

Softwarequalität und -test

4. Vorlesung

„Qualitätsmanagement und
Qualitätssicherung in der
Softwareentwicklung“

www.beuth-hochschule.de

Dipl.-Inform. Thomas Ziemer

Softwarequalität und -test

QM und QS in der Softwareentwicklung

„Warum haben Sie eigentlich nie Zeit, eine Sache richtig zu machen, aber immer Zeit, Sie noch einmal zu machen?“

(Gerald Marvin Weinberg, „Systemdenken und Softwarequalität“)

„Qualität heißt nicht nur, genau das zu liefern, was der Kunde wollte, sondern auch das, was er gewollt hätte, wenn er richtig beraten worden wäre.“

(Soo Chul Bang, Project Executive for IBM Global Service)

Warum scheitern
Softwareprojekte?

Softwarequalität und -test

QM und QS in der Softwareentwicklung

Unrealistische Zeitschiene (oder der Endtermin verschiebt sich nach vorne)	(55 %)
Externe Einflüsse	(50 %)
Fehleinschätzung von Projektrisiken	(50 %)
Unrealistische Ressourcenplanung (Urlaubsplanung vergessen, Krankheiten etc.)	(46 %)
Unternehmensinterne Widerstände	(46 %)
Personelle Fehlbesetzung (keine entsprechenden Voraussetzungen, keine Schulungen, keine Unterstützung)	(42 %)
Unklare Zieldefinition (oder das Ziel ändert sich während der Projektarbeit)	(42 %)
Scheitern des Projekts	(18 %)

(*Chefbüro – Das Magazin für Führungskräfte*, 10/98)

Softwarequalität und -test

QM und QS in der Softwareentwicklung

Was ist Qualität? (Fortsetzung)

1. Firmen stehen unter Druck, die **Kosten möglichst gering** zu halten und **schnell zu liefern**.
2. Softwareprojekte waren in der Vergangenheit **einfacher und nicht so umfangreich**.
3. **Scheinbar bessere Software-Engineering-Verfahren** und –Tools erübrigen systematische Qualitätssicherung.
4. **Hohe Kosten** für die Einrichtung eines Qualitätsmanagementsystems.

—

Softwarequalität und -test

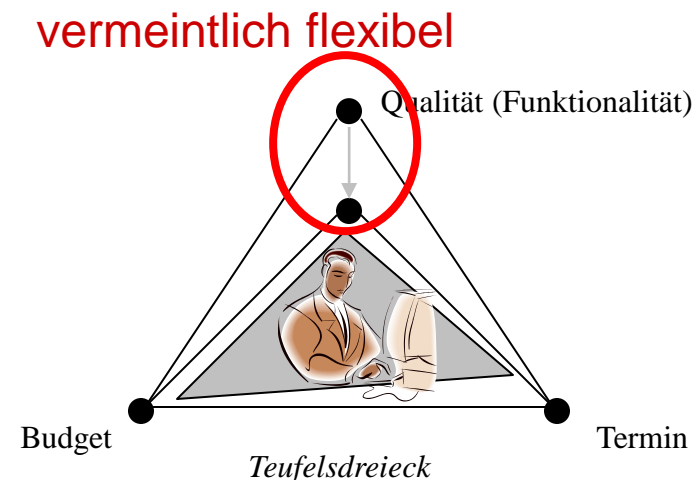
QM und QS in der Softwareentwicklung

Das Teufelsdreieck muss optimiert werden!

Ein Projekt bewegt sich im Spannungsfeld von **Qualität** (bzw. **Funktionalität**), **Budget** und **Termin**.

Richtig eingesetzt, können die Erkenntnisse dieses Teufelsdreiecks den Projektmanager bei seiner Arbeit unterstützen. **Werden jedoch nur zwei der Parameter im Auge behalten, so ist der dritte kaum zu kontrollieren.**

Die Vorgaben wurden in der Regel gestellt, ohne dass zuvor eine **Projektplanung** durchgeführt wurde.



Wartungsprobleme

Softwarequalität und -test

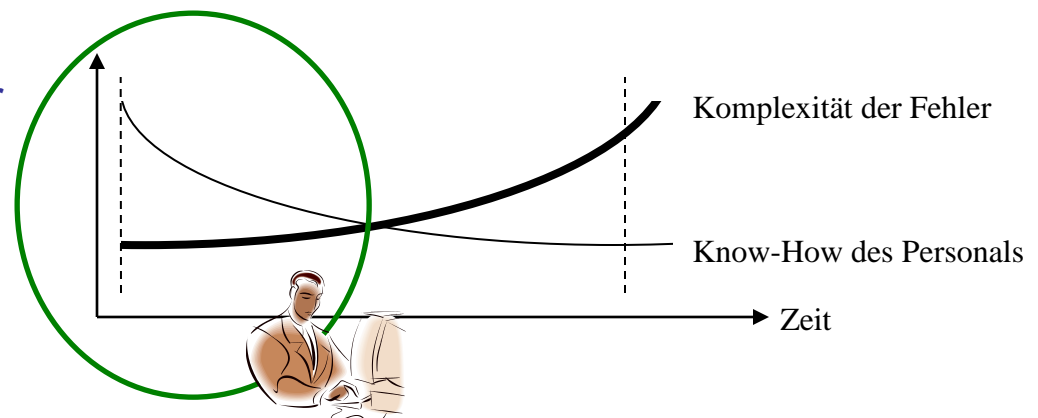
QM und QS in der Softwareentwicklung

Software-Wartungsprobleme

Wenn ein Softwaresystem in die Produktion eingeführt wird, sind die auftretenden Fehler in der Regel **von ihrer Komplexität her sehr gering**. Dagegen ist das Know-How des Wartungspersonals **sehr hoch**.

Es werden also leicht zu findende Fehler mit einem großen Know-How der Mitarbeiter behoben.

Konsequenz...

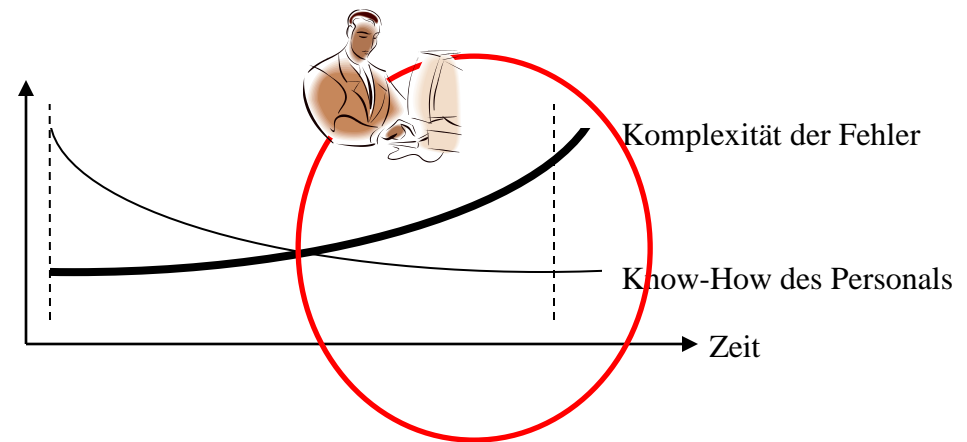


Softwarequalität und -test

QM und QS in der Softwareentwicklung

Software-Wartungsprobleme (Fortsetzung)

Dagegen wird zu einem späteren Zeitpunkt ein **sehr schwieriger Fehler** mit einem **niedrigen Know-How** der Mitarbeiter gesucht. Denn nach einer längeren Zeit verringert sich das Know-How über das Software-system, da die Mitarbeiter sich nicht mehr intensiv mit dem System auseinandergesetzt haben und mit anderen Arbeiten beauftragt wurden.

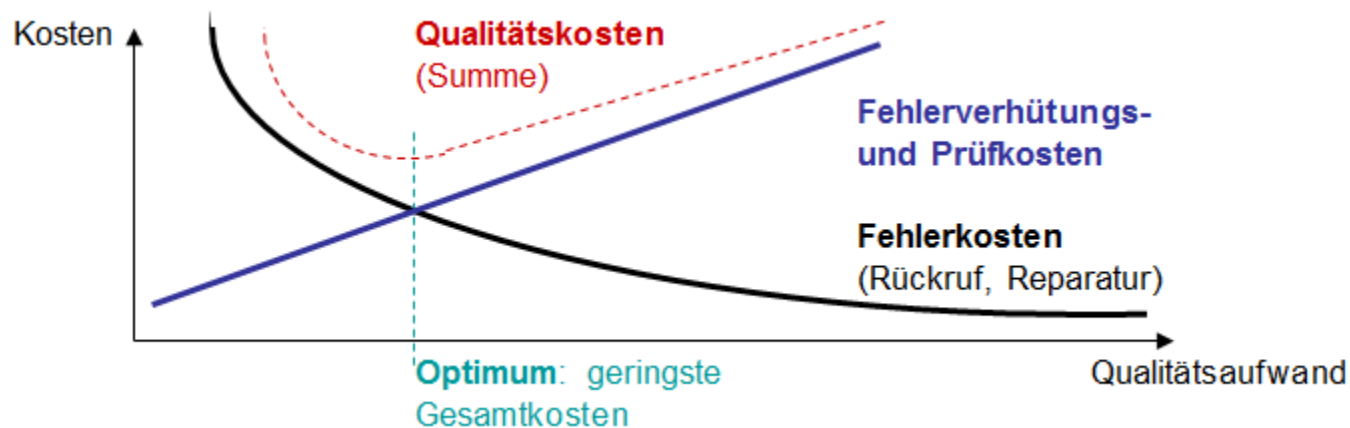


Softwarequalität und -test

QM und QS in der Softwareentwicklung

Kosten und Nutzen der Softwarequalität

Qualitätsmaßnahmen müssen sich lohnen. Sie werden eingesetzt, um die eigenen Kosten zu reduzieren. Zu den Kosten zählen sowohl die **Fehlerverhütungskosten** (was man für Qualitätsmaßnahmen ausgibt) als auch die **Fehlerkosten** (was man für Fehlerbeseitigung, Haftung oder Entschädigung ausgeben muss, wenn Fehler aufgetreten sind).



Softwarequalität und -test

QM und QS in der Softwareentwicklung

Softwarequalität (Fortsetzung)

Def.: „Qualität ist der Grad, in dem ein Satz inhärenter Merkmale die Anforderungen erfüllt.“

Softwarequalität erscheint zunächst einmal kein exakt definierter Begriff. Einigkeit herrscht jedoch darüber, dass er nicht nur die

Qualität eines Programms

als solches umfasst, sondern auch die

Qualität seiner Dokumentation sowie die

Qualität der Benutzeranweisungen,

die für den Erfolg eines Softwareprodukts ebenfalls von erheblicher Bedeutung sind.

Softwarequalität und -test

QM und QS in der Softwareentwicklung



geek & poke

SOMETIMES IT'S JUST SIMPLE

Softwarequalität und -test

QM und QS in der Softwareentwicklung

Softwarequalität (Fortsetzung)

Def.: „[...] *die Gesamtheit der Merkmale und Merkmalswerte eines Softwareprodukts, die sich auf die Eignung beziehen, festgelegte oder vorausgesetzte Erfordernisse zu erfüllen.*“ (DIN ISO 9126)

Unter dem Begriff der Qualität eines **Programms** ist dabei weit mehr zu verstehen als nur seine **Korrektheit**.

Es beinhaltet auch Eigenschaften wie

**Robustheit,
Übertragbarkeit und
Wartungsfreundlichkeit.**

Nicht zuletzt ist auch die **Qualität der Benutzerschnittstelle**, die **Benutzerfreundlichkeit**, einzuordnen.

Softwarequalität und -test

QM und QS in der Softwareentwicklung

Softwarequalität (Fortsetzung)

Die optimale Qualität ist dann erreicht, wenn das Ergebnis die Anforderungen genau erfüllt.

Ein *besseres Ergebnis* kommt einer *Übererfüllung der Anforderungen* gleich. Sie ist zu vermeiden, da diese in der Regel mehr Geld kostet.



Qualitätsmanagement

Die drei organisatorischen Einbettungen

Softwarequalität und -test

QM und QS in der Softwareentwicklung

Qualitätsmanagement

Def.: „**Alle Tätigkeiten der Gesamtführungsaufgabe, welche die Qualitätspolitik, Ziele und Verantwortlichkeiten festlegen, sowie diese durch Mittel wie Qualitätsplanung, -lenkung, -sicherung und –verbesserung im Rahmen des Qualitätsmanagementsystems verwirklichen.**“ (ISO 8402)

Qualitätsmanagement sind **strategische, projektübergreifende**, aufeinander abgestimmte Tätigkeiten zum Leiten und Lenken einer Organisation (eines Unternehmens, einer Firma) bezüglich Qualität, die in einem Qualitätsmanagementsystem, beispielsweise in einem **QM-Handbuch**, festgelegt werden.

Softwarequalität und -test

QM und QS in der Softwareentwicklung

Qualitätsmanagement (Fortsetzung)

▶ Traditionelles Qualitätsmanagement:

- erst das fertige Produkt betrachtet: Test
- **Nachteile:** hohe Kosten für Tests, hohe Kosten für Fehlerbeseitigung
→ Qualität zu spät betrachtet

▶ Modernes Qualitätsmanagement:

- Qualität des Produkts indirekt über Qualität des Entwicklungsprozesses verbessern

Aufbauorganisation

Erste organisatorische Einbettung

Softwarequalität und -test

QM und QS in der Softwareentwicklung

Qualitätsmanagement (Fortsetzung)

Aufbauorganisation der Softwarequalität (erste organisatorische Einbettung)

Softwarequalität muss in der Organisation einer Firma verankert sein
= **Softwarequalität muss aus Überzeugung gelebt werden!**

Dazu gehören die **Aufbauorganisation** und ein **Qualitätsmanagementsystem**.
In der Aufbauorganisation sind die **Hierarchieebenen** und **Leitungsfunktionen** geregelt, also wie die Softwarequalität in die **statische Struktur der Firma** eingebettet ist.

Softwarequalität und -test

QM und QS in der Softwareentwicklung

Qualitätsmanagement (Fortsetzung)

Aufbauorganisation der Softwarequalität (erste organisatorische Einbettung)

Damit gute Softwarequalität auch gegen mögliche Widerstände durchgesetzt werden kann, sind zwei Regeln zu beachten:

Qualität ist auf allen Unternehmensebenen vertreten und bildet eine eigene (Schatten-) Hierarchie, parallel zu der Leitungshierarchie der Softwareentwicklung.

Auf jeder Ebene agieren die „Qualitätsleute“ als **Stabsstellen der jeweiligen Führungskraft auf dieser Ebene**. Sie unterstützen die Führungskraft, ohne selbst Leitungsaufgaben wahrzunehmen.

—

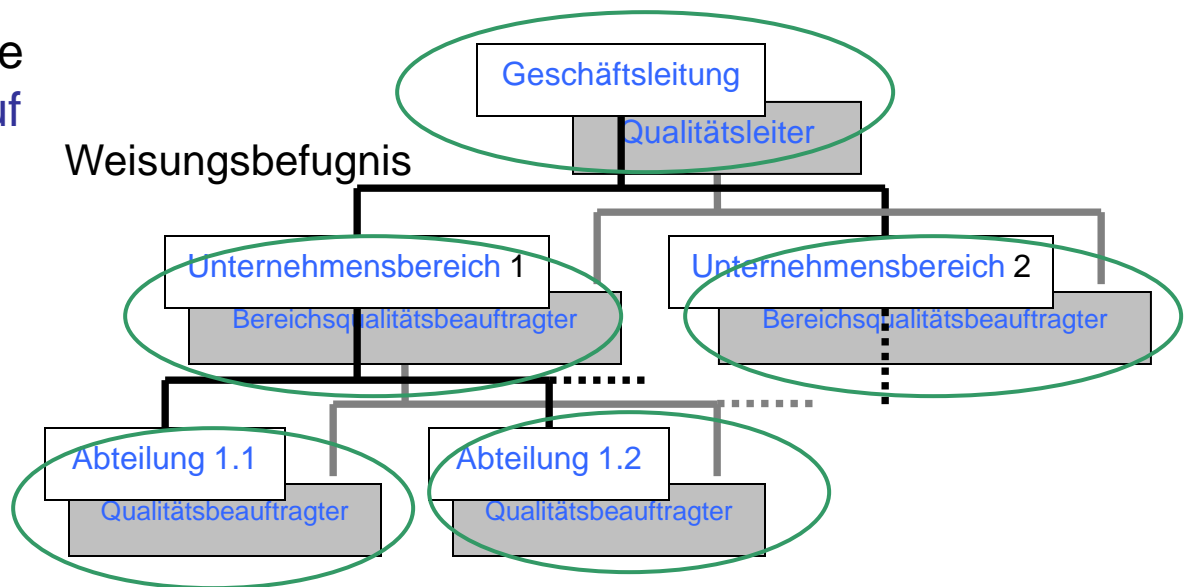
Softwarequalität und -test

QM und QS in der Softwareentwicklung

Qualitätsmanagement (Fortsetzung)

Aufbauorganisation der Softwarequalität

Durch diese Regeln sind die beiden Hierarchien zwar **auf allen Ebenen miteinander verbunden**, aber Leitungsverantwortliche können die Qualitätsfachleute **nicht zwingen**, ihre Qualitätsansprüche zu senken, denn **Weisungsbefugnis besteht nur entlang der dicken (schwarzen bzw. grauen) Linien**.



