anwendungen mit umfangreicher, komplexer Fachlogik ■ Rasche Einführung neuer fachlicher Geschäftsprozesse	■ Fachlich motivierter Komponenten- und Service-Schnitt nach Regeln von Quasar Enterprise ■ Werkzeugeinsatz ESB
Daten-Integration	■ Datenredundanz: geplant (Performance) oder bereits existent (z.B. historisch) ■ Ausfallsicherheit (Stand-by Datenbanken) ■ Unterschiedliche SLA's für Anwendungen (physische Separierung)	■ Pflegehöhe festlegen / Synchronisationskonflikte vermeiden ■ Anforderungen an Datenaktualität (Real-Time vs. Batch) berücksichtigen ■ Datenqualität ■ Werkzeugeinsatz ETL, MDM, EII, EAI

Abbildung 5: Einsatzkriterien und Erfolgsfaktoren für die Wahl der Integrationsebene

2.5 Wahl der adäquaten Integrationsebene

Serviceorientierte Architekturen sind das derzeit am meisten beachtete Paradigma zur Gestaltung von Anwendungslandschaften. Sollten daher nicht alle Anwendungen auf der Serviceebene integriert werden? Ist im Versicherungsbeispiel die Integration der Host-Anwendung mit der Web-Anwendung auf der Präsentationsebene adäquat? Im konkreten Beispiel kann die Frage bejaht werden. Die Begründung liegt in der Fachlichkeit der zu integrierenden Anwendungen. Die Vertragsanwendungen für die verschiedenen Sparten haben fachlich nur minimale Abhängigkeiten: nur einzelne Feldinhalte wie Name und Adresse werden übergeben. Übergreifende Fachlogik wie Prüfungen mit komplexen Abhängigkeiten in beide Anwendungen finden nicht statt. Daher ist die Integration auf Präsentationsebene aus architektonischer Sicht vertretbar. Die Integration auf Serviceebene ist teurer ohne zu einer evolutiv stabileren Architektur zu führen.

Doch Vorsicht! Die gute Werkzeugunterstützung der Präsentationsintegration in Portal Server Plattformen und die damit verbundenen Kostenvorteile verleiten dazu, auch bei fachlich größeren Abhängigkeiten zwischen den Anwendungen auf der Präsentations-ebene zu integrieren. In diesem Fall wandert umfangreiche Fachlogik unkontrolliert in den Portal Server – Redundanzen und schwer kontrollierbare Abhängigkeiten entstehen. Es ist die Aufgabe des Architekten, die adäquate Integrationsebene unter Berücksichtigung zukünftiger Erweiterungen festzulegen.

Abbildung 5 benennt beispielhaft Einsatzkriterien und Erfolgsfaktoren. Präsentationsintegration ist empfehlenswert, wenn bereits umfangreiche Dialoge zu integrierender Anwendungen existieren. Es ist aber entscheidend, dass die einzelnen Dialoge nur minimale fachliche Logik verbindet: die Aufrufreihenfolge und Übergabe einfacher Dialogdaten. Ist die fachliche Verzahnung enger so ist die Präsentation die falsche Integrationsebene. Dies gilt auch, wenn engere fachliche Abhängigkeiten in Zukunft zu erwarten sind.

Anwendungen mit umfangreicher und komplexer Fachlogik werden auf der Ebene der Serviceintegration integriert. Wenn die rasche Einführung neuer Geschäftsprozesse und Produkte in einer Anwendungslandschaft gefordert ist, soll die Integration auf dieser Ebene erfolgen, da hier die Abbildung der Geschäftsprozesse zentral realisiert wird.

Erscheint Serviceintegration als Mittel der Wahl, so ist diese aber oft deutlich aufwändiger und teurer als Präsentations- oder Datenintegration. Daher sollten umfangreiche SOA-Vorhaben nur auf der Basis von strategischen Entscheidungen gestartet werden.

