

Martin Glinz

Requirements Engineering I

Kapitel 11

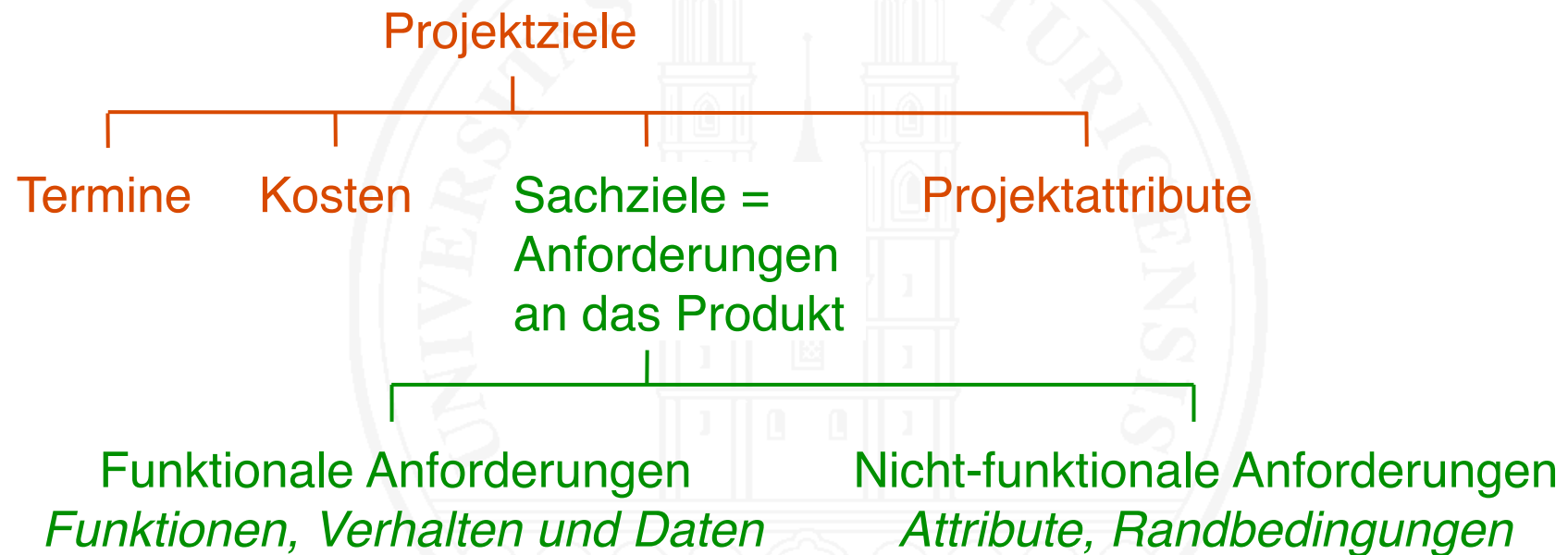
Nicht-funktionale Anforderungen



Universität Zürich
Institut für Informatik

11.1 Was sind nicht-funktionale Anforderungen?

Traditionelle Klassifikation



Definition – traditionell

Nicht-funktionale Anforderungen (non-functional requirements) – Anforderungen an die **Umstände**, unter denen die geforderte Funktionalität zu erbringen ist.

- Art und Weise, **wie** etwas zu tun ist („in weniger als 0,1 s“, „zuverlässig“)
- **Bedingungen**, unter denen etwas zu tun ist („muss auf PCs laufen“)

- Unscharfe Definition \Rightarrow Abgrenzungs- und Klassifikationsprobleme
 - Sache vs. Umstände ist **standpunktabhängig**
 - **Funktional** wird mit „operational repräsentiert“ gleichgesetzt
 - **Nicht-funktional** wird mit „weich“ gleichgesetzt

Probleme – 1: Sache vs. Umstände

- **Traditionelle** Auffassung:
 - Funktionale Anforderungen spezifizieren die zentrale Sache
 - Nicht funktionalen Anforderungen spezifizieren die Umstände
- Beispiel:

| | |
|-----------------------|----------------------------|
| zentrale Sache | Umstände |
| 100m Rennen laufen | Schneller als Konkurrenten |
- Gilt nicht immer:

Oft auch **genau umgekehrter** Standpunkt möglich
- Beispiel:

| | |
|-----------------------------------|-----------------------|
| zentrale Sache | Umstände |
| Schneller als Konkurrenten laufen | Die Strecke ist 100 m |

Probleme – 2: Abhängigkeit von der Repräsentation

- Beispiel: eine Sicherheitsanforderung

„Das System muss den unautorisierten Zugriff auf die Kundenstammdaten verhindern, soweit dies technisch möglich ist“
ist eine **nicht-funktionale** Sicherheitsanforderung

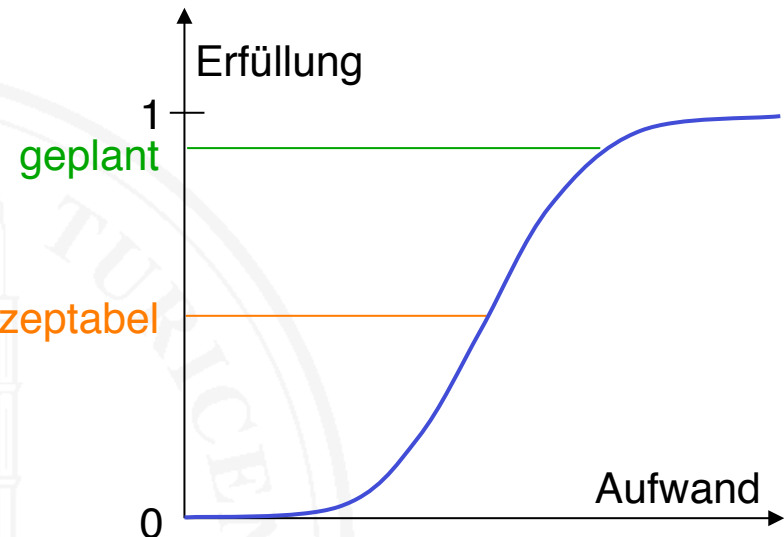
Um eine solche Anforderung prüfbar zu machen, wird sie häufig **operationalisiert**, beispielsweise durch

„Der Zugriff auf die Kundenstammdaten muss über eine Login-Prozedur mit Passworten geschützt werden“. „Die Kundenstammdaten müssen verschlüsselt gespeichert werden.“

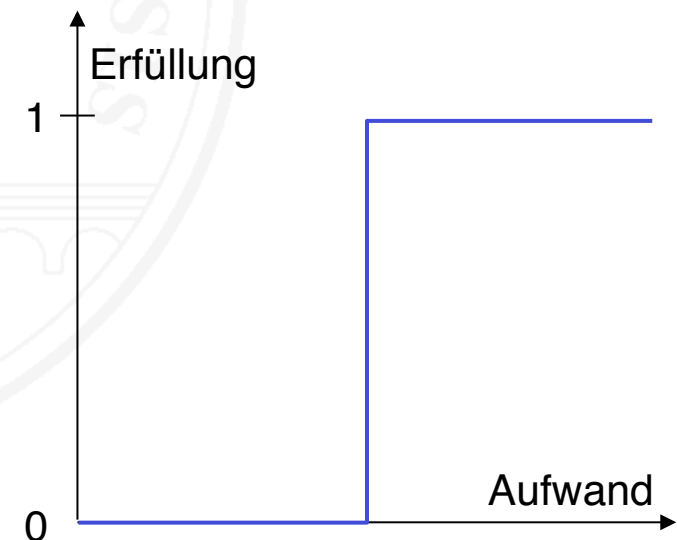
In der operationalen Repräsentation sind dies **funktionale** Anforderungen

Probleme – 3: nicht-funktional \neq weich

- Traditionell werden nicht-funktionale Anforderungen als Anforderungen aufgefasst, deren Erfüllung **weich** ist, d.h. für die es eine **Skala** gibt [Gilb 1997]



- Im Gegensatz dazu gibt es gerade im Bereich von Leistungsanforderungen auch **harte** Anforderungen, deren Erfüllungsverhalten **binär** ist

