

# Qualität, Software, Software Qualität?

Timo Hahn  
virtual 7 GmbH  
Karlsruhe

## Schlüsselworte

Qualität, Qualitätsmanagement, Quality Gates, Software-Entwicklungsprozess, ISO 9001, EFQM, CMMI, SonarQube, Test, Unit Test, Validierung, Verifizierung, Review

## Einleitung

Der Vortrag soll einen Überblick über Qualität, Qualitätsmanagement im Allgemeinen sowie in der Softwareentwicklung im speziellen vermitteln. In einem ersten Teil, der abstrakt mit den Begriffen umgeht werden die Begriffe erklärt und eingeordnet. Nach diesem allgemein gehaltenen Teil werden die Begriffe auf die Softwareentwicklung angewendet. Dabei wird der Einsatz spezieller Verfahren und Werkzeuge bei der Softwareentwicklung exemplarisch an einigen Punkten aufgezeigt.

## Definitionen

**Qualität:** Qualität ist der Grad, den ein Produkt, ein Dienst, eine Produktkomponente oder ein Prozess einem festgesetzten inhärenten Satz von Anforderungen eines Kunden erfüllt.

Bei der Qualität spricht man von inhärenten Merkmalen, wobei hier ‚inhärent‘ im Gegensatz zu ‚zugeordnet‘ vielmehr ‚einer Einheit innewohnend‘ bedeutet, insbesondere als ständiges Merkmal, wie z. B. im technischen Sinn der Durchmesser einer Schraube.

Inhärente Merkmale sind also solche, die der betrachteten Einheit – im modernen Qualitätsmanagement kann dies ein Produkt, ein Prozess oder ein System sein – innewohnen und damit ihre Beschaffenheit ausmachen. Im Gegensatz dazu gelten Merkmale wie Kosten oder Preis als zugeordnete Merkmale, die der betrachteten Einheit nicht innewohnen.

**Qualitätsmanagement:** Qualitätsmanagement (QM) bezeichnet alle organisatorischen Maßnahmen, die der Verbesserung der Prozessqualität, der Leistungen und damit den Produkten jeglicher Art dienen.

Der Begriff Leistungen umfasst im QM die Dienstleistungen, geht aber über den üblichen Begriff hinaus und betrifft vor allem die innerorganisatorischen Leistungen.

## Qualitätsmanagement

Qualitätsmanagement beschreibt also organisatorische Maßnahmen und ist als eine Kernaufgabe des Managements zu verstehen. In einigen Branchen wie z.B. der Luft- und Raumfahrt, Automobilindustrie oder der Medizintechnik ist ein Qualitätsmanagementsystem vorgeschrieben. Wichtig ist zu verstehen, dass Qualitätsmanagement somit nicht zwangsläufig zu einem höherwertigen Ergebnis führt, sondern nur die vorgegebene Qualität sicherstellt.

Auch der Herstellungsprozess eines Billigprodukts kann einem vollständigen Qualitätsmanagement unterliegen. Qualitätszertifizierungen etwa nach ISO sagen somit nichts über die Qualität eines Produkt oder einer Dienstleistung aus, sondern nur über das Qualitätsmanagement im Herstellungsprozess.

Im Wesentlichen lassen sich die Qualitätsmanagementmodelle in zwei Gruppen unterteilen

- a) Zertifizierbare Normen mit definierten Mindestanforderungen (z.B. EN ISO 9001)

- b) Selbstbewertung des eigenen Qualitätsmanagementsystems und Benchmarking zwischen Wettbewerbern um einen Qualitätspreis (z.B. European Foundation for Quality Management oder EFQM)  
 Für die Softwareentwicklung existieren auch spezielle Modelle wie z.B. das Capability Maturity Model Integration Development (CMMI-DEV) als ein spezialisiertes Prozessmodell. Durch die spezifische Ausrichtung auf Entwicklungsorganisationen kann CMMI detaillierter auf einzelne Prozessaspekte eingehen.

### Qualität in Softwareprojekten

Im Folgenden betrachten wir einen typischen Produktentwicklungsprozess, in seinen einzelnen Phasen.

