

Techniken der Projektentwicklung

Gerhard Sagerer M5-111, Tel 2935, Sprechstd Fr 12-13

- Vorlesung
- Tutorien
- Seminar
- Gruppenprojekt

Grundlagen und Projektorganisation

Softwaremodellierung

Projektanalyse und -präsentation

Projekt durchführen

Organisatorisches

auch noch Vorlesung zu Softwarepraktikum

verschiedene Informatik-Studiengänge

neben Vorlesung Seminar mit Übungen in kleineren Gruppen (ca 40 Teilnehmer)

Voraussetzung A&D-Modul erfolgreich abgeschlossen
(Stichtag 15.11.2005)

Kolloquium über Vorlesung

aktive Teilnahme in Seminar und Übungen

Projekterfolg der Gruppen

Betreuer und Tutoren

Franz Kummert

Ingo Lütkebohle

Dirk Stößel

Tutoren

- Mark Althoff

- Patrick Holthaus

- Daniel Katzberg

- Matthias Rolf

- Patrick Schwienteck

- Christoph Wortmeier

Software Praktikum

Vorlesung

Einzelprojekt

Gruppenprojekt

nur Wintersemester

Tutoren

- Christoph Broschinski
- Jan Paller
- Felix Reinhart

Softwaresysteme, Softwareprojekte

- SAP R/3
- HIS
- Projekte im SoPra
- Verbmobil
- SFB 360 Situierete Künstliche Kommunikatoren
- VAMPIRE
- Bilddatenbanken
- Hunde, Humanoide, etc
- Waschmaschinen
- Kraftfahrzeuge
- Fertigungsstraßen
- COGNIRON

Literatur

- Booch, Rambaugh, Jacobson: The Unified Programming Language User Guide. Addison-Wesley, 1999
- Bernd Oesterreich: Objektorientierte Softwareentwicklung. Oldenburg Verlag, 1998
- Ian Sommerville: Software Engineering. Addison-Wesley, 2001
- Herwig Mayr: Projekt Engineering. Fachbuchverlag Leipzig, 2001 www.projekt-engineering.at
- Chris Rupp, SOPHIST GROUP: Requirements-Engineering und –Management. Hanser 2001, www.sophist.de
- Kent Beck: eXtreme Programming eXplained. Addison-Wesley, 1999

Seminarinhalte

- Übungen
- Vorträge
- Poster

Werkzeuge: Eclipse, LaTeX, Debugging, CVS, Java Editing

- Anforderungsanalyse
- von der Analyse zum Entwurf
- statische Modellierung mit Klassendiagrammen
- dynamische Modellierung
- CRC Karten
- Hin zu Implementierung mit Java

Projektspezifische Inhalte

Aufbau der Vorlesung

Projekt- und System-Engineering

1. Begriffe
2. Projektentwicklungszyklus
3. Produktentwicklungszyklus
4. Grundlagen Projekt-Engineering
5. Zielerhebung- und Zielfindung
6. Organisationsformen
7. Risiken
8. Qualitätssicherung

Begriffe: System

System: Bezeichnung für jede Gesamtheit von Objekten, die sich in einem ganzheitlichen Zusammenhang befinden, wobei ihre Wechselwirkungen untereinander diejenigen mit der Umwelt im Allgemeinen stark überwiegen, sodass sie als ein von der Umwelt mehr oder weniger unabhängiges Ganzes (mit einer durch ihre Eigenschaften und Wechselwirkungen festgelegten Struktur) behandelt werden können.
[Meyers Lexikon]

Begriffe: System

Teilsystem: horizontale Unterteilung, Darstellung und Übersicht unter bestimmten Gesichtspunkten

Untersysteme: vertikale Unterteilung, geschlossene Subkomponenten

Modell: Abbildung eines Systems im Rechner

Sicht: Darstellung eines Teilsystems auf das Modell

Schnittstellen: dienen der Kopplung von Untersystemen zum Gesamtsystem

Begriffe: Organisation

Organisation: soziales Gebilde, das dauerhaft ein Ziel verfolgt und eine formale Struktur aufweist, mit deren Hilfe die Aktivitäten der Mitglieder auf das verfolgte Ziel ausgerichtet werden sollen

**Ziel – dauerhaft – Mitglieder –
formale Struktur – Aktivitäten der Mitglieder**

Instrumentale Organisation: Zielerreichung im Fokus

Institutionaler Organisationsbegriff: Gesamtheit des organisierten Gebildes

Begriffe: Organisation

Standardisierung: antizipierendes Durchdenken von Problemlösungswegen und die darauf aufbauende Festlegung von Aktivitätsfolgen, sodass diese im Wiederholungsfall routiniert und gleichartig ablaufen.

Programmierung: Strukturierung von Aktivitätsfolgen

Varietät: Anzahl von Verhaltensweisen bei der Abwicklung eines Prozesses

routiniert versus problemlösend

Aufbauorganisation: Strukturierung von Systemen

Ablauforganisation: befasst sich mit Systembeziehungen

Begriffe: Projekt

Projekt: innovatives, komplexes Vorhaben mit Randbedingungen (*Termine, Kosten, Ressourcen*)

Gemeinsamkeiten von Projekten:

- innovative Aufgabenkomplexe
- zeitlich abgegrenzte und komplexe Vorhaben
- keine reine Wiederholung früherer Arbeiten
- begrenzte verfügbare Mittel
- im Voraus geplante oder festgelegte Kosten bzw. Budget
- mehrere beteiligte Stellen, Zusammenarbeit erforderlich

Begriffe: Projekt

Kriterien zur Beurteilung und Klassifikation von Projekten sind deren:

- Umfang
- Dauer
- Besonderheit
- Komplexität
- Schwierigkeit
- Bedeutung
- Risiko
- Kosten
- Kontinuität
- Intensität
- Anzahl
- Organisations- und Führungsverständnis

Begriffe: internes versus externes Projekt

Einteilung nach der **Stellung des Auftraggebers**

Intern: Auftraggeber ist Unternehmensleitung; Ziel ist Erneuerung bzw. Verbesserung des Leistungspotentials des Unternehmens

Extern: externer Kunde; Marktleistung

Begriffe: Arten von Projektkooperationen

Einteilung nach Kooperation zwischen **Auftraggeber**
und Auftragnehmer

Consulting, Coaching: Projektbegleitung des Kunden;
Auftragnehmer ist Berater ohne Führungsaufgabe

Turn-Key-Entwicklung: erstellen eines Produkts durch
Auftragnehmer; Liefertermin;

Joint Venture: gemeinsame Entwicklung; kein
Liefertermin aber Projektende; geteilte
Verantwortung

Begriffe: Sichten auf Projekte

betriebswirtschaftlich: Projekt im Gleichgewicht zwischen Kosten, Zeit und Qualität des Produkts

technisch: Erfolg durch technische Eigenschaften bestimmt

soziologisch: Gleichgewicht zwischen Teamgeist und Selbstverwirklichung

Begriffe: Software Engineering

[Fairley 1985] *SE* ist die technische und organisatorische Disziplin zur systematischen Herstellung und Wartung von Softwareprodukten, die zeitgerecht und innerhalb vorgegebener Kostenschranken hergestellt und modifiziert werden.

