

PROJEKT ENGINEERING

Aufbauorganisation

Herwig Mayr

Fakultät für Informatik, Kommunikation und Medien
Fachhochschule OÖ, Hagenberg

Aufbauorganisation

beschreibt die **Struktur eines Systems**
(engl.: „organization structure“)

Unterscheide:

- **allgemeine Organisationsformen**
- **softwareentwicklungsspezifische Organisationsformen**



(Bild: macroart, fotolia.com)

Allgemeine Organisationsformen: Reine Projektorganisation

ausschließlich projektbezogen („Taskforce“); Projektmitarbeiter werden aus der bestehenden Organisation herausgelöst.

Vorteile:

- + eindeutige Aufgabenzuordnung
- + volle Konzentration auf Projekt
- + Identifikation mit der Aufgabe
- + rasche Entscheidungen
- + flexible Reaktion

Nachteile:

- Schwächung der Abteilungen
- Abstellung der am leichtesten entbehrlichen Mitarbeiter („Mr. Bean-Effekt“)
- Konflikte mit angestammten Abteilungen
- unterschiedliche Auslastungen

-> geeignet für schwierige, kleine Projekte

Beispiel: Wo findet man Reine Projektorganisationen?

Allgemeine Organisationsformen: Einfluss-Projektorganisation

funktionale Hierarchie unverändert; Projektleiter als **Stabsfunktion**

Vorteile:

- + Abteilungsorganisation unverändert
- + Sicherheitsgefühl für Mitarbeiter

Nachteile:

- mangelnde Autorität des Projektleiters
- Arbeiten am Projekt nur, "wenn Zeit ist"
- Konflikte werden verzögert

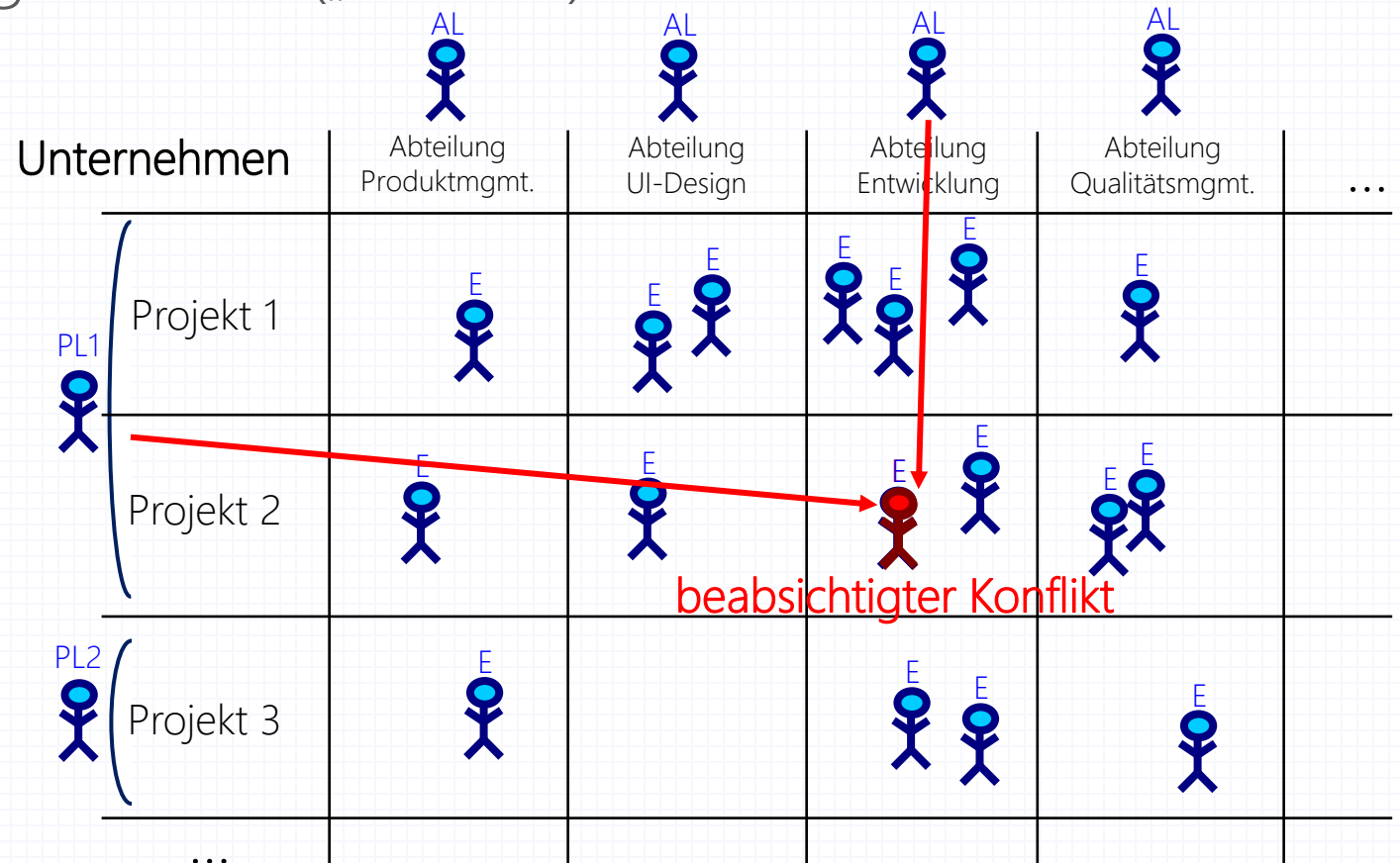
-> geeignet für gut strukturierte, inhaltlich einfache Projekte

