



# **Informatik - kurz gefasst**

© Beat Heil  
Chasernweg 20, CH-8302 Kloten

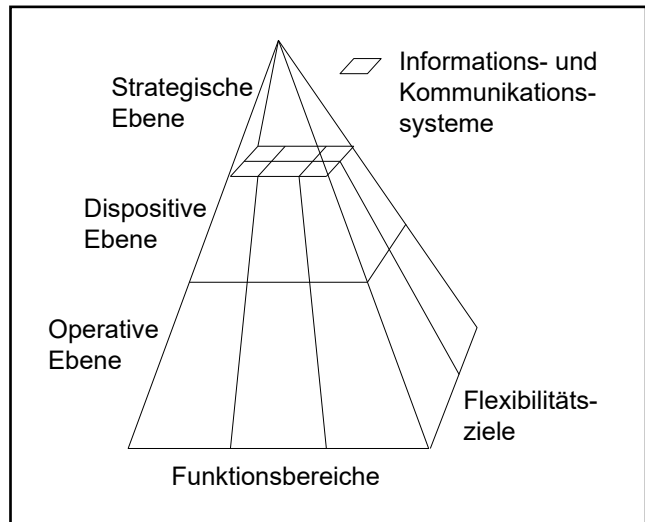
Verwendung mit Quellenangabe erlaubt

## INHALT

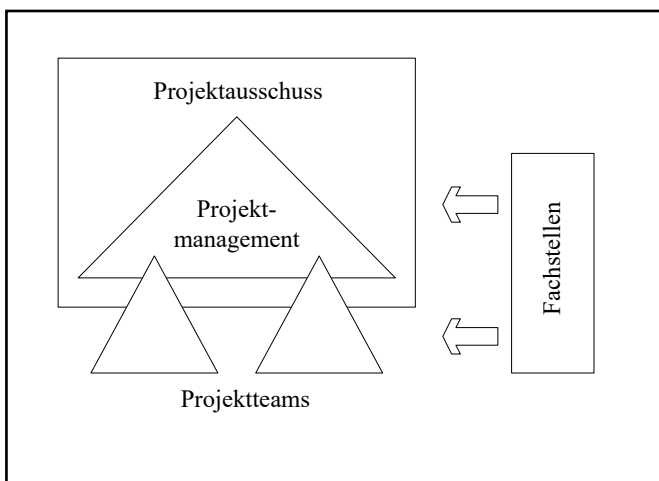
|    |  |   |
|----|--|---|
| 1  | Informationsmanagement.....            | 3 |
| 2  | Projektorganisation.....               | 3 |
| 3  | Verantwortlichkeiten.....              | 3 |
| 4  | Software-Engineering.....              | 4 |
| 5  | Projektmanagement.....                 | 4 |
| 6  | Vorgehensmodell .....                  | 4 |
| 7  | Projektbildung .....                   | 5 |
| 8  | Historisierung .....                   | 6 |
| 9  | XML – Extensible Markup Language.....  | 7 |
| 10 | Qualitätssicherung.....                | 8 |
| 11 | Quellen- und Literaturverzeichnis..... | 8 |

# 1 Informationsmanagement

Die sich rasant verändernden Rahmenbedingungen verlangen nach einer neuen Qualität in der Informationsversorgung. Der Trend, weg von blosser Rationalisierungsdanken zum Informationsmanagement, ist damit angesprochen [HMD90]. Informationsmanagement vereinigt Datenverarbeitungs-, Büro- und Kommunikationssysteme auf der Management-Ebene in einer Hand. Es bedeutet auch, dass Flexibilitätsziele für alle Funktionsebenen und -bereiche in die Gestaltung der Informations- und Kommunikationssysteme einfließen müssen.



