

Software Engineering

SS 2005

Prof. Dr. Barbara Paech, Jürgen Rückert



Institut für Informatik
Im Neuenheimer Feld 326
69120 Heidelberg
<http://www-swe.informatik.uni-heidelberg.de>
paech@informatik.uni-heidelberg.de



RUPRECHT-KARLS-UNIVERSITÄT HEIDELBERG



2. Beschreibungstechniken

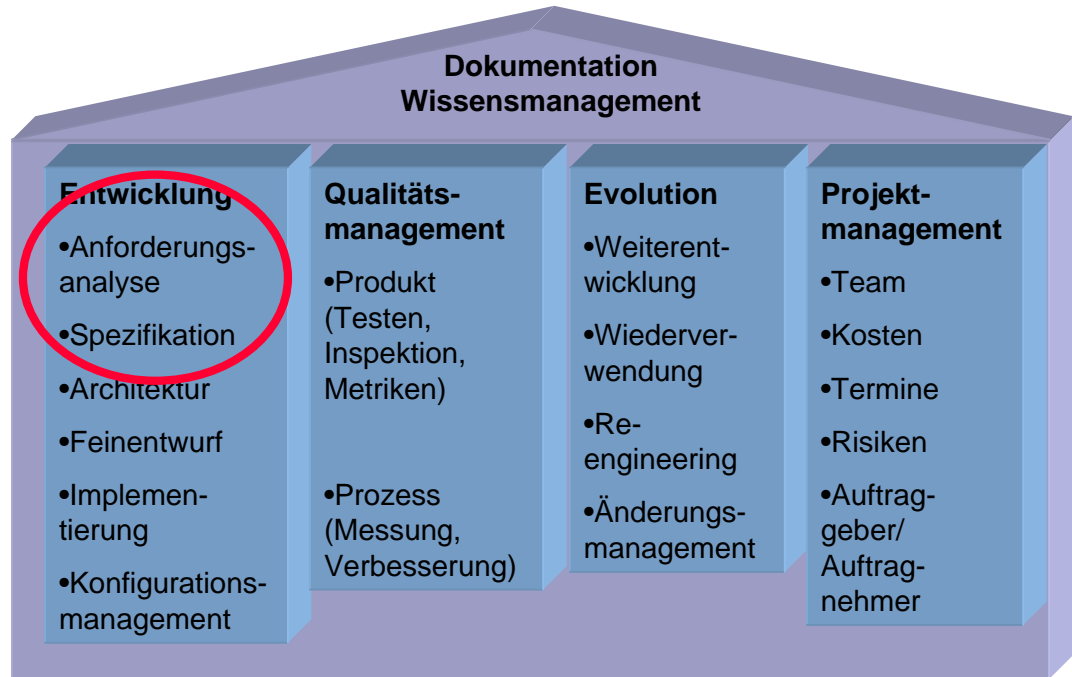
- 2.1. Modellierung
- 2.2. Objektorientierung
- 2.3. Klassendiagramme
- 2.4. Zustandsdiagramme
- 2.5. Nutzungsbeschreibung
- 2.6. NFR
- 2.7. Rationale



1.3.1. Aufgabenbereiche des Engineering

2. Beschreibungstechniken

- 2.1. Modellierung
- 2.2. Objektorientierung
- 2.3. Klassendiagramme
- 2.4. Zustandsdiagramme
- 2.5. Nutzungsbeschreibung
- 2.6. NFR
- 2.7. Rationale



2. Beschreibungstechniken

2. Beschreibungstechniken

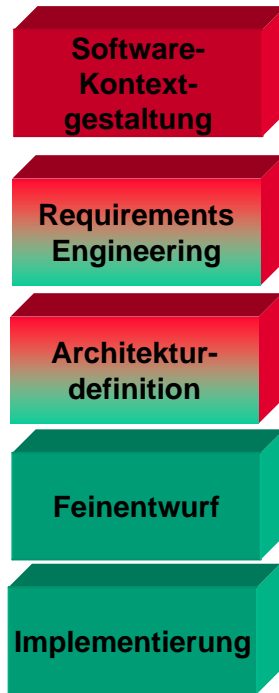
- 2.1. Modellierung
- 2.2. Objektorientierung
- 2.3. Klassendiagramme
- 2.4. Zustandsdiagramme
- 2.5. Nutzungsbeschreibung
- 2.6. NFR
- 2.7. Rationale

- ◆ 2.1. Modellierung
- ◆ 2.2. Objektorientierung
- ◆ 2.3. Klassendiagramme
- ◆ 2.4. Zustandsdiagramme
- ◆ 2.5. Nutzungsbeschreibung
- ◆ 2.6. Nicht-funktionale Anforderungen
- ◆ 2.7. Rationale

1.3.5. Entscheidung über Produktqualität

2. Beschreibungstechniken

- 2.1. Modellierung
- 2.2. Objektorientierung
- 2.3. Klassendiagramme
- 2.4. Zustandsdiagramme
- 2.5. Nutzungsbeschreibung
- 2.6. NFR
- 2.7. Rationale



2.6. Nicht-funktionale Anforderungen

2. Beschreibungstechniken

- 2.1. Modellierung
- 2.2. Objektorientierung
- 2.3. Klassendiagramme
- 2.4. Zustandsdiagramme
- 2.5. Nutzungsbeschreibung
- 2.6. NFR
- 2.7. Rationale

- ◆ 2.6.1. Übersicht
- ◆ 2.6.2. Produktqualitätsanforderungen
- ◆ 2.6.3. Kategorien von Produktqualitätsanforderungen

2.6.1. Übersicht: Welche NFR gibt es ?

