HUMBOLDT-UNIVERSITÄT ZU BERLIN

„Data Warehouse Project Management“

eine Seminararbeit
im Rahmen der Vorlesung

Data Warehousing

Autoren

Daniel Göhring
Stephan Weißleder

Dozent
Prof. Dr. Günther

Datum
12.02.2003

Inhaltsverzeichnis

1 Einleitung: Motivation (Warum braucht man ein Data Warehouse?)

1998 wurde die Größe des Data-Warehouse-Marktes auf 8 Milliarden US-Dollar geschätzt. Mehr als 900 Firmen boten Software, Hardware und andere Dienstleistungen in diesem Zusammenhang an. In der Zwischenzeit hat sich der Markt rasant weiterentwickelt.

Bei diesem Volumen stellt sich die Frage, was ein Data Warehouse so besonders macht, dass es unverzichtbar scheint. Eine einfache und zugleich zutreffende Antwort ist, dass man durch den Einsatz eines Data Warehouse Kosten verringern und/oder Gewinn erhöhen kann.
Doch auch hier ist der Erfolg von der Qualität des Data Warehouse abhängig – ein schlecht konzipiertes Data Warehouse kann sich am Ende durchaus zum reinen Kostenfaktor entwickeln. Die monetären Vorteile eines Data Warehouse sind allerdings eher mittel- oder langfristiger Natur, deshalb wird hier zuerst das Wesen eines Data Warehouse erörtert.

Das Ziel eines Data Warehouse ist, dass man jederzeit verfügbare, fachübergreifende, wohl strukturierte und qualitativ hochwertige Informationen für interne und externe Zugreifende zur Verfügung stellt (mit verschiedenen Zugriffsrechten). Diese Informationen können sich auf das Unternehmen generell bis hin zu detaillierten Fragen zu Prozessstandards oder einzelnen Prozessen beziehen.
Diese Informationen dienen vor allem aber als Entscheidungsgrundlage für Manager der entsprechenden Firma. Aus ihnen können zusammen mit externen Datenquellen wichtige Erkenntnisse über entscheidungsrelevante Fragen gewonnen werden, da das Data Warehouse Analysefunktionen zur Verfügung stellt, die für diesen Prozess unerlässlich sind.

Da der Aufbau eines Data Warehouse einen mittel- bis langfristigen Prozess darstellt, der zudem mit hohen finanziellen Risiken behaftet ist, muss er gut geplant, durchgeführt und überwacht werden. Zu diesem Zweck gibt es das Data Warehouse Projekt Management. Es befasst sich mit allen Aufgaben, die zur erfolgreichen Durchführung eines Projektes notwendig sind.
Der Projektmanager ist hauptverantwortlich und hat verschiedenste Aufgaben zu bewältigen, die unter anderem die Einhaltung der Zeit- und Kostenpläne und die Sicherung der Qualität des entstehenden Data Warehouses, sowie die Zufriedenheit und die möglichst optimale Auslastung der beteiligten Mitarbeiter beinhalten.

Den Lebenslauf eines Data Warehouses kann man grob in zwei Teile unterteilen: zum ersten den Aufbau und zweitens die Nutzung. Dabei sind diese beiden Teile gar nicht so sehr voneinander getrennt, wie es auf den ersten Blick scheinen mag. Ein Data Warehouse ist nämlich kein Produkt, das einmal hergestellt wird und danach nur noch benutzt wird – die Nutzung und Wartung des Data Warehouse beinhaltet auch die ständige Anpassung an die gegebenen Bedingungen, zu denen unter anderem Personalveränderungen und Veränderung des Marktes gehören.
Beim Aufbau hat man also schon Nutzensaspekte zu berücksichtigen und die Entwicklung möglichst so zu lenken, dass der spätere Betrieb erleichtert wird. Dies ist auch unter dem Gesichtspunkt wichtig, dass die Entwickler zumeist selbst die Kosten für die Wartung zu tragen haben. Das Beispiel eines intern genutzten Data Warehouse verdeutlicht dieses Problem. Wegen der Wichtigkeit der Planung des Aufbaus des Data Warehouse wird die Abteilung Projekt Management für den Gesamterfolg verantwortlich gemacht. Mit den auf sie zukommenden Aufgaben beschäftigt sich diese Arbeit.

2 Aufbau des Data Warehouse

Für den Aufbau eines Data Warehouse ist es vorteilhaft, sich schon im Vorfeld für die zu benutzende Referenzarchitektur zu entscheiden.
Sie ermöglicht Vergleiche zwischen Werkzeugen für ETL (Extraktion, Transformation und Laden der Daten), konkreten Data-Warehouse-Systemen bzw. zwischen einem speziellen Data-Warehouse-System und einer Referenzarchitektur. Auf der Basis einer Referenzarchitektur kann eine bestimmte Implementierung für ein Data-Warehouse-System geplant werden. Durch die Nutzung der Referenzarchitektur wird der vorhandene Bestand in seine Komponenten unterteilt. Diese Zerlegung in Komponenten dient der besseren Übersicht – sie ist somit ein Mittel zur Beschreibung, wodurch sich auch bisher unbekannte Beziehungen durch die Analyse finden lassen. Sie trägt zu einem gemeinsamen Verständnis bei Autoren und Lesern bei, da eine unternehmensweit einheitliche Sprache und Begriffsdefinition definiert werden muss.

2.1 Strategie

Bevor man also anfängt, ein Data Warehouse zu entwickeln, muss man sich über die Vorgehensweise – die Strategie – im Klaren sein. Auf sie entfällt ein wesentlicher Anteil des späteren Erfolgs oder Misserfolgs.

Der Prozess der Strategiefindung ist der Prozess umfassender integrierender Betrachtungen, welcher die Rahmenbedingungen für das Data Warehouse schaffen soll. Dazu müssen alle relevanten Faktoren, also die Organisation, die Technik, der Mensch und natürlich der Zweck des Data Warehouse miteinbezogen werden.
Die so entstehende Strategie muss projektübergreifend und langfristig flexibel nutzbar sein. Auf das durch das erste Projekt entstandene Data Warehouse sollen schließlich auch spätere Projekte zugreifen können.

Für die Strategiefindung des Data Warehouse muss zunächst die Unternehmensstrategie und damit insbesondere die IT-Strategie analysiert werden. Ein Data Warehouse ist zwar auf die Realisierung wirtschaftlicher Interessen ausgerichtet, hat sich dabei aber wie kein anderer Bereich an den technischen Möglichkeiten und Gegebenheiten zu orientieren.
Nach diesem ersten Schritt sollte die Machbarkeit eines Data Warehouse Systems analysiert werden. Dabei werden besonders die langfristigen Komponenten untersucht, die den langfristigen Charakter des Data Warehouse darstellen. Im folgenden werden typische Vorgehensweisen vorgestellt, um kostenintensive Fehler zu verhindern.

Mit der Data-Warehouse-Strategie schafft man also eine Grundlage, die sich in der resultierenden Vorgehensweise in der Projektorganisation und den Phasen eines Data-Warehouse-Projekts widerspiegelt.

2.1.1 Strategiefindung

So, wie die Data-Warehouse-Strategie von der IT-Strategie anhängig ist, so ist auch die Data-Warehouse-Strategiefindung abhängig von der IT-Strategiefindung. Doch was bedeuten diese beiden Begriffe überhaupt?

Die Unternehmensstrategie betrachtet die langfristige Entwicklung des Unternehmens. Die IT-Strategie ist Teil der Unternehmensstrategie und hat sich an dieser zu orientieren. Nach [8] besteht die IT-Strategie aus technologischen und organisatorischen Leitlinien, sowie dem

strategischen Informationsplan. Die angesprochenen Leitlinien beinhalten lediglich Freiräume und Richtlinien innerhalb der Informationsverarbeitung, die permanent angepasst werden müssen. Der strategische Informationsplan hingegen ist das konkrete Programm mit allen wichtigen Daten (Prioritäten, Zeiten, Kapazitäten, Ressourcen, etc.) für die Durchführung des Vorhabens. Dort werden alle wichtigen Entscheidungen für den Einsatz des Data Warehouse getroffen.
In Bereichen, die stark vom Fortschritt in der IT und von dem Stand der Technik abhängig sind, nimmt die IT-Strategie einen dementsprechend hohen Stellenwert ein.
Umgekehrt ist es gerade für Unternehmen in der IT-Branche wichtig, sich an dem Stand der Technologie und den technologischen Potentialen zu orientieren, beziehungsweise die Unternehmensstrategie entsprechend abzuändern. Die Konsequenz und Geschwindigkeit auf diesem Sektor entscheiden über den Erfolg. IT-Strategie und Unternehmensstrategie beeinflussen sich also gegenseitig.

Alles in allem ist der Prozess der Entwicklung der Architektur und der Strategie ein permanenter Prozess, in dem alle Erfahrungen von Mitarbeitern mit einfließen sollten. Dieses strategische Lernen oder auch „Knowledge Management“ ist besonders wichtig, da gerade im Data Warehouse die fach- und abteilungsübergreifende Kommunikation von entscheidender Bedeutung ist.

2.2 Data Warehousing als Teil der IT-Strategie

Das konkrete Data Warehouse ist ein Teil der IT-Strategie. Seine Einsatzfelder werden meist in Strategiestudien in den jeweils zuständigen Geschäftsbereichen festgelegt.
Besonders wichtig sind Studien in folgenden Fragen:

a) *Welche Rolle spielt das Data Warehouse im Rahmen der IT-Gesamtstrategie?*

b) *Wie und in welchem Umfang (zeitlich, räumlich, organisatorisch,...) wird das Data Warehouse eingesetzt? (verteilt/zentral, Intranet/Extranet,...)*

c) *Sollte man das Data Warehouse selbst entwickeln oder andere Konzepte übernehmen oder etwas von beidem?*

d) *Inwieweit ist es möglich, bestehende Systeme einzubinden?*

Die Ergebnisse dieser Strategiestudien fließen über das Management in die IT-Strategie in Form eines Bebauungsplanes für die Data-Warehouse-Landschaft mit ein. Der Bebauungsplan beinhaltet alle relevanten Unternehmensziele unter Berücksichtigung von Geschäftsprozessen. Organisationsstrukturen und Technologie, sowie der Anforderungen der Anwender.
Er bildet den Handlungs- und Orientierungsrahmen.

Bevor man sich jedoch aufmacht und diesen Plan umsetzt, sollte man eine Machbarkeitsstudie in Auftrag geben, welche die Risiken und sämtliche zu beachtenden Punkte miteinkalkuliert.

2.3 Machbarkeitsstudie

Wie bereits angedeutet sind die Probleme der Erstellung und der Organisation des Data-Warehouse-Systems nicht trivial. Dieses Projekt kann aus vielen Gründen fehlschlagen. Die technische und wirtschaftliche Machbarkeit ist also ein sehr wichtiger Punkt, der am Anfang der Entwicklung eines Data-Warehouse-Systems stehen sollte. Die Machbarkeitsstudie umfasst folgende Punkte:

a) Bedeutung des Data Warehouse im betrieblichen Umfeld
b) geplante Einsatzgebiete im Unternehmen nach Strategiestudie
c) Softwareauswahl
d) Hardwareauswahl
e) Risikobewertung und Abwägen der Erfolgsfaktoren
f) Wirtschaftlichkeitsbetrachtung
g) mögliche Alternativen
h) Vorgehensempfehlung

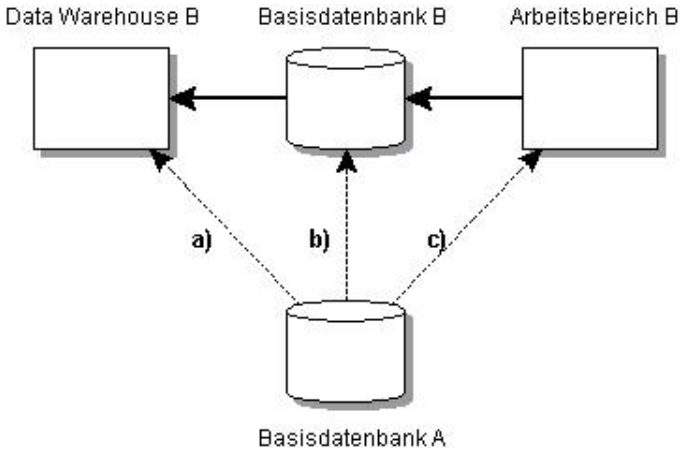
Eine solche Technologiestudie beschafft wichtige Informationen im Hinblick auf die Einführung eines Data-Warehouse-Systems. Darauf aufbauend kann entschieden werden, ob ein Data-Warehouse-System eingeführt wird. Es kann zudem nützliche Hinweise zur praktischen Umsetzung des Vorhabens geben. Den konkreten Aufbau hat – wie immer – das Projekt Management zu bestimmen und zu verantworten.