Beispiel: Wo findet man Einfluss-Projektorganisationen?

Allgemeine Organisationsformen: Matrix-Projektorganisation (I)

zweidimensionale Organisationsform (Unternehmenshierarchie und Projekt) mit doppelten Vorgesetzten („Matrix“)

Idee des
beabsichtigten Konflikts:



Allgemeine Organisationsformen: Matrix-Projektorganisation (II)

zweidimensionale Organisationsform (Unternehmenshierarchie und Projekt) mit doppelten Vorgesetzten („Matrix“)

Vorteile:

- + selbstständiger Projektleiter
- + Mitarbeiter bleiben in Abteilung
- + Expertenwissen leicht einbaubar

Nachteile:

- Verunsicherung bei Vorgesetzten
- Verunsicherung bei Mitarbeitern
- man „sitzt zwischen zwei Stühlen“
- Kompetenzkonflikte
- Möglichkeit, Vorgesetzte auszuspielen

-> geeignet für zeitunkritische Projekte

Beispiel: Wo findet man Matrix-Projektorganisationen?

Softwareentwicklungsspezifische Organisationsformen (I)

Probleme traditioneller, allgemeiner Organisationsformen:

- hierarchische (langsame) Kommunikationswege
- Problemstau bei Führungskräften
- gegenseitige Abhängigkeit von Stellen
- schlechte Koordination einzelner Bereiche
- unrealistische Planung



Softwareentwicklungsspezifische Organisationsformen (II)

Besondere Anforderungen an Projektorganisationsstrukturen in der Softwareentwicklung:

- zeitliche Begrenztheit
- hohe Flexibilität
- projektübergreifende Planung



Einzelkämpfer – Gruppe – Programmierheer

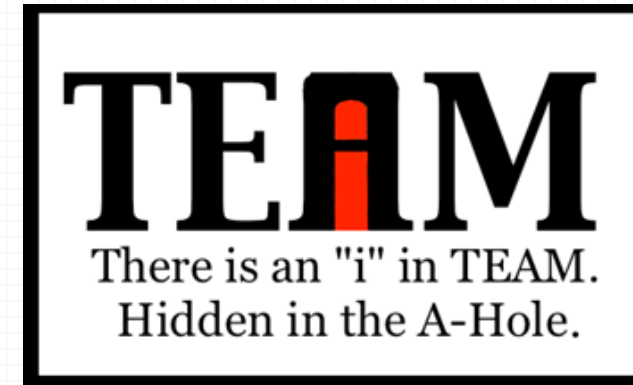
Kriterium	"Einzelkämpfer"	Gruppe	"Programmierheer"
Organisation	Improvisation, hohes Ausfallrisiko	gemeinsames Arbeiten, gruppendynamische Effekte	Aufgaben(zer)-teilung, geringes Produktrisiko
Motivation	extrem hoch, Selbstaussbeutung	hoch (Klima entscheidend), Tendenz zur Ausbeutung	niedrig, „Mitschleppen“ unproduktiver Mitarbeiter
Vorgangsweise	innovativ bis verspielt, unreflektierte Entscheidungen	innovativ, direkte Kommunikation, rasche Fehlerkorrektur	starr, indirekte Kommunikation, komplexe Methoden und Werkzeuge
Dokumentation	stark entwicklerbezogen, primär für Eigenbedarf	standardisiert, knapp	standardisiert, umfangreich
Testen / Kontrolle	ungenügend (Personalunion!)	führungsabhängig, Tendenz zu gering	umfangreich, sehr indirekt (Ablehnung!)
Projektgröße	klein	klein bis mittel	„immer“ groß

