



ARIS-Methode

ARIS Platform

Version 7.2 - Service Release 2

Oktober 2011

Copyright (C) Software AG, 1992 - 2011

Alle Rechte vorbehalten.

Der Inhalt dieser Publikation unterliegt dem Urheberrecht. Veränderungen, Kürzungen, Erweiterungen und Ergänzungen bedürfen der vorherigen schriftlichen Einwilligung durch Software AG, Darmstadt. Jede Vervielfältigung ist nur gestattet unter der Bedingung, dass dieser Urheberrechtsvermerk beim Vervielfältigen erhalten bleibt. Jede Veröffentlichung oder jede Übersetzung bedarf der vorherigen schriftlichen Einwilligung durch Software AG, Darmstadt.

Dieses Handbuch repräsentiert die Beschreibung von Einstellungen und Funktionalitäten, die zum Zeitpunkt der Drucklegung gültig waren. Da der Erstellungsprozess von Handbuch und Software verschiedenen Fertigungszyklen unterliegen, kann die Beschreibung von Einstellungen und Funktionalitäten von den tatsächlichen Gegebenheiten abweichen. Über diese Abweichungen informiert Sie die Readme-Datei, die dem Produkt beiliegt. Bitte lesen und berücksichtigen Sie diese Datei bei Installation, Einrichtung und Verwendung des Produkts.

Die Web-Sites, auf die hier verwiesen wird, unterliegen nicht der Kontrolle der Software AG. Die Software AG ist nicht verantwortlich für die Inhalte dieser oder weiterführender Web-Sites. Die Software AG bietet diese Links lediglich als einen Service für Sie an und durch die Einrichtung eines Links stimmt die Software AG nicht automatisch dem Inhalt der Web-Site zu.

"ARIS", "IDS", "ProcessWorld", "PPM",  und das Symbol  sind Marken oder eingetragene Marken der Software AG in Deutschland und in vielen anderen Ländern weltweit.

SAP NetWeaver ist eine Marke der SAP AG, Walldorf. Alle anderen Marken sind Eigentum ihrer jeweiligen Inhaber.

U.S. pat. D561,778, pat. D561,777, pat. D547,322, pat. D547,323, pat. D547,324

Der Funktionsumfang von ARIS hängt vom verwendeten Lizenz-Key ab.

Inhalt

1	Einführung.....	1
2	Architektur integrierter Informationssysteme (ARIS).....	3
2.1	Das Konzept der ARIS-Architektur.....	3
2.2	Die Beschreibungssichten.....	3
2.3	Die Beschreibungsebenen.....	7
3	Die Vorgangskettenanalyse.....	11
3.1	Beschreibung der betriebswirtschaftlichen Problemstellung.....	11
3.2	Das Vorgangskettendiagramm (VKD).....	12
4	Modellierung innerhalb der Sichten und Ebenen des ARIS-Konzeptes.....	14
4.1	Funktionssicht.....	14
4.1.1	Fachkonzept.....	14
4.1.2	DV-Konzept - Anwendungssystemtypdiagramm.....	23
4.1.3	Implementierung - Anwendungssystemtypdiagramm.....	28
4.2	Datensicht.....	31
4.2.1	Fachkonzept.....	31
4.2.2	DV-Konzept.....	66
4.2.3	Implementierung - Tabellendiagramm.....	71
4.3	Organisationssicht.....	74
4.3.1	Fachkonzept.....	74
4.3.2	DV-Konzept - Netztopologie.....	84
4.3.3	Implementierung.....	86
4.4	Prozesssicht/Steuerungssicht.....	91
4.4.1	Fachkonzept.....	91
4.4.2	DV-Konzept.....	143
4.4.3	Implementierung - Zugriffsdiagramm (phys.).....	151
4.5	Leistungsmodellierung.....	157
4.5.1	Leistungsaustauschdiagramm.....	158
4.5.2	Leistungsbaum.....	159
4.5.3	Produktzuordnungsdiagramm.....	160
4.5.4	Produktbaum.....	163
4.5.5	Produktauswahlmatrix.....	164
4.5.6	Wettbewerbskräftemodell.....	165
5	Unified Modeling Language in ARIS.....	166
5.1	Einleitung.....	166
5.2	ARIS UML Designer - Unterstützter UML-Standard.....	166
6	Verwendung der Modellierungsmethode Object Modeling Technique (OMT).....	167
6.1	Einleitung.....	167
6.2	Zusammenfassung der OMT-Methodologie.....	167
6.3	Verwendung der OMT-Diagrammtechniken in ARIS.....	168
6.3.1	OMT Object Model.....	168
6.3.2	OMT Dynamic Model.....	175
6.3.3	OMT Functional Model.....	177
6.3.4	Mögliche Hierarchisierungen von Objekten.....	180

7	Methoden für das Wissensmanagement	181
7.1	Einleitung	181
7.2	Objekttypen zur Modellierung der Wissensverarbeitung	182
7.2.1	Wissenskategorie	182
7.2.2	Dokumentiertes Wissen	184
7.3	Modelltypen zur Modellierung der Wissensverarbeitung	185
7.3.1	Wissensstrukturdiagramm	185
7.3.2	Wissenslandkarte	186
7.3.3	Darstellung der Wissensverarbeitung in Geschäftsprozessen	188
8	Anwendungsfälle	189
8.1	Allgemeine Unternehmensdokumentation	192
8.2	Datenbankmanagement/Data Warehousing	193
8.3	Groupware	194
8.4	PC-Hardware- und Netzwerkmanagement	195
8.5	Prozesskostenrechnung	196
8.6	Qualitätsmanagement	197
8.7	Reorganisationsmaßnahmen	198
8.8	SAP R/3-Einführung	199
8.9	Software-Entwicklung und -einführung	200
8.10	Wissensmanagement	201
8.11	Workflow-Management	202
9	Balanced Scorecard-Methode	203
9.1	Einleitung	203
9.2	Das Konzept der Balanced Scorecard	204
9.2.1	Kernaussagen des BSC-Ansatzes	204
9.2.2	Der strategische Managementprozess und die Balanced Scorecard	205
9.2.3	Vorteile und Nutzen der Balanced Scorecard	211
9.3	Entwicklung einer Balanced Scorecard mit ARIS BSC	212
9.3.1	Begriffe und Abkürzungen	212
9.3.2	Aufbau der Balanced Scorecards mit ARIS BSC	214
9.3.3	Beziehungen zu anderen Modellen	221
10	eBusiness Scenario Diagram	222
10.1	Einleitung	222
10.2	Die Methode des eBusiness Scenario Diagrams	224
10.2.1	Die Idee	224
10.2.2	Das Modell und seine Objekte	224
10.2.3	Attributgruppe Übertragungsart	225
10.3	Auswertungen durch Reporte	226
10.3.1	Überprüfung der Datensicherheit	226
10.3.2	Systemunterstützung	226
10.3.3	Informationsfluss	226
10.3.4	Collaborative Business Maps	227

10.4	Verbindung zu anderen Methoden und Komponenten	227
11	IT City Planning	231
11.1	Enterprise Architecture und IT City Planning	231
11.2	An welche Unternehmen richtet sich IT City Planning?	231
11.3	IT City Planning mit ARIS	232
11.4	Die IS-Sicht	233
11.5	Servicetypen und ihre Daten	237
11.6	Detailbeschreibung der Servicetypen	238
11.7	Zeitlich-logische Abläufe zwischen IS-Elementen	239
11.8	Die IT-Sicht	239
11.9	IT-Elemente und ihre Daten	241
11.10	Detailbeschreibung der IT-Elemente	241
11.11	Organisatorische Aspekte	242
11.12	Zeitlich-logische Abläufe zwischen IT-Elementen	242
11.13	Zeitlich-logische Abläufe innerhalb der Architektur	243
11.14	Mögliche Auswertungen	244
12	Business Process Modeling	245
12.1	Die Beschreibungssprache BPMN	245
12.1.1	Prozessklassen und das Business Process Diagram	245
12.1.2	Realisierung der BPMN in ARIS	247
12.1.3	Die Elemente des Business Process Diagrams	248
13	Literaturverzeichnis	262
13.1	Allgemeine Literaturliste	262
13.2	Themenbezogene Literaturliste	263
13.2.1	Unified Modeling Language in ARIS	263
13.2.2	Verwendung der Modellierungsmethode Object Modeling Technique (OMT)	264
13.2.3	Methoden für das Wissensmanagement	264
13.2.4	Balanced Scorecard-Methode	264
13.2.5	IT City Planning	264
13.2.6	Business Process Modeling	264
14	Abbildungsverzeichnis	A
15	Index	i

1 Einführung

Die zunehmende Standardisierung sowie der enorme Preisverfall auf dem Hardware-Sektor führen zu einer Verlagerung der Aufgabenstellungen im Bereich der Informationssystem-Entwicklung.

Lagen früher die Optimierungspotentiale hauptsächlich bei der Systemgestaltung und -integration, orientierte man sich in den letzten Jahren stärker an den fachlichen Fragestellungen. Weil dezentrale Informationssysteme verfügbar wurden und mit integrierten Informationssystemlandschaften vernetzt werden konnten, eröffneten sich neue Einsparpotenziale im Hinblick auf die organisatorische Gestaltung der Unternehmen.

Indem Organisationsstrukturen funktional untergliedert und zentral orientiert wurden, weil sie sich meist an den beschränkten Möglichkeiten zentral strukturierter Host-Welten orientierten, wurden die Unternehmen unflexibel. Die neuen Potenziale, die sich durch die zunehmende Dezentralisierung von Rechnerleistungen und den damit einhergehenden neuen Architektur-Konzepten für Informationssysteme (Bsp.: Client-Server, Workflow-Management) auftaten, wurden anfangs nur selten erkannt oder nicht beachtet.

Heute hingegen sind sie, hervorgerufen durch den ständig wachsenden Marktdruck, Thema Nummer Eins für alle Unternehmen. Flexible Strukturen, die sich konsequente an den betrieblichen Geschäftsabläufen orientieren, werden für die Unternehmen zum entscheidenden Wettbewerbsfaktor. Aber nur eine ganzheitliche Sicht auf die Unternehmensgeschehnisse ermöglicht es, zusammenhängende Abläufe zu erkennen, zu straffen und durch optimal gestaltete Informationssystem-Landschaften zu unterstützen. Das Management der neuen Strukturen wird jedoch im Vergleich zur zentral orientierten Unternehmenswelt nicht einfacher, sondern um ein Vielfaches komplexer. Nur klare und einheitliche Definitionen der Verantwortlichkeiten, hohe Transparenz der Strukturen, eine einheitliche Kommunikationsbasis über alle Ebenen des Unternehmens hinweg und ein straffes, an den Unternehmenszielen orientiertes Projektmanagement können zum gewünschten Ziel führen.

Methoden der Unternehmensmodellierung unterstützen die Bewältigung dieser komplexen Aufgabe. Unternehmensmodelle sind die elementare Voraussetzung, um Geschäftsprozesse zu analysieren, Projekte an der ganzheitlichen Zielsetzung des Unternehmens zu orientieren und schließlich die Informationssystem-Landschaft als Verbund verteilter, integrierter Systeme optimal zur Unterstützung dieser schlanken Organisationsstrukturen einzusetzen.

In den Vordergrund der Diskussion rücken damit die Modellierung der Realität und die Betrachtung ganzheitlicher Geschäftsabläufe. Durch die unterschiedlichen Modellierungsmethoden nahm die Methodenvielfalt enorm zu, was zu Unübersichtlichkeit und Unsicherheit führte. Um das Dilemma zu beseitigen, wurden einheitliche Rahmenkonzepte (Architekturen) für die Entwicklungs- und Modellierungsmethoden definiert.

Eine dieser Architekturen ist die von Scheer entwickelte **Architektur integrierter Informationssysteme (ARIS®)** (vgl. Scheer, Architektur integrierter Informationssystem 1992).

Dieses Architektur-Konzept dient zum einen dazu, Methoden bewertbar und durch die Fokus-

sierung auf ihre Schwerpunkte einordenbar zu machen, zum anderen dient es aber auch als Orientierungsrahmen für komplexe Entwicklungsprojekte, da es durch seine Strukturierungselemente implizit ein Vorgehensmodell für die Entwicklung integrierter Informationssysteme enthält.

Eine solche Architektur führt natürlich auch zu einer Vereinheitlichung des Methodeneinsatzes. Deshalb wurden aufbauend auf dieser Architektur bereits vorhandene sowie neue Modellierungsmethoden zu einer ganzheitlichen Methode der Geschäftsprozessmodellierung zusammengeführt.

Darüber hinaus diene die ARIS-Architektur als Grundlage der ARIS Platform. Sie integriert Produkte wie ARIS Business Architect innerhalb einer Produktfamilie der Software AG. Die Produkte unterstützen Berater und Unternehmen bei der Erstellung, Analyse und Auswertung der Unternehmensprozesse im Sinne des Business Process Reengineering. Zur leichten Erfassung und Modellierung der Unternehmensprozesse in den Fachabteilungen bietet ARIS Business Designer die notwendigen Funktionalitäten.

Das vorliegende Handbuch gibt Ihnen eine erste Einführung in die Modellierungsmethoden. Weiterführend werden Lösungsansätze aufgezeigt und Methoden angesprochen, die den vollen Systemumfang der ARIS-Produktfamilie nutzen. Darüber hinaus unterstützt Sie dieses Handbuch auch dann, wenn Sie sich unabhängig von Fragestellungen des Tool-Einsatzes mit Modellierungsmethoden beschäftigen.

2 Architektur integrierter Informationssysteme (ARIS)

2.1 Das Konzept der ARIS-Architektur

Die Konzeption der **Architektur integrierter Informationssysteme (ARIS)** basiert auf einem Integrationskonzept, das aus einer ganzheitlichen Betrachtung von Unternehmensprozessen abgeleitet wird. Zur Herleitung der Architektur wird zunächst ein Modell für Unternehmensprozesse entwickelt, das alle wesentlichen Merkmale zur Beschreibung von Geschäftsprozessen beinhaltet. Die hohe Komplexität des dabei entstehenden Modells wird durch die Zerlegung in einzelne Sichten reduziert. Die Sichtenzerlegung ermöglicht es nun, die Beschreibung der Sichteninhalte durch spezielle, für diese Sicht geeignete Methoden durchzuführen, ohne jeweils die vielfältigen Beziehungen und Zusammenhänge zu den anderen Sichten berücksichtigen zu müssen. Danach werden die Beziehungen zwischen den Sichten aufgenommen und redundanzfrei zu einer Gesamtbetrachtung der Prozessketten zusammengeführt.

Der zweite Ansatz, der ebenfalls zur Reduktion der Komplexität beiträgt, ist die Betrachtung verschiedener Beschreibungen. Dabei werden einem Life-Cycle-Konzept folgend die unterschiedlichen Beschreibungsmethoden für Informationssysteme nach ihrer Nähe zur Informationstechnik unterschieden. Dadurch wird eine durchgängige Beschreibung von der betriebswirtschaftlichen Problemstellung bis hin zur technischen Umsetzung sichergestellt.

Das ARIS-Konzept bildet somit einen Rahmen, in dem integrierte Informationssysteme entwickelt, optimiert und ihre Umsetzung beschrieben werden können. Dabei führt vor allem die Betonung der fachlichen Beschreibungsebene dazu, dass das ARIS-Konzept als Orientierung bei der Erstellung, Analyse und Auswertung von betriebswirtschaftlichen Prozessketten dient. Eine genauere Beschreibung der Architektur integrierter Informationssysteme gibt Scheer (vgl. Scheer, *Architektur integrierter Informationssysteme*, 1992, und Scheer, *ARIS - Vom Geschäftsprozess zum Anwendungssystem*, 1998).

2.2 Die Beschreibungssichten

Ausgangspunkt der Betrachtung ist ein Unternehmensprozess, wie er in folgender Abbildung dargestellt ist.

Der Prozess wird durch das Ereignis **Kundenauftrag ist eingetroffen** ausgelöst. Dieses Ereignis stößt die Funktion (Vorgang) **Kundenauftrag annehmen** an. Zur Durchführung dieses Vorgangs werden Zustandsbeschreibungen der relevanten Umwelt des Vorgangs benötigt. Hier sind insbesondere Daten über den Kunden und die Artikel relevant. Die Vorgangsbearbeitung kann zu einer Zustandsänderung der Umweltobjekte führen, z. B. können die Lagerbestandsdaten der Artikel mit den neuen Reservierungsdaten aktualisiert werden.

Die Vorgänge werden durch Sachbearbeiter durchgeführt, die wiederum Abteilungen zugeordnet werden können. Die Abteilung greift zur Durchführung ihrer Aufgaben auf bestimmte Ressourcen der Informationstechnik (Personal Computer, Drucker etc.) zu.

Mit Abschluss des Vorgangs **Kundenauftrag annehmen** tritt das Ereignis **Auftrag ist bestätigt** ein, das nun weitere Vorgänge, z. B. **Auftrag verfolgen** oder **Produktionsplan erstellen**,

zur Folge hat. Das Objekt **Auftrag** befindet sich nun in einem neuen Zustand, denn aus dem Objekt **Eingetroffener Auftrag** ist ein **Bestätigter Auftrag** geworden. Durch das Ausführen der Funktion **Kundenauftrag annehmen** wurde eine Leistung erbracht, die bei der Bearbeitung der folgenden Vorgänge, zusammen mit personellen und sachlichen Ressourcen, als Input verwendet wird.

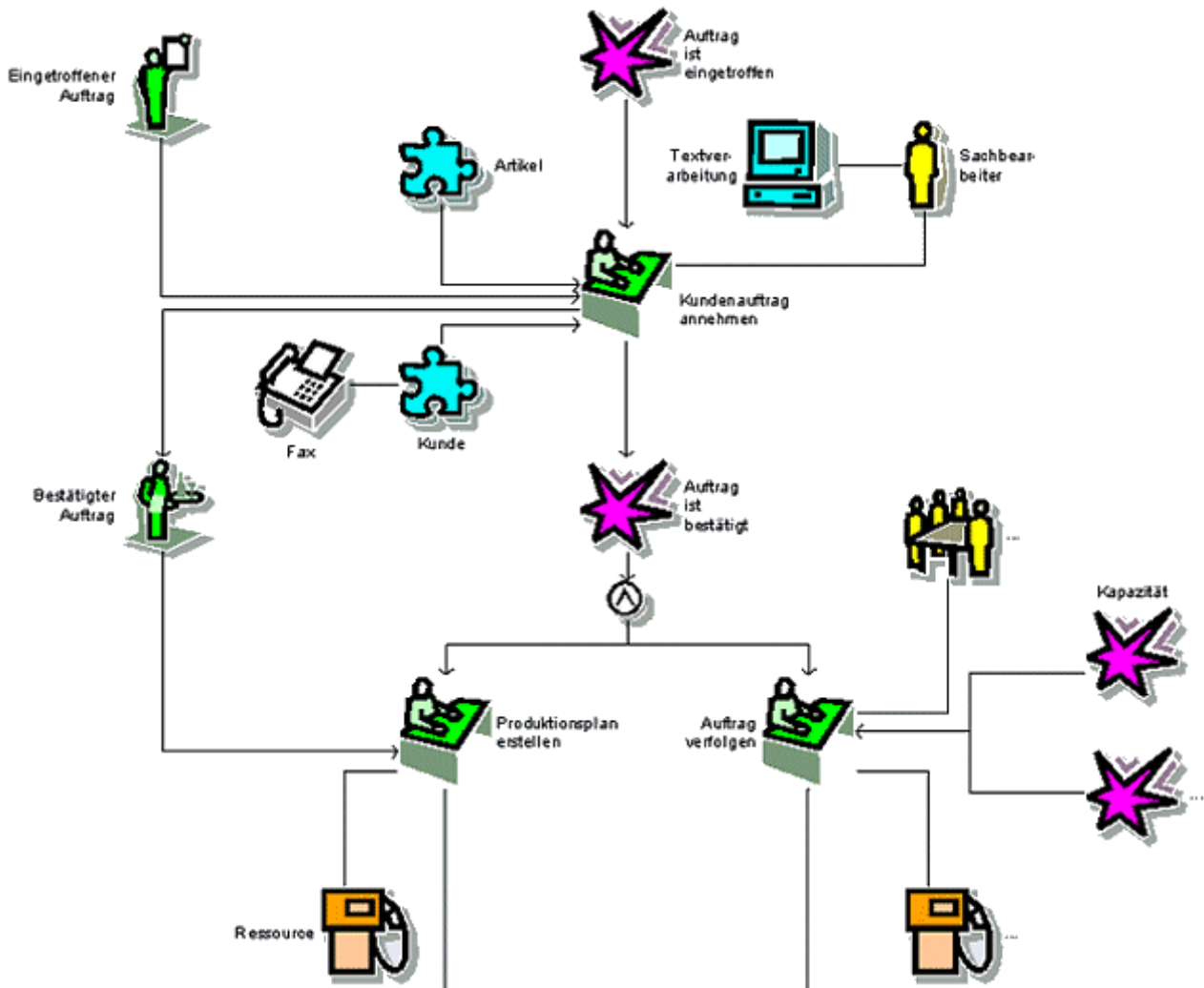


Abbildung 1: Modell eines Unternehmensprozesses

Die zur vollständigen Beschreibung eines Unternehmensprozesses benötigten Komponenten sind somit Vorgänge, Ereignisse, Leistungen (Zustände), Bearbeiter, Organisationseinheiten und Ressourcen der Informationstechnik. Müssten jeweils je betrachteten Vorgang alle Auswirkungen auf alle Elemente des Prozesses betrachtet werden, würde dies das Modell sehr komplizieren und auch zu Redundanzen in der Beschreibung führen.

Um diese Komplexität zu reduzieren, wird der Gesamtzusammenhang in einzelne Sichten zerlegt (vgl. nächste Abbildung), die jeweils eigene Modellierungs- und Entwurfsmethoden darstellen (vgl. Scheer, Architektur integrierter Informationssysteme 1992, S. 13 ff.). Diese können zunächst weitgehend unabhängig voneinander bearbeitet werden. Die Zerlegung der Sichten erfolgt

derart, dass die Beziehungen der Komponenten innerhalb einer Sicht sehr hoch sind, zwischen den Sichten jedoch nur eine relativ lose Kopplung besteht.

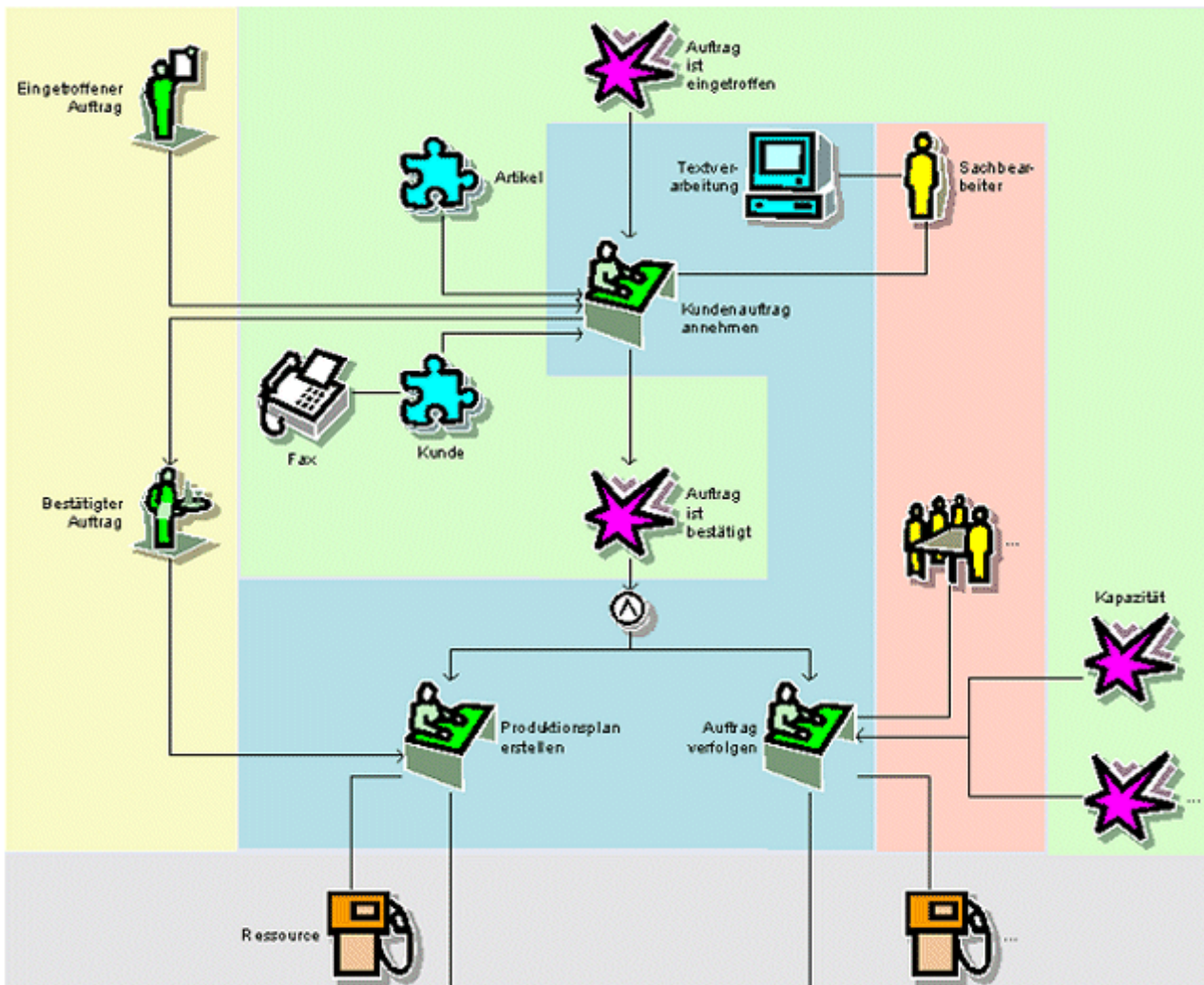


Abbildung 2: Sichten des Prozessmodells

Ereignisse wie **Kundenauftrag ist eingetroffen**, **Rechnung ist erstellt** definieren das Eintreten einer Zustandsänderung von Informationsobjekten (Daten). Sie werden in der Datensicht der ARIS-Architektur beschrieben.

Die Zustände im Umfeld der Objekte, z. B. im Umfeld des Auftrags, werden durch Leistungen repräsentiert. Eine Leistung kann entweder eine Sachleistung oder eine Dienstleistung sein. Dienstleistungen, die Informationen erzeugen und bereitstellen, sind Informationsdienstleistungen. Das Bereitstellen von finanziellen Mitteln gehört ebenfalls zu den Leistungen. Zusammenhänge zwischen Leistungen werden in der **Leistungssicht** der ARIS-Architektur beschrieben.

Die auszuführenden Funktionen (Vorgänge) sowie ihre Zusammenhänge untereinander bilden die zweite Sicht, die **Funktionssicht**. Sie beinhaltet die Beschreibung der Funktionen, die Aufzählung der einzelnen Teilfunktionen die zum Gesamtzusammenhang gehören, sowie die zwischen den Funktionen geltenden Anordnungsbeziehungen.

In der **Organisationssicht** werden die Mitarbeiter und Organisationseinheiten sowie deren Beziehungen und Strukturen zusammengefasst.

Die Ressourcen der Informationstechnik bilden ein weiteres Betrachtungsfeld, die **Ressourcensicht**. Diese ist aber für die fachliche Betrachtung eines Geschäftsprozesses nur insoweit von Bedeutung, als sie Rahmenbedingungen für die Beschreibung der anderen, stärker betriebswirtschaftlich ausgerichteten Komponenten bildet. Aus diesem Grunde werden die Komponentenbeschreibungen der anderen Sichten (Daten, Funktionen, Organisation) in Abhängigkeit von der Nähe zu den Ressourcen der Informationstechnik beschrieben. Die Ressourcen werden somit auf den Beschreibungen DV-Konzept und Implementierung der anderen Sichten behandelt (vgl. Kapitel **Die Beschreibungsebenen** (Seite 7)). Das durch die Ebenenbetrachtung definierte Life-Cycle-Modell ersetzt somit die Ressourcensicht als eigenes Beschreibungsfeld.

Mit der Zerlegung des Prozesses in einzelne Sichten wird zwar das Ziel der Komplexitätsreduzierung erreicht, allerdings gehen die Zusammenhänge der Prozesselemente zwischen den Sichten verloren. Aus diesem Grunde wird eine weitere Sicht, die **Steuerungssicht** aufgenommen, in der die Verbindungen zwischen den Sichten beschrieben werden. Die Aufnahme dieser Beziehungen in einer eigenen Sicht ermöglicht es, alle Beziehungen systematisch und redundanzfrei zu erfassen.

Die Steuerungssicht ist eine wesentliche Komponente von ARIS. Hierin liegt auch der hauptsächlichste Unterschied des ARIS-Konzeptes zu anderen Architekturvorschlägen (zum Vergleich mit anderen Architekturvorschlägen s. Scheer, Architektur integrierter Informationssysteme, S. 24 ff.).

Es ergeben sich somit insgesamt die fünf ARIS-Sichten, denen auch in den weiteren Methodenbeschreibungen gefolgt wird.

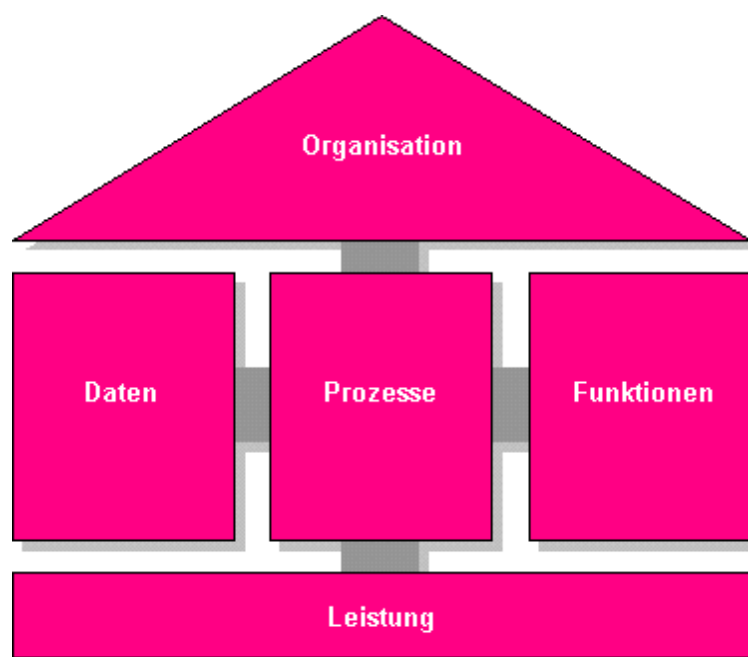


Abbildung 3: Zerlegungssichten des Prozessmodells

2.3 Die Beschreibungsebenen

Wie bereits erläutert wurde, wird die Ressourcensicht in ARIS durch ein Life-Cycle-Konzept der Beschreibungen eines Informationssystems aufgelöst.

Life-Cycle-Modelle in Form von Stufen- oder Phasenkonzepten beschreiben den Lebenslauf eines Informationssystems. Das Life-Cycle-Modell von ARIS hat jedoch nicht die Bedeutung eines Vorgehensmodells zur Entwicklung eines Informationssystems. Es definiert vielmehr unterschiedliche Beschreibungen, die sich in ihrer Nähe zur Informationstechnik unterscheiden.

Dabei unterscheidet ARIS die in der folgenden Abbildung gezeigte dreistufige Unterteilung (vgl. Scheer, Architektur integrierter Informationssysteme 1992, S. 16 f.).

Ausgangspunkt der Betrachtung ist die **betriebswirtschaftliche Problemstellung**. Die Beschreibung umfasst hier grobe Tatbestände, die sehr nahe an den fachlichen Zielsetzungen und der fachlichen Sprachwelt orientiert sind. Dabei werden auch die Möglichkeiten der Informationstechnik zur Unterstützung betriebswirtschaftlicher Prozesse und Entscheidungen mit einbezogen. Zur Darstellung werden deshalb auch nur halbformale Beschreibungsmethoden eingesetzt. Sie können aufgrund ihrer fehlenden Detailliertheit und des in der Fachwelt spezifischen Vokabulars nicht als Ausgangspunkt einer formalisierten Umsetzung in die Implementierung dienen.

Deshalb wird in einem **Fachkonzept** das zu unterstützende betriebswirtschaftliche Anwendungskonzept in einer soweit formalisierten Beschreibungssprache dargestellt, so dass es Ausgangspunkt einer konsistenten Umsetzung in die Informationstechnik sein kann. Dieser Vorgang wird auch als (semantische) Modellierung bezeichnet. Das Fachkonzept ist sehr eng mit der betriebswirtschaftlichen Problemstellung verbunden, wie es die Breite des Pfeils in der nächsten Abbildung ausdrückt.

Wird die Begriffswelt des Fachkonzeptes in die Kategorien der DV-Umsetzung übertragen, erreicht man die Ebene des **DV-Konzeptes**. Hier werden anstelle von fachlichen Funktionen die sie ausführenden Module oder Transaktionen definiert. Diese Ebene kann auch als Anpassung der Fachbeschreibung an generelle Beschreibungskonstrukte der Informationstechnik bezeichnet werden. Fachkonzept und DV-Konzept sind dabei nur lose gekoppelt. Ein DV-Konzept kann geändert werden, ohne dass dies Auswirkungen auf das Fachkonzept hat. Dies bedeutet jedoch nicht, dass Fachkonzept und DV-Konzept isoliert voneinander entwickelt werden können. Vielmehr soll nach Abschluss der fachkonzeptionellen Beschreibungen der betriebswirtschaftliche Inhalt so festgelegt sein, dass rein DV-bezogene Argumente wie z. B. das Leistungsverhalten eines Informationssystems keinen Einfluss auf die Fachinhalte nehmen.

Auf der Ebene der **Implementierung** wird das DV-Konzept auf konkrete hardware- und softwaretechnische Komponenten übertragen. Hiermit wird die Verbindung zur Informationstechnik hergestellt.

Die Beschreibungen sind durch unterschiedliche Änderungszyklen gekennzeichnet. Die Änderungsfrequenz ist auf der Ebene der Fachkonzepte am geringsten, auf der Ebene der Implementierung am höchsten.

Die Ebene der Implementierung ist eng an die Entwicklung der Informationstechnik gekoppelt und erfährt deshalb durch deren schnelle Innovationszyklen ebenfalls ständige Änderungen. Die Ebene des Fachkonzeptes ist von besonderer Bedeutung, da sie einmal Träger des langfristig geltenden betriebswirtschaftlichen Anwendungskonzeptes ist und gleichzeitig den Ausgangspunkt für die weiteren Generierungsschritte zur Umsetzung in die technische Implementierung darstellt. Fachkonzepte besitzen die größte Lebensdauer und dokumentieren durch ihre enge Nähe zur betriebswirtschaftlichen Problemstellung auch den fachlichen Nutzen des Informationssystems. Aus diesem Grunde steht die Sicht der Entwicklung von Fachkonzepten oder semantischen Modellen im Vordergrund. Die semantischen Modelle bilden die Verbindung zwischen dem Anwender und der ersten Umsetzung seiner Problemstellung in eine DV-nahe Sprache.

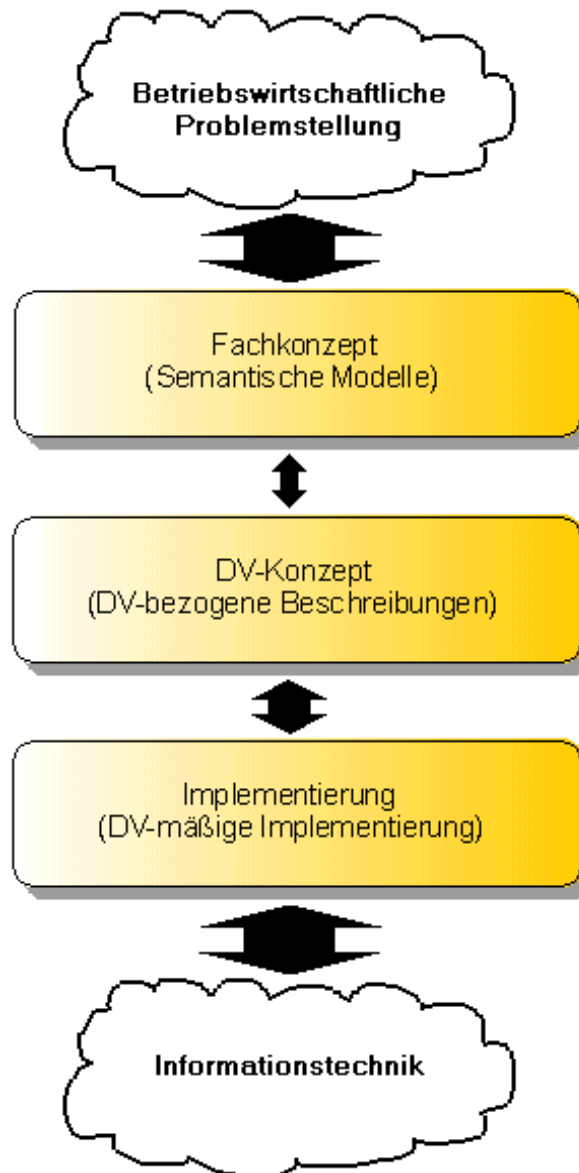


Abbildung 4: Beschreibungen eines Informationssystems

Mit der Sichtenbildung und den Beschreibungen einschließlich der betriebswirtschaftlichen Ausgangslösung ist das ARIS-Konzept entwickelt. Jede der Beschreibungssichten wird, wie in der

folgenden Abbildung gezeigt, in den drei Ebenen **Fachkonzept**, **DV-Konzept** und **Implementierung** beschrieben.

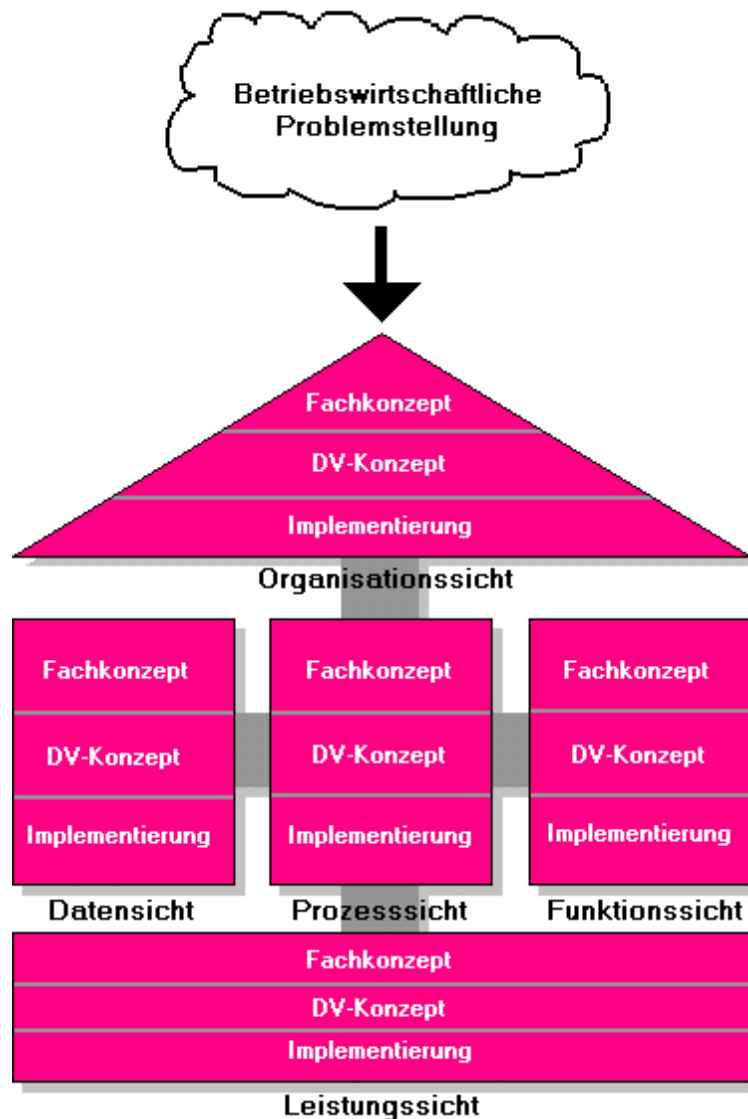


Abbildung 5: ARIS-Konzept

Mit dem entwickelten ARIS-Konzept liegen nun die Beschreibungsfelder, definiert durch die Beschreibungssichten und -ebenen der Architektur fest. Es ergeben sich somit einschließlich der als Ausgangspunkt der Betrachtung dienenden betriebswirtschaftlichen Problemstellung dreizehn Komponenten. Für jedes Betrachtungsfeld gilt es nun, die geeigneten Beschreibungsmethoden auszuwählen und zu erläutern.

Als Auswahlkriterien der Methoden gelten hierbei (vgl. Scheer, Wirtschaftsinformatik 1994, S. 18):

- die Einfachheit und Verständlichkeit der Darstellungsmittel,
- die Eignung für die speziell auszudrückenden Inhalte,
- die Möglichkeit, für alle darzustellenden Anwendungen einheitliche Methoden einsetzen zu können,

- der vorhandene oder zu erwartende Bekanntheitsgrad der Methoden sowie
- die weitgehende Unabhängigkeit der Methoden von technischen Entwicklungen der Informationstechnik.

Die einzelnen Methoden der Beschreibungsfelder werden nachfolgend beschrieben.

3 Die Vorgangskettenanalyse

3.1 Beschreibung der betriebswirtschaftlichen Problemstellung

Bevor die einzelnen Betrachtungsfelder innerhalb der ARIS-Architektur (Sichten und Ebenen) modelliert werden, muss der semantische Ausgangstatbestand, d. h. die betriebswirtschaftliche Problemstellung vorliegen. Beschrieben werden hierbei die Schwachstellen der gegenwärtig eingesetzten Informationssysteme in Bezug auf die Unterstützung der betrieblichen Geschäftsabläufe und die wesentlichen Inhalte des Soll-Konzepts des zu entwickelnden Systems, die auch die Ziele, die mit neuen Informationssystemen verfolgt werden, widerspiegeln.

Das zur Beschreibung dieser Problemstellung dienende Modell muss somit möglichst viele Tatbestände aus den Sichten Daten, Funktionen und Organisation einschließlich der zwischen ihnen bestehenden Zusammenhänge erfassen können. Darüber hinaus muss das Modell garantieren, soweit die Spezifizierung des Soll-Konzepts erlaubt, dass es als Ausgangspunkt der weiteren Modellierung dienen kann. Die Zerlegung in Sichten entsprechend des ARIS-Konzepts erfolgt somit erst mit der Entwicklung der Fachkonzepte.

Aufgrund der Forderung, die betriebswirtschaftliche Ausgangssituation zusammenhängend zu beschreiben und die Schwachstellen des gegenwärtigen Informationssystems in komprimierter Form darzustellen, ist der Einsatz von gängigen Modellierungsverfahren begrenzt. Diese sind aufgrund ihrer Darstellungsschwerpunkte vielmehr erst für die Modellierung einzelner Sichten einsetzbar.

Eine komprimierte Darstellung zur Erfassung dieser Zusammenhänge, die auch einen Überblick über das zu behandelnde Informationssystem gibt, ist das Vorgangskettendiagramm (VKD) (vgl. Scheer, EDV-orientierte Betriebswirtschaftslehre 1990, S. 39 f.).

3.2 Das Vorgangskettendiagramm (VKD)

In einem Vorgangskettendiagramm wird eine Vorgangskette geschlossen dargestellt. Dabei kommen alle herausgearbeiteten Sichten eines Geschäftsprozesses (Organisations-, Daten-, Funktions- und Ressourcensicht) mit ihren Beziehungen noch in zusammenhängender Form zum Ausdruck.

In folgender Abbildung wird ein Beispiel für die Vorgangskette der **Auftragsbearbeitung** gezeigt.

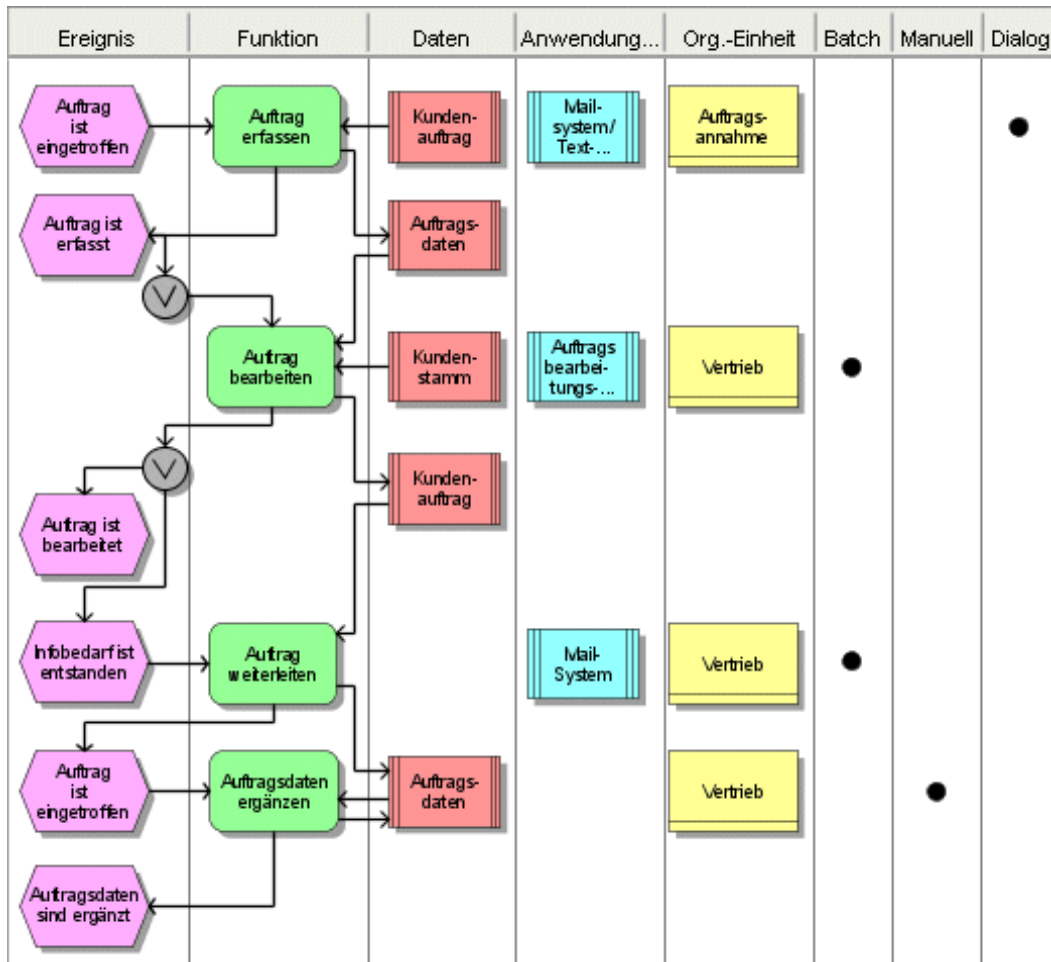


Abbildung 6: Beispiel für ein Vorgangskettendiagramm

Die beiden linken Spalten dienen zur Darstellung des zeitlich-logischen Ablaufs des betrachteten Geschäftsprozesses. Hierzu werden in der zweiten Spalte die einzelnen Funktionen des Vorgangs aufgeführt und mit den Ereignissen verknüpft, die sie aktivieren und erzeugen. Die Kanten zwischen Funktionen und Ereignissen definieren hierbei genau, durch welche Ereignisse Funktionen aktiviert und welche Ereignisse von Funktionen erzeugt werden; sie definieren somit den Kontrollfluss zwischen Funktionen. In dem gegebenen Beispiel wird die Funktion **Auftrag erfassen** durch das Ereignis **Auftrag ist eingetroffen** gestartet. Das Ergebnis dieser Funktion wird durch das Endereignis **Auftrag ist erfasst** definiert. Dieses Ereignis stößt die nächste Funktion **Auftrag bearbeiten** an. Durch diese Verknüpfung von Ereignissen und Funktionen

entsteht eine zeitlich-logische Ablauffolge von Funktionen, eine Prozesskette. Die logischen Abhängigkeiten möglicher Verzweigungen und Schleifen des Kontrollflusses können durch die Verwendung von Regeln berücksichtigt werden.

Die von den Funktionen benötigten Input- und Outputdaten werden in Form von Datenclustern in der nächsten Spalte dargestellt. Die Funktion **Auftrag bearbeiten** benötigt als Input-Daten die **Auftragsdaten** und **Kundenstammdaten**, sie erzeugt als Output-Datum den Kundenauftrag. Neben der reinen Darstellung der Informationsobjekte können auch die Informationsträger (Medien), auf denen sich die Informationen befinden, dargestellt werden. Es kann sich hierbei z. B. um ein Dokument, eine Liste, einen manuellen Beleg oder ein Speichermedium wie z. B. eine Platte handeln.

Die rechte Spalte dient der Festlegung der Organisationseinheiten (Abteilungen), die für die Ausführung der jeweiligen Funktion zuständig sind.

Die Spalten **Art der Bearbeitung** (dialog, batch, manuell) und **Anwendungssystem** geben zusätzliche Informationen über den Grad der DV-Unterstützung einer Funktion. In der Spalte **Anwendungssystem** können das eingesetzte Anwendungssystem oder Anwendungssystem-Komponenten eingetragen werden. Die Spalte **Art der Bearbeitung** dient zur näheren Spezifizierung der Funktionsausführung, wobei Dialog-, Batch- und manuelle Bearbeitung unterschieden werden.

In einem Vorgangskettendiagramm können bei der Analyse von Geschäftsprozessen, die eine vorliegende Ist-Situation beschreiben, Schwachstellen der gegenwärtigen Problemlösung aufgezeigt werden. Dies können sowohl Medienbrüche zwischen DV-bezogener und manueller Bearbeitung als auch organisatorische Brüche (häufiger Wechsel der verantwortlichen Abteilung/Organisationseinheit) sein. Insbesondere werden Datenredundanzen, Mehrfacherfassungen und Zeitverschleppungen innerhalb eines Ablaufs sichtbar, so dass eine Vielzahl von Anregungen für Verbesserungsmöglichkeiten des zu definierenden Soll-Ablaufs ableitbar sind.

Zur Beschreibung der Ausgangssituation werden Vorgangskettendiagramme auf einer relativ hohen Verdichtungsebene erstellt. Da sie vor allen Dingen zur Darstellung des Zusammenwirkens aller ARIS-Komponenten eingesetzt werden, dienen sie auch als Darstellungsmittel im Rahmen der Steuerungssicht von ARIS (vgl. hierzu Kapitel **Prozesssicht/Steuerungssicht** (Seite 91)). Neben den Vorgangskettendiagrammen werden in der Steuerungssicht auch ereignisgesteuerte Prozessketten (EPK) angeboten (vgl. Kapitel **Ereignissteuerung - ereignisgesteuerte Prozesskette (EPK)** (Seite 93)). Ereignisgesteuerte Prozessketten bieten die gleichen Modellierungsmöglichkeiten wie VKDs, sehen jedoch als Freiformdarstellung von einer Positionierung der Objekte in vordefinierten Spalten ab. Soll das Vorgehensmodell durchgehend nur durch einen Modelltyp (VKD oder EPK) unterstützt werden, kann der Soll-Ablauf auch in Form einer EPK dargestellt werden.

Die Beschreibung der weiteren Modellierungsmethoden folgt dem ARIS-Konzept. Dabei werden zunächst die Sichten (Funktionssicht, Datensicht, Organisationssicht, Steuerungssicht) und innerhalb dieser Sichten die Beschreibungsebenen (Fachkonzept, DV-Konzept und Implementierung) behandelt.

4 Modellierung innerhalb der Sichten und Ebenen des ARIS-Konzeptes

4.1 Funktionssicht

4.1.1 Fachkonzept

Modellierungsmethoden betrachten Funktionen oft im Zusammenhang mit Objekten der anderen Beschreibungssichten von ARIS. So wird z. B. die Verbindung von Daten und Funktionen dargestellt, um den Transformationsprozess einer Funktion über die Input-/Output-Daten der Funktion zu spezifizieren.

Bei der ARIS-Architektur wird jedoch eine strikte Trennung der Betrachtungsfelder verfolgt (vgl. Scheer, Architektur integrierter Informationssysteme 1992, S. 62). Innerhalb der Funktionssicht werden somit nur solche Darstellungsmittel behandelt, die Verbindungen zwischen Funktionen aufzeigen. So werden z. B. Verbindungen zwischen Funktionen und Daten in der Prozesssicht von ARIS dargestellt.

Eine Funktion ist eine fachliche Aufgabe oder Tätigkeit an einem Objekt zur Unterstützung eines oder mehrerer Unternehmensziele (vgl. Scheer, Architektur integrierter Informationssysteme 1992, S. 63).

Funktionen werden durch Rechtecke mit abgerundeten Ecken dargestellt:

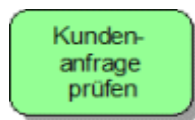


Abbildung 7: Darstellung der Funktion Kundenanfrage prüfen

Bildungskriterium für eine solche Funktion ist in der Regel ein Informationsobjekt wie z. B. eine Kundenanfrage oder ein Fertigungsauftrag. Dies sollte auch bei der Bezeichnung einer Funktion zum Ausdruck kommen. In obiger Abbildung wird dies verdeutlicht. **Kundenanfrage** definiert das Objekt, **prüfen** definiert die Verrichtung, die an diesem Objekt durchgeführt wird. Auf hoher Ebene wird jedoch meist nur ein Substantiv als Funktionsbezeichnung verwendet (Beschaffungslogistik, Produktion, Vertrieb).

4.1.1.1 Funktionsbaum

Funktionen können auf unterschiedlichen Verdichtungsstufen beschrieben werden. Oberste Verdichtungsstufe sind Funktionsbündelungen in Form von Geschäftsprozessen oder Vorgangsketten. Beispiel hierfür ist die Bearbeitung eines Kundenauftrags von der Bearbeitung der Kundenanfrage bis zum Versand. Ein solcher Geschäftsprozess stellt somit eine komplexe Funktion dar, die zur Reduzierung der Komplexität in Teilfunktionen zerlegt werden kann. Der Begriff Funktion kann somit auf allen Hierarchieebenen verwendet werden. Oft werden jedoch auch Begriffe wie Vorgang, Prozess, Teilfunktion oder Elementarfunktion zur Verdeutlichung der Hierarchieebene verwendet.

Die Zerlegung der Funktionen kann über mehrere Hierarchieebenen erfolgen. In semantischen Funktionsbäumen stellen hierbei Elementarfunktionen die unterste Ebene dar.

Elementarfunktionen sind Funktionen, die betriebswirtschaftlich nicht mehr sinnvoll zerlegbar sind.

Zur Darstellung dieser Untergliederung werden Funktionsbäume oder Hierarchiemodelle verwendet.

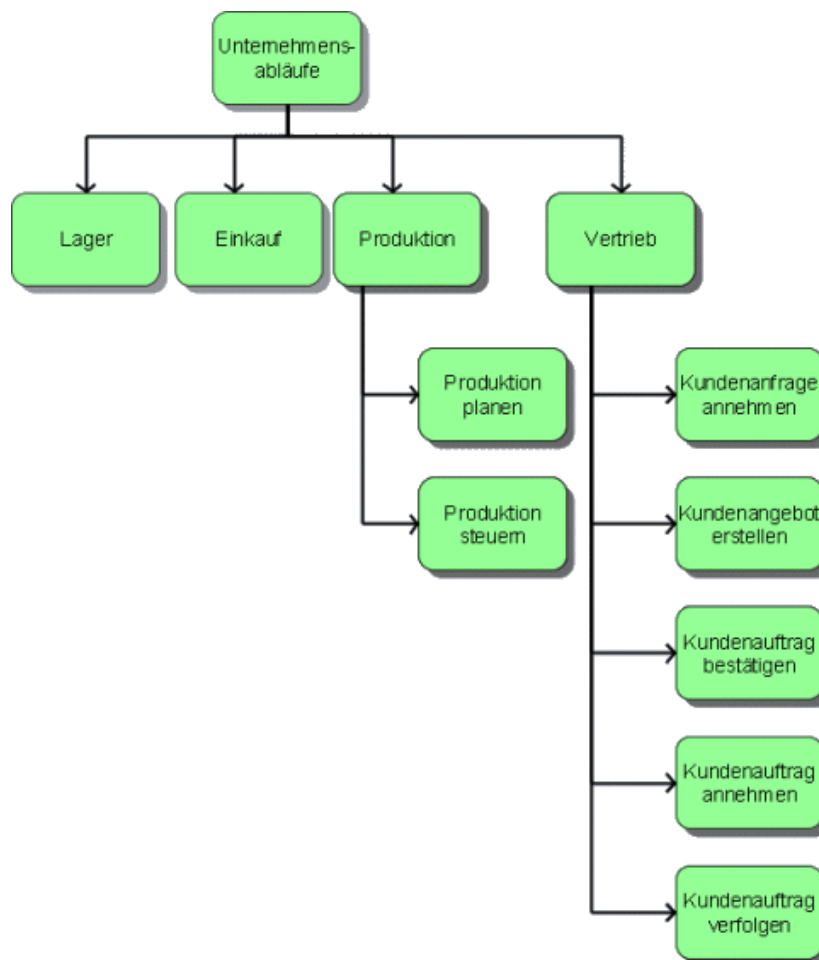


Abbildung 8: Funktionsbaum (Ausschnitt)

Die Gruppierung der Funktionen in einem Funktionsbaum kann nach verschiedenen Kriterien erfolgen (vgl. Brombacher/Bungert, Praxis der Unternehmensmodellierung 1992). Häufig verwendete Kriterien sind hierbei die Bearbeitung des gleichen Objekts (objektorientiert), die Zerlegung nach der Prozesszugehörigkeit (prozessorientiert) oder die Zusammenfassung von Funktionen nach gleichen Verrichtungen (verrichtungsorientiert).

Die nächste Abbildung zeigt das Beispiel einer objektorientierten Zerlegung. Die übergeordnete Funktion **Fertigungsauftrag bearbeiten** wird in die Funktionen **Fertigungsauftrag erstellen**, **Fertigungsauftrag rückmelden**, **Fertigungsauftrag ändern**, **Fertigungsauftrag stornieren**, **Fertigungsauftrag freigeben** und **Fertigungsauftrag überwachen** untergliedert. Die Funktionen beschreiben unterschiedliche Verrichtungen (erstellen, ändern, stornieren, ...), jedoch immer am gleichen Objekt **Fertigungsauftrag**.

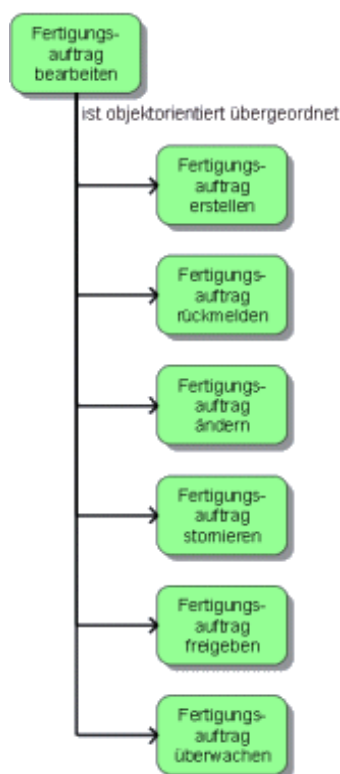


Abbildung 9: Funktionsbaum objektorientiert

Stellen die Funktionsbäume Ergebnisse der Modellierung von Geschäftsprozessen dar, bietet sich die Darstellung prozessorientierter Funktionsbäume an. Die folgende Abbildung zeigt das Beispiel einer prozessorientierten Funktionszerlegung.

Die Funktionen **Kundenauftrag annehmen**, **Kundenauftrag prüfen**, **Kundendaten anlegen**, **Kundenbonität prüfen**, **Produktverfügbarkeit prüfen** und **Kundenauftrag bestätigen** sind Funktionen des Geschäftsprozesses **Kundenauftrag bearbeiten**. Im Gegensatz zu einer objektorientierten Zerlegung werden hier Verrichtungen an verschiedenen Objekten durchgeführt (Kundenauftrag, Produktverfügbarkeit).

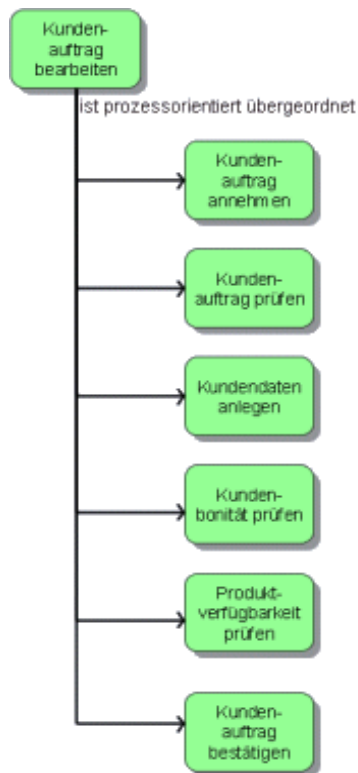


Abbildung 10: Funktionsbaum prozessorientiert

Verrichtungsorientierte Gruppierung bedeutet, dass alle Funktionen, die die gleiche Verrichtung (prüfen, anlegen, löschen) an unterschiedlichen Informationsobjekten durchführen, zusammengefasst werden. Ein Beispiel für die Verrichtung **Ändern** ist in der nächsten Abbildung dargestellt. Die gezeigten Funktionen können in unterschiedlichen Prozessen auftreten und bearbeiten auch unterschiedliche Objekte. Die Art der Verrichtung an den unterschiedlichen Objekten ist jedoch immer die gleiche.

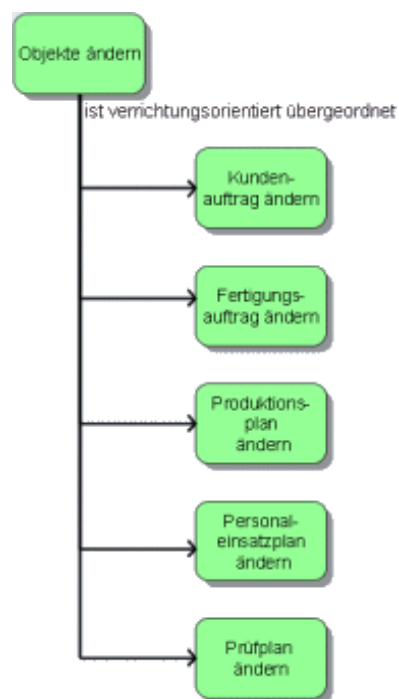


Abbildung 11: Funktionsbaum verrichtungsorientiert

Die Darstellung von Funktionen in einem Funktionsbaum dient zwar zur Reduzierung der Komplexität, ist jedoch statisch. Neben dieser statischen Darstellung kann aber auch die Ablauffolge im Sinne der zeitlichen Aufeinanderfolge von Funktionen von Interesse sein. Zur Darstellung zeitlich-logischer Ablauffolgen dienen so genannte ereignisgesteuerte Prozessketten (EPK). Diese beinhalten jedoch neben den Funktionen auch Ereignisse als Bindeglieder zwischen den Funktionen. Ereignisse sind der Datensicht von ARIS zuzuordnen. Der strikten Sichttrennung von ARIS folgend, werden ereignisgesteuerte Prozessketten in der Steuerungssicht von ARIS beschrieben (vgl. Kapitel **Fachkonzept** (Seite 14)).

Bei der fachkonzeptionellen Beschreibung von Funktionen sind neben der Eigenschaft der Funktionszergliederung in Teilfunktionen auch weitere Eigenschaften einer Funktion von Interesse; insbesondere solche Eigenschaften, die auf die Gestaltung der Geschäftsabläufe Einfluss nehmen.

So sollte für jede Funktion bereits grob definiert werden, ob ein Benutzereingriff notwendig ist oder die Funktion weitgehend automatisiert ablaufen kann. Gleichartige Funktionen, die ohne Benutzereingriff durchführbar sind, können somit gebündelt und geschlossen verarbeitet werden (Batch-Lauf).

Weitere Anhaltspunkte für die Neugestaltung der betrieblichen Abläufe bieten Informationen über das Mengengerüst einer Funktion (Bsp.: Anzahl der Anfragen, die an einem Tag bearbeitet werden) und die Gesamtdauer der Funktionsausführung. Die Gesamtdauer kann nochmals in Teilzeiten (Einarbeitungszeit, Bearbeitungszeit, Liegezeit) untergliedert werden. In ARIS können diese Informationen in den Attributen des Objekttyps **Funktion** abgelegt werden. Eine Auflistung der zur Verfügung stehenden Attributtypen befindet sich im Handbuch **ARIS-Methode – Tabellen** (Datei **method_tabulations_aris.pdf**) auf Ihrem Installationsmedium.

4.1.1.2 Y-Diagramm

Das Y-Diagramm dient zur Darstellung der Funktionen (Aufgaben) eines Unternehmens auf hoch aggregierter Ebene. Es handelt sich hierbei um umfassende Funktionalbereiche wie z. B. Produktentwurf, Materialwirtschaft, Instandhaltung. Die strukturierte Darstellung in Form des Y-CIM-Modells (vgl. Scheer, A.-W.: Wirtschaftsinformatik 1994, S. 87) drückt eine Klassifizierung der einzelnen Funktionen aus. Im linken Ast des Y ordnet Scheer die primär betriebswirtschaftlich-planerischen Funktionen der Produktionsplanung- und -steuerung an, im rechten die technisch orientierten Funktionen der Produktplanung und Produktrealisierung. Hierbei werden planerische Funktionen in den oberen Schenkeln des Y, Funktionen der Steuerung und Realisierung in den unteren Schenkeln des Y angeordnet.

Das Y-CIM-Modell stellt somit einen Ordnungsrahmen aller Funktionen eines Produktionsbetriebes dar.

In ARIS kann dieser Modelltyp für den funktionsorientierten Einstieg in komplexe Referenzmodelle verwendet werden. Die dargestellten Objekte sind vom Objekttyp **Funktion**. Dieser Objekttyp kann durch Hierarchisierung mit z. B. den Modelltypen **Funktionsbaum** und **EPK** verbunden werden.

Ein Beispiel zeigt diese Abbildung.

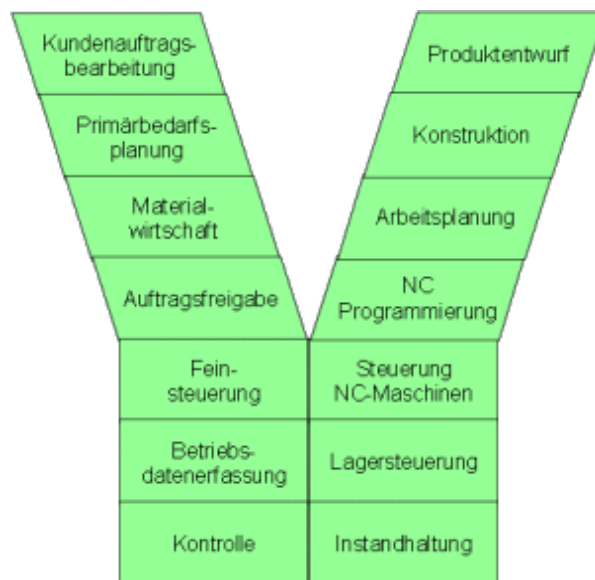


Abbildung 12: Y-Diagramm

4.1.1.3 SAP-Applikationendiagramm

Das SAP-Applikationendiagramm erlaubt einen an den Modulen des SAP-R/3-Anwendungssystems orientierten Einstieg in das SAP-R/3-Referenzmodell. Jedem Objekt dieses Modelltyps ist im R/3-Referenzmodell eine Prozessauswahlmatrix hinterlegt, die die in dem jeweiligen R/3-Modul zur Verfügung stehenden Hauptprozessen und den damit abbildbaren Prozessszenarien zeigt.

Das SAP-Applikationendiagramm des SAP-Systems ist in der folgenden Abbildung dargestellt.

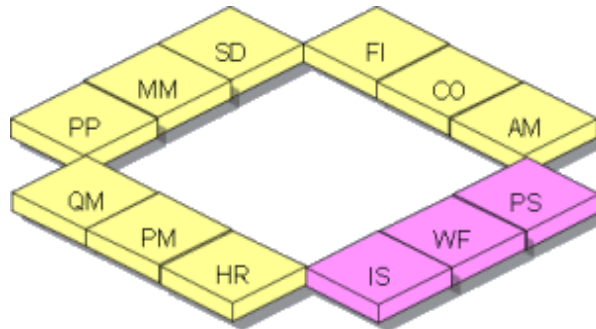


Abbildung 13: SAP-Applikationendiagramm

4.1.1.4 Zieldiagramm

Vor dem Beginn jeder Geschäftsprozessmodellierung, -analyse und -optimierung (Business Process Reengineering) sollte die Frage stehen, welche Ziele das Unternehmen mit der Geschäftsprozessmodellierung verfolgt.

Im Zieldiagramm können u. a. (Unternehmens-)Ziele definiert und Zielhierarchien aufgebaut werden.

Mit einem Ziel werden die zukünftigen Unternehmenszielsetzungen definiert, die durch Unterstützung der Erfolgsfaktoren und Realisierung neuer Geschäftsprozesse erreicht werden sollen.

Die möglichen Erfolgsfaktoren zur Zielerreichung können festgelegt, hierarchisiert und den Zielen, deren Erreichung sie unterstützen, zugeordnet werden.

Erfolgsfaktoren spezifizieren die zur Erreichung des jeweiligen Unternehmenszieles zu beachtenden Aspekte. Sie werden im Zieldiagramm den Unternehmenszielen zugeordnet.

Die Verbindung dieses Modelltyps zu den anderen Modelltypen des Fachkonzepts erfolgt über den Objekttyp **Funktion**. Für jedes Ziel kann dargestellt werden, welche(r) Funktion (Geschäftsprozess) des Unternehmens die Erreichung des Ziels unterstützt. In der Phase der Geschäftsprozessmodellierung und -optimierung sollten bei der Festlegung des Vorgehensmodells die hier festgelegte Zielpriorisierung und die zugeordneten Funktionen berücksichtigt werden.

Ein Beispiel für ein Zieldiagramm zeigt diese Abbildung.

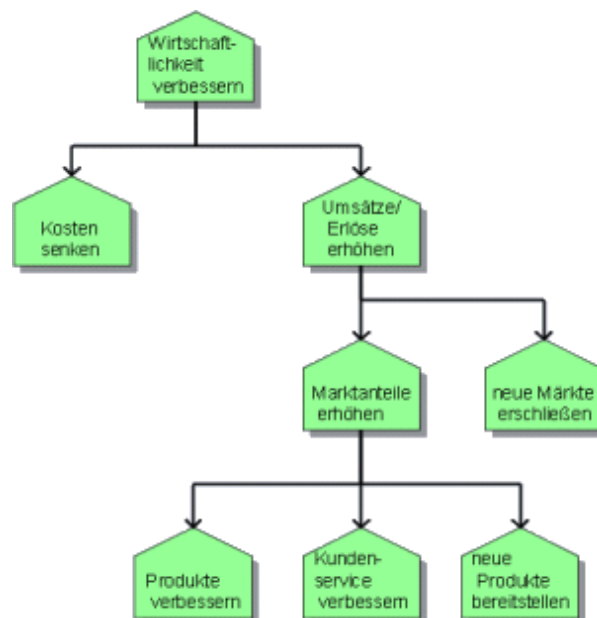


Abbildung 14: Zieldiagramm

4.1.2 DV-Konzept - Anwendungssystemtypdiagramm

Das DV-Konzept der Funktionssicht beinhaltet die Festlegung der Anwendungssystem- und Modultypen, den modularen Aufbau des Anwendungssystemtyps, den Entwurf einzelner Transaktionsschritte und die Definition der Ein- und Ausgabepräsentationen in Form von Listen- und Maskenentwürfen.

Zentrale Fragestellungen, die mit der Festlegung des DV-Konzepts der Funktionssicht beantwortet werden, sind:

- Wie können die im Fachkonzept definierten Funktionen durch Einsatz von Anwendungssystemtypen, Modultypen oder DV-Funktionen unterstützt werden?
- Wie sind Anwendungssystemtypen oder Modultypen modular aufgebaut?
- Welche Listen und Masken werden zur Durchführung einer Funktion benötigt?
- Welche Listen können mit einem Anwendungssystemtyp oder einem Modultyp erstellt werden und welche Masken verwenden Anwendungssystemtypen und Modultypen?
- Auf welcher technologischen Basis (Betriebssysteme, Benutzeroberflächen oder Datenbankmanagementsysteme) beruht ein Anwendungssystemtyp?
- Welche unternehmerischen Ziele werden beim Einsatz eines bestimmten Anwendungssystemtyps verfolgt?

Der zentrale Objekttyp des DV-Konzeptes der Funktionssicht ist somit der **Anwendungssystemtyp**.

Im Gegensatz zu einem konkreten Anwendungssystem, das erst auf der Implementierungsebene der Funktionssicht betrachtet wird und ein einzelnes z. B. durch eine Lizenznummer im Unternehmen identifizierbares Anwendungssystem darstellt, entsteht ein Anwendungssystemtyp durch Typisierung aller Anwendungssysteme, die sich auf exakt der gleichen technologischen Basis befinden.

Ein Anwendungssystemtyp stellt die Typisierung einzelner Anwendungssysteme dar, die sich auf exakt der gleichen technologischen Basis befinden.

Beispiel: ARIS Business Architect stellt einen Anwendungssystemtyp dar. Von diesem Anwendungssystemtyp kann man mehrere Lizenzen und damit mehrere einzelne Anwendungssysteme erwerben.

Anwendungssystemtypen werden grafisch wie folgt dargestellt:



Abbildung 15: Grafische Darstellung eines Anwendungssystemtyps

Anwendungssystemtypen sind meist modular aufgebaut. Das Anwendungssystemtypdiagramm ermöglicht die Darstellung dieses modularen Aufbaus. Die einzelnen Teile eines Anwendungssystemtyps sind Modultypen. Diese Abbildung zeigt hierzu ein Beispiel:

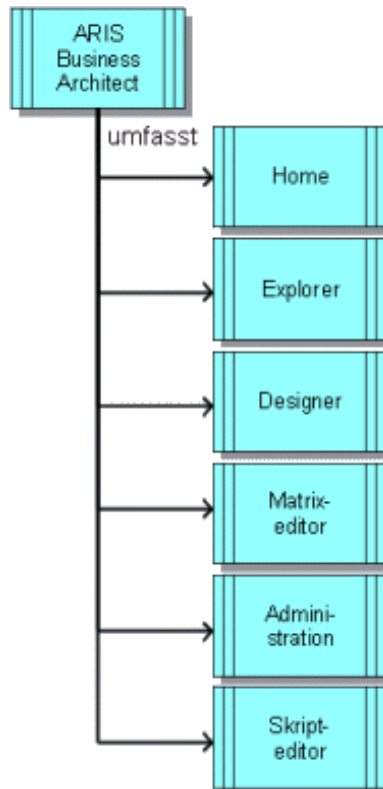


Abbildung 16: Modularer Aufbau eines Anwendungssystemtyps

ARIS Business Architect setzt sich in obigem Beispiel aus den Modultypen **Home**, **Explorer**, **Designer**, **Matrizen**, **Administration** und **Skripte** zusammen. Wie schon bei den Anwendungssystemtypen handelt es sich auch bei den Modultypen um die Typisierung einzelner Module, die sich auf exakt der gleichen technologischen Basis befinden. Modultypen sind Bestandteile von Anwendungssystemtypen. Sie stellen für sich alleine ablauffähige Komponenten dar.

Ein Modultyp ist ein für sich alleine ablauffähiger Bestandteil eines Anwendungssystemtyps. Modultypen stellen die Typisierung einzelner Module dar, die sich auf exakt der gleichen technologischen Basis befinden.

Anwendungssystemtypen und Modultypen sind beliebig hierarchisierbar. Modultypen können auf der untersten Stufe in DV-Funktionstypen unterteilt werden.

DV-Funktionstypen sind im Sinne einer Transaktion die kleinsten Einheiten eines Modultyps. Sie werden durch einzelne Programmteile realisiert und müssen zur Bearbeitung eines einzelnen Arbeitsschrittes immer vollständig durchgeführt werden.

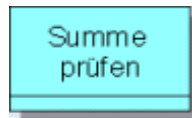


Abbildung 17: Grafische Darstellung des DV-Funktionstyps

Das Anwendungssystemtypdiagramm erlaubt auch die Festlegung der Funktionen des Fachkonzeptes, die durch die definierten Anwendungssystemtypen und Modultypen unterstützt werden. Diese Zuordnung bildet somit das Bindeglied zwischen dem Fachkonzept und dem DV-Konzept der Funktionssicht. Ein Beispiel hierzu wird in der folgenden Abbildung gezeigt.

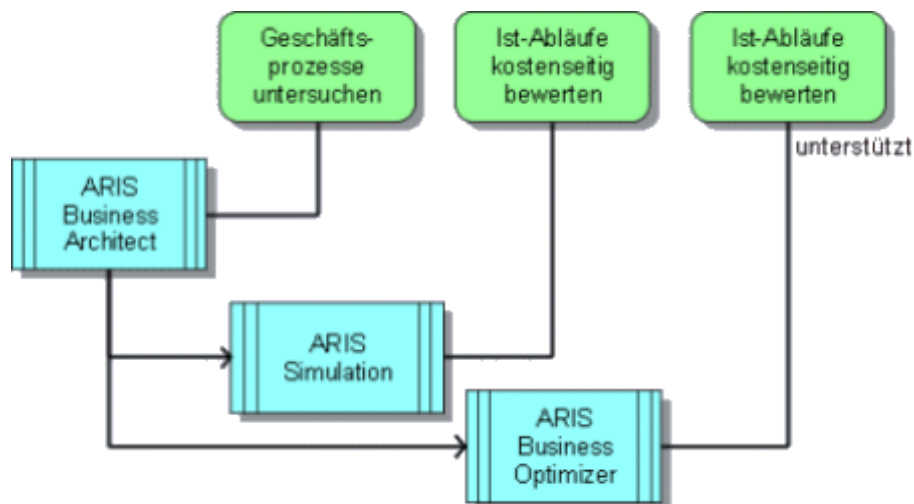


Abbildung 18: Zuordnung der Funktionen zu Anwendungssystemtypen

Zur genaueren Bestimmung der technologischen Basis von Anwendungssystemtypen und Modultypen können diesen auch die möglichen Typen von Benutzeroberflächen, Datenbankmanagementsystemen und Betriebssystemen, unter denen sie laufen, sowie die Programmiersprachen, mit denen sie realisiert sind, zugeordnet werden. Da es sich um Typen und nicht um konkrete Exemplare handelt, ist eine Mehrfachbeziehung durchaus möglich. So kann einem Anwendungssystemtyp die Benutzeroberfläche **Windows XP** und **Windows 7** zugeordnet sein, was bedeutet, dass die Version des Anwendungssystemtyps unter beiden Benutzeroberflächen lauffähig ist. Erst bei der Zuordnung der Benutzeroberfläche zu einem konkreten Exemplar (also einem Anwendungssystem) in der Implementierungsebene der Funktionssicht ist eine eindeutige Beziehung notwendig. Diese beschreibt dann genau die Konfigurierung der einzelnen Lizenz des Anwendungssystemtyps, die vom Unternehmen erworben wurde.

Ein Beispiel für die möglichen Zuordnungen im Anwendungssystemtypdiagramm zeigt diese Abbildung.

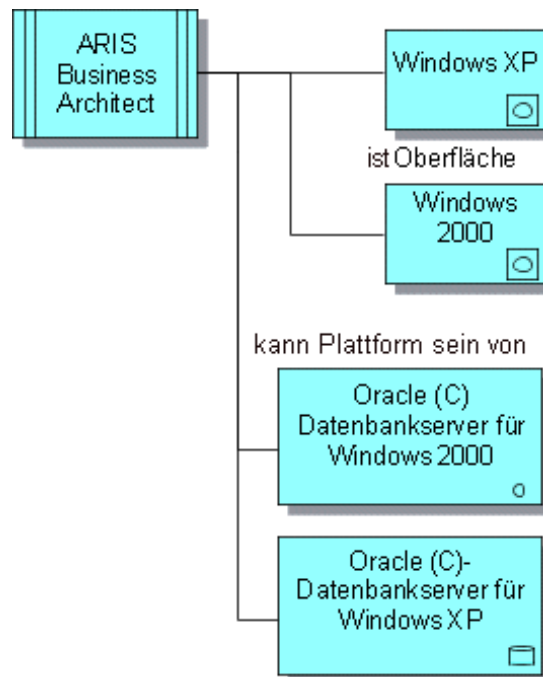


Abbildung 19: Konfiguration des Anwendungssystemtyps

Die Bearbeitung einer fachlichen Funktion mit Unterstützung eines Anwendungssystems führt zur Benutzung unterschiedlicher Bildschirmmasken und zur Erzeugung oder Verwendung unterschiedlicher Listen, die vom entsprechenden Anwendungssystem angeboten werden. Zur Darstellung dieses Sachverhalts stehen die Objekte **Liste** und **Maske** zur Verfügung, die entweder der fachlichen Funktion oder den Anwendungssystemtypen und Modultypen zugeordnet werden können.

Sollen zunächst allgemeine Ablauffolgen definiert werden, ohne Bezug auf konkrete Anwendungssystemtypen, so können zur Spezifikation der benötigten Masken und Listen auch die Objekte **Listenentwurf** und **Maskenentwurf** verwendet werden. Beide Objekttypen spezifizieren zunächst allgemein, welche Art von Liste oder Maske verwendet werden soll (Beispiel **Eingabe Kundendaten**) ohne einen konkreten Bezug zu Listen oder Masken eines Anwendungssystemtyps herzustellen. Danach können diese Listen- und Maskenentwürfe mit konkreten Listen und Masken verbunden werden. Die Zuordnung definiert dabei die zur Verfügung stehenden Realisierungsmöglichkeiten. Ein Beispiel hierzu zeigt folgende Abbildung.

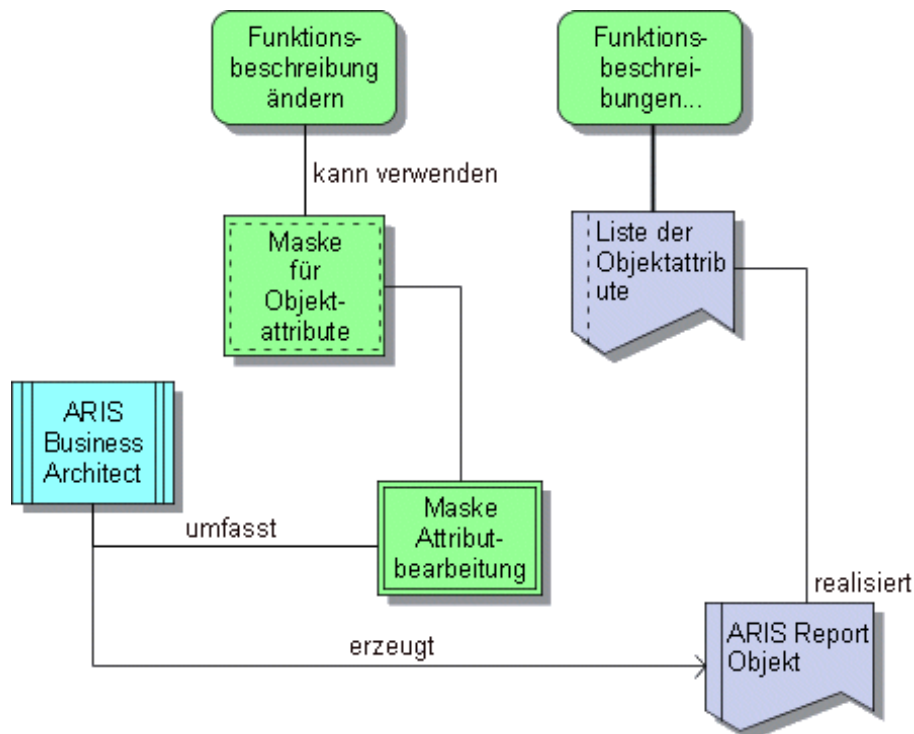


Abbildung 20: Masken- und Listenzuordnungen

Eine vollständige Zusammenstellung der Objekttypen des Anwendungssystemtypdiagramms und ihrer möglichen Beziehungen befindet sich im Handbuch **ARIS-Methode – Tabellen** (Datei **method_tabulations_aris.pdf**) auf Ihrem Installationsmedium.

4.1.3 Implementierung - Anwendungssystemtypdiagramm

Im Anwendungssystemtypdiagramm können den im DV-Konzept beschriebenen Anwendungssystem- und Modultypen die konkreten Anwendungssysteme und Module zugeordnet werden. Es handelt es sich hierbei um die im Unternehmen vorhandenen Exemplare eines Anwendungssystemtyps, die z. B. durch die Lizenznummern eindeutig identifizierbar sind.

Ein Anwendungssystem (Modul) ist ein einzelnes Exemplar eines Anwendungssystemtyps (Modultyps) das z. B. durch die Lizenznummer genau identifiziert werden kann.

Grafisch werden Anwendungssysteme und Module wie folgt dargestellt.

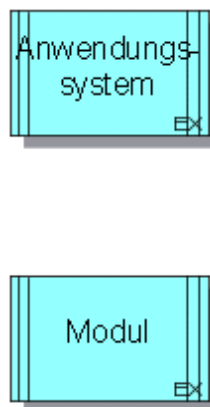


Abbildung 21: Grafische Darstellung des Anwendungssystems und des Moduls

Da von einem Anwendungssystemtyp (Modultyp) mehrere Lizenzen in einem Unternehmen vorhanden sein können, können somit auch im Anwendungssystemtypdiagramm mehrere Anwendungssysteme (Module) einem Anwendungssystemtyp (Modultyp) zugeordnet werden.

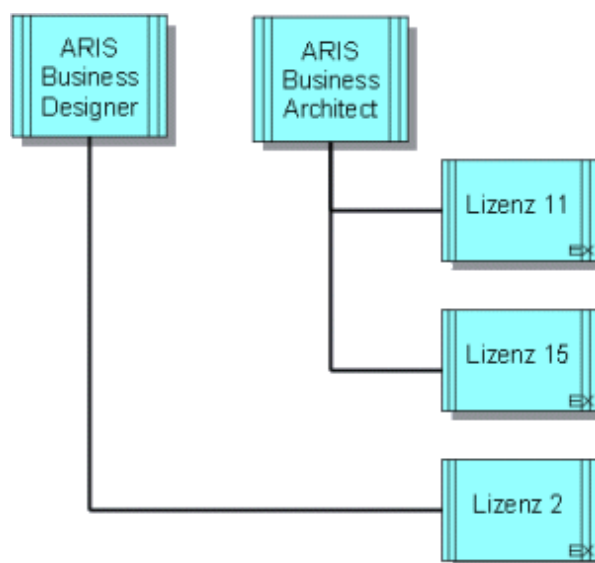


Abbildung 22: Zuordnung der Anwendungssysteme zu ihren Anwendungssystemtypen

Im Anwendungssystemtypdiagramm wird der tatsächliche modulare Aufbau eines Anwendungssystems dargestellt. Wurden im DV-Konzept alle möglichen modularen Bestandteile eines Anwendungssystemtyps aufgezeigt, liegt nun eine einzelne Lizenz des Anwendungssystems vor,

so dass die modularen Bestandteile jeder einzelnen Lizenz eindeutig definierbar sind. So können im Unternehmen von einem Anwendungssystemtyp mehrere Anwendungssysteme mit völlig unterschiedlichem modularem Aufbau vorhanden sein.

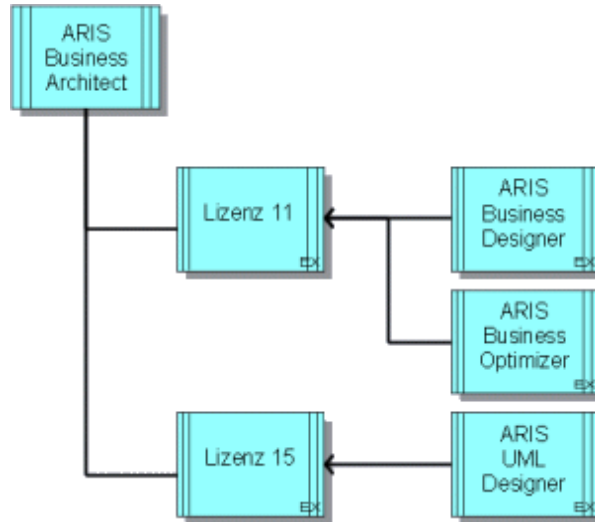


Abbildung 23: Unterschiedlicher modularer Aufbau zweier Anwendungssysteme des gleichen Typs

Neben der Darstellung der tatsächlich vorhandenen Anwendungssysteme und Module erlaubt die Implementierungsebene auch die Definition der programmtechnischen (physikalischen) Umsetzung der Anwendungssysteme in Form einzelner Programmdateien.

Hierzu kann im Anwendungssystemtypdiagramm dargestellt werden, welche Programmteiltypen zur Realisierung eines Anwendungssystemtyps oder Modultyps benötigt werden.

Ein Programmteil ist jede durch den Kauf einer Lizenz erworbene Programmdatei auf einem Datenträger (z. B. eine EXE-Datei oder COM-Datei). Ein Programmteiltyp entsteht durch Typisierung von Programmteilen, die sich auf exakt der gleichen technologischen Basis befinden.

Die Zuordnung von Programmteiltypen zu einem Anwendungssystemtyp und von einzelnen Programmteilen zu Programmteiltypen zeigt die folgende Abbildung.

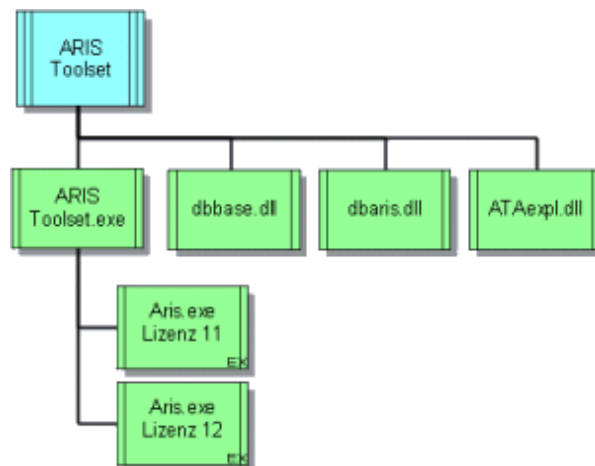


Abbildung 24: Zuordnung von Anwendungssystemtypen, Programmteiltypen und Programmteilen

Der Anwendungssystemtyp ARIS Toolset besteht u. a. aus den Programmteiltypen **ARIS Toolset.exe**, **dbbase.dll**, **abaris.dll** und **ATAexpl.dll**. Von jedem Programmteiltyp können im Unternehmen mehrere Exemplare (Programmteile) durch Erwerb mehrerer Lizenzen oder durch das Erstellen von Sicherungskopien vorhanden sein.

Programmteiltypen und Programmteile sind beliebig hierarchisierbar. Zur näheren technologischen Spezifikation des Programms können auch die Zugriffe von Programmteiltypen auf Programmbibliotheken im Anwendungssystemtypdiagramm dargestellt werden.

Eine Zusammenfassung der im Anwendungssystemtypdiagramm vorhandenen Objekttypen und Beziehungen befindet sich im Handbuch **ARIS-Methode – Tabellen** (Datei **method_tabulations_aris.pdf**) auf Ihrem Installationsmedium.

4.2 Datensicht

4.2.1 Fachkonzept

Das Fachkonzept der Datensicht beinhaltet die Beschreibung des semantischen Datenmodells des betrachteten Untersuchungsfeldes. Dem Zerlegungsprinzip von ARIS folgend werden hierbei sowohl die Objekte zur Spezifizierung der Start- und Endereignisse als auch die Zustandsbeschreibungen des relevanten Umfeldes einer Vorgangskette beschrieben.

Gegenüber der Funktionsmodellierung ist die Datenmodellierung methodisch besonders anspruchsvoll. In der Funktionssicht wird ausschließlich ein Objekt, die Funktion, betrachtet. Außerdem werden als Beziehungen zwischen Funktionen lediglich die Über- und Unterordnungen dargestellt.

Die am weitesten verbreitete Entwurfsmethode für semantische Datenmodelle ist das Entity-Relationship-Modell (ERM) von Chen (vgl. Chen, Entity-Relationship Model 1976). Bei dieser Modellierungsmethode werden vielfältige Begriffe wie Entitytyp, Beziehungstyp, Attribut usw. verwendet. Zwischen diesen Objekten bestehen zahlreiche und - im Vergleich zur Funktionsmodellierung - wesentlich schwieriger zu klassifizierende Beziehungen.

Im Folgenden wird die Modellierung mit Entity-Relationship-Modellen (ERM) vorgestellt. Dabei werden zunächst die Objekte und Beziehungen des Grundmodells nach Chen erläutert. Dieses Grundmodell wird dann im darauf folgenden Kapitel um einige Regeln erweitert.

4.2.1.1 Das ERM-Grundmodell

Das Grundmodell unterscheidet zwischen Entities, Attributen und Beziehungen. Hierbei kann grundsätzlich die Typebene von der Ausprägungsebene unterschieden werden.

Entities sind reale oder abstrakte Dinge, die für den betrachteten Ausschnitt der Aufgaben einer Unternehmung von Interesse sind.

Dieses Betrachtungsfeld kann z. B. ein Unternehmensprozess sein. Die interessierenden Datenobjekte sind dann entsprechend dem Zerlegungsmodell von ARIS die Objekte des Umfeldes und die Objekte, die Ereignisse spezifizieren. Beispiele für Entities eines Prozesses **Kundenauftragsbearbeitung** sind:

- Kunde 1235,
- Artikel 471,
- Auftrag 11.

Entities werden durch bestimmte Attribute (Eigenschaften) näher beschrieben. So kann z. B. ein Kunde über seinen Namen, Vornamen und die Anschrift näher spezifiziert werden.

Werden gleichartige Entities zu Mengen zusammengefasst, so werden diese als Entitytypen bezeichnet, deren einzelne Ausprägungen die Entities sind.

Die Gleichartigkeit von Entities drückt sich dadurch aus, dass sie durch gleiche Attribute beschrieben werden können. So wird der Kunde **Maier** und der Kunde **Müller** zu dem Entitytyp **Kunden** zusammengefasst, der Artikel **4710** und der Artikel **4712** zu dem Entitytyp **Artikel**. Entitytypen werden im ERM-Modell durch Rechtecke dargestellt (vgl. nächste Abbildung). Im Folgenden werden Entitytypen im Text immer durch Großschreibung gekennzeichnet.



