

Architekturzentriertes Vorgehen für Integrationsprojekte

Dr. (Ph.D.) Bernhard Humm, Boris Zech

sd&m Research
Herrnstraße 57
63065 Offenbach am Main
Bernhard.Humm@sdm.de
Boris.Zech@sdm.de

Abstract: In diesem Paper kondensieren wir das in zahlreichen Integrationsprojekten gesammelte Vorgehenswissen. Dabei spannen wir den Bogen von kleinen Integrationsprojekten bis zu großen Programmen und zeigen, dass ein auf die Architektur ausgerichtetes Vorgehen entscheidend für den Projekterfolg ist.

1 Einleitung

In den letzten Jahren verschwimmen die Grenzen zwischen Individualsoftware-Entwicklung und Standardsoftware-Lösungen. Auf der einen Seite werden Standardsoftware-Lösungen zu Integrationsplattformen ausgebaut, wie z.B. SAP NetWeaver; auf der anderen Seite werden individuelle Lösungen zunehmend aus Komponenten integriert.

Integration ist ein weites Feld (siehe z.B. [Kel02]). In diesem Paper fokussieren wir auf Vorhaben, in denen individuelle Lösungen aus Komponenten integriert werden. Wir betrachten zwei Ebenen, auf denen Integration stattfindet: innerhalb eines Softwaresystems und übergreifend über mehrere Softwaresysteme.

2 Architekturzentriertes Vorgehen

Ein wesentliches Merkmal von erfolgreichen Softwarearchitekturen ist die strikte Trennung von technischen (T) und anwendungsfachlichen (A) Aspekten. Die von sd&m entwickelte Standard-Softwarearchitektur Quasar [DS00] verfolgt diesen Grundsatz. Er ermöglicht unter Anderem eine effiziente Teambildung und parallele Bearbeitung einer Projektaufgabe. Konsequenterweise findet man die A/T-Trennung in unserem Vorgehensvorschlag wieder.

Integrationsprojekte stützen sich bei der Umsetzung der Anforderungen zu einem großen Teil auf Produkte oder integrieren Dienste von unternehmensinternen und -externen

Systemen. Die Systemerstellung ist daher auf die fachliche und technische Architektur der integrierten Bestandteile auszurichten.

3 i-Projekte

Definition: Wir bezeichnen ein Softwareprojekt als Integrationsprojekt oder kurz i-Projekt, wenn die Integration (Produkteinführung, Dienstnutzung, Infrastruktureinbindung) einen signifikanten Anteil am Gesamtprojektvolumen (> 30%) hat.

In i-Projekten muss sich die Systemerstellung an der gegebenen fachlichen und technischen Architektur der integrierten Produkte und Dienste ausrichten. Andernfalls lassen sich die erhofften Kosteneinsparungen nicht realisieren. Im Gegenteil, ein durch zahlreiche Anpassungen entstandenes „Individualprodukt“ bedeutet Probleme und erhöhte Aufwände in Weiterentwicklung und Wartung.

Abbildung 1 gibt einen Überblick über Stufen und Phasen eines i-Projekts.

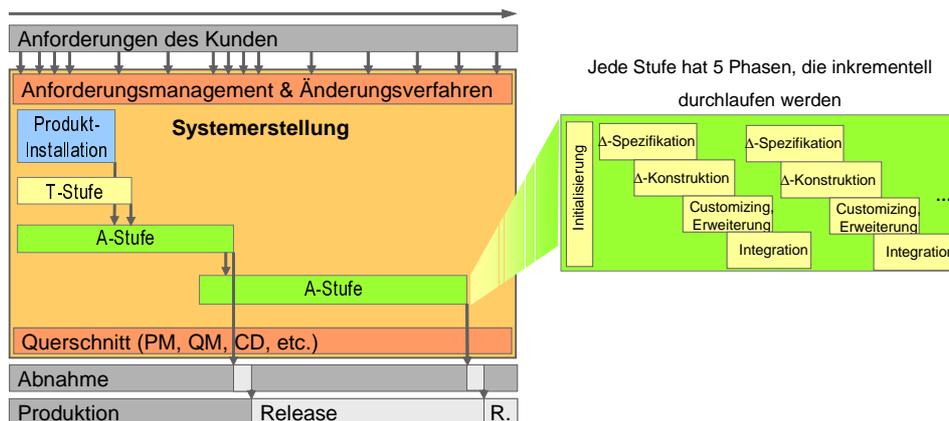


Abbildung 1: Projektmodell für i-Projekte

Wie in Individualsoftwareprojekten auch, so wird ein i-Projekt in T- und A-Stufen unterteilt, um die in der Architektur vorhandene Entkopplung und Aufgabentrennung organisatorisch zu nutzen. Stufen können, müssen aber nicht zu einem Release führen. Das Vorgehen in einer Stufe ist inkrementell mit einer Initialisierung und vier Phasen.