Ablauforganisation

Zweite organisatorische Einbettung

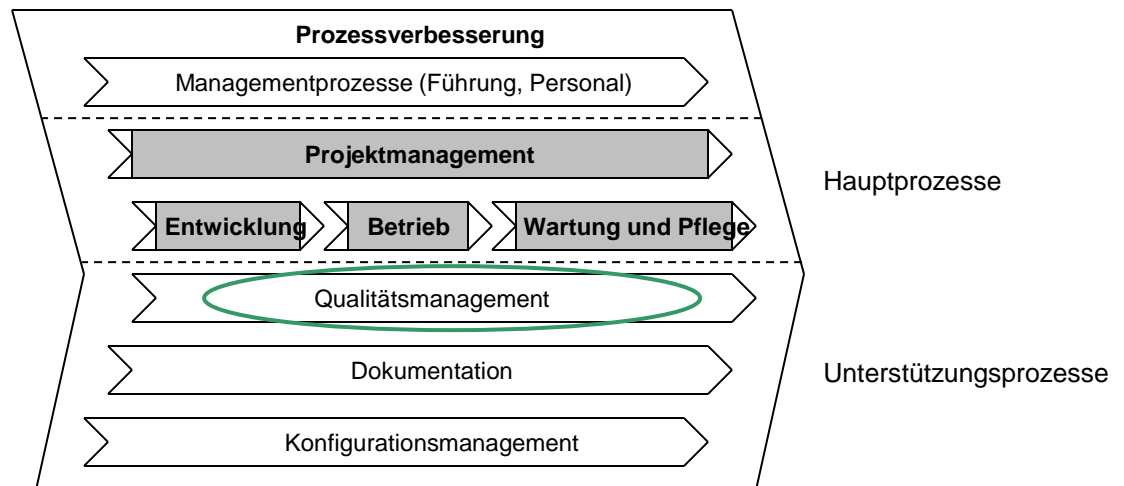
Softwarequalität und -test

QM und QS in der Softwareentwicklung

Qualitätsmanagement (Fortsetzung)

Ablauforganisation der Softwarequalität (zweite organisatorische Einbettung)

Bei der Ablauforganisation geht es um die **dynamischen Abläufe und Prozesse**. Aus der Sicht der Qualität ist daran interessant, dass das Qualitätsmanagement einfach einen **Unterstützungsprozess** für die Entwicklung bildet.



Qualitätsmanagement- system

Dritte organisatorische Einbettung

Softwarequalität und -test

QM und QS in der Softwareentwicklung

Qualitätsmanagement (Fortsetzung)

Qualitätsmanagementsystem (dritte organisatorische Einbettung)

Def.: „...zur Verwirklichung des Qualitätsmanagements erforderliche Organisationsstruktur, Verfahren, Prozesse und Mittel.“ (ISO 8402)

Die dritte organisatorische Einbettung ist das **Qualitätsmanagementsystem** (QMS). Es gehören mindestens sechs Felder zum QMS:

Softwarequalität und -test

QM und QS in der Softwareentwicklung

Qualitätsmanagement (Fortsetzung)

Qualitätsmanagementsystem (dritte organisatorische Einbettung)

Leiten und Lenken bezüglich Qualität umfasst üblicherweise das Festlegen der

Qualitätspolitik und der

Qualitätsziele, die

Qualitätsplanung, die

Qualitätslenkung, die

Qualitätssicherung und die

Qualitätsverbesserung.

Leiten und Lenken...

Softwarequalität und -test

QM und QS in der Softwareentwicklung

Qualitätsmanagement (Fortsetzung)

Qualitätspolitik

Qualitätspolitisch sind alle übergeordnete Absichten und die Ausrichtung einer Organisation zur Qualität, wie sie von der obersten Leitung formell ausgedrückt werden, sozusagen die Einstellung und Grundsätze der Firma zur Softwarequalität:

Senkung der Software-Entwicklungs- und -wartungskosten, beispielsweise durch Senkung des Personalaufwands für die Softwarewartung.

Qualitätsverbesserung des Softwareprodukts.

Transparenz des Projektfortschritts in dem Sinne, dass die zuvor definierte Qualität eines Teilprodukts und damit seine Fertigstellung oder Nichtfertigstellung feststellbar wird.

Softwarequalität und -test

QM und QS in der Softwareentwicklung

Qualitätsmanagement (Fortsetzung)

Qualitätsorganisation (die „anderen“ organisatorischen Einbettungen)

Qualitätsorganisatorisch sind die soeben beschriebene **Aufbau-** und **Ablauforganisation** zur Einbettung aller Softwarequalitätsaktivitäten in das Unternehmen.

Qualitätsziele

Qualitätsziele sind die **angestrebten, zu erreichenden Ziele** bezüglich Qualität. Ein Qualitätsziel oder mehrere davon bilden zusammen eine **Qualitätsanforderung**.

Softwarequalität und -test

QM und QS in der Softwareentwicklung

Qualitätsmanagement (Fortsetzung)

Qualitätsplanung

Qualitätsplanung ist der Teil des Qualitätsmanagements, der auf das Festlegen der für die Qualitätsziele notwendigen **Ausführungsprozesse** sowie der zugehörigen **Ressourcen** zur Erfüllung der Qualitätsziele gerichtet ist.

Qualitätslenkung

ist nach *EN ISO 9000:2005 Punkt 3.2.10* der Teil des Qualitätsmanagements, der auf die **Erfüllung der Qualitätsziele** gerichtet ist.

Qualitätslenkung umfasst dabei **Arbeitstechniken und Tätigkeiten sowohl zur Überwachung eines Prozesses als auch zur Beseitigung von Ursachen nicht zufriedenstellender Ergebnisse**. Maßnahmen der Qualitätslenkung und Maßnahmen der **Qualitätssicherung** stehen zueinander in Wechselwirkung.

Softwarequalität und -test

QM und QS in der Softwareentwicklung

Qualitätsmanagement (Fortsetzung)

Qualitätsverbesserung

Qualitätsverbesserung ist der Teil des Qualitätsmanagements, der auf die **Erhöhung der Fähigkeiten zur Erfüllung der Qualitätsziele** gerichtet ist.

Qualitätsmaßnahmen

zur Qualitätsverbesserung sind der Teil des Qualitätsmanagements, der auf die **Erhöhung der Fähigkeiten zur Erfüllung der Qualitätsziele** gerichtet ist. Es handelt sich um konkrete Aktivitäten zur Förderung der Qualität. Im Zentrum stehen **Prüfungen** (dabei besonders **Testen** und **Reviews**), aber auch **Fortbildungsmaßnahmen** gehören dazu.

Qualitätssicherungsmodell

Grundsatz

Softwarequalität und -test

QM und QS in der Softwareentwicklung

Qualitätsmanagement (Fortsetzung)

Qualitätssicherungsmodell

Die Softwarequalität gehört zu den **nicht-funktionalen Anforderungen** (*non-functional requirements*) an ein Softwareprodukt.

Funktionale Anforderungen sind die Funktionen und Abläufe sowie die dazu gehörenden Daten, die zur Problemlösung notwendig sind und vom Auftraggeber festgelegt werden.

Das Qualitätssicherungsmodell ist das **fest- und vorgelegte Vorgehensmodell**, nach dem sämtliche qualitätsrelevanten Prozesse und Verfahren eines Unternehmens in dokumentierten QM-Handbuch ablaufen.

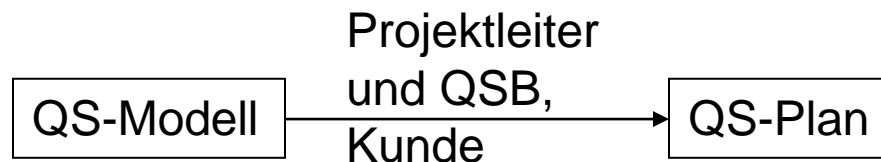
Softwarequalität und -test

QM und QS in der Softwareentwicklung

Qualitätsmanagement (Fortsetzung)

Qualitätssicherungsmodell

Es wird im Qualitätsmanagement (nur einmal!) in Form des **QM-Handbuchs** mit allen **konstruktiven und analytischen Qualitätssicherungsmaßnahmen** definiert, und **diejenigen Aspekte, die auf das konkrete Projekt zutreffen**, werden in der Qualitätssicherung vom Projektleiter gemeinsam mit dem Kunden durch entsprechende Aktivitäten erfüllt.



Softwarequalität und -test

QM und QS in der Softwareentwicklung

Qualitätsmanagement (Fortsetzung)

Qualitätssicherungsmodell für objektorientierte Systeme

Der **objektorientierte Ansatz** führt in der Regel dazu, dass Produkte mit Eigenschaften entstehen, die sich stark von solchen Produkten **unterscheiden**, die unter **prozeduralen Ansätzen** erzeugt wurden:

Wiederverwendbarkeit

...den Entwurf so gestalten, dass einzelne Entwurfskomponenten **einfach wiederverwendet** werden können.

—

Softwarequalität und -test

QM und QS in der Softwareentwicklung

Qualitätsmanagement (Fortsetzung)

Qualitätssicherungsmodell für objektorientierte Systeme

Geringe Komplexität der Anwendung

Um die **Vorteile der objektorientierten Softwareentwicklung** zu gewährleisten, müssen die Objekte und Klassen einer Anwendung eine **möglichst geringe Komplexität** aufweisen.

Diese Eigenschaft stellt einen wichtigen Faktor für die Erreichung der oben genannten Qualitätsmerkmale dar und setzt sich bei objektorientierten Anwendungen aus folgenden Einzelkriterien zusammen:

Softwarequalität und -test

QM und QS in der Softwareentwicklung

Qualitätsmanagement (Fortsetzung)

Qualitätssicherungsmodell für objektorientierte Systeme

Kopplung

...ist die Kommunikation mittels gesendeter Nachrichten zwischen verschiedenen Objekten, die nicht über eine Vererbungsbeziehung miteinander verbunden sind.

Je mehr ein Objekt mit anderen kommuniziert, desto schwieriger wird es, seine Funktion und Wirkungsweise zu durchschauen. Um die Komplexität einer Anwendung zu reduzieren, sollte die Kopplung möglichst gering sein.

—

Softwarequalität und -test

QM und QS in der Softwareentwicklung

Qualitätsmanagement (Fortsetzung)

Qualitätssicherungsmodell für objektorientierte Systeme

Vererbung

...ermöglicht eine Benutzung der vorhandenen Funktionalität in anderen Klassen des Gesamtsystems. Eine **tiefe Vererbungshierarchie** verschlechtert jedoch die Verständlichkeit und Wartbarkeit der in ihr enthaltenen Klassen. Jede Änderung in einer Oberklasse wird sich in allen Unterklassen sofort aus.

—

Softwarequalität und -test

QM und QS in der Softwareentwicklung

Qualitätsmanagement (Fortsetzung)

Qualitätssicherungsmodell für objektorientierte Systeme

Volumen

...ist ein wichtiges Komplexitätsmerkmal eines Objekts. Im Normalfall sind **große Objekte komplizierter und schwieriger zu verstehen als kleinere**. Große Objekte können vielfach durch Zerlegung in mehrere kleine, aber in sich geschlossene Einheiten an Komplexität verlieren. Man kann davon ausgehen, dass ein großes Objekt speziell für eine bestimmte Softwareanwendung entwickelt wurde und es dadurch kaum in anderen Programmen Wiederverwendung finden wird. Das Volumen eines Objekts sollte demnach möglichst gering sein.

—

Qualitätssicherungsmodell

QM-Handbuch
(Erster Meilenstein)

Softwarequalität und -test

QM und QS in der Softwareentwicklung

Qualitätsmanagement (Fortsetzung)

QM-Handbuch

Das QM-Handbuch gibt allen Mitarbeitern im Unternehmen in Inhalt und Struktur eines so genannten **Qualitätssicherungsmodells** vor, **auf welchem Wege** sie die Qualität des Entwicklungsprozesses selbst, sowie die Qualität aller Projektergebnisse gestalten können.

Im QM-Handbuch wird alles festgelegt, was **unternehmensweit für Softwarequalität und -qualitätssicherung** gilt.

Es enthält Angaben über alle **Vorgaben, Standards, Aktivitäten**, die Qualität und Qualitätssicherung betreffen.

Softwarequalität und -test

QM und QS in der Softwareentwicklung

Qualitätsmanagement (Fortsetzung)

QM-Handbuch

Die strukturelle Ausgestaltung des QM-Handbuchs ist **grundsätzlich dem Unternehmen freigestellt**. Seine spätere **Anwendung** sowie die damit verbundenen **Aufgaben des Handbuchs** legen jedoch bestimmte Strukturen der inhaltlichen Gliederung besonders nahe:

Das QM-Handbuch wird sowohl **im Unternehmen verteilt** als auch im Rahmen der Akquisition oder Kundenpflege **an Externe** ausgegeben. Da im zweiten Fall ein Missbrauch nicht ausgeschlossen werden kann, muss die unerwünschte Offenlegung firmenspezifischen Know-hows verhindert werden.

—

Softwarequalität und -test

QM und QS in der Softwareentwicklung

Qualitätsmanagement (Fortsetzung)

QM-Handbuch

Das Handbuch hat die Funktion eines **Nachschlagewerkes**.

Das Handbuch ist ein lebendes Dokument. Im Rahmen der laufenden Pflege muss ein leichtes und organisatorisch sicheres **Austauschen von Elementen des Handbuchs** gewährleistet sein. Die Verteilung und der Austausch des Dokuments sind im allgemeinen mit hohen Kosten verbunden. Eine **elektronische Dokumentation** bietet hier erhebliche Kosten- und Zeitvorteile.

—

Softwarequalität und -test

QM und QS in der Softwareentwicklung

Qualitätsmanagement (Fortsetzung)

QM-Handbuch

Das Handbuch ist als Sammlung einzelner Dokumente (häufig *dreiteilig*) aufgebaut:

Der erste Teil macht Angaben zur Organisation, zum Gebrauch, zur Herausgabe und zur Pflege des Handbuchs.

Der zweite Teil enthält Ausführungen, WAS Inhalt des QM ist

1. Zielsetzung
2. Anwendungsbereiche
3. Verantwortlichkeiten (und Zuständigkeiten, Befugnisse, Pflichten etc.)

Softwarequalität und -test

QM und QS in der Softwareentwicklung

Qualitätsmanagement (Fortsetzung)

QM-Handbuch

All das sind so genannte QM-Elemente (gemäß *ISO 9001:1994*)

- Verantwortung der Leitung
- Qualitätsmanagementsystem
- Vertragsprüfung
- Designlenkung
- Lenkung von Dokumenten und Daten
- Beschaffung
- Lenkung der vom Kunden beigestellten Produkte
- Prozesslenkung
- Interne Qualitätsaudits
- Korrektur- und Vorbeugungsmaßnahmen
- Schulung

Softwarequalität und -test

QM und QS in der Softwareentwicklung

Qualitätsmanagement (Fortsetzung)

QM-Handbuch

Der dritte Teil organisiert die formalen Anlagen

1. Beschreibung der angewandten Verfahren und Methoden (Arbeitsanweisungen)
2. Belege, Formblätter, mitgeltende Dokumente, Auflistung der zitierten Normen und Richtlinien etc.

Softwarequalität und -test

QM und QS in der Softwareentwicklung

Qualitätsmanagement (Fortsetzung)

QM-Handbuch

Diese Dreiteilung vereinfacht die Herausgabe und Verteilung des Handbuchs:

Der erste und zweite Teil werden an **Führungskräfte des Unternehmens** und an **Externe** ausgegeben, bzw. zugänglich gemacht.

Die Dokumente des dritten Teils stehen jeweils **im benötigten Umfang den Mitarbeitern der von den Inhalten betroffenen Bereiche** zur Verfügung. Da in diesem Teil präzise Angaben zu Abläufen und technischen Verfahren gemacht werden, die auch **schützenswertes Know-how** enthalten können, werden diese Dokumente **nicht an Externe weitergegeben**.

Qualitätsmerkmale und Qualitäts*teil*merkmale

nach DIN ISO 9126
(größtenteils aus Wikipedia)

Softwarequalität und -test

QM und QS in der Softwareentwicklung

Qualitätsmanagement

Operationalisierung der Qualitätsmerkmale

Softwarequalität wird bei Softwareprodukten durch **Qualitätsmerkmale** („Aspekte“) beschrieben. Diese Merkmale werden dann in **Teilmerkmale** oder **-kriterien** heruntergebrochen.