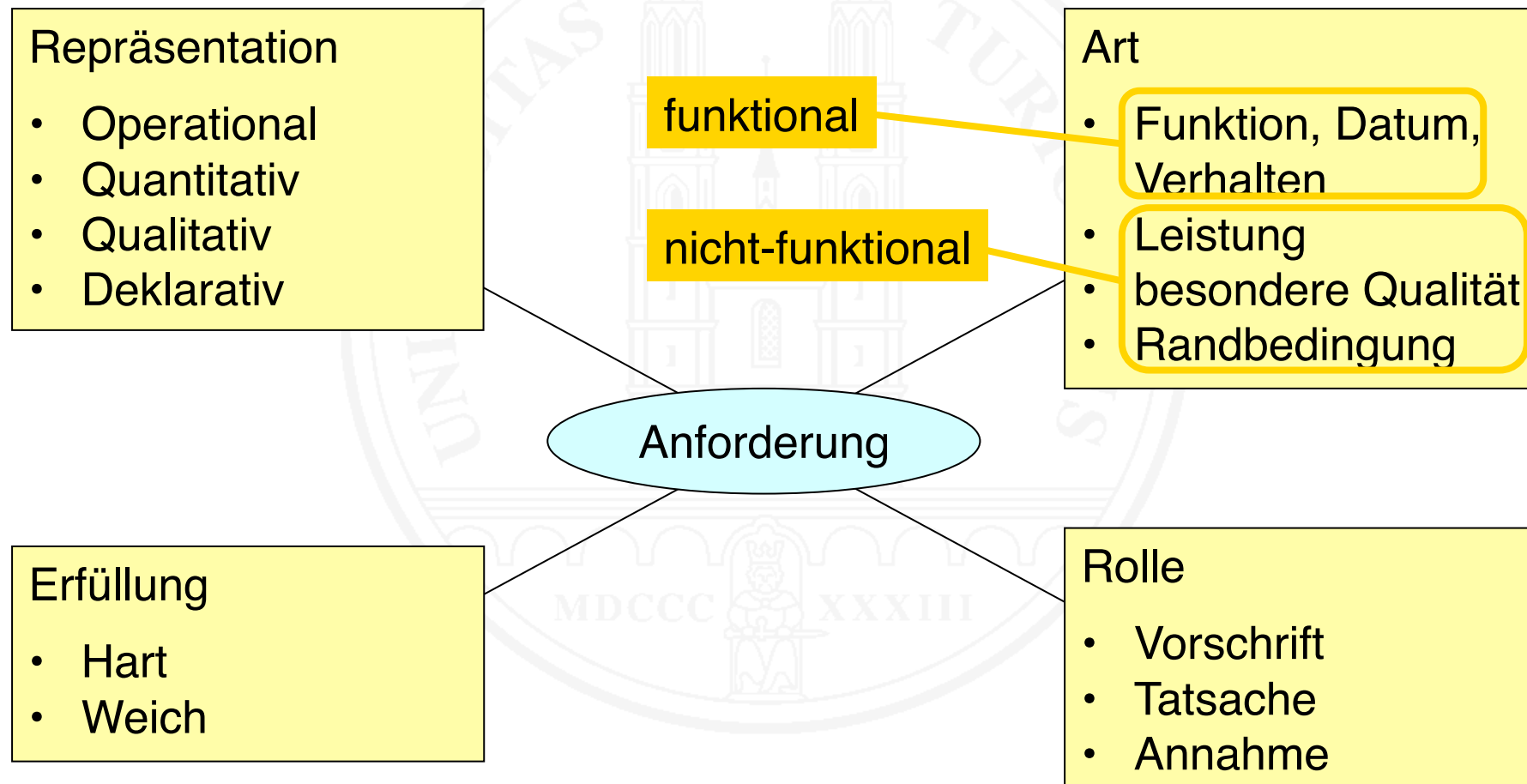


Konsequenzen

- Art, Repräsentation und Erfüllung von Anforderungen voneinander trennen
⇒ facettierte Klassifikation [Glinz 2005]
- Die Unterscheidung von funktionalen und nicht-funktionalen Anforderungen macht Sinn, wenn sie
 - als **verschiedene Arten** von Anforderungen verstanden werden
 - die Unterscheidung **unabhängig** von Repräsentation und Erfüllung ist⇒ führt zu einer neuen Definition und Taxonomie [Glinz 2007]

Anforderungen nach mehreren Facetten klassifizieren

[Glinz 2005]