Abbildung 25: Beispiele für Entitytypen

Attribute sind Eigenschaften, durch die Entitytypen beschrieben werden.

Attributausprägungen sind konkrete Werte von Attributen einzelner Entities. Z. B. kann der Kunde **1235** durch die Attributausprägungen **Maier**, **Heinz**, **München** etc. beschrieben werden. Die dazugehörigen Attribute lauten **Name**, **Vorname** und **Wohnort**.

Attribute werden meist durch ein Oval oder einen Kreis dargestellt. Im Folgenden wird als Darstellungsmittel das Oval verwendet. Die folgende Abbildung zeigt beispielhaft Attribute für den Entitytyp KUNDE.



Abbildung 26: Beispiele für Attribute des Entitytyps Kunden

Eine Unterscheidung zwischen Entitytypen und Attributen ist oft sehr schwierig und kann nur aus dem Kontext der Modellierung heraus getroffen werden. So können z. B. die Anschriften der Kunden auch als Entities aufgefasst werden und nicht als Attribute zu KUNDEN. In diesem Fall würde ein eigener Entitytyp ANSCHRIFT mit einer Beziehung zu KUNDEN modelliert werden. Ein hilfreiches Unterscheidungsmerkmal bei der Festlegung, ob es sich um einen Entitytyp oder ein Attribut handelt, ist hierbei, dass Entities Attribute besitzen, Attribute jedoch keine eigenen Attribute besitzen. Wird somit ein Attribut in einem ERM angelegt, das man später durch weitere Attribute beschreiben möchte, so wird es zum Entitytyp. Eine weitere hilfreiche Unterstützung bietet die Frage, ob man zu einem Objekt Beziehungen zu anderen Entitytypen herstellen möchte. Ist dies der Fall, so ist das betrachtete Objekt ebenfalls ein Entitytyp.

Eine Beziehung ist eine logische Verknüpfung zwischen Entities.

Beziehungen können damit nur existieren, wenn auch Entities existieren.

Werden gleichartige Beziehungen zu Mengen zusammengefasst, werden diese als Beziehungstypen bezeichnet.

Ein Beziehungstyp zwischen LIEFERANT und TEIL könnte z. B. LIEFERT heißen. Beziehungstypen werden im folgenden Text ebenfalls durch Großschreibung gekennzeichnet. Im ERM werden Beziehungstypen durch eine Raute dargestellt und über Kanten mit den Entitytypen verbunden (vgl. folgende Abbildung).

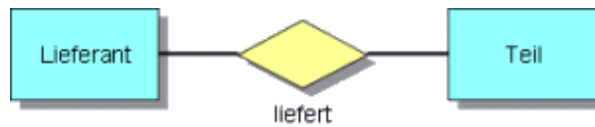


Abbildung 27: Beispiel für einen Beziehungstyp

Bei der Bezeichnung der Beziehungstypen kann oft nur eine Leserichtung zu sinnvollen Verbindungen führen. So soll im obigen Beispiel die Beziehung **Lieferant liefert Teil** ausgedrückt werden. In anderer Leserichtung wäre das Ergebnis **Teil liefert Lieferant**, was eindeutig nicht sinnvoll ist. Ist die Leserichtung nicht eindeutig bestimmbar, muss diese Schwierigkeit durch eine geschickte Wahl übergeordneter Begriffe umgangen werden.

Es werden mehrere Arten von Beziehungstypen unterschieden. Unterscheidungsmerkmale sind hierbei zum einen die Anzahl der durch sie verbundenen Entitytypen und zum anderen der Komplexitätsgrad einer Beziehung.

Bei der Anzahl der durch einen Beziehungstyp verbundenen Entitytypen können 1-, 2- oder n-stellige Beziehungen unterschieden werden.

Der Komplexitätsgrad oder die Kardinalität gibt an, wie viele Entities eines Entitytyps mit einem Entity des anderen Entitytyps in Beziehung stehen können.

Die dabei zu unterscheidenden Beziehungen werden in folgender Abbildung gezeigt (vgl. Scheer, Wirtschaftsinformatik 1994, S. 34).

Es ergeben sich somit vier verschiedene Beziehungsarten (Kardinalitäten):

- 1:1 - Beziehung,
- 1:n - Beziehung,
- n:1 - Beziehung und
- n:m - Beziehung.

Bei einer 1:1-Beziehung wird jeder Entity der ersten Menge genau einem Entity der zweiten Menge zugeordnet.

Eine 1:n-Beziehung drückt aus, dass jeder Entity der ersten Menge genau einem Entity der zweiten Menge zugeordnet wird, jeder Entity der zweiten Menge jedoch Beziehungen zu verschiedenen Entities der ersten Menge eingehen kann.

Die n:1-Beziehung drückt den gleichen Sachverhalt in umgekehrter Reihenfolge aus.

Werden jedem Entity der ersten Menge mehrere Entities der zweiten Menge zugeordnet und umgekehrt, so liegt eine n:m-Beziehung vor.

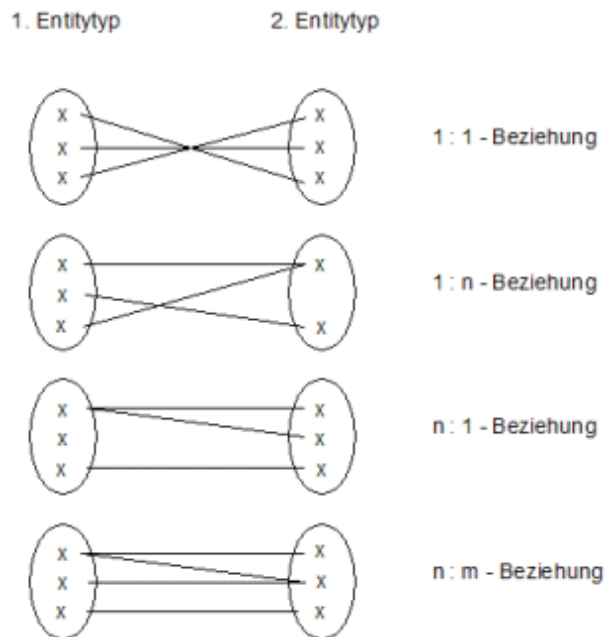


Abbildung 28: Kardinalitäten der Beziehungen zwischen zwei Entitytypen

Die Kardinalitäten des Beziehungstyps (Attributtyp **Komplexitätsgrad**) werden an die Kanten des Entity-Relationship-Modells geschrieben.

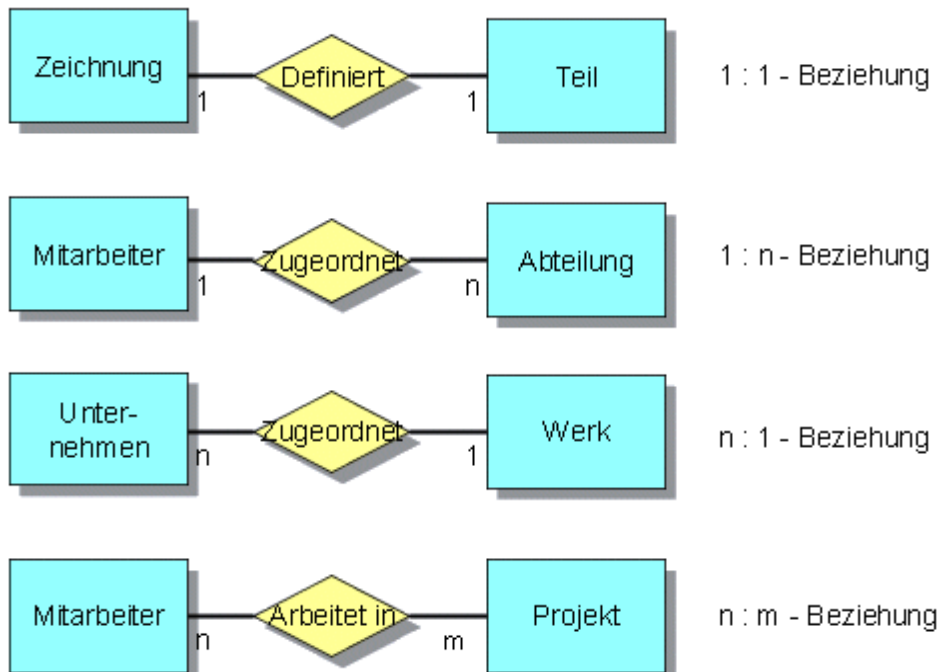


Abbildung 29: Darstellung der Kardinalitäten im ERM

Die Kardinalität gibt jeweils an, an wie vielen Beziehungen des Beziehungstyps ein Entity des Entitytyps maximal beteiligt sein kann. Im Beispiel der n:1-Beziehung der obigen bedeutet dies, dass ein Unternehmen des Entitytyps UNTERNEHMEN in mehreren Beziehungen ZUGEORDNET

vorkommen kann, da ein Unternehmen aus mehreren Werken besteht; ein konkretes Werk kann jedoch maximal an einer Beziehung ZUGEORDNET beteiligt sein, muss also einem Unternehmen eindeutig zugeordnet sein.

In der Originalarbeit von Chen werden die Kardinalitätsangaben in anderer Weise interpretiert. Die hier dargestellte Notation ermöglicht aber, insbesondere bei der Darstellung von Beziehungen zwischen mehreren Entitytypen, eindeutigere Formulierungen. Auf die Originalfassung von Chen wird an dieser Stelle daher nicht eingegangen, da dies nur zu unnötigen Verwirrungen führen würde.

Da auch Beziehungen zwischen Entities eines Entitytyps möglich sind, ist auch die Darstellung zweier paralleler Kanten zwischen einem Entitytyp und einem Beziehungstyp zulässig. Den Kanten können dann zur Unterscheidung Rollennamen vergeben werden. Ein Beispiel zur Darstellung rekursiver Beziehungen zeigt die folgende Abbildung. Ein übergeordnetes Teil besteht aus mehreren untergeordneten Teilen, ein untergeordnetes Teil kann aber auch als Komponente in mehreren übergeordneten Teilen verwendet werden.

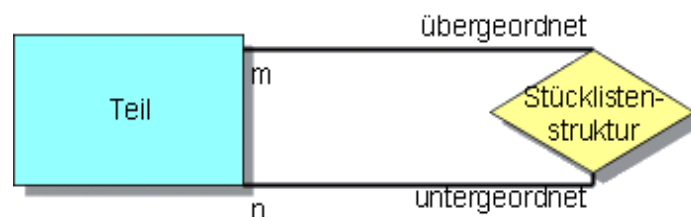


Abbildung 30: ERM für eine Stückliste

Sowohl Entitytypen als auch Beziehungstypen können durch Attribute beschrieben werden (vgl. folgende Abbildung).

Die Wertebereiche von Attributen werden als Domänen bezeichnet.

Die Zuordnungen von Elementen der Domäne zu Elementen der Entity- und Beziehungstypen stellen auch Beziehungen dar und können durch eine Kante dargestellt werden, die mit dem Namen gekennzeichnet ist.

Zwischen einem Entitytyp und mindestens einer Domäne muss eine 1:1-Beziehung bestehen. Die Werte dieser Domäne identifizieren die einzelnen Entities eindeutig und werden deshalb auch Schlüsselattribut des Entitytyps genannt.

Im Beispiel der nächste Abbildung (vgl. Scheer, Wirtschaftsinformatik 1994, S. 33) werden die Entities von KUNDE über das Schlüsselattribut **Kundennummer** eindeutig identifiziert.

Beziehungen werden durch die Verschmelzung der Schlüsselattribute der verbundenen Entities identifiziert. Schlüsselattribute des Beziehungstyps WOHNEN sind somit die Kundennummer und die Adressennummer.

Werte aus Domänen mit einer 1:n - Beziehung zu Entity- oder Beziehungstypen definieren die beschreibenden Attribute der jeweiligen Datenobjekte.

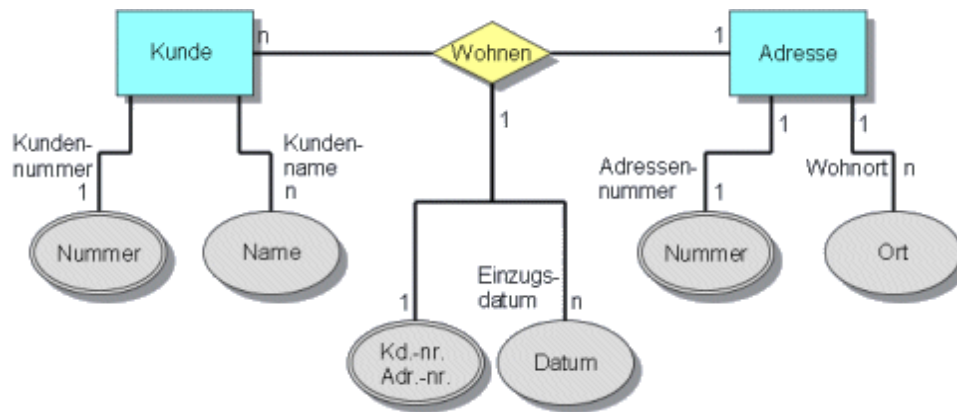


Abbildung 31: Zuordnung von Attributen im ERM

4.2.1.2 Erweiterungen des ERM - eERM

Das Grundmodell von Chen hat in den letzten Jahren zahlreiche Erweiterungen erfahren. Im Folgenden werden jedoch nur solche Erweiterungen angeführt, die für die Modellierung der Datensicht im ARIS-Konzept von Bedeutung sind.

4.2.1.2.1 Erweiterung um Konstruktionsoperatoren

Konstruktionsoperatoren bieten eine formale Hilfestellung bei der Erstellung eines Datenmodells. Sie sichern ab, dass Datenstrukturen nicht „willkürlich“ entstehen und geben dem Betrachter einer gegebenen Datenstruktur Hinweise über den Entwurfsprozess. Von eingeführten Begriffen ausgehend werden neue Begriffe durch Anwendung der Konstruktionsoperatoren erzeugt. Die Erstellung stellt dabei einen intellektuellen Prozess dar, der auf der Ebene des betriebswirtschaftlichen Fachwissens abläuft. Die Untersuchung der betriebswirtschaftlichen Tatbestände unter dem Gesichtspunkt der Datenstrukturen führt dabei entweder zur Strukturierung bekannter Tatbestände aus neuer Sicht oder auch zur Gewinnung neuer, bisher nicht berücksichtigter Zusammenhänge.

In der Vielzahl der unterschiedlichen Ansätze zur Erweiterung der ERM-Modellierung haben sich vier grundsätzliche Konstruktionsoperatoren durchgesetzt (vgl. Scheer, Wirtschaftsinformatik 1994, S. 35 ff):

- Klassifizierung,
- Generalisierung,
- Aggregation und
- Gruppierung.

Klassifizierung

Bei der Klassifizierung werden gleichartige Objekte (Entities) erkannt und einem Begriff (Entitytyp) zugeordnet. Ein Objekt ist mit einem anderen Objekt gleich, wenn es durch die gleichen Merkmale (Attribute) beschrieben wird.

Die Klassifizierung führt somit zur bereits beschriebenen Ermittlung von Entitytypen.



Abbildung 32: Klassifizierung von Kunden

Generalisierung/Spezialisierung

Bei der Generalisierung werden ähnliche Objekttypen zu einem übergreifenden Objekttyp zusammengefasst.

So werden in der nächsten Abbildung der Entitytyp **Kunde** und der Entitytyp **Lieferant** zu dem Gattungsbegriff **Geschäftspartner** generalisiert. Die Eigenschaften (beschrieben durch die Attribute), die beiden Ausgangsobjekten gemeinsam sind, werden auf den generalisierten Objekttyp übertragen. Damit müssen nur noch die Attribute beschrieben werden, in denen sich die Ausgangsobjekttypen unterscheiden. Die Bildung des neuen Entitytyps **Geschäftspartner** wird grafisch durch das Dreieck ausgedrückt, das auch als is-a-Beziehung bezeichnet wird.

Wird ein Gattungsbegriff in Teilbegriffe zerlegt (**Geschäftspartner** wird in **Kunde** und **Lieferant** aufgespalten), so liegt eine Spezialisierung vor.

Es handelt sich hierbei um die umgekehrte Form der Generalisierung. Die Eigenschaften des generalisierten Objektes werden auf die spezialisierten Objekte vererbt. Neben diesen vererbten Attributen können die spezialisierten Objekttypen auch eigene Attribute besitzen. Die Darstellung der Spezialisierung unterscheidet sich grafisch nicht von der Darstellung der Generalisierung.

Dies drückt sich in der Grafik auch dadurch aus, dass die Verbindungslinien nicht gerichtet dargestellt werden.

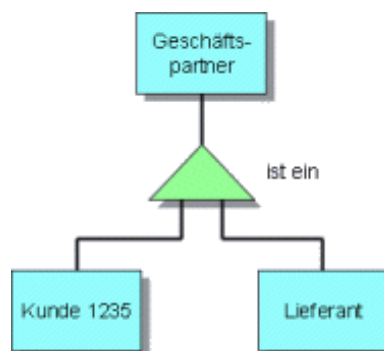


Abbildung 33: Generalisierung/Spezialisierung

Während die Spezialisierung hauptsächlich eine Top-down-Vorgehensweise bei der Erstellung eines Datenmodells unterstützt, wird die Generalisierung zur Unterstützung einer Bottom-up-Vorgehensweise eingesetzt.

Im Rahmen der Spezialisierung kann bei der Bildung von Teilmengen die Vollständigkeit und Disjunktheit (Alternative) der entstehenden Teilmengen spezifiziert werden.

Nicht-disjunkte Teilmengen liegen vor, wenn die Ausprägung eines Objektes sowohl in der einen Teilmenge als auch in der anderen Teilmenge vertreten sein kann. So kann im obigen Beispiel ein Kunde auch gleichzeitig Lieferant sein. Kann eine Ausprägung immer nur genau einer Teilmenge zugeordnet werden, so liegen disjunkte Mengen vor.

Eine vollständige Spezialisierung liegt dann vor, wenn für einen generalisierten Objekttyp alle möglichen spezialisierten Objekttypen eines Spezialisierungsgesichtspunktes aufgeführt werden. So kann z. B. der Entitytyp **Mensch** in die Entitytypen **Mann** und **Frau** spezialisiert werden (vgl.

folgende Abbildung). Die Spezialisierung unter dem Gesichtspunkt Geschlecht wäre damit vollständig.

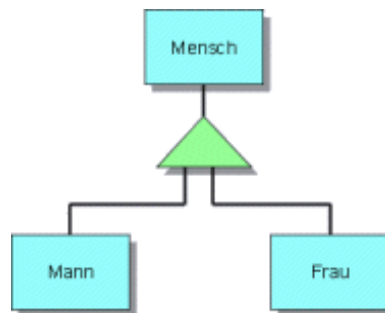


Abbildung 34: Vollständig beschriebene Spezialisierung

Die Kombinationen dieser Kriterien ergeben somit die folgenden vier Ausprägungen zur näheren Spezifizierung einer Generalisierung/ Spezialisierung:

- disjunkt/vollständig,
- disjunkt/nicht vollständig,
- nicht disjunkt/vollständig,
- nicht disjunkt/nicht vollständig.

Aggregation

Die Aggregation beschreibt die Bildung neuer Objekttypen durch die Zusammenfassung vorhandener Objekttypen. Der neue Objekttyp kann hierbei Träger neuer Eigenschaften sein.

Die Aggregation wird im ERM durch die Bildung von Beziehungstypen ausgedrückt (vgl. nächste Abbildung). Das neue Objekt **Auftragsarbeitsplan** entsteht durch Aggregation der Entitytypen **Fertigungsauftrag** und **Arbeitsplan**.



Abbildung 35: Beispiel für eine Aggregation

Der Aggregationsoperator kann aber auch auf Beziehungen angewendet werden. Ein dabei vorhandener Beziehungstyp wird dann als Entitytyp betrachtet und kann somit selbst wieder Ausgangspunkt für die Bildung neuer Beziehungen sein. Ein Beispiel hierfür wird in folgender Abbildung gezeigt.

Aus den Entitytypen **Fertigungsauftrag** und **Arbeitsplan** wurde durch eine erste Aggregation der Beziehungstyp **Auftragsarbeitsplan** gebildet. Die Schlüsselattribute **Fertigungsauftragsnummer** (FANR) und **Arbeitsplannummer** (APLNR) bilden den komplexen Schlüssel des Auftragsarbeitsplans. Dem Auftragsarbeitsplan können nun mehrere Arbeitsgänge zugeordnet werden. Aus diesem Grunde wird zwischen dem Beziehungstyp **Auftragsarbeitsplan** und dem Entitytyp **Arbeitsgang** die Beziehung **Auftragsarbeitsgang** gebildet. Da Beziehungen nur zwischen Entitytypen erzeugt werden können, muss der Beziehungstyp **Auftragsarbeitsplan**

Die nächste Abbildung zeigt die grafische Darstellung eines Datenclusters.

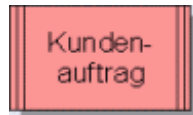


Abbildung 37: Datencluster (grafisches Symbol)

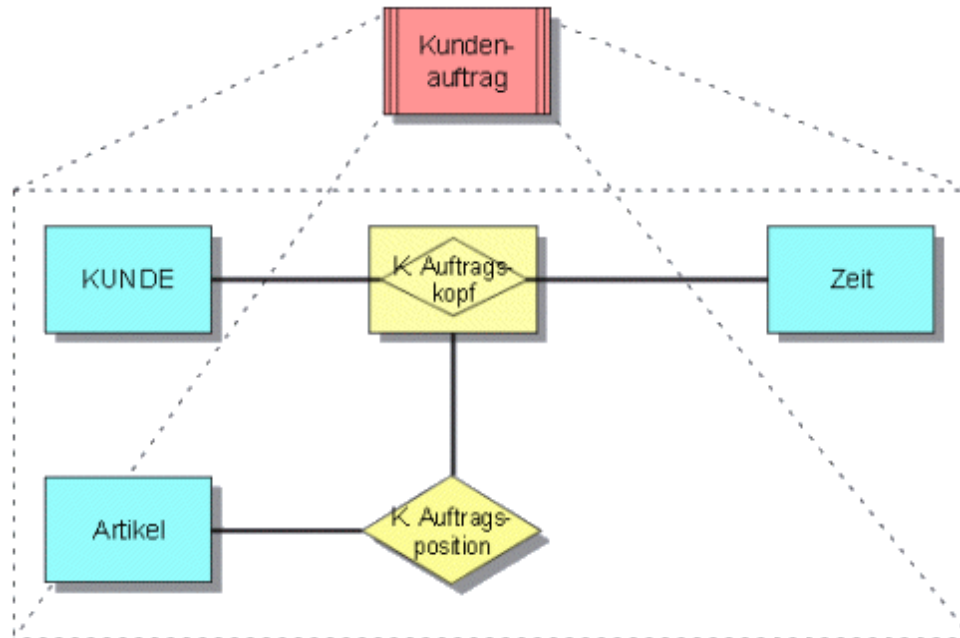


Abbildung 38: Datenclustersicht auf mehrere Objekte

Das Datencluster ist eine logische Sicht auf mehrere Entity- und Beziehungstypen. Zur Beschreibung des komplexen Objektes **Kundenauftrag** werden die Entity- und Beziehungstypen **Kunde**, **Zeit**, **Auftragskopf**, **Artikel** und **Auftragsposition** benötigt.

Gruppierung

Bei der Gruppierung werden aus den Elementen einer Entitymenge Gruppen gebildet.

So werden zum Beispiel in der folgenden Abbildung die Betriebsmittel zu Betriebsmittelgruppe zusammengefasst. Die Betriebsmittelgruppe ist ein eigenständiges Objekt, das durch weitere Attribute (Name der Betriebsmittelgruppe, Anzahl der Betriebsmittel), die nicht zu einzelnen Betriebsmitteln gehören, näher beschrieben werden kann. Weitere Beispiele sind die Gruppierung von Arbeitsplätzen zu Abteilungen oder die Zusammenfassung von Auftragszeilen zu Aufträgen.



Abbildung 39: Gruppierung

4.2.1.2.2 Erweiterung der Kardinalitäten

Bisher wurde bei der Angabe der Kardinalitäten jeweils immer nur die Obergrenze der möglichen Beziehungsausprägungen angezeigt. So wird mit den Kardinalitäten in der folgenden Abbildung ausgedrückt, dass einem Projekt maximal mehrere (m) Mitarbeiter zugeordnet werden können, ein Mitarbeiter maximal an mehreren (n) Projekten beteiligt sein kann.



Abbildung 40: Ober-/Untergrenze (1)

Neben dieser Angabe der Obergrenze kann aber auch die Untergrenze zur Spezifikation der minimalen Anzahl der Beziehungsausprägungen von Interesse sein. Hierzu können z. B. die Kardinalitäten als Buchstabenpaar (a, b) dargestellt werden (vgl. Scheer, Wirtschaftsinformatik). In der nächsten Abbildung wird durch das Buchstabenpaar (a_1, b_1) ausgedrückt, dass jedes Projekt an **mindestens** a_1 und **höchstens** b_1 Beziehungsausprägungen vom Typ **arbeitet in** teilnehmen kann, was bedeutet, dass jedem Projekt **mindestens** a_1 und **höchstens** b_1 Mitarbeiter zugeordnet werden können. Das Buchstabenpaar (a_2, b_2) drückt aus, dass ein Mitarbeiter **mindestens** an a_2 und **höchstens** an b_2 Projekten beteiligt sein kann.



Abbildung 41: Ober-/Untergrenze (2)

Jede Beziehung wird somit durch zwei Komplexitätsgrade (Minimum, Maximum) bestimmt. Als Werte für die Untergrenze werden hierbei oft nur 0 und 1 verwendet, die Obergrenze hat als Wertebereich $1 \leq \max \leq *$ (* bedeutet hierbei „beliebig viele“).

Eine Untergrenze $\min = 0$ bedeutet, dass ein Entity an einer Beziehung beteiligt sein kann, aber nicht muss; die Untergrenze $\min = 1$ bedeutet, dass ein Entity mindestens an einer Beziehung teilnehmen muss.

In der nächsten Abbildung wird durch die Untergrenzen ausgedrückt, dass ein Mitarbeiter an einer Beziehung teilnehmen kann, aber nicht muss ($\min = 0$), ein Projekt aber mindestens an einer Beziehung teilnehmen muss ($\min = 1$). Damit wird ausgedrückt, dass Mitarbeiter existieren können, die keinem Projekt zugeordnet sind, umgekehrt jedoch kein Projekt existieren kann, dem nicht mindestens ein Mitarbeiter zugeordnet ist.



Abbildung 42: Ober- Untergrenze (3)

Werden als Minimalwerte nur 0 oder 1 und als Maximalwerte 1 oder * zugelassen, können somit vier Fälle einer (\min, \max) - Notation unterschieden werden: (1,1), (1,m), (0,1) und (0,m). Ist diesem Fall kann auch folgende abgekürzte Schreibweise verwendet werden (vgl.

Schlageter/Stucky, Datenbanksysteme 1983, S. 51):

- 1 (entspricht (1,1))
- c (entspricht (0,1))
- m (entspricht (1,m))
- cm (entspricht (0,m)).

Die folgende Abbildung zeigt das vorherige Abbildungsbeispiel mit dieser Notation.



Abbildung 43: Ober- Untergrenze (4)

4.2.1.2.3 Identifizierungs- und Existenzabhängigkeit

Die in Kapitel **Erweiterungen des ERM - eERM** (Seite 38) diskutierte Erweiterung der Kardinalitäten um die Angabe von Unter- und Obergrenzen ermöglicht es nun, bestimmte Abhängigkeiten zwischen Datenobjekten zu definieren.

Beziehungstypen und uminterpretierte Beziehungstypen existieren definitionsgemäß nicht aus sich heraus, sondern aus der Existenz der Entitytypen, die sie verbinden. Sie sind somit identifikations- und existenzabhängig von anderen Entitytypen.

Daneben gibt es jedoch auch Entitytypen, die, obwohl sie ein eigenes Schlüsselattribut besitzen, von der Existenz anderer Entitytypen abhängen. Diese Abhängigkeiten sind z. B. durch die Operation der Gruppierung begründet. So ist in der nächsten Abbildung eine Abteilung nur dann sinnvoll, wenn ihr mindestens ein Arbeitsplatz zugeordnet ist, umgekehrt ist die Definition eines Arbeitsplatzes nur sinnvoll, wenn er einer Abteilung zugeordnet ist. Wie in der folgenden Abbildung dargestellt, werden diese existentiellen Abhängigkeiten durch die Angabe der Komplexitätsgrade ausgedrückt. Bei einer (min, max) - Notation durch (1,1) und (1,*). Die Definition existentieller Abhängigkeiten im Datenmodell führt in einer späteren Umsetzung zu Bedingungen der referentiellen Integrität von Daten. Einfach erklärt, stellt die Beachtung dieser Bedingungen sicher, dass sich die Inhalte einer Datenbank nach bestimmten Transaktionen immer noch in einem konsistenten Zustand befinden. Für das Beispiel bedeutet dies, dass eine Abteilung nur dann gelöscht werden kann, wenn auch alle Arbeitsplätze, die dieser Abteilung zugeordnet sind, gelöscht werden.



Abbildung 44: Existentielle Abhängigkeit

4.2.1.2.4 Modellierung von Fachbegriffen des Unternehmens-Fachbegriffsmodell

Eine häufig auftretende Schwierigkeit der Modellierung - insbesondere der Datenmodellierung - ist die Begriffsvielfalt, die sich für Informationsobjekte in größeren Unternehmen ergibt. So wird unter dem Begriff **Auftrag** in der Einkaufsabteilung etwas Anderes verstanden als in der Produktion. Die Akzeptanz der erhobenen Informationen kann jedoch durch die Verwendung der Unternehmens- oder Abteilungsbegrifflichkeit erheblich erhöht werden. Aus diesem Grunde bietet der Methodensatz von ARIS sog. Fachbegriffsmodelle an, die sowohl die Verwaltung der unterschiedlichen Begriffe im Sinne einer Synonym-Verwaltung für Datenobjekte als auch die Pflege der Beziehungen zwischen den Objekten der Datenmodelle (Entitytyp, Beziehungstyp,...) und den erhobenen Fachbegriffen des Unternehmens erlaubt.

Zur Darstellung dieser Beziehungen wird der Objekttyp des Fachbegriffs eingeführt. Jedem Informationsobjekt des Datenmodells können nun mehrere Fachbegriffe zugeordnet werden. Ein Beispiel hierfür zeigt diese Abbildung.

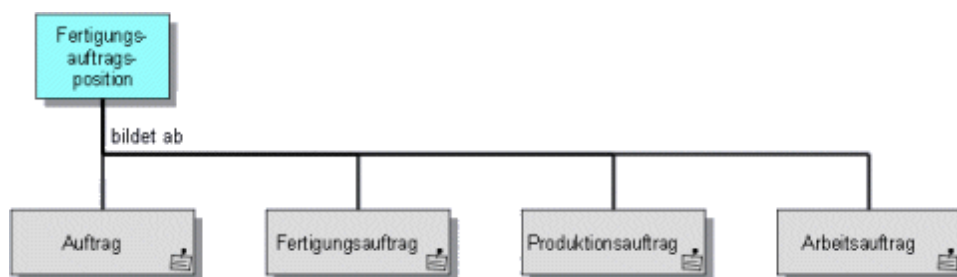


Abbildung 45: Fachbegriffe (1)

Fachbegriffe können untereinander in Beziehung stehen und hierarchisiert werden. Die folgende Abbildung verdeutlicht die Verwendung der verschiedenen Kantentypen zwischen Fachbegriffen.

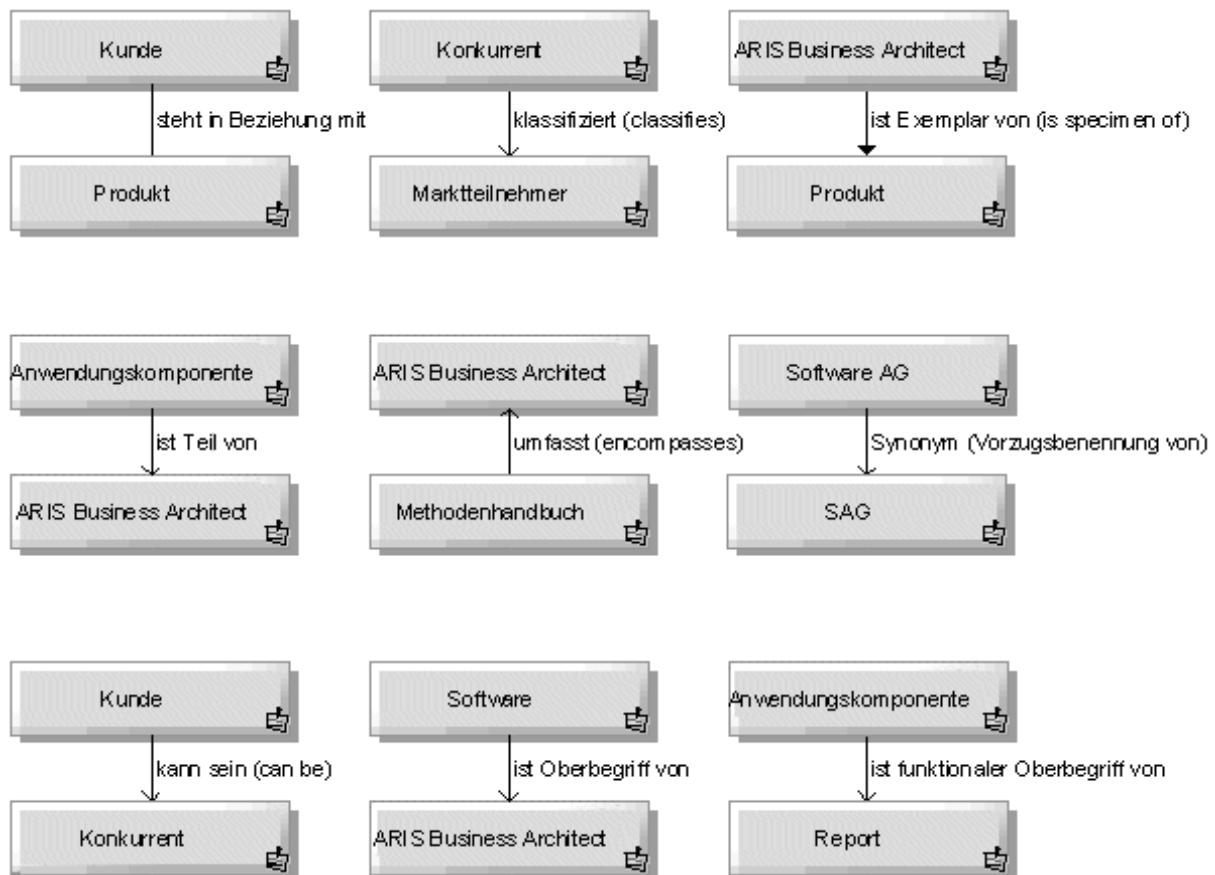


Abbildung 46: Fachbegriffe (2)

Die im Fachbegriffsmodell definierten Fachbegriffe können nun auch in anderen Modelltypen, in denen Informationsobjekte verwendet werden, herangezogen werden, so z. B. in Prozessketten zur Darstellung der Input-/Output-Daten einer Funktion.

4.2.1.2.5 eERM-Attributzuordnungsdiagramm

Datenmodelle in Form von eERM-Darstellungen, in denen nur die Entitytypen und Beziehungstypen dargestellt werden, weisen meist schon recht komplexe Strukturen auf. Eine Aufnahme der ERM-Attribute in diese Modelle führt dazu, dass die Modelle nicht mehr lesbar sind.

eERM-Attributzuordnungsdiagramme bieten die Möglichkeit, für jeden Entity- und Beziehungstyp die ERM-Attributzuordnungen in einem eigenen Modell zu hinterlegen. Der Objekttyp des eERM (Entitytyp oder Beziehungstyp) kann per Ausprägungskopie in dieses Modell aufgenommen werden und die Beziehungen zu ERM-Attributen modelliert werden. Hierbei kann unterschieden werden, ob es bei dem verbundenen ERM-Attribut um ein Schlüsselattribut, einen Fremdschlüssel oder ein beschreibendes Attribut handelt. Ein Beispiel hierzu ist in der nächsten Abbildung dargestellt.

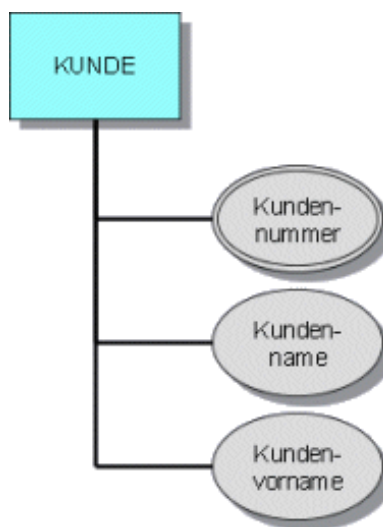


Abbildung 47: Zuordnung von ERM-Attributen zu einem Entitytyp

Neben der Darstellung und Zuordnung einzelner ERM-Attribute können in diesem Modelltyp auch Attributtypgruppen und ihre Zuordnungen dargestellt werden.

Eine Attributtypgruppe stellt die Gruppierung von ERM-Attributen eines Entitytyps dar, die eine enge semantische Zusammengehörigkeit aufweisen. So können z. B. die ERM-Attribute eines Entitytyps, die in ihrer Gesamtheit einen Sekundärschlüssel bilden, zu einer Attributtypgruppe zusammengefasst werden.

Attributtypgruppen werden wie folgt dargestellt:



Abbildung 48: Darstellung einer Attributtypgruppe

Eine Zusammenfassung der möglichen Beziehungen des ERM-Attributzuordnungsdiagramms finden Sie im Handbuch "ARIS-Methode – Tabellen" (Datei **method_tabulations_aris.pdf**) auf Ihrem Installationsmedium.

4.2.1.3 Alternative Darstellungsformen

4.2.1.3.1 SAP-SERM

Neben der hier beschriebenen Darstellungsform sind auch andere ERM-Darstellungen gebräuchlich. Häufig werden Beziehungstypen nicht als eigene Objekte dargestellt, sondern als Kanten zwischen Entitytypen. Beziehungstypen, die eigene Attribute besitzen, werden dann als (schwache) Entitytypen dargestellt.

Eine dieser Darstellungsformen wurde von Sinz mit der SERM-Darstellung (Strukturiertes Entity-Relationship-Modell) entwickelt (vgl. Sinz, Entity-Relationship-Modell 1990). Durch die Verwendung gerichteter Graphen zur Darstellung der originären und abhängigen Datenobjekte und die Anordnung der Modellobjekte von links (starke Entitytypen) nach rechts (schwache Entitytypen und Beziehungstypen) wird die Entwicklungsrichtung der Datenstruktur sichtbar. Die Kantenansatzpunkte werden durch das jeweilige Objekt beschränkt und unterstützen damit die Visualisierung der Entwicklungsrichtung. Der Hauptunterschied zwischen SER-Modellen und der bisher beschriebenen Modellierung nach der eERM-Methode liegt somit in der grafischen Darstellung.

Die von der SAP AG in Walldorf im Rahmen ihrer Informationsmodellierung entwickelte Modellierungsmethodik verbindet Gedanken des SER-Ansatzes mit dem hier erläuterten eERM-Ansatz (vgl. Keller/Hechler, Informationsmodell 1991).

Hierbei wird bei der Objektbildung grafisch nicht zwischen Entitytypen und Beziehungstypen unterschieden. Die Abhängigkeiten zwischen den Informationsobjekten werden durch die Beziehungskomplexitäten der Pfeildarstellungen zum Ausdruck gebracht. Hierbei wird zwischen hierarchischen, aggregierenden und referentiellen Beziehungen unterschieden (vgl. folgende Abbildung).

Eine hierarchische Beziehung definiert eine einseitig existentielle Abhängigkeit zwischen Informationsobjekten.

Aggregierende Beziehungen entsprechen der Bildung von Beziehungstypen des eERM-Ansatzes. Referentielle Beziehungen beschreiben die logischen Abhängigkeiten zwischen uminterpretierten Entitytypen und originären Entitytypen im Sinne des eERM-Ansatzes.

Die Spezialisierung wird analog zum eERM-Ansatz dargestellt.

In der nächsten Abbildung wird zur Verdeutlichung dieser Notation ein Beispiel in eERM-Darstellung gezeigt und in die SAP-ERM-Darstellung umgesetzt (vgl. hierzu auch: Seubert, SAP-Datenmodell 1991, S. 94).

Es wird deutlich, dass die zugrunde liegenden Fachinhalte ohne Informationsverlust in diese Darstellung überführt werden können.

Das SAP-ERM kann als eine Art der Repräsentation von Datenmodellen nach Abschluss des Erstellungsprozesses angesehen werden. Durch die graphenorientierte Anordnung der Informationsobjekte im SAP-ERM wird eine schnellere Navigation und Orientierung insbesondere in umfangreichen Datenmodellen unterstützt.

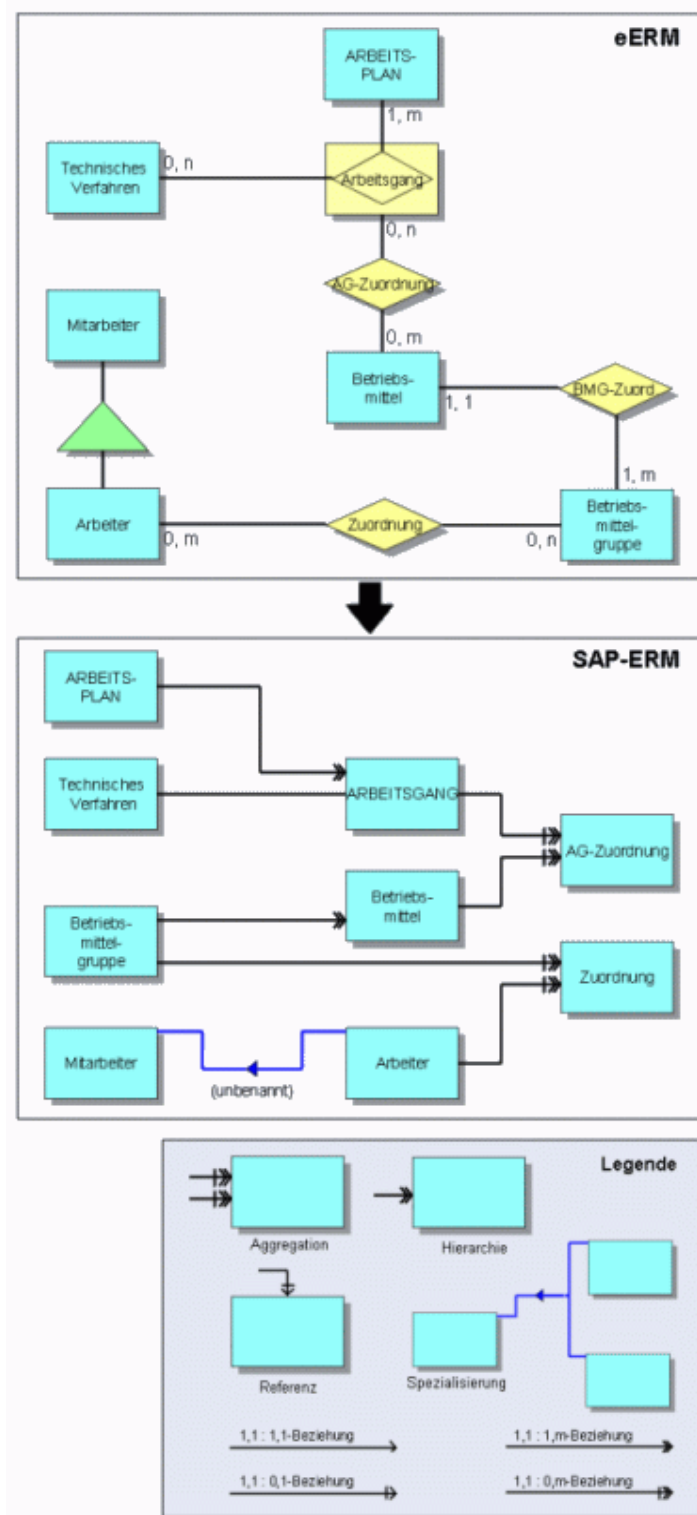


Abbildung 49: eERM und SAP-ERM-Darstellung

4.2.1.3.2 IE-Datenmodell

Das IE-Datenmodell entspricht der Notation der Datenmodellierung des CASE-Tools Information Engineering Facility (IE) der Texas Instruments Inc.

Wie auch die Notation von SAP-SERM kennt die IE-Notation keine eigenen Objekttypen für Beziehungen zwischen Entitytypen.

Ein Beispiel für ein Datenmodell in IE-Notation zeigt diese Abbildung.

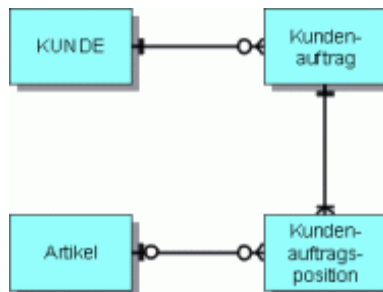


Abbildung 50: Datenmodell in IE-Notation

4.2.1.3.3 SeDaM - Modell

Die Datenmodell-Notation SeDaM (semantisches Datenmodell) ist eine Notation der BASF AG. Es handelt sich hierbei ebenfalls um eine Notation, die keine eigenen Objekttypen für Beziehungen zwischen Entitytypen kennt.

Eine strenge Anordnung der Entitytypen von links nach rechts (vgl. SAP-SERM-Notation) erfolgt nicht.

Die Objekttypen **Datencluster** und **Generalisierungstyp** stehen ebenfalls zur Verfügung.

Ein Beispiel eines Datenmodells in SeDaM-Notation zeigt die folgende Abbildung.

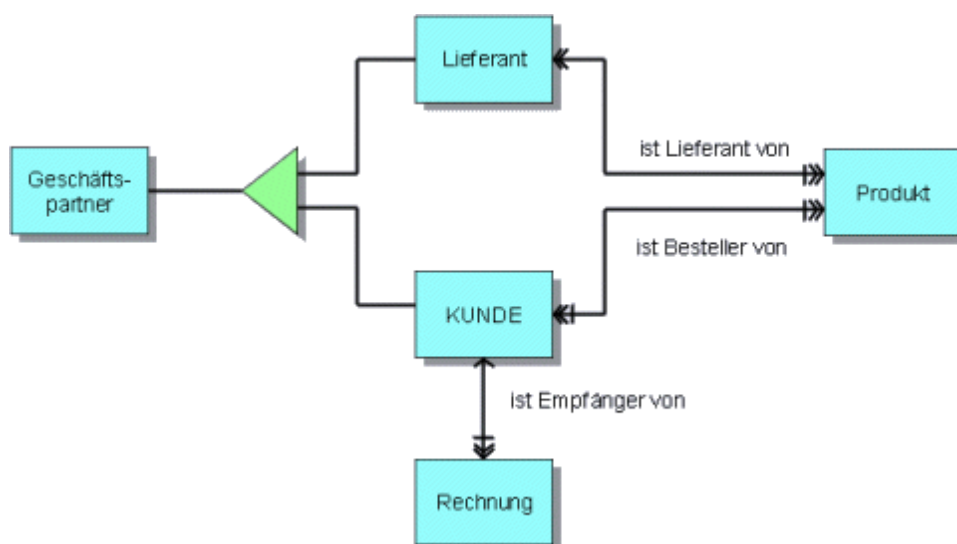


Abbildung 51: Datenmodell in SeDaM-Notation

Eine Zusammenfassung der möglichen Beziehungstypen des SeDaM-Modells findet sich im Handbuch **ARIS-Methode – Tabellen** (Datei **method_tabulations_aris.pdf**) auf Ihrem Installationsmedium.

4.2.1.4 Zusammenfassung der wichtigsten Begriffe und Darstellungsformen des eERM

Die Begriffe und Darstellungsformen der Strukturelemente und Konstruktionsoperatoren des erweiterten Entity-Relationship-Modells (eERM) sind zusammenfassend in der folgenden Abbildung dargestellt (vgl. Scheer, Wirtschaftsinformatik 1994, S. 45).

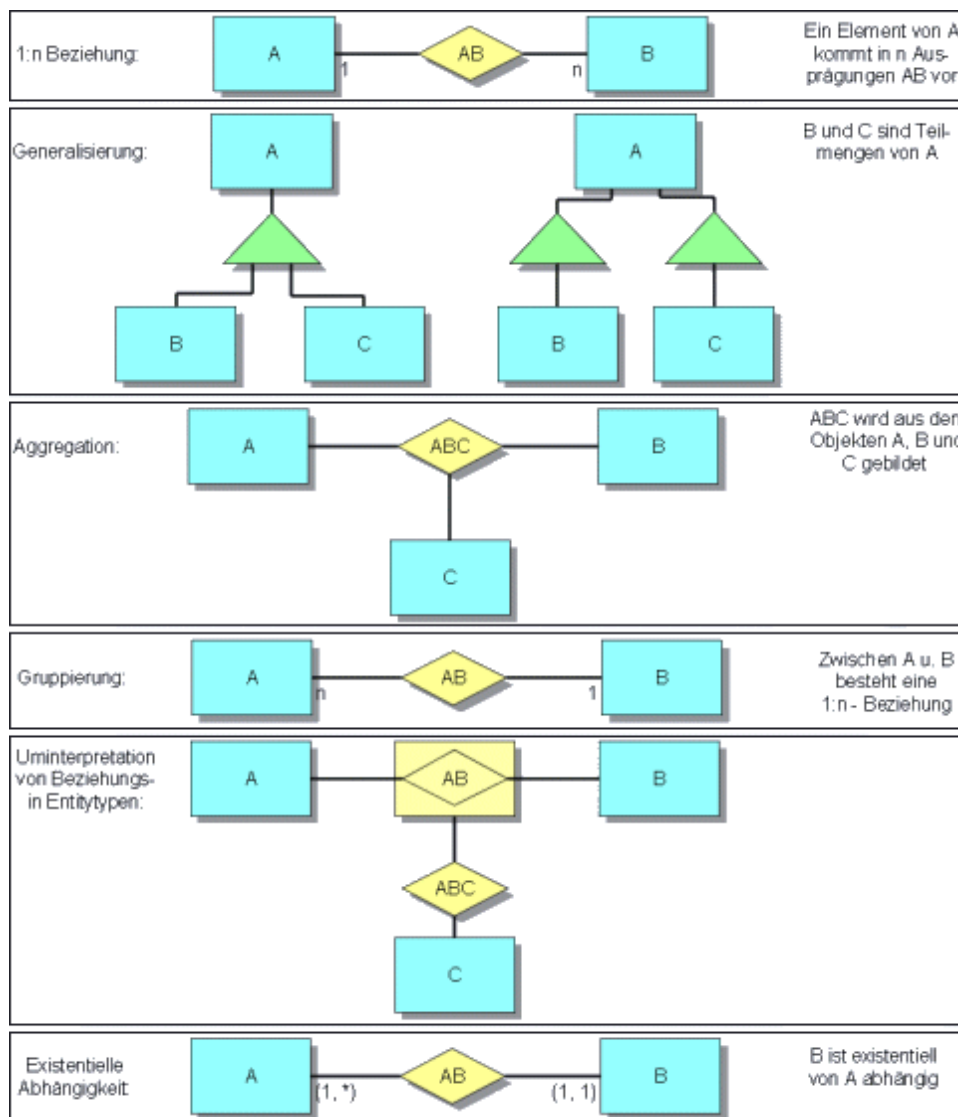


Abbildung 52: Begriffe und Darstellungsformen des eERM

4.2.1.5 Dokumenttyp-Definition

Ein Modell des Typs **DTD** (Dokumenttyp-Definition) beschreibt die Regeln, nach denen ein XML-Dokument eines bestimmten Typs aufgebaut sein muss. Die Beschreibung erfolgt in Form von Elementtypdeklarationen.

Sie können mit Hilfe einer DTD z. B. die allgemeine Struktur eines Belegtyps definieren. Ein gültiger Beleg eines in der DTD definierten Belegtyps kann als XML-Dokument erstellt werden. Dies bietet den Vorteil, dass der Beleg gemeinsam mit der zugehörigen DTD von verschiedenen Programmen verarbeitet werden kann.

Das Modell muss ausgehend von einem Objekt des Typs **Elementtyp** streng hierarchisch aufgebaut werden. Für das Ausgangsobjekt gilt, dass es keine eingehenden Kanten besitzen darf. Des Weiteren müssen Kanten immer vom übergeordneten Objekt zum untergeordneten verlaufen. Während Sie für übersichtliche Dokumentstrukturen alle Elementtypen in einem DTD-Modell beschreiben können, haben Sie bei komplexen Strukturen auch die Möglichkeit die Deklaration eines Elementtyps in einem hinterlegten DTD-Modell vorzunehmen.

Wenn Sie mit Hinterlegungen arbeiten, muss das hinterlegte Modell die vollständige Beschreibung des Elementtyps enthalten.

Mit Hilfe des Reports **DTDExport.rsm** können Sie ausgehend vom Basismodell über die gesamte DTD-Modellstruktur eine DTD-Datei erzeugen.

Wenn Ihnen eine Dokumenttyp-Deklaration in Form einer Textdatei vorliegt, können Sie mit Hilfe des Reports **DTDImport.rsg** ein Modell des Typs **DTD** generieren.

4.2.1.5.1 Elementtypen

Die wesentlichen Bestandteile einer DTD sind Elementtypen. Instanzen der Elementtypen, die in der Hierarchie von DTD-Modellen vorkommen, dürfen in einem gültigen XML-Dokument vorkommen und werden Elemente genannt.

Jeder Elementtyp wird durch seinen Inhalt und durch seine Attributtypen beschrieben.

Aufgrund des Inhalts kann man drei Arten von Elementtypen unterscheiden:

- Elementtypen, mit Text als Inhalt
- Elementtypen ohne Inhalt
- Elementtypen mit Text und/oder weiteren Elementtypen als Inhalt

Um einen Elementtyp zu beschreiben, der nur Text als Inhalt besitzt, platzieren Sie ein Objekt des Typs **Elementtyp** mit dem gewünschten Namen und ein Element des Typs **Inhalt** mit dem Symbol **#PCDATA**. Anschließend ziehen Sie eine Kante des Typs **beinhaltet** zwischen dem Elementtyp und dem Inhalt.

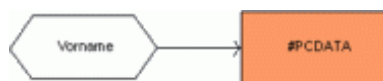


Abbildung 53: DTD-Elementtyp mit reinem Textinhalt

Um einen Elementtyp ohne Inhalt zu beschreiben, platzieren Sie ein Objekt des Typs **Elementtyp** mit dem gewünschten Namen und ein Element des Typs **Inhalt** mit dem Symbol **EMPTY**. Anschließend ziehen Sie eine Kante des Typs **beinhaltet** zwischen dem Elementtyp und dem Inhalt. Typisches Beispiel für einen Elementtyp ohne Inhalt ist der Tag **** in HTML. Der Nutzen von leeren Elementtypen liegt im Wesentlichen darin, dass sie nützliche Attribute besitzen können, z. B. die Attribute **SRC**, **ALIGN**, **ALT** und **ISMAP** des **img**-Tag.

Die komplexeste Form eines Elementtyps ist ein Elementtyp mit gemischtem Inhalt. In diesem Fall können einem Objekt des Typs **Elementtyp** Text oder Elementtypen zugeordnet werden. Um zu beschreiben, in welcher Form die Zuordnungen miteinander verbunden werden, können Regeln verwendet werden. Die möglichen Kanten zwischen dem Elementtyp und den Zuordnungen beschreiben, wie häufig eine Zuordnung im Inhalt des Elementtyps vorkommen darf. Zum Verknüpfen der Zuordnungen stehen Ihnen der Sequence-Operator und die XOR-Regel zur Verfügung.

Wenn im Inhalt des beschriebenen Elementtyps aus einer angegebenen Menge von Zuordnungen nur eine erlaubt ist, müssen die Zuordnungen über eine XOR-Regel miteinander verknüpft werden.

Wenn im Inhalt des beschriebenen Elementtyps mehrere Zuordnungen nur in einer gewissen Reihenfolge erlaubt sind, müssen sie über einen Sequence-Operator miteinander verknüpft werden. Um die gewünschte Reihenfolge eindeutig festzulegen, geben Sie an der Kante zwischen dem Operator und der Zuordnung die gewünschte Position für die Zuordnung an.

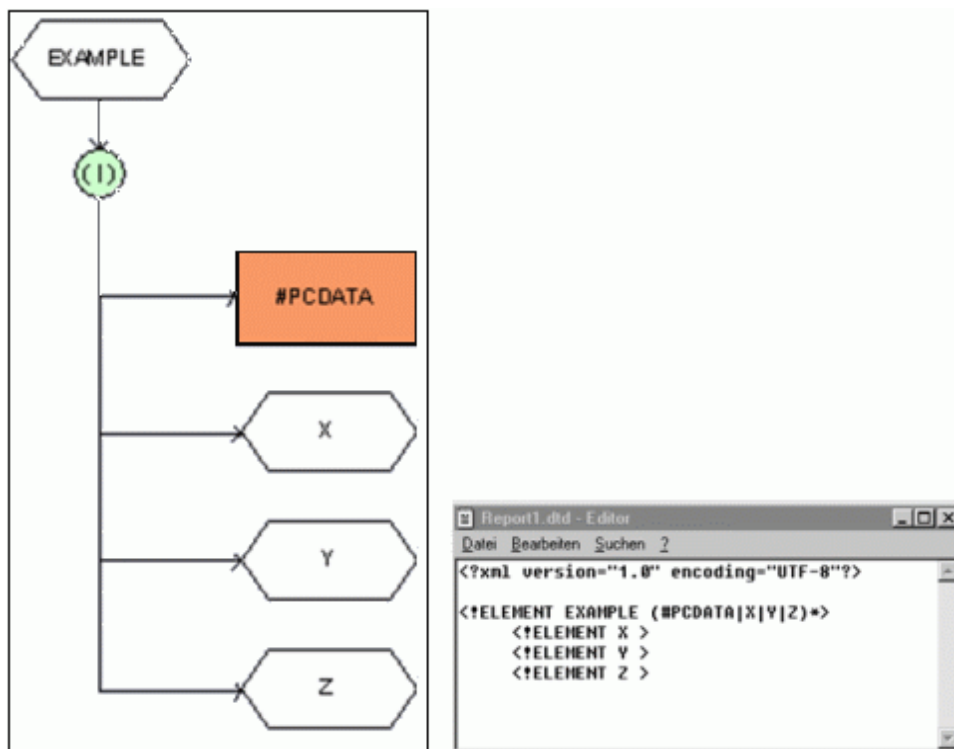


Abbildung 54: Elementtypen mit gemischtem Inhalt und Umsetzung in der DTD

Zuordnungen können auch ohne Verknüpfungsoperator mit dem beschriebenen Element verbunden werden. In diesem Fall wird standardmäßig von einer Sequence-Verknüpfung mit beliebiger Reihenfolge ausgegangen.

4.2.1.5.2 Kantentypen

Zuordnungen im Inhalt eines Elementtyps können unterschiedlich häufig vorkommen. Die Anzahl von erlaubten Vorkommen einer Zuordnung im Inhalt des Elementtyps wird durch den Kantentyp zwischen Elementtyp und Zuordnung bestimmt.

Folgende Kantentypen stehen zur Auswahl:

- hat beliebig oft
- hat mindestens einmal
- hat höchstens einmal
- hat genau einmal

Folgende Tabelle erläutert die verschiedenen Kantentypen und das jeweilige Zeichen, das in der erzeugten DTD deren Vorkommen symbolisiert:

Kantentyp	Beschreibung	Zeichen
hat beliebig oft	Die Zuordnung muss nicht im Inhalt des beschriebenen Elementtyps vorkommen. Sie kann einmal oder mehrfach vorliegen (min = 0, max = beliebige positive ganze Zahl).	*
hat mindestens einmal	Die Zuordnung muss im Inhalt des beschriebenen Elementtyps vorkommen, wobei sie auch mehrfach vorliegen kann (min = 1, max = beliebige positive ganze Zahl).	+
hat höchstens einmal	Die Zuordnung muss nicht im Inhalt des beschriebenen Elementtyps vorkommen. Sie darf nur einmal vorliegen (min = 0, max = 1).	?
hat genau einmal	Die Zuordnung muss im Inhalt des beschriebenen Elementtyps vorkommen. Sie darf nur einmal vorliegen (min = 1, max = 1).	kein Zeichen

4.2.1.5.3 Attributtypen

Neben der Strukturbeschreibung kann eine DTD auch Deklarationen von Attributtypen enthalten. Attributtypen beschreiben die Eigenschaften eines Elementtyps. Sie sind immer einem Elementtyp zugeordnet.

Es können einfache Attributtypen und Aufzählungsattributtypen in einer DTD deklariert werden.

Um einen einfachen Attributtyp zu definieren,

1. platzieren Sie ein Objekt des Typs **Attributtyp** mit dem gewünschten Namen im Modell und
2. ziehen eine Kante vom Elementtyp, dessen Eigenschaft durch den Attributtyp beschrieben wird, zum neuen Attributtyp.
3. Anschließend öffnen Sie die Attributbearbeitung, um weitere Angaben für die Deklaration des Attributtyps zu machen.

Zu diesen Angaben gehören:

- Datentyp des Attributwerts
- Attributvorgabe
- Vorgabewert

4.2.1.5.3.1 Datentyp des Attributwerts

Um den Datentyp des Attributwerts zu bestimmen, pflegen Sie das Attribut **Datentyp**. Folgende Tabelle erläutert die angegebenen Datentypen:

Datentyp	Beschreibung
CDATA	Zeichenketten können im Attributwert verwendet werden.
ID	Ein eindeutiger Identifizierer kann im Attributwert verwendet werden. Ist der Wert nicht eindeutig, meldet der XML-Prozessor einen Fehler.
IDREF	Ein Verweis auf einen Identifizierer, der an anderer Stelle im Dokument definiert ist, kann im Attributwert verwendet werden. Wenn ein Identifizierer als Wert verwendet wird, der nicht im aktuellen XML-Dokument vergeben wurde, meldet der XML-Prozessor einen Fehler.
IDREFS	Der Attributwert kann aus mehreren Attributwerten des Typs IDREF bestehen, die durch Leerzeichen voneinander getrennt sind. Wenn im Attributwert auf einen Identifizierer verwiesen wird, der nicht im aktuellen XML-Dokument vergeben wurde, meldet der XML-Prozessor einen Fehler.
ENTITY	Eine Referenz auf einen entfernten binären Entity, der innerhalb der DTD deklariert ist, kann als Attributwert verwendet werden.
ENTITIES	Der Attributwert kann aus mehreren Attributwerten des Typs ENTITY bestehen, die durch Leerzeichen voneinander getrennt sind.

Datentyp	Beschreibung
NMTOKEN	Beliebige Kombinationen aus Buchstaben, Ziffern, Punkten, Bindestrichen, Strichpunkten oder Unterstrichen können als Attributtyp verwendet werden.
NMTOKENS	Der Attributwert kann aus mehreren Attributwerten des Typs NMTOKEN bestehen, die durch Leerzeichen voneinander getrennt sind.
NOTATION	Ein Verweis auf eine Notation, die in der DTD deklariert ist, können als Attributwert verwendet werden.

Standardmäßig ist für das Attribut **Datentyp** der Wert **NMTOKEN** gepflegt.

4.2.1.5.3.2 Attributvorgabe

Für das Attribut **Attributvorgabe** können Sie einen der folgenden Werte wählen:

- #REQUIRED
- #IMPLIED
- #FIXED

Wenn für ein Attributtyp der Wert der Attributvorgabe auf #REQUIRED gesetzt ist und dieses Attribut im XML-Dokument für ein Element gepflegt wird, muss zwingend ein gültiger Wert für das Attribut angegeben werden. Fehlt der Wert, meldet der XML-Prozessor einen Fehler.

Wenn für ein Attributtyp der Wert der Attributvorgabe auf #IMPLIED gesetzt ist und dieses Attribut im XML-Dokument für ein Element gepflegt wird, ist die Angabe eines Wertes für das Attribut optional.

Wenn für ein Attributtyp der Wert der Attributvorgabe auf #FIXED gesetzt ist, wird ein feststehender Wert als Attributwert verwendet. Dieser Wert muss im Attribut **Vorgabewert** angegeben werden. Wird das Attribut im XML-Dokument nicht gepflegt, verhält sich der XML-Prozessor so, als stünde es im Dokument.

Standardmäßig ist für das Attribut **Attributvorgabe** der Wert **#IMPLIED** gepflegt.

Um einen Aufzählungs-Attributtyp zu deklarieren,

- platzieren Sie ein Objekt des Typs **Aufzählungs-Attributtyp** mit dem gewünschten Namen im Modell und
- ziehen Sie eine Kante vom Elementtyp, dessen Eigenschaft durch den Aufzählungs-Attributtyp beschrieben wird, zum neuen Attributtyp.

Anschließend platzieren Sie ein Objekt des Typs **Aufzählung** und geben als Namen die Werte ein, die der Aufzählungs-Attributtyp annehmen kann. Die einzelnen Werte müssen durch Kommata getrennt werden. Wenn die Liste der Werte mehr als 250 Zeichen umfasst, verteilen Sie die Werte über mehrere Aufzählungsobjekte.

Legen Sie nun zwischen dem Aufzählungs-Attributtyp und den Aufzählungen eine Kante an.

Abschließend pflegen Sie im Attribut **Vorgabewert** den Wert aus den Aufzählungen, der standardmäßig angenommen werden soll.

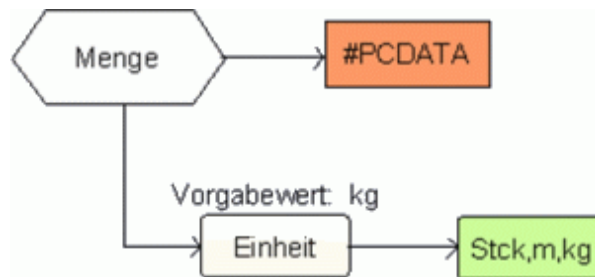


Abbildung 55: Elementtyp mit einem Aufzählungs-Attributtyp

Informationen, die des Attributs **Beschreibung/Definition** von DTD-Modellen und darin enthaltenen Elementtypen werden vom Report **DTDGenerator.rsm** als Kommentare in die erzeugte DTD eingefügt.

Um Parameter-Entities, interne oder externe Entities und Notationen in einer DTD zu deklarieren, stehen Ihnen mehrere Modellattribute zur Verfügung.

Parameter-Entity

Um einen Parameter-Entity zu deklarieren, geben Sie in der angegebenen Reihenfolge

- den symbolischen Namen,
- ein Schlüsselwort und
- einen Uniform Resource Identifier für die gewünschte Parameter-Entity-Deklaration ein.

Der symbolische Name kann als Parameter-Entity-Referenz in der aktuellen DTD verwendet werden.

Als Schlüsselwörter können Sie SYSTEM und PUBLIC verwenden. Wenn Sie mit dem Parameter-Entity auf eine Datei verweisen möchten, die nur innerhalb Ihres Unternehmens bekannt ist und verwendet wird, geben Sie das Schlüsselwort SYSTEM ein. Wenn Sie jedoch auf eine Datei verweisen, die Bestandteil einer Standardbibliothek ist, geben Sie PUBLIC ein.

Beispiel für einen URI, der auf das Schlüsselwort PUBLIC folgt: "-//w3c//ENTITIES Latin//EN//HTML" "http://www.w3.org/DTD/ISOLAT1.ent".

Interner Entity

Um einen internen Entity zu deklarieren, geben Sie in der angegebenen Reihenfolge

- den symbolischen Namen und
- einen Text ein, die zur Deklaration eines allgemeinen, internen Entity verwendet werden. Geben Sie zwischen Name und Text ein Leerzeichen ein und schreiben Sie den Text in Anführungszeichen.

Wenn ein XML-Dokument der aktuellen DTD eine Entity-Referenz mit dem angegebenen symbolischen Namen enthält, wird dieser Name durch den angegebenen Text ersetzt.

Externer Entity

Um einen externen Entity zu deklarieren, geben Sie in der angegebenen Reihenfolge

- einen symbolischen Namen,
- ein Schlüsselwort,
- einen Uniform Resource Identifier (URI) und
- eine Notation ein, die zur Deklaration eines allgemeinen, externen Entity verwendet werden.

Der symbolische Name kann als Entity-Referenz in einem XML-Dokument der aktuellen DTD verwendet werden.

Als Schlüsselwörter können Sie SYSTEM und PUBLIC verwenden. Wenn Sie mit dem externen Entity auf eine Datei verweisen möchten, die nur innerhalb Ihres Unternehmens bekannt ist und verwendet wird, geben Sie das Schlüsselwort SYSTEM ein. Wenn Sie jedoch auf eine Datei verweisen, die Bestandteil einer Standardbibliothek ist, geben Sie PUBLIC ein.

Ein URI teilt dem XML-Prozessor mit, wo das Objekt zu finden ist, auf das mit dem externen Entity referenziert wird.

Beispiel für einen URI, der auf das Schlüsselwort SYSTEM folgt:

"c:\images\test.gif".

Beispiel für einen URI, der auf das Schlüsselwort PUBLIC folgt:

"-//w3c//ENTITIES Latin//EN//HTML" "http://www.w3.org/DTD/ISOLAT1.ent".

Die Notation am Ende der Eingabe informiert den XML-Prozessor über den Typ des Objekts, auf das mit dem externen Entity referenziert wird. Die verwendete Notation muss in der aktuellen DTD deklariert sein.

Wenn Sie bspw. den URI des ersten Beispiels in Ihrer Entity-Deklaration verwenden, deklarieren Sie zuvor in der DTD eine Notation für das Datenformat GIF und fügen Sie als Abschluss Ihrer Eingabe in diesem Attributtyp NDATA GIF ein.

Notation

Um eine Notation zu deklarieren, geben Sie in der angegebenen Reihenfolge

- einen symbolischen Namen,
- ein Schlüsselwort und
- einen Uniform Resource Identifier (URI) ein.

Diese Informationen weisen den XML-Prozessor an, wie er Objekte dieses Typs, die in XML-Dokumenten der aktuellen DTD vorkommen, behandeln soll.

Der symbolische Name der Notation kann in Attribut- und Entity-Deklarationen der aktuellen DTD verwendet werden.

Als Schlüsselwörter können Sie SYSTEM und PUBLIC verwenden. Wenn Sie z. B. eine Notation für Objekte im Format GIF deklarieren und den XML-Prozessor anweisen möchten Objekte dieses Typs mit dem lokal verfügbaren Internet Explorer anzuzeigen, geben Sie das Schlüsselwort

SYSTEM ein. Wenn Sie jedoch eine Notation für Dateien des Typs TEX deklarieren und dazu den XML-Prozessor auf eine allgemein zugängliche Resource oder Quelle verweisen möchten, geben Sie das Schlüsselwort PUBLIC ein.

Der Uniform Resource Identifier informiert den XML-Prozessor darüber, wo er die Anwendung oder Informationen finden kann, die die Anweisungen für den Umgang mit Objekten des angegebenen Typs enthalten. Der URI für die GIF-Notation mit dem Schlüsselwort SYSTEM könnte z. B. "c:\Programme\Internet Explorer\iexplore.exe" sein, während als URI für die Notation mit dem Schlüsselwort PUBLIC "ISBN 0-201-13448-9: ://NOTATION TeX//EN" eingegeben werden kann.

4.2.1.5.4 DTD testen

Wenn Sie die gewünschte DTD erstellt haben und diese in zwei Schritten testen möchten, haben Sie mit Hilfe des Objekttyps **Bedingter Abschnitt** die Möglichkeit Teile der DTD, die im aktuellen Test nicht berücksichtigt werden, zu aktivieren oder zu deaktivieren.

Wenn Sie Elementtypen, die einem bedingten Abschnitt untergeordnet sind, aus einer DTD ausblenden möchten, aktivieren Sie das Attribut **Ignore** des entsprechenden bedingten Abschnitts.

Wenn Sie mit Hilfe des Reportskripts **DTDImport.rsg** aus einer textuellen DTD, die einen bedingten Abschnitt erhält, ein Modell des Typs **DTD** erzeugen, wird der Inhalt des Abschnitts nicht berücksichtigt.

Ein Modell des Typs **DTD** kann u. a. den Datenelementen der ARIS-Methode hinterlegt werden. Zu den Datenelementen gehören:

- Cluster/Datenmodell
- Package
- Entitytyp
- Beziehungstyp
- Fachbegriff

4.2.1.6 Modellierung des Materialflusses - Materialdiagramm

Zur Darstellung des Materialflusses in Prozessmodellen (EPK (Materialfluss), VKD (Materialfluss)) werden den einzelnen Funktionen des Geschäftsprozesses die Materialtypen als Funktionsinput oder Funktionsoutput zugeordnet. Ähnlich der Zuordnung von Informationsobjekten zu Funktionen (Darstellung der Transformation von Informationen durch Funktionen) wird mit dieser Zuordnung die Transformation der Inputmaterialtypen zu Outputmaterialtypen dargestellt.

Im Materialdiagramm können Materialtypen definiert, hierarchisiert und zu Materialklassen klassifiziert werden.

Ein Materialtyp stellt die Typisierung einzelner Materialien dar, die exakt die gleichen Materialeigenschaften aufweisen.

Gleichartige Materialtypen können zu einer Materialklasse zusammengefasst werden. Die Gleichartigkeit kann hierbei für verschiedene Klassifizierungsaspekte betrachtet werden. Ein Materialtyp kann somit auch mehreren Materialklassen zugeordnet werden.

Materialtypen können Packmitteltypen zugeordnet werden. Damit soll ausgedrückt werden, dass bestimmte Materialtypen nur in bestimmten Packmitteltypen transportiert werden können.

Packmitteltypen können ebenfalls hierarchisiert und klassifiziert werden. Damit sind z. B. der Aufbau und die Restriktionen komplexer Verpackungsgebilde darstellbar.

Ein Packmitteltyp stellt die Typisierung einzelner Packmittel dar, die exakt die gleichen Eigenschaften (z. B. Materialeigenschaften) aufweisen.

Gleichartige Packmitteltypen können zu einer Packmittelklasse zusammengefasst werden. Die Gleichartigkeit kann hierbei für verschiedene Klassifizierungsaspekte betrachtet werden. Ein Packmitteltyp kann somit auch mehreren Packmittelklassen zugeordnet werden.

Ein Beispiel für ein Materialdiagramm mit Hierarchisierungen und Klassifizierungen wird in dieser Abbildung dargestellt.

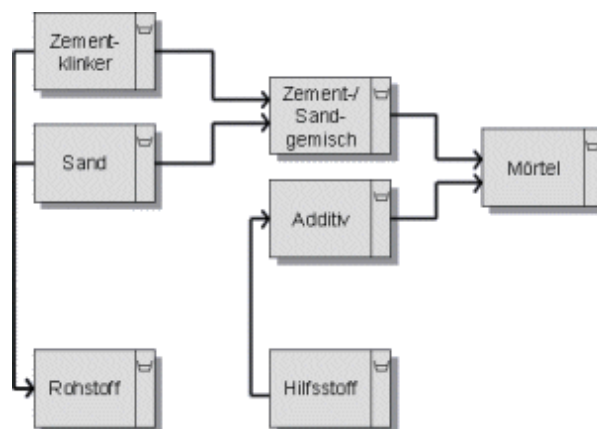


Abbildung 56: Beispiel für ein Materialdiagramm

4.2.1.7 Modellierung der Data Warehouse-Struktur

Das Data Warehouse-Strukturdiagramm dient der Beschreibung des Aufbaus eines Data Warehouse. Primär stellt es die statische Beschreibung dar, d. h. die Relationen der Daten untereinander und deren Standorte. In der ARIS-Architektur wird diese Art der Beschreibung in der Datensicht realisiert. Als zentraler Aspekt werden die Relationen der Informationen betrachtet sowie ihre Anordnung zueinander. Die Daten-Dimensionen werden durch den Info Cube beschrieben. Das Zusammenspiel der Dimensionen wird durch das Sternen-Schema (Star scheme) präsentiert (siehe folgende Abbildung). Hierbei kann eine Dimension als Schlüssel andere miteinander verbinden. Die Objekte der einzelnen Dimensionen können bestimmte Werte annehmen, die in den Faktentabellen katalogisiert und durch Kennzahlen genau bestimmt werden. Die Dimensionstabellen mit ihren Schlüsselattributen und ihren Merkmalen beschreiben die Abhängigkeiten. Die Hierarchiebeziehungen der Merkmale untereinander sind durch Baumstrukturen beschrieben. Schließlich kann die Zuordnung von Dimensionen zu Stammdatentabellen mit Hilfe des Strukturdiagramms erfolgen.

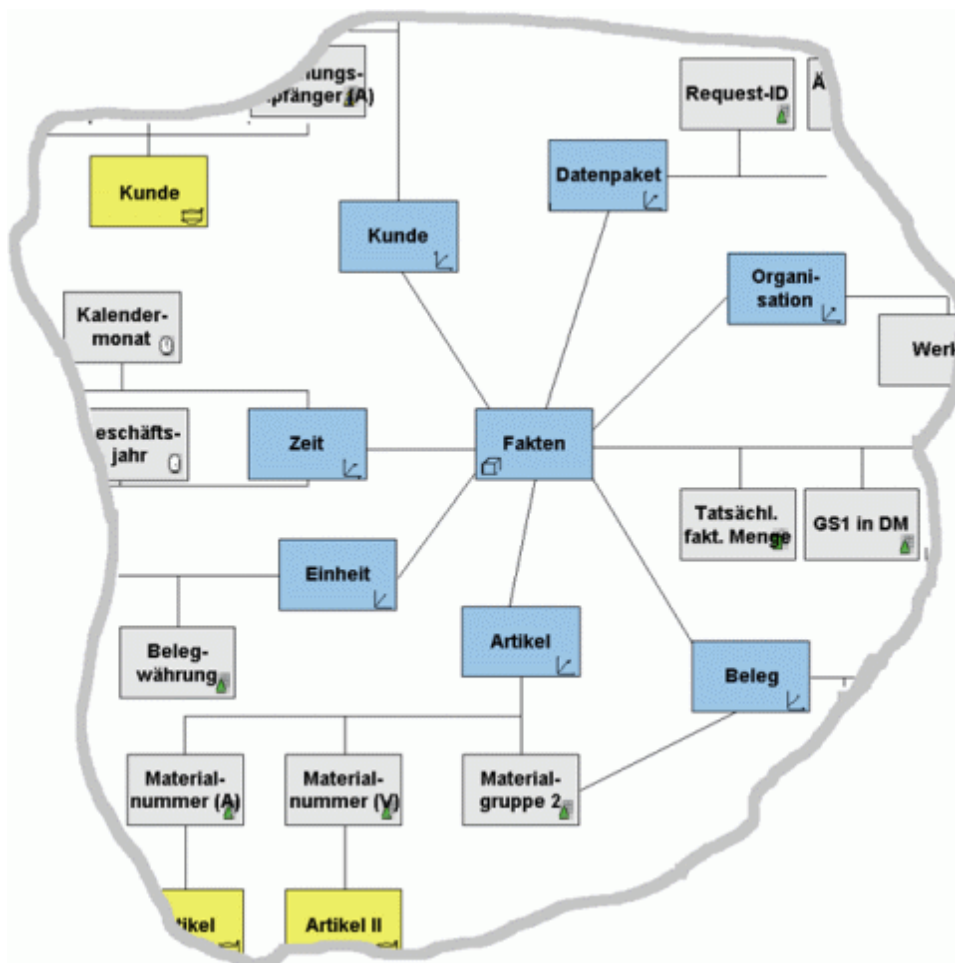


Abbildung 57: Data Warehouse im Sternen-Schema

4.2.1.8 Berechtigungshierarchie

Das Berechtigungshierarchiediagramm wird im Rahmen der Rollen- und Organisationsmodellierung verwendet. Es stellt die Beziehungen von Berechtigungen dar, die im Rollendiagramm definiert wurden. Es werden über- und untergeordnete Berechtigungen definiert, damit ein logischer Aufbau gewährleistet und Berechtigungskonflikte vermieden werden.

Das Berechtigungshierarchiediagramm ist mit dem Rollendiagramm eng verbunden. Die aufgeführten Berechtigungen werden in der Rollenbeschreibung zur Definition des Anforderungsprofils verwendet. Der Aufbau entspricht dem eines Funktionsbaums.

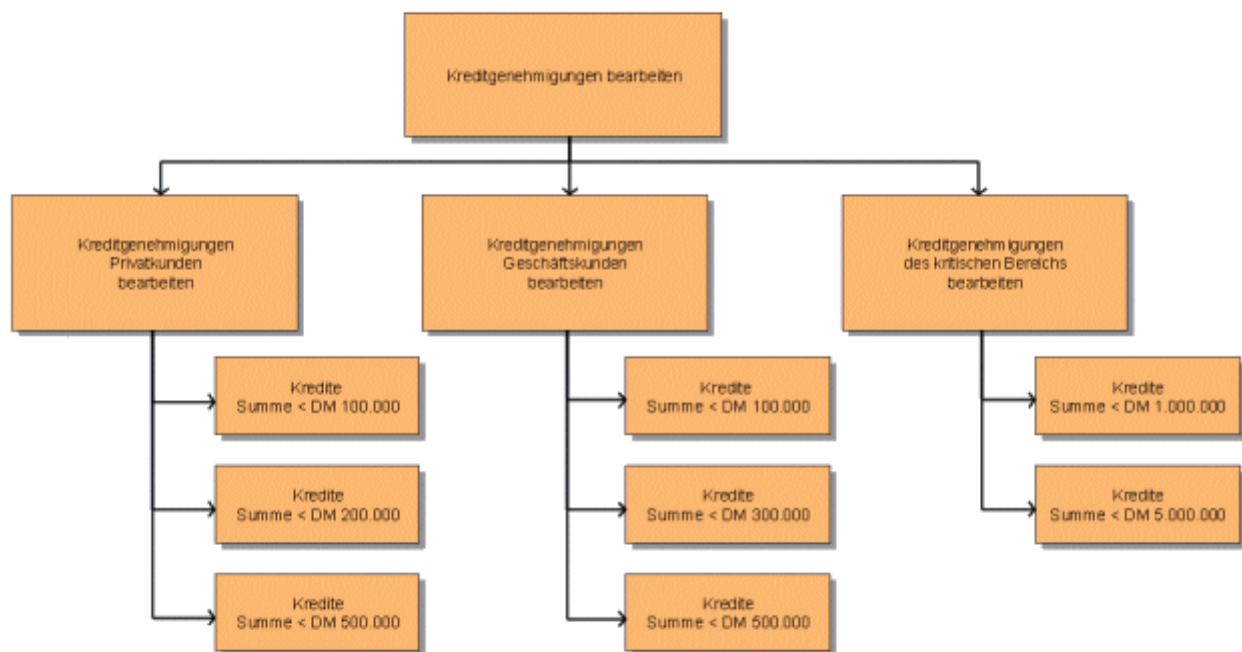


Abbildung 58: Berechtigungshierarchie

4.2.1.9 Datenmodelle der Prozesskostenrechnung

4.2.1.9.1 CD-Diagramm

Anwendungsgebiet des CD-Diagramms (Cost Driver-Diagramm) ist die Prozesskostenrechnung, zum Beispiel mit ARIS Business Optimizer. Im CD-Diagramm wird die Hierarchie der Cost Drivers abgebildet.

Ein Cost Driver (deutsch: Kostentreiber) ist eine für einen bestimmten Prozess aussagekräftige Maß-/Bezugsgröße der Kostenverursachung. Die Bezugsgröße sollte eine leicht aus verfügbaren Informationsquellen ableitbare operationale Größe sein, die sich proportional zur Kostenverursachung verhält.

Folglich lassen sich Kostentreiber nur für leistungsmengenvariable und leistungsmengeninduzierte Prozesse definieren. Für leistungsmengenneutrale Prozesse, wie z.B. **Abteilung leiten**, lassen sich keine Kostentreiber definieren. Beispiel eines Kostentreibers ist **Straßenlänge** für den Prozess **Straße teeren**.

Im CD-Diagramm wird die Hierarchie der Kostentreiber durch gerichtete Kanten des Typs **bestimmt Menge von** abgebildet. An diesen Kanten müssen die Attribute **CD-Verhältnis Zähler** und **CD-Verhältnis Nenner** gepflegt sein. Ist **CD-Verhältnis Nenner** nicht gepflegt, wird ein Wert von **1** angenommen. Der Quotient dieser beiden Attribute legt das Mengenverhältnis zwischen den beiden Kostentreibern für die Prozesskalkulation fest.

Ein Beispiel gibt die folgende Abbildung wieder: Es existieren die beiden Kostentreiber **Anzahl Pkw (Limousine)** und **Anzahl Türen**. Um die Tatsache darzustellen, dass jede Limousine vier Türen hat, muss an der Kante von dem Kostentreiber **Anzahl Pkws (Limousine)** zu dem Kostentreiber **Anzahl Türen** das Attribut **CD-Verhältnis Zähler** auf **4** gesetzt werden.

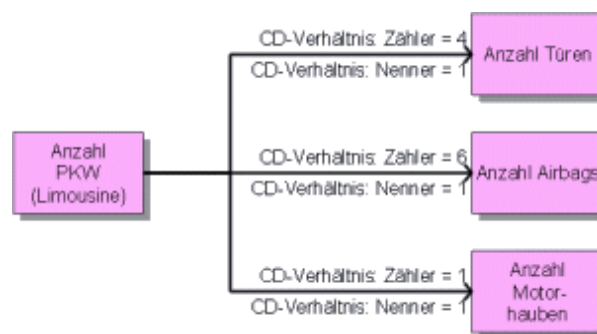


Abbildung 59: Beispiel eines CD-Diagramms

In den Tabellen der Prozesssicht werden den einzelnen Prozessen die Kostentreiber zugeordnet. Für jede Funktion in den Prozessen kann der Einsatzfaktor automatisch aus der Kostentreiberhierarchie ermittelt werden.

4.2.1.9.2 Kostenartendiagramm

Anwendungsgebiet des Kostenartendiagramms ist die Prozesskostenrechnung, zum Beispiel mit ARIS Business Optimizer. Im Kostenartendiagramm wird die Hierarchie von Kostenarten abgebildet.

Kostenarten dienen der systematischen Gliederung aller Kosten, die bei der Erstellung und Verwertung der Kostenträger (Leistungen) entstehen. Die Fragestellung lautet: Welche Kosten sind angefallen?

Beispielsweise sind Materialkosten die Kostenarten für den Verbrauch von Stoffen und Abschreibungen die Kostenart, die die Wertminderungen der Anlagegüter erfasst.

Die gesamten Kosten lassen sich nach verschiedenen Kriterien systematisieren. Werden die Kosten nach der Art der verbrauchten Produktionsfaktoren eingeteilt, ergibt sich eine Gliederung in Personalkosten (z. B. Gehälter, Provisionen), Sachkosten (z. B. Kosten für Rohstoffe, Abschreibungen auf Maschinen), Kapitalkosten, Kosten für Dienstleistungen Dritter (z. B. Transportkosten, Kosten für Strom) und Kosten für Steuern, Gebühren und Beiträge. Die Kostenarten lassen sich ferner nach den wichtigsten betrieblichen Funktionen gliedern in Kosten der Beschaffung, Kosten der Lagerhaltung, Kosten der Fertigung, Kosten der Verwaltung und Kosten des Vertriebs. Beide Gliederungen können weiter verfeinert werden.

Im Kostenartendiagramm wird die Hierarchie der Kostenarten durch gerichtete Kanten des Typs **ist übergeordnet** abgebildet.

Ein wichtiges Attribut für Kostenarten ist der **Leistungsmaßstab**. Er beschreibt die Einheit, in der die Leistung der Kostenart bemessen wird (z. B. Stunden für Löhne und Quadratmeter für Raumkosten).

Die folgende Abbildung zeigt ein Beispiel für ein Kostenartendiagramm entsprechend der oben angegebenen Gliederung anhand der Art der verbrauchten Produktionsfaktoren mit einer weiteren Untergliederung der Personalkosten.

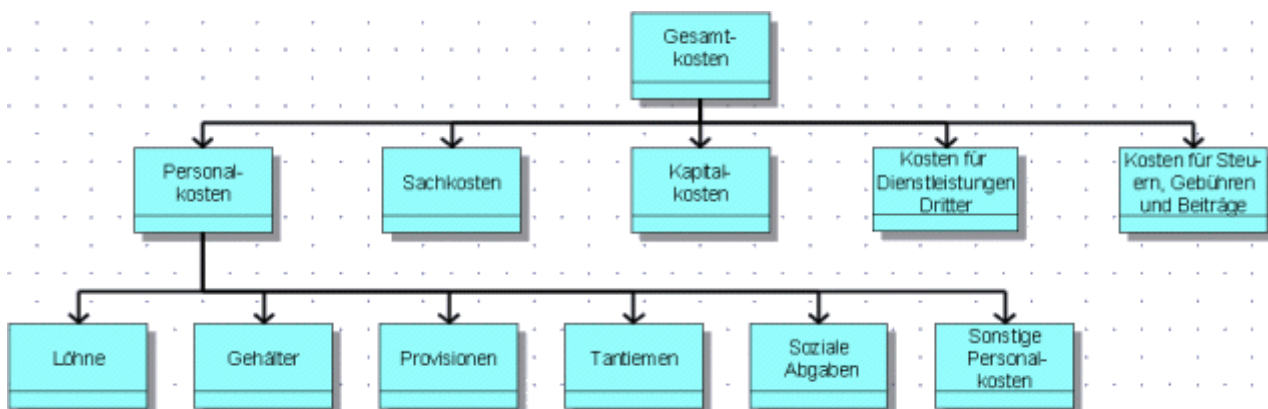


Abbildung 60: Beispiel eines Kostenartendiagramms

Das Kostenartendiagramm visualisiert die Abhängigkeiten zwischen den Kostenarten. Diese Hierarchisierung kann Teil einer Berechnung in ARIS Business Optimizer sein. So kann ARIS Business Optimizer beispielsweise so konfiguriert werden, dass die errechneten Kosten der einzelnen Kostenarten gemäß der Hierarchie im Kostenartendiagramm aggregiert werden.

4.2.1.10 Datenmodell des Projektmanagements

4.2.1.10.1 Informationsträgerdiagramm

Das Informationsträgerdiagramm ist ein optionaler Bestandteil des Projektmanagements mit ARIS Business Architect. Es ist dem Fachkonzept der Datensicht zugeordnet und dient der Dokumentation ein- und ausgehender Daten in Form von Dokumenten, Protokollen und ARIS-Modellen.

Als Hinterlegung einer Clusterausprägung können ARIS-Modelle in der PPK (Projektprozesskette, siehe Fachkonzept der Prozesssicht) beschrieben werden. Somit lässt sich eine grobe Spezifikation der Daten im zugehörigen Cluster erzielen. Die tatsächlich benötigten Dokumente, z. B. eine Textverarbeitungsdatei, sind somit explizit darstellbar und durch die Attribute **Verknüpfung 1** bis **Verknüpfung 4** von ARIS Business Architect aufrufbar.

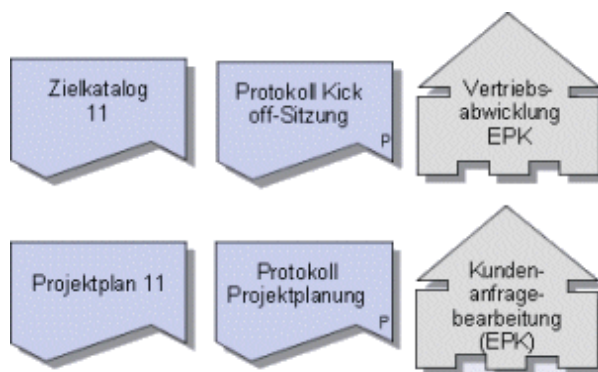


Abbildung 61: Informationsträgerdiagramm

4.2.2 DV-Konzept

4.2.2.1 Relationendiagramm, Attributzuordnungsdiagramm

Die im Fachkonzept entworfenen logischen Datenstrukturen werden im DV-Konzept in eine Beschreibungsform umgewandelt, auf die konkrete Datenbanksysteme aufbauen können. In ARIS wird hierfür das Relationenmodell angeboten.

Zur Definition der vorhandenen Relationen und Attribute sowie ihrer Beziehungen zu den im Fachkonzept eingeführten Informationsobjekten stehen das Relationendiagramm und das Attributzuordnungsdiagramm zur Verfügung.

Im Relationendiagramm können zunächst die benötigten Relationen definiert werden.

Eine Relation beschreibt einen Entitytyp durch seine Attribute. Sie ist eine Teilmenge der möglichen Kombinationen der Wertebereiche der einzelnen Attribute.

Relationen werden grafisch wie folgt dargestellt:

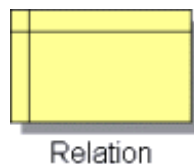


Abbildung 62: Grafische Darstellung der Relation

Jeder Entitytyp des eERM bildet nun im Relationenmodell eine Relation. Bei der Umsetzung der Beziehungstypen des eERM-Modells spielt die Kardinalität bei der Entscheidung, ob eine eigene Relation für den Beziehungstyp gebildet wird, eine große Rolle. $n:m$ - Beziehungen erfordern im Gegensatz zu $1:n$ - Beziehungen die Abbildung in eigenen Relationen.

Im Relationendiagramm kann nun für jede einzelne Relation gezeigt werden, welchen Entitytypen bzw. Beziehungstypen des eERM-Modells sie abbildet.

Zur näheren Spezifizierung der Relation ist auch eine Aufzählung ihrer Attribute möglich. Hierbei kann durch Auswahl der entsprechenden Kante zwischen Relation und Attribut auch definiert werden, ob es sich bei dem entsprechenden Attribut um ein Schlüsselattribut, ein Fremdschlüsselattribut oder ein beschreibendes Attribut handelt. Für jedes einzelne Attribut kann wiederum die Beziehung zum ERM-Attribut des Fachkonzeptes, das es abbildet, hergestellt werden.

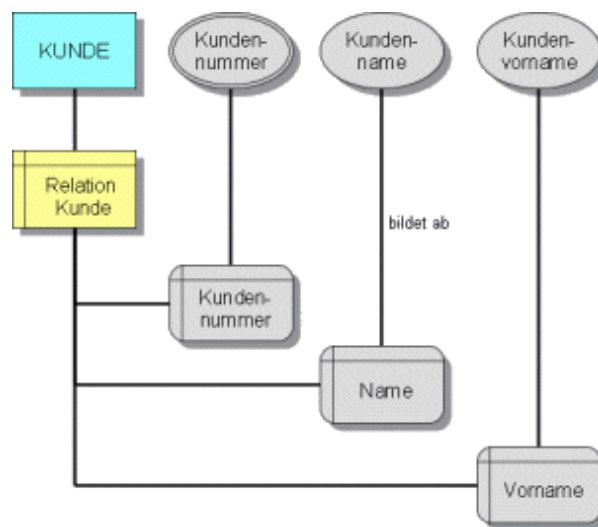


Abbildung 63: Zuordnung der Attribute und der Datenobjekte des Fachkonzeptes

Zur Reduktion der Abbildungskomplexität können die Attribute jeder Relation auch in einem Attributzuordnungsdiagramm definiert werden, das mit der Relation verknüpft wird. Ein Beispiel hierfür zeigt die nächste Abbildung.