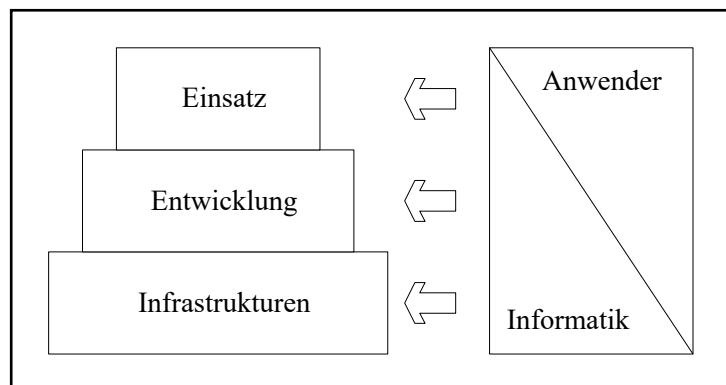
# 2 Projektorganisation



Projekte müssen beim späteren Anwender verankert werden. Wie dies geschehen kann, zeigt die Abbildung. Im Projekt-ausschuss als Entscheidungsträger sind Projektmanagement mit Mitspracherecht, und die Projektteams mit Antragsrecht vertreten. Die Leiter der Projektteams rekrutieren sich aus dem Kreis der Anwender. Ein professionelles Projektmanagement ist absolutes «Muss».

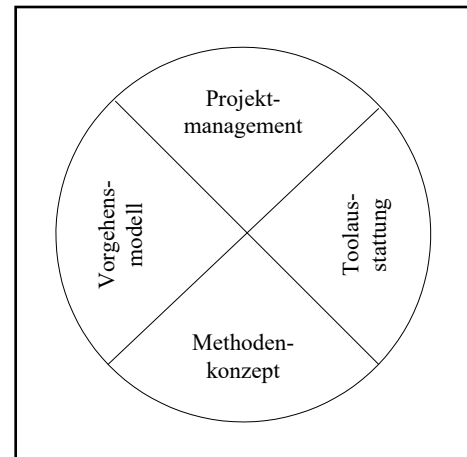
# 3 Verantwortlichkeiten

Während die Informatik für die Bereitstellung und Pflege der Infrastrukturen zuständig ist, bestimmen die Anwender den Einsatz der von beiden gemeinsam entwickelten Systeme. Optimal ist, wenn beide Seiten überlappendes Wissen einbringen.

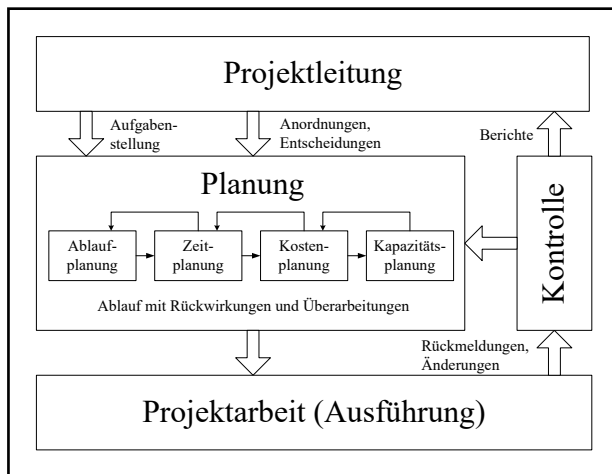


## 4 Software-Engineering

Software-Engineering benutzt wissenschaftlich fundierte, lehrbare Methoden und Prinzipien, die geeignet sind, in kohärenter Weise den Gesamtprozess der Erstellung von Software rechnergestützt, komfortabel, fehlerfrei, wirtschaftlich sowie hinsichtlich der Leistung, der Termine und der Kosten berechenbar zu machen. Die wesentlichen Komponenten sind in der Abbildung aufgeführt. [LEW88].



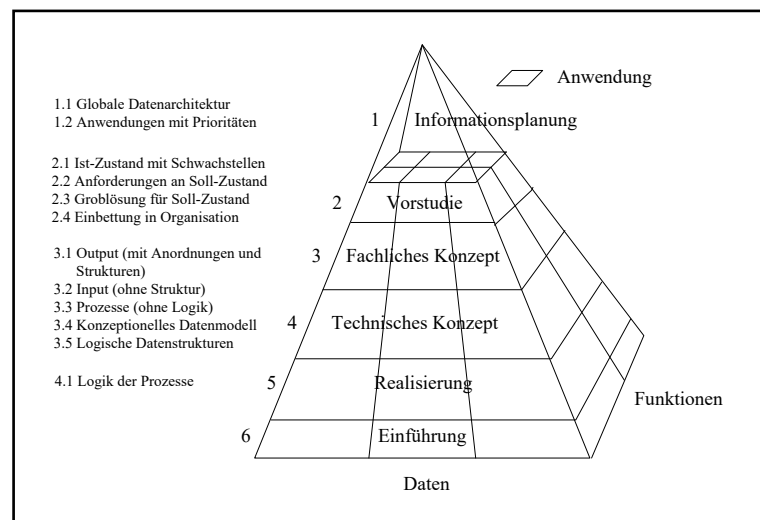
## 5 Projektmanagement



Das Projektmanagement ist auf zweckmässige Instrumente zur Planung und Überwachung von Kosten, Terminen, Einsatzmitteln, Fortschritt, kritischen Pfaden, Dokumentenfluss etc. angewiesen. Das Schwergewicht hat auf Klarheit und projektspezifischer Aussagekraft zu liegen; also keine hoch spezialisierte Methoden und Mittel. [MÜL88], [PIE90], [SVB88].

