

HANSER



Vorwort

Ernest Wallmüller

Software Quality Engineering

Ein Leitfaden für bessere Software-Qualität

ISBN: 978-3-446-40405-2

Weitere Informationen oder Bestellungen unter

<http://www.hanser.de/978-3-446-40405-2>

sowie im Buchhandel.

Geleitwort

Software ist zu einem wichtigen Wirtschaftsfaktor geworden, und die Softwareindustrie ist eine aus unserem Wirtschaftsleben nicht mehr wegzudenkende Wachstumsbranche. Softwareprodukte werden immer häufiger zentrale Bestandteile komplexer Systeme, die technische oder betriebswirtschaftliche Prozesse steuern oder unterstützen. Aber auch im täglichen Leben werden wir immer stärker – meist unbemerkt – abhängig von funktionierender Software. In einzelnen Bereichen, wie z. B. der Flugsicherung oder der Energietechnik, kann Sicherheit und Leben vom richtigen Funktionieren der dort eingesetzten Software abhängen. Die Funktionsfähigkeit und Zuverlässigkeit von Software ist Voraussetzung für die Funktionsfähigkeit des Gesamtsystems, dessen Bestandteil sie ist.

Die rasante Entwicklung auf dem Gebiet der Informations- und Kommunikationstechnik führte im Bereich der Softwareentwicklung aber zu solchen Schwierigkeiten, dass man bereits vor mehr als 40 Jahren den Begriff „Softwarekrise“ prägte. Das heißt zwar nicht, dass die Hardware stets fehlerfrei ist, die Praxis zeigt jedoch, dass der Grund für die Schwierigkeiten, die oft auftreten, wenn man computerunterstützte Systeme einsetzen will, meist in einer mangelbehafteten Software zu suchen ist. Es ist leicht nachzuweisen, dass Softwareprodukte fast immer eine geringere Qualität aufweisen als andere technische Produkte. Daher erhebt sich die Frage nach den Ursachen der oft miserablen Qualität von Software.

Software ist in Bezug auf die Konstruktion komplexer als vielleicht jedes andere vom Menschen errichtete Gebilde. Die Vergrößerung der Funktionalität eines Softwaresystems ist keine Wiederholung der gleichen Elemente in größerem Maßstab, sondern immer eine Vergrößerung der Anzahl unterschiedlicher Elemente. In den meisten Fällen ist die Beziehung zwischen den Elementen äußerst vielschichtig. Die Komplexität des Gesamtsystems steigt daher bei Erweiterungen viel stärker als linear. Die Komplexität ist eine unvermeidbare Eigenschaft von Software. Zu ihrer Meisterung reicht der gesunde Hausverstand nicht aus. Nur wer über ausreichende theoretische, technische und organisatorische Kenntnisse auf dem Gebiet der Softwaretechnik verfügt, ist in der Lage, die Probleme der Softwareentwicklung zu meistern.

Eine der Hauptursachen für die mangelnde Qualität von Softwareprodukten besteht darin, dass die in der Praxis eingesetzten Methoden und Techniken zur Konstruktion von Software oft hinter dem Stand der Technik, d. h. den wissenschaftlichen Erkenntnissen auf diesem Gebiet, hinterherhinken. Die Methodenkenntnisse der am Softwareentwicklungsprozess Beteiligten sind selten auf dem aktuellen Stand, weil – nicht zuletzt als Folge des Einsatzes unfachmännischer Techniken – der Entwicklungs- und Wartungsaufwand unnötig hoch ist und dadurch kaum Zeit bleibt für eine entsprechende „Nach- und Umrüstausbildung“ der Softwareentwickler. Statt saubere, fundierte Konstruktionsprinzipien zur Meisterung der Kom-

plexität und zur Qualitätssicherung anzuwenden, blicken Softwareentwickler oft mit Stolz auf ihre komplizierten Gebilde. Je undurchschaubarer das Ganze wird, desto wertvoller erachten sie es. Was nicht selten dazu führt, dass nur die Autoren und nach geraumer Zeit oft nicht einmal sie selbst den eigenen Code noch verstehen.