[Sommerville 1985] *SE* befasst sich mit dem Bau von Softwaresystemen, die nicht von einem Entwickler alleine hergestellt werden können. *SE* beruht auf der Anwendung ingenieurmäßiger Prinzipien und umfaßt sowohl technische als auch nichttechnische Aspekte.

[Bauer 1975] Das Ziel der *Softwaretechnik* ist die wirtschaftliche Herstellung zuverlässiger und effizienter Software.

Hauptprobleme des Software Engineering

- Spezifikation der Anforderungen
- Beherrschung der Komplexität
- Zerlegung eines Systems in Untersysteme
- Spezifikation der Schnittstellen
- Wiederverwendung/Wiederverwendbarkeit von Bausteinen
- Projektorganisation
- Effizienz
- Dokumentation und Wartbarkeit
- Änderbarkeit und Erweiterbarkeit
- Übertragbarkeit auf und Anpassbarkeit an verschiedene Hardwaresysteme

2. Projektentwicklungszyklus: Schritte

1. Initialisierung
 2. Angebotslegung und Auftragserteilung
 3. Vorprojekt
 4. Produkterstellung
 5. Nutzung
 6. Lebensdauer von Software
- Nicht rein sequentiell,
 - zurückgehen auf frühere Schritte,
 - gesamte Projektentwicklung neu beginnen

Projektinitialisierung

- Zieldefinition
- Abgrenzung des Problems
- Projektrestriktionen (Know-How, Ressourcen)

Alternativ:

6. Auftraggeber hat Lasten- oder Pflichtenheft
7. Ziele werden gemeinsam erarbeitet (Besprechung, Workshop, Präsentation)

Weitere Projektvorbereitende Maßnahmen:

- Projektgruppe
- Literatur, Experten, weitere Informationsquellen
- Feststellen von Auslastung und Reservierung von Zeit

Vorprojekt

- Feststellung der Machbarkeit
- Darstellung des Istzustands
- (Grob-)Darstellung des Sollzustands
- Grobe Anforderungsanalyse
- Aufwandsabschätzung (Arbeit, Zeit, Ressourcen)
- Vorgehenshandbuch
- Anforderungen an Entwicklungsumgebung
- Vom Auftraggeber zu stellende Ressourcen
- Projektgliederung mit Realiserungsplan
- Definition des Erfüllungsgrades (Bedeutung von *brauchbar*, *Mangel*, *Qualität*, etc.)
- Schulungsplan

Produkterstellung

Basisprodukt: notwendigste Anforderungen aus Sicht des Auftraggebers

Meilensteintechnik: Abschnitte mit entscheidungsreifen Zwischenprodukten

Spin-offs: Subprojekte an Meilensteinen initiieren

Adaptierung und Optimierung: Änderungs- und Erweiterungswünsche, Zusatz- und Folgeprojekte

Produktnutzung

Fragen nach dem Erfolg eines Softwareprodukts

Kriterien:

- Einbettung in Betriebsführungskonzept
- Verfügbares Personal
- Komplexität
- Motivation der AnwenderInnen
- Schulung
- Gestaltung

Lebensdauer

Produktlebensdauer von Hardware: 2-3 Jahre aber Ersatzproblem

Produktlebensdauer von Software: 10-20 Jahre, Pflege bis 50(!!!) Jahre

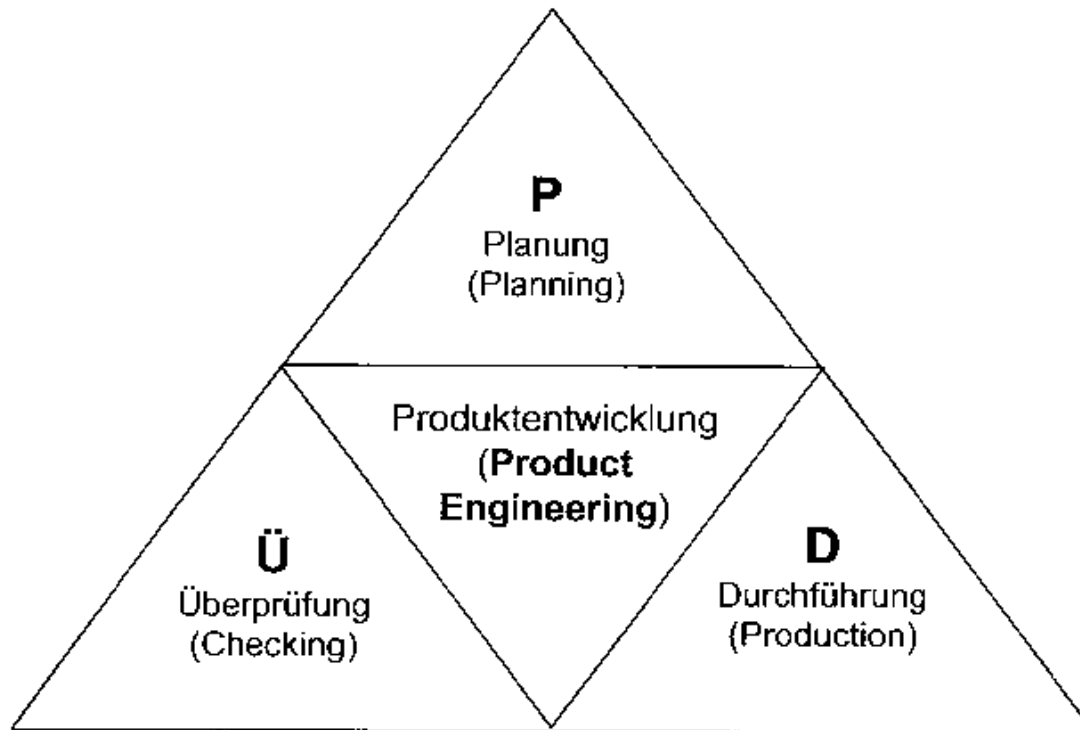
Zeitspannen:

- Labor bis Produkt ca 15 Jahre
- bei Software eher mehr
- Paradigmenwechsel in Softwareentwicklung alle 20 Jahre (1960, 1980, 2000)

3. Produktentwicklungszyklus: Komponenten

1. Planung
 2. Durchführung
 3. Überprüfung
- Planung ersetzt Zufall durch Irrtum
 - Planung ohne Überprüfung ist sinnlos
 - Überprüfung ohne Planung unmöglich

Grundkomponenten der Produktentwicklung



Quelle: H. Mayr: Projekt Engineering, Fachbuchverlag Leipzig, 2001

Planung

Ziel: Vorgaben zur Entwicklung in Bezug auf

- Aufgaben
- Aufwand
- Termine
- Ressourcen
- Kosten
- Finanzen
- Begleitende Maßnahmen (z.B. Qualitätssicherung)
- Vorsorgende Maßnahmen (z.B. Sicherungs- und Notfallplanung)

Planung

Aufgaben der Planung: Produkte so zergliedern, dass die einzelnen Arbeiten planbar und mit überprüfbaren Resultaten von möglichst kleinen Teams mit wirtschaftlich vertretbarem Aufwand an Zeit und technischen Ressourcen durchgeführt werden kann.

Die Projektplanung ist **Aufgabe der gesamten Projektorganisation unter dem Verantwortungsbereich des Projektleiters**

Durchführung

- Abarbeitung der Projektschritte (seriell, parallel)
- (End-)Auswahl der Werkzeuge
- Risikobehandlung
- (Fein-)Design
- Implementierung
- Testen
- Produkt-Einführung

Projekt in kleinerem Team beginnen,

Kommunikationsstruktur sukzessive aufbauen,

Priorisierung der Projektschritte (leicht, schwer, wichtig, etc. zuerst)

Überprüfung

Ziel der Überprüfung (checking) der Produkterstellung :

Während der Durchführung und möglichst schon in ihren Ansätzen Abweichungen von den Projektzielen zu erkennen und gegebenenfalls korrigierende Maßnahmen einleiten.

Überprüfung

Aufgaben der Überprüfung

- Durchsetzen der geplanten Lösungen
- Anleiten und motivieren der Mitarbeiter
- Herbeiführen notwendiger Entscheidungen
- Einleiten von Korrekturmaßnahmen
- Einleiten der Ziele
- Koordination zwischen Auftraggeber, Auftragnehmer und Projektgruppe
- Durchführung der Validierung
- Durchführung der Abnahme
- (externer) Projektabschluss
- (interne) Evaluierung und Nachkalkulation

Überprüfung

Hilfsmittel der Überprüfung

- Anforderungsbeschreibungen
- Projektpläne
- Besprechungsprotokolle
- Projektfortschrittsbericht

4. Grundlagen des **Projekt Engineering**

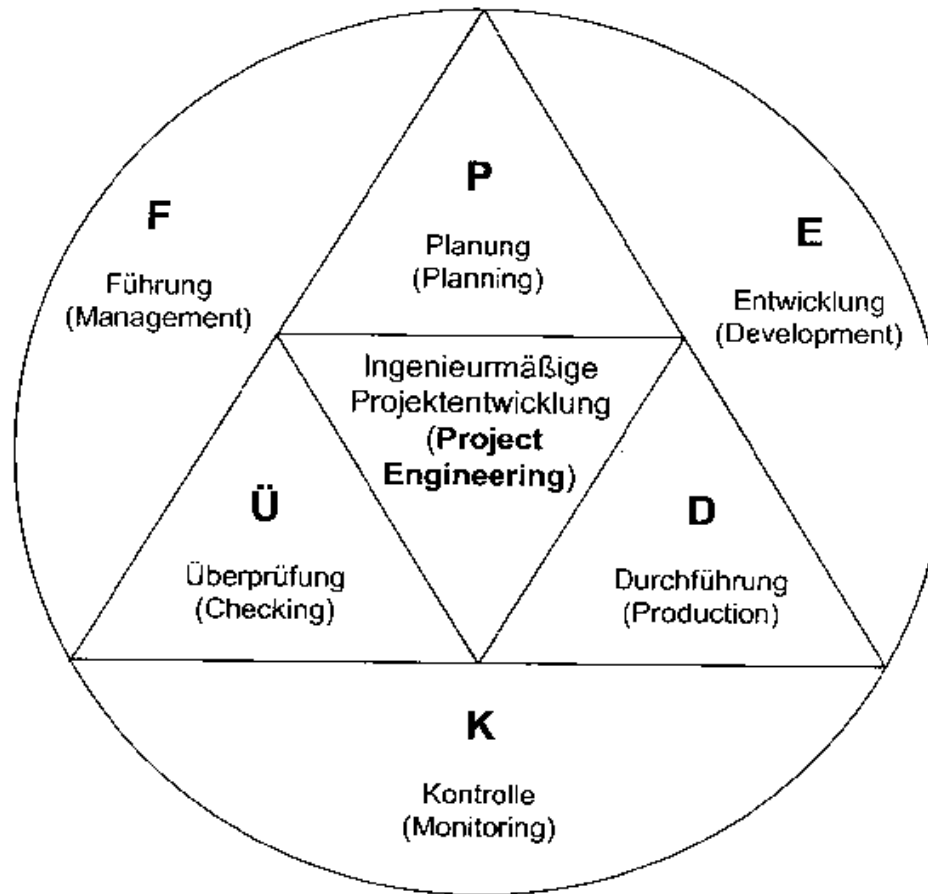
Führung (management): wirtschaftliche Belange der Planung und Überprüfung, nicht der Durchführung

Entwicklung (development): wirtschaftliche Belange der Planung und Durchführung, nicht der Überprüfung

Kontrolle (monitoring): wirtschaftliche Belange der Durchführung und Überprüfung, nicht der Planung

Auf **Kompetenztrennung** achten,
damit erfolgt auch weitgehende Aufgabentrennung