-> Gruppe führt (bei kleinen/mittleren Projekten) zu bester Leistung!

Softwareentwicklungsspezifische Organisationsformen

Besondere Eigenschaften der Softwareentwicklung aus Organisationssicht:

- Softwareentwicklung wird eher als planungszentrierter Prozess denn als Produktionsprozess gesehen.
- Es gibt oft mehrere gute, aber selten eine „beste“ Lösung.
- Softwareentwickler-Teams weisen einen sehr hohen Akademikeranteil auf.
- Projektfortschritt und Zielerreichungsgrad sind schwer messbar („Immaterialität“ von Software).



(Bild: www.hanselman.com)

Softwareentwicklungsspezifische Organisationsformen:

Hierarchische Teamorganisation

Zerlegung in Phasen und Einrichtung entsprechender Abteilungen und Leiter (z.B. Analyse, Programmierung, Testen)

Vorteile:

- + entspricht anderen Industriezweigen

Nachteile:

- Projektleiter weit vom Geschehen entfernt
- langsame, indirekte Kommunikation
- Peterprinzip!

-> für Softwareentwicklung zu bürokratisch und undemokratisch

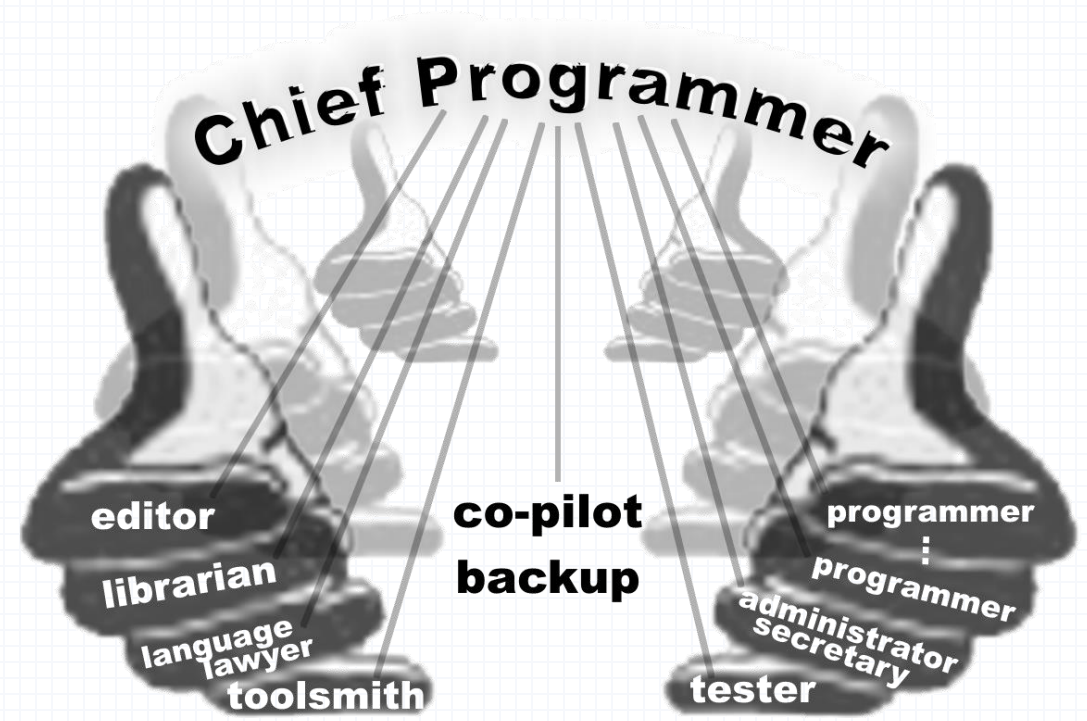
Softwareentwicklungsspezifische Organisationsformen:

Chefprogrammierer-Team (I)

dem Chirurgen-Team nachgebildet [Baker, 1972]; kleine Teamgröße, Spezialisten, Projektleiter arbeitet aktiv mit

Teammitglieder:

- Chefprogrammierer
- Projektassistent
- Projektsekretär
- mehrere Spezialisten (2-6)



(Bild: iguw.tuwien.ac.at)

Softwareentwicklungsspezifische Organisationsformen:

Chefprogrammierer-Team (II)

dem Chirurgen-Team nachgebildet [Baker, 1972]; kleine Teamgröße, Spezialisten, Projektleiter arbeitet aktiv mit

Vorteile:

- + Chefprogrammierer direkt eingebunden
- + gute Kommunikation
- + Berichtswesen institutionalisiert

Nachteile:

- nur kleine Teams möglich
- Personalanforderungen kaum erfüllbar
- undankbare Aufgabe des Projektsekretärs

-> Es wurde gezeigt, dass kleine Teams besser sein können als Pgm.-heere!

Softwareentwicklungsspezifische Organisationsformen:

Moderne Projektgruppe (I)

Änderungen in der Softwareentwicklung erfordern Anpassungen des Chefprogrammierer-Team-Konzepts:

- Ein Softwareingenieur muss alle Schritte des Entwicklungsprozesses beherrschen.
- Mitarbeiter müssen Kollegen vertreten können.
- Ein Projektleiter, der gleichzeitig ein guter Implementierer ist, existiert kaum (bzw. ist für die Implementierung zu teuer).
- Die Rolle des Projektsekretärs („Chefdokumentierers“) ist undankbar und risikoreich.

-> Entwicklung hin zur **Modernen Projektgruppe**

Softwareentwicklungsspezifische Organisationsformen: Moderne Projektgruppe (II)

Einbindung in die Organisation:

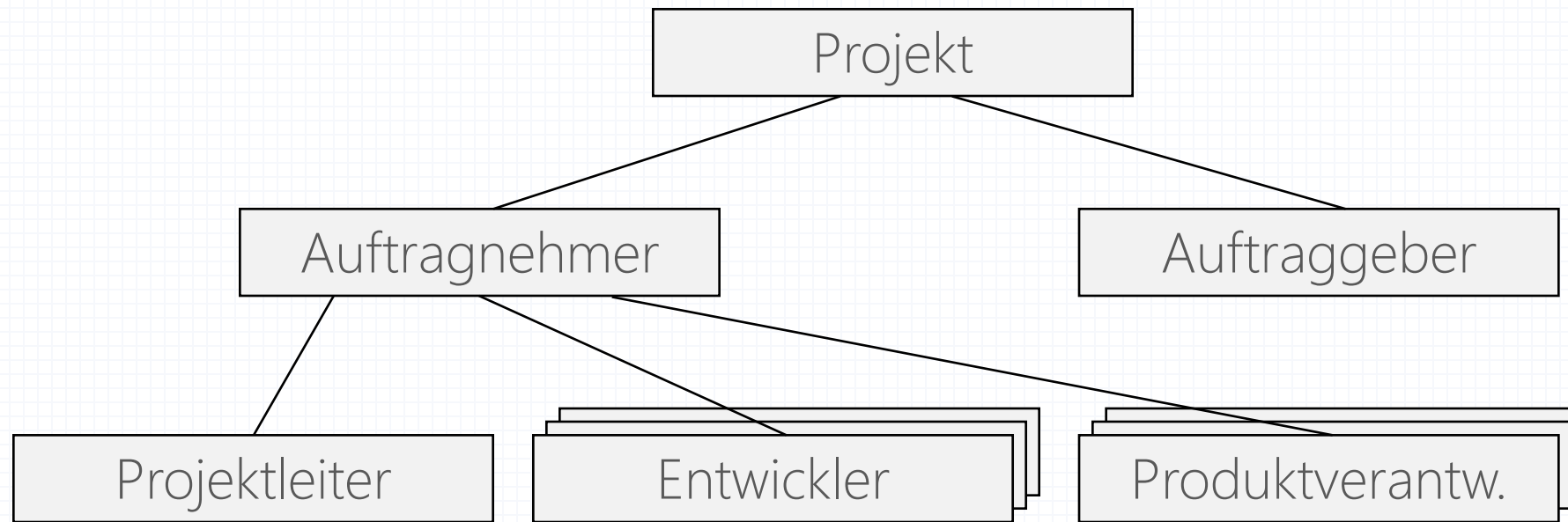
- als projektorientierte Teilorganisation („[agile] Teams“)
- durch Projektleitung in der Linienorganisation
- bei mehreren kooperierenden Gruppen:
Leitergruppe