Die Qualitätsmerkmale beschreiben im weitesten Sinne die **benutzerorientierte Sichtweise** (Fachseite bzw. IT-Seite).

Dagegen beschreiben die Teilmerkmale die **softwareorientierten Anforderungen**, also wie die benutzerorientierten Anforderungen in dem Softwareprodukt realisiert werden können.

Softwarequalität und -test

Qualitätsmerkmale

Was ist Softwarequalität?

Sechs Qualitätsmerkmale werden allgemein unterschieden. Diese können auf jede Art von Software angewandt werden:

Functional Requirements

1. **Funktionalität** (inwieweit besitzt das Produkt die geforderten Funktionen?)
 - **Richtigkeit** (liefern der *richtigen* oder *vereinbarten* Ergebnisse, z.B. der geforderten **Genauigkeit von Ergebnissen**)
 - **Interoperabilität** (Fähigkeit, mit **externen Systemen** *zusammenzuwirken*, und dabei *Standards* und *Vereinbarungen* einzuhalten [z.B. JDBC])
 - **Angemessenheit** (*Eignung* der Funktionen für die Aufgabe, z.B. *aufgabenorientiertes* Arrangement von Teilfunktionen)
 - **Ordnungsmäßigkeit** (Erfüllung anwenderspezifischer *Normen*, *Vereinbarungen*, gesetzlicher *Bestimmungen* und *Vorschriften* [z.B. DSGVO])

Softwarequalität und -test

Qualitätsmerkmale

Was ist Softwarequalität? (Fortsetzung)

Sechs Qualitätsmerkmale werden allgemein unterschieden. Diese können auf jede Art von Software angewandt werden:

1. **Funktionalität** (inwieweit besitzt das Produkt die geforderten Funktionen?)
 - **Sicherheit** (*beabsichtigten* oder *unbeabsichtigten/versehentlichen unberechtigten* Zugriff auf Daten und Programme verhindern)

Softwarequalität und -test

Qualitätsmerkmale

Was ist Softwarequalität? (Fortsetzung)

2. **Zuverlässigkeit** (Leistungsniveau über einen Zeitraum aufrechterhalten)
 - **Reife** (geringe *Versagenshäufigkeit* durch Fehlerzustände, „**Software reift beim Kunden**“)
 - **Fehlertoleranz** (spezifiziertes *Leistungsniveau* bei Softwarefehlern oder Nicht-Einhaltung ihrer spezifizierten Schnittstellen bewahren, **Zustandsänderung ablehnen**, Fehler automatisch *korrigieren*)
 - **Wiederherstellbarkeit** (bei einem Versagen das Leistungsniveau **wiederherstellen** und die direkt betroffenen Daten *wiedergewinnen*, unter Berücksichtigung der benötigten Zeit und des Aufwands; z.B. Recovery bei einem Datenbanksystem)

—

Softwarequalität und -test

Qualitätsmerkmale

Was ist Softwarequalität? (Fortsetzung)

3. **Benutzbarkeit** (Aufwand zum und Beurteilung beim Einsatz der Software)
 - **Verständlichkeit** (Schwierigkeit für den Benutzer, das *Konzept* und die Anwendung zu **verstehen**)
 - **Erlernbarkeit** (Schwierigkeit für den Benutzer, die *Bedienung* und die *Ein- und Ausgaben* der Softwareanwendung zu **erlernen**)
 - **Bedienbarkeit** (Schwierigkeit für den Benutzer, die Softwareanwendung zu **bedienen**)
 - **Attraktivität** („*Anziehungskraft*“ der Anwendung gegenüber dem Benutzer)

—

Softwarequalität und -test

Qualitätsmerkmale

Was ist Softwarequalität? (Fortsetzung)

4. **Effizienz** (Verhältnis zwischen Leistungsniveau und Betriebsmitteln)
 - **Zeitverhalten** (Antwort- und *Verarbeitungszeiten* sowie *Durchsatz* bei der Funktionsausführung)
 - **Verbrauchsverhalten** (Anzahl und Dauer der benötigten **Betriebsmittel** [Ressourcen] für die Erfüllung der Funktionen, z.B. CPU-Zeit, Festplattenzugriffe, Netzwerkauslastung)
 -

Softwarequalität und -test

Qualitätsmerkmale

Was ist Softwarequalität? (Fortsetzung)

5. **Änderbarkeit** (Aufwand zur Durchführung vorgegebener Änderungen)
 - **Analysierbarkeit** (Aufwand, um Mängel oder Ursachen von Versagen zu *diagnostizieren* oder um änderungsbedürftige Teile zu *bestimmen*)
 - **Modifizierbarkeit** (Aufwand zur *Ausführung* von *Verbesserungen*, zur Fehlerbeseitigung oder Anpassung an Umgebungsänderungen)
 - **Stabilität** (Wahrscheinlichkeit des Auftretens unerwarteter Wirkungen von Änderungen; funktionelle Fehler, die durch eine getätigte Änderung mittelbar an anderer Stelle auftreten → *Seiteneffekte*)
 - **Prüfbarkeit** (Aufwand, der zur *Prüfung* der geänderten Software notwendig ist)

—

Softwarequalität und -test

Qualitätsmerkmale

Was ist Softwarequalität? (Fortsetzung)

6. **Übertragbarkeit** (Aufwand, Software in eine andere Umgebung zu übertragen)
 - **Anpassbarkeit** (Möglichkeiten, die Software an verschiedene, festgelegte Soft- und Hardware-**Umgebungen** anzupassen)
 - **Installierbarkeit** (Aufwand, der zum Installieren der Software in einer festgelegten Umgebung notwendig ist)
 - **Konformität** (Grad, in dem die Software Standards, Normen oder Vereinbarungen **zur Übertragbarkeit** [z.B. Elster, HBCI, JSON] erfüllt)
 - **Austauschbarkeit** (Möglichkeit, diese Software anstelle einer spezifizierten anderen in der Umgebung jener Software zu verwenden)

—

Softwarequalität und -test

QM und QS in der Softwareentwicklung

Was ist Softwarequalität? (Fortsetzung)

Qualitätsmerkmal

Bei konkurrierenden Qualitätsmerkmalen oder schwieriger Definition der Prioritäten der einzelnen Merkmale kann die sogenannte **Präferenzmatrix** eingesetzt werden:

Dabei wird jedes Merkmal mit jedem verglichen und das wichtigere in den **Kreuzungspunkt** eingetragen. Anschließend werden die **Anzahlen der Nennungen** der einzelnen Merkmale ermittelt. Das mit der höchsten Anzahl hat die oberste Priorität usw.

Qualitätsmerkmal	Anz.						
Qualitätsmerkmal 1	0	A					
				A/B			
Qualitätsmerkmal 2	0	B			A/C		
				B/C		A/D	
Qualitätsmerkmal 3	0	C			B/D		A/E
				C/D		B/E	A/F
Qualitätsmerkmal 4	0	D			C/E		B/F
				D/E		C/F	
Qualitätsmerkmal 5	0	E			D/F		
				E/F			
Qualitätsmerkmal 6	0	F					

Qualitätssicherungsmodell

Operationalisierung der Qualitätsmerkmale

Softwarequalität und -test

QM und QS in der Softwareentwicklung

Qualitätsmanagement (Fortsetzung)

Operationalisierung der Qualitätsmerkmale

Damit diese Anforderungen mit analytischen Maßnahmen mess- und bewertbar gemacht werden können, müssen die Teilmerkmale durch **Qualitätsindikatoren** bzw. **Metriken** beschrieben werden.

Quantifizierbare Indikatoren werden mit Hilfe von **Qualitätsmaßen** quantitativ gemessen.

Das Messergebnis (Messwert) wird auf einer vorher festgelegten **Messskala** abgebildet.

Softwarequalität und -test

QM und QS in der Softwareentwicklung

Qualitätsmanagement (Fortsetzung)

Operationalisierung der Qualitätsmerkmale

Die Operationalisierung der Qualitätsmerkmale wird in drei Stufen durchgeführt:

In der ersten Stufe werden die **Qualitätsziele und Qualitätsmerkmale definiert**.

In der zweiten Stufe wird die **Gewichtung der Qualitätsmerkmale vorgenommen** (meist Gewichtung von 0 bis 3).

Und in der dritten Stufe wird die **Operationalisierung durchgeführt**, das heißt, es werden die Qualitätsmaße definiert.

Softwarequalität und -test

QM und QS in der Softwareentwicklung

Qualitätsmanagement (Fortsetzung)

Operationalisierung der Qualitätsmerkmale

Beispiel...

Abstrakte Qualitätsziele (Merkmal)

Auf der obersten Ebene werden die Qualitätsziele festgelegt. Dabei bedient man sich normierter Begriffe aus [der vorgegebenen Liste des QM-Handbuchs](#). Auf dieser Ebene wird der Kunde grob angegeben, was er meint, indem er ein Merkmal durch allgemeine Qualitätsmerkmale erklärt.

Merkmal:	Benutzerfreundlichkeit
Teilmerkmal:	Bedienbarkeit
Qualitätsindikator:	Prozessbearbeitungszeit
Zielwert:	2 Minuten
Qualitätsmaß:	Der Maßstab wird in Minuten angegeben, zusätzlich wird festgelegt, wie die Messung durchzuführen ist.

Softwarequalität und -test

QM und QS in der Softwareentwicklung

Qualitätsmanagement (Fortsetzung)

Operationalisierung der Qualitätsmerkmale

Konkrete Qualitätsaspekte (Teilmerkmal)

Die meiste Zeit wird für die mittlere Ebene aufgewendet. Hier geht es darum, die **Bedeutung der abstrakten Qualitätsziele auf die konkrete Situation** des Kunden und des Projekts zu beziehen und zu verfeinern.

Merkmal: Benutzerfreundlichkeit

Teilmerkmal: Bedienbarkeit

Qualitätsindikator: Prozessbearbeitungszeit

Zielwert: 2 Minuten

Qualitätsmaß: Der Maßstab wird in Minuten angegeben, zusätzlich wird festgelegt, wie die Messung durchzuführen ist.

Softwarequalität und -test

QM und QS in der Softwareentwicklung

Qualitätsmanagement (Fortsetzung)

Operationalisierung der Qualitätsmerkmale

Indikatoren und Metriken (Zielwert, Qualitätsmaß)

Hier geht es darum, **welches Maß an welches Objekt angelegt wird**. Das können quantitative Maße sein, oder auch qualitative Hinweise, wie Indikatoren. Metriken werden meist wieder verwendet, und eher selten neue entwickelt.

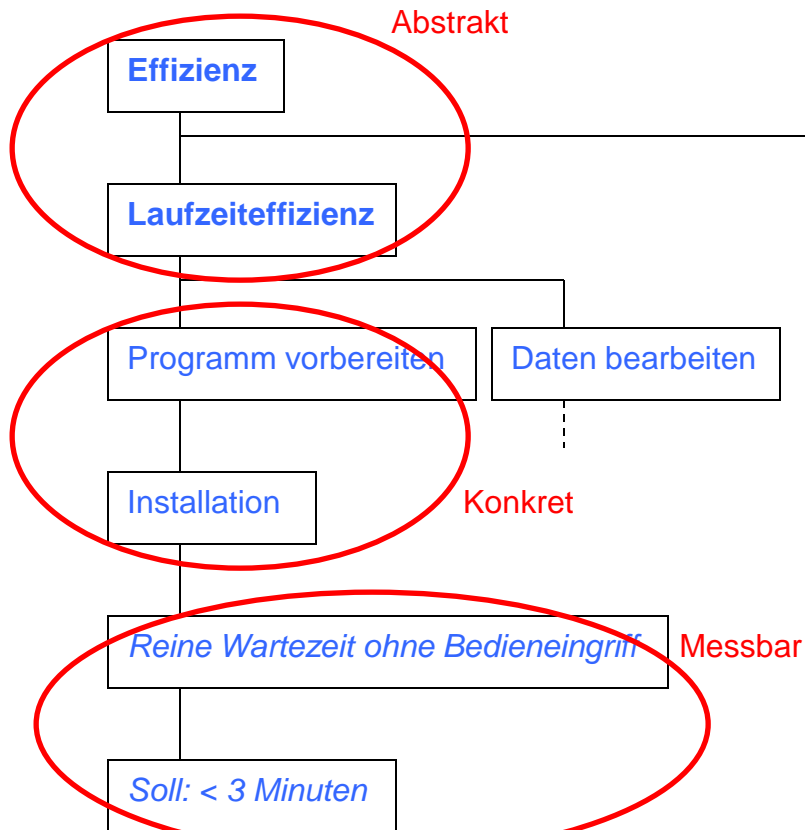
Merkmal:	Benutzerfreundlichkeit
Teilmerkmal:	Bedienbarkeit
Qualitätsindikator:	Prozessbearbeitungszeit
Zielwert:	2 Minuten
Qualitätsmaß:	Der Maßstab wird in Minuten angegeben, zusätzlich wird festgelegt, wie die Messung durchzuführen ist.

Softwarequalität und -test

QM und QS in der Softwareentwicklung

Qualitätsmanagement (Fortsetzung)

Operationalisierung der Qualitätsmerkmale



Abstrakt:	Allgemeine Qualitätsziele
Konkret:	Konkretisiert für die eigene Situation
Messbar:	Indikatoren und Metriken

Entstehung eines QM-Handbuchs

Softwarequalität und -test

QM und QS in der Softwareentwicklung

Qualitätsmanagement (Fortsetzung)

QM-Handbuch

Die Erstellung des QM-Handbuchs wird sinnvollerweise in folgenden Schritten bearbeitet:

1. Bildung einer Arbeitsgruppe

In der Regel ist das Handbuch ein **bereichsübergreifendes Dokument** mit einem **unternehmensweiten Gültigkeitsbereich**. Deshalb sollte es nicht von einer Person alleine, sondern immer **von einem Team bearbeitet** werden, das sich aus Vertretern der wichtigsten Aufgaben- und Funktionsbereiche zusammensetzt. Typischerweise wird der **Leiter des Qualitätswesens** dieses Team leiten und in einem ersten Treffen über die **Ziele** und anstehenden **Aufgaben** informieren.

Softwarequalität und -test

QM und QS in der Softwareentwicklung

Qualitätsmanagement (Fortsetzung)

QM-Handbuch

2. Sammlung der qualitätsrelevanten Unterlagen

Die vorhandenen **qualitätsrelevanten Unterlagen** werden durch das Team zusammengetragen und gesichtet.

Hierbei ist zu klären, ob die Unterlagen einem Kapitel des Handbauchs oder den Verfahrens- und Arbeitsanweisungen zugehörig sind und ob sie für die geplante Darstellung des QM-Systems ausreichen.

Softwarequalität und -test

QM und QS in der Softwareentwicklung

Qualitätsmanagement (Fortsetzung)

QM-Handbuch

3. Erstellung des organisatorischen Teils

Bevor mit der eigentlichen Beschreibung der QM-Elemente begonnen wird, empfiehlt es sich, den organisatorischen Rahmen für das Handbuch und für die Lenkung der Dokumente festzuschreiben. Hiermit sind zugleich die Verantwortlichkeiten und Aufgaben innerhalb des Projekt-Teams bestimmt.

Softwarequalität und -test

QM und QS in der Softwareentwicklung

Qualitätsmanagement (Fortsetzung)

QM-Handbuch

4. Erster Entwurf einer Beschreibung der Elemente des QM-Systems

Kapitelweise wird nun durch die Mitarbeiter des Projekt-Teams ein **erster Entwurf des Handbuchs** erstellt.

Softwarequalität und -test

QM und QS in der Softwareentwicklung

Qualitätsmanagement (Fortsetzung)

QM-Handbuch

5. Durchsicht und Prüfung des Entwurfs

Mit dem Abschluss des ersten Entwurfs liegt nun zum ersten Mal ein Gesamtdokument des QM-Handbuchs vor.

Jeder Mitarbeiter des Teams muss nun das Handbuch (nicht nur die von ihm erstellten Kapitel!) hinsichtlich **Korrektheit, Verständlichkeit, Vollständigkeit und Redundanzen prüfen**.

Änderungswünsche werden in einer Tabelle zusammengetragen und nach vollständiger Prüfung des Handbuchs im Team diskutiert.

Softwarequalität und -test

QM und QS in der Softwareentwicklung

Qualitätsmanagement (Fortsetzung)

QM-Handbuch

6. Abschließende Bearbeitung des Texts

Die durch das Team freigegebene Änderungstabelle ist Grundlage für die **abschließende Bearbeitung der Handbuchtexe** durch die verantwortlichen Mitarbeiter des Teams.