2. Beschreibungs- techniken

2.1. Modellierung

2.2. Objekt- orientierung

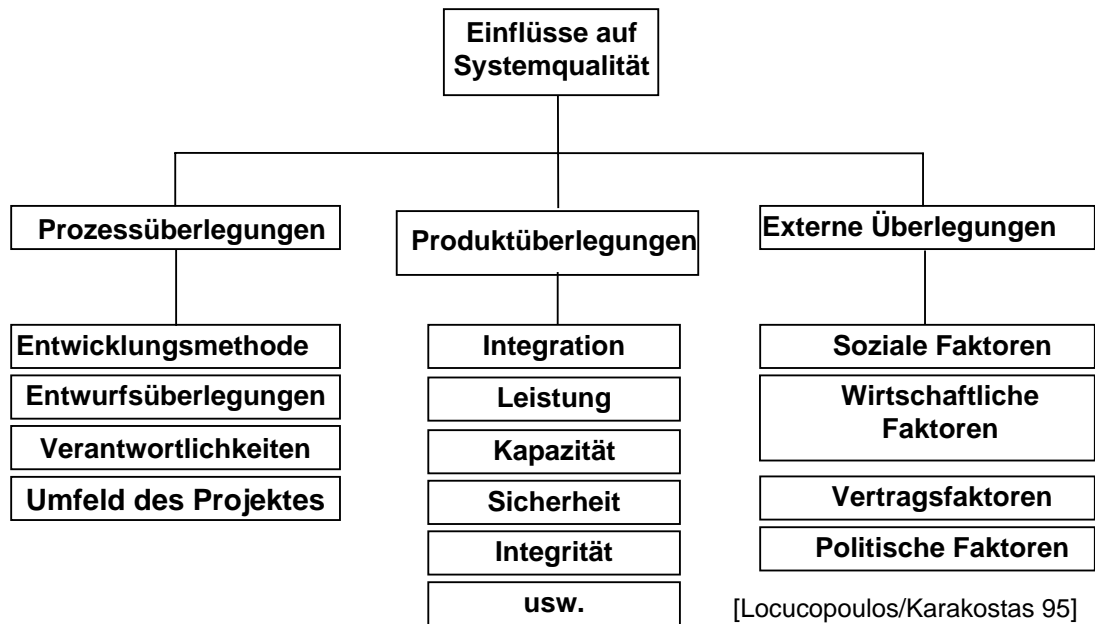
2.3. Klassen- diagramme

2.4. Zustands- diagramme

2.5. Nutzungs- beschreibung

2.6. NFR

2.7. Rationale



2.6.2. (Produkt-)Qualitäts-anforderungen (QA) (1)

2. Beschreibungs- techniken

2.1. Modellierung

2.2. Objekt- orientierung

2.3. Klassen- diagramme

2.4. Zustands- diagramme

2.5. Nutzungs- beschreibung

2.6. NFR

2.7. Rationale

- ◆ Definiert die Antwort auf die Frage: „**Wie gut** muss das durch die funktionalen Anforderungen beschriebene Produkt sein?“
- ◆ **Teilweise untrennbar mit funktionalen Anforderungen** verbunden
z.B. Reaktionszeit bei Kontrolle chemischer Prozesse
- ◆ **Stark beeinflusst von externen Faktoren**
z.B. wie schnell muss das System reagieren, damit es für die Nutzer noch akzeptabel ist
z.B. wie wichtig ist welche Performanz für den Markterfolg

2.6.2. (Produkt-)Qualitäts-anforderungen (QA) (2)

2. Beschreibungs- techniken

- 2.1. Modellierung
- 2.2. Objekt-orientierung
- 2.3. Klassen-diagramme
- 2.4. Zustands-diagramme
- 2.5. Nutzungs-beschreibung
- ▶ 2.6. NFR
- 2.7. Rationale

- ◆ Konkrete Werte müssen aufgrund von **Kosten/ Nutzenabwägungen** festgelegt werden
- ◆ Oft Konflikte zwischen mehreren QA, z.B. erhöht Kapselung die Wartbarkeit, verringert aber dafür oft die Performanz

2.6.2. QA können den Markterfolg entscheiden

2. Beschreibungs- techniken

- 2.1. Modellierung
- 2.2. Objekt-orientierung
- 2.3. Klassen-diagramme
- 2.4. Zustands-diagramme
- 2.5. Nutzungs-beschreibung
- ▶ 2.6. NFR
- 2.7. Rationale

- ◆ WASHINGTON (AP, Jan 17th 2002) -- Microsoft's chairman, Bill Gates, is steering his software empire onto a new strategic heading, **putting security and privacy ahead of new capabilities** [new functionality] in the company's products.
- ◆ In an e-mail to employees obtained by The Associated Press, Gates refers to the new philosophy as **"Trustworthy Computing"** and says his highest priority is to ensure that computer users continued to venture safely across an increasingly Internet-connected world.