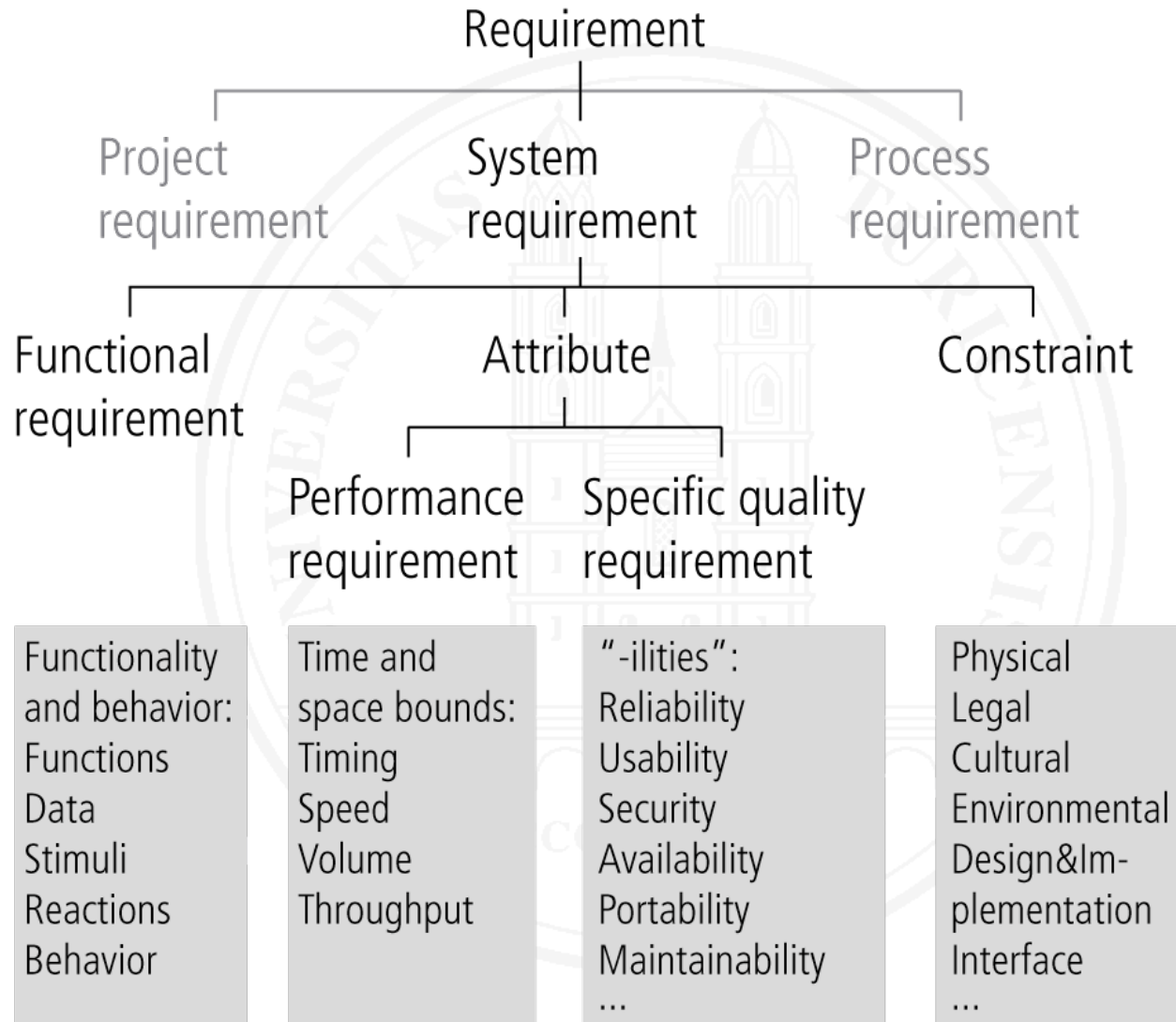


Definition – neu

[Glinz 2007]

