

Martin Glinz

Software-Qualität – Ausgewählte Kapitel

Kapitel 1

Einführung



Universität Zürich
Institut für Informatik

1.1 Zu dieser Vorlesung



Voraussetzungen

- In der Vorlesung Software Engineering wird in den folgenden Kapiteln ein **Grundwissen in Software-Qualität** vermittelt:
 - Kapitel 2: Ziele und Qualität
 - Kapitel 7: Validierung und Verifikation
 - Kapitel 8: Testen von Software
 - Kapitel 9: Reviews
 - Kapitel 10: Messen von Software
 - Kapitel 11: Statische Analyse
 - Kapitel 19: Software-Qualitätsmanagement
 - Kapitel 20: Bewertung und Verbesserung von Prozessen und Qualität
- **Dieses Grundwissen wird vorausgesetzt**; alle Kapitel sind elektronisch verfügbar: <http://www.ifi.uzh.ch/serg/courses/hs07/se/>

Lernziele

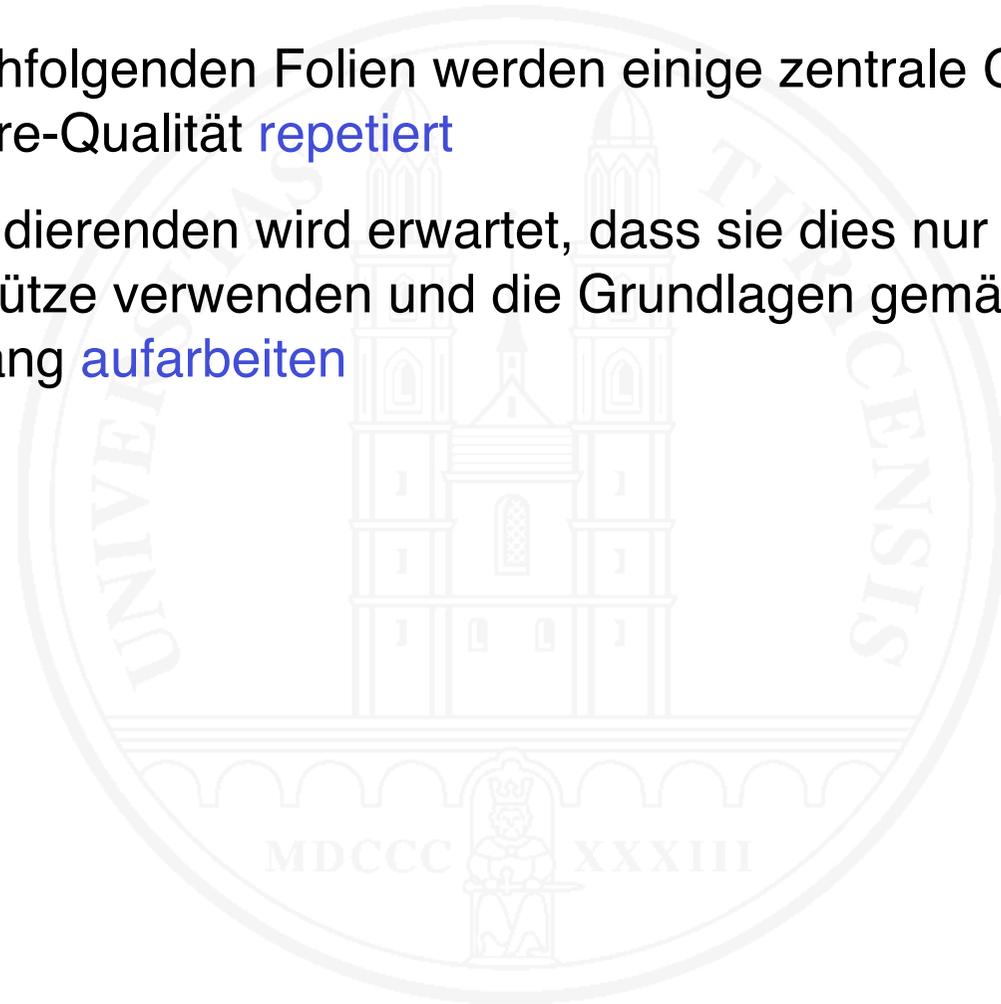
- Die Studierenden verfügen verbreitern und vertiefen ihr vorhandenes Grundwissen über Software-Qualität.
 - Sie **vertiefen** ihr Wissen über **Testen**, **Qualitätsmodelle**, **Produktqualität** und **Prozessqualität**
 - Sie erwerben ein vertieftes Wissen über die **Grundlagen modernen Qualitätsmanagements**
 - Sie lernen mit dem **Model Checking** ein weiteres, wichtiges Prüfverfahren kennen
 - Sie lernen **Debugging** als systematischen Prozess zur Bestimmung von Fehlerursachen kennen
 - Sie lernen mit **QFD** (Quality Function Deployment) ein weiteres bekanntes Qualitätsmanagement-Verfahren kennen
 - Sie lernen mit **CMMI** ein wichtiges Prozessbeurteilungs- und Prozessverbesserungsverfahren kennen