2.6.2. Beschreibung von QA

2. Beschreibungs- techniken

2.1. Modellierung

2.2. Objekt- orientierung

2.3. Klassen- diagramme

2.4. Zustands- diagramme

2.5. Nutzungs- beschreibung

2.6. NFR

2.7. Rationale

- ◆ QA werden in der Praxis meist sehr ungenau erfasst
- ◆ typische Entwicklungsmethoden wie RUP oder Notationen wie UML bieten keine Unterstützung zur Erfassung von QA
- ◆ Kann durch Checklisten verbessert werden

- ◆ Methode zur Erhebung von Beschreibung von QA ist aktueller Forschungsgegenstand am Lehrstuhl

2.6.3. Produktqualität nach ISO 9126/DIN66272

2. Beschreibungs- techniken

2.1. Modellierung

2.2. Objekt- orientierung

2.3. Klassen- diagramme

2.4. Zustands- diagramme

2.5. Nutzungs- beschreibung

2.6. NFR

2.7. Rationale

- ◆ **Funktionalität**
 - Angemessenheit
 - Sicherheit
 - Genauigkeit der Berechnung
 - Interoperabilität
 - Konformanz zu Standards
- ◆ **Zuverlässigkeit**
 - Reife
 - Fehlertoleranz
 - Wiederherstellbarkeit
- ◆ **Benutzbarkeit**
 - Verständlichkeit
 - Erlernbarkeit
 - Bedienbarkeit
- ◆ **Effizienz**
 - Zeitverhalten
 - Verbrauchsverhalten
- ◆ **Änderbarkeit**
 - Analysierbarkeit
 - Modifizierbarkeit
 - Stabilität
 - Prüfbarkeit
- ◆ **Übertragbarkeit**
 - Anpassbarkeit
 - Installierbarkeit
 - Konformanz zu Standards
 - Austauschbarkeit

2.6.3. Beispielanforderungen?

2. Beschreibungs- techniken

2.1. Modellierung

2.2. Objekt- orientierung

2.3. Klassen- diagramme

2.4. Zustands- diagramme

2.5. Nutzungs- beschreibung

2.6. NFR

2.7. Rationale

- ◆ **Zuverlässigkeit**
 - Reife
 - Fehlertoleranz
 - Wiederherstellbarkeit
- ◆ **Effizienz**
 - Zeitverhalten
 - Verbrauchsverhalten
- ◆ **Änderbarkeit**
 - Analysierbarkeit
 - Modifizierbarkeit
 - Stabilität
 - Prüfbarkeit
- ◆ **Übertragbarkeit**
 - Anpassbarkeit
 - Installierbarkeit
 - Konformanz zu Standards
 - Austauschbarkeit

2.6.3. Wie beschreibe ich QA Genauigkeit?

[nach Lauesen 2002]

2. Beschreibungs- techniken

2.1. Modellierung

2.2. Objekt- orientierung

2.3. Klassen- diagramme

2.4. Zustands- diagramme

2.5. Nutzungs- beschreibung

2.6. NFR

2.7. Rationale

- ◆ **Intervallgrenzen und Berechnungsgenauigkeit, z.B.**
 - Namensfeld soll 150 Zeichen umfassen
 - Buchungsdatum darf bis zu 2 Jahre das aktuelle Datum übersteigen
 - Sensordaten sollen mit 14 Bit Genauigkeit abgespeichert werden
 - Spracherkennung auch bei Hintergrundgeräuschen der Lautstärke...

2.6.3. Wie beschreibe ich QA Sicherheit?

[nach Lauesen 2002]

2. Beschreibungs- techniken

2.1. Modellierung

2.2. Objekt- orientierung

2.3. Klassen- diagramme

2.4. Zustands- diagramme

2.5. Nutzungs- beschreibung

2.6. NFR

2.7. Rationale

- ◆ Oft **lösungsorientiert** beschrieben (z.B. Firewall)
- ◆ Soll Misuse Cases verhindern => **Bedrohungen** müssen identifiziert werden
 - Typischerweise unterschiedlich für Eingabe, Datenspeicherung, Output
- ◆ Typischerweise 4 **Sicherheitsziele**: CIA+A
 - Confidentiality (nur autorisierte Nutzer)
 - Integrity (nur korrekte Daten)
 - Availability (Verfügbarkeit)
 - Authenticity (Authentizität der Benutzer)
- ◆ 3 typische **Maßnahmen**
 - Prävention (reduziert Auftreten der Bedrohungen, z.B. Passwort)
 - Prüfung (macht Bedrohung explizit, z.B. Logging)
 - Gegenmaßnahmen (reduziert Auswirkungen eines Angriffs, z.B. Back-up Kopie)

2.6.3. Wie beschreibe ich QA Sicherheit?

[nach Lauesen 2002]

2. Beschreibungs- techniken

2.1. Modellierung

2.2. Objekt- orientierung

2.3. Klassen- diagramme

2.4. Zustands- diagramme

2.5. Nutzungs- beschreibung

2.6. NFR

2.7. Rationale

- ◆ **Typische Bedrohungen und Maßnahmen**
 - Für Integrität: Verschmutzung (z.B. verlorene Nachrichten), meist durch Protokolle erkannt (z.B. Parity Check)
 - Für Verfügbarkeit: Zusammenbruch (z.B. Kabel), meist durch spezielle Infrastruktur ausgeglichen (z.B. Rerouting)
 - Für CIA+A: Einfache Nutzerfehler, durch System verhindert (z.B. Bar Code) oder Nutzertraining
 - Für CIA+A: Verarbeitungsfehler, durch Prüfung meist verhindert
 - Für CIA+A: Sabotage (z.B. Passwort stehlen, Lauschen, Programmveränderung), meist durch Zugangskontrollen verhindert
- ◆ **Beispiele für Anforderungen**
 - Es sollen Sicherheitskopien angelegt werden
 - Unabsichtliche Löschung von Daten soll verhindert werden
 - Auftragnehmer muss eine Bedrohungsanalyse erstellen

2.6.3. Wie beschreibe ich QA Effizienz?

[nach Lauesen 2002]

2. Beschreibungs- techniken

2.1. Modellierung

2.2. Objekt- orientierung

2.3. Klassen- diagramme

2.4. Zustands- diagramme

2.5. Nutzungs- beschreibung

2.6. NFR

2.7. Rationale

◆ Verbrauchsverhalten

- Vor allem Volumenangaben, z.B.
 - Max 16 MB Speicher
 - Bis zu 2000 gleichzeitige Nutzer
 - Bis zu 10.000 Kundeneinträge, mit 20% Zuwachs pro Jahr