2.4 Festlegen der Data-Warehouse-Architektur

An dieser Stelle stellt sich die Frage, für welche Art von Unternehmen man ein Data Warehouse entwerfen möchte. Die meisten Unternehmen, die ein Data Warehouse einzuführen gedenken, stellen große und dezentral organisierte Strukturen dar. Folglich sollte es nicht nur ein zentral organisiertes Data-Warehouse-System geben. Im Falle eines einzigen Data-Warehouse-Systems müssten viele Abteilungen ihr möglicherweise bereits entwickeltes System und damit auch ein Stück erkämpfte Unabhängigkeit aufgeben.
Man sollte demnach ein ebenso dezentral organisiertes Data-Warehouse-System einführen. Weitere Gründe für die dezentrale Lösung sind die damit einhergehende erhöhte Sicherheit, mögliche Lastverteilung und technische Restriktionen, deren Umgehung bei einer zentralen Lösung zu teuer würden.
Bei verteilten Daten ist allerdings zu beachten, dass es einen „single point of truth" gibt – eine Stelle, an der alle Daten korrekt vorhanden sind. Als Lösung hierfür kann man eine zentrale Datenbank nutzen, die von allen Systemen beschickt und gefragt wird. Aufgrund dieses „hot spots" sollte man hier besonders auf die Qualität und die Performanz der eingesetzten Technik achten.
Dass dieser Punkt des „single point of truth" so wichtig ist, zeigt die einfache Überlegung, dass gerade bei voneinander völlig unabhängigen Daten die Hauptargumente eines Data Warehouse, z.b. Konsistenz und Mehrfachnutzung außer Kraft gesetzt wären.

Für die Beschickung mit Daten kann man grundlegend drei Varianten definieren:

a) die direkte Beschickung der Endsysteme
b) die Beschickung über die Basisdatenbank
c) die Beschickung über den Arbeitsbereich



(Data Warehouse Systeme, von Bauer und Günzel)

Die Frage, wie man die Beschickung zu organisieren hat und welche Methode hierfür die beste ist, lässt sich nicht pauschal beantworten. Vielmehr muss man diese Frage individuell und vor Ort entscheiden. Jede Variante bringt nämlich Vor- und Nachteile mit sich. So kann man pauschal nur sagen, dass eine schnellere Einbindung meist eine schlechtere Qualität der Daten und umgekehrt eine hohe Qualität der Daten eine langwierige Einbindung fremder Daten in das eigene System nach sich zieht.

2.5 Gestaltung des Data-Warehouse-Systems

Bei der Einführung eines Data Warehouse gibt es zahlreiche Freiräume und Einschränkungen, die zu beachten sind. Faktoren, die diese beeinflussen, sind:

a) das Ideal der Referenzarchitektur,
b) die Vereinfachung der Architektur,
c) Restriktionen der eingesetzten Standardsoftware,
d) die existierende Infrastruktur und
e) der zeitliche und finanzielle Rahmen

des Data Warehouse.

Die gewählte Referenzarchitektur lässt einige Punkte, über die noch Entscheidungen zu fällen sind, aus. Diese sind fachübergreifender und langfristiger Natur und haben großen Einfluss auf das Data Warehouse. So gibt es in den folgenden Bereichen Bedarf nach organisatorischen Festlegungen:

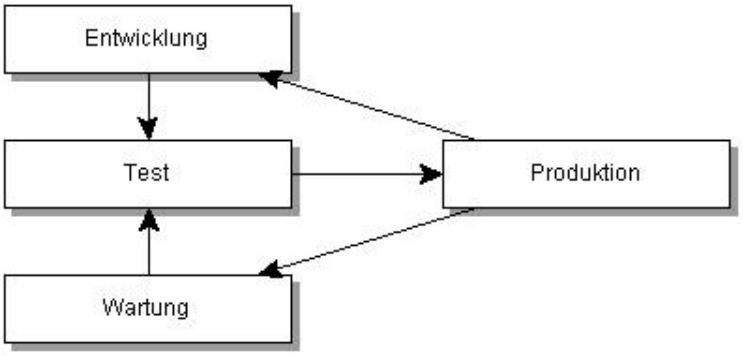
a) Repositorium (Fähigkeit zur Versionierung, Erweiterbarkeit, Webanbindung; Festlegung von Metadaten; Zuständigkeiten für Beschaffung, Pflege und Freigabe der Metadaten)
b) Arbeitsbereich (hardware, software, safety (backup) & security (Zugriff), Zuständigkeiten für Betriebsbereitschaft)
c) Basisdatenbank (Inhalt und Umfang, hardware, software, Verfügbarkeit, safety & security, Art und Umfang der Archivierung)
d) Organisation (Entscheidungsinstanzen für die Architektur, Besitzverhältnisse, Zuständigkeiten für den Betrieb)

Wie muss man das entstehende System im Hinblick auf Entwicklungen, Tests, Produktion und Wartung bewerten ?

Hier sei wieder zu erwähnen, dass ein Data Warehouse selten statisch ist und man sich schon im Vorfeld auf spätere Änderungen einstellen sollte – und zwar Änderungen auf allen möglichen Ebenen. Das Data Warehouse soll schließlich auch unter sich ständig ändernden Bedingungen funktionieren, was den Punkt der ständigen Weiterentwicklung und Bearbeitung des Data Warehouse begründet.

2.5.1 Umgebungen

Dabei sind die wichtigsten Umgebungen die Entwicklungsumgebung, die Testumgebung und die Wartungs- und die Produktionsumgebung. Je nach Bedarf kann man mehrere Bereiche ineinander übergehen lassen und muss nicht für jeden Einzelnen extra einen Bereich anzulegen.



(Data Warehouse Systeme, von Bauer und Günzel)

Gerade in Data Warehouses kommen oft neue Bestandteile hinzu, sodass man Wartung und Erweiterung schon fast synonym verwenden kann. Von den erwähnten vier Umgebungen sind aber je nach Anwendung und Kostensituation nicht alle notwendig. So bringen mehr Umgebungen höhere Sicherheit und Performanz, kosten dafür aber erheblich mehr Soft- und Hardware und somit Geld und Einarbeitungszeit. Weiterhin sind Aussagen über Scheduling, Accounting und die Qualitätssicherung zu treffen. Zur Qualitätssicherung gehört unter anderem, dass bereits vorhandene Standards in den Bereichen Programmierung, Vorgehensweise, Design, Dokumentation oder ETL genutzt werden. Das Projekt Management hat zudem dafür Sorge zu tragen, dass die entsprechenden Mitarbeiter diese Konzepte auch verinnerlichen, da sie sie sonst wahrscheinlich nicht anwenden werden.

2.5.2 Scheduling

Als weiteren wichtigen Punkt sollte man das Scheduling betrachten. Dieses gibt Auskunft über die Abfolge der Datenbeschaffungsprozesse, die Generierung und Beschickung der Data Warehouses, sowie über die Auswertung der Informationen. Das Scheduling kann manuell oder automatisiert erfolgen, wobei auch hier die Frage nach der zu nutzenden Software zu stellen ist: Sollte man eher ein Data-Warehouse-spezifisches Scheduling-System oder ein unabhängiges verwenden?

2.5.3 Accounting

Wie bewertet man jetzt die Nutzung des Data Warehouses? Wie kann man ein Art Abrechnungsverfahren für die Nutzung und Beschickung der Datenbank oder des Data Warehouse erstellen? Von welcher Art, Umfang und Organisation sollte dieses Abrechnungsverfahren sein?

Vorher sollte man sich aber fragen, was denn wirklich Kosten verursacht. Kosten verursacht die Nutzung des Data Warehouse. Dazu zählt sowohl die Beschickung der Datenbank, als auch Analyseanfragen an das Data Warehouse.

Hier bringt das alte Prinzip „Zuckerbrot und Peitsche“ eine Lösung. Man sollte die ausufernde Nutzung bestrafen (Kosten) und verantwortungsvolles und sparsames Umgehen belohnen (Vergütung). Diese Verfahren kann man auf Datenvolumen, Datenqualität und die Einhaltung vorgegebener Termine anwenden. Auf diese Weise verhindert man eine Überflutung mit unsauberen Daten und gibt Anreize für höhere Datenqualität und damit auch für eine höhere Qualität des Data Warehouse.

2.5.4 Qualitätssicherung

Die Qualität stellt einen wichtigen Erfolgsfaktor dar. Um die Qualität des Data Warehouse sicherstellen zu können, muss man erst mal Festlegungen über die Art, den Umfang und die Organisation der Qualitätssicherung treffen.
Diese Punkte betreffen im Speziellen die Sicherung der Qualität der Daten in der Basisdatenbank und den Data Warehouses, den Schedulingprozess für Extraktion, Transformation und Laden der Daten, die Standards für Data-Warehouse-Projekte bezüglich Vorgehensweise, Programmier-, Design- und Dokumentationsstandards, sowie Hardware- und Softwarestandards, sowie die Überwachung von Performanz, Datensicherheit und Zugriffsschutz.

Hier spielt die Weiterbildung der Mitarbeiter eine große Rolle. Nur wer diese Punkte verstanden hat, kann sie auch mit Überzeugung anwenden. Diese Überzeugungs- und Einführungsarbeit ist eine wichtige Aufgabe für das Projektmanagement.

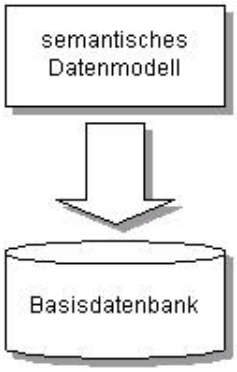
2.6 Vorgehensweise bei der Einführung

Das spezielle Vorgehen bei der Einführung eines Data Warehouse ist nicht festgelegt. Vielmehr entscheiden Termindruck und die finanzielle Situation, wie ein Data Warehouse aufzubauen ist. Generell gibt es aber drei verschiedene Ansätze, wie man ein solches Data Warehouse aufbauen könnte.

2.6.1 Top-down-Vorgehen

Bei dem Top-Down-Vorgehen wird zuerst die Datenbank anwendungsneutral anhand eines semantischen Datenmodells entworfen und mit allen relevanten Daten beschickt. Diese Aufgabe hat viele theoretische Aspekte zu berücksichtigen, gerade was den Aufbau der Datenbank angeht. Darauf aufbauend kann man dann das Data Warehouse entwerfen und laden.

Der Vorteil dieser Vorgehensweise ist ganz klar die ideale Konzipierung der Datenbank. Alle Daten können problemlos konsistent gehalten werden.
Der Nachteil ist, dass der Aufbau aufwendig ist und lange dauert. Man hat also jede Menge risikoreicher Vorleistungen zu leisten und die Anwender werden spät eingebunden. Der Mehrnutzen des so entstandenen Data Warehouse nach viel Vorarbeit muss also noch gezeigt werden.