Die Integration auf Datenebene kann sinnvoll sein, wenn Datenredundanz existent oder geplant ist. Dies kann aus Gründen der Performance, der Ausfallsicherheit, aufgrund des Einsatzes von Standardsoftware oder einfach aus historischen Gründen der Fall sein.

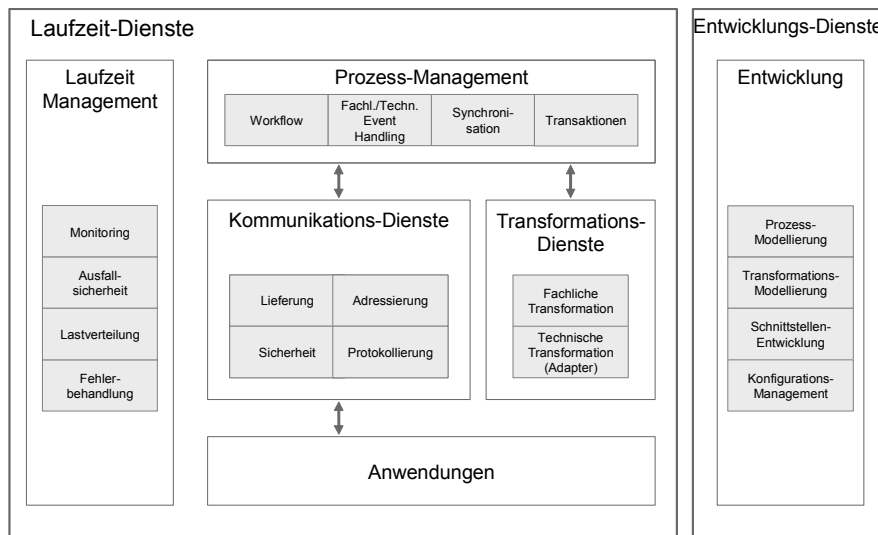


Abbildung 6: Referenzarchitektur für die Serviceintegration

3. Referenzarchitekturen und Produktlandkarten

Seit vielen Jahren werden Produkte, die Entwickler bei der Integration von Anwendungen unterstützen, angeboten. Der Markt für Integrationsplattformen ist jedoch noch nicht konsolidiert. Außerdem hat sich keine einheitliche Begriffsbildung für Integrationsplattformen und deren Dienste gebildet: verschiedene Hersteller verwenden unterschiedliche Begriffe für dieselben Dienste oder dieselben Begriffe mit unterschiedlichen Bedeutungen. Beide Effekte erschweren die Produktauswahl, die einen entscheidenden Einfluss auf die Qualität der Anwendungslandschaft hat. Hier helfen *Referenzarchitekturen*.

Für die Integration von Anwendungen werden verschiedene Dienste benötigt. Diese Dienste sind abhängig von der gewählten Integrationsebene und finden sich in Plattformen unterschiedlicher Produkthersteller in ähnlicher Form wieder. Zur Beschreibung und Klassifizierung der Dienste haben wir bei sd&m für die drei Integrationsebenen jeweils eine Referenzarchitektur entwickelt. In den Referenzarchitekturen werden die für eine Integrationsaufgabe notwendigen technischen Dienste festgehalten. Sie destillieren damit die Essenz von konkreten Produkten. Im Gegensatz zu den Produkten bleiben die Referenzarchitekturen über die Zeit relativ stabil. Referenzarchitekturen bilden damit einen Startpunkt für die Ermittlung individueller Projektanforderungen.