- Eine Anforderung bezeichnen wir als **funktional**, wenn das ihr zu Grunde liegende **Bedürfnis funktional** ist, d.h. sich auf **Gegenstände** der Informationsverarbeitung (Daten, Operationen, Verhalten) bezieht
- Eine Anforderung bezeichnen wir als **nicht-funktional**, wenn das ihr zu Grunde liegende **Bedürfnis** charakterisierbar ist als
 - ein **Leistungsmerkmal**
 - ein nicht auf Funktionserfüllung bezogenes **Qualitätsmerkmal**
 - eine **Einschränkung** der Menge der **zulässigen Lösungen** durch **Auftraggeber/Kunden** oder **nicht beeinflussbare äußere Faktoren**
- **Unterschied** zur traditionellen Definition:
 - **Nicht** die **Darstellung** ist funktional oder nicht-funktional,
 - **sondern** das **der Anforderung zu Grunde liegende Bedürfnis**
- Untergliederung nach Art auch mit der neuen Definition sinnvoll

Eine neue Taxonomie für Anforderungen



[aus Glinz 2007]

Bestimmung der Art einer Anforderung

- Grundsatz: nach der zu Grunde liegenden **Intention**, *nicht* nach der Darstellung der Anforderung
- **Schema:**

| | |
|--|-------------------------|
| Wird diese Anforderung gestellt, weil... | |
| ... Systemverhalten, Daten, Eingaben oder Reaktionen auf Eingaben zu spezifizieren sind – unabhängig davon, wie dies geschehen soll? | funktionale Anforderung |
| ... Restriktionen bezüglich Verarbeitungs-/ Reaktionszeiten, Datenmengen oder Datenraten zu spezifizieren sind? | Leistungsanforderung |
| ... eine spezielle Qualität, die das System aufweisen soll, zu spezifizieren ist? | besondere Qualität |
| ... irgend eine andere Restriktion zu spezifizieren ist? | Randbedingung |

Aufgabe 11.1: Anforderungen klassifizieren

Klassifizieren Sie die nachstehenden Anforderungen nach ihrer Art, Repräsentation, Erfüllung und Rolle.

1a $|x - \sqrt{x}| < \varepsilon$ (mit ε gemäß 1b)

1b $\varepsilon < 10^{-6} \wedge \varepsilon < 10^{-6}x$

1c Berechnungszeit für sqrt im Mittel unter $10 \mu\text{s}$, maximal $100 \mu\text{s}$

2 Die Zuverlässigkeit der vom Kunden beigestellten Module soll sehr hoch sein

3 Wenn das System einen Fehler erkennt, soll es eine für den Benutzer verstehbare Meldung erzeugen

11.2 Leistungsanforderungen

Leistungsanforderung (performance requirement) – eine Anforderung, deren zu Grunde liegendes Bedürfnis ein **Leistungsmerkmal** ist

Formen von Leistungsanforderungen:

- **Zeit**
 - für die Erledigung einer Aufgabe
 - für eine Reaktion
 - Minimum? Maximum? Innerhalb eines gegebenen Intervalls? Im Mittel? Tolerierte Abweichungen?
- **Menge**
 - von Daten
 - Minimum? Maximum?

Leistungsanforderungen – 2

○ Raten

- Datendurchsatz
- Transaktionsrate
- Häufigkeit der Verwendung einer Funktion
- im Mittel? Maximal? Verteilung bekannt?

○ Ressourcenverbrauch

- Rechnerkapazität
- Speicherkapazität
- Übertragungskapazität

○ Genauigkeit* (von Berechnungen)

- Auf wie viel Stellen genau? Festkomma oder Gleitkomma?