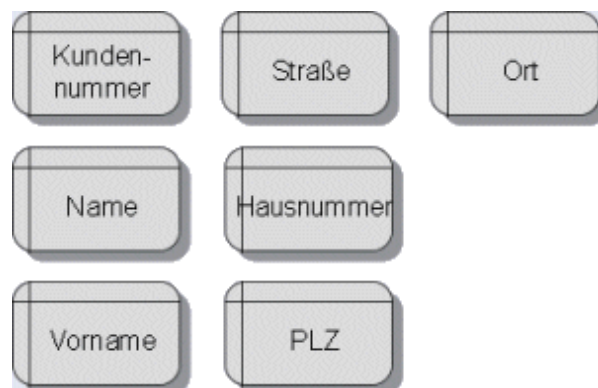


Abbildung 64: Attributzuordnungsdiagramm

Die Umsetzung der Datencluster des Fachkonzeptes erfolgt im DV-Konzept durch einen eigenen Objekttyp, die View. In Anlehnung an die Definition der Datencluster ist die View wie folgt definiert:

Eine View beschreibt eine logische Sicht auf eine Ansammlung von Relationen.

Welche Relationen einer View zugeordnet sind, kann ebenfalls im Relationendiagramm dargestellt werden.

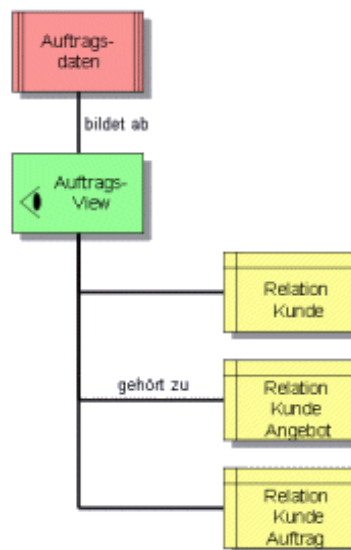


Abbildung 65: Definition einer View

1:n-Beziehungen des ERM-Modells werden nicht durch eigene Relationen im Relationenmodell abgebildet. Die Beziehung wird durch Aufnahme des Schlüsselattributs des übergeordneten Entitytyps in die Relation des untergeordneten Entitytyps abgebildet. Das ursprüngliche Schlüsselattribut wird hierbei zum Fremdschlüssel der Relation.

Welches Attribut des Relationenmodells den Beziehungstypen des ERM-Modells abbildet, kann ebenfalls durch eine Kante im Relationendiagramm dargestellt werden.

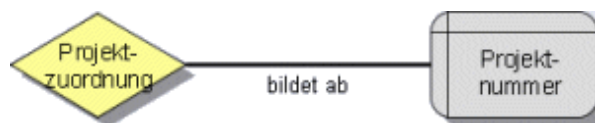


Abbildung 66: Zuordnung des ERM-Beziehungstyps zum Attribut

Eine Zusammenfassung der Objekttypen und Beziehungstypen des Relationenmodells befindet sich im Handbuch **ARIS-Methode – Tabellen** (Datei **method_tabulations_aris.pdf**) auf Ihrem Installationsmedium.

4.2.2.2 Modellierung der Systemschnittstellenmodelle - Systemattribute, Systemattributdomäne

Der Modelltyp **Systemattribute** ist insbesondere für datenexportorientierte Aufgabenstellungen aus ARIS Business Architect gedacht. Mit ihm lassen sich Entitytypen, Ereignisse, Fachbegriffe, Funktionen, Informationsträger, Organisationseinheiten und Personen hierarchisieren und entsprechend ihrer datentechnischen Anforderungen eindeutig und umfassend spezifizieren. Diese Daten lassen sich nach den datenbanküblichen Anforderungen in Primär- und Fremdschlüssel, beschreibende und Mussfelder typisieren. Um die Domärentypen dieser Datenobjekte zu bestimmen, ist hier wiederum eine Hinterlegung mit dem Modelltyp **Systemattributdomäne** (s. u.) möglich.

Im Gegensatz zu den ERM-Attributen liegt der Schwerpunkt bei den Systemattributen in der Repräsentation und Verwaltung von schnittstellenorientierten Daten. Um eine hohe Flexibilität bezüglich der zu exportierenden Inhalte zu gewährleisten, enthalten die Systemattributobjekte zwei Wertfelder, die mit den relevanten Informationen gefüllt werden können.

Das folgende Beispiel zeigt einen Ausschnitt aus der Projektkopfdefinition eines in ARIS Business Architect definierten Projektes für die Übertragung an ein System zum Projektmanagement.

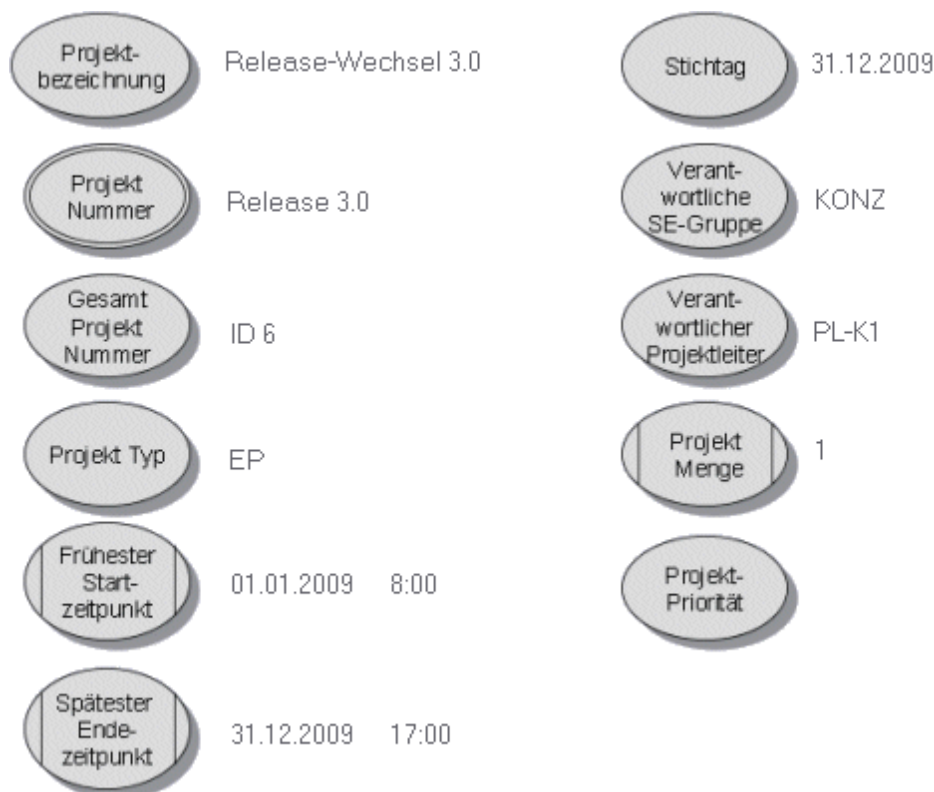


Abbildung 67: Beispiel für ein Modell "Systemattribute"

Der Modelltyp **Systemattributdomäne** dient der datentypspezifischen Definition der Systemattributobjekte, also beispielsweise der Angabe des Domäentyps (char, int, date, ...) und der Feldlänge. Der Hauptanwendungsfall liegt hier in der Bereitstellung von Informationen im Rahmen des Exports an Fremdsysteme.

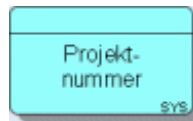


Abbildung 68: Systemattributdomäne

4.2.3 Implementierung - Tabellendiagramm

Das Tabellendiagramm erlaubt die Beschreibung der Tabellen und Felder eines Datenbanksystems. Die grafische Darstellung von Tabellen und Feldern wird in folgender Abbildung gezeigt.

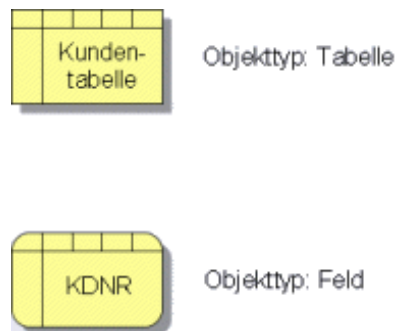


Abbildung 69: Grafische Darstellung von Tabelle und Feld

Für jede Tabelle können die einzelnen Felder, die dieser Tabelle zugeordnet sind, dargestellt werden. Dem einzelnen Feld kann zur näheren Spezifikation ein Sortier-Index und die Domäne zugeordnet werden. Ein Beispiel hierfür zeigt diese Abbildung.

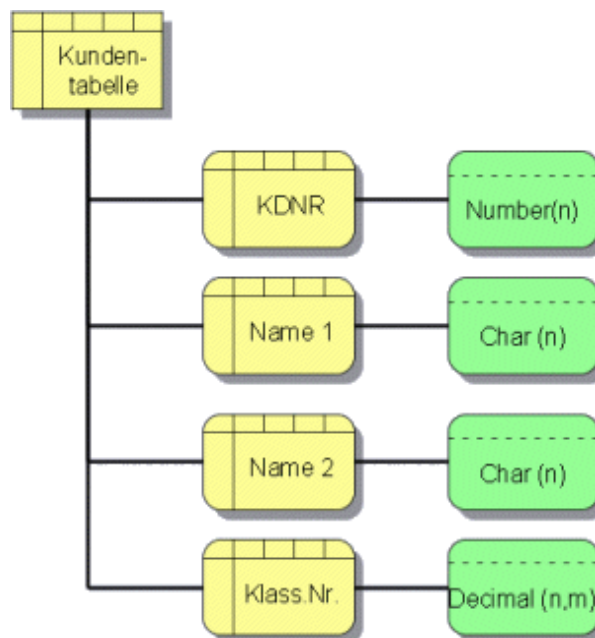


Abbildung 70: Feldzuordnungen

Da Relationen des Relationenmodells nicht unbedingt 1:1 in Tabellen und Felder umgesetzt werden (z. B. aus Gründen der Datenbank-Performance), sind auch mehrwertige Beziehungen zwischen Tabellen und Relationen oder Entitytypen denkbar. Im Tabellendiagramm sind diese Beziehungen durch Auswahl der entsprechenden Kanten darstellbar. Die im Fachkonzept definierten Datencluster oder im Relationenmodell definierten Views finden im Tabellendiagramm ihre Entsprechung in dem Objekt **View (phys.)**.

Da die Umsetzung oder Dokumentation der im Unternehmen verwendeten Datenbanktabellen und -Felder nicht unbedingt über die Definition eines Relationenschemas erfolgen muss, sind

sowohl die Realisierungsbeziehungen zwischen Relationen (bzw. Attributen) und Tabellen (bzw. Feldern) als auch zwischen Entitytypen (bzw. ERM-Attributen) und Tabellen (bzw. Feldern) darstellbar.

Es kann entweder gezeigt werden, welche Relationen und Attribute durch die dargestellten Tabellen und Felder realisiert werden oder - ohne den Weg über die Relationendefinitionen zu gehen - welche Entitytypen, Beziehungstypen und ERM-Attribute durch die dargestellten Tabellen und Felder abgebildet werden. Beispiele für beide Darstellungsformen werden in der nächsten Abbildung gezeigt.

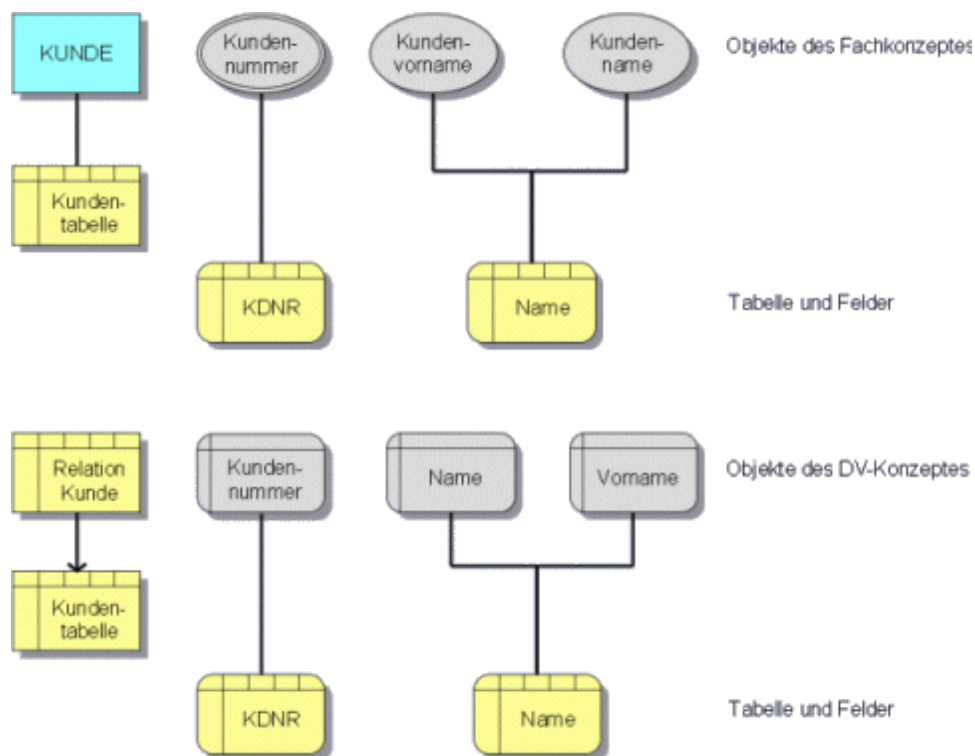


Abbildung 71: Zuordnung der Objekte des Fachkonzeptes und des DV-Konzeptes

Um den genauen Standort bestimmter Tabellen und Felder im Unternehmen definieren zu können, muss es auch möglich sein, jedes einzelne Exemplar einer Tabelle definieren zu können. Gleiches gilt dann, wenn die Zugriffsrechte von Organisationseinheiten auf Tabellen und Felder festgelegt werden sollen. Der bisher vorgestellte Objekttyp Tabelle definiert auf der **Typ-Ebene** den logischen Aufbau einer physikalischen Tabelle und ihrer Felder. Von jeder so definierten Tabelle können im Unternehmen jedoch mehrere Exemplare, verteilt an verschiedenen Standorten, gespeichert auf verschiedenen Medien existieren. Um dies darstellen zu können, werden die Objekttypen Tabelle (Exemplar) und Feld (Exemplar) eingeführt.

Mit Hilfe dieser Objekte kann nun genau definiert werden, wie viele Exemplare einer Tabelle oder eines Feldes existieren. Die folgende Abbildung verdeutlicht diesen Zusammenhang.

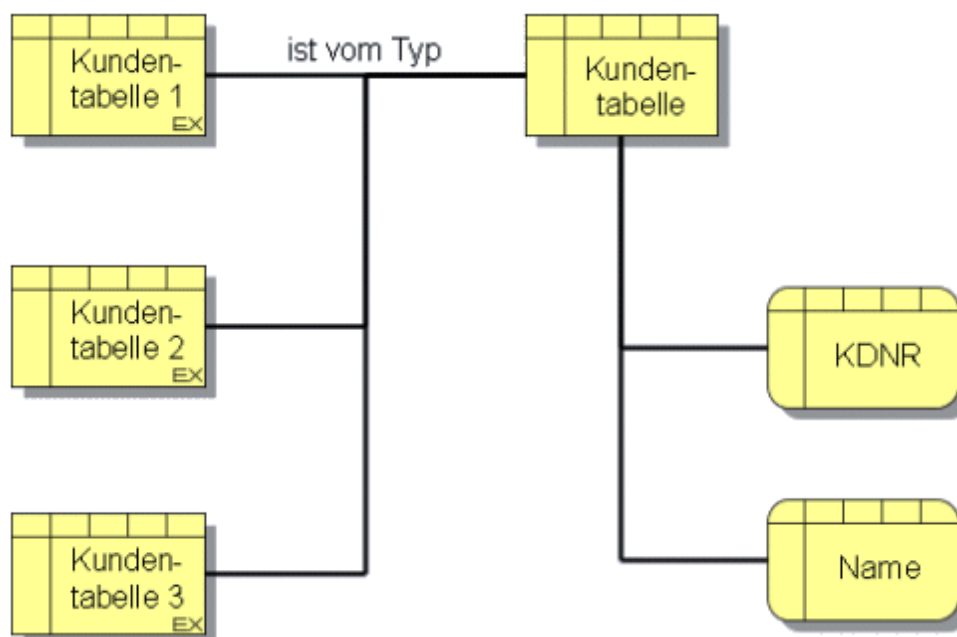


Abbildung 72: Tabellenexemplare

Eine Zusammenstellung der Objekte und Beziehungen des Tabellendiagramms befindet sich im Handbuch **ARIS-Methode – Tabellen** (Datei **method_tabulations_aris.pdf**) auf Ihrem Installationsmedium.

4.3 Organisationssicht

4.3.1 Fachkonzept

4.3.1.1 Organisatorische Gestaltung der Unternehmen

Unternehmen stellen komplexe soziale Gebilde dar, die in überschaubare Einheiten unterteilt werden. Zur Komplexitätsbewältigung werden Ordnungsmuster definiert und Regeln festgelegt. Das Ergebnis dieses Ordnungsprozesses wird dann als Organisation bezeichnet. Welche Rolle organisatorische Betrachtungen im Zusammenhang mit der Entwicklung von Informationssystemen spielen, wurde bisher nur selten untersucht. Neuere Unternehmenskonzepte wie z. B. Lean Produktion, Lean Management oder CIM haben jedoch einen engen Bezug zur organisatorischen Gestaltung des jeweiligen Betrachtungsumfeldes. Das ARIS-Konzept besitzt aus diesem Grunde eine eigenständige Beschreibungssicht für die Organisation.

Bei der organisatorischen Strukturierung des Unternehmens können die Aufbauorganisation und die Ablauforganisation unterschieden werden.

Die Aufbauorganisation umfasst die Regeln zur statischen Strukturierung des Unternehmens. Die Ablauforganisation beinhaltet die Regeln, die sich auf die auszuführenden Aufgaben des Unternehmens beziehen. Diese aufgabenbezogene Strukturierung im Sinne einer Verteilung von Funktionen auf Aufgabenträger wird in der Steuerungssicht des ARIS-Hauses behandelt. Die Organisationssicht umfasst somit hauptsächlich die Betrachtung der aufbauorganisatorischen Gestaltung.

Die Gestaltung der optimalen Organisation eines Unternehmens im Sinne der Erreichung eines möglichst geringen Koordinationsaufwands, richtet sich nach ihren Umfeldbedingungen und Zielsetzungen. Es können daher keine allgemein gültigen optimalen Organisationsstrukturen im Sinne von Referenzstrukturen definiert werden.

Die Strukturierung der organisatorischen Einheiten kann sich an verschiedenen Kriterien orientieren.

Sehr verbreitet ist die funktionale Gliederung. Hierbei wird jeweils eine Unternehmensfunktion (Beschaffung, Produktion, Finanz- und Rechnungswesen, Absatz) für alle Produkte und Gebiete zuständig gemacht. Dem Vorteil der hohen Spezialisierung der eingesetzten Mitarbeiter steht jedoch der Nachteil des hohen Kommunikations- und Koordinierungsaufwandes zwischen den Funktionalbereichen gegenüber.

Auch der Entwurf und Einsatz von Informationssystemen hat sich lange an dieser funktionalen Zergliederung der Unternehmen orientiert. Bei der Betrachtung ganzheitlicher Prozessketten im Sinne der zusammenhängenden Bearbeitung gleichartiger Datenobjekte ist aber bei einer solchen Gliederung der Zusammenhang der einzelnen Funktionen nur schwierig herzustellen.

Dies führte im Umfeld der Diskussion um die integrierte Datenverarbeitung zu der Forderung nach einer einheitlichen Datenbasis zur Unterstützung der unterschiedlichen Funktionen. Die angestrebte Integration der Funktionen führt jedoch dazu, dass der durch die funktionale Gliederung angestrebte Zweck der Komplexitätsreduzierung wieder aufgehoben wird.

Deswegen werden häufig bei der Zielsetzung der Funktionsintegration andere Kriterien der organisatorischen Zerlegung angewendet.

Dies kann z. B. eine Zerlegung nach Kriterien wie Gebiete oder Produkte sein. In folgender Abbildung wird eine Zergliederung nach Produkten schematisch dargestellt (vgl. Scheer, Wirtschaftsinformatik 1994, S. 26 f).

Bei einer gebietsorientierten Aufbauorganisation werden die Organisationseinheiten entsprechend der örtlichen Verteilung des Unternehmens oder Unternehmensbereiches festgelegt. Sie eignet sich insbesondere für Vertriebsfunktionen, da regionale Faktoren wie z. B. unterschiedliche Gesetzgebungen gezielter berücksichtigt werden können.

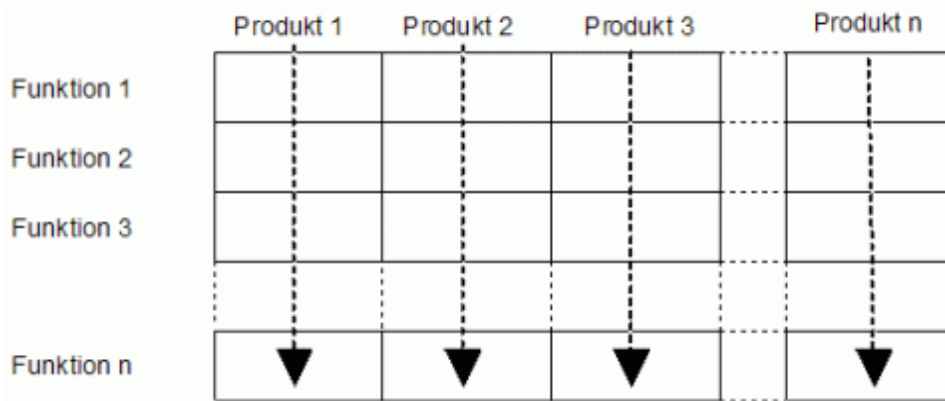


Abbildung 73: Organisatorische Zergliederung nach Produkten

Eine produktbezogene Aufbauorganisation definiert für Produkte bzw. Produktgruppen Organisationseinheiten. Innerhalb einer Produktgruppe werden dann möglichst viele Funktionen, die für die entsprechende Produktgruppe relevant sind, integriert. Zielsetzung ist hierbei, den durch eine funktionale Gliederung auftretenden Kommunikationsaufwand zu verringern. Auf der anderen Seite entsteht jedoch nun ein Vermittlungsaufwand zwischen den produktgruppen-orientierten Teilsystemen.

Um diesen Auswirkungen entgegenzuwirken werden oft gemischte Organisationsformen gebildet. Für das Beispiel Einkauf wird dies in der folgenden Abbildung dargestellt (vgl. Scheer, Wirtschaftsinformatik 1994, S. 26 f).

	Produktgruppe 1	Produktgruppe 2	Produktgruppe 3
zentraler Einkauf		Lieferantenauswahl	
		Vertragsgestaltung	
Disposition			
Bestellung			
Rechnungsprüfung			

Abbildung 74: Gemischte Organisationsformen

Eine rein funktionale Gliederung würde bedeuten, dass ein zentraler Einkauf für alle Produktgruppen zuständig wäre. Hierbei könnten zwar alle Synergieeffekte, die zwischen den Produktgruppen bestehen, genutzt werden; die Durchführung eines Einkaufsvorgangs durch alle Teil-

funktionen würde aber zu erheblichen Abstimmungsproblemen führen. Eine Aufteilung der Einkaufsfunktionen auf die unterschiedlichen Produktgruppen führt dazu, dass für jede Produktgruppe eine Einkaufsabteilung eingerichtet wird, die alle Einkaufsfunktionen wahrnimmt. Synergieeffekte z. B. bei der Lieferantenauswahl und bei Verhandlungen über Rahmenverträge könnten hierbei nur mit einem hohen Abstimmungsaufwand erzielt werden.

Bei der oben dargestellten Zerlegung werden die Einkaufsfunktionen, bei denen hohe Synergieeffekte erwartet werden, funktional zerlegt, d. h. sie werden von einer zentralen Einkaufsstelle wahrgenommen. Die Funktionen, die spezielle Bedürfnisse und Restriktionen der einzelnen Produktgruppen zu berücksichtigen haben, werden objektorientiert nach Produktgruppen zergliedert. Sie können somit zeitnah in die Prozessabläufe der einzelnen Produktgruppen eingebunden werden. Damit wird die operative Abwicklung der Prozesse in den dezentralen Einheiten durchgeführt, während die zwischen den dezentralen Einheiten bestehenden Beziehungen auf der übergeordneten, zentralen Koordinationsebene berücksichtigt werden.

Im ARIS-Konzept werden diesen flexiblen Organisationsformen aufgrund der betont prozessorientierten Betrachtung besondere Beachtung geschenkt. Die Bildung von Aufbauorganisationen unter paralleler Betrachtung mehrerer Zerlegungskriterien wird insbesondere auch von abrechnungsorientierten Ansätzen wie z. B. dem Profit-Center-Konzept gefordert.

4.3.1.2 Organigramm

Eine typische Darstellungsart für Organisationsstrukturen ist das Organigramm. In ihm werden je nach den gewählten Strukturierungskriterien die gebildeten Organisationseinheiten (als Aufgabenträger) und ihre Verknüpfungen untereinander dargestellt.

Organisationseinheiten sind Träger der zur Erzielung der Unternehmensziele durchzuführenden Aufgaben.

Die Verknüpfungen stellen die Verbindungen zwischen den Organisationseinheiten dar. In folgender Abbildung wird hierfür ein Beispiel gezeigt.

Zur genaueren Spezifizierung der Unterstellungsverhältnisse werden verschiedene Kantentypen zwischen Organisationseinheiten unterschieden. Eine Kante kann hierbei bedeuten:

- ist fachlich vorgesetzt
- ist disziplinarisch vorgesetzt
- ist bildend für

Werden die funktionalen Zuständigkeiten erfasst, bildet das Organigramm die Verteilung der betrieblichen Aufgaben ab.

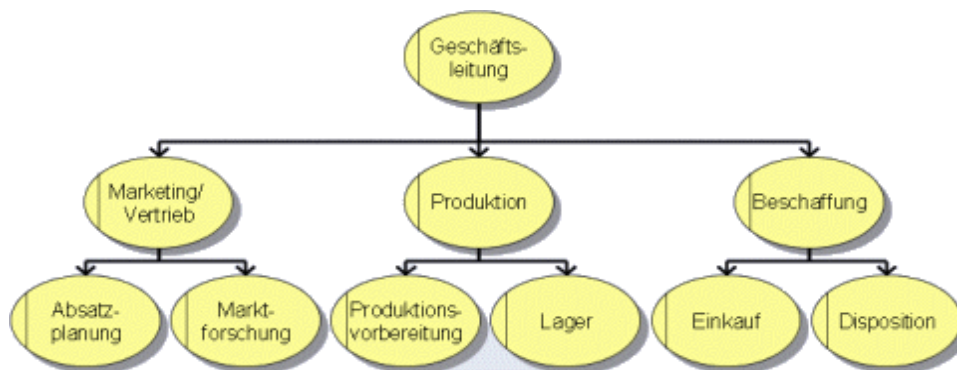


Abbildung 75: Organigramm

Zur Darstellung einzelner Stellen im Unternehmen, für die z. B. Stellenbeschreibungen existieren, steht ein eigener Objekttyp **Stelle** zur Verfügung. Dieser wird wie in der nächsten Abbildung dargestellt. Einer Organisationseinheit können mehrere Stellen zugeordnet werden. Die Bedeutung der Kanten entspricht der zwischen Organisationseinheiten.

Den Stellen und Organisationseinheiten können natürliche Personen, die diese Stellen besetzen, zugeordnet werden. Auch für Personen sind eigene Objekte in ARIS vorhanden, die ebenfalls in folgender Abbildung dargestellt werden. Die Zuordnung einer natürlichen Person zu einer Organisationseinheit drückt aus, dass diese Person als Mitarbeiter der Organisationseinheit zugeordnet ist; die Zuordnung zu einer einzelnen Stelle definiert die aktuelle Stellenbesetzung im Unternehmen. Ein Beispiel zeigt diese Abbildung.

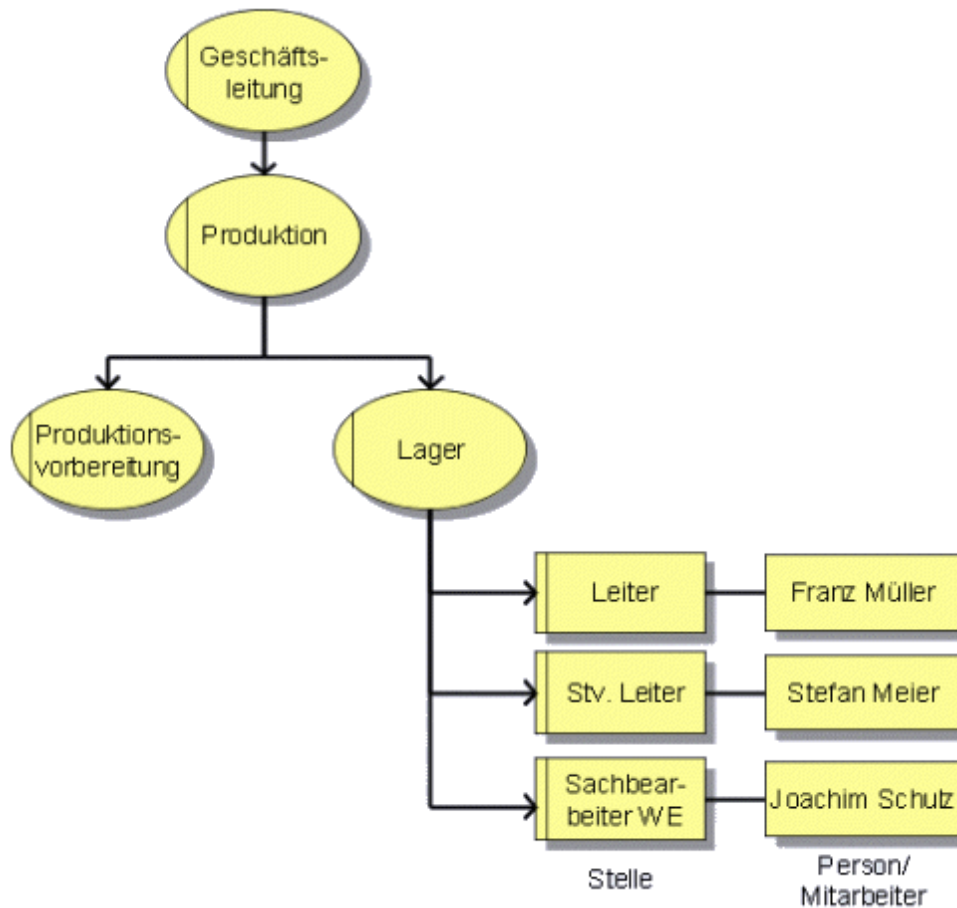


Abbildung 76: Organigramm mit Stellen- und Personenzuordnung

Organisationseinheiten und Personen können auch typisiert werden. Damit kann für eine Organisationseinheit z. B. definiert werden, ob es sich um eine Abteilung, Hauptabteilung oder Gruppe handelt; Personen können z. B. den Personentypen **Abteilungsleiter**, **Gruppenleiter** oder **Projektleiter** zugeordnet werden.

Zur Darstellung dieser Typisierung existieren die Objekte **Organisationseinheitstyp** und **Personentyp**. Ein Beispiel der erläuterten Typisierung von Organisationseinheiten und Personen zeigt diese Abbildung.

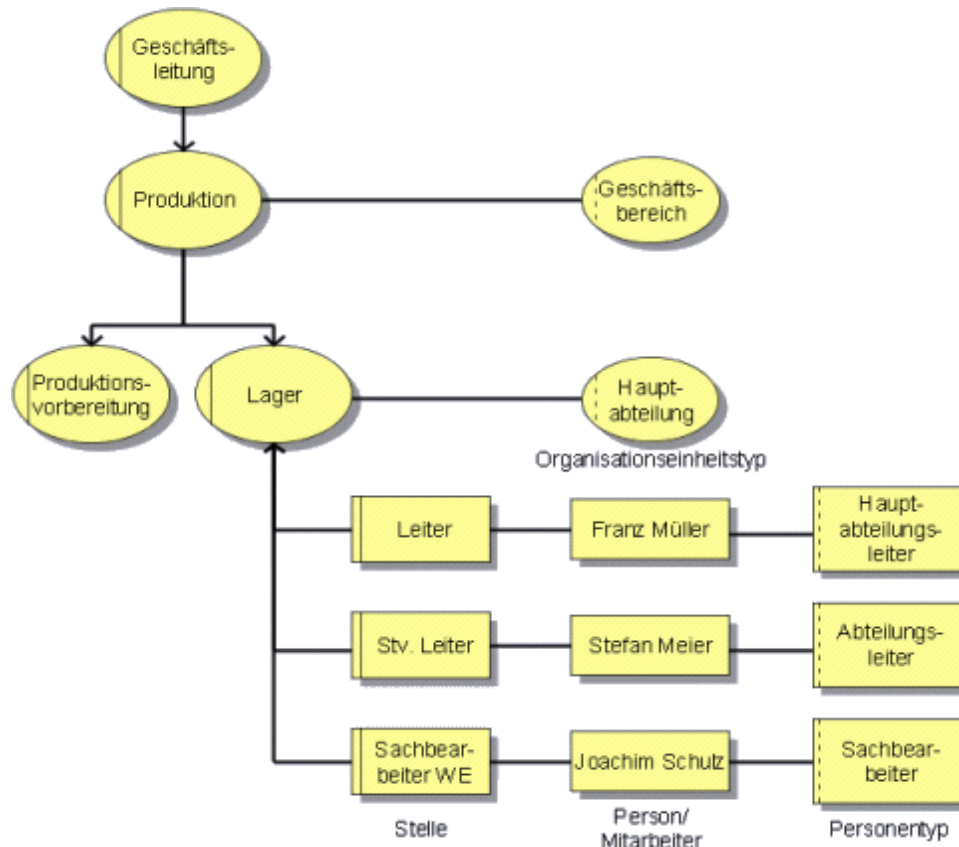


Abbildung 77: Personentypen

Die Verwendung dieser Objekttypen ermöglicht es, allgemeine Geschäftsregeln abzubilden, die von konkreten Organisationseinheiten oder Personen des Unternehmens abgeleitet sind. So besteht in Prozessketten die Möglichkeit zu definieren, dass nur bestimmte Personentypen eine Funktion ausführen dürfen oder nur bestimmte Personentypen Zugriff auf ein Informationsobjekt haben dürfen.

Die Modellierung der Aufbauorganisation des Unternehmens ist Ausgangspunkt für die in der DV-Konzept-Ebene zu definierenden Netztopologien, die diese Aufbauorganisation bestmöglich unterstützen sollen. Zur Definition der Netztopologie gehören Netzkanten und Netzknoten, die sich an bestimmten Standorten des Unternehmens befinden. Der Standort einer Organisationseinheit ist somit das wichtigste Bindeglied zwischen Fachkonzept und DV-Konzept der Organisationssicht. Jeder Organisationseinheit ist deshalb im Fachkonzept bereits der Standort zuzuordnen, an dem sie sich befindet. Ein Beispiel hierzu zeigt diese Abbildung.

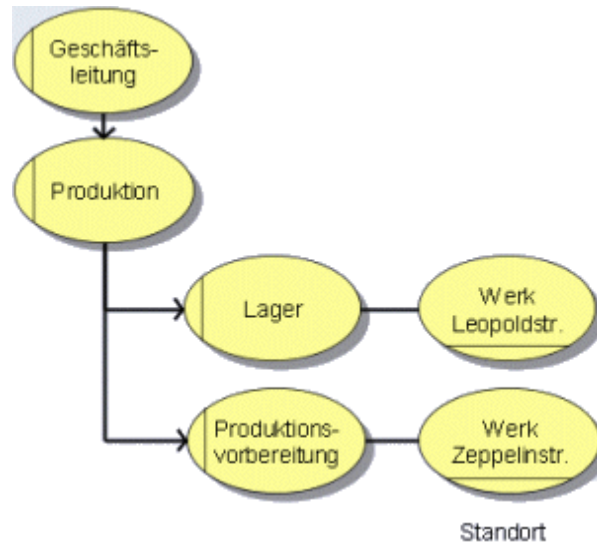


Abbildung 78: Standortzuordnungen

Standorte sind beliebig hierarchisierbar. Ein Standort kann ein gesamtes Werk, ein Gebäude, oder auch bei einer detaillierteren Betrachtung ein Büro bis hin zu einem einzelnen Arbeitsplatz in einem Zimmer sein. Damit ist es im DV-Konzept möglich, Netzknoten eines Netzes den einzelnen Arbeitsplätzen der organisatorischen Einheit zuzuordnen, also z. B. zu definieren, dass in einem Büro (Zimmer 102) insgesamt 3 Netzknoten zur Verfügung stehen müssen.

Diese Abbildung zeigt das Beispiel einer Standorthierarchisierung.

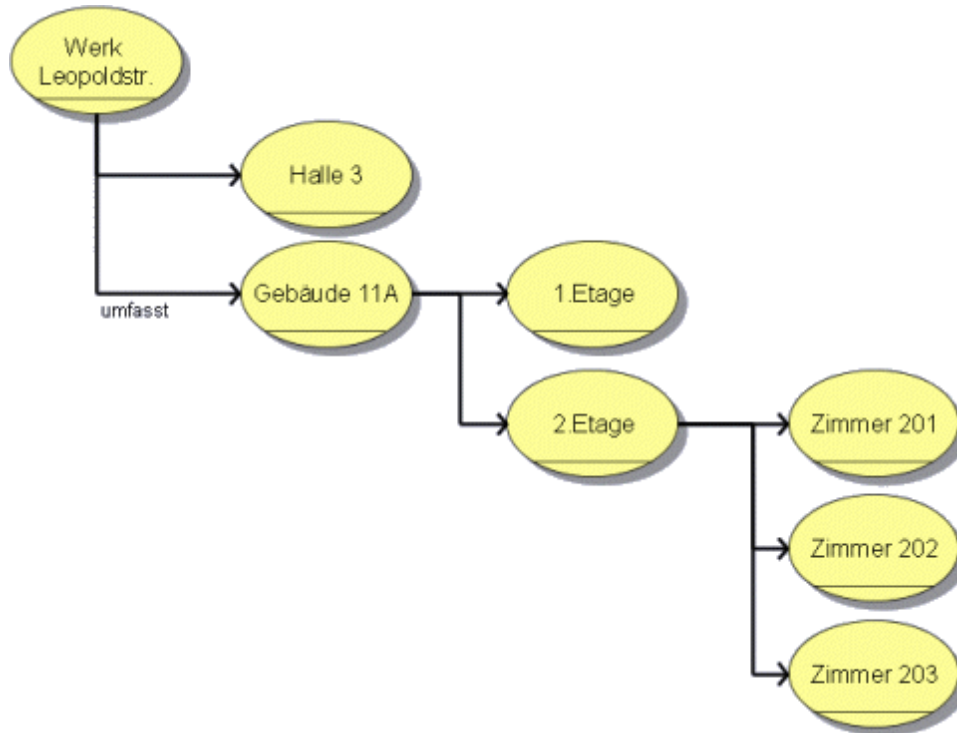


Abbildung 79: Standorthierarchien

4.3.1.3 Schichtkalender

Personal- und technischen Ressourcen können Schichtkalender hinterlegt werden, die spezifizieren, wann eine Ressource überhaupt verfügbar ist.

Die Zuordnung von Schichtkalendern zu Ressourcen geschieht im Organigramm oder in der EPK. Grundsätzlich können jeder Personal- bzw. technischer Ressource Schichtkalender zugeordnet werden. Bei einer Hierarchie von Personalressourcen gilt immer der jeweilige Kalender auf der am tiefsten liegenden Hierarchiestufe.

Der Schichtkalender ist ein mehrstufiges Objektmodell. Auf der untersten Stufe befinden sich Objekte vom Typ **Pause**. Eine Pause ist das tägliche Zeitintervall innerhalb einer Schicht, während dem nicht gearbeitet wird. Die Pause wird durch ihren relativen Anfang und ihre Dauer definiert. Dabei bezieht sich der relative Anfang immer auf die Schicht, der die Pause zugeordnet ist. Beginnt beispielsweise die Schicht um 8:00 Uhr, wenn die Pause einen relativen Anfang von 2 Stunden hat, beginnt die Pause um 10:00 Uhr.

Die nächste Hierarchiestufe beinhaltet Objekte vom Typ **Schicht**. Eine Schicht ist das tägliche Zeitintervall, während dem gearbeitet wird. Die Schicht wird durch ihren relativen Schichtanfang und ihre Dauer angegeben. Eine Schicht kann mehrere Pausen beinhalten. Die relativen Anfangszeiten der Pausen müssen innerhalb der Schichtzeiten liegen.

Typische Beispiele für Schichten sind Früh-, Mittags-, Nacht- und Tagesschicht. Jede Schicht wiederholt sich ständig im Abstand von 24 Stunden. Ein Schichtzyklus ist das wöchentliche oder mehrtägige Zeitintervall, während dem gearbeitet wird. Durch den Schichtzyklus wird also festgelegt, an welchen Tagen eine bestimmte Schicht gefahren wird und an welchen nicht. Der Schichtzyklus wird durch den relativen Zyklusanfang und die Zyklusdauer bestimmt. Soll ein Schichtzyklus ständig wiederholt werden, kann dies mit Hilfe des Attributs **Zyklisch wiederholen** definiert werden. Außerdem kann mit Hilfe des Attributs **Periode** bestimmt werden, in welchen Abständen ein Zyklus wiederholt werden soll.

Schichtzyklen umfassen häufig einen wöchentlichen oder zweiwöchentlichen Zeitraum. So kann ein Arbeiter beispielsweise in der einen Woche eine Frühschicht, in der nächsten Woche eine Mittagsschicht ableisten. Durch Schichtzyklen kann diese Sequenz permanent wiederholt werden.

Beispiel

Dem oben aufgeführten Beispiel folgend, kann man zwei Schichtzyklen definieren:

- Schichtzyklus Frühschicht:
 - Relativer Zyklusanfang = 0
 - Zyklusdauer = 5 Tage
 - Zyklisch wiederholen = ja
 - Periode = 14 Tage

- 2. Schichtzyklus Mittagsschicht:
 Relativer Zyklusanfang = 7 Tage
 Zyklusdauer = 5 Tage
 Zyklisch wiederholen = ja
 Periode = 14 Tage

Die einzelnen Schichten wiederholen sich in einem vierzehntägigen Rhythmus, d. h., die Periodizität beträgt zwei Wochen. Würde der gleiche Arbeiter alle vier Wochen samstags eine Frühschicht ableisten, könnte der dritte Schichtzyklus folgendermaßen definiert sein:

- Schichtzyklus Frühschicht:
 Relativer Zyklusanfang = 20 Tage
 Zyklusdauer = 1 Tag
 Zyklisch wiederholen = ja
 Periode = 28 Tage

Das beschriebene Beispiel finden Sie im Folgenden als Modell noch einmal dargestellt. Dabei ist eine 1 : n-Zuordnung von Schichten und Schichtzyklen zu erkennen.

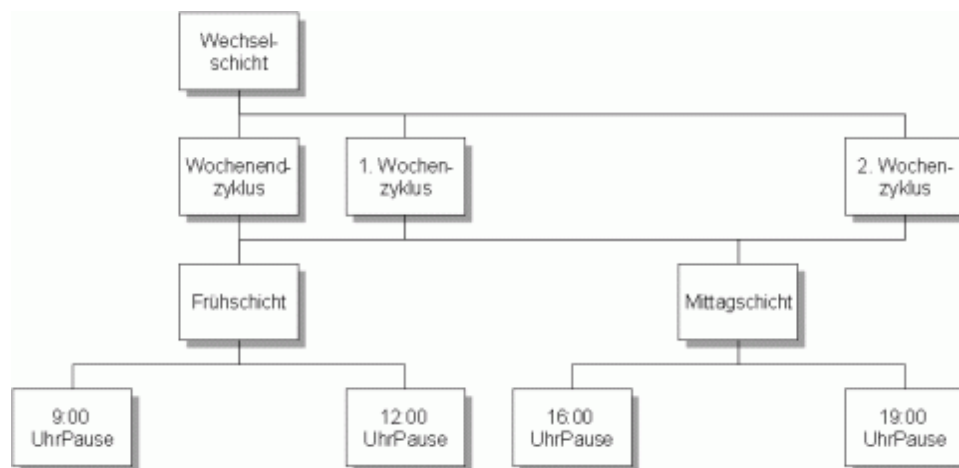


Abbildung 80: Beispiel eines Schichtkalenders

Ein Schichtkalender ist die Menge aller Schichtzyklen und zugehörigen Schichten, die die Arbeitszeit einer Ressource beschreiben. Diese Beschreibung enthält nur den periodisch wiederkehrenden Anteil; Sonderregelungen wie Urlaub, Krankheit, Feiertage oder sonstige arbeitsfreie Tage werden hier nicht erfasst.

Der Objekttyp **Schichtplan** besitzt die Attribute **Plananfang** und **Plandauer**. Diese Attribute spezifizieren den Zeitrahmen, während dem der Schichtkalender gültig ist. Außerdem existieren auch für den Schichtkalender die Attribute **Zyklisch wiederholen** und **Periode**.

4.3.2 DV-Konzept - Netztopologie

Die im Organigramm dargestellte Aufbauorganisation des Unternehmens kann nun durch Kommunikations- und Informationssystemlandschaften unterstützt werden. Die Anforderungen, die an den strukturellen Aufbau dieser Informationssysteme gestellt werden, sind im DV-Konzept zunächst allgemein in Form von Netztopologien definierbar.

Im Modell **Netztopologie** können zunächst unterschiedliche Netztypen aufgenommen werden. Ein Netztyp stellt eine Typisierung einzelner Netzexemplare dar, die sich auf exakt der gleichen technologischen Basis befinden.

Ein Beispiel für einen Netztyp zeigt diese Abbildung:

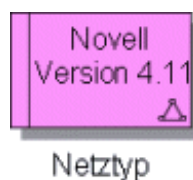


Abbildung 81: Grafische Darstellung eines Netztyps

Netztypen können untereinander verbunden sein und sind als logische Konstrukte auch hierarchisierbar.

Jedem Netztyp sind Des Weiteren mögliche Netzknontypen und Netzkantentypen zuordenbar. Damit können technologische Beschränkungen, die sich aus der Auswahl eines bestimmten Netztyps für ein Unternehmen ergeben, sofort erkannt werden. Für jeden Netzkantentyp kann dargestellt werden, in welchen Netzknontypen er enden kann.

Bei den Hardwarekomponententypen kann es sich zum einen um Netz-Hardware zur Realisierung der definierten Netzstrukturen oder um Hardwarekomponententypen handeln, die an Netzknontypen angeschlossen werden können.

Wie schon bei Anwendungssystemtypen oder Netztypen handelt es sich auch bei den Hardware-Komponententypen nicht um einzelne z. B. durch Inventarisierungsnummern des Unternehmens identifizierbare Exemplare von Hardwarekomponenten, sondern um Typisierungen auf der gleichen technologischen Basis. Hardwarekomponententypen sind beliebig hierarchisierbar. Ein Hardwarekomponententyp stellt eine Typisierung einzelner Exemplare von Hardwarekomponenten dar, die sich auf exakt der gleichen technologischen Basis befinden.

In Verbindung mit den Netzknontypen- und -kantentypen kann nun eine Art Referenzmodell der Netztopologie aufgebaut werden. Hierbei ist darstellbar, welche Hardwarekomponententypen zur Realisierung bestimmter Netzkantentypen oder Netzknontypen verwendet werden können. Bei den Kantentypen kann es sich dabei z. B. um einen bestimmten Typ eines Übertragungskabels handeln. Es kann aber auch dargestellt werden, welcher Hardwarekomponententyp an welchen Netzknontyp anschließbar ist. Netzknontypen können auch mit Hardwarekomponententypen, die zur Realisierung von Knontypen verwendet werden, in Beziehung stehen. Ein Beispiel hierzu zeigt diese Abbildung.

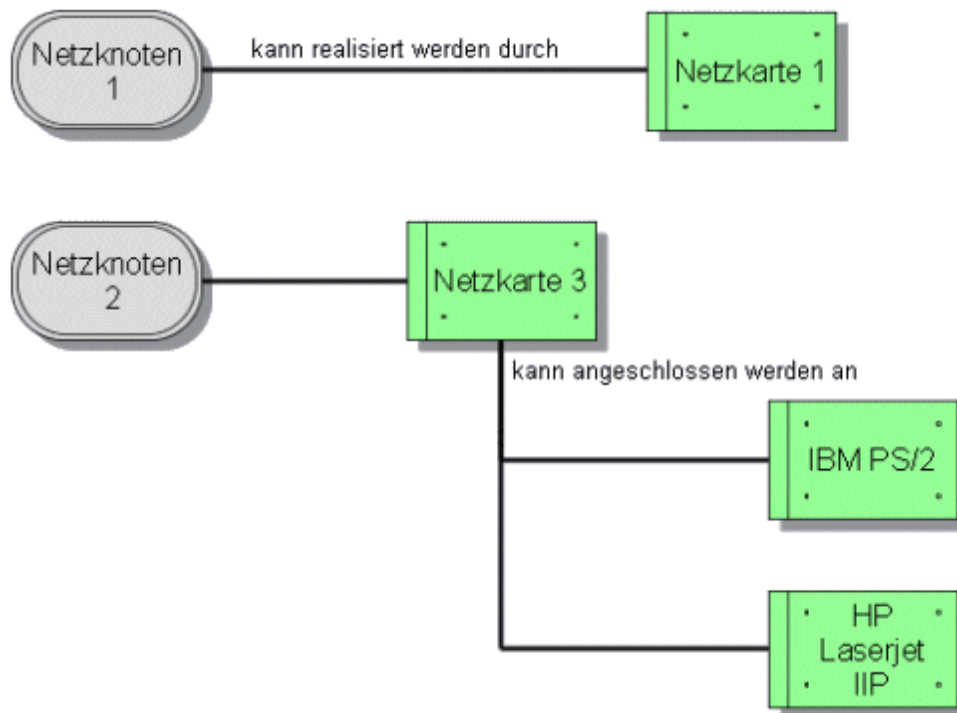


Abbildung 82: Netztopologie

Die Verbindung der Netztopologie zu den Objekten des Fachkonzeptes wird über zwei Konstrukte hergestellt.

Zum einen kann für jeden Hardwarekomponententyp festgelegt werden, welche Organisationseinheit oder Stelle für ihn verantwortlich sein kann.

Zum anderen kann für jeden Netztyp, jeden Netzknontyp und Netzkartentyp als auch für jeden Hardwarekomponententyp definiert werden, an welchem Standort des Unternehmens er sich befinden kann. Somit ist der Standort das zentrale Bindeglied zwischen dem Fachkonzept und dem DV-Konzept der Organisationssicht.

Eine Zusammenstellung der Objekt- und Beziehungstypen des Netztopologie-Modells befindet sich im Handbuch **ARIS-Methode – Tabellen** (Datei **method_tabulations_aris.pdf**) auf Ihrem Installationsmedium.

4.3.3 Implementierung

4.3.3.1 Netzdiagramm

Im Netzdiagramm erfolgt die Darstellung der konkreten Umsetzungen der im DV-Konzept spezifizierten Netztopologie.

Mit Hilfe des Objektes **Netz** werden die im Unternehmen konkret existierenden Netze aufgenommen. Für jedes Netz kann spezifiziert werden, aus welchen Netzknoten und Netzkanten es besteht.

Sowohl für jedes Netz als auch für jeden Netzknoten und jede Netzkante kann der genaue Standort im Unternehmen angegeben werden. Ein Standort kann hierbei ein ganzes Werk, ein bestimmtes Gebäude, ein Gebäudekomplex, ein Büro oder ein bestimmter Arbeitsplatz sein.

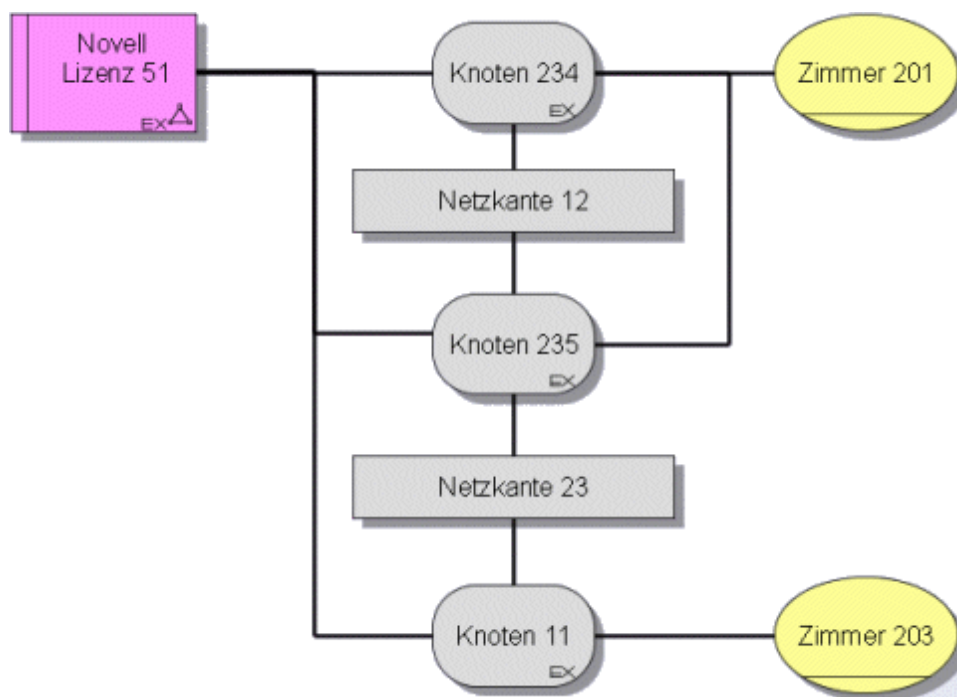


Abbildung 83: Netzdiagramm mit Standortzuordnung

Für jede Netzkante und jeden Netzknoten kann die Hardwarekomponente mit denen sie realisiert sind, aufgenommen werden. Daneben ist auch der strukturelle Aufbau jeder Hardware-Komponente darstellbar. Hardwarekomponenten dienen zum einen der Realisierung von Netzkanten und Netzknoten, können zum anderen aber auch an Netzknoten angeschlossen sein. Auch diese Beziehung ist im Netzdiagramm darstellbar. Für jedes Objekt auf Exemplar-Ebene ist auch die Beziehung zum entsprechenden Objekt der DV-Konzept-Ebene modellierbar. So kann gezeigt werden, dass es sich bei dem Netz im Werk **Zepelinstr.** um ein Netz des Typs **Novell Version 4.11** handelt.

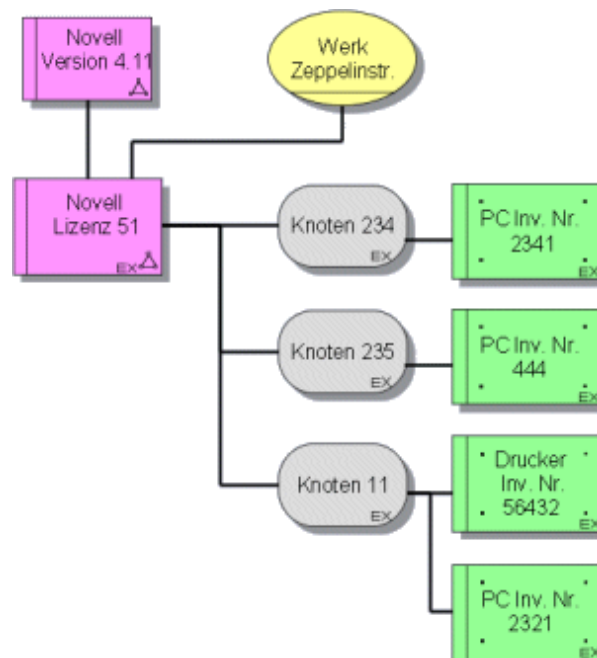


Abbildung 84: Netzdiagramm mit Hardwarekomponenten und Standortzuordnung

Im Netzdiagramm sind somit über die Typzuordnungen die Verbindungen zum DV-Konzept und über die Zuordnung der Netzkomponenten zu konkreten Standorten die Verbindungen zum Fachkonzept hergestellt.

Eine Zusammenstellung der Objekt- und Beziehungstypen des Netzdiagramms befindet sich im Handbuch **ARIS-Methode – Tabellen** (Datei **method_tabulations_aris.pdf**) auf Ihrem Installationsmedium.

4.3.3.2 Modellierung des Materialflusses - Technische Ressourcen

Zur Darstellung des Materialflusses in Prozessmodellen (EPK (Materialfluss), VKD (Materialfluss)) werden den einzelnen Funktionen des Geschäftsprozesses die Materialtypen als Funktionsinput oder Funktionsoutput zugeordnet. Ähnlich der Zuordnung von Informationsobjekten zu Funktionen (Darstellung der Transformation von Informationen durch Funktionen) wird mit dieser Zuordnung die Transformation der Inputmaterialtypen zu Outputmaterialtypen dargestellt. Zusätzlich kann in den Prozessketten aufgenommen werden, welche technischen Ressourcen zur Materialtransformation benötigt werden. Hierbei werden Betriebsmittel, Lagereinrichtungen, Transportsysteme und technische Hilfsmittel unterschieden.

Im Modelltyp **Technische Ressourcen** können diese Ressourcen hierarchisiert, typisiert und klassifiziert werden. Hierzu stehen folgende Objekttypen zur Verfügung:

Betriebsmittel

Betriebsmittel stellen die Exemplare verschiedener Betriebsmitteltypen dar, die einer Unternehmung zur Wahrnehmung ihrer Aufgaben zur Verfügung stehen. Betriebsmittel werden oft durch die Vergabe von Inventarisierungsnummern (Bsp.: Nummer einer Produktionsanlage) identifiziert.

Betriebsmitteltyp

Ein Betriebsmitteltyp stellt die Typisierung einzelner Betriebsmittel dar, die sich auf der exakt gleichen technologischen Basis befinden.

Betriebsmittelklasse

Gleichartige Betriebsmitteltypen können zu einer Betriebsmittelklasse zusammengefasst werden. Die Gleichartigkeit kann hierbei für verschiedene Klassifizierungsaspekte betrachtet werden. Ein Betriebsmitteltyp kann somit auch mehreren Betriebsmittelklassen zugeordnet werden.

Lagereinrichtung

Lagereinrichtungen stellen die Exemplare verschiedener Lagereinrichtungstypen dar, die einer Unternehmung zur Wahrnehmung ihrer Aufgaben zur Verfügung stehen. Lagereinrichtungen werden oft durch die Vergabe von Lagernummern identifiziert.

Lagereinrichtungstyp

Ein Lagereinrichtungstyp stellt die Typisierung einzelner Lagereinrichtungen dar, die sich auf der exakt gleichen technologischen Basis befinden.

Lagereinrichtungsklasse

Gleichartige Lagereinrichtungstypen können zu einer Lagereinrichtungsklasse zusammengefasst werden. Die Gleichartigkeit kann hierbei für verschiedene Klassifizierungsaspekte betrachtet werden. Ein Lagereinrichtungstyp kann somit auch mehreren Lagereinrichtungsklassen zugeordnet werden.

techn. Hilfsmittel

Ein techn. Hilfsmittel ist ein einzelnes Exemplar eines techn. Hilfsmitteltyps. Es kann i. d. R. über eine Inventarisierungsnummer identifiziert werden.

techn. Hilfsmitteltyp

Ein techn. Hilfsmitteltyp stellt die Typisierung einzelner techn. Hilfsmittel dar, die sich auf der exakt gleichen technologischen Basis befinden.

techn. Hilfsmittelklasse

Gleichartige techn. Hilfsmitteltypen können zu einer techn. Hilfsmittelklasse zusammengefasst werden. Die Gleichartigkeit kann hierbei für verschiedene Klassifizierungsaspekte betrachtet werden. Ein techn. Hilfsmitteltyp kann somit auch mehreren techn. Hilfsmittelklassen zugeordnet werden.

Transportsystem

Ein Transportsystem ist ein einzelnes Exemplar eines Transportsystemtyps. Es kann i. d. R. über eine Inventarisierungsnummer oder Anlagenummer identifiziert werden.

Transportsystemtyp

Ein Transportsystemtyp stellt die Typisierung einzelner Transportsysteme dar, die sich auf der exakt gleichen technologischen Basis befinden.

Transportsystemklasse

Gleichartige Transportsystemtypen können zu einer Transportsystemklasse zusammengefasst werden. Die Gleichartigkeit kann hierbei für verschiedene Klassifizierungsaspekte betrachtet werden. Ein Transportsystemtyp kann somit auch mehreren Transportsystemklassen zugeordnet werden.

Die Hierarchisierungsmöglichkeiten des Modelltyps **Techn. Ressourcen** ermöglichen es, den Aufbau komplexer techn. Anlagen zu beschreiben. So können z. B. die Komponenten einer komplexen Produktionsanlage und ihre Beziehungen untereinander dargestellt werden.

Neben den oben beschriebenen Modellierungsmöglichkeiten können sowohl Standortzuordnungen als auch organisatorische Verantwortlichkeiten für techn. Ressourcen definiert werden. Hierzu stehen die aus dem Modelltyp **Organigramm** bekannten Objekttypen **Standort**, **Organisationseinheit**, **Gruppe**, **Stelle** und **Person** zur Verfügung, die mit den Objekttypen **Betriebsmittel**, **Lagereinrichtung**, **techn. Hilfsmittel** und **Transportsystem** verbunden werden können.

Ein Beispiel für den Modelltyp **Techn. Ressourcen** zeigt diese Abbildung.

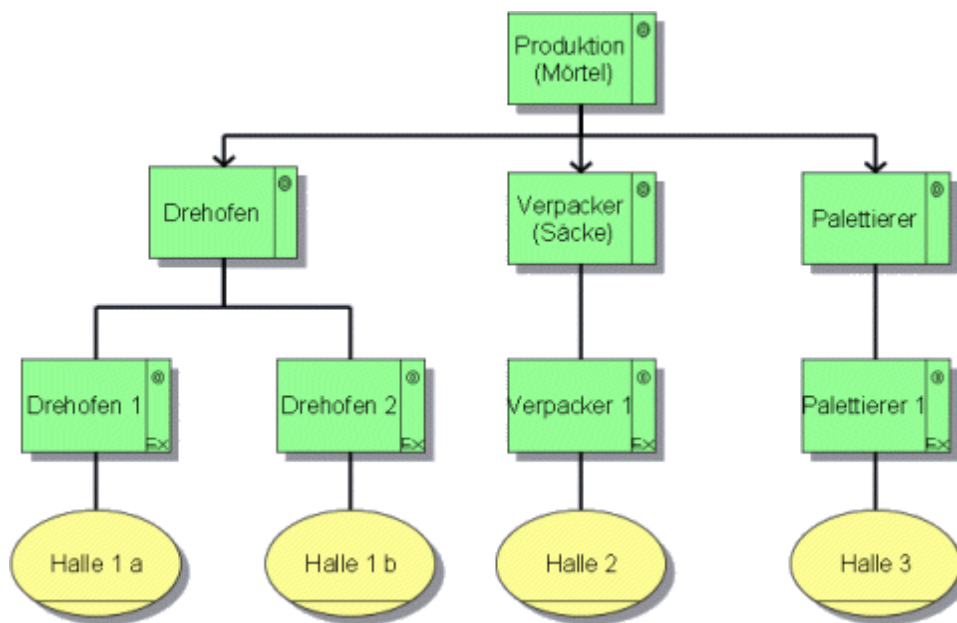


Abbildung 85: Beispiel für ein Modell "Technische Ressourcen"

4.4 Prozesssicht/Steuerungssicht

4.4.1 Fachkonzept

In der Steuerungs-/Prozesssicht werden die Beziehungen zwischen den Objekten der Daten-, Organisations- und Funktionssicht betrachtet. Die Beziehungen, die zu betrachten sind, ergeben sich aus den Kanten zwischen den Sichten.

Zunächst werden die Beziehungen jeweils zweier Sichten zueinander betrachtet, danach werden Diagramme für die Darstellung der Beziehungen zwischen allen drei Sichten vorgestellt.

4.4.1.1 Funktionen mit Organisation - EPK, Funktions-/Org.-Ebenendiagramm

Die Verbindung der Funktionssicht mit der Organisationssicht dient dazu, die im Funktionsbaum definierten Funktionen den Aufgabenträgern (Organisationseinheiten) des Organigramms zuzuordnen. Diese Zuordnung definiert die Verantwortung und die Entscheidungskompetenz einer Organisationseinheit für die zugeordneten Funktionen. Betrachtet man diese organisatorischen Zuordnungen entlang einer Prozesskette (Geschäftsprozesse), wird der Grad der Funktionsintegration definiert, d. h. die von einer Organisationseinheit zu bearbeitenden Funktionsschritte innerhalb eines Geschäftsprozesses.

Ein Beispiel für die Zuordnung von Organisationseinheiten zu Funktionen zeigt folgende Abbildung. Hierbei ist der links platzierten Funktion die für ihre Ausführung verantwortliche Organisationseinheit zugeordnet. Die Über-/Unterordnungen der Funktionen wurden in der Funktions-sicht (Funktionsbaum) und die Beziehungen zwischen den Organisationseinheiten in der Organisationssicht (Organigramm) dargestellt. Sie müssen somit an dieser Stelle nicht mehr definiert werden.

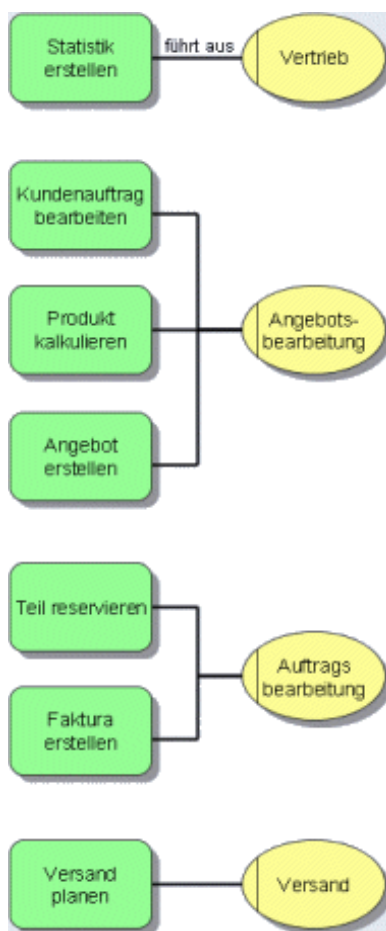


Abbildung 86: Zuordnung der Organisationseinheiten zu Funktionen

4.4.1.2 Funktionen mit Daten

4.4.1.2.1 Ereignissteuerung - ereignisgesteuerte Prozesskette (EPK)

Die Ablauffolge von Funktionen im Sinne eines Geschäftsprozesses wird in Prozessketten dargestellt. Dabei können für jede Funktion die Start- und Endereignisse angegeben werden. Ereignisse sind Auslöser von Funktionen und Ergebnisse von Funktionen.

Ein Ereignis ist das Eintreten eines betriebswirtschaftlich relevanten Zustandes eines Informationsobjektes, der den weiteren Ablauf des Geschäftsprozesses steuert oder beeinflusst. Ereignisse lösen Funktionen aus und sind Ergebnisse von Funktionen. Im Gegensatz zu einer Funktion, die ein zeitverbrauchendes Geschehen dargestellt, ist ein Ereignis auf einen Zeitpunkt bezogen.

Die Zustandsänderung eines Informationsobjektes kann sich zum einen auf das erste Auftreten dieses Informationsobjektes beziehen (Bsp.: **Kundenanfrage ist eingetroffen**) oder auf eine Zustandsänderung im Sinne einer Statusänderung, die in einer Attributausprägung festgehalten wird (Bsp.: **Angebot ist abgelehnt**). Da Informationsobjekte und Attribute in der Datensicht von ARIS beschrieben werden, ist die ereignisgesteuerte Darstellung von Prozessketten eine Verbindung zwischen der Daten- und Funktionssicht und ist somit der Steuerungssicht von ARIS zugeordnet.

Grafisch werden Ereignisse durch Sechsecke dargestellt. Die Bezeichnung sollte sowohl das Informationsobjekt (**Auftrag**) als auch die Zustandsänderung dieses Informationsobjektes (**ist eingetroffen**) beinhalten. In der nächsten Abbildung sind Ereignisse abgebildet.



Abbildung 87: Ereignisse (grafische Darstellung)

Ereignisse lösen Funktionen aus und sind Ergebnisse von Funktionen. Durch das Hintereinanderschalten dieses Ereignis-Funktionswechsels entstehen so genannte ereignisgesteuerte Prozessketten. Eine ereignisgesteuerte Prozesskette (EPK) zeigt den logisch-zeitlichen Ablauf eines Geschäftsprozesses.

Ein Beispiel für eine EPK zeigt die folgende Abbildung. Da Ereignisse definieren, durch welchen Zustand oder Bedingung eine Funktion gestartet wird und welcher Zustand den Abschluss einer

Funktion definiert, sind die Start- und Endknoten einer solchen EPK immer Ereignisse. Von einem Ereignis können mehrere Funktionen gleichzeitig ausgehen, andererseits kann eine Funktion mehrere Ereignisse als Ergebnis haben. Um diese Verzweigungen und Bearbeitungsschleifen in einer EPK darstellen zu können, wird eine Regel in Form eines Kreises verwendet. Die Verbindungen dienen jedoch nicht nur als grafische Verknüpfungen, sondern sie definieren die logischen Verknüpfungen der Objekte, die sie verbinden.

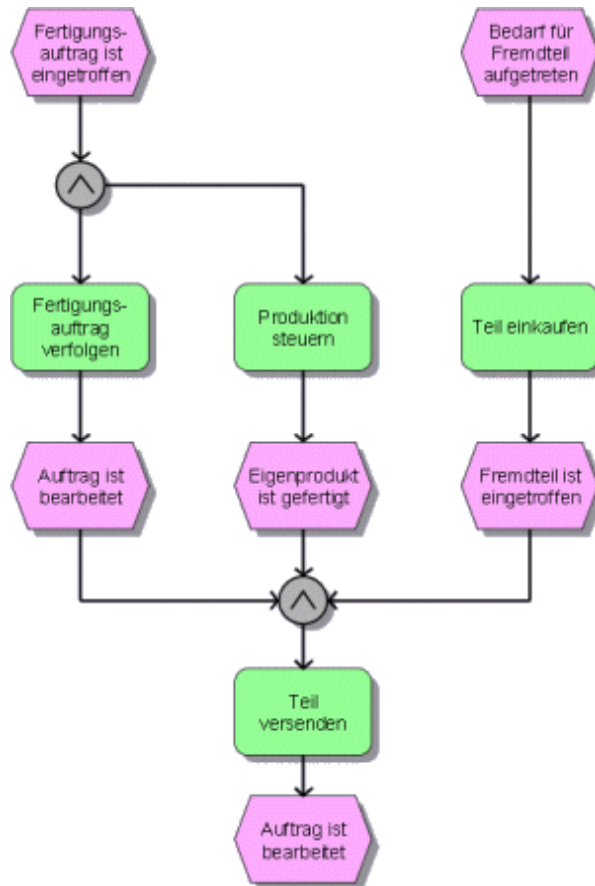


Abbildung 88: Beispiel für eine EPK

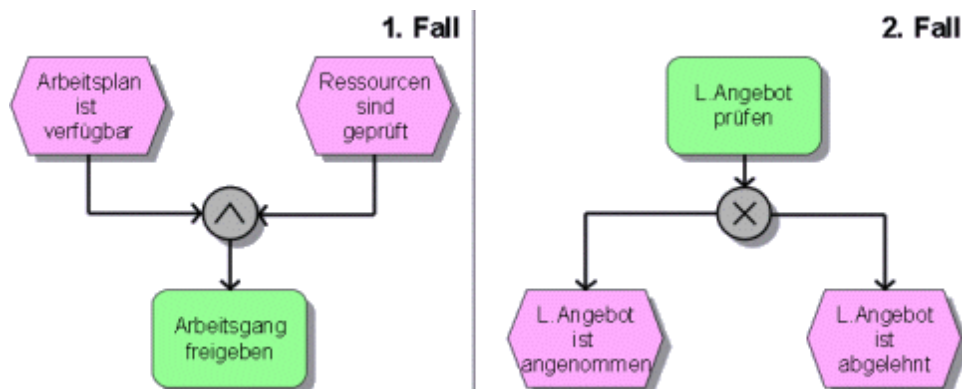


Abbildung 89: Beispiele für Regeln

Im ersten Fall dieser Abbildung liegt eine UND-Verknüpfung durch eine UND-Regel der Startereignisse vor. Dies bedeutet, dass der Vorgang **Arbeitsgang freigeben** nur dann gestartet wird,

wenn sowohl der Arbeitsplan vorhanden ist als auch die benötigten Ressourcen geprüft sind. Beide Ereignisse müssen also eingetreten sein, damit der Vorgang starten kann. Der zweite Fall stellt eine Exklusiv-oder-Verknüpfung (exklusives oder) mit Hilfe einer XOR-Regel dar. Ergebnis der Funktion **Lieferantenangebot prüfen** kann sein, dass das Angebot entweder angenommen oder abgelehnt wird. Beides kann jedoch nicht gleichzeitig eintreten. Neben diesen beiden Fällen und der Verknüpfung im Sinne eines „einschließenden oder“ sind auch komplexere Beziehungen denkbar. In diesem Fall kann in der EPK eine allgemeine Regel dargestellt werden, die dann detaillierter in Form eines Regeldiagramms beschrieben wird.

Grundsätzlich können somit zwei Arten von Verknüpfungen unterschieden werden:

1. Ereignisverknüpfungen und
2. Funktionenverknüpfungen.

Eine Übersicht der möglichen Ereignis- und Funktionsverknüpfungen zeigt folgende Abbildung (vgl. Hoffmann, Kirsch, Scheer, Modellierung mit Ereignisgesteuerten Prozessketten, 1993, S. 13).

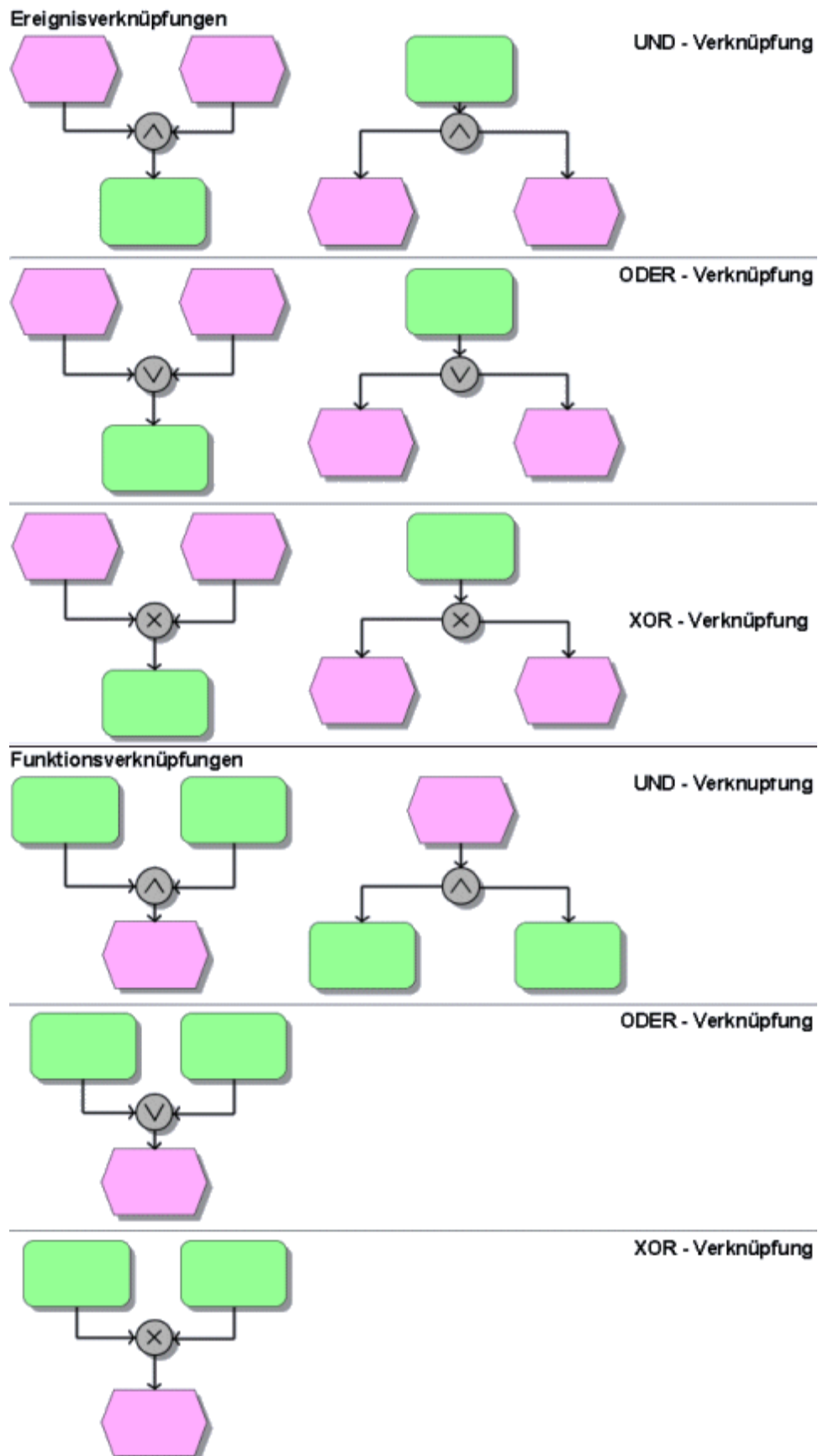


Abbildung 90: Verknüpfungsoperatoren (Regeln)

Zu beachten sind hierbei vor allem die Einschränkungen bezüglich der Funktionenverknüpfungen. Da Ereignisse keine Entscheidungen treffen können (dies können nur Funktionen), ist die Verknüpfung eines auslösenden Ereignisses mit ODER- und Exklusiv-ODER-Regel nicht erlaubt! Nachfolgend werden mögliche Verknüpfungen durch Beispiele erläutert.

Die Verknüpfung von auslösenden Ereignissen

UND-Regel

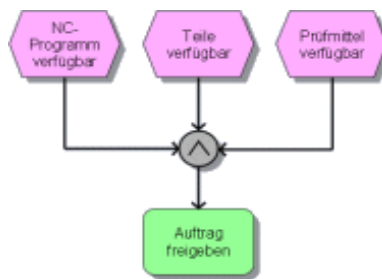


Abbildung 91: Und-Verknüpfung auslösender Ereignisse

Die Funktion kann erst gestartet werden, wenn alle Ereignisse eingetreten sind.

ODER-Regel

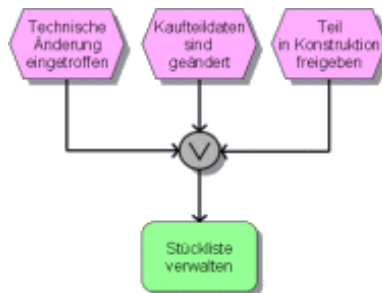


Abbildung 92: Oder-Verknüpfung auslösender Ereignisse

Die Funktion wird ausgeführt, wenn mindestens eines der Ereignisse eingetreten ist.

Exklusiv-ODER-Regel (Entweder-ODER-Regel)



Abbildung 93: Exklusiv-oder-Verknüpfung auslösender Ereignisse

Die Funktion wird gestartet, wenn genau ein und nur ein Ereignis eingetreten ist.

Die Verknüpfung von erzeugten Ereignissen

UND-Regel



Abbildung 94: Und-Verknüpfung erzeugter Ereignisse

Die Funktion führt dazu, dass alle Ereignisse eintreten.

ODER-Regel

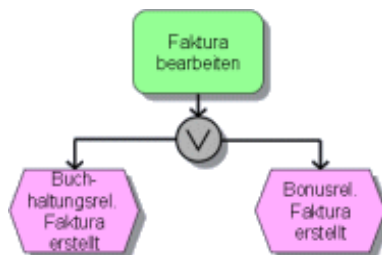


Abbildung 95: Oder-Verknüpfung erzeugter Ereignisse

Die Ausführung der Funktion führt dazu, dass mindestens eines der Ereignisse eintritt.

Exklusiv-ODER-Regel (Entweder-ODER-Regel)

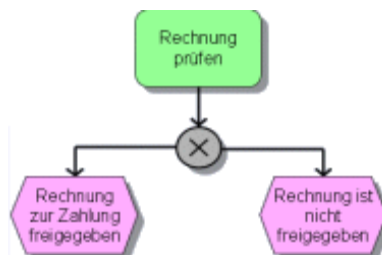


Abbildung 96: Exklusiv-oder-Verknüpfung erzeugter Ereignisse

Die Ausführung der Funktion führt dazu, dass höchstens eines der Ereignisse eintritt.

Die Verknüpfung von Funktionen mit erzeugten Ereignissen

UND-Regel

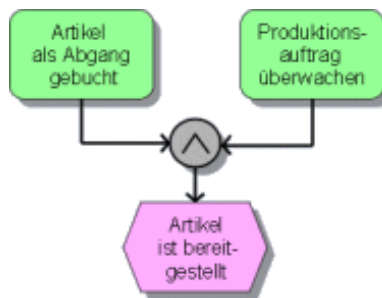


Abbildung 97: Und-Verknüpfung von Funktionen mit erzeugten Ereignissen

Das Ereignis ist erst eingetreten, wenn alle Funktionen ausgeführt sind.

ODER-Regel

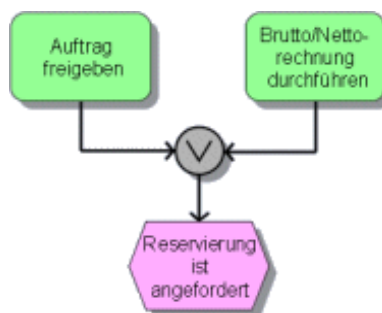


Abbildung 98: Oder-Verknüpfung von Funktionen mit erzeugten Ereignissen

Das Ereignis tritt ein, wenn mindestens eine der Funktionen durchgeführt ist.

Exklusiv-ODER (Entweder-ODER-Regel)

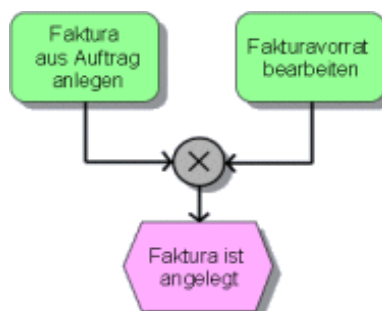


Abbildung 99: Exklusiv-oder-Verknüpfung von Funktionen mit erzeugten Ereignissen

Das Ereignis tritt ein, wenn genau eine und höchstens eine der Funktionen ausgeführt wurde.

Die Verknüpfung von Funktionen mit auslösenden Ereignissen

UND-Regel

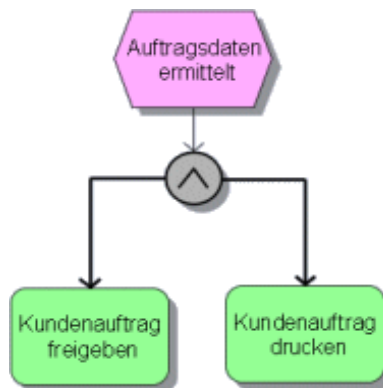


Abbildung 100: Und-Verknüpfung von Funktionen mit auslösenden Ereignissen

Alle Funktionen werden durch das Ereignis ausgelöst.

Oder-Regel

Ereignisse haben keine Entscheidungskompetenz! Diese Verknüpfung ist nicht möglich!

Exklusiv-ODER-Regel (Entweder-ODER-Regel)

Ereignisse haben keine Entscheidungskompetenz! Diese Verknüpfung ist nicht möglich!

Neben der Darstellung in Form von ereignisgesteuerten Prozessketten können diese Verzweigungen auch in tabellarischer Form in den Ereignis- und Funktionsspalten eines Vorgangskettendiagramms (vgl. Kapitel **Die Vorgangskettenanalyse** (Seite 11)) gezeigt werden. Die sequentielle Anordnung der Funktionen in einem Vorgangskettendiagramm führt jedoch dazu, dass Verzweigungen und Bearbeitungsschleifen nur sehr unübersichtlich darstellbar sind.

EPK: BPML-Export

ARIS unterstützte bisher das Dateiformat BPML. Da BPML nicht mehr weiterentwickelt wird und nur wenige Systeme noch BPML verstehen, wird BPML in den Desupport von ARIS aufgenommen und damit in naher Zukunft nicht mehr unterstützt.

4.4.1.2.2 Funktionszuordnungsdiagramm (I/O)

Neben der in Kapitel **Ereignissteuerung- ereignisgesteuerte Prozesskette (EPK)** (Seite 93) erläuterten Darstellung der Ereignissteuerung stellen auch die Transformation von Eingabedaten zu Ausgabedaten und die Darstellung der Datenflüsse zwischen Funktionen eine Verbindung zwischen der Datensicht und Funktionssicht des ARIS-Konzeptes dar. Die Transformation von Input-Daten zu Outputdaten kann in sog. Funktionszuordnungsdiagrammen (I/O), die im Wesentlichen der Darstellung reiner Input-/Outputdiagramme anderer Methoden entsprechen, dargestellt werden. Ein Beispiel für ein Funktionszuordnungsdiagramm (I/O) zeigt die folgende Abbildung. Die Funktion **Liefertermin bestimmen** hat als Input-Daten **Teiledaten**, **Bestandsdaten**, **Stücklistendaten** und **Versanddaten**. **Anfragedaten** sind sowohl Input-Daten als auch Outputdaten. Bestandteile eines Funktionszuordnungsdiagramms (I/O) sind somit Funktionen der Funktionssicht und Informationsobjekte der Datensicht. Die Pfeile definieren, ob ein Informationsobjekt nur Inputdatum, nur Outputdatum oder Input-/Outputdatum ist. Darüber hinaus sind genauere Spezifizierungen möglich, die z. B. aussagen, dass ein Informationsobjekt durch die Funktion erstellt oder gelöscht wird. Je nach dem Detaillierungsgrad kann es sich bei den Informationsobjekten um Datencluster, Entity-/Beziehungstypen oder auch Attribute der Datensicht handeln.

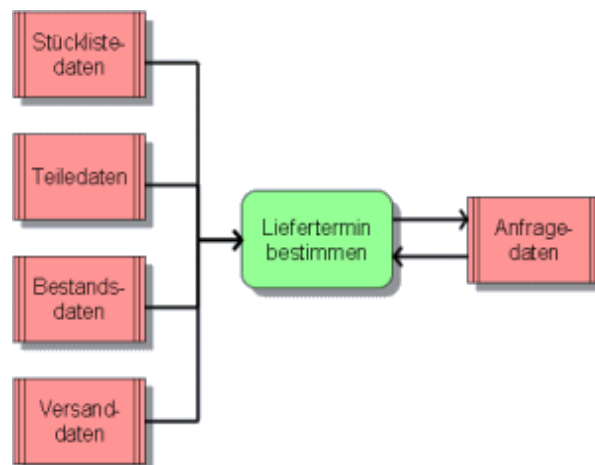


Abbildung 101: Beispiel für ein Funktionszuordnungsdiagramm (I/O)

Das hier gezeigte Beispiel verdeutlicht die eigentliche Zielsetzung der Funktionszuordnungsdiagramme (I/O), nämlich die Darstellung der Input-/Output-Daten einer Funktion.

Neben der Aufnahme der Input-/Output-Daten einer Funktion stehen außer den Ereignissen auch alle anderen Objekte zur Verfügung, die man in einer EPK den Funktionen zuordnen kann. Damit ist dem Anwender die Möglichkeit gegeben, bei der Modellierung von Prozessketten in EPK-Diagrammen nur die Ereignisse und Funktionen aufzunehmen und hinter jede Funktion ein Funktionszuordnungsdiagramm (I/O) mit den weiteren Beziehungen der Funktion zu legen. Dies ermöglicht eine wesentlich übersichtlichere Darstellung der Geschäftsprozesse und erklärt auch die Verwendung einer neuen Bezeichnung für diesen Modelltyp. Ein Beispiel für eine solche erweiterte Darstellung in einem Funktionszuordnungsdiagramm zeigt diese Abbildung.

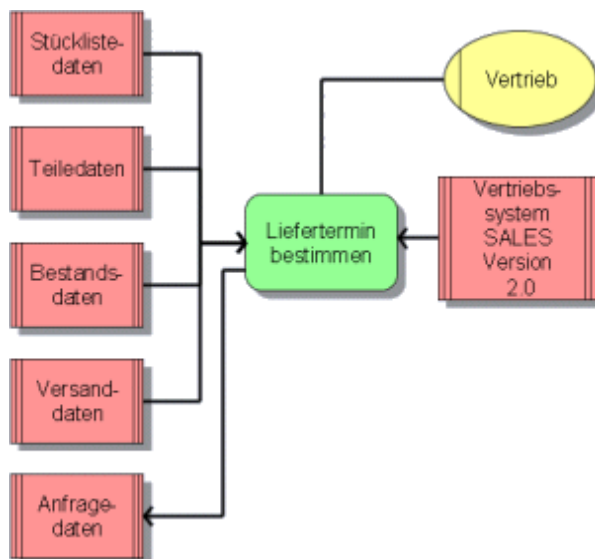


Abbildung 102: Erweiterte Darstellung des Funktionszuordnungsdiagramms

Neben der getrennten Darstellung der Datentransformation in Form von Funktionszuordnungsdiagrammen (I/O) können diese Informationen auch in einer EPK aufgenommen werden. Ein Beispiel hierfür ist in der nächsten Abbildung dargestellt. Die Verbindungen zwischen den Funktionen und Informationsobjekten haben hierbei die gleiche Bedeutung wie die des Funktionszuordnungsdiagramms (I/O), ihre Aufnahme in eine stark verzweigten Prozesskette führt jedoch unter Umständen zu unübersichtlichen Darstellungen.

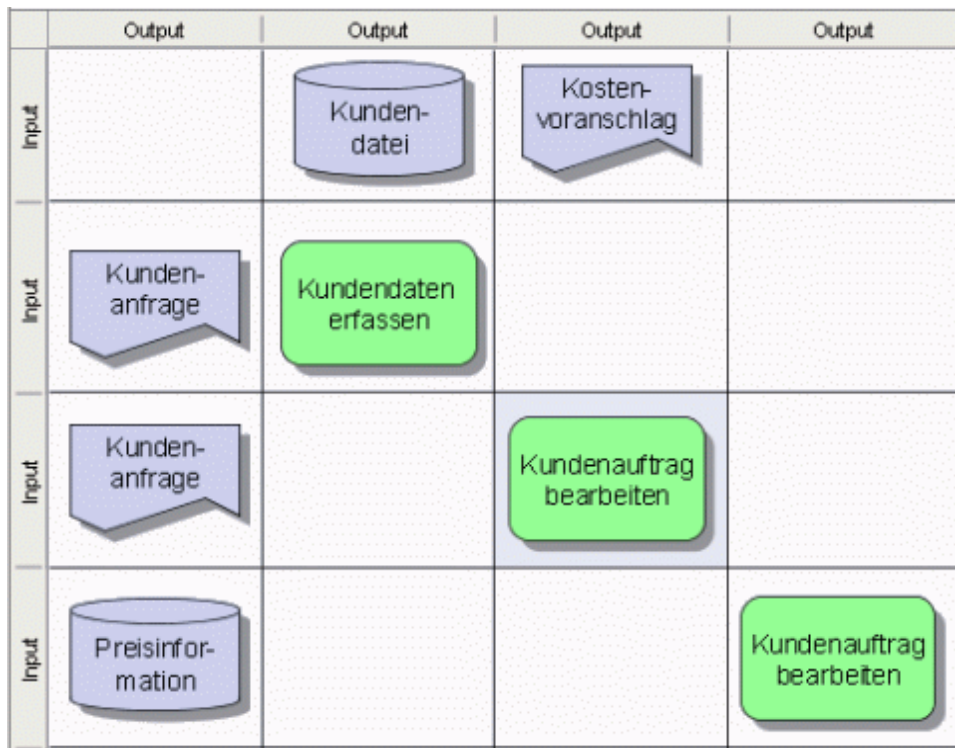


Abbildung 103: EPK mit Input-/Output-Daten

Im VKD (Vorgangskettendiagramm) müssen die Objekte entsprechend der Spaltenbezeichnung angeordnet werden. Die EPK-Darstellung erlaubt die freie Anordnung der Objekte. Allerdings kann die zusätzliche Aufnahme der Input-/Output-Daten zu unübersichtlichen Modellen führen. Daher ist gerade bei sequentiell ablaufenden Geschäftsprozessen die VKD-Darstellung zu empfehlen. In der folgenden Abbildung ist die EPK mit Input-/Output-Daten der vorherigen Abbildung als VKD realisiert (siehe auch Kapitel **EPK/VKD** (Seite 107)).

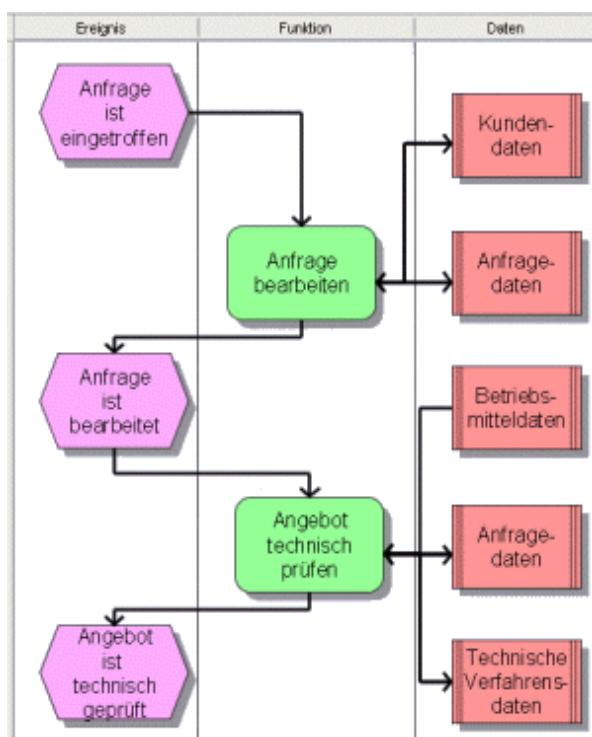


Abbildung 104: VKD mit Input-/Output-Daten

4.4.1.2.3 Informationsflussdiagramme

Informationsflussdiagramme eignen sich zur Darstellung der bereits angesprochenen Datenflüsse zwischen Funktionen. Zu diesem Zweck können in einem Informationsflussdiagramm zwei Funktionen über ein Datenflussobjekt miteinander verbunden werden. Dieses Objekt drückt aus, dass ein Datenfluss von der Quellfunktion zur Zielfunktion existiert. Zur genaueren Spezifizierung der Datenobjekte, die zwischen den dargestellten Funktionen fließen, kann durch Hierarchisierung dieses Datenflussobjekts ein Datenmodell hinter dieses Objekt gelegt werden, in dem die Informationsobjekte, die zwischen den Funktionen ausgetauscht werden, dargestellt werden. In Abhängigkeit vom Detaillierungsgrad der betrachteten Funktionen kann es sich bei den Informationsobjekten um Datencluster, Entitytypen oder ERM-Attribute handeln. Ein Beispiel für diese Darstellung zeigt diese Abbildung.