## 6 Vorgehensmodell

Ein Vorgehensmodell regelt, nach welchen zeitlichen und sachlichen Gesichtspunkten die Planung, Entwicklung und Realisierung von Anwendungen zu erfolgen hat. Es stützt das Vorgehensprinzip vom Groben zum Detail (top-down) und richtet sich auf die beiden Dimensionen Daten und Funktionen. [VET86].



*Informationsplanung:* Dient der Strategiefestlegung. Dabei werden die Anwendungen festgelegt, welche im Anschluss an den Planungsprozess zu entwickeln sind.

*Vorstudie:* Hier geht es in erster Linie darum, das Problemfeld einer Anwendung eindeutig abzugrenzen, die Anforderungen an das Soll-System festzulegen, sowie eine die Rahmenbedingungen berücksichtigende Groblösung zu entwickeln. Die Vorstudie stellt damit ein Klärungsprozess dar, dem eine Entscheidung bezüglich Fortführung oder Abbruch des Projektes folgen muss.

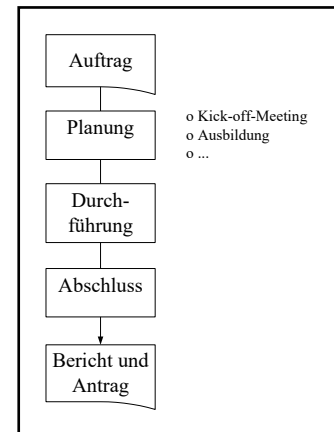
*Fachliches Konzept:* Auf der Vorstudie aufbauend wird das detaillierte Fachkonzept entwickelt. Es werden alle für die Anwendung relevanten Gesichtspunkte berücksichtigt, unabhängig davon, ob die dabei erkannten Prozesse zukünftig automatisiert werden oder nicht.

*Technisches Konzept:* Hier werden die im Fachkonzept enthaltenen Prozesse in eine Form gebracht, welche für die anschließende Programmierung geeignet ist (Programmieren im Grossen [LEW88]).

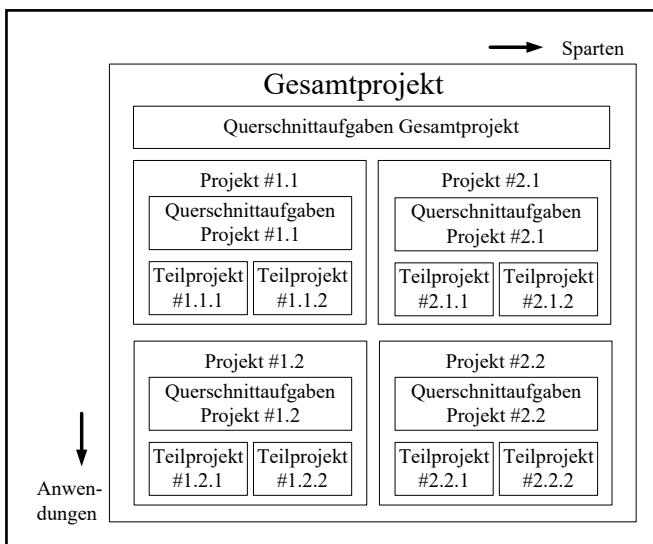
*Realisierung:* Hier werden die im technischen Konzept vorbereiteten Prozesse programmiert, getestet und dokumentiert (Programmieren im Kleinen [LEW88]). Darüber hinaus werden die rahmenorganisatorischen Belange ausgearbeitet, anwenderorientierte Dokumentationen und Bedienungsanweisungen erstellt, Informationswege organisiert und Katastrophenszenarien ausgearbeitet.