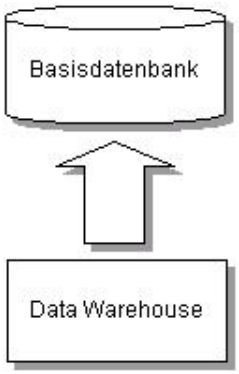


2.6.2 Bottom-Up-Vorgehen

Das Bottom-Up-Vorgehen verfolgt den entgegengesetzten Ansatz zum Top-Down-Verfahren. Hier ist es das Ziel, so früh wie möglich ein Data Warehouse bereitzustellen. Das Entwicklungsteam konzentriert sich auf ein frühzeitiges Ergebnis und die Umsetzung, vernachlässigt dafür zwangsweise die Qualität der Basisdatenbank, da diese auf den ersten Anwendungsfall (Data Mart) zugeschnitten wird. Zeitlicher Druck kann das Team sogar dazu nötigen, von sauberen Lösungen abzuweichen. Dieser Effekt ist übrigens auch sichtbar, wenn jede Abteilung ihr eigenes Data Warehouse entwickelt, bzw. wenn unkoordiniert vorgegangen wird. Hier ergeben sich wieder verantwortungsvolle Aufgaben für das Data Warehouse Projekt Management.

Der Vorteil dieses Vorgehens ist, dass die Anwender recht früh eingebunden werden und der Nutzen des Data Warehouse früh erkannt wird. Zusätzlich dazu können Missverständnisse mit den Anwendern durch diese frühen Anwendungsmöglichkeiten zu einem frühen Zeitpunkt ausgeräumt werden. Dadurch verringert sich das Entwicklungsrisiko des Data Warehouse erheblich. Der Nachteil dieser Variante besteht jedoch darin, dass die Struktur der Datenbank anwendungsspezifisch konzipiert wird.
Die Datenbank ist also nur bedingt wiederverwendbar. Wenn man die einzelnen Data Marts unabhängig voneinander oder ohne übergreifendes Konzept entwirft - oder existierende zusammenführen will, kommt es aller Wahrscheinlichkeit nach zu Inkonsistenzen. Es müsste für ein neues Projekt also auch die bereits entworfene Datenbank umstrukturiert und damit wieder von vorne angefangen werden.

Das Bottom-Up-Verfahren sollte man also nur für kurzfristige Überbrückungen einsetzen.



2.6.3 Think big – Start small

Beide bereits vorgestellte Vorgehensweisen haben ihre Vor - und Nachteile. Der Sinn dieses Ansatzes ist, die Vorteile beider Varianten miteinander zu kombinieren.
Man erstellt also ein semantisches Modell für den gesamten Anwendungsbereich - dabei muss man nicht von vorne beginnen und kann bereits existierende Modelle übernehmen. Davon ausgehend wird die Datenbank schrittweise spezifiziert und die für das jeweilige Data Warehouse notwendigen Daten identifiziert und bereitgestellt.
Mit diesem Ausschnitt des semantischen Modells kann ein Data Warehouse fertiggestellt werden. Gerade hierfür ist die enge Zusammenarbeit von zentralen und lokalen IT - Spezialisten gefordert – wieder eine Anforderung an das Projekt Management.

Der Vorteil dieser Variante liegt darin, globale Planung („think big“) und schnelle Umsetzung („start small“) miteinander zu kombinieren. Man lädt sich ein zeitlich und finanziell begrenztes Risiko auf, hat Anbindung an die global gedachte Datenbank und frühe Rücksprache mit den Anwendern.
Der Nachteil könnte sich daraus ergeben, dass einzelne Entscheidungen unvorteilhaft getroffen werden und man sich zu stark an einer der beiden vorherigen Vorgehensweisen orientiert. Damit würde man sich auch die Nachteile der jeweiligen Vorgehensweise einhandeln.

2.7 Projekt (Organisation und Phasen)

2.7.1 Data-Warehouse-Projektorganisation

Wie bereits angesprochen ist ein Data-Warehouse-Projekt ein IT-Projekt, dass in den Bereich der Datenbanken fällt. Für die Integration vieler heterogener Datenquellen ist ein genaues Verständnis der Semantik der Inhalte notwendig, was die Wichtigkeit für die Einbindung der Datenlieferanten, also der Mitarbeiter (vor allem nicht IT-Personal!) verdeutlicht.

In Data-Warehouse-Projekten fließen Daten aus den unterschiedlichsten Bereichen zusammen. In Bezug auf den zu erwartenden Abstimmungsaufwand kommt auf das Projekt Management hier wieder eine große Aufgabe zu.
Die Kommunikation zwischen allen Beteiligten und Verständnis auf allen Seiten ist unbedingt Voraussetzung für den Erfolg des Projektes. Die Vergangenheit hat gezeigt, dass fehlende Kommunikation oft der Grund für das Scheitern von solchen Projekten war.

Für das Management kommen dabei große und wichtige Aufgaben zu. Diese lassen sich grob in die Bereiche Planung, Organisation, Steuerung, Durchführung unterteilen. Dafür sind Standards unverzichtbar - in den meisten Firmen sind bereits Projektentwicklungsstandards vorgesehen oder bereits realisiert.

Wie bereits erwähnt sind Data-Warehouse-Projekte durch eine besondere Komplexität gekennzeichnet. Sie verbinden Quelldaten, die heterogen (strukturell, räumlich, technisch, organisatorisch) vorliegen und in eine einzige, zentrale Datenbank integriert werden sollen. Weiterhin wirken Personen mit unterschiedlichem Erfahrungsgrad aus total unterschiedlichen Fachrichtungen zusammen. Das erfordert Teamfähigkeit auf allen Seiten.
Da man aber nicht immer von dem guten Willen oder Teamfähigkeit aller ausgehen kann (manche fürchten den Verlust von Macht innerhalb der Firma), ist die Kompetenzverteilung und die Projektleitung besonders wichtig.

Die genaue Definition folgender Aufteilungen sind daher von grundlegender Bedeutung:

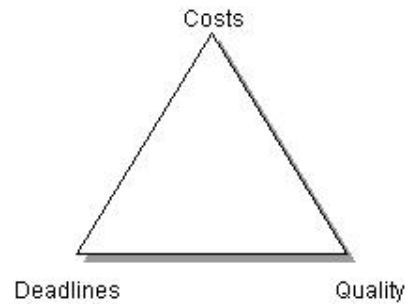
- Projektrollen
- Projektsteuerung
- Projektteam
- Rollenverständnis
- Kommunikation
- Konfliktmanagement
- Qualitätssicherung
- Dokumentation

Die Rollenverteilung ordnet jeder definierten Rolle eine Menge von Verantwortungen und Aufgaben zu, die derjenige, dem diese Rolle zugeteilt wurde bestmöglich wahrzunehmen hat. Das Management hat dafür Sorge zu tragen, dass dafür die geeigneten Personen ausgewählt werden. Denkbare Rollen sind die des Auftraggebers, des Auftragnehmers, der Entscheidungsträger, sowie einzelner Projektmitarbeiter.

Das Projektteam und das Rollenverständnis – diese Themenbereiche eines Data-Warehouse-Projektes lassen sich in das betriebliche Fachwissen, die zu verwendende Technologie und Projektmanagement und Methoden einteilen. Die durchzuführenden Tätigkeiten können davon abgeleitet werden. So sollte beispielsweise der Spezialist der Fachabteilung über sicheres, detailliertes Fachwissen verfügen oder der Anwender sich mit standardisierten Analysemethoden auskennen. Hier sollte die Auswahl der Mitarbeiter also nach Kenntnisstand und Erfahrung in den einzelnen Bereichen und Aufgaben geschehen. Nur so kann das geforderte Rollenverständnis durch die entsprechenden Mitarbeiter umgesetzt werden.

Die Kommunikation hat den Austausch von Informationen zum Zweck. Diesen wiederum ist kann man in zwei Ebenen unterteilbar. Die inhaltliche Ebene umfasst Ziele, Termine, Aufgaben und Kosten. Die Beziehungsebenen spiegelt Ängste, Hoffnungen, Visionen und das Verständnis eines jeden einzelnen wider.
Ein Modell zur Umsetzung der erfolgreichen Kommunikation ist die Datenkonferenz. Hier soll ein gemeinsames Verständnis der Quelldaten, sowie die Lösung von Integrationsproblemen als auch die Verteilung von Zuständigkeiten geklärt werden.
Ziel ist es, eine einvernehmliche Lösung zu finden und dabei alle Beteiligten einzubeziehen. Die Effizienz und Durchschaubarkeit sollen dadurch gesteigert und die auftretenden „Reibungsverluste“ zwischen den Mitarbeitern minimiert werden. Alle sollen eine gemeinsamen Sprache sprechen und nicht aneinander vorbei reden. Das setzt auch eine unternehmensweite Definition von Begriffsstandards voraus.

Das Konfliktmanagement tritt dann ein, wenn die „Reibungsverluste“ zwischen einzelnen Mitarbeitern oder ganzen Abteilungen unüberwindbar scheinen. Hier ist wieder das Projektmanagement gefragt. Ein Modell für den Ausgangspunkt vieler Konflikte verdeutlicht das „magische Dreieck“. Es verdeutlicht die Widersprüchlichkeit der drei Teilziele Kosten, Termine und Funktion und Qualität. Generell steht die Konfliktvermeidung aber vor dem Konfliktmanagement.



(Data Warehouse Systeme, von Bauer und Günzel)

Anhand dieses Beispiels sollen noch mal die bisher erarbeiteten Punkte erwähnt werden, die bei einer konfliktpräventiven Projektplanung zu berücksichtigen sind.
Die Auswahl eines geeigneten (unternehmensinternen) Sponsors zur Durchsetzung des Data Warehouse, die Benutzung des Ansatzes „Think big – start small“ und die Festlegung des Umfanges und der Rolle des Repositoriums sind dabei genau so wichtig, wie die Betriebsphase des Data Warehouse, die Priorität des Projektes und Verteilung der Verantwortungen und Rollen auf die Mitarbeiter. Dabei sollte man möglichst alle Interessen der Beteiligten berücksichtigen und zur Konfliktvermeidung regelmäßige Teammeetings durchführen. Für den Erfolg des Data Warehouse ist der Ansatz des „Rapid Prototyping“, also der ständige Kontakt mit dem Anwender notwendig.

Die Qualitätssicherung des Data Warehouse umfasst alle Maßnahmen, die zur Qualität der Prozesse, Produkte, Konzepte und Daten beitragen. Die Qualitätssicherung bezieht sich auf die Einheiten des Projektverlaufs und auf einzelne Projektphasen. Die Überprüfung der Qualität sollte in phasenbezogenen Projektergebnissen geschehen. Dazu ist eine Milestone-vorgabe von Bedeutung, damit nicht zu große Abstände zwischen den einzelnen Entwicklungsstufen entstehen.
Die Kriterien für die Qualitätssicherung sind vor allem Performanz, Zuverlässigkeit, Sicherheit, Erweiterbarkeit, Skalierbarkeit, Wiederverwendbarkeit, Wartungsfreundlichkeit, Bedienerfreundlichkeit und Robustheit.
Da das Data Warehouse ein stets fortlaufender Entwicklungsprozess ist, sollte auch der Prozess der Qualitätssicherung ein projektbegleitender Prozess und gut durchdacht sein. Man darf diesen Prozess nicht mit dem ans Ende verlagerten Prozess des Testens verwechseln, da es dort für qualitätssichernde Entscheidungen bereits zu spät sein kann.

Der Standard für die Beschreibung der Qualität sieht unter anderem die Beschreibung von Testfällen, den Aufbau einer Testumgebung mitsamt dem Test der Funktionalität und dem Durchführen sämtlicher Tests (Funktionalität, Integration, Belastung) vor. Anhand dieser Ergebnisse sollen die Vorgaben durch den Meilensteinentwurf und die Termine und Ressourcen des Projektplans überprüft werden.

