

# ERP Enterprise Resource Planning

## Zusammenfassung Theorie

**Version** 1.1  
**Letzte Aktualisierung** 11.06.2010

**Autoren** Flurin Müller  
Nicolas Amstalden  
Dominik Lustenberger  
Michael Baumli

<b>1.</b>	<b>Einführung in ERP-Systeme</b>	<b>6</b>
<b>1.1.</b>	<b>Datenintegration.....</b>	<b>6</b>
1.1.1.	Homogene ERP-Systemlandschaft.....	6
1.1.2.	Datenbestände von ERP-Systemen .....	6
1.1.3.	Stammdatenpflege .....	6
<b>1.2.</b>	<b>ERP-Systeme und ihre Anwender .....</b>	<b>6</b>
1.2.1.	SAP .....	6
1.2.2.	Microsoft Dynamics NAV .....	7
1.2.3.	Aufbau von ERP-Systemen .....	7
1.2.4.	Branchenneutralität, Internationalität, Kunden .....	7
1.2.5.	SAP NetWeaver.....	7
<b>1.3.</b>	<b>Bestandteile von SAP .....</b>	<b>7</b>
1.3.1.	Unternehmensmodell .....	7
1.3.2.	Customizing .....	7
1.3.3.	Offenes System.....	8
1.3.4.	Skalierbarkeit.....	8
<b>1.4.</b>	<b>Was ist ein SAP-System?.....</b>	<b>8</b>
1.4.1.	Die SAP 3-Tier Architektur .....	8
1.4.2.	Die MS Dynamics NAV 2009, 3-Tier Architektur .....	8
1.4.3.	Die LUW bei SAP .....	8
<b>1.5.</b>	<b>Die betriebswirtschaftliche Architektur .....</b>	<b>9</b>
1.5.1.	Organisationsstruktur.....	9
1.5.2.	Die Sonnenschein AG / Beispiel für ein Dienstleistungsunternehmen .....	10
<b>1.6.</b>	<b>Organisation der ERP Software-Einführung .....</b>	<b>10</b>
1.6.1.	Anforderung an den ERP Projektleiter.....	10
1.6.2.	Die Einführung eines ERP Systems in einem Unternehmen .....	10
1.6.3.	Probleme bei der ERP Einführung.....	10
1.6.4.	Die Roadmap.....	11
<b>2.</b>	<b>Evaluation, Beurteilung und Kosten</b>	<b>11</b>
<b>2.1.</b>	<b>Selektionsproblematik: Vielfalt der Anbieter .....</b>	<b>11</b>
<b>2.2.</b>	<b>IT-Strategieentwicklung .....</b>	<b>11</b>
<b>2.3.</b>	<b>Industrialisierung der IT .....</b>	<b>11</b>
2.3.1.	Webservice .....	11

<b>2.4. Beurteilung.....</b>	<b>12</b>
2.4.1. Nutzwertanalyse .....	12
<b>2.5. Ziele eines ERP-Projektes .....</b>	<b>12</b>
2.5.1. Notwendigkeit eines Pflichtenhefts .....	12
<b>2.6. Ablauf eines ERP-Projekts.....</b>	<b>13</b>
<b>2.7. Exkurs Investitionsbeurteilung .....</b>	<b>13</b>
2.7.1. Payback-Methode .....	13
<b>2.8. Stellenorientiertes Ablaufdiagramm .....</b>	<b>13</b>
<b>2.9. Anforderungsanalyse Checkliste .....</b>	<b>14</b>
<b>2.10. Methoden zur Identifikation von Anforderungen .....</b>	<b>14</b>
2.10.1. Beobachtung .....	14
2.10.2. Dokumente analysieren .....	14
2.10.3. Interviews und Umfragen.....	14
2.10.4. Workshops .....	14
<b>2.11. Total Cost auf Ownership (TCO).....</b>	<b>14</b>
 <b>3. Verfügbarkeit von Systemlandschaften .....</b>	 <b>15</b>
<b>3.1. IT-Risikomanagement .....</b>	<b>15</b>
<b>3.2. Verfügbarkeitsrechnung.....</b>	<b>15</b>
3.2.1. Verfügbarkeit für in Serie geschaltete Systeme .....	16
3.2.2. Verfügbarkeit für in parallel geschaltete Systeme .....	16
3.2.3. Beispiel zur Verfügbarkeitsrechnung .....	16
<b>3.3. Merkmale für die die Administration fehlertoleranter Systeme .....</b>	<b>16</b>
<b>3.4. Fehler-Möglichkeiten und Einfluss-Analyse (FMEA).....</b>	<b>16</b>
<b>3.5. Service Level Agreement.....</b>	<b>16</b>
3.5.1. Verfügbarkeit einer Leistung .....	17
3.5.2. Messmethode .....	17
3.5.3. Antwortzeit (response time).....	17
3.5.4. Reaktionszeit .....	17
3.5.5. Sanktionen .....	17
3.5.6. Umfeldbedingungen .....	17
3.5.7. Flexibilität .....	17
3.5.8. Schritte zur Erarbeitung einer SLA .....	17

<b>4.</b>	<b>Modellierung und Darstellung von Geschäftsprozessen</b>	<b>18</b>
4.1.	Prozess .....	18
4.2.	Modell .....	18
4.3.	Geschäftsprozess modellierung .....	18
4.3.1.	Geschäftsprozess in der Praxis .....	18
4.4.	Standards und Darstellungsformen der Prozessmodellierung .....	19
4.4.1.	Klassische Ablaufdiagramme .....	19
4.4.2.	Ereignisgesteuerten Prozesskette .....	19
4.4.3.	Business Process Modelling Notation (BPMN) .....	20
4.5.	Prozessmuster .....	20
4.5.1.	Grundmuster .....	20
4.5.2.	Sequenz .....	20
4.5.3.	Parallele Verzweigung (AND) .....	20
4.5.4.	Exklusive Fallunterscheidung (XOR) .....	20
4.5.5.	Mehrfachwahl (OR) .....	21
4.5.6.	Weitere Muster .....	21
4.5.7.	Geschäftsprozessoptimierung .....	21
<b>5.</b>	<b>IT-Management und ITIL</b>	<b>22</b>
5.1.	Application Service Provider (ASP) .....	22
5.2.	Information Technology Infrastructure Library (ITIL) .....	22
5.2.1.	„Service Delivery“: .....	22
5.2.2.	„Service Support“: .....	22
<b>6.</b>	<b>Analytische Informationssysteme</b>	<b>23</b>
6.1.	Data Warehouse (DWH) .....	23
6.1.1.	Die DWH Architektur: .....	23
6.1.2.	Der Aufbau eines Business Informations (BI) Systems: .....	23
6.2.	Eine praktische Einführung in OLAP .....	24
6.3.	Data Mining .....	25
6.4.	Managementunterstützungssysteme MUS .....	25
6.5.	Abfrage- und Berichtssysteme .....	25

<b>6.6. Entscheidungsunterstützungssysteme EUS (Decision Support Systems, DSS) ..</b>	<b>25</b>
<b>6.7. Expertensystem XPS .....</b>	<b>25</b>
<b>6.8. Executive Information System .....</b>	<b>25</b>
<b>7. Buch „ERP für KMU“ .....</b>	<b>26</b>
<b>7.1. Grundlagen ERP .....</b>	<b>26</b>
7.1.1. Definition und Übersicht .....	26
7.1.2. Abgedeckte Prozesse .....	26
7.1.3. Wichtige Module, Funktionsumfang .....	26
7.1.4. Kostenstruktur ERP .....	26
7.1.5. Gründe für den ERP-Einsatz .....	26
<b>7.2. ERP-Auswahl .....</b>	<b>26</b>
7.2.1. Erwartungshaltung .....	26
7.2.2. Stolpersteine in ERP Projekten .....	26
7.2.3. Erfolgsfaktoren und Lösungsansätze .....	26
7.2.4. Vorgehen für Evaluation und Einführung .....	26
7.2.5. Vertragsabschluss .....	27
<b>7.3. Blick auf den ERP-Markt .....</b>	<b>27</b>
7.3.1. Standardsoftware .....	27
7.3.2. Branchensoftware .....	27
7.3.3. Individualsoftware .....	27
7.3.4. Open-Source-Software (OSS) .....	27
7.3.5. Integration in die Softwarelandschaft .....	27
7.3.6. SOA und Integration .....	27
7.3.7. IT extern betreiben? .....	27

# 1. Einführung in ERP-Systeme

Als Enterprise-Resource-Planning Systeme (ERP) werden **integrierte betriebswirtschaftliche Standardanwendungssoftware-Pakete** bezeichnet, die nahezu alle **Aufgabenbereiche und Prozesse innerhalb des Unternehmens unterstützen**, wie z.B. Beschaffung, Produktion, Vertrieb, Rechnungswesen und Personalwirtschaft.

Ziel eines ERP's (Enterprise Resource Planning) ist es eine **Konformität zwischen Geschäftsprozessen und Informatikprozessen** herzustellen.

ERP's unterstützen **unternehmensinterne Prozesse**, Supply Chain Management System sind für außerhalb der Unternehmung zuständig (Lieferanten).

## 1.1. Datenintegration

Viel Unternehmen benutzen unterschiedliche Programme, um ihre betrieblichen Aufgaben abzudecken  $\Rightarrow$  **Insellösungen**. Jedes Programm besitzt einen eigenen Datenpool.

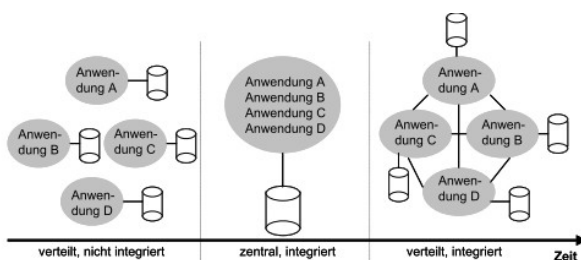
Probleme:

- **Übergreifende Nutzung** der gemeinsamen Daten
- **Bereitstellung von Daten** für einen anderen Arbeitsbereich.

Vorteil eines ERP: Nutzung eines gemeinsamen Datenpools (zentrale Datenbank)  $\Rightarrow$  **Datenintegration** / integrierte Datenstruktur genannt.

**Schnittstellen** werden somit überflüssig und es fallen keine „**doppelten Daten**“ an.

Möglichkeiten der **technischen Integration**:



(Abbildung: **verteilt, nicht integriert** / **zentral, integriert** / **verteilt, integriert**)

**Horizontale Integration:** Entlang der Wertschöpfungskette (Lieferant  $\Rightarrow$  Kunde)

**Vertikale Integration:** hierarchisch (Kostenstelle  $\Rightarrow$  Geschäftsleitung)

### 1.1.1. Homogene ERP-Systemlandschaft

Die homogene ERP-Systemlandschaft ist eine Fiktion. Ergibt im Kontext der Veränderungsdynamik wenig Sinn. Es kann durchaus sinnvoll sein, verschiedene Funktionen auf verschiedenen Systemkomponenten abzubilden.

### 1.1.2. Datenbestände von ERP-Systemen

Es werden zwei Typen von Daten unterschieden:

- **Stammdaten:** Daten, die längerfristig im System genutzt werden. z.B. Kunden- und Lieferantendaten, Materialdaten.
- **Bewegungsdaten:** Vorgangsbezogene Daten (temporär), die in Belegen gespeichert werden. z.B. Auftragsdaten, Daten in Buchungsbelegen.

### 1.1.3. Stammdatenpflege

Hauptgrund für ein schlechtes Systemverhalten sind **schlecht gepflegte Stammdaten**.

ERPs geben **keine Stammdatenpflegeprozesse** vor, deshalb sind **Workflow-Funktionen**

## 1.2. ERP-Systeme und ihre Anwender

Schwere **funktionale Defizite** gibt es bei den Spitzen-ERP-Systemen **kaum noch**. Heutzutage steht nicht mehr der Funktionsumfang im Vordergrund, sondern deren Eigenschaft/Fähigkeit, die Geschäftsprozesse der Anwender möglichst präzise und realitätskonform abzubilden.

Folgende **Kriterien** sind für die Bewertung eines ERP-Systems massgebend:

- **Dynamische Prozesse**  
Grundsätzlich kommen ERP-Systeme sehr gut mit **dynamischen Prozessen** (Veränderungen) zu recht, vielfach ist einfach das **Wissen des Anwenders zu klein**. Dies führt zu **zweigleisigem Arbeiten** und schlussendlich zu einer **schlechten Planung**.
- **Technologie**  
Programmiertechniken, Architektur, Eignung der Lösung für die globale Anwendung.

### • Funktionsumfang

### • Benutzerfreundlichkeit

### 1.2.1. SAP

SAP (**S**ysteme **A**nwendungen und **P**rodukte) ist der weltweit führende Anbieter von **betriebswirtschaftlicher Standardsoftware**.

Das SAP-System Deckt fast alle betriebswirtschaftlichen Anwendungsbereiche (**Rechnungswesen, Logistik, Personalwirtschaft**) ab.

**Funktionalität der Software:**

- Datenintegration
- Modularer Aufbau
- Internationalität
- Branchenneutralität
- Anpassungsfähig
- Realtime Verarbeitung
- riesiges Datenmodell.

### 1.2.2. Microsoft Dynamics NAV

Seit 2002 in Geschäftsbereich von Microsoft integriert. Vor allem von KMUs eingesetzt. Enthält Produktspezifische Programmiersprache: **C/AL**. Es gibt verschiedene **Branchenlösungen** (Handel, Dienstleistung, Verwaltung etc.)

**SIFT Technologie** (SumIndex Field Technology): Summe aus beliebig grossen Datenmengen aus lediglich **drei Datenbankzugriffen** bilden. Die aufnummerierten Werte werden mitgespeichert.

