

„Kurzfristig mehr - langfristig  
weniger“

Die langfristige Rentabilität hoher  
Anfangsinvestitionen in die IT-  
Einführung oder Umstellung



Prof. Dr. Alfred Holl

# „Kurzfristig mehr - langfristig weniger“

## 1. Kostenexplosion durch mangelnde Softwarequalität

Die Kostensituation im Vergleich: Niedrige Anfangsinvestitionen führen zu hohen Folgekosten

## 2. Kostenreduktion durch erhöhte Softwarequalität

Was ist Softwarequalität ?

## 3. Erhöhte Softwarequalität durch zeitliche Betonung der Problemanalyse

a) Die Problemanalyse steht am Anfang der Entwicklung

b) Die Analyse muß 55% der Zeit in Anspruch nehmen, die Implementierung dagegen nur 15%

## 4. Sinn und Zweck der Problemanalyse

a) Steht das Nachdenken am Beginn, bleiben böse Überraschungen aus

b) Um die Schwierigkeiten der Tätigkeiten während eine Systemanalyse zu meistern, benötigen Sie den Informatiker

c) Am Ende steht weder ein Programm, noch Papierwust, sondern ein Konzept

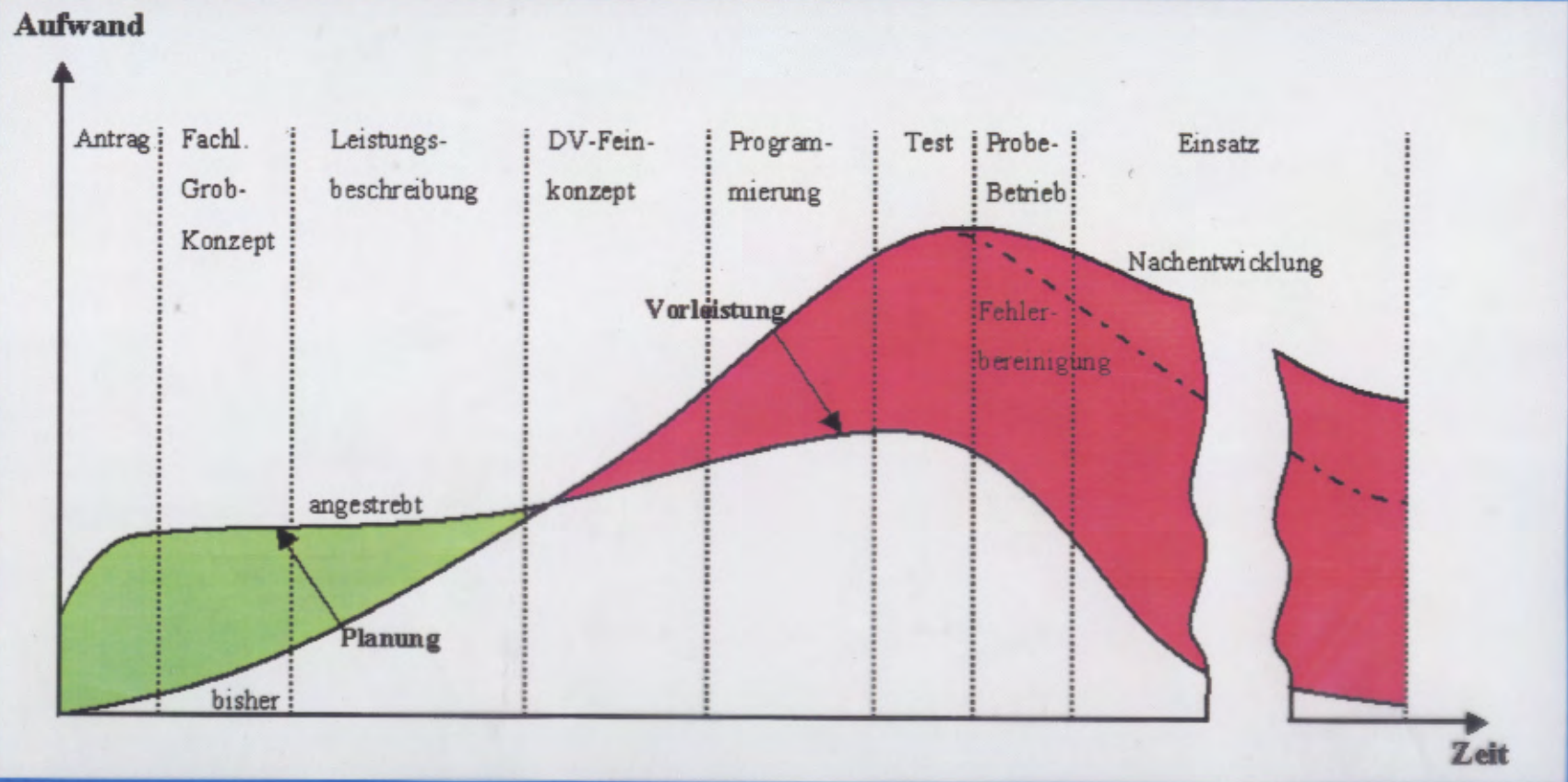
## 5. Vorgehensweisen zur Problemanalyse

## 6. Sie benötigen die Problemanalyse

a) Sie verschafft Ihnen einen klaren Überblick

b) Sie bewahrt Sie vor der Kostenexplosion

# Schematischer Aufwandsverlauf



Gesundheitswesen

## IT-Projekte scheitern

**Aufgrund unvollständiger Anforderungsprofile scheitern nach Analysen von First Consulting 84 Prozent der IT-Projekte im Gesundheitswesen.**

„Projektziele müssen deutlicher definiert werden“, fordert Frank Müller, Geschäftsführer der First Consulting Group. Sein Unternehmen hat die IT-Implementierung im Gesundheitswesen untersucht und ist zu erschreckenden Ergebnissen gekommen: So wurden nur 16% der Projekte erfolgreich abgeschlossen. Nahezu ein Drittel wurde völlig eingestellt, und mehr als die Hälfte kostete doppelt soviel wie veranschlagt. Neben einem unvollständigen Anforderungsprofil sind laut Müller die fehlende Unterstützung von seiten des Auftraggebers und mangelnde technische Kompetenz der Anbieter dafür verantwortlich.