Die Aufgabe der Dokumentation ist für eventuell folgende Data-Warehouse-Projekte wichtig. Unter Zuhilfenahme der Dokumentationen vorhergegangener Projekte kann man alte Fehler vermeiden und die Entwicklungskosten so reduzieren. Am besten eignet sich hierfür eine Art Projektdatenbank, die den Aufwand für jeden Einzelnen so gering wie möglich hält und ständig Informationen liefern kann. („Knowledge Management“)

In dem bereits erwähnten Repositorium sind diese Ergebnisdokumente zu hinterlegen. Das Management sollte vorher Vorgaben bezüglich Inhalt, Form, Organisation und Zugriffslegitimation zu den Quell-, bzw. Zieldaten festlegen.

2.7.2 Data-Warehouse-Projektphasen

Das hier vorgestellte Modell betrachtet die Machbarkeitsstudie nicht mehr. Diese könnte man auch als erste Projektphase betrachten. Die hier betrachteten Phasen behandeln die Analyse, das Design und die Implementierung des Data Warehouse.

Das Prozessmodell legt den organisatorischen Rahmen eines Projektes fest. Bisher entwickelte und häufig verwendete Modelle sind das Wasserfallmodell, das Prototypenmodell, das evolutionäre/inkrementelle Modell, sowie das Spiralmodell.

Welches Modell gewählt wird oder ob ein neuer Typ entwickelt wird, ist dem Management überlassen. (Obwohl es von entscheidendem Vorteil ist, auf bewährte Standards zurückzugreifen)

Das gewählte Phasenmodell hat im Wesentlichen folgende Aufgaben zu erfüllen.
Es muss sicherstellen, dass alle wichtigen Teilaufgaben einbezogen sind, alle (projektspezifischen) Abhängigkeiten berücksichtigt werden, alle Teilaufgaben ausreichend behandelt und alle Teilergebnisse und Empfehlungen rechtzeitig geprüft und dokumentiert wurden.

Die Projektabwicklung unterteilt sich somit in die Teilbereiche der Analysephase, in der das externe Know-how geprüft wird und eine Ist-Analyse erstellt wird, das Soll-Konzept, die Analyse der Datenquellen, die erste Fokussierung und die Entscheidung über die multidimensionale oder relationale Realisierung auf Basis fachlicher Anforderungen.

Die Kapazitätsplanung und Hardwareauswahl ist dabei ebenso wichtig, wie die in der Designphase auszuführenden Prozesse der Modellierung der Datenquellen, dem Design der Basisdatenbank, des Data Warehouse, des Datenflusses, der Benutzeroberfläche und eines Sicherheits- und Berechtigungskonzeptes.

In der Implementierungsphase gilt es, besonders die Metadatenhaltung, die Implementierung eines Sicherheits- und Berechtigungskonzeptes, sowie das Aufbauen einer Entwicklungs-, Test-, Produktions- und Wartungsumgebung zu realisieren. Das Prinzip des „Rapid Prototyping“ sollte dabei auf jeden Fall verfolgt werden.

2.8 Erfolgsfaktoren und Wirtschaftlichkeit

2.8.1 Erfolgsfaktoren beim Aufbau eines Data Warehouses

Jedes Data Warehouse muss individuell entwickelt werden.
Somit ist klar, dass man nur allgemeine Kriterien für den Erfolg oder Misserfolg angeben und dabei nicht zu sehr ins Detail gehen kann. Aufgrund vieler Erfahrungsberichte kristallisieren sich aber einige Punkte heraus, denen man besondere Aufmerksamkeit widmen sollte. Einer dieser Punkte ist das Projektmanagement. Dabei sind sowohl die institutionellen als auch die funktionalen Aufgaben gemeint.

2.8.1.1 Projektorganisation

Der Projektleiter sollte anwenderorientiert handeln, da ein Data Warehouse hauptsächlich ein fachspezifisches Problem darstellt. Daher ist es von Vorteil, wenn der Leiter auch mehr Fachexperte als IT-Spezialist ist. Wichtig ist die permanente Ausrichtung auf das betriebliche Umfeld.
Ein geeigneter Sponsor sollte auch auf jeden Fall vorhanden sein, da man ansonsten Gefahr läuft, dass das Data Warehouse Projekt in der Masse vorhandener Projekte untergeht. Die Unternehmensführung sollte also vom Nutzen und der Wichtigkeit des Data Warehouse Projektes überzeugt sein und das Projekt auch dementsprechend unterstützen, da es sonst leicht am Widerstand einzelner Interessengruppen scheitern könnte.

Der Nutzen für die Anwender darf nie vergessen werden. Man muss also im ständigen Kontakt zu diesen Gruppen bleiben und in möglichst kurzen Abständen neue Teilergebnisse

präsentieren, um diese mit den Anwendern zu diskutieren. Das ist deshalb besonders wichtig, da die meisten Endanwender erst dann wissen was sie wollen, wenn sie Produkt vor der Nase haben.

Gerade bei der Einführung oder dem erstmaligen Entwurf eines Data Warehouses verfügt man höchstwahrscheinlich noch nicht über das notwendige Wissen, um das Projekt erfolgreich durchzuführen. Aus diesem Grund sollte man auf Berater zurückgreifen, die sich mit der Problematik besser auskennen. Ein Vertrag mit einer externen Beraterfirma kann die Lösung sein. Dabei sollte man natürlich sehr sorgfältig bei der Auswahl der Berater vorgehen, da ein erheblicher Teil des Erfolges von der Qualität der gelieferten Informationen abhängen wird.

2.8.1.2 Projektabwicklung

Ein Vorgehensmodell, dass die Teilergebnisse mit den gesetzten Zielen vergleicht, ist unentbehrlich für ein Data-Warehouse-Projekt. Der Erfolg und der Nutzen des Projektes muss anhand dieser Evaluierungen deutlich nachgewiesen werden. Die Projektbeschreibung zusammen mit der Machbarkeitsstudie am Anfang des Projektes sind also von entscheidender Wichtigkeit.
Dabei ist zunächst einmal für ein unternehmensweites Begriffsverständnis zu sorgen. Nur so reden wirklich alle Beteiligten eine gemeinsame Sprache. Auch der Vorteil der Basisdatenbank als Informationspool ist nur so gewährleistet.
Der ständige Kontakt zu den Anwendern ist zu suchen. Man sollte also in regelmäßigen Abständen Weiterentwicklungen zur Verfügung stellen. Dafür eignen sich die vorgestellten Verfahren „Rapid Prototyping“ und „Think big, Start small“.

Der Widerstand innerhalb eines Unternehmens – bedingt durch die Angst vor Machtverlust oder mühsam aufgebauter Strukturen – muss durch intensive interne Werbung möglichst gering gehalten werden.
Bei der Projektplanung sind nicht nur die Kosten für den Aufbau des Data Warehouse zu betrachten, sondern auch die Kosten für die Administration, Wartung und Pflege. Der Gedanke, dass ein Data Warehouse niemals abgeschlossen ist, sondern stets weiterlebt, kommt somit besser zum Tragen. Der Weg ist das Ziel.
Ebenso sollte nicht vergessen werden, dass selbst bei all zu guter Planung nie alles gut geht. Dementsprechend sollte man genug Zeit für Tests, Fehlerbeseitigungen und Datenbereinigung einplanen.
Besonders in den ersten Projekten sollte man auf externe Berater zurückgreifen um erste schwerwiegende und meist auch kostenintensive Fehler zu vermeiden.
Ganz wichtig ist, dass sich die Nutzung des Data Warehouse wohl nicht nur auf interne Daten beschränken sollte – jeder Manager versucht ebenso, möglichst viele externe Quellen nutzen zu können. Man sollte sich also überlegen, inwieweit man extern Quellen in das Data Warehouse mit einbeziehen könnte.

2.8.2 Wirtschaftlichkeitsbetrachtungen

Bei einem solch komplexen System, wie einem Data-Warehouse-System stellt sich die Frage, welche Daten zur Rentabilitätsbetrachtung herangezogen werden sollen. Man muss in jedem Fall zeigen, dass das Data Warehouse langfristig erfolgversprechend ist und trotz der hohen Komplexität überschaubar bleibt. Die folgenden Versuche, die beiden Alternativen (Einführen eines Data Warehouse oder nicht) einander gegenüberzustellen, zielen insgesamt auf alle relevanten Faktoren. Oftmals stellt sich dabei das Problem, dass man einen Nutzen/Vorteil nicht in Euro oder Dollar aufrechnen kann.

2.8.2.1 Kostenbetrachtung

Die wichtigsten Kostentreiber sind Studien und Testumgebungen, Beschaffung, Bereinigung und Speichern von Daten, die eingesetzte Soft- und Hardware, das Know-how für die Organisation eines Data Warehouse, sowie die Schulung der Anwender und die Wartung im laufenden Betrieb.
Die Frage, die sich häufig stellt ist, was dieses konkrete Data Warehouse kostet. Dabei ist es schwierig, alle Kosten dem ersten Data Warehouse zuzurechnen. Folgende Data Warehouses werden erfahrungsgemäß durch die Informationen, die aus dem ersten Data Warehouse gewonnen wurden weit billiger sein als ihre Vorgänger.
Was statt dessen betrachtet werden sollte sind die Möglichkeiten, wie man entstehende Kosten vermeiden könnte. So führt der Einsatz externer Berater im Normalfall dazu, Lehrgeld für entstandene Fehler zu sparen. Dies ist gerade bei dem wichtigen Punkt der Projektorganisation und des –managements stark ausschlaggebend.

2.8.2.2 Nutzenbetrachtung

Der Nutzen eines Data Warehouse lässt sich im Vorfeld meist nur schwer quantifizieren. Anstelle von ausgedachten Zahlen versucht man, den Nutzen eines Data Warehouse in verschiedene Kriterien zu unterteilen: Prozess-, Produktivitäts-, Wahrnehmungs- und Produktkriterien. Diese Einteilung hilft, die Effizienz-, Visualisierungs-, Präsentations- und Qualitätssteigerungen in verschiedene Gruppen zu unterteilen um somit allen Beteiligten die vielfachen Einsatzmöglichkeiten bewusst werden zu lassen.
Interessant ist am Ende, ob die vorhergesagten Vorteile wirklich eingetreten sind, ob sich das Data Warehouse tatsächlich rentiert hat. Problematisch bei den entsprechend nötigen genauen Datenerhebungen könnte sein, dass sich einzelne Mitarbeiter überwacht fühlen oder dass sich einige Vorteile nicht monetär bewerten lassen. (Wie misst man die Bereitschaft eines Kunden, den Lieferanten zu wechseln – gibt es ein Maß für Loyalität?)

Alles in allem ist ein Data Warehouse Projekt ein Anwendungsprojekt, dass an der vorgegebenen Situation im Unternehmen ausgerichtet und auch immer wieder an diese angepasst werden wird. Die größte Fehlerquelle liegt in der Komplexität. Laut Bauer und Günzel umfasst schon ein Data-Warehouse-Projekt mittlerer Größe 500 bis 100 Einzelaktivitäten. Dies zeigt auch, dass hier nur ein verhältnismäßig grober Überblick über die Aufgaben, die auf das Data Warehouse Projekt Management zukommen, gegeben werden kann, da eine detaillierte Auflistung den Rahmen dieser Arbeit sprengen würde.