Sicht der Projektentwicklung auf die Produktentwicklung



Quelle: H. Mayr: Projekt Engineering, Fachbuchverlag Leipzig, 2001

Führung: Ziele

- Projektplanung
- Projektstrukturierung
- Adaptieren der Planung auf Basis der Überprüfung

Projektaufgabe in Projektziele umsetzen

Realisierbarkeit der Aufgabe und der so definierten Ziele prüfen

Festlegung der organisatorischen Regelungen

- Einrichten eines Projektträgers
- Festlegen der Form der Projektorganisation
- Festschreiben der Kompetenzen des Projektleiters

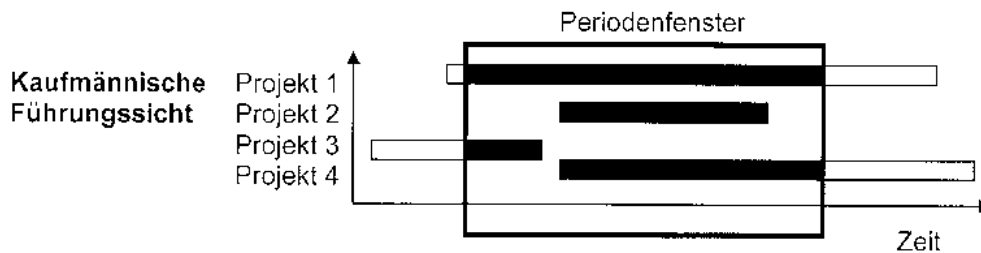
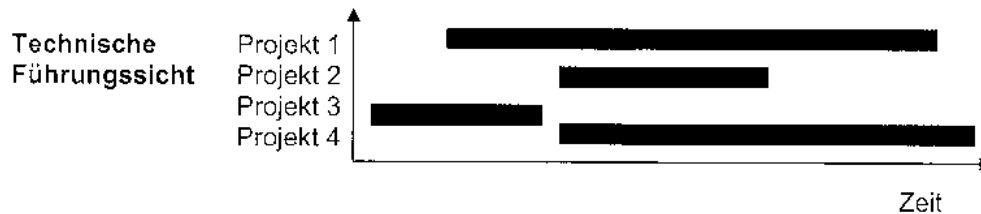
Management in diesem Sinne ist nicht Steuerung sondern
Regelung

Führung: Aufgaben

- Projektleitung und –organisation festlegen
- Aufgaben, Kompetenzen und Verantwortung verteilen
- Projekt und Projektziel strukturieren
- Machbarkeit feststellen
- Termin und Kosten planen und adaptieren
- Durchführung organisieren
- Information sicherstellen
- Dokumentationserstellung planen

Führung: Arten

Technische und kaufmännische Sichtweise der Projektführung:



Quelle: H. Mayr: Projekt Engineering, Fachbuchverlag Leipzig, 2001

Denken und Handeln in Projekten ist zentrales Konzept der Unternehmensführung. Projektmanagement ist zentrale Qualifikation

Entwicklung: Ziel

**Die Entwicklung ist die Umsetzung des Projektziels
gemäß der Planung**

Entwicklung: Aufgaben

- Verfeinerung des Entwurfs von der Analyse zum inhaltlichen Design
- Weiterentwicklung von Prototypen der Zielerforschung zu Schnittstellenprototypen und Architekturprototypen
- Aufstellen eines Ressourcenbelegungsplans (auch Mitarbeiter)
- Erstellen des Pflichtenhefts
- Durchführung der Implementierung auf Grundlage des Lastenheftes und des Designs
- Erstellen und Abnehmen der Testsuite
- Erstellen der Produktdokumentation (System- und Nutzungsdokumentation)

Entwicklung: Arten

- Einzelentwicklung
- Verteilte Entwicklung
- Serielle Entwicklung
- Parallele Entwicklung
- Konkurrierende Entwicklung

Monitoring: Ziel

ist die Erkennung von Planabweichungen während der Projektdurchführung, deren Ursache und Auswirkungen, Die Planadaption durch korrigierende Maßnahmen

Monitoring muss aktiv durchgeführt werden!

- Planungsunterlagen
- Fortschrittsberichte
- Interne Reviews
- Milestones
- Externe Reviews

Monitoring: Aufgaben und Arten

- Termine und Abläufe
- erbrachte Leistung
- Kosten und Finanzierung
- Ressourcen: Kapazität und Auslastung
- Qualität

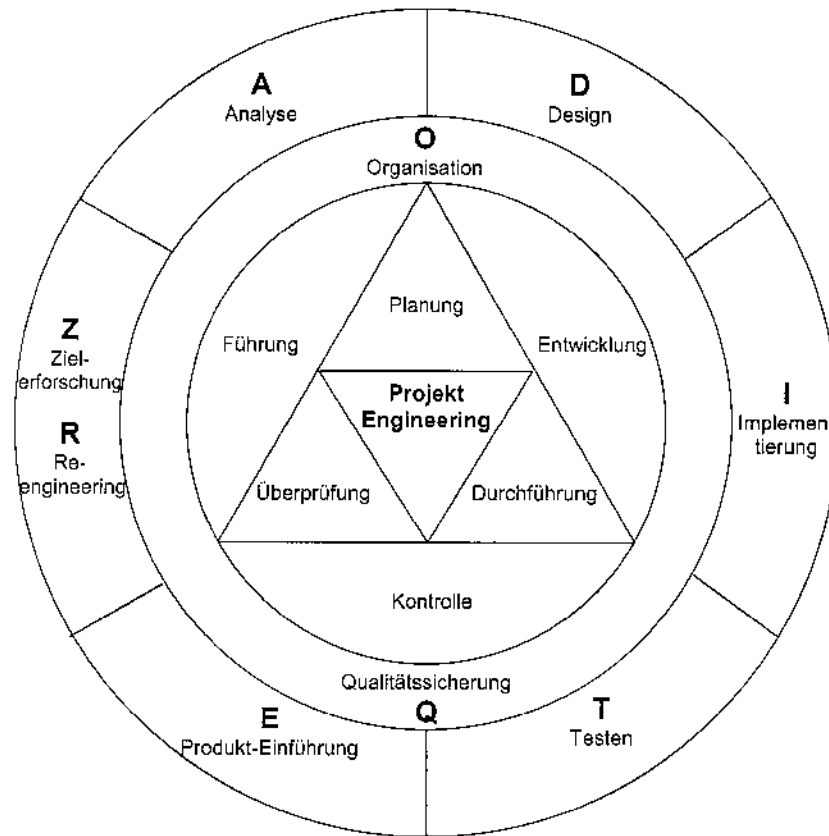
Werkzeugunterstützung

Softwareentwicklungsumgebungen

Case-Tools, Case-Systeme

- Planungswerkzeuge
- Prototypingwerkzeuge
- Analysewerkzeuge und Designwerkzeuge für funktionsorientierte, datenorientierte und objektorientierte Zerlegung
- Implementierungs- und Testwerkzeuge wie Editoren, Compiler, Interpreter, Debugger, Browser, Versionskontrolle, Codegeneratoren, etc
- Dokumentationswerkzeuge und Werkzeuge zur Dokumentationsverwaltung