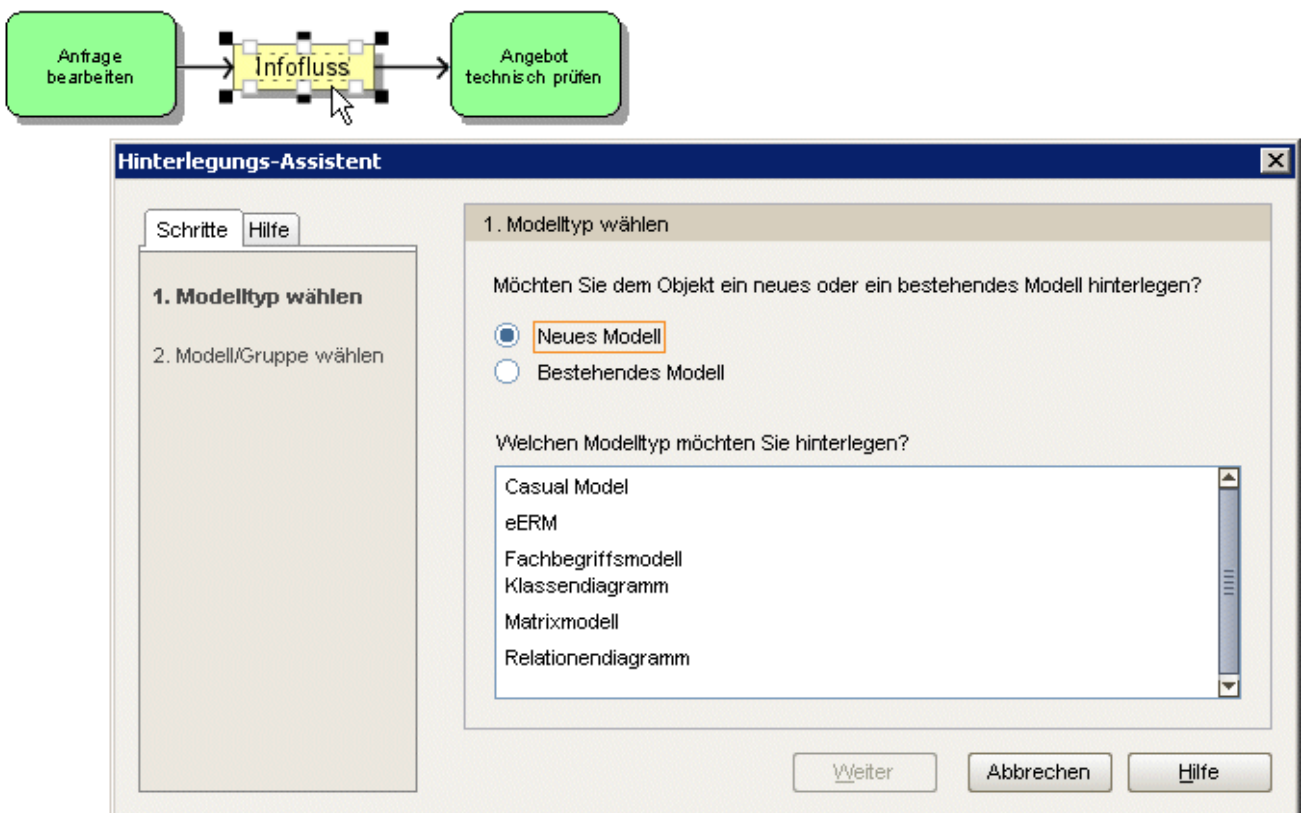


Abbildung 105: Informationsflussdiagramm mit geöffnetem Hinterlegungs-Assistenten

4.4.1.2.4 Ereignisdiagramm

Ereignisse definieren das Eintreten von Zustandsänderungen von Informationsobjekten. Jedes Ereignis referenziert somit auf bestimmte Informationsobjekte des Datenmodells und definiert den Status des jeweiligen Informationsobjektes zu einem bestimmten Zeitpunkt.

Ereignisse werden in einer Top-down-Vorgehensweise zunächst grob spezifiziert (Bsp. **Auftrag ist bearbeitet**). Auf der nächsten Detaillierungsstufe der Prozessmodellierung werden nun die detaillierteren Ereignisse definiert, die in bestimmter Kombination dazu führen, dass das Ereignis auf grober Ebene eingetreten ist. So kann z. B. das Eintreten der Ereignisse **Machbarkeit ist geprüft**, **Auftragskopf ist erfasst** und **Auftragspositionen sind erfasst** in der Summe den Zustand **Auftrag ist bearbeitet** definieren.

Diese Ereigniszusammenhänge auf grober und feiner Modellierungsebene können mit Hilfe des Ereignisdiagramms dargestellt werden. Hierzu kann hinter ein Ereignis auf grober Ebene ein Ereignisdiagramm gelegt werden (Hierarchisierung!!), in dem die Ereignisse auf feiner Ebene und ihre Verknüpfungen (auch über Regel-Verknüpfungen) untereinander dargestellt werden. Zusätzlich können in diesem Modelltyp auch Informationsobjekte des Datenmodells aufgenommen werden und mit den Ereignissen verknüpft werden. Damit wird definiert, welches Ereignis die Zustandsänderung welches Informationsobjektes definiert.

Folgende Abbildung zeigt ein Beispiel.

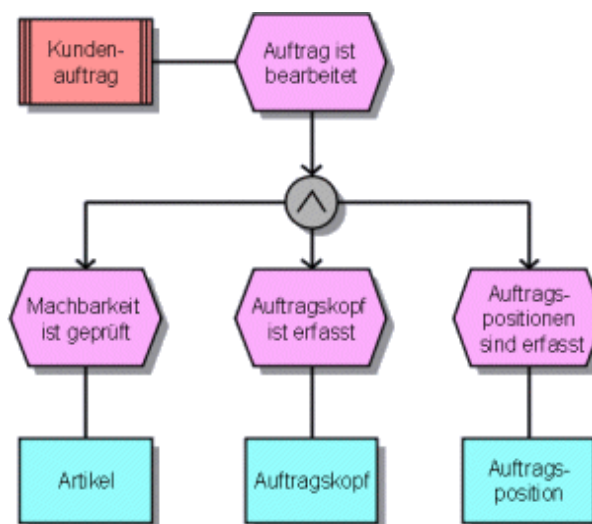


Abbildung 106: Beispiel für ein Ereignisdiagramm

4.4.1.3 Funktionen - Organisation - Daten

4.4.1.3.1 EPK/VKD

Sowohl in der EPK als auch im VKD werden dieselben Sachverhalte abgebildet.

Die bisherige Betrachtung zweier Sichten wird nun um die dritte Sicht erweitert. Damit werden die Teilsichten der Prozesskette wieder in einer Gesamtsicht zusammengefasst und das Zusammenwirken aller Komponenten des ARIS-Konzeptes betrachtet. Die Prozesskette, die auch Ausgangspunkt der Betrachtung war, wird also nun wieder in detaillierter Form abgebildet. Hierbei sind jedoch nicht die in den einzelnen Sichten erarbeiteten Details der betrachteten Objekte Schwerpunkt der Untersuchung, sondern die Verbindungen zwischen diesen Objekten. In der nächsten Abbildung ist eine Vorgangskette mit allen Sichten einer Prozesskette dargestellt. In der ersten Spalte werden die Ereignisse, die Objekte der Datensicht repräsentieren, dargestellt. Die Pfeile führen in die Vorgangsspalte, in der die Funktionen der Prozesskette aufgeführt sind. Die erste und zweite Spalte definieren somit die Ereignissteuerung. In der dritten Spalte werden die Datenobjekte aufgenommen und ihre Beziehungen zu den Funktionen dargestellt. Die Sicht auf die zweite und dritte Spalte der VKD definiert somit den Datenfluss der Prozesskette. Gegenüber der in Kapitel **Das Vorgangskettendiagramm** (Seite 12) vorgestellten VKD verfügt das Vorgangskettendiagramm des Fachkonzeptes nicht über Spalten zur Definition der Bearbeitungsart und des DV-Systems. Diese Tatbestände werden zur Aufnahme der Ist-Situation in einem Unternehmen benötigt, sind jedoch nicht Bestandteil der fachlichen Beschreibung eines Geschäftsprozesses. Die vierte Spalte definiert somit die Organisationseinheiten der Organisationssicht, die für die Ausführung der einzelnen Funktionen der Prozesskette verantwortlich sind.

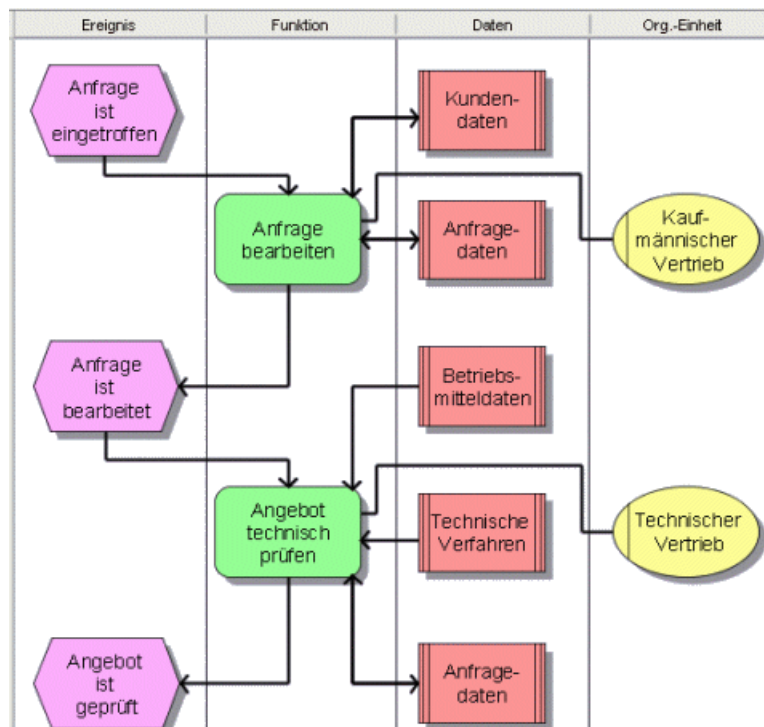


Abbildung 107: Beispiel für eine Vorgangskette (Fachkonzept)

Die in dieser Abbildung dargestellte Prozesskette ist auch in einer EPK darstellbar.

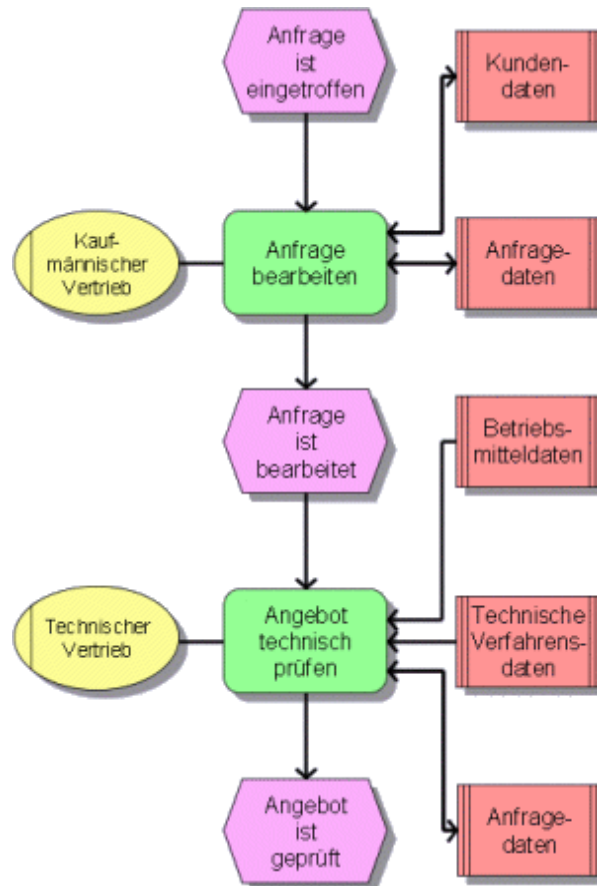


Abbildung 108: EPK mit Funktionen, Daten, Organisationseinheiten und Ereignissen

4.4.1.3.2 Wertschöpfungskettendiagramm

Das Wertschöpfungskettendiagramm dient in erster Linie dazu, die Funktionen im Unternehmen zu spezifizieren, die direkt an der Wertschöpfung des Unternehmens beteiligt sind. Diese Funktionen können in Form einer Funktionsfolge miteinander verbunden werden und bilden damit eine Wertschöpfungskette. Ein Beispiel für eine Wertschöpfungskette zeigt diese Abbildung.

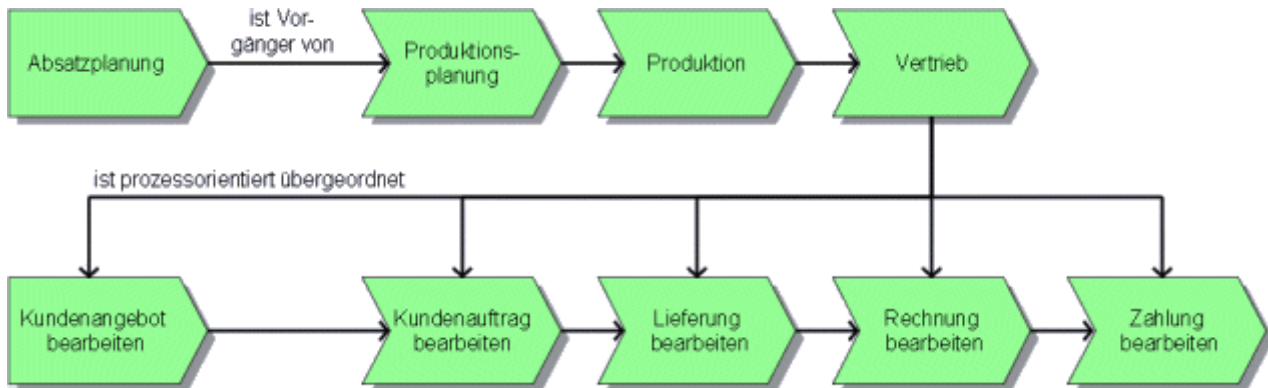


Abbildung 109: Wertschöpfungskette

Funktionen können im Wertschöpfungskettendiagramm ähnlich einem Funktionsbaum hierarchisiert werden. Hierbei wird immer die prozessorientierte Über-/Unterordnung dargestellt.

Neben der Über-/Unterordnung von Funktionen können in einem Wertschöpfungskettendiagramm auch die Verbindungen der Funktionen zu Organisationseinheiten und zu Informationsobjekten dargestellt werden. Bei der Zuordnung der Organisationseinheiten zu Funktionen werden wie auch schon in den Prozessketten die fachliche Verantwortung, die DV-Verantwortung und die eigentliche Ausführung einer Funktion unterschieden.

Weitere Beziehungen, die im Wertschöpfungskettendiagramm zur Verfügung stehen, finden sich im Handbuch **ARIS-Methode – Tabellen** (Datei **method_tabulations_aris.pdf**) auf Ihrem Installationsmedium.

4.4.1.3.3 Regeldiagramm

In Prozessketten können die Regeln als Verknüpfungen verwendet werden, um die Verknüpfungen von Ereignissen und Funktionen zu spezifizieren. Oft kommt es jedoch vor, dass diese Regeldarstellungen zur Abbildung der logischen Verknüpfungen sehr komplex werden - insbesondere dann, wenn Regeln untereinander verknüpft werden. Um die Prozesskette durch solche Darstellungen nicht zu komplex werden zu lassen, kann in diesem Fall in der EPK oder VKD die allgemeine Regel-Verknüpfung verwendet werden. Diese allgemeine Regel-Verknüpfung kann mit einem Regeldiagramm verknüpft werden (Hierarchisierung!), in dem dann die komplexe Regel im Detail dargestellt wird.

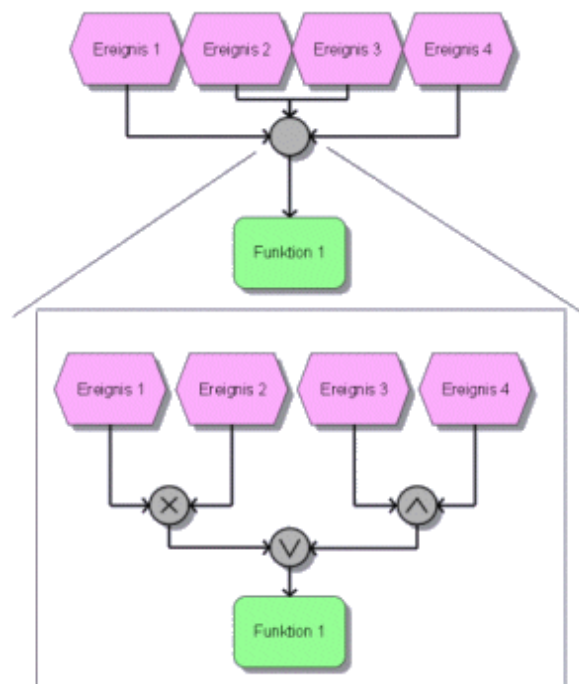


Abbildung 110: Abbildung komplexer Verknüpfungen im Regeldiagramm

4.4.1.3.4 Kommunikationsdiagramm

Große Referenzmodelle umfassen eine Vielzahl von Prozessmodellen. Innerhalb dieser Prozessmodelle wird durch die Einbeziehungen der Elemente der Organisationssicht dargestellt, wer innerhalb eines Prozessablaufs mit wem kommuniziert. Das Kommunikationsdiagramm bietet nun die Möglichkeit, alle Prozesse unter dem Aspekt der Kommunikation zwischen organisatorischen Einheiten zu gruppieren.

Im Kommunikationsdiagramm werden hierzu die organisatorischen Einheiten dargestellt, die miteinander kommunizieren. Es wird z. B. die Organisationseinheit **Vertrieb** über ein Objekt des Typs **Kommunikation** mit der Organisationseinheit **Kunde** verbunden. Objekte des Typs **Kommunikation** sind hierarchisierbar. Sie können mit dem Modelltyp **Prozessauswahlmatrix** verbunden werden. In dieser Prozessauswahlmatrix werden dann alle Prozesse dargestellt, in denen der Vertrieb mit dem Kunden kommuniziert.

4.4.1.3.5 Klassifizierungsdiagramm

Das Klassifizierungsdiagramm bietet die Möglichkeit, Funktionen zu klassifizieren, indem Funktionen Objekttypklassen zugeordnet werden. Die Klassifizierung erfolgt hierbei nach verschiedenen Klassifizierungsaspekten. Zur Spezifikation der Klassifizierungsaspekte kann der Objekttyp **Objekttypklasse** mit dem Objekttyp **Klassifizierungsaspekt** verbunden werden.

4.4.1.3.6 Input/Output Diagram

Das Input/Output Diagram dient der übersichtlichen Darstellung von ein- und ausgehenden Daten und Informationsträgern. Für dieses Modell gilt, dass in jedes Diagrammraster, d. h. durch Linien abgetrenntes Feld, nur ein Symbol gesetzt werden kann. In der obersten Zeile werden die von einer bestimmten Funktion erzeugten Daten oder Informationsträger (Output) platziert. Analog werden in der linken Spalte die in eine bestimmte Funktion eingehenden Daten- oder Informationsträgersymbole (Input) modelliert. Falls eine Funktion mehrere Input- oder Output-Symbole benötigt, werden diese durch Ausprägungskopien erzeugt.

Im Input/Output Diagram werden beim Anlegen von Funktionen und Daten- oder Informationsträgersymbolen automatisch die nicht sichtbaren (impliziten) Beziehungen **liefert Input für** oder **erzeugt Output auf** angelegt.

Nachfolgend sehen Sie ein einfaches Beispiel eines Input/Output Diagramm.

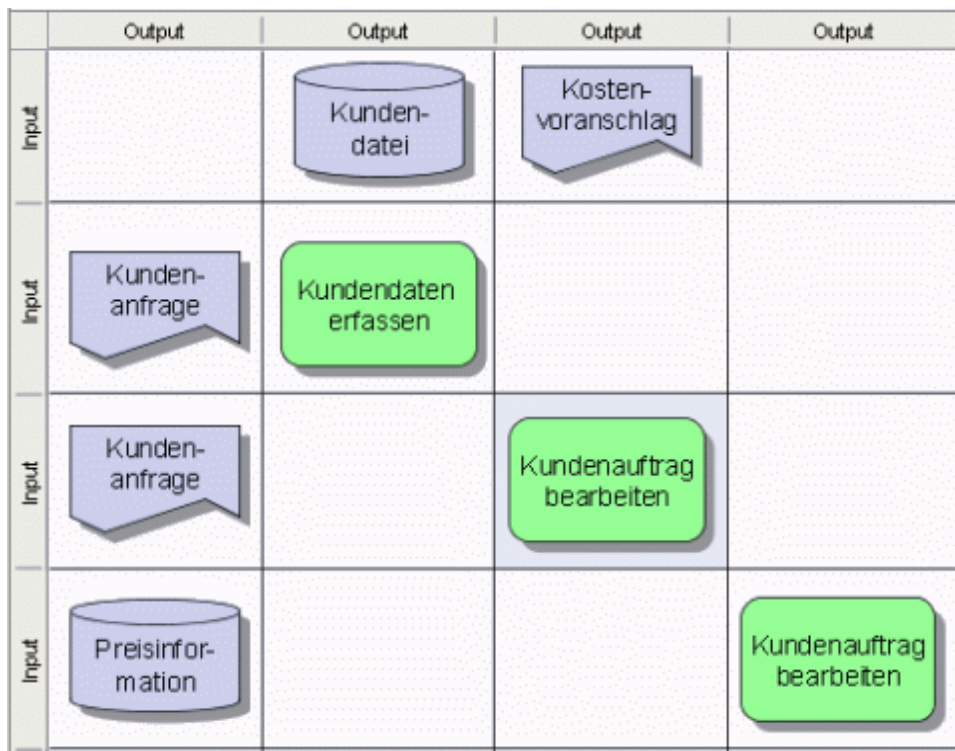


Abbildung 111: Input/Output Diagramm

4.4.1.4 Objektorientierte Modellierung

4.4.1.4.1 Klassendiagramm

Im objektorientiert erweiterten Konzept von ARIS besteht die Möglichkeit, Informationsobjekten (Clustern, Entitytypen, Beziehungstypen) die Rolle von Klassen zuzuordnen. Dies bedeutet, dass alle in ARIS innerhalb der Datensicht definierbaren Informationsobjekte Klassencharakter erhalten können, indem ihnen Klassenfunktionen (/methoden) und beschreibende Attribute (Dateninhalte) zugeordnet werden. Diese Zuordnung geschieht in so genannten **Klassendiagrammen**. Durch die Zuordnung eines Klassendiagramms zu einem Informationsobjekt wird diesem Informationsobjekt die Rolle einer Klasse zugeschrieben.

Ein Klassendiagramm ist eindeutig einem Informationsobjekt zugeordnet und enthält folgende Elemente (Symbole für Objekttypen der ARIS-Architektur):

- das Informationsobjekt, das als Klasse beschrieben wird,
- eine Liste von Attributen, die der Klasse zugeordnet sind;
- eine Liste von Ereignissen, die aufgrund des Eintretens bestimmter Status der Klasse auftreten;
- eine Liste von Funktionen der Funktionssicht, die der Klasse zugeordnet sind und deren auslösende bzw. von ihnen ausgelöste Ereignisse.

Diese Abbildung zeigt ein Beispiel für die Definition der Klasse **Kundenauftrag**.

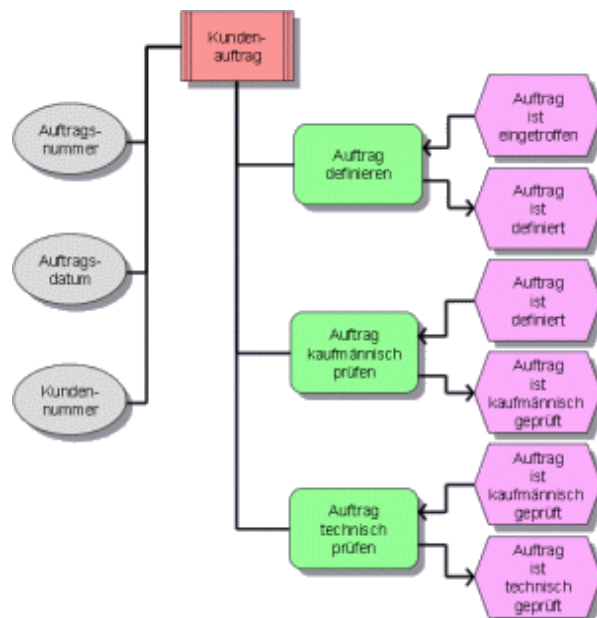


Abbildung 112: Klassendiagramm für die Klasse Kundenauftrag

4.4.1.5 Prozessvarianten

4.4.1.5.1 Prozessauswahlmatrix

In der Prozessauswahlmatrix werden verschiedene Prozessszenarien dargestellt, indem Hauptprozesse einzelnen Szenarien zugeordnet werden.

Der Anwender kann danach festlegen, welche Funktionen der Szenarioprozesse im Unternehmen auftreten. Hierzu sollten immer alle Hauptfunktionen (Szenariofunktionen) eines Anwendungssystems oder eines Branchenreferenzmodells als Prozesse enthalten sein.

Für die Modellierung einer Prozessauswahlmatrix stehen folgende Symboltypen zur Verfügung:

- Szenario
- Prozess
- Hauptprozess

Ein Szenario repräsentiert einen Szenarioprozess in der Auswahlmatrix der verschiedene Hauptprozesse gruppiert.

Der Prozess repräsentiert Funktionen des Szenarioprozesses, die im Referenzmodell durch Prozessmodelle näher beschrieben werden.

Der Hauptprozess repräsentiert die Hauptfunktionen in den Funktionsbäumen, denen die Prozesse (Funktionen aus den Szenarioprozessen) zugeordnet sind.

Ein Beispiel für eine Prozessauswahlmatrix zeigt die folgende Abbildung.

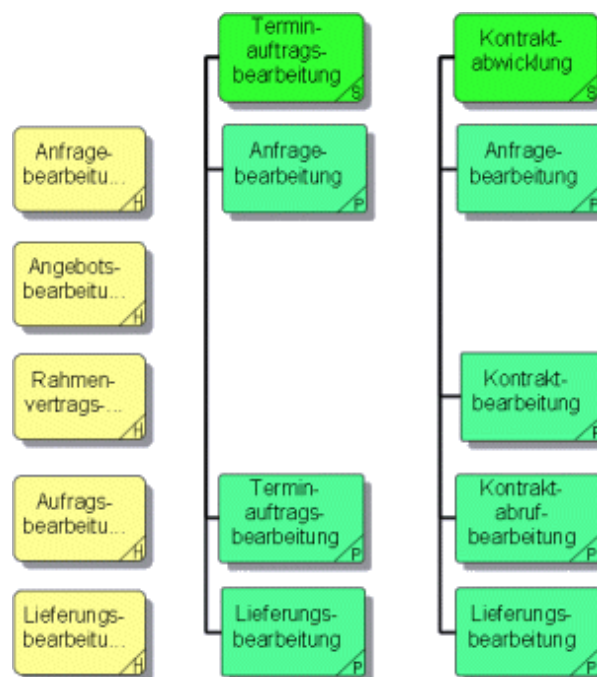


Abbildung 113: Prozessauswahlmatrix (Auszug aus dem Referenzmodell R/3 der SAP AG)

4.4.1.6 Modellierung des Materialflusses

Neben der Darstellung des Informationsflusses eignen sich Prozessmodelle (EPK und VKD) auch zur Darstellung der Materialtransformation. Zur Darstellung des Materialflusses innerhalb von Geschäftsprozessen wird in ARIS ein eigener Modelltyp angeboten - EPK (Materialfluss) -, der eine Erweiterung des Modelltyps **EPK** darstellt.

4.4.1.6.1 EPK (Materialfluss)

Neben den Objekttypen der EPK stehen zusätzlich in der EPK (Materialfluss) folgende Objekttypen zur Verfügung:

- Materialtyp
- Packmitteltyp
- Betriebsmitteltyp
- Betriebsmittel
- techn. Hilfsmitteltyp
- techn. Hilfsmittel
- Lagereinrichtungstyp
- Lagereinrichtung
- Transportsystemtyp
- Transportsystem

Der Objekttyp **Materialtyp** kann mit einer eingehenden oder einer ausgehenden Kante mit dem Objekttyp **Funktion** verknüpft werden. Im Falle einer eingehenden Kante wird definiert, welche Materialien von einer Funktion als Input benötigt werden. Hierbei kann durch Auswahl des entsprechenden Kantentyps definiert werden, ob das Material durch die Funktion nicht, teilweise oder ganz verbraucht wird. Eine ausgehende Kante spezifiziert die Materialtypen, die durch die Funktion erzeugt werden.

Zur Materialtransformation werden techn. Ressourcen benötigt. Diese können ebenfalls in den Prozessketten mit dem Objekttyp **Funktion** verbunden werden. Hierzu wird neben dem Kantentyp **benötigt** auch ein Kantentyp **benötigt alternativ** angeboten, um eventuell zur Verfügung stehende Alternativressourcen spezifizieren zu können.

Werden Materialien innerhalb einer Funktion verpackt, werden bestimmte Packmitteltypen benötigt. Zur Spezifikation der entsprechenden Packmitteltypen kann eine Beziehung zwischen der Funktion und den benötigten Packmitteltypen modelliert werden.

In folgender Abbildung ist eine EPK (Materialfluss) und den dazugehörigen techn. Ressourcentypen und Packmitteltypen dargestellt.

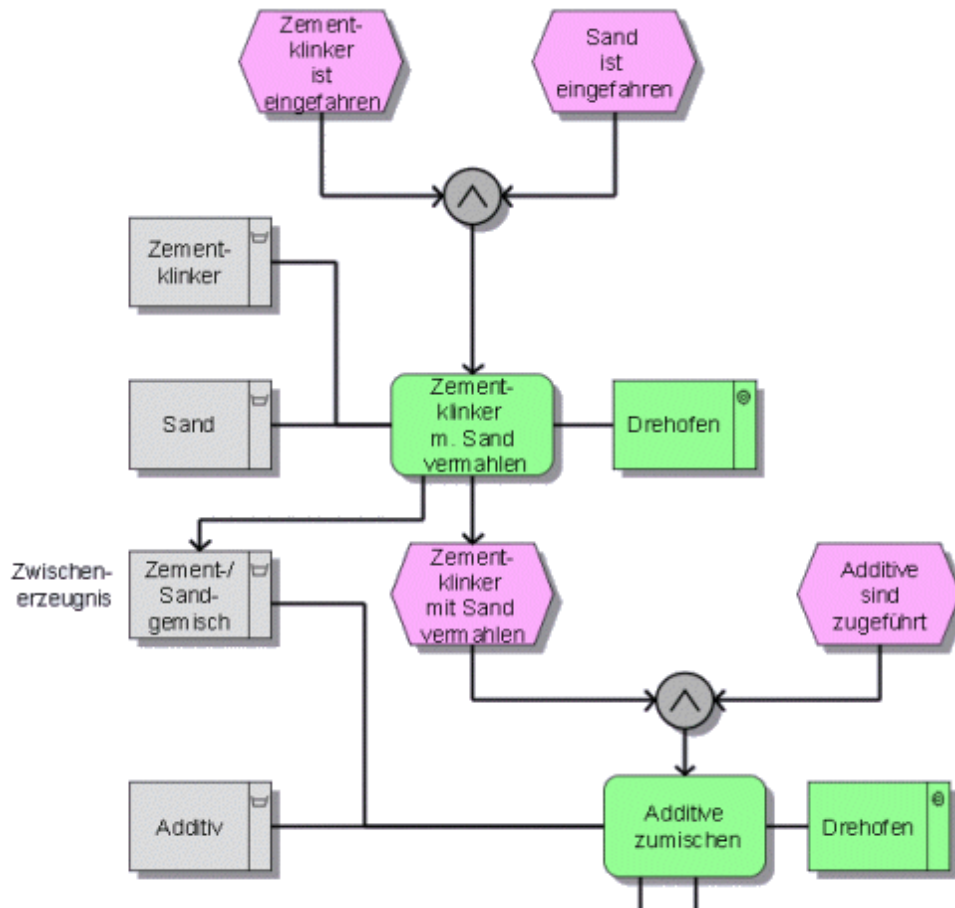


Abbildung 114: Auszug einer EPK (Materialfluss)

4.4.1.6.2 Materialflussdiagramm

Materialflussdiagramme eignen sich zur Darstellung von Materialflüssen zwischen Funktionen. Sie werden in der Modellierung ähnlich behandelt wie Informationsflussdiagramme. In einem Materialflussdiagramm werden zwei Funktionen über eine Materialflusskante miteinander verbunden. Die Kante drückt aus, dass ein Materialfluss von der Quellfunktion zur Zielfunktion existiert. Zur genaueren Spezifizierung der Materialien, die zwischen den dargestellten Funktionen fließen, kann durch Hierarchisierung dieser Materialflusskante ein Materialdiagramm hinter diese Kante gelegt werden, in dem die Materialien oder Materialtypen, die zwischen den Funktionen ausgetauscht werden, dargestellt werden.

4.4.1.6.3 EPK (Spaltendarstellung/Zeilendarstellung)

Die folgende Beschreibung gilt analog für die EPK (Zeilendarstellung).

Für den Modelltyp **EPK (Spaltendarstellung)** gelten im Wesentlichen die Ausführungen zur EPK mit dem Unterschied, dass in diesem Modell alle Symbole in verschiedenen Spalten verteilt werden. Das hat den Vorteil, dass die Übersichtlichkeit der EPK durch diese Darstellung deutlich verbessert wird. Organisatorische- und Anwendungssystem-Elemente werden in der Kopfzeile des Diagramms platziert. Alle anderen Symbole werden in der zweiten Zeile der Spalten positioniert.

Besonderes Merkmal sämtlicher Swimlane-Modelle, also von Modellen, die in Spalten und/oder Zeilen modelliert werden, ist die automatische Erstellung von nicht sichtbaren (impliziten) Beziehungen. Beispielsweise wird in den standardmäßig vorgegebenen Spalten der EPK (Spaltendarstellung) beim Modellieren von Anwendungssystemen und Funktionen automatisch die implizite Beziehung unterstützt angelegt. Zwischen organisatorischen Elementen und Funktionen werden führt aus-Beziehungen implizit erzeugt. Darüber hinaus hat der Anwender die Möglichkeit, folgende weitere Spalten hinzuzufügen, die jeweils nach den impliziten Beziehungen benannt wurden:

- Wirkt mit bei
- Entscheidet über
- Ist dv-verantwortlich für
- Ist fachlich verantwortlich für
- Muss bei Abbruch informiert werden
- Muss informieren über Ergebnis von
- Muss informiert werden über
- Stimmt zu
- Wirkt beratend mit

Die folgende Abbildung zeigt exemplarische eine EPK (Spaltendarstellung).

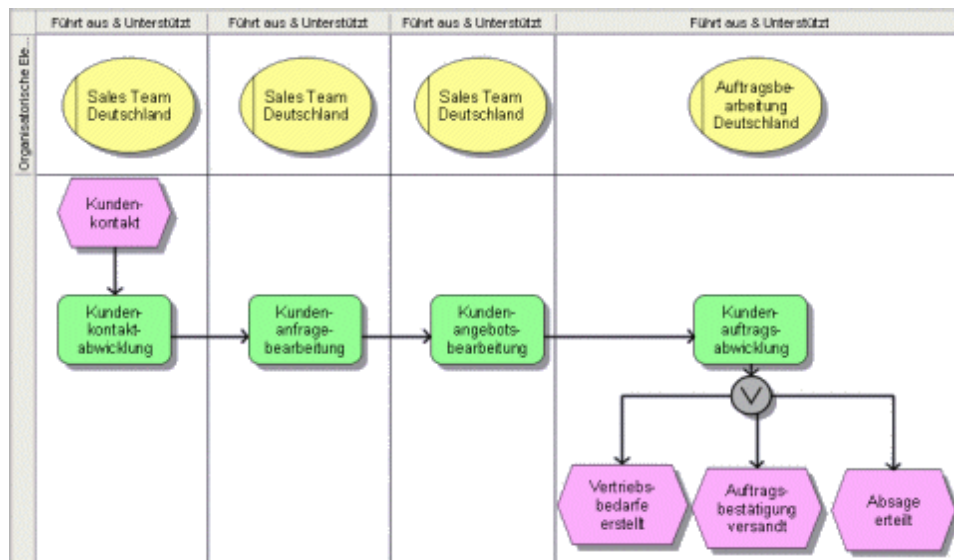


Abbildung 115: EPK (Spaltendarstellung)

Ein Unterschied zwischen der EPK (Spaltendarstellung) und der EPK (Zeilendarstellung) ist lediglich durch die sich unterscheidende Richtung der Modellierung gegeben. In der EPK (Spaltendarstellung) wird von oben nach unten, bei der EPK (Zeilendarstellung) von links nach rechts modelliert.

4.4.1.7 SAP ALE-Modelle

Modelle des Typs **SAP ALE** (SAP-Application Linking Enabling) dienen der Modellierung von Verbindungen zwischen Funktionen und Applikationen.

4.4.1.7.1 SAP ALE-Filtermodell

Durch das SAP ALE-Filtermodell wird wiedergegeben, nach welchen Kriterien ein Funktionstyp mehrfach in einem System (in einer Applikation) verwendet werden kann. Insofern dient das Objekt **Filterobjekttyp** als Teilungskriterium innerhalb eines Verteilungsmodells.

4.4.1.7.2 SAP ALE-Nachrichtenflussmodell

Durch das SAP ALE-Nachrichtenflussmodell werden mit Hilfe des Objekttyps **Nachrichtenfluss** die Nachrichtenflüsse samt ihrer Richtung zwischen Systemen und Funktionstypen wiedergegeben.

4.4.1.7.3 SAP ALE-Nachrichtentypmodell

Durch das SAP ALE-Nachrichtentypmodell werden mit Hilfe des Objekttyps **Nachrichtentyp** die zwischen verteilten Systemen ausgetauschten Nachrichten in Abhängigkeit von den beteiligten Daten und Vorgängen klassifiziert.

4.4.1.8 Role Allocation Diagram (RAD)

Das SAP-Referenzmodell wird mit Hilfe von EPKs dargestellt. Sie bilden die betrieblichen Abläufe auf unterschiedlichen Detaillierungsstufen ab. In den EPKs mit dem höchsten Detaillierungsgrad, die der SAP-Terminologie als **Prozesse** bezeichnet werden, werden die Handlungsabläufe beim Abarbeiten von Transaktionen im SAP-System modelliert. Diesen Prozessen können sowohl Rollen als auch Transaktionen zugeordnet werden.

In ARIS geschieht dies im Funktionszuordnungsdiagramm, wobei die EPK, die den modellierten Prozess enthält, der entsprechenden Funktionsdefinition hinterlegt sein muss. Im Funktionszuordnungsdiagramm ist also ersichtlich, welche Rollen zur Ausführung der Transaktionen notwendig sind. Da es jedoch keine direkten Beziehungen zwischen Rollen und Transaktionen gibt, kann bei mehreren auftretenden Rollen nicht entschieden werden, für welche Transaktionen diese zuständig sind. Das Zuordnen von Rollen zu Transaktionen erfolgt deshalb im Rollenzuordnungsdiagramm (RAD). Pro Spalte wird eine Rolle dargestellt. In den Spalten werden die Transaktionen platziert, wodurch implizite Beziehungen erzeugt werden.

Die Informationen können während der R/3-Einführung verwendet werden, um die für den Betrieb des SAP-Systems notwendigen Benutzerprofile und Berechtigungskonzepte zu erstellen.



Abbildung 116: Rollenzuordnungsdiagramm (RAD - Role Allocation Diagram)

4.4.1.9 Weitere Modelle

4.4.1.9.1 Business Controls Diagram

In einem Business Controls Diagram werden zu einem Prozess bzw. einer Funktion mögliche Risiken und Risikobewältigungen abgebildet.

Ein Risiko (Risk) stellt eine mögliche Gefahr für einen Prozess dar, das angestrebte Prozessziel nicht zu erreichen.

Eine Risikobewältigung bzw. -bekämpfung (Control) stellt einen allgemeinen Weg dar, Risiken zu eliminieren oder zu vermindern.

Eine Risikolösung (Solution) stellt die Umsetzung einer Risikobekämpfung (Control) bezogen auf ein Risiko (Risk) dar.

Das Layout des Business Controls Diagram entspricht einer Matrix bzw. Tabelle. Auf der Abszisse werden die möglichen Prozessrisiken abgetragen und auf der Ordinate mögliche Risikobewältigungen. Risikolösungen werden nun als Verknüpfungen eines Risikos mit einer Risikobekämpfung eingefügt. Zusätzlich können dem Modell noch Organisationseinheiten (im Sinne von Benutzeranforderungen) und Dokumente hinzugefügt werden, die ebenfalls eine Umsetzung einer Risikobewältigung hinsichtlich eines Risikos unterstützen.

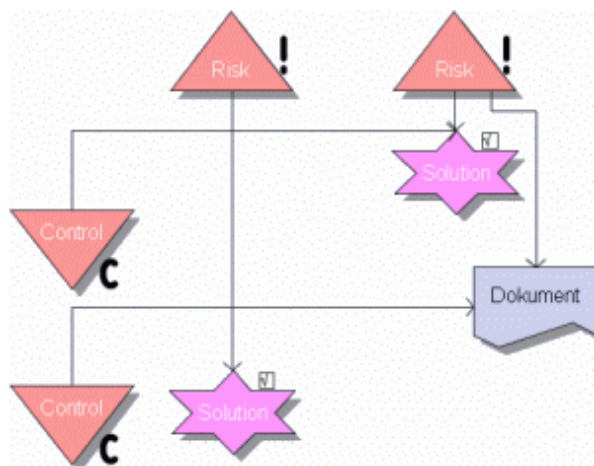


Abbildung 117: Beispiel eines Business Controls Diagramm

Angewendet wird der Modelltyp üblicherweise zur Beschreibung von SAP-Standardprozessen. Hierbei zeigt das Modell, welche Risiken und Risikobewältigungen eine SAP-Lösung bezüglich des betrachteten Prozesses aufweist.

4.4.1.9.2 DW-Transformation

Das Data Warehouse-Datentransformationsdiagramm dient der Beschreibung eines Data Warehouse. Im Vordergrund steht die Beschreibung der dynamischen Aspekte, die im ARIS-Konzept in der Prozesssicht beschrieben werden.

Im Modell wird die Überführung von allgemeinen Daten der Informationsobjekte in Datenformate des Info Cube dargestellt. Damit bei der Transformation effizient und ohne Datenverlust vorgegangen werden kann, ist eine Methodenbestimmung zur Modellierung erforderlich. Die Methodik beinhaltet Transformationsregeln und Vorgehensweisen, welche im Modell grafisch dargestellt werden.

Die Überführung erfolgt in zwei Schritten. Zuerst werden die Elemente der Transferstruktur in Elemente der Kommunikationsstruktur gewandelt. Dann werden diese in den Info Cube transferiert.

Es besteht die Möglichkeit diesen Vorgang auf unterschiedlichen Ebenen ablaufen zu lassen. Einerseits kann auf einer sehr hohen (fachkonzeptionellen) Abstraktionsebene gezeigt werden, welche Transfer- und Kommunikationsstrukturen vorliegen, um den Info Cube zu füllen. Andererseits kann auf einer tieferen (implementierungsnahen) Ebene abgebildet werden, wie die einzelnen Datenelemente ineinander überführt werden.

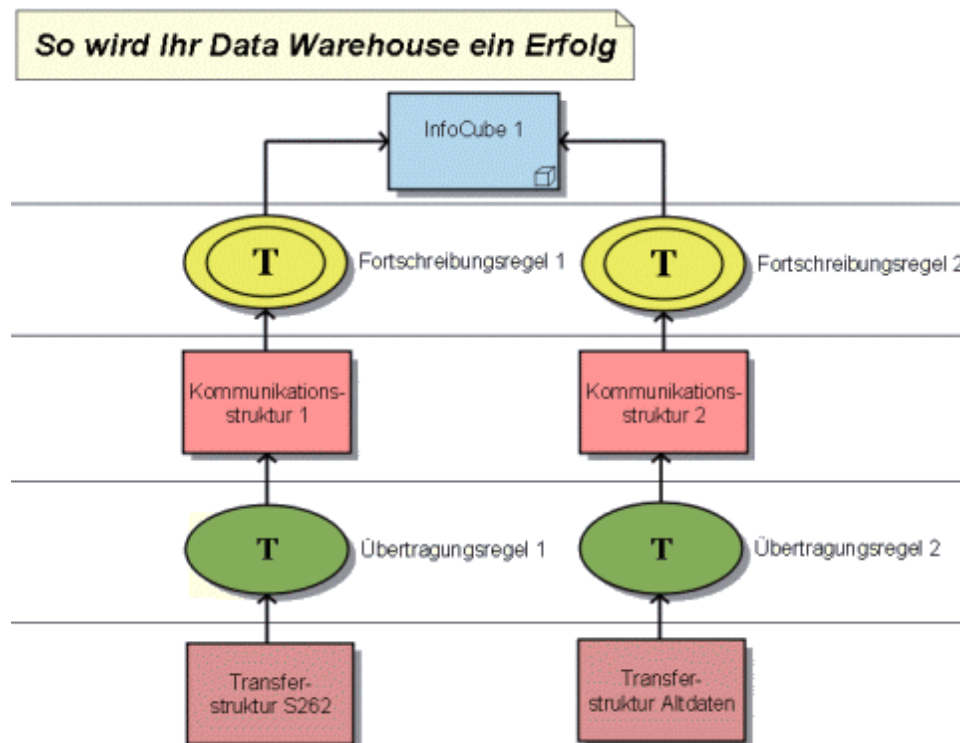


Abbildung 118: DW-Transformation - Datentransformation eines Data Warehouse

4.4.1.9.3 eBusiness Szenario Diagram

Die reibungslose Abfolge von unternehmensübergreifenden Geschäftsprozessen gewinnen ständig an Bedeutung. Dabei stehen auf der einen Seite der Ablauf bestimmter Vorgänge an den Schnittstellen zwischen den Unternehmen, auf der anderen Seite zwischen Unternehmen und ihren Kunden im Blickpunkt. Die Kontakte sollen übersichtlich, schnell, einheitlich und direkt erfolgen.

Auch das schnelle Finden von adäquaten Geschäftspartnern aus Unternehmenssicht und Anbietern aus Verbrauchersicht wird ständig wichtiger. Eine optimale Gestaltung dieser Prozesse bedeutet einen Wettbewerbsvorteil. Die geeignete Plattform zur Unterstützung dieser bilateralen Beziehungen ist das Internet. Da die Vorgänge innerhalb des oben beschriebenen Umfeldes sehr vielschichtig sind, muss definiert werden, was unter **E-Business** verstanden wird.

Der Begriff E-Business bezeichnet alle computergestützten Prozesse zwischen zwei Wirtschaftssubjekten und den Versuch durch neue Medien einen Mehrwert zu erzielen.

Dies kann somit der simple Kauf eines Artikels via Internet sein, ein hoch komplexes Projekt zweier Unternehmen oder die Erstellung einer Web-Site zur Unternehmenspräsentation.

Beziehungen Unternehmen - Unternehmen werden als Business-To-Business (kurz: B2B) bezeichnet, Beziehungen Kunde - Unternehmen als Business-To-Customer (kurz: B2C).

Zur Unterstützung des E-Business wurde das eBusiness Szenario Diagramm entwickelt.

Durch die Möglichkeit eine Wertschöpfungskette ganzheitlich zu betrachten, d.h. vom Endkunden bis zu allen an einem Vorgang beteiligten Unternehmen, soll Optimierungspotential entwickelt werden. Ziel hierbei ist beispielsweise die Verbesserung der Supply Chain, Senkung von Beschaffungs- und Vertriebskosten oder Optimierung der Architektur des Informationssystems. Die durch die Ziele dargestellten Inhalte werden durch diese Methode modellierbar.

Die Wirtschaftsobjekte werden im Diagramm in der oberen Zeile eingeordnet und als **Business Participant** bezeichnet. Die beteiligten Unternehmen können durch ein Organigramm hinterlegt werden. Zentrale Objekte sind hierbei die am Gesamtprozess beteiligten Einzelprozesse der Wirtschaftsobjekte und ihre Schnittstellen zueinander. Ein Einzelprozess, auch Business Process, ist ein wesentlich an unternehmensübergreifenden Kooperationen beteiligter Unternehmensprozess, der dem Prozessmodell hinterlegt werden kann. Der Business Process wird von Anwendungssystemen (Business Components) unterstützt, z. B. durch das R/3-System.

Auch die Rollen der am Prozess beteiligten Mitarbeiter kann definiert werden. Diese werden im Modell als **Employee Role** bezeichnet.

Die Schnittstellen sind durch die Übertragung prozessspezifischer Informationen charakterisiert. Sie sind in Business Documents zusammengefasst und können die Form eines XML- oder eines HTML-Dokuments annehmen. Auch das Business Document kann als Datenmodell hinterlegt werden. Alternativ zu diesem Objekt können auch die Objekte **Money Transaction** (zur Darstellung des Geldstroms), **Goods Shipment** (zur Darstellung des Güterstroms), **E-Mail**, **Intranet**, **Extranet** und **Handy** (zur Spezifizierung des technologischen Aspektes der Datenübertragung) verwendet werden.

Alle ein Unternehmen betreffenden Abläufe werden im Modell in der Zeile unter dem Business Participant, aber in derselben Spalte modelliert.

Daraus folgt, dass die Spaltenränder die abstrakten Schnittstellen bilden. Diesen gilt besonderes Augenmerk, da sie das Hauptoptimierungspotential beinhalten und deshalb für Modellierungszwecke besonders interessant sind.

Begriffserläuterung: Im folgenden Beispielmodell steht OEM für **Original Equipment Manufacturer** und MRP für **Material-Resource-Planing Controller**.

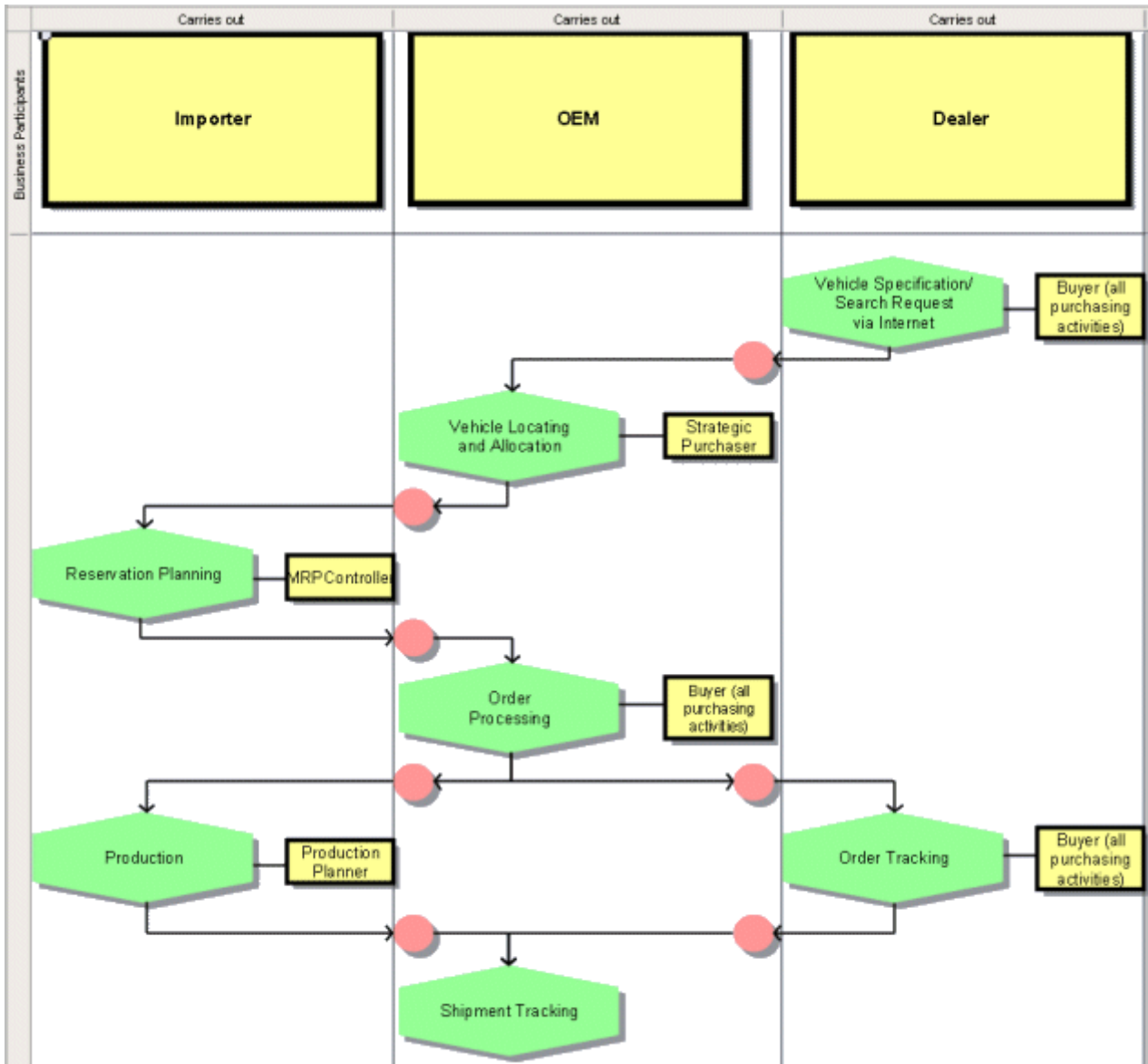


Abbildung 119: eBusiness Szenario Diagram-Beispiel: Automobilbranche

Im Beispielmodell wird das Zusammenspiel eines Herstellers, eines Importeurs und eines Händlers dargestellt. Jeder hat im Gesamtgebilde seine spezifischen Prozesse, die an den Schnittstellen zu den Prozessen anderer Geschäftspartner via Business Documents Informationen austauschen. Auch die an den Geschäftsprozessen beteiligten Personen werden mit ihren Rollen erfasst und zugeordnet.

4.4.1.9.4 Gliederungsmodell

Das Gliederungsmodell dient allgemein der Hierarchisierung bzw. Systematisierung eines Sachverhaltes (Spezialisierung bzw. Generalisierung von Sachverhalten).

Ein Gliederungselement stellt einen Sachverhalt (in Richtung der beabsichtigten Systematisierung) dar.

Hinter den einzelnen Gliederungselementen der Sachverhaltshierarchie können Modelle hinterlegt werden, die dem Sachverhalt zugeordnet werden.

Gebräuchlichstes Anwendungsgebiet des Gliederungsmodells ist das Qualitätsmanagement, speziell die Zertifizierung. Dort wird durch das Gliederungsmodell eine Norm in ihre einzelnen Komponenten untergliedert, und bei den einzelnen Gliederungselementen werden die Modelle hinterlegt, die zur Erfüllung des Qualitätskriteriums beitragen.

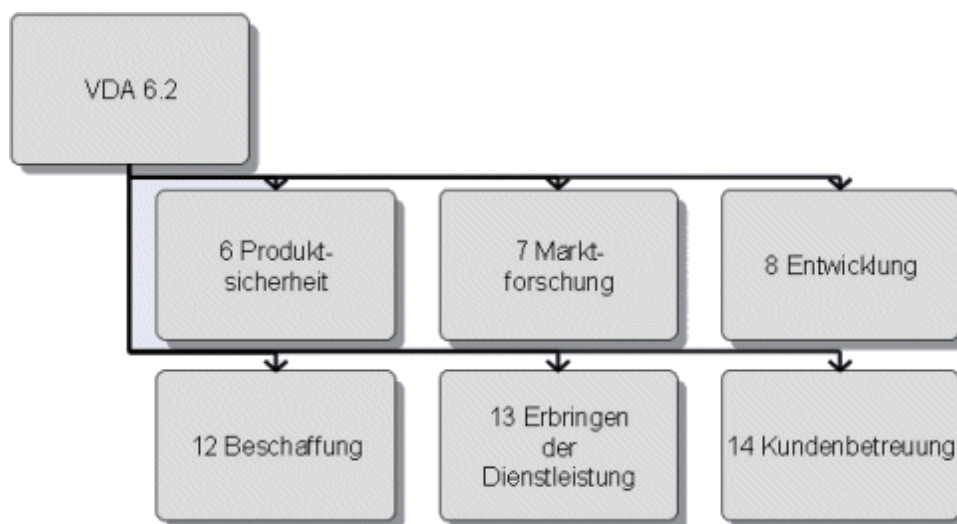


Abbildung 120: Beispiel eines Gliederungsmodells (Auszug aus der Norm VDA 6.2)

Durch einen Report können diese Sachverhalte leicht ausgewertet werden bzw. für Dokumentationszwecke benutzt werden.






















4.4.1.9.5 Industrial Process und Office Process




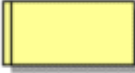
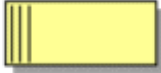

















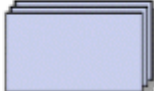






































Mit den Modelltypen **Industrial Process** und **Office Process** werden im Wesentlichen die gleichen Sachverhalte abgebildet wie mit dem Modelltyp EPK bzw. EPK (Materialfluss), jedoch steht in diesen beiden Modellen nur eine beschränkte Auswahl von Objekten zur Verfügung und die Symbole weisen eine bildhafte Darstellung auf.

Die bildhafte Darstellung hat den Vorteil, dass Mitarbeiter in den Fachabteilungen ohne Schulung die Modelle verstehen und selbst anpassen bzw. entwickeln können. So ist z.B. für jeden leicht ersichtlich, dass ein Symbol, welches drei Personen darstellt, eine Gruppe repräsentiert; hingegen ist dies bei der abstrakten Symbolik der EPK (doppelt umrandetes Oval) nicht für jedermann sofort ersichtlich. Ziel der beiden Modelltypen ist folglich, die Prozessmodellierung, -optimierung und -nutzung in die Fachabteilungen zu bringen.

Um die Identifikation mit der Symbolik zu maximieren, werden zwei Prozessarten (Modelltypen) unterschieden: der Industrial Process, der Fertigungsprozesse abbildet (Erstellung eines materiellen Gutes/Produktes) und der Office Process, der Büroprozesse abbildet (Erstellung eines immateriellen Gutes/Dienstleistung).

Die folgende Tabelle stellt die Symbole des Industrial Process und des Office Process denen der EPK bzw. EPK (Materialfluss) gegenüber.

Objekttyp	Mögliche Symbole innerhalb des Modelltyps		
	EPK	Industrial Process	Office Process
Ereignis			
Funktion		 Funktion (Fertigung)	 Funktion (Büro)
Regel			
Anwendungssystemtyp			
Standort	 Arbeitsplatz Standort		
Organisationseinheit			
Gruppe			

Objekttyp	Mögliche Symbole innerhalb des Modelltyps		
	EPK	Industrial Process	Office Process
Stelle			
Personentyp	 Personentyp  Stellenbeschreibung		
Person	 Person intern  Person extern	 Person (männlich)  Person (weiblich)	 Person (männlich)  Person (weiblich)
Wissenskategorie			
Dokumentiertes Wissen			
Informationsträger	 Barcode  Datei  Fax  Kartei  Magnetband  Microfiche  Telefon  Dokument  Ordner  Know-how	 Drucker  Dokument  Diskette  Karteikasten  Notizblock  Telefon  Ablagekorb  Email	 Drucker  Dokument  Diskette  Karteikasten  Notizblock  Telefon  Ablagekorb  Email  Zeitplaner  Papierkorb  Internet  Buch  Festplatte  CD-ROM  Fax  Brief  Akte  Magnetband
Materialtyp			-
Transportsystemtyp		 Flugzeug  LKW  Transport-system	-

Objekttyp	Mögliche Symbole innerhalb des Modelltyps		
	EPK	Industrial Process	Office Process
Betriebsmitteltyp		 Maschine Roboter	-
Techn. Hilfsmitteltyp			-
Packmitteltyp			-
Lagereinrichtungstyp			-

Die Modelle können in allen drei Modelltypen dargestellt werden (soweit die Objekte im jeweiligen Modelltyp existieren), indem man den Inhalt des einen Modelltyps in einen anderen kopiert. Beim Kopieren wird in ARIS automatisch eine Symbolkonvertierung durchgeführt. Ein Beispiel für die Darstellung des gleichen Sachverhaltes mit den drei Modelltypen **EPK**, **Industrial Process** und **Office Process** ist in folgende Abbildung dargestellt.

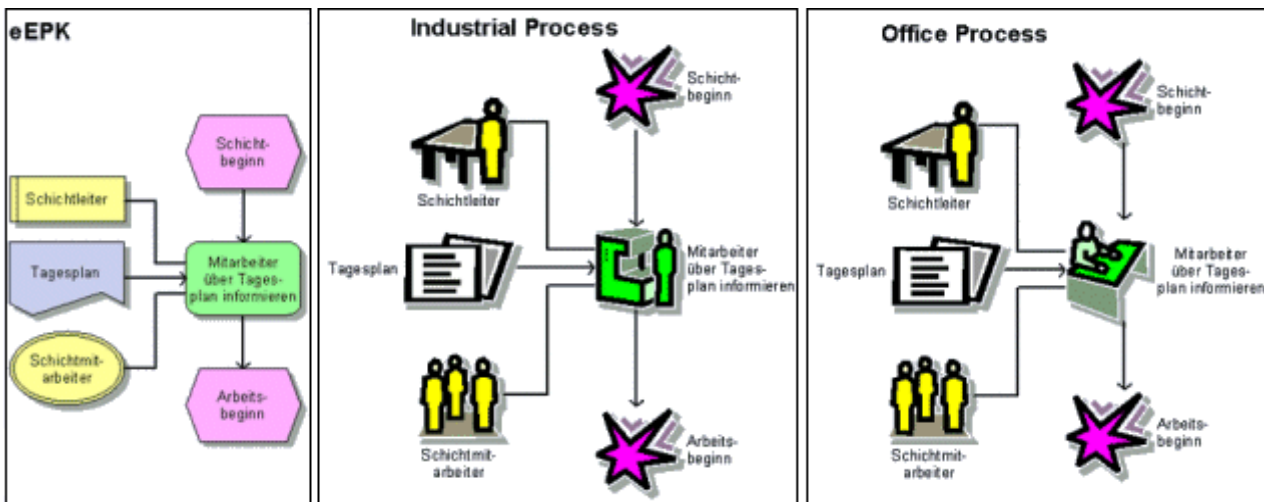


Abbildung 121: Beispielsachverhalt in den Modelltypen EPK, Industrial Process und Office Process

4.4.1.9.6 Projektprozesskette (PPK)

Der Modelltyp **PPK** stellt das Bindeglied zwischen ARIS und MS Project dar. Die Ablauffolge von Funktionen im Sinne eines Geschäftsprozesses kann im ARIS Business Architect durch ereignisgesteuerte Prozessketten (EPKs) dargestellt werden. Zur Kapazitäts- und Zeitplanung eines Projektes ist es jedoch nicht ausreichend, auf dieser abstrakten Ebene zu bleiben. Vielmehr müssen konkret auftretende Ereignisinstanzen und Aufgaben betrachtet und spezifiziert werden. Diesem Sachverhalt trägt die PPK dadurch Rechnung, dass sie an Stelle der Objekttypen **Ereignis** und **Funktion** jeweils einen eigenen Objekttyp auf Ausprägungsebene zur Verfügung stellt.

Mit Hilfe der früheren ARIS MS Project-Schnittstelle konnten aus EPKs automatisch PPKs generierte werden. Die heutige Schnittstelle zwischen ARIS Business Architect und MS Project 2000 greift nicht mehr auf die PPK zurück. Der Modelltyp spielt heute nur noch eine untergeordnete Rolle und wird im Wesentlichen noch zum Anzeigen der früheren Modelle verwendet.

Eine Ereignisinstanz ist ein in einer konkreten Prozessinstanz vorkommendes Ereignis. Die Ereignisinstanz kann evaluiert werden, d. h., es kann entschieden werden, ob sie wahr oder falsch ist.

Eine Aufgabe ist eine in einer konkreten Prozessinstanz vorkommende Funktion. Ihr können ein eindeutiges Startdatum und Endedatum sowie weitere notwendige Attribute zugeordnet werden. Die Projektstrukturelemente (Aufgabe, Ereignisinstanz, Regel und Kanten) dienen dazu, die zeitlich-logische Abfolge des Projektes darzustellen. Daneben beinhaltet die PPK die Ressourcenobjekte **Person** (intern/extern), **Betriebsmittel** und **Allgemeine Ressource**. Sie dienen der Planung von Terminen und Kapazitäten.

Eine Allgemeine Ressource ist eine nicht näher definierbare Ressource, die weder eine Person noch ein Betriebsmittel sein muss. Die allgemeine Ressource dient der Durchführung von Vorgängen.

Daneben besteht die Möglichkeit, innerhalb der PPK die Aufgaben durch Clusterinstanzen näher zu spezifizieren.

Eine Clusterinstanz ist eine Instanziierung des Objektes Cluster/Datenmodell. Sie stellt eine logische Sicht auf eine Ansammlung von Datenobjekten bzw. Strukturen dar.

In der PPK werden die Clusterinstanzen eingesetzt, um die Beziehung zwischen Aufgaben und Daten darzustellen. Der Objekttyp **Clusterinstanz** ist mit einem Modell des Typs **Informationsträgerdiagramm** (siehe Fachkonzept der Datensicht) hinterlegbar. Dadurch kann dargestellt werden, auf welchem Informationsträger die Daten abgelegt sind.

In der folgenden Abbildung ist das Beispiel einer durch Umwandlung aus einer EPK entstandenen Projektprozesskette wiedergegeben.

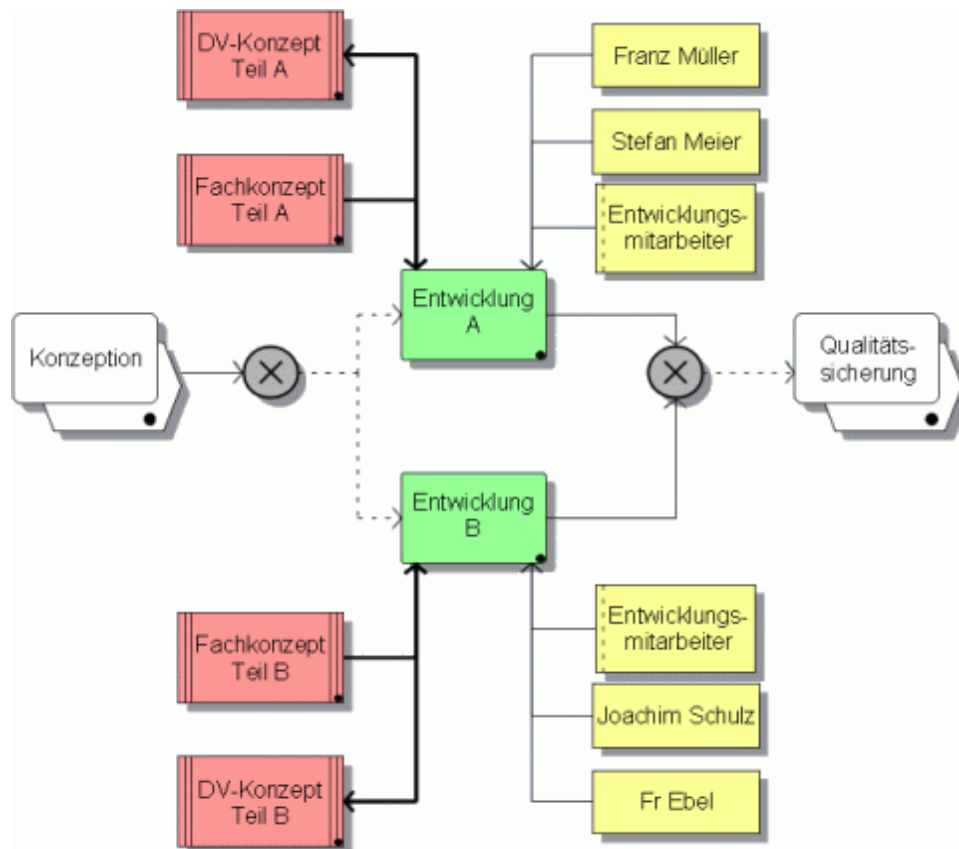


Abbildung 122: Beispiel einer aus einer EPK entstandenen PPK

Die XOR-Regel im angegebenen Beispiel zeigt, dass in der umgewandelten EPK an dieser Stelle Verzweigungen aufgetreten sind, die für das Projekt als alternative Pfade interpretiert werden und eindeutig spezifiziert werden müssen.

Objekte, die mit Modellen hinterlegt sind, werden durch einen schwarzen Punkt am unteren Rand gekennzeichnet. Im obigen Beispiel sind dies die Objekte **Konzeption**, **Qualitätssicherung** (beides Projektwegweiser) und Fachkonzept Teil B (Clusterinstanz).

Der Modelltyp **PPK** kann vom Benutzer auch direkt in ARIS Business Architect modelliert werden.

4.4.1.9.7 Prozessinstanziierungsmodell

Hauptaspekt einer dynamischen Simulation ist die Analyse von Prozessabläufen während ihres dynamischen Ablaufs. Die zu analysierenden Prozesse werden an Startereignissen instanziiert (gestartet bzw. erzeugt). Der Anwender muss die Möglichkeit besitzen, gemäß seinem Anwendungsgebiet selbst vorzugeben, wann und wie oft Prozesse instanziiert werden sollen. Zusätzlich muss es möglich sein, Prozesse zu priorisieren, um z. B. Eilprozesse berücksichtigen zu können.

Die Priorisierung wird durch die ARIS-Methode realisiert, indem das Attribut **Priorität** (Attributtypgruppe **Simulation**) von Startereignissen gepflegt wird und alle Prozesse, die an dem entsprechenden Startereignis instanziiert werden, diese Priorität erhalten.

Die beschriebene Anforderung wird durch das Prozessinstanziierungsmodell erfüllt. Dieses wird als mehrstufiges Objektmodell entwickelt. Auf unterster Ebene befindet sich das Objekt **Instanzierungsintervall**. Ein solches Intervall enthält die Attribute **Relativer Intervallanfang**, **Intervalldauer**, **Anzahl Prozessinstanzen** und **Verteilung**, **Zyklisch wiederholen** und **Periode**. Dabei ist es zulässig, dass die Intervalldauer gleich 0 ist, um einen bestimmten Zeitpunkt ausdrücken zu können. Während die Intervalle kleinere Zeiträume beschreiben, dienen die Prozessinstanzierungszyklen dazu, eine immer wiederkehrende Folge von Intervallen zu wiederholen. So kann ein Tag z. B. durch vier verschiedene Intervalle modelliert werden, die in einem Zyklus während des gesamten Simulationszeitraums (z. B. eine Woche) wiederholt werden. Es ist aber auch möglich, den Simulationszeitraum in mehrere Zyklen zu unterteilen (z. B. Werktag und Wochenende), wobei jeder Zyklus auch unterschiedliche Intervalle enthalten kann. Ein Prozessinstanzierungsplan kann einen oder mehrere Zyklen enthalten. Zur Verdeutlichung des Objektmodells soll folgendes Beispiel dienen:

Es liegt ein Prozessmodell als EPK mit einem Startereignis vor. Für diesen Prozess gelten folgende Annahmen: Werktags (Montag - Freitag) werden zu Dienstbeginn um 8:00 Uhr 50 Prozesse gestartet. In der Zeit zwischen 8:00 Uhr und 12:00 Uhr sowie zwischen 13:00 Uhr und 17:00 Uhr werden gleichverteilt 20 Prozesse gestartet; zwischen 12:00 Uhr und 13:00 Uhr sowie außerhalb der Dienstzeit fallen keine Prozesse an. Am Samstag werden dreiecksverteilt zwischen 9:00 Uhr und 15:00 Uhr 60 Prozesse gestartet. Sonntags fallen generell keine Prozesse an. Dieser Wochenrhythmus gilt von Januar bis Dezember mit Ausnahme der Urlaubszeit von Juli bis August. Während dieser Zeit wird samstags nicht gearbeitet.

Auf der Grundlage des beschriebenen Beispiels kann folgendes Modell erstellt werden:

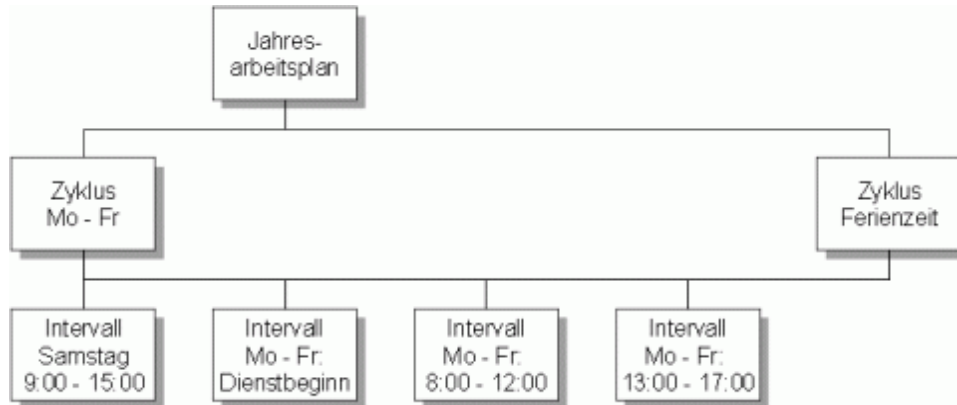


Abbildung 123: Prozessinstanziierungsmodell

4.4.1.9.8 RAMS

RAMS (Requirements Analysis for Management Systems) ist eine von Digital Equipment entwickelte Methode zur Unternehmensanalyse.

RAMS ist eine modellierte Vorgehensweise für die Sichtung und Beurteilung des Integrationspotentials von Informationstechnologie sowie zur Entwicklung von Lösungsszenarien für die Anforderungen an Informationssysteme. Das Ergebnis ist eine Anforderungsspezifikation, die sicherstellt, dass Unternehmensziele, Geschäftsabläufe, Informationsflüsse und Informationssysteme aufeinander abgestimmt sind.

Im Modell werden auf einer Diagonalen alle zu betrachtenden Abteilungen, Aktivitäten und bestehenden Anwendungen dargestellt. Diese Diagonale wird in Form einer Matrix um die wichtigsten Informationsflüsse zwischen den einzelnen funktionalen Einheiten ergänzt. Hinzuzufügen sind unter Umständen auch wichtige Güter-, Geld- oder Materialströme.

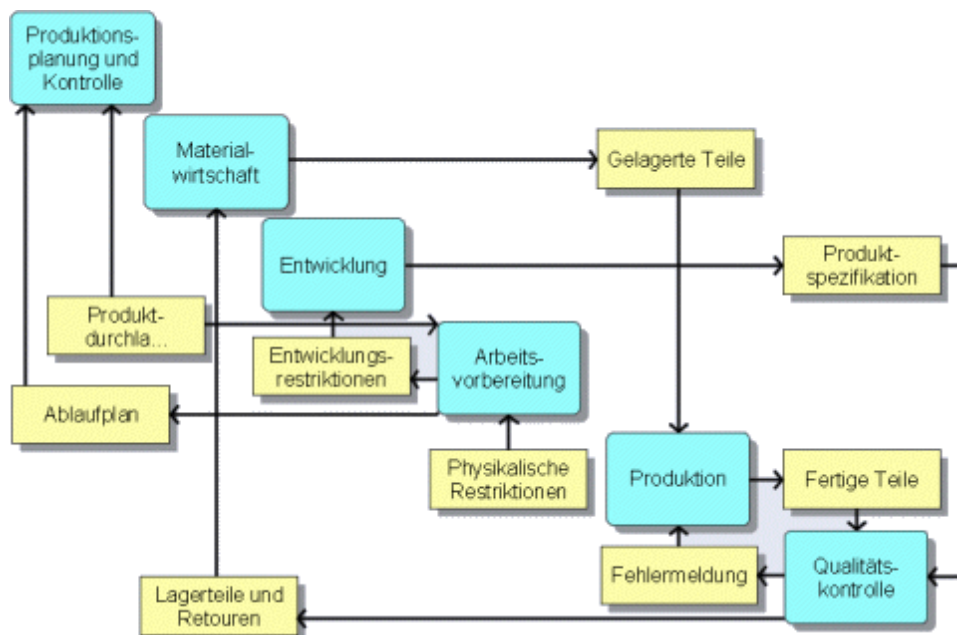


Abbildung 124: Beispiel eines RAMS-Diagramms

Vorgehensmodell einer RAMS-Studie:

1. Schritt 1: Der erste Schritt beinhaltet die Definition der erwarteten Studienergebnisse, die Benennung der einzubeziehenden Teilnehmer und eine erste Festlegung des Zeitplans.
2. Schritt 2: Die zweite Phase beginnt mit der Auswahl der verschiedenen Abteilungen, Aktivitäten und bestehenden Anwendungen, die in der Studie betrachtet werden sollen. Diese werden auf einer Diagonalmatrix dargestellt, auf der es anschließend auch gilt, die wichtigsten Informationsflüsse zwischen den einzelnen funktionalen Einheiten deutlich zu machen. Hinzuzufügen und sichtbar zu machen sind unter Umständen auch wichtige Güter-, Geld- oder Materialströme. Ebenfalls festgelegt werden zudem diejenigen Abteilungen oder Funktionen, für die eine detaillierte Anforderungsanalyse gewünscht wird.

3. Schritt 3: Nachdem der Umfang der Studie definiert worden ist, werden in Schritt 3 die ausgewählten Abteilungen oder Funktionsgruppen hinsichtlich ihrer Geschäftsziele und -prozesse sowie Aufgaben und verwandte Informationsbedürfnisse im Detail analysiert. Der kreative Gebrauch von Zeichnungen und Illustrationen unterstützt das Verständnis komplexer Prozesse oder Vorgehensweisen. Häufig werden zur Verdeutlichung auch originale Geschäftsformulare, Berichte oder Bildschirmmasken verwendet. Eine wichtige Aufgabe dieser Analyse liegt darin, bestehende Aktivitäten und Systeme auf mögliche Unregelmäßigkeiten im Verhältnis zwischen Informationsflüssen und Geschäftsabläufen zu durchleuchten. Weiterhin werden Möglichkeiten zur Verbesserung bereits vorhandener Aktivitäten und Systeme untersucht. Im Verlauf der Analyse sollen Probleme, Fragen und Vorschläge für optimale Lösungen herausgestellt werden. Diese Informationen werden anschließend strukturiert und genau auf ihre Ursachen und Wirkungen hinsichtlich der Geschäftsaktivitäten untersucht. Zeigen sich potentielle Verbesserungschancen, werden diese erfasst und ihr möglicher Nutzen bewertet.
4. Schritt 4: Die Ergebnisse der ausführlichen Ist-Analyse dienen nun als Basis der folgenden Anforderungsspezifikation. Für die jetzt klar umrissenen Problemfelder werden neue Ideen und alternative Lösungen entwickelt. Es gilt dabei stets, dass die vorgeschlagenen Lösungen - bei denen es sich sowohl um komplexe Systeme wie auch um einfache Prozessänderungen handeln kann - in engem Bezug zu der vorher erfassten Ausgangslage stehen müssen. Diese Abgleichung mit der Ausgangslage hat für sämtliche Funktionen und Aktivitäten in den relevanten Bereichen zu erfolgen. Das Ergebnis der Studie kann sowohl die Funktionalität generischer Lösungen sein als auch etwa die Erstellung und den Abgleich von Nutzeranforderungen umfassen.
5. Schritt 5: Die während der einzelnen Analyseschritte gesammelten Ergebnisse gehen ein in die Formulierung der abschließenden Anforderungsspezifikationen. Alle Informationen, Detailuntersuchungen und Vorschläge, die im Verlauf der Studie erarbeitet wurden, werden als Abschlussbericht zusammengefasst und bilden die Basis für Anforderungen an künftige Systeme. Als nächster Schritt zur Implementierung der vorgebrachten Lösungen hat dann eine funktionale Systemspezifikation zu erfolgen.

4.4.1.9.9 Rollendiagramm

Das Rollendiagramm dient generell der näheren Beschreibung von Prozessen. Im Vordergrund stehen die Organisationseinheiten, die an den Prozessen beteiligt sind, und deren Rollen. Charakterisiert werden die Objekte und ihre Beziehungen durch folgende Eigenschaften:

Eine Rolle ist unter Berücksichtigung von Berechtigungen an Prozessen beteiligt. Nicht nur die Durchführungsmöglichkeit ist bedeutend, sondern auch die Spezifizierung der Berechtigungsart im Prozess (eine Rolle ist „durchführend beteiligt“). Bei der Durchführung eines Prozesses mit einer bestimmten Berechtigung wird also die Beziehungskette Rolle - Beteiligung - Prozess (mit Beteiligung - Berechtigungsbedingung sowie Beteiligung - Berechtigungswert) aufgebaut.

Eine Rolle kann von Personen, Stellen oder Informationssystemen besetzt werden. Die Rolle bildet das Bindeglied zwischen den Prozessen und den an ihnen beteiligten Ressourcen. Sie wird durch eine Aggregation von Erwartungen an die an den Prozessen beteiligten Ressourcen definiert.

Die Durchführung eines Prozesses erfordert Fertigkeiten (Skills), die die beteiligte Rolle bzw. die zugewiesene Ressource aufweisen muss. Um Rollen überhaupt prozessorientiert definieren zu können, müssen die Prozesse begutachtet und die Anforderungen der Prozesse an die beteiligten Personen/Systemen zu bestimmt werden. Anforderungen an Personen/Systeme heißt genauer: Unter Anforderungen wird hier das Wissen und die Fähigkeiten (Skills) der Personen/Systeme verstanden. Die Bewertung einer Fähigkeit wird durch eine zugeordnete Bewertungsskala standardisiert.

Im Rollendiagramm besteht damit die Möglichkeit, Prozesse und bestimmte Elementarprozesse abzubilden, die daran beteiligten Ressourcen darzustellen, ihre Fähigkeiten bzw. benötigte Fähigkeiten festzuhalten und ihre Berechtigungen wiederzugeben.

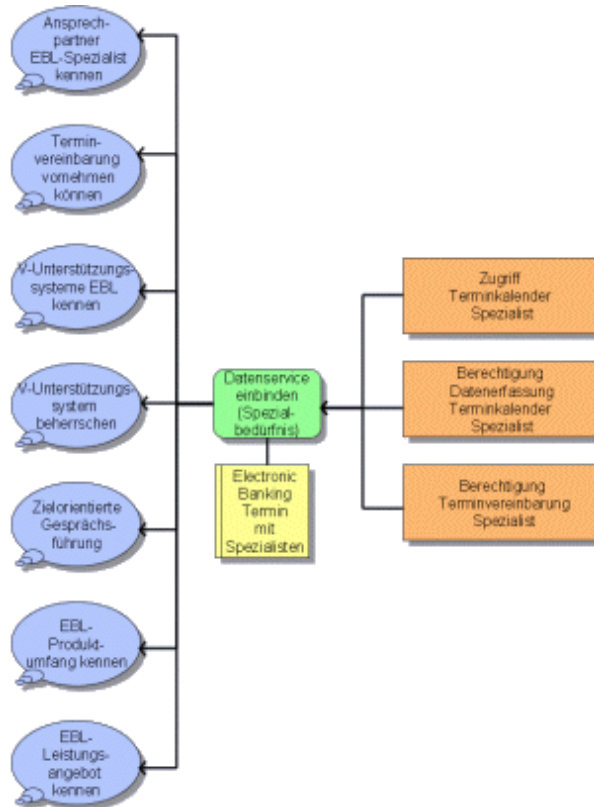


Abbildung 125: Rollendiagramm

Im Beispielmodell werden die Anforderungen des Elementarprozesses an die Rolle (Fähigkeiten und Berechtigungen) und ebenso die Anforderungen der Elementarrolle an die Ressource hinsichtlich der Fähigkeiten und Berechtigungen gezeigt.

Das Diagramm wird dem jeweiligen Elementarprozess und der jeweiligen Elementarrolle hinterlegt. Durch die Hinterlegung des Diagramms am Elementarprozess können die Anforderungen aus der zugrunde liegenden Prozess-EPK (entspricht dem Prozessreferenzmodell) eingesehen werden. Durch die Hinterlegung an der Elementarrolle kann die Anforderung der Elementarrolle an die Ressource hinsichtlich ihrer Fähigkeiten und Berechtigungen aus dem Rollenstrukturdiagramm eingesehen werden.

4.4.1.9.10 Casual Model

Das Modell des Typs **Casual Model** bietet Ihnen die Möglichkeit ohne methodische Einschränkungen zu modellieren. Innerhalb des Casual Model ist der Objekttyp **Casual Object** verfügbar, für den über 30 verschiedene Symbole angeboten werden. Zwischen den Casual Objects können Beziehungen des Typs **steht in Beziehung mit** angelegt werden. Es sind mehrere Kanten dieses Typs zwischen zwei Objekten erlaubt.

Für Modell, Objekt und Kante können die jeweiligen Standardattribute gepflegt werden.

Jedem Objekt eines beliebigen Objekttyps der ARIS-Methode können mehrere Casual Models hinterlegt werden. Einem Casual Object können wiederum beliebige viele Modelle der ARIS-Methode hinterlegt werden, unabhängig vom Modelltyp.

Sie können die Modelle des Typs **Casual Model** und/oder die Objekte des Typs **Casual Object** mit Hilfe des Semantics Generator in ARIS in methodenbasierte Modelle/Objekte transformieren.

4.4.1.9.11 C3-Methode

Mit einem Modell des Typs **c3-Methode** können Sie in einem Change Management-Projekt den Prozesseinstieg oberhalb der Prozessebene beschreiben.

Im Mittelpunkt steht hierbei der Prozess, der verbessert werden soll. Für jeden betrachteten Prozess wird eine Anzahl von verschiedenen Objekten modelliert, die in Listenform relevante Informationen zum Projekt abbilden. Hierzu gehören Informationen über

- organisatorische Aspekte, z. B. Prozessverantwortlichkeit und Stellvertreterregelung
- Aufgaben, die zur Prozessverbesserung durchgeführt werden
- Kennzahlen, mit denen die Prozessverbesserung gemessen wird
- Werkzeuge, die zur Prozessverbesserung eingesetzt werden
- Aktivitäten, die in naher Zukunft zur Veränderung des Prozesses geplant sind
- Verbesserungspotentiale für den betrachteten Prozess
- die Kompetenzen, die für die Ausführung des Prozesses benötigt werden
- Ziele, die mit dem Prozess verfolgt werden
- Tools, die heute eingesetzt werden (Software, Methoden, kontinuierliche Trainings)
- Tools, die zur Verbesserung des Prozesses und Integration in die Gesamtsystematik eingesetzt werden sollen

Der Aufbau des Typs **c3-Methode** wird durch folgende Grafik wiedergegeben:



Abbildung 126: Aufbau eines C3-Modells

Nur zwischen Prozess/Aufgabe und den Prozessbesitzern, -verantwortlichen oder Prozessbetreuern können Sie eine Kante ziehen.

Die übrigen Beziehungen ergeben sich aus der Position der Objekte im Modell.

Im Modell des Typs **c3-Methode** können die später erfassten Prozesse hinterlegt werden wie auch den Objekten der Organisationssicht.

Zur kurzen Beschreibung eines Objektes kann das Attribut **Beschreibung/Definition**: genutzt werden. Dieses Attribut wird durch den Report **c3-Projektblatt.rsm** ausgewertet.

4.4.1.9.12 Maskendesign

In einem Maskendesign können Sie im Rahmen des Software-Entwurfs mit ARIS die fachlichen Anforderungen spezifizieren, die an einen Dialog oder ein Webformular gestellt werden.

In der Spalte **Layout** bestimmen Sie die Struktur des Dialogs oder Webseiten-Formulars. Das Vorgehen beim Entwurf z. B. eines Dialogs ähnelt dem Arbeiten mit einem Ressource-Editor in einer Entwicklungsumgebung.

Zu den grafischen Bestandteilen, die in der Spalte **Layout** platziert werden können, gehören Textfelder, Drehfelder, Optionsfelder und Kontrollkästchen, Kombinationsfelder, Schaltflächen, Tree- und Listen-Steuerelemente sowie Bitmaps und statischer Text. Mit Hilfe des Attributtyps **Tabindex** können Sie die Reihenfolge festlegen, in der die einzelnen Maskenelemente mit der Tabulatortaste angesprochen werden können.

In den Spalten **Daten** und **Funktionen** können verschiedene Datenelemente bzw. Funktionsobjekte platziert werden. Über eine Kante des Typs **stellt dar** können die Objekte jeweils mit den Datenelementen und Funktionen, die sie editieren, in Beziehung gesetzt werden.

Jedes Maskendesign kann dem entsprechenden Maskenobjekt hinterlegt werden, das z. B. in einer EPK oder einem Modell des Typs **Maskennavigation** verwendet wird. Darüber hinaus kann ein Maskendesign aber auch dem Entitytyp, dem Cluster, dem komplexen Objekttyp, der Klasse oder der Funktion/DV-Funktion hinterlegt werden, welche mit der Maske bearbeitet werden.

Mit Hilfe des Reports **Maskendesign.rsm** können die Informationen des Maskendesign in eine Textdatei exportiert werden. Die Textdatei kann in der Entwicklungsumgebung **C++** weiterverwendet werden.

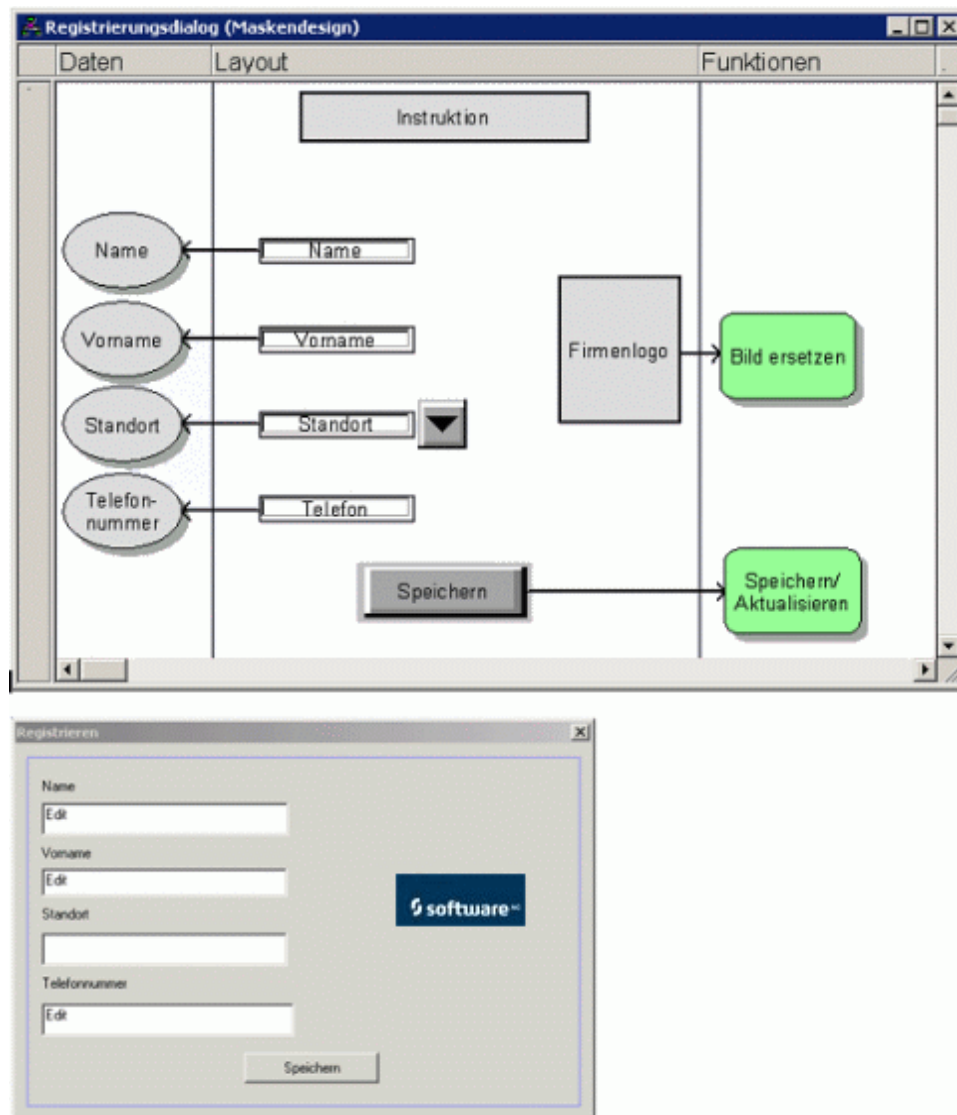


Abbildung 127: Beispiel für das Maskendesign eines Registrierungsdialogs und die Umsetzung in C++

4.4.1.9.13 Maskennavigation

In einem Modell des Typs **Maskennavigation** können Sie entweder den Aufbau einer Maske spezifizieren, die aus mehreren Teilmasken besteht, z. B. eine Website mit mehreren Formularfeldern oder Frames. Oder Sie können die Übergänge zwischen verschiedenen Masken beschreiben. Die Übergänge zwischen den Masken können detailliert werden.

Beispiel

Sie möchten verdeutlichen, dass ein Maskenelement betätigt werden muss um in eine andere Maske zu wechseln. Ordnen Sie das auslösende Maskenelement (des Modells **Maskendesign**) mit Hilfe der Kante **beinhaltet** der Maske zu. Ziehen Sie danach eine Kante des Typs **ruft auf** vom Maskenelement zur nachfolgenden Maske.