Datum	Betrag	SumIndex
2006-01-10	50	50
2006-02-01	100	150 (= 50 + 100)
2006-06-01	50	200 (= 150 + 50)
2006-12-12	200	400 (= 200 + 200)
2006-12-30	40	440 (= 400 + 40)

### 1.2.3. Aufbau von ERP-Systemen

Die Software ist in **verschiedene Module** aufgeteilt. Zum Teil sind diese Module voneinander **abhängig**. Der Benutzer merkt dies nicht, da er **prozessorientiert** und somit **modulübergreifend** arbeitet.

Beispiele Module:

- Vertrieb (**SD** - Sales and Distribution)
- Materialwirtschaft (**MM** - Material Management)
- Qualitätsmanagement (**QM** - Quality Management)
- Finanzwesen (**FI** - Finance)
- Personalwirtschaft (**HR** - Human Resources)
- Branchenlösungen (**IS** - Industry Solutions): Zusätzliche branchenspezifische Funktionen, welche vormodelliert ausgeliefert werden (Banken, Öl und Gas, Handel,...).

### 1.2.4. Branchenneutralität, Internationalität, Kunden

Multinationale Konzerne können ein **einheitliches System** **länderübergreifend** nutzen.

**Möglichkeit des internationalen Einsatzes:**

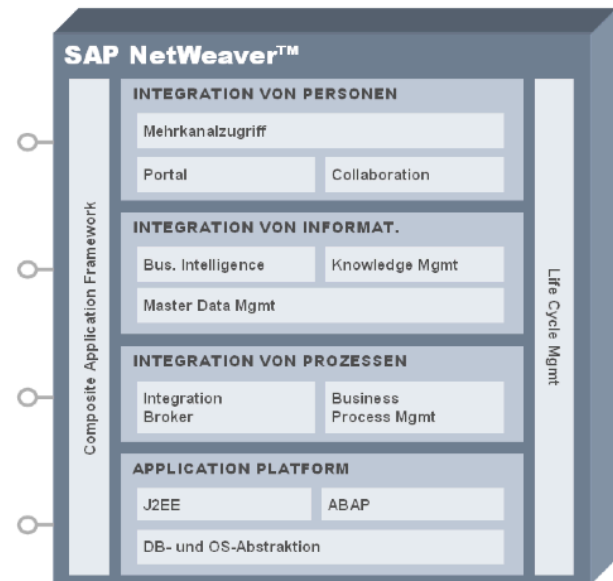
- Unterschiedliche Sprachen / Datumsformate
- Unterschiedliche Kontenpläne
- Länderspezifisches Verfahren zur Lohn-/Gehaltsberechnung
- Weltweite Planung und Abwicklung von Geschäften.

### 1.2.5. SAP NetWeaver

Plattform für Geschäftsanwendungen (SOA), Grundlage ist der **SAP NetWeaver Application Server**. Somit können auch **Nicht-SAP-Systeme** angeschlossen werden.

- Hat zahlreiche **Komponenten** und ist **flexibel** zu gestalten.
- Anwendungen können in erster Linie durch **Modellieren** anstatt durch **Programmieren** erstellt werden.
- **Life-Cycle-Management**: liefert die nötige Technologie (Installation, Upgrades, Lizenzen, Customizing,...) für den gesamten Lebenszyklus der Anwendung.

- **Master Data Management**: effiziente Verwaltung von Stammdaten: Stammdaten werden über Netzwerke gesucht und identische Daten identifiziert.
- **Multi-Channel Access** (Mehrkanalzugriff): mobiler Zugriff (Handheld etc.) auf SAP



## 1.3. Bestandteile von SAP

### 1.3.1. Unternehmensmodell

Die „**innere Logik**“ von SAP repräsentiert das Unternehmensmodell (ERP - **Entity Relationship Management**), welches sich nicht verändern lässt. Somit ist eine Anpassung des Prozessmanagements bis zu einem gewissen Punkt der Software nötig. Standardsoftware ist **keine individuell** für ein Unternehmen **massgeschneiderte Unterstützung**, sondern:

- **vereinheitlicht Abläufe**.
- beinhaltet **variierbaren Bausteinen**
- ist weit verbreitet und besitzt **standardisierte Schnittstellen**.
- wird gesetzlich und anderen üblichen **Rahmenbedingungen angepasst**.

Trend Richtung **Standardsoftware** anstatt **Individuallösungen**, weil:

- kostengünstig (Entwicklungskosten verteilen sich)
- Zeitersparnis (Anpassung)
- Mangel an Personal mit nötigem Know-how ist nicht mehr so tragisch
- Zukunftssicherheit (ständige Weiterentwicklung der Software)

### 1.3.2. Customizing

**Customizing**: Zuschneiden der Software auf individuelle Bedürfnisse und Abläufe. Es werden **unternehmensspezifische Einstellungen** gemacht. Im

**laufenden Betrieb** können **keine Customizing** Einstellungen verändert werden.

### 1.3.3. Offenes System

Aufgrund des **hardwareunabhängigen Systemdesigns** ist SAP auf vielen unterschiedlichen Rechnern lauffähig. Neben der **SAP-Präsentationssoftware** (Client-SW) kann auch mit dem **Browser** oder **Mobilen Geräten** auf SAP zugegriffen werden.

### 1.3.4. Skalierbarkeit

Unter Skalierbarkeit versteht man die **leichte Anpassbarkeit der installierten Rechnerleistung bei geänderten Lasten**, z.B. Bei steigender Benutzerzahl oder Einsatz zusätzlicher Anwendungen. SAP ist diesbezüglich **stufenlos erweiterbar**.

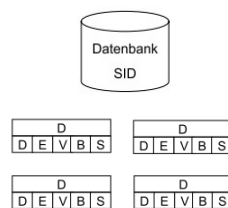
## 1.4. Was ist ein SAP-System?

Ein SAP-System besteht aus **einer Datenbank** und **einer oder mehreren Instanzen**.

**Zentrales System:** System mit einer einzigen Instanz und dies gemeinsam mit der Datenbank auf einem Rechner läuft.

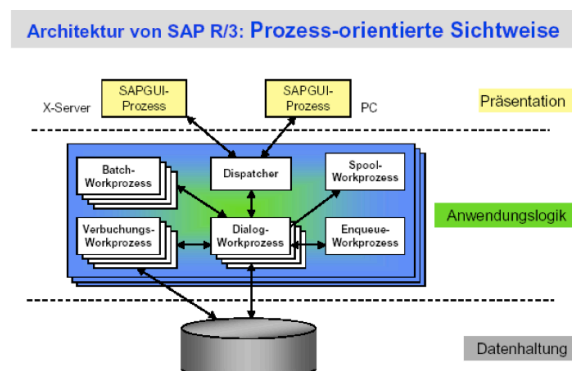
Eine **Instanz** besteht aus:

- Dispatcher
- Dialog- Workprozesse
- Enque- Workprozesse
- Verbuchungs- Workprozesse
- Batch- Workprozesse
- Spool- Workprozesse



### 1.4.1. Die SAP 3-Tier Architektur

Ein SAP-System besteht aus einer 3-Tier Architektur: **Präsentation** (reines Frontend, grafische Bedienoberfläche), **Anwendungslogik** (Application Server, entlastet den Datenbankserver indem er Daten zwischenspeichert), **Datenhaltung** (DBMS, Speicherung, Aufbereitung und Bereitstellung der Daten)



**Vorteile der Client/Server Architektur:**

- Kosteneinsparungen (Wartung)
- Verteilung der Rechnerleistung
- erhöhte Verfügbarkeit (Clustering)
- Skalierbarkeit kann Lastspitzen ausgleichen

**Dispatcher:** koordiniert und verteilt die Useranfragen auf die Work-Prozesse im Applikationsserver (FCFS - First Come First Server)

Weitere **Work-Prozesse:**

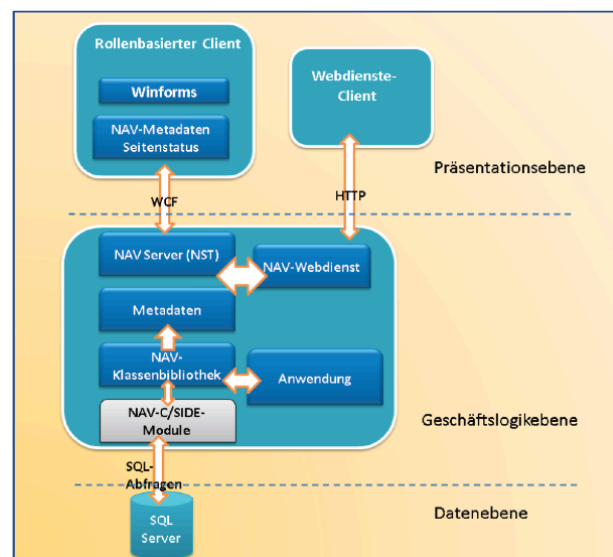
- **Enqueue Workprozess** (Sperrung von Business-objekten, wie Debitor, Kreditor) ⇒ Ermöglicht das konfliktfreie Arbeiten mit parallelen Usern.
- **Verbuchungs Workprozess** (Update in der Datenbank)
- **Batch Workprozess** (Hintergrundverarbeitung für rechenintensive Jobs)
- **Spool Workprozess** (Ausgabe: Drucker, Fax).

### 1.4.2. Die MS Dynamics NAV 2009, 3-Tier Architektur

Microsoft Dynamics NAV 2009 enthält sowohl ein zweistufiges als auch ein dreistufiges Modell:

- **Rollenbasierter Client:** Clientbenutzeroberfläche, von der das Front-End bereitgestellt wird. Daten werden auf Seiten angezeigt, welche im Page Designer entworfen werden können.
- **Microsoft Dynamics NAV Server:** Middle-Tier-Server, von dem die Geschäftslogik bereitgestellt wird.
- **SQL-Server:** Datenbankserver, von dem die Datenebene bereitgestellt wird.

Der NAV Server bietet ausserdem eine zusätzliche **Sicherheitsebene zwischen den Clients und der Datenbank** (mithilfe des Authentifizierungsfeatures von *Windows Communication Framework*).



### 1.4.3. Die LUW bei SAP

Eine LUW ist die **Zeitspanne bis eine betriebswirtschaftliche Buchung** (über mehrere Dialogschritte hinweg) **vollzogen ist**. Während dieser Zeit sind die betriebswirtschaftlichen Objekte (Material, Debitoren,...) für andere User gesperrt. ⇒ **Rollback** sofern bei der Verbuchung ein Laufzeitfehler auftritt.



Eine **SAP-Transaktion** ist seine Sequenz betriebswirtschaftlich logisch zusammengehörender Dialog-Schritten

Eine **SAP-Verbuchung** ist die dauerhafte Verbuchung in BWL-Tabellen  $\Rightarrow$  Daten aus den vorbereiteten Schritten der SAP-Transaktion benötigt.

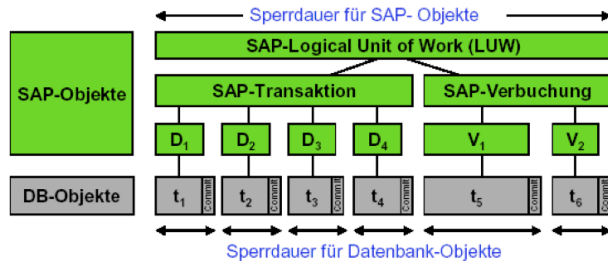


Abbildung 18: Quelle SAP; Logical Unit of Work

(Legende: **D**: Dialogschritte, **V**: Verbuchungsschritte, **t**: DB-Transaktionen)

**SAP-Objekte** werden für die Dauer der LUW gesperrt. Sperrdauer: Lang.

**DB-Objekte** (Tabellen) nur während der DB-Transaktion. Sperrdauer: kurz.

Datenmanipulationen aus den einzelnen Dialogschritten werden in so genannten **Verbuchungstabellen zwischenspeichert**. Erst nach dem letzten Schritt erfolgt die Übertragung. Bei einem Laufzeitfehler wird jetzt nur der Datensatz in der Verbuchungstabelle gelöscht. Die **Sperrungen** werden bei jedem **COMMIT** und **ROLLBACK aufgehoben**. Die Sperrtabelle befindet sich auf dem **Applikationsserver**.

## 1.5. Die betriebswirtschaftliche Architektur

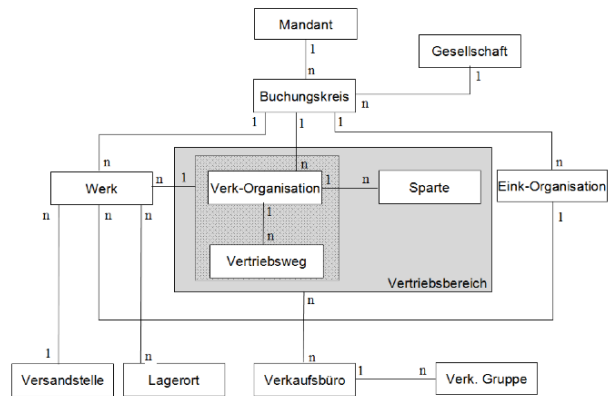
### 1.5.1. Organisationsstruktur

Jede Unternehmensstruktur kann mit Hilfe der SAP-Organisationseinheit abgebildet werden. Unter **Organisationseinheiten** werden die einzelnen Teilbereiche des Unternehmens und deren betriebliche Funktion verstanden.

