UNIVERSITÄT LEIPZIG
Institut für Informatik
Betriebliche Informationssysteme

Software-Qualitätsmanagement

Kernfach Angewandte Informatik

Sommersemester 2006

Prof. Dr. Hans-Gert Gräbe

UNIVERSITÄT LEIPZIG
Institut für Informatik
Betriebliche Informationssysteme

1. Aufgaben im Qualitätsmanagement
2. Konstruktive und analytische Maßnahmen
3. Aktivitäten im Qualitätsmanagement
4. Prinzipien der Software-Qualitätssicherung
5. Beispiel: Qualitätssicherung im V-Modell

UNIVERSITÄT LEIPZIG
Institut für Informatik
Betriebliche Informationssysteme

Qualitätsmanagement umfasst alle Tätigkeiten der Gesamtführungsaufgabe, welche die Qualitätspolitik, Ziele und Verantwortlichkeiten festlegt sowie diese durch Mittel wie Qualitätsplanung, Qualitätslenkung, Qualitätssicherung und Qualitätsverbesserung im Rahmen des Qualitätsmanagementsystems verwirklichen.

[DIN ISO 8402]

- **Q.-Planung**: Vorbereitende Maßnahmen
- **Q.-Sicherung**: Begleitende Maßnahmen mit
  - **Q.-Lenkung**: administrative Maßnahmen
  - **Q.-Prüfung**: diagnostische Maßnahmen

sowie

- **Q.-Verbesserung**: Prozess-strukturelle Maßnahmen

UNIVERSITÄT LEIPZIG
Institut für Informatik
Betriebliche Informationssysteme

Produktorientiertes Q.-Management

- Produkte und Zwischenergebnisse werden auf vorher festgelegte Qualitätsmerkmale überprüft
  - Qualität wird im Nachhinein festgestellt
  - Gütebedingungen und Prüfbestimmungen
  - eher im Bereich der Komponentensoftware und Standardsoftware mit konstanten Q.-Anforderungen

- **Grundansatz**: Qualität als messbare Größe des Produkts
  - Qualität kann durch Zertifikat (Prüfung durch unabhängige Seite) bestätigt werden
  - Relevante Bestimmungen: ISO 9126
- **Kontext**: analytische und konstruktive QS-Maßnahmen

UNIVERSITÄT LEIPZIG
Institut für Informatik
Betriebliche Informationssysteme

Prozessorientiertes Q.-Management

- Gerichtet auf den Erstellungsprozess der Software selbst
  - eher für Firmen, die anwenderspezifische Spezialsoftware herstellen, mit variierenden Q.-Anforderungen und dynamischem Qualitätsoptimum
  - Ziel ist die Herausbildung eines Qualitätsbewusstseins bei den Mitarbeitern

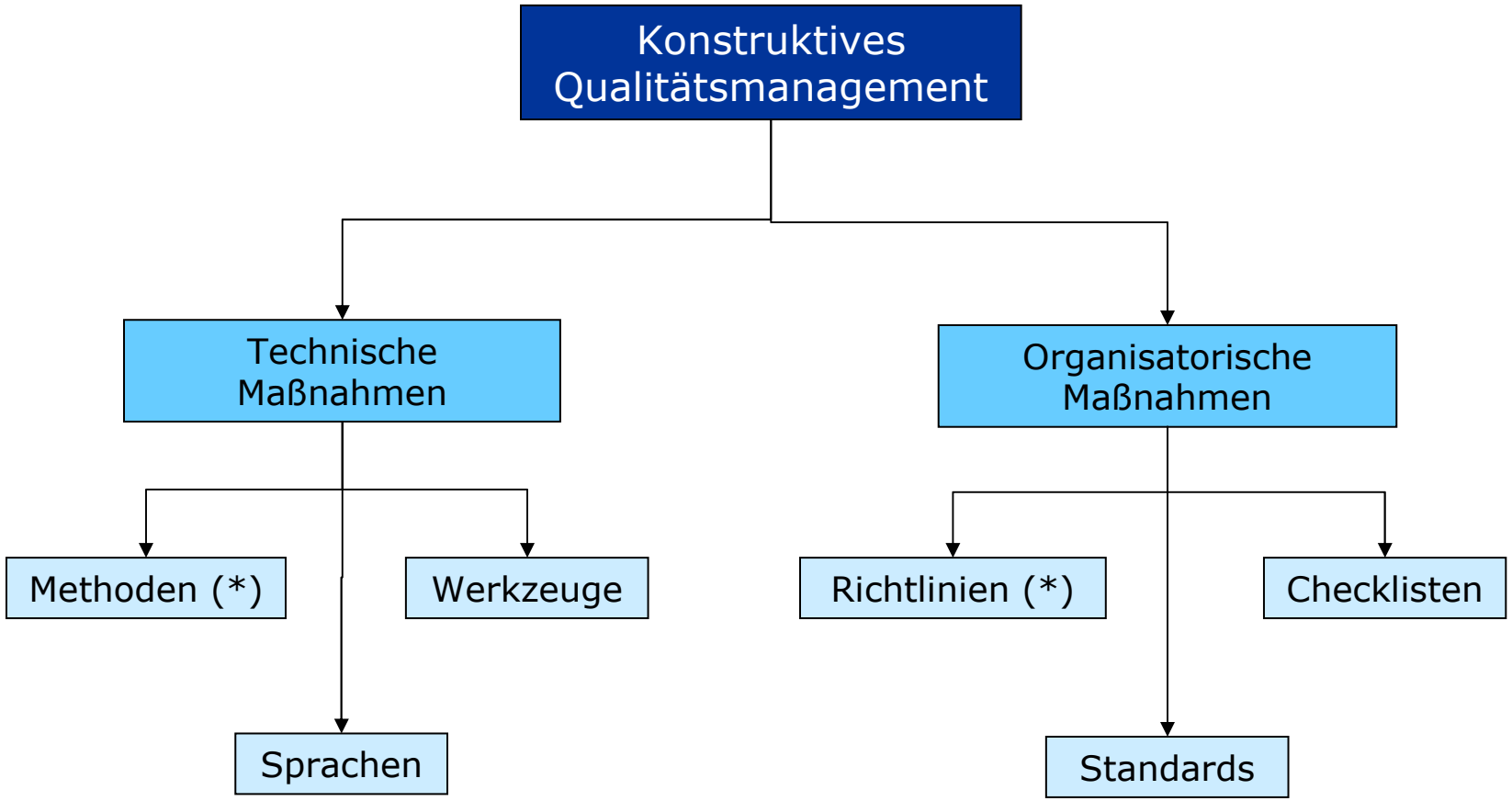
- **Grundansatz**: Qualität durch den Erstellungsprozess selbst
- **Faktoren**: Planbarkeit, Effizienz (im Kosten/Nutzen-Sinn), Produktqualität
- **Kontext**: Prozesszertifizierung, Prozessverbesserung

