

# Software-Qualitätssicherung

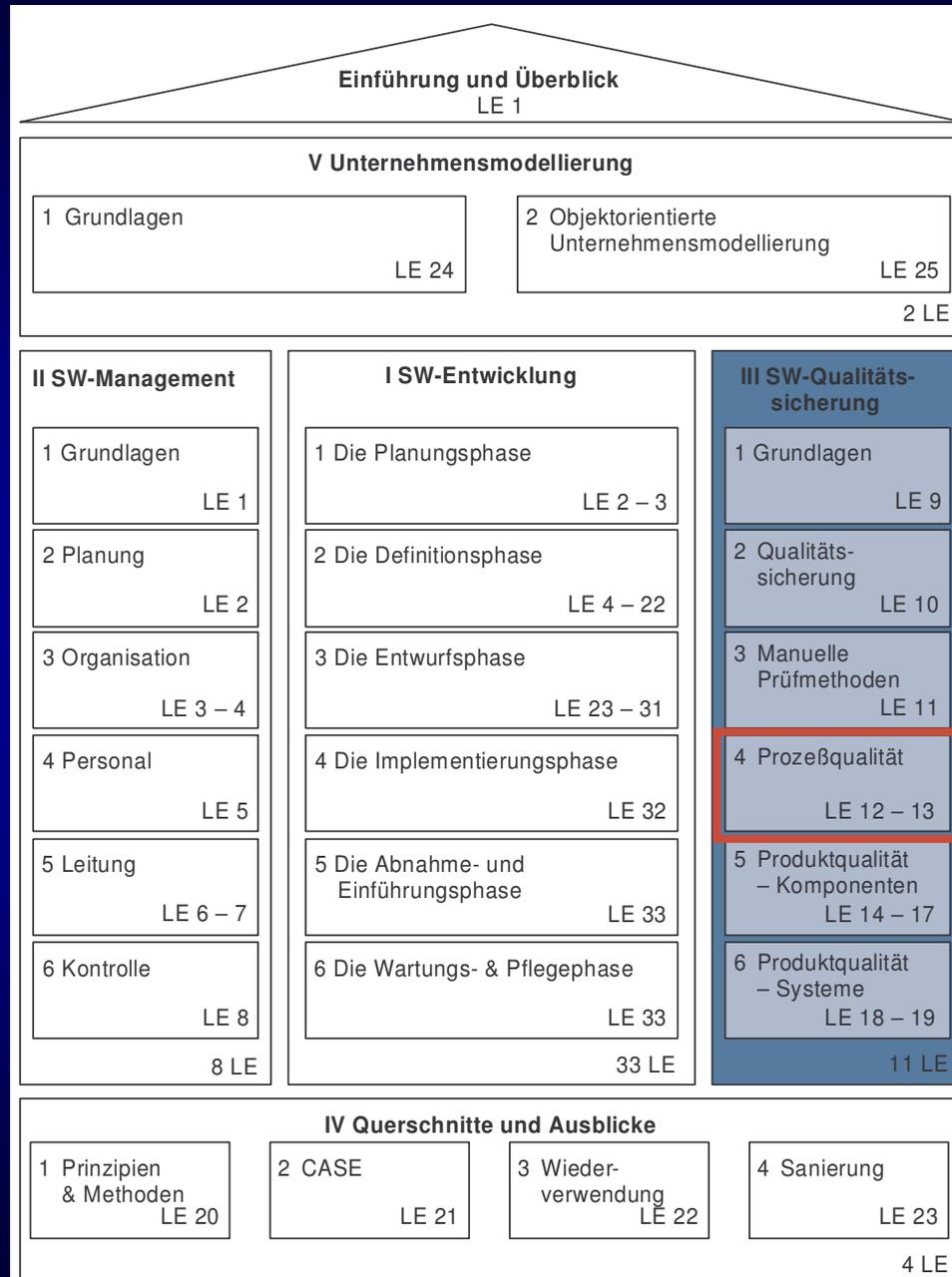
## 4 Verbesserung der Prozeßqualität – CMM und Spice

Prof. Dr. Joachim Hertel  
Fachrichtung Informatik  
Universität des Saarlandes



# III Software-QS - Prozeßqualität - CMM - Spice

LE 13  
2



Legende: LE = Leereinheit (für jeweils 1 Unterrichtsdoppelstunde)

## Lernziele

- ♦ **Aufbau und Struktur sowie Vor- und Nachteile des CMM-Ansatzes kennen und erläutern können**
- ♦ **Die Reifegradstufen des CMM-Ansatzes und des SPICE-Ansatzes mit ihren Prozeßcharakteristika erklären können**
- ♦ **Aufbau und Struktur des SPICE-Ansatzes darstellen können**
- ♦ **Die Forderungen des *Business Engineering* an Geschäftsprozesse schildern können**
- ♦ **Die behandelten Ansätze ISO 9000, TQM, CMM, SPICE und *Business Engineering* vergleichend darstellen können.**

## **Inhalt**

### **4.3 Der CMM-Ansatz**

**4.3.1 Die fünf Reifegradstufen**

**4.3.2 Die Hauptkriterien**

**4.3.3 Durchführung von  
Prozeßverbesserungen**

**4.3.4 Aufwand und Nutzen**

**4.3.5 Vergleiche CMM vs. ISO 9000 vs. TQM**

**4.3.6 Vor- und Nachteile.**

## Inhalt

### 4.4 Der SPICE-Ansatz

#### 4.4.1 Die Struktur von SPICE

#### 4.4.2 Die Prozeß-Dimension

#### 4.4.3 Die Reifegrad-Dimension

#### 4.4.4 Vor- und Nachteile

### 4.5 *Business Engineering.*

## 4.3 Der CMM - Ansatz

### ◆ Zur Historie

#### ◆ **Watts S. Humphrey**

Direktor des Prozeß-Programms  
am SEI (*Software  
Engineering Institute*)

#### ◆ Autor der Bücher

- »*Managing the Software Process*« (1989)
- »*A Discipline for Software Engineering*« (1995)

#### ◆ 1959 bis 1986 bei IBM.



## 4.3 Der CMM - Ansatz

### ◆ CMM-Historie

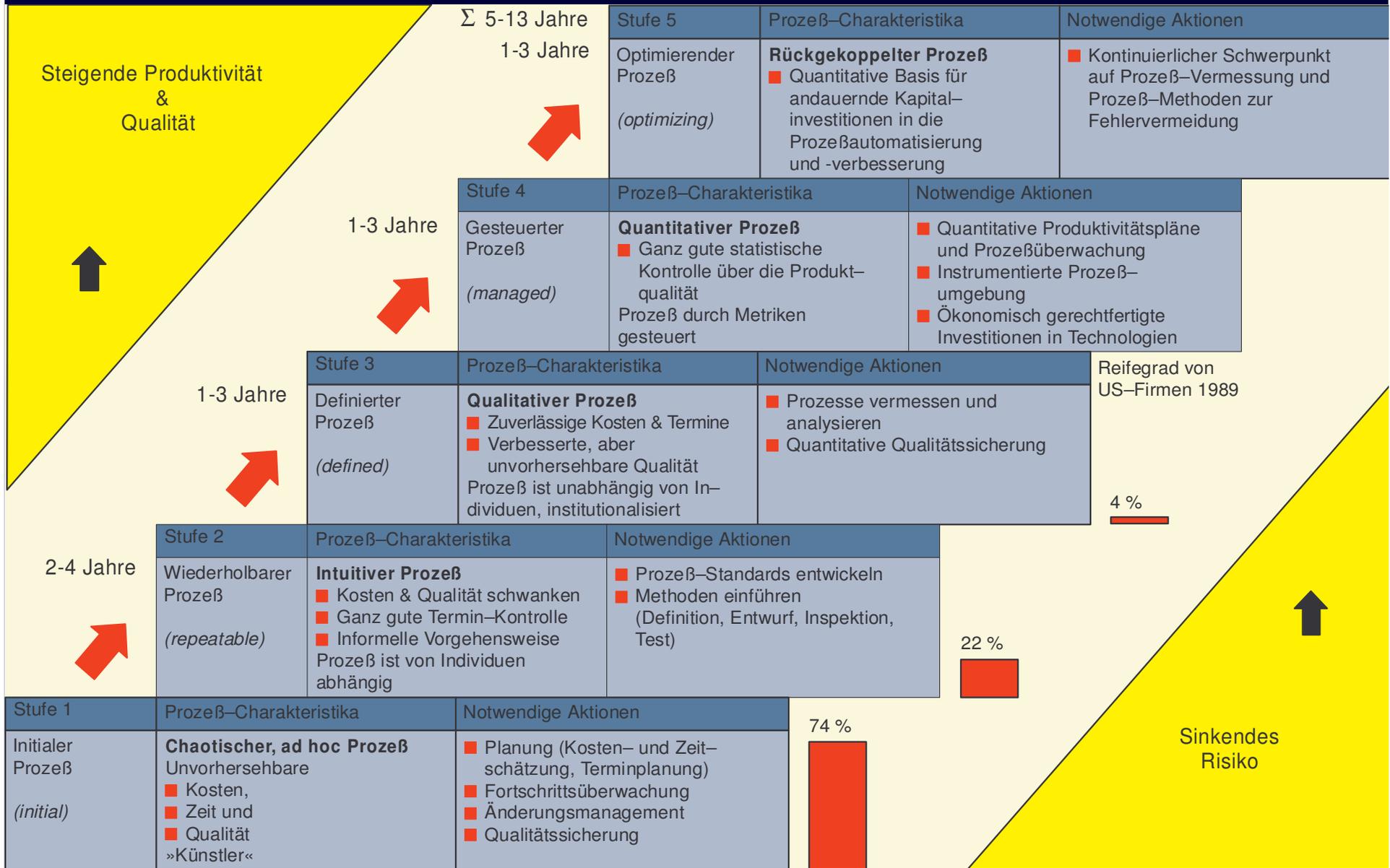
- ◆ 1987: Fragebogen zur Bewertung der Leistungsfähigkeit von **Software-Lieferanten** durch das *Software Engineering Institute* (SEI) der *Carnegie Mellon University*
- ◆ Fragebogen wurde zu einem **Referenzmodell** als **Vergleichsnorm** für **Software-Lieferanten**
- ◆ Namen des Referenzmodells
  - *Capability Maturity Model* (**CMM**)
- ◆ 1991 Version 1.0
- ◆ 1993 verbesserte Version 1.1
- ◆ 1997 überarbeitete Version 2.0  
Bewertungsverfahren mit Hilfe eines Fragebogens = **Assessments**.

## 4.3.1 Die 5 Reifegradstufen

### ◆ Allgemein

- ◆ CMM bezieht sich auf die **Qualität des Software-Entwicklungsprozesses** eines Unternehmens oder eines Bereiches innerhalb eines Unternehmens
- ◆ Unterscheidung von **5 unterschiedlichen Qualitätsstufen** von Software-Entwicklungsprozessen
- ◆ Jede Qualitätsstufe beschreibt einen bestimmten Reifegrad (**maturity**) eines Entwicklungsprozesses
- ◆ Stufen bauen aufeinander auf
- ◆ Eine Stufe setzt die Erfüllung der Anforderungen der anderen Stufen voraus.

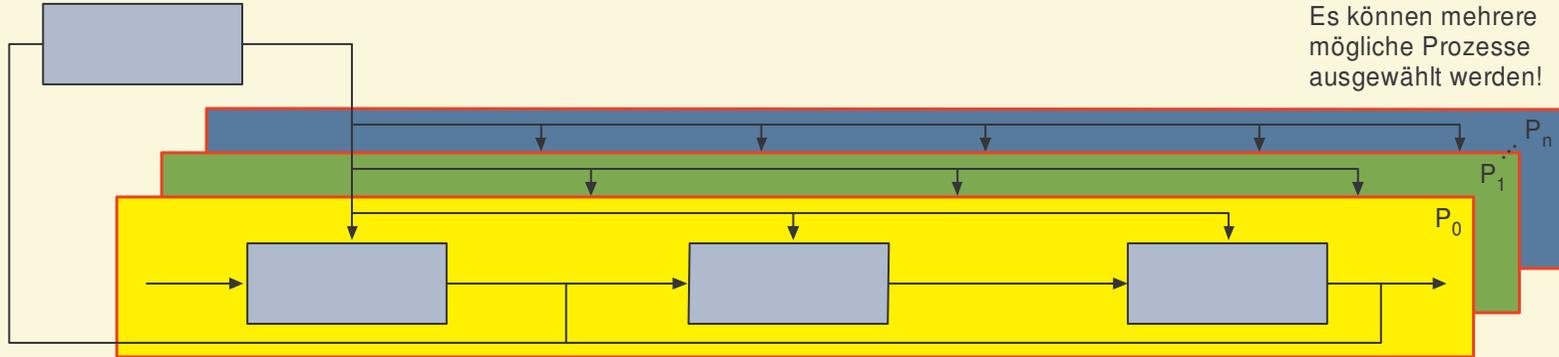
# III Software-QS - Prozeßqualität - CMM - Spice