3. Software - und Hardwareaspekte

3.1 Softwareauswahl – Wichtige zu beachtende Punkte

Die Softwareauswahl stellt in der Entwicklungsphase eines Data Warehouses einen wichtigen Punkt dar, da von ihr wichtige weitere Schritte abhängen. Gewünscht wäre natürlich ein perfektes Produkt, welches den Anforderungen des einzelnen Projektes in jeder Hinsicht genügt. Jedoch sind auf dem Markt meist nur Kompromisslösungen zu finden. Um die Güte der einzelnen Produkte voneinander abgrenzen zu können, sind bestimmte Bewertungskriterien nötig. Fehlentscheidungen können sich zu einem späteren Zeitpunkt als sehr kostenintensiv erweisen, beispielsweise, wenn zusätzlich gewünschte Funktionalitäten nicht mehr integrierbar sind, Datenvolumina die Kapazität überschreiten, Wartung zu teuer ist oder Funktionalität und Oberfläche aufgrund hoher Komplexität vom Benutzer nicht akzeptiert werden.
Spätestens dann müsste auf ein neues Produkt umgestiegen werden was wiederum mit Kosten in Form von Umschulungen, Lizenzen usw. zu Buche schlagen würde. Meist ist die Produktauswahl aber schon auf einen bestimmten Rahmen beschränkt, weil bestimmte Komponenten vorhanden sind, die mit der neuen Software zusammenarbeiteten müssen. Oft besitzen Firmen eine Vielzahl solcher Insellösungen und die verschiedenen Arbeitsgruppen sind nicht bereit, auf ihre Lösung zu verzichten. In Hinblick auf eine homogene Zusammenarbeit der einzelnen Abteilung sollte auf Dauer jedoch eine Vereinheitlichung der Softwarelösungen vorangetrieben werden.

Überblick über verschiedene Produktklassen anhand von Referenzarchitekturen

Zum Aufbau eines Data Warehouse Systems benötigt man i.A. folgende Komponenten:

1. Produkte zur Datenbeschaffung (ETL-Werkzeuge): Diese Produkte dienen der Überführung der Daten aus der Datenquelle in das Data Warehouse System. ETL steht dabei für Extraktion, Transformation und Laden

2. Datenbanksystem für das Data Warehouse: Meist reichen die bereits im Unternehmen vorhandenen Datenbanksysteme für die geforderten Zwecke aus.

3. Analysewerkzeuge: Die Palette der Analysemöglichkeiten reicht von Reporting bis hin zu Mining, an dieser Stelle werden wir uns auf OLAP-Produkte beschränken.

4. Repositorien: Diese werden zur Metadatenvewaltung im Rahmen von Data Warehouse Systemen eingesetzt. Ausreichend sind oft jedoch schon die werkzeugeigenen Werkzeuge zur Metadatenverwaltung, wie die der ETL-Tools.

Weitere Produktklassen wie Modellierungs- und Portierungswerkzeuge sind an dieser Stelle denkbar.

Bewertungskriterien für die Softwareauswahl

Im folgenden werden wichtige Kriterien für eine erfolgreiche Bewertung von Softwareprodukten vorgestellt.
Objektivität ist ein guter Vorsatz für die Produktbewertung, bei genauem Hinsehen jedoch sehr schwierig durchzuführen. Man sollte versuchen, Objektivität in Form von allgemein anerkannten Kriterien einzubringen und politische Entscheidungen außen vor zu lassen.

Einbindung relevanter Personengruppen für die Anforderungsanalyse ist sehr wichtig für die spätere Akzeptanz des neuen Produktes durch die Anwender. Evtl. sollte den späteren Anwendern auch ein Mitspracherecht zugesichert werden. Problematisch ist hierbei, das zumeist lediglich 20% der Anwender 80% des Bedarfs beschreiben sollen (80:20 Regel). Leider werden Anwender von den Entwicklung oft missverstanden und somit sinkt die Akzeptanz des Produktes nach der Einführung.

Eine ***Kostenkalkulation*** sollte zu einem frühen Zeitpunkt durchgeführt werden, da in vielen Fällen, die Soft- und Hardwarekosten deutlich unterschätzt werden. Eventuell können somit Lösungen ausgeschlossen werden, die das Budget deutlich überschreiten. Allerdings ist hier auch immer zu prüfen, ob der höhere Preis nicht doch durch eine höhere Funktionalität gerechtfertigt ist.

Nachvollziehbarkeit der Entscheidung dient zum einen der Akzeptanz durch den Anwender als auch der Objektivität.

Einsatz von ***Marktstudien*** gerade auf dem Bereich der ETL- und OLAP-Werkzeuge sind meist sehr hilfreich, da sie häufig genaue technische Informationen enthalten, anhand derer man erkennen kann, welche der Funktionen des Produktes in Hinblick auf die Firmenzwecke ausreichend bzw. überflüssig sind.

Eine mögliche Vorgehensweise bei der Produktbeschaffung könnte die folgende sein:

1. Aufnahme der Anforderungen: Es gilt nun, sowohl technische als auch finanzielle Anforderungen zu erfassen. Finanzielle Anforderungen ergeben sich aus der IT-Strategie, technische am besten durch Interviews mit allen betroffenen Personenkreisen. Es ist dabei sinnvoll, sich zusammenzusetzen und Minimalanforderungen zu definieren. Auch der Einsatz eines standardisierten Fragebogens kann von Vorteil sein, um Anforderungskonflikte besser offenzulegen.

2. Bewertungsschema festlegen: Nicht alle geforderten Kriterien sind erfüllbar, da sie auch oft im Widerspruch zueinander stehen. Man unterscheidet deshalb auch zwischen Musskriterien, wichtigen als auch optionalen Kriterien. Bei der Erstellung eines Bewertungskriteriums ist insofern Vorsicht geboten, als dass man die Objektivität wahrt. Dabei hilft auch nicht unbedingt ein Punktesystem, da die Punktevergabe für bestimmte Kriterien auch subjektiv erfolgt sein kann. Kriterien wie Benutzerfreundlichkeit und Zuverlässigkeit lassen sich zudem nicht objektiv bewerten.

3. Informationsbeschaffung: Am Markt wird man leicht von der Informationsflut überwältigt. Eine Fokussierung ist deshalb unbedingt notwendig. Marktstudien als auch Consultingleistungen können deshalb hilfreich sein.

4. Vorauswahl: Im weiteren Verlauf der Entscheidungsfindung sollte man sich auf zwei bis drei Produkte festgelegt haben. Nun sollten Detailfragen die Auswahl beeinflussen, z.B. Plattform, Schnittstellen, Architektur, etc. Steht die Entscheidung unter hohem Zeitdruck, kann es von Vorteil sein, sich für Marktführer zu entscheiden. Falls keine besonderen Funktionalitäten gefordert sind, hat man damit eine hohe Chance auf den Erwerb einer befriedigenden Lösung, wenn auch nicht optimal.

5. Feinevaluierung: Am Ende sollte der gesamte Anforderungskatalog gegen die Produkte abgeglichen werden, dazu sollte man sich auf ein geeignetes Vokabular einigen und festlegen, was unter den einzelnen Begrifflichkeiten zu verstehen ist.

6. Prototypenaufbau: Falls man Prototypen einsetzen möchte, sollte man sich auf das Wesentliche beschränken, um gewisse Grundfunktionalitäten zu testen.

7. Entscheidung: Aus dem Prototypentest sollte sich idealerweise herauskristallisieren, welches Produkt am geeignetsten für das Unternehmen ist. Auch hier sind wiederum die einzelnen Kriterien gegeneinander abzuwägen, ohne sich in Detailfragen zu verlieren.

Für die Produktauswahl sind außerdem Kriterien wie das Herstellerprofil, Support, Hardware- und Softwarevoraussetzungen, Preismodell, Benutzerfreundlichkeit, Zuverlässigkeit, Performanz als auch Skalierbarkeit zu untersuchen.
Für einzelne Produktklassen ergeben sich zudem noch spezielle Auswahlkriterien, die an dieser Stelle kurz vorgestellt werden sollen.

Kriterien für Datenbeschaffungswerkzeuge (DBW) sind zum ersten, ob man eine Eigenentwicklung oder ein Fremdprodukt vorziehen möchte. Dabei spielen Kosten, Erfahrung mit der Entwicklung als auch Betriebsgeheimnisse eine wichtige Rolle. So ist eine Eigenentwicklung in der Regel besser auf die eigenen Bedürfnisse zugeschnitten, bei sehr komplexen Anforderungen sollte man aber eher auf Systeme von externen Spezialisten zurückgreifen. Die Architektur des DBW sollte dem Administrator aus Performanzgründen Optionen zu Lastverteilung bereitstellen.
Man sollte auch auf Möglichkeiten für die Steigerung des Datendurchsatzes Wert legen. Auf dieses Tuning wird in einem der nächsten Abschnitte noch genauer eingegangen. Es gibt noch weitere entscheidungsrelevante Kriterien wie Schnittstellen, Scheduling, Metadaten und Fehlerbehandlung die an dieser Stelle jedoch nicht weiter beschrieben werden sollen.
OLAP-Produkte können nach der Datenhaltung klassifiziert werden. Jede Alternative, egal ob relationaler oder multidimensionaler Art hat ihre Vor- und Nachteile. Direkte Vergleiche sind schwierig, nicht nur aufgrund architektonischer Unterschiede, sondern auch wegen unterschiedlicher Anwendungsgebiete. Die Architektur stellt das wichtigste Kriterium dar, da es hier im späteren Verlauf kaum noch Veränderungsspielräume gibt. Anfragewerkzeuge sollten grafisch und intuitiv sein. Optionen zur Anfragedefinition sind genauso wie eine individuelle Errichtung von Filterfunktionen von Vorteil. Es sollten Reportingfunktionen vorhanden und vorgefertigte Berichte für unerfahrene Anwender enthalten sein. Eine hohe Analysefunktionalität ist erwünscht. Ein Berechtigungskonzept ist gerade bei großen Systemen mit einer hohen Anwenderzahl unverzichtbar, da nur so Sabotage bzw. Spionage bekämpft werden können. Auch eine Web-Anwendung ist heutzutage erwünscht und reicht vom einfachen Abspeichern von Berichten im HTML-Format bis hin zu Java-Implementierungen.

Bis hierher haben wir uns mit der Auswahl geeigneter Software beschäftigt und gehen nun kurz auf die Hardwarekriterien ein.

3.2 Hardwareauswahl

Viele Data Warehouse Systeme sind zentral angelegt, d.h. die gesamte Datenmenge wird auf einem zentralen Server abgelegt. Dies hat den Vorteil, dass nur ein System gewartet werden muss jedoch kann es schnell zu Überlastungen und Engpässen führen, da die gesamten Anfragen von dem einen zentralen System zu bewältigen sind.
An dieser Stelle werden einige wichtige Punkte vorgestellt, die es bei der Auswahl von Hardwaresystemen zu beachten gilt.

Verfügbarkeit und Fehlertoleranz: Bestimmte Serversysteme müssen rund um die Uhr aktiv sein, Ausfälle sind hier nicht akzeptabel und Fehlertoleranz ist enorm wichtig.

Datensicherung beschreibt den Aspekt, dass für den Fall eines Datenverlustes Vorkehrungen getroffen werden, mit deren Hilfe das System schnell wieder einsatzbereit gemacht werden kann. Eine Art davon ist das Datenbackup.

Antwortzeiten sollen möglichst gering gehalten werden. Gerade bei „interaktiven" Systemen mit vielen Nutzern ist dieser Aspekt nicht trivial. Als Lösung bieten sich Cluster- und Multiprozessorensysteme an.

Skalierbarkeit bei steigender Nutzer- und Datenzahl des Systems sorgt dafür, dass das System auch im späteren Einsatzbereich lauffähig und erweiterbar bleibt.

Speicherungsarten für die Daten

In einem Data Warehouse System müssen Daten im Bereich von mehreren 100 Gigabyte bis hin zu mehreren Terabyte gespeichert werden. Wichtig dabei ist, dass der Zufluss von dem Quelldatensystem ins Warehouse gewährleistet bleibt und das auch bei steigendem Datenvolumen.
Wichtige Leistungsmerkmale eines Speicherungssystems ist die Zugriffszeit und die Lebensdauer der Daten. Zugriffszeit und Übertragungsdauer bestimmen die Gesamtzugriffsdauer. Durch die Anwendung von inkrementellen Speicherungstechniken können die Daten im Falle eines Verlustes nicht schnell rekonstruiert werden. Um die Daten effizient wieder herstellen zu können, kommen RAID-Systeme (redundand array of independent drives) als auch Storage Area Networks zum Einsatz.