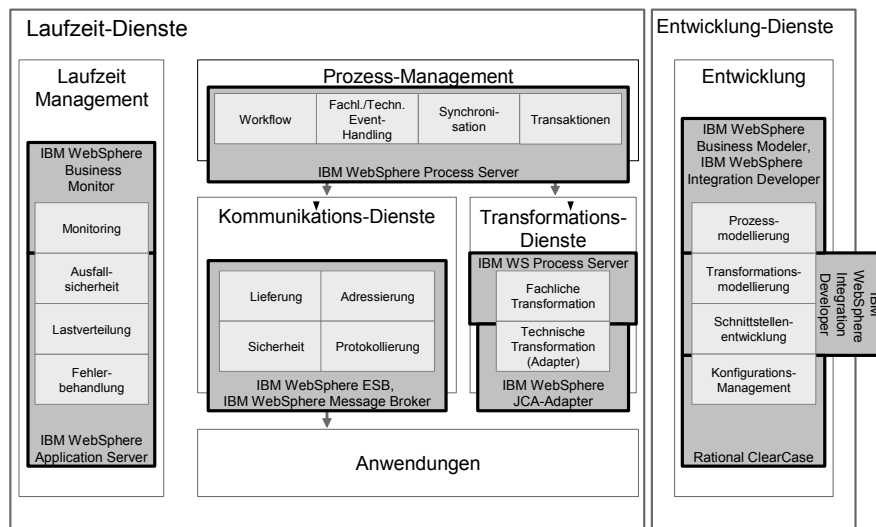


Abbildung 7: Produktlandkarte Serviceintegration für die IBM WebSphere Produktfamilie

Im Folgenden erläutern wir die Referenzarchitektur für die Integration auf der Serviceebene (Abbildung 6). In der Referenzarchitektur werden auf der obersten Ebene Dienstgruppen wie z.B. Prozess-Management und Kommunikations-Dienste unterschieden. Den Gruppen werden technische Dienste zugeordnet. Um das Beispiel aus Abschnitt 3.1 aufzugreifen: Ruft eine Anwendung den von einer anderen Anwendung bereitgestellten Web-Service auf, so werden im ersten Schritt Transformationen durchgeführt (Services *fachliche, technische Transformation*). Dann wird die zu rufende Anwendung *adressiert*

und die Parameterdaten transportiert (*Lieferung*). Die Kommunikation der beiden Anwendungen wird technisch gesteuert (*fachl. / techn. Event Handling*).

Referenzarchitekturen helfen auch bei der Wahl einer konkreten Integrationsplattform. In *Produktlandkarten* haben wir die wichtigsten Produkte auf der Basis der Referenzarchitektur kategorisiert. Abbildung 7 zeigt dies für die IBM WebSphere Produktfamilie.

4. Vorgehen

Auf Basis der eingeführten Elemente hat sich in Projekten der sd&m AG folgende systematische Vorgehensweise für die technische Anwendungsintegration bewährt:

1. *Fachliche und technische Analyse*: In diesem Schritt werden die fachlichen Anforderungen geklärt und identifiziert, welche Anwendungen dafür zu integrieren sind. Dann werden die potenziellen Integrationspunkte dieser Anwendungen identifiziert.
2. *Festlegung der logischen Integrationsarchitektur*: Nun wird festgelegt, welche Integrationspunkte welcher Anwendungen auf welchen Ebenen integriert werden und so die logische Integrationsarchitektur entwickelt. Hier helfen die Regeln für die Wahl der adäquaten Integrationsebene.
3. *Identifikation der benötigten Integrationsdienste*: Von der Integrationsarchitektur können die benötigten technischen Integrationsdienste abgeleitet werden. Hierbei unterstützen die Referenzarchitekturen.
4. *Auswahl der Integrationsplattform*: Auf der Basis der benötigten technischen Integrationsdienste können Integrationsplattformen oder einzelne Produkte daraus ausgewählt werden. Hier helfen Produktlandkarten.

5. Praxiserfahrung

Im Folgenden illustrieren wir die obige Vorgehensweise anhand eines Projektes aus der ITK-Branche. Ziel des Projekts war die Bereitstellung einer Vielzahl von Diensten auf Settop-Boxen (Geräte, welche Fernsehen und Internet miteinander verbinden). Die zu integrierenden Anwendungen stellen Dienste wie eine elektronische Programmzeitschrift oder das Auswählen und Abspielen von Videos on-demand bereit. Die zu entwickelnde Weboberfläche sollte durch eine klassische Fernbedienung für Fernseher steuerbar sein.