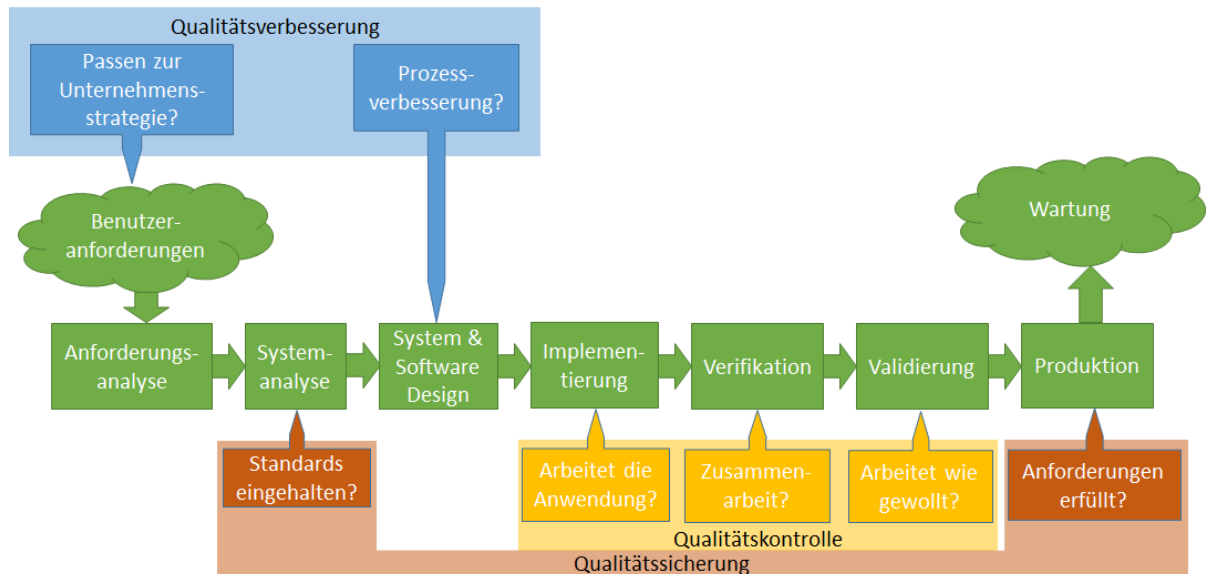


Abb. 1: Produktentwicklungsprozess und Qualitätsmerkmale

Abbildung 1 zeigt wie Qualitätsmanagement in einem typischen Produktentwicklungsprozess (hier grün hinterlegt) Anwendung findet. Im Prozess werden unterschiedliche Techniken zu unterschiedlichen Prozessschritten angewendet. Die gelb hinterlegten Fragen betreffen die Qualitätskontrolle. Diese werden durch Tests und Reviews geprüft. Qualitätssicherung wird durch die braun hinterlegten Fragen abgedeckt. Diese werden durch Audits und 'Muster'-Prüfungen gesichert. Abschließend bleiben die blau hinterlegten Fragen, die der Qualitätsverbesserung dienen. Hier kommen spezielle Qualitätsverbesserungsprojekte und ein kontinuierlicher Verbesserungsprozess zum Einsatz.

Software-Qualitätsmanagement Prinzipien und Best Practices werden für die Entwicklung, Evolution und den Betrieb der Software angewendet. Software-Qualitätsmanagement gilt für Produkte (z.B. um sicherzustellen, dass Anforderungen an die Zuverlässigkeit erfüllt werden), Prozesse (z.B. Umsetzung von Defekterkennung Techniken), Projekte (z.B. Design für Qualität) und Menschen (z.B. Weiterentwicklung von Qualitätstechniken).

Softwarequalität ist ein ernstzunehmendes, aber wegen oftmals fehlender Definition von Qualitätsstandards für Methoden und Ziele, ein schwieriges Feld. Nicht zuletzt da eine absolute Qualität fehlt. Softwarequalität hat viele Facetten, wie z.B. Zuverlässigkeit, Wartbarkeit, Sicherheit usw. Auch die Kosten können als Qualitätsmerkmal angesehen werden. Hier ist nicht gemeint, dass Software oder ein Produkt billig hergestellt werden kann,

um anschließend durch hohe Nachfolgekosten teurer zu werden (z.B. durch notwendige Nachbesserungen).

Durch seine ökonomische Bedeutung hat Qualität auch Einfluss auf die Unternehmensstrategie bzw. wird durch diese beeinflusst. Es macht keinen Sinn zu sagen ‘Wir sind qualitätsbewusst’ oder ‘Wir brauchen ein perfektes Produkt’, wenn niemand dafür bezahlen kann oder will. Aus diesen Grund sollte Qualität immer als ein über den gesamten Lebenszyklus eines Produkts inhärentes Merkmal betrachtet werden.

Qualitätsmerkmale können sich auch gegenseitig widersprechen. So kann z.B. eine Steigerung der Sicherheit mittels Verschlüsselung einen negativen Einfluss auf die Effizienz und die Geschwindigkeit hervorrufen.

Schon in der Konzeption und Strategiephase sollte Qualität betrachtet und entworfen werden, um anschließend während des gesamten Prozess gemessen und verifiziert werden zu können. Qualitätsmanagement ist keine feststehende Wissenschaft. Im Gegenteil, mit der Geschwindigkeit mit der Technologien auftauchen, muss auch das Qualitätsmanagement reagieren. Da sich die grundsätzlichen Prinzipien nicht ändern, haben sich Standards wie ISO 9001 (zur Definition von Produkt und Service Standards) und CMMI (Capability Maturity Model Integration; ein Verbesserungsframework) und weitere entwickelt.