Softwarequalität und -test

QM und QS in der Softwareentwicklung

Qualitätsmanagement (Fortsetzung)

QM-Handbuch

7. Freigabe und Verteilung des Handbuchs

Die abschließende Freigabe des QM-Handbuchs erfolgt durch den Leiter des Qualitätswesens und die Geschäftsführung.

Sie wird durch Unterschriften bestätigt. Zusätzlich bestätigen die Verantwortlichen für Erstellung und Prüfung der Handbuchkapitel durch Unterschrift, das das entstandene Dokument korrekt und vollständig ist.

Qualitätssicherung

Softwarequalität und -test

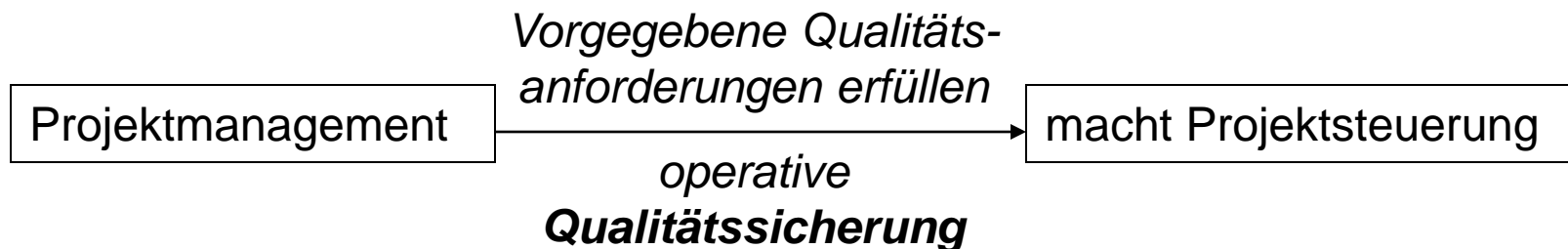
QM und QS in der Softwareentwicklung

Qualitätssicherung

Qualitätssicherung wird heute vielfach immer noch synonym zu „Testen“ verwendet.

Dabei hat sich die Qualitätssicherung in den letzten Jahren stark weiterentwickelt. Es geht nicht mehr nur um den **Test und die Abnahme** des fertigen **Produkts**, sondern vielmehr um einen...

...Prozess, der von Beginn an das Vorgehen im Projekt begleitet und mit geeigneten Aktivitäten die Unterstützung bietet, um das Projektmanagement um eine qualitätssichernde Projektsteuerung zu ergänzen .



Softwarequalität und -test

QM und QS in der Softwareentwicklung

Qualitätssicherung (Fortsetzung)

Def.: „**Qualitätssicherung ist der während der Durchführung des Projekts stattfindende, operative Teil des Qualitätsmanagements, der auf das Erzeugen von Vertrauen darauf gerichtet ist, dass vorgegebene Qualitätsanforderungen erfüllt werden.**“

Das heißt, das Qualitätsmanagement beschreibt in der Regel dasjenige, **was** im Projekt im Rahmen der Qualitätssicherung **wie** durchzuführen ist.

Softwarequalität und -test

QM und QS in der Softwareentwicklung

Qualitätssicherung (Fortsetzung)

Die Qualitätssicherung wird oft systematisch in drei Bereiche eingeteilt: Konstruktive, analytische und organisatorische Maßnahmen.

Konstruktive QS-Maßnahmen versuchen schon vor dem Beginn einer Entwicklungsphase (wie klein auch immer diese sein möge), Probleme gar nicht erst entstehen zu lassen.

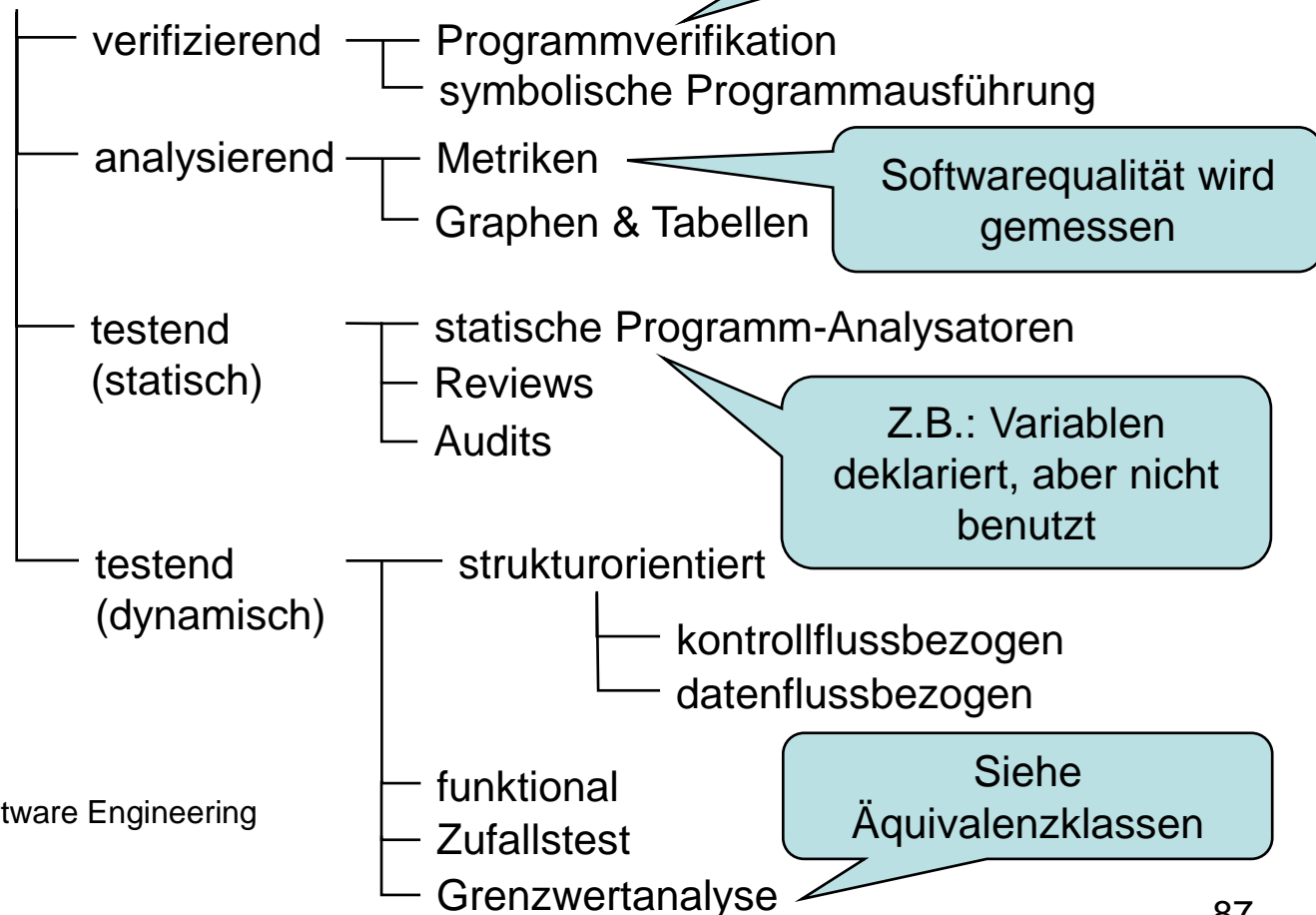
Analytische Qualitätssicherungsmaßnahmen hingegen suchen Fehler in den Objekten der Qualitätssicherung, damit man sie beheben kann.

Softwarequalität und -test

QM und QS in der Softwareentwicklung

Qualitätssicherung
(Fortsetzung)

Qualifikation der
Verfahren der
analytischen
Qualitätssicherung



D. Macos nach K. Bothe, Software Engineering

Softwarequalität und -test

QM und QS in der Softwareentwicklung

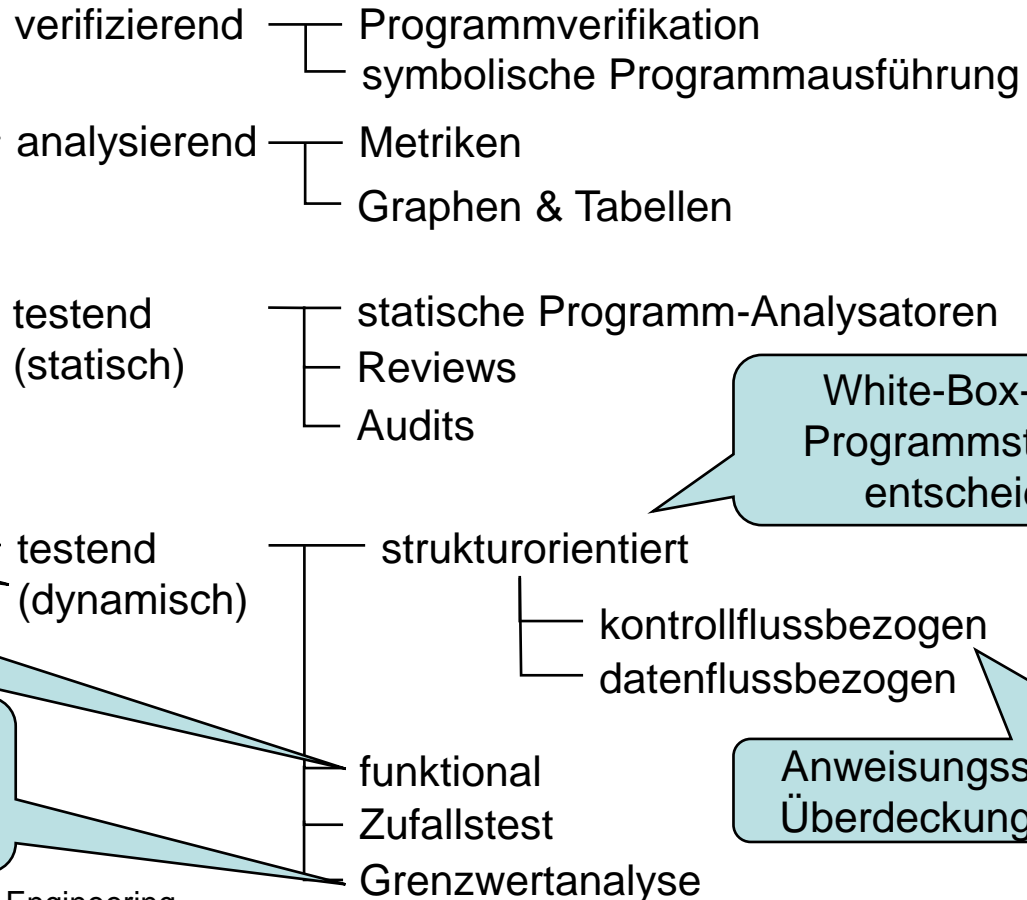
Qualitätssicherung
(Fortsetzung)

Qualifikation der
Verfahren der
analytischen
Qualitätssicherung

Anwendung des
Programmes auf Testdaten

Black-Box-Test: funktionale
Anforderungen

fehlerorientierte Verfahren:
Ausnahmesituationen,
Grenzwerte



White-Box-Test:
Programmstruktur
entscheidet

Anweisungsstruktur,
Überdeckungsgrade

Qualitätssicherungs- maßnahmen

Softwarequalität und -test

QM und QS in der Softwareentwicklung

Qualitätssicherung (Fortsetzung)

Objekte der Qualitätssicherung

Grundsätzlich sind alle Arbeitsergebnisse eines Projekts Objekte der Qualitätssicherung. Dabei ist es unerheblich, ob es sich dabei um **Dokumente** oder **Software** im eigentlichen Sinne handelt, oder ob die Zielgruppe des erstellten Ergebnisses **intern** (Projektmitarbeiter) oder **extern** (Auftraggeber, Benutzer, Management) ist.

Softwarequalität und -test

QM und QS in der Softwareentwicklung

Qualitätssicherung (Fortsetzung)

Qualitätssicherungsmaßnahmen

Ziel der Qualitätssicherungsmaßnahmen ist es, dass die vom Auftraggeber sowie die im Qualitätssicherungsmodell (QM-Handbuch) vorgegebenen **Qualitätsanforderungen an das zu erstellende Produkt erfüllt werden.**

Die Qualitätssicherungsmaßnahmen sollten **während des ganzen Entwicklungsprozesses** durchgeführt werden, und zwar möglichst von Anfang an, da sich die Qualität in ein Produkt nicht „hineintesten“ lässt.

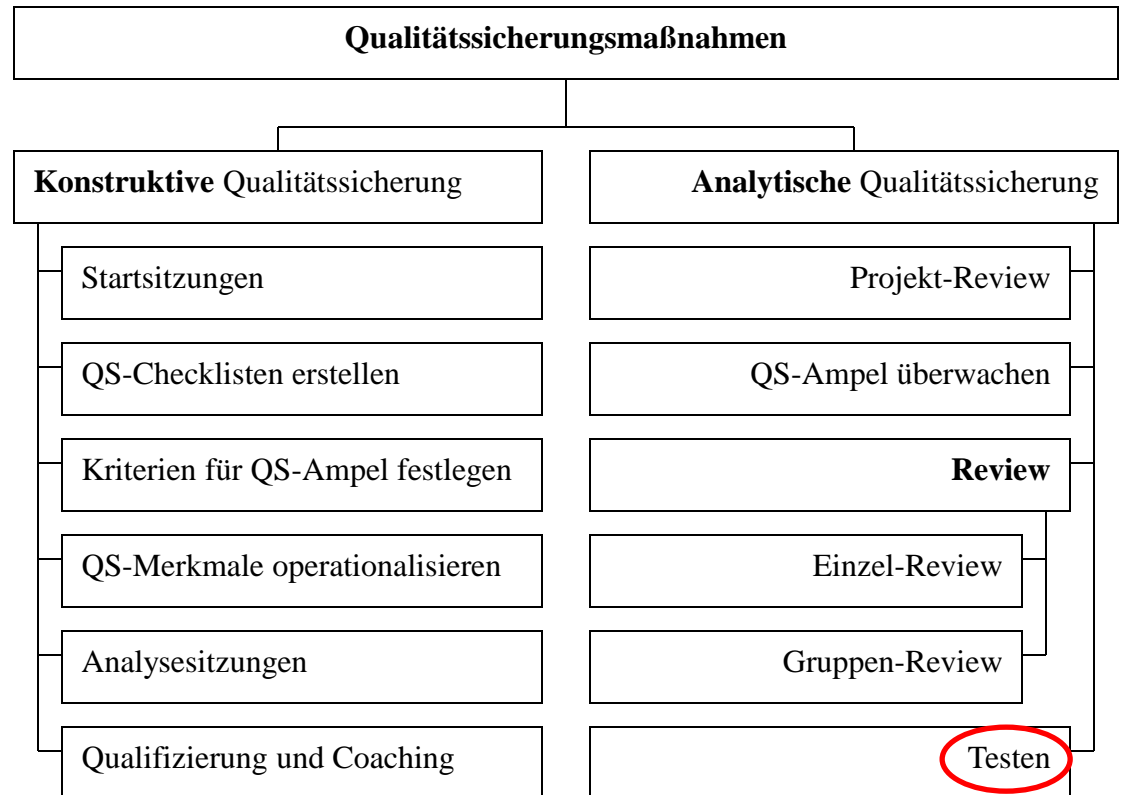
Softwarequalität und -test

QM und QS in der Softwareentwicklung

Qualitätssicherung

Qualitätssicherungsmaßnahmen

Die Qualitätssicherungsmaßnahmen werden unterschieden in **konstruktive** und **analytische** Qualitätssicherung.



Guck' mal, so klein ist es...

Softwarequalität und -test

QM und QS in der Softwareentwicklung

Qualitätssicherung (Fortsetzung)

Qualitätssicherungsmaßnahmen

Die **konstruktiven Qualitätssicherungsmaßnahmen** umfassen alle Maßnahmen und Hilfsmittel, die ~~prozess-vorbereitend~~ zur Erzeugung qualitativ hochwertiger Anwendungssysteme ergriffen werden können. Sie unterstützen den Projektmitarbeiter schon vor und während des Entwicklungsprozesses, die Qualitätsziele zu erreichen, die durch Qualitätsanforderungen vom Auftraggeber festgelegt wurden.