# Kostenbeispiel

## Kosten richtig

Problemanalyse 35.000,-

Implementierung 35.000,-

Wartung 15.000,-

Fehlerbehebung 15.000,-

100.000,- DM

## Kosten falsch

Problemanalyse 10.000,-

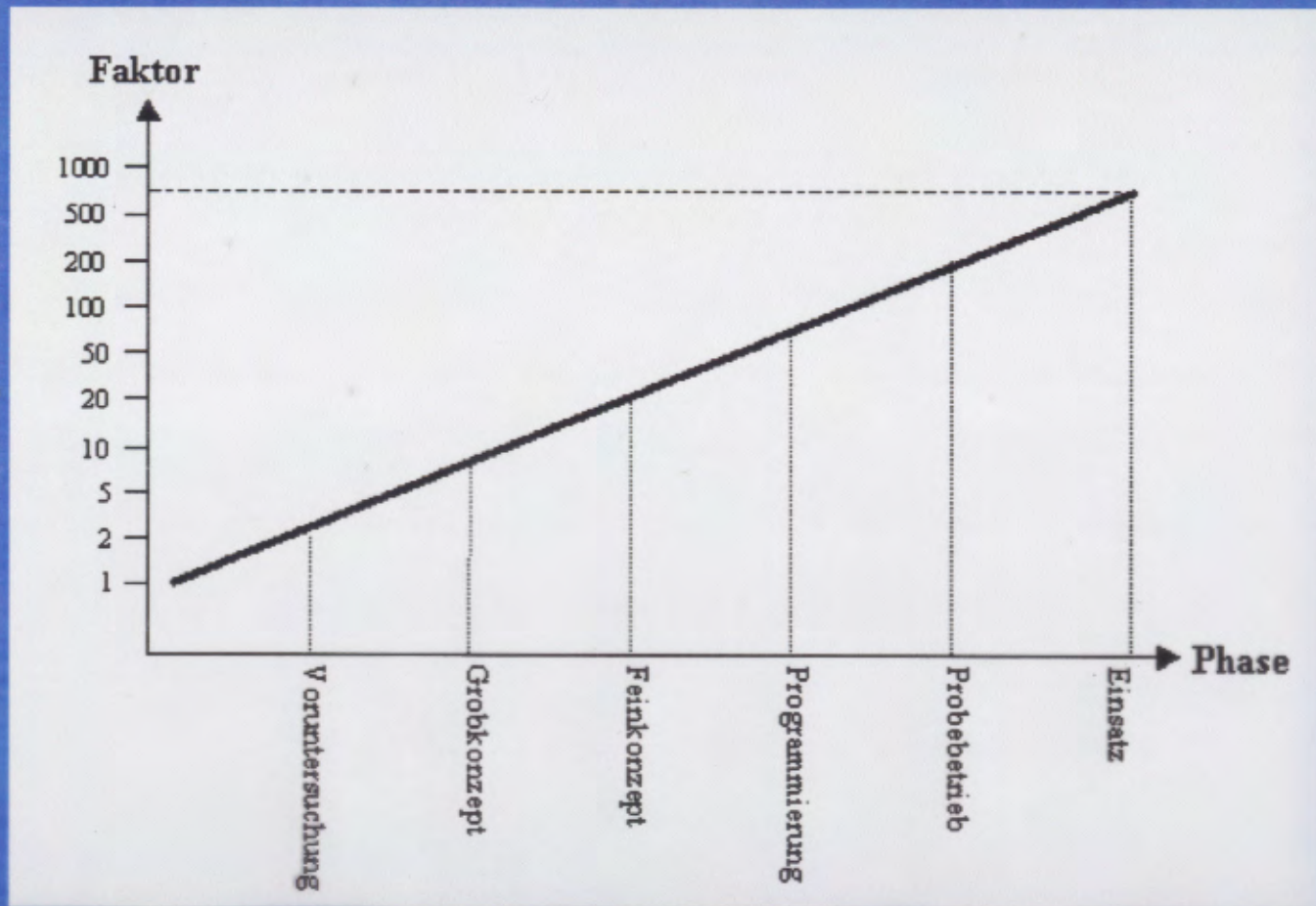