Vorlesungsplan (vorläufig)

01	18.2.08	Einführung, Model Checking I
02	25.2.08	Model Checking II
03	3.3.08	Qualität bei agiler und evolutionärer Entwicklung
04	10.3.08	Fortgeschrittene Testverfahren I
05	17.3.08	Fortgeschrittene Testverfahren II; Debugging I
	24.3.08	Keine Vorlesung (Osterferien)
06	31.3.08	Debugging II
07	7.4.08	Debugging III
	14.4.08	Keine Vorlesung (Sechseläuten)
08	21.4.08	Produktqualität I
09	28.4.08	Produktqualität II
10	5.5.08	Prozessqualität I
	12.5.08	Keine Vorlesung (Pflingstmontag)
11	19.5.08	Prozessqualität II
12	26.5.08	Zusammenfassung, Ausblick
	2.6.08	Klausur

1.2 Qualität – Grundlagen

- Auf den nachfolgenden Folien werden einige zentrale Grundlagen über Software-Qualität **repetiert**
- Von den Studierenden wird erwartet, dass sie dies nur als Gedankenstütze verwenden und die Grundlagen gemäss Folie 3 in vollem Umfang **aufarbeiten**



Der intuitive Qualitätsbegriff

Qualität

- **Hochwertige** Produkte oder Arbeitsweise
- **Langlebige** Produkte **hoher Güte**



Der industrielle Qualitätsbegriff

Qualität (quality) – der Grad, in dem ein Satz **inhärenter Merkmale** **Anforderungen** erfüllt. [ISO 9000:2000]

Inhärentes Merkmal (inherent characteristic) – eine **kennzeichnende Eigenschaft** einer Einheit (Produkt, Dienstleistung, Prozess, System, Person, Organisation, etc.), welche diese aus sich selbst heraus hat und die ihr nicht explizit zugeordnet ist.

Beispiel:

Ein Medikament

- **Inhärentes Merkmal:**
Zusammensetzung
- **Explizit zugeordnet:**
Preis

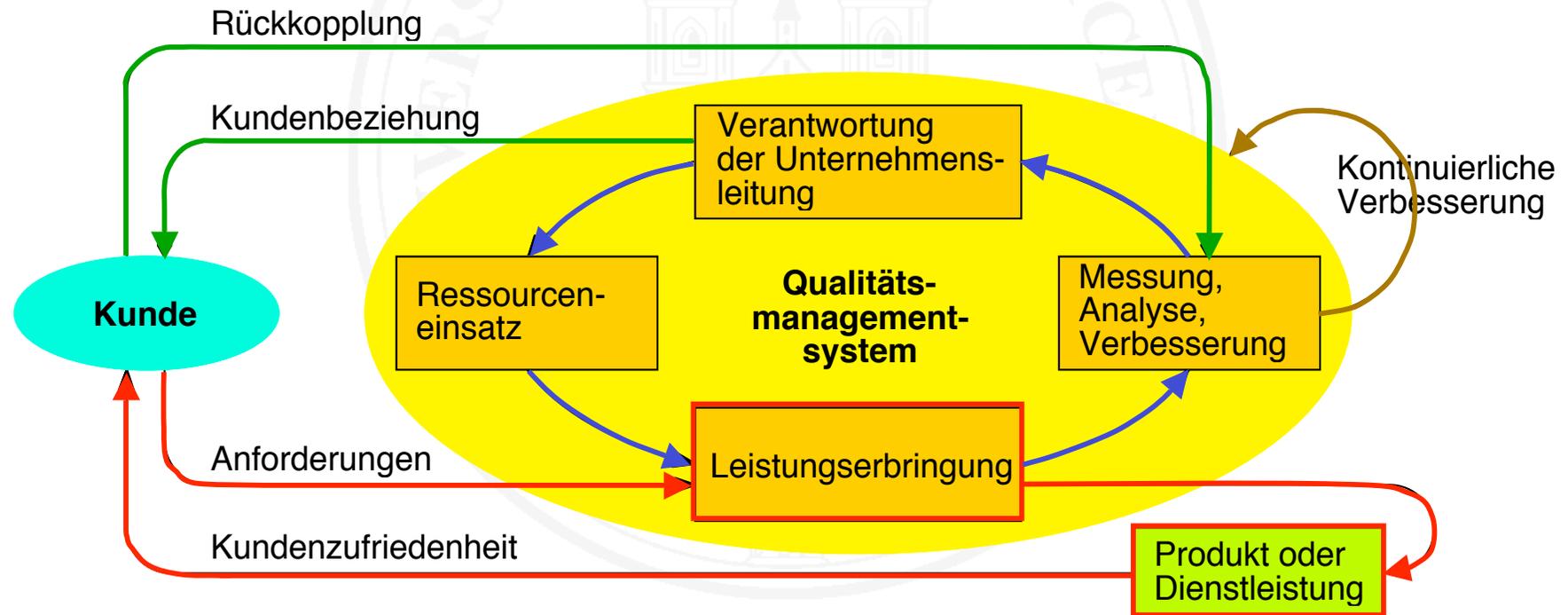


Bemerkungen

- Qualität ist **Zielerfüllung**. Die Ziele (Anforderungen) können **explizit festgelegt** oder **implizit** durch gemeinsame Vorstellungen der Beteiligten **gegeben** sein.
- Eine Auffassung von Qualität als reine **Zweckeignung** oder **Kundenzufriedenheit** greift zu kurz. Sie erfasst den Qualitätsbegriff nicht in seiner Gesamtheit.
- Qualität ist kein absolutes Maß für die **Güte** einer Einheit.
- Qualität entsteht nicht von selbst. Sie muss **definiert** und **geschaffen** werden.
- Qualität bezieht sich sowohl auf **Produkte** als auch auf **Prozesse** und **Projekte**.