UNIVERSITÄT LEIPZIG
Institut für Informatik
Betriebliche Informationssysteme

Konstruktive Maßnahmen

- Vorgabe von Konstruktionstechniken und Richtlinien
  - strukturiertes Vorgehen
  - werkzeuggestützte Entwicklung
  - höhere Programmiersprachen
- Vorteile:
  - Erfahrungen projektübergreifend sammeln und nutzen
  - Aufwertung der Planungsaktivitäten in frühen Projektphasen
  - Werkzeugunterstützung
- Nutzen:
  - Steigerung der Qualität um bis zu 50 %
  - Steigerung der Produktivität um bis zu 30 %

Konstruktive Maßnahmen sorgen durch Einschränkung der Variabilität in der Systementwicklung von vornherein dafür, dass gewisse Fehler nicht auftreten können und damit ein gewisses Maß an Qualität per se erreicht wird.


Konstruktives Qualitätsmanagement
Technische Maßnahmen
Organisatorische Maßnahmen
Methoden (*)
Werkzeuge
Sprachen
Richtlinien (*)
Checklisten
Standards

Konstruktive Verfahren: Methoden

- Ziel: strukturierte Vorgehensweise
- Technik: Vorgabe von Zwischenprodukten
  - Vorgabe von Modellen (Bsp.: objektorientiert)
  - Vorgabe von Einzelschritten (Bsp.: Anwendungsfall-Modellierung)
  - Vorgabe von Erstellungsmitteln (Bsp.: Klassendiagramm, Anwendungsfall-Diagramm)
- Vorteile:
  - Strukturierung unterstützt gute Granularität, Änderbarkeit
  - Werkzeugunterstützung

UNIVERSITÄT LEIPZIG
Institut für Informatik
Betriebliche Informationssysteme

Konstruktive Verfahren: Richtlinien

- Ziel: Produkteigenschaften a-priori festlegen
- Technik:
  - Vorgabe von Checklisten, Schablonen
  - Überprüfung der Richtlinien
- Beispiele:
  - Strukturierung der Analyse durch SCR-Tabellen
  - Anwendung von Design Pattern
  - Einsatz von Coding Standards
- Vorteile:
  - Erfahrungen mit Richtlinien werden projektübergreifend wirksam
  - Unterstützung durch Werkzeuge und Vorlagen

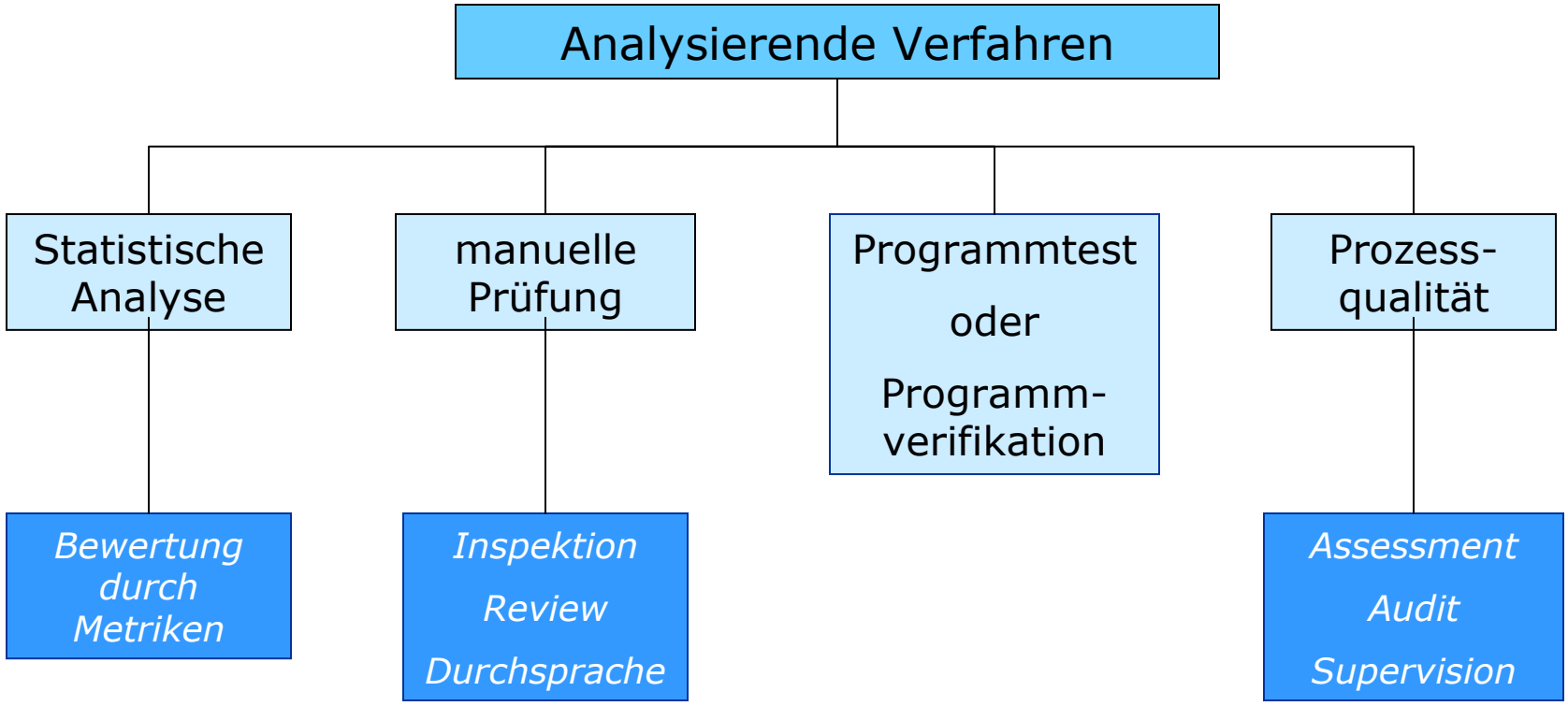
Analytische Maßnahmen

- diagnostische Maßnahmen, bringen keine Qualität per se
- sind zur Messung der Qualität der End- bzw. Zwischenprodukte
- Gliederung nach verschiedenen Gesichtspunkten:
  - Bezug der Prüfung (Produkt oder Prozess)
  - Automatisierungsgrad der Prüfung (manuell / mit Werkzeug)
  - Nachvollziehbarkeit der Prüfung (Selbstprüfung / Nachweis)
  - Einsatzbereich der Prüfung (in welcher Phase des SW-Zyklus)