Für die am Entwicklungsprozess beteiligten Aufgabenstellungen (siehe Abb. 1) wurden entsprechende Qualitätssicherungsmaßnahmen entwickelt:

| Prozess                            | Verfahren  |
|------------------------------------|--|
| Anforderungsanalyse, Systemanalyse | Review (Spezifikationsdokumente)   |
| Design                             | Review (Dokumente), Usability Tests, Barrierefreiheit  |
| Implementierung                    | Prüfungen auf Compiler Ebene   |
| Implementierung                    | Statische Code-Analyse mit autom. Tools  |
| Implementierung                    | Manuelle Code Reviews und Code Inspektionen von z.B. kritischen Codeteilen                       |
| Implementierung                    | Erweiterte Reviews auf Basis voriger Fehler, Komplexität, Änderungshistorie, Risikoanalysen usw. |
| Implementierung                    | Unit Tests   |
| Implementierung                    | Systematische Tests durch Code Abdeckung   |
| Verifikation, Validierung          | Operationale Tests   |
| Verifikation, Validierung          | Regressionstests   |

|            |   |
|------------|---|
| Produktion | Systemtests mittels unterschiedlicher Profile<br>u.a. Lasttests |
|------------|---|

## Quality Gates in Softwareprojekten

Zitat:

Die einzige Maßnahme, die die Qualität einer Software sicherstellen kann ist die Einführung von Quality Gates. Alle anderen "Qualitätssicherungsmaßnahmen" wie ‚Pair Programming‘, Tests, Metriken etc. können die Qualität aufzeigen oder auch verbessern, aber niemals sicherstellen.

"Quality Gates sind Punkte im Ablauf eines Entwicklungsprojekts, bei denen anhand von im Voraus eindeutig bestimmten Qualitätskriterien über die Freigabe des nächsten Projektschrittes entschieden wird." - Jochen Peter Sondermann

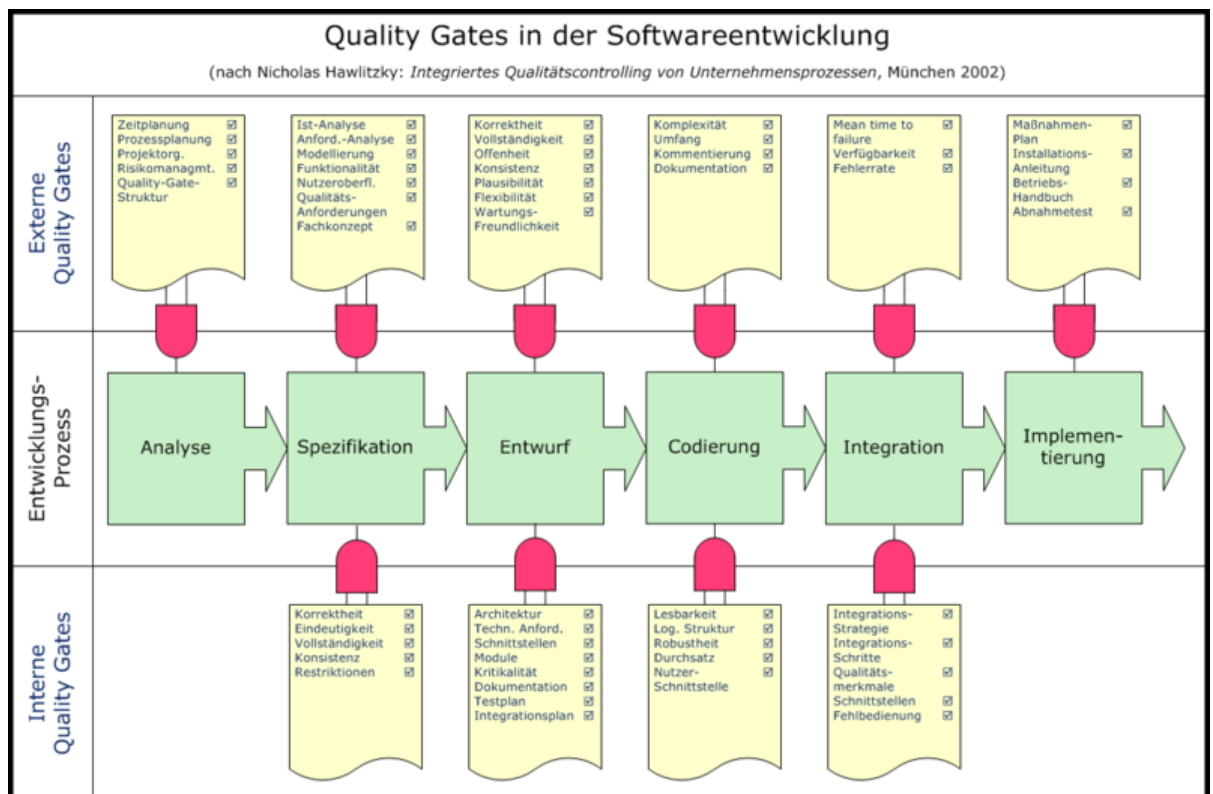


Abb. 2: *Qualitygates in der Softwareentwicklung* (Wikipedia)