(Bild: Michael Marsiglia;
spin.atomicobject.com)

Softwareentwicklungsspezifische Organisationsformen:

Agile Teams (I) – Rollen



Softwareentwicklungsspezifische Organisationsformen:

Agile Teams (II) – Projektleiter*in vs. Produktverantwortliche*r

Projektleiter*in (PL)	Produktverantwortliche*r (PV)
„organisatorische Projektleiterin“	„inhaltliche Projektleiterin“
prozessbezogene Projektleiterin	produktbezogene Projektleiterin, Pilotanwenderin
auch <i>Scrum Master, Gewissen, Coach, Big Boss</i>	auch <i>Product Owner, On-site Customer, Virtual Customer, Tester</i>
1 mit Stv.	1-n mit Stv. (pro Anforderung genau 1!)
teilzeitlich! passt gut zu RM, schlecht zu PR	teil- oder vollzeitlich! passt gut zu PR, schlecht zu RM

Softwareentwicklungsspezifische Organisationsformen:

Agile Teams (III) - Aufgabenverteilung PL / PV / Entwickler

- Projektleiter*in:** überwacht Ablauf
versteht, steuert und dokumentiert Prozess
teilt Aufgaben zu
- Produktverantwortliche*r:** bewahrt Produktsicht (\neq *Entwickler* !)
treibt dadurch Entwicklung voran (Anforderungsauswahl!)
erstellt Testfälle
- Entwickler (2 - 7):** schätzen und wählen Aufgaben
entwickeln (primär Designen-Implementieren-Testen)
übernehmen andere Aufgaben (z.B. Systemadministratorin,
Weiterbildungskoordinatorin) teilzeitlich
(*Gruppenorganisation ist vorgehensspezifisch!*)

Gruppendynamische Prozesse

Kommunikation der Mitglieder sowie deren Koordination durch den Projektleiter ist der entscheidende Erfolgsfaktor!

Problem: oft sehr technische Führungssicht

Gruppendynamische Prozesse führen zur Selbstordnung.

Hohe **Ethik und Moral** notwendig -> Profession des Softwareingenieurs



Motivation der Mitarbeiter

Software Engineering ist eine **humanzentrierte Tätigkeit!**

- realistische Planvorgaben
- klare Kompetenzen
- Gefühl individueller Wichtigkeit,
- akzeptable Arbeitsbedingungen,
- auch privat gutes Gesprächsklima notwendig
- Konflikte ansprechen und beheben!
(Gewinner *und* Verlierer)



Parkinsonsches Gesetz: Man braucht immer (mindestens) so viel Zeit, wie man hat! -> Dauernde leichte Überforderung führt zu besserer Arbeit!

Produktivität (I)

Leistungsfähigkeit gleich gebildeter Programmierer:

- Programmumfang: 5 : 1
- Programmierzeit: 25 : 1
- benötigte Testzeit: 26 : 1
- Ergebnislaufzeit: 13 : 1



-> Mit Mittelwerten kalkulieren ist problematisch!

Beispiel: Welche Auswirkungen hat diese Streuung?