Ebenso ist es möglich, die Abhängigkeit der Navigation von Ereignissen darzustellen. Beim Verlassen einer Maske können unterschiedliche Ereignisse eintreten. Wenn z. B. ein Benutzer die Registrierungsseite eines Online-Shops ausgefüllt hat, kann das Registrieren erfolgreich durchgeführt werden oder fehlschlagen. Abhängig davon wechselt er entweder auf die Inhaltsseite des Katalogs oder er kehrt zur Registrierungsseite zurück.

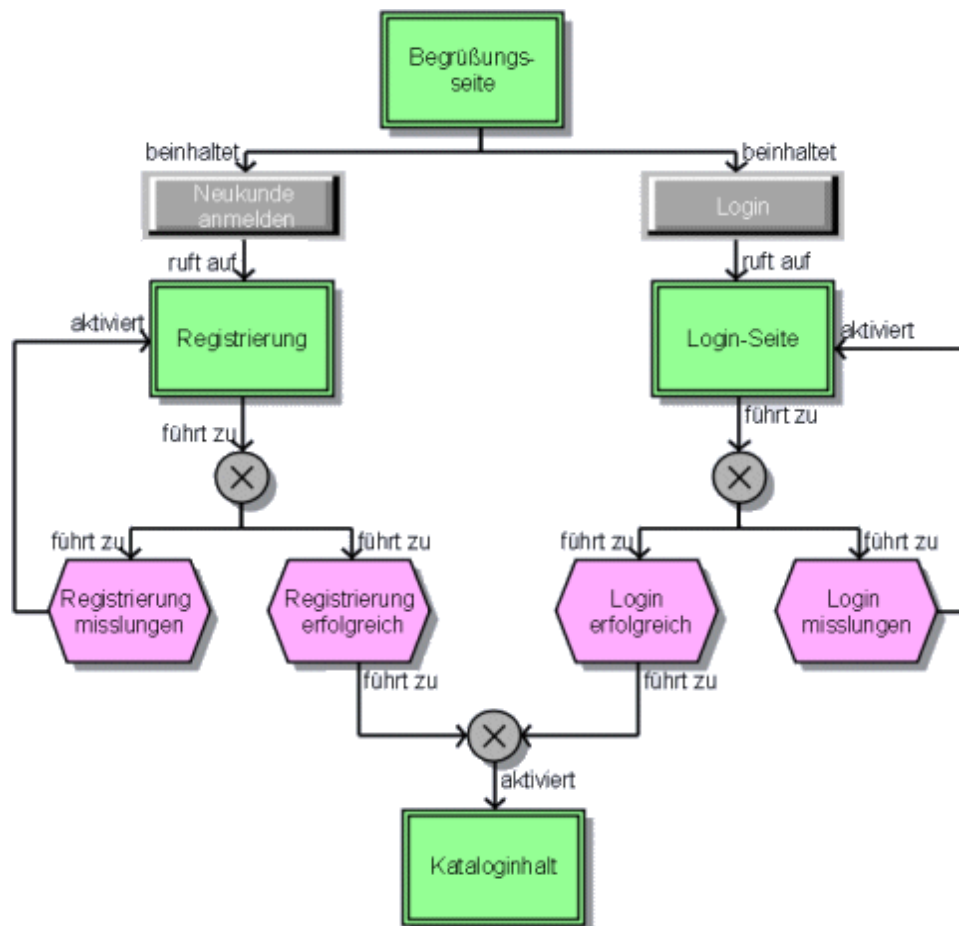


Abbildung 128: Beispiel für die Maskennavigation mit Ereignissen

4.4.1.9.14 Geschäftsfelder-Matrix

In der Geschäftsfelder-Matrix werden die verschiedenen Märkte, in denen ein Unternehmen tätig ist, im Überblick dargestellt und ihre Bedeutung für den Unternehmenserfolg visualisiert.

Jeder Markt wird beschrieben

- durch das angebotene Produkt oder die angebotene Dienstleistung und
- die Kundengruppe, an die sich das Angebot richtet.

Produkte und Dienstleistungen (Objekte des Typs **Leistung**) werden in den Zellen der ersten Spalte der Geschäftsfeldermatrix platziert. Die Zielgruppe (verschiedene Organisationselemente) in den Zellen der ersten Zeile. Sie definieren den Markt, indem Sie ein Geschäftsfeld-Objekt in der Zelle platzieren, in der sich Produktzeile und Zielgruppenspalte treffen. Dabei werden zwischen dem Produkt/der Dienstleistung und dem Organisationselement implizite Beziehungen des Typs **gehört zu Geschäftsfeld** aufgebaut.

Um die Bedeutung eines Geschäftsfeldes gegenüber den anderen hervorzuheben, stehen fünf verschiedene Symbole zur Verfügung, mit denen die Wichtigkeit von **fast unbedeutend** bis **sehr bedeutend** zum Ausdruck gebracht werden kann.

Beim Modellieren müssen Sie beachten, dass jedes Geschäftsfeld nur einmal in der Matrix platziert werden darf.

Für jedes Geschäftsfeld kann angegeben werden, im Rahmen welcher Unternehmensstrategie es von Bedeutung ist. Eine Strategie beschreibt langfristige Verfahrensweisen, mit denen sich das Unternehmen gegenüber seinem Umfeld zur Verwirklichung seiner langfristigen Ziele zu behaupten versucht.

Die folgende Abbildung zeigt eine Geschäftsfelder-Matrix aus dem pharmazeutischen Bereich.

	Gehört zu	Gehört zu	Gehört zu	Gehört zu	Gehört zu	Gehört zu	Gehört zu
Markt							
Gehört zu							
Gehört zu							

Abbildung 129: Beispiel Geschäftsfelder-Matrix

Geschäftsfelder können auch mit einem Zieldiagramm hinterlegt werden. Das Zieldiagramm enthält die für das Geschäftsfeld gesetzten Ziele sowie die Prozesse und Erfolgsfaktoren, die die Zielerreichung unterstützen.

Die Erfolgsfaktoren im Zieldiagramm können Basis für eine Erfolgsfaktorenanalyse sein, wenn in der gleichnamigen Attributtypgruppe die Attribute **Erfolg - Ist**, **Erfolg - Soll** und **Erfolg -**

Wettbewerber gepflegt sind. Der Erfolg kann anhand einer 5-stufigen Skala von **sehr gering** bis **sehr groß** beurteilt werden.

Vorgehen

Um eine Erfolgsfaktorenanalyse durchzuführen,

1. starten Sie mit dem Kontextmenü des Geschäftsfeldes den ARIS-Report (**Auswerten/Report**)
2. und wählen im Report-Assistenten das Reportskript **HEF-Analyse (Objekt).rso** der Gruppe **BPM** im Standardpfad.

Die Reportausgabe erfolgt im Format HTML.

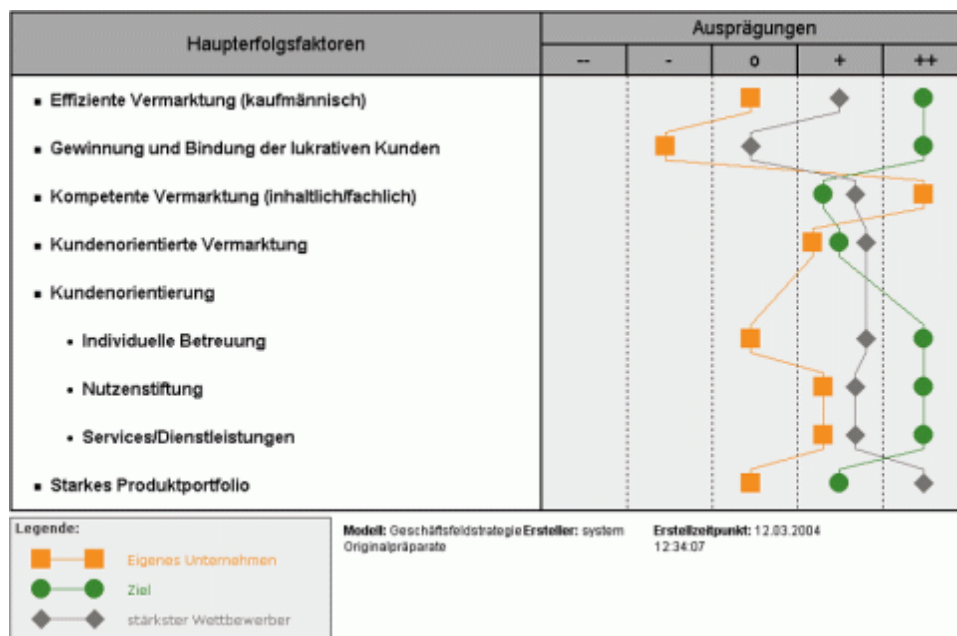


Abbildung 130: Report

Alternativ kann die Erfolgsfaktorenanalyse auch mit dem Kontextmenü des Ziendiagramms gestartet werden. Wählen Sie dann das Reportskript **HEF-Analyse (Modell).rsm**.

4.4.2 DV-Konzept

4.4.2.1 Zugriffsdiagramm

Die nachfolgend dargestellten Beziehungen zwischen den Objekten, die in den DV-Konzept-Beschreibungen der anderen Sichten erläutert wurden, können im Zugriffsdiagramm der Steuerungssicht aufgenommen werden. Zur besseren Übersichtlichkeit der Darstellung werden die einzelnen Zweierbeziehungen getrennt behandelt.

4.4.2.1.1 Funktionen mit Daten

Hier können zunächst die Informationsflüsse die zwischen Anwendungssystemtypen, Modultypen oder DV-Funktionen bestehen, definiert werden. Zu diesem Zweck wird zwischen den jeweiligen Anwendungssystemtypen oder Modultypen ein Informationsflussobjekt angelegt. Um den Informationsfluss zwischen den Systemtypen näher spezifizieren zu können, wird mit dem Informationsflussobjekt ein eERM-Diagramm, ein Relationendiagramm oder ein Tabellendiagramm verbunden. Die Objekte des Informationsflusses können sich somit auf der Fachkonzept-Ebene, der DV-Konzept-Ebene oder der Implementierungsebene befinden.

Ein Beispiel zeigt diese Abbildung.

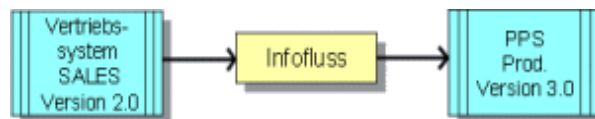


Abbildung 131: Datenfluss zwischen Anwendungssystemtypen

Neben den Informationsflüssen können auch für jeden Anwendungssystemtyp, Modultyp und jeden DV-Funktionstyp die Input- und Outputdaten in Form von Datenobjekten des Fachkonzeptes oder des DV-Konzeptes dargestellt werden. Anhand der Pfeilrichtung kann dabei spezifiziert werden, ob es sich um einen eingehenden (Input) oder einen ausgehenden (Output) Datenstrom handelt.

Ein Beispiel zeigt folgende Abbildung.

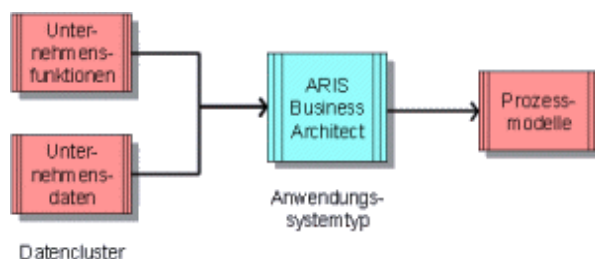


Abbildung 132: I/O-Daten auf DV-Konzept-Ebene

4.4.2.1.2 Organisation mit Daten

Die wesentlichen Aufgabenstellungen bei der Verbindung von Datensicht und Organisationsicht auf DV-Konzept-Ebene bestehen in der Definition der Verantwortlichkeiten von organisatorischen Einheiten für Datenobjekte des Unternehmens und der Festlegung von Zugriffsrechten, die im Wesentlichen definieren, welche Organisationseinheiten auf bestimmte Daten des Unternehmens zugreifen dürfen.

Die Beziehungen, die dabei aufgebaut werden, verbinden fachliche Objekte der Organisationsicht (Organisationseinheit, Stelle, Personentyp, Person, ...) mit Datenobjekten des Relationendiagramms auf DV-Konzept-Ebene (Relation, Attribut, View). Aus diesem Grunde sind diese Beziehungen auch der DV-Konzept-Ebene der Steuerungssicht zugeordnet.

Zur Definition der Zugriffsberechtigungen auf Relationen oder einzelne Felder können sowohl Stellen als auch Personentypen diesen Datenobjekttypen zugeordnet werden. Damit kann zum einen definiert werden, dass eine bestimmte Stelle Zugriffsberechtigungen auf bestimmte Felder besitzt, es können zum anderen aber auch mit der Zuordnung der Personentypen Geschäftsregeln in der Form **auf dieses Feld darf nur ein Abteilungsleiter zugreifen** definiert werden. Ein Beispiel hierzu zeigt folgende Abbildung.

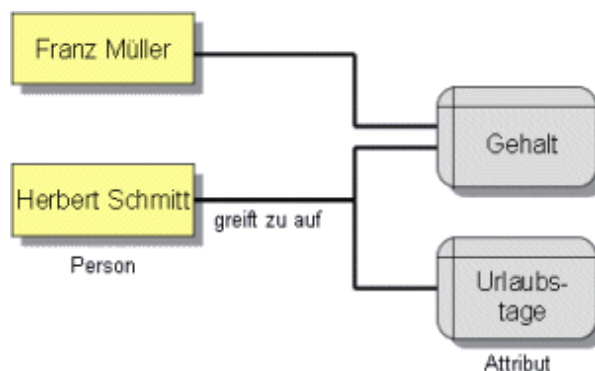


Abbildung 133: Zugriffsrechte

Neben der Zugriffsberechtigung ist auch die Definition der Verantwortlichkeit für die Inhalte eines Feldes oder einer gesamten Relation von Bedeutung. Aus diesem Grunde steht zwischen den organisatorischen Einheiten und den Datenobjekten des Relationendiagramms eine zweite Kante **ist verantwortlich für** zur Verfügung. Verantwortlichkeiten für Datenobjekte sind im Gegensatz zu den Zugriffsberechtigungen meist nur einer Stelle im Unternehmen zugeordnet. Auch hier besteht die Möglichkeit, durch die Zuordnung von Personentypen, Geschäftsregeln in der oben beschriebenen Form zu definieren, die sich dann natürlich auf die Verantwortlichkeit für ein Datenobjekt beziehen.

Ein Beispiel hierfür zeigt diese Abbildung.



Abbildung 134: Definition der Verantwortlichkeiten

4.4.2.1.3 Organisation mit Funktion

Die Verbindungen der organisatorischen Aspekte mit den auf DV-Konzept-Ebene definierten funktionalen Aspekten dienen im Wesentlichen zur Beantwortung folgender Fragestellungen:

- Wer (welche Organisationseinheiten, Stellen, Personen,...) ist für die in der Funktionsicht auf DV-Konzept-Ebene spezifizierten Anwendungssystemtypen und Modultypen verantwortlich oder ist Anwender dieser Systeme?
- An welchen Standorten (Organisationssicht) im Unternehmen werden Anwendungssystemtypen oder Modultypen eingesetzt?
- Auf welchen zur Verfügung stehenden Plattformen des Unternehmens (Hardware-Komponententypen (Organisationssicht)) können Anwendungssystemtypen laufen?

Zur Beantwortung der ersten Fragestellung können im Zugriffsdiagramm Kanten zwischen den organisatorischen Einheiten des Organigramms (Organisationseinheiten, Stellen und Personen) und den Objekten des Anwendungssystemtypdiagramms (Anwendungssystemtyp, Modultyp, DV-Funktion,...) gezogen werden. Hierbei kann nochmals genauer spezifiziert werden, welche Bedeutung diese Beziehung haben soll. Es werden unterschieden:

- Eine Organisationseinheit **kann fachlich** für einen Anwendungssystemtyp **verantwortlich** sein.
- Eine Organisationseinheit **kann für die Entwicklung** eines Anwendungssystemtyps **verantwortlich** sein.
- Eine Organisationseinheit kann **Anwender** eines Anwendungssystemtyps sein.

Zur Beantwortung der Standortfrage können Standorte der Organisationssicht den Anwendungssystemtypen, Modultypen und DV-Funktionstypen zugeordnet werden.

Da es sich im DV-Konzept nicht um einzelne Anwendungssysteme im Sinne einer einzelnen Lizenz, sondern um Anwendungssystemtypen handelt, werden mit dieser Beziehung nicht die konkreten Standorte von Anwendungssystemen definiert (diese Zuordnungen erfolgen auf Implementierungsebene), sondern es wird eine Aussage darüber getroffen, an welchen Standorten sich ein Anwendungssystemtyp befinden kann.

Im DV-Konzept der Organisationssicht wird definiert, welche Hardwarekomponententypen im Unternehmen zur Verfügung stehen. In der Steuerungssicht können nun diese Hardware-Komponententypen mit den Anwendungssystemtypen in Beziehung gesetzt werden. Dadurch wird festgelegt, auf welchen Hardware-Plattformen Anwendungssystemtypen, Modultypen oder DV-Funktionstypen lauffähig sind. An dieser Stelle können auch die in der Funktionsicht aufgenommenen Benutzeroberflächentypen, Betriebssystemtypen und DBMS-Typen den Hardware-Komponententypen zugeordnet werden.

Die Gesamtheit der möglichen Beziehungen des Zugriffsdiagramms befindet sich im Handbuch **ARIS-Methode – Tabellen** (Datei **method_tabulations_aris.pdf**) auf Ihrem Installationsmedium.

In folgender Abbildung sind beispielhaft Beziehungen dargestellt.

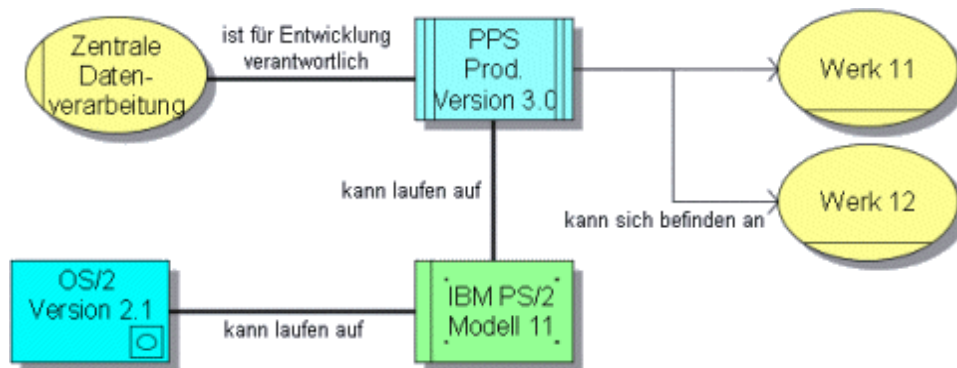


Abbildung 135: Zugriffsdiagramm (Auszug)

4.4.2.2 Programmablaufdiagramm

Im Zugriffsdiagramm können für die im Anwendungssystemtypdiagramm spezifizierten Anwendungssystemtypen, Modultypen und DV-Funktionstypen die Beziehungen zu den Objekttypen der Organisationssicht und Datensicht hergestellt werden (vgl. Kapitel **Zugriffsdiagramm** (Seite 143)). Die direkte Darstellung der Zuordnung von Funktionen des Fachkonzeptes ist in diesem Modelltyp nicht möglich. Diese Zuordnung erfolgt im Anwendungssystemtypdiagramm. Ebenfalls nicht direkt darstellbar sind die möglichen zeitlich-logischen Ablauffolgen der Anwendungssystemtypen, Modultypen und DV-Funktionstypen. Der ARIS-Architektur streng folgend, sind diese Verknüpfungen nur mit Hilfe der Navigation über verschiedene Modelltypen hinweg nachvollziehbar.

Gerade im Umfeld des Systemdesigns haben sich jedoch Modelltypen etabliert (z. B. Programmablaufpläne (PA), siehe Seite 1), die eine integrierte Sicht auf alle Aspekte des Systemdesigns zulassen.

Aus diesem Grunde wird in **ARIS** der Modelltyp **Programmablaufdiagramm** angeboten, in dem ungeachtet der Sichtenzerlegung von **ARIS** alle Beziehungen zu Anwendungssystemtypen, Modultypen und DV-Funktionstypen, die in den anderen Modelltypen von **ARIS** zur Verfügung gestellt werden, modellierbar sind. Daneben können auch die zeitlich-logischen Ablauffolgen der genannten Objekttypen dargestellt werden. Hierzu werden auch Ereignisse in diesem Modelltyp zur Verfügung gestellt. Ähnlich der Zuordnung von Funktionen und Ereignissen in der EPK können hiermit im Programmablaufdiagramm Modulfolgen definiert werden. Das Ereignis ist hierbei im Sinne eines Triggers zu verstehen, das Modultypen oder Anwendungssystemtypen anstößt. Verzweigungen können mit den aus der EPK bekannten Regeln dargestellt werden. Im Gegensatz zur EPK können im Programmablaufdiagramm jedoch auch Ablauffolgen ohne das Zwischenschalten von Ereignissen definiert werden.

4.4.2.3 Programmablaufplan (PA)

Der Programmablaufplan (PA) dient zur Darstellung von Verarbeitungsfolgen in einem Programm. Die Beziehungen zwischen den Objekten zeigen dabei die Reihenfolgen der Verarbeitung auf. Daten werden durch dieses Diagramm nicht wiedergegeben.

Die folgende Abbildung zeigt ein vereinfachtes Beispiel für die Verarbeitungsfolgen eines Geldautomaten. In der Darstellung der Verarbeitungsfolgen wird eine starke Orientierung an die Implementierung sichtbar.

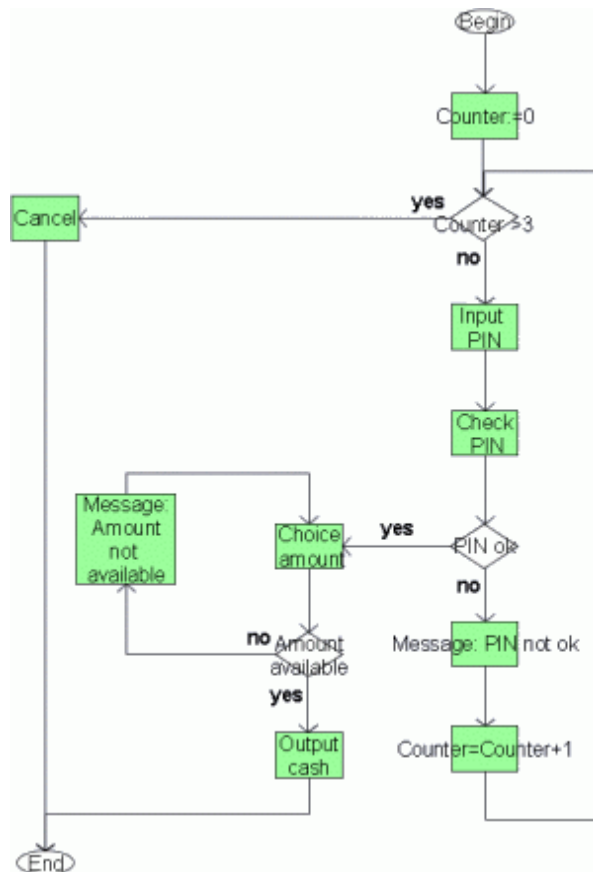


Abbildung 136: Beispiel eines Programmablaufplans (PA)

4.4.2.4 Maskendiagramm

Anwendungsgebiet des Maskendiagramms ist die Beschreibung von Bildschirmmasken bei der Software-Entwicklung. Ziel ist dabei eine automatische Ableitung der Bildschirmmasken aus dem Maskendiagramm.

Folglich bildet das Maskendiagramm den Aufbau und in gewissem Grad die Funktionalität von Bildschirmmasken ab. Es gilt dabei: der Aufbau des Maskendiagramms von links nach rechts und von oben nach unten entspricht der Geometrie der beschriebenen Oberfläche.

Zentrales Symbol ist die Maske; es stellt entsprechend der Windows-Terminologie ein Fenster dar. Dieses Fenster kann mehrere Registerkarten aufweisen (Symbol **Seite**). Generell kann die Oberfläche mittels einer Tabellenform geographisch in Bereiche eingeteilt werden (Symbol **Sektion** für eine Zeile und Symbol **Spalte** für eine Spalte). Die Symbole **Sektion** und **Spalte** sind zur Bildung komplexer Oberflächen beliebig schachtelbar. Auf der Oberfläche können Tabellen (Symbol **Maskentabelle**), Texteingabefelder (Symbol **COT-Attribut**) sowie Grafiken (Symbol **Bitmap**) und Erläuterungstext (Symbol **Text**) platziert werden. Mit dem Symbol **Layout** können den Objekten **Maske**, **Seite**, **Sektion**, **Spalte**, **Maskentabelle**, **COT-Attribut** und **Text** Darstellungseigenschaften zugewiesen werden.

Weitere Symbole können für die Beschreibung der Bildschirmoberfläche verwendet werden.

In folgender Abbildung ist beispielhaft ein Maskendiagramm abgebildet. Die zweite Abbildung zeigt die daraus abgeleitete Bildschirmmaske.

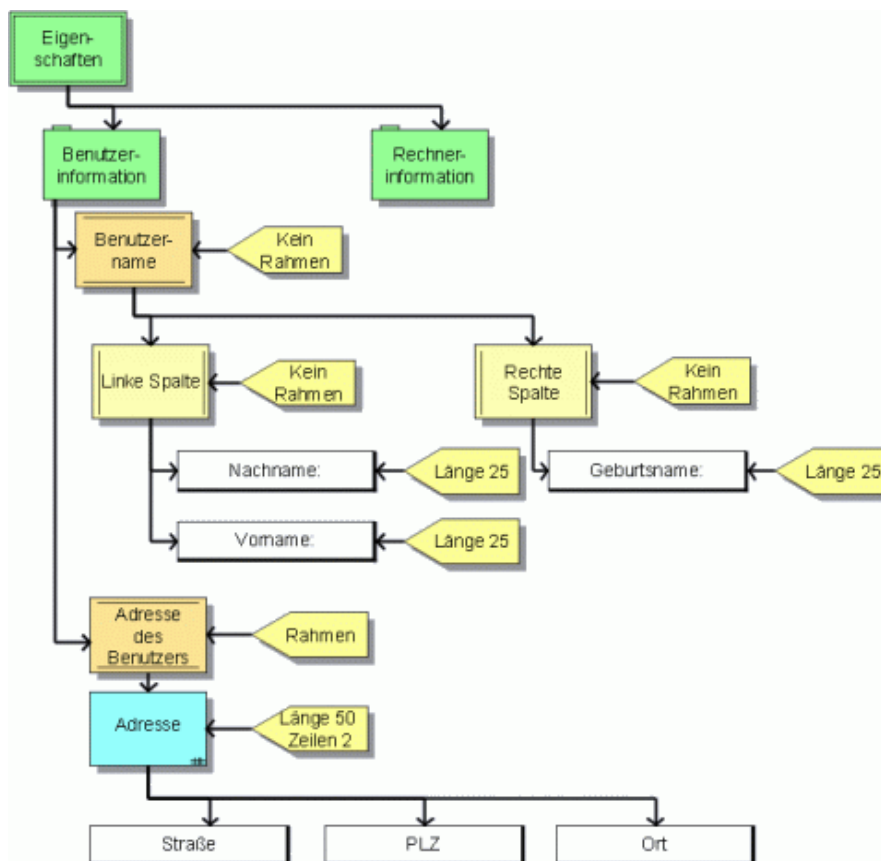
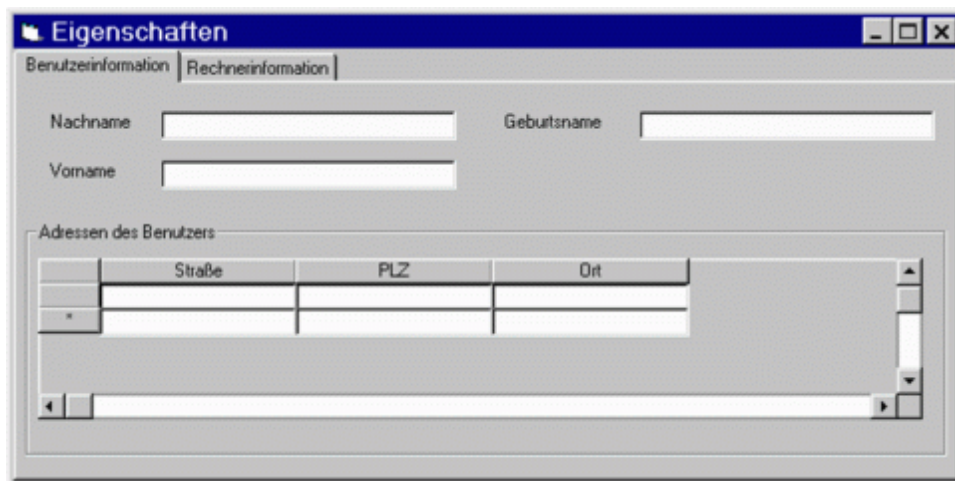


Abbildung 137: Beispiel eines Maskendiagramms



	Straße	PLZ	Ort
*			

Abbildung 138: Aus dem Maskendiagramm der vorherigen Abbildung abgeleitete Bildschirmmaske

4.4.2.5 SAP Integrationsprozess (XI)

In einem Modell des Typs **SAP Integrationsprozess (XI)** werden Geschäftsprozesse aus der SAP-Exchange Infrastructure dargestellt.

Ein solcher Geschäftsprozess beschreibt im Detail die Schritte, die durchgeführt werden, wenn Nachrichten zwischen Applikationen ausgetauscht werden.

Jeder Integrationsprozess hat genau ein Startobjekt und ein Prozessende.

Im Verlauf des Integrationsprozesses können folgende Konstrukte angelegt werden:

- Empfangen
- Senden
- Warten
- Blockbegrenzung (Anfang)
- Blockbegrenzung (Ende)
- Parallelität (Anfang)
- Parallelität (Ende)
- Steuerung
- Empfängerermittlung
- Zuweisung
- Transformation
- Switch (Anfang)
- Switch (Ende)
- Leer
- Schleifenbegrenzung (Anfang)
- Schleifenbegrenzung (Ende)

Verzweigt der Pfad nach einem Blockanfang, dem Anfang einer Parallelität, einem Switch- oder Schleifenanfang in mehrere Pfade, müssen diese in einem entsprechenden Endesymbol zusammengeführt werden.

4.4.3 Implementierung - Zugriffsdiagramm (phys.)

Die im DV-Konzept der Steuerungssicht behandelten Fragestellungen sind auch Gegenstand der Implementierungsebene. Im Unterschied zur DV-Konzept-Ebene werden hier jedoch nicht die Objekttypen betrachtet, sondern die konkreten Exemplare der einzelnen Objekte. So werden z. B. die Beziehungen von konkreten Anwendungssystemen zu Organisationseinheiten betrachtet und nicht die Beziehungen von Anwendungssystemtypen zu Organisationseinheiten.

Die im Folgenden dargestellten Beziehungen werden im Zugriffsdiagramm (physikalisch) modelliert.

4.4.3.1 Funktionen mit Daten

Zur Beantwortung der Frage, welche Daten zwischen Anwendungssystemen fließen, können Informationsflussobjekte zwischen den Anwendungssystemobjekten der Funktionssicht angelegt werden. Im Unterschied zur DV-Konzept-Ebene handelt es sich bei diesen Anwendungssystemobjekten jedoch nicht um Anwendungssystemtypen, sondern um konkrete Exemplare (einzelne Lizenzen). Es können also Anwendungssysteme, Module und Programmteiltyper über Datenflusskanten miteinander verbunden werden. Wurde auf der DV-Konzept-Ebene definiert, dass der Modultyp **Vertriebssystem SD Version 2.1** Daten mit dem Modultyp **Materialwirtschaftssystem MM Version 1.2** austauschen kann, wird auf der Implementierungsebene dargestellt, dass das konkret installierte Modul **SD Lizenznummer 1234** Daten mit dem Modul **MM, Lizenznummer 2352** und mit dem Modul **MM, Lizenznummer 34234** austauscht. Beide Module MM sind vom Modultyp **Materialwirtschaftssystem MM Version 1.2**. Dieser Sachverhalt wird auch in folgender Abbildung dargestellt.

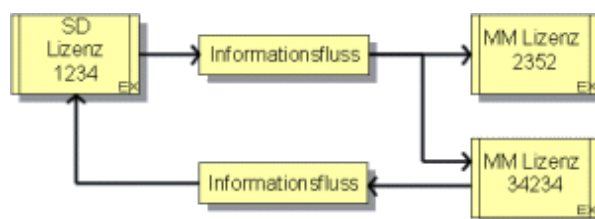


Abbildung 139: Datenfluss

Um die Datenobjekte, die zwischen den Systemen ausgetauscht werden, näher spezifizieren zu können, werden den Informationsflussobjekten entsprechende Modelltypen der Datensicht hinterlegt.

Neben den Datenflüssen zwischen Anwendungssystemen können auch für jedes Anwendungssystem die Input-/Output-Daten spezifiziert werden. Dabei gibt es zwei Gründe dafür, dass die Beziehungen im Zugriffsdiagramm (physikalisch) dargestellt werden. Im ersten Fall handelt es sich bei den Datenobjekten um Objekte des Tabellendiagramms (Tabelle, Feld, View (phys.)), welches sich in der Datensicht auf Implementierungsebene befindet. Diese Datenobjekte können über Input-/Outputbeziehungen mit Anwendungssystemobjekten der DV-Konzept-Ebene oder der Implementierungsebene verbunden werden. Im zweiten Fall handelt es sich bei den An-

wendungssystemobjekten um die konkreten Anwendungssysteme oder Module der Implementierungsebene, die mit den Objekten der Datensicht verbunden werden.

Als allgemeine Regel kann somit definiert werden:

Stammt einer der Objekttypen bei Input-/Outputbeziehungen aus der Implementierungsebene der entsprechenden Sicht, werden auch die Beziehungen in der Steuerungssicht auf Implementierungsebene (Zugriffsdiagramm (phys.)) dargestellt.

Ein Beispiel hierzu zeigt diese Abbildung.

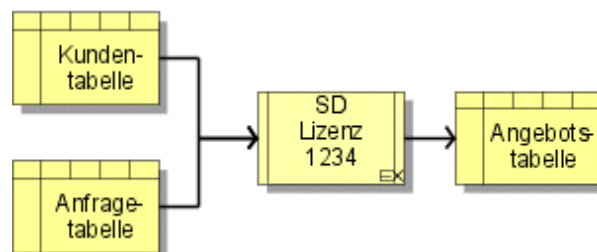


Abbildung 140: Input-/Outputbeziehungen

4.4.3.2 Organisation mit Daten

Im Vordergrund stehen hier die gleichen Fragestellungen, die auch im DV-Konzept betrachtet wurden:

- Welche organisatorischen Einheiten sind für Datenobjekte verantwortlich?
- Wer darf auf welche Datenobjekte zugreifen?
- Auf welchen Hardwarekomponenten sind Datenobjekte gespeichert?

Im Gegensatz zum DV-Konzept werden hier die Beziehungen jedoch zu den Datenobjekten hergestellt, die auf der Implementierungsebene der Datensicht dargestellt werden.

Die Verantwortlichkeit für Datenobjekte wird also nicht mehr für Relationen und Attribute des Relationendiagramms, sondern für die physischen Strukturen, also Tabellen, Felder und deren Exemplare (Tabelle (Exemplar), Feld (Exemplar)) definiert.

Zur Darstellung dieser Abhängigkeiten werden im Zugriffsdiagramm (phys.) Kanten zwischen den Objekten der Organisationssicht (Organisationseinheit, Stelle, Person, ...) und den bereits erwähnten Objekten des Tabellendiagramms (Tabelle, Feld, View [phys.], ...) angelegt.

Beim Anlegen der Kante zwischen den organisatorischen Einheiten und den Tabellen und Feldern muss dann jeweils die Bedeutung der Beziehung definiert werden. **Ist verantwortlich für** bedeutet, dass die organisatorische Einheit für den Inhalt der jeweiligen Tabelle oder des jeweiligen Feldes verantwortlich ist; **greift zu auf** bedeutet, dass die Stelle oder Person die Zugriffsberechtigung auf die dargestellten Datenobjekte hat.

Neben der Definition der Zugriffsrechte und Verantwortlichkeiten kann nun mit dem Objekt Hardwarekomponente (Organisationssicht/Implementierung) definiert werden, auf welchen konkret vorhandenen Hardwarekomponenten - die z. B. über die Inventarisierungsnummer des Unternehmens eindeutig bestimmt werden - sich bestimmte Informationsobjekte des Unternehmens befinden. Hierzu können im Zugriffsdiagramm (phys.) mit dem Objekt **Hardwarekomponente** Informationsobjekte der Implementierungsebene (Tabellen, Felder...), der DV-Konzept-Ebene (Relationen, Attribute) oder auch der Fachkonzept-Ebene (Entitytypen, Datencluster, ...) verbunden werden.

Ein Beispiel hierzu zeigt folgende Abbildung.

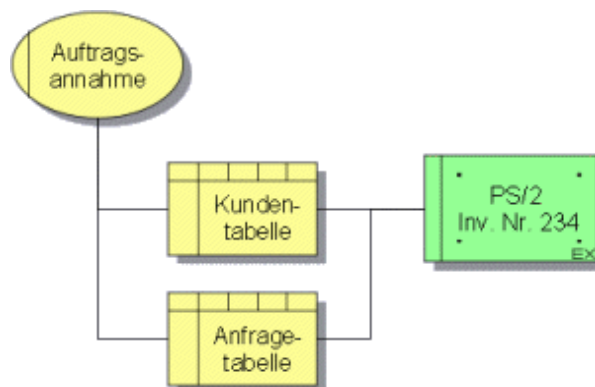


Abbildung 141: Zuordnungen zur Hardwarekomponente

4.4.3.3 Organisation mit Funktionen

Die im Zugriffsdiagramm (phys.) definierten Beziehungen zwischen den Objekten der Organisationssicht und der Funktionssicht erlauben die Beantwortung der folgenden Fragen:

Auf welchen Hardwarekomponenten laufen welche Anwendungssysteme und welche Anwendungssystemtypen können darauf laufen?

Zur Darstellung dieser Abhängigkeiten können die Beziehungen **ist Plattform von** und **kann Plattform sein von** zwischen den Anwendungssystemobjekten der Implementierungsebene (Anwendungssystem, Modul, Programmteil, ...) oder der DV-Konzept-Ebene (Anwendungssystemtyp, Modultyp, ...) und dem Objekttyp **Hardwarekomponente** aus der Organisationssicht modelliert werden.

Ein Beispiel zeigt diese Abbildung.

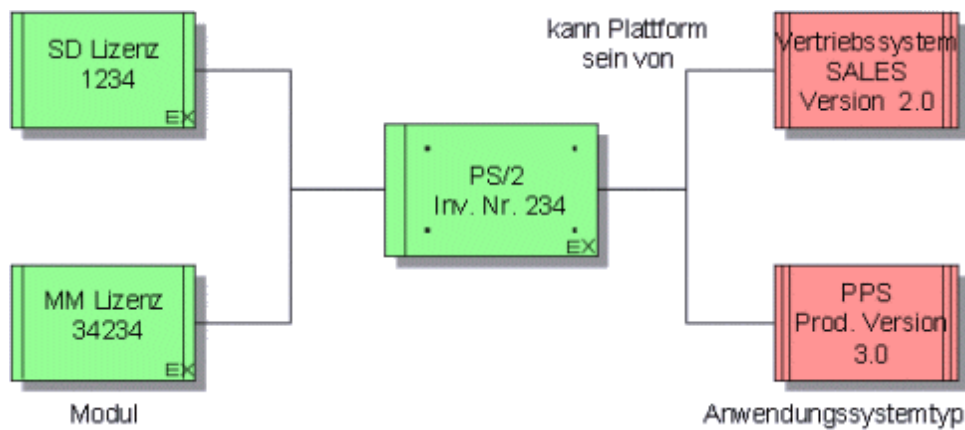


Abbildung 142: Hardwarekomponente als Plattform

Welche organisatorische Einheit ist Anwender eines konkreten Anwendungssystems?

Werden auf DV-Konzept-Ebene die Anwender definiert, die auf Anwendungssystemtypen zugreifen, kann nun auf Implementierungsebene diese Beziehung für konkrete Anwendungssysteme (einzelne Lizenzen) definiert werden. So können bspw. von einer Version des Anwendungssystemtyps ARIS Toolset mehrere Lizenzen in unterschiedlicher Konfigurierung im Unternehmen zur Verfügung stehen. Im Zugriffsdiagramm (phys.) kann nun dargestellt werden, welcher Anwender mit welcher Lizenz arbeitet. Hierzu können die Objekttypen **Organisationseinheit**, **Stelle** und **Person** über die Kante **ist Anwender** mit den Objekttypen **Anwendungssystem** und **Modul** verbunden werden. Ein Beispiel hierzu zeigt folgende Abbildung.

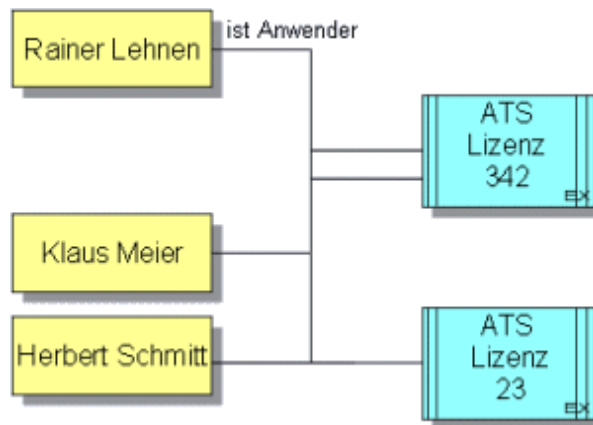


Abbildung 143: Anwender und Anwendungssystem

An welchen Standorten des Unternehmens sind Anwendungssysteme installiert?

Im DV-Konzept konnte über die Beziehung **Anwendungssystemtyp - Standort** definiert werden, welche Anwendungssystemtypen sich an bestimmten Standorten des Unternehmens befinden können. Um nun auch konkret definieren zu können, an welchen Standorten des Unternehmens sich die einzelnen Lizenzen, die von einem Anwendungssystemtyp erworben wurden, eingesetzt werden, bietet das Zugriffsdiagramm (phys.) die Möglichkeit, Standorte auch mit den Objekttypen **Anwendungssystem**, **Modul** und **DV-Funktion** zu verbinden.

Ein Beispiel hierzu zeigt die nächste Abbildung.

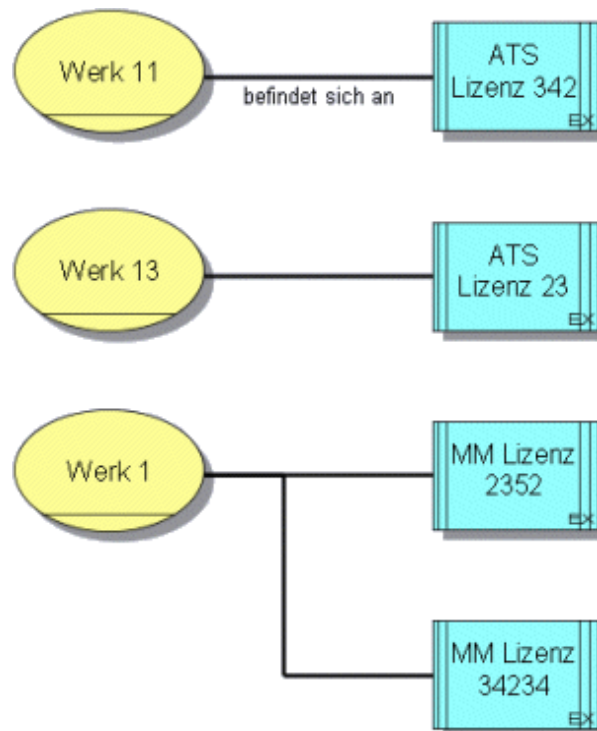


Abbildung 144: Standortzuordnungen

Im Anhang sind alle Beziehungen zusammengefasst, die im Zugriffsdiagramm (phys.) zur Verfügung stehen.

4.5 Leistungsmodellierung

ARIS stellt zur Beschreibung von Leistungen in einer Unternehmung verschiedene Modelltypen zur Verfügung.

Eine Leistung wird im Verlauf eines Wertschöpfungsprozesses erbracht. Sie ist das Ergebnis einer menschlichen Handlung oder eines technischen Vorgangs. Eine Leistung kann eine Dienstleistung oder eine Sachleistung sein.

Sachleistungen können einem Verbrauchsprodukt, Materialtyp, Betriebsmitteltyp, technischem Hilfsmitteltyp oder einem Packmitteltyp entsprechen. Auslöser zur Erstellung einer Leistung ist immer der Bedarf einer Organisationseinheit oder eines Kunden. Eine Sachleistung wird dem Kunden in Form eines materiell existierenden Guts angeboten.

Eine Dienstleistung ist ein immaterielles Gut, für das Produktion und Verbrauch zeitlich zusammenfallen.

Anbieter von reinen Dienstleistungen sind z. B. Banken, Versicherungen oder die öffentliche Verwaltung.

Je stärker die Kundenorientierung im Marktsegment eines Anbieters von Sachleistungen betont wird, desto wichtiger es für ihn die Dienstleistung im Produktumfeld genauer zu betrachten und weiterzuentwickeln.

Die verschiedenen Modelltypen von ARIS sind daher sowohl für die Beschreibung reiner Sach- oder Dienstleistungen als auch für die Beschreibung kombinierter Leistungen geeignet.

Folgende Modelltypen stehen Ihnen für die Leistungsmodellierung zur Verfügung:

- Leistungsaustauschdiagramm
- Leistungsbaum
- Produktzuordnungsdiagramm
- Produktbaum
- Produktauswahlmatrix
- Wettbewerbskräftemodell

4.5.1 Leistungsaustauschdiagramm

Das Leistungsaustauschdiagramm dient dazu, die Leistungserstellung sowie den Leistungsaustausch im Unternehmen abzubilden. Eine Leistung kann entweder eine Dienstleistung oder ein Produkt sein und wird durch entsprechende Symbole repräsentiert. Produkte können Materialtypen, Betriebsmitteltypen, technischen Hilfsmitteltypen und/oder Packmitteltypen entsprechen, die man z. B. aus der EPK (Materialfluss) kennt. Leistungen als Input und/oder Output von Funktionen können mit den Anfangs- und/oder Endereignissen dieser Funktionen in Verbindung gesetzt werden.

Dieser Leistungsaustausch zwischen betriebswirtschaftlichen Funktionen kann bereits auf einer Abstraktionsebene sinnvoll eingesetzt werden, die zwischen Wertschöpfungskettendiagramm und EPK liegt. Neben den Austauschbeziehungen aus funktionaler Sicht sind die Austauschbeziehungen von Leistungen aus organisatorischer Sicht abzubilden. Hierzu werden im Leistungsaustauschdiagramm ebenfalls Modellierungsmöglichkeiten geboten.

Die folgende Abbildung zeigt ein Beispiel für ein Leistungsaustauschdiagramm.

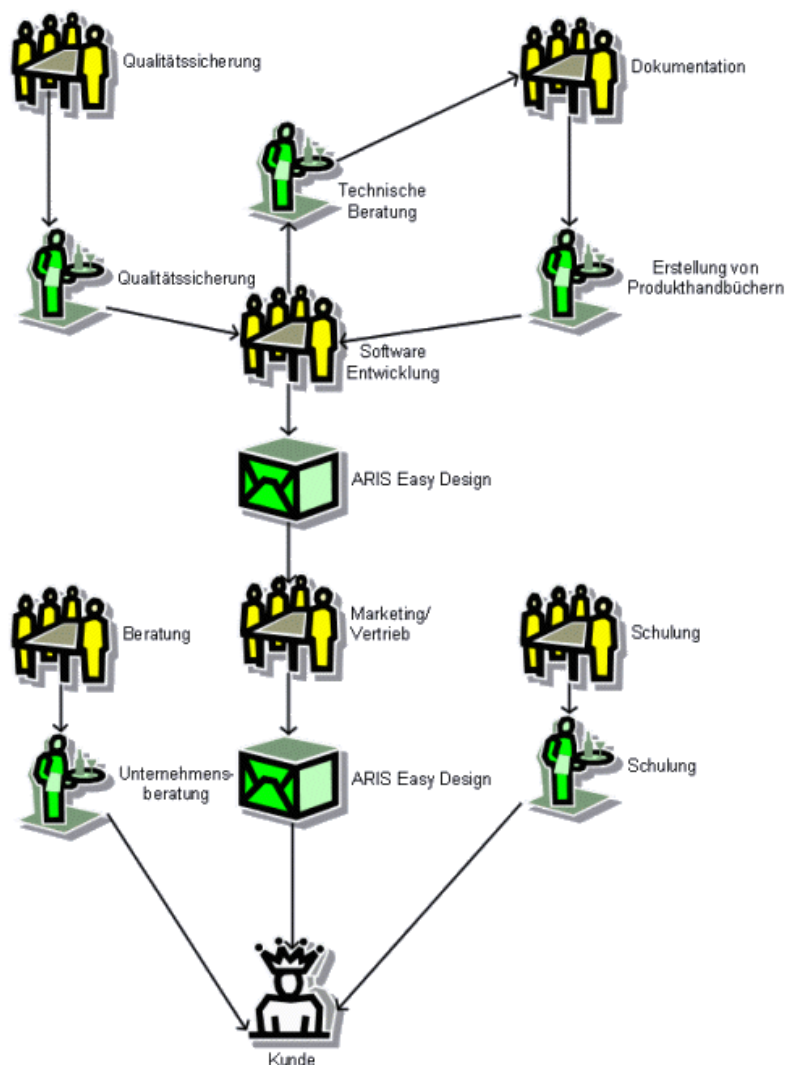


Abbildung 145: Exemplarischer Leistungsaustausch in einem Softwarehaus

4.5.2 Leistungsbaum

Leistungen können auf verschiedenen Abstraktionsebenen betrachtet werden. Dazu ist es sinnvoll, diese Beziehungen in einem Modell abzulegen, in dem dargestellt wird, aus welchen Teilleistungen sich eine Leistung zusammensetzt. Dieser statische Aspekt wird im Leistungsbaum abgebildet. So enthält beispielsweise ein komplexes Produkt häufig verschiedene Baugruppen, die wiederum aus diversen Einzelteilen bestehen. Jedes dieser Elemente kann als Leistung verstanden werden.

Mit Hilfe der Kante **steht in Beziehung mit**, die auch zwischen Leistungen im Leistungsbaum erlaubt ist, können Abhängigkeiten anderer Art beschrieben werden. Hierzu gehört z. B. die Beziehung zwischen Konsumentenkredit und Girokonto, über das die Zahlungen abgewickelt werden können.

Zudem können auch Substitutionsbeziehungen zu anderen Leistungen wie z. B. (potentiellen) Ersatzprodukten bzw. -dienstleistungen dargestellt werden.

In diesem statischen Modell werden auch die Beziehungen der Leistungen zu den (Unternehmens-) Zielen abgebildet.

Die folgende Abbildung zeigt beispielhaft einen Leistungsbaum.

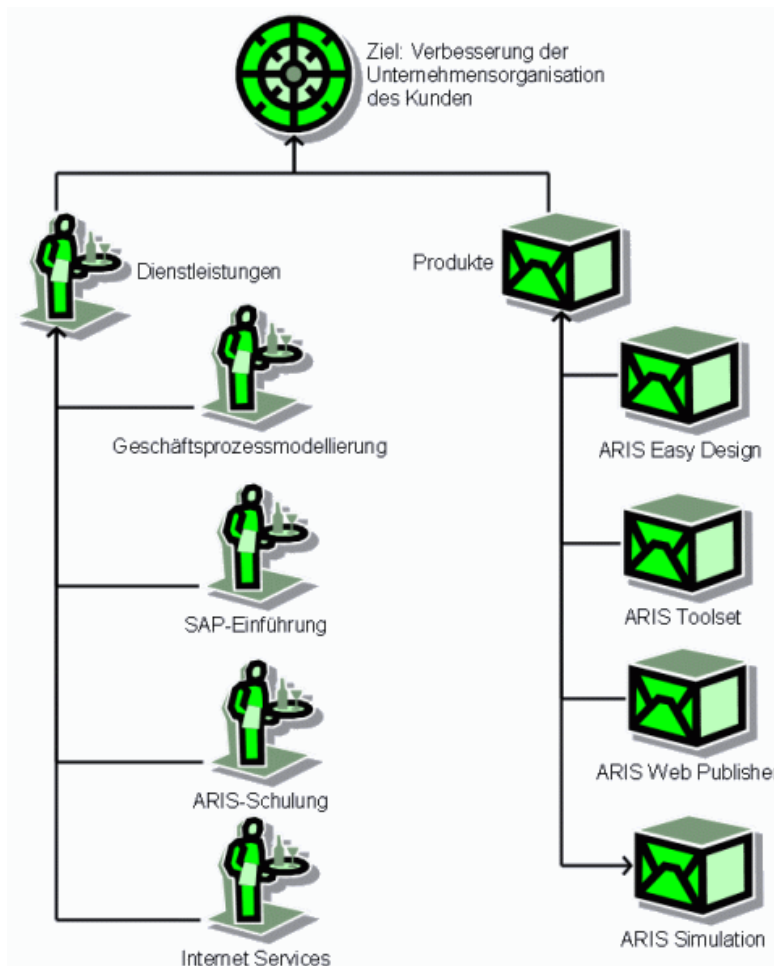


Abbildung 146: Leistungsbaum

4.5.3 Produktzuordnungsdiagramm

Neben den allgemeinen Leistungsdiagrammen, die zu den bildhaften Modellen gehören, bieten sich die Produktmodelle an, um eine eher abstrakte Darstellung zu ermöglichen. Das Produktzuordnungsdiagramm dient in erster Linie der Analyse der Produkterstellung in öffentlichen Verwaltungen. Ähnlich wie im Leistungsaustauschdiagramm kann mit diesem Modelltyp aufgezeigt werden, welche organisatorischen Einheiten Produkte bereitstellen oder nutzen und welche Funktionen zur Erstellung der Produkte benötigt werden oder für welche Funktionen die Produkte einen Input liefern. Ergänzend werden hier die (gesetzlichen) Auftragsgrundlagen der jeweiligen Produkte aufgezeigt. Daneben können auch die Ziele, die mit den Produkten erreicht werden sollen, dargestellt werden.

Die folgende Abbildung zeigt einen Ausschnitt aus einem Produktzuordnungsdiagramm der öffentlichen Verwaltung.

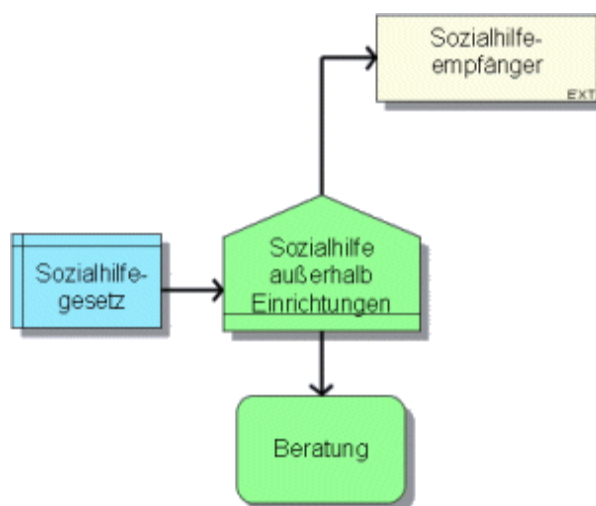


Abbildung 147: Beispiel eines Produktzuordnungsdiagramms

Der Modelltyp bietet aber auch die Möglichkeit Aspekte zu beschreiben, die mit der Vermarktung des Produktes in Zusammenhang stehen.

Diese Aspekte sollen am vereinfachten Beispiel von Bankprodukten verdeutlicht werden.

Mit der Verbreitung des Internets und der steigenden Anzahl von Internet-Anschlüssen in privaten Haushalten der letzten 10 Jahre stieg auch die Verbreitung des Online-Banking. Gleichzeitig stieg auch die Finanzkraft der Jugendlichen, ihre Bedeutung als Zielgruppe stieg.

Dies hat dazu geführt, dass die ursprüngliche Leistung **Girokonto** in unterschiedlichen Formen angeboten wird:

Es kann z. B. als Seniorenkonto angeboten werden, dessen Inhaber von den Mitarbeitern einer Geschäftsstelle betreut werden. Dieses Produkt wendet sich u. a. an ältere Kunden, die weniger mit neuen Technologien vertraut sind, Wert auf persönliche Betreuung und Beratung durch ihnen bekannte Personen legen und aufgrund altersbedingter Gebrechlichkeiten nicht mehr sehr mobil sind. Die für ein solches Konto anfallenden Gebühren dürfen über dem Durchschnitt liegen.

Das Girokonto kann aber auch als günstiges Teeny-Konto im Rahmen des Online-Banking angeboten werden. Dieses Produkt wendet sich an Jugendliche zwischen 12 und 20, die mit der Internet-Technologie vertraut sind, aber über eher geringe finanzielle Mittel verfügen. Die Gebühren müssen sich daher im unteren Bereich bewegen.

Folgende Grafiken zeigen Produktzuordnungsdiagramme für diese beiden Produktvarianten:

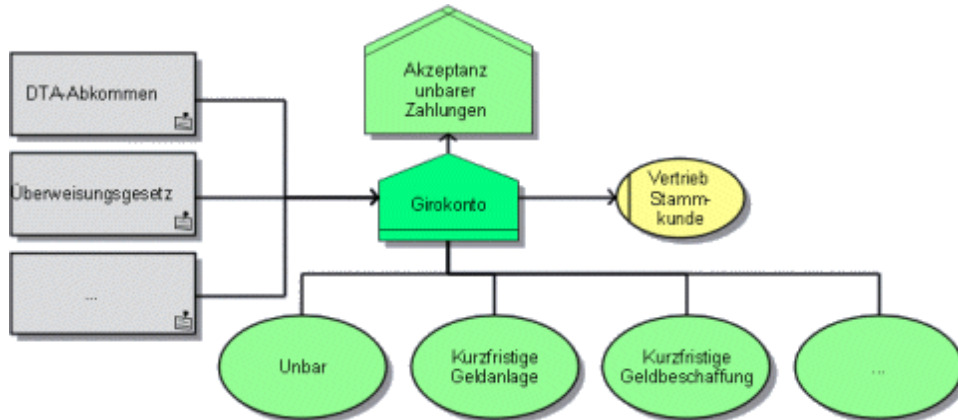


Abbildung 148: Produktzuordnungsdiagramm - Girokonto

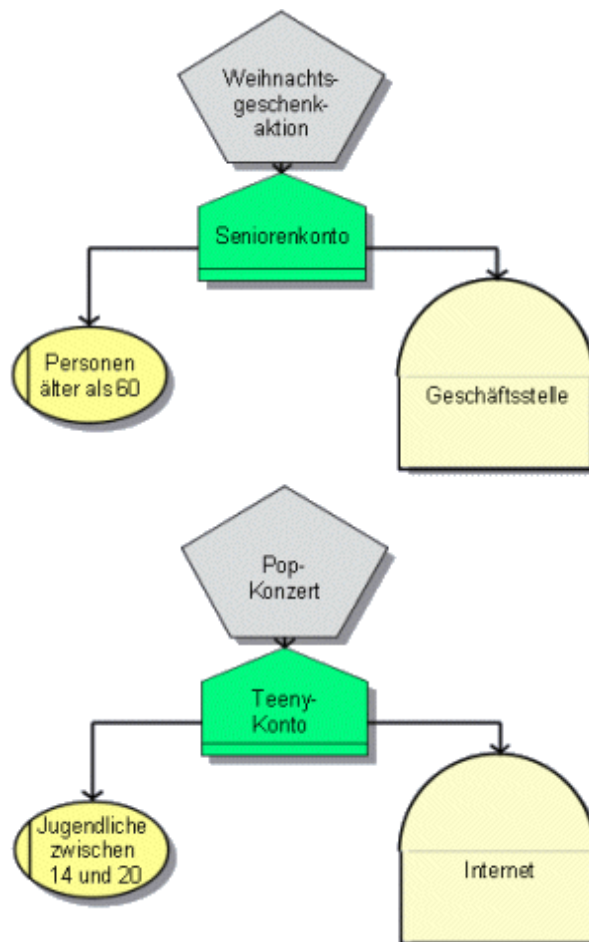


Abbildung 149: Produktzuordnungsdiagramme - Verkaufsprodukte

Die Leistungen **Teeny-** und **Seniorenkonto** wurden als Objektvarianten des Girokontos angelegt und durch das Attribut **Verkaufsprodukt** gekennzeichnet. Ein Verkaufsprodukt ist eine vom Unternehmen erstellte Leistung, die unter verschiedenen Namen in verschiedenen Marktsegmenten angeboten wird. Für die verschiedenen Verkaufsprodukte werden in der Regel jeweils unterschiedliche marketingpolitische Instrumente verwendet.

Mit Hilfe der Komponente ARIS Varianten können ausgehend von einem Produkt beliebig viele Verkaufsprodukte angelegt werden.

4.5.4 Produktbaum

Zweck des Produktbaumes ist es, die Zusammensetzung von Produkten in der öffentlichen Verwaltung zu analysieren. Dieses Modell entspricht im Wesentlichen dem Leistungsbaum, wobei auf die Möglichkeit der Modellierung von Ersatzprodukten verzichtet wurde. Der Produktbaum ist in der Fachkonzeptebene der Leistungssicht angesiedelt.

In folgender Abbildung wird ein Produktbaum beispielhaft dargestellt.

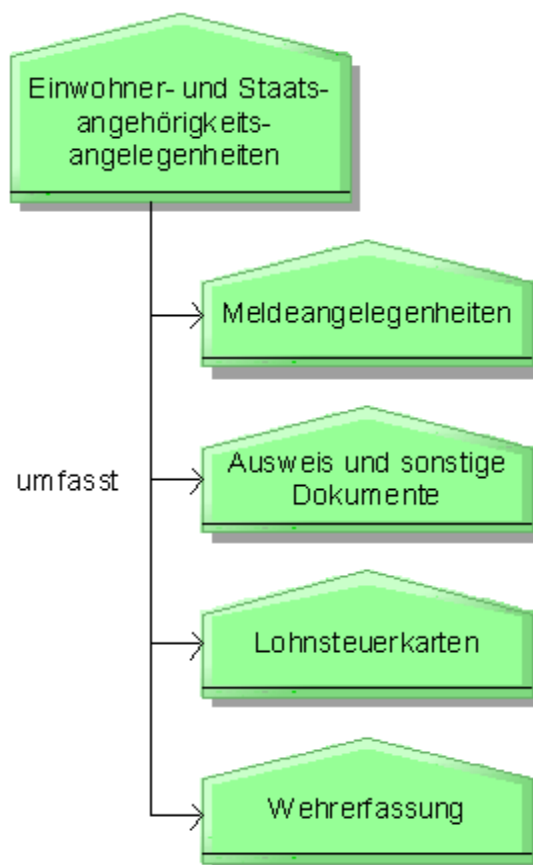


Abbildung 150: Klassifizierung der Produktgruppe "Einwohner- und Staatsangehörigkeitsangelegenheiten" mit Hilfe eines Produktbaumes

4.5.5 Produktauswahlmatrix

In der Produktauswahlmatrix steht eine Organisationseinheit mit den von ihr zu verantworteten Produkten im Vordergrund. Diesen Produkten können die für ihre Erstellung notwendigen Funktionen zugeordnet werden. Das Modell eignet sich als Einstiegspunkt, von dem zu Organigrammen, zu Produktbäumen und zu den für die Erstellung der Leistungen relevanten Prozessen navigiert werden kann. Ein Beispiel für eine Produktauswahlmatrix zeigt diese Abbildung.

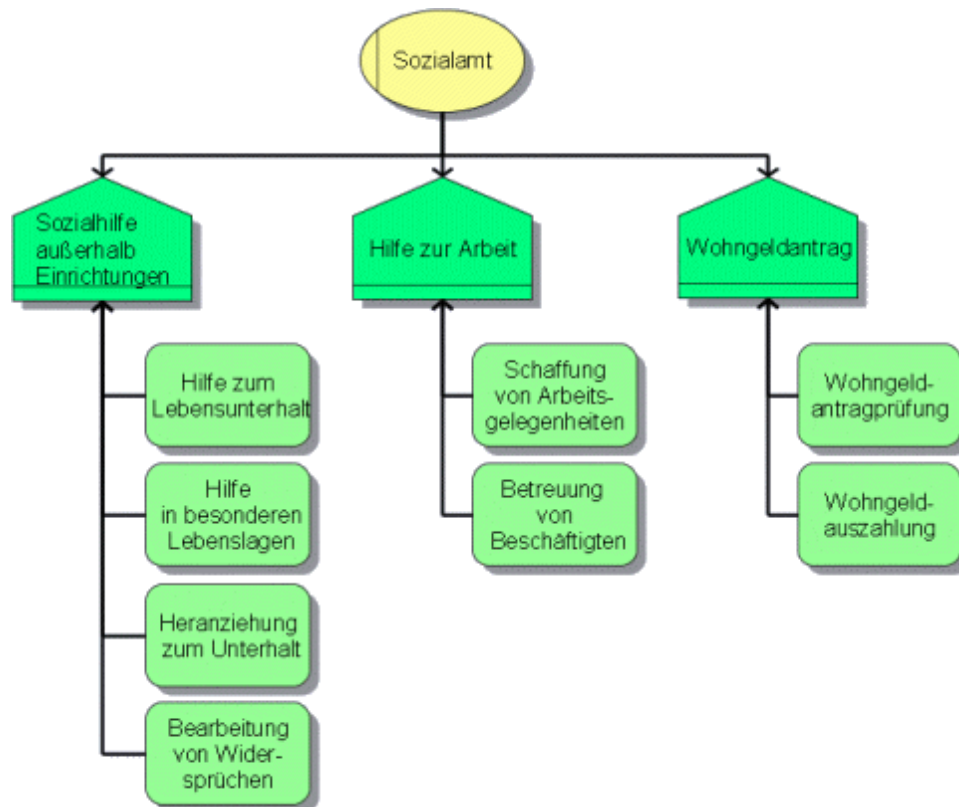


Abbildung 151: Produktauswahlmatrix des Sozialamts

4.5.6 Wettbewerbskräftemodell

Dieses Modell unterstützt die Analyse und Bewertung eines Wettbewerbsumfeldes, in dem ein Unternehmen konkurriert. Die Branchenstruktur beeinflusst in starkem Maß die Strategien, die dem Unternehmen potentiell zur Verfügung stehen.

In diesem Modell können die Beziehungen zwischen eigenem Unternehmen, den hergestellten Produkten und Dienstleistungen und den Marktpartnern dargestellt werden. Es kann dargestellt werden, welchen Kunden welche Leistungen in Anspruch nehmen, welche Leistungen von Lieferanten zur Verfügung gestellt werden, und welche Ersatzleistungen von (potentiellen) Wettbewerbern angeboten werden. Somit kann ein Ausschnitt der Wettbewerbssituation des Unternehmens dargestellt werden

Die folgende Abbildung zeigt ein Beispiel für ein Wettbewerbskräftemodell.

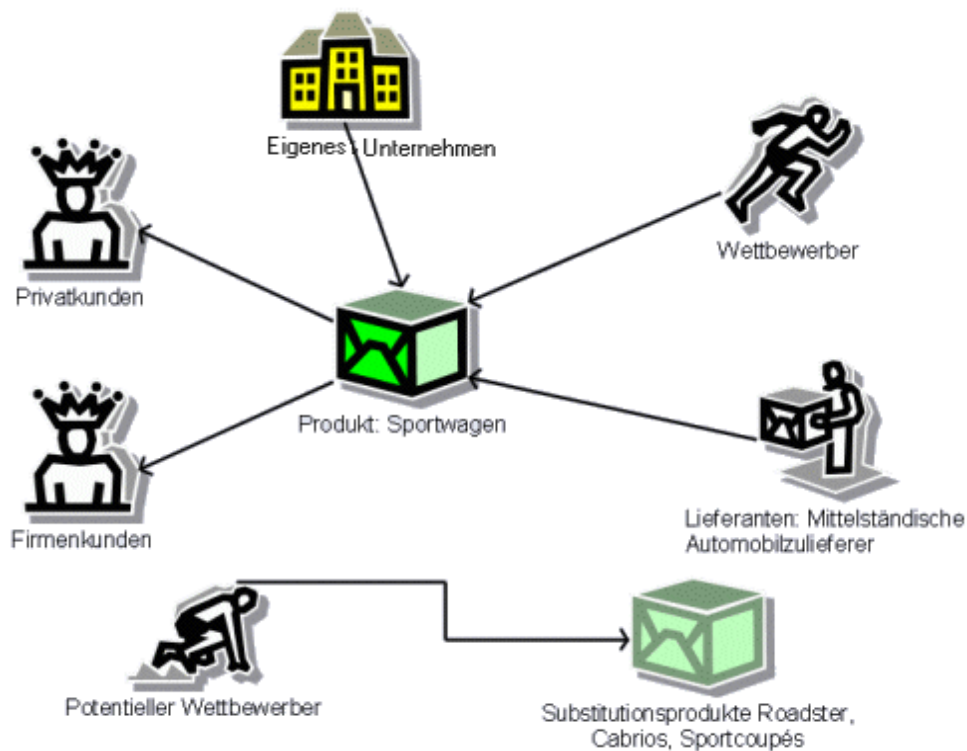


Abbildung 152: Wettbewerbskräfte im Sportwagenmarkt

5 Unified Modeling Language in ARIS

5.1 Einleitung

UML (Unified Modeling Language) ist eine objektorientierte Modellierungssprache, deren Sprachkonstrukte durch eine Arbeitsgruppe der OMG (Object Management Group) standardisiert werden. UML basiert auf den objektorientierten Ansätzen von OMT, Booch und OOSE.

5.2 ARIS UML Designer - Unterstützter UML-Standard

ARIS UML Designer 7.x unterstützt den kompletten UML-Standard 1.4.

Der UML-Standard 1.5 wird mit einem Abdeckungsgrad von ca. 90% unterstützt. Die UML-Spezifikation finden Sie im Dokument **Unified Modeling Language Specification, Version 1.4.2** unter <http://www.omg.org/cgi-bin/doc?formal/05-04-01.pdf>.

6 Verwendung der Modellierungsmethode Object Modeling Technique (OMT)

6.1 Einleitung

Innerhalb von ARIS wird als weitere Möglichkeit zur objektorientierten Modellierung die grafische Notation **Object Modeling Technique (OMT)** angeboten (vgl. Rumbaugh, J.; Blaha, M.; Premerlani, W.; Eddy, F.; Lorenzen, W.: Objektorientiertes Modellieren und Entwerfen. München et al. 1993). OMT ist in der Zwischenzeit gemeinsam mit den Methoden Booch und OOSE integriert und weiterentwickelt worden zu der Unified Modeling Language (UML), die ebenfalls in ARIS angeboten wird (siehe Kapitel **Unified Modeling Language in ARIS** (Seite 166)). OMT wird aus Kompatibilitätsgründen zu älteren Versionen von ARIS noch angeboten. Es wird jedoch empfohlen, statt OMT die Unified Modeling Language zu nutzen.

Im Folgenden werden die Bestandteile der OMT-Methodologie beschrieben. Darüber hinaus erhalten Sie Informationen darüber, wie die innerhalb von ARIS eingestellten Modellierungstechniken zu verwenden sind. Diese Beschreibung ersetzt nicht die ausführliche Beschreibung der Modellierungstechnik innerhalb der Originalliteratur. Sie hat im Wesentlichen zum Ziel, die Handhabung der Modellierungstechniken innerhalb von ARIS zu erläutern.

6.2 Zusammenfassung der OMT-Methodologie

Die Methodologie OMT wurde entworfen, um verschiedene Blickwinkel bei der Beschreibung eines Systems zu präsentieren. Dazu werden folgende Methoden verwendet:

- Object Modeling
- Dynamic Modeling
- Functional Modeling

Diese drei Sichten sind orthogonal zueinander, aber nicht völlig unabhängig zu sehen.

Object Modeling repräsentiert die statischen, strukturellen und datenbezogenen Aspekte eines Systems. Hier werden die Struktur von Objekten, ihre Beziehungen zu anderen Objekten und ihre Attribute und Beziehungen dargestellt.

Dynamic Modeling bildet die zeitlichen, verhaltensmäßigen, steuerungsbezogenen Aspekte eines Systems ab. Hier wird die Reihenfolge von Operationen durch die Abbildung von Ereignisfolgen beschrieben.

Functional Modeling verdeutlicht die Übergangs- und Funktionsaspekte eines Systems. Hier wird die Transformation von Werten beschrieben.

Diese Modelle enthalten untereinander Verweise auf die anderen Modelle: Das Object Model beschreibt beispielsweise Datenstrukturen, die im Dynamic Model und Functional Model verwendet werden. Die Prozesse im Functional Model entsprechen den Operationen im Object Model. Ein Zustandsdiagramm des Dynamic Model beschreibt ganz oder teilweise das Verhalten eines Objektes einer Klasse des Object Model.

6.3 Verwendung der OMT-Diagrammtechniken in ARIS

Im Folgenden ist aufgelistet, wie die in der OMT-Methodik vorgedachten Konstrukte in ARIS dargestellt, verwendet und miteinander in Beziehung gesetzt werden. Die für OMT definierten Modellierungskonstrukte (z. B. Class, Process, State) haben keine Überlappung zu anderen Modellierungskonstrukten innerhalb von ARIS (z. B. Ereignis, Funktion, Entitytyp usw.) und können somit nicht außerhalb der OMT-Modelle wieder verwendet werden. Die OMT-Modelltechniken sind somit als „selbstständige“ Methodik zu verstehen.

6.3.1 OMT Object Model

Darstellung von Instanzen

Im Rahmen objektorientierter Modellierung ist es normalerweise üblich, Objekte auf Typebene (d. h. Klassenebene) zu dokumentieren. Trotzdem kann es sinnvoll sein, auch einzelne Instanzen zu modellieren. ARIS bietet dafür mit dem blauen, abgerundeten Rechteck ein eigenes Symbol an.



Abbildung 153: Darstellung von Instanzen

Darstellung von Klassen

Klassen repräsentieren die wesentlichen Strukturen des zu modellierenden Anwendungsgebietes und werden in ARIS durch ein blaues Rechteck (mit horizontalen Linien) dargestellt.



Abbildung 154: Darstellung von Klassen

Zuordnung von Instanzen zu Klassen

Wenn die Darstellung von Instanzen für nötig gehalten wird, wird es möglich, die Zuordnung zu den zugehörigen Klassen deutlich zu machen. Die entsprechende Kante hat die Semantik **is instance of** wie folgende Abbildung zeigt.

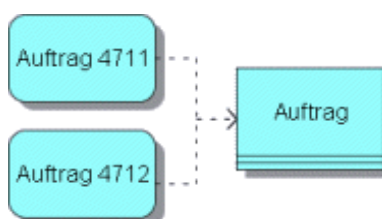


Abbildung 155: Verbindung von Instanzen zu Klassen

Zuordnung von Attributen zu Klassen

Die Eigenschaften von Klassen werden über Attribute beschrieben. Bei der Modellierung in ARIS sind dies eigene Objekte (und damit auch eigene Symbole), die über Kanten (mit der Semantik **has attribute**) mit den dazugehörigen Klassen verbunden werden (vgl. folgende Abbildung). Diese Trennung in zwei unterschiedliche Objekttypen (Klassen und Attribute) ist nötig, um alle Vorteile von ARIS bzgl. der Erstellung von Reporten und Navigation zu nutzen.

Bei jedem Attribut kann vermerkt werden, ob es sich dabei um ein Klassen- (Wert ist für alle Instanzen der Klasse relevant) oder Instanzenattribut handelt.

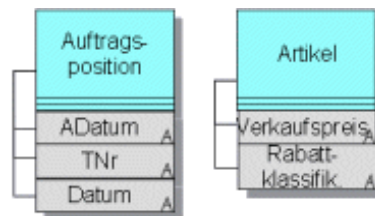


Abbildung 156: Zuordnung von Attributen zu Klassen

Zuordnung von Operationen zu Klassen

Die Klassen zugeordnete Funktionalität wird über die Definition von Operationen (Methoden) beschrieben. Auch hierfür wurde ein eigener Objekttyp definiert, der mit Klassen in Beziehung (mit der Semantik **has operation**) gesetzt werden kann (vgl. folgende Abbildung).

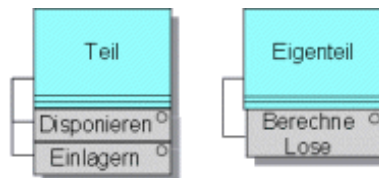


Abbildung 157: Zuordnung von Operationen zu Klassen

Assoziationen zwischen Instanzen

Zwischen einzelnen Instanzen können Verknüpfungen bestehen. Diese werden in ARIS durch eine ungerichtete Kante vom Typ **is linked to** dargestellt.

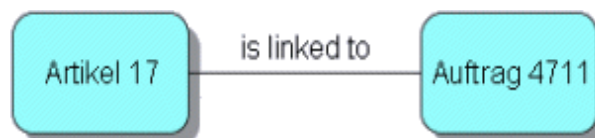


Abbildung 158: Assoziationen zwischen Instanzen

Assoziationen zwischen Klassen

Auch zwischen Klassen können Verknüpfungen (Assoziationen) bestehen, wie sie auch aus einem Entity-Relationship-Modell schon bekannt sind. Zu deren Darstellung wird ein eigenes Symbol (gelbe Raute) verwendet, um in gleicher Art und Weise auch mehrstellige Beziehungen darzustellen (siehe unten). Die Kanten werden dabei jeweils vom Klassensymbol zum Rautensymbol gezogen und können mit Komplexitätsgraden belegt werden, wozu das Attribut **Multiplicity** der Kante verwendet wird. Folgende Einträge für Multiplicity sind möglich und führen teilweise auch zu einer entsprechenden grafischen Darstellung der Kante:

- 1
- c
- cn
- n

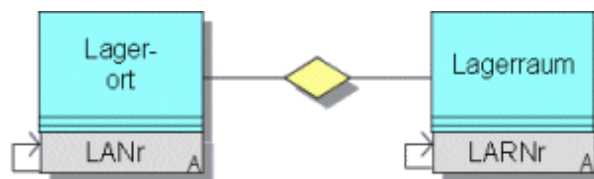


Abbildung 159: Assoziationen zwischen Klassen

Mehrstellige Assoziationen zwischen Klassen

Dreistellige (oder auch n-stellige) Assoziationen zwischen Klassen werden dargestellt, indem auch noch eine dritte Klasse oder noch mehr Klassen mit der die Verknüpfung definierenden Raute verbunden wird.

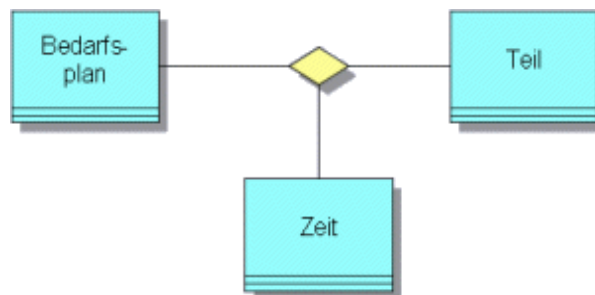


Abbildung 160: Ternäre Beziehung zwischen Klassen

Modellieren einer Assoziation als Klasse

Eine Assoziation kann wiederum als selbständiges Objekt verstanden und als Klasse interpretiert werden. Dies kann dargestellt werden, indem eine gerichtete Kante von dem Rautensymbol zu einem Klassensymbol gezogen wird, wo dann alle Attribute und Operationen aufgelistet werden können (vgl. nächste Abbildung). Diese „uminterpretierte“ Klasse kann dann natürlich wiederum Assoziationen zu anderen Klassen eingehen.

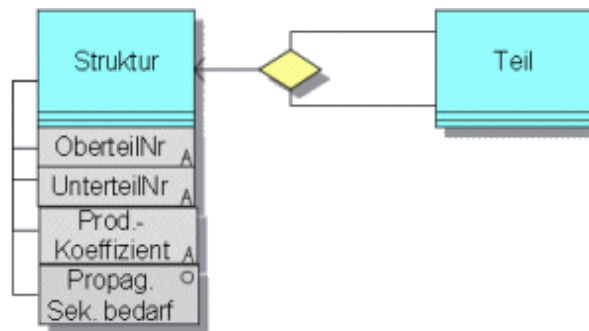


Abbildung 161: Modellieren einer Assoziation als Klasse

Darstellung einer qualifizierten Assoziation

Eine qualifizierte Assoziation ergänzt eine normale Assoziation um eine Qualifikationsangabe, die als ausgezeichnetes Attribut die Kardinalität einer Assoziation reduziert. Sinn macht dies für 1:m und n:m-Assoziationen, da damit die Objekte auf der m-Seite einer Assoziation unterschieden werden.

Die Darstellung einer qualifizierenden Assoziation erfolgt, indem an der Kante die Qualifikationsangabe vermerkt wird. Dazu ist ein eigenes Attribut **Qualifier** vorgesehen, das natürlich (wie alle Attribute) auch in die Grafik eingeblenet werden kann.



Abbildung 162: Darstellung einer qualifizierten Assoziation

Darstellung von Ordnungen bei Assoziationen

Falls die Objekte auf der n-Seite einer Assoziation eine Ordnung haben, kann dies explizit in der Grafik vermerkt werden. Hierfür existiert ein eigenes Attribut an der Kante zwischen Class und Assoziations-Symbol.



Abbildung 163: Darstellung von Ordnungen bei Assoziationen

Aggregation zwischen Klassen

Eine Aggregation stellt eine Teil-Ganzes-Beziehung dar und kann als Spezialfall einer Assoziation angesehen werden. Diese Beziehung wird als gerichtete Beziehung zwischen Klassen modelliert (der Kantenotyp ist dabei **aggregates**), wobei dann in der Grafik eine weiße Raute die Klasse, die das „Ganze“ repräsentiert (Komponentengruppe), angezeigt wird.



Abbildung 164: Aggregation zwischen Klassen

Generalisierung und Vererbung

Ein grundlegendes Konstrukt objektorientierter Modellierung ist die Definition von Hierarchien zwischen Klassen, womit auch eine Vererbung von Attributen und Operationen von der Ober- zu den Unterklassen verbunden ist. In ARIS gibt es hierfür einen eigenen Objekttyp (grünes Dreieck), der mit den beteiligten Klassen verbunden wird (vgl. nächste Abbildung). Auch Mehrfachvererbung ist damit darzustellen.

Bei dem Generalisierungsoperator kann als Attribut vermerkt werden, welcher Aspekt bei der Generalisierung/Spezialisierung verwendet wird und ob die Spezialisierung disjunkt oder nicht disjunkt ist.

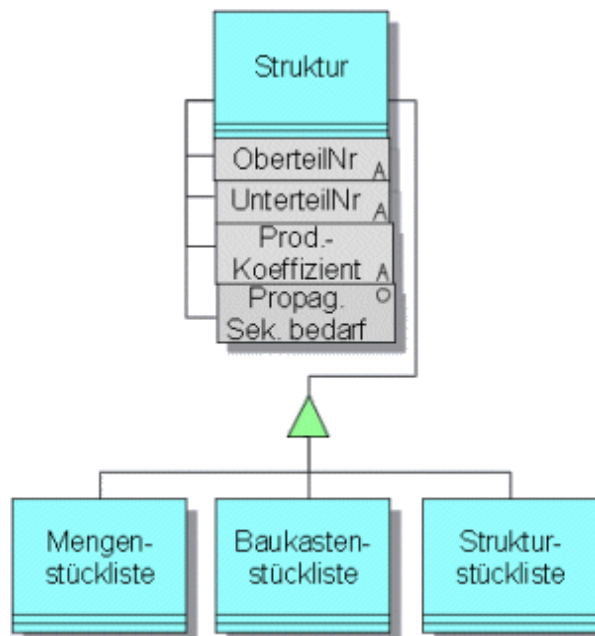


Abbildung 165: Generalisierung und Vererbung

Einschränkungen (Constraints) bei Klassen, Attributen und Assoziationen

Einschränkungen (Constraints) sind funktionale Beziehungen zwischen den Klassen, Attributen und Assoziationen eines OMT Object Model. In ARIS sind eigene Objekttypen (Punkt) für Constraints auf Attributen definiert. Dies zeigt das folgende Beispiel, das ausdrückt, dass sich das Verhältnis von Höhe zu Breite bei Fenstern zwischen 0,7 und 1,7 bewegt.

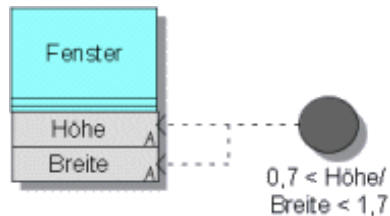


Abbildung 166: Darstellung von Constraints auf Attributen

Auch zwischen Assoziationen können Einschränkungen definiert werden. Die folgende Abbildung zeigt ein Beispiel, wo ausgedrückt werden soll, dass die Personenmenge, die den Vorsitz eines Ausschusses ausmacht, natürlich eine Teilmenge der gesamten Mitgliedschaft darstellt. Hierzu kann eine gerichtete Kante zwischen den Assoziationssymbolen gezogen werden.

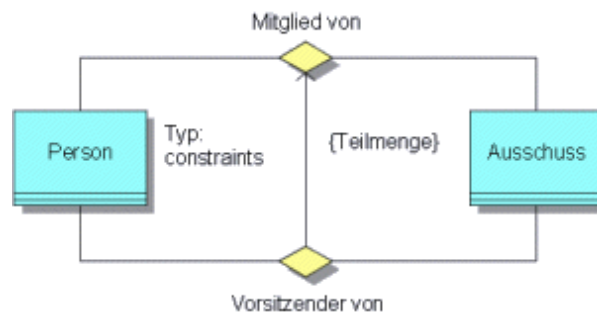


Abbildung 167: Darstellung der Constraints bei Assoziationen

Beispiel eines OMT Object Model

Die nächste Abbildung zeigt ein typisches Beispiel eines OMT Object Model mit den wesentlichen Modellierungskonstrukten.

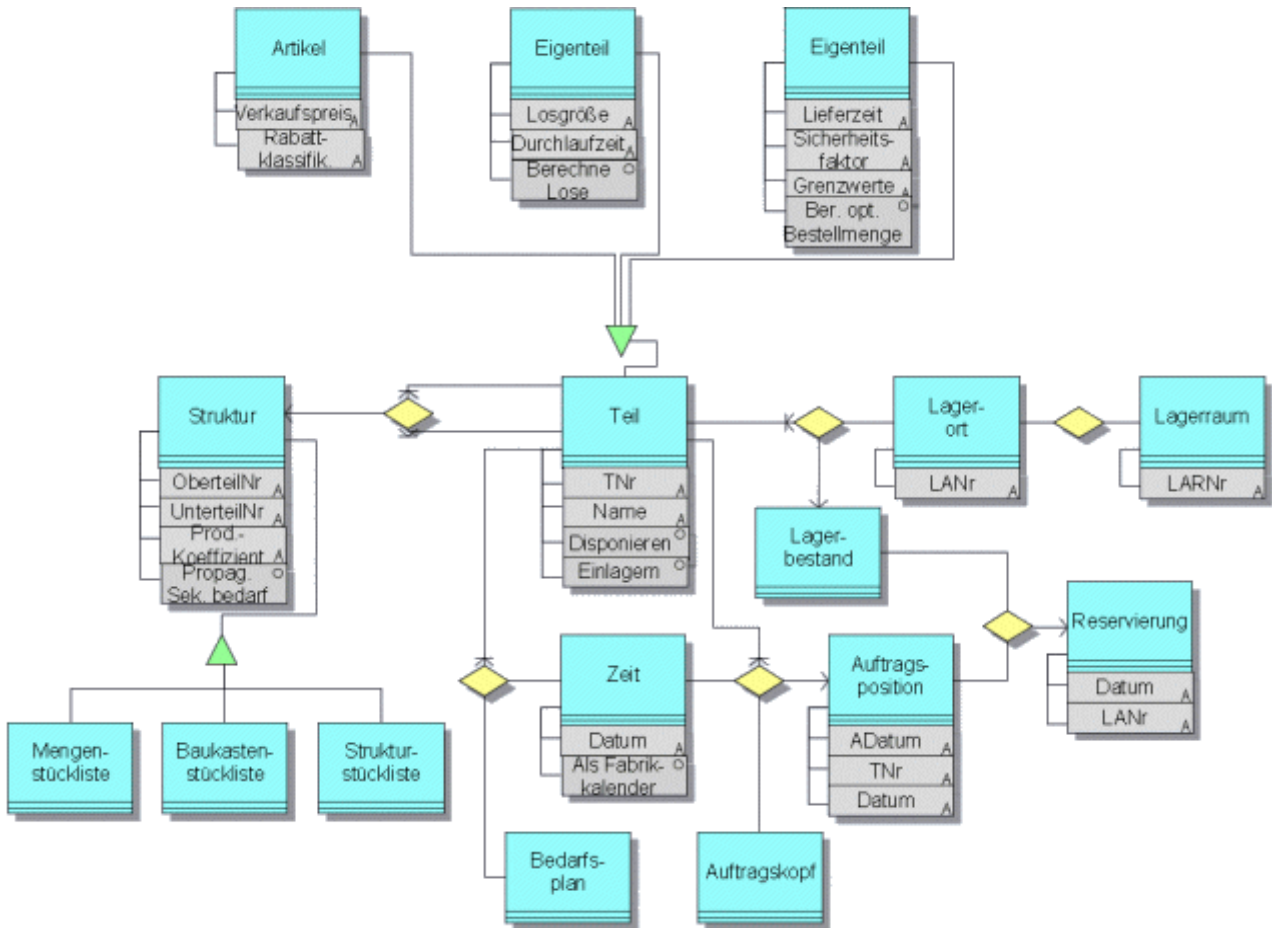


Abbildung 168: Beispiel eines OMT Object Model

6.3.2 OMT Dynamic Model

Das OMT Dynamic Model ist eine Menge von Zustandsübergangsdiagrammen, wobei i. a. ein Zustandsübergangsdiagramm das Verhalten einer Klasse beschreibt. Die Zustände sind dabei über gerichtete Kanten miteinander verbunden, die Ereignisse repräsentieren.

Darstellung von Initial-, Final- und Übergangszuständen

In ARIS gibt es drei Symbole, um Initial States, Final States und normale States zu unterscheiden.



Abbildung 169: Darstellung von Initial-, Final- und Übergangszuständen

Übergang zwischen Zuständen

Zustandsübergänge (Transitionen) werden durch Ereignisse ausgelöst; die Kante zwischen zwei Zuständen hat den Typ **has transition to**.

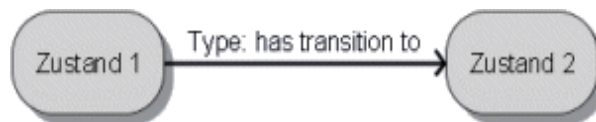


Abbildung 170: Darstellung des Übergangs zwischen Zuständen

Sowohl die Zustände, als auch die Übergänge können mit weiteren Informationen angereichert werden. Die Attribute **do/action**, **entry/action**, **exit/action** und **event/action** beim State können verwendet werden, um Eingangs-, Ausgangs- und interne Aktionen zu beschreiben. Auch an der Kante zwischen Zuständen kann die Bedingung für den Übergang detaillierter beschrieben werden.



Abbildung 171: Darstellung weiterer Informationen beim Zustandsübergang

Beispiel für ein OMT Dynamic Model

Die folgende Abbildung zeigt ein typisches Beispiel für ein OMT Dynamic Model.

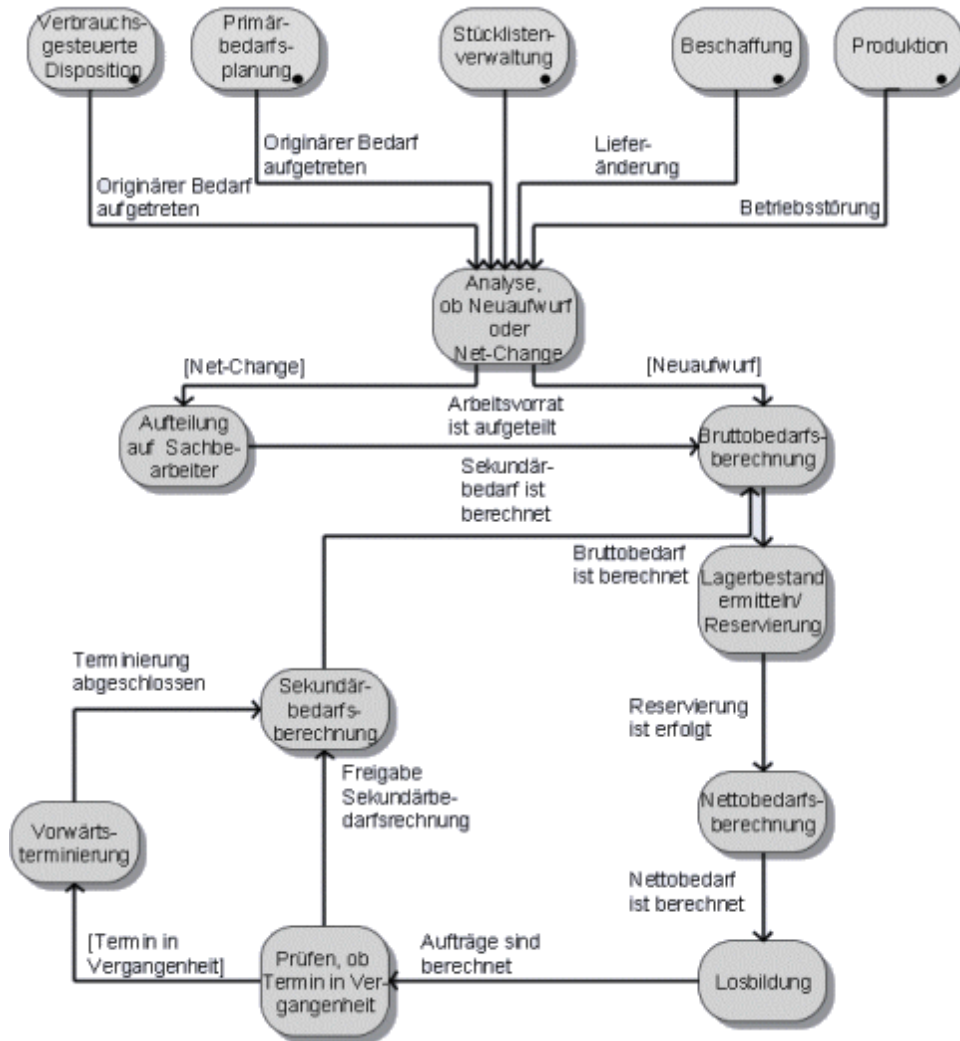


Abbildung 172: Beispiel eines OMT Dynamic Model

6.3.3 OMT Functional Model

Das OMT Functional Model zeigt mit Hilfe von Datenflussdiagrammen, wie bei einer Berechnung die Ausgabewerte aus den Eingabewerten abgeleitet werden.