◆ Zeitverhalten

- Angaben zu **Systemreaktionszeit** (Aufgabenreaktionszeit in bezug zu Benutzungsqualität)
- Angabe von **Durchschnittszeit und Maximum**, oft unter Angaben bestimmter Wahrscheinlichkeiten
- Setzt **Annahmen über Häufigkeit der Benutzereingaben** (Systemlast) und Zeitverhalten der Hardware voraus
- Z.B.
 - 100 Zahlungstransaktionen pro Sekunde unter höchster Systemlast
 - CPU Verbrauch unter Standardlast bis zu 50% (Rest für Batch-Jobs)
 - Keyword-Suche soll max. 5 Sekunden dauern
 - Blättern in 200 Seiten Dokumente soll max. 1 Sekunde dauern

Folie 17

2.6.3. Wie beschreibe ich QA Benutzbarkeit?

[nach Lauesen 2002]

2. Beschreibungs- techniken

2.1. Modellierung

2.2. Objekt- orientierung

2.3. Klassen- diagramme

2.4. Zustands- diagramme

2.5. Nutzungs- beschreibung

2.6. NFR

2.7. Rationale

- ◆ Benutzbarkeit umfasst manchmal auch Aufgabenangemessenheit (in der Norm unter Funktionalität)
- ◆ Im engeren Sinn: **Leichtigkeit der Nutzung**
 - Verständlichkeit
 - Erlernbarkeit
 - Bedienbarkeit
 - Unterstützung sporadischer Nutzer
 - Nutzerzufriedenheit
- ◆ Verschiedene Möglichkeiten Nutzungsqualität zu beschreiben
 - Je nachdem ist das Risiko für **Auftraggeber** (G) oder **Auftragnehmer** (N) höher

Folie 18

2.6.3. Wie beschreibe ich QA Benutzbarkeit?

[nach Lauesen 2002]

- ◆ **Performanzorientiert (N)**
 - Unerfahrene Benutzer sollen Aufgabe X in 15 Minuten durchführen können, erfahrene Benutzer sollen das in 2 Minuten können
- ◆ **Fehlerorientiert (N)**
 - 80% der Benutzer sollen Aufgabe X fehlerfrei durchführen können
- ◆ **Prozessorientiert (G)**
 - Es sind 3 Prototypen zu entwickeln, jeder muss auf Benutzbarkeit getestet werden
- ◆ **Subjektorientiert (N)**
 - 80% der Benutzer sollen das System leicht erlernbar einschätzen
- ◆ **Eingabeorientiert (G)**
 - Eingabe des Kundenauftrags muss mit 5 Tastenanschlägen möglich sein (keine Verwendung der Maus)
- ◆ **Entwurfsorientiert (G)**
 - Die in X gezeigten Bildschirmmasken sind zu verwenden
- ◆ **Richtlinienorientiert (G)**
 - Der MS-StyleGuide ist zu verwenden. Menüs dürfen höchstens eine Tiefe von 3 haben

2. Beschreibungs- techniken

2.1. Modellierung

2.2. Objekt- orientierung

2.3. Klassen- diagramme

2.4. Zustands- diagramme

2.5. Nutzungs- beschreibung

2.6. NFR

2.7. Rationale

2.6.3. Zuordnung zu den FR-Ebenen (1)

- ◆ **Domänenfaktoren**
 - Ergeben sich aus der Anwendungsdomäne
z.B. Sisyphus: Bis zu 20 EntwicklerInnen in einem Projekt
- ◆ **Qualitätsanforderungen an die Aufgabe**
 - Ergeben sich aus dem Aufgabenkontext
z.B. Sisyphus: Inspektion von 40 Elementen kürzer als 30 min.
- ◆ **Globale funktionale Anforderungen**
 - Übergreifende funktionale Anforderungen, die noch in einzelne funktionalen Anforderungen zerlegt werden müssen
z.B. Sisyphus: weltweiter Zugriff