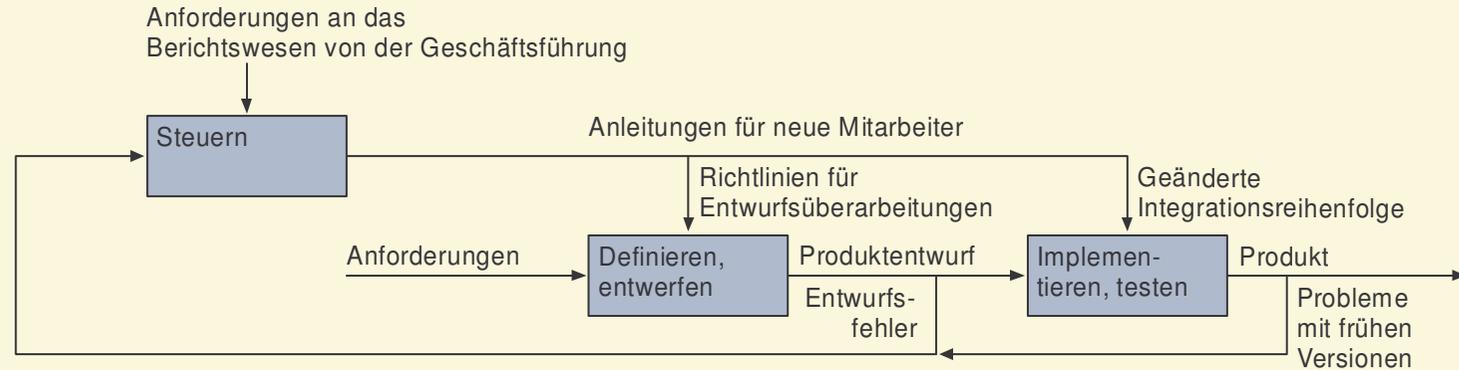
Quelle: /Humphrey 89, S. 56/, /Humphrey, Kitson, Kosse 89/

# III Software-QS - Prozeßqualität - CMM - Spice

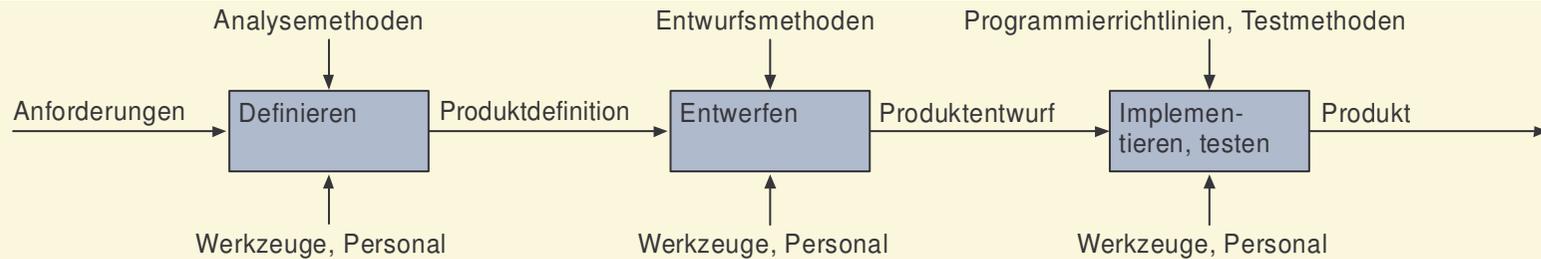
**d**  
Stufe 5  
Optimierender  
Prozeß



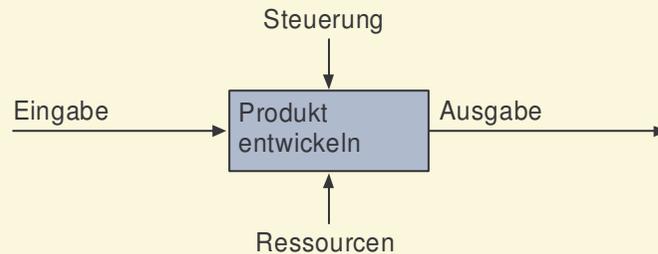
**c**  
Stufe 4  
Gesteuerter  
Prozeß



**b**  
Stufe 3  
Definierter  
Prozeß

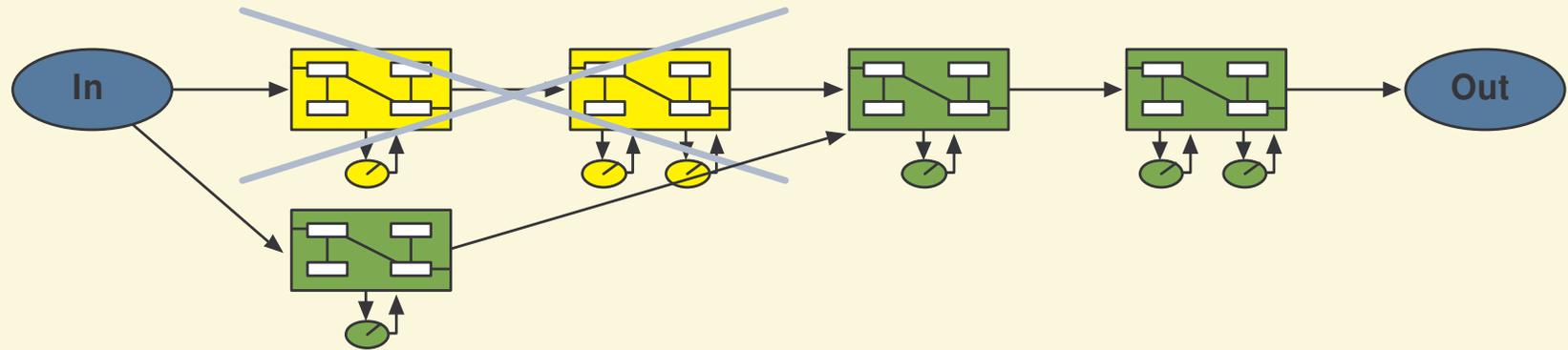


**a**  
Stufe 2  
Wiederholbarer  
Prozeß

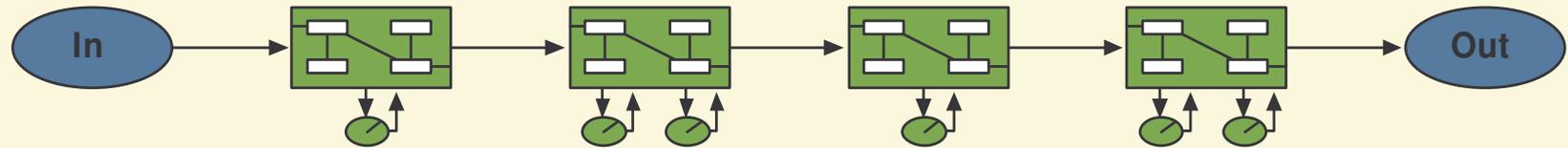


# III Software-QS - Prozeßqualität - CMM - Spice

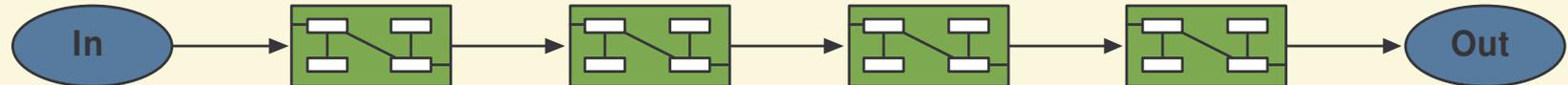
Stufe 5  
Optimierender  
Prozeß



Stufe 4  
Gesteuerter  
Prozeß



Stufe 3  
Definierter  
Prozeß



Stufe 2  
Wiederhol-  
barer Prozeß



Stufe 1  
Initialer  
Prozeß



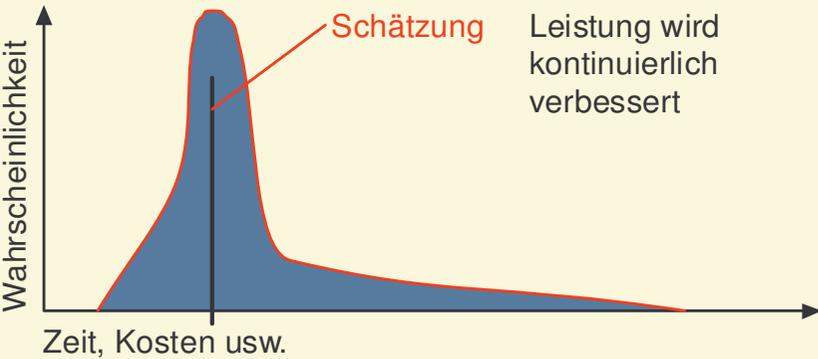
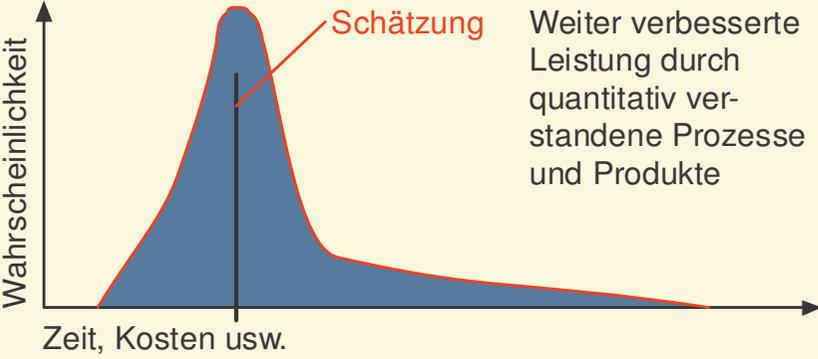
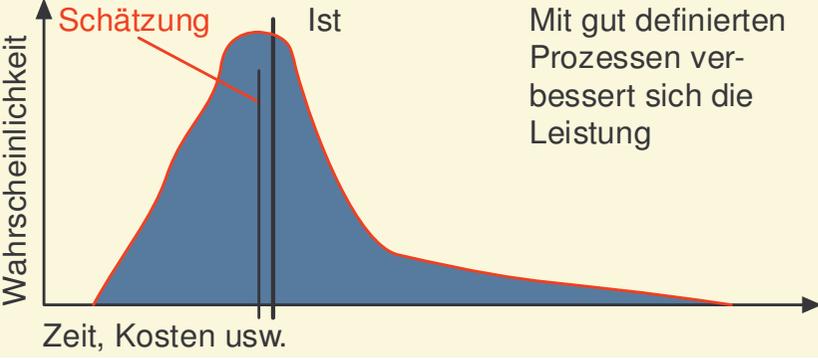
LE 13  
12

## 4.3.1 Die 5 Reifegradstufen

### ♦ Auswirkungen der CMM-Ebenen (1/2)

Stufe 2				
Wiederholbarer Prozeß <i>(repeatable)</i>		Pläne, basierend auf vergangenen Projekten, sind realistischer	Techniken unterstützen einige Tätigkeiten	Erfahrene Mitarbeiter halten den Prozeß am Leben
Stufe 1				
Initialer Prozeß <i>(initial)</i>		Zeit- und Kostenziele werden überschritten. Zu optimistische Schätzungen	Einführung neuer Techniken ist risikoreich	Regelmäßige Chaosbeseitigung, geringe Effizienz

## LE 13 4.3.1 Die 5 Reifegradstufen

Stufe 5	...auf die Prognosequalität	...auf die Technik	...auf die Mitarbeiter
Optimierender Prozeß <i>(optimizing)</i>	 <p>Wahrscheinlichkeit</p> <p>Zeit, Kosten usw.</p> <p>Schätzung</p> <p>Leistung wird kontinuierlich verbessert</p>	Technik und Prozeß unterstützen sich gegenseitig	Probleme werden verhindert, Mitarbeiter verbessern aktiv
Gesteuerter Prozeß <i>(managed)</i>	 <p>Wahrscheinlichkeit</p> <p>Zeit, Kosten usw.</p> <p>Schätzung</p> <p>Weiter verbesserte Leistung durch quantitativ verstandene Prozesse und Produkte</p>	Quantitative Basis für Techniken vorhanden	Verständnis für Zusammenhänge ist vorhanden
Definierter Prozeß <i>(defined)</i>	 <p>Wahrscheinlichkeit</p> <p>Zeit, Kosten usw.</p> <p>Schätzung</p> <p>Ist</p> <p>Mit gut definierten Prozessen verbessert sich die Leistung</p>	Qualitative Basis für Techniken vorhanden	Prozeß ist definiert, Mitarbeiter kennen und befolgen ihn

## 4.3.1 Die 5 Reifegradstufen

- ◆ **Reifegrade und ihre Erwartungen**
  - ◆ **Je höher der Reifegrad eines Prozesses ist, desto größer sind die erwarteten Verbesserungen bezüglich des Erreichens von Zielen**
    - **Zeit, Kosten usw.**
  - ◆ **Je höher der Reifegrad ist, desto geringer wird der Unterschied zwischen den geplanten Ergebnissen und den Ist-Ergebnissen**
    - **Organisationen auf der Stufe 1**
      - **Verfehlung ursprünglich geplanter Termine in einem großen Rahmen**
    - **Reifere Organisationen sollten die geplanten Termine mit besserer Genauigkeit einhalten.**

## 4.3.1 Die 5 Reifegradstufen

- ◆ Je höher der Reifegrad ist, desto geringer ist die Schwankungsbreite der aktuellen Ergebnisse um die Soll-Ergebnisse herum
  - In Organisationen der Stufe 1
    - Auslieferungstermine für Projekte gleichen Umfangs sind unvorhersehbar und variieren stark
  - Ähnliche Projekte werden in reiferen Organisationen innerhalb eines engeren Zeitbereichs fertiggestellt.