RAID Systeme spiegeln die Daten auf mehreren Platten gleichzeitig. Außerdem können aufeinanderfolgende Blöcke auf verschiedenen Platten gespeichert werden. Dadurch werden sowohl die Datensicherheit als auch die Datendurchsatzrate erhöht. Die einzelnen Festplatten werden als eine logische Einheit adressiert, die Daten aber auf allen Platten redundant verteilt (RAID 1). Fällt ein Plattensystem aus, so können die Daten noch von den restlichen Platten gelesen werden. Defekte Platten können während der Laufzeit ausgewechselt und neu beschrieben werden.
RAID-Systeme erhöhen zwar die Sicherheit und den Datendurchsatz, nicht jedoch die Zugriffszeit.

Storage Area Networks beruhen auf einem lange bekannten Prinzip. Dabei wird eine n:m-Beziehung zwischen Rechnern und Festplatten aufgebaut. Verschiedene Rechner können auf dieselben Daten zugreifen. Kombinieren kann man dieses System mit RAID und erhält eine erhöhte Ausfallsicherheit sowohl auf Festplatten als auch auf Rechnerseite.

Möchte man die Daten langfristig speichern, so greift man im allgemeinen auf Archivspeichermedien, früher meist Magnetbänder oder nun verstärkt optische Datenträger zurück. Magnetbänder haben eine Lagerdauer von 5 bis 25 Jahren, sind jedoch sehr empfindlich gegenüber Magnetfeldern, schon heute ist eine Ablösung durch beschreibbare Optische Medien wie DVRs mit Kapazitäten um 9 GB und Lebenszeiten von bis 150 Jahren abzusehen.

Multiprozessorsysteme (MP)

Besonders auf dem Gebiet der Datenbank- und Serveranwendungen finden Multiprozessoren ein breites Anwendungsfeld, da die Verarbeitung vieler Anfragen gleichzeitig auf diese Weise kostengünstig ermöglicht wird. Es gibt dabei verschiedene Architekturen für solche Systeme, die sich alle je nach Anforderung unterschiedlich gut eignen.

Symmetrische Multiprozessorsysteme: hier greifen verschiedene Prozessoren über denselben Datenbus auf den Speicher zu. Dies ist bis zu einer gewissen Zahl von Prozessoren vertretbar. Jedoch gerät man ab einer bestimmten Prozessoranzahl an die Grenze der Übertragungsmöglichkeit durch den Datenbus und die Prozessoren fangen an, sich gegenseitig zu blockieren. Die Programmierung unterscheidet sich nicht sehr stark von der für herkömmliche Systeme, verschiedene Programmteile (Threads) werden auf einzelne Prozessoren ausgelagert und später wieder synchronisiert.

Bei ***Rechner mit nichtuniformen Speicherzugriffen*** besitzen die einzelnen Prozessoren im Gegensatz dazu ihren eigenen Speicher, jedoch gibt es auch gemeinsame Adressbereiche. Speicherzugriffe auf den eigenen Speicher gehen sehr schnell, auf fremde Bereiche dementsprechend länger. Programme müssen nicht speziell angepasst werden, jedoch sollte man sie aus Performanzgründen so anpassen, dass die Prozessoren nur ihren (affinen) Speicherbereich benutzen.

Cluster stellen eine weitere Architekturmöglichkeit da, wenn es darum geht, gut skalierende Systeme zu verwenden. Es handelt sich hierbei um einen Rechnerverbund, der als eine Einheit agiert. Eine Reihe eigenständiger Systeme ist hier über ein Netzwerk verbunden. Die Netzwerkgeschwindigkeit sollte dementsprechend hoch sein, weil Daten zwischen den einzelnen Systemen über das Netzwerk ausgetauscht werden sollen. Diese Architektur ist sehr einfach erweiterbar, weitere Rechner werden einfach an das Netzwerk angeschlossen, allerdings stellt die Stabilität teilweise ein Problem dar. Außerdem ist die Geschwindigkeit und Latenzzeit des Netzwerkes bei hohen Anfragefrequenzen von ausschlaggebender Bedeutung. Mit der Anwendung von Clustern kann man sich außerdem gut gegen Ausfälle einzelner Rechner absichern.

Backupstrategien

Ursachen für Systemausfälle reichen von menschlichem über technisches Versagen bis hin zu Katastrophen wie Bränden und Erdbeben, Sabotage und Terrorismus. Um schnell wieder ein lauffähiges System einrichten zu können, ist die Aufstellung von Notfallplänen sinnvoll. Die Errichtung eines Backups beeinflusst jedoch die Antwortzeiten für den Anwender und sollte deshalb zu Nachtstunden und Wochenenden passieren. So sollte auch nur in bestimmten Intervallen der gesamte Datenbestand gesichert werden und dafür zumeist nur die Änderungen der Daten. Möchte man die Daten wiederherstellen, so installiert man zunächst das vollständige Backup und nachfolgend die durchgeführten Änderungen.
Es gibt dabei verschiedene Arten von Backups: Cold Backup bei inaktivem System, Hot Backup bei laufendem System und Logisches Backup in Exportdateien.

Die Lagerung der Backups sollte fernab der Firma passieren um die Wahrscheinlichkeit einer Involvierung der Backupdaten in eine lokale Katastrophe möglichst gering zu halten. Weiterhin können bestimmte Watchdog Timer die Systemfunktionalität überprüfen und die Verantwortlichen zu gegebener Zeit benachrichtigen. Regelmäßige Proben des „Ernstfalles“ bieten zudem eine gute Möglichkeit die eingesetzten Verfahren zu testen und Schwachstellen aufzuzeigen.

4 Betrieb eines DWS

Im folgenden Abschnitt werden die Aufgaben beschrieben, die für den Betrieb eines DW-Systems notwendig sind. Die Aufgaben, die für seine Erstellung notwendig waren, sind nun bereits bekannt.

Es kann nicht oft genug betont werden, dass ein Data Warehouse System kein Produkt ist, das man nur einmal besorgen muss, sondern eher eine Aufgabe, die niemals beendet ist. Diese Aussage begründet sich darin, dass ein DW-Projekt zwar nur einmal aufgebaut, um es dauerhaft verwenden zu können aber immer wieder aktualisiert und gepflegt werden muss. Schwerpunkte liegen hierbei in der Administration, der Datenbeschaffung, der optimierten Datenspeicherung und der Analyse. Darüber hinaus werden die Arbeitsbereiche Repositorium und Sicherungsmanagement etwas näher betrachtet.

4.1 Administration

Die Wartung eines Datawarehoussystems und die damit verbundene Qualitätssicherung kann den Aufwand für seine Erstellung um ein vielfaches übersteigen. Dabei gibt es eine Vielzahl von Aspekten, die nicht vereinzelt sondern eher als ein verzahnter Prozess betrachtet werden müssen. An dieser Stelle werden wir die wichtigsten dieser Aspekte vorstellen.

Systemtechnische Aspekte bezeichnen die Pflege der Hard- und Software. Während sich die Hardwarepflege auf die Pflege und Reparatur von Rechnern bezieht, ist der Teil der Softwarepflege bei weitem komplexer. Die Funktionalität des Systems muss sichergestellt werden, Patches installiert, Schwachstellen gefunden und beseitigt werden. In Absprache mit dem Anwender sollen Verbesserungen herausgearbeitet werden. Die gegebene Heterogenität der Hard- und Softwaresysteme soll außerdem für die verschiedenen Systeme in Kombination mit den einzelnen Spezialisten transparent gemacht werden.
Unter dem Begriff ***Performanzmanagement*** versteht man die Erstellung einer Ist-Analyse des Systems zur weiteren Optimierung desselben. Im Vordergrund steht dabei die Prüfung, ob die momentane Systemkonfiguration den an sie gestellten Ansprüchen gerecht werden kann. Veränderbare Parameter sind dabei bestimmte Vorverarbeitungsschritte z.B. bei der Datenbeschaffung, bei der auf eine optimale Abfolge von Extraktions- und Aufbewahrungsvorgängen aus verschiedenen Quellsystemen geachtet werden muss. Eine Veränderung solcher Regelgrößen kann großen Einfluss auf die Systemleistung haben.
Ziel der ***Qualitätsüberwachung*** ist es, Engpässe aufzuzeigen. Mit Hilfe der ***Kapazitätsplanung*** können diese im folgenden beseitigt werden. Anfangs wird das Qualitätsmanagement eher beiläufig eingesetzt werden, im Laufe der Zeit sollte man jedoch zu einem globalen, komponentenübergreifenden Qualitätssicherungssystem gelangen. In der Kapazitätsplanung werden bestimmte Daten aus der Qualitätsüberwachung mit anderen Messwerten wie Netzwerkauslastung ausgewertet, um Engpässe festzustellen. Diese Daten können nicht nur für Hard- und Software benutzt werden sondern bspw. auch für die Personalpolitik. Gerade Engpässe in der Datenbeschaffung können somit erkannt und durch Systemaufrüstung oder -optimierung beseitigt werden.
Die Anwenderbetreuung stellt ein weiteres wichtiges Kriterium dar, weil nur mit zufriedenen Anwendern die gewünschten Ergebnisse mit einem Data Warehouse System zu erreichen sind. Kontakt mit dem Anwender zwecks Rückfragen und Problembehebung muss somit sichergestellt werden. Es muss dabei auch zwischen verschiedenen Nutzerprofilen unterschieden werden und die Hilfe kenntnisspezifisch angepasst werden (je nach Anfänger oder Experte).
Das ***Schutz- und Sicherheitsmanagement*** befasst sich zum einen mit der Verteilung von Zugriffberechtigungen für die Nutzer, was für den Betrieb eines großen DW-Systems

unabdingbar ist als auch mit Konzepten, die im Falle eines Systemausfalles anzuwenden sind. Es gibt dabei zwei verschiedene Punkte. Systemtechnische Schutz- und Sicherheitskonzepte sollen im Falle eines Systemschadens dafür sorgen, das andere Teile des Systems nicht in Mitleidenschaft gezogen werden. Datenorientierte Maßnahmen sorgen für die zweckgemäße Zuordnung von Zugriffsrechten zu den Nutzern.
Unter der ***Evolutionskontrolle*** versteht man im allgemeinen die Aufgabe, dafür zu sorgen, dass Weiterentwicklungen des Systems auf eine Art eingeführt werden die den weiteren Betrieb des Systems nicht beeinträchtigen. Aus langfristiger Sicht betrachtet gibt es dazu den Aufgabenbereich der DW-Strategie und –Plattform, die sich zwar auch mit der Weiterentwicklung des Systems beschäftigt, dies aber eher auf langfristiger Basis und mit Hinblick auf ein Nachfolgesystem..

Diese genannten Aufgaben werden nun von verschiedenen Rollen übernommen, die jeweils im Rechenzentrum als auch im DW-Kompetenzzentrum arbeiten. Die Verteilung der einzelnen Aufgaben an diese zwei Zentren ist von einer übergeordneten Stelle zu klären. Das ***Rechenzentrum*** befasst sich mit den klassischen Aufgaben der Installation und des Betriebs der Rechenanlagen bzw. Software. Auch Datenbankadministrationen werden häufig übernommen und die automatisierten Datenbeschaffungsvorgänge überwacht. Demgegenüber hat das ***DW-Kompetenzzentrum*** die Aufgabe, für einen reibungslosen Betrieb des DW zu sorgen. Eine Zusammenarbeit mit dem Rechenzentrum ist dazu oft notwendig. Der Datenmanager ist für den korrekten Fluss der Daten in das Data Warehouse verantwortlich. Er hat technische (Machbarkeitsanalysen) als auch politische Aufgaben (z.B. Abtretungsverhandlungen über Zuständigkeiten) zu bewältigen. Daneben ist der Datenadministrator in Zusammenarbeit mit dem Datenbankadministrator für die Erstellung und Pflege von konzeptionellen Schemata verantwortlich. Er ist auch verantwortlich für die Pflege der Metadaten. Weiterer und letzter Punkt des DW-Kompetenzzentrums ist die Anwenderbetreuung in Form von z.B. Schulungen.
Beide Bereiche können sowohl firmenintern betrieben als auch ausgegliedert werden. Das Rechenzentrum für das DW sollte sich den gegebenen Strukturen des Rechnerbetriebs anpassen. Für das DW-Kompetenzzentrum ist abzuwägen, ob Kostenvorteile mit Firmenpolitik und Spezialisierung kollidieren oder nicht. Im Falle einer Ausgliederung sollte eine genaue Schnittstellendefinition zwischen Firma und Außenwelt definiert werden.