In der Analyse wurden als erstes die zu integrierenden Anwendungen und deren Rahmenbedingungen betrachtet. Es wurde bestimmt, welche Arten von Integrationspunkten die Anwendungen besitzen, welcher Aufwand und technischen Fähigkeiten nötig sind, um neue Integrationspunkte bereitzustellen und wie komplex die Integrationslogik ist.

Aufgrund der geringen Komplexität bei der Kopplung der einzelnen Anwendungen wurde die Präsentationsintegration als logische Integrationsarchitektur festgelegt. Dabei

stellen alle Anwendungen als Integrationspunkte HTML-Oberflächen bereit. Diese werden mit einer Integrationsplattform zu einer einheitlichen Oberfläche zusammengeführt.

Im nächsten Schritt wurde analysiert, welche Integrationsdienste der Referenzarchitektur für Präsentationsintegration in welcher Ausprägung benötigt werden. Hierbei stellte sich heraus, dass die größte Herausforderung in der fachlichen Transformation der Oberflächen steckt: Einfache HTML-Formulare mussten in einer HTML-Oberfläche zusammengeführt werden, die ihrerseits aber komplexe Anforderungen an die Steuerungsmechanismen zur Bedienung dieser Oberfläche stellte.

Auf Basis der Analyse wurde eine Integrationsplattform ausgewählt, welche ausgewiesene Stärken in der fachlichen Transformation von Datenströmen hat und die benötigten technischen Dienste zur Präsentationsintegration hinreichend abdeckt.

Ausgehend nur von den funktionalen Anforderungen wären im Bereich der Integrationsarchitektur, wie auch bei der Auswahl des Produkts andere Lösungen möglich gewesen. Im Nachhinein betrachtet wären diese allerdings aufwändiger und riskanter in der Implementierung. Auf Basis der systematisch entwickelten Integrationslösung wurde das Projekt im Budget- und Zeitrahmen zur vollsten Zufriedenheit des Kunden umgesetzt.

6. Fazit und Ausblick

Anwendungen unter Zeit- und Kostendruck ad-hoc zu integrieren kann auf die Dauer katastrophale Folgen für die Wartung einer IT-Anwendungslandschaft haben. In diesem Artikel haben wir ein praxiserprobtes Rahmenwerk für die technische Anwendungsintegration vorgestellt. Das Rahmenwerk kondensiert die Erfahrungen aus dutzenden Integrationsprojekten mit einem Gesamtaufwand von weit mehr als 100 Mitarbeiterjahren.

Unsere Arbeit an Quasar Enterprise geht weiter. Nachdem sich das Rahmenwerk als stabil erwiesen hat, liegt noch viel Detailarbeit vor uns: Praxiserfahrungen mit neu aufkommenden Technologien und Produkten gewinnen, kondensieren und aufbereiten. Unsere Erfahrung zeigt, dass diese Investition sich auszahlen.

7. Literatur

- [DK76] F. DeRemer, H. Kron: *Programming-in-the-Large versus Programming-in-the-Small*. IEEE Transactions on Software Engineering, Vol. 2, No. 2, pp 321-327, 1976.
- [Hes04] A. Hess: *Sie sind es uns wert!*, IT Management 08/2004.
- [HHV06] A. Hess, B. Humm, M. Voß: *Regeln für serviceorientierte Architekturen hoher Qualität*. Informatik Spektrum 6/2006, Springer Verlag, 2006.
- [HW05] B. Humm, F. Wietek: *Architektur von Data Warehouses und Business Intelligence Systemen*. Informatik Spektrum 3/05, S. 3-14, Springer Verlag, 2005.