State-of-the-art bei Quality Gates ist die Verwendung von Sonar. Sonar kann - basierend auf den meisten hier erwähnten Tools - für ein gesamtes Projekt Qualitätsberichte in einem Dashboard zusammenfassen und ebenfalls in den Build dermaßen integriert werden, dass der Build fehlschlägt, wenn bestimmte Quality Gates nicht erreicht wurden.

Im Folgenden gehen wir auf einzelne Quality Gates mit besonderem Bezug auf die Softwareentwicklung ein und stellen einige Kriterien zu deren Bewertung vor.

**Quality Gates für Architektur:** Die Qualität einer Architektur zu bewerten ist objektiv sehr schwierig. Es gibt nur wenige unstrittige Kriterien:

a) Die Architektur muss Zyklen frei sein

b) Die Architektur muss korrekt umgesetzt werden

**Quality Gates für Softwaredesign:** Problematisch ist, das es zwar viele allgemeine Vorgaben gibt, jedoch nur wenige Metriken. Eine davon ist die Erkennung von Codewiederholungen, da diese in der Objektorientierung immer als ein Zeichen für schlechtes Design angesehen werden.

Quality Gates für Implementierung: Als Metrik für dieses Quality Gate können z.B. Implementierungs-, Größen- und Komplexitätsmetriken herangezogen werden.

**Quality Gates für Unit-Tests:** Kenngröße für die Qualität von Unit-Tests ist deren Code-Coverage (Line- und Branch-Coverage sind die Aussagekräftigsten). Die Wartungsfreundlichkeit einer Software ließe sich am besten mit der Kennzahl "Wahrscheinlichkeit, dass zukünftige Codeänderungen durch Unit-Tests erkannt werden" beschreiben. Leider gibt es dafür (noch) keine geeigneten Tools (,mutation-testing‘ Tools gehen in diese Richtung, sind aber in der Praxis viel zu langsam). Neben der Coverage sollte natürlich auch überprüft werden, dass alle Unit-Tests durchlaufen werden und zumindest ein ‚Assert‘ Statement enthalten.

Als Ergebnis lässt sich festhalten, dass sich einige (bei weitem nicht alle) Qualitätskriterien in der Java Entwicklung automatisiert durch Quality Gates sicherstellen lassen. Die Nutzung der Technik der Quality Gates erhöht die Qualität und verringert gleichzeitig die Kosten für Wartung und Weiterentwicklung.

#### **Referenzen:**

Abb. 2: <http://de.wikipedia.org/wiki/Datei:QualityGatesSoftwareentwicklung.png>

*Quality Management:* Ebert, C. and R. Dumke: Software Measurement. Springer, Heidelberg, New York, 2007

*Software Quality Management:*

[http://vector.com/portal/medien/vector\\_consulting/publications/Ebert\\_QualityManagement.pdf](http://vector.com/portal/medien/vector_consulting/publications/Ebert_QualityManagement.pdf)

*ISO 9001:* 2000. Quality management systems – Requirements. ISO, <http://www.iso.org> (2000)

*CMMI-Dev:* [http://cmmiinstitute.com/wp-content/uploads/2013/01/10tr033de\\_v11.pdf](http://cmmiinstitute.com/wp-content/uploads/2013/01/10tr033de_v11.pdf)

*Quality Gates:* [http://www.berner-](http://www.berner-mattner.com/cms/upload/pdf/presse/BernerMattner_Sonderdruck_ETR_QualityGates.pdf)

[mattner.com/cms/upload/pdf/presse/BernerMattner\\_Sonderdruck\\_ETR\\_QualityGates.pdf](http://www.berner-mattner.com/cms/upload/pdf/presse/BernerMattner_Sonderdruck_ETR_QualityGates.pdf)

Zitat Jochen Peter Sondermann aus Masing: [Handbuch Qualitätsmanagement](#)

#### **Kontaktadresse:**

Timo Hahn

Virtual 7 GmbH

Zeppelinstrasse 2

D-76185 Karlsruhe

Telefon: +49 (0) 721-61901759

Fax: +49 (0) 721-61901729

E-Mail [hahn@virtual7.de](mailto:hahn@virtual7.de)

Internet: [www.virtual7.de](http://www.virtual7.de)