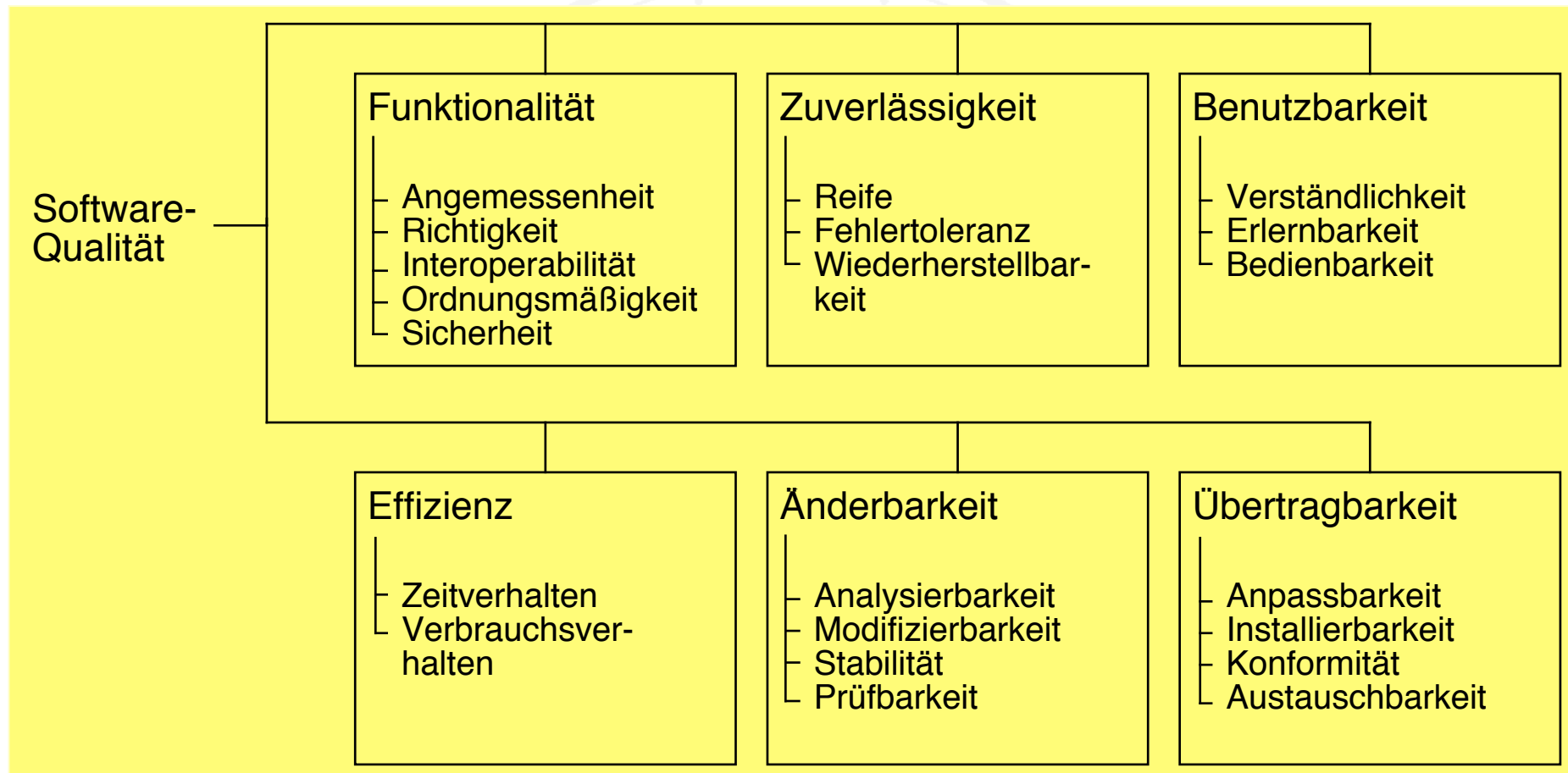
* Wird manchmal als funktionale Anforderung betrachtet, beschreibt aber eine Eigenschaft und ist daher nach neuer Definition nicht-funktional

11.3 Besondere Qualitätsanforderungen

Besondere Qualität (specific quality requirement) – eine Anforderung, deren zu Grunde liegendes Bedürfnis ein nicht auf Funktionserfüllung bezogenes **Qualitätsmerkmal** ist

- Ein **Qualitätsmodell** hilft bei der Identifikation der benötigten Qualitäten
- Beispiel: Qualitätsmodell aus **ISO/IEC 9126** (DIN 66272)
- Auf **Funktionserfüllung** bezogene Qualitäten wie **Angemessenheit** oder **Richtigkeit** werden dabei **nicht betrachtet**; dies sind **funktionale Anforderungen**

Qualitätsmodell nach ISO/IEC 9126 (DIN 66272)



11.4 Randbedingungen

Randbedingung (constraint) – eine Anforderung, deren zu Grunde liegendes Bedürfnis eine **Einschränkung** der Menge der **zulässigen Lösungen** durch den **Auftraggeber/Kunden** oder durch **nicht beeinflussbare äußere Faktoren** ist