4.2 Iterativer Datenbeschaffungsprozess

Im folgenden soll auf Anforderungen, die der Datenbeschaffungsprozess mit sich bringt, eingegangen werden. Die operativen Daten sollen dabei in die Basisdatenbanken und danach ins Data Warehouse überführt werden. Dieser Vorgang der in viele einzelne Untervorgänge aufzuteilen ist, nimmt ca. 60 – 70 % des Gesamtaufwandes für den Betrieb eines DWS in Anspruch, vor allem der Bereich Qualität. Man unterscheidet zwei Datenbeschaffungsvorgänge: das initiale Laden und iterativ auftretende Aktualisierungen. Die Administration der Datenbeschaffungsprozesse beschäftigt sich mit folgenden Aufgabenbereichen: ***Planung*** (Konfiguration, Fehlerbehandlung, Protokollierung) sollte mit grafischen Tools wie Flussdiagrammen vorgenommen werden. Die ***Prozesssteuerung*** muss für ein Zusammenspiel der einzelnen Prozesse sorgen. Der gesamte Prozess sollte einer ständigen ***Beobachtung*** und ***Protokollierung*** unterworfen werden, um Fehler frühzeitig zu erkennen. Hilfsmittel können wiederum grafische Softwarekomponenten sein. Im Falle von Fehlern ist eine ***Ausnahmebehandlung*** zu aktivieren, die den Betrieb jedoch nicht aufhalten sondern lediglich fehlerhafte Daten zwischenspeichern. Eine Fehlertabelle mit markierten Primärschlüsselattributen ist dabei denkbar. Für den Fehlerfall ist eine Fehlerbehandlung vorzusehen, die das System anhält ohne Daten bzw. Konsistenzen zu zerstören und möglichst schnell und ohne großen Aufwand wieder zum Laufen bringt.

Anwender müssen von Datenbeschaffungsmaßnahmen rechtzeitig informiert werden, um ihre Arbeit gegebenenfalls für den betreffenden Zeitraum umplanen zu können.

Auch der Datenbeschaffungsprozess muss einer gewissen Wartung unterliegen, gerade bei sich ändernden Quelldatenbank oder Datenformaten. Eine Überwachung bezüglich Qualitätssicherung, Engpässen, Redundanzen und Konsistenzen ist somit erforderlich. Mit der Zeit wird es wahrscheinlich dazu kommen, dass immer größere Datenmengen transportiert werden müssen, was die Forderung nach Optimierung der Datenbeschaffung nach sich zieht. Eine Optimierung befasst sich mit Fragen wie der Redundanzvermeidung, der Parallelisierung, der Datenbereinigung, dem Einsatz neuer Werkzeuge, der Skriptanpassung, dem parallelem Laden von Basisdatenbank und Data Warehouse.

4.3 Performanztuningmöglichkeiten eines DWS

Als Performanztuning bezeichnet man die Verbesserung der Antwortzeitcharakteristik eines Softwaresystems. Dieser wird hier als mehrstufiger und hierarchischer Prozess vorgestellt, vom Informationsmanagement bis hin zur rein hardwaretechnischen Betrachtung.
Aus Sicht des ***Informationsmanagements*** können Umstrukturierungsmaßnahmen Systemreserven freisetzen. Schwachpunkte im System müssen dazu erst einmal lokalisiert werden. Dies kann mithilfe des Auditing, der systembezogenen Protokollierung oder mit dem Tracing, der benutzerbezogenen Ablaufverfolgung geschehen. Man muss zwischen subjektiv kritischen und objektiv kritischen Anfragen unterscheiden, um zu sinnvollen Lösungen zu gelangen. Es macht daher keinen Sinn, subjektiv kritische Probleme zu verfolgen, die objektiv unkritisch sind, z.B. wenn die Anfrage nun mal sehr viel Rechenaufwand benötigt. Die Anwendungsumgebung ist sinnvoll zu organisieren. Test- und Produktivsysteme sind voneinander zu trennen. Komplexe Anfragen sollten zu Zeitpunkten mit geringer Systemauslastung gestellt werden bzw. auf örtlich verteilten Systemen ausgeführt werden. Hardwareplattformerweiterungen sind meist günstiger, genaue Untersuchung über ihr Potential sollten jedoch vorher veranlasst werden.
Als nächstes können ***Datenbankschemata denormalisiert*** werden, indem man Redundanzen hinzufügt, die dann zwar mehr Speicher benötigen, die Anfragezeiten aber teilweise drastisch reduzieren. So können z.B. Fremdschlüssel aufgelöst und an ihre Stelle die Attribute der referenzierten Entität geschrieben werden. Auch Aggregatspeicherung, bspw. einer Gesamtsumme kann bei häufiger Wiederverwendung sinnvoll sein. Umfangreiche Tabellen können sowohl horizontal als auch vertikal partitioniert werden. Dabei werden bei der horizontalen Partitionierung einzelne zusammengehörende (semantisch homogene) Tupel einer Mastertabelle in verschiedene Teiltabellen übertragen. Anfragen, die sich auf eine solche zusammengehörende Menge beziehen, gehen danach schneller vonstatten.
Die Vertikale Partitionierung sieht vor, semantisch homogene Attribute in Untertabellen zu speichern und an ihre Stelle im Originaleintrag Fremdschlüssel zu setzen. Unnötige Daten werden danach nicht mehr mittransportiert. Weitere Möglichkeiten seien an dieser Stelle nur erwähnt, das Clustering, Indexdesign und Konfiguration der Tabellenfüllung.
Maßnahmen aus Sicht der Applikationsumgebung bezeichnen alle systemunabhängigen Maßnahmen, zur Leistungssteigerung aus Sicht der Applikationsumgebung. So kann man bestimmte Prozesse parallel ausführen oder Applikationslogik auf den Datenbankserver auslagern. Anfrageergebnisse können anderen Nutzern zur Verfügung gestellt werden, um doppelte Berechnungen zu vermeiden. Beim Schreiben in eine Tabelle muss diese kurz für andere Benutzer gesperrt werden. Ziel muss es aber sein, diese Sperrung nur so kurz wie möglich vorzunehmen. Dies bezeichnet man auch als Locking Strategien. Es gibt hierbei wiederum mehrere Verfahren, nur erwähnt sei hier als Beispiel das „optimistische Schreiben“. Die Wahl der Anzahl der Kontrollpunkte bestimmt wiederum die Systemsicherheit auf Kosten der Performanz.

Weiterhin kann man auf Datenbankebene Leistungsreserven freisetzen. Dabei können verschachtelte Selektionsanfragen bei semantischer Gleichheit so optimiert werden, dass hohe Selektionen gleich anfangs ausgeführt werden um die Menge des Selektionsergebnisses schon anfangs stark zu reduzieren, spätere Selektionen können dann schneller erfolgen. Als Selektivität bezeichnet man das Verhältnis aller möglichen Tupeln zu den selektierten Tupeln. Eine hohe Selektivität reduziert also die Ergebnismenge sehr stark. Das Anlegen von Indizes kann die Reaktionszeit auch erhöhen, aber nicht zwingend. Regelbasierte Optimierer führen eine Vielzahl der hier vorgestellten Verfahren automatisch durch.
Aus Sicht der Datenbankkonfiguration sind wiederum Optimierungsmöglichkeiten auf dem Gebiet der parallelen Verarbeitung, des Speichermanagements als auch mit Hilfe von Verteilungskonzepten möglich.
Maßnahmen aus Sicht des Netzwerks betrifft die Sicherstellung der Funktionalität des Netzwerkes, z.T. mit automatischen Überwachungssystemen. Um Engpässe zu vermeiden, werden Verfahren wie ***Connection Pooling*** sowie ***Connection Multiplexing*** verwendet. Beim Connection Pooling teilen sich die Anwender eine Begrenzte Anzahl von Verbindung aus dem Verbindungspool mit Hilfe eines Verteilungsprozesses. Beim Multiplexing werden viele logische Netzwerkverbindungen zu einer physischen Verbindung zusammengeführt. Auf Hardwareebene lassen sich Optimierungen auf dem Gebiet der Systemprozessoren, Primärspeicher, Cache, Sekundärspeicher als auch bei den I/O-Kontrollern durchführen.

4.4 Probleme für die Anwenderakzeptanz von DW-Systemen

Es werden an dieser Stelle drei Problemgruppen unterschieden: systembedingte, organisationsbedingte, anwenderorientierte Probleme.
Systembedingte Probleme umfassen zumeist Perfomanz- und Datenqualitätsaspekte als auch Softwareprobleme. Eine Lösung ist hier die konsequente Weiterentwicklung des DWS mit Hilfe von Analysewerkzeugen.
Organisatorische Probleme treten oft dann auf, wenn das Management die Stellung der Datenanalyse nicht mit der nötigen Priorität beachtet und Anwender mit ihren Problemen nicht ernst genug unterstützt werden. Eine Anpassung der organisatorischen Abläufe zur Unterstützung des DWS ist somit notwendig.
Anwenderorientierte Probleme können objektiver Natur, wenn der Anwender nicht in der Lage ist, das System zu bedienen, als auch subjektiver Natur sein, wenn Anwender aufgrund eines Inneren Widerstandes nicht die Notwendigkeit erkennen wollen, sich in ein neues System einzuarbeiten. Eine differenzierte Anwenderbetreuung je nach Wissens- und Aufgabengebiet ist damit dringend zu empfehlen und kann in Form von Schulungen, Hotlines oder Ansprechpartnern umgesetzt werden.

4.5 Weitere Aspekte und Komponenten

Repositorium

Gerade bei einer sehr heterogenen Struktur der Softwareplattformen ist ein Repositorium notwendig für eine Effiziente Administration und Wartung. Das Repositorium enthält dabei wichtige Daten über die Struktur des DWS auf Software- Hardware- und Metadatenebene. Besonders technische Metadaten sind hierbei von Interesse. Dies umfasst Metadaten über Primärdaten (Strukturen der Basisdatenbanken und der Quelldatensysteme) als auch Prozessmetadaten wie Regeln für Datenbeschaffungsprozesse (z.B. Reihenfolge). Durch das Repositorium kann auf alle Metadaten über eine einheitliche Schnittstelle zugegriffen und die Steuerung des Datenbeschaffungsprozesses durch zugriff auf Metadaten im Repositorium automatisiert werden.

Softwareänderungen lassen sich systematischer durchführen und nachvollziehen. Auch wenn meist ein zentrales Repositorium aufgrund der Heterogenität der Softwarekomponenten nicht möglich ist, so sollte doch wenigstens eine logische Sicht auf alle Metadaten des DWS existieren.

Entsorgung von Daten

In bestimmten Fällen kann es passieren, dass man Daten entsorgen möchte. Dies kann in Form des unwiderruflichen Löschens passieren als auch mit Hilfe der Archivierung. Gründe können sein, dass die Daten nicht mehr verwendet werden oder von schlechter Qualität sind, bzw. dass das DWS den verwendeten Platz und die Performanz besser nutzen kann. Mit Hilfe von Protokolldateien kann man die Verwendung von Daten gut überprüfen. Auf Konsistenzerhalt in der Datenbank ist unbedingt zu achten. Archivierungsaspekte sind zum einen die Reaktivierungszeit, Zuverlässigkeit in Hinblick auf die Datenlesbarkeit nach der Lagerung, Zugriffserlaubnisse und Archivierungs-/Reaktivierungskosten. Bei der Reaktivierung kann es zu Problemen mit der Software oder Plattformeben kommen oder es ändert sich die Datenstruktur. Teilweise werden deshalb auch Metadaten, und Software zu sichern, manchmal wird das gesamte System inklusive Hardware archiviert, was dann auch als „Schnappschuss total“ bezeichnet wird.