Eine der wichtigsten Maßnahmen zur Herstellung von Qualitätsprodukten, nämlich die Institutionalisierung des Qualitätsmanagements, fehlt bei vielen Softwareproduzenten gänzlich. Beobachtet man den Herstellungsprozess von Produkten, die sowohl aus Hardware- als auch aus Softwarekomponenten bestehen, z. B. im Maschinen- und Anlagenbau, fällt häufig auf, dass die Hardwareteile des Systems den üblichen Qualitätskontrollen und Sicherungsmaßnahmen unterzogen werden, während die Softwarekomponenten lediglich auf ihre Funktionsfähigkeit getestet werden. Qualitätsmängel, die das Design und die Struktur betreffen und sich später bei Änderungen und Erweiterungen katastrophal auswirken können, bleiben meist unerkannt.

Dies sind nur einige der Ursachen für die schlechte Qualität von Softwareprodukten und Systemen, in denen Software ein wesentlicher Bestandteil ist. Die Liste ließe sich durchaus verlängern. Weil die Informations- und Kommunikationstechnologie und damit die Softwaretechnik eine Schlüsseltechnologie für unsere hochtechnisierte Gesellschaft darstellen, kann und darf man sich mit dem unbefriedigenden Qualitätsniveau von Software nicht abfinden. Im Gegenteil – man muss alles daran setzen, diesen Zustand zu verbessern. Ein Handbuch bzw. Leitfaden zur Erzielung einer besseren Softwarequalität ist daher naturgemäß ein willkommener Beitrag.

o. Univ.-Prof. Dipl.-Ing. Dr. Gustav Pomberger

Institut für Wirtschaftsinformatik, Software Engineering,
Johannes Kepler Universität Linz

Vorwort

Zahlreiche Ausbildungen im Bereich Qualitätssicherung und Qualitätsmanagement im IT-Umfeld sind in den letzten 15 Jahren entstanden und kontinuierlich besser geworden. Die Anzahl der Ausbildungen und die Anzahl von zertifiziertem Qualitätspersonal sind stark gestiegen. Vor allem in den Bereichen des Testengineering, Testmanagement, Quality System Management und Auditing wurden europaweit und sogar weltweit anerkannte Standardausbildungen geschaffen. Mehrere Lücken in der Ausbildung wurden aber noch nicht geschlossen – etwa die Anforderungen an einen Qualitätsmanager in einem IT-Projekt. Die bis vor wenigen Jahren umfassendste Wissenssammlung war diesbezüglich die des Software Quality Engineering Body of Knowledge der amerikanischen Gesellschaft für Qualität (ASQ).

Am vierten Weltkongress über Software-Qualität im September 2008 in Bethesda, Maryland USA, stellte die japanische IBM-Projektleiterin Yasuko Okazaki den japanischen „Guide to the Software Quality Body of Knowledge (SQuBOK)“ vor. Diese Wissenssammlung über Software Quality Management wurde in einer Zusammenarbeit von der Union of Japanese Scientists and Engineers (JUSE), der Japanese Society for Quality Control (JSQC) sowie einem Konglomerat aus japanischen Universitäten und Unternehmen (bspw. Fujitsu, Hitachi, IBM, NEC, NTT Data, Omron, Panasonic, Unisys u. a.) erstellt. Die Breite und Tiefe der japanischen Wissenssammlung ist beeindruckend und geht weiter als die des Software Quality Engineering Body of Knowledge der amerikanischen Gesellschaft für Qualität (ASQ).

Die europäische Gesellschaft für Qualität (EOQ) hat weder dem amerikanischen noch dem neuen japanischen Body of Knowledge etwas entgegenzustellen. Im Rahmen eines Projektes zur Ergänzung von Ausbildungspfaden für IT-Qualitätsmanagement mit Ansätzen des IT Quality Engineering wurde zusammen mit dem Autor dieses Buches in einem großen Schweizer Finanzinstitut eine Analyse der beiden Wissenssammlungen durchgeführt und mit anderen „Bodies of Knowledges“ benachbarter Disziplinen – etwa dem PMBoK® [URL:PMBOK], dem BABOK® vom internationalen Institut für Business Analysis (IIB) [URL:IIBA] oder dem SWEBOK® [URL:SWEBOK] – verglichen, um ein Konzept für eine komponentenbasierte Ausbildungslandkarte zu erhalten. Die Ergebnisse wurden im Januar 2009 bei den Austrian Quality Days in Wien vorgestellt [Stei09]. Dieses Buch hat unter anderem zum Ziel, das amerikanische und japanische Software Quality Engineering Body of Knowledge weiten Kreisen in Europa bekannt zu machen.

Michael K. Steinhöfel-Córdova

Head of Quality Management, SIX Telekurs Ltd., Zürich