*Einführung:* Vor einer Übernahme des Systems in die Produktion werden die Anwender geschult. Dazu wird das Gesamtsystem unter möglichst echten Bedingungen getestet. Nach Einführung wird der Erfolg geprüft. Die Resultate sind von Bedeutung, wenn das realisierte System verbessert, oder ein ähnlich konzipiertes System zu gestalten ist.

Die einzelnen Phasen des Modells werden am besten nach einem Standard abgewickelt, welcher mit dem Auftrag beginnt, mit Bericht und Antrag endet.



## 7 Projektbildung

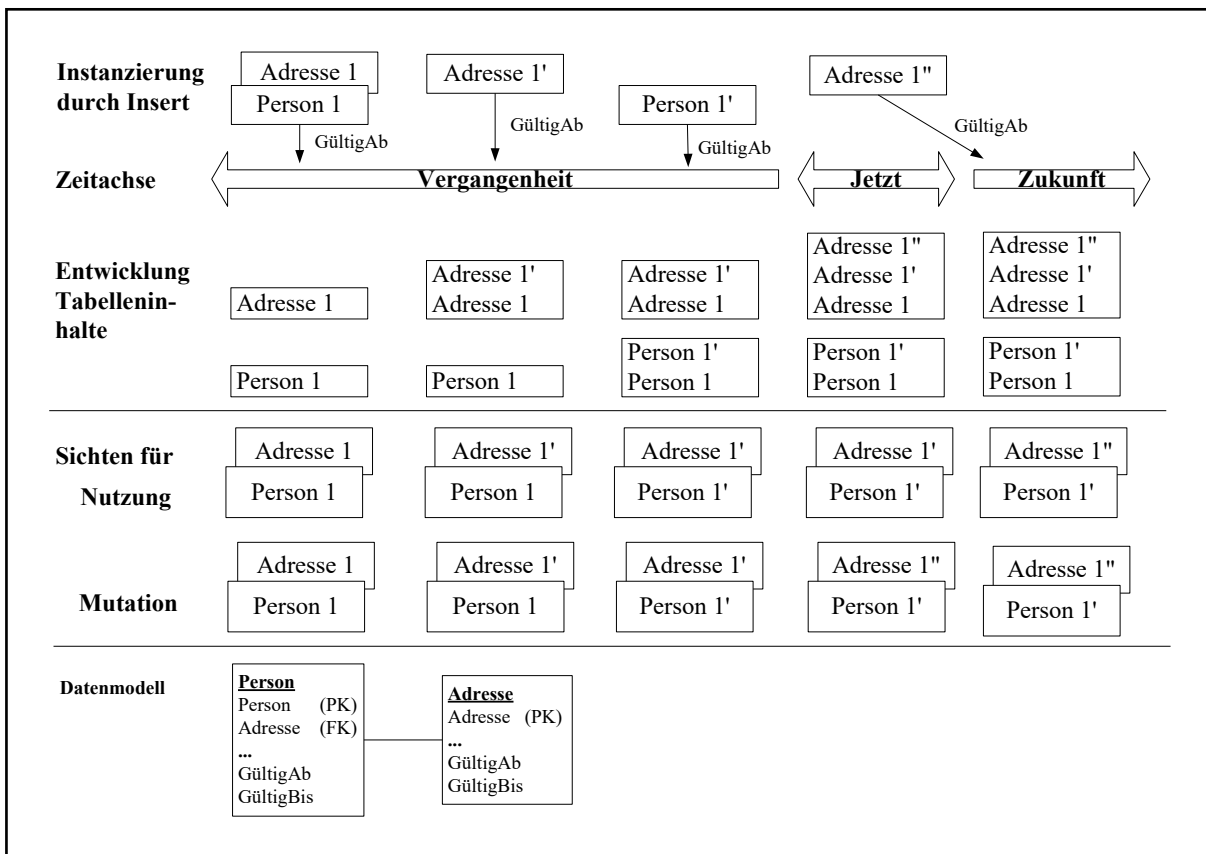


Der Projektbildung muss hinreichende Aufmerksamkeit geschenkt werden, besonders im Hinblick auf die spartenübergreifenden Querschnittsaufgaben. Die Praxis zeigt, dass sich hierbei gemachte Fehler langfristig auswirken und das Gesamtergebnis negativ beeinflussen.