Darstellung von DataStores (Datenspeichern)

DataStores werden verwendet, um Daten passiv zu speichern. In ARIS werden zu deren Darstellung 2 horizontale Linien als Symbol verwendet.



Abbildung 173: Darstellung von Datenspeichern

Darstellung von Processes (Prozessen)

Prozesse transformieren Daten und werden in ARIS als gelbe Ellipse dargestellt.



Abbildung 174: Darstellung von Prozessen

Darstellung von Actors (Handlungsobjekten)

Ein Actor ist ein Objekt, das den Datenfluss aktiviert, indem es Datenwerte erzeugt oder verbraucht. Sie können somit als Quellen und Senken des Graphen verstanden werden und werden durch ein Quadrat dargestellt.

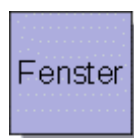


Abbildung 175: Darstellung von Actors (Handlungsobjekten)

Darstellung von Datenflüssen

Datenflüsse verbinden die Eingabe eines Prozesses oder Objektes mit der Eingabe eines anderen. Sie werden als Objekte des Typs **DataValue** zwischen Objekten modelliert und i. a. mit der Beschreibung der Daten beschriftet.



Abbildung 176: Darstellung von Datenflüssen

Splitten von Datenflüssen

Wenn ein Datenwert an unterschiedliche Stellen geschickt werden soll, können Datenflüsse gespalten werden. In ARIS werden zur Darstellung eines solchen Splittings eigene Symbole (Connection) verwendet.

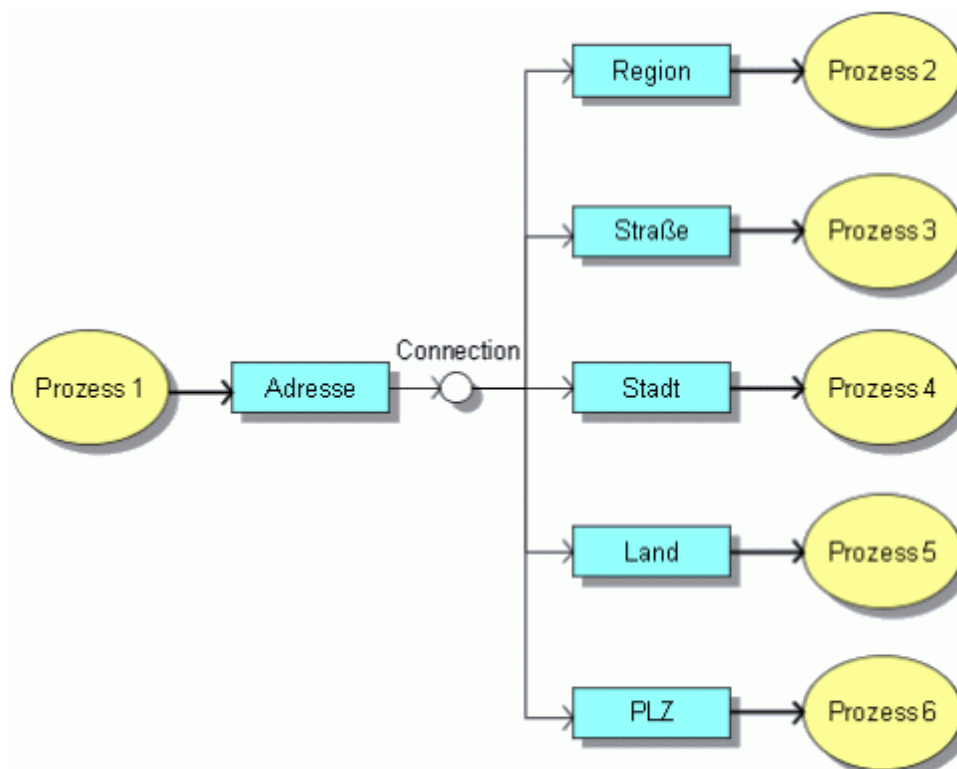


Abbildung 177: Darstellung des Splittings von Datenflüssen

Beispiel für ein OMT Functional Model

Diese Abbildung zeigt ein typisches Beispiel eines OMT Functional Model.

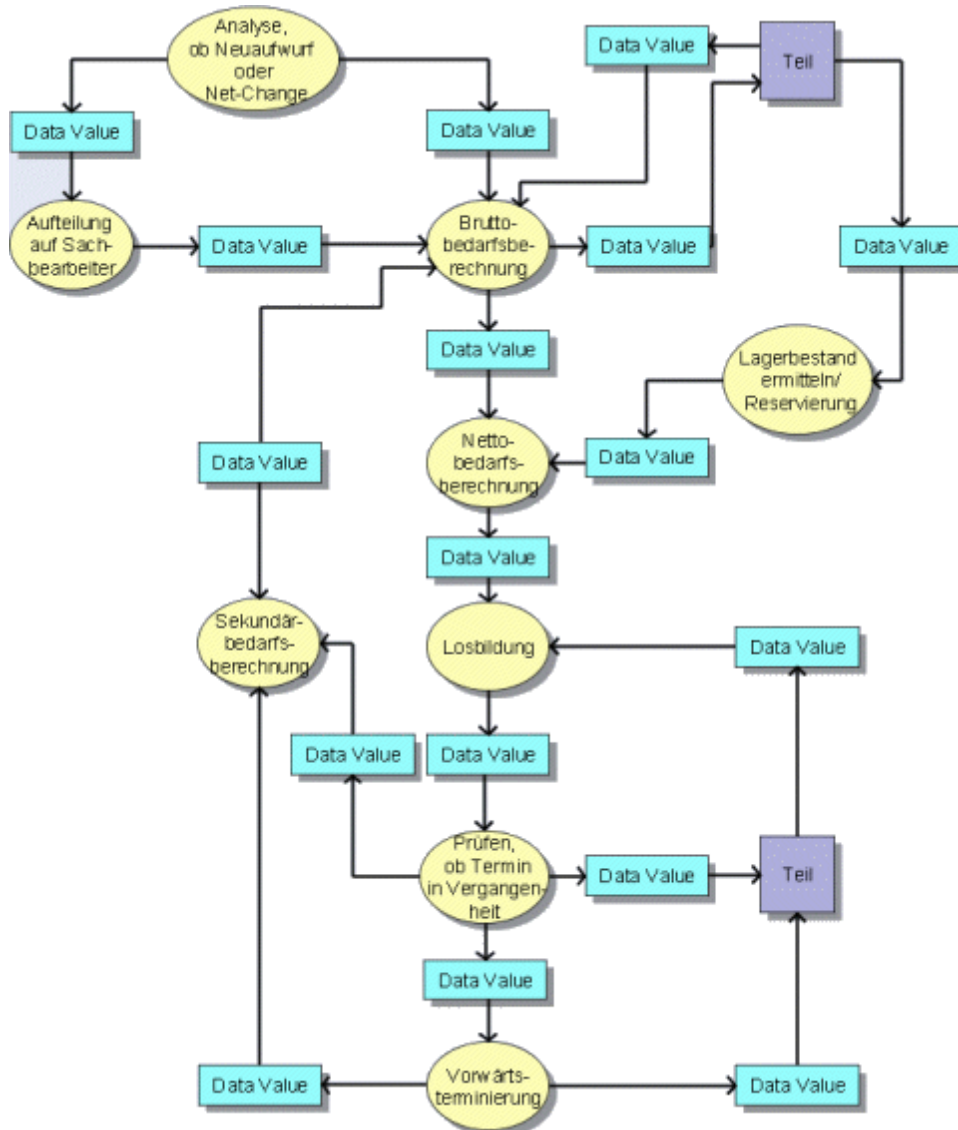


Abbildung 178: Beispiel eines OMT Functional Model

6.3.4 Mögliche Hierarchisierungen von Objekten

- Der Objekttyp **Class** kann durch ein OMT Dynamic Model hierarchisiert werden, um die Zustandsübergänge innerhalb dieser Klasse zu dokumentieren.
- Der Objekttyp **Class** kann durch ein OMT Class Description Model hierarchisiert werden, um Attribute und Operationen dieser Klasse zuzuordnen. Die möglichen Beziehungen in diesem Modell machen eine Teilmenge des OMT Object Model aus.
- Die Objekttypen **State**, **Final State** und **Initial State** können wiederum durch ein OMT Dynamic Model verfeinert werden, um eine Zustandsbeschreibung auf unterschiedlichen Hierarchieebenen zu ermöglichen.
- Der Objekttyp **Process** kann durch ein OMT Functional Model verfeinert werden, um Datenflussbeschreibungen auf unterschiedlichen Levels zu ermöglichen.
- Der Objekttyp **DataStore** kann durch ein OMT Object Model hierarchisiert werden.
- Die Kantentypen im OMT Functional Model können durch ein OMT Object Model und durch ein OMT DataValue Decomposition Modell verfeinert werden, um die Einzelheiten eines Datenflusses zu dokumentieren.

7 Methoden für das Wissensmanagement

7.1 Einleitung

Wissensmanagement bezweckt die systematische Bewirtschaftung der zunehmend wichtigen Unternehmensressource Wissen. Es umfasst die Entwicklung, Überwachung, Unterstützung und Weiterentwicklung von Strategien, Prozessen, Organisationsstrukturen und Technologien für die effektive Wissensverarbeitung im Unternehmen. Hierzu gehören alle Aktivitäten, die den Erwerb, die Aufbereitung, die Übertragung und die Nutzung von Wissen betreffen. Diese Aktivitäten der Wissensverarbeitung finden in der Regel nicht isoliert statt, sondern hauptsächlich in den operativen und dispositiven Geschäftsprozessen des Unternehmens. Es ist daher eine integrierte Betrachtung von Geschäftsprozessen, Wissensverarbeitung, Organisationsstrukturen, Informationssystemen usw. erforderlich.

Ein Großteil dieser Aspekte lässt sich mit bereits seit längerem existierenden ARIS-Methoden abbilden, z. B. EPKs, Organigrammen, Funktionszuordnungsdiagrammen, eERMs usw. Ist jedoch eine genaue Darstellung, Analyse und Verbesserung der Wissensverarbeitung gewünscht, sind zusätzliche Darstellungsmittel notwendig, z. B. zur Identifikation und inhaltlichen Strukturierung relevanter Wissenskategorien, zur Beschreibung der Verteilung von Wissen in der Organisation und zur Modellierung der Wissensentstehung und -nutzung in den Geschäftsprozessen.

Aus diesem Grunde wurden die zwei neuen Objekttypen **Wissenskategorie** und **Dokumentiertes Wissen** sowie die zwei neuen Modelltypen **Wissensstrukturdiagramm** und **Wissenslandkarte** aufgenommen. Außerdem wurden die Modelltypen zur Darstellung von Geschäftsprozessen (EPK, VKD, Office Process u. ä.) um Konstrukte für die Wissensentstehung und -nutzung erweitert. Die neuen Objekt- und Modelltypen sind methodisch voll in den wichtigsten sonstigen Modelltypen des Fachkonzeptes (wie z. B. eERM, Organigramm, Funktionsbaum) integriert, so dass eine durchgängige Betrachtung gewährleistet wird. Hierdurch können beispielsweise Modelle aus einem Projekt zur Geschäftsprozessoptimierung auch zur Analyse und Verbesserung der Wissensverarbeitung herangezogen werden.

Das Wissensstrukturdiagramm befindet sich in der Datensicht des Fachkonzeptes, die Wissenslandkarte - ebenso wie die erweiterten Modelltypen für die Geschäftsprozessmodellierung - in der Steuerungssicht des Fachkonzeptes.

7.2 Objekttypen zur Modellierung der Wissensverarbeitung

7.2.1 Wissenskategorie

Der Objekttyp **Wissenskategorie**, dargestellt durch eine ovale Denkblase (vgl. Abbildung **Wissensstrukturdiagramm** (Seite 185)) repräsentiert einen inhaltlichen Gegenstand, auf den sich bestimmtes Wissen bezieht. Beispiele für **Wissenskategorien** sind: Wissen über Projektmanagement, Wissen über eine bestimmte Branche, Wissen über eine bestimmte Technologie, Wissen über Kunden und Wettbewerber usw. Mit Hilfe derartiger Kategorien lässt sich das im Unternehmen vorhandene oder benötigte Wissen einteilen.

Bei dem in eine bestimmte Wissenskategorie eingeteilten Wissen kann es sich sowohl um implizites Wissen handeln, also um nicht vollständig dokumentierbares, als Fähigkeiten von Mitarbeitern oder Gruppen vorhandenes Wissen, als auch um explizites Wissen, das - etwa in Form einer Beschreibung oder einer technischen Zeichnung - dokumentiert sein kann. Häufig beinhalten Wissenskategorien beides. Beispielsweise kann Wissen über Projektmanagement sowohl die Erfahrungen von Projektleitern als auch Hinweise in einem Handbuch zum Projektmanagement umfassen.

Zur näheren Beschreibung von Wissenskategorien dienen - neben allgemeinen Attributen wie Beschreibung, Bemerkung, Quelle etc. - folgende spezifischen Attribute:

Attributname	Wertebereich	Beschreibung/Beispiel
Umschlaghäufigkeit	Aufzählungstyp: stündlich, täglich, wöchentlich, monatlich, jährlich, selten, nie	Die Umschlaghäufigkeit gibt an, in welchen Zeitabständen das Wissen der entsprechenden Kategorie erneuert werden muss, um aktuell zu sein. Beispielsweise sind Grundkenntnisse der Trigonometrie nur selten oder - für praktische Zwecke - nie zu erneuern, wogegen Wissen über bestimmte Börsenkurse ggf. täglich oder gar stündlich zu erneuern ist.
Bedeutung	Prozentzahl: 0..100	Die Bedeutung der Wissenskategorie für das Unternehmen kann zwischen 0% (völlig unwichtig) und 100% (extrem wichtig) angegeben werden.
Abdeckungsgrad	Prozentzahl: 0..100	Der aktuelle Abdeckungsgrad des betreffenden Wissens im Unternehmen kann zwischen 0% (überhaupt nicht abgedeckt) und 100% (maximal denkbare Abdeckung) angegeben werden. Soll der Abdeckungsgrad der Wissenskategorie durch eine bestimmte Organisationseinheit oder Person dargestellt werden, kann dies in einem entsprechenden Attribut der Kante des Typs verfügt über in einer Wissenslandkarte erfolgen.

Attributname	Wertebereich	Beschreibung/Beispiel
Wissensvorsprung	Prozentzahl: 0..100	Der relative Wissensvorsprung gegenüber dem Wettbewerb kann zwischen 0% (Wettbewerb verfügt über maximal vorstellbaren Wissensvorsprung) und 100% (eigenes Unternehmen verfügt über maximalen Wissensvorsprung) angegeben werden.
Wissensnutzung	Prozentzahl: 0..100	Der Nutzungsgrad einer speziellen Wissenskategorie kann zwischen 0% (betreffendes Wissen wird überhaupt nicht genutzt) und 100% (betreffendes Wissen wird optimal genutzt) angegeben werden.
Angestrebter Abdeckungsgrad	Prozentzahl: 0..100	Der anzustrebende Abdeckungsgrad des betreffenden Wissens durch das Unternehmen kann zwischen 0% (überhaupt nicht abgedeckt) und 100% (maximal denkbare Abdeckung) angegeben werden.
Künftige Bedeutung	Aufzählungstyp: stark sinkend, sinkend, gleich bleibend, steigend, stark steigend	Künftige Bedeutung stellt die erwarteten Veränderungstendenzen bzgl. der Bedeutung einer Wissenskategorie für das Unternehmen dar.
Strukturelle Änderungsgeschwindigkeit	Prozentzahl: 0..100	Die strukturelle Änderungsgeschwindigkeit gibt an, wie schnell sich die angewandten Methoden zum Erwerb des entsprechenden Wissens ändern müssen (0%: keine Änderung, 100%: maximale Änderungsgeschwindigkeit).

Diese Attribute dienen der Beurteilung der jeweiligen **Wissenskategorie** in Bezug auf das eigene Unternehmen. Sie können daher als Grundlage für die Identifizierung von wichtigen und dringenden Maßnahmen zur Verbesserung des eigenen Wissensmanagements herangezogen werden. Häufig ist es nützlich, derartige Werte grafisch darzustellen. Eine einfache Möglichkeit hierfür besteht darin, sie per Kopieren und Einfügen aus dem Fenster **Attribute** in ein Tabellenkalkulationsprogramm zu übertragen und dort entsprechende Modelle zu erstellen. Beispielsweise können für die betrachteten **Wissenskategorien** in einem Balkendiagramm jeweils der derzeitige und der angestrebte **Abdeckungsgrad** gegenübergestellt werden.

7.2.2 Dokumentiertes Wissen

Im Gegensatz zum Objekttyp **Wissenskategorie**, der implizites und explizites Wissen umfassen kann, bezieht sich der Objekttyp **Dokumentiertes Wissen** ausschließlich auf solche **Wissenskategorien**, die explizit dokumentiert sind oder zumindest prinzipiell dokumentierbar wären. Ein Beispiel hierfür ist das in einem Handbuch dokumentierte Wissen über die Bedienung einer Software. Bei der Einteilung von Wissen in Wissenskategorien hilft die Unterscheidung zwischen allgemeinen Wissenskategorien und dokumentiertem Wissen dabei, Möglichkeiten und Grenzen einer Informationssystemunterstützung der Wissensverarbeitung zu identifizieren, da nur dokumentiertes Wissen elektronisch gespeichert, übertragen und verarbeitet werden kann. Der Objekttyp **Dokumentiertes Wissen** wird durch eine rechteckige Denkblase dargestellt. Er beinhaltet die gleichen spezifischen Attributtypen wie der Objekttyp **Wissenskategorie** (siehe Seite 7-1).

7.3 Modelltypen zur Modellierung der Wissensverarbeitung

7.3.1 Wissensstrukturdiagramm

Mit Hilfe eines Wissensstrukturdiagramms lassen sich Wissenskategorien inhaltlich untergliedern. Dies ist in folgender Abbildung an einem Beispiel dargestellt. Eine Wissenskategorie kann sowohl andere Wissenskategorien umfassen, als auch dokumentiertes Wissen. Dokumentiertes Wissen kann wiederum in verschiedene Unterkategorien dokumentierten Wissens unterteilt werden. Es kann jedoch keine allgemeinen Wissenskategorien umfassen.

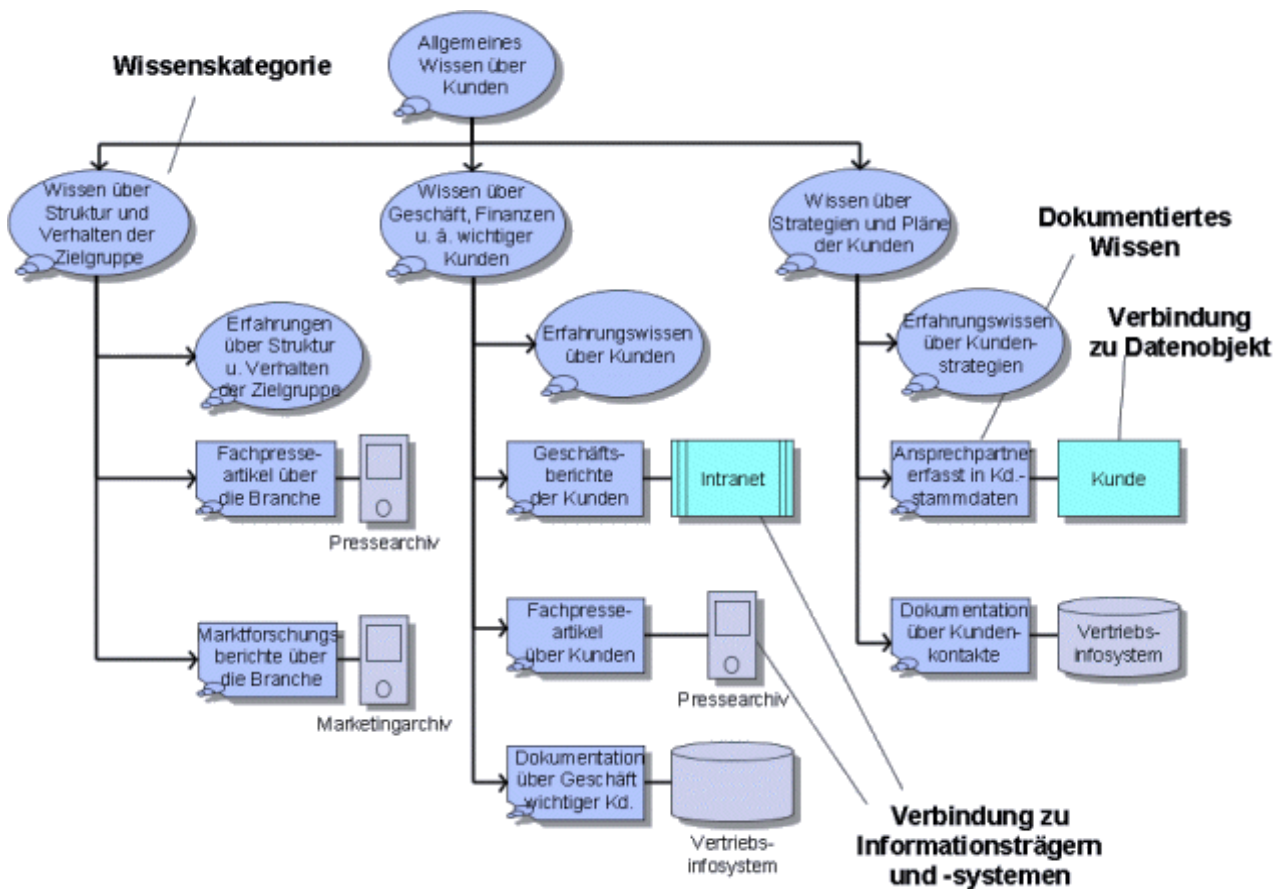


Abbildung 179: Wissensstrukturdiagramm

Für dokumentiertes Wissen kann im Wissensstrukturdiagramm dargestellt werden, auf welchen Informationsträgern es dokumentiert ist, bzw. welche Informationsobjekte eines Datenmodells oder welche Klassen eines objektorientierten Systems der Dokumentation dieses Wissens dienen. Schließlich lässt sich modellieren, mit welchen Typen oder Klassen von Anwendungssystemen dokumentiertes Wissen verwaltet wird.

7.3.2 Wissenslandkarte

In einer Wissenslandkarte lässt sich die organisatorische Verteilung der unterschiedlichen Wissenskategorien abbilden. Hierzu lassen sich unterschiedliche Objekttypen der Organisationssicht (z. B. Organisationseinheit, Stelle, Person, Standort, Gruppe) mit Hilfe von Kanten des Typs **verfügt über** mit Wissenskategorien verbinden. Neben der Tatsache, dass eine bestimmte Person oder Organisationseinheit über Wissen einer bestimmten Kategorie verfügt, lässt sich hierfür auch noch ein Abdeckungsgrad angeben. Die Kante **verfügt über** enthält das Attribut **Abdeckungsgrad**, mit dessen Hilfe in Prozentzahlen ausgedrückt werden kann, in welchem Maße das Wissen der gewählten Kategorie von der entsprechenden Organisationseinheit abgedeckt wird. Ein Wert von 100% steht für eine maximale Abdeckung; ein Wert von 0% bedeutet, dass keinerlei Wissen der genannten Kategorie vorhanden ist. Dies ist gleichbedeutend mit dem völligen Fehlen der Kante. Neben dieser quantitativen Angabe ist auch eine grafisch darstellbare qualitative Bewertung möglich. Hierzu dient das Kantenattribut **Abdeckung qualitativ**, das die Werte **wenig**, **mittel**, **hoch** und **maximal** annehmen kann. Diese Angaben können durch grafische Symbole an den Kanten visualisiert werden, wie dies in folgender Abbildung dargestellt ist. Zwischen den Werten der Attribute **Abdeckungsgrad** und **Abdeckung qualitativ** besteht kein unmittelbarer Zusammenhang. Sollen beide Attribute gemeinsam verwendet werden, ist es sinnvoll, die Qualifikation **wenig** bei einem Abdeckungsgrad bis zu 25% zu verwenden, entsprechend **mittel** zwischen 25% und 50%, **hoch** zwischen 50% und 75% und **maximal** zwischen 75% und 100%.

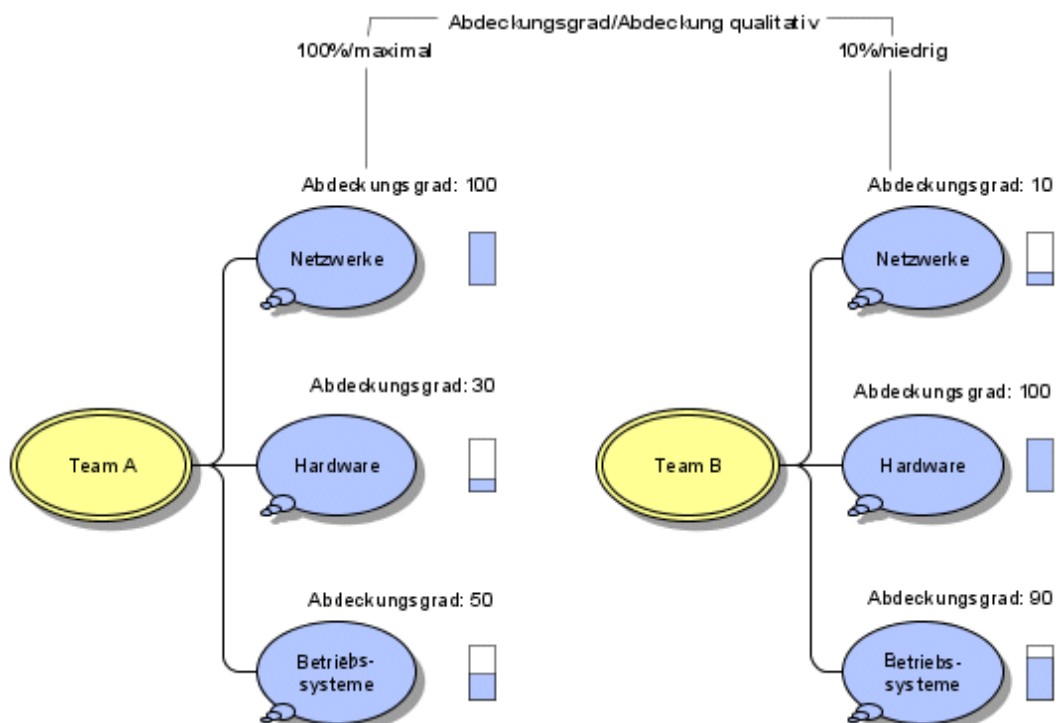


Abbildung 180: Wissenslandkarte - organisationseinheitenbezogen

Die Wissenslandkarte im Beispiel dieser Abbildung ist organisationseinheitenbezogen aufgebaut, d. h. für jede Organisationseinheit sind alle relevanten **Wissenskategorien** angegeben. Um-

gekehrt ist es natürlich auch möglich, die **Wissenseinheiten** als zentralen Betrachtungsgegenstand zu wählen und alle betreffenden Organisationseinheiten hinzu zu modellieren. Über die Navigationsmöglichkeiten von ARIS (Registerkarte **Beziehungen** des Dialogs **Eigenschaften - Objekt**) lassen sich in beiden Fällen leicht die sonstigen Verbindungen einer Organisationseinheit oder einer **Wissenskategorie** finden. Häufig wird für Wissenslandkarten eine Matrixdarstellung verwendet. Diese kann über die spaltenförmige Anordnung mehrerer Ausprägungen der gleichen Wissenskategorie erreicht werden, wie dies in der nächsten Abbildung dargestellt ist. In diesem Beispiel wird der Name nur für die - ähnlich wie in der Kopfzeile einer Tabelle - oben dargestellten **Wissenskategorien** angezeigt, für die anderen Ausprägungen wurde der Name über die Attributplatzierung entfernt. Hier wird außerdem eine alternative optische Darstellungsart für unterschiedliche Abdeckungsgrade gezeigt, indem die **Wissenskategorien** unterschiedlich groß skaliert wurden.

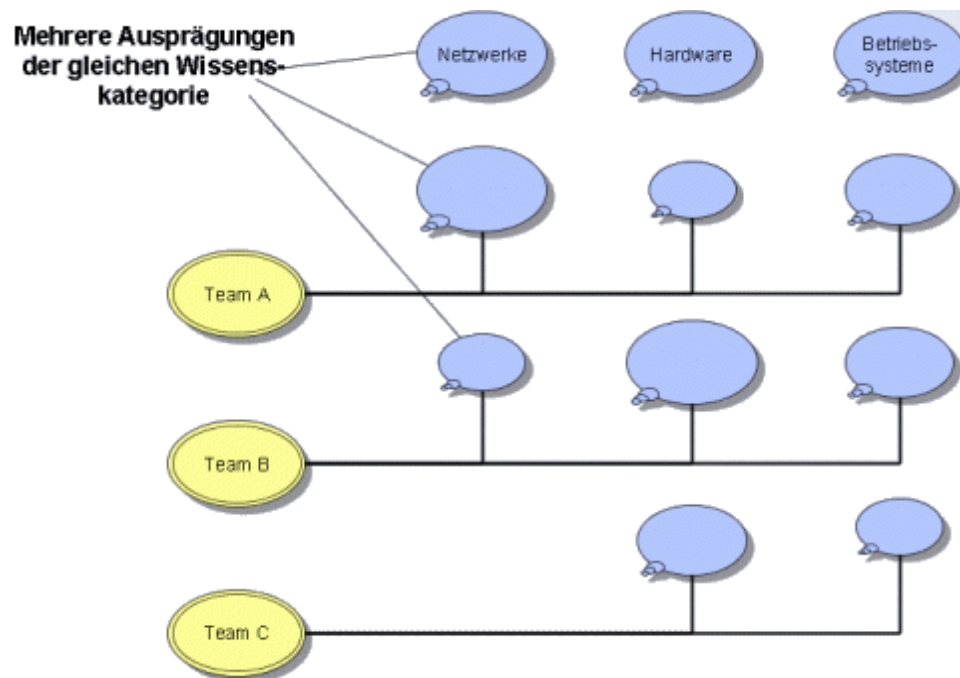


Abbildung 181: Wissenslandkarte - Matrixdarstellung

7.3.3 Darstellung der Wissensverarbeitung in Geschäftsprozessen

Die Nutzung und das Entstehen von Wissen in den Geschäftsprozessen des Unternehmens wird in den Modelltypen für die Geschäftsprozessdarstellung (EPK, EPK (Materialfluss), Office Process, Industrial Process, VKD, VKD (Materialfluss)) modelliert. In diesen Modelltypen stehen nun auch die Objekttypen **Wissenskategorie** und **Dokumentiertes Wissen** zur Verfügung. Zu einer Funktion kann nun einerseits angegeben werden, welches (allgemeine oder dokumentierte) Wissen für ihre Durchführung benötigt, andererseits welches Wissen bei ihrer Durchführung entsteht bzw. dokumentiert wird. Mit Hilfe einer derartigen Darstellung können Geschäftsprozesse im Hinblick auf ihre Wissensverarbeitung untersucht werden. Beispielsweise können nicht gedeckte Wissensbedarfe aufgedeckt werden. Ebenso kann z. B. das benötigte Qualifikationsprofil zur Durchführung einer Funktion ermittelt werden.



Abbildung 182: Wissensverarbeitung in einer EPK

8 Anwendungsfälle

Dieses Kapitel soll helfen, für bestimmte betriebswirtschaftliche Problemstellungen die richtige Unterstützung durch ARIS zu finden. Hierzu ist das Kapitel in Anwendungsszenarien (Unterkapitel) aufgeteilt.

Für jedes Anwendungsszenario ist kurz beschrieben, was unter dem Szenario zu verstehen ist und welche Tätigkeiten standardmäßig in dem Szenario durchgeführt werden. Anschließend sind Aufgaben beschrieben, die typischerweise in dem Szenario vorkommen. Zu jeder Aufgabe ist dargestellt, wie man ARIS bei der Lösung der Aufgabe einsetzen kann.

Eine Übersicht der beschriebenen Anwendungsfälle nebst zu verwendenden Modelltypen zeigt die nachfolgende Tabelle:

Szenario	Szenarioaufgaben	Modelltypen
Allgemeine Unternehmensdokumentation	Dokumentation der Unternehmensziele Dokumentation der Unternehmenswertschöpfung Dokumentation der Organisationsstruktur Dokumentation der Unternehmensfunktionen Prozessdokumentation Process Warehousing	Zieldiagramm Wertschöpfungskettendiagramm Organigramm Funktionsbaum Office Process Industrial Process EPK VKD
Datenbankmanagement/ Data Warehousing (Seite 193)	Datenstrukturierung/Datenbankentwurf Datenbank-/Zugriffsverwaltung	ERM SAP-SERM SeDaM IEF-Datenmodell Relationendiagramm Tabellendiagramm Class Diagramm Class Description Diagramm
Groupware (Seite 194)	Integration von Lotus Notes und ARIS	
PC-Hardware- und Netzwerkmanagement (Seite 195)	Ermittlung des informationstechnischen Infrastrukturbedarfs Dokumentation der informationstechnischen Infrastruktur	Netztopologie Netzdiagramm

Szenario	Szenarioaufgaben	Modelltypen
	Zugriffsberechtigungen	
Prozesskostenrechnung (Seite 196)	Beschreibung der Prozess- und Organisationsstrukturen Kostenstellenanalyse Prozesskalkulation	EPK VKD Organigramm CD-Diagramm Kostenartendiagramm
Qualitätsmanagement (Seite 197)	Aufbau von QM-Dokumentationen Zertifizierungsvorgehen Zertifizierungsunterlagen	Produktbaum Produktauswahlmatrix EPK VKD Office Process Diagramm Industrial Process Diagramm Wertschöpfungskettendiagramm Gliederungsmodell Organigramm
Reorganisations- maßnahmen	Projektdokumentation Reorganisationsdurch- führung	Wertschöpfungskettendiagramm EPK Organigramm Produktmodell Leistungsmodell Zieldiagramm VKD
SAP R/3-Einführung (Seite 199)	Phase Analyse, Spezifikation (Project Preparation) Phase Design (Business Bluep- rint); 3 Aufgabenfälle vorhan- den	EPK Organigramm
Software-Entwicklung und -einführung (Seite 200)	Projektdokumentation Festlegung von Anwendungs- systemen und Modulen Beschreibung der DV-Prozesse Entwicklung der Systemober- fläche	Wertschöpfungskettendiagramm Organigramm EPK Use-Case-Diagramm Anwendungssystemtypdiagramm Programmablaufplan Maskendiagramm

Szenario	Szenarioaufgaben	Modelltypen
Wissensmanagement (Seite 201)	Wissenslandkarte bzw. Yellow Page Wissenskategorisierung Wissensverarbeitung in Ge- schäftsprozessen	Wissenslandkarte Wissensstrukturdiagramm EPK VKD Office Process Industrial Process Funktionszuordnungsdiagramm
Workflow-Management (Seite 202)	Prozess-Customizing von Workflow-Management-System en	Rechtediagramm EPK Funktionszuordnungsdiagramm Anwendungssystemdiagramm Anwendungssystemtypdiagramm

8.1 Allgemeine Unternehmensdokumentation

Zu Schulungs-, Präsentations- und Bewertungszwecken jedweder Art können Unternehmenscharakteristika, z. B. Abläufe, Strukturen und Daten, in geeigneter Form dokumentiert werden. Die wichtigsten Aufgaben der Unternehmensdokumentation sind im Folgenden kurz beschrieben.

Aufgabe: Dokumentation der Unternehmensziele

ARIS-Unterstützung: Zur Hierarchisierung von Unternehmenszielen und den dazugehörigen Erfolgsfaktoren können Zieldiagramme verwendet werden.

Aufgabe: Dokumentation der Unternehmenswertschöpfung

Grundlage vieler Unternehmensentscheidungen ist die Identifikation der an der Wertschöpfung einer Unternehmung beteiligten Funktionen.

ARIS-Unterstützung: Die an der Wertschöpfung beteiligten Unternehmensfunktionen können durch das Wertschöpfungskettendiagramm dargestellt werden. Dieses Modell macht sowohl die Reihenfolge der aufeinander folgenden Funktionen als auch über- und untergeordnete Funktionen deutlich.

Aufgabe: Dokumentation der Organisationsstruktur

ARIS-Unterstützung: Die Struktur einer Unternehmung kann durch Organigramme dokumentiert werden, die Hierarchie und Beziehungen von Organisationseinheiten darstellen.

Aufgabe: Dokumentation der Unternehmensfunktionen

ARIS-Unterstützung: Durch einen Funktionsbaum können einzelne Unternehmensfunktionen übersichtartig dargestellt werden. Die Funktionen werden entweder objektorientiert, prozessorientiert oder verrichtungsorientiert gegliedert.

Aufgabe: Prozessdokumentation

ARIS-Unterstützung: Unternehmensabläufe können je nach Branche bzw. Prozessart durch Office oder Industrial Process Diagramme erfasst werden, ohne notwendigerweise auf Methodenwissen zurückzugreifen. Bei Vorhandensein von Methodenwissen und zur weiteren Nutzung der Prozessmodelle für SAP-Anwendungen, Simulation, Workflow usw. wird die Modellierung mittels EPK bzw. zur Schwachstellenanalyse mittels Vorgangskettendiagramm (VKD) empfohlen.

Aufgabe: Process Warehousing

Unter einem Process Warehouse versteht man die systematische Erfassung, Speicherung und Pflege des Geschäftsprozesswissens in einem Repository.

ARIS-Unterstützung: Zur Modellierung des Prozesswissens in den dezentralen Einheiten wird die Nutzung von Office und Industrial Process Diagrammen empfohlen, da die operativ arbeitenden Mitarbeiter i.d.R. kein methodisches Know-how haben. Zur Pflege und Verwaltung im zentralen Modell-Repository hingegen wird die Umwandlung der Modelle in EPKs empfohlen, angereichert durch Dokumente, Bilder, Videos, da sie so zu anspruchsvolleren Auswertungen, wie Simulation oder Prozesskostenrechnung, genutzt werden können.

8.2 Datenbankmanagement/Data Warehousing

Das Speichern von Unternehmensdaten in Datenbanken reduziert eine redundante Datenhaltung und ermöglicht den programmunabhängigen Zugriff auf unternehmensweite Daten. Durch Data Warehousing soll die Qualität, die Integrität und die Konsistenz der zugrunde liegenden Daten sichergestellt werden. Der Begriff des Data Warehouse bezeichnet generell eine von den operationalen DV-Systemen isolierte Datenbank, die als unternehmensweite Datenbasis für alle Ausprägungen managementunterstützender Systeme dient und durch eine strikte Trennung von operationalen und entscheidungsunterstützenden Daten und Systemen gekennzeichnet ist. Der Fokus des Data Warehouse-Konzeptes liegt auf der effizienten Bereitstellung und Verarbeitung großer Datenmengen für die Durchführung von Auswertungen und Analysen in entscheidungsrelevanten Prozessen.

Aufgabe: Datenstrukturierung/Datenbankentwurf

Die Struktur von Datenbanken wird durch die zugrunde liegenden Datenmodelle vorgegeben.

ARIS-Unterstützung: Die verbreitetste Methode der Datenmodellierung ist die des Entity-Relationship-Modells (ERM), welches die Grundlage zur Implementierung einer relationalen Datenbank legt.

Produkt- bzw. firmenspezifische Abwandlung des ERM ist das SAP-ERM zur Informationsmodellierung der SAP AG, das SeDaM (semantisches Datenmodell) als Notation der BASF AG und das IEF-Datenmodell (Information Engineering Facility) zur Datenmodellierung des CASE-Tools der Texas Instruments Inc.

Zur konkreteren Beschreibung der logischen Datenstrukturen kann das Relationendiagramm genutzt werden, welches auf den im ERM entworfenen Datenstrukturen aufsetzt.

Tabellen und Felder eines Datenbanksystems werden durch das Tabellendiagramm beschrieben. Objektorientierte Datenbanksysteme können mit Hilfe der Unified Modeling Language (UML) oder mit der Object Modeling Technique (OMT) entworfen werden. In der UML werden mit dem Class Diagramm die statischen Datenbeziehungen dargestellt, das Class Description Diagramm ermöglicht eine zusätzliche Darstellung von Attributen, Objekten, Schnittstellen usw.

Aufgabe: Datenbank-/Zugriffsverwaltung

Zuordnung von Anwendern und Systemverantwortlichen zu den Datenbanksystemen.

ARIS-Unterstützung: Durch das Zugriffsdiagramm können in Verbindung mit Relationen und Systemkomponenten Zugriffsrechte von Organisationseinheiten, Stellen und Personen auf das Datenbanksystem festgelegt werden.

8.3 Groupware

Groupware ist eine Technik, die das Arbeiten in einer Gruppe unterstützt. Ein wichtiges Merkmal von Groupware besteht darin, dass eine Gruppe von Anwendern gemeinsam Aufgaben erledigt und diese mit Hilfe einer Software strukturiert bearbeitet. Hierbei kann man zwischen Workgroup Computing (Koordination von Arbeitsgruppen), Collective Writing (Erstellung gemeinsamer Dokumente) und Group Decision Support Systems (Unterstützung der Entscheidungsfindung in Gruppen) unterscheiden.

Aufgabe: Integration von Lotus Notes und ARIS

Die Systeme ARIS und Lotus Notes sind für unterschiedliche Arten der Dokumentation geeignet. Während der Vorteil von ARIS vorwiegend darin liegt, die Zusammenhänge unterschiedlicher Aspekte, z. B. Prozesse oder Organisationsstruktur, grafisch einfach und komfortabel zu visualisieren sowie weitreichende Analyseverfahren zur Verfügung zu stellen, können in Lotus Notes umfangreiche Textdokumente, wie Arbeitsanweisungen und Formulare, effizient verwaltet werden. Es ist sinnvoll, beide Formen der Dokumentation zu nutzen und zu integrieren.

ARIS-Unterstützung: Durch ARIS Connectivity for Lotus Notes ist es möglich, von einem beliebigen Objekt, einem Modell, einer Gruppe oder einer Datenbank in ARIS, per Verknüpfung in ein zugehöriges Dokument, eine Ansicht oder eine Datenbank in Lotus Notes zu verweisen.

Für die Zertifizierung und Schulung ist es sinnvoll, Prozessbeschreibungen in ARIS, Arbeitsanweisungen und Formulare hingegen in Notes zu pflegen.

Zur Verwaltung von ARIS-Modellen kann die Katalogisierungs- und Retrieval-Funktionalität von Lotus Notes genutzt werden.

8.4 PC-Hardware- und Netzwerkmanagement

Unter Netzwerkmanagement versteht man die Steuerung, Überwachung und Koordination aller (verteilter) Ressourcen (Datennetze, Rechner, Daten und Anwendungen), die eine Kommunikation in einem Rechnernetz ermöglichen.

Aufgabe: Ermittlung des informationstechnischen Infrastrukturbedarfs

Aus einer bestehenden Aufbauorganisation soll abgeleitet werden, durch welche Kommunikations- und Informationssystemlandschaften diese effizient unterstützt werden kann.

ARIS-Unterstützung: Mit dem Modelltyp Netztopologie sind die Anforderungen an den strukturellen Aufbau von Informationssystemen abbildbar. Bei der Darstellung der Anwendungssysteme, Netztypen und Hardwarekomponenten handelt es sich nicht um einzeln identifizierbare Exemplare (z. B. PC mit Inventarisierungsnummer 3423), sondern um Typisierungen auf gleicher technologischer Basis.

Aufgabe: Dokumentation der informationstechnischen Infrastruktur

Aufgabe ist die Abbildung einer bestehenden oder zu installierenden IT-Infrastruktur mit konkreten Hardwarekomponenten, Netzen und Anwendungssystemen.

ARIS-Unterstützung: Eine IT-Infrastruktur lässt sich mit Hilfe des Netzdiagramms darstellen, als konkrete Umsetzung einer Netztopologie.

Aufgabe: Zugriffsberechtigungen

Aufgabe ist aufzuzeigen, welche Applikationen und Benutzer in welcher Weise Zugriff auf welche Daten haben.

ARIS-Unterstützung: Mit dem Zugriffsdiagramm kann beschrieben werden, welche Applikationen bzw. Applikationsmodule in welcher Form (schreiben/lesen/verändern) Zugriff auf Datenbestände und Informationsträger haben und ob die Daten als Input oder Output agieren. Weiterhin kann abgebildet werden, welche Benutzerberechtigungen und Ansichten bestimmte Benutzer bzw. Benutzergruppen auf die Applikationen bzw. Applikationsmodule besitzen.

8.5 Prozesskostenrechnung

Die Kostenrechnung stellt durch die Erfassung und Zurechnung der bei der betrieblichen Leistungserstellung anfallenden Kosten eine Dispositionsgrundlage und ein Kontrollinstrument zur Verfügung. Durch die Änderung der Kostenstrukturen, insbesondere durch den Anstieg der Gemeinkosten, werden die traditionellen Kostenrechnungsmethoden durch die Prozesskostenrechnung ersetzt. Die Prozesskostenrechnung bestimmt die Kosten von Prozessen kostenstellenübergreifend. Budgetplanung, Kostentransparenz in den indirekten Leistungsbereichen, Preisfindung und Unterstützung bei Make-or-Buy-Entscheidungen sind die Hauptvorteile der Prozesskostenrechnung.

Teile der nachfolgend beschriebenen ARIS-Unterstützung sind nur mit ARIS Business Optimizer möglich.

Aufgabe: Beschreibung der Prozess- und Organisationsstrukturen

Festlegen der Prozesse, auf die die Prozesskostenrechnung angewendet werden soll, sowie Beschreibung der Kostenstellen.

ARIS-Unterstützung: Prozesse werden mit den Standardmodelltypen, z. B. EPK und VKD, abgebildet. Wichtig für die Prozesskostenrechnung sind die Pflege der Zeitenattribute sowie eine Zuordnung von Organisationseinheiten.

Die Unternehmensorganisation wird in einem Organigramm beschrieben, wobei die Organisationseinheiten Kostenstellen (u. a. mit den Attributen **Kostensatz** und **Leistung**) entsprechen.

Aufgabe: Kostenstellenanalyse

ARIS-Unterstützung: Für die Kostenstelleninanspruchnahme können Kostentreiber verwendet werden, die in einem CD-Diagramm definiert werden.

Die Kalkulation kann für beliebig viele Kostenarten durchgeführt werden. Die Kostenartenstruktur ist in einem Kostenartendiagramm abzubilden.

Zusätzlich ist die Erstellung einer Kostenartentabelle und einer Funktionstabelle notwendig, in denen die zu analysierenden Objekte beschrieben werden.

Aufgabe: Prozesskalkulation

ARIS-Unterstützung: Voraussetzung ist eine vollständige Kostenstellenanalyse mit einer Ermittlung der Prozesskostensätze. Für die Durchführung der Prozesskalkulation müssen keine weiteren Modelle gepflegt werden. Die Ergebnisse werden in einer Kalkulationstabelle ausgewiesen.

8.6 Qualitätsmanagement

Unter Qualitätsmanagement (QM) versteht man alle Tätigkeiten, die die Qualitätspolitik, Ziele und Verantwortungen einer Unternehmung festlegen. Mittel zur Verwirklichung dieser Tätigkeiten sind Qualitätsplanung, Qualitätslenkung (Prozessmanagement), Qualitätssicherung und Qualitätsverbesserung (Qualitätsförderung).

Aufgabe: Aufbau von QM-Dokumentationen

Um die Qualität von Produkten und Prozessen innerhalb einer Unternehmung zu sichern, ist eine geeignete Dokumentation anzufertigen, die Produkte und Prozesse bewertbar, vergleichbar und damit verbesserbar macht.

ARIS-Unterstützung: Zur Dokumentation von Produkten eignen sich Produktbäume, mit deren Hilfe sich Produkte klassifizieren lassen. Diese Art der Darstellung wird zunehmend im Dienstleistungsbereich und dort speziell in der öffentlichen Verwaltung angewendet. Mit der Produktauswahlmatrix ist weiter darstellbar, welche Unternehmensfunktionen zur Erstellung welcher Produkte notwendig sind und welche Organisationseinheiten für die Produkterstellung verantwortlich sind.

Weiterhin geht es bei QM-Dokumentationen hauptsächlich um die Dokumentation von Abläufen, die mit Hilfe von EPKs, VKDs oder Office- und Industrial Processes festgehalten werden, durch Reports ausgewertet werden und auf Dokumente und Anwendungen im Unternehmen verweisen können.

Aufgabe: Zertifizierungsvorgehen

Unterstützung des Projektmanagements bei der Zertifizierung nach national und international anerkannten Normen, wie ISO oder VDA, durch Vorgehensmodelle.

ARIS-Unterstützung: Ein Vorgehensmodell zur Zertifizierung (z. B. das ARIS-Vorgehensmodell) kann mit Hilfe eines Wertschöpfungskettendiagramms dargestellt werden. Die einzelnen Schritte können durch Hinterlegung weiterer Prozessmodelle detaillierter beschrieben werden.

Aufgabe: Zertifizierungsunterlagen

Erstellung der notwendigen Qualitätsunterlagen für die Zertifizierung.

ARIS-Unterstützung: Durch das Gliederungsmodell werden einzelne Zertifizierungsnormen in ihre Komponenten untergliedert. Den einzelnen Elementen des Gliederungsmodells können zur Qualitätslenkung Unternehmensmodelle hinterlegt werden. Diese Modelle können beispielsweise Prozessmodelle in Form von EPKs, Office oder Industrial Processes sein, Organigramme oder auch Wertschöpfungskettendiagramme.

Mit ARIS Business Architect können Reporte generiert werden, die als QM-Handbuch zur Zertifizierung anerkannt sind.

8.7 Reorganisationsmaßnahmen

Im Rahmen von Reorganisationsmaßnahmen zur Kosten- oder Zeitreduktion, Verbesserung der Ergebnisqualität oder Verbesserung der Arbeitsqualität werden Unternehmensabläufe entweder verändert (Prozess-Redesign) oder komplett neu entworfen (Prozess-Reengineering).

Aufgabe: Projektdokumentation

Dokumentation der Planung, des Vorgehens und der Ergebnisse der Reorganisationsmaßnahmen.

ARIS-Unterstützung: Die Hauptprojektphasen der Reorganisation können als Vorgehensmodell durch ein Wertschöpfungskettendiagramm beschrieben werden.

Die einzelnen Projektaktivitäten im Reorganisationsprojekt können in ihren Abläufen mit Hilfe von EPKs dokumentiert werden.

Die Organisation der am Projekt teilhabenden Personen und Einheiten kann durch Organigramme dargestellt werden.

Aufgabe: Reorganisationsdurchführung

Im Rahmen einer Reorganisation folgen nach der Projektvorbereitung und der strategischen Planung die Analyse der Ist-Situation, die Entwicklung des Soll-Konzepts und schließlich die Implementierung der Lösungen.

ARIS-Unterstützung: Zur Dokumentation der strategischen Rahmenbedingungen werden Produkt- und Leistungsmodelle sowie Zieldiagramme eingesetzt, damit die wesentlichen Geschäftsfelder des Unternehmens mit ihren Produkten, Dienstleistungen und Kundengruppen erfasst und die kritischen Erfolgsfaktoren sowie die Zielhierarchie des Unternehmens abgebildet werden.

Während der Ist-Analyse wird mit Hilfe von Wertschöpfungskettendiagrammen zunächst ein Rahmen mit den wesentlichen Geschäftsprozessen erarbeitet. Diese Geschäftsprozesse werden danach detailliert auf der Grundlage von Interviews mit den Mitarbeitern als EPK oder Vorgangskettendiagramm (VKD) erfasst. Das VKD eignet sich besonders gut zur Identifikation von Schwachstellen, die durch Medienbrüche und wechselnde Prozessverantwortlichkeiten entstehen.

Nach einer Schwachstellenanalyse unter Berücksichtigung von Durchlaufzeiten, Prozesskosten, organisatorischen Brüchen, System- und Medienbrüchen, Datenredundanzen u. a. werden alternative Soll-Abläufe definiert. Diese werden, wie zuvor in der Ist-Erfassung geschehen, mit Hilfe von EPKs modelliert.

Zur Unterstützung der Implementierung werden System-, Organisations- und Datenkomponenten am Ende der Soll-Konzeption konkreter dargestellt, beispielsweise kann die Angabe **Textverarbeitung** als Anwendungssystemkonstrukt nun als Microsoft Word spezifiziert werden.

Bemerkung: Die Phase der Schwachstellenanalyse kann durch Auswertungen mit Hilfe von Simulation und ARIS Business Optimizer unterstützt werden.

8.8 SAP R/3-Einführung

Die Unterstützungspotentiale von ARIS bei der Einführung der Standardsoftware R/3 der SAP AG orientieren sich an dem Lebenszyklus des Einführungsansatzes ASAP. Neben ASAP werden jedoch auch Ansätze unterstützt, die mehr an der Geschäftsprozessoptimierung (im weitesten Sinne) orientiert sind. Teile der nachfolgend beschriebenen ARIS-Unterstützung sind nur mit ARIS for SAP möglich.

Aufgabe: Phase Analyse, Spezifikation (Project Preparation)

Aufgabe ist die Identifikation des Abdeckungsgrades der unternehmensspezifischen Prozesse durch das SAP-System sowie die rechtzeitige Identifikation von möglichen Schwachstellen.

ARIS-Unterstützung: Wird die Untersuchung nicht direkt durch das SAP R/3-Referenzmodell durchgeführt, so können die zu unterstützenden, „betriebswirtschaftlich optimalen“ Prozesse mit ARIS modelliert werden (siehe allgemeine Unternehmensdokumentation). Anschließend kann durch Reporte eine erste Schätzung des Abdeckungsgrades ermittelt werden, indem das Unternehmensmodell mit dem SAP R/3-Referenzmodell gemappt wird.

Aufgabe: Phase Design (Business Blueprint)

Prozess- und/oder Funktionsdefizite wurden in dem SAP R/3-Referenzmodell identifiziert.

ARIS-Unterstützung: Mit den aus dem SAP R/3-Referenzmodell bestehenden Bausteinen können in ARIS neue Prozess- und Szenariovarianten abgeleitet werden. Zusätzlich können die Prozess- und Szenariobausteine um neue Funktionen, Ereignisse und Regeln erweitert werden (d.h. ggf. durch die Entwicklung neuer SAP-ABAP-Funktionen).

Aufgabe: Phase Design (Business Blueprint)

Entwurf von Schnittstellen zwischen SAP R/3 und Non-SAP-Anwendungen.

ARIS-Unterstützung: Die Dokumentation der auszutauschenden Attribute kann im ARIS-Repository in Prozessmodellen durch Daten- bzw. Objektzuordnungen zu Funktionen und Daten- bzw. Objektmodellen detailliert beschrieben werden. Diese Informationen können als Reporte ausgegeben werden und bilden die Grundlage für die Schnittstellenentwicklung.

Aufgabe: Phase Design (Business Blueprint)

Entwicklung der organisatorischen Gestaltung der Geschäftsprozesse und des SAP-Systems.

ARIS-Unterstützung: Mittels Organigrammen kann die soziale Unternehmensstruktur und die SAP-Organisationsstruktur dargestellt und einander gegenübergestellt werden.

8.9 Software-Entwicklung und -einführung

Aufgabe: Projektdokumentation

Dokumentation der Planung, des Vorgehens und der Ergebnisse der Software-Entwicklung und -einführung.

ARIS-Unterstützung: Die Hauptprojektphasen des Projektes können als Vorgehensmodell durch ein Wertschöpfungskettendiagramm beschrieben werden.

Die einzelnen Projektaktivitäten während Entwicklung und Einführung können in ihren Abläufen mit Hilfe von EPKs dokumentiert werden.

Die Organisation der am Projekt teilhabenden Personen und Einheiten kann durch Organigramme dargestellt werden.

Aufgabe: Festlegung von Anwendungssystemen und Modulen

Aufgabe ist es, bestehend auf den Systemanforderungen, den strukturellen Aufbau eines Informationssystems darzulegen.

ARIS-Unterstützung: Das Use-Case-Diagramm kann zur Identifikation der Anwendungsfälle des zu entwickelnden Softwaresystems dienen. Weiterhin werden dadurch die Systemanwender definiert und einzelnen Anwendungsfällen zugeordnet. Das Use-Case-Diagramm bildet häufig den Ausgangspunkt für die detaillierte Prozessmodellierung. Einzelnen Use-Cases können Prozessmodelle hinterlegt werden.

Mit dem Anwendungssystemtypdiagramm kann der hierarchischen Aufbau von Anwendungssystemen auf Typebene durch Modultypen und DV-Funktionstypen beschrieben werden.

Für die spezielle Beschreibung der einzelnen Typen durch konkrete Ausprägungen kann das Anwendungssystemdiagramm verwendet werden.

Aufgabe: Beschreibung der DV-Prozesse

Aufgabe ist es, den zeitlich-logischen Ablauf innerhalb von Modulen und modulübergreifend zu beschreiben.

ARIS-Unterstützung: Mit dem Programmablaufdiagramm können DV-Prozesse modelliert werden.

Aufgabe: Entwicklung der Systemoberfläche

Aufgabe ist die Entwicklung und Dokumentation der Benutzerschnittstelle.

ARIS-Unterstützung: Der geographische und funktionale Aufbau einer Bildschirmmaske (Fenster) kann mit einem Maskendiagramm beschrieben werden. Hieraus kann - ähnlich dem Übergang vom ERM zum Relationenmodell - der Programmcode abgeleitet werden.

8.10 Wissensmanagement

Ausgangspunkt für die Gestaltung eines umfassenden Wissensmanagements ist die These, dass Wissen de facto zum dominanten Produktionsfaktor von Unternehmen geworden ist bzw. wird. Daraus ergibt sich die Notwendigkeit, Wissen als ebenso bewirtschaftbare Größe aufzufassen, wie die klassischen betriebswirtschaftlichen Produktionsfaktoren.

Folglich beschäftigt sich Wissensmanagement mit der Gewinnung, Darstellung und Verbreitung von Wissen. Wissensmanagement ist die Summe der Methoden, Maßnahmen und Systeme, die eine Organisation einsetzt, um Wissen zu entwickeln, es transparent zu machen und es zeit-, personen- und ortsunabhängig verfügbar zu machen. Ziel des Wissensmanagements ist es, das Wissen zu vergrößern und die Wissensbasis im Unternehmen optimal anzuwenden.

Aufgabe: Wissenslandkarte bzw. Yellow Page

Es soll abgebildet werden, welches Wissen wo im Unternehmen vorhanden ist.

ARIS-Unterstützung: Mit dem Modelltyp **Wissenslandkarte** lässt sich die organisatorische Verteilung unterschiedlicher Wissenskategorien abbilden. Es kann dargestellt werden, welche Organisationseinheit, Stelle oder welcher Mitarbeiter Know-how in bestimmten Wissenskategorien aufweist, und wie groß diese Kompetenz ist.

Aufgabe: Wissenskategorisierung

Aufgabe ist es, das Intellectual Capital einer Organisation analytisch zu klassifizieren, d.h. die verschiedenen Arten und Gruppen von Wissen darzustellen, um z. B. eine Wissensablagestruktur zu entwerfen.

ARIS-Unterstützung: Mit dem Wissensstrukturdiagramm lässt sich darstellen, wie sich die Wissensbasis einer Organisation in verschiedene Wissenskategorien aufteilt und wie sich diese in weitere Wissenskategorien und dokumentiertes Wissen auffächert. Für dokumentiertes Wissen kann zusätzlich abgebildet werden, auf welchen Informationsträgern dieses Wissen gespeichert ist.

Aufgabe: Wissensverarbeitung in Geschäftsprozessen

Aufgabe ist es darzustellen, wo in den Geschäftsprozessen Wissen erzeugt, verändert und benötigt wird, um eine möglichst effiziente Nutzung der Ressource Wissen zu gewährleisten.

ARIS-Unterstützung: Innerhalb der Modelltypen **EPK**, **Vorgangskettendiagramm**, **Office Process**, **Industrial Process** und **Funktionszuordnungsdiagramm** können die Objekte **Wissenskategorie** und **Dokumentiertes Wissen** verwendet werden. Der Aufbau und die organisatorische Verteilung des Wissens kann durch das Wissensstrukturdiagramm und die Wissenslandkarte gesondert beschrieben werden.

8.11 Workflow-Management

Ein Workflow kann im weitesten Sinne als Geschäftsprozess interpretiert werden. Der Begriff Workflow steht für arbeitsteilige Prozesse, die zur Abwicklung von Geschäftsvorfällen initiiert werden. Dabei kann es sich sowohl um sehr einfache Geschäftsprozesse als auch um vielschichtige, organisationsübergreifende Prozesse handeln. Schwerpunkt der Betrachtung ist der dynamische Ablauf der Prozesse von Anfang bis Ende. Workflow-Management ist die Summe von Methoden, Maßnahmen und Systemen, die eingesetzt werden, um Workflows zu entwickeln, zu steuern und zu optimieren.

Ein Workflow-Management-System ist eine flexibel gestaltbare, nach einem organisatorischen Regelwerk arbeitende, aktiv einwirkende Software, die einen über mehrere Arbeitsplätze gehenden Vorgang steuert und bestehende technische Basiskomponenten einbindet. Mit Systemen der Vorgangssteuerung ist es möglich, komplexe Aufgabenzusammenhänge, an denen eine Vielzahl von Mitarbeitern und Stellen arbeiten, zu unterstützen.

Aufgabe: Prozess-Customizing von Workflow-Management-Systemen

ARIS unterstützt insbesondere die Überführung von allgemeinen Geschäftsprozessmodellen in Workflow-Modelle, mit denen dann verschiedene Workflow-Management-Systeme (halb-)automatisch konfiguriert werden können.

ARIS-Unterstützung: Mit dem Rechediagramm wird beschrieben, welche Workflows (Prozesse) es gibt und welche Personen bzw. Personengruppen diese initiieren dürfen.

Mit der EPK wird analog zur Prozessmodellierung der Tätigkeitsfluss abgebildet. Hierbei ist auf eine streng methodenkonforme Modellierung zu achten! Zu jeder Funktion muss ein Funktionszuordnungsdiagramm erstellt werden, in dem der Funktion ein Bearbeiter sowie die Input- und Output-Daten zugeordnet werden - soweit dies nicht schon in der EPK abgebildet wurde.

Damit zur Run-Time die zu den Daten gehörigen Applikationen automatisch gestartet werden, muss in einem Anwendungssystemtypdiagramm bzw. Anwendungssystemtypdiagramm eine Zuordnung von Dateien zu Applikationen modelliert werden.

9 Balanced Scorecard-Methode

9.1 Einleitung

Um im turbulenten Unternehmensumfeld und dem globalen Wettbewerb heutiger Märkte bestehen zu können, benötigen Unternehmen optimal gestaltete Geschäftsprozesse. Ebenso bedeutsam ist aber auch die Fähigkeit, schnell und, in Bezug auf die strategischen Unternehmensziele, auf neue Entwicklungen im Unternehmensumfeld reagieren zu können. Dazu sind effiziente Managementprozesse notwendig, die eine konsequente Realisierung der Unternehmensstrategien und der strategischen Ziele sowie deren Umsetzung im Tagesgeschäft anhand operativer Maßnahmen erlauben.

Viele der klassischen Managementansätze schaffen keine Verbindung zwischen der Formulierung von Unternehmensstrategien und deren Umsetzung durch strategieorientierte Maßnahmen sowie einer konsequenten Kontrolle der Zielerreichung strategischer Ziele.

Zusätzlich werden viele Unternehmen noch mit reinen Finanzkennzahlen geführt, deren Aussagefähigkeit jedoch begrenzt ist, da sie größtenteils vergangenheitsorientiert sind und deshalb nur wenig zukunftsrelevante Steuerungsinformationen liefern können. Erst die Berücksichtigung der Ursachen des finanziellen Erfolgs (Kunden, Prozesse, Innovationen etc.) ermöglichen es, frühzeitig Abweichungen bezüglich der Erreichung strategischer Ziele zu erkennen.

Der Balanced Scorecard-Ansatz stellt eine übersichtlich strukturierte und einfach zu implementierende Methodik zur Verfügung, mit der diese Schwächen ausgeglichen werden können.

9.2 Das Konzept der Balanced Scorecard

9.2.1 Kernaussagen des BSC-Ansatzes

Der Balanced Scorecard-Ansatz ist ein strategisches Managementsystem, das erstmals von Robert Kaplan und David Norton im Jahre 1992 (The Balanced Scorecard-Measures that drive Performance, Harvard Business Review Januar/Februar 1992) vorgestellt wurde. Es entstand als Weiterentwicklung aus den Ergebnissen einer Studie zum Thema **Performance-Measurement-Ansätze**. Im Rahmen der Studie wurde festgestellt, dass die rein auf monetäre Kennzahlen ausgerichteten Performance-Measurement-Systeme viele Unternehmen an wertschöpfenden Tätigkeiten eher hindern. Ausgehend von dieser Erkenntnis versuchten Kaplan und Norton in Zusammenarbeit mit innovativen Unternehmen ein Kennzahlensystem aufzubauen, das es ermöglichen sollte, die Umsetzung von Visionen und Strategien der Unternehmen möglichst optimal zu messen.

Beim Balanced Scorecard-Ansatz werden Kennzahlen verschiedenen Sichtweisen auf das Unternehmen zugeordnet (sog. Perspektiven). Diese beinhalten sowohl interne (z. B. Lern- und Entwicklungsperspektive, Prozessperspektive) als auch externe Performance-Perspektiven (z. B. Kundenperspektive, finanzwirtschaftliche Perspektive). Durch diese Anordnung der Kennzahlen wird eine gewisse Ausgewogenheit hinsichtlich kurzfristiger und langfristiger Ziele, monetärer und nicht-monetärer Kennzahlen, Früh- und Spätindikatoren sowie interner und externer Sichtweisen erreicht. Mit dem Einbezug von branchenspezifischen Kennzahlen wird zusätzlich noch eine Benchmarking-Komponente in das Konzept integriert.

Aus dem reinen Performance-Measurement-Ansatz entwickelte sich ein umfassendes Führungssystem, das eine zielorientierte Unternehmensführung enthält, ausgehend von der Unternehmensvision über die einzelnen Wettbewerbsstrategien bis hin zur Formulierung und Überwachung von Maßnahmen anhand ausgewogener Kennzahlen. Der Balanced Scorecard-Ansatz ist daher mehr als nur ein Kennzahlen-System zum Performance Measurement. Er unterstützt Unternehmen bei der Kommunikation und Umsetzung der Unternehmensstrategie und dem daraus resultierenden strategischen Lernprozess (double loop learning).

9.2.2 Der strategische Managementprozess und die Balanced Scorecard

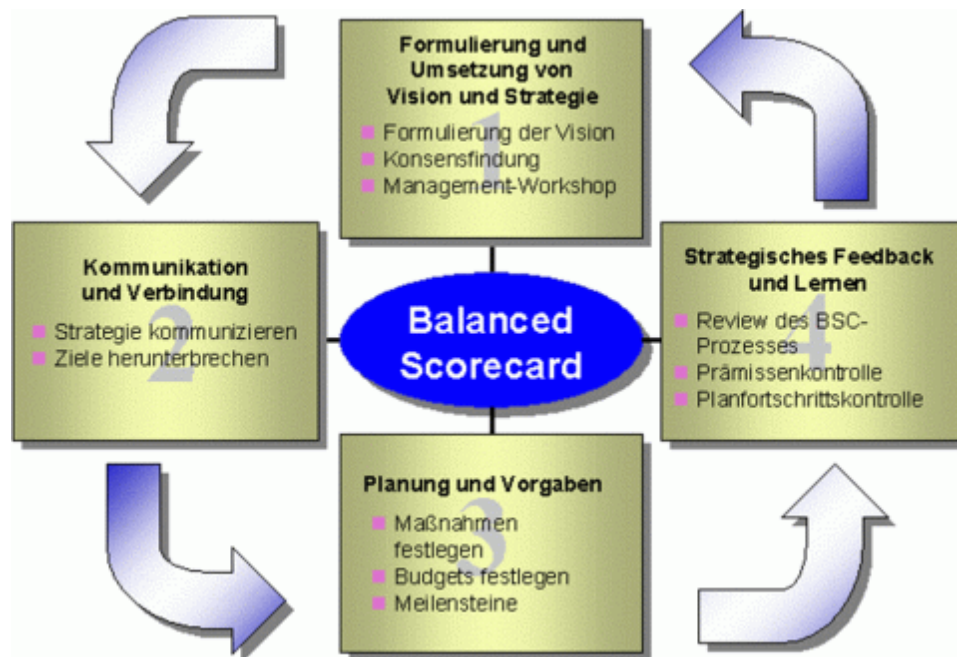


Abbildung 183: Die BSC als Handlungsrahmen für das strategische Management

Der strategische Managementprozess im Rahmen des Balanced Scorecard-Ansatz besteht aus zwei Phasen:

1. In der ersten Phase muss im Rahmen einer strategischen Analyse die Strategie des Unternehmens gefunden werden. Im Rahmen der strategischen Analyse werden alle wettbewerbsrelevanten Informationen erfasst. Ziel der Analyse ist es, Trends, Chancen und Risiken der Entwicklung des Unternehmensumfelds zu identifizieren und zu bewerten sowie unternehmenseigene Kernkompetenzen zu erfassen. Am Ende dieser Phase wird die individuelle Unternehmensstrategie definiert.
2. Die Umsetzung der Unternehmensstrategie findet in der zweiten Phase statt. Für diese Phase bietet die BSC-Methodik Unterstützungspotential. Die Unternehmensstrategie kann anhand von Teilstrategien (z. B. geschäftsfeldspezifische Strategien) verfeinert werden. Daraus leiten sich wiederum strategische Ziele ab. Die strategischen Ziele werden anhand von Vorgaben für bestimmte Maßgrößen konkretisiert. Wie die Zielvorgaben erreicht werden sollen, wird durch ein Aktionsprogramm in Form von Handlungsanweisungen bestimmt. Das Aktionsprogramm wird in der Unternehmensplanung auf unterschiedliche Unternehmensbereiche bzw. Abteilungen „heruntergebrochen“. Durch eine Budgetierung können Aktionen weiter konkretisiert werden. Mit Hilfe einzelner strategiebezogener Scorecards wird die Zielerreichung anhand der generierten Kennzahlen gemessen. Aufbauend auf den definierten Scorecards findet ein Review-Prozess statt, indem auf der Grundlage möglicher Abweichungen weitere Maßnahmen festgelegt werden oder die Strategie „überarbeitet“ wird.

9.2.2.1 Formulierung und Übersetzen der Vision und der Strategie

Die Strategie eines Unternehmens resultiert aus der Unternehmensvision.

Die Vision stellt ein übergeordnetes Leitbild für das betrachtete Unternehmen dar, sie ist als pauschale, aber sehr plakative Aussage wenig konkret, gilt aber als oberste Handlungsmaxime und stellt so eine Art Kurzfassung der Unternehmensphilosophie dar.

Für unterschiedliche strategische Geschäftseinheiten eines Unternehmens werden entsprechende Strategien festgelegt. Diese Strategien müssen sich am Erfolgsziel des Unternehmens orientieren. Daher ist es notwendig, dass zu Beginn der Einführung einer BSC das Top-Management des Unternehmens in einem Workshop die Vision und die daraus resultierenden Strategien für die strategischen Geschäftseinheiten bestimmt. In den meisten Fällen wird das Ergebnis eine finanzielle Zielsetzung (finanzielle Perspektive) sein. Der Schwerpunkt kann unterschiedlich sein: Mögliche Ziele sind bestimmte Werte für Kapitalrentabilitäten, ROI, Shareholder Value, Umsatzerlöse oder Cash Flow. Die finanziellen Ziele müssen durch ein spezifisches Verhalten auf dem zu bearbeitenden Markt realisiert werden (Kundenperspektive) Daher werden nach der Zielsetzung die entsprechenden Markt- und Kundensegmente ausgewählt. Auch für diese Kundenperspektive werden strategische Ziele definiert und die relevanten Kennzahlen erfasst. Sie können sich z. B. auf Marktanteile oder Wachstumsraten bestimmter Kundensegmente beziehen.

Zur Realisierung spezifischer marktorientierter Strategien sind entsprechende Unternehmensressourcen notwendig. Bei der Festlegung einer Balanced Scorecard werden die Ressourcen auf zwei Arten erfasst:

1. Nachdem die Finanz- und Kundenziele festgelegt sind, werden die bedeutenden Geschäftsprozesse (Prozessperspektive) betrachtet und Zielsetzungen festgelegt sowie Maßnahmen und Kennzahlen generiert. Im Mittelpunkt der Betrachtung stehen hier in der Regel Prozesszeiten und -kosten.
2. Im Rahmen der so genannten Lern- und Entwicklungsperspektive werden aus den strategischen Zielvorgaben der Finanzperspektive, Kundenperspektive und Prozessperspektive die strategischen Ziele für Personalentwicklung, Informationstechnologie und Innovationen abgeleitet.

Alle definierten strategischen Ziele stehen durch diese Vorgehensweise bei der Definition einer Balanced Scorecard in einem gegenseitigen Wirkungsgefüge, das als Ursache-Wirkungs-Kette beschrieben wird.

Wichtig für diesen ersten Schritt ist, dass auf Managementebene ein allgemeiner Konsens über die Vision und die abgeleiteten Strategien und strategischen Zielen gefunden wird.

9.2.2.1.1 Die Standardperspektiven einer Balanced Scorecard

Kaplan und Norton schlagen vier Standardperspektiven für den Aufbau einer Balanced Scorecard vor:

1. Finanzielle Perspektive - Erwartungen der Shareholder: „Wie wirkt sich die Strategie auf die Finanzen aus?“
2. Kundenperspektive - Erwartungen der Kunden: „Wie positionieren wir uns auf den Zielmärkten?“
3. Prozessperspektive - Anforderungen an Prozesse: „Welche Prozesse sind strategisch bedeutend?“
4. Lernperspektive - Anforderungen an das organisatorische Lernen und Innovationen: „Wie entwickeln wir uns zu einer lernenden Organisation? Wie fördern wir das Wachstum?“

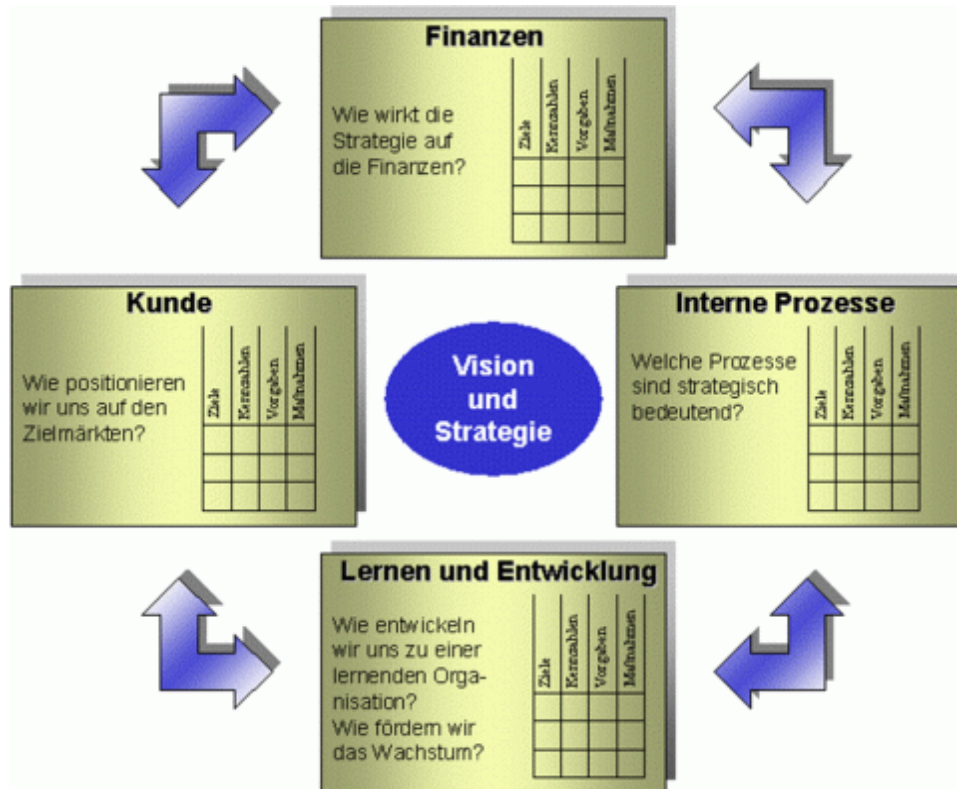


Abbildung 184: Die Perspektiven der BSC

Die Standardperspektiven bilden eine implizite Logik, die bei der Strategieumsetzung und der Formulierung von konkreten Ursache-Wirkungsbeziehungen hilfreich ist. Es besteht jedoch grundsätzlich die Möglichkeit, weitere für ein Unternehmen relevante Perspektiven zu definieren (z. B. Umweltperspektive), die strategierelevant sind und deren Ziele und Kennzahlen in Beziehung zu den anderen Perspektiven stehen können.

9.2.2.1.2 Die Ursache-Wirkungs-Kette

Die Ursache-Wirkungs-Kette verdeutlicht die Ursache-Wirkungsbeziehungen zwischen den einzelnen Teilzielen einer Strategie, also über die definierten Perspektiven der Kennzahlen hinweg. D. h., sie beschreibt, wie die Ziele der Perspektiven erreicht werden können. Dabei bilden die Perspektiven selbst sozusagen die Rahmenbedingung für die Ursache-Wirkungs-Kette („Kraftfeld des Unternehmens“). Man geht davon aus, dass Verbesserungen im Umfeld der Lern- und Entwicklungsperspektive einen direkten und positiven Einfluss auf Ziele und Kennzahlen der internen Prozessperspektive haben. Weiterhin wirken sich die Entwicklungen der Prozessperspektive auch positiv auf die Ziele und Kennzahlen der Kundenperspektive aus, was letztlich zu einer Verbesserung der Finanzziele führt.

Während des Aufbaus eines unternehmensweiten Balanced Scorecard-Systems werden für jede Strategie Ziele und Kennzahlen definiert und deren Wirkungsbeziehungen innerhalb der Perspektiven sowie perspektivenübergreifend dargestellt. Man geht von den Zielen und Kennzahlen der finanziellen Perspektive aus und sucht entsprechende „positive Verursacher“ (Leistungstreiber) in der Kundenperspektive. Diese werden dann über die interne Prozessperspektive bis auf die Lern- und Entwicklungsperspektive heruntergebrochen. Diese Vorgehensweise ermöglicht es, Strategien eines Unternehmens oder einer Division in Teilziele und Kennzahlen aufzuspalten und auch entsprechende operative Maßnahmen abzuleiten.

Durch die Festlegung der Ursache-Wirkungs-Kette und die Definition der Kennzahlen ergibt sich implizit ein Gleichgewicht zwischen Frühindikatoren und Spätindikatoren.

9.2.2.1.3 Definition der Früh- und Spätindikatoren

Durch den Aufbau von Ursache-Wirkungs-Ketten wird auch ein zeitlicher Zusammenhang zwischen den einzelnen Zielen und Kennzahlen vorgegeben. So gehören den Perspektiven **Lern- und Entwicklungsperspektive** und **Prozessperspektive** in der Regel Leistungstreiber, sog. Frühindikatoren, an, die den Wirkungen der Kundenperspektive und der finanziellen Perspektive zeitlich vorangehen. In der Kunden- und finanziellen Perspektive sind die meisten der Kennzahlen Ergebniskennzahlen, die den Geschäftserfolg messen; sie sind als sog. Spätindikatoren in die Vergangenheit gerichtet.

Es sollten in jeder Perspektive der Balanced Scorecard sowohl Früh- als auch Spätindikatoren definiert sein, damit der Zusammenhang zwischen den Maßnahmen und dem Erreichen der Ziele dargestellt werden kann. Durch die Frühindikatoren können frühzeitig Zielabweichungen erkannt werden.

9.2.2.2 Kommunikation und Ableiten weiterer Scorecards

Die Wirksamkeit einer Unternehmensstrategie und ihrer Umsetzung anhand des Balanced Scorecard-Konzepts ist unter anderem abhängig von der unternehmensweiten Bekanntheit und Akzeptanz der Unternehmens-Scorecard. Aus diesem Grund ist es überaus wichtig, dass die Unternehmensstrategien und die Unternehmens-Scorecard durch ein umfassendes Kommunikationsprogramm auf allen Hierarchiestufen des Unternehmens publiziert werden.

Aus der übergeordneten Unternehmensstrategie (Unternehmens-Scorecard) werden mit Hilfe eines Top-Down-Ansatzes Zielvorgaben für die unteren Hierarchiestufen des Unternehmens abgeleitet sowie übergeordnete strategische Ziele an die bereichsspezifischen Ziele angepasst. Für sie werden ebenfalls Kennzahlen generiert, die den Grad der Zielerreichung messen. Um die Ziele erreichen zu können, sind in den Funktionsbereichen eines Unternehmens verschiedene Maßnahmen notwendig, die ebenfalls in den Scorecards der darunter liegenden Hierarchiestufen erfasst werden. Alle Zielvorgaben müssen auf der Basis langfristiger strategischer Überlegungen gemacht werden und dürfen nicht auf kurzfristige Ergebnisse ausgerichtet sein, die nicht im Einklang mit der übergeordneten Strategie stehen. Ein Beispiel für ein kurzfristiges Ergebnis wäre eine Kostenreduzierungen aufgrund einer kurzfristigen Erhöhung der Produktionsintensität.

9.2.2.3 Planung und Vorgaben

Die BSC kann dazu genutzt werden, den Prozess der Strategieumsetzung auf die unterschiedlichen Hierarchiestufen der Unternehmung mit dem Budgetierungsprozess zu verknüpfen. Zweck der Kopplung ist die Ausrichtung aller Ressourcen auf die Unternehmensstrategie.

In vier Schritten wird die BSC in den langfristigen strategischen Planungs- und Budgetierungsprozess integriert.

1. Herausfordernde Zielvorgaben werden gemacht, die allgemein kommuniziert und von den Mitarbeitern akzeptiert werden. Durch die Ursache-Wirkungsbeziehungen zwischen den Kennzahlen werden die relevanten Kennzahlen identifiziert, die auf die Ergebniswerte der Kunden- und Finanzperspektive bezogen sind.
 2. Die strategischen Maßnahmen müssen genau an den Kennzahlen ansetzen, deren aktueller Wert sich am deutlichsten von den Vorgabewerten unterscheidet. Dieses Vorgehen bündelt langfristig die Kapitalinvestitionen und Aktionsprogramme konsequent auf strategisch bedeutsame Zielwerte.
 3. Kritische, unternehmensweite Initiativen werden identifiziert, so dass Synergieeffekte mit den strategischen Zielen realisiert werden können.
 4. Die Planwerte für die Ergebniskennzahlen werden mit der Unternehmensplanung verknüpft, indem sie an Budgetvorgaben mit unterschiedlichen Fristigkeiten (5-, 3- und 1-Jahresplanung) gekoppelt werden.

9.2.2.4 Strategisches Lernen und Feedback

Das Balanced Scorecard-Konzept trägt dazu bei, dass im Unternehmen strategisches Lernen stattfindet und Feedback zur Strategieerreichung gegeben wird.

Das strategische Lernen bezieht sich auf jeden einzelnen Mitarbeiter sowie auf das gesamte Unternehmen als lernende Organisation. Mitarbeiter sollen die Strategie kennen und selbstständig dafür sorgen, dass sie ihre Handlungen an ihr ausrichten. Über einen definierten Feedback-Prozess werden Leistungsdaten gesammelt, an denen die Strategieerreichung gemessen wird. Auf diesem Weg findet eine Prämissenkontrolle statt, indem Hypothesen über die vorab getroffenen Ursache-Wirkungsbeziehungen strategischer Ziele geprüft werden.

Auf diesem Weg kommt es zum so genannten Double-Loop-Lernen, wenn die strategischen Zielvorgaben in einem Bottom-Up-Prozess einer kritischen Revision unterzogen und danach neue Ursache-Wirkungsbeziehungen generiert werden, die aufgrund einer ständig erweiterten Informationsbasis zu einer effizienteren Strategieverfolgung führen.

Dem Feedback kommt dabei eine große Bedeutung zu: es repräsentiert den Prozess, der dem „Herunterbrechen“ von BSCs entgegen läuft. Das Feedback beinhaltet nicht nur Rückmeldungen von Kennzahlen in Form von Planfortschrittszahlen, sondern auch Vorschlägen für weitere Maßnahmen sowie neue Wirkzusammenhänge. Es weist damit auf eventuelle Inkonsistenzen in der Strategieverfolgung hin.

9.2.3 Vorteile und Nutzen der Balanced Scorecard

Der Vorteil der Balanced Scorecard-Methode liegt in der konsequenten strategieorientierten Unternehmensführung, welche die untersten Unternehmensebenen und die strategiekonforme Ableitung operativer Maßnahmen auf diesen Unternehmensebenen umfasst. Alle Unternehmensressourcen und alle Mitarbeiter werden auf die Unternehmensstrategie ausgerichtet. Die Unternehmensstrategie wird durch Balanced Scorecards konsequent kommuniziert und kontrolliert.

Die Balanced Scorecard-Methode überwindet die Nachteile traditioneller Kennzahlensysteme zur Unternehmenssteuerung:

- Keine Beschränkung auf finanzielle Kennzahlen
- Keine reine Vergangenheitsorientierung
- Berücksichtigung auch qualitativer Daten
- Komplexitätsreduktion und Konsensfindung
- Strategische Ziele als Ausgangspunkt des Balanced Scorecard-Ansatzes
- Orientierung an Führungsebenen, nicht an vorhandenen Daten
- Beschleunigung notwendiger Veränderungen

Die Balanced Scorecard-Methode eignet sich hervorragend zur strategieorientierten Unternehmensführung, da sie die folgenden Barrieren überwindet:

- Konkretisierungsbarriere: Bei klassischen Managementansätzen sind Vision und Strategie Wunschformulierungen und die Ableitung konkreter (operativer) Aktionsformulierungen bleibt offen.
- Visionsbarriere: Die Strategie wird oftmals nicht von jenen Mitarbeitern verstanden, die sie umsetzen müssen.
- Commitment-Barriere: Strategien sind nicht mit konkreten Abteilungs- oder Individualzielen verbunden.
- Implementierungsbarriere: Das Berichtswesen ist an operativ-monetären Zielen, statt an strategischen Zielen ausgerichtet.
- Operative Barriere: Der Budgetierungsprozess ist vom strategischen Planungsprozess getrennt.

9.3 Entwicklung einer Balanced Scorecard mit ARIS BSC

9.3.1 Begriffe und Abkürzungen

Vision: Eine Vision gibt die zukünftige strategische Stoßrichtung und den Auftrag eines Unternehmens an und wird durch verschiedene Strategien realisiert.

Strategie: Ausgehend von der Vision werden Strategien entwickelt. Die Strategien können in spezifische Teilstrategien aufgespalten werden.

Strategisches Ziel: Jede Strategie besteht aus einer bestimmten Anzahl strategischer Ziele. Die strategischen Ziele werden in der Regel vom verantwortlichen Management in Workshops festgelegt. Zwischen den Zielen bestehen eine Ursache-Wirkungsbeziehung und eine zeitliche Logik.

Erfolgsfaktor: Ein Erfolgsfaktor ist ein für die Geschäftsabwicklung des Unternehmens erfolgskritischer Einflussfaktor, der in einem Wirkungsgefüge zu anderen Erfolgsfaktoren stehen kann und diese in eine bestimmte Wirkungsrichtung mit einem bestimmten Wirkungsgrad beeinflusst.

Sicht/Perspektive: Eine Perspektive ist die Konkretisierung einer bestimmten Sicht auf das Unternehmen. Grundsätzlich sollte die Auswahl von Perspektiven folgende Sichtweisen auf ein Unternehmen beinhalten: humanorientierte Sicht, interne Sicht, prozessorientierte Sicht und externe Sicht.

Kennzahl: Jedem strategischen Ziel oder Erfolgsfaktor werden Kennzahlen zur Leistungsmessung und Zielerreichung zugeordnet.

Ist-Wert: Der aktuelle Wert den eine Kennzahl, ein strategisches Ziel oder ein Erfolgsfaktor annehmen kann. Aus dem Ist-Wert kann durch einen Plan-Ist-Vergleich der aktuelle Zielerreichungsgrad abgeleitet werden.

Plan-Wert: Der für diesen Zeitraum geplante und zu erreichende Plan-Wert einer Kennzahl, eines strategischen Zieles oder eines Erfolgsfaktors. Bei strategischen Zielen handelt es sich in der Regel um einen Prozentwert, der sich aus der gewichteten Zielerreichung einzelner Kennzahlen ergibt. Der Plan-Wert kann auf einzelne Perioden heruntergebrochen werden und kann dann auch periodisch bedingte Schwankungen einbeziehen. Der Plan-Wert entspricht einem Zielerreichungsgrad von 100 %.

Ziel-Wert: Der für einen zukünftigen Ziel-Zeitraum geplante und zu erreichende Wert. Er wird in der Regel zum Plan-Wert, wenn der Ziel-Zeitraum erreicht ist.

Minimalwert: Der kleinste Wert, den ein Ziel, ein kritischer Erfolgsfaktor oder eine Kennzahl annehmen kann. In ARIS wird als Vorgabewert für den Minimalwert 0 vorgegeben. Er kann auch geändert werden.

Maximalwert: Der größte Wert (Obergrenze), den ein strategisches Ziel, ein Erfolgsfaktor oder eine Kennzahl annehmen kann. Er wird in ARIS in der Regel nur dazu verwendet, um die Darstellung mehrerer Kennzahlen zu normieren und vergleichbar zu machen.

Warngrenze: Die Warngrenze entspricht dem Plan-Wert, d. h., der Plan-Wert entspricht der Schwelle, an welcher der Wert einer Kennzahl, eines strategischen Ziels oder eines Erfolgsfaktors unter eine gewünschte Vorgabe fällt.

Toleranzbereich: Der Toleranzbereich ist die negative Abweichung vom Plan-Wert, die für ein strategisches Ziel, eine Kennzahl oder einen Erfolgsfaktor noch als tolerierbar gilt.

Alarmgrenze: Die Alarmgrenze entspricht dem Plan-Wert abzüglich des Toleranzbereichs. Alle Werte eines strategischen Ziels, einer Kennzahl oder eines Erfolgsfaktors, die unterhalb der Alarmgrenze liegen, werden nicht mehr toleriert.

Maßnahme: Eine Maßnahme wirkt auf ein Ziel oder auf mehrere strategische Ziele. Maßnahmen haben in der Regel einen Verantwortlichen, ein Anfangs- und einen Endzeitpunkt, Ressourcen etc.

Indikatortyp: Eine Kennzahl kann vom Indikatortyp **Frühindikator** oder vom Indikatortyp **Spätindikator** sein. Ein Frühindikator misst einen Leistungstreiber und ist eine in die Zukunft gerichtete Kennzahl. Ein Spätindikator misst ein Ergebnis (Ergebniskennzahl) und ist retrospektiv. Finanzwirtschaftliche Ziele sind in der Regel Spätindikatoren, Ziele der Prozess-, Lern- und Entwicklungsperspektive sind zunehmend Frühindikatoren. Zusätzlich sollten auch Spätindikatoren innerhalb der Perspektiven erfasst werden, um die Ursache-Wirkungsbeziehung zwischen den Kennzahlen darzustellen.

Datenquelle: Jede Kennzahl besitzt eine Datenquelle, aus der sie in das Balanced Scorecard-System extrahiert wird.

9.3.2 Aufbau der Balanced Scorecards mit ARIS BSC

9.3.2.1 Perspektiven festlegen

Zu Beginn eines Balanced Scorecard-Projektes sollten im Rahmen der strategischen Planung die Perspektiven festgelegt werden. Diese Perspektiven werden für alle Scorecards eines Unternehmens verwendet.

Dafür steht der Objekttyp **Perspektive** zur Verfügung. Perspektiven können im BSC-Ursache-Wirkungsdiagramm modelliert werden.

Perspektiven werden mit Hilfe der Kante **wird beeinflusst von** verbunden. Die Festlegung der Reihenfolge ist methodisch nicht notwendig. Durch die logische Festlegung der Perspektivenstruktur (d. h. die Reihenfolge der einzelnen Perspektiven) wird allerdings eine automatische Modellgenerierung wesentlich vereinfacht und der logische Aufbau der Ursache-Wirkungs-Kette verdeutlicht.

9.3.2.2 Struktur des Balanced Scorecard-Systems festlegen

Den Rahmen für den Aufbau des Balanced Scorecard-Systems bilden folgende Modelle:

- Organigramm

Ein Balanced Scorecard-System kann in Anlehnung an die Organisationsstruktur eines Unternehmens aufgebaut werden. Dabei können den Objekten des Organigramms beliebig viele BSC-Ursache-Wirkungsdiagramme hinterlegt werden (z. B. zur Variantenbildung, Historienverwaltung etc.). Diese Hinterlegung ordnet die für die Strategieumsetzung erforderlichen Ziele den entsprechenden Organisationseinheiten zu. Das Balanced Scorecard-System eines Unternehmens kann somit durch die Orientierung am Organigramm von der obersten Unternehmensebene bis auf die Ebene einzelner Abteilungen oder Mitarbeiter heruntergebrochen werden.

- Gliederungsmodell

Möchte ein Unternehmen den Aufbau des Balanced Scorecard-Systems nicht am Organigramm, sondern beispielsweise anhand von strategischen Geschäftsfeldern festlegen, bietet sich das Gliederungsmodell als Ausgangsmodell für die Balanced Scorecard-Struktur an. In diesem Fall können beliebig viele BSC-Ursache-Wirkungsdiagramme einem Gliederungsobjekt hinterlegt werden.

- Wertschöpfungskettendiagramm oder Funktionsbaum

Da sich die Balanced Scorecard als Instrument zum Performance Management und Performance Measurement von Unternehmen versteht, kann in ARIS das Balanced Scorecard-System auch wertschöpfungsorientiert aufgebaut werden. Dazu können das Wertschöpfungskettendiagramm oder der Funktionsbaum genutzt werden.

9.3.2.3 Ursache-Wirkungs-Beziehungen festlegen

Den Objekten des Organigramms, den Funktionen oder den Gliederungselementen im Gliederungsmodell, die den Aufbau des unternehmensweiten Balanced Scorecard-Systems festlegen, werden BSC-Ursache-Wirkungsdiagramme hinterlegt.