Analytische Maßnahmen dienen zur Datenerhebung, um Ist- und Soll-Zustand zu vergleichen und so den Grad der erreichten Qualität im Nachhinein festzustellen.

UNIVERSITÄT LEIPZIG
Institut für Informatik
Betriebliche Informationssysteme

Analysierende Verfahren sammeln gezielt Informationen über den Prüfling mit analytischen Mitteln.



Produktqualität: Statistische Analyse

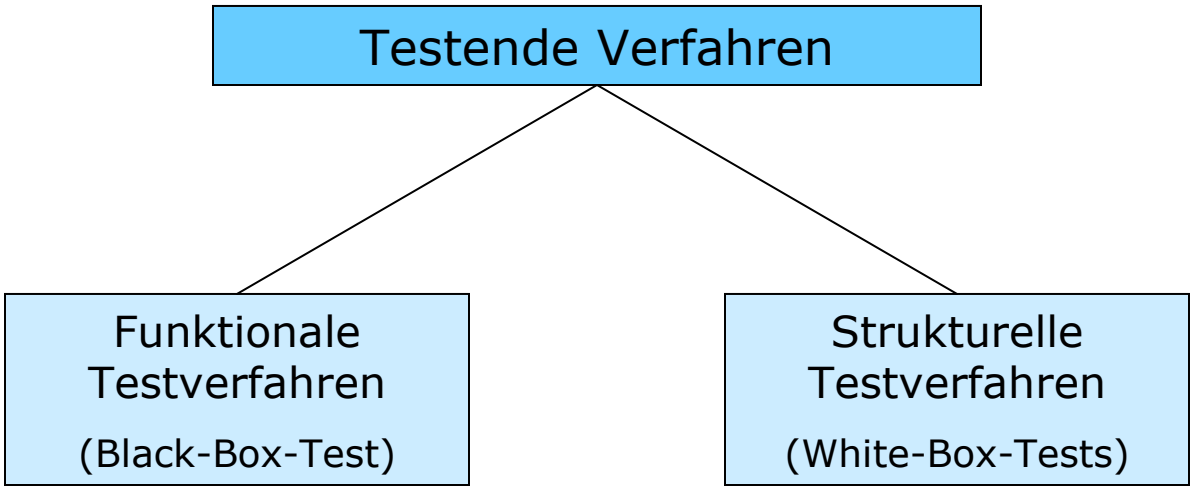
- **Ziel**: **Bewertung** eines Produkts (Entwurfsdokuments, Grob/Fein-Entwurf, Code, Designdokumentation usw.) mittels **Metriken**
- **Schwerpunkt:** Zuverlässigkeit, Änderbarkeit

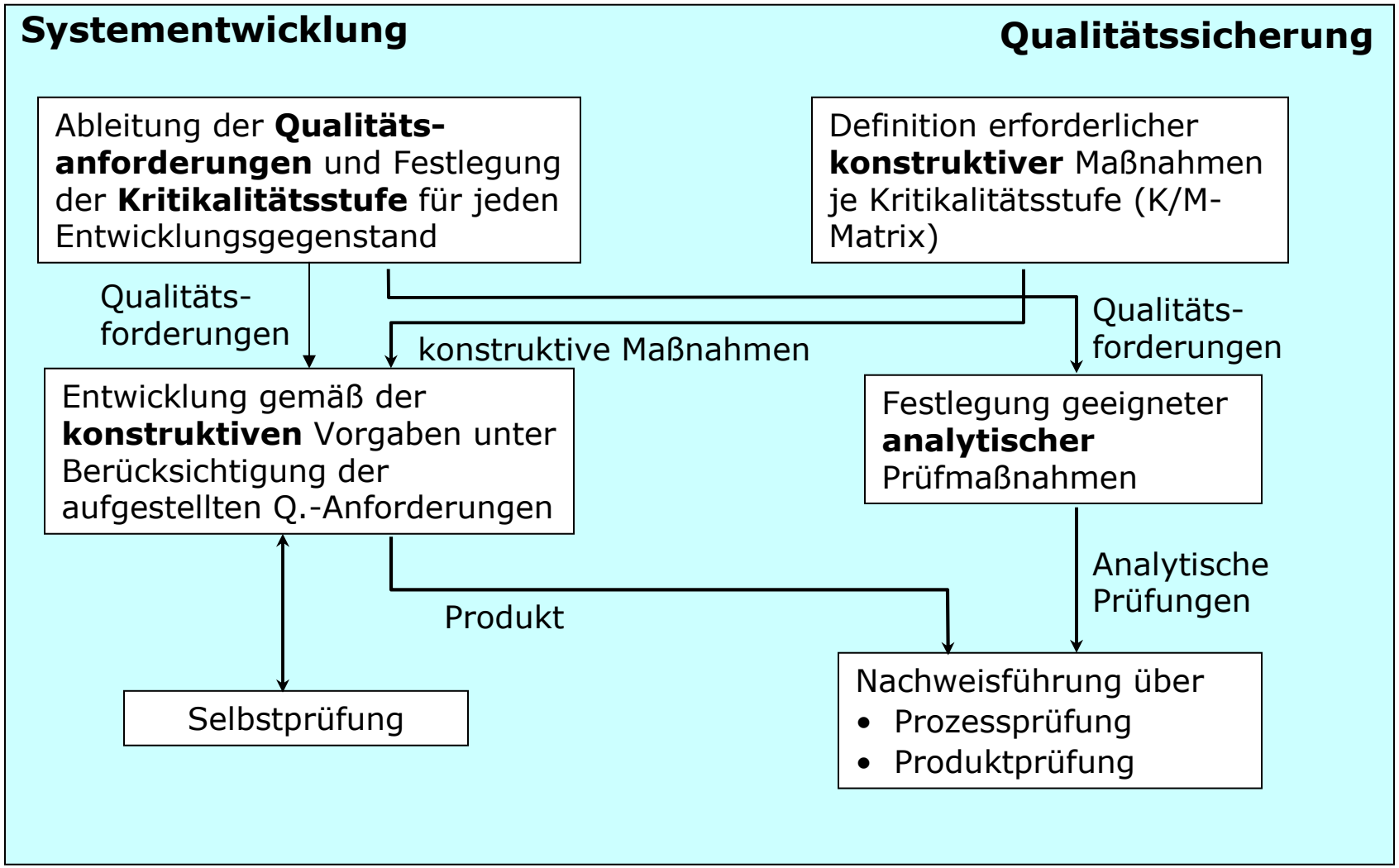
Einsatzgebiet	Kriterium	Metrik
Komponenten-analyse	Umfang	lines of code
	innere Struktur	Kontrollfluss-komplexität
	Schnittstelle	# Methoden pro Klasse Schnittstellenbreite

