

## Problemstellung

---

Trotz aller Methoden, Techniken usw. zeigen Untersuchungen sehr negative Ergebnisse:

### Slide 1

- große Software-Systeme werden im Schnitt ein Jahr zu spät ausgeliefert
- nur ein Prozent der Projekte wird rechtzeitig, ohne Budgetüberschreitung und zur vollen Kundenzufriedenheit beendet
- 25 % der Projekte werden überhaupt nicht beendet
- mehr als 60 % der IT-Produktmanager hat keine oder nur wenig Erfahrung mit modernen Software-Engineering-Techniken

Eine wichtige Aufgabe ist zuerst, Fehler und Probleme bei Projekten überhaupt zu identifizieren.

## Prozeß

---

Sehr oft werden in kritischen Situationen sowohl Prozeß- als auch Qualitätsmanagement vernachlässigt. Warnzeichen dafür sind:

### Slide 2

- Tests werden vernachlässigt, weil dafür keine Zeit mehr ist und der Kunde die Fehler sowieso melden wird
- Das System ist nur ein Testprodukt und nicht für den Endkunden, deswegen wird es nicht getestet
- Das Qualitätsmanagement kommuniziert nicht mit den Entwicklern

## Prozeß

---

- Der Prozeß ist nicht flexibel
- Niemand kann konkret sagen, warum das Projekt existiert, ob etwas Konkretes erstellt wird, und falls nicht, wer dafür verantwortlich ist

### Slide 3

- Es werden keine prozeß- oder produktbezogenen Kennzahlen erfaßt, der aktuelle Status des Projektes ist unklar
- Nicht angepaßte Pläne oder Pläne, die nicht an die Erfordernisse, sondern die Gegebenheiten angepaßt werden

## Produkt

---

Die Identifikation von Produktproblemen ist schwieriger, da Zwischenprodukte oft nicht existieren. Trotzdem gibt es auch hier typische Anzeichen:

- Die Problemlösung wurde nicht mit minimalem Aufwand erstellt
- Die aktuelle Version aller Produktbestandteile und die Konfiguration des Gesamtsystems ist unklar
- Es existiert entweder zu wenig, nicht aktuelle oder zu viel Dokumentation
- Bei der Integration von Arbeitsergebnissen treten wiederkehrende und nachhaltige Probleme auf

### Slide 4

## Personen

---

Zu den Personen, die direkt und indirekt an Projekten beteiligt sind, gehören die Kunden, die Benutzer, das Management und die Entwickler. Allgemeine Punkte sind dabei:

### Slide 5

- Unabhängig vom Prozeß kann ein gutes Entwicklerteam erfolgreich Projekte bearbeiten, ein schlechtes Team wird jedoch auch trotz eines guten Prozesses nicht erfolgreich sein.
- In kritischen Situationen werden entweder die Entwickler beschuldigt (aus der Sicht des Managements oder der Kunden), oder es wird das Projekt verlassen (aus Entwicklersicht)

## Entwickler

---

Aus Entwicklerperspektive befindet sich das Projekt in einer schwierigen Phase, wenn:

### Slide 6

- Probleme der Entwickler von Managern als „Chance“ oder „Herausforderung“ angesehen werden
- Manager schwer erreichbar sind bzw. kein Feedback liefern
- Organisatorische Tätigkeiten dominieren und kreative Tätigkeiten verdrängt werden
- Probleme mit dem Einsatz von mehr Entwicklern bekämpft werden
- Aktivität mit Produktivität verwechselt wird

## Management

---

Aus der Sicht des Managements gibt es folgende Warnsignale:

### Slide 7

- Die Mitglieder des Entwicklerteams kennen die kritischen Erfolgsfaktoren nicht
- Die Entwickler wissen nicht, inwieweit sie zum Erfolg des Projekts beitragen
- Die Entwickler sind unzufrieden, ohne dies jedoch klar zu ausgesprechen oder Gelegenheit für die Äußerung von Kritik haben
- Niemand fühlt sich für Fortschrittsberichte verantwortlich
- Riskante Entscheidungen werden aus Angst vor Versagen vermieden

## Kunden

---

Die Gruppe der Kunden schließt alle Personen ein, die direkten oder indirekten Nutzen aus dem in Auftrag gegebenen System haben. Für sie sind folgende Punkte kritisch:

### Slide 8

- Die Anforderungen sind nicht von den eigentlichen Kunden/Nutzern definiert worden
- Der Kunde ist nicht in allen Stufen des Projekts beteiligt
- Die Erwartungen des Kunden werden nicht erfüllt
- Nicht-Funktionale Anforderungen spielen eine zu geringe Rolle

## Kunden

---

Diese Warnsignale sind abhängig vom Grad der Involviertheit des Kunden:

**initial:** wenig Projekterfahrung, Vertrauen in die Entwickler, wissen, was sie vermutlich wollen

Slide 9

**besorgt:** hatten eventuell vorherige schlechte Erfahrung, wissen aber nicht, wie sie die Situation verbessern können, schätzen externe Entwickler eher als interne

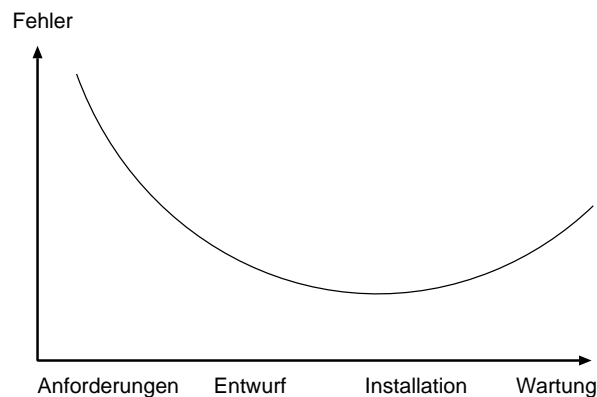
**über-involviert:** glauben, zu wissen, wie ein Projekt geführt werden muß (Projektmanagement aus der Distanz), wissen was sie wollen und wie dies erreicht werden kann

**ausgewogen:** verstehen und akzeptieren ihre Rolle, setzen auf Zusammenarbeit

## Zeitliche Verteilung

---

Die Verteilung der verursachten Softwarefehler im Lebenszyklus stellt sich folgendermaßen dar:



Slide 10

Damit sind auch die Phasen identifiziert, auf die geachtet werden sollte.

## Kosten für Fehlerkorrektur

---

In Verbindung mit den relativen Kosten für die Fehlerkorrektur in den einzelnen Phasen (hier bzgl. der Implementierungsphase) ergeben sich klare Schwerpunkte für das Projektmanagement:

Slide 11