- Mögliche **Klassifikation** von Randbedingungen
 - **Technisch**: Plattformen, Schnittstellen, Nachbarsysteme,...
 - **Organisatorisch**: zum Beispiel Prozesse und Organisationsformen, die unverändert bleiben müssen
 - **Normativ**: Gesetze, Verordnungen, Normen,...
 - **Kulturell**: Sprache, Gebräuche, Traditionen,...
 - Andere **explizite Vorgaben** des Auftraggebers
- Randbedingungen werden zusammen mit den übrigen Anforderungen erhoben, aber **als eigene Anforderungsart dokumentiert**

11.5 Gewinnung nicht-funktionaler Anforderungen

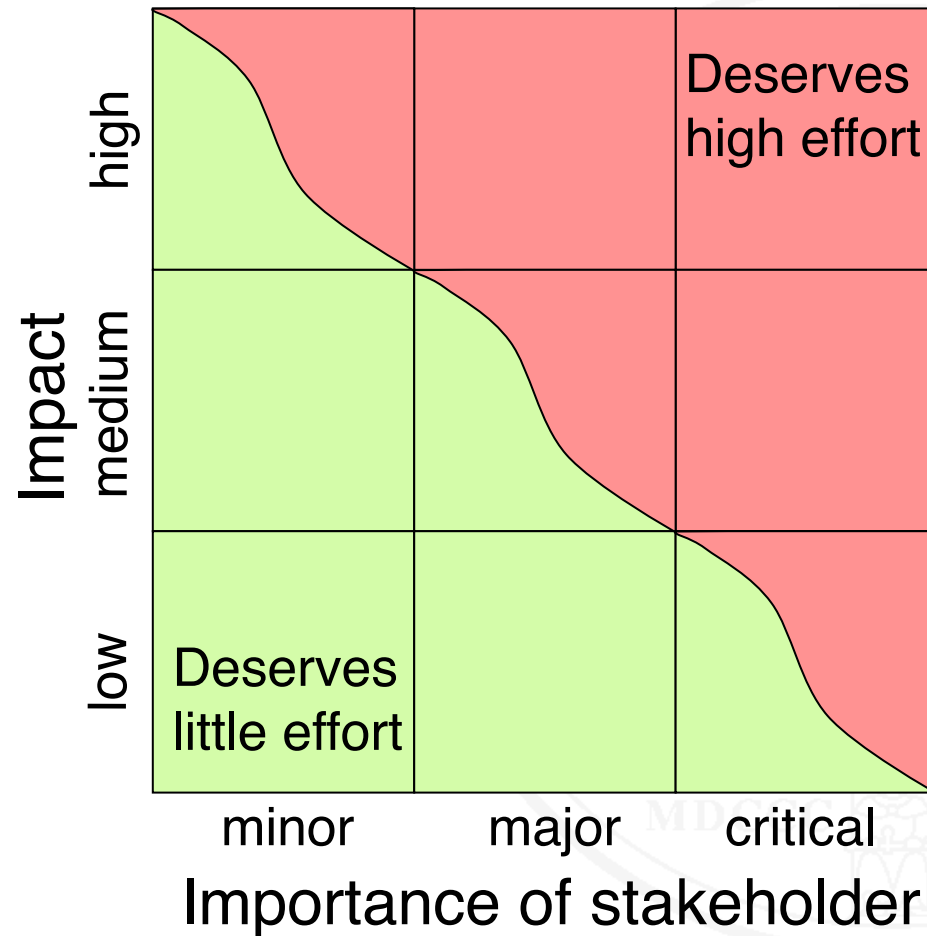
- Nicht-funktionale Anforderungen können ebenso **kritisch** für den Erfolg eines Systems sein wie die funktionalen Anforderungen
- Bei der Gewinnung von Anforderungen werden die nicht-funktionalen Anforderungen dennoch häufig **vergessen** oder stiefmütterlich behandelt
- Im Gewinnungsprozess müssen die nicht-funktionalen Anforderungen **explizit thematisiert werden**
- Zum gezielten Stellen von Fragen können die in den Abschnitten 11.2 bis 11.4 genannten Kategorien als **Checkliste** dienen