## 4.3.1 Die 5 Reifegradstufen

- ◆ Je reifer eine Organisation wird...
  - desto stärker sinken die Kosten
  - desto kürzer wird die Entwicklungszeit
  - desto mehr steigen Produktivität und Qualität
- In Stufe 1 kann die Entwicklungszeit durch viele notwendige Nacharbeiten sehr lang sein
- In reiferen Organisationen steigt die Prozeßeffizienz
- Es reduzieren sich die Nacharbeiten
- Die Entwicklungszeit verkürzt sich.

## 4.3.2 Die Hauptkriterien

- ◆ **Feststellung des Reifegrads**
  - ◆ Aufstellung von **Hauptkriterien** (*key process areas*) pro Reifestufe, die erfüllt sein müssen
  - ◆ Zuordnung von **Aspekten** (*key practices*) zu jedem Hauptkriterium
    - Geben an, was zu tun ist, um das jeweilige Hauptkriterium zu erfüllen
    - Geben aber **nicht** an, wie dies zu tun ist.

## 4.3.2 Die Hauptkriterien

- ◆ Fragebogen
  - ◆ Deckt alle 18 Hauptkriterien ab
  - ◆ Er bezieht sich auf die Ziele der Hauptkriterien, aber **nicht** auf alle Aspekte der Hauptkriterien
  - ◆ Pro Hauptkriterium gibt es 6 bis 8 Fragen
  - ◆ Jede Frage kann beantwortet werden mit
    - »Ja«
    - »Nein«
    - »nicht anwendbar«
    - »Ich weiß nicht«.

## Hauptkriterien zum Erreichen einer Reifestufe

- ◆ **Stufe 1: Initialer Prozeß (*initial*)**
  - ◆ entfällt
- ◆ **Stufe 2: Wiederholbarer Prozeß (*repeatable*)**
  - ◆ Ziel: Einführung einer grundlegenden Projektsteuerung und -überwachung
  - ◆ Anforderungsmanagement
    - Gemeinsames Verständnis zwischen Kunde und Projektteam über die Anforderungen herstellen
  - ◆ Projektplanung
    - Projektpläne einführen
  - ◆ Projektverfolgung und -überwachung
    - Transparenter Entwicklungsfortschritt, um frühzeitig Korrekturmaßnahmen einzuleiten.

## Hauptkriterien zum Erreichen einer Reifestufe

- ◆ Unterauftragsmanagement
  - Qualifizierte Unterlieferanten auswählen und effektiv steuern und überwachen
- ◆ Qualitätssicherung
  - Transparenter Prozeß und transparente Produkte
- ◆ Konfigurationsmanagement
  - Integrität der Produkte während ihrer Lebenszyklen sicherstellen
- ◆ **Stufe 3: Definierter Prozeß (*defined*)**
  - ◆ Ziel: Einführung einer projektübergreifenden Infrastruktur für Entwicklung und Management
  - ◆ Konzentration auf Prozeßorganisation
    - Gruppe einrichten, die für die Verbesserung des Software-Prozesses verantwortlich ist.

## Hauptkriterien zum Erreichen einer Reifestufe

- ◆ Definieren von Prozessen
  - Entwickeln und Pflegen einer brauchbaren Menge von Prozeßwerten, um die Prozesse zwischen Projekten zu verbessern
- ◆ Aufstellen eines Trainingsprogramms
  - Für das Training der Mitarbeiter ist eine organisatorische Einheit verantwortlich
- ◆ Integration von Software-Entwicklung
  - Entwicklungs- und Managementaktivitäten sind in einem zusammenhängenden, definierten Prozeß integriert
  - Standardprozesse können auf Projekte zugeschnitten werden (*tailored*).

## Hauptkriterien zum Erreichen einer Reifestufe

- ◆ **Software-Produkt-Engineering**
  - Konsistente Durchführung eines gut definierten Prozesses, der alle technischen Aktivitäten integriert, um korrekte, konsistente Produkte effektiv und effizient zu produzieren
- ◆ **Koordination aller beteiligten Gruppen**
  - Koordinationsmechanismen einführen, damit das Projektteam an den Ergebnissen anderer Gruppen partizipiert, um die Kundenwünsche besser zu erfüllen
- ◆ **Frühzeitige Fehlerbehebung**
  - Frühe und effiziente Fehlerbeseitigung aus Arbeitsprodukten z.B. durch *peer reviews*.

## Hauptkriterien zum Erreichen einer Reifestufe

- ◆ **Stufe 4: Gesteuerter Prozeß (*managed*)**
  - ◆ Ziel: Quantitatives Verstehen der Prozesse und der Arbeitsprodukte
  - ◆ Quantitatives Prozeßmanagement
    - Prozeßdurchführung quantitativ steuern und überwachen
  - ◆ Quantitatives Qualitätsmanagement
    - Quantitatives Verständnis der Produktqualität entwickeln, um spezifische Qualitätsziele zu erreichen
- ◆ **Stufe 5: Optimierender Prozeß (*optimizing*)**
  - ◆ Ziel: Einführung einer kontinuierlichen und meßbaren Prozeßverbesserung.

## Hauptkriterien zum Erreichen einer Reifestufe

- ◆ Fehlervermeidung
  - Identifizieren von Fehlerursachen und Fehlervermeidung durch Änderung des definierten Prozesses
- ◆ Innovationsmanagement
  - Identifizieren von neuen, nützlichen Techniken sowie deren geordnete Einführung
- ◆ Prozeßverbesserungsmanagement
  - Kontinuierliche Verbesserung der Prozesse mit folgenden Zielen:
    - Qualitätsverbesserung
    - Produktivitätssteigerung
    - Entwicklungszeitverkürzung.

## Beispiel für Verfeinerung eines Hauptkriteriums

- ◆ Qualitätssicherung, Hauptkriterium der Stufe 2
  - ◆ Ziele
    - 1 Die Übereinstimmung des Software-Produkts und des Software-Prozesses mit festgelegten Standards, Verfahren und Produktanforderungen wird von einer unabhängigen Instanz bestätigt
    - 2 Gibt es Übereinstimmungsprobleme, dann ist das Management darüber informiert
    - 3 Das höhere Management entscheidet Streitfragen
  - ◆ Managementvoraussetz. (*commitment to perform*)
    - 1 Die Organisation richtet sich nach einer schriftlichen Verfahrensweise zur Einführung der Software-Qualitätssicherung.

## Beispiel für Verfeinerung eines Hauptkriteriums

- ◆ Technische Voraussetzungen (*ability to perform*)
  - 1 Es stehen ausreichende Ressourcen und ein ausreichendes Budget zur Verfügung, um die QS-Aktivitäten durchzuführen
  - 2 Die QS-Mitarbeiter sind geeignet geschult
  - 3 Alle an der Software-Entwicklung beteiligten Personen werden über die Rolle, den Verantwortungsbereich und die Befugnisse der QS-Gruppe informiert
- ◆ Aktivitäten (*activities performed*)
  - 1 Ein QS-Plan ist entsprechend einem definierten Verfahren pro Projekt zu erstellen
  - 2 Die QS-Aktivitäten werden in Übereinstimmung mit dem QS-Plan durchgeführt.

## Beispiel für Verfeinerung eines Hauptkriteriums

- 3** Die QS-Gruppe ist an der Vorbereitung, der Überprüfung und der Genehmigung der Projektpläne, Prozeßspezifikationen, Standards und Verfahren beteiligt
- 4** Die QS-Gruppe prüft die Entwicklungs-aktivitäten, um die Übereinstimmung mit dem Prozeß sicherzustellen
- 5** Die QS-Gruppe prüft repräsentative Teile des Software-Produkts um sicherzustellen, daß sie mit den festgelegten Prozeßanforderungen übereinstimmen
- 6** Die QS-Gruppe erstattet regelmäßig der Entwicklung und dem Management Bericht über die Prüfergebnisse.

## Beispiel für Verfeinerung eines Hauptkriteriums

- 7 Festgestellte Abweichungen von den Entwicklungsaktivitäten werden dokumentiert und entsprechend dem festgelegten Verfahren abgehandelt
- 8 Die QS-Gruppe führt regelmäßige Überprüfungen ihrer Aktivitäten und Ergebnisse zusammen mit dem QS-Personal des Kunden durch
  - ◆ Überwachung der Aktivitäten (*monitoring implementation*)
- 1 Messungen werden durchgeführt, um die Kosten und den Zeitaufwand der QS-Aktivitäten zu bestimmen.

## Beispiel für Verfeinerung eines Hauptkriteriums

- ◆ Überprüfung der Aktivitäten (*verifying implementation*)
  - 1 Die QS-Aktivitäten werden regelmäßig vom höheren Management überprüft
  - 2 Die QS-Aktivitäten werden regelmäßig vom Projektleiter überprüft
  - 3 Die Aktivitäten der QS-Gruppe werden durch Manager überwacht, die nicht zum Software-Projekt gehören.

## 4.3.2 Die Hauptkriterien

### ◆ Beispiel

#### ◆ Die Fragen zu dem Hauptkriterium

»Qualitätssicherung« (Stufe 2) lauten:

- 1 Sind QS-Maßnahmen geplant?
- 2 Stellt die QS objektiv sicher, daß die Software-Produkte und -Aktivitäten den festgelegten Standards, Verfahren und Anforderungen entsprechen?
- 3 Werden die Ergebnisse der QS-Überprüfungen (*reviews, audits*) den betroffenen Gruppen und Personen zur Verfügung gestellt:
  - Denjenigen, die die Arbeit ausführen
  - Denjenigen, die für die Arbeit verantwortlich sind.

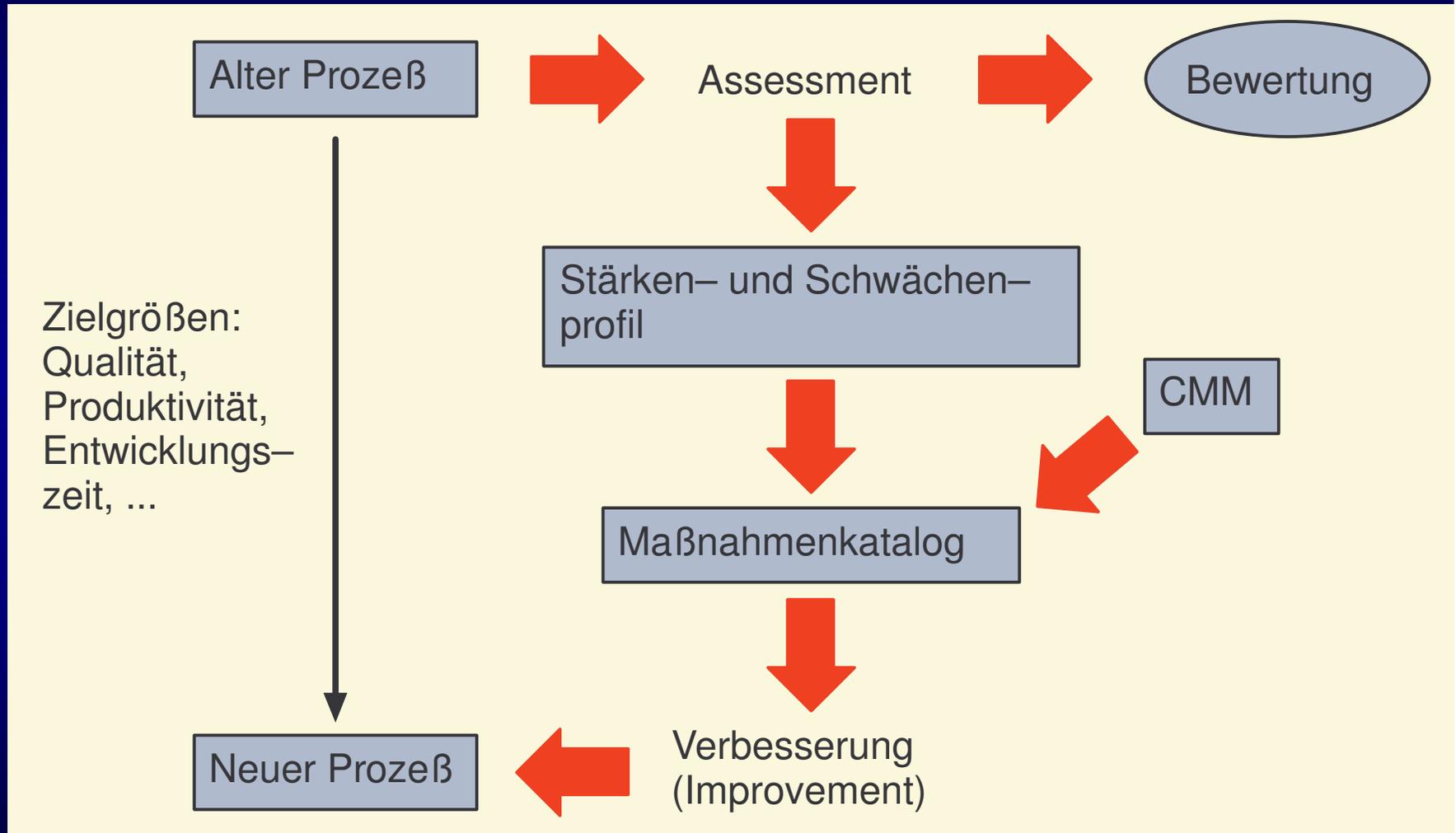
## 4.3.2 Die Hauptkriterien

- 4** Werden Abweichungen, die nicht innerhalb des Projekts gelöst werden, an das höhere Management berichtet
  - Beispiel: Abweichungen von festgelegten Standards
- 5** Folgt das Projekt einer schriftlich festgelegten QS-Politik?
- 6** Sind ausreichende Ressourcen vorhanden, um QS-Aktivitäten durchzuführen
  - Beispiel: Geld und ein verantwortlicher Manager, der bei Abweichungen aktiv wird.