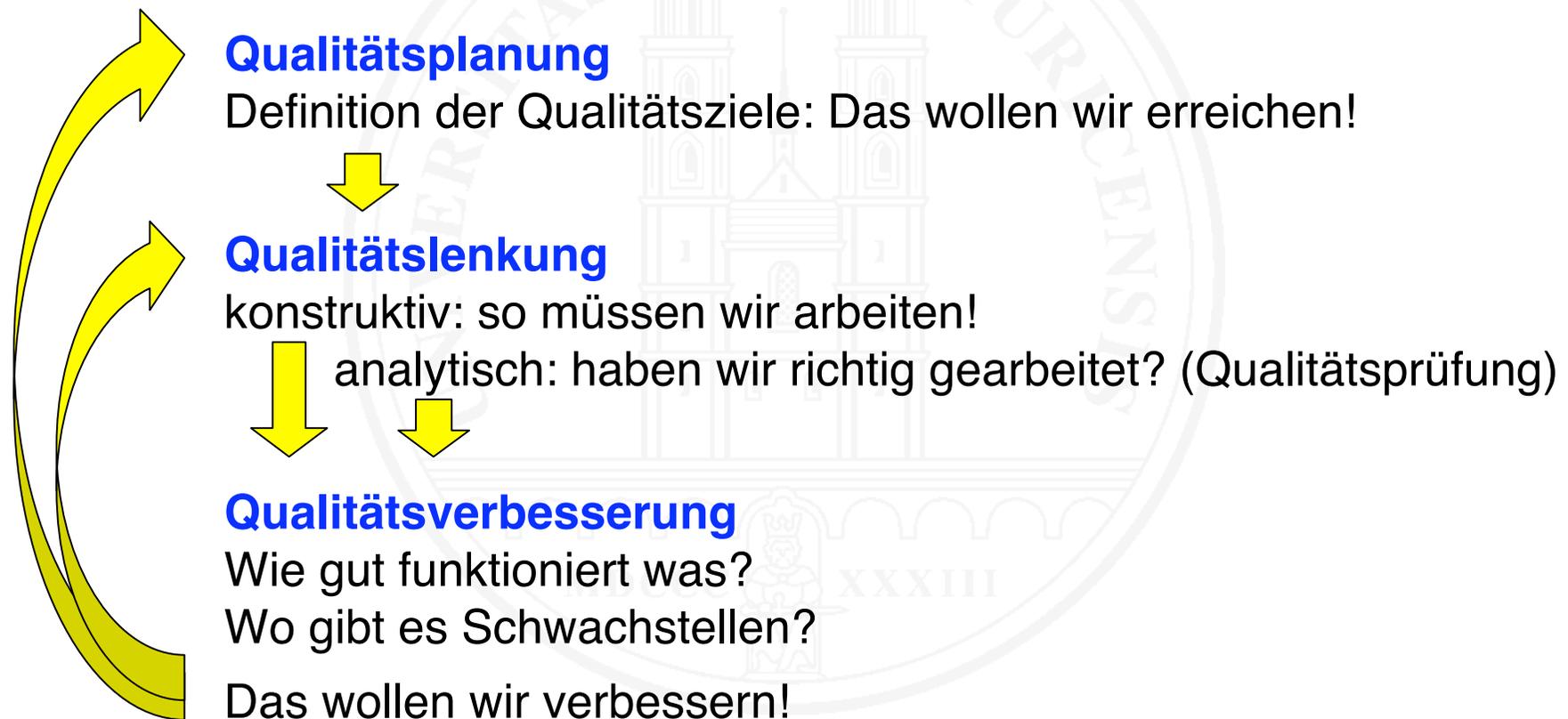
Das Qualitätsmanagementsystem – Prinzipien

- Orientiert an
 - Selbstverantwortung aller Beteiligten
 - Kundenzufriedenheit
- Prozessorientiert, systemischer Ansatz zur Realisierung



Verfahren des Qualitätsmanagements: Überblick

generell: planen – lenken – verbessern



Qualitätsplanung

Qualitätsplanung (quality planning) – Teil des Qualitätsmanagements, der auf das Festlegen der **Qualitätsziele** und der notwendigen **Ausführungsprozesse** sowie der zugehörigen **Ressourcen** zur Erfüllung der Qualitätsziele gerichtet ist. (ISO 9000:2000)

- ⇒ **Qualitätsziele** bestimmen
- ⇒ Kein Qualitätsmanagement **ohne** eine saubere, quantifizierte **Spezifikation der Anforderungen**.
- **Qualitätsplanung heißt**
 - **Im Allgemeinen:** Aufbau und Dokumentation des QM-Systems, allgemeine Qualitätsziele
 - **Im Speziellen:** Festlegung der Qualitätsziele für individuelle Projekte

