

KÖNIGSDISZIPLIN DER SOFTWAREARCHITEKTEN



Ich bin ein großer Fan von **Gerhard Wohland**. Auch wenn, oder gerade weil er in seinem Buch über „Denkwerkzeuge für dynamische Märkte“ kein Hype-Thema geschaffen hat, bekomme ich in vielen Vorträgen, in denen ich ihn zitiere, nach kurzem Staunen viel Zustimmung. Er sagt: „Jeder Arbeitsprozess wird von einem Problem angestoßen und durch seine Lösung beendet. Bei trägen Problemen besteht ein Prozess fast nur aus Ereignissen, mit deren Wiederholung gerechnet werden kann. Der Anteil überraschender Probleme ist vernachlässigbar gering. Die Prozessbeschreibung kann sich ohne Schaden auf die Struktur des Prozesses beschränken. Bei dynamischen Problemen besteht der Lösungsprozess fast nur aus überraschenden Ereignissen. Die Behandlung einer Überraschung benötigt Ideen auf Basis von Prinzipien. Nur qualifizierte und motivierte Menschen können mit Überraschungen sinnvoll umgehen.“

Das klingt zunächst unspektakulär, begründet aber eine Theorie, die nicht nur viele aktuelle Diskussionen um Möglichkeiten und Grenzen einer agilen Entwicklungsorganisation aus dem Religiösen in das Wissenschaftliche hebt. Sie beschreibt auch eine solide Grundlage für die Notwendigkeiten eines modernen Business Process Management (BPM).

Bereits vor 20 Jahren ging nämlich eine Welle der Geschäftsprozess-Automatisierung durch die Großraumbüros. Nach dem Vorbild der Automatisierung industrieller Produktionsprozesse sollte auch die Verwaltung vollautomatisiert werden. Zentrale Betriebsorganisationen schwangen sich dazu auf, erfahrenen Sachbearbeitern vorzurechnen, wie sie nun endlich schneller und effizienter ihre Arbeit erledigen konnten. Große Erfolge wurden vor allem in den Bereichen verzeichnet, in denen Dokumente nicht nur gescannt, sondern automatisch erkannt und in meist einfachen und kurzen „Dunkelprozessen“ automatisiert werden konnten, also typischen Beispielen für Wohlands träge Prozesse. Ihre Grenzen fanden diese Systeme aber immer dann, wenn die zu unterstützenden Prozesse eben nicht nur auf der sich wiederholenden Anwendung von Regelwissen basierten. Schwierig wurde es, wenn unvorhergesehene Umstände eine Bearbeitung auf der Basis von Erfahrung und Können erforderten oder wenn die zu unterstützenden Prozesse auf einer komplexen Landschaft von Anwendungssystemen – vielleicht sogar über Unternehmensgrenzen hinweg – aufsetzen mussten.

Hier setzen moderne, offene BPM-Ansätze an. Inzwischen werden die Kommunikation und Synchronisation von Anwendungen sehr viel besser beherrscht und auch für die Beschreibung der Prozesse selbst gibt es spätestens mit BPMN 2.0 eine geeignete Standardsprache. Unter Nutzung dieser Möglichkeiten und in Anerkennung dieser Prozesskomplexität muss nun nicht mehr auf Gedeih und Verderb automatisiert werden: „Anwendungsintegration“ heißt das Schlagwort. **Bernd Rücker** beklagt in seinem Artikel „Befreit die Prozesse“ deswegen die Zero-Code-Lügen der Black-Box-BPM-Suiten. **Hanno Hüther** und **Tobias Walter** beschreiben eine Reihe von Anwendungsszenarien für moderne BPM-Systeme, die dieser Komplexität Rechnung tragen. **Jonas Grundler** und **Lars Rübesamen** geben in ihrer „BPM-Brautschau“ wertvolle Tipps, wie man für seine Prozesslandschaft das geeignete Werkzeug findet.

Ich gehe aber noch einen Schritt weiter. Mit der Fokussierung auf die Anwendungsintegration ist Business Process Engineering eine Königsdisziplin für Softwarearchitekten geworden. Mit der Einführung eines BPM-Systems wird eine historisch gewachsene, heterogene Technologielandchaft nicht mehr in einem aussichtslosen Kampf bekämpft, sondern gezähmt und zum Diener sich gegebenenfalls schnell verändernder Geschäftsprozesse gemacht. Nur wenn ich diese Integration softwaretechnisch und -architektonisch sauber mache, habe ich eine langfristige Investitionssicherung geschafft. Damit ist die Einführung solcher BPM-Systeme ein Paradebeispiel dafür, dass Softwarearchitektur und Software-Engineering zur Kernkompetenz jedes Unternehmens mit komplexen und dynamischen Prozesslandschaften gehören müssen, auch und gerade wenn inzwischen große Teile der Anwendungslandschaft aus Standardsoftware bestehen und größere Individualentwicklungen in den letzten Jahren immer häufiger ausgelagert werden. Aber wir – die Leser und die Redaktion des OBJEKTSpektrum – haben das ja schon immer gewusst oder zumindest ganz tief in unserem Herzen gefühlt, oder?

Viel Spaß bei der Lektüre dieser Ausgabe des OBJEKTSpektrum wünscht


Ihr Thorsten Jauning