Ein i-Projekt beginnt mit einer Stufe zur Installation der zu integrierenden Produkte (beziehungsweise dem Verfügbarmachen von Diensten anderer Systeme). Ziel ist es dabei, dass das Entwicklungsteam mit diesen arbeiten kann.

Das Vorgehen in den einzelnen Stufen orientiert sich an der fachlichen und technischen Architektur des Produkts und wird in die folgenden Phasen unterteilt:

- *Delta-Spezifikation*: Spezifikation der vom Produkt abweichenden Funktionalität, möglichst gemeinsam mit den Nutzern anhand des Produkts. Dabei sind die Kosten für gewünschte Abweichungen vom Produktstandard transparent zu machen – vor allem solche, die zukünftige Upgrades erschweren. Besonderes Augenmerk ist auf nichtfunktionale Anforderungen zu legen.
- *Delta-Konstruktion*: Design von projektspezifischem Customizing und Erweiterungen. Dabei gibt das Produkt die technische Architektur vor. Um Upgrades zu erleichtern, sollten Erweiterungen möglichst nur an vom Hersteller vorgesehen Stellen vorgenommen werden.
- *Customizing und Erweiterung*: fachliche / technische Konfiguration und Programmierung. Dafür ist ein sorgfältiges Konfigurationsmanagement notwendig.
- *Integration*: Test im Zusammenspiel mit Nachbarsystemen. Dabei sollten neben den projektspezifischen Anpassungen und Customizing zusätzlich die kritische Standardfunktionalität des Produkts getestet werden.

Der Namenszusatz „Delta“ bringt dabei zum Ausdruck, was in der Literatur oft als „bridging the architectural gap“ beschrieben wird: in i-Projekten werden Lücken zwischen den Anforderungen beziehungsweise der Zielarchitektur und dem Produkt geschlossen.

Ein inkrementelles Vorgehen bietet sich auch deshalb an, weil Leistungen und Beschränkungen von Produkten sukzessive im Projektverlauf ermittelt werden müssen. Diese weichen oft erheblich von der Prospektlage oder den Erwartungen der Nutzer ab. [WHS02] beschreibt ein Verfahren, wie die dabei auftretenden Risiken abgeschätzt werden können.

4 i-Programme

i-Programme etablieren eine Architektur und Infrastruktur für die Integration vieler Anwendungen. Auf dieser Basis wird eine weiterentwickelte Anwendungslandschaft implementiert.

Programme umfassen ein ganzes Portfolio von Einzelprojekten. i-Projekte werden häufig im Rahmen von Programmen durchgeführt. Beispiele sind Business Intelligence-, CRM-, EAI- und Portal-Programme. Abbildung 2 gibt eine Übersicht über die Phasen eines i-Programms.

