Geleitwort

Softwarearchitektur: Das ist die Königsdisziplin des Software-Engineering. Sie befasst sich mit der Gestaltung von Softwaresystemen, mit deren Struktur im Kleinen und im Großen. Bei der Gestaltung im Großen geht es um den Aufbau eines Systems aus Komponenten, um die Prinzipien und Kriterien, die eine gute Architektur auszeichnen. Im Kleinen dagegen um die Gestaltung einer einzelnen Komponente: um ihre Außensicht einerseits, d.h. um ihre Schnittstelle zu anderen Komponenten, und um ihre innere Struktur andererseits, also darum, mit welchen Daten sie operiert und wie sie ihrerseits aus Komponenten bzw. Modulen zusammengesetzt ist.

Softwarearchitektur ist der Gegenstand dieses Buches, und zwar in einer besonderen Ausprägung: Quasar, die Qualitätssoftwarearchitektur von sd&m. Dieses Wortungetüm kann man auf zweierlei Weise interpretieren: erstens als Qualitätssoftware-Architektur, d.h. eine Architektur für qualitativ hochwertige Software, und zweitens als Qualitäts-Softwarearchitektur, mithin eine qualitätsvolle Architekturblaupause zur Gestaltung von Software. Mit diesem Buch ist ein wichtiger Meilenstein erreicht in einem Vorhaben, das ich selbst vor zehn Jahren initiiert habe. Ich wollte eine Standardarchitektur betrieblicher Informationssysteme schaffen, eine Blaupause, anhand derer Software bei sd&m entwickelt werden sollte, aufbauend auf Ideen und Konzepten, die über lange Jahre entstanden und in dem sog. Schwarzbuch »Software-Engineering« von Johannes Siedersleben und mir Anfang der 90er Jahre publiziert worden waren. Es fing mit einem geförderten Forschungsprojekt an und hat viel länger gedauert als gedacht. Aber nun ist dieses Vorhaben am Ziel. Quasar existiert, nicht nur als Vorlage für den Entwurf von Softwaresystemen, sondern auch implementiert durch eine Reihe lauffähiger Komponenten.

Quasar gründet auf drei bis vier Jahrzehnten Entwicklung im Software-Engineering und inkorporiert viele der guten, bekannten Konzepte. In deren Integration und Nutzung ist Quasar bewahrend und innovativ zugleich. Ein zentraler Gedanke des Buches sind die »Software-Blutgruppen«. Das ist nicht nur eine pfiffige und griffige Metapher, sondern eine neue Art und Weise, das Konzept des

»Separation of Concern« so zu formulieren, dass ein handhabbares Kriterium entsteht zur Beurteilung der Güte eines Softwareentwurfs.

Dieses Buch enthält harten Stoff und vermeidet die häufig anzutreffenden wolkigen Ausführungen über Architektur ebenso wie die Beschränkung auf reine Programmiertechnik. Es ist methodisch fundiert und technisch konkret – Softwarearchitektur für die gehobene Praxis.

Ernst Denert
Berlin, im Mai 2004

Kloster St. Josef in Zangberg: ein Ort zum Lehren, Lernen und Diskutieren

Vorwort

Das Quasar-Projekt begann am 5. März 1998 im Gasthaus Wörnbrunn in Grünwald bei München. Es war ein wunderbarer Tag im Vorfrühling, die Sonne schien in den Besprechungsraum, und *Quasar* war geboren.

Was wir wollten: die vielen Architekturideen vieler sd&m-Projekte einsammeln und in eine wieder verwendbare Form bringen. Grundlage waren Datenabstraktion und Trennung der Zuständigkeiten – das hatten wir schon lange vor der Objektorientierung mit Cobol, VSAM und MVS erfolgreich umgesetzt.