## 4.3.2 Die Hauptkriterien

- 7 Werden Messungen dazu benutzt, Kosten und Zeitaufwand der durchgeführten QS-Aktivitäten zu bestimmen
  - Fertiggestellte Arbeit, Aufwand und Kosten verglichen mit dem Plan
- 8 Werden die QS-Aktivitäten regelmäßig vom höheren Management überprüft?

## 4.3.3 Durchführung von Prozeßverbesserungen



## Durchführung von *Assessments*

- ◆ **Vorbereitung**
  - ◆ Betroffene über CMM, *Assessments* und ihre Rolle informieren
  - ◆ Mitarbeiter aus den betreffenden Organisationseinheiten unter Umständen schulen
  - ◆ Atmosphäre des Vertrauens schaffen
- ◆ **Durchführung**
  - ◆ Unterschiedliche Personengruppen befragen (Management, Entwicklung, Qualitätssicherung)
  - ◆ Es sind sowohl die dokumentierte Prozeßdefinition (Soll-Situation) als auch ihre Umsetzung in die Praxis (Ist-Situation) zu bewerten.

## Durchführung von *Assessments*

- ◆ Es sind offene Interviews zu führen, d.h. es werden offene Fragen gestellt
  - Beispiel: »Wie wird bei Ihnen die Qualität und Eignung der Testfälle festgestellt?«
  - statt »Werden Testfälle formalen *Reviews* unterzogen?«
- ◆ Eine Bewertung der Fragen aufgrund der Schilderung vornehmen, unter Umständen Zusatzfragen stellen
- ◆ Das alleinige Beantworten der Fragen führt zu unzuverlässigen, unvollständigen Ergebnissen
- ◆ Wesentliche Aussagen mitschreiben als wichtige Information für das Stärken-und Schwächenprofil und als Basis für Verbesserungsvorschläge.

## Durchführung von *Assessments*

- ◆ Die Bewertungen der Fragen sind mit den Befragten unmittelbar anschl. zu diskutieren, um Mißverständnisse zu vermeiden
  - Beispiel: »Ich habe die Frage „Werden Testfälle formalen *Reviews* unterzogen?“ mit Nein bewertet, weil ... Habe ich Sie das richtig verstanden?«
- ◆ **Nachbereitung**
  - ◆ Aufzeigen...
    - der Soll-Situation
    - der Ist-Situation (Umsetzung in die Praxis)
    - des Verbesserungspotentials
  - ◆ Detailliertes Stärken- und Schwächenprofil nach Themenkomplexen ermitteln.

## 4.3.3 Durchführung von Prozeßverbesserungen

- ◆ **Maßnahmenkatalog & Einführungsplan**
  - ◆ Anhand des ermittelten Stärken- und Schwächenprofils
- ◆ **Prozeßverbesserungs-Team**
  - ◆ Koordiniert Umsetzung der Maßnahmen
  - ◆ Die Maßnahmen sollten in Teams mit Fach- und Anwendungswissen überarbeitet werden
    - Beispiel: Testexperte gemeinsam mit Testern.

## 4.3.4 Aufwand und Nutzen

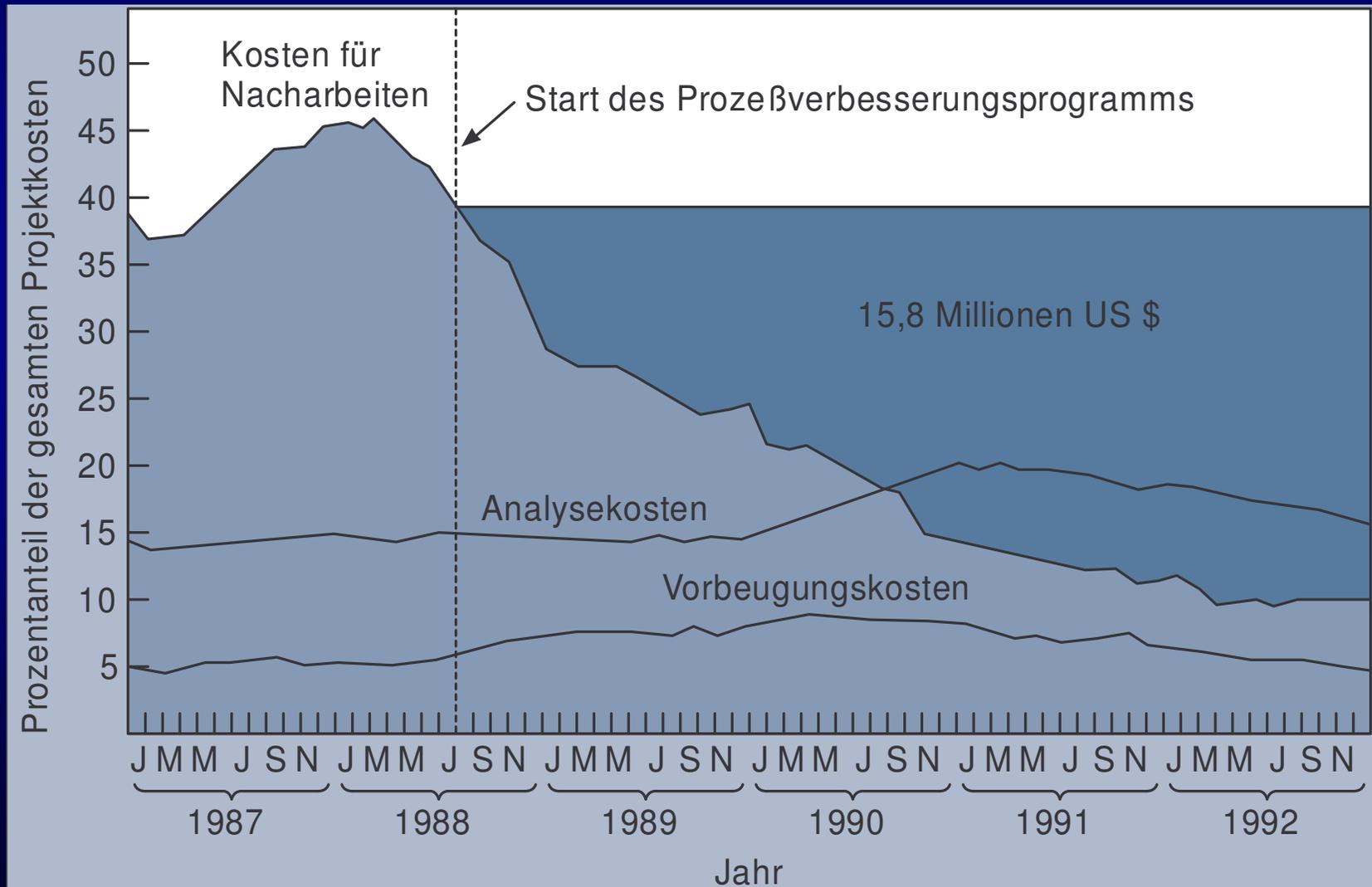
- ◆ Aufwand & Nutzen von Prozeßverbesserungen
- ◆ Beispiel 1
  - ◆ Firma **Hughes Aircraft**
    - 1987: Einordnung der Software-Entwicklung in die **Stufe 2**
  - ◆ **3 Jahre**, um die Prozeßqualität auf die **Stufe 3** anzuheben
  - ◆ Kosten für die *Assessments*: **45.000\$**
  - ◆ Kosten für **2 Jahre** Prozeßverbesserung: **400.000\$**
  - ◆ **Jährliche Einsparung** durch die bessere Prozeßqualität: **ca. 2 Mio. \$.**

## 4.3.4 Aufwand und Nutzen

- ◆ **Beispiel 2**
  - ◆ *Software Systems Laboratory* der Firma Raytheon
    - Führte 1988 ein eigenes *Assessment* durch
  - ◆ Ergebnis: Reifestufe 1
  - ◆ 5 Jahre lang: pro Jahr 1 Million \$ für die Prozeßverbesserung
  - ◆ Ende 1991: Reifestufe 3
  - ◆ Analyse der Kosten über 15 Projekte hinweg
    - Größte Einsparungen durch die **Reduzierung von Nacharbeiten.**

## 4.3.4 Aufwand und Nutzen

### ◆ Kostenverläufe bei Firma Raytheon



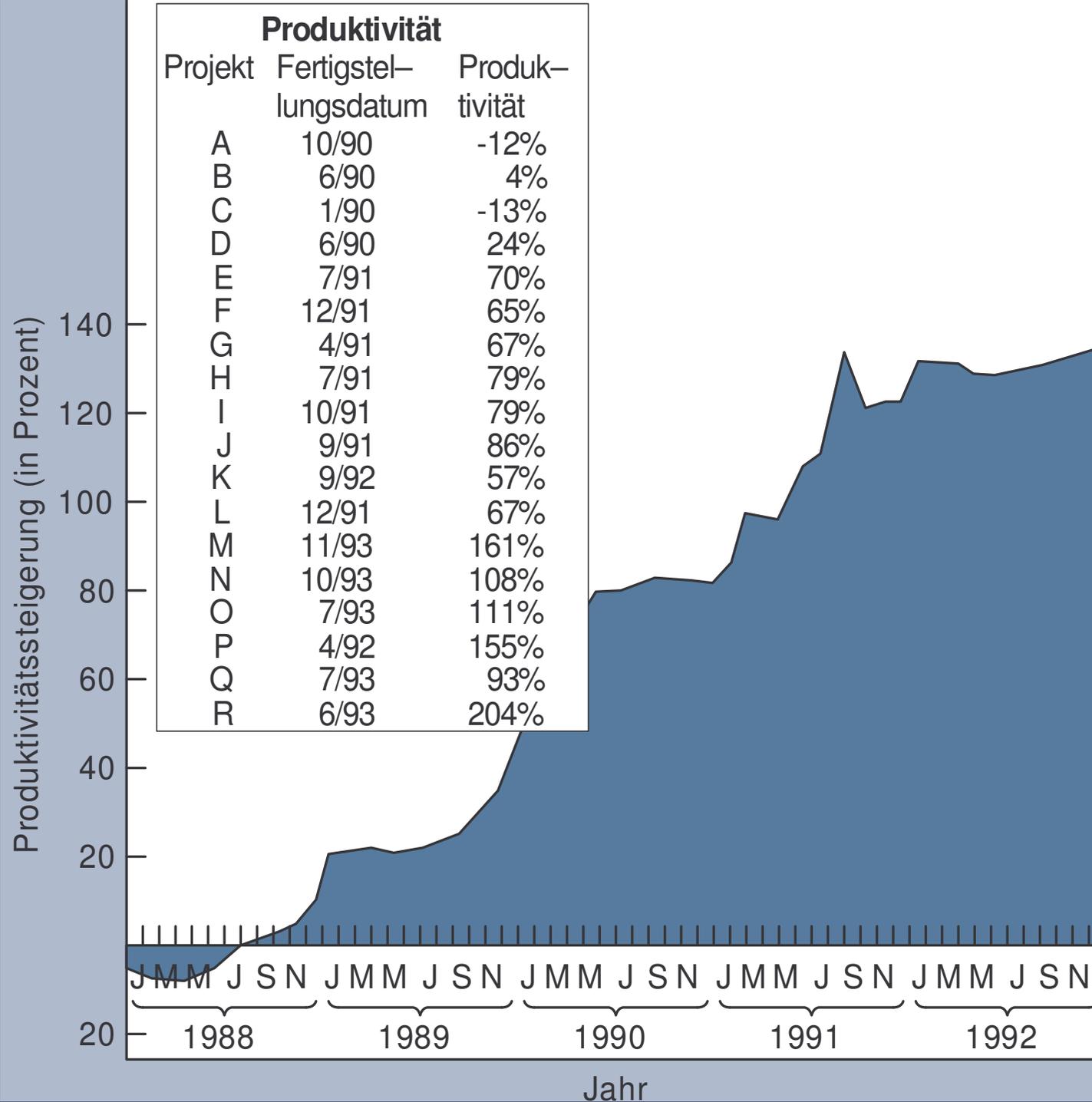
## 4.3.4 Aufwand und Nutzen

- ◆ 3 Kostenarten:
  - **Analysekosten**, um ein Produkt auf Fehler zu prüfen
  - **Nachbearbeitungskosten**, um Fehler zu beheben
  - **Vorbeugungskosten**, um zu verhindern, daß Fehler entstehen
- ◆ Bis 1992: 15,8 Mio. \$ **Nachbearbeitungskosten** eingespart
- ◆ **Analysekosten**: Erhöhung um 5%
  - Verursacht durch strengere Entwurfs- und Code-Inspektionen
  - Formale Inspektionen statt informale *Reviews*
- ◆ **Nachbearbeitungskosten**: Von 41% auf 11% der gesamten Prozeßkosten reduziert.