Implementierung 70.000,-

Wartung 60.000,-

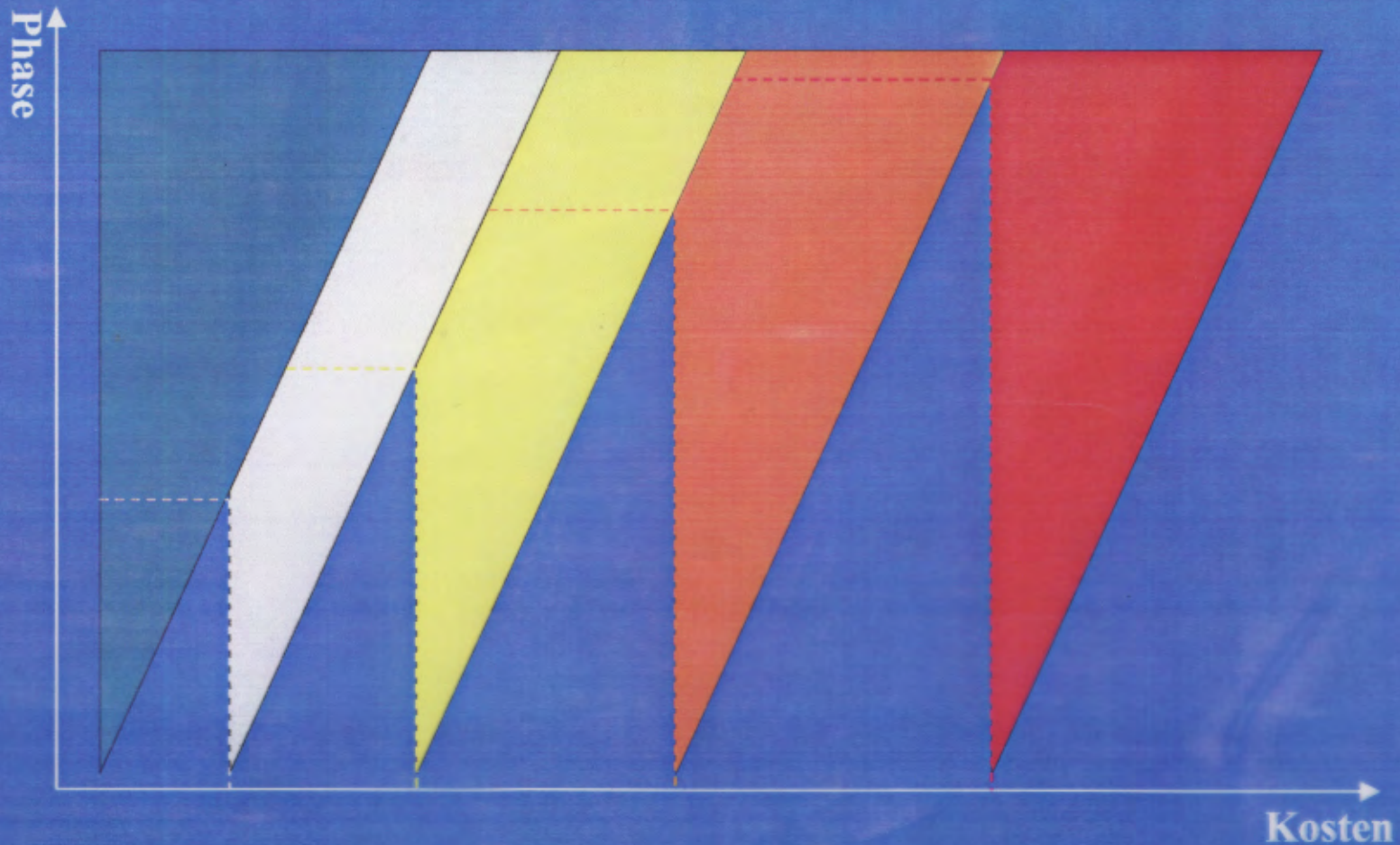
Fehlerbehebung 60.000,-

200.000,- DM

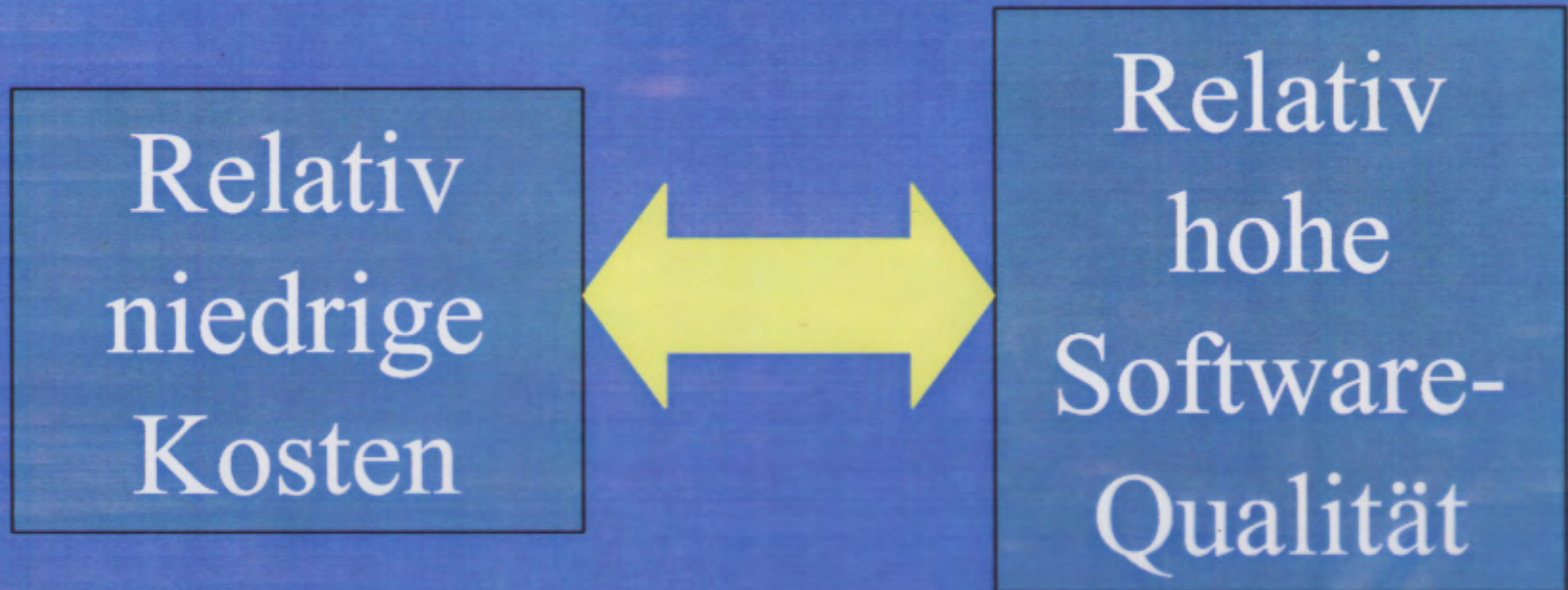
# Kostenentwicklung bei der Fehlerbehebung