2. Beschreibungs- techniken

2.1. Modellierung

2.2. Objekt- orientierung

2.3. Klassen- diagramme

2.4. Zustands- diagramme

2.5. Nutzungs- beschreibung

2.6. NFR

2.7. Rationale

2.6.3. Zuordnung zu den FR-Ebenen (2)

2. Beschreibungs-
techniken

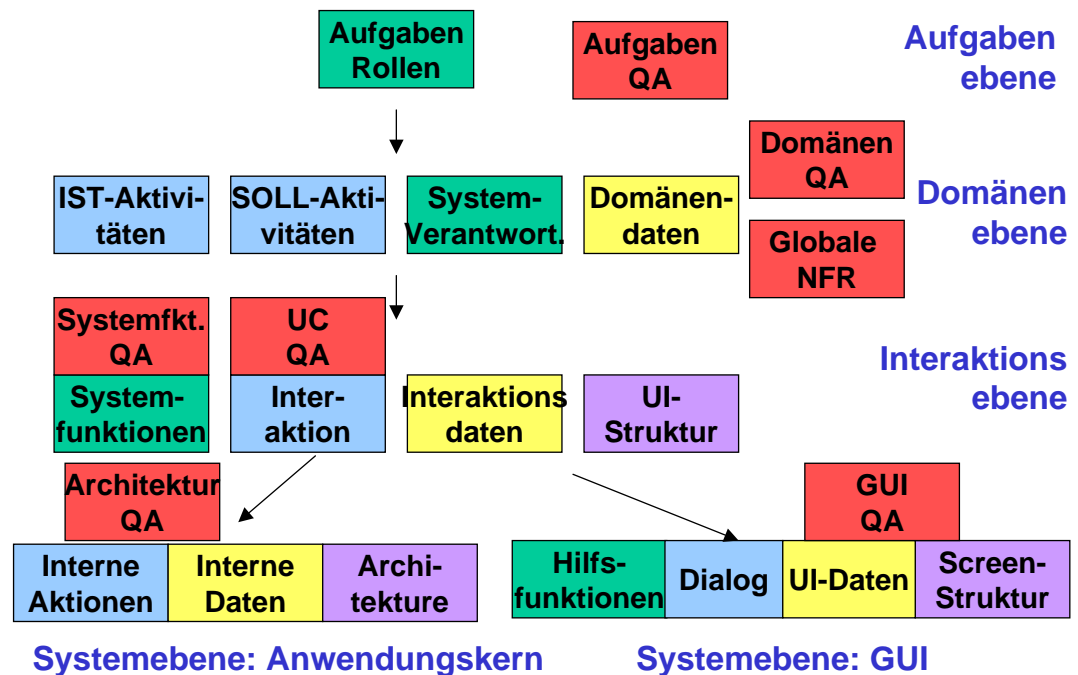
- 2.1. Modellierung
- 2.2. Objekt-
orientierung
- 2.3. Klassen-
diagramme
- 2.4. Zustands-
diagramme
- 2.5. Nutzungs-
beschreibung
- 2.6. NFR
- 2.7. Rationale

- ◆ Qualitätsanforderungen an die Systemunterstützung (Use Cases und Systemfunktionen)
 - QA im engeren Sinn
 - Z.B. Sisyphus: Das Laden einer Sicht darf nicht länger als 1 sec. dauern
- ◆ Qualitätsanforderungen an das GUI
 - QA im engeren Sinn, Anforderung an Dialog und Layout
 - Z.B. Sisyphus: Ausloggen muss jederzeit möglich sein
- ◆ Architektur Anforderungen
 - QA im engeren Sinn, Anforderung an die interne Systemstruktur
 - Z.B. Sisyphus: keine zusätzliche Hardware für neue Funktionalität

2.6.3. Beschreibungsebenen für QA

2. Beschreibungs-
techniken

- 2.1. Modellierung
- 2.2. Objekt-
orientierung
- 2.3. Klassen-
diagramme
- 2.4. Zustands-
diagramme
- 2.5. Nutzungs-
beschreibung
- 2.6. NFR
- 2.7. Rationale



2. Beschreibungs-
techniken

2.1. Modellierung

2.2. Objekt-
orientierung

2.3. Klassen-
diagramme

2.4. Zustands-
diagramme

2.5. Nutzungs-
beschreibung

2.6. NFR

2.7. Rationale

QA zu dem Thema Sicherheit auf **allen** Ebenen

- ◆ **Aufgaben-QA:** Aufgabe X ist sicherheitskritisch, d.h. Informationen über Ergebnisse dürfen nicht nach außen gelangen und sie darf nur von besonders geschulten Personen durchgeführt werden (*diese Aussage vererbt sich auf die Systemunterstützung!*)
- ◆ **Domänen-QA:** An Aufgabe X sind immer mind. 5 Rollen beteiligt, von denen nur Rolle1 und Rolle2 Aktion A ausführen dürfen.
- ◆ **Globale NFR:** Datenschutzrichtlinien sind einzuhalten

2. Beschreibungs-
techniken

2.1. Modellierung

2.2. Objekt-
orientierung

2.3. Klassen-
diagramme

2.4. Zustands-
diagramme

2.5. Nutzungs-
beschreibung

2.6. NFR

2.7. Rationale

QA zu dem Thema Sicherheit auf **allen** Ebenen

- ◆ **UC-QA:** UC Y ist sicherheitskritisch, d.h. Informationen über Ergebnisse dürfen nicht nach außen gelangen
- ◆ **Funktions-QA:** Funktion Z darf keinen Zugang zu den Daten ABC erhalten
- ◆ **GUI-QA:** Auf dem Bildschirm dürfen Benutzerdaten nur anonymisiert dargestellt werden (z.B. Passwort durch ***)
- ◆ **Architektur-QA:** Die Verbindung zwischen Komponente I und J muss abhörsicher sein.

2. Beschreibungs- techniken

2.1. Modellierung

2.2. Objekt- orientierung

2.3. Klassen- diagramme

2.4. Zustands- diagramme

2.5. Nutzungs- beschreibung

2.6. NFR

2.7. Rationale

- ◆ S. Lauesen: Requirements Engineering, Addison-Wesley, 2002
- ◆ Chung, L., Nixon, B.A., Yu, E., Mylopoulos, J., "Non-Functional Requirements in Software Engineering", Kluwer Academic Publishers, 2000

2. Beschreibungs- techniken

2.1. Modellierung

2.2. Objekt- orientierung

2.3. Klassen- diagramme

2.4. Zustands- diagramme

2.5. Nutzungs- beschreibung

2.6. NFR

2.7. Rationale

- ◆ NFR umfassen Prozessanforderungen, Qualitätsanforderungen und externe Faktoren
- ◆ QA müssen **integriert mit FA** erfasst und spezifiziert werden
- ◆ Beschreibung von QA oft vernachlässigt, oft unsystematisch
- ◆ Festlegung einzelner QA muss **Kosten/Nutzen** berücksichtigen
- ◆ Für verschiedene Typen von QA **verschiedene Beschreibungsstile** verwendbar