- Ein **Mandant** repräsentiert eine für sich handelsrechtlich, organisatorisch und **datentechnisch abgeschlossene Einheit** innerhalb eines Systems (getrennten Stammsätzen, eigener Satz von Tabellen). Auch **Konzern** in BWL.
- Ein **Buchungskreis** ist eine **selbstständig bilanzierende Einheit** (z.B. Firma innerhalb einer Konzerns). Auf Buchungskreisebene werden die vom Gesetzgeber geforderte Bilanz sowie die Gewinn- und Verlustrechnung (GuV) erstellt.
  - Der **Kontenplan** ist eine **strukturierte Liste aller Hauptbuch-Konten** (kann einem oder mehreren Buchungskreisen zugeordnet sein).
  - Der **Kostenrechnungsbereich** ist die **organisatorische Einheit innerhalb eines Unterneh-**

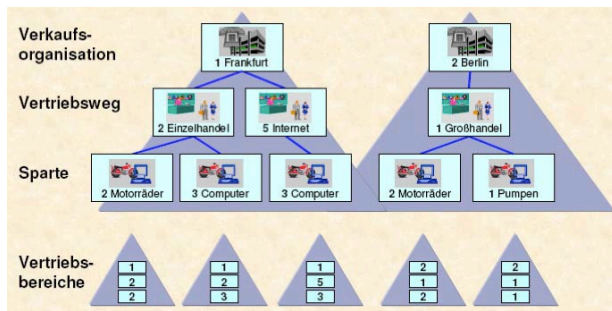
**mens**, für die eine vollständige, **in sich geschlossene Kostenrechnung durchgeführt werden kann** (kann mehrere operierende Buchungskreise umfassen, diese müssen aber den selben Kontenplan nutzen).

- **Ergebnisbereich** stellt einen Teil des Konzerns dar, für den eine einheitliche Segmentierung des Absatzmarktes vorliegt.



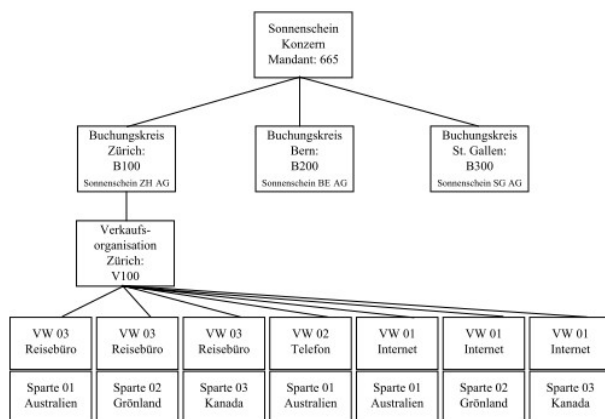
(Abbildung: Organisationseinheiten in der Logistik/Vertrieb/Materialwirtschaft)

- Ein **Werk** stellt Güter her (**Produktionswerk**), bietet Dienstleistungen (**Werksverkauf**) an oder stellt Güter für die Verteilung (**Auslieferungswerk**) bereit. Ein Werk kann mehrere Lagerorte enthalten.
- Ein **Lagerort** ist für die Bestandsführung vorgesehen.
- Die **Einkaufsorganisation** beschafft Materialien oder Dienstleistungen für ein oder mehrere Werke beschafft und allgemeine Einkaufskonditionen mit den Lieferanten aushandelt.
- Die **Verkaufsorganisation** ist dem Buchungskreis eindeutig zugeordnet. Sie stellt die **Organisation im Betrieb** dar. Sie ist verantwortlich für den Vertrieb von Gütern und Dienstleistungen sowie die Verkaufsverhandlungen und führt betriebswirtschaftliche Transaktionen durch.
- Unter einer **Sparte** versteht man eine **Gruppe von Produkten**, die von einer Verkaufsorganisation vertrieben werden können.
- Ein **Vertriebsweg** ist der Weg, auf dem Waren oder Dienstleistungen zum Kunden gelangen (z.B. Einzelhandelsverkauf, Großhandelsverkauf, Direktverkauf, Werksverkauf, ...).
- Der **Vertriebsbereich** besteht aus Verkaufsorganisation, Vertriebsweg und Sparte. Damit ist es möglich, festzustellen, welches Produkt / Dienstleistung auf einem Vertriebsweg verkauft werden können. Innerhalb von Vertriebsbereichen können **Auswertungen** durchgeführt werden. Sämtliche **vertriebsrelevanten Daten** können je Vertriebsbereich definiert werden.



- Ein **Verkaufsbüro** ist für den Verkauf innerhalb eines geografischen Gebiets zuständig. Die Definition ist optional und ist im Sinne einer Niederlassung zu verstehen.
- Die **Verkäufergruppe** bezeichnet die personelle Besetzung eines Verkaufsbüros.
- An der **Versandstelle** findet die Versandaktivität statt. Jede Lieferung wird von einer Versandstelle bearbeitet.

### 1.5.2. Die Sonnenschein AG / Beispiel für ein Dienstleistungsunternehmen



- **Buchungskreise** weisen Ende Jahr alle eine eigene Bilanz und ER aus.
- Es werden Reisen verkauft, also brauchen wir noch eine **Verkaufsorganisation**.
- Wir brauchen **kein Werk**, da es sich nur um Dienstleistungen handelt.
- **Vertriebsbereiche** werden anhand der Sparten (Australien, Grönland und Kanada) und der Vertriebswege (Internet, Telefon, Reisebüro) definiert.
- Auf ein **Verkaufsbüro** wird **verzichtet**, da die Verkaufsorganisation schon die geografische Unterteilung führt.

## 1.6. Organisation der ERP Software-Einführung

Die **Top-5 kritischen Erfolgsfaktoren** bei der Einführung sind (gem. empirischer Untersuchung von Somers und Nelson):

1. Top-Management Unterstützung,
2. Qualifikation des Projektteams

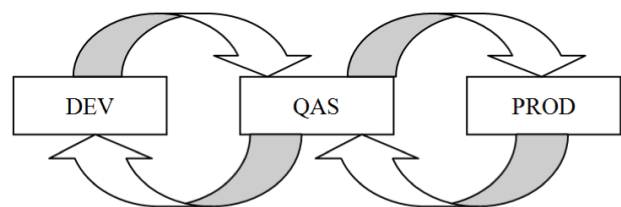
3. Abteilungsübergreifende Kooperation
4. Klare Projektziele
5. Projektmanagement

### 1.6.1. Anforderung an den ERP Projektleiter

Die **Erfahrung** eines Projektleiters bei der Einführung von ERP-Systemen unabdingbar. **Branchenkenntnisse** und sehr gute **Kenntnisse des jeweiligen ERP-Systems** sind nötig. Blosser zeitliche Überwachung, mitlaufender Kostenüberwachung und Ressourceneinsatzplanung ist nicht ausreichend.

### 1.6.2. Die Einführung eines ERP Systems in einem Unternehmen

Da ein ERP-System i.d.R. in einem nicht angepassten Standardformat ausgeliefert wird, müssen weitreichende **Customizing-Arbeiten** (Einarbeitung der Kostenstellen, Übernahme von Daten aus den Altsystemen) vorgenommen werden. Aus diesem Grund empfiehlt es sich, drei unabhängige Systeme aufzubauen:



- **DEV (Development System)**

An diesem System werden alle Customizing Einstellungen gemacht. Danach wird die SW in das QAS System transportiert.

- **QAS (Quality assurance System)**

An diesem System werden alle Softwarequalitäts-tests durchgeführt. Sobald diese Testphase vorbei ist, wird die SW in das produktive System transportiert.

- **PROD (Production System)**

Nach dem „Go live“ wird mit dem produktiven System gearbeitet und DEV und QAS sind bereit für Updates oder Customizing Änderungen usw.

### 1.6.3. Probleme bei der ERP Einführung

- Wechselnde Verantwortlichkeiten
- Fehlende Integration und unzureichende Kommunikation
- Schlechte Zeit-, Ressourcenplanung und Koordination
- Zu starke Orientierung am Altsystem
- Schlechte Dokumentation
- Eigene Geschäftsprozesse unklar
- Einführungsteam zu gross / klein

### 1.6.4. Die Roadmap

Einführungsleitfaden für die Projektleitung und die Mitglieder des Projektteams. Sie führt zielsicher durch das Projekt.



**Phase 1:** Projektvorbereitung (Planung, Vorbereitung)

**Phase 2:** Business Blueprint, Beschreibung der Ergebnisse aus den Workshops zu den Anforderungen der Geschäftsprozesse.

**Phase 3:** Realisierung, Konfiguration der Geschäfts- und Prozessanforderungen des BP im R/3 System.

**Phase 4:** Produktionsvorbereitung, Tests, Schulung, Datenübernahme

**Phase 5:** Go-Live und Support, Wechsel Testsystem - Livesystem

⇒ Glossar Skript Seite 49 - 53

## 2. Evaluation, Beurteilung und Kosten

**Auslöser** die Einführung eines neuen Systems:

1. Veraltetes System
2. Geänderte Anforderungen / Prozesse
3. Veränderung Unternehmensstruktur

**Ziele** bei der Einführung von Business-Software:

1. Prozesse beschleunigen
2. Schneller Zugriff auf Informationen
3. Bessere Informationen

**Projektherausforderungen**

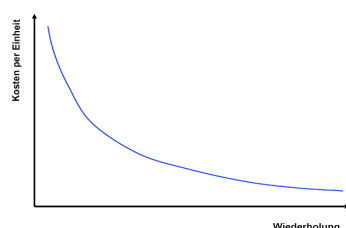
1. Datenaufbereitung, Datenmigration
2. Knapper Zeitplan
3. Zu viele Systemanpassungen

### 2.1. Selektionsproblematik: Vielfalt der Anbieter

Um den „Lieferantenwildwuchs“ zu strukturieren, gruppieren sich die Lieferanten zu **Lieferantengruppierungen** (HW-Hersteller, Softwarehäuser, Systemhäuser, ...)

**Merkmale des Marktes:** Rasche Entwicklung, Konsolidierungstendenzen, Relativ unübersichtlich

(Abbildung: Ein Geschäftsprozess im ERP-System rentiert dann, wenn er eine hohe Wiederholungsrate hat)



## 2.2. IT-Strategieentwicklung

Die Flexibilität des ERP-Systems entscheidet oft über den Ruf der IT-Service-Abteilung.

**Hauptgrund:** Ein Business-Prozess wandelt sich stetig ⇒ **Notlösung:** Systeme werden mit Funktionalität aufgeblasen ⇒ **Resultat:** Die daraus Resultierende Komplexität verursacht hohe IT-Unterhaltungskosten (heute 70% der IT-Ausgaben), grosser Schulungsaufwand, Qualitätsprobleme ⇒ **mangelnde Produktivität**

Problematik der unterschiedlichen Geschwindigkeit:

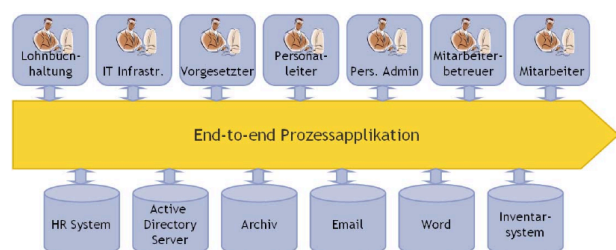
Geschäftsstrategie 3-5 Jahre
Geschäftsprozesse Wochen, Monate, wenige Jahre
Informations und Kommunikationssysteme 10-15 Jahre

## 2.3. Industrialisierung der IT

Die Entwicklung ging von **Individual-** über **Standardsoftware** bis heute zu **flexibelkomponentenbasierter** Software.

Die Industrialisierung in Hardware/Software brachte im Allgemeinen eine **Kostensenkung** mit sich. Problematik bei ERP's ist, dass sie sich nicht gut **standardisieren** lassen. Denn mehr als 50% aller Prozesse sind **nicht identisch**. Ein Standard müsste **einfach, plattform- und herstellerunabhängig** sein und von **Seiten Lieferant** garantiert werden.

Optimal für ERP's ist die **Integration** verschiedenster Systeme (EAI), häufig über einen **Enterprise Bus** realisiert.



### 2.3.1. Webservice

Ein Webservice ist eine **Software.Anwendung, die irgendwo im Web läuft** (z.B. Google-Suche, Wetterauskunft, Börsenkurse, ...). Das revolutionäre an Ihnen ist, dass er nicht nur von einer **Person** genutzt werden kann, sondern auch durch **Maschinen** ⇒ Standards machen es möglich (TCP/IP, HTTP, XML).

## 2.4. Beurteilung

1. **Grobevaluation** (2-5 Marktführer, 1-3 NewComer, 2-5 Branchenführer auf der Longlist).
  - Sichtung des geeigneten Angebots
  - Elimination anhand von Killer-Kriterien (Grob-Pflichtenheft)
2. **Feinevaluation** (Favoritisierte Systeme, ca. 6)
  - Vorstellung der Anbieter (ca. 1/2 Tag)
  - Telefonische Überprüfung der Referenzen
  - Überprüfung der Funktionalen Anforderungen
3. **Endevaluation** (Endrunde, ca. 2)
  - Aufstellung des Prozessdrehbuchs (Kritische Prozesse)
  - Durchführung von Workshops mit Anbieter (ca. 2-3 Tage)
  - Referenzbesuche
4. **Vertragsverhandlungen** (Ausgewähltes System)
  - Umstellung vom Pflichtenheft auf Lastenheft (Integration aller bisherigen Erkenntnisse)
  - Verhandlung der Zahlungskonditionen und Abnahmevergang

### 2.4.1. Nutzwertanalyse

Anhand verschiedener **Beurteilungskriterien**, werden **Produkte** einander **gegenübergestellt**. Diese werden dann **entsprechend der Wichtigkeit gewichtet** und summiert  $\Rightarrow$  Je grösser die **Summe** desto besser das geeignete Produkt.

- **Kriterium Funktionalität**  
Da die Bedürfnisse der jeweiligen Unternehmen von den meisten Anbietern abgedeckt werden, ist dieses Kriterium nicht so wichtig. Es ist ausserdem schwierig, Funktionalität objektiv zu messen.
- **Kriterium Zukunftssicherheit**  
Die grosse Anzahl der Anbieter wird langfristig keinen Bestand haben. Folglich ist dieses Kriterium sehr wichtig.  $\Rightarrow$  **Investitionsrechnung** deutlich schlechter wenn **Nutzungsdauer** kürzer.  
Ein weiterer Aspekt dabei ist die Einschätzung der zukünftigen Anforderungen.  
**Messwerte** für die **Zukunftssicherheit** sind: Firmengrösse (Anzahl Mitarbeiter, Umsatz), Gewinn, EK-Quote.
- **Kriterium Preis**  
Oft wird beim **Preis** von den **Investitionskosten** gesprochen. Nicht zu vernachlässigen sind jedoch auch die Unterhaltskosten. Gewisse Anbieter verlangen einen gewissen **Prozentsatz der Softwarekosten für Updates**. Zum Vergleich können die zukünftigen Kosten auf heute diskontiert und zu den Investitionskosten addiert werden (nicht bei der Grobevaluation).