# Darum werden spät erkannte Kosten teuer



„Kurzfristig mehr - langfristig weniger“

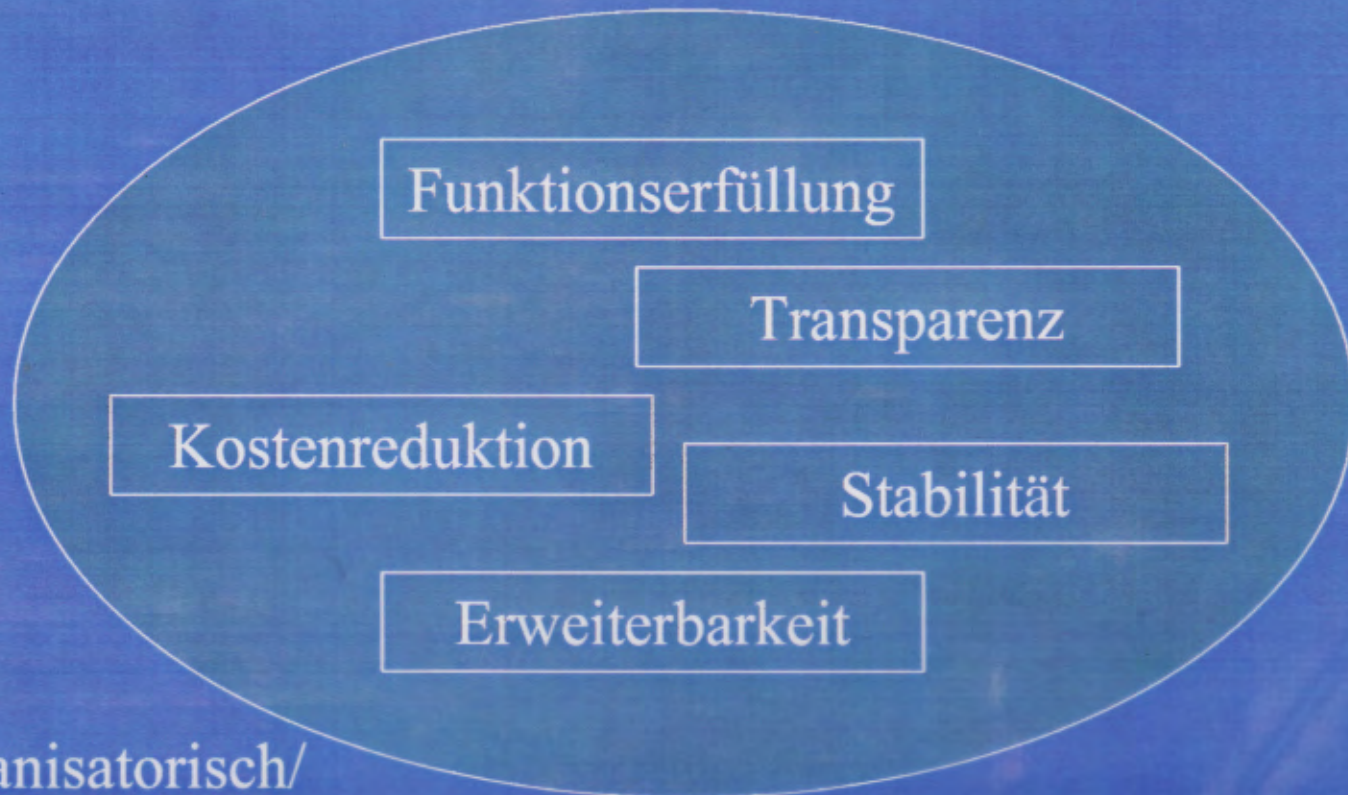




## Qualitätsmerkmale aus Benutzersicht (Sachbearbeiter)

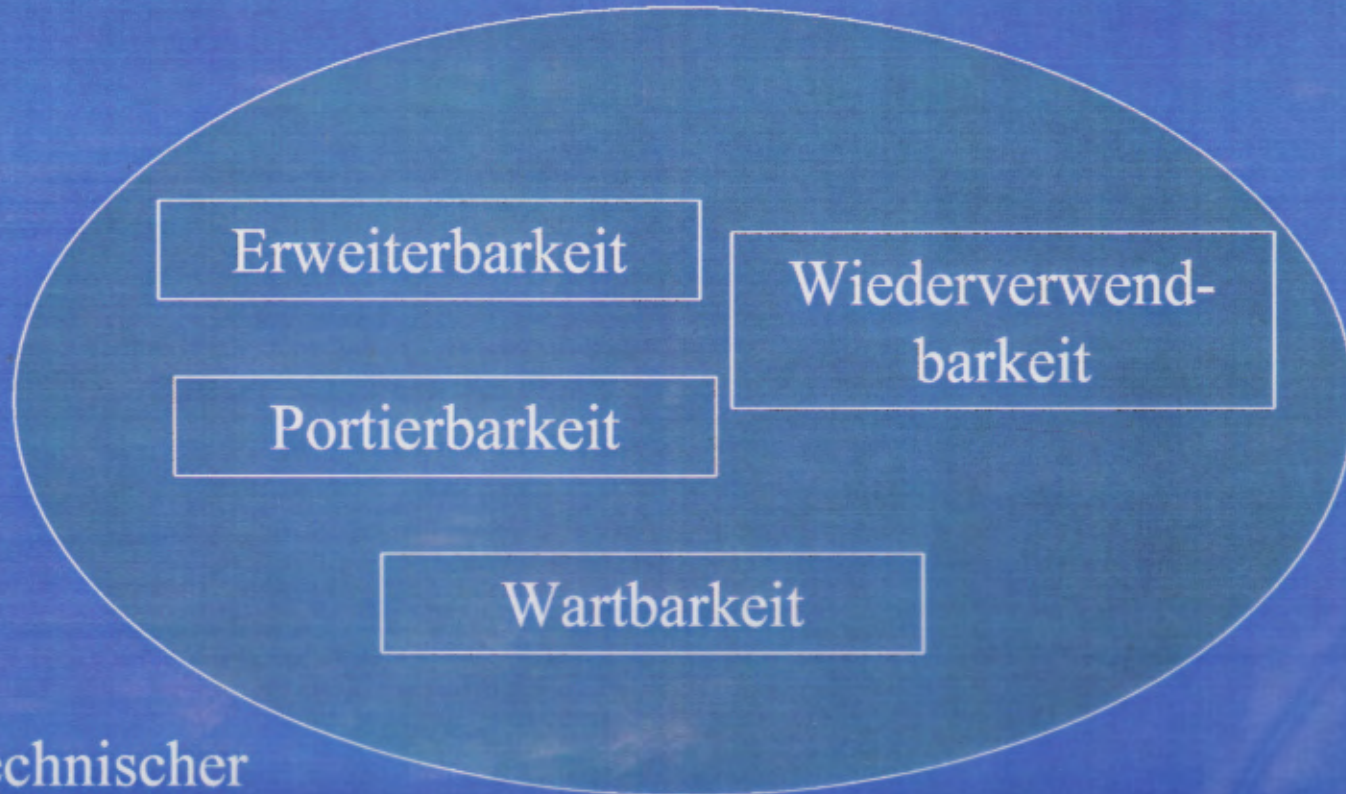


## Qualitätsmerkmale aus Anwendersicht (Institution)



organisatorisch/  
finanziell

# Qualitätsmerkmale aus Entwicklersicht



technischer  
Aspekt

# Qualitätsmerkmale



Die unterschiedlichen oberflächlichen Ansprüche der Beteiligten haben dasselbe gemeinsame Ziel: Softwarequalität