Einsatzgebiet	Kriterium	Metrik
Systemanalyse	Umfang	lines of code
	Kopplung	# Aufrufe in/aus Komponenten
	OO-Strukturierung	OO-Metriken
Prozessanalyse	Aufwandsoptimierung	Zeiterfassung
	Dokumentenqualität	entdeckte Fehler pro Seite
	Prüfprozessqualität	# vorab gefundener Fehler / # in der Sitzung gefundener Fehler

Produktqualität: Testende Verfahren

Diese prüfen das Verhalten des Prüfling bei konkreten Eingaben.




Systementwicklung
Qualitätssicherung
Ableitung der **Qualitäts-anforderungen** und Festlegung der **Kritikalitätsstufe** für jeden Entwicklungsgegenstand
Definition erforderlicher **konstruktiver** Maßnahmen je Kritikalitätsstufe (K/M-Matrix)
Qualitäts-forderungen
konstruktive Maßnahmen
Qualitäts-forderungen
Entwicklung gemäß der **konstruktiven** Vorgaben unter Berücksichtigung der aufgestellten Q.-Anforderungen
Festlegung geeigneter **analytischer** Prüfmaßnahmen
Produkt
Analytische Prüfungen
Selbstprüfung
Nachweisführung über
• Prozessprüfung
• Produktprüfung

UNIVERSITÄT LEIPZIG
Institut für Informatik
Betriebliche Informationssysteme

Qualitätssicherung und Systementwicklung

- konstruktive Maßnahmen sind im Rahmen der Anforderungsanalyse und vor Beginn des Entwicklungsprozesses zu fixieren
- analytische Maßnahmen werden Entwicklungsprozess begleitend oder zu Meilensteinen, mit denen Entwicklungsprozess-Etappen abgeschlossen werden, wirksam
- analytische Maßnahmen können konstruktive Vorgaben erfordern, um die Produktion der zu analysierenden Daten zu initiieren.

Analytische und konstruktive QM-Maßnahmen beeinflussen sich gegenseitig: Vorausschauende konstruktive Planung erspart analytischen Aufwand

Qualitätsplanung und Qualitätssicherung

Qualitätsmanagement besteht aus den Phasen

- **Qualitätsplanung**
  - vorbereitende Aktivitäten zur Festlegung von Standards, zu erreichender Parameter und von Verantwortlichkeiten
  - Ergebnis: Qualitätssicherungsplan und Prüfplan

- **Qualitätssicherung**
  - Systementwicklung begleitende Aktivitäten zur Umsetzung und Dokumentierung der erreichten Qualitätsparameter
  - Ergebnis: Protokolle, Zertifikate, Vorschläge für die Projektsteuerung
  - **Qualitätslenkung** und **Qualitätsprüfung**

UNIVERSITÄT LEIPZIG
Institut für Informatik
Betriebliche Informationssysteme

Qualitätsplanung

- **Qualitätszielbestimmung**: Festlegung von Qualitätsanforderungen an den Prozess und an das Produkt in überprüfbarer Form.
- **Qualitätslenkung**: Umsetzung, Steuerung, Überwachung und Korrektur des Entwicklungsprozesses mit dem Ziel, die vorgegebenen Anforderungen zu erfüllen.
- **Qualitätsprüfung**: Durchführung der im Rahmen der Qualitätsplanung festgelegten Maßnahmen zur
  - Erfassung von Istwerten der Qualitäts-Indikatoren
  - Überwachung der Umsetzung der konstruktiven Maßnahmen
  - Tests, Reviews, Audits, Inspektionen
- **Qualitätsverbesserung**: Auswertung der Qualitätssicherungs-Ergebnisse und Prozessverbesserung.
  - Mängel- und Fehleranalyse (Verbesserung der Prozessqualität)

UNIVERSITÄT LEIPZIG
Institut für Informatik
Betriebliche Informationssysteme

Qualitätssicherungsplan und Prüfplan

- Ergebnisse der Qualitätsplanung werden in einem **Qualitäts-Sicherungsplan** dokumentiert (prozess-orientiert), die begleitenden Maßnahmen in einem **Prüfplan** festgelegt (produkt-orientiert).
  - **Festlegung der Aufgaben**
    - Was ist zu tun?
    - Identifizierung der zu sichernden Produkte
    - Identifizierung der relevanten Qualitätsmerkmale, ihre relative Bedeutung und ihre Quantifizierung in Form von Metriken
  - **Festlegung der Vorgaben und Hilfsmittel**
    - Wie ist es zu tun?
    - Auswahl der zur Datenerfassung und Qualitätsprüfung geeigneten Techniken und Methoden
    - konstruktive Vorgaben (etwa Richtlinien, Vorlagen)
    - analytische Vorgaben (Verfahren, Werkzeuge)

Qualitätssicherungsplan und Prüfplan

- **Festlegung der Termine**
  - (Bis) wann ist es zu tun?
  - Festlegung der Zeitpunkte für die den gesamten Entwicklungsprozess begleitende Datenerfassung
  - Einordnung des Prüfplans in den Projektplan
- **Festlegung der Verantwortlichkeiten**
  - Wer hat es zu tun?
  - Festlegung der Verantwortlichkeiten für die Qualitätsprüfung und -lenkung.
  - Definition und Besetzung von Rollen (Q-Manager, Prüfer, Autor, Gutachter)

- Gliederungsschema für Qualitätssicherungspläne ist im IEEE-Standard 730-1984 festgelegt.