Def.: „Konstruktive Qualitätssicherungsmaßnahmen sind alle die Maßnahmen, die bereits bei der Konstruktion von Software auf die Verbesserung ausgewählter Qualitätsaspekte abzielen und nicht erst nachträglich durch Prüfung und Korrektur.“

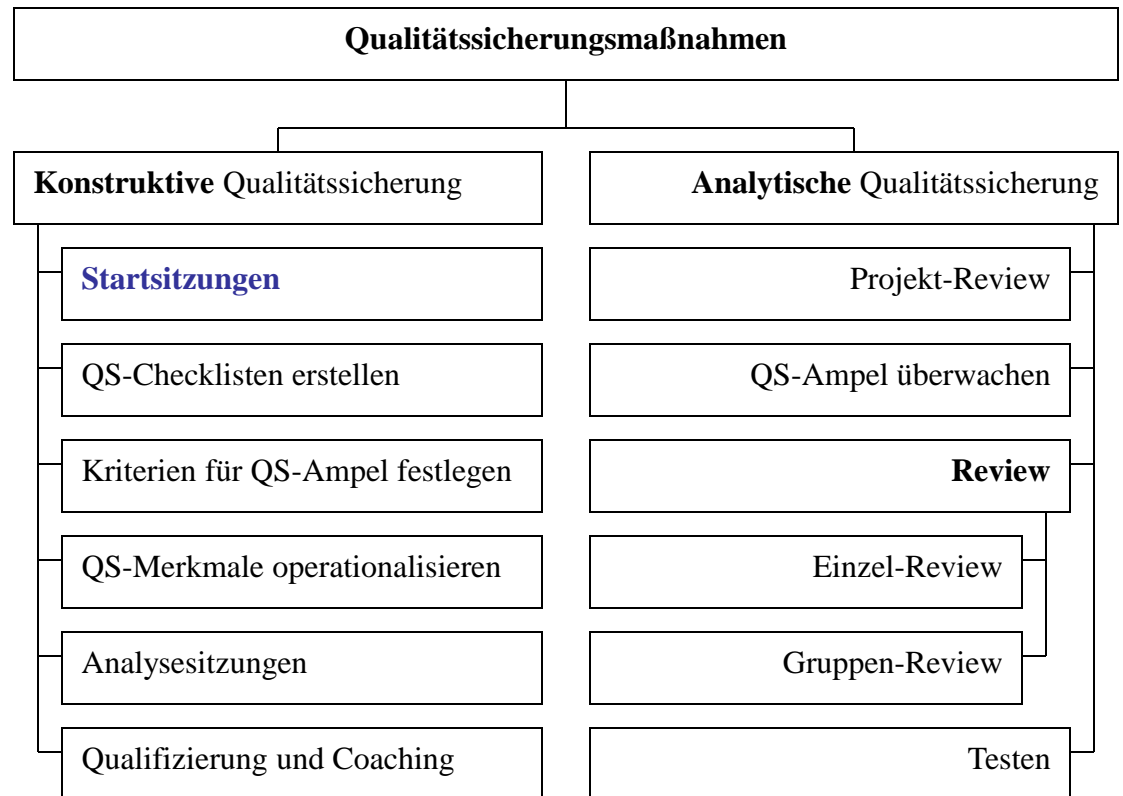
Softwarequalität und -test

QM und QS in der Softwareentwicklung

Qualitätssicherung

Startsitzung

Einheitlicher Basisinformationsstand aller Beteiligten (Auftraggeber „allgemeines Marketing“, Mitarbeiter „das Team“, Projektleiter, Qualitätssicherungsbeauftragter, Vorgesetzte) zu Beginn einer Phase oder eines Arbeitsabschnitts. Die Mitarbeiter sollen auf die neue Aufgabe eingeschworen und motiviert werden.



Softwarequalität und -test

QM und QS in der Softwareentwicklung

Qualitätssicherung (Fortsetzung)

Kickoff (Startsitzung)

Das Kickoff (Startsitzung) ist eine gemeinsame Sitzung zum Start einer Prozessphase. Darin werden die Ziele der Phase erläutert und die konstruktiven Qualitätssicherungsmaßnahmen zu deren Erreichen vorgestellt.

Softwarequalität und -test

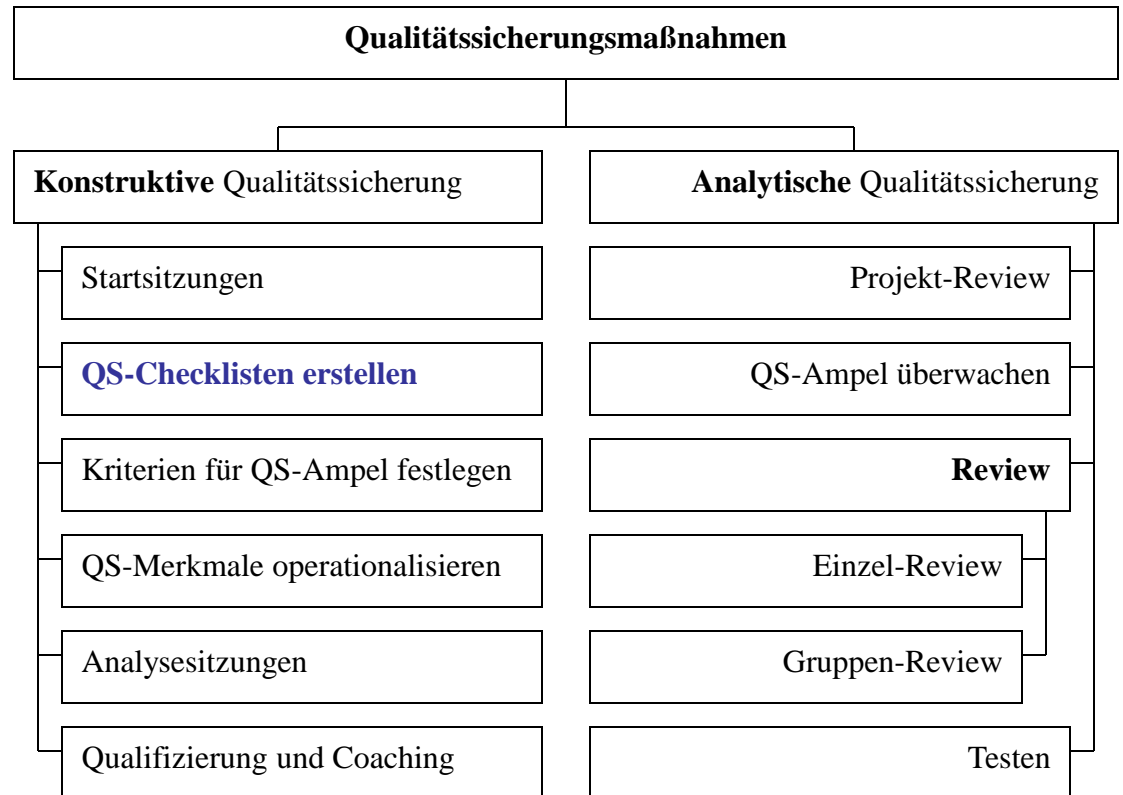
QM und QS in der Softwareentwicklung

Qualitätssicherung

QS-Checklisten erstellen

Checklisten enthalten die Kriterien und Standards, nach denen Ergebnisse und Teilergebnisse entwickelt wie auch geprüft werden sollen.

Sie spielen beim Review eine entscheidende Rolle.



Softwarequalität und -test

QM und QS in der Softwareentwicklung

Qualitätssicherung (Fortsetzung)

QS-Checklisten erstellen

QS-Merkmal

Sind nachfolgende Ergebnisse der Entwurfsdokumente vorhanden?

- () Use-Case-Modell?
- () Klassendiagramm?
- () Sequenzdiagramm?
- () Zustandsübergangsdigramm?
- () Objektmodell?

Wie ist die Dokumentenqualität?

Leserfreundlich, vollständig, korrekt, modular, änderbar, strukturiert

...

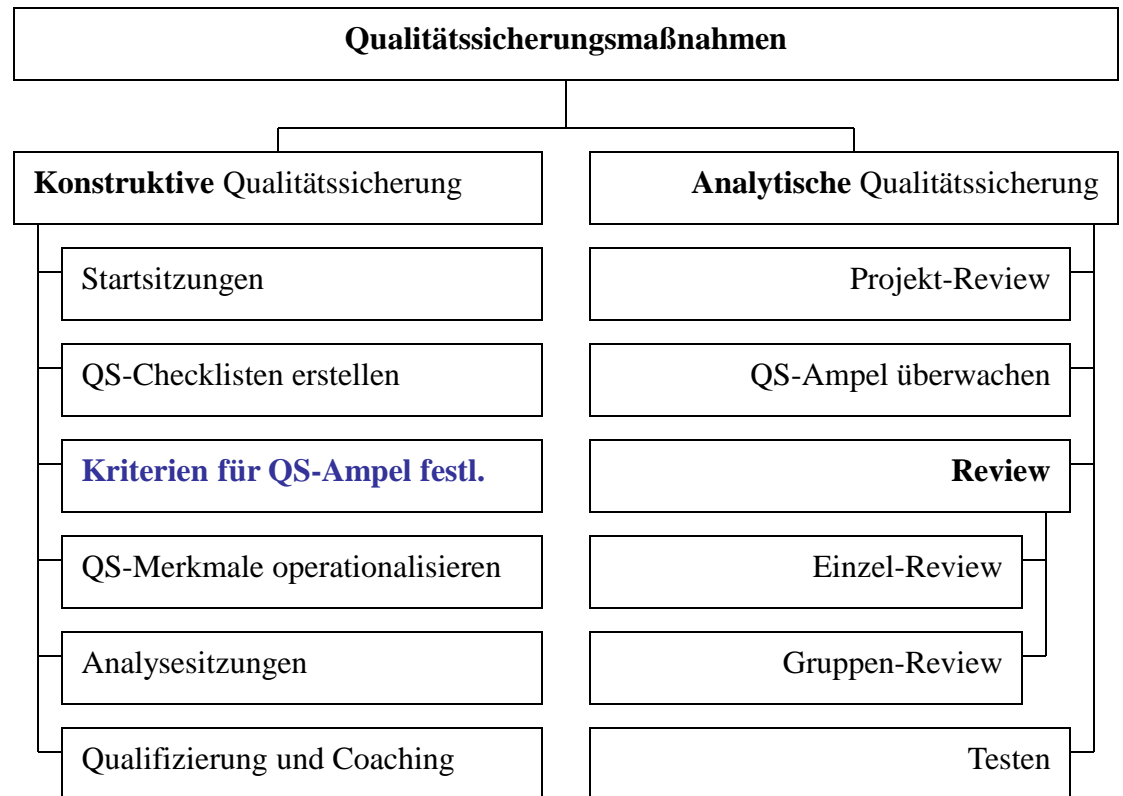
Softwarequalität und -test

QM und QS in der Softwareentwicklung

Qualitätssicherung

Kriterien für QS-Ampel festlegen

Die Qualitätssicherungsmaßnahmen sollen analog zu allen anderen Aktivitäten **im Projekt eingeplant werden**. Nur so wird der Ablauf des Projekts für alle Beteiligten nachvollziehbar und man kann die Gesamtaufwände erkennen.



Softwarequalität und -test

QM und QS in der Softwareentwicklung

Qualitätssicherung (Fortsetzung)

Kriterien für QS-Ampel festlegen (und gewichten!)

- Die im Projektplan eingetragenen Termine werden eingehalten? x 3
- Protokolle und Formulare für formale und inhaltliche Bewertungen sind vollständig ausgefüllt? x 3
- Die für das Projekt definierten Methoden und Verfahren werden eingehalten? x 2
- Die Ergebnisse der inhaltlichen Prüfungen stimmen mit den Vorgaben überein? x 3
- Statistische Auswertungen werden durchgeführt? x 1
- ...

Hinterher Formel zur Berechnung der Ampelfarbe aufstellen.

Softwarequalität und -test

QM und QS in der Softwareentwicklung

Qualitätssicherung

QS-Merkmale operationalisieren

Konkrete Definition der im Projekt berücksichtigten Qualitätsmerkmale.



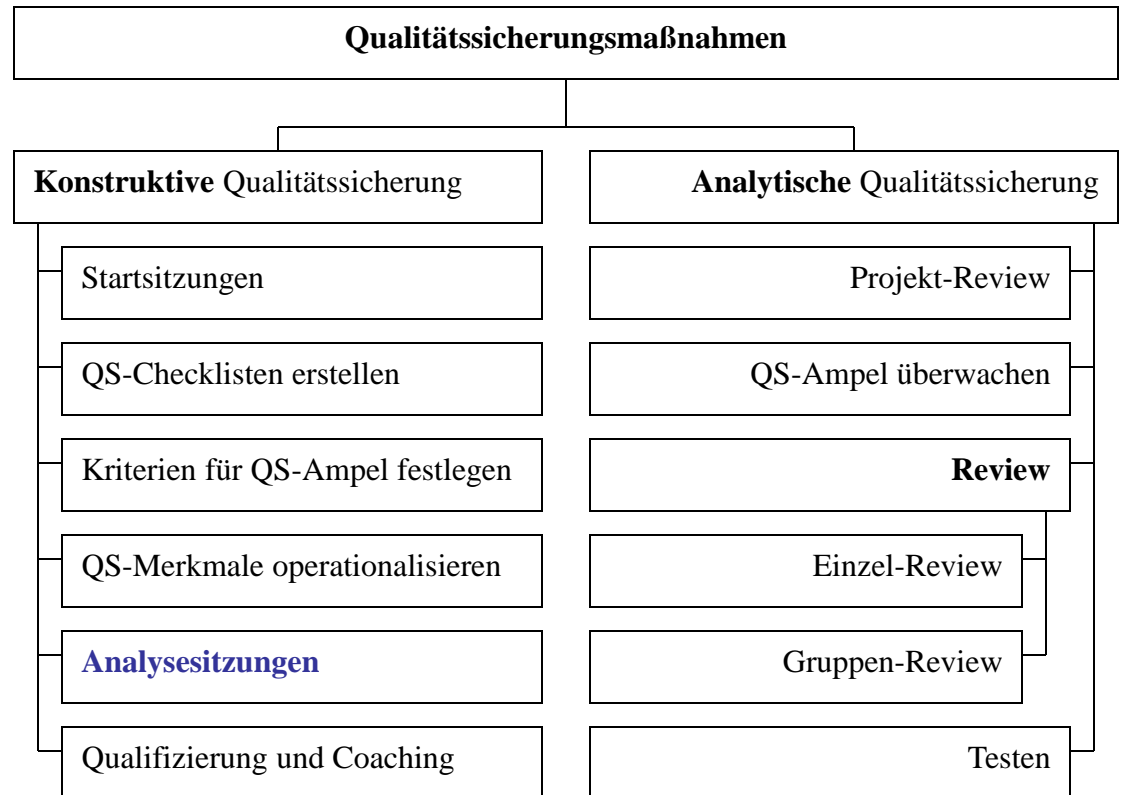
Softwarequalität und -test

QM und QS in der Softwareentwicklung

Qualitätssicherung

Analysesitzungen

Die Analysesitzungen vertiefen die Auseinandersetzung der Mitarbeiter mit den **von ihnen verursachten Fehlern**, wobei durch moderierte Gruppenarbeit, Vertraulichkeit der Einzelheiten und Ausschluss der Vorgesetzten eine konstruktive Atmosphäre sichergestellt werden muss.



Softwarequalität und -test

QM und QS in der Softwareentwicklung

Qualitätssicherung

Analysesitzungen

Wie ist der Fehler entstanden? (Ursache)

Wann ist er entstanden? (Phase)

Wie kann er künftig vermieden werden? (Maßnahme)

Sind ähnliche Fehler denkbar? (Querschluss)

Wo könnten oder müssten sie auftreten? (potenzieller Querschluss)

Sind sie dort zu finden? (tatsächlicher Querschluss)

Wie können auch sie künftig vermieden werden? (Maßnahme)

—

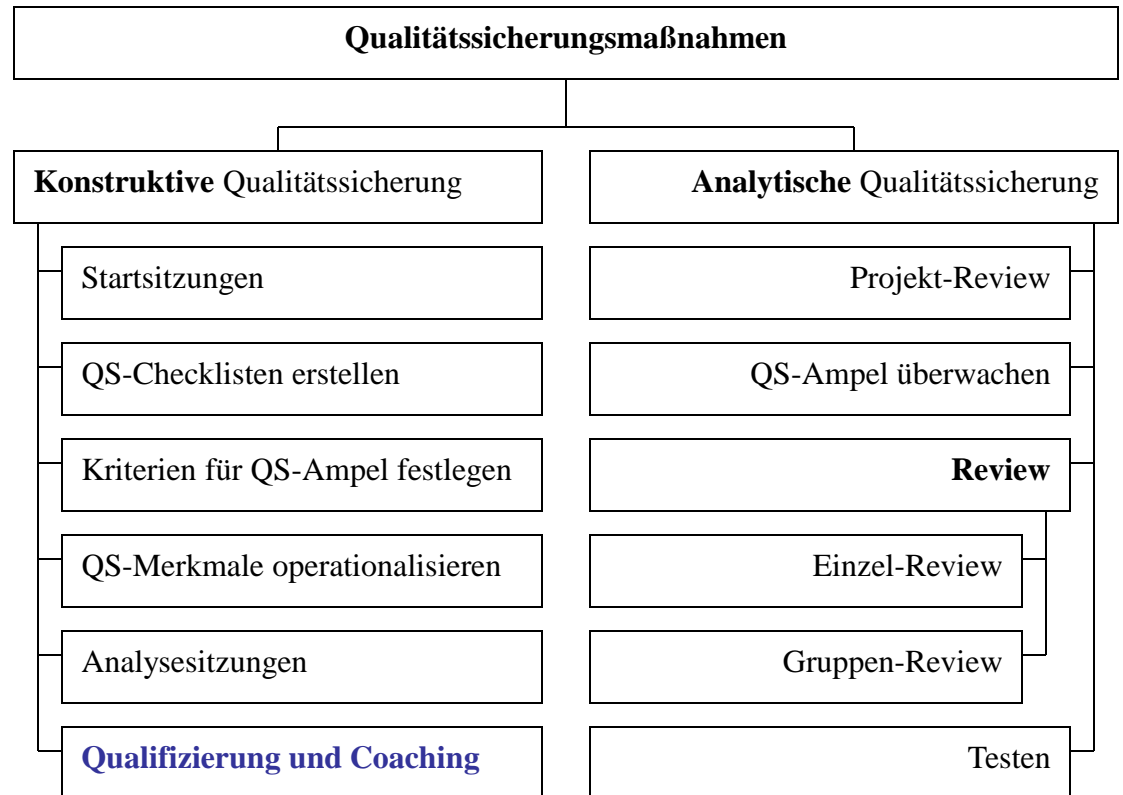
Softwarequalität und -test

QM und QS in der Softwareentwicklung

Qualitätssicherung

Qualifizierung und Coaching

Unter dem Begriff **Qualifizierung** versteht man den Vorgang zur Erlangung von Fähigkeiten, um eine bestimmte Aufgabe oder Anforderung erfüllen zu können. Mit **Coaching** sind alle Konzepte gemeint, die in professioneller Form individuelle Beratung im beruflichen Kontext anbieten.