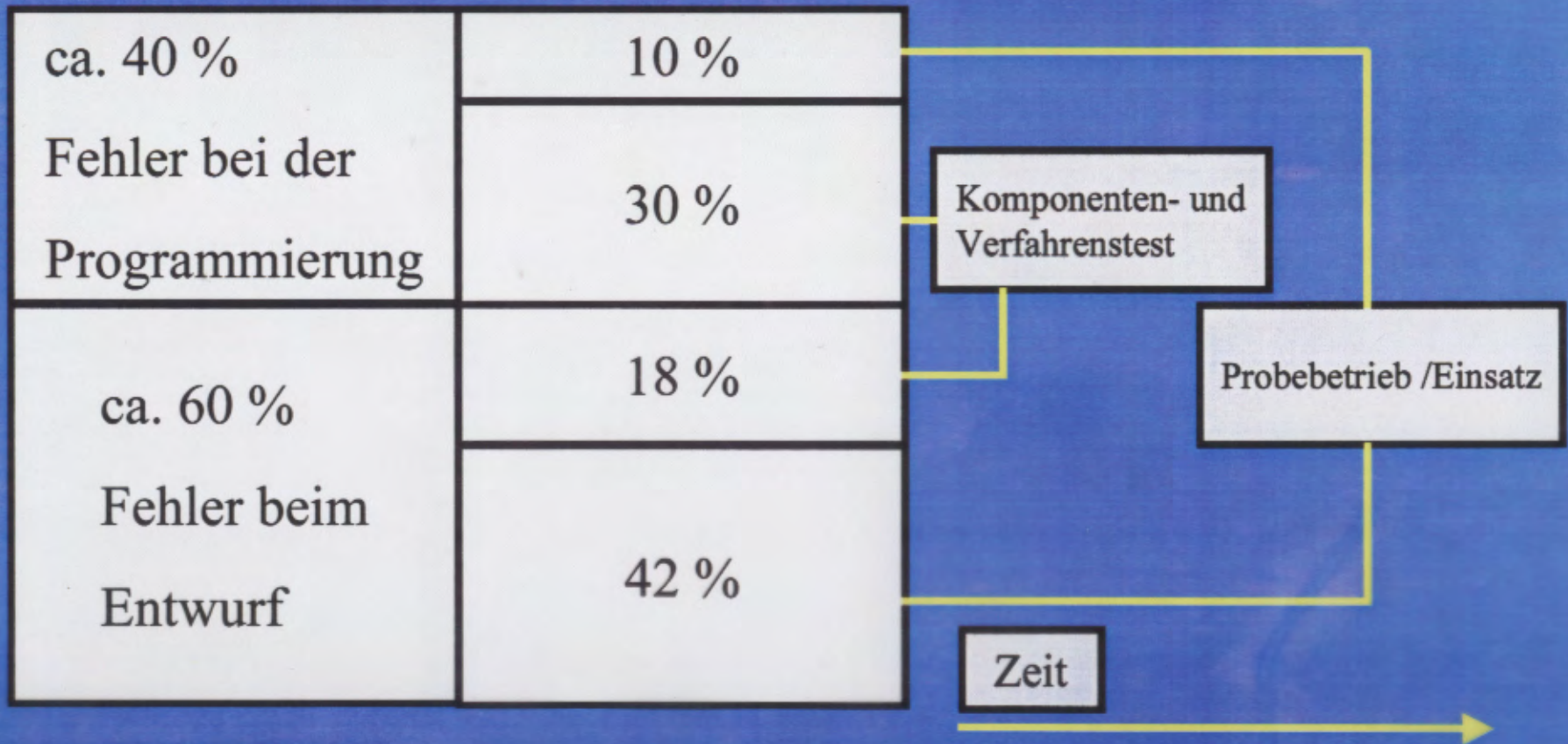
# Phasen der Software-Entwicklung

- Analytische Phase (DV-unabhängig):  
Problemanalyse, Fachkonzept
- Synthetische Phase (DV-abhängig)  
Implementierung, Test
- Wartung, Pflege

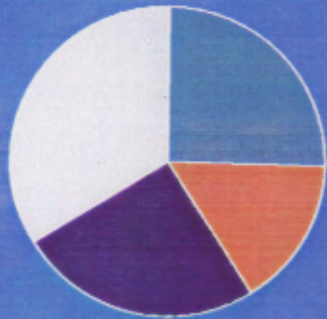
# Vorgehensmodell zur Software-Entwicklung

| Phase (Tätigkeit)                    | Ergebnis   | Abstraktionsebene           |
|--------------------------------------|--|-----------------------------|
| <u>Fachkonzept</u><br><b>55%</b>     | <u>Schlüssel-Schloß-Prinzip:</u><br>Formale Modellierung des betrieblichen Einsatzbereiches und der zu entstehenden Software | Informationsmodell          |
| <u>DV-Konzept</u><br><b>15%</b>      | Beschreibung der programm-technischen Komponenten und Zusammenhänge (Daten-, Programmstrukturen usw.)                        | Implementationsmodell       |
| <u>Implementierung</u><br><b>15%</b> | Realisierung der Beschreibungen des DV-Entwurfs in Form von Programmen, Datenbanken usw.                                     | Anwendungssystem            |
| <u>Test</u><br><b>15%</b>            |  | Getestetes Anwendungssystem |