## 8 Historisierung

Für eine konsequente Historisierung sind (a) Instanzierung durch Insert und (b) Speicherung von Gültig-Ab- und GültigBis-Informationen in allen Instanzen zwingend notwendig. GültigBis wird abgefüllt, wenn eine Instanz ohne direkten Nachfolger endet, andernfalls bleibt es leer (NULL). Warum Historisierung? Sekundärfolgen aus Zustandsübergängen in den Daten können nur dann automatisch abgewickelt werden, wenn gleichzeitiger Zugriff auf beide Datenzustände (Vorgänger und Nachfolger) gewährleistet ist: z.B. Berechnung der Prämierendifferenz bei Änderung von Versicherungsverträgen, oder Abwicklung von Verwaltungshandlungen in den Wertpapieren u.a. bei Änderung der Währung oder des Kapitals. Der Verzicht auf eine Historisierung hat schwerwiegende Auswirkungen, wenn dieser Entschluss später korrigiert werden soll. Es handelt sich hier um ein strukturelles Problem, welches sich nur mit horrenden Folgekosten korrigieren lässt. Nachteile einer konsequenten Historisierung sind (a) der höhere Speicherbedarf und (b) komplexere SQL-Statements. Der höhere Speicherbedarf lässt sich gezielt entschärfen durch (a) geeignete Archivierungs-/Rearchivierungskonzepte, (b) Konsolidierung von Zahlenmaterial und (c) moderate Historisierung, welche Änderungen unkritischer Attribute mit Replace vollzieht.

Die Grafik veranschaulicht die Elemente einer konsequenten Historisierung am Beispiel einer Person und ihrer Adresse.



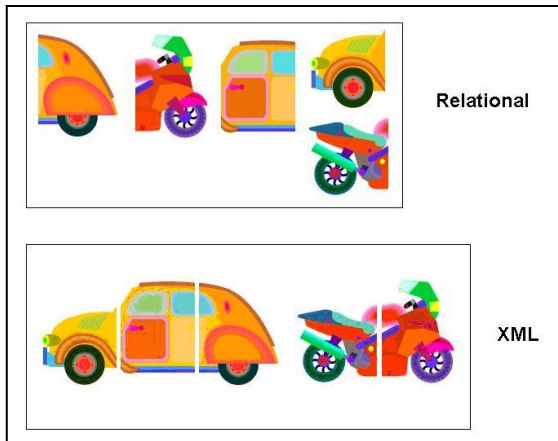
```

Select für Nutzung
SELECT * FROM Person AS P1, Adresse AS A1 WHERE A1.Adresse = P1.Adresse AND P1.GültigAb IN
(SELECT MAX(P2.GültigAb) FROM Person P2 WHERE P2.Person = P1.Person AND P2.GültigAb
<= Jetzt() AND (P2.GültigBis IS NULL OR P2.GültigBis >= Jetzt())) AND A1.GültigAb IN
(SELECT MAX(A2.GültigAb) FROM Adresse A2 WHERE A2.Adresse = P1.Adresse AND A2.GültigAb
<= Jetzt() AND (A2.GültigBis IS NULL OR A2.GültigBis >= Jetzt()));
    
```

**Select für Mutation**

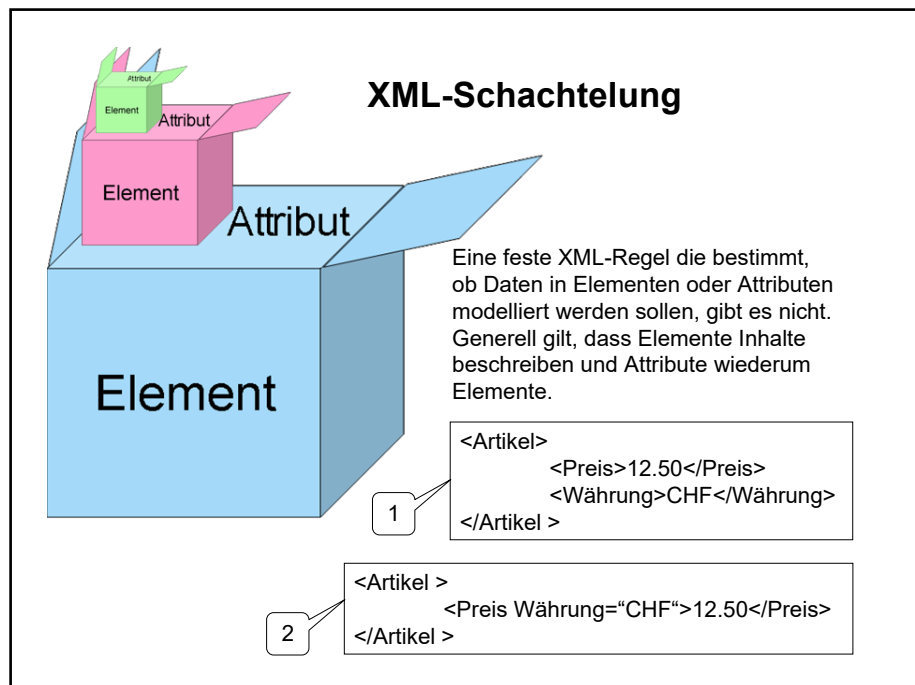
```
SELECT * FROM Person AS P1, Adresse AS A1 WHERE A1.Adresse = P1.Adresse AND P1.GültigAb IN
(SELECT MAX(P2.GültigAb) FROM Person P2 WHERE P2.Person = P1.Person AND A1.GültigAb IN
(SELECT MAX(A2.GültigAb) FROM Adresse A2 WHERE A2.Adresse = P1.Adresse AND (A2.GültigBis
IS NULL OR P1.GültigBis IS NULL OR A2.GültigBis >= P1.GültigBis)));
```