## 4.3.4 Aufwand und Nutzen

### ◆ *Return on investment*

- 1990: In 6 Projekten **58%** der Mitarbeiter gearbeitet
- Geht man davon aus, daß die Vorteile der Investitionen von 1 Mio. \$ ebenfalls zu **58%** den Projekten zugute gekommen sind, dann wurden durch **reduzierte Nacharbeiten** in diesen Projekten **4,48 Mio. \$** eingespart
- Durch eine Investition von **0,58 Mio. \$** erhielt man eine Ersparnis von **4,48 Mio. \$**
- *Return on Investment: 7,7 bzw. 770%.*



## 4.3.4 Aufwand und Nutzen

### ◆ Produktivität

- Maß: ausgelieferten Quellcode-Anweisungen pro Mitarbeitermonat
- Eingebettete Echtzeitanwendungen im Umfang zwischen 70.000 und 300.000 Anweisungen
- Über 15 Projekte hinweg ergab sich von 1988 bis 1990 ein Produktivitätszuwachs von **130%**
- Produktivität stieg um den Faktor **2,3** in **4 ½ Jahren.**

## 4.3.4 Aufwand und Nutzen

- ◆ **Vor der Prozeßverbesserungsinitiative**
  - Die meisten Projekte lagen hinter dem Zeitplan und über dem Budget
  - In vielen Fällen wurden Experten zu Projekten versetzt, die kritisch wurden
  - Es wurde immer versucht, das »Feuer zu bekämpfen«
- ◆ **Heute**
  - Die meisten Projekte sind im Zeitplan oder früher fertig und unter Budget.

## 4.3.4 Aufwand und Nutzen

### ◆ Beispiel 3

#### ◆ Einstufung amerikanischer Firmen in das Reifegrad-Modell

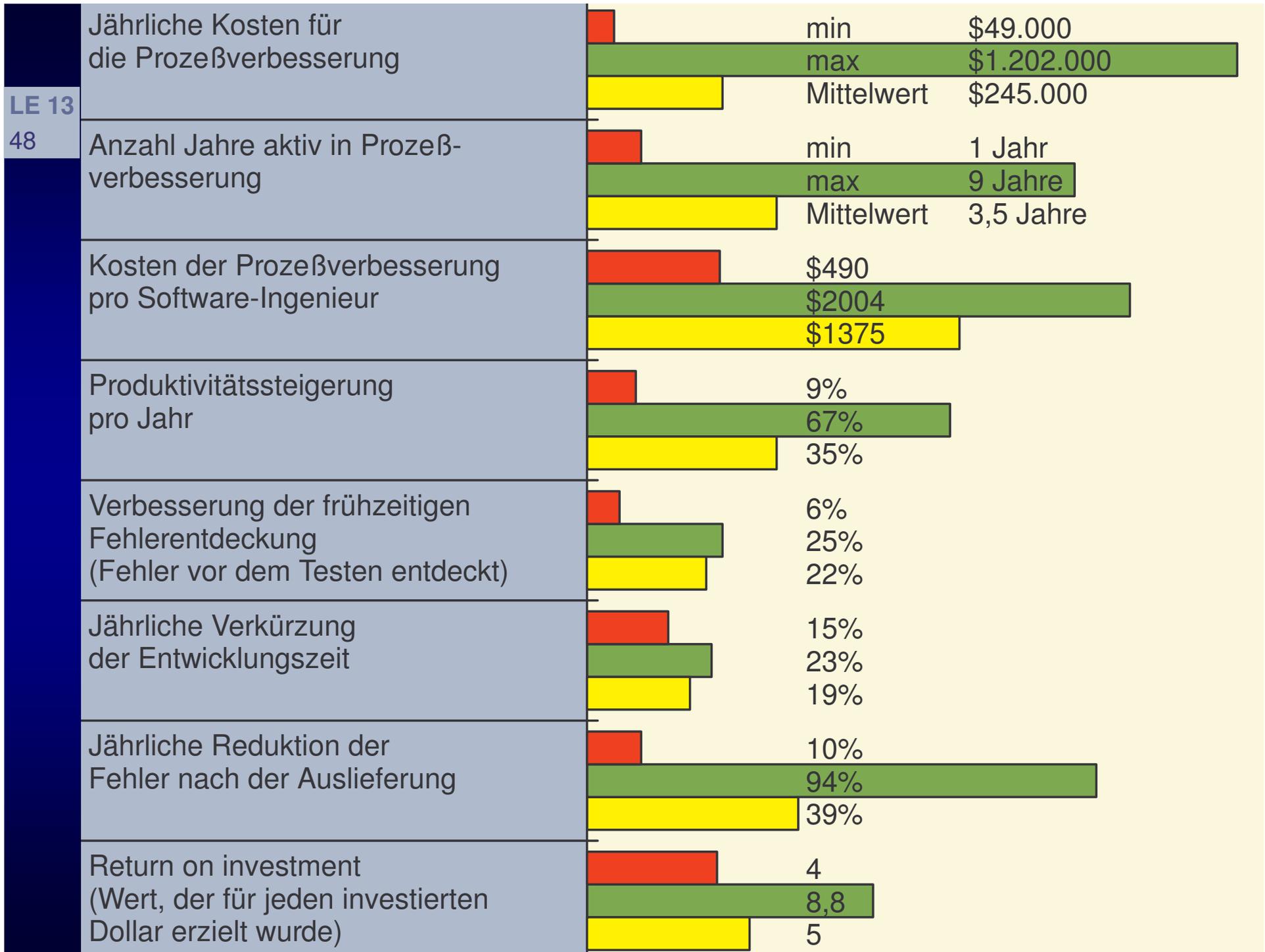
Amerikanische SW-Firmen Reifegrad des Unternehmens <sup>1)</sup>	1989	1991 Ein Standort <sup>2)</sup>	1991 Ein Projekt <sup>2)</sup>	IBM: Space Shuttle SW
Stufe 5 Optimierender Prozeß	0%	0%	2%	
Stufe 4 Gesteuerter Prozeß	0%	0%	0%	
Stufe 3 Definierter Prozeß	3%	7%	5%	
Stufe 2 Wiederholbarer Prozeß	22%	12%	5%	
Stufe 1 Initialer Prozeß	74%	81%	88%	

<sup>1)</sup> Quelle: Humphrey, Kitson, Tim 89

<sup>2)</sup> Quelle: Yourdon, »Decline and the Fall of the American Programmer«, S.86, SEI Assessments

## 4.3.4 Aufwand und Nutzen

- ◆ **Beispiel 4**
  - ◆ **Untersuchung von 13 Organisationen, die sich auf unterschiedlichen Reifestufen befinden, hat identifiziert...**
    - **substantielle Produktivitätssteigerungen**
    - **frühe Fehlerentdeckung**
    - **kurze Entwicklungszeiten**
    - **gute Qualität.**



LE 13

48

## 4.3.5 Vergleiche CMM vs. ISO 9000 vs. TQM

### ◆ CMM vs. ISO 9000

#### ◆ Frage 1

- Wenn meine Organisation die Stufe 3 in CMM erreicht hat, sind dann auch alle Anforderungen von ISO 9000 erfüllt?

#### ◆ Frage 2

- Wenn meine Organisation nach ISO 9000 zertifiziert ist, was hat CMM zusätzlich anzubieten?

## 4.3.5 Vergleiche CMM vs. ISO 9000 vs. TQM

### CMM-Assessments

### DIN ISO 9001

#### Gegenstand

Zur Zeit für reine Software-Entwicklungsprozesse vorgesehen

Vielzahl industrieller Organisationen, Produkte und Abläufe

#### Ziel

Detaillierte Ziel- und Prioritätenvorgaben zur Verbesserung des Prozesses

Nachweis der Qualifikation zur Erzeugung qualitätsgerechter Resultate

#### Status

Nützliches Hilfsmittel zur Problemanalyse und Prozeßverbesserung

Fester Industriestandard.

## 4.3.5 Vergleiche CMM vs. ISO 9000 vs. TQM

### CMM-Assessments

### DIN ISO 9001

#### Forderungen

Hierarchie von Forderungen in  
Abhängigkeit der Stufen

Minimalanforderungen  
(ausnahmslos zu erfüllen)

#### Basis

Flexibles *Capability Maturity  
Model*

Starrer Normentext

#### Ergebnis

Ist-Stand, Stärken- und  
Schwächen-Profil

Anerkanntes Zertifikat

#### Kosten vs. Nutzen

Einsparungen durch Prozeß-  
verbesserung vs. Kosten für  
die *Assessments* und die  
Verbesserungsaktivitäten

Nutzen ist durch das  
erteilte Zertifikat begründet .

## 4.3.5 Vergleiche CMM vs. ISO 9000 vs. TQM

- ◆ **CMM vs. ISO 9000**
  - ◆ Inhaltlich betrachtet gibt es zwischen beiden Ansätzen sowohl **Überschneidungen** als auch **Differenzen**, die sich im wesentlichen aus der Zielsetzung und dem Gegenstand der Betrachtung ergeben
  - ◆ CMM enthält...
    - Einführung von statistischen Methoden
    - Definition von Standards
    - Einführung von Techniken.

## 4.3.5 Vergleiche CMM vs. ISO 9000 vs. TQM

### ◆ ISO 9001 enthält...

- Abnahmekriterien für jede Phase des Entwurfs und der Entwicklung (4.4.4)
- Lenkung der Dokumente und Daten (4.5)
- Lenkung von Qualitätsaufzeichnungen (4.16)
- Festlegung von Verantwortungsbereichen (4.1.2.1) und kompetentes Personal (4.18)
- Handhabung, Lagerung, Verpackung, Konservierung und Versand (4.15), Wartung (4.19)
- Kriterien für die Beurteilung von Unterauftragnehmern (4.6.2), zugekauften Produkten (4.6.4) und beigestellten Produkten des Kunden (4.7).

## 4.3.5 Vergleiche CMM vs. ISO 9000 vs. TQM

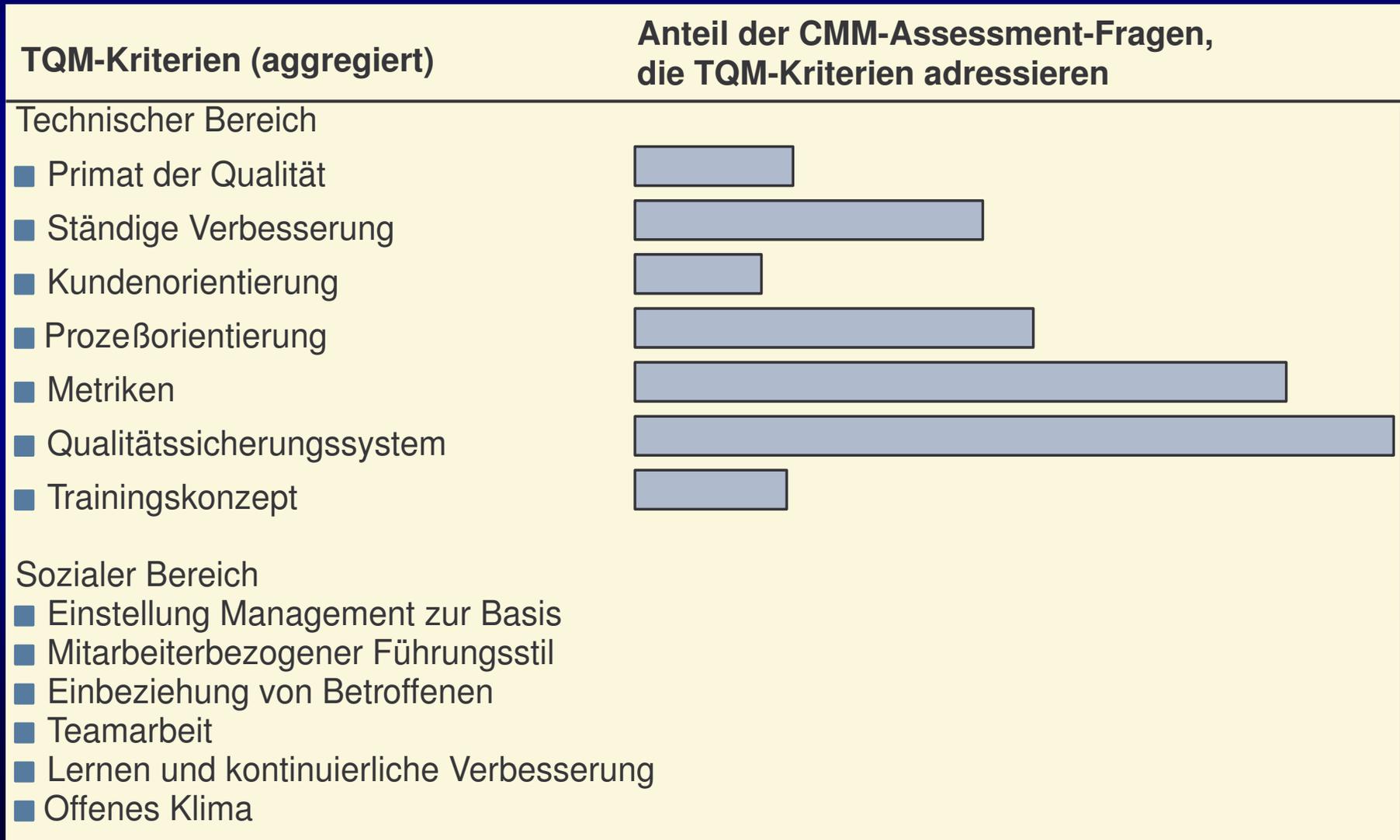
- ◆ Beide Ansätze enthalten...
  - unabhängige Audits der Entwicklungsaktivitäten
  - korrigierende Aktionen
  - Mitarbeitertraining
  - detaillierte Definition des Prozeß- und Lebenszyklus
- ◆ Zu Frage 1
  - ◆ Ist die CMM-Stufe 3 erreicht, dann ist noch einiges zu tun, um ISO9001 zu erfüllen, da einige Bereiche durch CMM nicht abgedeckt werden
- ◆ Zu Frage 2
  - ◆ Ist ISO 9001 erfüllt, dann gibt CMM zusätzliche Hilfestellung insbesondere auf den Gebieten der Technik, der Prozeßdefinition und der Metriken.