Im BSC-Ursache-Wirkungsdiagramm werden die zur Strategieumsetzung festzulegenden strategischen Ziele und Erfolgsfaktoren definiert und deren gegenseitige Beeinflussung über die verschiedenen Perspektiven der Balanced Scorecard dargestellt. Dazu werden zuerst die einzelnen Perspektiven als Objekte in ARIS angelegt, die in allen Ursache-Wirkungsketten des Unternehmens genutzt werden.

Beim BSC-Ursache-Wirkungsdiagramm handelt es sich um ein Lane-Modell, das in der Spalte **relevante Perspektiven** die Ablage der Objekttypen **Perspektive** und in der Zeile **Strategie** die Ablage der Objekttypen **Strategie** erlaubt. In den Ursache-Wirkung-Spalten können die Objekttypen **Ziel**, **Erfolgsfaktor** und **Kennzahl** modelliert werden. In der Ursache-Wirkung-Spalte werden die Wirkungsstärke der strategischen Ziele und ihre Erfolgsfaktoren beschrieben. Die jeweiligen Pfeile werden mit zunehmender Einflussstärke dicker dargestellt.

Im BSC-Ursache-Wirkungsdiagramm können weiterhin Erfolgsfaktoren für eine Balanced Scorecard definiert werden. Durch die Verwendung des Objekttyps **Erfolgsfaktoren** lassen sich vom Unternehmen nicht direkt beeinflussbare Faktoren und deren Kennzahlen in die Balanced Scorecard mit aufnehmen, z. B. Marktwachstum.

Das BSC-Ursache-Wirkungsdiagramm finden Sie auf der Fachkonzeptebene der Steuerungssicht.

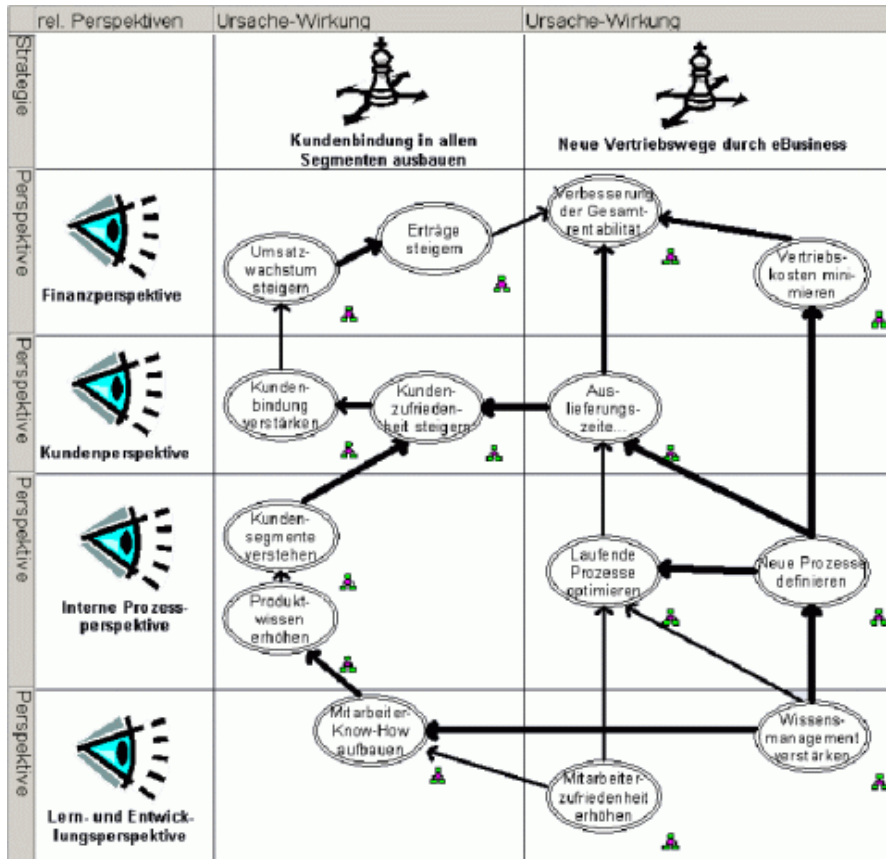


Abbildung 185: BSC-Ursache-Wirkungsdiagramm

Die Objekttypen **Ziel** und **Erfolgsfaktor** in der Modellierungsspalte des BSC-Ursache-Wirkungsdiagramms können ausschließlich mit einem BSC-Kennzahlenzuordnungsdiagramm hinterlegt werden. Folgende Symbole werden im BSC-Ursache-Wirkungsdiagramm verwendet:

Symbol	Symbolname	Objekttyp
	Perspektive	Perspektive
	Strategie	Strategy
	Strategisches Ziel	Ziel
	Erfolgsfaktor	Erfolgsfaktor
	Kennzahleninstanz	Kennzahleninstanz

9.3.2.4 Maßnahmen und Kennzahlen zur Zielkontrolle festlegen

Nach der Definition der strategischen Ziele und der Erfolgsfaktoren im BSC-Ursache-Wirkungsdiagramm wird den einzelnen Objekten je ein BSC-Kennzahlenzuordnungsdiagramm hinterlegt. Darin werden den Zielen und Erfolgsfaktoren die Kennzahlen zugeordnet, die als Maßstab für das jeweilige Ziel oder den Erfolgsfaktor dienen sollen. Diese Kennzahlen werden anschließend ermittelt und kanalisiert. Werden einem Objekt mehrere Kennzahlen zugeordnet, kann mit der Kante des Typs **wird gemessen durch/misst** eine Gewichtung der jeweiligen Kennzahl festgelegt werden. Diese gibt an, mit welchem Gewicht die Kennzahl in den Zielerreichungsgrad des Zieles oder Erfolgsfaktors eingeht. Zur erfolgreichen Ausführung der Analyseskripte von ARIS BSC muss das Attribut **Gewichtung** bei allen Kanten des Typs **wird gemessen durch/misst** gepflegt sein.

Neben Kennzahlen können den Zielen oder Erfolgsfaktoren im BSC-Kennzahlenzuordnungsdiagramm auch Objekte für die Definition der Datenherkunft zugeordnet werden, z. B. Entitytyp, Datei, Dokument, Datenbank, Datenträger etc. Durch diese Zuordnung wird eine Verbindung zur Data Warehouse-Methode von ARIS geschaffen. Hier kann genau definiert werden, mit welchen Datenelementen der Data Warehouse-Methode die jeweilige Balanced Scorecard-Kennzahl verbunden ist.

Aufgrund der Möglichkeit, mit dem Kantentyp **wirkt auf/wird beeinflusst von** auch das Wirkungsgefüge von Kennzahlen zu beschreiben, können die Wirkungsweisen von Frühindikatoren auf Spätindikatoren im BSC-Kennzahlenzuordnungsdiagramm beschrieben werden. Wir empfehlen dafür ein zusätzliches BSC-Kennzahlenzuordnungsdiagramm zu entwerfen, in dem nicht die Zuordnung einzelner Kennzahlen zu Zielen oder Erfolgsfaktoren zueinander im Vordergrund steht, sondern das Zusammenführen und die Definition der genutzten Kennzahlen aus den anderen BSC-Kennzahlenzuordnungsdiagrammen und die Beschreibung ihrer Wirkungen.

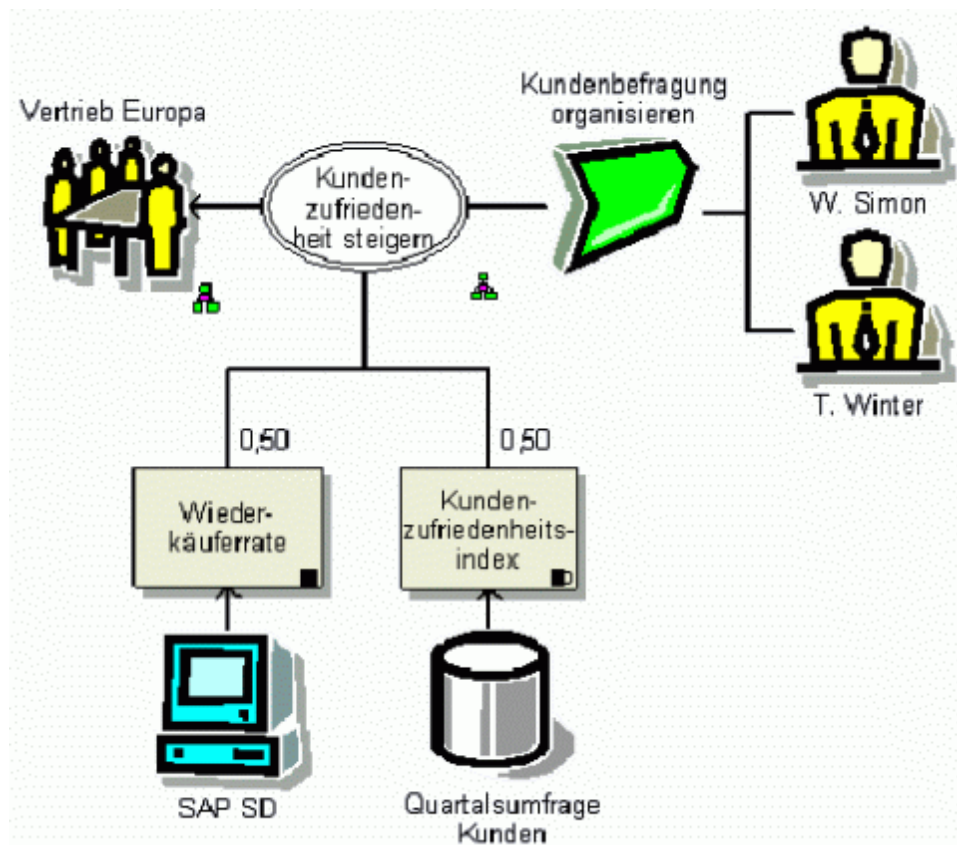

















Abbildung 186: BSC-Kennzahlenzuordnungsdiagramm

Das BSC-Kennzahlenzuordnungsdiagramm finden Sie auf der Fachkonzeptebene der Prozesssicht.

Folgende Symbole werden im BSC-Kennzahlenzuordnungsdiagramm verwendet:

Symbol	Symbolname	Objekttyp
	Strategisches Ziel	Ziel
	Erfolgsfaktor	Erfolgsfaktor
	Kennzahlinstanz	Kennzahlinstanz
	Maßnahme	Aufgabe
	Organisationseinheit	Organisationseinheit

Symbol	Symbolname	Objekttyp
	Organisationseinheitstyp	Organisationseinheitstyp
	Stelle	Stelle
	Person (m) Person (w)	Person
	Personentyp	Personentyp
	Gruppe	Gruppe
	Computer	Anwendungssystemtyp
	Fachbegriff	Fachbegriff
	Entitytyp	Entitytyp
	Beziehungstyp	Beziehungstyp
	ERM-Attribut	ERM-Attribut
	Cluster	Cluster/Datenmodell
	Wissenskategorie	Wissenskategorie
	Dokumentiertes Wissen	Dokumentiertes Wissen
	Class	Class

Symbol	Symbolname	Objekttyp
	Datei, Dokument, Telefon, Internet, Akte, Buch, E-Mail, CD-ROM, Karteikasten, Festplatte, Brief, Diskette, Magnetband	Informationsträger

9.3.2.5 Kennzahlen und deren Zusammenhänge beschreiben

Nach der Festlegung der Kennzahlen zur Messung strategischer Ziele und kritischer Erfolgsfaktoren können diese Kennzahlen und ihre Zusammensetzung durch den Modelltyp **Kennzahlenbaum** detaillierter definiert werden.

In einem Kennzahlenbaum werden verschiedenen Kennzahlen mit Hilfe des Kantentyps **beeinflusst** hierarchisch strukturiert. Für diese Kanten kann das Attribut **Gewichtung** gepflegt werden, sodass innerhalb eines Kennzahlenbaums die Berechnung einer Gesamtkennzahl anhand von Gewichtungen möglich wird. Der Kennzahlenbaum wird der Kennzahlinstanz hinterlegt, die die Gesamtkennzahl repräsentiert. Die Hinterlegung wird durch die BSC-Auswertungen **Management View erzeugen** und **Plan-Ist-Vergleich durchführen** berücksichtigt.

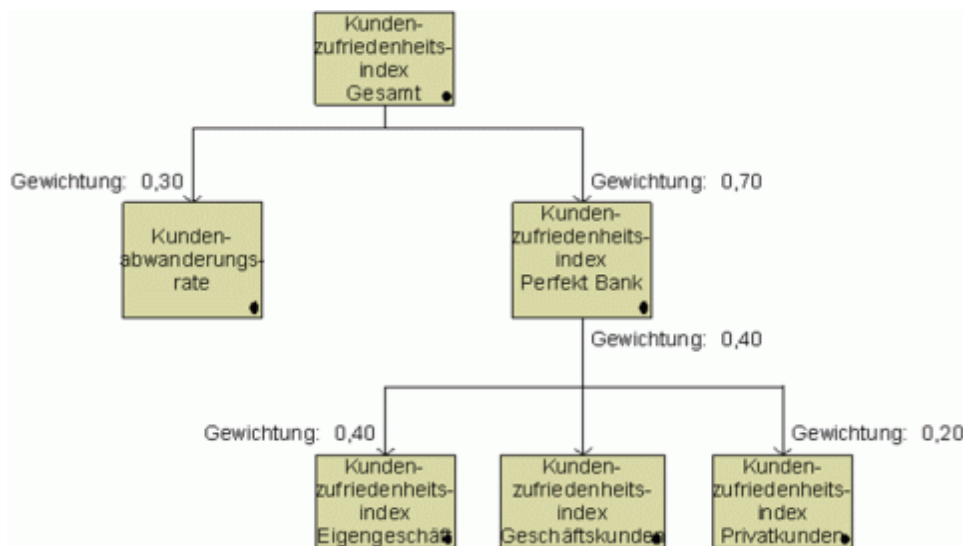


Abbildung 187: Kennzahlenbaum

Der Modelltyp **Kennzahlenbaum** verwendet lediglich den Objekttyp **Kennzahlinstanz**.

Den Kennzahlenbaum finden Sie auf der Fachkonzeptebene in der Datensicht.

9.3.3 Beziehungen zu anderen Modellen

Die Objekte der Balanced Scorecard-Methode können zu folgenden ARIS-Modellen in Beziehung stehen:

- EPK
- Zieldiagramm
- DW-Struktur
- DW-Transformation

10 eBusiness Scenario Diagram

10.1 Einleitung

Die reibungslose Abfolge unternehmensübergreifender Geschäftsprozesse gewinnt zunehmend an Bedeutung. Dabei steht auf der einen Seite der Ablauf bestimmter Vorgänge an den Schnittstellen zwischen den Unternehmen, auf der anderen Seite die Schnittstelle zwischen Unternehmen und ihren Kunden im Blickpunkt. Die Kontakte sollen übersichtlich, schnell, einheitlich und direkt erfolgen.

Größte Relevanz besitzt auch das schnelle Finden von adäquaten Geschäftspartnern - aus Unternehmenssicht - und Anbietern - aus Verbrauchersicht. Eine optimale Gestaltung dieser Prozesse bedeutet einen Wettbewerbsvorteil. Die geeignete Plattform zur Unterstützung der multilateralen Beziehungen ist das Internet. Da die Vorgänge innerhalb des oben beschriebenen Umfeldes sehr vielschichtig sind, muss definiert werden, was unter E-Business verstanden wird. E-Business ist ein Oberbegriff für die Unterstützung der Geschäftstätigkeit eines Unternehmens durch Informations- und Kommunikationstechnologien. Es umfasst die Unterstützung der Beziehungen und Prozesse zwischen Geschäftspartnern, Mitarbeitern und Kunden durch elektronische Medien (Herrmans, Sauter, 1999).

Somit kann E-Business die Erstellung einer Web-Site zur Unternehmenspräsentation, der Kauf eines Artikels via Internet, ein hoch komplexes Projekt zweier Unternehmen oder auch vielschichtige Beziehungen unter beliebig vielen Partnern bedeuten, die auf einem Marktplatz zusammenkommen.

Daraus leiten sich folgende Benennungen ab:

B2B (Business to Business)

Business to Business bezeichnet die Abwicklung von Transaktionen zwischen Unternehmen. Dies wird beispielsweise durch die Vernetzung der Supply Chains der Unternehmen ermöglicht.

B2C (Business to Consumer)

Business to Consumer bezeichnet die Abwicklung von Transaktionen zwischen Unternehmen und Ihren Kunden. Beispiel hierfür sind die Einkäufe von Kunden in Online-Shops.

B2A/C2A (Business/Consumer to Administration)

Business/Consumer to Administration bezeichnet alle Transaktionen zwischen Unternehmen oder Einzelpersonen und der öffentlichen Verwaltung. Besonders im Bereich der Kontakte zwischen Unternehmen und Verwaltung wird ein enormes Potential zur Kosteneinsparung gesehen.

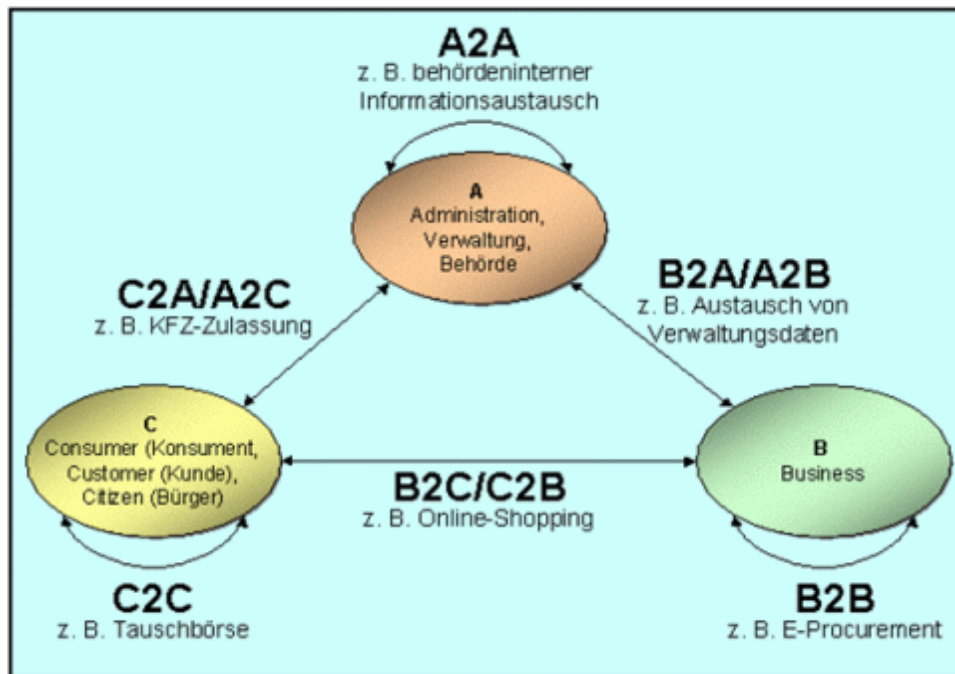


Abbildung 188: Transaktionsmöglichkeiten im E-Business

Neben der Unterscheidung in verschiedene Partner ist auch eine Unterscheidung im Umfang der Beziehungen zwischen den verschiedenen Partnern möglich: one to one, one to many, many to many. Große Bedeutung hat vor allem das Marktplatz-Szenario erlangt.

Marktplätze

Elektronische Marktplätze sind virtuelle Plätze, auf denen eine (beliebige) Zahl Käufer und Verkäufer Waren und Dienstleistungen (offen) handeln und Informationen tauschen.

Zur Unterstützung dieser denkbaren Szenarien wurde das eBusiness Scenario Diagram entwickelt. Gemeinsam mit anderen Methoden und verschiedenen Komponenten, die ARIS zur Verfügung stellt, kann die Realisierung von E-Business-Projekten optimal unterstützt werden. Dieses Kapitel zu den eBusiness Scenario-Diagrammen beschreibt zunächst die Methode mit allen Objekten und Auswertungsmöglichkeiten und geht danach auf die Verbindung zu anderen Methoden ein. Abschließend wird ein Anwendungsfall vorgestellt, der die komplexen Möglichkeiten aufzeigt.

10.2 Die Methode des eBusiness Scenario Diagrams

10.2.1 Die Idee

Durch die Möglichkeit, eine Wertschöpfungskette ganzheitlich zu betrachten, d.h. vom Endkunden bis zu allen an einem Vorgang beteiligten Unternehmen, soll Optimierungspotential entwickelt werden. Ziel hierbei ist beispielsweise die Verbesserung der Supply Chain, Senkung von Beschaffungs- und Vertriebskosten oder Optimierung der Architektur des Informationssystems. Die Darstellung mittels des eBusiness Scenario Diagram ermöglicht die Visualisierung der Inhalte, die zur Erreichung der genannten Ziele untersucht werden müssen. Durch die gewählte Form der Spaltendarstellung werden die Schnittstellen zwischen unterschiedlichen Prozesspartnern abstrahiert über die Spaltenränder abgebildet. Verschiedene Reporte ergänzen die Modelle und bieten wichtige Analysemöglichkeiten.

10.2.2 Das Modell und seine Objekte

Die im Modell betrachteten Wirtschaftssubjekte werden in der Kopfzeile eingeordnet und als **Business Participant** bezeichnet. Sie entstammen der Organisationssicht und können mit Organigrammen hinterlegt werden, die beispielsweise die Unternehmensstruktur oder auch die Beziehungen zwischen den verschiedenen Objekten der einzelnen Spalten verdeutlichen können. Zentrale und strukturbildende Objekte des Modells sind die am Gesamtprozess beteiligten Einzelprozesse der Wirtschaftssubjekte und ihre Schnittstellen zueinander. Ein Einzelprozess, auch **Business Process**, ist ein wesentlich an unternehmensübergreifenden Kooperationen beteiligter Unternehmensprozess. Die genauere Darstellung und Analyse eines solchen Einzelprozesses erfolgt durch Hinterlegung eines Prozessmodells. Alle Abläufe eines Unternehmens werden im Modell in der Zeile unter dem Business Participant aber in der gleichen Spalte modelliert. Die unternehmensübergreifende Abstimmung bedarf auch der genauen Analyse der **Anwendungssysteme** und Hardware, die von den verschiedenen Wirtschaftsobjekten zur Unterstützung ihrer Einzelprozesse eingesetzt werden, z.B. der ERP-Systeme. Sie werden durch **Business Components** dargestellt. Zur Abstimmung dieser unterschiedlichen Komponenten ist es notwendig, die genauen Verantwortlichkeiten für die Systeme zu bestimmen. Dazu können Objekte vom Typ **Organizational Unit Type** verwendet werden. Auch die Rollen der am Prozess beteiligten Mitarbeiter können definiert werden. Sie werden im Modell als **Employee Role** bezeichnet. Eine besondere Anforderung an eBusiness-Modellierung im Allgemeinen stellt die Einbeziehung der Schnittstellen dar. Dabei kommt den Spaltenrändern eine zentrale Bedeutung zu, da sie die Schnittstellen zwischen den Prozessteilnehmern symbolisieren. Ihre Betrachtung erfolgt unter verschiedenen Gesichtspunkten.

Ein Schwerpunkt kann die Übertragung prozessspezifischer Informationen sein. Dazu dienen die **Business Documents**, die die Form eines XML- oder eines HTML-Dokuments annehmen können. Das Business Document kann mit einem Modell der Datensicht hinterlegt werden, beispielsweise mit einer Document Type Definition. Alternativ dazu kann auch der Geld- oder Güterstrom durch die Objekte **Money Transaction** bzw. **Goods Shipment** abgebildet werden.

Ein weiterer Aspekt ist von großer Bedeutung: Die Sicherheit der via Internet übermittelten Daten und besonders auch der elektronisch abgewickelten Zahlungen muss gewährleistet sein. Dazu werden verschiedene Verschlüsselungstechniken verwendet, z. B. SET (Secure Electronic Transaction) oder SSL (Secure Socket Layer). Dieser Sicherheitsaspekt wird durch die Einbeziehung des Objektes **Security Protocol** berücksichtigt. Auch für die Sicherung von Transaktionen können Verantwortliche bestimmt werden, die ebenfalls durch den **Organizational Unit Type** dargestellt werden. Zum Schwerpunkt kann auch die Analyse eines eher technischen Aspekts, nämlich die Betrachtung der technischen Ausgestaltung der Datenübertragung an den Schnittstellen bestimmt werden. Hierzu werden im Modell verschiedene Informationsträger verwendet. Die Einzelprozesse können über **Intranet**, **Extranet** oder **Internet** verbunden werden. Die Datenübertragung kann per **E-Mail** erfolgen. Als Übertragungsmedium erlangt das Handy immer größere Bedeutung.

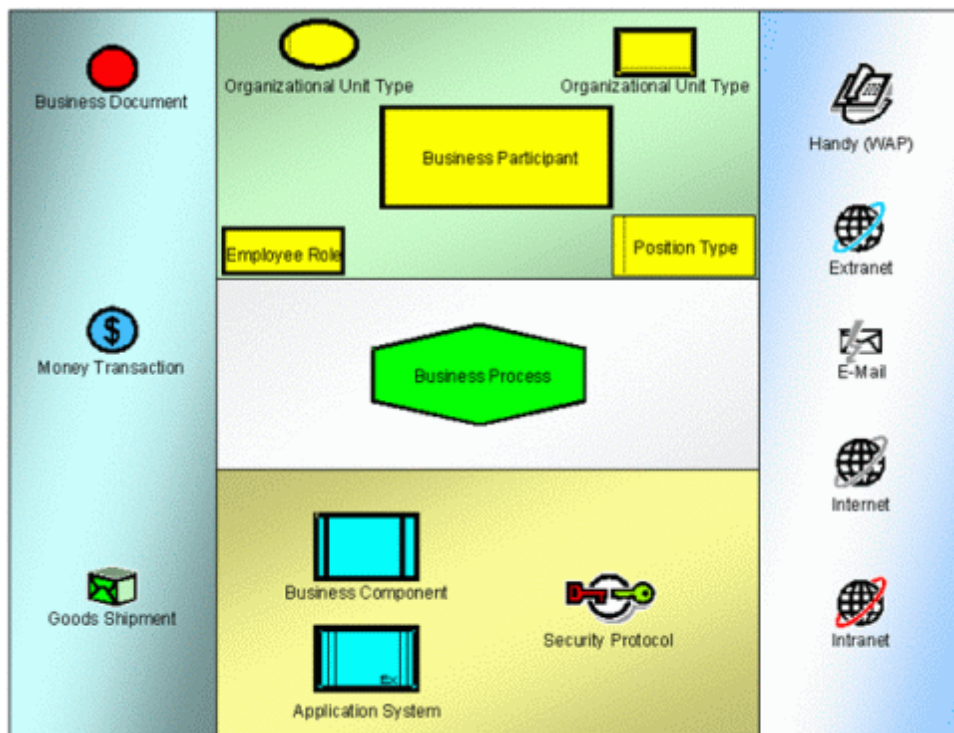


Abbildung 189: Die Objekte im eBusiness Szenario Diagramm

10.2.3 Attributgruppe Übertragungsart

Die modellierten Objekte können durch die Pflege ihrer Attribute weiter spezifiziert werden. Ein Attribut ist besonders auf die Anforderungen des E-Business zugeschnitten.

Die Attributgruppe **Übertragungsart** der Objekte **Business Document**, **Money Transaction** und **Goods Shipment** ist hervorzuheben. Durch Pflege eines der Übertragungsart-Attribute wird nicht nur der Übertragungsweg gekennzeichnet, es kann zugleich die Notwendigkeit der Sicherung der Transaktion zum Ausdruck gebracht werden. Handelt es sich beispielsweise um eine Online-Übertragung, darf die bereits angesprochene notwendige Sicherung vertraulicher Informationen und Daten nicht fehlen.

10.3 Auswertungen durch Reporte

Verschiedene Auswertungsmöglichkeiten unterstützen die Modellierung der eBusiness Scenarios. Diese Auswertungen erfolgen in Form von Reporten. Mehrere vordefinierte Auswertungen werden durch ARIS angeboten, aber auch eigene, benutzerspezifische sind denkbar. Für die eBusiness Scenarios werden die im Folgenden erläuterten Reporte ausgeliefert.

10.3.1 Überprüfung der Datensicherheit

Die Sicherheit online übertragener Daten ist einer der wichtigsten Punkte, die die Akzeptanz von E-Business beeinflussen. Persönliche Daten oder gar Zahlungen vor den Zugriffen unberechtigter Personen zu schützen, ist eine Aufgabe, die gelöst werden muss, will man das Vertrauen der Kunden und Partner nicht verlieren. Mittels eines Reports lassen sich alle ausgetauschten Leistungen (Money Transaction und Goods Shipment) und Daten (Business Documents) diesbezüglich überprüfen. Dazu wird die bereits angesprochene Attributgruppe **Übertragungsart** ausgewertet und für den Fall, dass es sich um eine Online-Übertragung handelt, überprüft, ob eine Verschlüsselung der Daten erfolgt. Somit werden eventuelle Sicherheitslücken identifiziert, aber auch veraltete Verschlüsselungsmethoden aufgezeigt und damit wird eine leichtere Aktualisierung ermöglicht.

10.3.2 Systemunterstützung

Ein zweiter wichtiger Aspekt bei der Realisierung von E-Business-Projekten ist die Abstimmung der Anwendungssysteme. Dadurch ergeben sich für Unternehmen viele Fragen. Welche Prozesse müssen durch welche Systeme unterstützt werden? Wer zeichnet sich für die Bedienung welcher Systeme verantwortlich? Wo entsteht gegebenenfalls Schulungsaufwand? Welche Anpassungen der bestehenden Systeme müssen gemacht werden? Auch hierzu lassen sich mittels eines Reports Aussagen treffen. Es werden die einzelnen Prozesse mit den dazugehörigen Systemen und den dafür Verantwortlichen aufgelistet

10.3.3 Informationsfluss

Im Gegensatz zu anderen Prozessmodellen steht bei den eBusiness Scenarios die Transaktion im Vordergrund. Besonderes Augenmerk liegt auf den Daten und Leistungen, die ausgetauscht werden. Daher werden auch hierzu Auswertungen angeboten. Wichtig ist die Frage, welche Daten und Leistungen an welcher Stelle erzeugt werden und wo sie Verwendung finden. Hierzu gibt ein Report die modellierten Daten und Leistungen sowie die jeweiligen Prozesse aus, denen sie als Input oder Output zuzurechnen sind.

10.3.4 Collaborative Business Maps

Eine spezielle Form der Modelle stellen die von der SAP verwendeten Collaborative Business Maps dar. Sie stellen die Sicht auf verschiedene Partner besonders in den Vordergrund. Es werden zwei Sichten unterschieden, die auf verschiedene Zielgruppen ausgerichtet sind. Eine „aggregated view“, die nur die Business Participants sowie die Prozesse enthält und die an das führende Management adressiert ist, und eine „detailed view“ mit den Dokumenten und Rollen der Prozessverantwortlichen für die Fachabteilungen. Die Reporte ermöglichen es jederzeit, die im eBusiness Scenario Diagram dargestellten Informationen in beide Sichten zu übertragen.

10.4 Verbindung zu anderen Methoden und Komponenten

ARIS Business Architect bietet mit den verschiedenen Modellierungsmethoden die Möglichkeit, einen Sachverhalt in unterschiedlichen Sichten zu betrachten und mit einem jeweils speziellen Fokus verschiedenen Zielgruppen zugänglich zu machen. Das eBusiness Scenario ist Ausgangspunkt dieser Betrachtungen. Seine Objekte bieten die Möglichkeit, zielgruppenspezifisch Detaillierungen vorzunehmen. Hierdurch lässt sich die ganzheitliche Darstellung eines E-Business-Projektes erreichen. Außerdem ermöglichen die unterschiedlichen Komponenten von ARIS Auswertungen für die erzeugten Modelle, so dass Projekte im E-Business Umfeld optimal unterstützt werden können.

Beispiel: Einführung eines Online-Shops

In einem ersten Schritt werden mit Hilfe der ARIS-Komponente Balanced Scorecard (siehe Kapitel **Balanced Scorecard-Methode** (Seite 203)) die Ziele definiert, die durch die Einführung von eBusiness verwirklicht werden sollen. Die Prozesse, die zur Erreichung dieser Ziele optimiert werden sollen, werden dabei identifiziert. In vorliegendem Beispiel lässt sich die Erschließung eines neuen Vertriebsweges, nämlich über Internet, als Ziel identifizieren. Um diesen neuen Weg optimal zu beschreiten ist eine genaue Dokumentation und Planung notwendig

Nicht nur der Prozessablauf als solcher, sondern auch die Organisation der Verantwortlichkeiten, der Schnittstellen zwischen verschiedenen Systemen, der Datensicherheit etc. ist in die Überlegungen einzubeziehen.

Ausgangspunkt der Betrachtung ist das eBusiness Scenario Diagram. Business Participants sind in unserem Beispiel das Unternehmen, welches das Shop-System implementiert, und der Kunde, der dieses Angebot nutzen wird. Der gesamte Prozess vom „Betreten des Shops“ bis zum „Verlassen“ wird in seine wichtigsten Teilschritte zerlegt. Die Darstellung beinhaltet die Sicht des Kunden und des Unternehmens. Das eBusiness Scenario Diagram dient als Einstieg in das Einführungsprojekt. Diese Abbildung enthält eine Aufteilung des Gesamtprozesses.

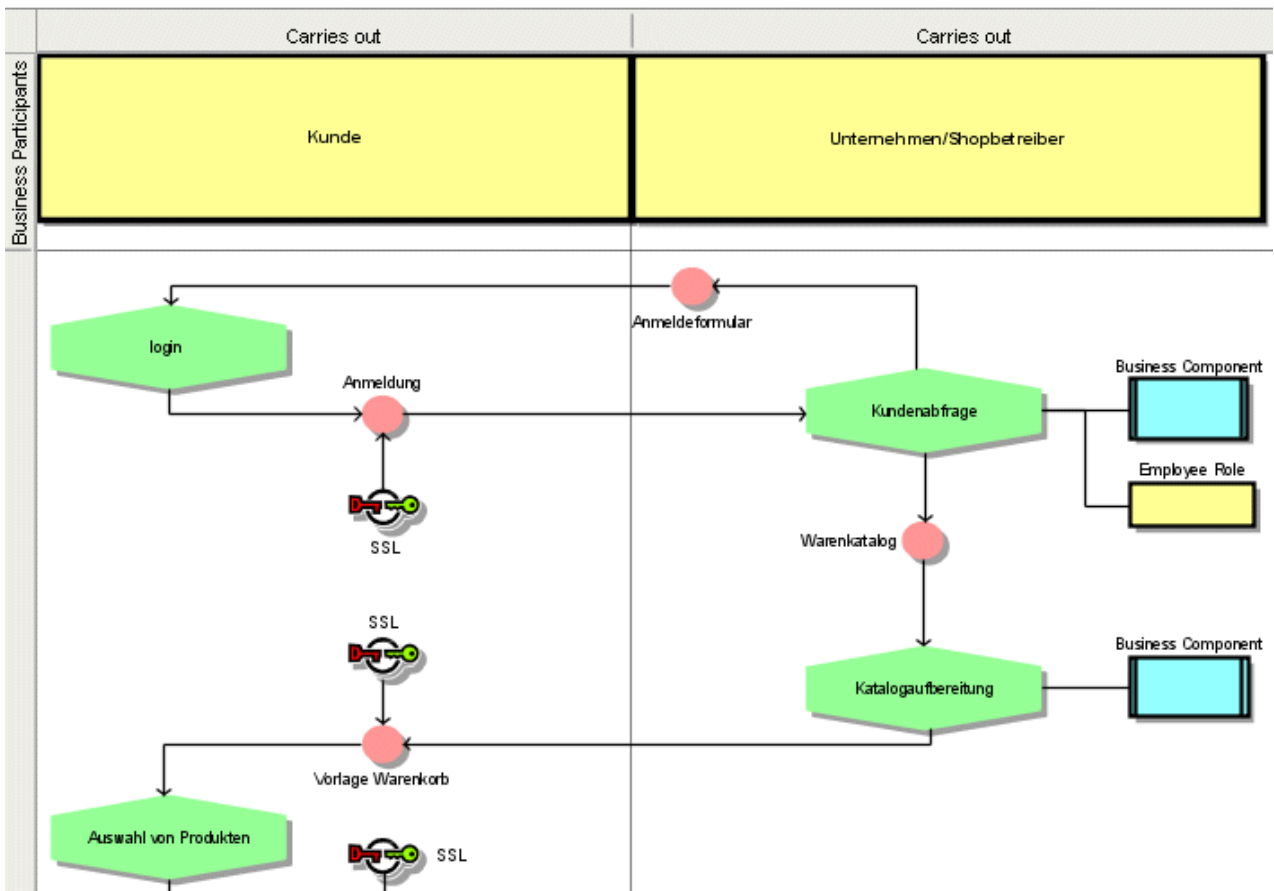


Abbildung 190: Ausschnitt aus eBusiness Scenario "Online-shop"

Die verschiedenen Prozessschritte lassen sich nun durch EPKs verfeinern, z. B. mit der Simulations-Komponente überprüfen, nach ihrer Optimierung durch Pipeline-Diagramme darstellen und durch Intershop Enfinity in ein fertiges Shop-System umwandeln und weiter verbessern.

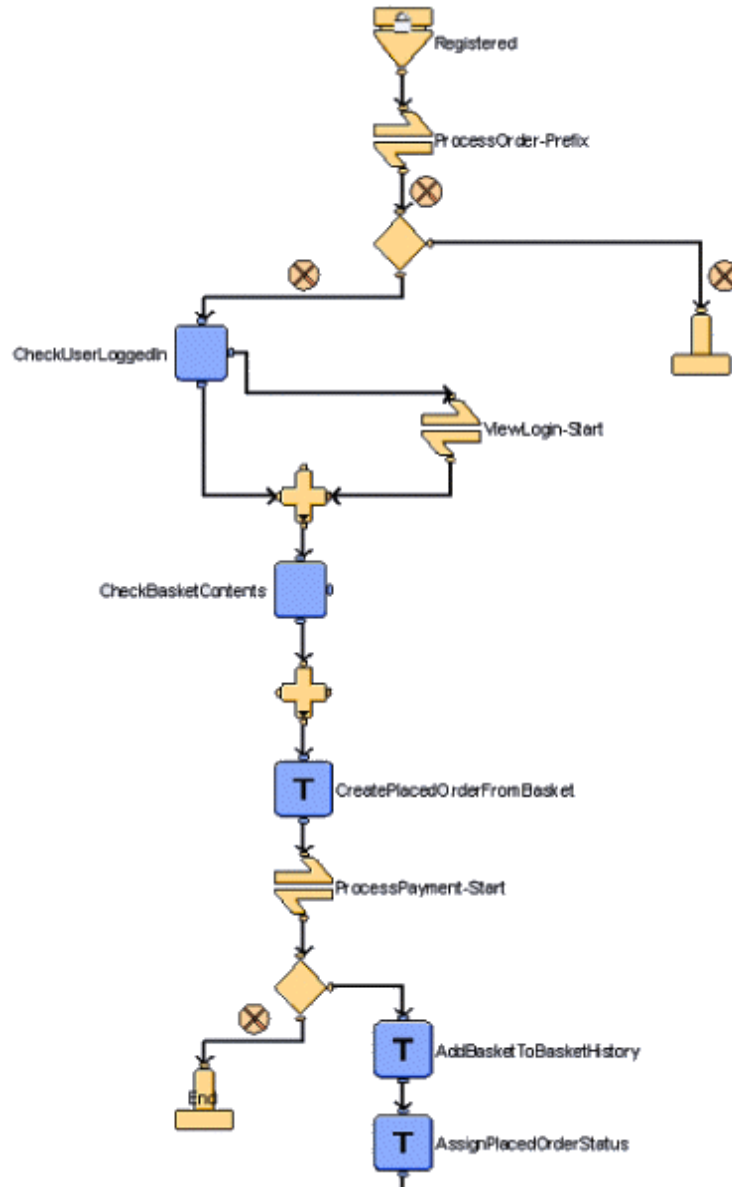


Abbildung 191: Ausschnitt aus dem Pipeline Diagram

Soll eine Anbindung eines Shops an das nachgelagerte ERP-System (Enterprise Resource Planning-System) erfolgen, benötigt man die korrekten Formate der zu übergebenden Daten. Um dies zu ermöglichen, gibt es verschiedene Wege, die Dokumente und Daten zu standardisieren. Eine Möglichkeit der Standardisierung bietet die Verwendung der Extensible Markup Language (XML). Um die zu erstellenden Dokumente in Ihrer Struktur und mit Ihren benötigten Inhalten zu definieren, kann man sie mittels DTDs hinterlegen. Da XML eine Sprache ist, die völlig unterschiedlich weiterentwickelt wird, ist auch hier eine einheitliche Grundlage notwendig. Unterschiedliche Organisationen, bestehend aus Unternehmen und wissenschaftlichen Instituten, bemühen sich um die Vereinheitlichung von XML für verschiedene Bereiche.

Die Verwendung von mit XML vereinheitlichten Dokumenten erleichtert die Anbindung von ERP-Systemen.

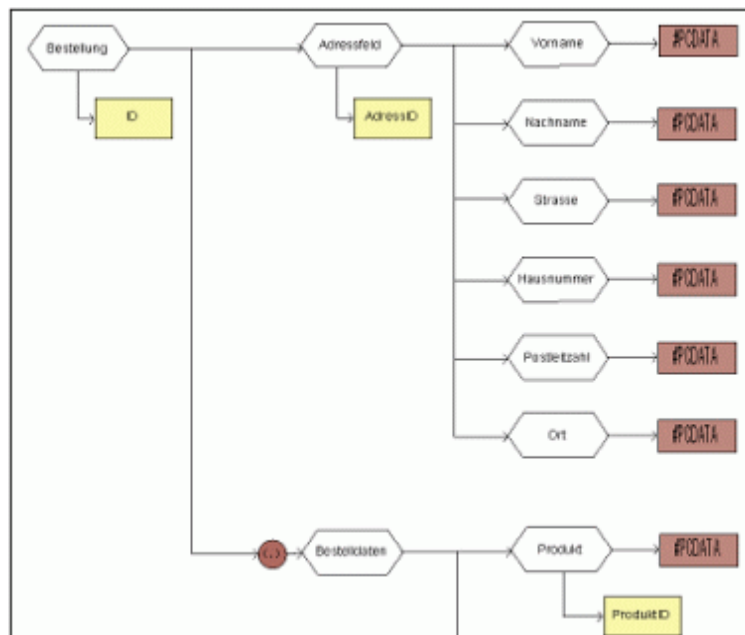


Abbildung 192: Ausschnitt aus der DTD: Bestellung

Die Problematik verschiedener Komponenten, die aufeinander abgestimmt werden müssen, wurde bereits angesprochen. Eine Visualisierung der Systeme kann im Anwendungssystemtypdiagramm erfolgen, das den Application Systems oder auch Business Components als Modell hinterlegt werden kann und die Zusammenhänge der Systeme verdeutlicht.

Auch die Organisationsstruktur ist von der Einführung von E-Business betroffen. Eventuell müssen neue Verantwortlichkeiten definiert oder Zuordnungen neu getroffen werden. Im eBusiness Scenario Diagramm können Rollen und Organisationseinheiten für die Prozessschritte beschrieben werden. Welche Position sie in der Organisation der Unternehmen oder auch im Prozess einnehmen, kann durch Organigramme weiter analysiert werden.

Der Schritt der Implementierung beginnt mit der Umsetzung der modellierten Inhalte. Bei Verwendung der Intershop-Methode werden die modellierten Inhalte mit Intershop Enfinity in ein laufendes System umgewandelt.

11 IT City Planning

11.1 Enterprise Architecture und IT City Planning

IT City Planning ist ein in den 80er Jahren entwickelter Architekturansatz des Franzosen Jacques Sassoon. Ziel des IT City Planning ist es, Stimmigkeit in eine heterogene Systemlandschaft zu bringen, indem die Interaktionen, also der Informationsaustausch zwischen den Applikationen innerhalb eines Systems genauer betrachtet werden.

In Anlehnung an die Städteplanung ist das Vorgehen beim Erstellen eines IT City Plan von dem Gedanken bestimmt, ein langfristiges, strategisches IT-Management zu ermöglichen, das nicht nur die Gegenwart berücksichtigt, sondern auch die Aspekte der Vergangenheit (Legacy-Systeme) und die der Zukunft einbezieht.

Dabei wird kein Redesign für das gesamte System durchgeführt. Vielmehr wird inkrementell von Projekt zu Projekt vorgegangen.

Ähnlich wie bei der Model Driven Architecture (MDA), die durch die ARIS-Produkte unterstützt wird, erfolgt der Einstieg über Modelle, die das Informationssystem losgelöst von technologiebezogenen Informationen beschreiben. Allerdings wird im Rahmen des IT City Planning nicht auf UML zurückgegriffen, was den Einstieg für Personen mit weniger technischem Hintergrund vereinfacht und die Akzeptanz erhöht.

11.2 An welche Unternehmen richtet sich IT City Planning?

IT City Planning richtet sich an folgende Unternehmen:

- Unternehmen, mit einem großen Bestand an Anwendungen.
- Unternehmen, die schon sehr lange Informationstechnologien einsetzen.
- Unternehmen, in denen die Informationstechnologie von strategischer Bedeutung ist.
- Unternehmen, die fusionieren.

Ziele des IT City Planning:

- Wiederverwendung von Software-Ressourcen, um das Anlegen weiterer Redundanzen zu vermeiden.
- Verringerung von Wartungskosten, indem durch "blockweise" Überarbeitung und Definition neuer Software-Ressourcen die verschiedenen, bisher verwendeten, ersetzt und die verschiedenartigen Anwendungsfälle abgedeckt werden können.
- Konsolidierung der Informationssysteme.
- Vorbereitung des Einsatzes von EAI-Software auf einer höheren Ebene.

Das Erstellen eines IT City Plan ist Aufgabe des Integration Competence Center. Der Plan selbst muss sowohl die Design Pattern als auch die Informations- und Technologiearchitektur ansprechen.

EAI = Enterprise Application Integration. Unternehmensweite Integration von Anwendungen. EAI liefert die E-Business-Infrastruktur. EAI-Software ist die technische Middleware und Voraussetzung für die Realisierung einer E-Business-Strategie.

11.3 IT City Planning mit ARIS

ARIS betrachtet folgende Sichten eines Informationssystems:

- Datensicht
- Funktionssicht
- Organisationssicht
- Leistungssicht
- Prozesssicht

Jede dieser Sichten ist in die Beschreibungsebenen **Fachkonzept**, **DV-Konzept** und **Implementierung** untergliedert. Sie orientieren sich am Lebenszyklus eines Informationssystems und der Nähe zur Informationstechnik.

Die DV-Konzept- und Implementierungsebenen beschreiben im Wesentlichen das Software-System. Die Begriffswelten dieser Ebenen sind eng miteinander verwandt und die "Übersetzung" ist unproblematisch.

Anders verhält es sich mit dem Übergang zwischen Fachkonzept und DV-Konzept. Bei der DV-Konzepterstellung muss die betriebswirtschaftliche Sicht mit der Standardsoftware abgeglichen werden. Dies erfordert betriebswirtschaftliches Know-how ebenso wie DV-nahes Wissen (vgl. Scheer, A.-W., ARIS - Vom Geschäftsprozess zum Anwendungssystem, 1998, 3. Auflage, S. 7).

Die Informationssystem-Sicht (IS-Sicht) des IT City Planning gibt hier als Vermittler zwischen den Ebenen Hilfestellung. Die Objekttypen der IS-Sicht sind in ARIS zwischen Funktion und Anwendungssystemen anzusiedeln und erweitern somit die Funktionssicht in ARIS. Wie Funktionen werden IS-Elemente mit den verschiedenen Konstrukten aus den bekannten Sichten des ARIS-Hauses in Beziehung gesetzt. Diese Erweiterungen betreffen im Wesentlichen die Prozesssicht und Datensicht. Wenn im Folgenden von IS-Sicht gesprochen wird, sind darunter die Modelltypen aus Funktions- und Prozesssicht des ARIS-Hauses zu verstehen, in denen Beziehungen zwischen IS-Elementen beschrieben werden oder die einer Detailbeschreibung von IS-Elementen im Kontext der anderen ARIS-Sichten dienen.

Anwendungssystemtypen, DV-Funktionstypen und Sockets werden im Folgenden als Elemente der so genannten IT-Sicht (IT-Sicht = Informationstechnologie-Sicht) betrachtet. In Analogie zur IS-Sicht gehören alle Modelltypen der IT-Sicht an, in denen Beziehungen zwischen Anwen-

dungssystemtypen, DV-Funktionstypen und dem neuen Objekttyp **Socket** beschrieben werden oder die einer Detailbeschreibung eines dieser Elemente dienen.

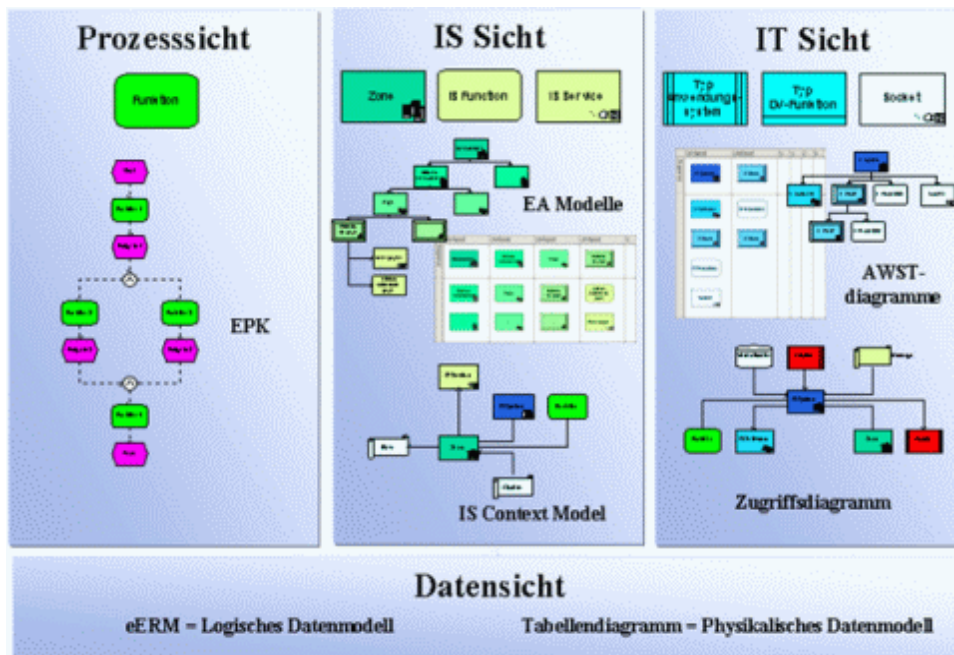


Abbildung 193: Prozess-, IS- und IT-Sicht

11.4 Die IS-Sicht

Zum Beschreiben der IS-Sicht stellt ARIS folgende Modelltypen zur Verfügung:

- Service-Architekturdiagramm
- Service-Architekturdiagramm (Column Display)
- Service-Zuordnungsdiagramm
- Service-Kollaborationsdiagramm

In den beiden Service-Architekturdiagramm-Modellen wird das Informationssystem in hierarchischer Form gegliedert

Eine IS-Hierarchie kann folgende Ebenen umfassen:

- Zone
- District
- Building Cluster
- Functional Block
- Fähigkeit
- IS Service
- Geschäftsservice

Zone, District, Building Cluster, Functional Block und Geschäftsservice sind vom Objekttyp **Servicetyp**. Servicetypen dienen der funktionalen Gliederung eines Informationssystems in unabhängige Einheiten/Blöcke.

Jeder Servicetyp zeichnet sich dadurch aus, dass es "Eigner" der von ihm verwendeten Daten und der zugehörigen Verarbeitungsmethoden ist. Andere Servicetypen können auf diese Daten und Verarbeitungsmethoden nur zugreifen, wenn sie einen Dienst des "Eigner-Servicetyp" aufrufen. In einem Servicetyp werden ähnliche Daten verwendet und gleiche Aktivitäten und Geschäftsfunktionen ausgeführt.

Auf der höchsten Ebene wird das Informationssystem in Zonen gegliedert. Eine Zone entspricht z. B. einem Arbeits- und Entwicklungsbereich.

Folgende Abbildung zeigt die Zonen, in die ein Informationssystem eines Unternehmens typischerweise untergliedert werden kann.



Abbildung 194: Zonen eines Unternehmens-Informationssystems

Jede Zone kann wiederum einen Distrikt oder mehrere Distrikte enthalten.

Distrikte einer Zone zeichnen sich durch ähnliche Prozesse und große Ähnlichkeit bzgl. zeitlicher Merkmale aus, z. B. ähnliche Lebens- und Informationsbearbeitungszyklen. Distrikte sind beispielsweise Zahlungskonditionen, Preiskonditionen, Personalverwaltung, Reiserichtlinien etc.

Die Zone **Ressourcen** kann u. a. die Distrikte **Human Resources** und **Rechnungswesen** umfassen:

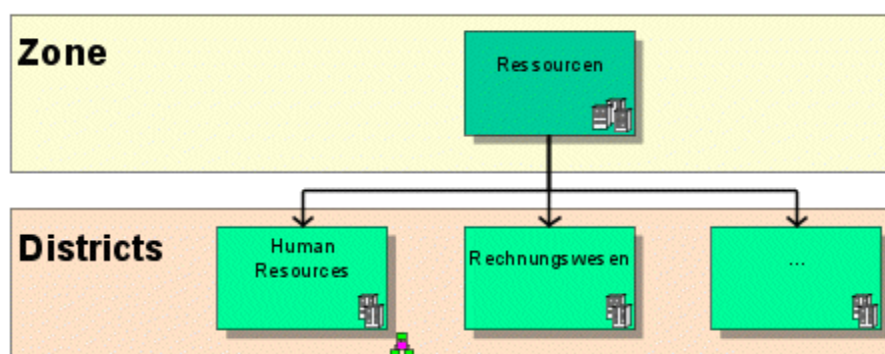


Abbildung 195: Zone in Distrikte untergliedert

Innerhalb eines Distrikts gibt es einen oder mehrere Building Cluster, die einen funktionalen Zweck erfüllen, z. B. Lohnzahlungen, Fakturierung usw.

Der Distrikt Human Resources umfasst das Building Cluster Bereichsadministration, Recruiting, Personalentwicklung und Personalbetreuung.

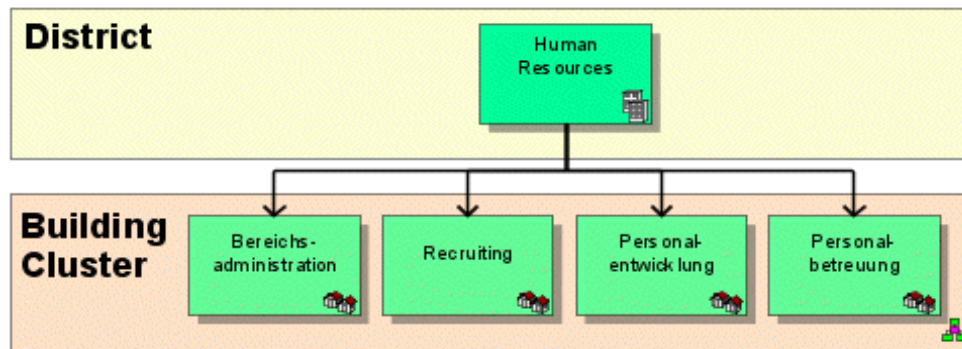


Abbildung 196: Distrikt in Building Cluster untergliedert

Jedes Building Cluster kann wiederum ein oder mehrere Functional Blocks umfassen. Die Functional Blocks eines Building Cluster zeichnen sich durch hohe Ähnlichkeit der von ihnen verwalteten Business-Objekte und Ereignisse aus.

Ein Functional Block stellt eine unabhängige, wieder verwendbare funktionale Komponente dar. In einem Functional Block werden Fähigkeiten und IS Services nach folgenden Regeln zusammengefasst:

- Sie weisen einen starken Zusammenhang zwischen den von ihnen verwalteten Objekten und unterstützten Funktionen auf.
- Der Austausch mit anderen funktionalen Blöcken ist minimal.

Das Building Cluster **Personalbetreuung** unseres Beispiels umfasst die Functional Blocks **Stammdatenpflege**, **Zeugnisse**, **Controlling** und **Gehälter**.

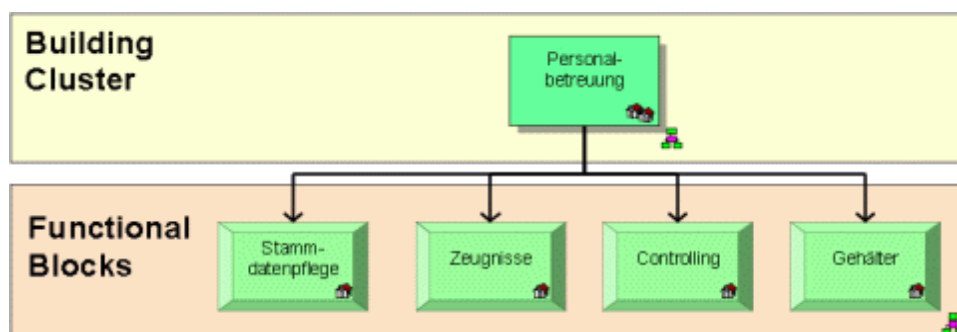


Abbildung 197: Aufteilung des Building Cluster "Personalbetreuung" in Functional Blocks

Eine Fähigkeit beschreibt eine elementare Funktionalität innerhalb eines Systems. Sie unterstützt die Realisierung einer Aktivität innerhalb eines Prozesses.

Ein IS Service beschreibt eine Schnittstelle eines Servicetyps oder einer Fähigkeit. Sie ermöglichen anderen IS-Elementen einen kontrollierten Zugriff auf Daten und Verarbeitungsmethoden des IS-Elements, das den Service zur Verfügung stellt.

Über diese Schnittstellen können Nachrichten mit anderen Elementen der IS-Sicht ausgetauscht werden.

Folgende Abbildung zeigt die Fähigkeiten und IS Services des Functional Block **Gehälter**.

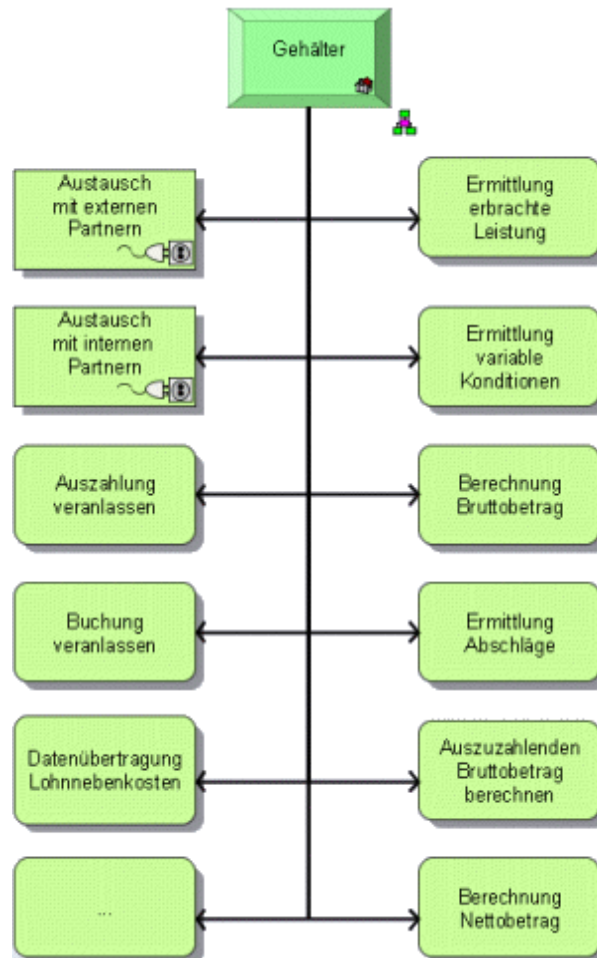


Abbildung 198: Fähigkeiten und IS Services des Functional Block "Gehälter"

Für die Beschreibung der IS-Hierarchie ist es nicht notwendig alle hier beschriebenen Ebenen auszumodellieren. Die IS-Elemente **Fähigkeit** und **IS Service** werden im Rahmen des IT City Planning nicht als Elemente des Städteplans gesehen. Das Aufgabenfeld des Städteplaners endet auf der Ebene der Building Blocks. Fähigkeiten und IS Services gehören in das Aufgabengebiet des Architekten (vgl. Longépé, Christoph: Le projet d'urbanisation du système d'information, S. 18).

11.5 Servicetypen und ihre Daten

Um zu beschreiben, welche Daten innerhalb eines Servicetyps oder einer Fähigkeit gekapselt sind, wird das eERM verwendet. Im Rahmen des City Planning werden im eERM die Symbole der IS-Sicht angeboten. Mit Hilfe einer Kante des Typs **ist Eigner von** können diese Objekte mit Beziehungstypen und Entitytypen in Beziehung gesetzt werden.

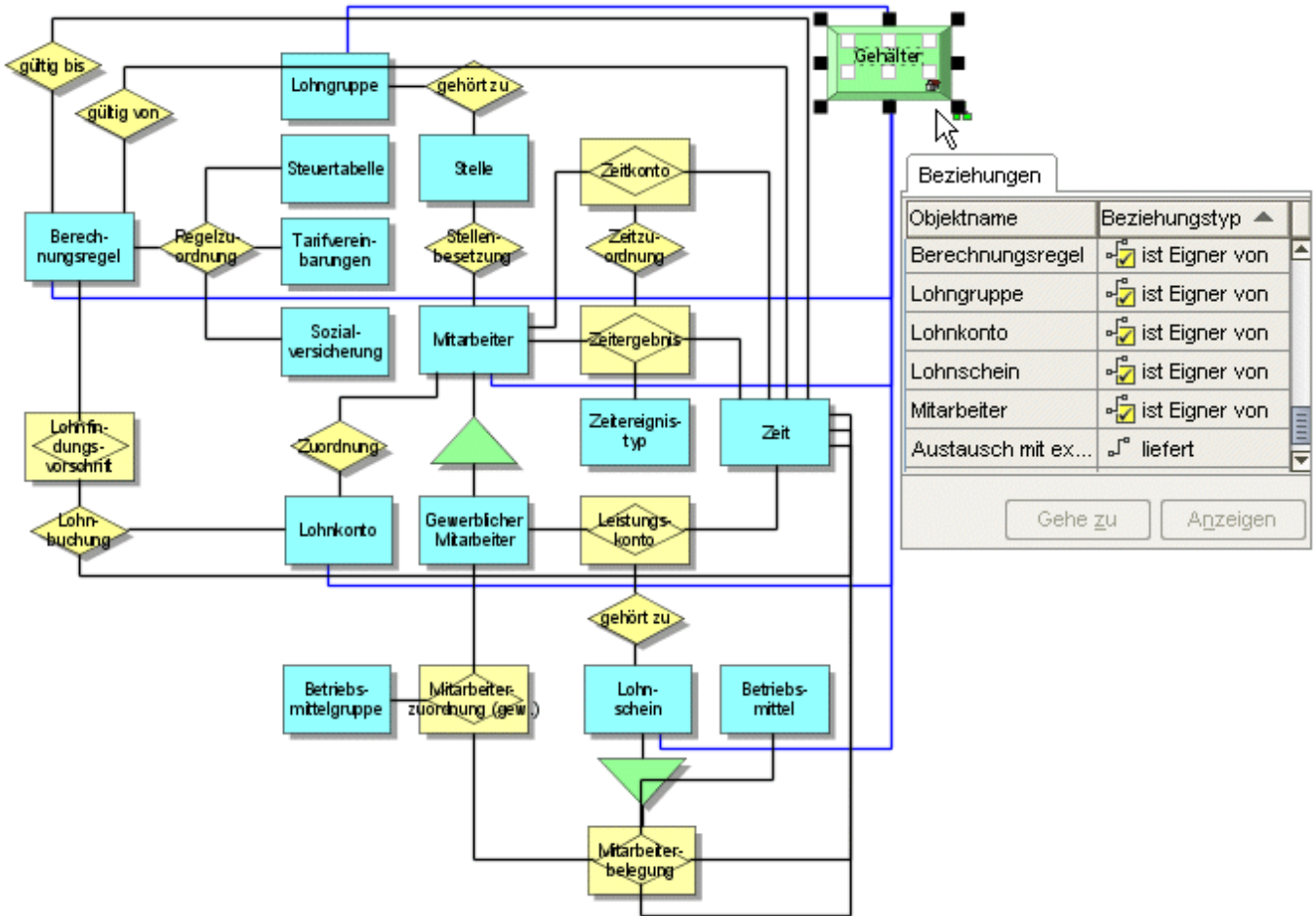


Abbildung 199: Kante "ist Eigner von" zwischen Symbolen der IS-Sicht und Beziehungs- und Entitytypen

11.6 Detailbeschreibung der Servicetypen

Eine detaillierte Beschreibung der Servicetypen und Fähigkeiten eines Informationssystems wird im Service-Zuordnungsdiagramm vorgenommen. Hierzu gehören

- die Schnittstellen eines Blocks,
- die Interaktionen zwischen den Blöcken,
- die Anwendungssysteme, die den Block unterstützen, und
- betriebswirtschaftliche Funktionen, die durch den Block unterstützt werden.

Zonen, Distrikte, Building Cluster, Functional Block und Fähigkeit können mittels der Kante **liefert** mit einem IS Service verbunden werden.

Mit Hilfe von Input/Output-Kanten zwischen IS- und Datenelementen können die Informationsflüsse zwischen den Servicetypen beschrieben werden.

Die verschiedenen Anwendungssystem- und DV-Funktionstypobjekte können durch eine Kante des Typs **unterstützt** den Objekten der IS-Sicht zugeordnet werden. Interpretiert man den City Plan nun wie den Bebauungsplan einer Stadt, kommt durch diese Kante zum Ausdruck, welche Informationssystembereiche von welchen Anwendungssystemen "besiedelt" werden. Die Kante **unterstützt** steht auch zwischen IS-Elementen und der Funktion zur Verfügung.

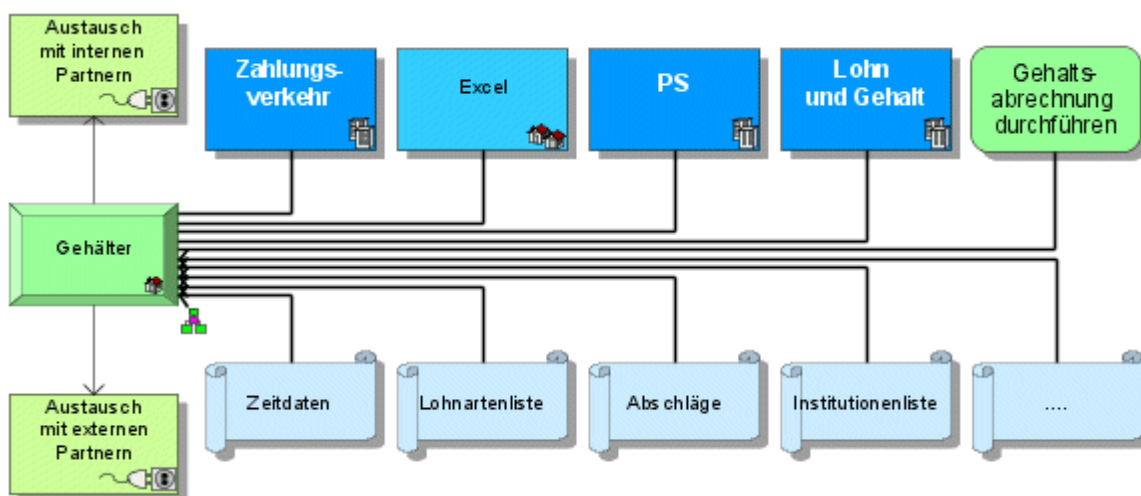


Abbildung 200: Kante "unterstützt" zwischen IS-Elementen und Funktion

11.7 Zeitlich-logische Abläufe zwischen IS-Elementen

Im Service-Zuordnungsdiagramm können für die im Service-Architekturdiagramm spezifizierten Servicetyp, Fähigkeit und IS Services die Beziehungen zu den Objekttypen der Organisations-, Daten- und Prozesssicht hergestellt werden. Die möglichen zeitlich-logischen Ablauffolgen der IS-Elemente können nicht dargestellt werden.

Das Service-Kollaborationsdiagramm wird im Rahmen des IT City Planning verwendet, um zeitlich-logische Ablauffolgen der IS-Elemente wiederzugeben, also den dynamischen Aspekte innerhalb des Informationssystems zu beschreiben. Es stellt das Pendant zum Programmablaufdiagramm (vgl. Kapitel **Programmablaufdiagramm** (Seite 146)) der IT-Sicht dar. Durch diesen Modelltyp werden zur Darstellung der Ablauffolge Ereignisse zur Verfügung gestellt. Ähnlich der Zuordnung von IT-Elementen und Ereignissen im Programmablaufdiagramm können hiermit im Service-Kollaborationsdiagramm Folgen funktionaler Module definiert werden. Das Ereignis ist hierbei im Sinne eines Triggers zu verstehen, das Servicetypen, Fähigkeiten oder IS Services anstößt. Verzweigungen können mit den aus der EPK oder dem Programmablaufdiagramm bekannten Regeln dargestellt werden. Die Ablauffolgen können auch ohne das Zwischenschalten von Ereignissen definiert werden.

11.8 Die IT-Sicht

In Analogie zur IS-Sicht stehen in der IT-Sicht folgende Modelltypen zur Verfügung:

- Anwendungssystemtypdiagramm
- Anwendungssystemtypdiagramm (Spaltendarstellung)
- Zugriffsdiagramm
- Programmablaufdiagramm

Anwendungssystemhierarchie

Im Rahmen des City Planning wird die aktuelle Anwendungssystemhierarchie im Unternehmen mit Hilfe des Anwendungssystemtypdiagramms bzw. des Anwendungssystemtypdiagramms (Spaltendarstellung) abgebildet. Das Anwendungssystemtypdiagramm (Spaltendarstellung) ist ein Swimlane-Diagramm (Diagramm in Spalten- und/oder Zeilendarstellung), das genau die Objekttypen, Symbole und Beziehungstypen aus dem Anwendungssystemtypdiagramm anbietet, die für das City Planning notwendig sind.

Folgende Ebenen einer Anwendungssystemtyphierarchie können abgebildet werden:

- IT System
- Subsystem
- IT Software
- IT Block
- IT Procedure

- Socket

IT System, Subsystem, IT Software und IT Block sind Symbole des Objekttyps **Anwendungssystemtyp**. Die Hierarchie wird durch den Beziehungstyp **umfasst** hergestellt.

Auf der obersten Ebene der Anwendungssystemtyphierarchie befinden sich die IT-Systeme. Ein IT-System beschreibt eine strukturierte Menge von IT Elementen, i. d. R. Subsysteme. Verwaltung und Betrieb eines IT Systems unterliegt einer eindeutig bestimmten Organisationseinheit.

Ein Subsystem beschreibt eine Komponente eines IT Systems. Die Bausteine eines Subsystems werden IT Software genannt.

IT-Software unterstützt einen homogenen Satz von Funktionen. Sie ist benutzerorientiert und unterstützt einen oder mehrere Geschäftsprozesse. Bestandteile einer IT Software sind IT Blöcke.

In einem IT Block werden in der Regel solche IT Procedures zusammengefasst, die auf gleiche Daten (Datenbanken, Tabellen, Dateien etc.) zugreifen.

IT Procedures sind Objekte des Typs **DV-Funktionstyp**. Jede IT Procedure unterstützt eine bestimmte Funktionalität.

Ein Socket entspricht dem IS Service, d. h., er beschreibt eine Schnittstelle, die ein IT-Element anderen IT-Elementen zur Verfügung stellt, um ihnen den Zugriff auf seine Daten und Verarbeitungsmethoden zu ermöglichen.

Folgende Abbildung zeigt beispielhaft die Subsystemstruktur des Systems DATEV:



Abbildung 201: Subsystemstruktur des DATEV-Systems

11.9 IT-Elemente und ihre Daten

Um zu beschreiben, welche Daten innerhalb eines IT-Elements gekapselt sind, steht wie für die die IS-Elemente im eERM die Kante **ist Eigner von** zwischen Anwendungssystemtyp, DV-Funktionstyp oder Socket und Entity- oder Beziehungstyp zur Verfügung.

11.10 Detailbeschreibung der IT-Elemente

Eine detaillierte Beschreibung der IT-Elemente innerhalb des IT City Plan wird im Zugriffsdiagramm vorgenommen. Es entspricht dem Service-Zuordnungsdiagramm der IS-Sicht.

Beschrieben werden können

- In- und Output-Beziehungen des betrachteten IT-Elements
- unterstützte Geschäftsfunktionen
- unterstützte IS-Elemente
- Aufruf anderer IT-Elemente durch das betrachtete
- Plattform, auf der das IT-Element läuft
- Anwender des IT-Elements

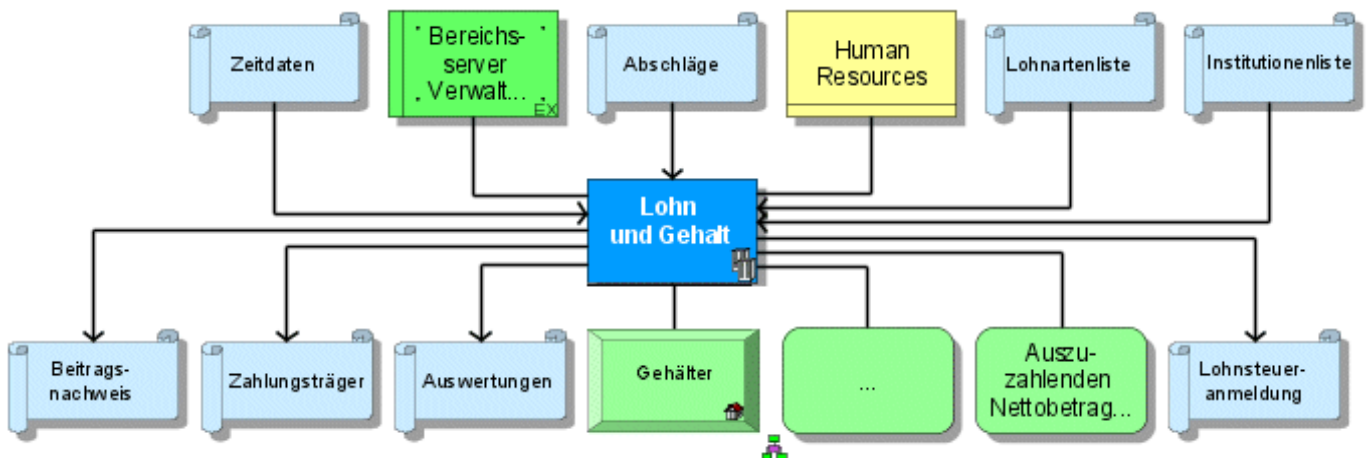


Abbildung 202: Detaillierte Beschreibung von IT-Elementen im Zugriffsdiagramm

11.11 Organisatorische Aspekte

In die Detailbeschreibung eines IT-Elements fließen auch Informationen der Organisationsicht ein. Hierzu zählen nicht nur die Informationen darüber, welche organisatorischen Elemente Anwender eines IT-Elements sein können. Mit Hilfe des Netzdiagramms können auch Einflüsse und Auswirkungen bzgl. der technischen Infrastruktur verdeutlicht werden.

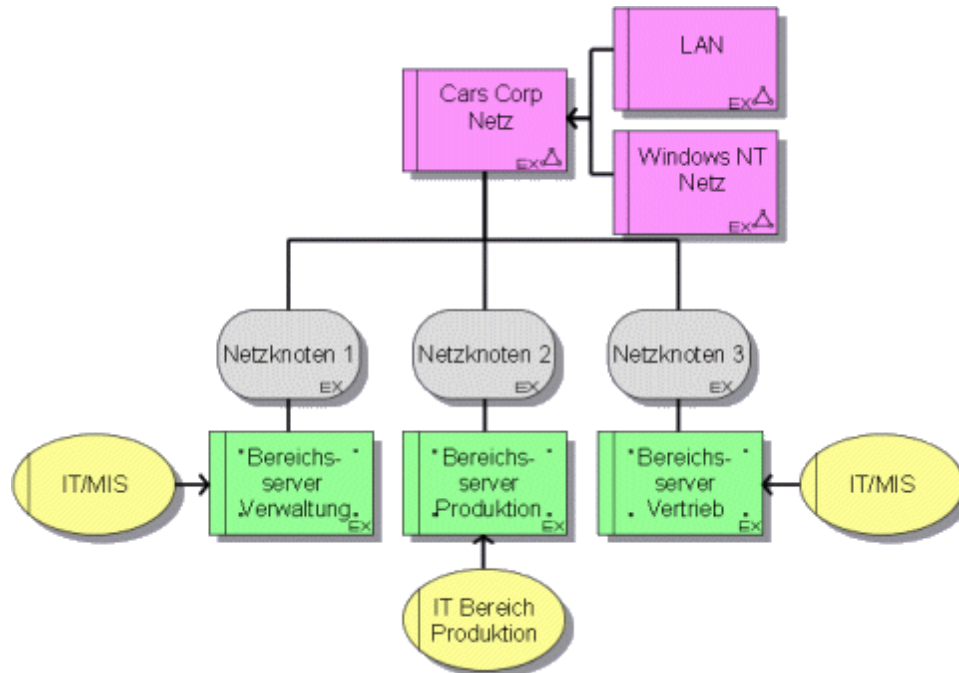


Abbildung 203: Einflüsse und Auswirkungen der technischen Infrastruktur

11.12 Zeitlich-logische Abläufe zwischen IT-Elementen

In Analogie zum Service-Kollaborationsdiagramm dient das Programmablaufdiagramm der Beschreibung der zeitlich-logischen Abläufe zwischen den verschiedenen IT-Elementen **Anwendungssystemtyp**, **DV-Funktionstyp** und **Socket**.

Näheres zum Programmablaufdiagramm finden Sie in Kapitel **Modellierung der System-schnittstellenmodelle - Systemattribute, Systemattributdomäne** (Seite 69).

11.13 Zeitlich-logische Abläufe innerhalb der Architektur

Um die Einbindung von IS- und IT-Elementen in einen zeitlich-logischen Ablauf darzustellen, stehen entsprechende Objekte in den verschiedenen Prozessmodellen (alle Variationen der EPK) und im Programmablaufdiagramm zur Verfügung.

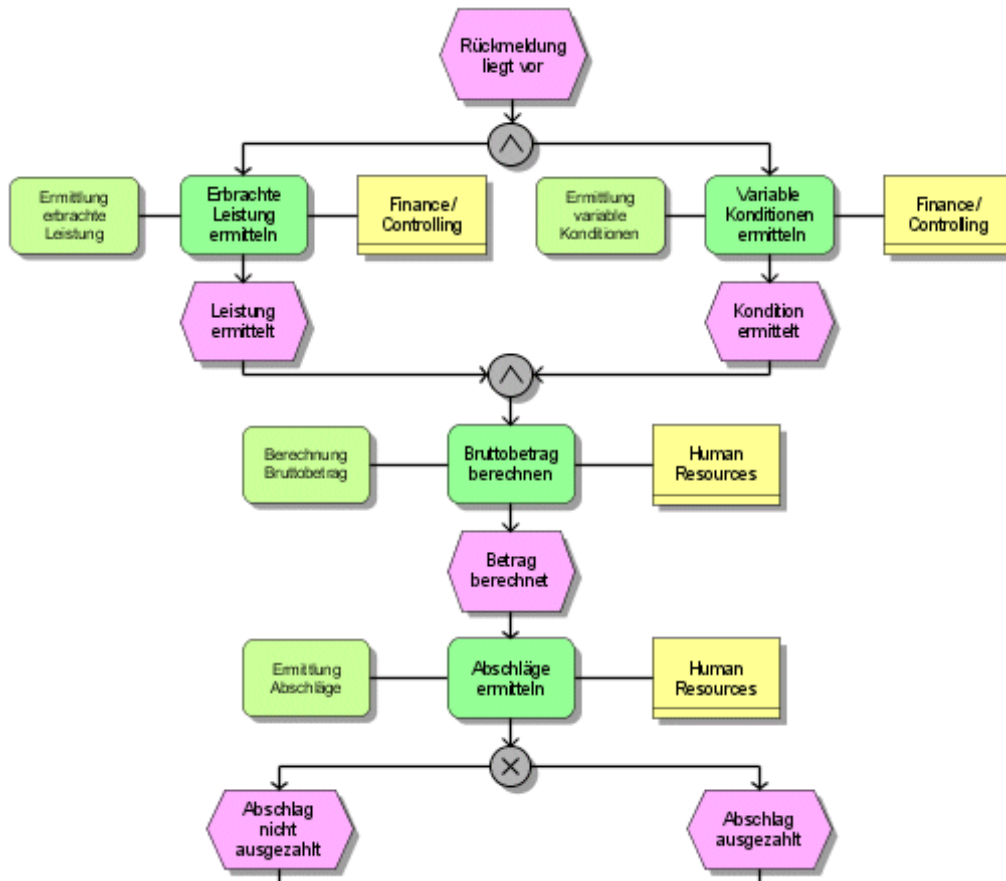


Abbildung 204: Einbindung von IS- und IT-Elementen in einen zeitlich-logischen Ablauf

11.14 Mögliche Auswertungen

Basierend auf der in den vorhergehenden Abschnitten beschriebenen Modellierung sind Auswertungen möglich, die folgende Fragen beantworten und beim Erstellen des zukünftigen Informationssystems hilfreich sind:

- Welche Daten werden von einem bestimmten IS-Element verwaltet?
- Welche Anwendungssysteme unterstützen ein IS-Element?
- Welche Funktionen werden von einem bestimmten IS-Element unterstützt?
- Welche Daten werden von den IT-Elementen eines bestimmten IS-Elements verwendet?
- Welche Daten werden von den IT-Elementen eines bestimmten IS-Elements erzeugt?
- Welche IS Services werden von einem IS-Element zur Verfügung gestellt und in welchen Prozessen werden sie genutzt?
- Auf welchen Hardwarekomponenten laufen die Anwendungssysteme eines bestimmten IS-Elements?

Für gewählte Anwendungssystemtypen, DV-Funktionstypen und Sockets werden folgende Auswertungen angeboten:

- Daten, die von einem IT-Element verwendet werden.
- IS-Elemente, die von einem IT-Element unterstützt werden.
- Funktionen, die von einem IT-Element unterstützt werden.
- Daten, die von den durch ein IT-Element unterstützten IS-Elementen verwendet werden.
- Daten, die von den durch ein IT-Element unterstützten IS-Elementen erzeugt werden.
- Hardwarekomponenten, auf denen ein IT-System läuft.

12 Business Process Modeling

Neue Informations- und Kommunikationstechnologien führen zu immer komplexeren Interaktionen und Transaktionen zwischen Unternehmen, ihren Partnern, Lieferanten und Kunden. Zunehmend zeigt sich, dass die weitere Entwicklung und Leistungsfähigkeit der Geschäftsprozesse von der engen Zusammenarbeit der einzelnen Geschäftspartner abhängt.

Zum einen möchte man sicherstellen, dass ein Unternehmen sein eigenes Handeln und das seiner Geschäftspartner besser verstehen kann, zum anderen sollen Organisationen die Fähigkeit erhalten, sich schneller an interne und marktbedingte Veränderungen anzupassen. Dabei kann eine standardisierte Prozessmodellierungssprache den Unternehmen helfen, ihre internen und externen Geschäftsabläufe transparent und flexibel zu beschreiben. Die Unternehmen sollen außerdem in der Lage sein, die modellierten Prozesse in einer angemessenen, anschaulichen und verständlichen Weise den beteiligten Geschäftspartnern kommunizieren zu können. Alle Beteiligten sollen die gleiche „Prozesssprache“ sprechen.

Die Business Process Management Initiative (BPMI.org) bietet zur Realisierung dieser Ziele die eigens entwickelte, standardisierte Modellierungssprache Business Process Model and Notation (BPMN) an. BPMN ist eine grafische Notation zum Beschreiben von Geschäftsprozessen.

Die Notation soll für alle Anwender leicht verständlich sein. Damit ist sie für Geschäftsprozess-Analysten und Personen geeignet, die die Prozesse überwachen und verwalten, aber auch für Entwickler, die Technologien zur Prozessausführung implementieren.

Daneben soll sichergestellt sein, dass XML-basierte Sprachen für die Geschäftsprozessautomation mit dieser Notation visualisiert werden können, z. B. Business Process Execution Language for Webservices (BPEL4WS).

12.1 Die Beschreibungssprache BPMN

12.1.1 Prozessklassen und das Business Process Diagram

Business Process Model and Notation (BPMN) sieht zur Prozessbeschreibung den Modelltyp **Business Process Diagram (BPD)** vor. Dieses Modell bildet drei Klassen von Geschäftsprozessen und deren Beziehung untereinander ab:

- **Private Business Processes** (interne Geschäftsprozesse)
- **Abstract Business Processes** (öffentliche Geschäftsprozesse)
- **Collaboration Processes** (globale Geschäftsprozesse)

Private Business Processes sind Geschäftsprozesse, die nur innerhalb einer Organisation ablaufen. Sie sind allgemein bekannt unter dem Namen Workflow oder BPM-Prozesse.

Verschiedene interne Geschäftsprozesse werden als Sequence flow innerhalb der Grenzen von einzelnen Pools (siehe Kapitel **Realisierung der BPMN in ARIS** (Seite 247)) modelliert, deren Interaktion untereinander mittels Message flow dargestellt wird.

BPMN verwendet anstelle des Begriffs Kontrollfluss die Begriffe Sequence flow und Message flow, da nicht nur die Ereignisse den Prozess steuern, sondern auch die ausgetauschten Nachrichten.

Abstract Business Processes beschreiben Interaktionen zwischen Private Processes unterschiedlicher Pools, zwischen Objekten verschiedener Pools oder Kombinationen aus beiden. Hier ist - neben dem Sequence flow innerhalb des Private Process - der Message flow zwischen den einzelnen Prozessen von besonderer Bedeutung. Die Interaktionen werden über Message flows modelliert.

Abstract Business Processes sind in einzelne Pools eingebunden und können separat oder innerhalb eines ganzen BPMN-Diagramms modelliert werden. Wenn ein Abstract Business Process im gleichen Modell erscheint wie sein entsprechender Private Business Process, können sie miteinander assoziiert werden.

Collaboration Processes beschreiben ausschließlich die Interaktionen zwischen zwei oder mehreren Business Entities (Geschäftspartnern). Dabei wird eine Abfolge von Aktivitäten modelliert, die das Nachrichtenaustauschmuster zwischen den verschiedenen Geschäftspartnern beschreibt. Der Sequence flow spielt dabei keine Rolle mehr.

Relevante Sprachen für Collaborations sind beispielsweise bXML BPSS, RosettaNet, W3C Choreography Working Group. Die Mapping-Spezifikation ist für weitere Versionen der BPMN-Spezifikation geplant.

Collaboration Processes können in Pools eingebunden werden. Die Interaktionen der beteiligten Partner werden in einzelnen Lanes beschrieben. Die Prozesse können auf diese Weise separat oder innerhalb eines umfassenden BPMN-Diagramms modelliert werden. Wenn eine Collaboration im gleichen Diagramm erscheint wie einer ihrer internen Prozesse, können Aktivitäten, die in beiden gebräuchlich sind, miteinander assoziiert werden.

Aus diesen drei Prozessklassen können wiederum verschiedene Arten von Geschäftsprozessen abgeleitet werden:

- Private Business Processes auf höherer Ebene
- Private Business Processes auf Detailebene (Soll- oder Ist-Prozesse)
- Abläufe zwischen Detailprozessen und externen Entities
- Abläufe zwischen Detailprozessen
- Abläufe zwischen Detailprozessen und abstrakten Prozessen
- Abläufe zwischen Detailprozessen und Collaboration-Prozessen
- Abläufe zwischen abstrakten Prozessen
- Abläufe zwischen abstrakten Prozessen und Collaborations
- Abläufe zwischen Collaborations
- Abläufe zwischen mehreren Detailprozessen durch ihre abstrakten Prozesse
- Abläufe zwischen mehreren Detailprozessen durch eine Collaboration
- Abläufe zwischen mehreren Detailprozessen durch ihre abstrakten Prozesse und einer Collaboration

Folgende Abbildung zeigt beispielhaft ein BPD mit zwei Geschäftspartnern, denen ein eigener Prozess zugeordnet ist. Beide Detailprozesse bestehen aus einem Startereignis, Aktivitäten, Sequence flow-Kanten und einem Endereignis. Zwischen den Aktivitäten der beiden Detailprozesse sind Message flow-Kanten modelliert.

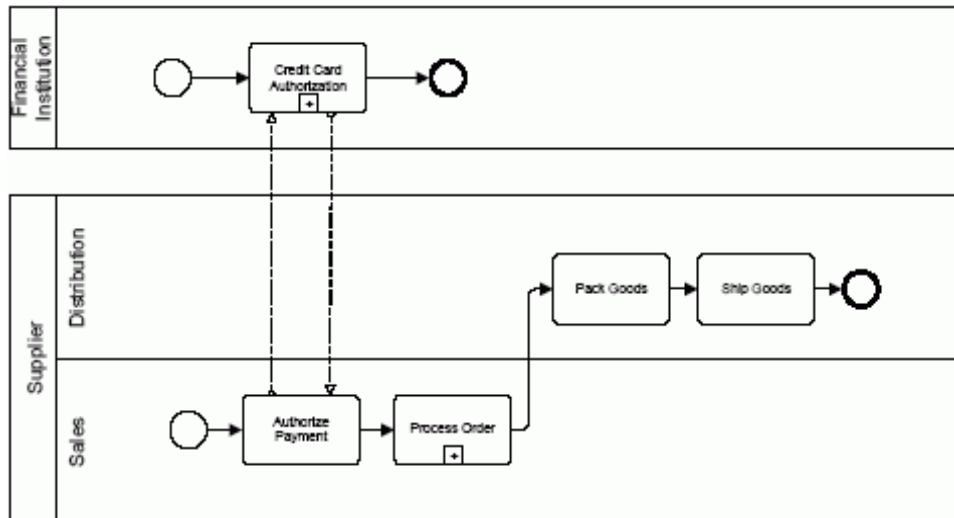


Abbildung 205: Zwei Pools mit Sequence- und Message flow

Da innerhalb eines BPD Prozesse mehrerer Geschäftspartner dargestellt werden können und jeder Geschäftspartner eine andere Sicht auf den gleichen Prozess hat, ist es sinnvoll einen "Standpunkt" (Point of view) anzugeben. Die BPMN schreibt nicht vor, wie der Standpunkt in einem BPD hervorgehoben wird. Am einfachsten ist es, die Namen der zugeordneten Business Entities (Geschäftspartner) im Attribut **Beschreibung/Definition** zu pflegen (vgl. Abbildung).

12.1.2 Realisierung der BPMN in ARIS

Obwohl die BPMN nur den Modelltyp **Business Process Diagram (BPD)** vorsieht, können in ARIS zwei Modelltypen genutzt werden: die EPK und der neue Modelltyp Business Process Diagram (BPD). Auf diese Weise können bereits vorliegende Prozesse in ARIS als Private Processes wieder verwendet werden. Die EPK besitzt alle Modellattribute, die laut BPMN für das Business Process Diagram vorgesehen sind. Indem Sie jedoch den Modelltyp Business Process Diagram (BPD) verwenden, bleiben existierende Modelle des Typs EPK frei von Aspekten des B2B-Kontextes. Damit wird die Komplexität der EPK-Modelle nicht durch zusätzliche Beziehungstypen erhöht.

Das neue Business Process Diagram erbt wiederum alle BPMN-relevanten Modellattribute der EPK und alle Sequence flow-relevanten Objekte, Kanten und Symbole. Im neuen Modelltyp Business Process Diagram (BPD) können Sequence flow-relevante Konzepte der EPK wieder verwendet werden. Zusätzlich können aber auch Pools, Lanes und Message flows abgebildet werden.

12.1.3 Die Elemente des Business Process Diagrams

12.1.3.1 Pools und Lanes

Ein Business Process Diagram durch Pools ist strukturiert.

Ein Pool ist ein grafischer Container, in dem ein Satz von Aktivitäten eines Business Entity zusammengefasst wird.

Ein Business Entity kann entweder eine Funktion, ein Anwendungssystem, ein Organisationselement (dazu zählen Organisationseinheit, Organisationseinheitstyp, Gruppe, Personentyp, Stelle, Person, Standort, Systemorganisationseinheit und Systemorganisationseinheitstyp) **oder ein Datenelement** (dazu zählen Fachbegriff, Cluster/Datenmodell, Entitytyp, Beziehungstyp, ERM-Attribut, Business-Objekt, Komplexer Objekttyp, COT-Attribut, Class und Informationsträger.) **sein.**

In BPMN repräsentieren zwei Pools zwei verschiedene Business Entities. Die Strukturierung eines Modells in Pools wird typischerweise im B2B-Kontext verwendet.

Ein Pool bindet die unterschiedlichen Aktivitäten eines Prozesspartners, die mittels Lanes strukturiert und organisiert werden. Auf diese Weise wird eine Abgrenzung zu den Aktivitäten anderer Prozesspartner deutlich (vgl. vorherige Abbildung **Zwei Pools mit Sequence- und Message flow** (Seite 245)).

In einem BPD muss ein Pool nicht notwendigerweise Prozesselemente enthalten. Man kann auch einen leeren Pool („Black Box“) in ein Modell einfügen, wenn man beispielsweise die Zusammenhänge eines beteiligten, aber in den Details unbekanntem Teilprozesses (z. B. eines Geschäftspartners) in einen Gesamtprozess integrieren möchte. Vielleicht möchte man aber auch nur aus Komplexitätsgründen auf die Darstellung von Details eines Teilprozesses verzichten (vgl. Abbildung **E-Mail Voting Process** (Seite 259)).

Pools bestehen aus mindestens einer Lane. Eine Lane kann wiederum weitere Lanes enthalten, die ineinander geschachtelt sind, oder als Matrix definiert werden. Besteht ein Pool aus einer einzigen Lane, erhält der Pool den gleichen Namen wie die Lane. Wenn ein Pool mehr als eine Lane umfasst, müssen die verschiedenen Lane-Namen und ein spezieller Pool-Name angegeben werden.