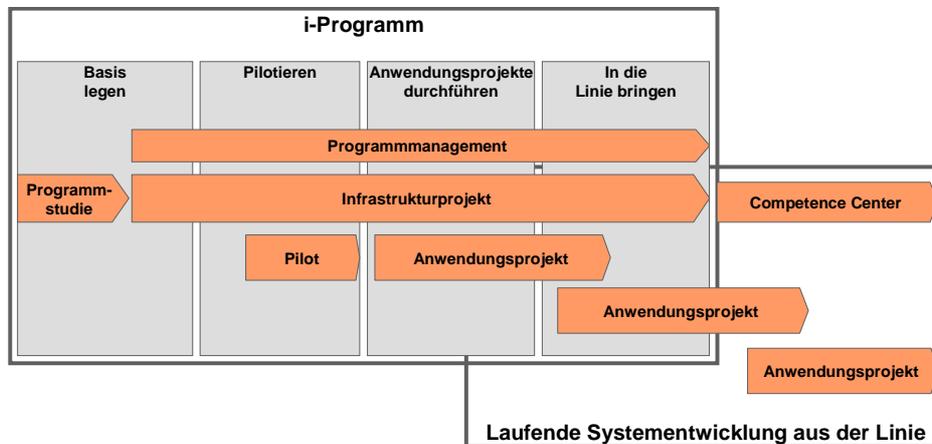


Abbildung 2: Phasen eines i-Programms

In der ersten Phase wird mit einer Studie die *Basis* für ein i-Programm gelegt. Als Ergebnis ist das Programm geplant, die Gesamtarchitektur festgelegt, die wichtigsten Produkte ausgewählt und die ersten Projekte aufgesetzt. Nach der Einführung der wichtigsten Infrastrukturprodukte kann in der zweiten Phase anhand eines ausgewählten Anwendungsprojekts die *Pilotierung* erfolgen. Als Ergebnis ist die Architektur und Infrastruktur validiert und detailliert, Organisation und Prozesse aufgesetzt und ein erster fachlicher Nutzen erzielt (*Quick Win*). In der dritten Phase werden mehrere *Anwendungsprojekte* durchgeführt. Hier wird der eigentliche fachliche Nutzen erzielt. Architektur und Infrastruktur sind im Programm etabliert; Organisation und Prozesse sind eingespielt. In der vierten Phase schließlich wird die Weiterentwicklung und Pflege der entstandenen Systeme in die *Linienorganisation* des Unternehmens gebracht. Aus Mitarbeitern des Programm-Teams wird ein *Competence-Center* aufgebaut, welches Entwicklung, Support und Betrieb weiterführen.

Der Schnitt der Projekte eines i-Programms richtet sich nach der Architektur der Ziel-Anwendungslandschaft. Wir wollen im Weiteren nur auf die architektonischen Aspekte des Vorgehens eingehen.

4.1 Referenzarchitekturen und Produktlandkarten

In der Programmstudie wird die Basis für das Gesamtvorhaben gelegt. Sowohl die Analyse als auch die Sollkonzeption erfolgen *architekturzentriert*. Mittels *Referenzarchitekturen* wird die Ist-Architektur erhoben, die Anforderungen kategorisiert, sowie die Soll-Architektur entworfen. So können Migrationspfade von Ist- nach Sollarchitektur aufgezeigt werden. sd&m hat Referenzarchitekturen für BI, CRM, EAI, Portale, und Anwendungslandschaften als Ganzes entwickelt und setzt diese in i-Projekten und Programmen ein [sdm04].

Produktlandkarten geben eine Übersicht über die wichtigsten kommerziellen und nicht-kommerziellen Produkte. Wir strukturieren Produktlandkarten nach den Diensten der Referenzarchitekturen. Sie helfen bei der raschen Identifikation von Shortlists für die Produktauswahl.

4.2 Infrastruktur- und Anwendungsprojekte

Das *Infrastrukturprojekt* bildet das Rückgrat des i-Programmes. Es verkörpert die Gesamtvision und bietet den *Anwendungsprojekten* eine Integrationsinfrastruktur als Dienstleitung an. Die Anwendungsprojekte implementieren Anwendungssysteme unter Berücksichtigung und Nutzung von Dienstleistungen und Vorgaben des Infrastrukturprojektes.

Trotz der Trennung ist entscheidend für den Erfolg, dass Infrastruktur- und Anwendungsprojekte eng zusammenarbeiten. Integrationsvorhaben ohne fachliche Nutzer enden sonst in technischen Investitionsruinen.

5 Fazit

i-Projekte erfolgreich durchzuführen erfordert viel Erfahrung. Für die Integration verschiedener fachlicher und technischer Systeme und Produkte – mit jeweils unterschiedlichen Rahmenbedingungen – hat sich ein architekturzentriertes Vorgehen bewährt. Dabei bleiben hersteller- und hypeunabhängige Referenzarchitekturen wichtig, um Produkte und Trends einordnen und vergleichen zu können.

Weiterhin entscheidend für den Projekterfolg ist der Faktor Mensch. i-Projekte haben viele Beteiligte: Auftraggeber, Fachbereiche, IT-Organisationen und Partner – jeweils mit eigenen Zielen und Vorstellungen. Diese Menschen zu einer konstruktiven und ziel führenden Zusammenarbeit zu bringen ist ein wesentlicher Erfolgsfaktor von i-Projekten.

Literaturverzeichnis

- [DS00] Denert, Ernst: Siedersleben, Johannes: Wie baut man Informationssysteme? Überlegungen zur Standardarchitektur. Informatik Spektrum Nr. 4/2000, S. 247-257.
- [Kel02] Wolfgang Keller: Enterprise Application Integration – Erfahrungen aus der Praxis. dpunkt.verlag, Januar 2002.
- [sdm04] sd&m: Leistungsangebot für Integrationsprojekte.
<http://www.sdm.de/de/angebot/leistungen/iprojekte/index.html>
- [WHS02] Kurt C. Wallnau, Scott A. Hissam, Robert C. Searcord: Building systems from commercial components. Addison-Wesley, 2002