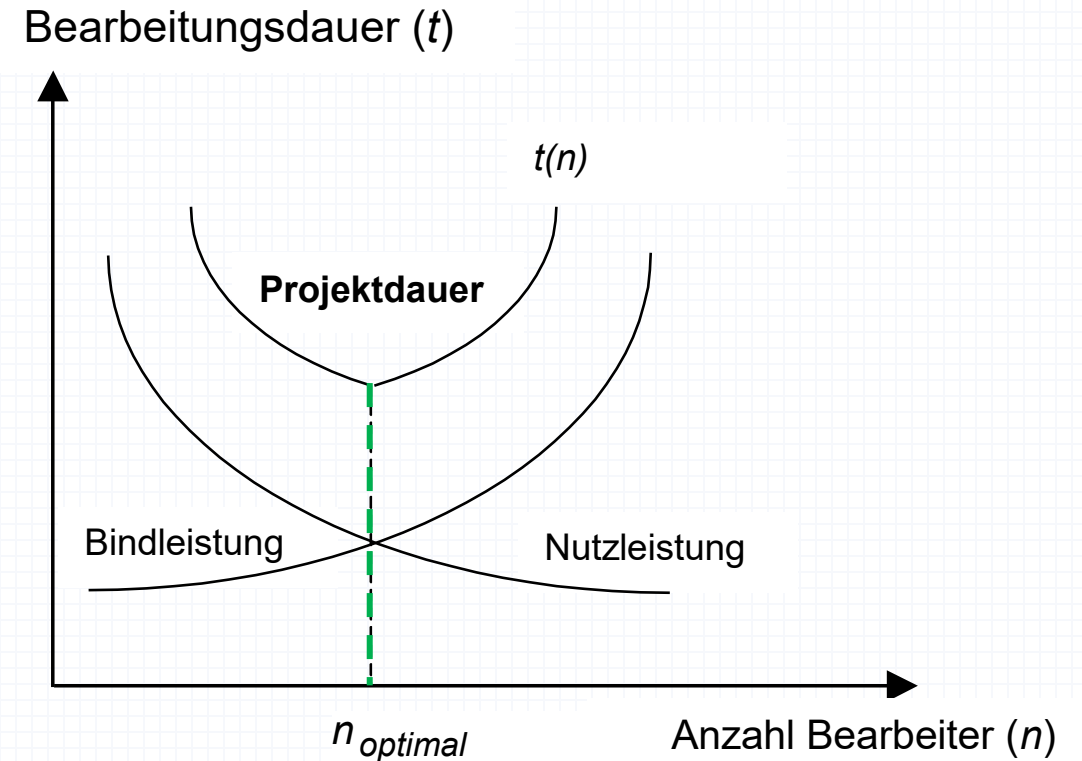
Produktivität (II)

Produktivität einer Projektgruppe:

- produktive Arbeit (**Nutzleistung**)
- Kommunikation (**Bindleistung**)

Optimale Gruppengröße:

- $n_{\text{optimal}} = \sqrt{t}$, wobei
[t] = Personenmonate



Produktivität (III)

Entwicklungszeit in der Gruppe (Brookssches Gesetz):

- $t = 1/n + k n^2/2$, wobei
 - t ... Projektentwicklungszeit
 - n ... Anzahl der Mitarbeiter
 - k ... durchschn. Kommunikationszeit

-> Ein Projekt, das in Verzug ist, gerät durch Hinzufügen neuer Mitarbeiter nur noch mehr in Verzug!

(Voraussetzung:
Projekt ist nicht unterbesetzt.)



Interdisziplinäre Zusammenarbeit

Moderne Projektentwicklungsgruppen bestehen oft aus Personen mit verschiedensten Fertigkeiten (**heterogene Projektgruppen**).

Probleme:

- fehlende gegenseitige Ein- und Wertschätzung der Arbeit
- bei Überlappungen Konflikte
- viele verschiedene Methoden und Werkzeuge

Wichtig:

- Konflikte rasch lösen!
- Schulung, Weiterbildung anbieten!
- Geordnete Änderungs politik der Entwicklungsumgebung(en) beachten!

Organisationsmängel bei (Software-)Projekten

Hauptmängel:

- Fehlen von Planung, Organisation, Standards
- Inkompetenz der Projektführung
- Fehlen aktueller Dokumentation
- Mangel an Fortschritts- und Qualitätskontrolle
- Mangel an Kostenkontrolle

Ansätze zur Verbesserung:

- Verbesserung der Abläufe (Prozesse) -> *Kapitel Ablauforganisation*
- Verbesserung von Information und Kommunikation im Team
-> *Kapitel Informationsaustausch*
- Verbesserung der Qualität -> *Kapitel Qualitätssicherung*



Beispiel: Welche konkreten Projektängel haben Sie schon erlebt?

PROJEKT ENGINEERING

Aufbauorganisation

Herwig Mayr

Fakultät für Informatik, Kommunikation und Medien
Fachhochschule OÖ, Hagenberg