UNIVERSITÄT LEIPZIG
Institut für Informatik
Betriebliche Informationssysteme

Grundsätze für die Qualitätssicherung in der Software-Entwicklung

- produkt- und prozessabhängige Qualitätszielbestimmung
- quantitative Qualitätssicherung
- maximale konstruktive Qualitätssicherung
- frühzeitige Fehlerentdeckung und -behebung
- entwicklungsbegleitende, integrierte Qualitätssicherung
- unabhängige Qualitätssicherung

Prinzip der produkt- und prozessabhängigen Qualitätszielbestimmung

- Nur 50% der Betriebe legen Qualitätsmerkmale fest [Spillner et al. 94].
  - Qualitätsmerkmale mit hoher Priorität sind Robustheit, Verständlichkeit, Wartbarkeit und Laufzeiteffizienz.
  - Erst in zweiter Linie werden Korrektheit, Vollständigkeit und Benutzungsfreundlichkeit festgeschrieben.
- explizite und transparente Qualitätszielbestimmung ist vor Beginn des Entwicklungsprozesses äußerst hilfreich.
  - Die in der Qualitätszielbestimmung festgelegten Qualitätsanforderungen werden vom Auftraggeber für den Abnahmetest verwendet.
  - Für den Software-Lieferanten ergeben sich aus den Qualitäts-anforderungen die Maßnahmen für den Entwicklungsprozess und die Qualitätsprüfung.
- Prinzip der produkt- und prozessabhängigen Qualitätszielbestimmung vor Beginn des Entwicklungsprozesses bringt Planungs- und Kalkulations-sicherheit.

UNIVERSITÄT LEIPZIG
Institut für Informatik
Betriebliche Informationssysteme

Prinzip der quantitativen Qualitätssicherung

- Schwierigkeiten bei der Quantifizierung von Soll- und Istwerten:
  - Metriken sind ziel- und kontextabhängig
  - Anzahl der Variationsparameter ist um ein Vielfaches höher als bei traditionellen Produktionsprozessen
  - kreativer Charakter vieler Aspekte der Software-Entwicklung
  - unkontrollierte Variabilität von Entwicklungsprozessen
- **Vorteile:**
  - Messen ist geeignet
    - o zum besseren Verständnis unterschiedlicher Qualitätsmerkmale
    - o zur besseren Planung und Sicherung von Qualitätsmerkmalen
    - o zur Verbesserung von Entwicklungsansätzen
  - Methoden und Werkzeuge zur Planung und Durchführung der Datenerfassung sind schon vorhanden.
  - Auch zur Auswertung und Präsentation von Messdaten können vorhandene Werkzeuge verwendet werden.

Prinzip der frühzeitigen Fehlerentdeckung und –behebung

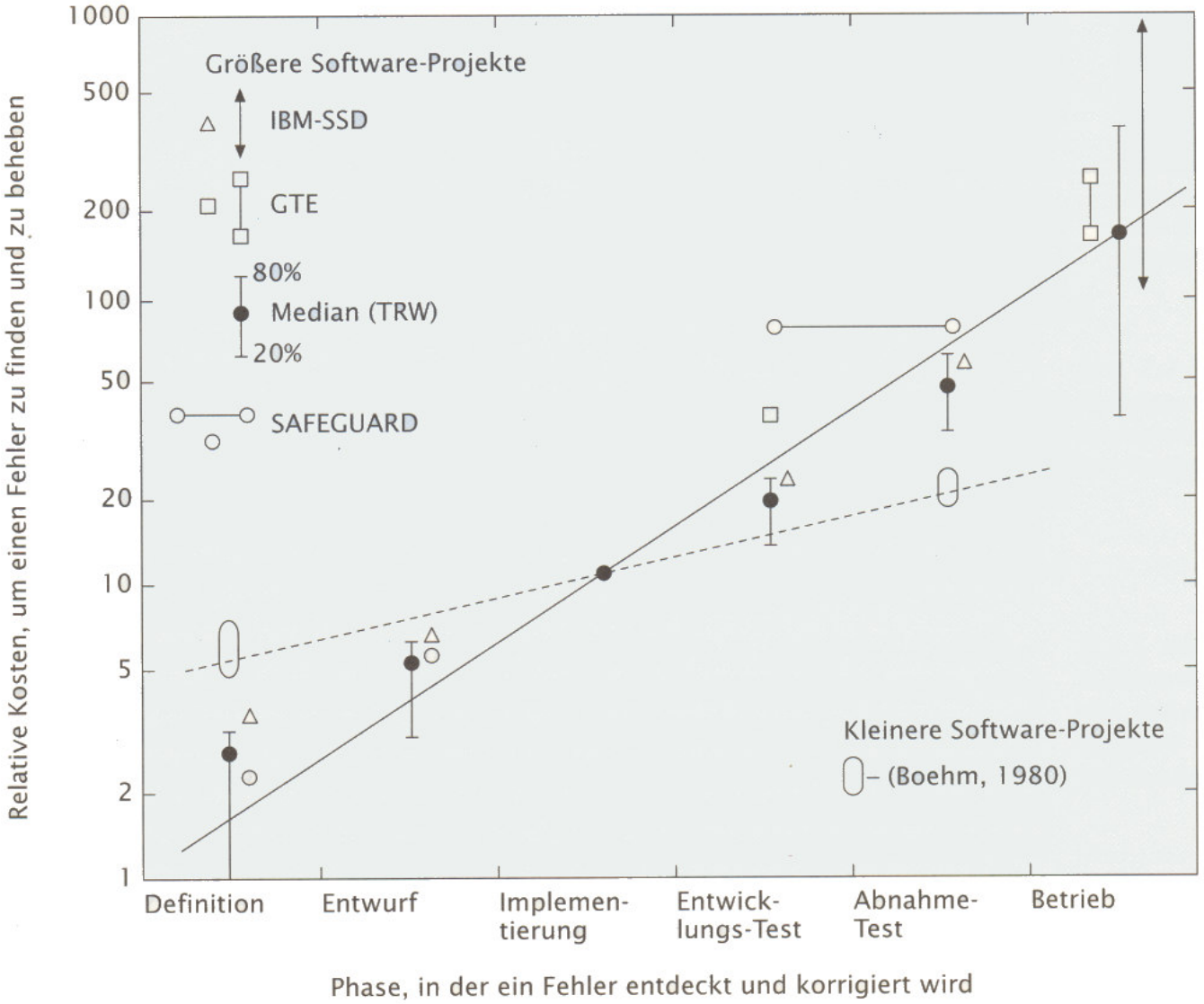
- Fehler ist:
  - Abweichung von den Anforderungen des Auftraggebers
  - Inkonsistenz in den Anforderungen