2.7. Rationale (Begründungszusammenhang)

2. Beschreibungs- techniken

2.1. Modellierung

2.2. Objekt- orientierung

2.3. Klassen- diagramme

2.4. Zustands- diagramme

2.5. Nutzungs- beschreibung

2.6. NFR

2.7. Rationale

- ◆ 2.7.1. Motivation
- ◆ 2.7.2. Wissensbereiche
- ◆ 2.7.3. Rationale-Beschreibung
- ◆ 2.7.4. Vor- und Nachteile
- ◆ 2.7.5. Rationale in Sysiphus

2.7.1. Motivation

2. Beschreibungs- techniken

2.1. Modellierung

2.2. Objekt- orientierung

2.3. Klassen- diagramme

2.4. Zustands- diagramme

2.5. Nutzungs- beschreibung

2.6. NFR

2.7. Rationale

- ◆ Dokumente enthalten meist nur die letzte Entscheidung
- ◆ Verworfenne Optionen und Kriterien sind daraus nicht ablesbar
- ◆ Führt dazu, dass
 - Entscheidungen oft nicht alles berücksichtigen, weil Kriterien und Optionen nicht systematisch untersucht wurden
 - Entscheidungen oft nicht überzeugend sind für Leute, die nicht dabei waren
 - Entscheidungen nachträglich (z.B. bei Änderungen) völlig umgeworfen werden, weil die Leute die Gründe nicht verstehen
 - Verworfenne Optionen (Sackgassen) bei Änderungen noch mal durchgegangen werden

2. Beschreibungs-
techniken

2.1. Modellierung

2.2. Objekt-
orientierung

2.3. Klassen-
diagramme

2.4. Zustands-
diagramme

2.5. Nutzungs-
beschreibung

2.6. NFR

2.7. Rationale

- ◆ Erfassung der **Begründungen für Gestaltungsentscheidungen** während der Softwareentwicklung
- ◆ unterstützt z.B.
 - den **Diskussionsprozess** in der Anforderungsphase (WinWin)
 - die **Erfassung von Domänenwissen** (SCRAM)
 - die **Strukturierung und Erfassung von Entwicklertreffen** (gIBIS)
 - die **Identifikation von Auswirkungen von Änderungen** und die Durchführung von Änderungen
 - die **Einarbeitung** neuer Projektbeteiligter
 - die **Qualität von Anforderungsdokumenten** (Vermeidung alter Sackgassen, Expliziter Trade-Off)

2. Beschreibungs-
techniken

2.1. Modellierung

2.2. Objekt-
orientierung

2.3. Klassen-
diagramme

2.4. Zustands-
diagramme

2.5. Nutzungs-
beschreibung

2.6. NFR

2.7. Rationale

	Wissen über System	Wissen über Prozess (Rollen, Aktivitäten, Dokumente)
Wissen auf Produkt-ebene	Inhalte: Spezifikation, Entwurf, Kode, Testpläne etc.	Inhalte: Projektplan, Kostenplan, Ausgaben, Richtlinien
Wissen auf Organisations-ebene	Inhalte: Domänenmodell, Systemarchitektur, Entwurfsmuster	Inhalte: Prozessmodell, Best Practices, Erfahrungen

2.7.3. Wie beschreibe ich Rationale?

2. Beschreibungs-
techniken

2.1. Modellierung

2.2. Objekt-
orientierung

2.3. Klassen-
diagramme

2.4. Zustands-
diagramme

2.5. Nutzungs-
beschreibung

2.6. NFR

2.7. Rationale

◆ Inhalt

- Fragen (Issues)
- Optionen
- Kriterien
- Argumente (Diskussionen)
- Entscheidungen

◆ Erfassung

- **implizit:** Rationale in Gesprächsnotizen, Protokollen etc. versteckt
- **später:** wird nach eigentlicher Entwicklung zusammengestellt, enthält keine Alternativen
- **kontinuierlich:** während der Entwicklung, Überarbeitung nach der eigentlicher Entwicklung
- **integriert:** auch Überarbeitung während der Entwicklung

Folie 31

2.7.3. Beispiel Beschreibungstechnik: Question Option Criteria (QOC)

2. Beschreibungs-
techniken

2.1. Modellierung

2.2. Objekt-
orientierung

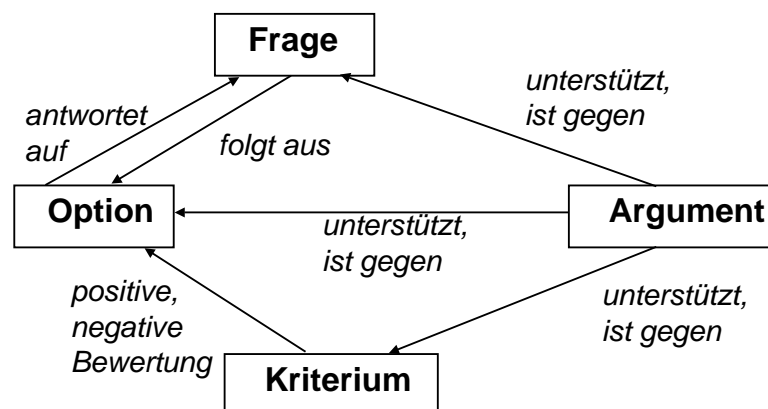
2.3. Klassen-
diagramme

2.4. Zustands-
diagramme

2.5. Nutzungs-
beschreibung

2.6. NFR

2.7. Rationale



[MacLean et al. 1991]

Folie 32

2. Beschreibungs-
techniken

2.1. Modellierung

2.2. Objekt-
orientierung

2.3. Klassen-
diagramme

2.4. Zustands-
diagramme

2.5. Nutzungs-
beschreibung

2.6. NFR

2.7. Rationale

- ◆ Sind konkrete Probleme, die keine eindeutige Lösung haben

Welcher Authentisierungsmechanismus?: Frage

- ◆ Typische Arten bei Dokumentationselementen:
 - Frage bzgl. der Form
 - Klärungsfrage
 - Übersehenes
 - Frage bzgl. des Inhalts, die andere Optionen aufwirft
 - Inkonsistenz
 - Begründung