## 2.5. Ziele eines ERP-Projektes

Bei der Zielformulierung ist es wichtig, **messbare Ziele** zu formulieren:

- **Kostenziele** (Budget)
  - Einführungskosten von Fr. ...
  - Unterhaltskosten von Fr. ...
  - Kosteneinsparungen von Fr. ...
- **Organisatorische Ziele**
  - Reduktion manueller Abläufe
  - Optimierung der Prozesse
  - Beschleunigung der Durchlaufzeit
  - Sicherstellung der Wartung
- **Terminziele**
  - Evaluation 6 Monate
  - Einführung 9 Monate
  - Tests & Schulung 3 Monate
  - Gesamtdauer 18 Monate

### Investitionsbeurteilung

- Die finanziellen Auswirkungen einer ERP-Beschaffung sind wichtig. Sie sollten **vor Abschluss der Feinevaluation** mittels einer **Investitionsrechnung** ermittelt werden.
- Die einfachste Methode dafür ist die Berechnung der erwarteten **Gewinn Veränderung** (Nutzen - Unterhalt - Abschreibungen).

### Nutzen

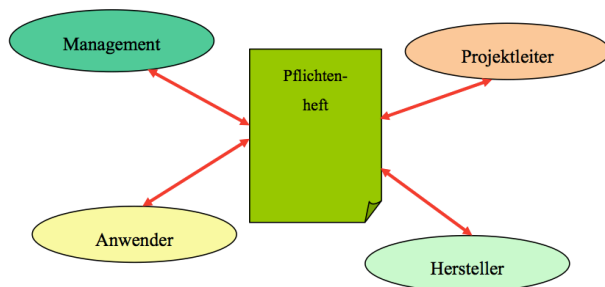
- kann in **quantitativ** und **qualitativer Aspekte** aufgeteilt werden.
- nur **quantitative Aspekte** können in die **Investitionsrechnung** aufgenommen werden
- Quantifizieren lässt sich z.B. **Effizienzgewinn bei der Kundenauftragsabwicklung, geringeren Unterhalt, Unterhalt des Altsystems fällt weg, Unterhalt des neuen System fällt an**
- Schwieriger zu quantifizieren sind z.B. **grössere Ausfallsicherheit, besserer Kundendienst, bessere Informationen** oder **kürzere Lieferfrist**

### 2.5.1. Notwendigkeit eines Pflichtenhefts

Die Erstellung eines Pflichtenhefts wird in der Praxis oft vernachlässigt. Bekannte Thesen dafür sind:

- Das Pflichtenheft ist zu **aufwendig** und zu **teuer**, die Zeit reicht nicht aus.  
 $\Rightarrow$  Spätestens vor der Einführung erstellt man auf alle Fälle ein Konzept oder einen Anforderungskatalog.
- Man wählt am besten den **bekanntesten Anbieter**, dann geht schon nichts schief.  
 $\Rightarrow$  Es gibt genügend gegenteilige Praxis-Beispiele jedes Lieferanten.
- Es gibt genügend **schlüsselfertige Standardsoftware**.  
 $\Rightarrow$  Dann kann man ja den Funktionskatalog für das Pflichtenheft verwenden.

Das Pflichtenheft ist ein **Vorvertrag**!

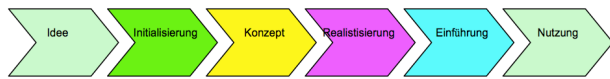


(Abbildung: Pflichtenheft und die Anspruchsgruppen)

## 2.6. Ablauf eines ERP-Projekts

Ein ERP-Projekt läuft gleich wie andere Projekte ab. Wichtig sind die **Meilensteine** und dass sie mittels Audit zu einem Qualitätschritt werden.

Ein **neutrales review Team** sollte ausserdem die Meilensteine mit *pass* oder *fail* bewerten.  $\Rightarrow$  wenn nicht bestanden, nachbessern.



### Hauptprobleme während des Betriebs:

1. Mangelnde Flexibilität der Software
2. Aufwand zur Implementierung von Upgrades / Neuerungen
3. Mangelnde Bedienerfreundlichkeit

## 2.7. Exkurs Investitionsbeurteilung

### 2.7.1. Payback-Methode

Die Payback-Methode liefert Hinweise auf **die mit dem Investitionsvorhaben verbundenen Risiken**. Man geht von der Überlegung aus, dass das Risiko umso geringer ist, je schneller das Unternehmen die investierten Mittel zurückerhält.

Investitionskapital	195'000
Unterhalt pro Jahr	22'000
Einsparungen pro Jahr	78'000

Jahr	0	1	2	3	4	5
Investition	-195'000					
Kosten		-22'000	-22'000	-22'000	-22'000	-22'000
Nutzen		78'000	78'000	78'000	78'000	78'000
Gewinn		56'000	56'000	56'000	56'000	56'000

$\Rightarrow$  Payback-Dauer 3.5 Jahre

**Formel zur Berechnung der Investitionskosten**  
(von Thomas Widmer, ETH Zürich):

$$\text{Ungenaue Investitionskosten} = \text{Benutzerzahl} \times (5'000 \text{ bis } 30'000) \text{ Fr.}$$

Wichtige Kostenfaktoren zur Berechnung der Bandbreite sind:

- Komplexität
- Spezielle Funktionen
- Schnittstellen
- Fehlende interne Ressourcen
- Fehlende Erfahrung des Anbieters
- Datenübernahme des Altsystems

Die folgende Kostenschätzungsformel ist genauer, weil die Anzahl user und die Komplexität des Systems berücksichtigt wird:

$$\begin{aligned} \text{Erwartete Gesamtkosten} &= 160'000 + 9500 * \text{Benutzerzahl} + 56'000 * \text{Teamgrösse} \\ \text{Erwartete Investitionen} &= 160'000 + 5000 * \text{Benutzerzahl} + 34'000 * \text{Teamgrösse} \end{aligned}$$

### Tipps für ERP-Anwender:

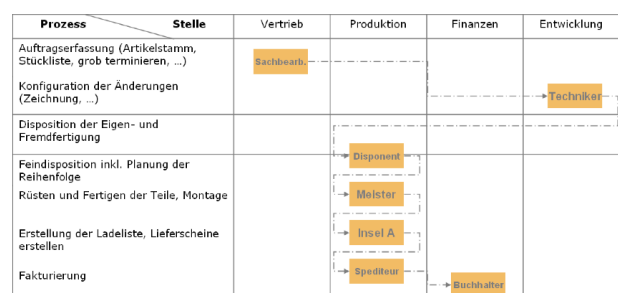
- **Dienstleistungskosten** lassen sich von allen Kostenarten am besten anpassen (z.B. beim Verzicht auf Anpassungen)
- Bei KMUs kennen sich oft Projekt- / Teamleiter nicht genügend im ERP-Markt aus.  $\Rightarrow$  In die Beratung während der **Evaluationsphase** investieren (nur 26% der beobachteten Unternehmen hat einen **externen Berater** dazu gezogen)

### Tipps für ERP-Hersteller:

- **Wachstumsstrategie** für kleine Hersteller selten eine Strategie  $\Rightarrow$  da direkte Konfrontation mit den Branchenführern. **Kundennähe** und **Spezialisierung** (Regional, Branche) wichtiger!
- **Schulungskosten** machen einen **wesentlichen Teil der Dienstleistungskosten** aus  $\Rightarrow$  besser in **Ergonomie der Software** investieren anstatt in zusätzliche Funktionen.

## 2.8. Stellenorientiertes Ablaufdiagramm

- Ein Stellenorientiertes Ablaufdiagramm hilft ein **Prozess einfach darzustellen**.
- Dabei sind **viele Übergänge** (Stellenwechsel) und Medienbrüche **schädlich**.
- Für die Erstellung des Diagramms sind **Bearbeitungszeiten, Liegezeiten und Transportzeiten** zu messen.



$\Rightarrow$  Aufgabe S. 26 / 27 Skript



## 2.9. Anforderungsanalyse Checkliste

- **Strategische-Sicht: WARUM?**  
Wertschöpfungskette, Organisation, Bedeutung, Ziel
- **Fachliche-Sicht: WAS?**  
Prozesse + Geschäftsfälle, Funktionen, Informationen. Geschäftsregeln
- **Technische-Sicht: WIE?**  
Schnittstellen, Plattformen + Basissoftware, Wartung + Betrieb, Security
- **Umsetzungs-Sicht: WOMIT?**  
Zeit, Know-How / Kapazität, Risiken, Infrastruktur / Finanzen

## 2.10. Methoden zur Identifikation von Anforderungen

Zur Identifikation der Systemanforderungen eignen sich die folgenden Methoden:

### 2.10.1. Beobachtung

- **Wie?**
  - Rundgang durch das Unternehmen
  - Sacharbeiter bei der Auftragseingabe beobachten
  - Produktion beobachten
- **Nachteile**
  - Aussenstehenden fallen wichtige Details oft nicht auf.
  - Die aufwändigen Ausnahmefälle treten in der Beobachtungszeit nicht auf.

### 2.10.2. Dokumente analysieren

- **Vorteile**
  - geringer Aufwand bei der Datenerhebung
  - keine Notwendigkeit für weitere Dokumentationen wie z. B. bei einem Interview,
  - Untersuchungsanlass verfälscht die Daten nicht
  - Betriebsablauf bleibt ungestört
- **Nachteile**
  - Oft veraltet
  - Anderer Fokus
  - Daten lassen Spielraum für Interpretationen
  - Unterlagen sind möglicherweise nicht vollständig oder für die Untersuchung nicht passend

### 2.10.3. Interviews und Umfragen

- **Vorbereitung**
  - Leitfaden für Interviews
  - Getesteter Fragebogen und Auswertungskonzept für Umfragen
  - Experteninterview
- **Resultat**
  - Persönliche, gegenteilige Meinungen.
  - Mitarbeiter fühlen sich ernst genommen.

### Vorteile

- Ermöglicht Diskussion
- Gemeinsame Lösungsfindung

### Nachteile

- Aufwändig

### 2.10.4. Workshops

- Gute Moderation wichtig
- Aufwändig

In der Praxis finden diese Methoden viel zu Wenig Einsatz. Es ist jedoch **wichtig**, die Phase der Systemanforderungen zu planen.

## 2.11. Total Cost auf Ownership (TCO)

- **Abrechnungsverfahren** um alle **anfallenden Kosten von Investitionsgütern** (z.B. Software, Hardware) **abzuschätzen**.
- Die Idee ist es, eine Abrechnung zu erhalten, welche nicht nur die **Anschaffungskosten**, sondern auch die **Kosten der späteren Nutzung** (Reparatur, Wartung) berücksichtigt.
- Wichtig beim Verfahren ist die Unterscheidung von **direkten** und **indirekten Kosten**.

Kostenkategorie	Kostenart	Beispiel
Indirekte Kosten ca. 40%	Datenmanagement	Kosten, die durch die vom Anwender benötigte Zeit für Backup, Recovery, Datenorganisation entstehen.
	Entwicklungen durch Anwender	Z.B. Makroprogrammierung oder Skripting etc. durch Anwender.
	Formales Lernen	Produktiver Ausfall
	Lernen am Fallbeispiel	Z.B. Lesen von Online-Hilfen und Suchen im Internet
	Fun Factor	Z.B. Software-Spielerien wie z.B. Gestaltung von Oberflächen, Extra Einstellungen etc.
	Unterstützung von Kollegen/innen	Produktivitätsausfälle durch unterstützen von Kolleginnen und Kollegen.
	Support	Produktivitätsausfall durch Beanspruchung des Help Desk
Direkte Kosten ca. 60%	Helpdesk	Kosten für den Betrieb eines Help Desk
	Dokumentation	Erstellen von Dokumentationen zum Betrieb der IT
	Datenextraktion	Aufbereiten von Daten Ex- und Import
	Anwenderschulung	Kosten für Beratung und Schulung der Endanwender
	Installation und Upgrade	Installation und Upgrade von Anwenderinstallationen.
	Service	Kosten für eingekaufte Dienstleistungen
	Sicherheit	Kosten für den Schutz des LAN gegen unbefugten Zugriff.
	Formale Audits	Kosten für die Sammlung von Daten für Audits.
	Applikationsunterstützung	Kosten für : Support von Applikations-Server, Upgrades, Fixes, Service Packs, Troubleshooting
	Beschaffung	Einkauf von IT-Produkten, Gespräche mit Herstellern, Rechnungsstellung, Zahlungsüberwachung etc. .
Kapitalkosten	Hardware	Kosten für angeschaffte Hardware
	Software	Lizenzkosten für Software
	Zinskosten	Kosten für Finanzierung

Eine **Analyse der Zahlen** kann zu folgenden Aussagen führen:

- **Einfluss** der reinen **Hardwarekosten** auf die Gesamtkosten ist zu gering.
- Mehr als ein Drittel der Gesamtkosten ist **nicht budgetierbar**.
- Je mehr **dezentrale Eingriffe** in ein **vernetztes System** gemacht werden, desto weniger lassen sich die Kosten des Gesamtsystems kontrollieren.

⇒ Vergleiche Skript S. 31

### 3. Verfügbarkeit von Systemlandschaften

#### 3.1. IT-Risikomanagement

Das IT-Risikomanagement kann schon **im Vorfeld mögliche Gefahren** um Unternehmen **erkennen** und entsprechende Gegenmassnahmen vorschlagen.

Unter **Risiken** werden **Bedrohungen** zusammengefasst, die sich nachteilig auf den Betrieb und die **Verfügbarkeit** von Prozessen, Systemen, einzelnen Daten / Informationen im Unternehmen auswirken können (z.B. Softwarefehler, Hardwarefehler, Betriebsfehler).