- Ziel ist es, Fehler gar nicht erst zu machen (konstr. Maßnahmen) oder zum frühestmöglichen Zeitpunkt zu erkennen und zu beheben

- **Vorteile:**
  - Fehler in späteren Phasen werden vermieden
  - Kosten werden reduziert
  - mit höherer Wahrscheinlichkeit werden Fehler richtig korrigiert
  - die Fehlerfortpflanzung wird reduziert

- Folgerung: Viel Aufmerksamkeit den frühen Projektphasen

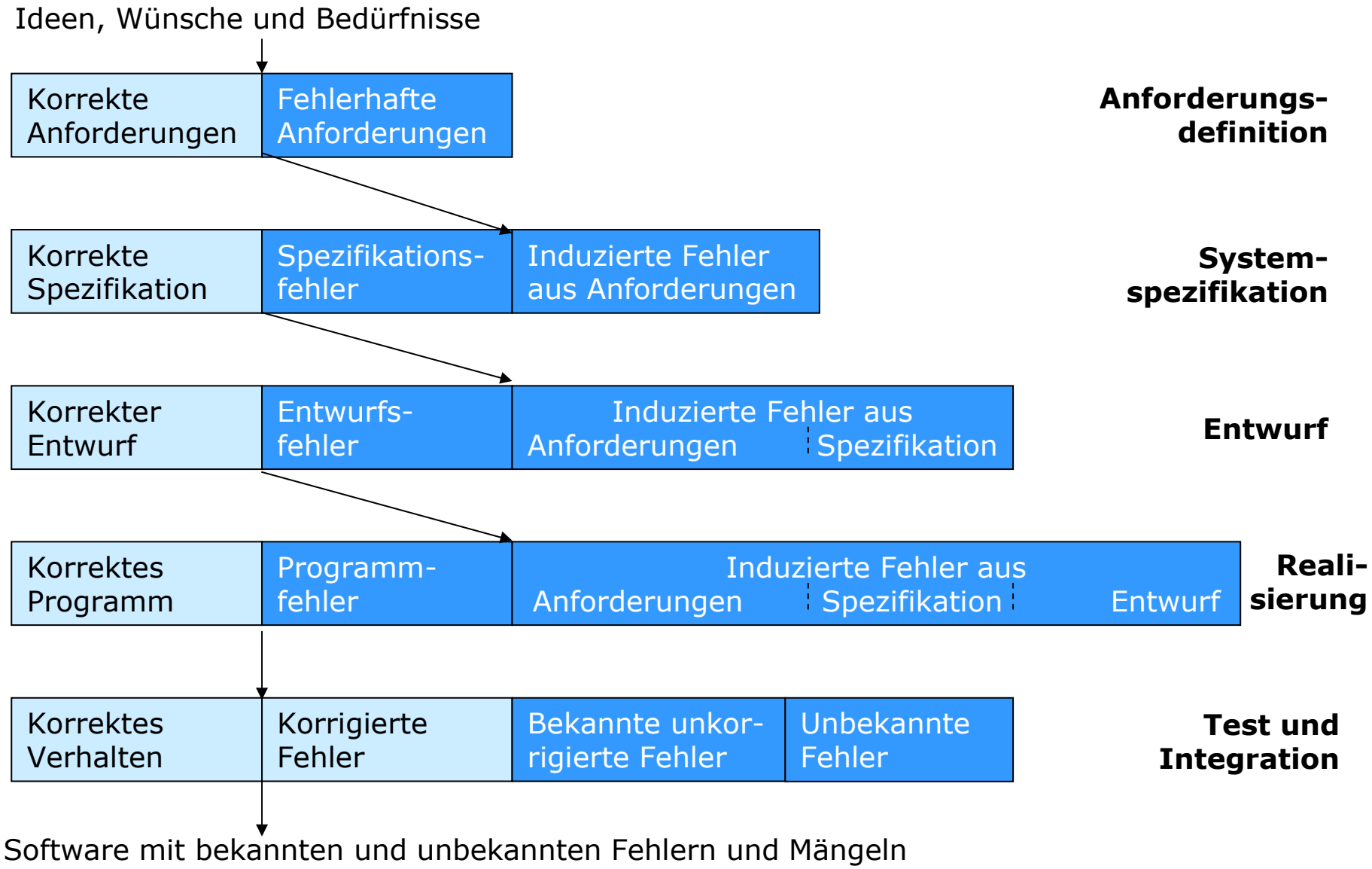
Prinzip der frühzeitigen Fehlerentdeckung