# Bedeutung des Entwurfs

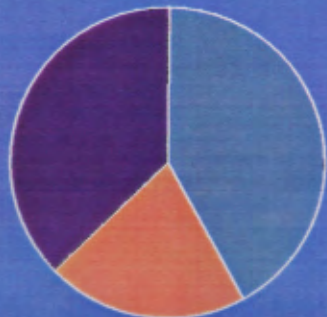


# Ideen für Phasenaufteilung (1)



- Planung
- Codierung
- Komponententest
- Systemtest

Lt. Brooks

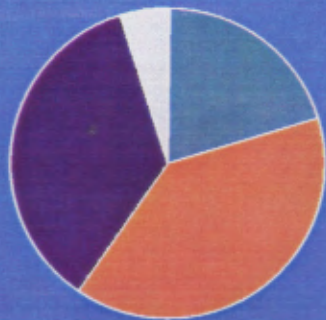


- Analyse & Entwurf
- Codierung
- Test

Lt. Wolverton



# Ideen für Phasenaufteilung (2)

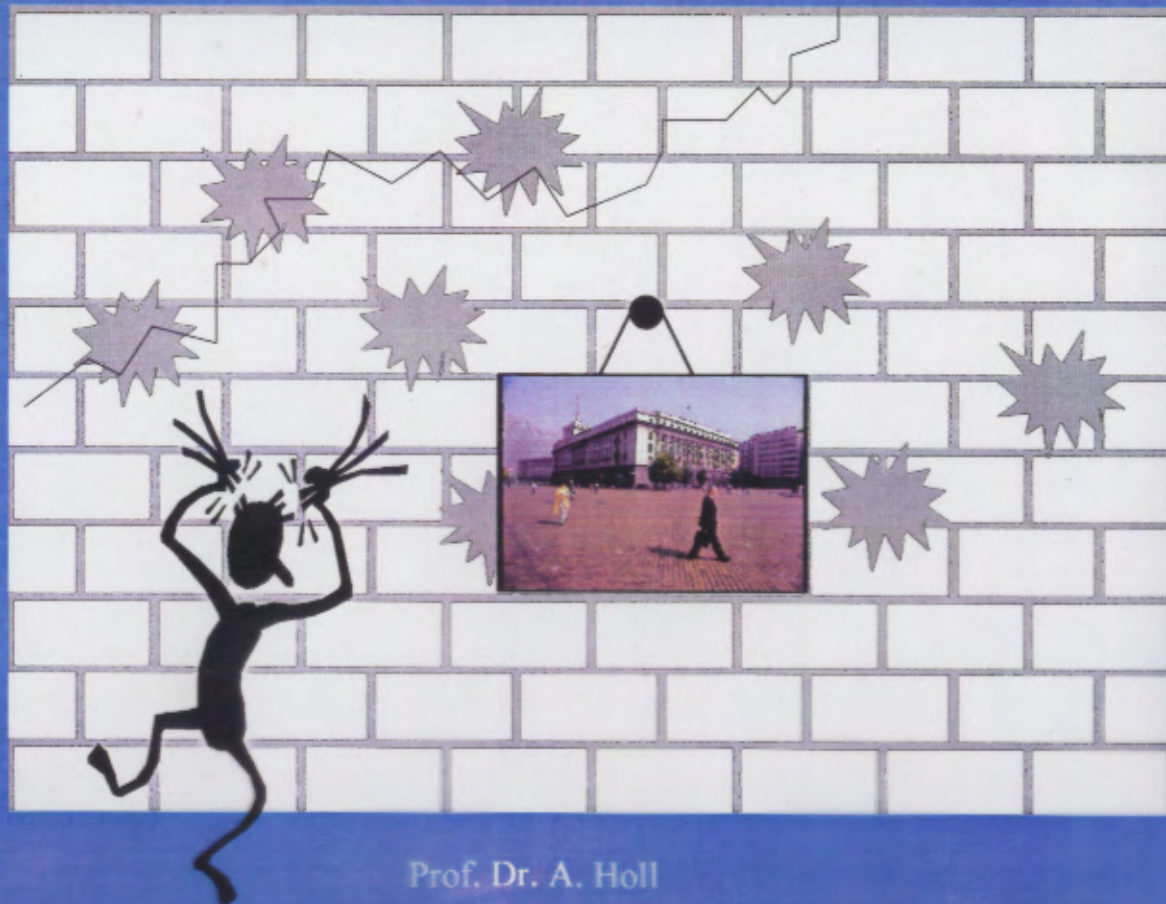


- Analyse
- Entwurf
- Implementierung
- Evaluation

Lt. Mobil

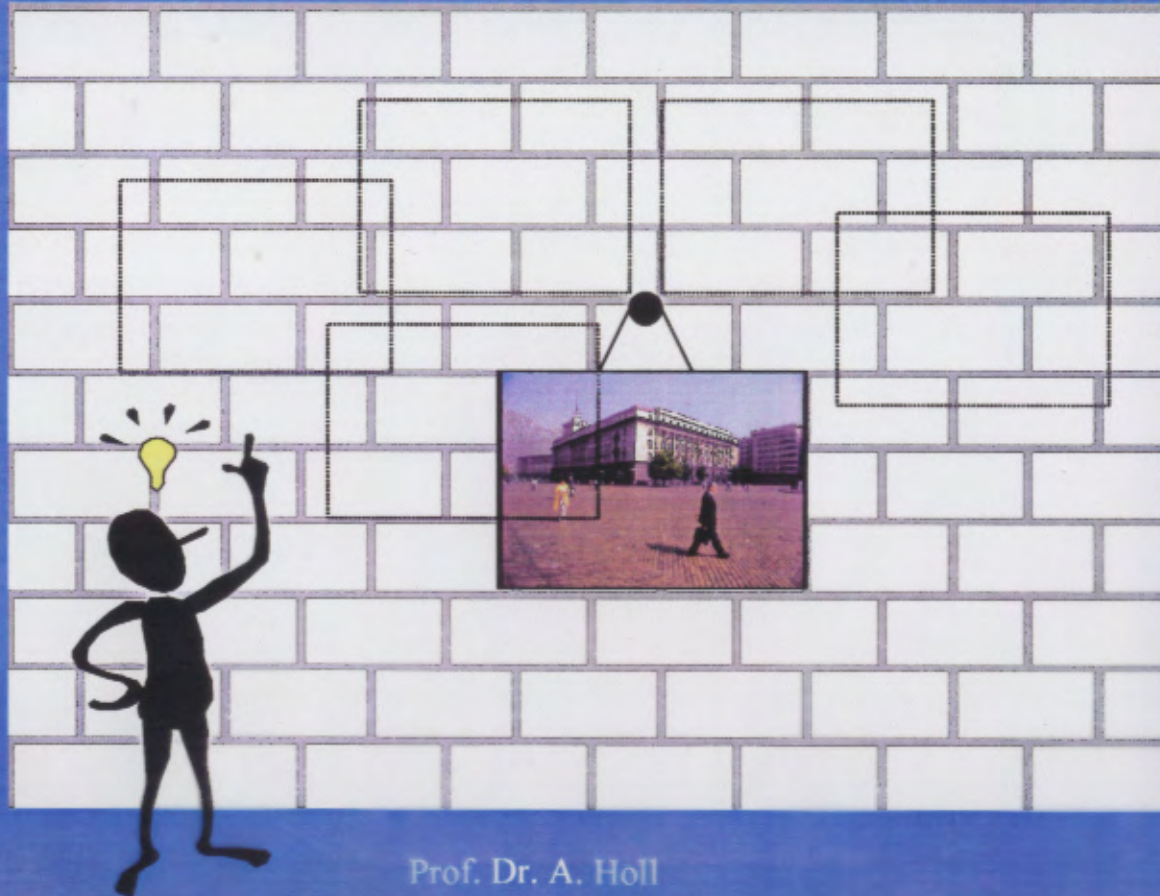
# Sinn und Zweck der Analyse-Phase (1)