Dieses Buch präsentiert unter dem Namen *Quasar* viele bewährte Ideen – vielleicht etwas aufbereitet und modernisiert – und auch ein paar neue. Aber was ist mit der Gretchenfrage, die so manchen erfahrenen Redner aus dem Konzept bringt: Warum stimmt das alles, warum sollte der Zuhörer bzw. Leser irgendetwas davon glauben?

Hier tun sich Softwareingenieure schwer: Zwar kann man die Korrektheit eines Algorithmus beweisen und seine Komplexität berechnen – aber hat schon mal jemand den Nutzen der Objektorientierung oder den von Entwurfsmustern nachgewiesen oder gar quantifiziert? Ich glaube nicht. Objektorientierung und Muster haben sich auch ohne Nachweis durchgesetzt – sie haben sich durchgesetzt, weil sie in der Praxis funktionieren und weil wir den Nutzen auch ohne wissenschaftliche Messungen spüren.

Quasar hat sich bei sd&m durchgesetzt – Denken in Komponenten, Kategorien und Schnittstellen ist zum Standard geworden, ohne den Einzelnen in seiner Freiheit zu beeinträchtigen. Die Schulungen (Schools), die sd&m für die eigenen Mitarbeiter veranstaltet (und gelegentlich auch für andere), sind ein wichtiger Nährboden für Quasar und eine wertvolle Plattform der Kommunikation: Da gibt es Programmer-Schools, Designer- und Chefdesigner-Schools (und noch ein paar andere). Jede School dauert drei bis fünf Tage; sd&m-Kollegen halten sie für sd&m-Kollegen. Dort werden die Elemente von Quasar gelehrt, gelernt und vor allem aber auch diskutiert und weiterentwickelt. Diese Schools gewährleisten, dass Quasar bei sd&m gut bekannt ist und angewendet wird. Sie finden traditionsgemäß im Kloster Zangberg statt – in the middle of nowhere, irgendwo im

oberbayerischen Hinterland. Zangberg ist der ideale Ort für arbeitsintensive Veranstaltungen: Die Ruhe ist vollkommen, die Ablenkung gleich null, und die Teilnehmer lassen sich abends nur widerwillig in den Feierabend befördern.

Ein wichtiger Schritt in der Entwicklung von Quasar war ein sd&m-internes Projekt mit dem wenig aussagekräftigen Namen *Themenarbeit*. Themen waren z.B. Fehlerbehandlung, Transaktionen, Persistenz und andere, die sich teilweise als Kapitel dieses Buches wiederfinden. *Themenarbeit* bedeutete vor allem die Analyse konkreter Projekte – welche Lösungen wurden eingesetzt, was hat gut funktioniert und was nicht? Im Rahmen der Themenarbeit wurde ein knappes Dutzend sd&m-Projekte untersucht; über 20 Kollegen waren beteiligt.

Ein Praktikum in Software-Engineering, das ich im Wintersemester 2003/04 an der Fachhochschule Rosenheim gehalten habe, war nichts anderes als ein Quasar-Feldversuch: 26 Studenten entwickelten in sechs Teams ebenso viele Softwaresysteme nach den Prinzipien von Quasar. Sie bauten einen Forth-Interpreter, eine Fahrstuhlsimulation, eine Simulation des Schafkopf-Kartenspiels, den *StatementManager* der Persistenzkomponente von Kapitel 9, ein Framework für Suchalgorithmen (z.B. für das Acht-Damenproblem) und eine Bibliothek als Erweiterung von `java.math`. Einige dieser Systeme haben auch Spuren in diesem Buch hinterlassen. Das Ergebnis des Feldversuchs: In allen sechs Projekten standen Softwarekategorien und Architektur rasch fest und sie änderten sich bis zum Ende nur noch geringfügig. Es gab keine Sackgassen, keine Umbauten – alle Projekte kamen ans Ziel, und zwar nicht nur irgendwie oder mit Ach und Krach; alle Systeme sind gut dokumentiert und laufen stabil; alle Teams sind stolz auf ihr System (zu besichtigen unter www.fh-rosenheim.de/~siedersleben).