Das IT-Risikomanagement ist nicht nur eine Pflicht, sondern auch eine **Chance**, IT-Prozesse zu optimieren und das Sicherheitsniveau in einem Unternehmen insgesamt zu verbessern.

In den **ISO-Standards** der 2700x-Reihe werden Normen zur Informationssicherheit definiert.

#### 3.2. Verfügbarkeitsrechnung

Unter **Verfügbarkeit** eines technischen Systems versteht man die Wahrscheinlichkeit, dass das System bestimmte Anforderungen innerhalb eines vereinbarten Zeitraums erfüllt.

$$\text{Verfügbarkeit} = \frac{\text{Gesamtzeit} - \text{Ausfallzeit}}{\text{Gesamtzeit}}$$

Die Verfügbarkeit V wird **in % ausgedrückt**: 98% bedeutet also, dass ein System  $0.98 \times 365 = 357.7$  Tage pro Jahr verfügbar ist  $\Rightarrow$  Das System hat eine **Ausfallzeit** (Downtime) von 7.3 Tage im Jahr. Bei der Budgetierung ist diese Angabe und die dadurch entstehenden indirekten Kosten wichtig.

Die Verfügbarkeit eines Systems wird i.d.R. in einem **Vertrag**, dem (Service Level Agreement, SLA) geregelt.

##### Beispiel:

Ein System, das 12 Stunden / Tag, an 5 Wochentagen, in 52 Wochen im Jahr ( $12 \times 5 \times 52$ ) zur Verfügung steht (3120 Stunden):

Verfügbarkeit	Minimale erwartete Betriebszeit	Maximale erlaubte Ausfallzeit	Restzeit
99%	3088,8	31,2	5640
99,5%	3104,4	15,6	5640
99,7%	3110,64	9,36	5640
99,9%	3116,88	3,12	5640
99,95%	3118,44	1,56	5640
100%	3120	0	5640

$\Rightarrow$  Bei einem System, dass über **keine Restzeit** verfügt, muss die **Systemwartung** in der erlaubten **Ausfallzeit** durchgeführt werden.

Die Verfügbarkeit kann mittels einem **Verfügbarkeitsverbund** optimiert werden. Darin sind die wichtigsten Komponenten redundant ausgelegt. Einsatzbereite: Überall dort wo eine hohe Betriebssicherheit benötigt wird: z.B. Flugüberwachung.

**Kennzahlen** der Verfügbarkeit:

- **maximale Dauer eines einzelnen Ausfalls**
- **Zuverlässigkeit** (Fähigkeit, über einen gegebenen Zeitraum hinweg unter Bestimmten Bedingungen korrekt zu arbeiten)
- **Fehlersicherer Betrieb** (Robustheit gegen Fehlbedingungen, Sabotage oder höhere Gewalt)
- **System- und Datenintegrität**
- **Wartbarkeit**
- **Reaktionszeit** (benötigte Dauer, bis eine Aktion ausgeführt wird)
- **Mean Time to Repair** (mittlere Dauer der Wiederherstellung nach einem Ausfall)
- **Mean Time between Failure** (MTBF, mittlere Betriebszeit zwischen zwei auftretenden Fehlern ohne Reparaturzeit)
- **Mean Time to Failure** (MTTF, gleich wie MTBF, bei Komponenten die nicht repariert werden können)

##### Beispiel MTBF:

Bei einer Festplatte sind **MTBF-Werte** von 137 Jahren üblich. Daraus kann die Wahrscheinlichkeit berechnet werden, dass es während der **Nutzungsdauer** von 5 Jahren zu einem Ausfall kommt.

$$p(T) = 1 - e^{-\frac{T}{MTBF}}$$

$$p(5a) = 1 - e^{-\frac{5a}{137a}} = 3,6\%$$

$\Rightarrow$  Die Verfügbarkeit bei aufeinander aufbauenden Teilsystemen wird als **Netto-Verfügbarkeit** bezeichnet (Multiplikation der Verfügbarkeitswerte der einzelnen Teilsysteme)

In der Praxis kennt man die **Verfügbarkeitsklassen** 2 bis 7 :

Verfügbarkeitsklasse	Verfügbarkeit	Ausfallzeit/Jahr	Systemart
1	90%	36.5 Tage	Nicht gemanaged
2	99%	3.6 Tage	Gemanaged
3	99.9	8.76 Stunden	Gut gemanaged
4	99.99	52 Minuten	Hochverfügbar
5	99.999	5 Minuten	Fehlertolerant
6	99.9999	30 Sekunden	Sehr hochverfügbar
7	99.99999	3 Sekunden	Ultra hochverfügbar

Typische **Komponenten** um eine Hochverfügbarkeit zu erreichen sind: *unterbrechungsfreie Stromversorgung, Doppelte Netzteile, ECC-Speicher, Einsatz von RAID-Systemen, Clustering.*

Schlüsselkomponenten bezeichnet man als „Single Point of Failure“ (**SPOF**). Das Umschalten auf redundante Komponenten heisst **Failover**.





### 3.5.1. Verfügbarkeit einer Leistung

Die **Verfügbarkeitsquote** ist das **Verhältnis** von **tatsächlicher** zu **geforderter Verfügbarkeit** in einem Bezugsraum und wird **in %** ausgedrückt.

### 3.5.2. Messmethode

Um die Einhaltung von SLAs zu überprüfen, bedarf es objektiven **Bewertungsmassstäben**. **Messtools** und Messmethoden können dies erleichtern.

Gemessen wird **stichprobenartig** und die Messtools sollten nicht gewechselt werden.

### 3.5.3. Antwortzeit (response time)

**Zeit zwischen** einer **Aktion** des Anwenders und der **Reaktion** des Systems inkl. aller Bearbeitungs- und Reaktionszeiten (z.B. Empfangen einer Nachricht).

**Ablaufende Prozesse** vom Anstoss bis zur Erledigung werden **Transaktionen** genannt.

### 3.5.4. Reaktionszeit

**Zeitspanne** zwischen dem **Eingang der Fehlermeldung** beim Auftragnehmer und einer **Reaktion** in definiertem Umfang. Die Reaktionszeiten werden aufgrund der verschiedenen **Fehler-Stufen** (First-Level, Second-Level, Third-Level) definiert  $\Rightarrow$  Aufwändigere Fehler haben eine höhere Reaktionszeit.

### 3.5.5. Sanktionen

Sofern die **Leistungserfüllung mangelhaft** ist, kommen **Sanktionen** zur Anwendung (z.B. nicht bezahlen des vereinbarten Entgelts). Es ist jedoch wichtig zwischen **sanktionsrelevanten** und **-nicht relevanten** Leistungen zu unterscheiden (nicht Geschäftskritische Service-Levels können ausgenommen werden).

### 3.5.6. Umfeldbedingungen

Definieren das **Bedingungen des Umfelds** (z.B. unterbrechungsfreie Stromversorgung, Klimatisierung von Serverräumen), **welche vorherrschen müssen**, damit der Auftragnehmer seine Leistungen vollbringen kann. Falls diese nicht gegeben sind, kann er solche Störungen als **Grund für eine Verzögerung/Verhinderung** seiner Leistung geltend machen.

### 3.5.7. Flexibilität

Eine SLA sollte **Änderungen von Umgebungsbedingungen** berücksichtigen (z.B. Anzahl Arbeitsplätze erhöht sich  $\Rightarrow$  Preis pro Arbeitsplatz definieren). Somit muss die SLA nicht angepasst werden.

### 3.5.8. Schritte zur Erarbeitung einer SLA

1. **Definition** der Dienstleistungen (Service-Levels)
2. **Kennzahlen** für Service Levels definieren und Überwachungs-Methode definieren
3. **Verrechnungsmodell** erarbeiten
4. **Plan** für die Einführung / Exekution des SLA erstellen

## 4. Modellierung und Darstellung von Geschäftsprozessen

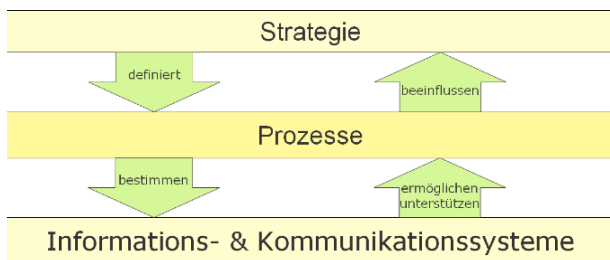
### 4.1. Prozess

Ein **Prozess** ist eine Menge von Aufgaben, die in einer kontrollierten Ablauffolge zu erledigen sind und durch Anwendungen der Informatiktechnik unterstützt werden können

Ein **Geschäftsprozess** beschreibt eine Folge von Einzeltätigkeiten, die schrittweise ausgeführt werden, um ein geschäftliches oder betriebliches Ziel zu erreichen.

#### Prozessarten:

- **Kernprozesse** (Leistungsprozesse): Tätigkeiten, die der direkten Erfüllung der Kundenbedürfnisse dienen. Sie leiten sich aus der Kernkompetenz einer Organisation ab.
- **Unterstützungsprozess**: Unterstützen die Kernprozesse, erzeugen selbst aber keinen direkten Kundennutzen.
- **Managementprozess** (Führungsprozesse): Steuerung von Kernprozessen in Organisationen, mit dem Fokus auf der Strukturierung der organisatorischen Rollen und deren Aufgaben.



(Abbildung: Wechselwirkung zwischen Strategie, Prozesse, Informations- und Kommunikationssysteme)

- **Aufgabe**: Betriebliche Funktion mit einem bestimmtem Ergebnis.
- **Aufgabenkette**: Wichtigste Aufgaben eines Prozesses und ihre Ablauffolge.
- **Leistungen**: Ergebnisse eines Prozesses
- **Prozessmanagement**: Strategiekonforme Gestaltung, Lenkung und (Weiter-)Entwicklung betrieblicher Prozesse

### 4.2. Modell

Ein Modell ist ein **vereinfachtes Abbild der Wirklichkeit**, wobei die „wichtigen“ Eigenschaften hervorgehoben werden.

#### Warum Modellieren?

- **Einheitliche Vorgehensweise** unabhängig vom Betrachteten System
- **Kosten** der Modellherstellung / Analyse sehr tief im Vergleich zum realen System

- **Kein Risiko** für reales System
- **Zeitlicher Ablauf** des dynamischen Verhaltens kann erheblich verkürzt werden.

### 4.3. Geschäftsprozess modellierung

Bei der Geschäftsprozessmodellierung werden **Geschäftsprozesse** oder Ausschnitte daraus abstrahiert und (grafisch) dargestellt. Der **Schwerpunkt** liegt dabei auf der Darstellung des **Ablaufs**.

#### Ziele der Geschäftsprozessmodellierung:

- **Dokumentation der Geschäftsprozesse** des Unternehmens um
  - Kenntnisse über Geschäftsprozesse erlangen
  - Geschäftsprozesse an andere Standorte zu übertragen
  - Mitarbeiter zu schulen oder einzuarbeiten
  - Wissensverlust zu vermeiden
- **Verbesserung der Geschäftsprozesse** (Ist-Analyse) um:
  - Neue Organisationsstrukturen einzuführen
  - Unternehmensaufgaben auszulagern
  - Unternehmensabläufe umzugestalten / straffen
- **Festlegung von Prozesskennzahlen** und Überwachung der Prozessleistung
- **Modularisierung** der Unternehmensaufgaben
- **Benchmarking** (Vergleich mit Messlatte) zwischen Unternehmensteilen, Partnern, Konkurrenten
- **Best Practice** finden (Best bekannte)

Bei der **Geschäftsprozessvisualisierung** ist zu beachten: 1. **Weglassen** (Notwendigkeit der Aktivität) 2. **Auslagern** (z.B. Outsourcing) 3. **Zusammenfassen** (Zusammenlegen von Aktivitäten) 4. **Parallelisieren** 5. **Verlagern** (früher mit Aktivität beginnen) 6. **Beschleunigen** (zeitlich verkürzen)

#### 4.3.1. Geschäftsprozess in der Praxis

Die **Geschäftsstrategie** gibt dem Betrieb die Vorgabe für die operationelle Umsetzung der Geschäftstätigkeit in einer Organisation.

Oft verunmöglichen die installierten Prozesse jedoch, **bestehende Marktmöglichkeiten** auszuschöpfen ⇒ die **konsequente Umsetzung** der **Geschäftsstrategie** in **operative Prozesse** ist gefordert.

Weitere **Auslöser** sind: *Veränderungen im Markt, Operationalisierung der Strategie (Neuausrichtung der Geschäftsstrategie), Erhöhte Kundenorientierung*

Fragen bei der Gestaltung der Prozesse:

- **Welche Aufgaben** sind erforderlich um die Leistung zu erbringen
- Wie ist die sinnvolle **Ablauffolge** der Aufgaben?
- Wie ist die **Interaktion der Prozesse**?
- **Wer führt** die Aufgaben aus?

## 4.4. Standards und Darstellungsformen der Prozessmodellierung

In der Prozessmodellierung gibt es **unterschiedliche Sprachen** je nach Anwendungsgebiet.

Mittels **Modellierungssprachen** werden die Anforderungen an ein Organisationssystem oder ein Softwaresystem sowie dessen Strukturen und inneren Abläufe auf einer höheren Ebene festgelegt.