Typisches Vorgehen

Gewinnung von Leistungs- und Qualitätsanforderungen

- Fragen stellen:
«Wie fehlertolerant soll das System sein?»
- Antworten analysieren und in geeigneter Form darstellen
 - Klassisch:
 - Quantitativ mit den zugehörigen Maßen
oder
 - Operational, so dass die Anforderung durch Testfälle überprüfbar ist
 - Nicht immer optimal ⇨ Darstellungsform an Wert und Risiko orientieren
 - Prüfbarkeit anstreben

Typisches Vorgehen – 2: Risikoanalyse



- Kritikalität beurteilen
- Weitere Faktoren berücksichtigen (nächste Folie)

[Glinz 2008]

Risikoanalyse: Weitere Faktoren



- Erforderlicher Aufwand für Quantifizierung
- Validität der gewählten Maße
- Bedeutung der Anforderung
- Gemeinsames Verständnis
- Existenz von Referenzsystemen
- Länge des Rückkopplungszyklus
- Art der Auftraggeber-Auftragnehmer-Beziehung
- Notwendigkeit von Zertifizierung

Typisches Vorgehen – 3: Maße

- Bestimmung oder Definition geeigneter Maße
- Direkte Maße:
 - «Die Fehlertoleranz wird in MTTF gemessen und soll im Mittel kleiner als 10^6 Betriebsstunden sein»
- Indirekte Maße als Indikatoren:
 - «Die Bedienung des System gilt als erlernbar, wenn
 - pro Person nicht mehr als zwei Tage Schulung aufgewendet werden müssen,
 - für jede Hauptfunktion der Lernaufwand für ihre erfolgreiche Anwendung im Mittel weniger als eine Stunde beträgt.»

Typisches Vorgehen – 4: Randbedingungen

Gewinnung von Randbedingungen

- Gezielte **Fragen stellen**
- Genannte Randbedingungen **hinterfragen**: Sind es als als Lösungsvorgaben getarnte Anforderungen?
- Beispiel:
 - Aussage: «**Das System muss mit einem Magnetbandkassetten-Laufwerk ausgestattet sein.**»
 - Dahinter verborgene Anforderung: «**Das System muss die Sicherung der Daten in einfacher Weise ermöglichen**»
- Resultate in **prüfbarer Form darstellen**

Aufgabe 11.2: Nicht-funktionale Anforderungen

Gegeben sei die Fallstudie Institutsbibliothek.

- a) Identifizieren Sie in der verfügbaren Information Leistungsanforderungen, besondere Qualitäten und Randbedingungen.
- b) Formulieren sie die gefundenen Anforderungen in prüfbarer Form.
- c) Wo hat es mit hoher Wahrscheinlichkeit weitere nicht-funktionale Anforderungen, die Sie erfragen müssen? Mit welchen Beteiligten im Projekt reden Sie?

Literatur

- Chung, L., B. Nixon, E. Yu, and J. Mylopoulos. (2000). *Non-Functional Requirements in Software Engineering*. Kluwer Academic Publishers.
- Gilb, T. (1997). Towards the Engineering of Requirements. *Requirements Engineering* **2**, 3 165-169.
- Glinz, M. (2005). Rethinking the Notion of Non-Functional Requirements. *Proceedings of the Third World Congress for Software Quality (3WCSQ 2005)*, München, Vol. II, 55-64.
- M. Glinz (2007). On Non-Functional Requirements. *Proceedings of the 15th IEEE International Requirements Engineering Conference*, Delhi, India. 21-26.
- Glinz, M. (2008). A Risk-Based, Value-Oriented Approach to Quality Requirements. *IEEE Software* **25**, 2. 34-41.
- IEEE (1990). *Standard Glossary of Software Engineering Terminology*. IEEE Standard 610.12-1990.
- IEEE (1993). *IEEE Recommended Practice for Software Requirements Specifications*. IEEE Standard 830-1993.
- ISO/IEC 9126-1 (2001). *Software engineering - Product quality - Part 1: Quality model*. International Organization for Standardization.

Literatur – 2

Mylopoulos, J., L. Chung, B. Nixon (1992). Representing and Using Nonfunctional Requirements: A Process-Oriented Approach. *IEEE Transactions on Software Engineering* **18**, 6 (June 1992). 483-497.