Ein kleines Beispiel: Sie wollen ein Bild aufhängen. Methode 1:



# Sinn und Zweck der Analyse-Phase (2)

Ein kleines Beispiel: Sie wollen ein Bild aufhängen. Methode 2:



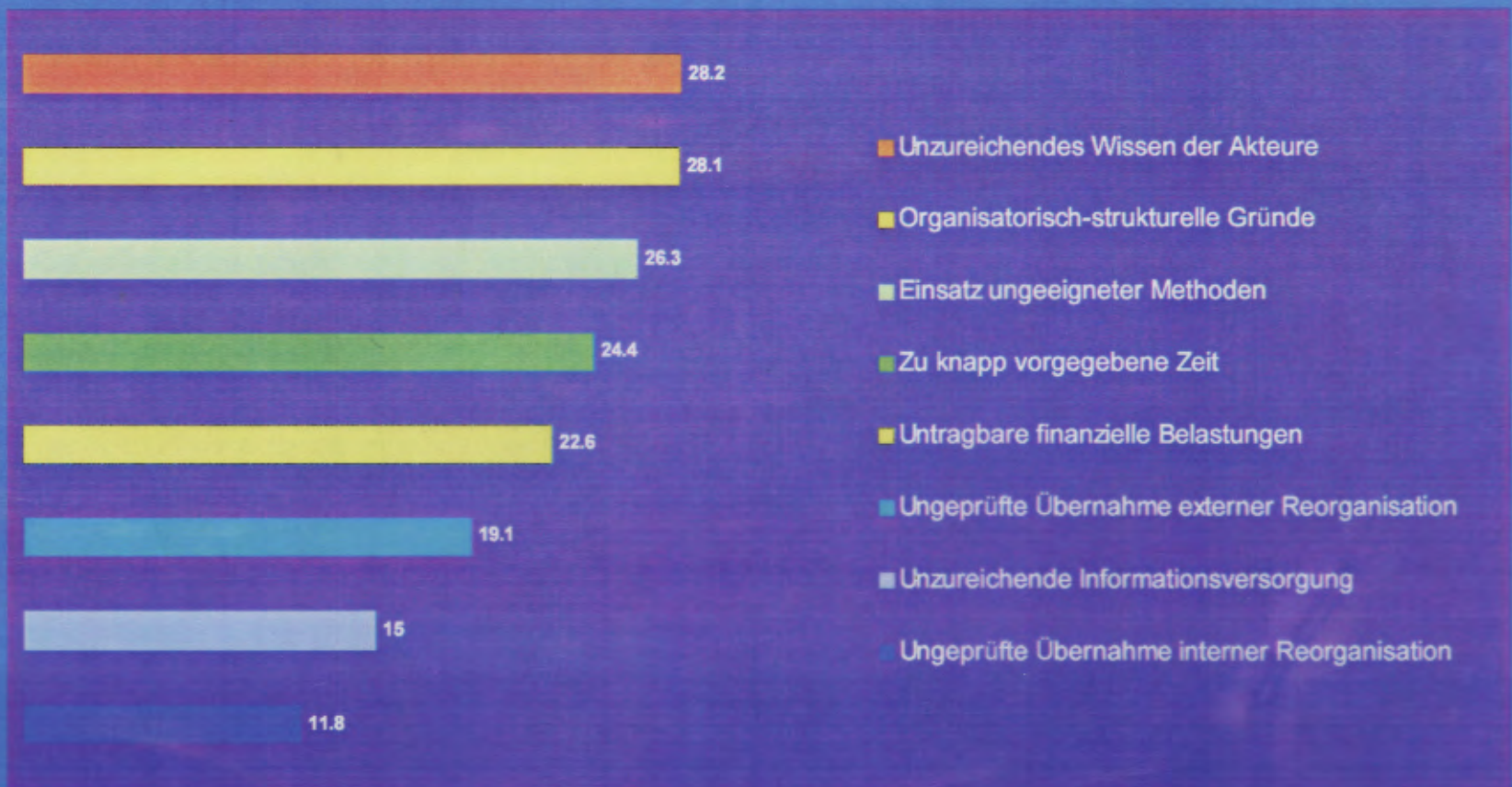
# Einstellungs- und Verhaltensakzeptanz

|                            |      | Verhaltensakzeptanz     |                               |
|----------------------------|------|-------------------------|-------------------------------|
|                            |      | ja                      | nein                          |
| Einstellungs-<br>Akzeptanz | ja   | Überzeugter<br>Benutzer | Verhinderter<br>Benutzer      |
|                            | nein | Gezwungener<br>Benutzer | Überzeugter<br>Nicht-Benutzer |

## Gründe für das Scheitern von Organisationsprojekten (1)



## Gründe für das Scheitern von Organisationsprojekten (2)



## Tätigkeiten während der Problemanalyse

- Erhebung des Ist-Zustandes
- Analyse des Ist-Zustandes
- Diskussion von Lösungsalternativen
- Sollkonzept