### Beispiele von Modellierungssprachen:

- Klassische Ablaufdiagramme
- Ereignisgesteuerte Prozessketten (EPK)
- Business Process Modelling Notation (BPMN)
- Unified Modeling Language (UML)

### 4.4.1. Klassische Ablaufdiagramme

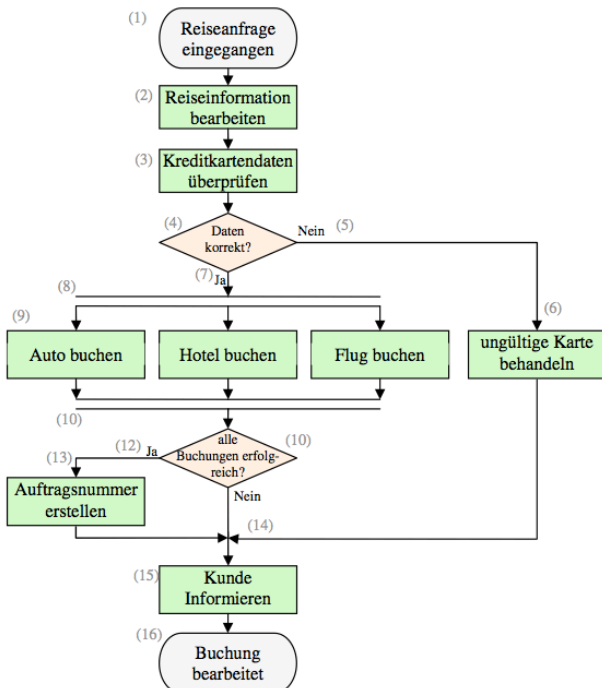
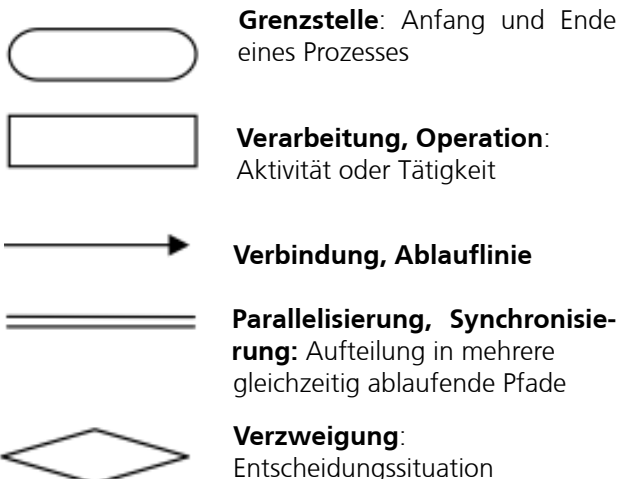


Abbildung 8: Buchungsprozess als traditionelles Ablaufprogramm

#### Legende:



### 4.4.2. Ereignisgesteuerten Prozesskette

Die Ereignisgesteuerte Prozesskette (EPK) ist eine **grafische Modellierungssprache** zur Darstellung von Geschäftsprozessen. Sie ist umfangreicher als der klassische Ablaufplan. Jede Aktivität (**Funktion**) wird **startet und endet** mit einem **Ereignis**.

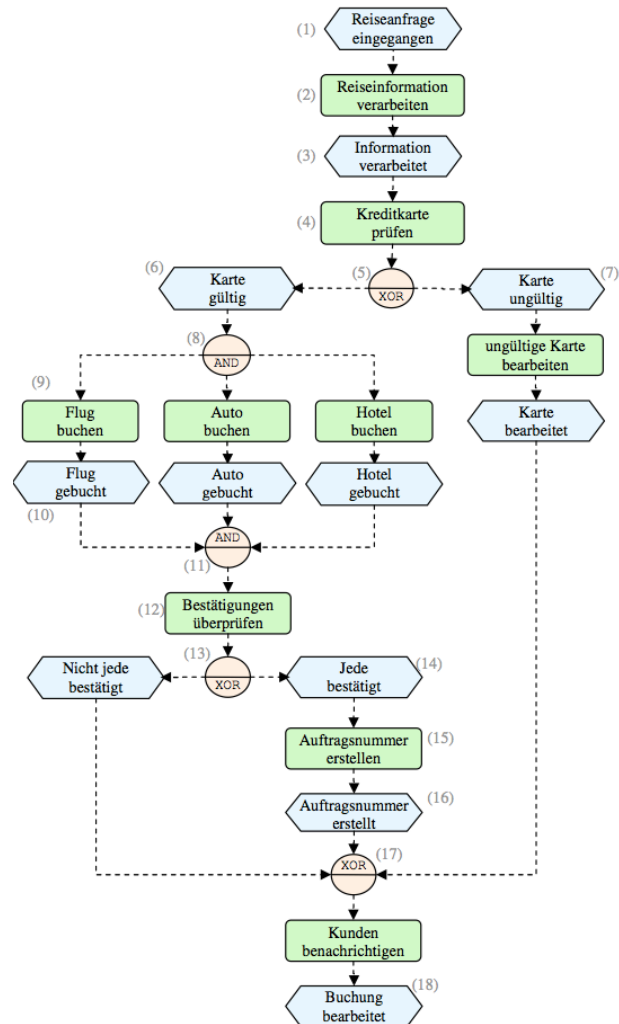
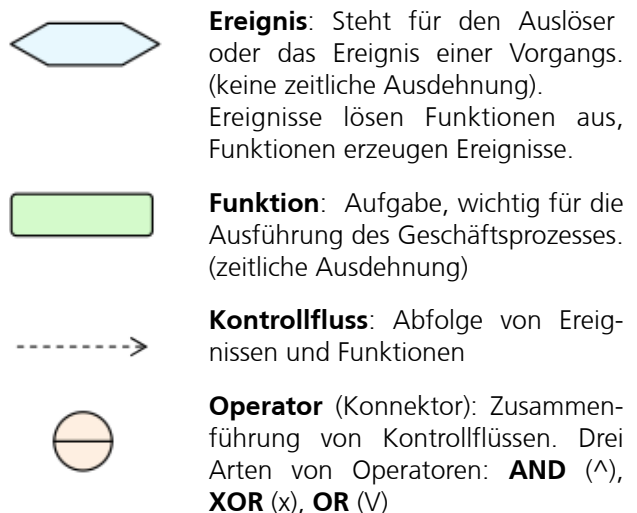
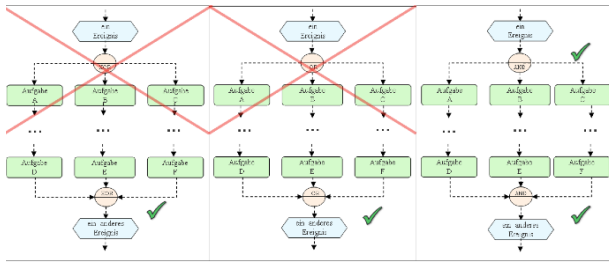


Abbildung 9: Buchungsprozess als Ereignisgesteuerte Prozesskette

#### Legende:





⇒ **Kein OR / XOR nach einem einzelnen Ereignis!**

Im **erweiterten EPK** gibt es noch Symbole für: Dokumente, Informationsobjekte, Prozesswegweiser, Organisatorische Einheit, Organisatorische Rolle.

#### 4.4.3. Business Process Modelling Notation (BPMN)

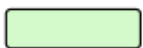
BPMN ist eine **ablaufplanbasierte Notation von Business Prozessen**. Die Modellierungssprache wurde auf das Verlangen nach einer einheitlichen Notation entwickelt. Sie erlaubt es, Geschäftsprozesse intuitiv darzustellen und **kombiniert Vorteile existierender Sprachen**.



**Legende:**



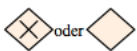
**Ereignis:** Auslöser oder Ergebnis eines Vorgangs. Drei Grundtypen: **Startereignis** (normaler Kreis), **Zwischenereignis** (Doppelter Kreis), **Endereignis** (Fetter Kreis)



**Aufgabe:** Ausdruck für Arbeit. Kann sowohl atomar oder zusammengesetzt sein (Subprozesse)



**Parallele Verzweigung / Synchronisation:** AND-Verzweigung. Bei der Sync. werden zwei oder mehrer parallele Pfade zu einem zusammengefasst.



**Exklusive Entscheidung / Zusammenführung:** XOR-Verzweigung



**Inklusive Entscheidung und synchronisierende Zusammenführung:** OR-Verzweigung



**Komplexe Entscheidung / Zusammenführung:** Erlaubt es, komplexe Bedingungen für die Fortführung des Prozesses zu definieren.



**Kontrollfluss** (Sequence Flows): Abfolge von Aufgaben oder Ereignissen

#### Regeln zur Modellierung mit BPMN

- Ein Prozess beginnt mit dem Startereignis
- Ein Prozess endet mit dem Endereignis
- Der Fluss durch den Prozess wird über die Gateways und den Zwischenereignissen gesteuert

#### 4.5. Prozessmuster

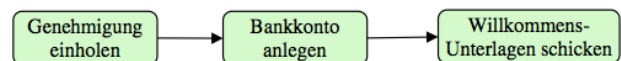
Die Prozessmuster bilden die **Grundlage der Geschäftsprozessmodellierung**. Sie sollen helfen, Prozesse besser zu konstruieren und können als eine **Ansammlung von Aufgaben** angesehen werden, die so angeordnet sind, dass ein anstehendes Problem gelöst werden kann.

##### 4.5.1. Grundmuster

Das Grundmuster deckt die fundamentale Prozessfähigkeit ab (sequentiell ausführen, Verzweigen und Zusammenführen, parallele Ausführung, ...).

##### 4.5.2. Sequenz

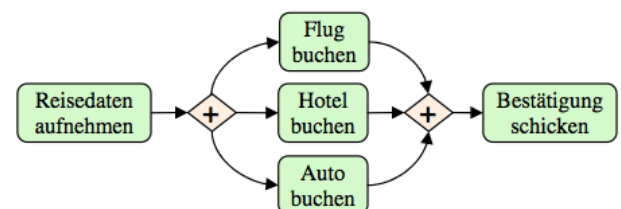
Das Ziel des Sequenzmusters ist es, Aufgaben **nacheinander** auszuführen. Erst A, dann B, dann C.



##### 4.5.3. Parallele Verzweigung (AND)

Von einer einzigen Aufgabe in mehrere **parallele Pfade** verzweigen (auch **AND-Split**).

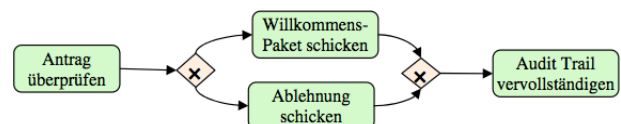
Wird eingesetzt, wenn **mehrere Aufgaben gleichzeitig** ablaufen sollen, ohne direkt voneinander abhängig zu sein. Beispiel Buchung einer Reise (Hotel-, Flugzeug-, und Autoreservierung).



Die **Synchronisation** bewirkt, dass parallele Pfade zu einer Aufgabe führen, welche **wartet**, bis alle parallelen Pfade beendet sind, bevor diesen Ihren Prozess startet.

##### 4.5.4. Exklusive Fallunterscheidung (XOR)

Von einer Aufgabe auf genau **einen von mehreren Pfaden** verzweigen (auch **XOR-Verzweigung**). Die Auswirkung ist die einer *if*-Aussage im Prozess.



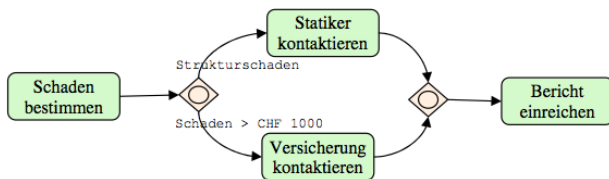
Am Endpunkt der exklusiven Verzweigung, benötigt es eine **Zusammenführung** (siehe Grafik).

#### 4.5.5. Mehrfachwahl (OR)

Von einer Aufgabe können **einen oder mehrere Pfade gewählt werden**. Jede dieser Pfade wird jedoch nur dann gewählt, **wenn eine spezifische Bedingung wahr** ist (auch **OR-Verbindung**).

Bei der **Zusammenführung** wartet die gemeinsame Aufgabe wieder, bis sämtliche parallele Prozesse abgeschlossen sind.

Der Unterschied zur XOR-Verbindung ist, dass bei dieser immer nur genau ein „Strang“ aktiviert werden kann. Bei der OR-Verbindung mehrere.



#### 4.5.6. Weitere Muster

Im Skript sind noch weitere Muster beschrieben (Mehrfachverbindung, Diskriminator und N-aus-M-Verbindung, Strukturmuster, Muster mit mehreren Instanzen, Zustandsbasierte Muster, Annullierungsmuster). Ich erachte diese als äusserst exotisch und somit nicht als Prüfungsrelevant.

#### 4.5.7. Geschäftsprozessoptimierung

Geschäftsprozesse können mithilfe von **Computersimulationen** und **Optimierungsmethoden** verbessert werden.

## 5. IT-Management und ITIL

Das **IT-Management** ist verantwortlich für eine **bestmögliche Unterstützung der Geschäftsprozesse** durch die IT. Zudem regelt es die Überwachung und Anpassungen der IT-Dienstleistung ⇒ Steigerung Effizienz, Qualität und Wirtschaftlichkeit. Die IT-Abteilung ist für IT-Infrastruktur verantwortlich (Planung, Beschaffung, Installation, Wartung usw.)

**IT-Outsourcing** bezeichnet **Auslagerung** (unabhängiges Unternehmen) oder **Ausgliederung** (selbständiges, aber rechtlich verbundenes Unternehmen) der gesamten IT-Dienstleistung oder nur Teile davon (z.B. Support).

- **Vorteile:** Kostenvorteil, Nutzung von Synergien, Aktualität/moderne Technologien, Steigerung Performance...
- **Nachteile:** Abhängigkeit, Sicherheit und Verfügbarkeit, Konflikte, Kulturelle Unterschiede, Verlust IT-Kompetenzen...

⇒ Eine klare **Planung und Koordination** des Outsourcing nötig!

### 5.1. Application Service Provider (ASP)

APS ist ein Dienstleistungsunternehmen, welches **anwendungsorientierte Services** anbietet. Alle Anwendungen laufen dabei auf den Servern des Anbieters. Über Internet können die Software-Funktionen bezogen werden. Es umfasst alle Dienste, die in Verbindung mit der Bereitstellung von Applikationssoftware zu sehen sind. Dazu gehören die **Verwaltung** und **Bereitstellung** einzelner **Komponenten der Anwendungssoftware**, sowie die Zugriffskontrolle und Authentifizierung einzelner Benutzer oder Benutzergruppen.

