



Engpässe

Als ich vor einigen Tagen mit einem Vertreter der „Theory of Constraints“ zusammensaß, schleuderte er meinen Kollegen und mir den Satz entgegen: „Ihr müsst in Entwicklungsprozessen keine Engpässe analysieren. Der Engpass ist immer die Konzeption!“ Auch wenn diese Aussage natürlich in ihrer Absolutheit falsch ist, hat sie doch einen wahren Kern. Wer kann nicht von vielen Beispielen erzählen, wo über die langsame Entwicklungsgeschwindigkeit in der IT geklagt wird, ein kurzer Blick auf den Gesamtprozess aber zeigt, dass mindestens doppelt so viel Zeit für die Analyse und Spezifikation gebraucht wird wie für die Entwicklung.

Spätestens an dieser Stelle kommen in der Informatik immer wieder Modelle der verschiedensten Art auf den Plan. Waren es in grauen Informatik-Vorzeiten Datenmodelle, die die IT revolutionierten, so waren in den letzten Jahren Geschäftsmodelle der einen oder anderen Art en vogue. Letztere sind auf jeden Fall ein Beispiel dafür, wie versucht wird, Anforderungen oder das Außenverhalten von Anwendungen zu abstrakt zu beschreiben. Damit bewegen sich Prozessmodelle in der Tradition von Modellen in anderen Disziplinen, beispielsweise Architektur und Ingenieursdisziplinen.

Modelle sind leicht, schnell und kostengünstig zu bauen und verdeutlichen den Auftraggebern schnell, was für ein Produkt sie zu erwarten haben. Ganz ohne Tools geht das natürlich besonders schnell. **Martin Klewitz** stellt in diesem Sinne in seinem Artikel „Ein Bild sagt mehr als tausend Worte“ einen sehr konsequenten und praktikablen Ansatz vor, um Modelle ganz ohne elektronische Tools zu skizzieren und diese damit auch als reines Kommunikationsinstrument klarzustellen.

Datenmodelle dagegen sind der Urvater aller Konstruktionsmodelle in der IT, denen Objektmodelle, Klassendiagramme und vieles mehr folgten. Bei diesen Modellen steht nicht das Außenverhalten, sondern die Struktur der Anwendung oder der verwendeten Daten im Vordergrund. Diese Modelle sind als Konstruktionshilfe extrem wertvoll und auf diese Weise sogar die Grundlage für Codegeneratoren verschiedenster Couleur geworden. Hier nimmt das Unglück seinen Lauf.

Modellgetriebene Architekturen haben einige Jahre lang die Anwendungsentwicklung geprägt. Auf der Basis von Modellen und vorimplementierten Architekturmustern werden Softwaregeneratoren angeworfen. Damit wird tatsächlich sehr schnell gut strukturierter Code produziert. **Thomas Stahl** und **Christoph Gutmann** geben in ihrem Artikel in dieser Ausgabe von OBJEKTSpektrum einen wirklich wertvollen Überblick über die inzwischen reichhaltigen Erfahrungen und eine interessante Klassifikation modellgetriebener Entwicklungsansätze.

Das Problem beginnt aber mit der Wartung. Mir geht der Dialog mit einem Entwicklungsverantwortlichen nicht aus dem Kopf, als er mir mit stolzgeschwellter Brust erklärte, dass seine Anwendung aus unvorstellbar vielen Millionen Codezeilen bestehe und damit eines der komplexesten Softwaresysteme überhaupt sei. Obwohl die generierte Grundstruktur des Codes gut war, fing das Problem bei der Suche nach und der Reparatur von Fehlern an, weil der zu analysierende Code zu komplex war und der gesamte Generator-Apparat selbst bei nur kleinen Reparaturen verdammt schwerfällig war. Nicht zuletzt deswegen sind die modellgetriebenen Entwicklungsansätze etwas außer Mode gekommen. Spannend ist vor diesem Hintergrund daher die Diskussion, warum nun auch das modellbasierte Testen nicht so recht vorankommt, die **Christian Brandes** in seinem Beitrag anstößt.

Was also ist die Quintessenz? Brandes sagt hierzu: „Es ist müßig zu erwähnen, dass die Komplexität von IT-Lösungen in Zukunft weiter zunehmen wird. Ohne Abstraktion und eine zielgruppenkonforme Sichtenbildung – also Modellierung – wird diese Komplexität nicht beherrschbar sein. Die Nutzung von Modellen ist also keine Spielerei, sondern ein unverzichtbares Hilfsmittel.“ Damit wird uns ein wichtiges Werkzeug an die Hand gegeben, um den Engpass der Konzeption ein wenig auszuweiten. Ich persönlich habe aber meine Zweifel, ob modellbasierte Generierungsansätze noch eine große Zukunft haben, solange sie nicht ihre bisherige Schwergewichtigkeit verlieren. So bilden sie nur einen zusätzlichen Engpass im Gesamtprozess.

Ihr Thorsten Janning, Chefredakteur OBJEKTSpektrum

NEWS
LETTER

Alle zwei Monate kostenlos

• Heftinhalte • ausgewählte Artikel im PDF-Format • ergänzende Weiterbildungsangebote
Anmeldung unter www.sigs-datacom.de/os/newsletter/