Dieses Buch präsentiert Quasar in dem heute erreichten Stand. Aber Quasar ist im Grunde eine Vision, die erst dann Wirklichkeit wird, wenn wir für die wichtigen Standardprobleme dieser Welt (von Autorisierung über Persistenz bis hin zu Bedienoberflächen) präzise dokumentierte Standardschnittstellen und Standardarchitekturen besitzen. Bis dahin ist es noch ein weiter Weg – aber einer, der sich lohnt.

Dank

Das Quasar-Projekt wäre schnell im Sande verlaufen, hätte nicht *Ernst Denert* die Rolle des Förderers, Mentors und PR-Managers übernommen. Sein 1992 erschienenes Buch »Software-Engineering« ist auch heute eine wertvolle Grundlage; viele Gedanken von Quasar sind dort bereits enthalten.

Nach Denerts Weggang von sd&m im Juni 2001 fanden sich andere Quasar-Förderer: Uwe Dumslaff, Kai Grambow, Burkhard Kehrbusch, Edmund Küpper und Dirk Taubner haben Quasar immer nach Kräften unterstützt und mit Budget versorgt.

Trennung der Zuständigkeiten, Datenabstraktion, Geheimnisprinzip – diese Ideen verdanken wir *Dave Parnas*, dessen wichtigste Publikationen Dave Weiss und Dan Hoffman in einem Sammelband vereinigt haben (siehe [Hoffmann 2001]). Dort steht eigentlich schon alles – das ist mitunter frustrierend. Normalen Sterblichen wie dem Autor bleibt nur, den Urtext zu interpretieren und anzuwenden.

Viele Freunde und Kollegen haben das Manuskript oder Teile davon gelesen, kritisiert und manchmal auch für gut befunden. Das waren Gerd Beneken, Manfred Broy, Ernst Denert, Uwe Dumslaff, Stephan Kurpjuweit, Bernhard Humm, Klaus Nünninghoff, Karl Prott und Johannes Weigend.

Martin Haft hat alle Kapitel sorgfältig und mehrfach gelesen und dabei zahllose Fehler, Ungenauigkeiten und Inkonsistenzen gefunden. Für die verbleibenden bin ich selbst verantwortlich.

Thomas Nietsch ist unser QSL-Experte. Er hat alle QSL-Beispiele selbst geschrieben oder korrigiert, und er ist Koautor von Anhang A. Stephan Kurpjuweit hat den Abschnitt 4.8 über SAAM beigesteuert.

Viele sd&m-Kollegen haben an den Quasar-Komponenten mitgearbeitet, aber einer hat die Basis geschaffen, vorzugsweise im Urlaub oder während verletzungsbedingter Zwangspausen: Oliver Juwig.

Die beiden Quasar-Broschüren [Quasar 2003a] und [Quasar 2003b] sind die Vorläufer dieses Buches: In nur 18 Monaten fanden rund 3500 Exemplare ihren Weg auf die Schreibtische von Kunden und Kollegen von sd&m. Sie waren eine Teamarbeit im besten Sinn des Wortes, und alle Koautoren der Broschüren sind

indirekt auch Koautoren dieses Buches. Hier sind sie in alphabetischer Reihenfolge: Gerd Beneken, Axel Burghof, Till Dalkowski, Frank Dörr, Helmut Duschinger, Ulrike Hammerschall, Andreas Hess, Alex Hofmann, Bernhard Humm, Alexander Jost, Christian Kamm, Klaus Nünninghoff, Stefan Scheidle, Boris Zech und Jürgen Zeller.

Die Zusammenarbeit mit dem dpunkt.verlag war professionell und in jeder Beziehung erfreulich – das Verdienst von Christa Preisendanz.

Bei allen Genannten bedanke ich mich herzlich; ohne sie gäbe es dieses Buch nicht.

Johannes Siedersleben
München, im Mai 2004