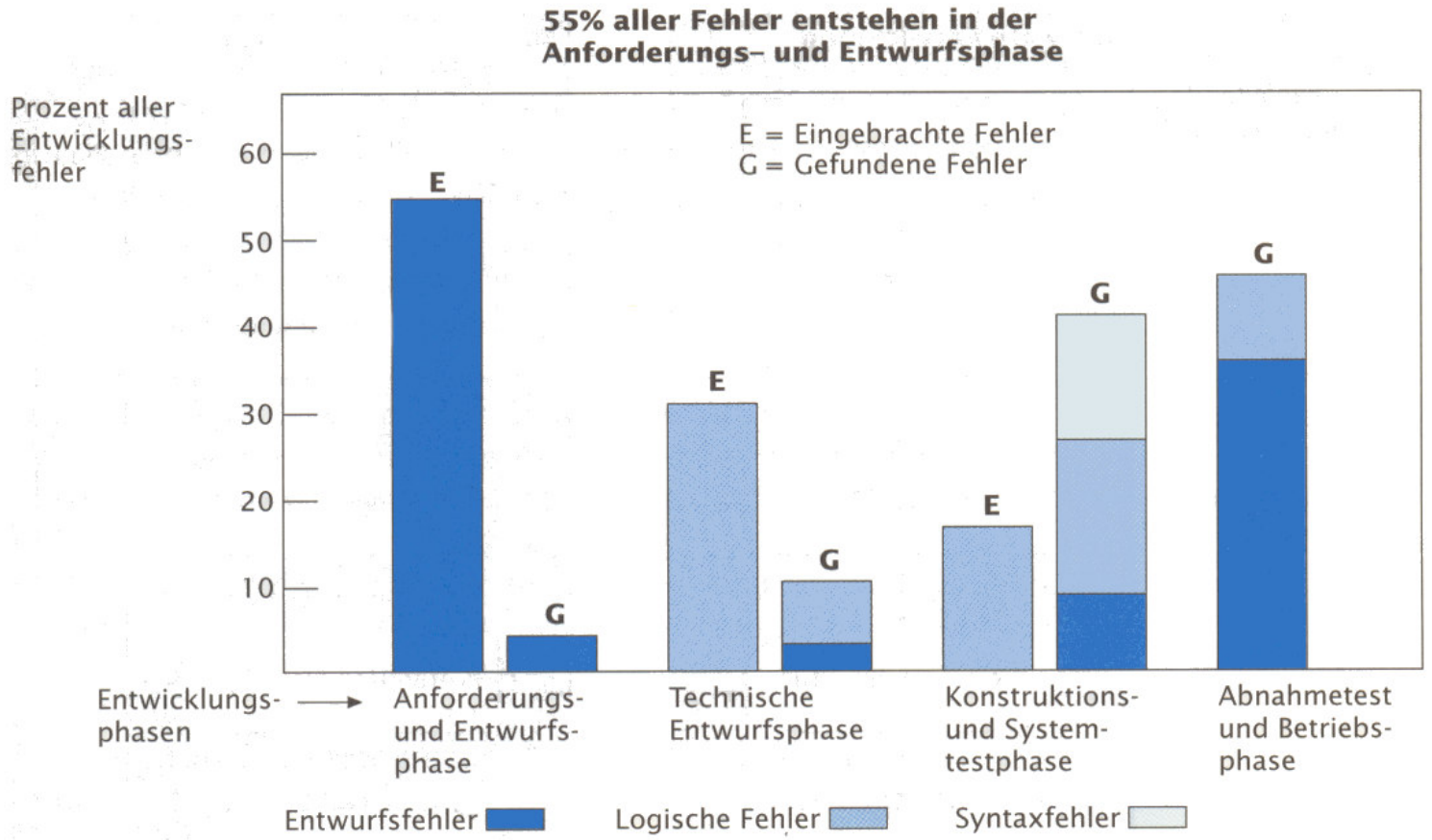
Kosten einer verzögerten Fehlerentdeckung [Boehm 76]

Folgen später Fehlerentdeckung

UNIVERSITÄT LEIPZIG
Institut für Informatik
Betriebliche Informationssysteme



Fehlerbeseitigungskosten



Fehlerbeseitigungskosten (abgeleitet von Alberts 1985)

Prinzip der maximalen konstruktiven Qualitätssicherung

- „Fehler die nicht gemacht werden, brauchen auch nicht behoben werden" ist das Ziel, das von der maximalen konstruktiven Qualitätssicherung verfolgt wird.

- Reduzierung der analytischen Maßnahmen über Einschränkung der Variationsbreite durch vorausblickende konstruktive Maßnahmen.

- **Vorteile:**
  - direkte Verbesserung der Produktivität
  - Reduktion analytischer Maßnahmen
  - Voraussetzung für analytische Maßnahmen
  - Vermeidung von Fehlern

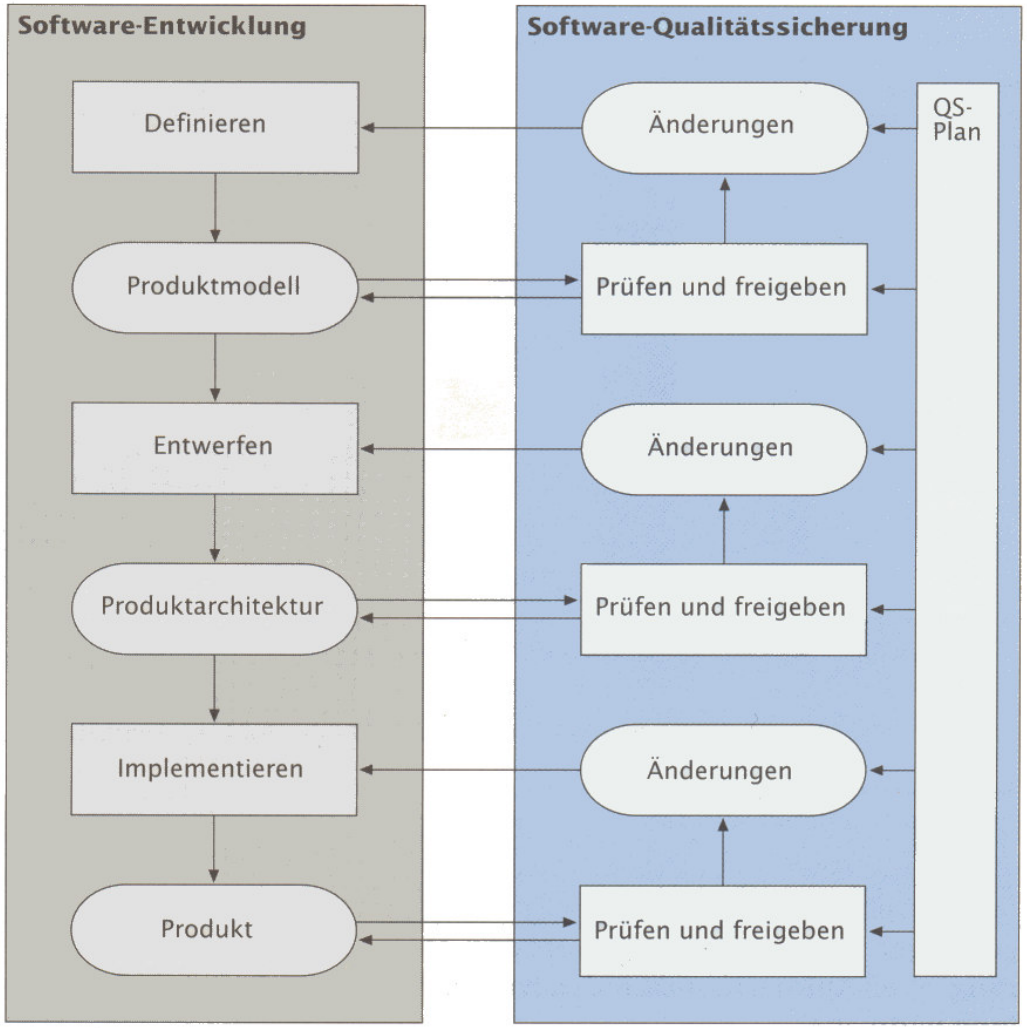
Prinzip der entwicklungsbegleitenden, integrierten Qualitätssicherung