## 9 XML – Extensible Markup Language

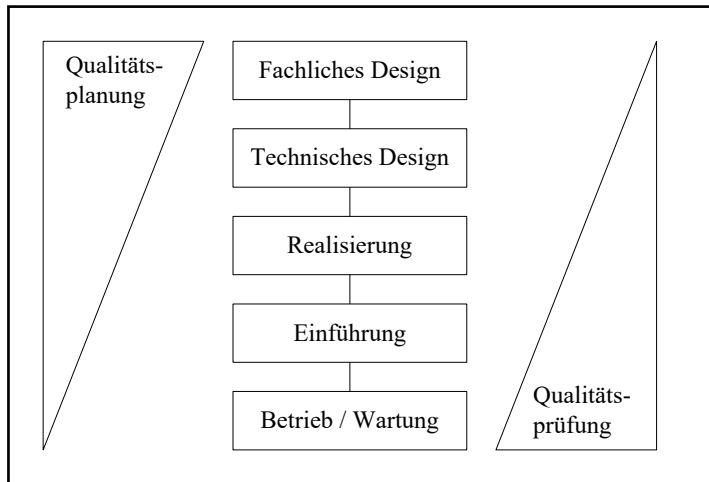


XML ist eine noch junge Technologie, die laufend weiterentwickelt (standardisiert) wird. Mittlerweile hat sie sich auch im Datenbank-Sektor etabliert.

Das nebenstehende Bild enthält die beiden möglichen Modellierungsarten anhand eines Beispiels. Für Datenbanken ist die Modellierung in Attribute (Währung in Beispiel 2) nicht empfehlenswert. MS Access z.B. importiert nur Elemente.



## 10 Qualitätssicherung



Qualitätsplanung soll im Vordergrund stehen und verhindern, dass Mängel erst bei der späteren Prüfung festgestellt, und mit entsprechenden Kosten behoben werden müssen. Erkenntnisse aus der Qualitätsprüfung müssen in die Planung einfließen.

## 11 Quellen- und Literaturverzeichnis

HMD90: Theorie und Praxis der Wirtschaftsinformatik: HMD Hefte 153/154, Forkel-Verlag, Wiesbaden, 1990..... 3

LEW88: Lewerentz,C.: Interaktives Entwerfen grosser Programmsysteme (Konzepte und Werkzeuge). Informatik-Fachberichte 194, Springer-Verlag, Berlin, 1988 ..... 4, 5

MÜL88: Müller,K.: Management für Ingenieure - Grundlagen, Technik, Instrumente. Springer-Verlag, Berlin, 1988 ..... 4

PIE90: Pieper,R., Richter,K.: Management - Bedingungen, Erfahrungen, Perspektiven. Verlag Die Wirtschaft, Berlin, 1990..... 4

SVB88: van Zijl,N., Lang,R., Burger,P., Sutter,H., Banthien,Ch.: Projektmanagement, Die Orientierung Nr.92. Schweizerische Volksbank, Bern, 1988 ..... 4

VET86: Vetter,M.: Aufbau betrieblicher Informationssysteme. Teubner, Stuttgart, 1986 ..... 4