Aufgabenschema des Projekt Engineering



Quelle: H. Mayr: Projekt Engineering, Fachbuchverlag Leipzig, 2001

5. Projektspezifische Zielerhebung und Zielfindung

Ziele und Anforderungen ändern und entwickeln sich im Laufe eines Projekts.

Ziele ändern sich schleichend. Eine der schwierigsten Aufgaben ist das Erkennen, ob und wann Planadaption oder Umplanung erforderlich sind.

Ergebnis der Zielerhebung und –findung ist eine möglichst klare Beschreibung der unveränderlichen Gesamtziele eines Projekts

Systemanalyse

Erhebung

meist (unbezahlte) Vorleistung dient der
Projektanbahnung

Ziel (objectives capturing):

Feststellung der Projektausgangslage
durch potentiellen Auftragnehmer

im Umfeld des potentiellen Auftraggebers

Erhebung: Aufgaben

- Zusammenstellen der Projektvorarbeiten seitens des potentiellen Auftraggebers
- Definieren des Projektumfeldes

Projektumfeld:

- Organisation
 - Operationale Struktur
 - Strategisches Entwicklung
- möglichst von außen beobachten

Erhebung: Kriterien für Techniken

2. nach der Art der Interaktion

- auditiv: persönliches oder telefonisches Interview
- schriftlich: Fragebogen oder Auswertung von Dokumenten

3. nach dem Strukturierungsgrad

- nicht strukturiert, offen
- halb strukturiert, vorgegebene Fragen
- strukturiert, geschlossen, Fragen und Antworten vorgegeben

4. nach dem Objekt der Erhebung

- Primärerhebung: Befragung und Beobachtung
- Sekundärerhebung: Dokumentenauswertung

5. nach der Anzahl der Erhebungsobjekte

- Einzelerhebung: ein Element
- Fallstudie: einzelne Elemente
- Repräsentative Erhebung
- Totalerhebung

Erhebung: Interview

Interaktionsprozess mit einer Rollenaufteilung zwischen Interviewer und Interviewtem

Einzel- oder Gruppeninterview

Von Angesicht zu Angesicht oder telefonisch

Multimedial oder auditiv (oder textuell)

Strukturierungsgrad

Erhebung: nicht und halb strukturierte Interviews

Offenes Gespräch bei dem **nur ein Thema** vorgegeben ist

Hohe Anforderungen an **Interviewer**

- Interessante Antworten erzwingen
- Auf Antworten flexibel reagieren
- Fragen richten sich nach vorangegangenen Antworten
- Reaktion auf verbale und nicht verbale Signale
- Aufbau eines roten Fadens
- Vertrautheit mit Themenbereich und Umfeld

Interviewleitfaden

- Thema strukturiert in Teilthemen
- Fragen teilweise vorgegeben

Erhebung: Fragetypen

- Aufschließende versus abschließende Fragen
- Offene versus geschlossene Fragen
- Direkte versus indirekte Fragen

Erhebung: Strukturiertes Interview

Mischung aus Interview und Fragebogen

Interviewer übernimmt das Ausfüllen eines Fragebogens

Fragebogen ist auch Protokoll

Fragebogen bei strukturiertem Interview weniger formalisiert als bei reiner Fragebogenaktion

Erhebung: Fragebogen

- hoher Strukturierungsgrad
- hohe Entwicklungskosten
- niedrige Durchführungskosten
- große Stichproben
- Totalerhebung

Interviewer: strukturiertes Interview

Ausfüllen **von Befragten**

- direkt
- schnell
- ohne Rückfragen

Fragebogenerhebung

2. Zielgruppe auswählen

3. Fragen formulieren

- *KROKUS-Regel*: kurze, nicht redundante, offene, konkrete, keine Unter- und Kettenfragen, keine Suggestivfragen
- *Bearbeitungsformen*: Ankreuzen oder Nennung von Alternativen, Reihung von Alternativen, Skalierung

4. Fragebogen zusammenstellen

Dramaturgie: Motivation zur Mitarbeit, Einleitungsfragen um Interesse zu wecken, schwierig und heikel nicht am Anfang, Abschluss durch Ausfüller wie Personendaten oder generelles Statement

5. Fragebogen testen: Vortest als strukturiertes Interview

6. Fragebogen einsetzen: Rücklaufquote berücksichtigen

7. Fragebogen auswerten: Problem ungenaue und unvollständige Antworten, Problem unwichtige und nicht sensitive Antworten

Erhebung: Dokumentenauswertung

Auftraggeber besitzt Dokumente aus (internen) Projektvorarbeiten
zur Verfügung

nach diesen Dokumenten fragen

- Daten- und Informationsfluss
- Ergebnisse einer Projektentwicklung
- Bestellscheine
- Lieferscheine
- Arbeitsanweisungen
- Ablaufdiagramme
- Vorschläge im Rahmen eines Verbesserungswesens
- Konkurrenzprodukte
- Produktvergleiche

Erhebung: Ergebnisauswertung

Ist ein Dokument, das die Auswertung der Dokumente zur Projektanbahnung enthält

- Interviews
- Besprechungen
- Fragebogenaktionen
- vorhandene Einzeldokumente
- etc

Ist die Problembeschreibung aus Sicht des Auftragnehmers und damit kein bindendes Dokument

Bindende Festlegung erst durch nachfolgendes Lastenheft oder Auftrag

Zielfindung

Projektziele des Auftraggebers oft nebulös
Durch Erhebung ergeben sich zusätzliche Aspekte, die
Projektziel ändern oder erweitern