- Um das Prinzip der frühzeitigen Fehlerentdeckung zu realisieren, ist Softwareentwicklung begleitende und in den Entwicklungsprozess integrierte Qualitätssicherung nötig.

- **Vorteile:**
  - Einbettung der Qualitätssicherung in das organisatorische Ablaufmodell der Software-Entwicklung
  - Qualitätssicherung findet zu dem Zeitpunkt statt, zu dem sie im Entwicklungsprozess angebracht ist
  - Qualitätssicherung wird nicht als Fremdkörper empfunden, sondern gehört per se zur Software-Erstellung

Prinzip der integrierten Qualitätssicherung

UNIVERSITÄT LEIPZIG
Institut für Informatik
Betriebliche Informationssysteme



Prinzip der unabhängigen Qualitätssicherung

*„... Testing is a **destructive** process, even a sadistic process ...“* [Myers 79]

Vorteile:

- psychologische
  - Der Entwickler eines Produkts ist am schlechtesten geeignet, um durch Anwendung analytischer QS-Maßnahmen die Ergebnisse seiner Tätigkeit zu betrachten.
  - Betonung der Bedeutung der QS im Gesamtprojekt
- organisatorische
  - geringere Zielkonflikte beim Ressourceneinsatz
  - klare Budget- und Ressourcenaufteilung

UNIVERSITÄT LEIPZIG
Institut für Informatik
Betriebliche Informationssysteme

Qualitätssicherung – integriert und doch unabhängig

Vorteile der Unabhängigkeit

- Entwicklung und QS als gleichberechtigte Projektziele
- Neutralität der QS
- klare Budgetaufteilung
- Betonung der Qualitätssicherung

Vorteile der Integration

- QS „bekommt alles mit"
- bessere Gewährleistung einer gleichmäßigen Personalauslastung
- Erleichterung der Teamarbeit

Qualitätssicherung – integriert und doch unabhängig

Personalalternativen

- Personal arbeitet nur in der Qualitätssicherung
  - Mitarbeiter mit hohen Spezialisierungsgrad, vor allem zur Abdeckung besonderer Anforderungen im QS-Prozess

- Mitarbeiter rotieren in festgelegten Abständen zwischen organisatorisch selbstständiger QS und Entwicklung
  - Ermöglicht systematischen Wissenstransfer, Schulung und Weiterbildung des Personalstamms zu QS-spezifischen Aspekten

- Jeder Mitarbeiter arbeitet im selben Projekt sowohl in der QS als auch in der Entwicklung (in der Praxis üblich)
  - Ermöglicht einen flexiblen Personaleinsatz, aber Gefahr, dass QS-Aspekte unterbelichtet bleiben.

Qualitätssicherung – integriert und doch unabhängig

Vorgehensweisen in der Praxis

Trennung von Entwicklung und QS bereits im laufenden Entwicklungsprozess

- Die Entwicklung erstellt Produkte und die Qualitätssicherung ist für die Überprüfung zuständig.
- Entwickler kann sich auf die konstruktiven Aspekte konzentrieren, wird aber nicht zu Sorgfalt angehalten
- Realisierung im Ansatz des Pair Programming

UNIVERSITÄT LEIPZIG
Institut für Informatik
Betriebliche Informationssysteme

Qualitätssicherung – integriert und doch unabhängig

Vorgehensweisen in der Praxis

Integration von Entwicklung und laufender QS; spezielle QS-Phasen (Meilensteine)

- Die Entwicklung und jeder Entwickler ist für einen definierten Qualitätszustand seiner Produkte selbst zuständig.
- Laufende QS erfolgt entwicklungsintegriert auf niedrigem Intensitätsniveau (Durchsprachen, Reviews, Datenerfassung)
- Intensivere und evtl. externe QS setzt in Meilensteinphasen ein
- klar definierte, transparente Verantwortlichkeiten
- Eigenverantwortlichkeit der Entwickler
- erfordert messbare Qualitätsstufen und Nachweis, dass sie erreicht wurden
- Vorteile einer unabhängigen QS werden vor allem durch administrative Vorgaben gesichert

Qualitätssicherung – integriert und doch unabhängig
Vorgehensweisen in der Praxis

Bedeutung der Datenerfassung

- Besondere Bedeutung für die Sicherung der Unabhängigkeit der QS trotz Integration hat die instrumentelle Erfassung von Prüfparametern
  - instrumentell, automatisch, unabhängig, für die Logik des Entwicklungsprozesses transparent
  - Bedeutung des Prinzips der quantitativen Qualitätssicherung
  - QS kann sich auf Fragen der Interpretation der Messwerte konzentrieren.
- Sinnvolle Infrastruktur:
  - Sammlung von Daten, Validierung, Datenbank.
  - Weiterer Vorteil: Qualitätsvergleiche werden möglich.