Konstruktive Maßnahmen

- Fehlerverhindernde / fehlervermeidende Prozesse definieren
- Prüf- und Korrekturverfahren in die Prozesse integrieren
- Prüfergebnisse zur Verbesserung des Prozesses verwenden
- Eine systematische, ingenieurmäßige Vorgehensweise, welche die Erreichung gegebener Qualitätsanforderungen **garantiert**, gibt es für **Software** bis heute **nicht**.
- Konstruktive Maßnahmen werden so weit als möglich eingesetzt, um das **generelle Qualitätsniveau** zu heben
- **Rigorese Qualitätsprüfung** (und Behebung der festgestellten Mängel) während aller Phasen der Entwicklung ist heute **das Mittel** zur Sicherstellung der konkreten Qualitätsanforderungen an Software.

Analytische Maßnahmen: Qualitätsprüfung

- Prüfung der Produkte
 - Statische Prüfung
 - Review
 - Statische Analyse
 - Formale Programmverifikation
 - Model Checking
 - Dynamische Prüfung
 - Test
 - Simulation
 - Prototypen
- Prüfung der Prozesse
 - Audits
 - Prozessbeurteilung

Qualitätsverbesserung

Qualitätsverbesserung (quality improvement) – Teil des Qualitätsmanagements, der auf die **Erhöhung der Fähigkeit** zur **Erfüllung** von **Qualitätsanforderungen** gerichtet ist. (ISO 9000:2000)

- **Behebung** der bei der **Produktprüfung** gefundenen **Qualitätsmängel**
 - Notwendig zur Erreichung von Produktqualität
 - Häufig jedoch nur Symptombekämpfung
 - **Modifikationen** im **Entwicklungsprozess** und im **Qualitätsmanagementsystem** aufgrund von
 - Auswertung von Fehlerursachen
 - Resultaten von Audits
 - Messungen
- ⇒ **Prozessverbesserung**

Schaffung von Vertrauen: Qualitätssicherung

Qualitätssicherung (quality assurance) – Teil des Qualitätsmanagements, der auf das **Erzeugen von Vertrauen** darauf gerichtet ist, dass **Qualitätsanforderungen erfüllt** werden. (ISO 9000:2000)

- Regelmäßige **Überprüfung** der Wirksamkeit des Qualitätsmanagementsystems durch Experten in **Audits**
- Publikation von qualitätsrelevanten **Messgrößen**
- **Dokumentation** (und teilweise Offenlegung) der Prozesse und Qualitätsverfahren
- **Zertifizierung** des Qualitätsmanagementsystems
- Aktionsprogramme zur **Verbesserung** der Prozesse für die Entwicklung, Pflege und Verwaltung von Software

Literatur

Deming, W.E. (1986). *Out of the Crisis*. Cambridge, Mass.: M.I.T. Press.

Frühauf, K., J. Ludewig, H. Sandmayr (2000). *Software-Projektmanagement und -Qualitätssicherung*. 3. Auflage. Zürich: vdf Hochschulverlag.

Glinz, M. (2005). *Software Engineering*. Vorlesungsskript, Universität Zürich.

ISO 9000:2000. *Qualitätsmanagementsysteme – Grundlagen und Begriffe*. Deutsche Fassung der Europäischen Norm EN ISO 9000 (deutsch/englisch/französisch)

ISO 9001:2000. *Qualitätsmanagementsysteme – Anforderungen*. Deutsche Fassung der Europäischen Norm EN ISO 9001 (deutsch/englisch/französisch)

ISO 9004:2000. *Qualitätsmanagementsysteme – Leitfaden zur Leistungsverbesserung*. Deutsche Fassung der Europäischen Norm EN ISO 9004 (deutsch/englisch/französisch).

Liggemeyer, P. (2002). *Software-Qualität: Testen, Analysieren und Verifizieren von Software*. Berlin: Spektrum Akademischer Verlag.