Softwarequalität und -test

QM und QS in der Softwareentwicklung

Qualitätssicherung (Fortsetzung)

Qualitätssicherungsmaßnahmen

Die **analytischen Qualitätssicherungsmaßnahmen** haben hingegen das Ziel, in **Reviews** mit Hilfe der Abnahmekriterien Fehler zu entdecken, die in Arbeitsergebnissen (beispielsweise **Programm**, **Datenmodell**, **Zieldefinition**, **Dokument** etc.) in mehr oder minder großem Umfang enthalten sind.

Analytische Maßnahmen sind **diagnostische Maßnahmen**, das heißt, sie bringen in das Produkt keine Qualität per se, sondern sie können nur noch nachträglich Verbesserungen **erwirken**.

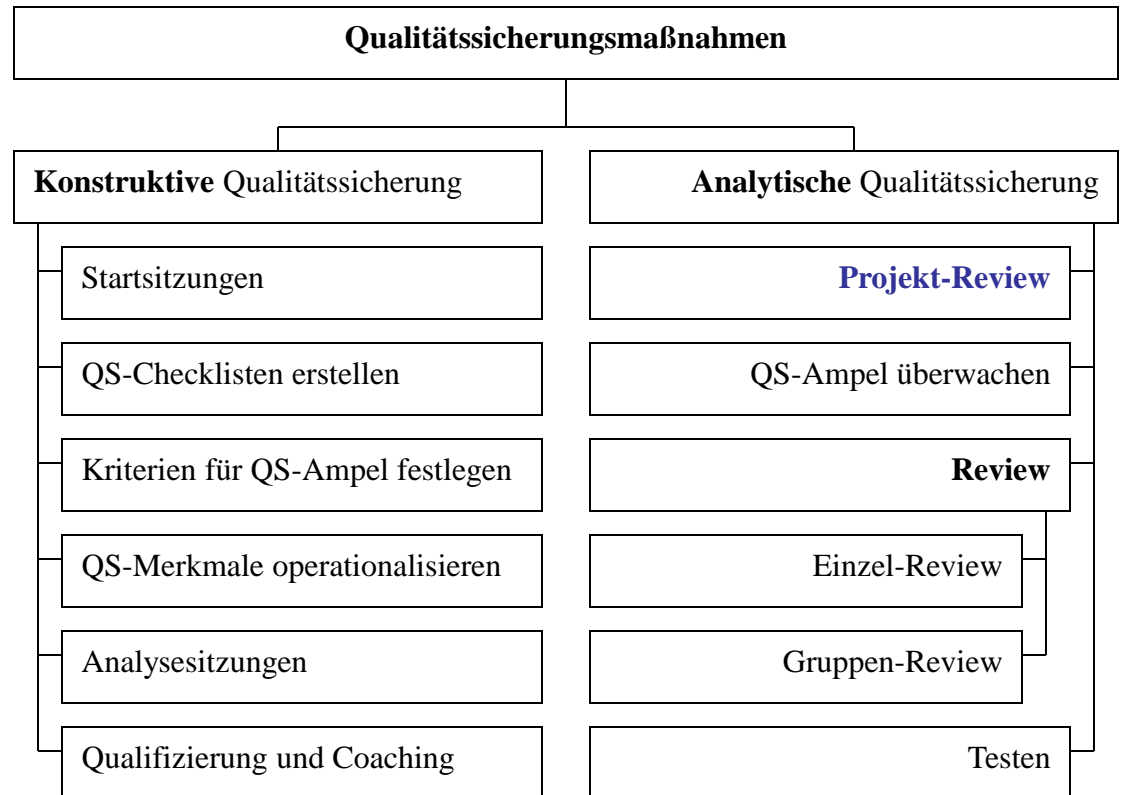
Softwarequalität und -test

QM und QS in der Softwareentwicklung

Qualitätssicherung

Projekt-Review

Projekt-Reviews werden durchgeführt, um den Projektstand aus der Sicht des Managements zu bewerten und evtl. Abweichungen vom ursprünglichen Plan zu erkennen. Es verfolgt zwei Ziele, erstens die Steuerung und Überwachung des Projektes und zweitens die Steuerung und Überwachung der Qualität.



Softwarequalität und -test

QM und QS in der Softwareentwicklung

Qualitätssicherung (Fortsetzung)

Review

Der Überbegriff Review (auch *technische Inspektion* genannt) steht für Prüfverfahren, bei denen ein **Arbeitsergebnis** (der so genannte *Prüfling*) eines **Autors** von einem oder mehreren **Gutachtern** betrachtet und geprüft wird.

Die Gutachter konzentrieren sich gemäß ihrer Expertise auf **einige ausgewählte Aspekte**.

Die Review-Ergebnisse werden meist in einer **gemeinsamen Sitzung** zusammengeführt.

Softwarequalität und -test

QM und QS in der Softwareentwicklung

Qualitätssicherung (Fortsetzung)

Review

Wie Tests haben Reviews das Ziel, **Fehler zu finden** (man spricht allerdings zunächst von „**Befunden**“ statt von Fehlern, weil sich ein Gutachter beim Bearbeiten ja irren kann).

Wie im Test werden auch im Review **nur Befunde gesammelt**, Korrekturen oder Verbesserungen sollen aber nicht vorgeschlagen werden. Die führt der *Autor* in der so genannten Nacharbeit durch.

Softwarequalität und -test

QM und QS in der Softwareentwicklung

Qualitätssicherung (Fortsetzung)

Review

Moderator

Er **organisiert den Ablauf** und leitet insbesondere die gemeinsamen Sitzungen.

Gutachter

Zwei bis fünf Gutachter werden in der Regel vom Moderator benannt. Sie **bearbeiten und prüfen den Prüfling** und berichten darüber.

Schriftführer

Er **schreibt in der Sitzung mit**, welche Befunde gesammelt wurden.

—

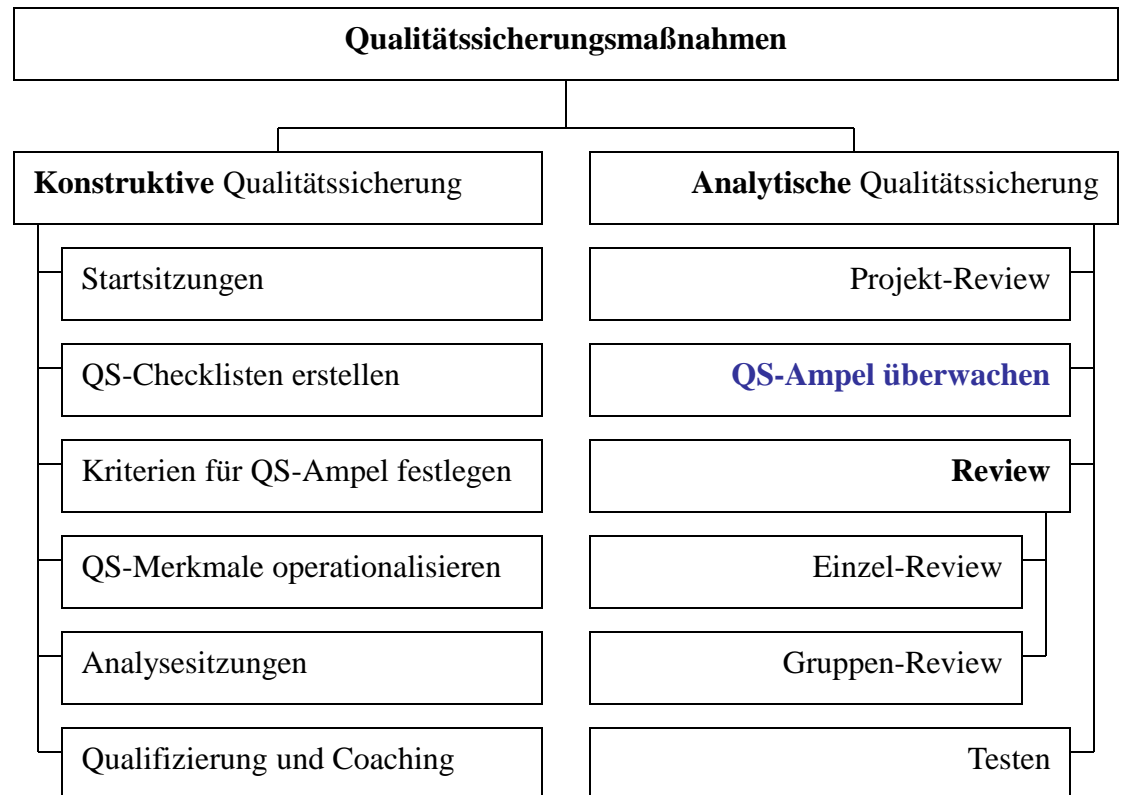
Softwarequalität und -test

QM und QS in der Softwareentwicklung

Qualitätssicherung

QS-Ampel überwachen

Die Kriterien des QS-Ampelverfahrens werden im Rahmen der konstruktiven Maßnahmen zu Projekt- bzw. Phasenbeginn festgelegt. Die Erfüllung der Kriterien überwacht der Qualitäts-sicherungsbeauftragte und leitet die notwendigen Maßnahmen ein.



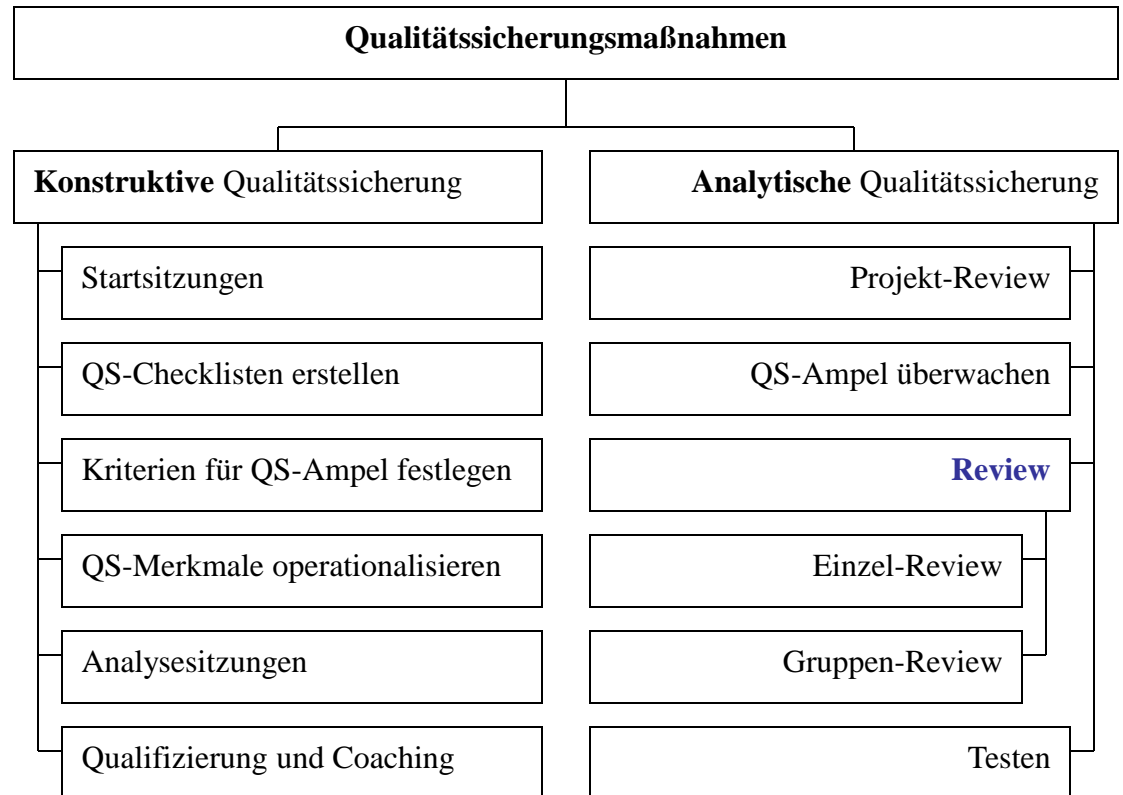
Softwarequalität und -test

QM und QS in der Softwareentwicklung

Qualitätssicherung

Review

Ein Review dient zur Überprüfung der Qualität eines Arbeitsergebnisses.



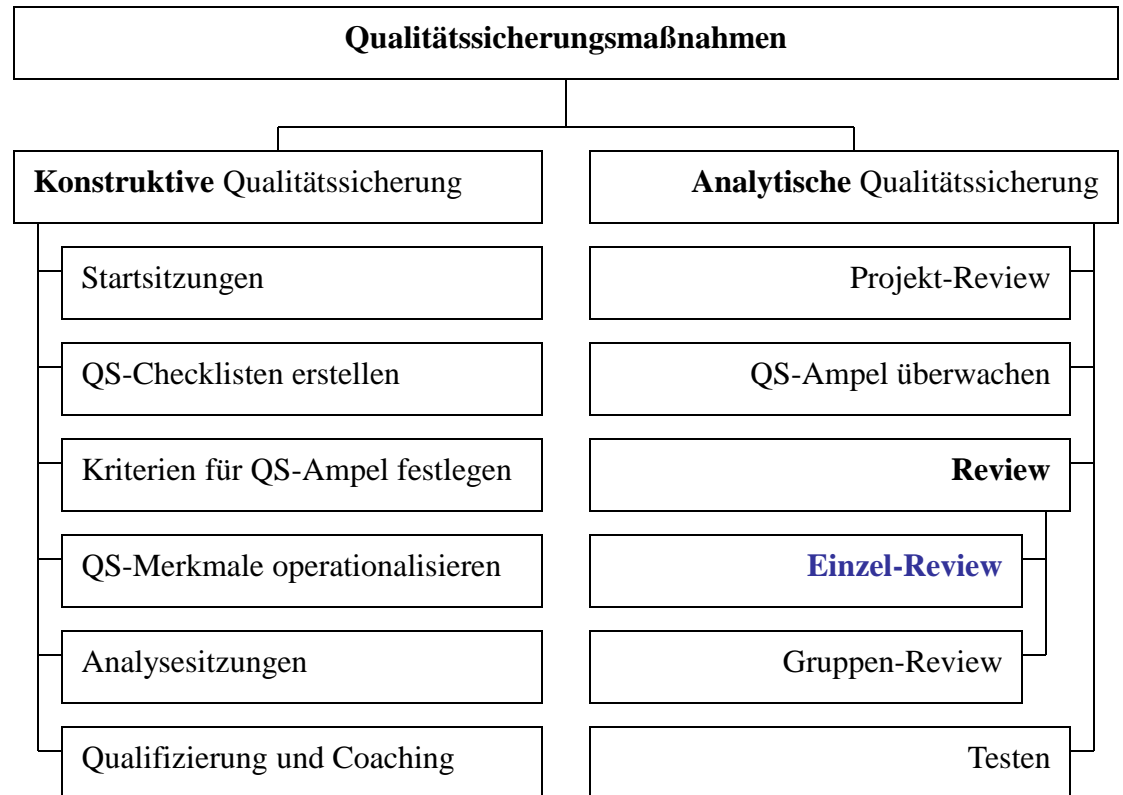
Softwarequalität und -test

QM und QS in der Softwareentwicklung

Qualitätssicherung

Einzel-Review

In einem **Einzel-Review** prüft der Autor seine eigene Arbeit (Schreibtischtest, Selbstprüfung), übergibt das Ergebnis einem Kollegen (Autor/Leser-Prinzip) oder verteilt das Ergebnis an mehrere Prüfer (Einzelprüfung mit QS-Protokoll).