## 4.3.5 Vergleiche CMM vs. ISO 9000 vs. TQM

- ◆ **CMM vs. ISO 9000: Zusammenfassung**
  - ◆ Schwerpunkt der ISO 9001-Zertifizierung ist der Nachweis eines QM-Systems entsprechend der Norm
  - ◆ CMM-Ansatz konzentriert sich demgegenüber auf die Qualitäts- und Produktivitätssteigerung des gesamten Software-Entwicklungsprozesses
  - ◆ ISO 9000 und CMM sind keine Alternativen, sondern **ergänzen sich**
  - ◆ Aufgrund der unterschiedlichen Schwerpunkte gibt es keine »Umrechnungsformel« zwischen ISO 9000-Zertifizierung und den CMM-Ebenen.

## 4.3.5 Vergleiche CMM vs. ISO 9000 vs. TQM

### ◆ CMM vs. TQM



## 4.3.5 Vergleiche CMM vs. ISO 9000 vs. TQM

### ◆ CMM vs. TQM

- ◆ TQM-Ansatz ist **umfassender** als der CMM-Ansatz
- ◆ **CMM**-Ansatz bezieht sich ausschließlich auf **technische Aspekte** eines sozio-technischen Systems
- ◆ CMM-Schwerpunkte liegen auf dem QS-System und den Metriken
  - **Unterrepräsentiert** sind die Bereiche
    - Primat der Qualität
    - Kundenorientierung
    - Trainingskonzepte.

## 4.3.6 Vor- und Nachteile

- ◆ **CMM-Ansatz einschl. der *Assessments***
  - + Bietet eine systematische Möglichkeit zur Verbesserung der Prozeßqualität
  - + Bei sorgfältiger Anwendung der *Assessments* werden Schwächen des Entwicklungsprozesses identifiziert, deren Behebung sehr wirksam ist
  - + Der Nutzen ist wesentlich größer als die Kosten
  - + Es gibt bei den Entwicklungsprozessen ein großes Verbesserungspotential, da viele Firmen sich auf Techniken und Werkzeuge konzentriert haben und den Entwicklungsprozeß vernachlässigt haben
  - + Erlaubt die Evaluierung des gegenwärtigen Prozeßzustandes einer Organisation und damit auch einen Vergleich mit anderen Organisationen.

## 4.3.6 Vor- und Nachteile

- Kein garantierter Zusammenhang zwischen hohem Reifegrad und erfolgreicher Software-Produktion
- Stark technikbezogen, wenig personalbezogen
- Für die Stufen 4 und 5 gibt es nur wenige gesicherte Erkenntnisse
- Der Zusammenhang zwischen dem Fragenkatalog und dem CMM ist nicht immer sichtbar
- Wichtige Kerngebiete fehlen, z.B. Risikomanagement
- Um eine hohe Stufe zu erreichen, müssen alle Forderungen der niedrigeren Stufen erfüllt sein
- Für technische Anwendungsbereiche nicht optimal
- Fragen können nur mit **Ja** oder **Nein** beantwortet werden; es gibt keine Abstufungen.

## 4.3.6 Vor- und Nachteile

### ◆ CMM V2.0

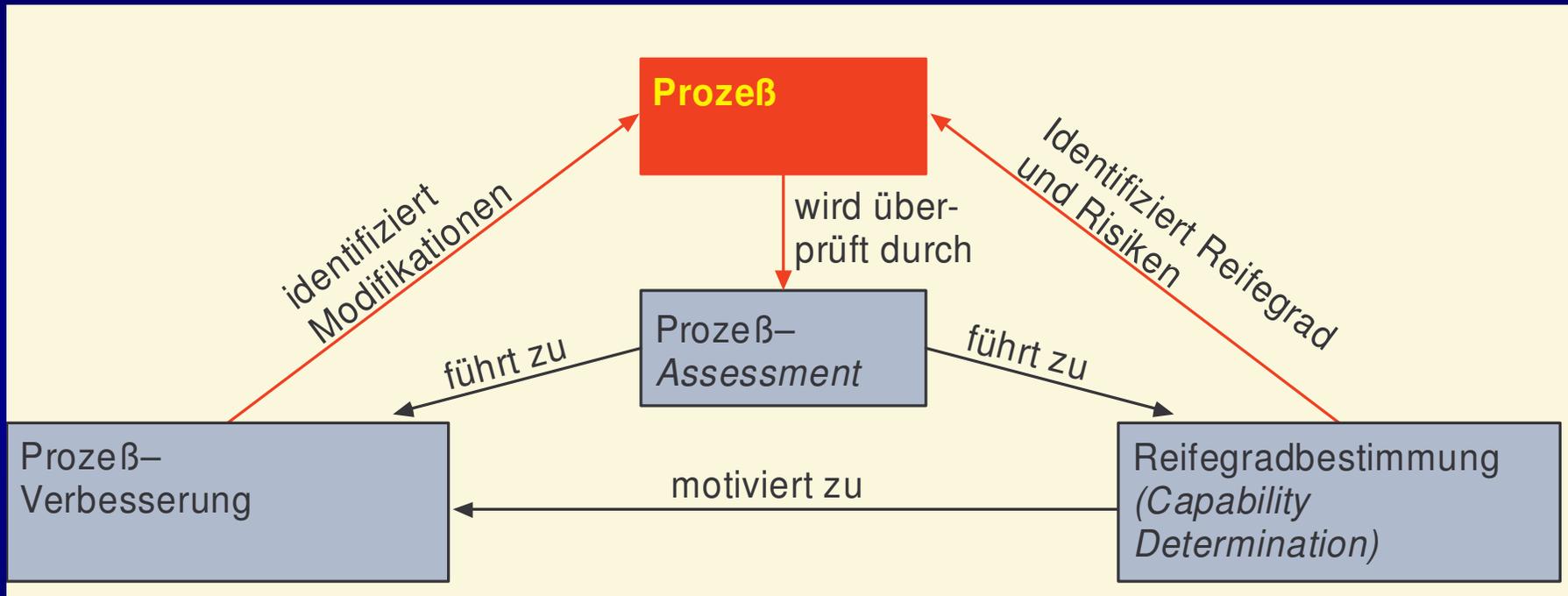
- ◆ Hauptkriterien (*key process areas*) werden **nicht** einer Reifegradstufe zugeordnet, sondern werden sich über mehrere Stufen erstrecken
- ◆ Die Beschreibungen und Empfehlungen für die Stufen 4 und 5 werden detaillierter und verbessert
- ◆ Neu: Ermittlung von Kundenanforderungen
- ◆ Befähigung der Mitarbeiter und die Unterstützung des Wandels der Unternehmenskultur bekommen ein stärkeres Gewicht
- ◆ Stärkere Betonung der Schnittstelle zwischen Kunden und Lieferanten
- ◆ Einige Hauptkriterien, z.B. Testen von Software, werden aufgewertet.

## 4.4 Der SPICE - Ansatz

### ◆ SPICE

- ◆ *Software **P**rocess **I**mprovement and **C**apability **D**etermination*
- ◆ Umfassender, ordnender Rahmen zur Bewertung und Verbesserung von Software-Prozessen
- ◆ Vorhandene Ansätze wie ISO 9000 und CMM sollen integriert und vereinheitlicht werden
- ◆ SPICE lehnt sich an das CMM an
- ◆ Mit der Verabschiedung von SPICE als ISO-Norm 15504 ist 1998 zu rechnen.

## 4.4.1 Die Struktur von SPICE

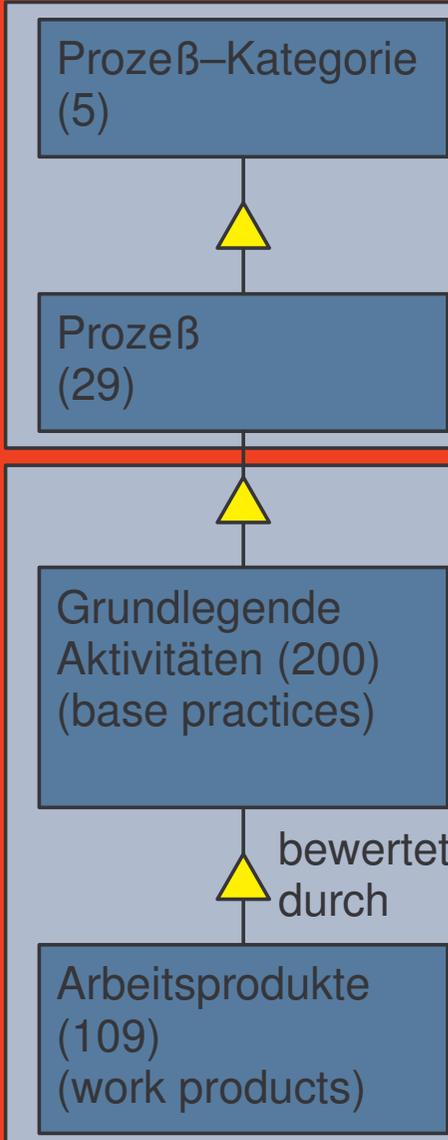


## 4.4.1 Die Struktur von SPICE

- ◆ **Prozeß-Assessments**
  - ◆ Stehen im Mittelpunkt
  - ◆ Dienen sowohl zur Reifegradbestimmung der Prozesse als auch zum Aufzeigen von Prozeßverbesserungen durch geeignete Modifikationen der Prozesse
  - ◆ Bewertung der eigenen Software-Entwicklung
  - ◆ Bewertung anderer Unternehmen
  - ◆ Schwerpunkt
    - Selbstbewertung (*Self-Assessment*), **nicht** Zertifizierung.

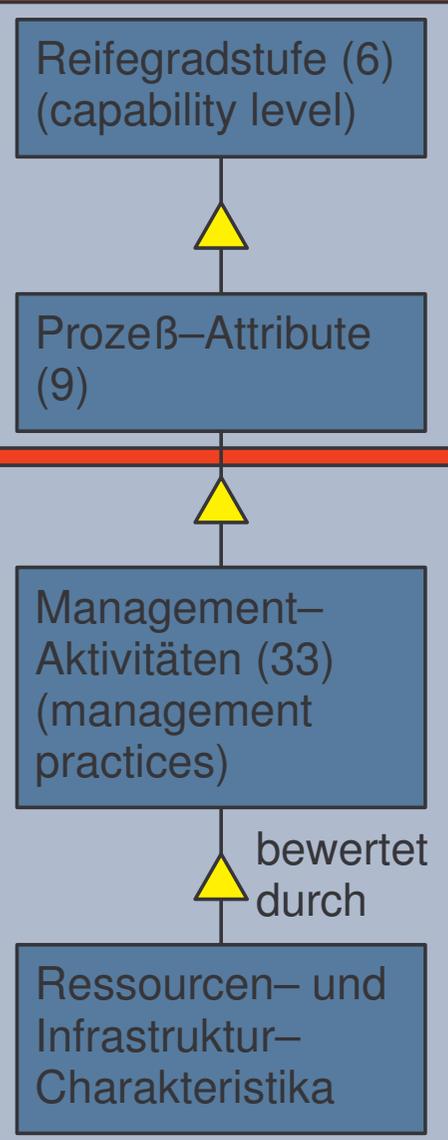
# Assessment-Modell

## Prozeß-Dimension



Referenzmodell

## Reifegrad-Dimension



Assessment-Indikatoren

Legende:  
A | B ist Teil von A  
▲

## 4.4.1 Die Struktur von SPICE

- ◆ **Prozeß-Assessments**
  - ◆ Werden anhand des SPICE-Referenzmodells durchgeführt
- ◆ **Referenz- und *Assessment*-Modell**
  - ◆ 2 Dimensionen:
    - Prozeß-Dimension dient zur Kennzeichnung der Vollständigkeit von Prozessen
    - Reifegrad-Dimension dient zur Kennzeichnung der Leistungsfähigkeit von Prozessen.

## 4.4.1 Die Struktur von SPICE

- ◆ **Bewertung von Prozessen**
  - ◆ **Durchführungsindikatoren in Form von grundlegenden Aktivitäten (*base practices*) und Arbeitsprodukten (*work products*) mit ihren Charakteristika**
- ◆ **Bewertung von Prozeß-Attributen**
  - ◆ **Management-Aktivitäten (*management practices*) sowie Ressourcen- und Infrastrukturcharakteristika.**

## 4.4.2 Die Prozeß-Dimension

### ◆ Prozeß-Dimension

- ◆ Jeder Prozeß wird einer von 5 Kategorien zugeordnet

#### 1 **Kunden-Lieferanten-Prozeßkategorie** (*Customer-Supplier process category*)

- Beschreibt Prozesse, die unmittelbar den Kunden betreffen, wie Software-Akquisition, Kundenbetreuung, Software-Lieferung, Software-Einsatz, Kundendienst

#### 2 **Entwicklungsprozeß-Kategorie** (*Engineering process category*)

- Umfaßt Prozesse, die dazu dienen, ein Software-Produkt zu definieren, zu entwerfen, zu implementieren und zu warten.