- **Vorteile:** Schnelle Softwareeinführung/Aktualisierung, evtl. Kostenreduktion
- **Nachteile:** Datenschutz/-Sicherheit, Erfahrung des IT-DL genügend?

### 5.2. Information Technology Infrastructure Library (ITIL)

ITIL ist ein **IT-Rahmenwerk** um **Abläufe** im IT-Service-Management **effizient** und **erfolgreich** zu gestalten. Standard für den Betrieb einer IT-Infrastruktur mit deren Prozessen. Die Hauptziele des IT Service Managements sind Kundenorientierung, qualitative Verbesserung der IT-Services, höhere Kundenzufriedenheit und Kostenreduzierung. Dazu bietet ITIL systematische und praxisorientierte Methoden um die Effizienz des IT-Managements zu steigern (WAS getan werden muss!)

Die einzelnen **Service-Bereiche** für das IT-Service-Management sind in einem ITIL-Referenzmodell beschrieben. Diese Prozesse sind aufgeteilt in:

#### 5.2.1. „Service Delivery“:

Die Prozesse haben insgesamt zum Ziel, der Unternehmung eine IT-Infrastruktur und IT-Dienste zu liefern, die deren Geschäftstätigkeit optimal unterstützen. Sie sorgen für den nötigen Grad von Verfügbarkeit von IT-Diensten (Availability Management), ausreichende Speicher- und Leistungskapazität (Capacity Management), eine umfassende Eventualfallplanung (IT Service Continuity Management), den effizienten Einsatz der Finanzmittel (Financial Management for IT) und eine konstante Kontrolle und Verbesserung der Qualität der IT-Dienstleistungen (Service Level Management).

#### 5.2.2. „Service Support“:

Die Prozesse sichern die Stabilität und Kontinuität der IT-Dienste, indem sie Störungen überbrücken (Incident Management), deren Ursachen diagnostizieren und beheben (Problem Management), Änderungen in der Infrastruktur planen, regeln und überwachen (Change Management), die Einführung neuer Hard- und Software planen und durchführen (Release Management) und die Komponenten der IT-Infrastruktur mit ihren Eigenschaften und gegenseitigen Verbindungen in einer Datenbank nachweisen (Configuration Management).

- **Incident Management:** Schnellstmögliche Behebung von Störungen
- **Problem Management:** Bestimmung und Behebung der Ursachen für (potentielle) Störungen
- **Configuration Management:** Bereitstellung gesicherter und genauer Informationen über die IT-Infrastruktur
- **Chance Management:** Kontrolle der effizienten und kostengünstigen Durchführung von Änderungen unter Sicherstellung der Qualität der IT-Services

ITIL – Implementierungen sind normalerweise gross und komplex. Beginnt mit der Ist-Analyse und definiert danach die Soll-/Ist Abweichung. Eine Einführung kann deshalb als Projekt gesehen werden, nützlich eine Projektmanagement!

## 6. Analytische Informationssysteme

**Hauptproblematik** bei heutigen Unternehmen sind die **heterogenen IT-Systeme**. Das heisst es existieren auch **unterschiedliche Datenbanken**.

**Unterschiedliche Ordnungsbegriffe** können nun die Zusammenführung erschweren:

- **Homonyme** (Doppelbedeutungen): Partner bedeutet im ersten Fall Lieferant, im zweiten Fall Kunde.
- **Synonyme**: Personal und Mitarbeiter

Ein weiteres Problem ist, dass der einzelne Anwender **keine umfassende Analysen** abrufen kann. Operative Datenbank-Systeme sind meist nur auf die **Bearbeitung einzelner Transaktionen** (Erfassen eines Auftrags) optimiert.

### 6.1. Data Warehouse (DWH)

**Ziel** eines Data Warehouse ist es die Mitarbeiter in den **Kontroll- und Entscheidungsprozessen** zu **unterstützen**. Zum Beispiel durch Auswertungen wie „Wodurch zeichnen sich gewinnbringende Kunden aus?“ oder „Welcher Vertriebskanal ist am wirksamsten?“.

**Anforderungen** an ein DWH:

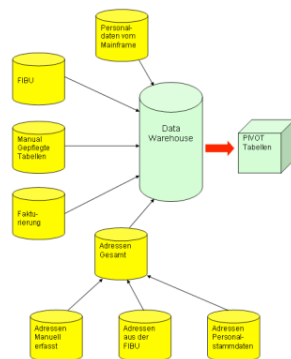
- DWH muss **Informationen über Kunden oder Produkte** liefern (weniger über innerbetriebliche Prozesse), welche zur **Entscheidungsunterstützung** dienen.
- DWH ist eine **Zusammenfassung verschiedener Datenbanken**, welche in Bezug auf die **Struktur** und das **Format vereinheitlicht** wurden (Sternschema, s.h. Wikipedia).
- **Alle Daten** erhalten den **Bezug zu einem Zeitraum**, somit können auch historische Zahlen verwendet werden.

**Beispiel** eines DWH:

Alle verschiedenen Tabellen werden schlussendlich zu einer PIVOT Tabelle zusammengefasst, welche dann individuell angepasst werden kann (z.B. Durch Drop Down Menüs).

**Merksätze** zum Thema DWH:

- **Abfragen** sind **zeitaufwendig** (Joins) und **performancelastig**.
- Daten eines operativen Systems genügen meist nicht ⇒ **heterogene Systemlandschaft**
- Operative Systeme haben keine **historische Daten**, diese sind aber wichtig für Entscheide.



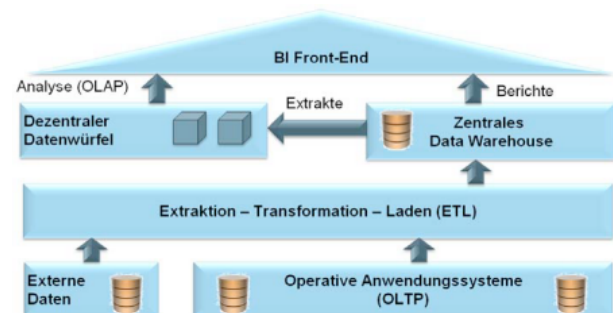
- Oft sind auch **externe Daten** nötig (z.B. Marktanteile von Wettbewerbern, Telefonnummern von Swisscom).
- **DWH** kann so **gross** werden, dass es **schwer handlebar** wird. ⇒ Lösung: Teilgebiete werden in Teilmengen (Data Mart) des gesamten DWH gebildet.

#### 6.1.1. Die DWH Architektur:

Ein DWH besteht aus einem Bereich **Datenbereitstellung** und einem der **Informationsgewinnung**.

Bei der **Datenbereitstellung** werden im ersten Schritt **Daten** aus bestehenden Datenbanken **ausgelesen**.

- Hier ist wichtig in welchen **Zeitabständen** das passieren soll. Am Effizientesten ist wenn man nur die seit letzter Übertragung **veränderten Daten überträgt**.
- Wichtigster Schritt ist nachher die **Transformation der Daten** von den operativen Datenbanken in einen einheitlichen Datenbestand im DWH.  
**Beispiel:** Geschlecht: m/w, männlich/weiblich, 0/1 etc. → vereinheitlichen
- Zusätzlich müssen auch **Plausibilitätsprüfungen** ausgeführt werden (z.B. liegen Daten in richtigen Formaten vor ⇒ Datumsformat).
- Ausserdem können **Daten selber berechnet** werden und in die DWH Datenbank abgelegt werden (z.B. hat man Gesamtkosten und variable Kosten, dann könnten auch Fixkosten berechnet werden).
- Grundlage für eine Datenbank sind die **Metadaten** (Daten über Daten). Sie beschreiben die **Struktur** und **Zusammenhänge eines Systems**. Ändert man zum Beispiel ein Datenfeld in der operativen Datenbank, so muss man wissen welche Felder im DWH betroffen sind.



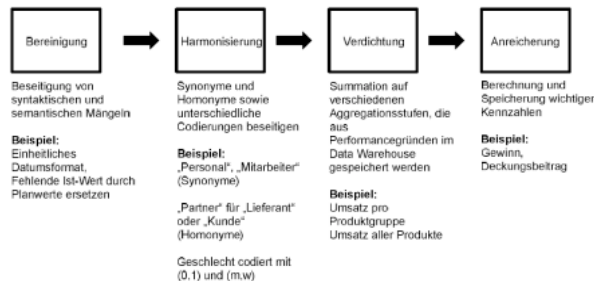
#### 6.1.2. Der Aufbau eines Business Informations (BI) Systems:

- **Operative Anwendungssysteme (OLTP:** Online Transactional Processing) **speichern** typischerweise die **Transaktionen eines Geschäfts** (z.B. Tagesverkäufe von Migros werden in OLTP Datenbanken gespeichert.).
- **Online Analytical Processing (OLAP)** hingegen ist die **Zusammenfassung, Aufbereitung und Auswertung der operativen Daten**. OLAP ermöglicht die **mehrdimensionale Analyse** betriebswirtschaftlicher Variablen. Zum Beispiel Um-



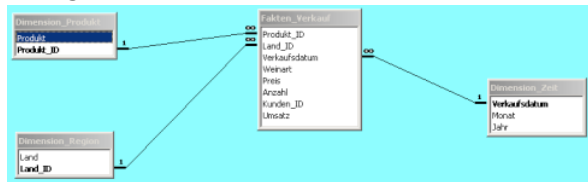
satz, Gewinn, nach verschiedenen Dimensionen, z.B. Kunden, Produkte, Regionen. Die typische Darstellung dieser mehrdimensionalen Sicht ist der **OLAP-Würfel**. Die Kanten des Würfels sind die Analyse Kriterien oder Dimensionen.

- Beim **ETL (Extraktion – Transform – Laden)** Prozess werden die operativen Datenbanken (OLTP) „datawarehousekonform“ gemacht. ETL beinhaltet folgende Phasen:



„datawarehousekonform“ gemacht. ETL beinhaltet folgende Phasen.

- Da Manager **multidimensional** (Zeit, Produkt, Region, Kundengruppe) denken, eignet sich das **Sternschema** sehr gut als Datenmodell. Hierbei werden die Daten sternförmig um eine so genannte **Faktentabelle** (Kennzahlen, Umsatz) angeordnet. Die Strahlen des Sterns bilden die **Dimensionsstabellen** (Zeitdaten, Produktdaten, Regionsdaten, Kundengruppendaten). Das Sternschema kommt bei **analytischen Abfragen (OLAP)** zum Einsatz und vergibt an den Benutzer lediglich **Lesezugriffe**.



- Das **Normalisierte Schema** (schrittweise Zerlegung der Daten in einzelne Tabellen, um Redundanzen zu vermeiden) hingegen wird bei **produktiven Datenbanken (OLTP)** verwendet. Bei diesen OLTP Systemen hat der Benutzer Schreib- und Lesezugriffe.



## 6.2. Eine praktische Einführung in OLAP

**Ausgangslage:** Man hat einen Webshop, der Wein vertreibt. Um den Gewinn zu überprüfen und gegebenenfalls eine Reorganisation der Logistik zu vollziehen, muss man gewisse **Kennzahlen** (Umsatz) haben.

**Problem:** Die verschiedenen Mitarbeitern werden jeweils **verschiedene Darstellungen** der Kennzahlen benötigen (Produktmanager  $\Rightarrow$  Umsatz pro Zeit und Produkt, Bereichsleiter  $\Rightarrow$  Umsatz pro Zeit und Bereich). Vielleicht werden **andere Dimensionen** verlangt, zum Beispiel Regionen (CH, Ö, D), Produktgruppe (Rot-, Weisswein) oder Zeit (Jahr, Quartal, Monat). Bei so vielen verschiedenen Tabellen verliert man dann schnell den **Überblick**.

**Lösung:** Hier kommt dann **OLAP** ins Spiel. Es **erleichtert** die **Analyse von Kennzahlen** unter **verschiedenen Gesichtspunkten** (Dimensionen). So können interaktive Abfrageprozesse **ad-hoc** erzeugt werden. Ebenfalls hat man die Möglichkeit **graphische Darstellung** in Form von Kuchen- oder Balkendiagrammen zu erzeugen.

OLAP-Anwendungen werden häufig als **Würfel** symbolisiert. Da in den meisten Fällen mehr als drei Dimensionen involviert sind, spricht man häufig von **Hypercubes**. Die Elemente einzelner Dimensionen werden **Ausprägungen** oder **Members** genannt (z.B. Merlot ist eine Ausprägung der Dimension Produkte). Die Daten selber werden in so genannten **Zellen** gespeichert.

Ein DWH kennt die Funktionen **Drill-Down**, **Roll-Up**, **Slicing** und **Dicing**:

- Drill-Down** bedeutet nichts anderes als die **Detaillierung** der Information.
- Roll-Up** ist das Gegenteil. Man springt auf die nächst **höhere Ebene**.
- Slicing** bedeutet, dass eine **beliebige Kombination von Dimensionen** und **Ausprägungen** angezeigt wird. Es wird eine Scheibe aus dem Würfel geschnitten.
- Dicing** bedeutet, dass die **Achsen** in einer Darstellung **beliebig ausgetauscht** werden können. Das heisst **Dimensionen** (Region, Produkt etc.) können **mal** als **Spalte**, **mal** als **Zeile** stehen.

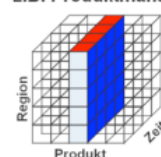
Das Diagramm zeigt zwei Tabellen, die die Operationen Drill-Down und Roll-Up illustrieren. Die obere Tabelle ist eine detaillierte Auflistung der Umsätze pro Produkt und Region, die untere Tabelle ist eine aggregierte Auflistung der Umsätze pro Produkt und Region.