Ziel (objectives elicitation):
Gesamtziel des Projekts und damit verbundene
Grobziele erarbeiten

Zielfindung: Aufgaben

- Ziele gewichten und klassifizieren
 - Mussziele** (*essential objectives*)
 - Sollziele** (*conditional objectives*)
 - Wunschziele** (*optinal objectives*)Zielkonflikte beachten
- Beweglichkeit der Ziele und Auswirkungen analysieren
- Organisationsvorgaben festlegen
 - Unternehmens-, Abteilungs-, Projektziel**
- Zielarten herausarbeiten
 - technische, organisatorische, finanzielle, etc.**

Zielfindung ist kreativer, d.h. schöpferischer, Prozess

Zielfindung: Kreativitätstechniken

sind Methoden zur Definition und zur Lösung schlecht strukturierter Probleme durch Anwendung intuitiver Probiervverfahren

gemeinsam sind Gruppensitzungen mit Teilnehmern aus verschiedenen

Tätigkeitsbereichen, verschiedenen hierarchischen Ebenen und unterschiedlichen Kenntnissen und Fertigkeiten

Merkmale:

- das Zerlegen von Problemen in Teilprobleme
- Das Generieren von Lösungsvorschlägen zu den Teilproblemen
- Das Auswählen der besten Teillösungen und deren Synthese zur Problemlösung
- Das Steuern des Gruppenprozesses durch Moderator

Zielfindung: Kreativitätstechniken

Brainstorming:

- keine Kritik während der Sitzung
- freie Gedankenspiele, originelle und neuartige Ideen
- viele Ideen produzieren
- Ideen anderer Mitglieder weiterentwickeln

Vorteile: Viele Ideen in kurzer Zeit, verschiedene Sichten, einfaches Verfahren

Nachteile: manches realitätsfremd, Verlauf der Ideenfindung beeinflusst Ergebnis (Gruppendynamik)

Zielfindung: Kreativitätstechniken

Brainstorming mit Osborne-Verfremdung:

Brainstorming um Verfremdungsphase erweitert zwischen
Ideenfindung
und Ideenauswertung

- Assoziationen: kombinieren, anders verwenden
- Ähnlichkeiten: anpassen, kleiner machen, größer machen
- Abwandlungen: abändern, ersetzen, neu anordnen
- Gegensätze: umkehren

Zielfindung: Kreativitätstechniken

Synektik:

Problem spontan definieren,
dann durch Fragen umformulieren,
bis Problem zufriedenstellend definiert

Untergliederung:

6. Spontandefinition
7. Umformulierung
8. Antwortenermittlung
9. Problemdefinition
10. Eignungsprüfung
11. Rückkopplung (neue Iteration)

Vorteile: geringe Gefahr vollständig am Ziel vorbeizulaufen, bei größeren Problemen anwendbar, Alternativen begrenzt

Nachteil: mehrere Personen, die mit Problem vertraut sind

Zielfindung: Kreativitätstechniken

Morphologischer Kasten:

Vielzahl von Lösungsmöglichkeiten systematisch suchen,
Problem (Objekt) in Teilprobleme (Parameter) zerlegen,
Für Parameter Ausprägungen festhalten

Schritte:

6. Problem präzise beschreiben
7. zerlegen, Problemelemente bestimmen
8. Für jedes Element Lösungen suchen
9. Lösungen kombinieren und bewerten
10. Bestmögliche Lösung auswählen

Vorteile: fördert systematisches Vorgehen, verhindert voreilige
Entscheidungen, fachübergreifend, Lösung

Nachteil: Gefahr der Unübersichtlichkeit, Auswahl der Lösung
schwierig

6. Projektorganisation

1. Aufbauorganisation
2. Ablauforganisation
3. Dokumentation von Softwareprojekten
4. Dokumente der Projektorganisation

Aufbauorganisation

beschreibt Struktur eines Systems

klassische Organisationslehre

spezifische Organisation für Softwareprojekte

Organisationsformen

- reine Projektorganisation
- Einfluss-Projektorganisation
- Matrix-Projektorganisation
- Einzelkämpfer – Gruppe – Programmierheer
- Hierarchische Teamorganisation
- Chefprogrammierer-Team
- Projektgruppe

Reine Projektorganisation

Taskforce: Ausschließlich projektbezogene Organisation, parallel zur Unternehmensorganisation

Projektleiter leitet direkt Mitarbeiter, ist direkt „nach oben“ verantwortlich, verfügt über Entscheidungskompetenzen und Anordnungsbefugnisse

Projektmitarbeiter werden für Projekt aus bestehender Organisation für die Dauer des Projekts herausgelöst

Vorteile: eindeutige Zuordnung, Anlaufstelle, Identifikation mit Projekt, projektbezogene Entscheidungsfindung, flexible Reaktionen

Nachteile: Schwächung der Abteilungen, „schwache“ Mitarbeiter, Störungen und Konflikte, Wiedereingliederung, unterschiedliche Auslastung

Einfluss-Projektorganisation

funktionale Hierarchie im Unternehmen unverändert

Projektleiter übernimmt Stabsfunktion, damit wenig Kompetenzen, aber für Planung, Verlauf und Durchführung verantwortlich, reine Koordinations- und Informationsfunktion zwischen **Projektmitarbeitern** und beteiligten **Stellen**

Vorteile: Organisation unverändert, Mitarbeiter bleiben in Abteilungen

Nachteile: Nichtübereinstimmung zwischen Autorität und Verantwortung, kein entscheidungsberechtigter und kompetenter Gesprächspartner für Auftraggeber

Matrix-Projektorganisation

zweidimensionale Organisation:

Hierarchie (vertikal) und Projekte (horizontal)

Projektleiter ist Fachvorgesetzter

Abteilungsleiter ist Personalvorgesetzter

Projektmitarbeiter unterliegen Doppelunterstellung

fordert hohes Führungsverständnis und
Konfliktlösungsstrategien und -fähigkeit

Vorteile: Projektleiter selbständig, Projektmitarbeiter bilden
Frühwarnsystem, Spezialwissen verfügbar, keine Versetzungen

Nachteile: Nichtübereinstimmung zwischen fachlicher Autorität und
personeller Verantwortung, Kompetenzkonflikte, Ausspielen der
Vorgesetzten durch Mitarbeiter