## 4.4.2 Die Prozeß-Dimension

### 3 Kategorie »**Unterstützende Prozesse**« (*Support process category*)

- Prozesse, die andere Prozesse unterstützen, z.B. Dokumentation, Konfigurationsmanagement, QS

### 4 **Managementprozeß-Kategorie**

- Prozesse, die notwendig sind, um Software-Projekte zu planen, zu steuern und zu kontrollieren, z.B. Projektmanagement, QM, Risikomanagement, Lieferantenmanagement

### 5 **Organisationsprozeß-Kategorie**

- Prozesse, die es ermöglichen, Unternehmensziele zu definieren und durch Ressourcen zu erreichen, z.B. Prozeßdefinition, -verbesserung, Personalmanagement, CASE.

## 4.4.2 Die Prozeß-Dimension

- ◆ **29 Prozesse**
  - ◆ 29 Prozesse sind definiert
  - ◆ Sind jeweils einer Prozeßkategorie zugeordnet
  - ◆ Jeder Prozeß selbst wird durch grundlegende Aktivitäten beschrieben, die die Aufgaben definieren, um das Prozeßziel zu erreichen
- ◆ **200 Aktivitäten (*base practices*)**
  - ◆ 200 Aktivitäten sind definiert
- ◆ **Ein- und Ausgabeprodukte**
  - ◆ Mit ihren Charakteristika beschrieben, die jedem Prozeß zugeordnet sind.

## 4.4.2 Die Prozeß-Dimension

### ◆ Beispiel

- ◆ Prozeß »Integriere und teste Software« gehört in die **Entwicklungsprozeß-Kategorie**
- ◆ Aufgabe
  - Software-Einheiten mit anderer produzierter Software zu integrieren, um die Software-Anforderungen zu erfüllen
- ◆ Prozeß wird schrittweise durch Einzelne oder Teams durchgeführt
- ◆ Durch 7 Aktivitäten beschrieben:
  - Ermittle eine Regressionsteststrategie
    - Lege eine Strategie für das erneute Testen von Aggregaten fest, wenn eine Änderung in einer gegebenen Software-Einheit durchgeführt wird.

## 4.4.2 Die Prozeß-Dimension

- **Bilde Aggregate von Software-Einheiten**
  - Identifiziere Aggregate von Software-Einheiten und bilde eine Sequenz oder partielle Ordnung, um sie zu testen.
- **Entwickle Tests für die Aggregate**
  - Beschreibe die Tests, die mit den Aggregaten durchgeführt werden, und lege die zu prüfenden Software-Anforderungen, Eingabedaten und Abnahmekriterien fest
- **Teste die Software-Aggregate**
  - Teste jedes Software-Aggregat gegen die Abnahmekriterien und dokumentiere die Ergebnisse
- **Integriere die Software-Aggregate**
  - Integriere die aggregierten Software-Komponenten, um ein vollständiges System zu bilden.

## 4.4.2 Die Prozeß-Dimension

- **Entwickle Tests für die Software**
  - Beschreibe die Tests, die mit der integrierten Software ausgeführt werden sollen, und lege die zu prüfenden Software-Anforderungen, Eingabedaten und Abnahmekriterien fest
  - Die Tests sollen die Übereinstimmung mit den Software-Anforderungen demonstrieren und die interne Struktur der Software überdecken
- **Teste die integrierte Software**
  - Teste die integrierte Software gegen die Abnahmekriterien und dokumentiere die Ergebnisse.

## 4.4.2 Die Prozeß-Dimension

- ◆ Dem Prozeß sind Arbeitsprodukte zugeordnet:

### **Eingabeprodukte**

Systemanforderungen  
Software-Anforderungen  
Wartungsanforderungen  
Änderungskontrolle  
Software-Entwurf  
(Architekturentwurf)  
Software-Entwurf  
(Implementierungsentwurf)  
Systementwurf/-architektur  
Software-Einheiten (Code)  
*Release-Strategie/-Plan.*

### **Ausgabeprodukte**

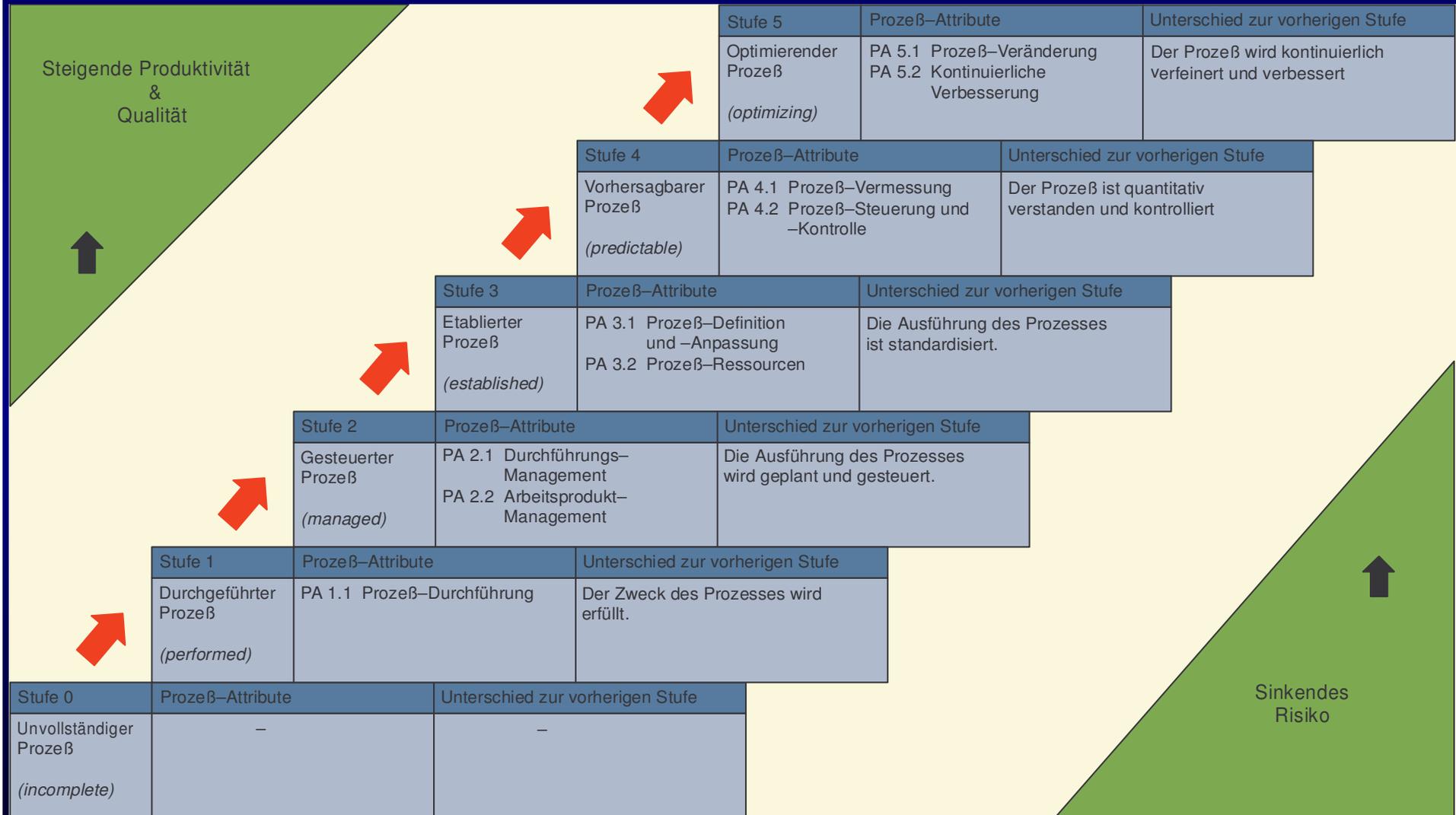
Regressionstest-Strategie  
*Traceability*-Aufzeichnung  
Integrationsteststrategie/-plan  
Integrationstestskript  
Software-Testplan  
Software-Testskript  
Testfälle  
Testergebnisse  
Integrierte Software

## 4.4.3 Die Reifegrad-Dimension

- ◆ **Leistungsfähigkeit von Prozessen**
  - ◆ wird durch Prozeß-Attribute ausgedrückt
  - ◆ Prozeß-Attribute sind zu Reifegradstufen gruppiert
  - ◆ 6 Reifegradstufen.

## 4.4.3 Die Reifegrad-Dimension

### ◆ Die SPICE-Reifegradstufen



## 4.4.3 Die Reifegrad-Dimension

- ◆ **Unterschied zum CMM**
  - ◆ Mit den Reifegraden werden nicht Unternehmen oder Projekte beurteilt, sondern Prozesse
  - ◆ Reifegradstufen können benutzt werden, um die Vollständigkeit und Leistungsfähigkeit der Prozesse eines Unternehmens zu bewerten
  - ◆ Beschreibungen des jeweils nächsthöheren Reifegrads geben Hinweise für Prozeßverbesserungen
  - ◆ Leistungsfähigkeit der in der Prozeß-Dimension beschriebenen 29 Prozesse wird durch 9 Prozeß-Attribute beurteilt
  - ◆ Ein Prozeß-Attribut repräsentiert eine meßbare Charakteristik jedes Prozesses.

## 4.4.3 Die Reifegrad-Dimension

- ◆ **Bewertung eines Prozeß-Attributs:**
  - ◆ vollständig erfüllt
  - ◆ weitgehend erfüllt
  - ◆ teilweise erfüllt
  - ◆ nicht erfüllt
- ◆ **Jedes Prozeß-Attribut ist einer Reifegradstufe zugeordnet**
  - ◆ Zur Überprüfung, inwieweit die Prozeß-Attribute durch einen Prozeß erfüllt werden, sind jedem Prozeß-Attribut **Management-Aktivitäten** zugeordnet.

## 4.4.3 Die Reifegrad-Dimension

### ◆ Beispiel

- ◆ Der Reifegradstufe 1 ist das Prozeß-Attribut **PA 1.1 »Prozeß-Durchführung«** zugeordnet:
  - Grad, in dem bei der Ausführung des Prozesses Aktivitäten durchgeführt werden, so daß festgelegte Eingabeprodukte verwendet werden, um festgelegte Ausgabeprodukte zu erzeugen, die dazu geeignet sind, den Zweck des Prozesses zu erfüllen
- ◆ Dem Attribut **PA 1.1** ist folgende **Management-Aktivität** zugeordnet:
  - Sicherstellen, daß die grundlegenden Aktivitäten ausgeführt werden, um den Zweck des Prozesses zu erfüllen.

## 4.4.3 Die Reifegrad-Dimension

- ◆ Die **Leistungscharakteristika** für diese **Management-Aktivität** lauten
  - Die Prozeß-Verantwortlichen können zeigen, daß die grundlegenden Aktivitäten für den Prozeß durchgeführt werden (selbst wenn der Prozeß nicht dokumentiert ist), um den Prozeßzweck zu erreichen
  - In jeder überprüften organisatorischen Einheit spricht vieles dafür, daß die grundlegenden Aktivitäten auch wirklich durchgeführt werden
  - Muster für die Eingabe- und Ausgabeprodukte, die für den betreffenden Prozeß festgelegt sind, existieren und besitzen die geforderten Charakteristika.

### 4.4.3 Die Reifegrad-Dimension

- Es gibt einen Verteilungsmechanismus für die dem Prozeß zugeordneten Arbeitsprodukte
- Die Ressourcen, die für die Ausführung des Prozesses benötigt werden, stehen zur Verfügung
- Die erstellten Arbeitsprodukte erfüllen den Prozeßzweck.

## 4.4.4 Vor- und Nachteile

- ◆ **Bewertung des SPICE-Ansatzes**
  - + **Prozeß-Assessments** dienen sowohl zur Reifegradbestimmung des Prozesses als auch dazu, Prozeßverbesserungen aufzuzeigen
    - Sie erlauben relativ genaue Einblicke in die Prozesse
  - + **Prozeß-Assessments** können auch zu Prozeß-Profilen führen, die einen schnellen Überblick über Stärken und Schwächen einzelner Prozesse ermöglichen
  - + Orientierung an bereits existierenden Ansätzen, z.B. CMM und ISO 9000.

## 4.4.4 Vor- und Nachteile

- + **Zusätzliche Reifegradstufe 1, die besonders für kleine Organisationen sinnvoll ist**
  - **Dieser Reifegrad entspricht der Durchführung bestimmter Tätigkeiten, ohne daß diese einem klaren Management unterliegen**
- + **Genereller Rahmen für die Bewertung und Verbesserung von Software- Prozessen, unabhängig von speziellen Methoden, Konzepten und Werkzeugen**
- + **Kundenorientierung wird im Gegensatz zum CMM explizit berücksichtigt.**

## 4.4.4 Vor- und Nachteile

- + Die einzelnen Prozesse können sich auf einer unterschiedlichen Reifegradstufe befinden
- + Umfangreiches, durchdachtes Referenz- und *Assessment*-Modell
- Außer einigen Feldstudien liegen bisher noch keine Anwendungserfahrungen aus der Unternehmenspraxis vor
- Die für die Reifegradstufen 4 und 5 festgelegten Prozesse und Prozeß - Attribute sind weder theoretisch fundiert noch empirisch abgesichert
- Ein weiterer Ansatz zur Verbesserung der Prozeßqualität; dadurch wird die Auswahl eines Ansatzes weiter erschwert.