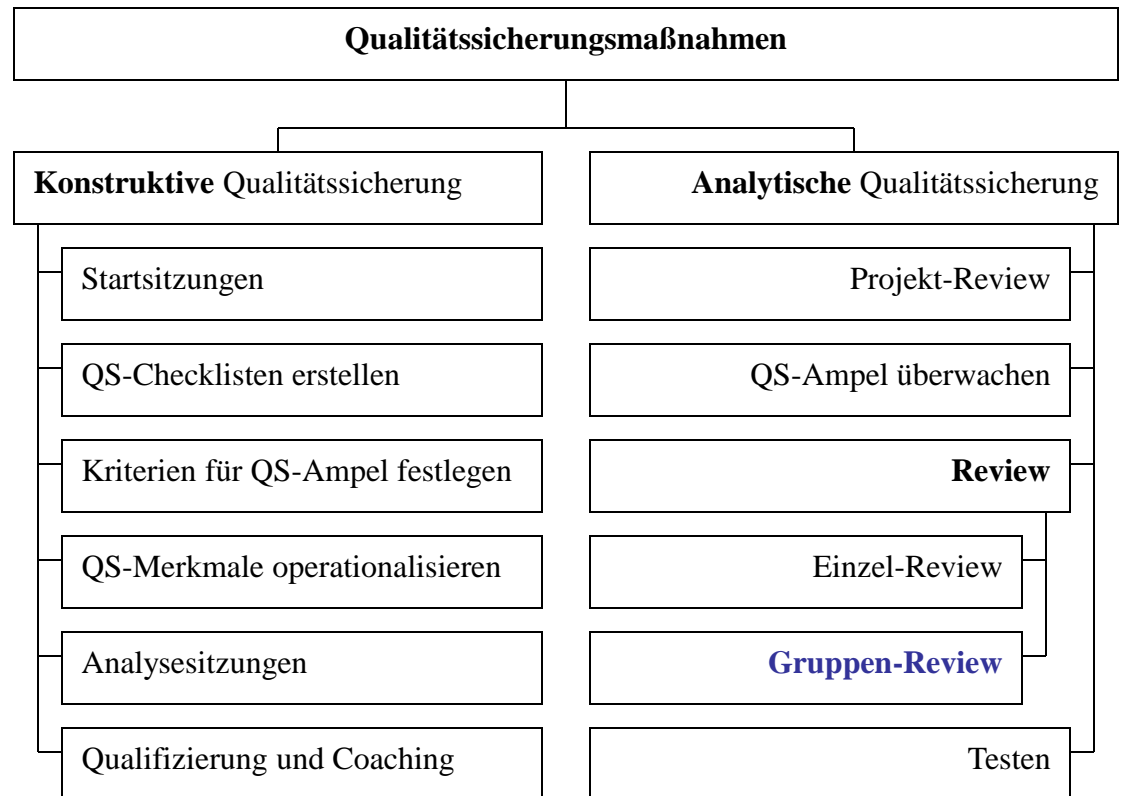
Softwarequalität und -test

QM und QS in der Softwareentwicklung

Qualitätssicherung

Gruppen-Review

dienen zur methodischen Durchführung der Ergebnisprüfung durch mehrere Personen: Offene Fragen besprechen, nach Checkliste das Ergebnis prüfen, Ergebnis vom Autor vorstellen lassen, Testfallmethode, Paraphrasieren, Walk-Through, Inspektion).



Softwarequalität und -test

QM und QS in der Softwareentwicklung

Qualitätssicherung

Gruppen-Review

Paraphrasieren

Die Aussage des Gesprächspartners noch einmal in anderen Worten wiederholen, und damit sicherstellen, dass keine Missverständnisse aufkommen.

Inspektion

Überprüfung mindestens zweier Dokumente in Bezug auf Konsistenz, Korrektheit und Vollständigkeit nach einer festen Prozedur.

Softwarequalität und -test

QM und QS in der Softwareentwicklung

Qualitätssicherung (Fortsetzung)

Walk-Through

Eine weitere klassische Variante ist das Walk-Through, bei dem sich die Gutachter überhaupt **nicht auf die Sitzung vorbereiten** müssen.

Der Autor stellt, ähnlich wie beim Kickoff, **den Prüfling Seite für Seite vor**, und die Gutachter nehmen spontan Stellung. Die Einzelprüfung wurde eingespart.

Es sollte aber klar sein, dass damit auch ein großer Teil der Wirkung und des Nutzens verloren geht. Die Gutachter müssen mit dem Umfeld des Prüflings vertraut sein, sonst können sie in aller Schnelle nur oberflächliche Kommentare abgeben.

Softwarequalität und -test

QM und QS in der Softwareentwicklung

Qualitätssicherung

Review eines Software-Tickets

Review-Checkliste für Ticket _____

Funktionalität

- Alle geforderten Anforderungen aus der Spezifikation sind erfüllt?
- Alle Masken sind gemäß der Maskenbeschreibung umgesetzt?
- Alle Interaktionen der Interaktionsbeschreibung sind erreichbar?
- Alle Interaktionen der Interaktionsbeschreibung sind vollständig umgesetzt?

...

Softwarequalität und -test

QM und QS in der Softwareentwicklung

Qualitätssicherung

Review eines Software-Tickets

Review-Checkliste für Ticket _____

Quelltext

- Der Quelltext entspricht den Code-Konventionen?
- Der Quelltext entspricht den Sprach-Konventionen (Programmiersprache)?
- Alle Klassen, Interfaces und Methoden sind dokumentiert?
- Nicht genutzte Code-Fragmente sind entfernt oder für spätere Nutzung dokumentiert?
- Der Code entspricht den Architekturvorgaben (Schichten, Abhängigkeiten etc.)?

...

Softwarequalität und -test

QM und QS in der Softwareentwicklung

Qualitätssicherung

Review eines Software-Tickets

Review-Checkliste für Ticket _____

Entscheidungen

- Getroffene Architekturentscheidungen sind dokumentiert?
- Getroffene Spezifikationsänderungen sind abgesprochen und dokumentiert?

...

Softwarequalität und -test

QM und QS in der Softwareentwicklung

Qualitätssicherung

Review eines Software-Tickets

Review-Checkliste für Ticket _____

Tests

- Alle Unit-Tests laufen erfolgreich durch?
- Die Unit-Tests testen *public*, *protected* und *private* Methoden?
- Anweisungsüberdeckung (C⁰) liegt bei mindestens 80 Prozent?
- Manuelle View-Tests bringen erwartetes Ergebnis (gegenüber Spezifikation)?

...

Softwarequalität und -test

QM und QS in der Softwareentwicklung

Qualitätssicherung

Review eines Software-Tickets

Review-Checkliste für Ticket _____

Integration

- Git-Branch ist in den *master*-Branch überführt?
- Tests, Build und Deployment auf dem Integrationsserver sind erfolgreich?

...

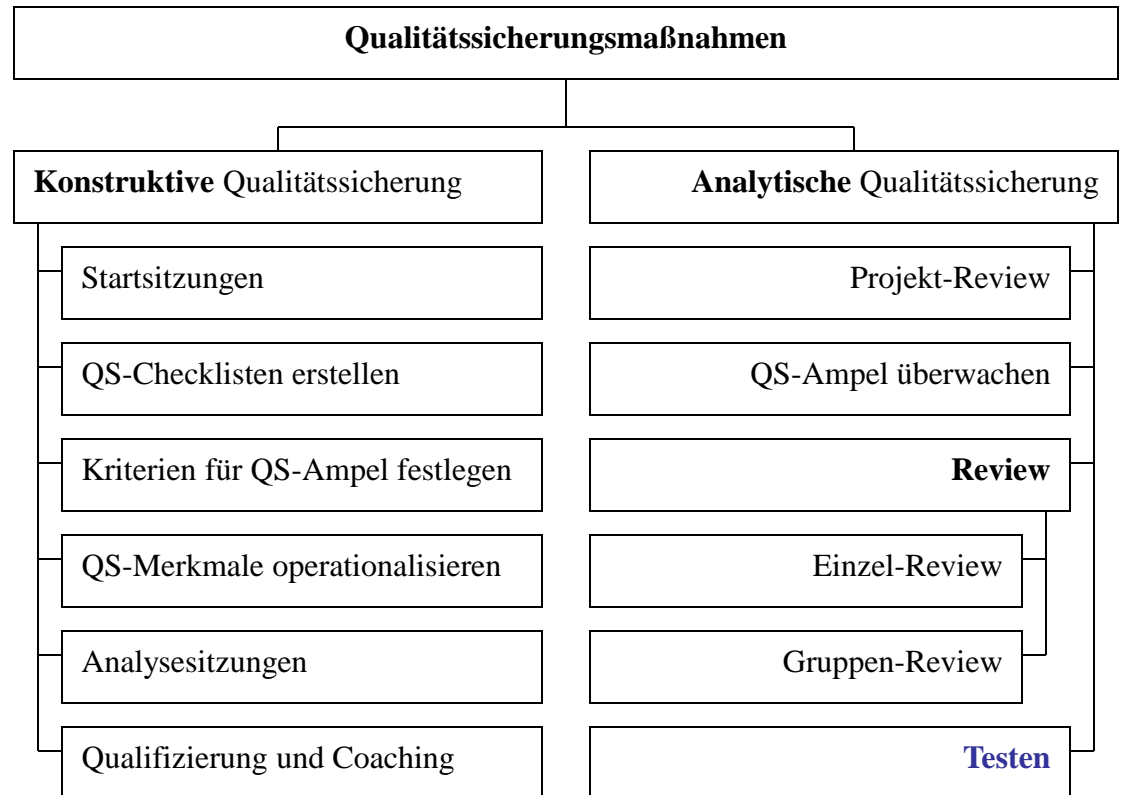
Softwarequalität und -test

QM und QS in der Softwareentwicklung

Qualitätssicherung

Testen

Kommt in der Vorlesung
Testmanagement in
Softwareprojekten.



Qualitätssicherungs- beauftragter

Softwarequalität und -test

QM und QS in der Softwareentwicklung

Qualitätssicherung (Fortsetzung)

Qualitätssicherungsbeauftragter (QSB)

Softwareprojekte stehen meist unter hohem Druck. Auch Projektleiter, die die Bedeutung von Softwarequalität verstanden haben und hoch achten, können **nicht aus Perfektionismus** ihr Projekt aufhalten. Alles, was zum Gelingen des Projekts beiträgt, ist hochwillkommen – und alles andere wird gemieden...

Die Aufgaben **Qualitätsplanung**, **Qualitätslenkung** und **Qualitätsverbesserung** werden deshalb durch den Qualitätssicherungsbeauftragten (QSB), der für die Qualitätssicherung im Projekt verantwortlich ist, wahrgenommen.

Softwarequalität und -test

QM und QS in der Softwareentwicklung

Qualitätssicherung (Fortsetzung)

Qualitätssicherungsbeauftragter (QSB)

Die Rolle des QSB sollte möglichst **nicht von einer Person eingenommen werden**, die schon die Rolle des Projektleiters ausübt.

Wenn beide Rollen von einer einzigen Person übernommen werden, kann es zu Problemen kommen, denn Zielkonflikte zwischen Termin, Funktionalität, Budget und Qualität werden unter Umständen **eher verdeckt und nicht transparent gemacht**.

—

Softwarequalität und -test

QM und QS in der Softwareentwicklung

Qualitätssicherung (Fortsetzung)

Qualitätssicherungsbeauftragter (QSB)

Da der Projektleiter die Gesamtverantwortung für das Projekt hat, sollte der QSB das „Qualitätsgewissen“ des Projektleiters sein.

Der QSB ist nur für die Qualitätssicherungsmaßnahmen verantwortlich, nicht aber für die Qualität selbst!

Softwarequalität und -test

QM und QS in der Softwareentwicklung

Qualitätssicherung (Fortsetzung)

Qualitätssicherungsbeauftragter (QSB)

Aufgaben, Kompetenzen und Verantwortlichkeiten (AKV)

Am wichtigsten:

- Informatik-Hintergrund
- Kommunikationsfähigkeit
- Eigeninitiative
- Pragmatische Einstellung
- Durchsetzungs- und Konfliktfähigkeit

Softwarequalität und -test

QM und QS in der Softwareentwicklung

Qualitätssicherung (Fortsetzung)

Qualitätssicherungsbeauftragter (QSB)

Aufgaben, Kompetenzen und Verantwortlichkeiten (AKV)

Ziemlich wichtig:

- Grundlegende QM-/QS-Techniken (Test, Review)
- Speziellere Techniken (Usability, Qualitätsmodelle)
- Gängige Standards (DIN, ISO, IEEE)

Softwarequalität und -test

QM und QS in der Softwareentwicklung

Qualitätssicherung (Fortsetzung)

Qualitätssicherungsbeauftragter (QSB)

Aufgaben, Kompetenzen und Verantwortlichkeiten (AKV)

Vorteilhaft:

Kenntnis des Anwendungsbereichs

Die QS-Phasen im Vorgehensmodell

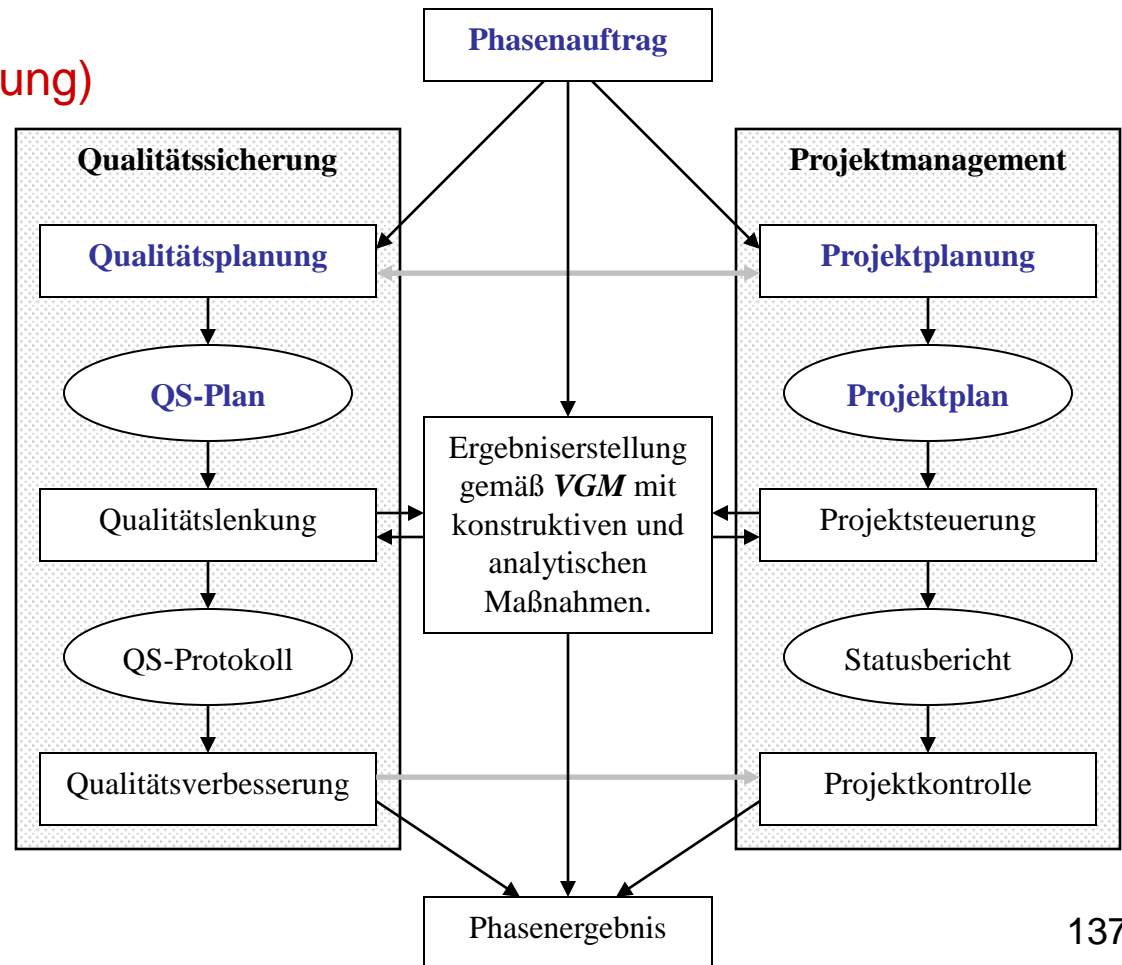
Softwarequalität und -test

QM und QS in der Softwareentwicklung

Qualitätssicherung (Fortsetzung)

Phasen im Vorgehensmodell

Am Beginn jeder Phase wird das Ziel der Phase sowohl unter **Qualitätssicherungs-** als auch unter **Projektmanagement-Aspekten** geplant.