Umsatz, Jun					
	CH	D	A	Alle Regionen	
Rotweine	200	100	50	350	
Weissweine	120	110	30	260	
Alle Produkte	320	210	80	610	

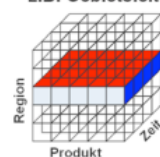
Umsatz, Jun					
	CH	D	A	Alle Regionen	
Rotweine	200	100	50	350	
Chardonnay	25	34	22	81	
Zinfandel	12	9	32	53	
Weissweine	120	110	30	260	
Alle Produkte	320	210	80	610	

Produkt Sicht  
z.B. Produktmanager



- alle Regionen
- gesamter Zeitraum
- ein Produkt (Filter)

Regionale Sicht  
z.B. Gebietsleiter



- alle Produkte
- gesamter Zeitraum
- eine Region (Filter)



**Unterschiede** zwischen OLTP und OLAP:

Transaktionsorientierte Systeme <i>Operative Systeme</i>	Auswertungsorientierte Systeme
OLTP (Online Transaction Processing)	OLAP (Online Analytical Processing)
Häufige, einfache Anfragen	Weniger häufige, komplexe Anfragen
Kleine Datenmengen je Anfrage	Grosse Datenmengen je Anfrage
Operieren hauptsächlich auf aktuellen Daten	Operieren auf aktuellen und historischen Daten
Schneller Update wichtig	Schnelle Kalkulation wichtig
→ Datenbanksystem kann nicht gleichzeitig für OLTP- und für OLAP-Anwendungen optimiert werden	
Paralleles Ausführung von OLAP-Anfragen auf operationalen Datenbeständen könnte Leistungsfähigkeit der OLTP-Anwendungen beeinträchtigen	

- Warum kann man für OLAP-Auswertungen nicht direkt auf die operativen Daten zugreifen? ⇒ Für **OLAP-Auswertungen** werden meist **mehr als nur ein operatives System** verwendet.

Ein **Data Mart** ist ein **kleineres abteilungsbezogenes DWH**. Er kann abhängig von einem zentralen DWH sein oder auch dezentral unabhängig aufgebaut sein.

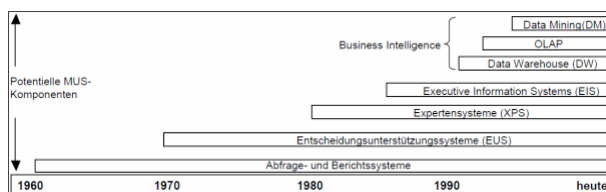
### 6.3. Data Mining

**Data Mining** ist die systematische Anwendung von **Methoden**, die meist **statistisch-mathematisch** begründet sind. Das **Ziel** ist die **Mustererkennung**. Typisch ist, dass die **gewonnenen Daten nie so gespeichert** werden.

**Beispiele** sind: Warenkorbanalyse zur Erforschung des Kaufmusters, Verhaltensmuster (Terroristenerkennung), Einteilung der Kunden in Risikoklassen bezüglich Bonität.

⇒ Einkaufskombination von Windeln und Sixpacks an Samstagen

### 6.4. Managementunterstützungssysteme MUS



### 6.5. Abfrage- und Berichtssysteme

Sind nichts anderes als **Datenbankabfragen** auf produktive Datenbestände (Kundenadressen, Verkaufszahlen etc.). Sie geben aber nur eine rückwärtsgerichtete **Momentaufnahme** (Verkaufszahlen im letzten Quartal), **ohne Zukunftsszenarien**.

### 6.6. Entscheidungsunterstützungssysteme EUS (Decision Support Systems, DSS)

Sind **Werkzeuge**, welche die **Entscheidungsfindung erleichtern** sollen. Es handelt sich meistens um eher **mathematische Modelle/Simulationen**.

**Beispiele:** Nutzwertanalyse, Was-Wenn-Analyse und lineare Optimierung, zu der anschliessend Anwendungsfälle zu finden sind:

- Produktionsplanung:** Wieviel soll von jedem Produkt produziert werden, damit der Profit maximiert und die Ressourcenbeschränkung nicht verletzt wird?
- Mischungsprobleme:** Wie erstelle ich aus Rohmaterialien mehrere Endprodukte mit minimalem Kostenaufwand und Berücksichtigung der Mindest- und Höchstgrenzen einzelner Stoffe?
- Routing** (Telekommunikation/Verkehr): Wie kann der Verkehr so geroutet werden, dass alle Anforderungen erfüllt und keine Kapazitätsbedingungen verletzt werden?
- Spieltheorie:** versucht Systeme mit mehreren Akteuren zu analysieren und das rationale Entscheidungsverhalten abzuleiten. Hierzu wird eine reale Situation in eine Art Spiel übertragen (Gefangendilemma). Die Spieltheorie ist eng mit der Wahrscheinlichkeitsrechnung verbunden.

### 6.7. Expertensystem XPS

Ein XPS besteht aus drei Elementen: einem **Regelsystem**, einer **Faktenbasis** und einem **Regelinterpret**.

- Die **Regelbasis** besteht aus einer Menge von Regeln (wenn – dann)
- Die **Faktenbasis** besteht aus einer Menge Fakten (Wissensdatenbank)
- Der **Regelinterpret** leitet aus den vorhandenen Fakten mit Hilfe der Regeln neue Fakten ab und führt diese in die Faktenbasis.

### 6.8. Executive Information System

Mit dem EIS wird dem oberen Management versucht anhand **einiger Indikatoren** einen Überblick über das **gesamte Unternehmen** zu geben (z.B. mit Balance Scorecard). Diese Indikatoren werden dann in einem sogenannten **Management Cockpit grafisch** dargestellt.

## 7. Buch „ERP für KMU“

### 7.1. Grundlagen ERP

Business Software ist heutzutage fast in jedem schweizerischen Unternehmen anzutreffen.

#### 7.1.1. Definition und Übersicht

Software für Firmen: Computerprogramme zur Unterstützung von Arbeitsabläufen in Firmen, also Software zur Geschäftsprozessunterstützung. Man spricht von „Betriebswirtschaftlicher-Software“ oder „Produktionsplanungs -Software“. Ganz allgemein: „Business-Software“

Unter „**Enterprise Resource Planning**“-Systemen versteht man Softwarepakete, welche alle für ein Unternehmen relevante Daten zur Bewirtschaftung der Ressourcen – Finanzen, Arbeitskräfte, Maschinen, Material, Zeit usw. – integrieren und damit operative und strategische Entscheidungen ermöglichen.

#### 7.1.2. Abgedeckte Prozesse

Die Prozesse in Firmen:

- **Wertschöpfende Prozesse** (Produktentwicklung, Auftragsgewinnung, Auftragserfüllung, Kundendienst)
- **Unterstützende Prozesse** (Evolutionsprozesse, Supportprozesse)

#### 7.1.3. Wichtige Module, Funktionsumfang

Entsprechen Wertschöpfenden und Unterstützenden Prozessen. Beispiel für Module: Verkauf, Einkauf, Materialwirtschaft, Kalkulation, Service/CRM, Produktion, Finanzbuchhaltung, Debitoren, Kreditoren usw.

#### 7.1.4. Kostenstruktur ERP

Kosten für ein ERP:

- **Investitionskosten** (Hardware, Software (Lizenzen), Dienstleistungen für Auswahl, Anpassung, Einführung)
- **Unterhaltungskosten** (Softwareupdate, Support, Dienstleistungen für Betrieb)

Je besser die Software zu den Anforderungen passt, desto kleiner der Anpassungsaufwand. Eine einfache Abschätzung der Investitionskosten durch „Cash out“:

**Zu erwartende Investitionen = k \* Benutzeranzahl**

k = Komplexität, üblich zwischen CHF 10'000 bis CHF 15'000.

Man unterscheidet **concurrent User** und **named User**. Concurrent User arbeiten gleichzeitig im System oder Modul. Z.B. braucht eine Firma 50 named

User – für ein bestimmtes Modul mag jedoch ein Maximum von 2 concurrent Usern genügen – wenn man davon ausgeht dass gleichzeitig nie mehr als zwei Personen in diesem Modul arbeiten.

Alternative zum Kauf: Teil der Leistungen extern zu beziehen. Dies nennt sich Application Service Providing (ASP)

#### 7.1.5. Gründe für den ERP-Einsatz

- Kontrollierter Auftragsablauf
- Datenaustausch mit Kunden und Lieferanten
- Konsistente zentrale Datenhaltung
- Präzise Lagerübersicht
- Materialengpässe für die Fertigung vermeiden

### 7.2. ERP-Auswahl

#### 7.2.1. Erwartungshaltung

Folgende Grafik zeigt die Erwartungshaltung der Schweizer KMU welche mit dem Start eines Projektes zur Reorganisation und IT-Einführung verbunden ist. Helle Balken ⇒ Welche Ziele für Unternehmen wichtig. Dunkle Balken ⇒ Grad der Zielerreichung.

⇒ Grafik s. 61

#### 7.2.2. Stolpersteine in ERP Projekten

- Projekt hat nicht genügend Personal, Geld und Zeit
- Projekt hat zu wenig Priorität
- Mangelnde Identifikation des Managements
- Unrealistische zeitliche Vorgaben.

#### Erfolgsfaktoren und Lösungsansätze

- Nötig für den Erfolg eines Projektes:
- Schnelle u. effektive Grobevaluation
- Werkzeuge für das Projektmanagement
- Kommunikation und Kooperation unterstützen
- Professionelle Berater

#### 7.2.3. Vorgehen für Evaluation und Einführung

Durch die Einführung einer Business Software wird im ERP System ein Abbild der Firma geschaffen. Die Einführung zusammen mit der Gestaltung der Organisation ist erforderlich. Intensiver Austausch zwischen Organisationsgestaltung und IT-Gestaltung ist wichtig.

#### Organisatorisches Grobkonzept

1. Ist-Prozesse abbilden
2. Soll-Prozesse festzulegen.

Es gilt die wichtigsten Prozesse des Unternehmens zu identifizieren. Die Ist-Prozesse werden mit einem „stellenorientierten Ablaufdiagramm“ dargestellt. Die Soll-Prozesse dienen als Basis für den Prozess der IT-Gestaltung, welcher mit der Phase Evaluation beginnt.

## Vom Marktüberblick zur Einführung

Bis zu den Vertragsverhandlungen ist mit mehreren Monaten zu rechnen.

⇒ Grafik s. 66

## Marktüberblick ERP-Systeme

Durch Grobevaluation gilt es einen Marktüberblick zu erreichen. Dazu gibt es im Internet Datenbanken mit Produktanbietern.

## Grobevaluation – Hauptkriterien

Ziel ist aus der Marktübersicht 6 bis 10 Systeme bzw. Anbieter zu selektionieren. Hauptkriterien:

- Funktionale Abdeckung der Bedürfnisse
- Anpassbarkeit an zukünftige Bedürfnisse
- Referenzkunden
- Preis-Leistungs-Verhältnis

## Feinevaluation

Hier **detailliert** man die **Prozessschritte**. Basis hierfür sind die in den Soll-Prozessen der Grobevaluation festgehaltenen Ablaufdiagramme. Anschliessend kann die **Offert-Anfrage** an den Anbietern zugesandt werden. Das **Pflichtenheft enthält**: Firmenbeschreibung, Prozessdiagramme, Prozessanforderungen, Einsatzkonzept.

## Einsatzkonzept

Mit dem Abschluss der Feinevaluation sollte ein Beschaffungsantrag vorliegen. Im Einsatzkonzept steht welche Mitarbeiter mit welchen IT-Funktionen arbeiten werden. Betrifft also IT- als auch Organisationsentwicklung.

## Drehbuch und Prototyping

Um aus enger Auswahl zu entscheiden, kann eine Vorführung organisiert werden. Unter Umständen lohnt es sich ein Prototyping zu verlangen.

## 7.2.4. Vertragsabschluss

Es muss überlegt werden, wer den Vertrag aufsetzt, welche Konsequenzen daraus resultieren. Ist das Machtverhältnis der Partnerschaft ausgeglichen?

## 7.3. Blick auf den ERP-Markt

### 7.3.1. Standardsoftware

Ist „fertige“ Software und darauf ausgerichtet von mehreren Unternehmen gekauft zu werden. Deckt das Bedürfnis einer Branche ab, kann aber durchaus angepasst werden. Eignet sich, wenn Sie die Prozesse des Unternehmens genügend gut stützt.

### 7.3.2. Branchensoftware

Fokussiert einen bestimmten Wirtschaftszweig. Wird erfolgreich dort eingesetzt wo mehre Anwender mit ähnlichen Prozessen arbeiten.

### 7.3.3. Individualsoftware

Wird ausschliesslich im dafür vorgesehenen Unternehmen eingesetzt. Für die Entwicklung dominieren die Technologien .net und J2EE.

### 7.3.4. Open-Source-Software (OSS)

Viele begegnen OSS mit Ablehnung, weil Geschäftsapplikationen auf idealistische Freizeitprogrammierer abgestützt werden. ERP stösst auf zusätzliche Barrieren wie gesetzliche und lokale Spezialitäten. Für die OSS-ERP Evaluation gelten jedoch dieselben Kriterien wie für andere ERP Systeme.

### 7.3.5. Integration in die Softwarelandschaft

Es empfiehlt sich eine gründliche Bestandsaufnahme der IT-Landkarte des Unternehmens zu zeichnen. Probleme bei der Integration sind:

- Aufwand für Schnittstellen Programmierung
- Laufende Anpassungen sind notwendig

### 7.3.6. SOA und Integration

Mit Service Oriented Architecture (SOA) werden mehrfach verwendbare Programme als Services definiert und diese technisch eigenständig implementiert. Mit SOA kann eine bestehende heterogene Software Landschaft schrittweise abgelöst und zusammengeführt werden.

### 7.3.7. IT extern betreiben?

Im Laufe der Grobevaluation muss festgestellt werden, ob die IT intern oder extern betrieben werden soll. Gerade kleine Firmen welche Sicherheitsfragen unbedacht lassen, profitieren von professionellen ASP (Application Service Provider). ASP-Lösungen haben Kostenvorteile bei der Infrastruktur.