## 4.5 *Business Engineering*

- ◆ **Verbesserung der Prozeßqualität**
  - ◆ evolutionäre Ansätze
  - ◆ **revolutionärer Ansatz:**
    - *Business Engineering*
    - *Business (Process) Reengineering*
    - *Business Improvement*
  - ◆ Ziel
    - Unternehmen und ihre Geschäftsprozesse **ingenieurmäßig** gestalten
  - ◆ Forderung
    - Geschäftsprozesse in Unternehmen müssen grundlegend neu überdacht und gestaltet werden, um **gravierende** Verbesserungen und nicht nur marginale Verbesserungen zu erzielen.

## 4.5 *Business Engineering*

- ◆ **Begründung für revolutionären Ansatz**
  - ◆ Die heutige Informations- und Kommunikationstechnik ist in der Lage, benötigte Informationen zu jeder Zeit im gewünschten Umfang am geforderten Arbeitsplatz zur Verfügung zu stellen
  - ◆ Da die meisten Unternehmen diese technischen Möglichkeiten noch nicht nutzen oder nur auf die veralteten Prozeßabläufe »aufpfropfen«, sind radikale Änderungen erforderlich
  - ◆ Frage?
    - Trifft diese Analyse auch auf die Prozesse der **Software-Entwicklung** zu.

## 4.5 *Business Engineering*

- ◆ **Forderungen an Geschäftsprozesse**
  - ◆ **Bezogen auf Software-Entwicklungsprozesse**
  - ◆ **Mehrere Positionen werden zusammengefaßt**
    - **Geschäftsprozesse werden oft segmentiert und durch verschiedene Personen oder Gruppen bearbeitet**
    - **Wird ein Geschäftsprozeß an einem einzigen integrierten Arbeitsplatz bearbeitet – oder bei komplexen Prozessen an wenigen Arbeitsplätzen – dann entfallen Übernahmeprozeduren**
    - **Daraus resultierende Fehler und Nacharbeiten nehmen ab, und die Verwaltungsgemeinkosten sinken.**

## 4.5 *Business Engineering*

- **Software-Entwicklung**

- Diese Forderung ist meistens erfüllt
- Gerade diese Situation führt in der Software-Entwicklung aber zu Problemen, da das benötigte Spezialwissen heute so hoch ist, daß kein Mitarbeiter mehr alle Aufgaben in einer Software-Entwicklung optimal bearbeiten kann

- ◆ **Mitarbeiter fällen Entscheidungen**

- Entscheidungen werden vertikal komprimiert, d.h. Mitarbeiter treffen selbständig Entscheidungen, wo sie früher Vorgesetzte fragen mußten
- **Software-Entwicklung**
  - Durch das hohe Qualifikationsniveau von Software-Ingenieuren dürfte diese Forderung schon immer weitgehend erfüllt gewesen sein.

## 4.5 *Business Engineering*

- ◆ **Die einzelnen Prozeßschritte werden in eine natürliche Reihenfolge gebracht**
  - Prozeßschritte sind zu »entlinearisieren«
  - Die Reihenfolge der Prozeßschritte orientiert sich an der Frage »Was muß auf einen bestimmten Schritt folgen?«
  - Dadurch können viele Arbeitsschritte gleichzeitig erledigt werden
  - **Software-Entwicklung**
    - Die verschiedenen Vorgehensmodelle in der Software-Technik zeigen – in Abhängigkeit von Zielen und Randbedingungen – geeignete Reihenfolgen
    - Das nebenläufige Vorgehensmodell versucht, möglichst viele Arbeiten parallel zu erledigen.

## 4.5 *Business Engineering*

### ◆ **Es gibt mehrere Prozeßvarianten**

- **Es werden mehrere Varianten des gleichen Geschäftsprozesses benötigt, die jeweils auf die Anforderungen unterschiedlicher Märkte, Situationen und Informationen zugeschnitten sind**
- **Beispielsweise ist eine Bestellung unter 100 DM anders zu behandeln als über 10.000 DM**
- **Das bedeutet das Ende der Standardisierung**
- **Software-Entwicklung**
  - **Das Spiralmodell ist ein Metamodell, das die Auswahl geeigneter Vorgehensmodelle in Abhängigkeit von der jeweiligen Entwicklungssituation unterstützt**
  - **Die Reifegradstufe 5 »optimierender Prozeß« im CMM-Ansatz strebt ebenfalls Prozeßvarianten an.**

## 4.5 *Business Engineering*

- ◆ **Die Arbeit wird dort erledigt, wo es sinnvoll ist**
  - Arbeit über organisatorische Grenzen hinweg neu verteilen, um Prozeßleistung zu verbessern
  - Prozeß kann auf Prozeßkunden verlagert werden, um Übergabeprozeduren zu eliminieren und Gemeinkosten zu senken
  - Beispiel: Beschaffungen unter 500 DM werden von jeder Einheit selbst durchgeführt
  - Manchmal effizienter, wenn der Lieferant seinem Kunden den Prozeß ganz oder teilweise abnimmt
  - **Software-Entwicklung**
    - Diese Forderung **widerspricht** teilweise dem Prinzip der unabhängigen Qualitätssicherung
    - Produktqualität wird am sinnvollsten dort geprüft, wo das Produkt erstellt wird.

## 4.5 *Business Engineering*

- ◆ **Weniger Überwachungs- und Kontrollbedarf**
  - Unternehmensprozesse werden nur in dem Maße kontrolliert, in dem dies wirtschaftlich sinnvoll ist
  - An die Stelle einer starren Überprüfung aller durchgeführten Arbeiten treten oftmals pauschale oder nachträgliche Kontrollen
  - **Software-Entwicklung**
    - In der Software-Entwicklung ist teilweise ein **umgekehrter Trend** auszumachen
    - Vorgangsmodele wie das V-Modell sowie der ISO-9000-Ansatz fordern eine starre Einzelfallprüfung unabhängig von der Relevanz eines Teilprodukts.

## 4.5 *Business Engineering*

- ◆ **Abstimmungsarbeiten reduzieren sich auf ein Minimum**
  - Abgleichen von Unstimmigkeiten wird minimiert
  - Dies ist wegen der verringerten Anzahl externer Kontaktpunkte im Unternehmensprozeß möglich
  - **Software-Entwicklung**
    - Bezogen auf den Kunden eher ein **gegenläufiger Trend**
    - Der Kunde soll immer stärker in die Entwicklungsarbeit einbezogen werden (z.B. durch Prototypen), so daß die Kontaktpunkte zum Kunden zunehmen
    - Auf der anderen Seite werden durch durchgängige Entwicklungsmethoden und prozeßumfassende CASE-Werkzeuge Unstimmigkeiten, z.B. in Dokumenten, von vornherein vermieden
    - Dies gilt auch in Bezug auf Lieferanten, wenn sie dieselben Methoden und Werkzeuge verwenden.

## 4.5 *Business Engineering*

- ◆ **Ein *Casemanager* als einzige Anlaufstelle**
  - Der »*Casemanager*« dient als Puffer zwischen einem komplexen Geschäftsprozeß und dem Kunden
  - Er verhält sich gegenüber dem Kunden so, als wäre er für die Durchführung des gesamten Prozesses verantwortlich, selbst wenn dies in Wirklichkeit nicht der Fall ist
  - **Software-Entwicklung**
    - Da Software-Entwicklungen oft in Form von Projekten abgewickelt werden, ist der Projektleiter der primäre Ansprechpartner des Kunden
    - Durch den Trend, den Kunden möglichst stark in die Entwicklung zu involvieren, ist die Forderung einer einzigen Anlaufstelle aber nicht durchzuhalten.

## 4.5 *Business Engineering*

- ◆ **Eine Mischung aus Zentralisierung und Dezentralisierung**
  - Die Informations- und Kommunikationstechnik erlaubt es Unternehmen so zu arbeiten, als seien ihre individuellen Geschäftseinheiten völlig autonom
  - Auf der anderen Seite können die Unternehmen weiterhin die Vorteile nutzen, die eine Zentralisierung hat, z.B. Zugriff auf eine Unternehmensbank
  - **Software-Entwicklung**
    - Umfangreiche Software-Entwicklungen nutzen oft bereits die Möglichkeiten der Dezentralisierung, gekoppelt mit den Vorteilen der Zentralisierung.

## Zusammenhänge

### ◆ Prozesse und ihre Qualität

#### Niedrige Prozeßqualität

Improvisierter, ad hoc- Prozeß

Reaktion bei Problemen

Kosten- und Terminpläne im  
allgemeinen nicht eingehalten

Qualitäts- und Funktionsreduktion  
bei Terminproblemen

QS-Aktivitäten werden bei Termin-  
problemen nicht durchgeführt

#### Hohe Prozeßqualität

Professionell durchgeführter  
Prozeß

Vermeiden von Problemen

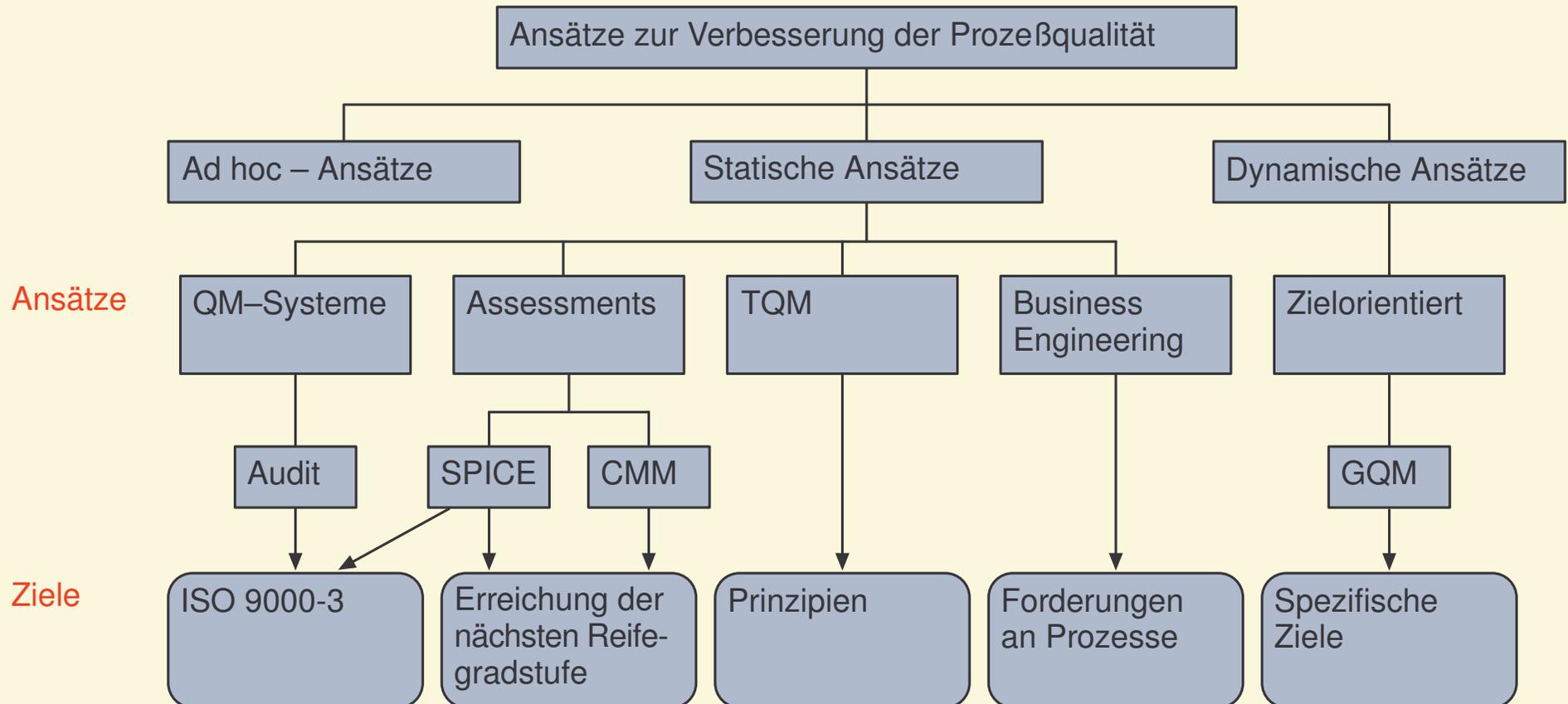
Bessere Planung durch  
geignete Prozeßverfahren

Probleme werden frühzeitig  
erkannt und behoben

Der Prozeß wird kontinuierlich  
verbessert.

# Zusammenhänge

## ◆ Ansätze zur Verbesserung der Prozeßqualität



- ◆ **Danke!**
- ◆ **Aufgaben**
- ◆ **Diese Präsentation bzw. Teile dieser Präsentation enthalten Inhalte und Grafiken des **Lehrbuchs der Software-Technik** (Band 2) von Helmut Balzert, Spektrum Akademischer Verlag, Heidelberg 1998**