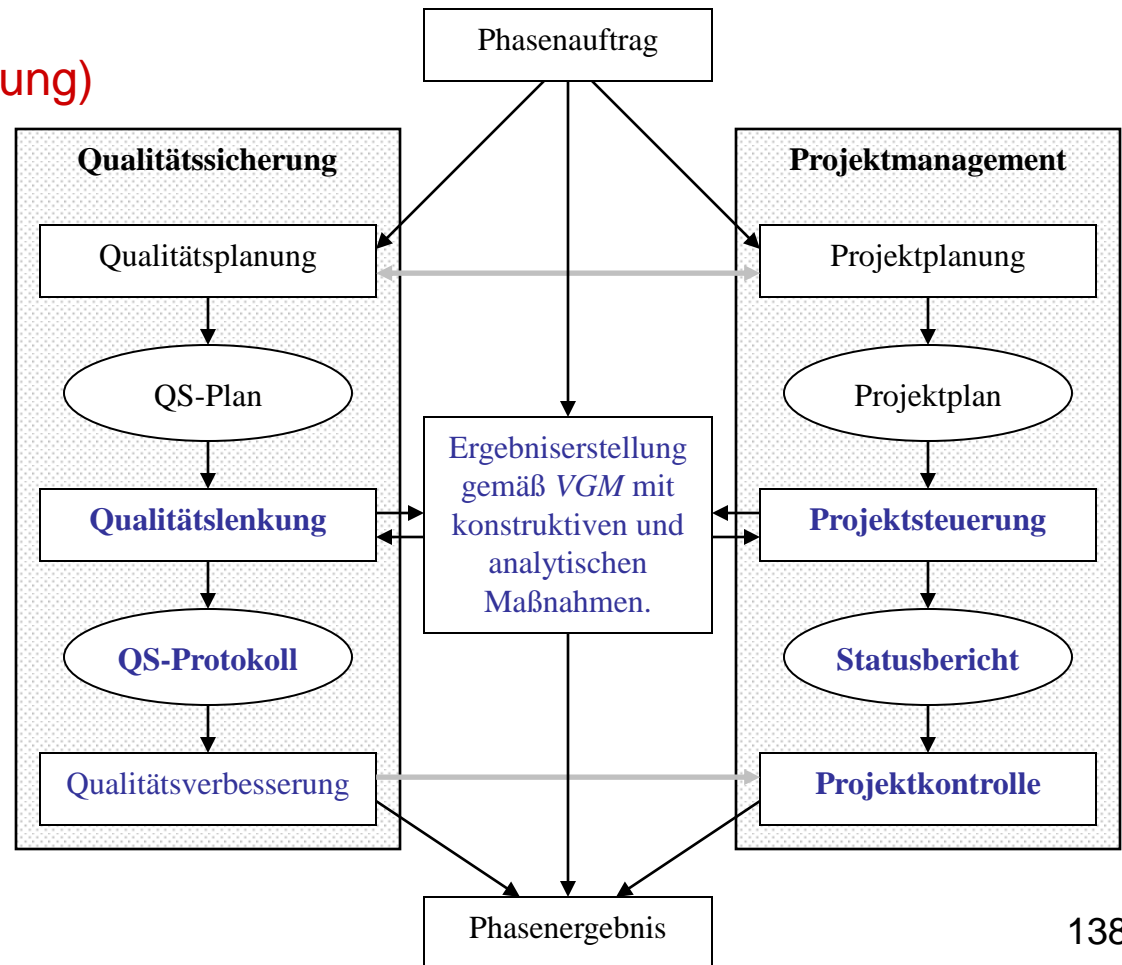
Softwarequalität und -test

QM und QS in der Softwareentwicklung

Qualitätssicherung (Fortsetzung)

Phasen im Vorgehensmodell

Während der eigentlichen Phasendurchführung (der Erstellung der im Vorgehensmodell festgelegten Ergebnisse) sind geeignete Steuerungsmaßnahmen (laufende Überprüfung und Korrekturmaßnahmen) zu ergreifen.



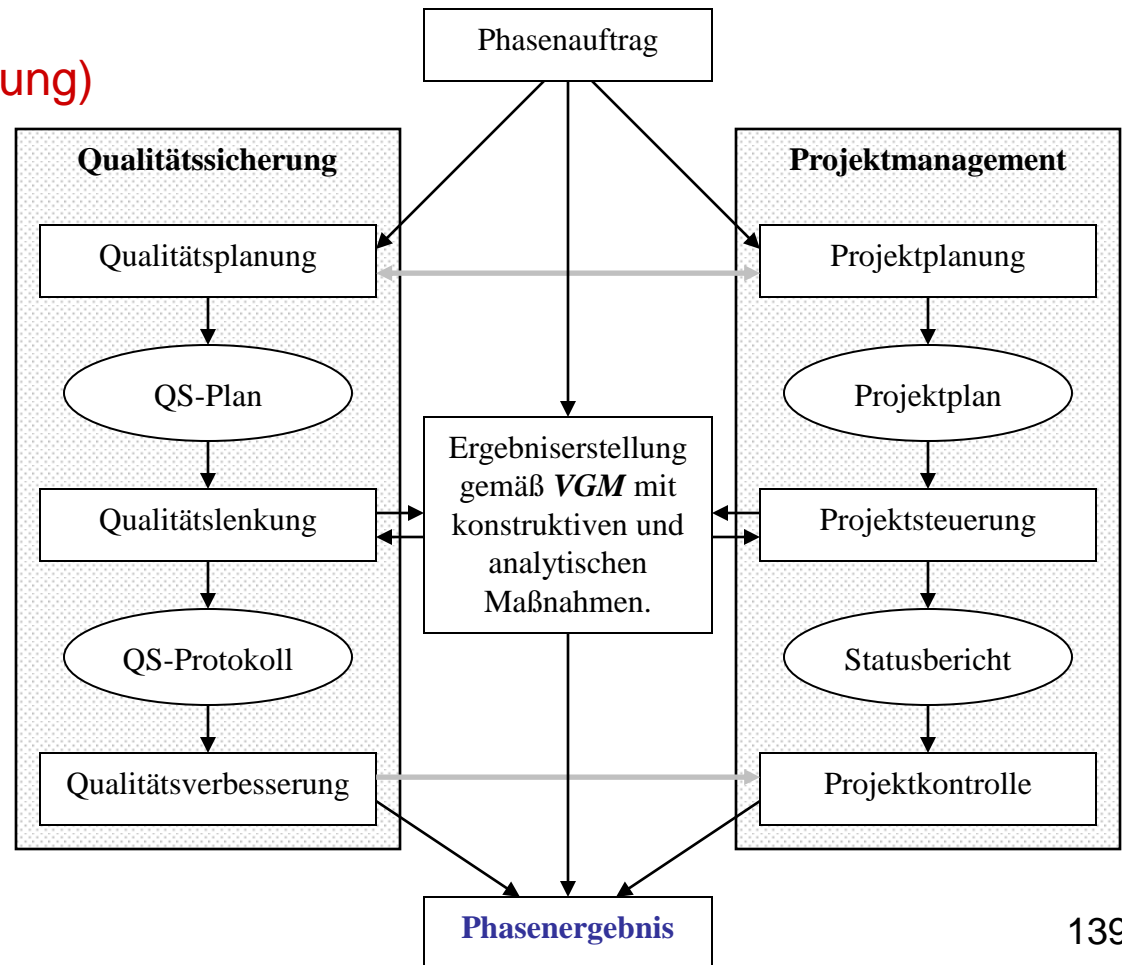
Softwarequalität und -test

QM und QS in der Softwareentwicklung

Qualitätssicherung (Fortsetzung)

Phasen im Vorgehensmodell

Eine Projektphase gilt erst dann als beendet, wenn die geforderten Qualitätskriterien erfüllt wurden.



Normen für Qualitätsmanagement

Softwarequalität und -test

QM und QS in der Softwareentwicklung

Normen für Qualitätsmanagement

Bezeichnung	Titel
DIN EN ISO 9000	Normen zum Qualitätsmanagement und zur Qualitätssicherung/ QM-Darlegung
Teil 1	Leitfaden zur Auswahl und Anwendung
Teil 2	Allgemeiner Leitfaden zur Anwendung von ISO 9001, ISO 9002 und ISO 9003 (Entwurf)
Teil 3	<i>Leitfaden für die Anwendung von ISO 9001 auf die Entwicklung, Lieferung und Wartung von Software</i>
DIN EN ISO 9001	<i>Qualitätsmanagementsysteme</i> – Modell zur Qualitätssicherung/ QM-Darlegung in Design, Entwicklung, Produktion, Montage und Wartung
DIN EN ISO 9002	Qualitätsmanagementsysteme – Modell zur Qualitätssicherung/ QM-Darlegung in Produktion, Montage und Wartung
DIN EN ISO 9003	Qualitätsmanagementsysteme – Modell zur Qualitätssicherung/ QM-Darlegung bei der Endprüfung

Softwarequalität und -test

QM und QS in der Softwareentwicklung

Normen für Qualitätsmanagement (Fortsetzung)

Bezeichnung	Titel
DIN ISO 8402	Qualitätsmanagement und Qualitätssicherung - Begriffe (Entwurf)
DIN EN ISO 9004 Teil 1 Teil 2 Teil 7	Qualitätsmanagement und Elemente eines QM-Systems Leitfaden Leitfaden für Dienstleistungen Leitfaden für Konfigurationsmanagement (Entwurf)
DIN ISO 10 011 Teil 1 Teil 2 Teil 3	Leitfaden für das <u>Audit</u> von QM-Systemen <u>Auditdurchführung</u> Qualifikationskriterien für <u>Qualitätsauditoren</u> Management von <u>Auditprogrammen</u>
DIN ISO 10 013	Leitfaden für die <i>Erstellung</i> von <i>Qualitätsmanagement - Handbüchern</i>

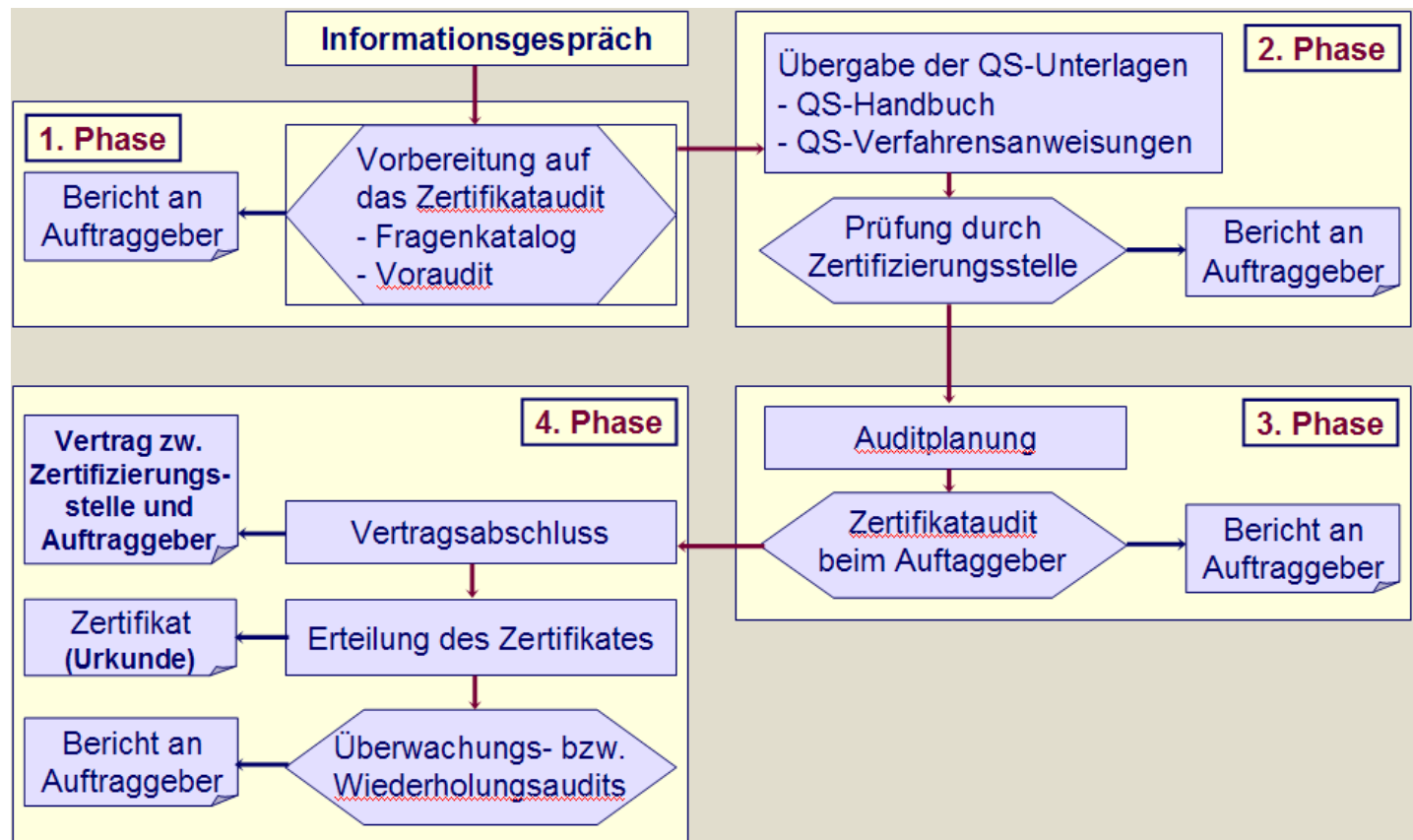
D. Macos nach K. Bothe, Software Engineering

Softwarequalität und -test

QM und QS in der Softwareentwicklung

Normen für
Qualitätsmanagement
(Fortsetzung)

Zertifizierung
nach ISO 9000

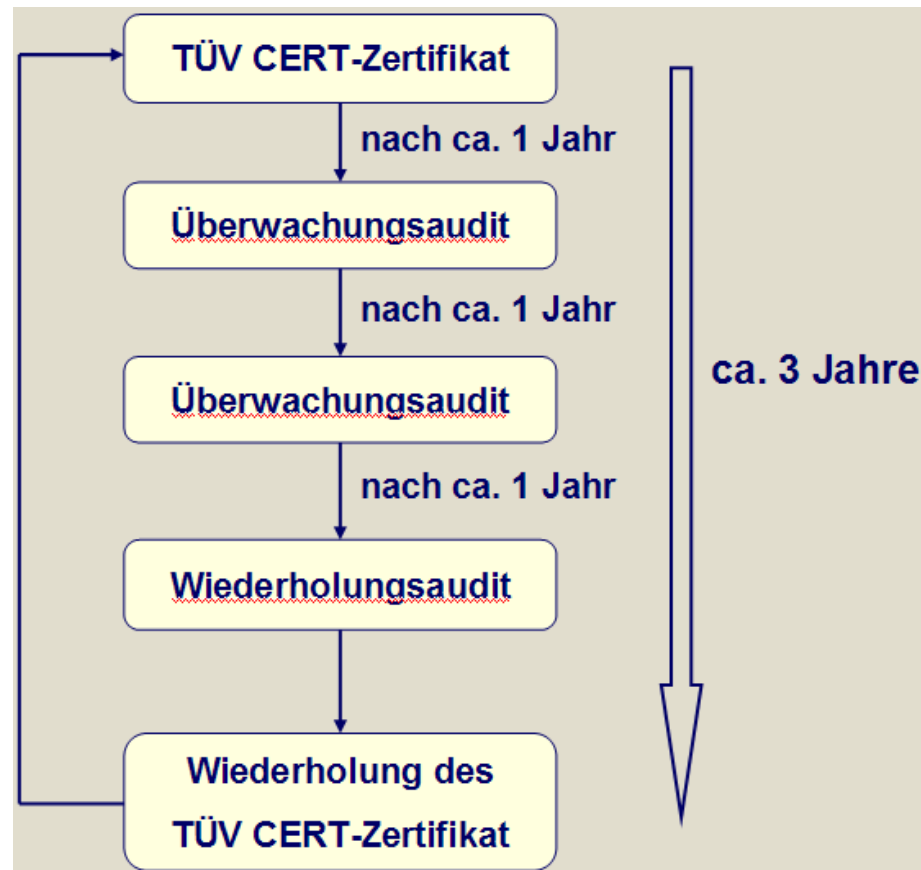


Softwarequalität und -test

QM und QS in der Softwareentwicklung

Normen für
Qualitätsmanagement
(Fortsetzung)

Zertifizierung
nach *ISO 9000*



Softwarequalität und -test

QM und QS in der Softwareentwicklung

Qualitätssicherung (Fortsetzung)

Puh, fertig...