2. Beschreibungs-
techniken

2.1. Modellierung

2.2. Objekt-
orientierung

2.3. Klassen-
diagramme

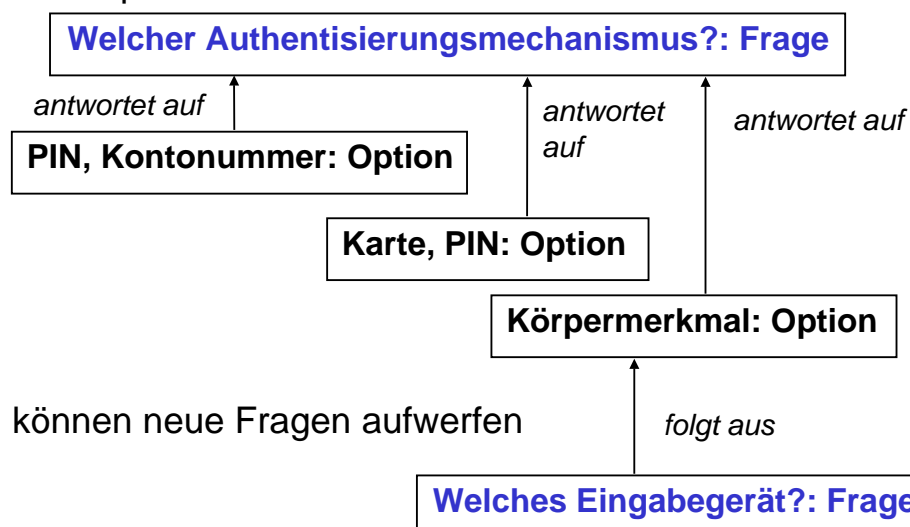
2.4. Zustands-
diagramme

2.5. Nutzungs-
beschreibung

2.6. NFR

2.7. Rationale

- ◆ Beschreiben Alternativen zur Lösung der Probleme
- ◆ eine Option kann sich auf mehrere Probleme beziehen



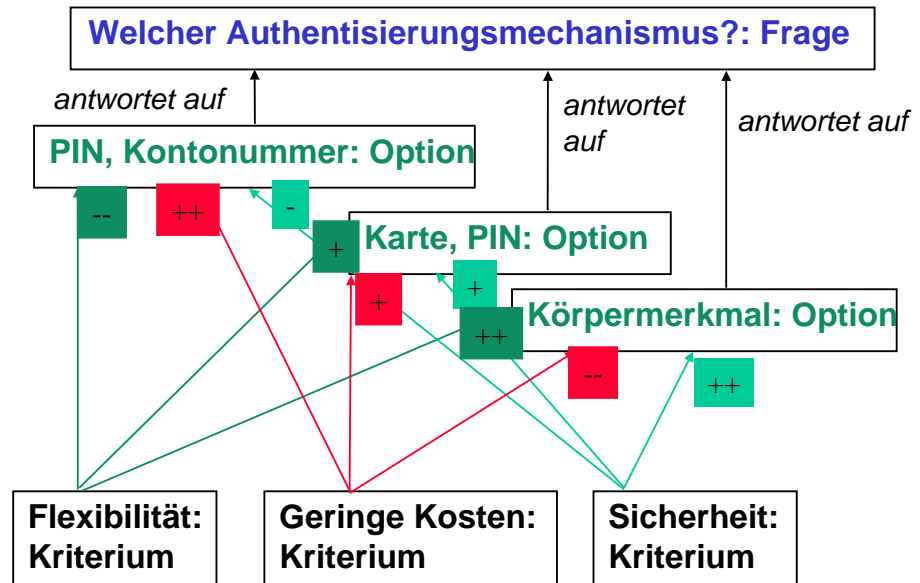
- ◆ können neue Fragen aufwerfen

2.7.3. Kriterien und Bewertung

- ◆ Sind oft Qualitätsanforderungen

2. Beschreibungs-
techniken

- 2.1. Modellierung
- 2.2. Objekt-orientierung
- 2.3. Klassen-diagramme
- 2.4. Zustands-diagramme
- 2.5. Nutzungs-beschreibung
- 2.6. NFR
- ▶ 2.7. Rationale

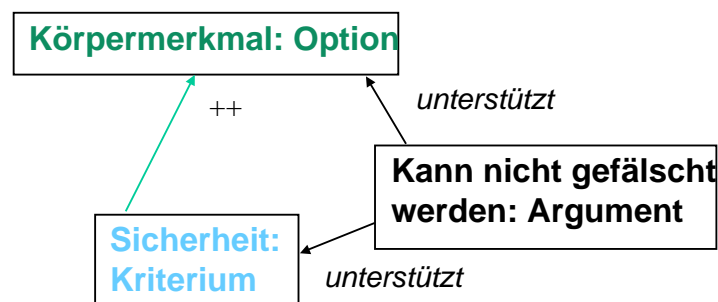


2.7.3. Argumente

- ◆ Kondensieren die Diskussionen
- ◆ bilden den umfangreichsten Teil des Rationale

2. Beschreibungs-
techniken

- 2.1. Modellierung
- 2.2. Objekt-orientierung
- 2.3. Klassen-diagramme
- 2.4. Zustands-diagramme
- 2.5. Nutzungs-beschreibung
- 2.6. NFR
- ▶ 2.7. Rationale



2.7.3. Entscheidungen

2. Beschreibungs- techniken

2.1. Modellierung

2.2. Objekt- orientierung

2.3. Klassen- diagramme

2.4. Zustands- diagramme

2.5. Nutzungs- beschreibung

2.6. NFR

2.7. Rationale

- ◆ Bezieht sich auf ein oder mehrere *offene* Fragen
- ◆ Fasst die gewählte Optionen und die unterstützenden Argumente zusammen.
- ◆ Daraufhin gelten die Fragen als *geschlossen*
- ◆ kann auch revidiert werden. Dann sind die entsprechenden Fragen wieder geöffnet.