# Warum Informatiker für Problemanalyse ?

- **Erfahrung**
- **Distanz**
- **Fähigkeit zu**
  - Formalisierung und Mathematisierung
  - Abstraktem, formallogischem Denken
  - Requirements-Engineering
  - Analyse des IT-Bedarfs
  - Systemabgrenzung und Begriffsdefinition
  - Kommunikation mit Sachbearbeitern und Programmierung



# Ziele der Problemanalyse

## Klärung der Machbarkeit (was, wie):

- fachlich
- technisch
- organisatorisch
- finanziell
- personell
- zeitlich

# Was Sie nicht erhalten



Fertige Programme mit  
undurchschaubarem  
Code-Wust

# Was Sie nicht erhalten



Unüberblickbaren  
Dokumentationswust

# Nutzen der Ergebnisse

- Klares Konzept
- Bezug zum betrieblichen Umfeld
- Schnittstelle zur synthetischen Phase

# Techniken zur Problemanalyse

## **Techniken**

### Planungstechniken

Zielformulierung/Aufgaben-  
beschreibung, Netzplantechnik

### Erhebungstechniken

Workshop, Fragebogen, Beobach-  
tung, Dokumentenstudium,  
Schätzungen

### Analysetechniken

Geschäftsfeldanalyse, HEF-Analyse,  
Geschäftsprozeßanalyse,  
Aufgabenanalyse, Kommunikations-  
analyse, Soziometrie

# Erhebungstechnik - Beobachtung

- Offene und verdeckte Beobachtung
- Strukturierte und unstrukturierte Beobachtung
- Stichproben-Erhebungen aus punktuellen Beobachtungen
- Besichtigung des Betriebs

# Erhebungstechnik - Fragebogen

## Bedingungen:

- Erhebung quantitativer Sachverhalte
- Bekannte Erhebungsthematik
- Nicht erklärungsbedürftige Fragen
- Großer, aber homogener Kreis von Befragten

# Erhebungstechnik - Workshop

- Persönliche Ansprache
  - Steuern, Schwerpunkte setzen
  - „Informelle Informationen“, Akzeptanz
  - Feedback, Erklärungsmöglichkeit
  - Kontrollierbare Situation (Psychologie)
  - Anwendung von Beobachtungstechnik
- ➔ Nur geschulte Informatiker !**



# Analyse-Techniken

- **Geschäftsfeld-Analyse**

Definition und Beurteilung der Geschäftsfelder hinsichtlich ihres strategischen und wirtschaftlichen Nutzens

- **Haupterfolgs-Faktoren (HEF)-Analyse**

Ableitung der für den Erfolg eines Unternehmens typischen Faktoren unter Berücksichtigung der Unternehmensstrategie

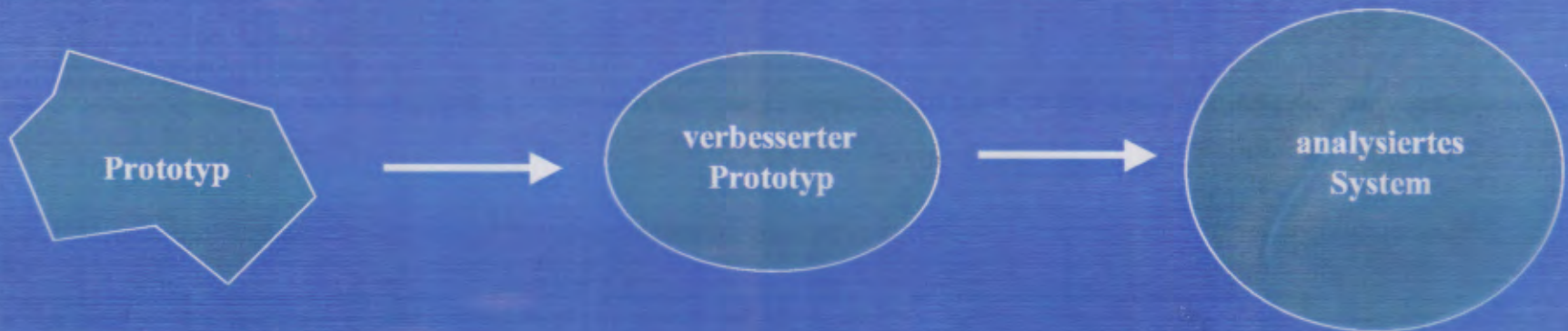
- **Geschäftsprozeß-Analyse (Business Process Reengineering)**

Branchentypische Geschäftsprozesse und unternehmensspezifische Besonderheiten werden zu einem Soll-Modell des Unternehmens

# Prototyping

## Definition:

„Ein Prototyp ist ein - mit wesentlich geringerem Aufwand als das geplante Produkt hergestelltes - vereinfachtes, einfach zu änderndes und zu erweiterndes Modell eines Systems, das nicht notwendigerweise alle Eigenschaften des Zielsystems aufweisen muß, jedoch so geartet ist, daß vor der eigentlichen Systementwicklung die wesentlichen Systemeigenschaften ersichtlich sind.



# Die Methoden des Prototyping

explorativ

→ Problemanalyse

Ausgehend von ersten Vorstellungen wird ein Prototyp erstellt. Maßgeblich ist nicht die Qualität, sondern Funktionalität, leichte Veränderbarkeit und Kürze der Entwicklungszeit. Prüfung anhand von Beispielen.

experimentell

Zerlegung des Systems in einzelne Komponenten, Spezifikation der Teilsysteme durch Beobachtung der Wechselwirkungen mit den Schnittstellen; Prüfung der Flexibilität bei Erweiterung.

evolutionär

Prototyp wird nach klaren Benutzeranforderungen aufgebaut und inkrementell weiterentwickelt, indem neue Benutzeranforderungen integriert werden.

# Sie benötigen die Problemanalyse!

## Ziel-Situation nach Problemanalyse

### Klarheit über:

- Eigene Situation
- Anfallende Kosten und Kostenersparnis
- Geschäftsabläufe
- Vorgehensweise

### und vor allem

- Beziehungen im Unternehmen → Gruppen-  
Dynamik