Besondere Eigenschaften der Softwareentwicklung

Elzer 1987: „ Die Mehrheit der Softwarespezialisten haben einen akademischen Grad erworben, wenn auch vielfach nicht in Software. Das bedeutet, dass sie in einer Tradition ausgebildet wurden, wo sie an Fähigkeiten gemessen wurden, einmalige oder zumindest außerordentliche Ergebnisse zu erreichen. Außerdem haben gewöhnliche Hochschulabsolventen nie die Notwendigkeit gelernt, Regeln strikt einzuhalten. Diese beiden Ausgangsvoraussetzungen machen es sehr schwer, mit diesem Menschentyp größere Teams zu bilden. Andererseits sind aber bis heute bei Softwareprojekten die wiederholbaren und feststehenden Arbeitsanteile wesentlich kleiner als in traditionellen Konstruktionsprojekten.“

Probleme traditioneller Organisationsformen:

- Hierarchische Kommunikationswege beeinträchtigen Informationsfluss
- Problemstau bei Führungskräften
- Gegenseitige Abhängigkeit von Stellen
- Realisierbarkeit, Kosten, Zeit werden falsch beurteilt
- Mangelndes Ineinandergreifen der Unternehmensbereiche

Einzelkämpfer – Gruppe - Programmierheer

in Klein- und Mittelunternehmen lange

EDV-Universalexperte

große Hard- und Softwarehersteller lange

hunderte bis tausende MJ/PJ pro Projekt

Schrittweise von beiden Seiten hin zu

Projekt- oder Arbeitsgruppe

Kostenanteile EDV inzwischen IT senken

Hierarchische Teamorganisation

- Projektleiter setzt ein
- verantwortliche Leiter für Projektphasen diese
- Gruppenleiter oder Mitarbeiter
- beratender und administrativer Stab für Leiter

Schwächen:

- Projektleiter weit entfernt
- Kommunikationsprobleme
- *Peterprinzip*: Aufrücken bis zur Stufe der Unfähigkeit

Chefprogrammierer-Team

- Projektleiter an Systementwicklung beteiligt
- Einsatz sehr guter Spezialisten
- Beschränkung der Teamgröße

Zusammensetzung

- ChefprogrammiererIn
- ProjektassistentIn
- ProjektsekretärIn
- die Spezialisten

Vorteile: Kontrollfunktion durch involvierten Leiter, Kommunikation einfach, kleine Teams, Berichtsweg institutionalisiert

Nachteile: nur kleine Teams, Personalanforderungen, Organisationsmodell berücksichtigt keinen Ausfall, **Sekretärsfunktion** Bedeutung vs Ansehen

Projektgruppe

aus Chefprogrammierer-Team entwickelt

Anforderungen

- Softwareingenieur: Design, Testen, Implementieren
- flexible Rollenwechsel in der Gruppe
- Projektleiter besitzt Management und inhaltliches Verständnis
- Dokumentationsaufgaben verteilt (statt SekretärIn)

Vorraussetzungen:

- Identifikation aller mit den Projektzielen
- Fähigkeit zur Zusammenarbeit
- Fachwissen und fachbez. Allgemeinwissen aller Gruppenmitglieder
- Führungsvermögen des Projektleiters
- andauernde Motivation und Konfliktlösungen

Projektgruppe

Integration in klassische Organisationsformen als

projektorientierte Teilorganisation

- Projektgruppe aus Mitarbeitern einer Abteilung
- Übertragung von Kompetenzen an Projektleiter
- Projektleiter aus Abteilung
- für Dauer des Projekts ist Projektleiter der Abteilung übergeordnet

Projektleitung in der Linienorganisation:

- Verantwortung für Projekt an Abteilung(sleiter)

Leistungsfähigkeit

Faktoren

- Realistische Planvorgaben
- Klare Kompetenzen
- Jedes Gruppenmitglied bedeutend für Erfolg
- Arbeitsbedingungen
- Projektleiter Anlaufstelle auch für persönliche Probleme
- Konflikte artikulieren und beheben

Produktivität

- Programmumfang 5:1
- Programmierzeit 25:1
- Testzeit 26:1
- Laufzeit der Ergebnisse 13:1

Kleine Subgruppen, Zusammenschweißen, Nutzleistung vs Bindleistung

„Ein Projekt, das in Verzug ist, gerät durch Hinzufügen neuer MitarbeiterInnen nur noch

mehr in Verzug“ (Brook'sches Gesetz)

Rollen: Organigramm plus Stellenbeschreibungen

Rollen:

- Projektaufsicht, Projektkoordinator
- Projektleiter
- Systembetreuer
- Werkzeugkoordinator
- Sicherheitsbeauftragter
- Bildungskoordinator
- Experte
- PR-Verantwortlicher
- Administrativer Projektleiter, Stellvertretender Projektleiter, Projektmanager
- Archivar, Klassenmanager, Librarian
- Dokumentkoordinator
- Softwareingenieur
- Testsuite-Verantwortlicher
- (Pilot-)Anwender
- Qualitätsmanager
- Standard-Überwacher

Ablauforganisation

Arbeitsvorgänge organisieren

Verantwortlichkeiten formalisieren

Möglichkeiten zum Rationalisieren, Vereinfachen, Standardisieren

Vorgehensmethoden

- Phasenmodell
- Wasserfallmodell
- Spiralmodell
- Objektorientiertes Modell (Tool-Smithing)
- Prototypingorientiertes Vorgehen
- Schablonenmodell

Besprechungen

Reviews

Dokumentation von Softwareprojekten

- Dokumentationsplanung
- Berichtswesen
- Dokumentationsorganisation
- Dokumentverwaltung
- Dokumentablage
- Dokumentkonfiguration

Dokumente der Projektorganisation

Grundlegende Dokumente

- Brief
- Fax
- Aktennotiz
- Telefonnotiz, Gesprächsnotiz
- Agenda
- Protokoll
- Bericht

Projekthandbuch

Projekttagbuch

Projektfortschrittsbericht

Arbeitsbericht

Projektbibliothek

7. Risiken und Prozessvariablen

1. Risiken
2. Grundlegende Variablen
3. Metapher für Entwicklungsprozess
4. Werte als Kriterien

nach Kent Beck: Extreme Programming

Risiken

- **schedule slips**
- **project canceled**
- **system goes sour**
- **defect rate**
- **business change**
- **false feature rich**
- **stuff turnover**
- **business misunderstood**

Lösungsideen

schedule slips:

- kurze Zyklen
- 1-4 Wochen Iterationen
- 1-3 Tage pro Aufgabe
- implementiere wichtige Sachen zuerst

project canceled

Kunde wird aufgefordert
kleinsten Aufgabenkatalog auszusuchen,
der für sein Geschäft am sinnvollsten ist

Lösungsideen

system goes sour:

- verständliche abgeschlossene Suite von Tests aufbauen;
- Tests müssen nach jeder Änderung funktionieren
- Kontroverse a priori versus a posteriori Suite

defect rate:

Tests aus Entwickler- **und** Kundenperspektive

Lösungsideen

business misunderstood:

- Kunde integraler Bestandteil des Teams
- Spezifikation flexibel halten
- vom Kunden lernen

business change:

- kurze Zyklen
- Kunde willkommen neue Funktionalität zu ergänzen

Lösungsideen

false feature rich:

nur höchst priore Aufgaben werden angegangen

stuff turnover:

Entwickler akzeptieren Verantwortlichkeit
für eigene Arbeit

Variablen

Vier Variablen der Softwareentwicklung

- **Kosten**
- **Zeit**
- **Qualität**
- **Aufgabenbereich**

Vorsicht: Externe Kräfte

Metapher

Learning to Drive

Driving is not about getting the car in the right direction.

Driving is about constantly paying attention,

making a little correction this way,

making a little correction that way.

Kriterien

Vier Werte

- **Kommunikation**
- **Einfachheit**
- **Rückkopplung**
- **Mut, Courage**

Kurz-Zeit-Ziele versus Lang-Zeit-Ziele

Individuum versus Gesellschaft

Balance vermittelt über

Mythen, Rituale, Strafen, Belohnungen

Kriterien

Grundprinzipien zur Auswahl zwischen Alternativen

- **Schnelle Rückkopplung**
- **Einfachheit**
- **Stufenweise Änderung**
- **Wahrnehmbare Änderung**
- **Quaitätsarbeit**

8. Qualitätssicherung

1. Ziele
2. Aufgaben
3. Techniken
4. Qualitätshandbuch
5. Normen und Richtlinien

Ziele: Qualität von Software

- **Qualität** ist gegeben, wenn gestellte Anforderungen erfüllt wird
- Qualität ist nicht Spitzenqualität
- Ziel ist die geplante nicht die bestmögliche Qualität zu erreichen
- **Qualitätsmerkmale** legen Anforderungen fest
- keine allgemein anerkannte Definition
- keine eindeutige, objektive Bewertung möglich

Ziele: Qualitätsmerkmale

- Korrektheit
- Zuverlässigkeit
- Nutzungsfreundlichkeit: Adäquatheit, Erlernbarkeit, Robustheit
- Wartungsfreundlichkeit: Lesbarkeit, Erweiterbarkeit, Testbarkeit
- Effizienz
- Portabilität

Ziele

Wechselwirkungen zwischen Qualitätsmerkmalen und Wirkung auf **Kosten** (Entwicklung, Betrieb, Wartung, Übertragung) und **Zeitgrößen** (Entwicklungs-, Lebenszeit)

Qualitätssicherung ist Sicherung der Produktqualität über den gesamten Produktentwicklungsprozess

Qualitätssicherung bezieht sich auf

- Projektziele,
- organisatorische Abläufe,
- Zusammenwirken von Ressourcen,
- Arbeitsweise

Aufgaben

- Qualitätsplanung: Festlegen der Qualitätsmerkmale
- Qualitätsprüfung: Prüfung der Projektgrundlagen
- Qualitätslenkung: Organisatorische Maßnahmen
 - *Konstruktive Maßnahmen*: Anwenden von Methoden, Einsetzen von Entwicklungswerkzeugen, Einsatz von Halbfabrikaten, Entwicklungsdokumentation
 - *Organisatorische Maßnahmen*: Vorgehensmodelle, Weiterbildung, Institutionalisierung der Q.-Sicherung, Dokumentationsrichtlinien
 - *Analytische Maßnahmen*: statische und dynamische Programmanalyse, systematische Testfälle, Protokollieren der Analyseergebnisse, system. Melden und Behandeln von Fehlern,
 - Prüfen und Beurteilen des Qualitätssicherungssystems selbst

Techniken: prozessorientierte Qualitätssicherung

- **Capability Maturity Modell (CMM):** Reifegrade, die den Grad der Beherrschung des Entwicklungsprozesses charakterisieren; von US-Luftwaffe initiiert; von SEI der CMU entwickelt
- **ISO 9000:** Qualitätsnorm des ISO; „Qualität ist die Fähigkeit alle Erwartungen eines Kunden von Produkten und Dienstleistungen zu erfüllen“; sehr umfangreiche Sammlung von Standards; System von Normen
- **Bootstrap:** EU-ESPRIT-Projekt; Grundlage ist Prozessmodell der ESA; ISO 9000 Gap-Analyse
- **SPICE:** Software Process Improvement and Capability dEtermination; „follow CMM, certify ISO“; Sammlung von 7 Handbüchern für Zulieferer und Anbieter; Werkzeug zur kontinuierlichen, messbaren Prozessverbesserung

Qualitätshandbuch

Enthält:

- **Priorisierung der Q.-Merkmale**
- **Q.-Ziele in messbaren Größen der Q.-Merkmale**
- **Q.-Merkmale für Produkt und Entwicklungsprozess**
- **Maßnahmen zur Q.-Lenkung**
- **Festlegung von Kompetenzen und Verantwortungen für alle Q.-Sicherungsmaßnahmen**

Tendenz zu ISO 9000ff

Normen und Richtlinien (Standards)

Normungsgremien:

ISO, IEC, ANSI, CEN, CENELEC, DIN, DEK, CCITT, VDA

Gremien für Industriestandards:

European Computer Manufacturer Association (ECMA), Ethernet-Konsortium, weiter lose Vereinigungen von Herstellern

Gremien für Richtlinien:

VDI, VDE, IEEE, IFIP, IFAC

Normen und Richtlinien für Softwareentwicklung:

Nutzungsschnittstellen, Datenbanken, Schnittstellentechnik, Anwendungsprogrammierung, Dokumentation

Normenauswahl

- von Normen zu Richtlinien von international zu national
- Unternehmensintern dann übrige Standards (sinnvolle)
- „The beauty of standards is that there so many to choose from!“
[Tanenbaum, 1990]