2.7.4. Probleme des Rationale

2. Beschreibungs- techniken

2.1. Modellierung

2.2. Objekt- orientierung

2.3. Klassen- diagramme

2.4. Zustands- diagramme

2.5. Nutzungs- beschreibung

2.6. NFR

2.7. Rationale

- ◆ Aufwand zur Erfassung ist hoch, Rationale-Nutzen nicht sofort sichtbar
 - muss **gut motiviert** werden
- ◆ Aufwand zur Konsolidierung ist noch höher
 - **eigene Rolle** für Rationale-Wartung
- ◆ Werkzeuge zur Erfassung nicht gut in den Entwicklungsprozess integriert
 - **neue Werkzeuge** entwickeln (Sysiphus)
- ◆ vollkommen freie Erfassung nicht ausreichend
 - **Vorgaben** nicht zu streng, aber auch nicht zu locker
- ◆ Interaktion über Werkzeug ist oft kompliziert
 - **Groupware**-Elemente einbauen
- ◆ Information, Nutzung ist komplex
 - gute **Sichten, Filter- und Suchmöglichkeiten**

2.7.4. Vorteile des Rationale (1)

2. Beschreibungs- techniken

2.1. Modellierung

2.2. Objekt- orientierung

2.3. Klassen- diagramme

2.4. Zustands- diagramme

2.5. Nutzungs- beschreibung

2.6. NFR

2.7. Rationale

- ◆ Unterstützt Zusammenarbeit
 - Koordinierung, da Entscheidungen untereinander transparent
 - Fokussierte Diskussion, da Optionen und Kriterien verschiedene Sichtweisen transparent machen
 - Mitarbeit, da gezielt Fragen gestellt werden können
 - Konsens, da nachvollziehbare Entscheidungen getroffen werden
- ◆ Unterstützt Wiederverwendung / Änderungen
 - Folgen von Änderungen besser abzuschätzen
 - Folgen von Wiederverwendung besser abzuschätzen

2.7.4. Vorteile des Rationale (2)

2. Beschreibungs- techniken

2.1. Modellierung

2.2. Objekt- orientierung

2.3. Klassen- diagramme

2.4. Zustands- diagramme

2.5. Nutzungs- beschreibung

2.6. NFR

2.7. Rationale

- ◆ Unterstützt Qualität
 - Konsistenz, da Kriterien explizit werden
 - Nachvollziehbarkeit von Verfeinerungen (z.B. Verbindung von Anforderungen zu Entwurfselementen)
 - Wartung, da Folgen besser abschätzbar
- ◆ Unterstützt Wissenstransfer
 - Aus der Vergangenheit lernen, da auch Fehlentscheidungen besser erkennbar und nachvollziehbar
 - Konsolidierung von Begründungen zu Mustern
 - Einarbeitung neuer MitarbeiterInnen
 - Langzeitgedächtnis

2.7.4. Wichtige Einsatzgebiete

- ◆ Besonders wichtig, in
 - Verteilten Projekten
 - Produktlinien Projekten
 - Sicherheitskritischen Systemen
 - COTS (Auftraggeber und Auftragnehmer getrennt)

2. Beschreibungs-
techniken

2.1. Modellierung

2.2. Objekt-
orientierung

2.3. Klassen-
diagramme

2.4. Zustands-
diagramme

2.5. Nutzungs-
beschreibung

2.6. NFR

2.7. Rationale

2.7.5. Rationale in Sysiphus

- ◆ QA sind die Kriterien für die Gestaltung der FA

2. Beschreibungs-
techniken

2.1. Modellierung

2.2. Objekt-
orientierung

2.3. Klassen-
diagramme

2.4. Zustands-
diagramme

2.5. Nutzungs-
beschreibung

2.6. NFR

2.7. Rationale

Schränkt ein	Aufgaben-QA	Domänen-QA	Globale NFR	UC-QA	Funktions-QA	GUI-QA	Architektur-QA
Aufgaben		x					x
UC	x	x	x	x			x
Funktionen	x	x	x	x	x		x
GUI	x	x	x	x	x	x	
Architektur	x	x	x	x			x

2. Beschreibungs- techniken

2.1. Modellierung

2.2. Objekt- orientierung

2.3. Klassen- diagramme

2.4. Zustands- diagramme

2.5. Nutzungs- beschreibung

2.6. NFR

2.7. Rationale

- ◆ Dutoit, A., Paech, B., “Rationale Management”, in Handbook of Software and Knowledge Engineering, World Scientific Publishing, 2002
- ◆ Dutoit, A., McCall, R., Mistrik, I., Paech, B., “Rationale Management in Software Engineering”, Springer Verlag, 2005 to appear

2. Beschreibungs- techniken

2.1. Modellierung

2.2. Objekt- orientierung

2.3. Klassen- diagramme

2.4. Zustands- diagramme

2.5. Nutzungs- beschreibung

2.6. NFR

2.7. Rationale

- ◆ Rationale stellt **wichtiges Wissen** im Softwareentwicklungsprozess dar
- ◆ **Hemmschwelle** zur Erfassung und Nutzung ist groß (wie beim Wissensmanagement allgemein)
- ◆ Geeignete Erfassung, Werkzeugunterstützung ist **noch Forschungsthema**