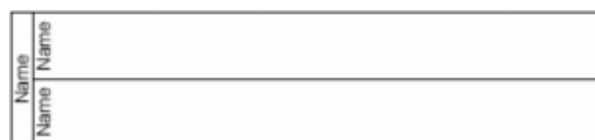


Abbildung 206: Pool mit zwei Lanes nach BPMN

In ARIS sind Pools und Lanes eigene Objekttypen, die zunächst im Modell platziert werden. Innerhalb des Pools kann der Prozess vergleichbar einer EPK modelliert werden. Alle Funktionen, Ereignisse und Regeln des Prozesses werden auf dem Pool-Objekt platziert. Um die Zugehörigkeit dieser Objekte zu einem Pool zu beschreiben, kann die Kante **gehört zu** verwendet werden. Es empfiehlt sich, diese als implizite Beziehung anzulegen. Das Pool-Objekt wird durch die Beziehung des Typs **bildet ab** mit einem Organisationselement, einem Anwendungssystem-

typ-Objekt, einem Datenelement oder einer Funktion verbunden. Hierbei ist zu beachten, dass jeder Pool datenbankweit nur eine Kante dieses Typs besitzen darf. Auch diese Beziehungen sollen implizit angelegt werden

Ein Pool muss gemäß der BPMN nicht durch ein Symbol im Modell repräsentiert werden. Auch dürfen die Grenzen **eines** Pools ausgeblendet sein, insbesondere wenn das Diagramm nur einen Pool enthält (vgl. Abbildung **E-Mail Voting Process** (Seite 259)). Um die Übersichtlichkeit eines Modells mit mehreren Pools sicherzustellen, wird empfohlen, auf den Gebrauch dieser Optionen zu verzichten.

12.1.3.2 Modellierungsrichtlinien für Pools und Lanes

- Innerhalb eines Diagramms darf es nur einen Pool geben, dessen Grenzen nicht sichtbar sind.
- Wenn das Attribut **Pool type** auf **Collaboration** gesetzt ist, soll kein Owner (Attribut **Verantwortlicher**) gepflegt sein.
- Jede Lane darf nur einen übergeordneten Pool besitzen.

12.1.3.3 Sequence flow

Ein Prozess in Form eines Sequence flow beschreibt die Reihenfolge, in der die Aktivitäten eines Prozesses ausgeführt werden. Der Sequence flow verbindet die Objekttypen **Ereignis**, **Aktivität** und **Gateway** miteinander. Sequence flows sind nur innerhalb der Grenzen von Pools erlaubt und dürfen diese nicht überschreiten (vgl. nächste Abbildung).

Der Sequence flow wird durch eine durchgezogene Linie mit einer schwarzen Pfeilspitze dargestellt:



Abbildung 207: Sequence Flow-Kante

Abhängig von der Art des Quell- und Zielobjekts der Kante werden entsprechende Kantentypen gepflegt, z. B. aktiviert, wird ausgewertet von, erzeugt, verknüpft, aktiviert oder führt zu.

12.1.3.4 Modellierungsrichtlinien für Sequence flow-Kanten

- Für Sequence flows, die auf ein XOR-database Gateway oder ein Inclusive Gateway folgen, muss für das Attribut **Condition** ein Wert gesetzt sein.
- Wenn im Attribut **Condition** der Wert **Expression** gesetzt ist, soll am Kantenanfang der Diamant als Symbol platziert werden.
- Wenn das Attribut **Condition** den Wert **Default** hat und das Quellobjekt eine Funktion ist, soll am Kantenanfang das Zeichen **** (umgekehrter Schrägstrich) als Symbol platziert werden.

- Das Zeichen \ (umgekehrter Schrägstrich) darf nicht platziert werden, wenn das Quellobjekt ein Gateway ist.
- Wenn es sich bei dem Quellobjekt um eines der folgenden Symbole handelt, soll keine Condition gesetzt werden:
 - Event-based Gateway
 - Complex Gateway
 - Parallel Gateway
 - Start Event
 - Intermediate Event
- Wenn für eine Sequence flow-Kante der Wert **Default** des Attributs **Condition** aktiviert ist, darf keine Condition gepflegt sein.
- Das Attribut **Condition** darf den Wert **Default** haben, wenn das Quellobjekt eine Funktion oder ein XOR-database Gateway ist.
- Wenn für das Attribut **Condition** der Wert **Expression** gesetzt ist, muss auch das Attribut **Expression** gepflegt sein.

12.1.3.5 Message flow

Ein Message flow beschreibt den Austausch von Informationen zwischen zwei Pools. Der Message flow kann sich entweder direkt zwischen den beiden Pool-Objekten befinden oder zwischen Objekten im Sequence flow der Prozesse des jeweiligen Pools. Nur Message flows dürfen Pool-Grenzen überschreiten und eine Message flow-Kante darf nicht zwischen zwei Objekten des gleichen Pools angelegt werden (vgl. Abbildung **Zwei Pools mit Sequence- und Message flow** (Seite 245)).

Die Kante wird durch eine gestrichelte Linie dargestellt. Am Anfangspunkt der Linie befindet sich ein Kreis, am Endpunkt eine weiße Pfeilspitze.

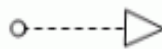


Abbildung 208: Message flow-Kante

Jeder Message flow besteht aus einem Sender-Objekt, einer Kante des Typs **sendet**, einer Kante des Typs **wird empfangen von** und dem Empfänger. Von einem Starterereignis und einem Zwischenereignis gehen keine Message flow-Kanten aus. Ein Endereignis empfängt hingegen keine Message flows, kann aber selbst Sender sein. Lanes, Gateways, Datenobjekte und Text-annotationen sind von Message flows ausgeschlossen.

12.1.3.6 Modellierungsrichtlinien für Message flow-Kanten

Quell- und Zielobjekt müssen unterschiedlichen Pools angehören.

12.1.3.7 Association

Eine Association wird verwendet, um die Bestandteile von Sequence- oder Message flow mit Informationen zu versehen. Diese Informationen können textueller oder grafischer Art sein. Wenn mehrere verschiedene Prozesse im gleichen Diagramm erscheinen, können ihre einzelnen Prozesselemente miteinander über Kanten assoziiert werden.

Die Association wird durch eine gepunktete Linie dargestellt, die bei Bedarf mit offenen Pfeilspitzen versehen werden können. Dies ist insbesondere dann der Fall, wenn es sich um Zuordnungen von Artefakten des Typs **Data Object** handelt.

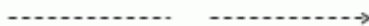


Abbildung 209: Association-Kanten

Abhängig von der Art des Quell- und Zielobjekts der Kante werden entsprechende Kantentypen gepflegt, z. B. **hat Output**, **ist Input für**, **liefert Input für** oder **erzeugt Output auf**.

Von besonderer Bedeutung ist in der BPMN die Zuordnung von Artefakten des Typs Data Object zu Aktivitäten.

Diese Zuordnung ist gerichtet und beschreibt, wie Informationen innerhalb eines Prozesses verwendet und verändert werden. Sie wird im BPD (BPMN) durch folgende Beziehungen vorgenommen:

Funktion - erzeugt Output auf - Datenelemente (insbesondere Informationsträger)

Datenelement (insbesondere Informationsträger) - liefert Input für - Funktion

12.1.3.8 Ereignisse

Ein Ereignis (Event) ist ein Zustand, der während des Ablaufs des Geschäftsprozesses eintritt. Ereignisse beeinflussen den Ablauf des Prozesses. Gewöhnlich stellen sie Auslöser oder Wirkungen innerhalb der Prozesse dar. In Abhängigkeit davon, wann ein Ereignis eintritt, werden drei Ereigniskategorien differenziert: Startereignis, Zwischenereignis und Endereignis. Die drei Kategorien von Events werden in BPMN durch unterschiedliche Symbole dargestellt:



Abbildung 210: Ereigniskategorien

Innerhalb dieser Kategorien werden weitere Spezialisierungen vorgenommen. Die Symbole der drei Ereigniskategorien werden mit der Pflege des Attributs **Event Type** durch zusätzliche Symbole erweitert, wie die drei folgenden Beispiele zeigen:



Abbildung 211: Beispiele für Ereignistypen

Alle für den Objekttyp **Ereignis** relevanten Attribute werden in der Attributtypgruppe **BPMN** zusammengefasst.

12.1.3.9 Modellierungsrichtlinien für Ereignisse

- Für Start events darf der Attributtyp **Event type** nur einen der folgenden Werte besitzen: **Message, Timer, Rule, Link** oder **Multiple**.
- Für End events darf der Attributtyp **Event type** nur einen der folgenden Werte besitzen: **Message, Exception, Cancel, Compensation, Rule, Link, Multiple** oder **Terminate**.
- Für Intermediate events darf der Attributtyp **Event type** nur einen der folgenden Werte besitzen: **Message, Timer, Exception, Cancel, Compensation, Rule, Link** und **Multiple**.
- In Abhängigkeit vom gesetzten Ereignistyp müssen entsprechende Attribute mit Zusatzinformationen gepflegt werden.
- Ein Start event darf mehrere ausgehende Sequence flow-Kanten besitzen. Für das Attribut **Condition** dieser Kanten darf kein Wert gesetzt sein.
- Intermediate events, die eine Exception oder eine Compensation anzeigen, sollen am Rand der Funktion platziert werden.

- Ist ein Intermediate event am Rande einer Funktion platziert, muss ein Wert, jedoch nicht **Link**, gepflegt sein.
- Die Werte **Multiple**, **Rule** und **Cancel** dürfen nicht für Intermediate events gesetzt werden, die sich im normale Sequence flow eines Prozesses befinden.
- Der Wert **Cancel** darf nicht gesetzt werden, wenn
 - das Intermediate event am Rande einer Funktion platziert und das Attribut **Transaction** der Funktion nicht aktiviert ist oder
 - das Ereignis nicht zu einem Prozess gehört, der eine Transaktion beschreibt.
- Wenn ein Intermediate event am Rande einer Funktion platziert ist, darf es nicht Zielobjekt einer Sequence flow-Kante sein.
- Wenn sich ein Intermediate event innerhalb des normalen Sequence flow eines Prozesses befindet, d. h., es nicht am Rande einer Funktion platziert ist, darf es genau eine eingehende Sequence flow-Kante besitzen. Für das Attribut **Event type** des Ereignisses muss dann kein Wert oder darf einer der folgenden Werte gepflegt sein: **Message**, **Timer**, **Exception**, **Link**, **Compensation**.
- Der Wert **Link** darf für Intermediate events im normalen Sequence flow nur dann gesetzt werden, wenn das Quellobjekt ein Gateway ist, für dessen Attribut **Gateway type** der Wert **XOR (Event-based)** gepflegt ist.
- Jedes Intermediate event muss genau eine ausgehende Sequence flow-Kante besitzen.
- Ein Intermediate event, für dessen Attribut **Event type** der Wert **Message** gepflegt ist, darf einen eingehenden Message flow besitzen (eingehende Kante des Typs **wird empfangen von**).
- Ein Intermediate event darf keinen ausgehenden Message flow besitzen (ausgehende Kante des Typs **sendet**).
-

12.1.3.10 Aktivitäten

Eine Aktivität ist eine Tätigkeit, die innerhalb eines Prozesses ausgeführt wird. Sie kann atomar oder nichtatomar (zusammengesetzt) sein. BPMN kennt drei Kategorien von Aktivitäten: Process, Subprocess und Task.

Folgende Symbole sind gemäß BPMN für Aktivitäten vorgesehen:

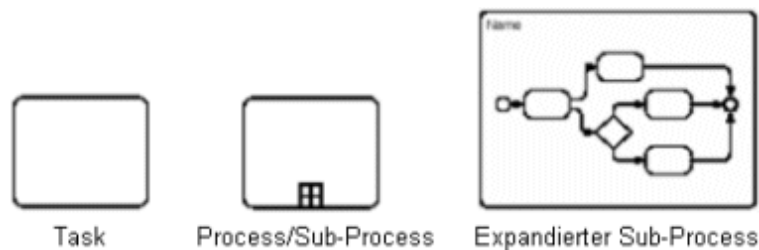


Abbildung 212: Aktivitäten nach BPMN

In ARIS werden Aktivitäten standardmäßig als Funktionen abgebildet:

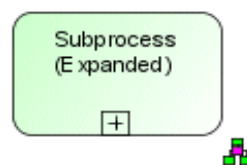


Abbildung 213: Hinterlegte Funktion als Aktivität in ARIS

Die Funktion erhält alle Attribute, die seitens der BPMN für Processes, Subprocesses und Tasks vorgesehen sind. Wie für Ereignisse wird die Attributtypgruppe **BPMN** genutzt, die zusätzliche Untergruppen für die Activity-Typen enthält.

Im Sinne der BPMN beschreibt ein Process eine Aktivität, die innerhalb eines Unternehmens oder einer Organisation durchgeführt wird. Ein Prozess wird durch einen Graphen mit Flussobjekten beschrieben, die einen Satz von anderen Aktivitäten und Steuerungsobjekten darstellen. Prozesse sind hierarchisch strukturiert und können somit auf jeder Detaillierungsstufe definiert werden. Im Gegensatz zum Prozess beschreibt ein Business Process (Geschäftsprozess) in der BPMN einen Satz von Aktivitäten, die über Unternehmens-/Organisationsgrenzen hinweg ausgeführt werden.

Im Sinne der BPMN ist ein Sub-Process eine zusammengesetzte Aktivität, die eine Detailbeschreibung besitzt. Ein Sub-Process erscheint als Objekt innerhalb eines Prozessablaufs (Process flow).

Ein Sub-Process ist in der Regel mit einem detaillierten Prozess hinterlegt. In ARIS wird eine hinterlegte Aktivität nicht wie in BPMN durch ein Pluszeichen markiert, sondern auch durch das Hinterlegungssymbol gekennzeichnet.

Neben der Kennzeichnung einer hinterlegten Funktion sieht BPMN auch vor, den Detailprozess auf der nächst höheren Prozessebene einzublenden. Dies geschieht durch Klicken auf das Pluszeichen.

12.1.3.11 Modellierungsrichtlinien für Aktivitäten

Process

- Wenn das Attribut **Ad hoc = True** ist, dann muss das Attribut **Completion Condition** gepflegt sein.
- Wenn ein Ad hoc-Prozess verfeinert ist, dürfen innerhalb des hinterlegten Modells keine Sequence flows modelliert sein.

Sub-Process

- Wenn für das Attribut **Subprocess Type** der Wert **Independent** verwendet wird, muss auch das Attribut **Process Reference** gepflegt werden.
- Wenn für einen Subprozess das Attribut **Transaction** aktiviert ist, muss auch das Attribut **Transaction ID** gepflegt sein.
- Wenn das Attribut **Loop Type** gepflegt ist, muss auch das Attribut **Loop Condition** gepflegt sein.
- Wenn die Modelle in BPEL4WS übertragen werden sollen, sollte auch geprüft werden, ob für Prozesse mit dem Wert **Standard** für das Attribut **Loop Type** das Attribut **Maximum** gepflegt ist.
- Wenn für das Attribut **Loop Type** der Wert **Standard** gepflegt ist, muss auch das Attribut **Test before** gepflegt sein. Das Attribut **Test before** soll standardmäßig deaktiviert sein.
- Wenn für das Attribut **Loop Type** der Wert **Multi Instance** gepflegt ist, soll auch das Attribut **Parallel Instance Generation** gepflegt sein. Das Attribut **Parallel Instance Generation** soll standardmäßig deaktiviert sein.
- Wenn das Attribut **Loop Type** eines Subprozesses den Wert **Multi Instance** besitzt und zugleich das Attribut **Parallel Instance Generation** aktiviert ist, muss auch das Attribut **Loop Flow Condition** gepflegt sein.
- Wenn in einem Prozess für das Attribut **Loop Flow Condition** der Wert **Complex** gesetzt ist, muss für das Attribut **Complex** ein Ausdruck gepflegt werden, der bestimmt, wann und wie viele Prozessmarken nach dem Subprozess weitergegeben werden.

Task

- Wenn für das Attribut **Task Type** der Wert **Receive** gepflegt ist, soll die Funktion keine ausgehenden Message flow-Kanten besitzen.
- Wenn für das Attribut **Task Type** der Wert **Send** gepflegt ist, soll die Funktion keine eingehenden Message flow-Kanten besitzen.
- Wenn das Attribut **Task Type** nicht gepflegt ist oder die Werte **Script** oder **Manual** gesetzt sind, soll die Funktion keine ausgehenden und keine eingehenden Message flow-Kanten besitzen.

- Für Funktionen, für deren Attribut **Task Type** der Wert **Abstract** gepflegt ist, muss auch das Attribut **Abstract Type** gepflegt werden. Darüber hinaus dürfen sie nur in Pools des Typs **Abstract** oder in **Collaborations** verwendet werden.

12.1.3.12 Gateway

Gateways beschreiben, wie Sequence flows innerhalb eines Prozesses zusammenlaufen oder verzweigen. Sie bestimmen das Verhalten einkommender und ausgehender Kanten. In ARIS werden sie als Objekte des Typs **Regel** dargestellt.

Ähnlich den Events können Gateways unterschiedlichen Typs spezifiziert werden. Abhängig vom Typ werden weitere Symbole in der Mitte des Gateway-Symbols angezeigt.

Folgende Gateway-Symbole werden differenziert:

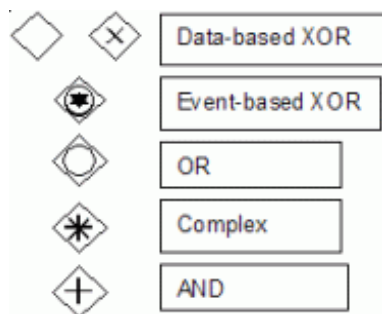


Abbildung 214: Gateway-Typen

Die Spezifikation der BPMN sieht vor, dass für jedes Gateway eine Anzahl von Gates definiert werden muss. In ARIS wird die Anzahl der Gates durch die Anzahl der ein- und ausgehenden Kanten bestimmt. Gate-abhängige Attribute werden daher an den ein- und ausgehenden Sequence flow-Kanten der Regel gepflegt.

Einen Spezialfall stellt das Complex Gateway dar, bei dem die besonderen Attribute **Incoming Condition** und **Outgoing Condition** angegeben werden. Diese Attribute müssen gepflegt sein, wenn es mehrere ein- oder ausgehende Sequence flow-Kanten am betrachteten Gateway gibt. Der Attributinhalt der Incoming Condition kann Sequence flow-Namen und Prozesseigenschaften (Daten) enthalten. Die Outgoing Condition enthält Referenzen auf Sequence flow-IDs und Prozesseigenschaften (Data).

12.1.3.13 Modellierungsrichtlinien für Gateways

- Databased XOR-Gateways: Für alle ausgehenden Kanten eines Data-XOR-Gateways muss für das Attribut **Condition** der Wert **Expression** gesetzt sein und für das Attribut **Condition Expression** ein gültiger Ausdruck verwendet werden.

Sequence flow, speziell nach Gateways:

- Für jedes XOR-Gateway des Typs **XOR (Data-based)** sollte an genau einer ausgehenden Sequence flow-Kante (Kantentyp **aktiviert**) das Attribut **Default-Gateway** gepflegt sein. Es dürfen in keinem Fall mehrere ausgehende Kanten mit diesem Attribut gekennzeichnet sein.
- Für jedes XOR-Gateway des Typs **XOR (Event-based)** muss es mindestens zwei ausgehende Sequence flow-Kanten geben (Typ **aktiviert** oder **führt zu**).

- Für alle ausgehenden Kanten eines Event-based XOR-Gateway darf für das Attribut **Condition** kein Wert gepflegt sein. Das Attribut **Condition Expression** sollte nicht gepflegt werden.
- Folgende Zielobjekte sind für ausgehenden Sequence flow-Kanten eines Event-based XOR-Gateway erlaubt:
 - Funktion, für die der Task Type **Receive** gesetzt ist.
 - Intermediate Events, deren Attributtyp **Event Type** einen anderen Wert besitzen als **Compensation** oder **Multiple**.
- Befindet sich in der Menge der Zielobjekte eine Funktion, darf in dieser Menge kein Event des Typs **Message** vorkommen.
- Besitzt ein Gateway des Typs **OR** keine oder genau eine eingehende Sequence flow-Kante, müssen mindestens zwei ausgehende Sequence flow-Kanten existieren.
- Für alle ausgehenden Sequence flow-Kanten eines OR-Gateway ist für das Attribut **Condition** der Wert **Expression** gesetzt und für das Attribut **Condition Expression** wird ein gültiger Ausdruck verwendet. Der Ausdruck muss auf das aktuelle Gateway bezogen eindeutig sein.
- Besitzt ein OR-Gateway genau eine ausgehende Sequence flow-Kante, darf für das Attribut **Condition** dieser Kante kein Wert gepflegt sein.
- Besitzt ein Gateway des Typs **Complex** keine oder genau eine eingehende Sequence flow-Kante, müssen mindestens zwei ausgehende Sequence flow-Kanten existieren.
- Für alle ausgehenden Kanten eines Complex Gateway muss für das Attribut **Condition** der Wert **None** gepflegt sein, speziell dann, wenn es nur eine ausgehende Kante gibt.
- Wenn ein Complex Gateway mehrere eingehende Sequence flow-Kanten besitzt, muss für das Attribut **Incoming Condition** eine Bedingung gepflegt werden, die Sequence flow-Namen und Prozesseigenschaften (Data) referenziert.
- Wenn ein Complex Gateway mehrere ausgehende Sequence flow-Kanten besitzt, muss für das Attribut **Outgoing Condition** eine Bedingung gepflegt werden, die Sequence flow-Namen und Prozesseigenschaften (Data) referenziert.
- Wenn ein AND-Gateway keine oder genau eine eingehende Sequence flow-Kante besitzt, müssen mindestens zwei ausgehende Sequence flow-Kanten existieren.
- Für alle ausgehenden Sequence flow-Kanten eines AND-Gateway darf für das Attribut **Condition** kein Wert gepflegt sein.

12.1.3.14 Artifact

Artifacts stellen Informationen über den Prozess zur Verfügung, die nicht zu Sequence Flow und nicht zum Message flow gehören. Es werden insgesamt drei Artifact-Typen differenziert: **Data Objects, Groups und Annotations** (die Typliste ist beliebig erweiterbar).

Data Objects sind mit dem Informationsträger oder den Datenelementen in ARIS vergleichbar. Im weitesten Sinne könnten sie aber alle Zuordnungen umfassen. Data Objects beeinflussen weder den Sequence flow noch den Message flow, sie liefern aber Informationen darüber, was während des Prozesses geschieht. Sie geben Aufschluss darüber, wie sich Dokumente, Daten und andere Objekte während des Prozesses verändern.

Bei einer **Group** handelt es sich um die grafische Hervorhebung zusammengehörender Prozesselemente. In ARIS bieten sich grafische Objekte dazu an.

Alternativ können auch Gruppierungen in Betracht gezogen werden. Dies ist allerdings nur dann sinnvoll, wenn die Gruppierung eine Grafik umfasst (vgl. nächste Abbildung, rechts oben).

Annotations entsprechen Bemerkungen zu Objekten oder Kanten. In ARIS werden sie mit Hilfe des Attributs **Bemerkung/Beispiel** realisiert. Wichtig ist, dass das Attribut im Modell platziert wird.

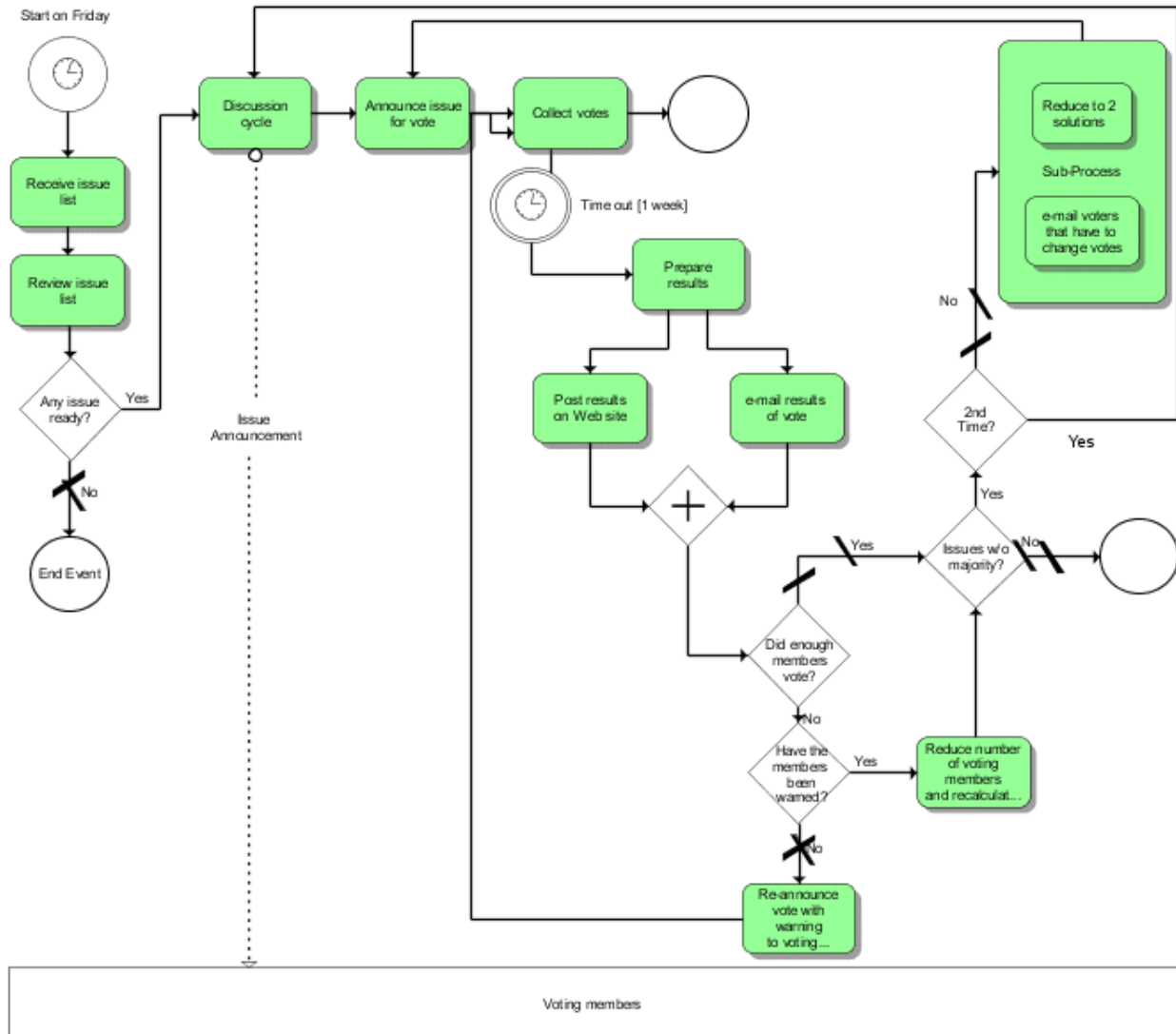


Abbildung 215: E-Mail Voting Process

Diese Abbildung zeigt die beispielhafte Realisierung eines Business Process Diagrams gemäß der BPMN in ARIS. Das Diagramm umfasst zwei Pools, wobei die Pool-Grenzen des oberen Pools ausgeblendet sind. Bei dem unteren Pool wurde auf die Darstellung der einzelnen Prozesselemente verzichtet.

12.1.3.15 Quellenangabe zu den Abbildungen

Abbildung **Zwei Pools mit Sequence- und Message flow** (Seite 245):

Business Process Modeling Notation, Working Draft (1.0); BPMI.org; August 25, 2003; Seite 85.

Abbildung **Pool mit zwei Lanes nach BPMN** (Seite 248):

Business Process Modeling Notation, Working Draft (1.0); BPMI.org; August 25, 2003; Seite 87.

Abbildung **Ereigniskategorien** (Seite 252) und Abbildung **Beispiele für Ereignistypen** (Seite 252):

Business Process Modeling Notation, Working Draft (1.0); BPMI.org; August 25, 2003; Seite 27.

Abbildung **Aktivitäten nach BPMN** (Seite 254):

Business Process Modeling Notation, Working Draft (1.0); BPMI.org; August 25, 2003; Seite 28.

Abbildung **Gateway-Typen** (Seite 257):

Business Process Modeling Notation, Working Draft (1.0); BPMI.org; August 25, 2003; Seite 28.

13 Literaturverzeichnis

13.1 Allgemeine Literaturliste

- Brombacher, R.; Bungert, W.: Praxis der Unternehmensmodellierung 1992
Praxis der Unternehmensmodellierung, Seminar der IDS Prof. Scheer GmbH, Bad Soden/Taunus, 12./13. November 1992.
- Chen, P. P.: Entity-Relationship Model 1976
The Entity-Relationship model: Toward a Unified View of Data, in: ACM Transactions on Database-Systems, Vol. 1 (1976), No. 1, S. 9 - 36.
- Hoffmann, W.; Kirsch, J.; Scheer, A.-W.: Modellierung mit Ereignisgesteuerten Prozeßketten 1993
Modellierung mit Ereignisgesteuerten Prozeßketten (Methodenhandbuch, Stand: Dezember 1992), in: Scheer, A.-W. (Hrsg.), Veröffentlichungen des Instituts für Wirtschaftsinformatik, Heft 101, Saarbrücken Januar 1993.
- Keller, G.; Hechler, H.-J.: Informationsmodell 1991
Konzeption eines integrierten Informationsmodells für die Kostenrechnung des SAP-Systems, in: Scheer, A.-W. (Hrsg.): 12. Saarbrücker Arbeitstagung Rechnungswesen und EDV: Kritische Erfolgsfaktoren in Rechnungswesen und Controlling, Heidelberg 1991, S. 67 - 106.
- Scheer, A.-W.: Architektur integrierter Informationssysteme 1992
Architektur integrierter Informationssysteme - Grundlagen der Unternehmensmodellierung, 2. Aufl., Berlin et al. 1992.
- Scheer, A.-W.: EDV-orientierte Betriebswirtschaftslehre 1990
EDV-orientierte Betriebswirtschaftslehre - Grundlagen für ein effizientes Informationsmanagement, 4. Aufl., Berlin et al. 1990.
- Scheer, A.-W.: Wirtschaftsinformatik 1994
Wirtschaftsinformatik - Referenzmodelle für industrielle Geschäftsprozesse, 5. Aufl., Berlin et al. 1994.
- Schlageter, G.; Stucky, W.: Datenbanksysteme 1983
Datenbanksysteme: Konzepte und Modelle, 2. Aufl. Stuttgart 1983.
- Seubert, M.: SAP-Datenmodell 1991
Entwicklungsstand und Konzeption des SAP-Datenmodells, in: Scheer, A.-W. (Hrsg.): Datenbanken 1991 - Praxis relationaler Datenbanken: Vom Datenmodell zur Implementierung, Fachtagung vom 04.-05.06.1991 in Saarbrücken, Saarbrücken 1991, S. 87 - 109.
- Sinz, E. J.: Entity-Relationship-Modell 1990
Das Entity-Relationship-Modell (ERM) und seine Erweiterungen, in: HMD Theorie und Praxis der Wirtschaftsinformatik, 27 (1990), Heft 152, S. 17 - 29.

- Scheer, A.-W.: ARIS - Vom Geschäftsprozeß zum Anwendungssystem. 3. Aufl. Berlin et al. 1998
- Scheer, A.-W.: ARIS - Modellierungsmethoden, Metamodelle, Anwendungen. 3. Aufl. Berlin et al. 1998
- Scheer, A.-W., Jost, W.: ARIS in der Praxis 2002
Gestaltung, Implementierung und Optimierung von Geschäftsprozessen, Berlin, Heidelberg, New York 2002.
- Scheer, A.-W., Abolhassan, F., Jost, W., Kirschmer, M.: Business Process Excellence 2002
ARIS in Practice , Berlin, Heidelberg, New York 2002.

13.2 Themenbezogene Literaturliste

13.2.1 Unified Modeling Language in ARIS

13.2.1.1 UML-Spezifikation

UML-Spezifikation usw.: <http://www.uml.org>.

13.2.1.2 Anwendung von UML

- Burkhardt, R.: UML Unified Modeling Language, Objektorientierte Modellierung für die Praxis, Bonn 1997.
- Fowler, M.; Scott, K.: UML Distilled - Applying the Standard Object Modeling Language, Reading et al. 1997.
- Oesterreich, B.: Objektorientierte Softwareentwicklung mit der UML, 3. Aufl., München-Wien 1997.

13.2.1.3 UML und Geschäftsprozessmodellierung

- Ambler, S. W.: What's Missing from the UML? Techniques that can help model effective business applications, Object Magazine 7 (1997)8
- Loos, P.; Allweyer, Th.: Process Orientation and Object-Orientation - An Approach for Integrating UML and Event-Driven Process Chains (EPC), Publication of the Institut für Wirtschaftsinformatik, Paper 144, Saarbrücken 1998.

13.2.2 Verwendung der Modellierungsmethode Object Modeling Technique (OMT)

Rumbaugh, J.; Blaha, M.; Premerlani, W.; Eddy, F.; Lorensen, W.: Objektorientiertes Modellieren und Entwerfen. München et al. 1993.

13.2.3 Methoden für das Wissensmanagement

13.2.3.1 Wissensmanagement allgemein

- Probst, G.; Raub, S.; Romhardt, K.: Wissen managen. Wie Unternehmen ihre wertvollste Ressource optimal nutzen. Frankfurt Wiesbaden 1998.
- Bürgel, H. D. (Hrsg.): Wissensmanagement. Schritte zum intelligenten Unternehmen. Berlin et al. 1998.

▪

13.2.3.2 ARIS-Einsatz für das Wissensmanagement

- Allweyer, Th.: Modellbasiertes Wissensmanagement. In: IM Information Management & Consulting 13 (1998) 1, S. 37-45.
- Allweyer, Th.: Wissensmanagement mit ARIS-Modellen. In: Scheer, A.-W.: ARIS - Vom Geschäftsprozeß zum Anwendungssystem. 3. Aufl. Berlin et al. 1998, S. 162-168.

▪

13.2.4 Balanced Scorecard-Methode

Kaplan, Robert/Norton, David: The Balanced Scorecard-Measures that drive Performance, Harvard Business Review Januar/Februar 1992.

13.2.5 IT City Planning

- Schulman, Jeff: A New View of Architectures Needed for New Business Drivers, Gartner Briefing Presentations.
- Longépé, Christoph: Le projet d'urbanisation du système d'information, Dunod, Paris, 2001

▪

13.2.6 Business Process Modeling

Business Process Modeling Notation, Working Draft (1.0); BPMI.org; August 25, 2003.

14 Abbildungsverzeichnis

Abbildung 1: Modell eines Unternehmensprozesses	4
Abbildung 2: Sichten des Prozessmodells	5
Abbildung 3: Zerlegungssichten des Prozessmodells	6
Abbildung 4: Beschreibungen eines Informationssystems	8
Abbildung 5: ARIS-Konzept	9
Abbildung 6: Beispiel für ein Vorgangskettendiagramm	12
Abbildung 7: Darstellung der Funktion Kundenanfrage prüfen	14
Abbildung 8: Funktionsbaum (Ausschnitt)	15
Abbildung 9: Funktionsbaum objektorientiert	16
Abbildung 10: Funktionsbaum prozessorientiert	17
Abbildung 11: Funktionsbaum verrichtungsorientiert	18
Abbildung 12: Y-Diagramm	20
Abbildung 13: SAP-Applikationendiagramm	21
Abbildung 14: Zieldiagramm	22
Abbildung 15: Grafische Darstellung eines Anwendungssystemtyps	23
Abbildung 16: Modularer Aufbau eines Anwendungssystemtyps	24
Abbildung 17: Grafische Darstellung des DV-Funktionstyps	25
Abbildung 18: Zuordnung der Funktionen zu Anwendungssystemtypen	25
Abbildung 19: Konfigurierung des Anwendungssystemtyps	26
Abbildung 20: Masken- und Listenzuordnungen	27
Abbildung 21: Grafische Darstellung des Anwendungssystems und des Moduls	28
Abbildung 22: Zuordnung der Anwendungssysteme zu ihren Anwendungssystemtypen	28
Abbildung 23: Unterschiedlicher modularer Aufbau zweier Anwendungssysteme des gleichen Typs	29
Abbildung 24: Zuordnung von Anwendungssystemtypen, Programmteiltypen und Programmteilen	29
Abbildung 25: Beispiele für Entitytypen	32
Abbildung 26: Beispiele für Attribute des Entitytyps Kunden	33
Abbildung 27: Beispiel für einen Beziehungstyp	34
Abbildung 28: Kardinalitäten der Beziehungen zwischen zwei Entitytypen	35
Abbildung 29: Darstellung der Kardinalitäten im ERM	35
Abbildung 30: ERM für eine Stückliste	36
Abbildung 31: Zuordnung von Attributen im ERM	37
Abbildung 32: Klassifizierung von Kunden	38

Abbildung 33: Generalisierung/Spezialisierung	39
Abbildung 34: Vollständig beschriebene Spezialisierung	40
Abbildung 35: Beispiel für eine Aggregation	40
Abbildung 36: Aggregation mit uminterpretiertem Beziehungstypen	41
Abbildung 37: Datencluster (grafisches Symbol)	42
Abbildung 38: Datenclustersicht auf mehrere Objekte	42
Abbildung 39: Gruppierung	42
Abbildung 40: Ober-/Untergrenze (1)	43
Abbildung 41: Ober-/Untergrenze (2)	43
Abbildung 42: Ober- Untergrenze (3)	43
Abbildung 43: Ober- Untergrenze (4)	44
Abbildung 44: Existentielle Abhängigkeit	44
Abbildung 45: Fachbegriffe (1)	45
Abbildung 46: Fachbegriffe (2)	46
Abbildung 47: Zuordnung von ERM-Attributen zu einem Entitytyp	47
Abbildung 48: Darstellung einer Attributtypgruppe	47
Abbildung 49: eERM und SAP-ERM-Darstellung	49
Abbildung 50: Datenmodell in IE-Notation	50
Abbildung 51: Datenmodell in SeDaM-Notation	50
Abbildung 52: Begriffe und Darstellungsformen des eERM	51
Abbildung 53: DTD-Elementtyp mit reinem Textinhalt	52
Abbildung 54: Elementtypen mit gemischtem Inhalt und Umsetzung in der DTD	53
Abbildung 55: Elementtyp mit einem Aufzählungs-Attributtyp	57
Abbildung 56: Beispiel für ein Materialdiagramm	60
Abbildung 57: Data Warehouse im Sternen-Schema	61
Abbildung 58: Berechtigungshierarchie	62
Abbildung 59: Beispiel eines CD-Diagramms	63
Abbildung 60: Beispiel eines Kostenartendiagramms	64
Abbildung 61: Informationsträgerdiagramm	65
Abbildung 62: Grafische Darstellung der Relation	66
Abbildung 63: Zuordnung der Attribute und der Datenobjekte des Fachkonzeptes	67
Abbildung 64: Attributzuordnungsdiagramm	67
Abbildung 65: Definition einer View	68
Abbildung 66: Zuordnung des ERM-Beziehungstyps zum Attribut	68
Abbildung 67: Beispiel für ein Modell "Systemattribute"	69
Abbildung 68: Systemattributdomäne	70

Abbildung 69: Grafische Darstellung von Tabelle und Feld	71
Abbildung 70: Feldzuordnungen	71
Abbildung 71: Zuordnung der Objekte des Fachkonzeptes und des DV-Konzeptes	72
Abbildung 72: Tabellenexemplare	73
Abbildung 73: Organisatorische Zergliederung nach Produkten	75
Abbildung 74: Gemischte Organisationsformen	75
Abbildung 75: Organigramm	77
Abbildung 76: Organigramm mit Stellen- und Personenzuordnung	78
Abbildung 77: Personentypen	79
Abbildung 78: Standortzuordnungen	80
Abbildung 79: Standorthierarchien	81
Abbildung 80: Beispiel eines Schichtkalenders	83
Abbildung 81: Grafische Darstellung eines Netztyps	84
Abbildung 82: Netztopologie	85
Abbildung 83: Netzdiagramm mit Standortzuordnung	86
Abbildung 84: Netzdiagramm mit Hardwarekomponenten und Standortzuordnung	87
Abbildung 85: Beispiel für ein Modell "Technische Ressourcen"	90
Abbildung 86: Zuordnung der Organisationseinheiten zu Funktionen	92
Abbildung 87: Ereignisse (grafische Darstellung)	93
Abbildung 88: Beispiel für eine EPK	94
Abbildung 89: Beispiele für Regeln	94
Abbildung 90: Verknüpfungsoperatoren (Regeln)	96
Abbildung 91: Und-Verknüpfung auslösender Ereignisse	97
Abbildung 92: Oder-Verknüpfung auslösender Ereignisse	97
Abbildung 93: Exklusiv-oder-Verknüpfung auslösender Ereignisse	97
Abbildung 94: Und-Verknüpfung erzeugter Ereignisse	98
Abbildung 95: Oder-Verknüpfung erzeugter Ereignisse	98
Abbildung 96: Exklusiv-oder-Verknüpfung erzeugter Ereignisse	98
Abbildung 97: Und-Verknüpfung von Funktionen mit erzeugten Ereignissen	99
Abbildung 98: Oder-Verknüpfung von Funktionen mit erzeugten Ereignissen	99
Abbildung 99: Exklusiv-oder-Verknüpfung von Funktionen mit erzeugten Ereignissen	99
Abbildung 100: Und-Verknüpfung von Funktionen mit auslösenden Ereignissen	100
Abbildung 101: Beispiel für ein Funktionszuordnungsdiagramm (I/O)	101
Abbildung 102: Erweiterte Darstellung des Funktionszuordnungsdiagramms	102
Abbildung 103: EPK mit Input-/Output-Daten	103
Abbildung 104: VKD mit Input-/Output-Daten	104

Abbildung 105: Informationsflussdiagramm mit geöffnetem Hinterlegungs-Assistenten	105
Abbildung 106: Beispiel für ein Ereignisdiagramm	106
Abbildung 107: Beispiel für eine Vorgangskette (Fachkonzept)	107
Abbildung 108: EPK mit Funktionen, Daten, Organisationseinheiten und Ereignissen	108
Abbildung 109: Wertschöpfungskette	109
Abbildung 110: Abbildung komplexer Verknüpfungen im Regeldiagramm	110
Abbildung 111: Input/Output Diagramm	112
Abbildung 112: Klassendiagramm für die Klasse Kundenauftrag	113
Abbildung 113: Prozessauswahlmatrix (Auszug aus dem Referenzmodell R/3 der SAP AG)	114
Abbildung 114: Auszug einer EPK (Materialfluss)	116
Abbildung 115: EPK (Spaltendarstellung)	118
Abbildung 116: Rollenzuordnungsdiagramm (RAD - Role Allocation Diagram)	119
Abbildung 117: Beispiel eines Business Controls Diagram	120
Abbildung 118: DW-Transformation - Datentransformation eines Data Warehouse	121
Abbildung 119: eBusiness Szenario Diagram-Beispiel: Automobilbranche	123
Abbildung 120: Beispiel eines Gliederungsmodells (Auszug aus der Norm VDA 6.2)	124
Abbildung 121: Beispielsachverhalt in den Modelltypen EPK, Industrial Process und Office Process	127
Abbildung 122: Beispiel einer aus einer EPK entstandenen PPK	129
Abbildung 123: Prozessinstanzierungsmodell	131
Abbildung 124: Beispiel eines RAMS-Diagramms	132
Abbildung 125: Rollendiagramm	135
Abbildung 126: Aufbau eines C3-Modells	137
Abbildung 127: Beispiel für das Maskendesign eines Registrierungsdialogs und die Umsetzung in C++	139
Abbildung 128: Beispiel für die Maskennavigation mit Ereignissen	140
Abbildung 129: Beispiel Geschäftsfelder-Matrix	141
Abbildung 130: Report	142
Abbildung 131: Datenfluss zwischen Anwendungssystemtypen	143
Abbildung 132: I/O-Daten auf DV-Konzept-Ebene	143
Abbildung 133: Zugriffsrechte	144
Abbildung 134: Definition der Verantwortlichkeiten	144
Abbildung 135: Zugriffsdiagramm (Auszug)	146
Abbildung 136: Beispiel eines Programmablaufplans (PA)	147
Abbildung 137: Beispiel eines Maskendiagramms	148

Abbildung 138: Aus dem Maskendiagramm der vorherigen Abbildung abgeleitete Bildschirmmaske	149
Abbildung 139: Datenfluss	151
Abbildung 140: Input-/Outputbeziehungen	152
Abbildung 141: Zuordnungen zur Hardwarekomponente	153
Abbildung 142: Hardwarekomponente als Plattform	154
Abbildung 143: Anwender und Anwendungssystem	155
Abbildung 144: Standortzuordnungen	156
Abbildung 145: Exemplarischer Leistungsaustausch in einem Softwarehaus	158
Abbildung 146: Leistungsbaum	159
Abbildung 147: Beispiel eines Produktzuordnungsdiagramms	160
Abbildung 148: Produktzuordnungsdiagramm - Girokonto	161
Abbildung 149: Produktzuordnungsdiagramme - Verkaufsprodukte	161
Abbildung 150: Klassifizierung der Produktgruppe "Einwohner- und Staatsangehörigkeitsangelegenheiten" mit Hilfe eines Produktbaumes	163
Abbildung 151: Produktauswahlmatrix des Sozialamts	164
Abbildung 152: Wettbewerbskräfte im Sportwagenmarkt	165
Abbildung 153: Darstellung von Instanzen	168
Abbildung 154: Darstellung von Klassen	168
Abbildung 155: Verbindung von Instanzen zu Klassen	168
Abbildung 156: Zuordnung von Attributen zu Klassen	169
Abbildung 157: Zuordnung von Operationen zu Klassen	169
Abbildung 158: Assoziationen zwischen Instanzen	169
Abbildung 159: Assoziationen zwischen Klassen	170
Abbildung 160: Ternäre Beziehung zwischen Klassen	170
Abbildung 161: Modellieren einer Assoziation als Klasse	171
Abbildung 162: Darstellung einer qualifizierten Assoziation	171
Abbildung 163: Darstellung von Ordnungen bei Assoziationen	171
Abbildung 164: Aggregation zwischen Klassen	172
Abbildung 165: Generalisierung und Vererbung	172
Abbildung 166: Darstellung von Constraints auf Attributen	173
Abbildung 167: Darstellung der Constraints bei Assoziationen	173
Abbildung 168: Beispiel eines OMT Object Model	174
Abbildung 169: Darstellung von Initial-, Final- und Übergangszuständen	175
Abbildung 170: Darstellung des Übergangs zwischen Zuständen	175
Abbildung 171: Darstellung weiterer Informationen beim Zustandsübergang	175

Abbildung 172: Beispiel eines OMT Dynamic Model	176
Abbildung 173: Darstellung von Datenspeichern	177
Abbildung 174: Darstellung von Prozessen	177
Abbildung 175: Darstellung von Actors (Handlungsobjekten)	177
Abbildung 176: Darstellung von Datenflüssen	178
Abbildung 177: Darstellung des Splittings von Datenflüssen	178
Abbildung 178: Beispiel eines OMT Functional Model	179
Abbildung 179: Wissensstrukturdiagramm	185
Abbildung 180: Wissenslandkarte - organisationseinheitenbezogen	186
Abbildung 181: Wissenslandkarte - Matrixdarstellung	187
Abbildung 182: Wissensverarbeitung in einer EPK	188
Abbildung 183: Die BSC als Handlungsrahmen für das strategische Management	205
Abbildung 184: Die Perspektiven der BSC	207
Abbildung 185: BSC-Ursache-Wirkungsdiagramm	216
Abbildung 186: BSC-Kennzahlenzuordnungsdiagramm	218
Abbildung 187: Kennzahlenbaum	220
Abbildung 188: Transaktionsmöglichkeiten im E-Business	223
Abbildung 189: Die Objekte im eBusiness Szenario Diagram	225
Abbildung 190: Ausschnitt aus eBusiness Szenario "Online-shop"	228
Abbildung 191: Ausschnitt aus dem Pipeline Diagram	229
Abbildung 192: Ausschnitt aus der DTD: Bestellung	230
Abbildung 193: Prozess-, IS- und IT-Sicht	233
Abbildung 194: Zonen eines Unternehmens-Informationssystems	234
Abbildung 195: Zone in Distrikte untergliedert	234
Abbildung 196: Distrikt in Building Cluster untergliedert	235
Abbildung 197: Aufteilung des Building Cluster "Personalbetreuung" in Functional Blocks	235
Abbildung 198: Fähigkeiten und IS Services des Functional Block "Gehälter"	236
Abbildung 199: Kante "ist Eigner von" zwischen Symbolen der IS-Sicht und Beziehungs- und Entitytypen	237
Abbildung 200: Kante "unterstützt" zwischen IS-Elementen und Funktion	238
Abbildung 201: Subsystemstruktur des DATEV-Systems	240
Abbildung 202: Detaillierte Beschreibung von IT-Elementen im Zugriffsdiagramm	241
Abbildung 203: Einflüsse und Auswirkungen der technischen Infrastruktur	242
Abbildung 204: Einbindung von IS- und IT-Elementen in einen zeitlich-logischen Ablauf	243
Abbildung 205: Zwei Pools mit Sequence- und Message flow	247
Abbildung 206: Pool mit zwei Lanes nach BPMN	248

Abbildung 207: Sequence Flow-Kante	249
Abbildung 208: Message flow-Kante	250
Abbildung 209: Association-Kanten	251
Abbildung 210: Ereigniskategorien	252
Abbildung 211: Beispiele für Ereignistypen	252
Abbildung 212: Aktivitäten nach BPMN	254
Abbildung 213: Hinterlegte Funktion als Aktivität in ARIS	254
Abbildung 214: Gateway-Typen	257
Abbildung 215: E-Mail Voting Process	260

15 Index

A

- A2A • 222
- A2B • 222
- A2C • 222
- Abdeckung • 186
- Abdeckungsgrad
 - Angestrebt • 182
 - Wissen • 182
 - Wissenslandkarte • 186
- Abhängigkeit
 - Identifizierungs- und Existenzabhängigkeit • 44
 - SAP - SERM • 48
- Ablauforganisation • 74
- Abstract Business Process • 245
- Abteilung • 11
- Aggregation • 38
- Aggregationsoperator • 38
- Aktionsprogramm • 209
- Aktivität
 - Process • 254
 - Sub-Process • 254
 - Task • 254
- Alarmgrenze • 212
- Analyse • 199
 - Strategie • 205
- Änderungsgeschwindigkeit • 182
- Änderungszyklen • 7
- Anforderungsprofil • 62
- Annotation • 259
- Anwendungssystem
 - Beschreibung der betriebswirtschaftlichen Problemstellung • 11
 - Komponente • 11
- Anwendungssystemtyp
 - Datenflüsse zwischen Anwendungssystemtypen • 143
 - Definition • 28
 - DV-Konzept - Anwendungssystemtypdiagramm • 23
 - Exemplare eines Anwendungssystemtyps • 28
 - Modularer Aufbau • 23
- Anwendungssystemtypdiagramm • 23
- Anwendungssystemtyphierarchie • 239
- Anwendungsszenario
 - Aufgaben • 189
 - Lösung • 189
 - Übersicht • 189
- Applikationen • 195
- Applikationsmodule • 195
- Arbeitsanweisungen • 194
- Arbeitsqualität • 198
- Architektur integrierter Informationssysteme (ARIS)
 - Beschreibungsebenen • 7
 - Einführung • 1
 - Gesamtsicht • 107
 - Konzept • 3
 - Organisationssicht - Fachkonzept • 74
 - Steuerungs-/Prozesssicht des ARIS-Hauses • 74
- Architekturen
 - Rahmenkonzepte • 1
- ARIS
 - Architektur integrierter Informationssysteme • 1
 - ARIS for SAP • 199
 - Integration von Lotus Notes • 194
 - Objektorientiertes, erweitertes Konzept • 113
- Artifact • 259
- ASAP • 199
- Attribut
 - Abdeckung qualitativ • 186
 - Abdeckungsgrad • 186
 - Ausprägungen • 32
 - Domäne • 32
 - ERM-Grundmodell • 32
 - Erweiterung um Konstruktionsoperatoren • 38
 - Fachkonzept • 31
 - Relationendiagramm, Attributzuordnungsdiagramm • 66
 - Unterscheidung zum Entitytyp • 32
 - Vererbt • 38
 - Wertebereich • 32
- Attributtypen
 - DTD • 55
- Attributtypgruppe • 47
 - Definition • 47

- Attributzuordnungsdiagramm • 66
- Aufbau von QM-Dokumentationen • 197
- Aufbauorganisation • 74
 - Aufbauorganisatorische Gestaltung • 74
 - Gebietsorientiert • 74
 - Produktbezogen • 74
- Aufgabe
 - Aufbau von QM-Dokumentationen • 197
 - Beschreibung der DV-Prozesse • 200
 - Beschreibung der Prozess- und Organisationsstrukturen • 196
 - Datenbank-/Zugriffsverwaltung • 193
 - Datenstrukturierung/Datenbankentwurf • 193
 - Dokumentation der informationstechnischen Infrastruktur • 195
 - Dokumentation der Organisationsstruktur • 192
 - Dokumentation der Unternehmensfunktionen • 192
 - Dokumentation der Unternehmenswertschöpfung • 192
 - Dokumentation der Unternehmensziele • 192
 - Entwicklung der Systemoberfläche • 200
 - Ermittlung des informationstechnischen Infrastrukturbedarfs • 195
 - Festlegung von Anwendungssystemen und Modulen • 200
 - Integration von Lotus Notes • 194
 - Kostenstellenanalyse • 196
 - Phase Analyse, Spezifikation (Project Preparation) • 199
 - Phase Design (Business Blueprint) • 199
 - Process Warehousing • 192
 - Prozess-Customizing von Workflow-Management-Systemen • 202
 - Prozessdokumentation • 192
 - Prozesskalkulation • 196
 - Reorganisationsdurchführung • 198
 - Reorganisationsmaßnahmen • 198
 - Software-Entwicklung und -einführung • 200
 - Wissenskategorisierung • 201
 - Wissenslandkarte bzw. Yellow Page • 201
 - Wissensverarbeitung in Geschäftsprozessen • 201
 - Zertifizierungsunterlagen • 197
 - Zertifizierungsvorgehen • 197
 - Zugriffsberechtigungen • 195
- Aufgabenträger
 - Aufgabenträger Organigramm • 77
- Funktionen mit Organisation - EPK, Funktions-/Org.-Ebenendiagramm • 91
- Organisatorische Gestaltung der Unternehmen • 74
- Ausgangstatbestand • 11
- Ausprägungsebene • 32
- B**
- B2A • 222
- B2B • 122, 222
- B2C • 122, 222
- Balanced Scorecard
 - Akzeptanz • 209
 - Ansatz • 204
 - Bekanntheit • 209
 - BSC-Kennzahlenzuordnungsdiagramm • 217
 - eBusiness Senario • 227
 - Funktionsbaum • 214
 - Gliederungsmodell • 214
 - Konzept • 204
 - Methode • 203
 - Objektbeziehung zu ARIS-Modellen • 221
 - Organigramm • 214
 - Performance Management • 214
 - Performance Measurement • 214
 - Phasen • 205
 - Struktur des Balanced Scorecard-Systems festlegen • 214
 - Ursache-Wirkungs-Beziehungen festlegen • 215
 - Wertschöpfungskettendiagramm • 214
- Balanced Scorecard-Ansatz • 211
- Balanced Scorecard-System • 208
- Barriere • 211
- Batch- Bearbeitung • 11
- Bearbeiter • 3
- Bedeutung • 182
 - Künftig • 182
 - Wissenskategorie • 182
- Begriff
 - Alarmgrenze • 212
 - Datenquelle • 212
 - Erfolgsfaktor • 212
 - Indikator typ • 212
 - Ist-Wert • 212
 - Kennzahl • 212
 - Maßnahme • 212
 - Maximalwert • 212
 - Minimalwert • 212
 - Perspektiv • 212

- Plan-Wert • 212
- Strategie • 212
- Strategisches Ziel • 212
- Toleranzbereich • 212
- Vision • 212
- Warngrenze • 212
- Ziel-Wert • 212
- Benutzer • 195
- Benutzerberechtigungen • 195
- Benutzergruppen • 195
- Benutzeroberfläche • 23
- Benutzerprofil • 119
- Berechtigung • 134
- Berechtigungsart • 134
- Berechtigungskonflikt • 62
- Berechtigungskonzept • 119
- Berichtswesen • 211
- Beschreibendes ERM-Attribut • 47
- Beschreibung der DV-Prozesse • 200
- Beschreibungsebene • 7
- Betrieblich • 196
- Betriebsmittel • 115
 - Definition • 88
- Betriebsmittelklasse
 - Definition • 88
- Betriebsmitteltyp • 115
 - Definition • 88
- Betriebssystem • 23
- Betriebssystemtyp • 23
- Betriebswirtschaftliche Problemstellung
 - Anwendungsfälle • 189
 - ARIS-Architektur (Konzept) • 3
 - Beschreibung • 11
 - Beschreibungsebenen • 7
- Bewertungsskala • 134
- Beziehung
 - Aggregierend • 48
 - Arten • 32
 - B2B • 122
 - B2C • 122
 - Beziehung • 32
 - Existenz • 32
 - Hierarchisch • 48
 - Implizit • 1, 117
 - Komplexitätsgrad • 32
 - Referenziell • 48
- Beziehungsausprägung
 - Minimalen Anzahl • 43
- Beziehungskomplexität • 48
- Beziehungstyp
 - Arten • 32
 - Bildung • 38
 - Fachkonzept • 31
 - Identifizierungs- und Existenzabhängigkeit • 44
 - Schwach • 48
- Bezugsumfeld
 - Zustände • 3
- Bild • 192
- Bildhafte Darstellung
 - Industrial Process • 125
 - Office Process • 125
- Bildschirmmaske • 23, 148, 200
 - Automatisch ableiten • 148
- Black Box • 248
- Bottom-Up-Prozess • 210
- BPEL
 - Einführung • 1
 - Process Modeling • 245
- BPMN • 245
- Branche • 192
- Brüche
 - Medienbrüche • 11
 - Organisatorisch • 198
- BSC-Kennzahlenzuordnungsdiagramm
 - BSC-Kennzahlenzuordnungsdiagramm • 217
 - Symbole • 217
- BSC-Ursache-Wirkungsdiagramm
 - Struktur • 214
 - Symbole • 215
 - Ursache-Wirkungs-Beziehungen festlegen • 215
- Budget
 - Budgetierungsprozess • 209
 - Vorgaben • 209
- Budgetierungsprozess • 211
- Budgetplanung • 196
- Building Cluster • 233
- Business Blueprint • 199
- Business Controls Diagram • 120
- Business Documents • 224
- Business Participant
 - eBusiness Scenario Diagram • 122
 - eBusiness Scenario Diagramm und seine Objekte • 224

- Business Process Diagram
 - Business Process Modeling • 245
 - Prozessklassen • 245
 - Realisierung der BPMN • 247
 - Business Process Execution Language
 - Business Process Modeling • 245
 - Business Process Modeling
 - Informationstechnologien • 245
 - Kommunikationstechnologien • 245
 - Business Process Modeling Notation • 245
 - Business to Business • 222
 - Business to Consumer • 222
 - Business/Consumer to Administration • 222
- C**
- C2A • 222
 - C2B • 222
 - C3-Methode • 137
 - Cash Flow • 206
 - Casual Model • 136
 - Casual Object • 136
 - Change Management-Projekt • 137
 - CIM • 74
 - Cluster • 38
 - Clusterinstanz • 128
 - Collaboration Processes • 245
 - Collaborative Business Maps • 227
 - Collective Writing • 194
 - Cost Driver
 - CD-Diagramm • 63
 - Hierarchie • 63
 - CRM-System • 231
- D**
- Data Object • 259
 - Data Warehouse • 121
 - Data
 - Warehouse-Datentransformationsdiagramm • 121
 - Data Warehouse-Methode • 217
 - Data Warehouse-Strukturdiagramm • 61
 - Data Warehousing • 193
 - Datenbank-/Zugriffsverwaltung • 193
 - Datenbankmanagementsystem • 23
 - Datenbankmanagementsystemtyp • 23
 - Datenbanksystem • 71
 - Objektorientiert • 193
 - Datencluster • 38
 - Datenfluss
 - Prozesskette • 107
 - Datenherkunft • 217
 - Datenmodell
 - Erstellungsprozess • 48
 - Semantisch • 31
 - Datenobjekt • 48
 - Datenquelle • 212
 - Datenredundanzen • 198
 - Datensicht • 11
 - DV-Konzept • 66
 - Fachkonzept • 31
 - Implementierung • 71
 - Datenstrukturen • 193
 - Datenstrukturierung/Datenbankentwurf • 193
 - Definition • 60, 88
 - Dialogbearbeitung • 11
 - Dienstleistung • 157
 - Dimensionstabelle • 61
 - Dispositionsgrundlage • 196
 - District • 233
 - Dokument • 192
 - HTML • 122
 - XML • 122
 - Dokumentation
 - Organisationsstruktur • 192
 - Unternehmensfunktionen • 192
 - Unternehmenswertschöpfung • 192
 - Unternehmensziele • 192
 - Dokumentiertes Wissen • 181
 - Dokumenttyp-Definition • 52
 - Domäne
 - ERM-Grundmodell • 32
 - Implementierung • 71
 - Double loop learning • 204
 - Double-Loop-Lernen • 210
 - DTD • 227
 - Attributtypen • 55
 - Attributvorgabe • 56
 - Dokumenttyp-Definition • 52
 - Elementtyp • 52
 - Externer Entity • 56
 - Interner Entity • 56
 - Kantentyp • 54
 - Notation • 56
 - Parameter-Entity • 56
 - Uniform Resource Identifier • 56

- DTD-Modell • 52
- DV-Funktion • 23
- DV-Funktionstyp (Definition) • 23
- DV-Konzept
 - Beschreibungsebenen • 7
 - Beschreibungssichten • 3
 - Datensicht • 66
 - Funktionssicht • 23
 - Organisationssicht • 82
 - Steuerungs-/Prozesssicht • 143
- DV-Prozesse • 200
- DV-Unterstützung • 11
- DW-Struktur
 - Beziehung zu anderen Modellen • 221
 - Modellierung • 61
- DW-Transformation • 221
- Dynamic Modeling • 167
- E**
- E-Business
 - Definition • 222
 - Senario Diagram • 122
- eBusiness Scenario
 - Balanced Scorcard • 227
 - Diagram • 122
 - Einleitung • 222
 - Idee • 224
 - Modell und Objekte • 224
 - Report • 226
 - Sicherung • 225
 - Transaktionssicherung • 225
 - Übertragungsweg • 225
- eERM
 - Ansatz • 48
 - Begriffe und Darstellungsformen • 51
 - Konstruktionsoperatoren • 51
 - Methode • 48
 - Strukturelemente • 51
- eERM-Attributzuordnungsdiagramm • 47
- Einflussfaktor • 212
- Einheit • 74
- Einschränkungen (Constraints) bei Klassen, Attributen und Assoziationen • 168
- Einzelprozess • 122
- Element
 - Anwendungssystem • 117
 - Entweder-oder-Verknüpfung • 93
 - Organisatorisches • 117
 - Regelblatt • 93
- Elementarfunktion • 15
- Elementarprozess • 134
- Elementarrolle • 134
- E-Mail • 224
- Entitymenge • 38
- Entity-Relationship-Modell
 - ERM • 31
 - Erweiterungen • 38
 - Grundmodell • 32
 - Modellierung • 31
 - Strukturiert • 48
- Entitytyp
 - Aggregation • 38
 - Beispiele • 32
 - Schwach • 48
 - Stark • 48
 - Unterscheidung zu Attribut • 32
- Entscheidungskompetenz • 91
- Entwicklung der Systemoberfläche • 200
- Entwicklungsumgebung C++ • 138
- Entwurf von Schnittstellen
 - Zwischen SAP R/3 und Non-SAP-Anwendungen • 199
- Entwurfsprozess • 38
- EPK
 - Beziehungen zu anderen Modellen • 221
 - Leistungsaustauschmodellierung • 158
 - Materialfluss • 115
 - Prozessauswahlmatrix • 114
 - Spalten-/Zeilendarstellung • 117
 - Symbole • 125
- EPK (Zeilendarstellung) • 117
- EPK Materialfluss • 60
- Ereignis
 - Funktionsbaum • 15
 - Grafische Darstellung • 93
 - Start- und Ergebnisereignisse • 31
 - Verknüpfung • 93
- Ereignisdiagramm • 106
- Ereignisgesteuerte Prozesskette (EPK)
 - Ereignissteuerung • 93
 - Funktionsbaum • 15
 - Verzweigungen und Bearbeitungsschleifen • 93
- Ereignissteuerung • 107
- Erfolgsfaktor
 - Definition • 22
 - Erfolgskritischer Einflussfaktor • 212
 - Hierarchisierung • 22
 - Unternehmensdokumentation • 192

- Wirkungsgefüge • 212
 - Ergebniskennzahl • 209
 - Ergebnisqualität • 198
 - ERM-Attribut • 47
 - ERM-Attributzuordnungen • 47
 - ERM-Beziehungstyp • 47
 - ERM-Grundmodell
 - ERM-Grundmodell • 32
 - Erweiterung • 43
 - Untergrenze • 43
 - ERP-System • 231
 - Anbindung • 227
 - XML • 227
 - Existentielle Abhängigkeit • 44
 - Extensible Markup Language (XML) • 227
 - Extranet • 224
- F**
- Fachbegriffsmodell • 45
 - Fachkonzept
 - Beschreibungsebenen • 7
 - Datensicht • 31
 - Funktionssicht • 14
 - Organisationssicht • 74
 - Fähigkeit • 233
 - Faktentabelle • 61
 - Feedback-Prozess • 210
 - Feld • 71
 - Feld (Exemplar) • 71
 - Fertigkeit • 134
 - Festlegung von Anwendungssystemen und Modulen • 200
 - Finanzkennzahl • 203
 - Finanzziel • 208
 - Formular • 194
 - Formularfeld • 140
 - Frame • 140
 - Frühindikator • 208
 - Führungseingpass • 211
 - Führungssystem • 1, 204
 - Functional Block • 233
 - Functional Modeling • 167
 - Funktion
 - Ablauffolge • 93
 - Ausführung • 15
 - Automatisiert • 15
 - Batch-Lauf • 15
 - Bearbeitungszeit • 15
 - Einarbeitungszeit • 15
 - Grad der Integration • 91
 - Gruppierungskriterien • 15
 - Hierarchieebenen • 15
 - Integration • 74
 - Liegezeit • 15
 - Mengengerüst • 15
 - Prozessorientierte Zerlegung • 15
 - Regel • 93
 - Spezifizierung der Ausführung • 11
 - Teilzeiten • 15
 - Verdichtungsstufen • 15
 - Verrichtungsorientierte Gruppierung • 15
 - Vorgang • 3, 11
 - Zuordnung zu Daten • 93
- Funktionen
- Ablauffolge • 15
- Funktions-/Org. Ebenendiagramm • 91
- Funktionsbaum
- Objekt-orientiert • 15
 - Prozess-orientiert • 15
 - Semantisch • 15
 - Verrichtungs-orientiert • 15
- Funktionssicht
- Beschreibungssichten • 3
 - Betriebswirtschaftliche Problemstellung • 11
 - Fachkonzept • 14
 - Funktionen mit Organisation • 91
- Funktionszuordnungsdiagramm • 101
- G**
- Gateway
- Complex Gateway • 257
 - Symbole • 257
 - Typen • 257
- Generalisierung
- Ausprägungen • 38
 - Gattungsbegriff • 38
 - Gliederungsmodell • 124
 - Teilbegriff • 38
- Gerichteter Graph • 48
- Geschäftseinheiten • 206
- Geschäftsprozess
- Einleitung • 222
 - Ganzheitliche Betrachtung • 1
 - Geschäftsprozess • 15
 - Gestaltung • 199
 - Logisch-zeitlicher Ablauf • 93
 - Modelle • 202
 - Modellierung • 1

- Optimierung • 199
- Sichten • 11
- Wissen • 192
- Zeitlich-logische Ablauffolge • 11
- Ziele der Modellierung • 22
- Geschäftsprozessautomation • 245
- Gestaltung
 - Ermitteln • 195
 - Geschäftsprozesse • 199
 - Informationstechnischen
 - Infrastrukturbedarf
- Dokumentation • 195
 - Organisatorische Gestaltung der Geschäftsprozesse • 199
 - Organisatorische Gestaltung des SAP-Systems • 199
- Gleichartige Entity • 32
- Gliederungselement • 124
- Gliederungsmodell • 124
 - Generalisierung • 124
 - Spezialisierung • 124
- Goods Shipment • 122
- Group • 259
- Group Decision Support Systems • 194
- Groupware • 194
- Gruppierung • 38
- H**
- Handy • 122
- Hardwarekomponententyp
 - Definition • 82
- Hauptprozess
 - Prozessauswahlmatrix • 114
 - SAP-Applikationendiagramm • 21
- Hierarchisierung
 - Erfolgsfaktoren • 22
 - Hierarchiemodell • 15
- Hierarchisierungen von Objekten • 180
- HTML-Dokument • 122
- I**
- Identifizierungs- und Existenzabhängigkeit • 44
- IE-Datenmodell • 50
- IE-Notation • 50
- Implementierung
 - Datensicht • 71
 - Funktionssicht • 28
 - Implementierung • 7
- Implizite Beziehung • 117
- Indikator
 - Frühindikator • 208
 - Spätindikator • 208
 - Typ • 212
- Industrial Process
 - Bildhafte Darstellung • 125
 - Fertigungsprozess • 125
 - Symbole • 125
- Info Cube • 121
- Informationsfluss • 114
- Informationsflussdiagramm • 101
- Informationssystem
 - Betriebswirtschaftl. Problemstellung • 11
 - Lebenslauf • 7
 - Unterstützung • 184
 - Vorgehensmodell zur Entwicklung • 7
- Informationstechnologie • 206
- Informationsträger • 11
- Informationsträgerdiagramm • 65
- Informationsträgersymbol • 111
- Initial-, Final- und Übergangszustände • 175
- Innovationen • 206
- Input- und Outputdaten • 11
- Input/Output Diagram • 101
- Input-Daten • 202
- Integrationskonzept • 3
- Integrationspotential von Informationstechnologie • 130
- Intellectual Capital • 201
- Interviews • 198
- IS Service • 233
- ISO • 197
- IS-Sicht • 233
- Ist-Wert • 212
- IT Blöcke • 239
- IT City Planning-Report
 - Auswertung • 244
 - IT City Planning • 231
 - Sichten • 232
 - Tangierte Unternehmen • 231
 - Ziel (detailliert) • 231
 - Ziel (übergeordnet) • 231

IT Procedures • 239

IT Software • 239

IT-Sicht • 239

K

Kalkulationstabelle • 196

Kante

Typ ist übergeordnet • 64

Typen • 54

Kapitalinvestition • 209

Kapitalrentabilität • 206

Katalogisierung • 194

Kennzahlensystem • 211

Kernkompetenz • 205

Klassen • 113

Klassendiagramm • 113

Klassifizierung • 38

Klassifizierungsdiagramm • 111

Kommunikationsdiagramm • 111

Kommunikationsprogramm • 209

Kommunikationsstruktur • 121

Komplexitätsgrad • 32

Komplexitätsreduktion • 211

Konnektor • 93

Konsensfindung • 211

Konstruktionsoperator

Arten • 38

eERM • 51

Konstruktionsoperatoren

Ausprägungsebene • 32

Extern • 56

Intern • 56

Konstruktionsoperatoren • 38

Typ-Ebene • 32

Kontrollfluss • 11

Kontrollinstrument • 196

Kosten

Reduktion • 198

Strukturen • 196

Transparenz • 196

Kostenartendiagramm • 63

Hierarchie • 64

Prozesskostenrechnung • 63

Kostenrechnung • 196

Kostenreduzierung • 209

Kostenstellenanalyse • 196

Kostentreiber • 63

Künftige Bedeutung • 182

L

Lagereinrichtung (Definition) • 88

Lagereinrichtungs-kategorie (Definition) • 88

Lane • 248

Lean Management • 74

Lean Produktion • 74

Leistung • 157

Leistungsaustauschdiagramm • 158

Leistungsbaum • 159

Leistungsmaßstab • 64

Leistungsmengeninduziert • 63

Leistungsmengenneutral • 63

Leistungstreiber • 208

Leitbild • 206

Life-Cycle-Konzept • 3

Linienstelle • 77

Liste • 23

Listenentwurf • 23

Lotus Notes in ARIS • 194

M

Make-or-Buy-Entscheidungen • 196

Managementprozess (Strategie) • 205

Managementunterstützende Systeme • 193

Manuelle Bearbeitung • 11

Marktwachstum • 215

Maskendesign • 138

Maskendesign.rsm • 138

Maskendiagramm • 148

Maskenentwurf • 23

Maskennavigation • 140

Maßnahme • 212

Materialdiagramm • 60

Materialfluss

EPK • 60

In Prozessmodellen • 60

VKD • 60

- Materialfluss (EPK) • 115
- Materialflussdiagramm • 115
- Materialressource • 82
- Materialtyp (Definition) • 60
- Maximalwert • 212
- Medienbrüche • 198
- Medium • 11
- Message flow • 245
- Methoden • 7
- Minimalwert • 212
- Modell
 - Anwendungssystemtypdiagramm • 23, 28
 - Attributzuordnungsdiagramm • 66
 - Berechtigungshierarchie • 62
 - Business Controls Diagram • 120
 - C3-Methode • 137
 - Casual Model • 136
 - Collaborative Business Maps • 227
 - Cost Driver-Diagramm (CD-Diagramm) • 63
 - Data
 - Warehouse-Datentransformationsdiagramm • 121
 - DTD • 52
 - DW-Struktur • 61
 - eBusiness Scenario Diagram • 222
 - eBusiness Szenario Diagram • 122
 - eERM • 38
 - eERM-Attributzuordnungsdiagramm • 47
 - EPK (ereignisgesteuerte Prozesskette) • 93, 107
 - EPK (Materialfluss) • 115
 - EPK (Spalten-/Zeilendarstellung) • 117
 - EPK (Zeilendarstellung) • 117
 - ERM-Grundmodell • 32
 - Fachbegriffsmodell • 45
 - Funktions-/Org. Ebenendiagramm • 91
 - Funktionsbaum • 15
 - Funktionszuordnungsdiagramm • 101
 - Gliederungsmodell • 124
 - IE-Datenmodell • 50
 - Industrial Process • 125
 - Informationsflussdiagramm • 101
 - Informationsträgerdiagramm • 65
 - Input/Output Diagram • 101, 111
 - Klassendiagramm • 113
 - Klassifizierungsdiagramm • 111
 - Kommunikationsdiagramm • 111
 - Kostenartendiagramm • 63
 - Leistungsaustauschdiagramm • 158
 - Leistungsbaum • 159
 - Maskendiagramm • 148
 - Materialdiagramm • 60
 - Materialflussdiagramm • 115
 - Netzdiagramm • 86
 - Netztopologie • 82
 - Office Process • 125
 - OMT Class Description Model • 180
 - OMT DataValue Decomposition • 180
 - OMT Dynamic Model • 175
 - OMT Functional Model • 177
 - Organigramm • 77
 - Produktauswahlmatrix • 164
 - Produktbaum • 163
 - Produktzuordnungsdiagramm • 160
 - Programmablaufdiagramm • 144
 - Programmablaufplan (PA) • 144
 - Projektprozesskette (PPK) • 128
 - Prozessauswahlmatrix • 114
 - Prozessinstanziierungsmodell • 130
 - RAMS • 130
 - Rechtediagramm • 202
 - Regeldiagramm • 110
 - Relationendiagramm • 66
 - Role Allocation Diagram • 119
 - Rollendiagramm • 134
 - SAP - SERM • 48
 - SAP ALE-Filtermodell • 118
 - SAP ALE-Nachrichtenflussmodell • 118
 - SAP ALE-Nachrichtentypmodell • 119
 - SAP-Applikationendiagramm • 21
 - Schichtkalender • 82
 - SeDaM-Modell • 50
 - Semantisch • 7
 - Service-Zuordnungsdiagramm • 238
 - Swimlane • 1
 - Systemattributdomäne • 69
 - Systemattribute • 69
 - Tabellendiagramm • 71
 - Technische Ressourcen • 88
 - VKD (Vorgangskettendiagramm) • 107
 - Wertschöpfungskettendiagramm • 107
 - Wettbewerbskräftemodell • 165
 - Wissenslandkarte • 186
 - Wissensstrukturdiagramm • 185
 - Y-Diagramm • 20
 - Zieldiagramm • 22
 - Zugriffdiagramm • 143
 - Zugriffdiagramm (phys.) • 151
- Modellierung
 - Betriebliche Realität • 1
 - Materialfluss • 114
 - Materialflusses im Materialdiagramm • 60
 - Methoden • 1
 - Semantisch • 7

- Modelltyp
 - Maskendesign • 138
 - Maskennavigation • 140
- Modul (Definition) • 28
- Modultyp (Definition) • 23
- Money Transaction
 - eBusiness Szenario Diagram • 122
 - Objekte • 224
- N**
- Netz • 86
- Netzdiagramm • 86
- Netzexemplar • 82
- Netzkante • 86
- Netzkantentyp • 82
- Netzknoten • 86
- Netzknotentyp • 82
- Netztopologie • 82
- Netztyp (Definition) • 82
- Netzwerkmanagement • 195
- Notation
 - Identifizierungs- und Existenzabhängigkeit • 44
 - Kardinalitäten • 43
- O**
- Object Management Group • 166
- Object Modeling • 167
- Object Modeling Technique • 193
- Objektmodell
 - Mehrstufig • 130
 - Prozessinstanzierungsmodell • 130
 - Schichtkalender • 82
- Objektorientiert
 - Erweitertes Konzept • 113
 - Modellierung • 113
 - Unternehmensdokumentation • 192
- Objektorientierte Datenbanksysteme • 193
- Objekttypen
 - Dokumentiertes Wissen • 184
 - Wissenskategorie • 182
- Oder-Verknüpfung • 93
- Office Process
 - Bildhafte Darstellung • 125
 - Büroprozesse • 125
 - Symbole • 125
- OMT • 167
- OMT Class Description Model • 180
- OMT DataValue Decomposition • 180
- OMT Dynamic Model • 175
- OMT Functional Model
 - Darstellung von Actoren (Handlungsobjekten) • 177
 - Darstellung von Datenflüssen • 177
 - Darstellung von Processes (Prozessen) • 177
 - Splitten von Datenflüssen • 177
- OMT Object Model
 - Aggregation zwischen Klassen • 168
 - Assoziationen zwischen Klassen • 168
 - Darstellung einer qualifizierten Assoziation • 168
 - Darstellung von Instanzen • 168
 - Darstellung von Klassen • 168
 - Darstellung von Ordnungen bei Assoziationen • 168
 - Generalisierung und Vererbung • 168
 - Mehrstufige Assoziationen zwischen Klassen • 168
 - Modellieren einer Assoziation als Klasse • 168
 - Zuordnung von Attributen zu Klassen • 168
 - Zuordnung von Instanzen zu Klassen • 168
 - Zuordnung von Operationen zu Klassen • 168
- OMT-Methodologie • 167
- Online-Shop
 - Beispiel • 227
 - Modell • 227
- Organigramm • 77
- Organisation • 74
- Organisationseinheit
 - Beschreibungssichten • 3
 - Betriebswirtschaftliche Problemstellung • 11
 - Organigramm • 77
 - Zuordnung zu Funktionen • 93
- Organisationseinheitentyp • 77
- Organisationsform • 74
- Organisationssicht
 - Beschreibungssichten • 3
 - DV-Konzept • 82
 - Fachkonzept • 74
 - Funktionen mit Organisation • 91

- Organisationsstruktur
 - Allgemeine Unternehmensdokumentation
 - 192
 - Balanced Scorecard-Systems • 214
- Organisatorisch • 74
- Organisatorisch
 - Einheit • 74
 - Organisatorische Einheiten • 74
 - Strukturierung • 74
 - Zerlegung • 74
- Organisatorisch
 - Element • 117
- Organisatorisch
 - Brüche • 198
- Organisatorische Komplexitätsbewältigung • 74
- Organizational Unit Type • 224
- OR-Verknüpfung • 93
- Output-Daten • 202
- P**
- Packmitteltyp • 115
- Packmitteltyp (Definition) • 60
- Parameter-Entity • 56
- Pause • 82
- Performance-Measurement
 - Ansatz • 204
 - Balanced Scorecard-System • 214
 - System • 204
- Periode • 82
- Personalentwicklung • 206
- Personalressource • 82
- Personentyp • 77
- Perspektiv • 212
- Perspektive
 - Externe Performance-Perspektive • 204
 - Finanzielle • 207
 - Finanzwirtschaftliche • 204
 - Humanorientiert • 212
 - Interne Performance-Perspektive • 204
 - Kundenperspektive • 207
 - Lern- und Entwicklungsperspektive • 204
 - Prozessperspektive • 207
 - Standard • 207
- Pfad (spezifizieren) • 128
- Phase Analyse, Spezifikation (Project Preparation) • 199
- Phase Design (Business Blueprint) • 199
- Philosophie • 206
- Pipeline-Diagramm • 227
- Planfortschrittszahl • 210
- Plan-Ist-Vergleich • 212
- Planung (strategisch) • 198
- Planungsprozess • 211
- Plan-Wert • 212
- Pool • 248
- PPK • 128
- Preisfindung • 196
- Private Business Process • 245
- Private Process • 247
- Problemstellung (betriebswirtschaftlich) • 189
- Process Warehousing • 192
- Produktauswahlmatrix • 164
- Produktbaum • 163
- Produktionsintensität • 209
- Produktzuordnungsdiagramm • 160
- Programmablaufdiagramm • 144
- Programmablaufplan • 144
- Programm-Code • 200
- Programmdatei • 28
- Programmiersprache • 23
- Programmteil (Definition) • 28
- Programmteiltyp
 - Definition • 28
 - Zuordnung zu Anwendungssystemtypen • 28
 - Zuordnung zu Programmteilen • 28
- Project Preparation • 199
- Projekt-Dokumentation
 - Reorganisationsmaßnahmen • 198
 - Software-Entwicklung und -einführung • 200
- Projektprozesskette • 128
- Projektstrukturelement • 128
- Prozess
 - Beschreibungssichten • 3
 - Definition der Prozessauswahlmatrix • 114
 - Feedback • 210
 - Funktionsbaum • 15

- Kosten • 206
- Leistungsmengeninduziert • 63
- Leistungsmengenneutral • 63
- Priorisieren • 130
- Zeiten • 206
- Prozess- und Organisationsstrukturen • 196
- Prozess- und/oder Funktionsdefizite • 199
- Prozessart • 192
- Prozessauswahlmatrix • 114
- Prozessbeschreibungen • 194
- Prozess-Customizing • 202
- Prozessdokumentation • 192
- Prozessinstanziierungsintervall • 130
- Prozessinstanziierungsmodell
 - Intervall • 130
 - Prozessinstanziierungsplan • 130
 - Prozessinstanziierungszyklus • 130
- Prozessinstanziierungsplan • 130
- Prozessinstanziierungszyklus • 130
- Prozesskalkulation • 196
- Prozesskette
 - ARIS-Architektur • 3
 - Datenfluss • 107
 - Ereignisgesteuerte • 93
 - Organisatorische Gestaltung der Unternehmen • 74
- Prozesskostenrechnung
 - Begriff • 196
 - Kostenartendiagramm • 64
 - Leistungsmaßstab • 64
- Prozesskostensatz • 196
- Prozessmanagement • 197
- Prozessmodelle • 60
- Prozessorientiert • 192
- Prozessreferenzmodell • 134
- Prozessrisiken • 120
- Prozesssicht • 74
 - ARIS • 93
 - Fachkonzept • 91
- Prozessszenarien • 21
- Prozessvarianten • 199
- Prozesswissens • 192

Q

- QM-Dokumentationen • 197
- QM-Handbuch • 197
- Qualität
 - Förderung • 197

- Kriterium • 124
- Lenkung • 197
- Management • 197
- Planung • 197
- Politik • 197
- Sicherung • 197
- Unterlagen • 197
- Verbesserung • 197

R

- RAD • 119
- RAMS
 - Requirements Analysis for Management Systems • 130
 - Vorgehensmodell • 130
- Rechtediagramm • 202
- Redundanz
 - Datenredundanzen • 11
 - Mehrfacherfassungen • 11
 - Zeitverschleppungen • 11
- Referentielle Integrität • 44
- Regeldiagramm • 110
- Relation • 66
- Relationendiagramm • 66
- Reorganisation
 - Durchführung • 198
 - Maßnahmen • 198
- Report
 - eBusiness Scenario • 226
 - Maskendesign.rsm • 138
- Ressource
 - Erfassen • 206
 - Informationstechnik • 3
 - Materialressource • 82
 - Objekt • 128
 - Personalressource • 82
 - Schichtkalender • 82
- Ressourcensicht • 3
- Retrieval-Funktionalität • 194
- Risiko
 - Prozessrisiken • 120
 - Risikobekämpfung (Control) • 120
 - Risikobewältigung (Control) • 120
 - Risikolösung (Solution) • 120
 - Risk • 120

- ROI • 206
- Role Allocation Diagram • 119
- Rolle • 119
- Rollendiagramm • 134
- Rollenname • 32
- Rollenzuordnungsdiagramm • 119
- S**
- Sachleistung • 157
- Sachverhaltshierarchie • 124
- SAP - SERM • 48
- SAP ALE-Filtermodell • 118
- SAP ALE-Nachrichtenflussmodell • 118
- SAP ALE-Nachrichtentypmodell • 119
- SAP-Anwendungen • 192
- SAP-Applikationsdiagramm • 21
- SAP-R/3-Referenzmodell • 21
- SAP-Referenzmodell • 119
- SAP-System • 199
- Schicht • 82
- Schichtkalender • 82
 - Pause • 82
 - Periode • 82
 - Schicht • 82
 - Schichtkalender • 82
 - Schichtzyklus • 82
 - Zuordnung • 82
 - Zyklisch wiederholen • 82
- Schichtzyklus • 82
- Schlüssel
 - Einfach • 38
 - Komplex • 38
- Schlüsselattribut • 32
- Schnittstellenentwicklung • 199
- Schulung • 194
- Schwachstelle (Problemlösung) • 11
- Schwachstelle (SAP) • 199
- Schwachstellenanalyse • 192
- SCM-Lösung • 231
- Security Protocol • 224
- SeDaM - Modell • 50
- Semantic Generator • 136
- Sequence Flow • 245
- SER
 - Ansatz • 48
 - Modell • 48
- Servicetyp • 233
- Service-Zuordnungsdiagramm
 - Cluster • 238
 - IT-Elemente • 241
- Shareholder Value • 206
- Sicht
 - Bildung • 7
 - Geschäftsprozess • 11
 - Humanorientiert • 212
 - Zerlegung • 3
- Simulation • 192
- Skill • 134
- Soll-Ablauf • 11
- Soll-Konzept • 11
- Sortier-Index • 71
- Spätindikator • 208
- Spezialisierung
 - Ausprägungen • 38
 - Disjunktheit • 38
 - Vollständigkeit • 38
- Spezifikation • 199
- Stabsstelle • 77
- Standardperspektive • 207
- Start- und Endereignisse • 93
- Stelle • 77
- Sternen-Schema • 61
- Steuerungs-/Prozesssicht
(Zugriffsdiagramm) • 143
- Steuerungsinformation • 203
- Steuerungssicht
 - ARIS • 93
 - ARIS-Haus • 74
 - Beschreibungssichten • 3
 - Betriebswirtschaftliche Problemstellung • 11
 - Fachkonzept • 91
- Strategie
 - Aktionsformulierung • 211
 - Analyse • 205
 - Begriff • 212
 - Geschäftseinheiten • 206
 - Geschäftsfeldspezifisch • 205
 - Informationstechnologie • 206
 - Inkonsistenz • 210
 - Innovationen • 206
 - Lernen • 210
 - Managementprozess • 205
 - Marktorientiert • 206
 - Personalentwicklung • 206
 - Planungsprozess • 211

- Teilstrategie • 205
- Übergeordnete • 209
- Umsetzung • 209
- Unternehmensstrategie • 206
- Verfolgung • 210
- Wettbewerb • 204
- Wunschformulierung • 211
- Ziel • 212
- Strategisches Ziel • 212
- Strukturelemente des eERM • 51
- Strukturelle Änderungsgeschwindigkeit • 182
- Stufen- oder Phasenkonzept • 7
- Substitutionsbeziehung • 159
- Subsystem • 239
- Symbol
 - BSC-Kennzahlenszuordnungsdiagramm • 217
 - BSC-Ursache-Wirkungsdiagramm • 215
- Synergieeffekt • 209
- System
 - CRM • 231
 - ERP • 231
 - Managementunterstützend • 193
- Systemattributdomäne • 69
- Systemattribute • 69
- Systemoberfläche • 200
- Systemschnittstellenmodelle • 69
- Szenario
 - Definition in der Prozessauswahlmatrix • 114
 - Varianten • 199
- T**
- Tabelle • 71
- Tabelle (Exemplar) • 71
- Tabellen und Felder • 71
- Tabellendiagramm • 71
- Tätigkeitsfluss • 202
- Techn. Hilfsmittel (Definition) • 88
- Techn. Hilfsmittelklasse (Definition) • 88
- Techn. Hilfsmitteltyp • 115
 - Definition • 88
- Technische Ressourcen • 88
- Teilfunktion • 15
- Teilstrategie • 205
- Teilziel • 208
- Toleranzbereich • 212
- Top-Down-Ansatz • 209
- Top-Management • 206
- Transaktion • 23, 119
- Transaktionsschritte • 23
- Transferstruktur • 121
- Transformationsregel • 121
- Transportsystem (Definition) • 88
- Transportsystemklasse (Definition) • 88
- Typ-Ebene • 32
- U**
- Umfeld
 - Organisatorische Gestaltung • 74
 - Zustandsbeschreibung • 3
- Uminterpretierter Beziehungstyp • 38
- UML-Report • 166
 - UML Glossary • 166
 - UML Notation Guide • 166
 - UML Semantics • 166
 - UML Summary • 166
- UML-Version • 166
- Umsatzerlös • 206
- Umschlaghäufigkeit • 182
- Und-Verknüpfung • 93
- Unified Modeling Language • 166
- Uniform Resource Identifier • 56
- Unternehmen
 - Erfolgsziel • 206
 - In Datenbanken • 193
 - Leitbild • 206
 - Perspektiven • 204
 - Philosophie • 206
 - Sichtweisen • 204
 - Strategie • 204, 206
 - Vision • 204

Unternehmensanalyse • 130
 Unternehmensdokumentation • 192
 Unternehmensentscheidungen • 192
 Unternehmensführung • 211
 Unternehmensfunktionen • 192
 Unternehmensmodell • 199
 Unternehmenswertschöpfung • 192
 Unternehmensziele
 Balanced Scorecard • 203
 Hierarchisierung • 192
 Organigramm • 77
 Prozessinstanzierungsmodell • 130
 Zieldiagramm • 22
 Unternehmerische Ziele • 23
 Ursache-Wirkungsbeziehungen
 Balanced-Scorecard • 207
 Ursache-Wirkungskette • 208
 Ursache-Wirkungs-Kette • 208

V

Variante • 160
 Prozess • 199
 Szenario • 199
 VDA • 197
 Vergangenheitsorientierung • 211
 Verknüpfung
 Allgemeine Regel • 93
 Entweder-oder • 93
 Ereignis • 93
 Funktion • 93
 Komplexe Beziehungen • 93
 Logische Entweder-oder-Verknüpfung • 93
 Logische Oder-Verknüpfung • 93
 Logische Und-Verknüpfung • 93
 ODER • 93
 Operator • 11
 Regeln • 93
 UND • 93
 XOR • 93
 Verrichtungsorientiert • 192
 Video • 192
 View
 Definition • 66
 View (phys.) • 71
 Virtueller Platz • 222
 Vision • 212
 VKD
 Betriebswirtschaftliche Problemstellung • 11

Materialfluss • 60, 115
 Spalten • 107
 Vorgang
 Beschreibungssichten • 3
 Funktionsbaum • 15
 Vorgangskette
 EPK/VKD • 107
 Funktionsbaum • 15
 Vorgangskettenanalyse • 11
 Vorgangskettendiagramm
 Betriebswirtschaftliche Problemstellung • 11
 Ereignis- und Funktionsspalten • 93
 Fachkonzept • 107
 Verzweigungen und
 Bearbeitungsschleifen • 93
 Vorgehenskonzept • 1
 Vorgehensweise
 Bottom-up • 38
 Top-down • 38

W

Warngrenze • 212
 Webformular • 138
 Webservices • 245
 Weisungsbefugnisse • 77
 Wertebereich • 66
 Wertschöpfung • 192
 Wertschöpfungskette • 122
 Wertschöpfungskettendiagramm • 107
 Wettbewerbskräftemodell • 165
 Wettbewerbssituation • 165
 Wettbewerbsstrategie • 204
 Wirkungsgefüge • 206, 212
 Wirkungsgrad • 212
 Wirkungsrichtung • 212
 Wirkzusammenhang • 210
 Wirtschaftsobjekt • 122
 Wissen
 Abdeckungsgrad • 182
 Dokumentiert • 181
 Wissenskategorisierung • 201
 Wissenslandkarte • 186
 Wissensmanagement
 Anwendung • 201
 Begriff • 181
 Wissenskategorie • 182

- Wissensnutzung • 182
 - Wissensstrukturdiagramm • 185
 - Wissensverarbeitung in Geschäftsprozessen
 - 201
 - Wissensvorsprung • 182
 - Workflow • 192
 - Workflow-Management-Systeme • 202
 - Workflow-Modell • 202
 - Workgroup Computing • 194
 - Workshop • 212
- X**
- XML • 227
 - XML-Dokument • 122
 - XOR-Regel • 93, 128
- Y**
- Y-CIM-Modell • 20
 - Y-Diagramm • 20
 - Yellow Page • 201
- Z**
- Zeitintervall • 82
 - Zeitlich-logische Ablauffolge • 11
 - Zeitreduktion • 198
 - Zerlegung (Organisatorisch) • 74
 - Zertifizierung
 - Gliederungsmodell • 124
 - Unterlagen • 197
 - Vorgehen • 197
 - Ziel
 - Abteilungsziel • 211
 - Definition • 22
 - Finanzwirtschaftlich • 212
 - Individual • 211
 - Operativ-monetär • 211
 - Strategisch • 203, 211
 - Zieldiagramm • 22
 - Zielerreichung • 209
 - Zielerreichungsgrad • 212
 - Zielhierarchien • 22
 - Zielvorgabe • 209
 - Zielwert • 209
 - Ziel-Wert • 212
 - Ziel-Zeitraum • 212
 - Zone • 233
 - Zugriffsberechtigungen • 195
 - Zugriffsdiagramm • 143
 - Zugriffsdiagramm (phys.) • 151
 - Zugriffsdiagramm (physikalisch) • 151
 - Zugriffsrecht
 - Organisationseinheiten • 71
 - PC-Hardware- und Netzwerkmanagement
 - 195
 - Zustand • 3
 - Zustandsänderung • 93
 - Zyklisch wiederholen • 82