Phasen eines Recoveryplans

In Fällen von Archivierung als auch beim Datenbackup ist ein Plan zu erstellen, wie eine mögliche Datenwiederherstellung von statten gehen könnte. Am Anfang sollte jedoch eine Risikoanalyse stehen (wie wahrscheinlich sind bestimmte Ausfallgründe), gefolgt von Recoveryanforderungen (wie lange soll die Wiederherstellung bei verschiedenen Fehlern dauern). Nun wird ein Plandokument erstellt und anschließend an die betreffenden Stellen verteilt. Dieses Dokument enthält Telefonnummern oder Emailadressen von zu benachrichtigenden Personen, Prioritäten sowie Zuständigkeiten, Hardwareersatzteillieferanten als auch System- und Konfigurationsdaten. Wie oben bereits erwähnt, ist es von Vorteil einen suchen Plan im simulierten Ernstfall einmal zu testen, allerdings nicht als Totalausfall, sondern eher durch den schrittweisen Ausfall einzelner Komponenten. Ergebnisse sind an die betreffenden Personen zu verteilen.

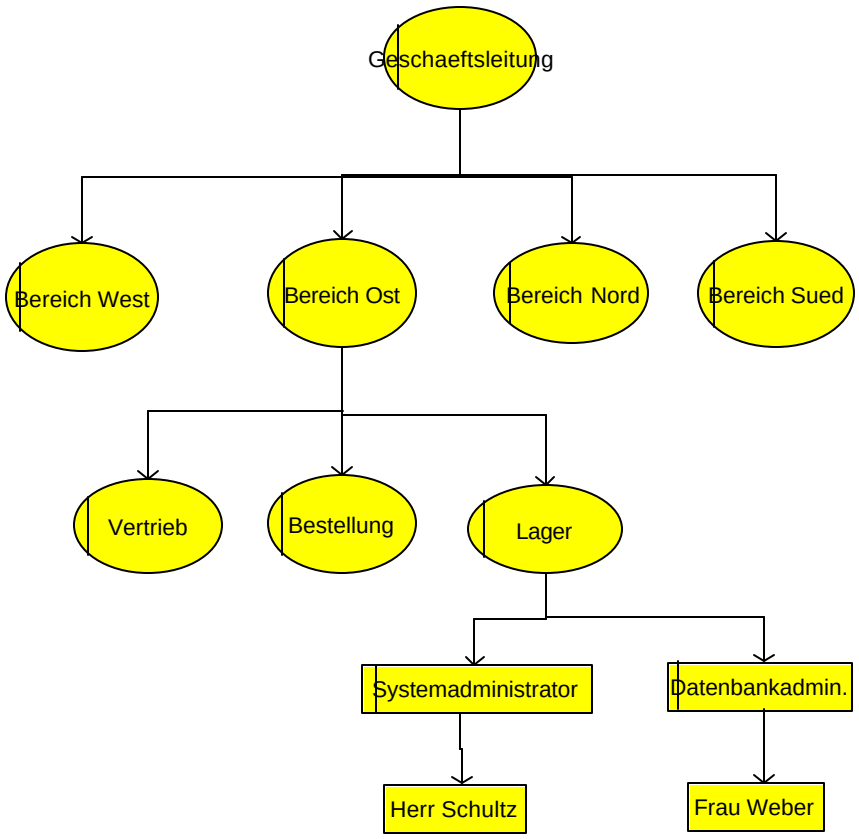
Zusammenfassung: Die Wartung und Pflege eines Data Warehouse Systems ist ein Prozess, der niemals beendet ist. Komponentenübergreifende Administration ist von großer Wichtigkeit. Auch die Pflege des Repositoriums mit all seinen Metadaten sollte eine hohe Priorität besitzen. Die Datensicherung sollte das Rückgrat eines jeden DWS darstellen, gerade in Hinblick auf den großen Entwicklungs- und Pflegeaufwand.

5 Software zur Unterstützung des Projektmanagements

5.1 ARIS-Toolset

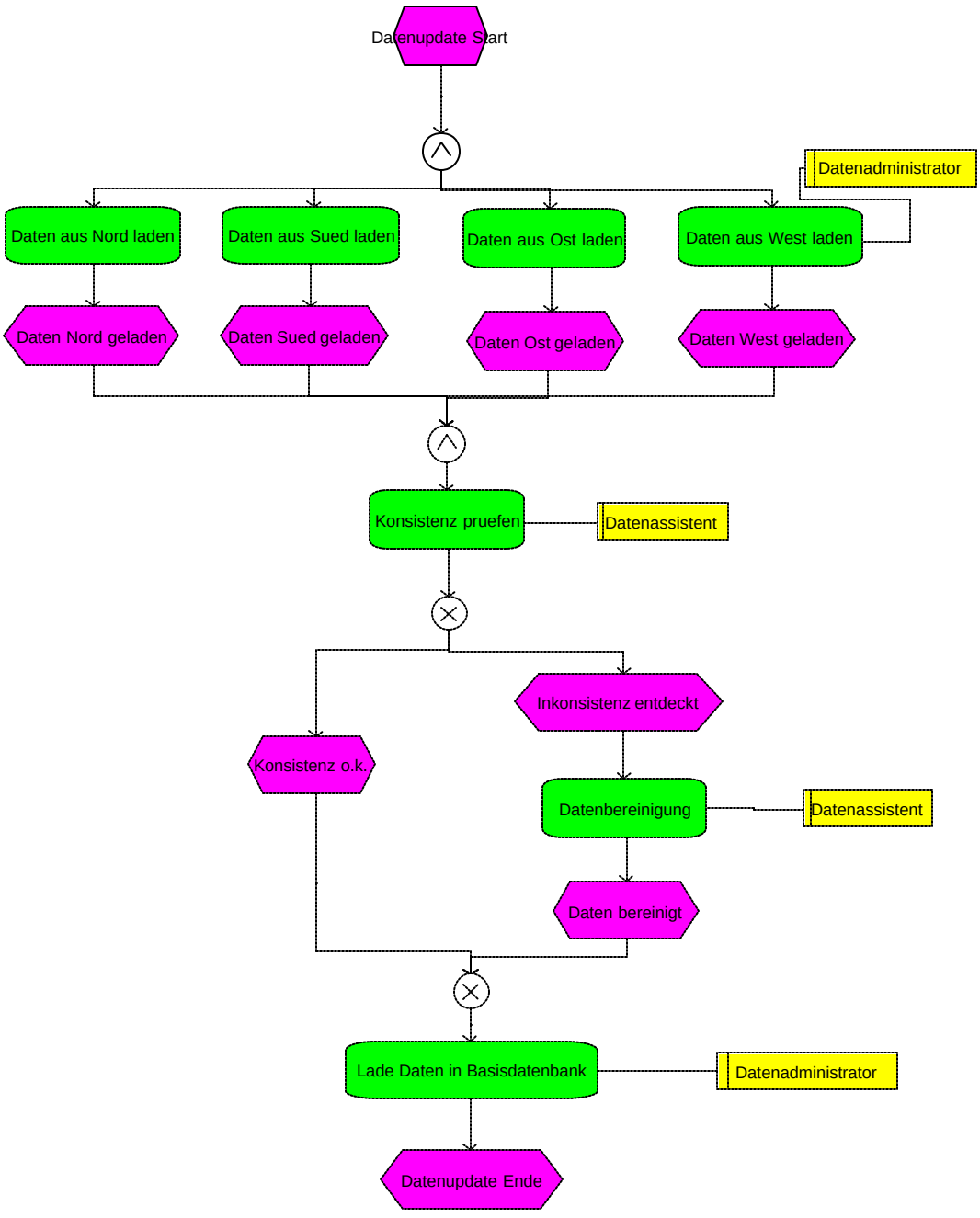
ARIS [5] ist ein Werkzeug von der IDS Scheer AG und wurde zur Modellierung von Geschäftsprozessen entwickelt. Dabei kann sowohl die Struktur/Aufbau eines Unternehmens als auch sein Verhalten modelliert werden. ARIS nutzt dabei verschiedene Sichten: Daten-, Funktions-, Steuerungs- und Organisationssicht. Bei der Erstellung eines Data Warehouse Projektes kann es sehr hilfreich sein, sich die Unternehmensstruktur und das Unternehmensverhalten anhand eines Modells vor Augen zu führen.
Im Data Warehouse Projekt kann man damit bestimmte Datenflüsse erkennen, die erforderlich sind, welche Stellen und Einheiten dafür zusammenarbeiten müssen und wer für welche Bereiche die Verantwortung trägt.

Im folgenden „Organigramm“ wird die Struktur eines Unternehmens modelliert. Das Unternehmen bestehe aus 4 Teilbereichen, die aufgrund ihrer geographischen Lage gebildet wurden. In jedem Teilbereich gibt es die Abteilungen Vertrieb, Bestellung und Lager, ggf. hat jeder dieser Bereiche eine eigene IT-Abteilung mit Administratoren. Diese Modellierung sollte Voraussetzung weiterer Entscheidungen sein.



ARIS – Organigramm

Nicht nur die Struktur sondern auch das Verhalten kann mit einem ARIS-Modell beschrieben werden. Man kann so zum Beispiel bestimmte Abläufe modellieren, die für den Betrieb eines Data Warehouse notwendig sind und durch Teilautomatisierung evtl. effizienter erfolgen können wie z. B. das Update der Daten in der Basisdatenbank.



ARIS – eEPK

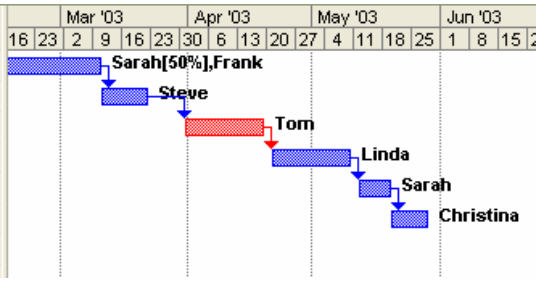
Legende: purpur: Ereignisse; grün: Funktionen; gelb: Stellen

5.2 Microsoft Project

Microsoft Project [6] ist hingegen nicht auf die Geschäftsprozessmodellierung ausgelegt. Hier handelt es sich vielmehr um ein Softwaretool, das zur Organisation von Projekten mit Termin- und Ressourcenplanung bis hin zur Festlegung von Meetings, der Festlegung von Zwischenzielen (Milestones), Deadlines und der Aufteilung des Projektes in Unterziele behilflich sein kann. Einige Grundlegende Funktionalitäten werden hier anhand von Beispielen, die für das Data Warehouse Projektmanagement relevant sind, veranschaulicht.
Nachdem nun die Struktur des Unternehmens mit ARIS modelliert wurde, und die Struktur des Data Warehouses daran angelehnt werden könnte, kann im weiteren Verlauf die Softwareauswahl an der Reihe sein. Diese soll anhand eines Beispieles dargestellt werden.
MS Project erlaubt es, die Arbeitszeiten der Mitarbeiter so zu planen, dass alle Angestellten möglichst voll ausgelastet sind – gleichzeitig aber Überstunden vermieden werden. Auch wird der zeitliche Verlauf grafisch dargestellt. Dabei werden auch die benötigte Ausstattung sowie die anfallenden Kosten berücksichtigt.
Wenn der Projektplan einmal aufgestellt wurde, kann man ihn mit MS Project kontrollieren, was durch Analyse- und Prüfwerkzeuge ermöglicht wird. Notwendige Abänderungen können somit schnell erkannt und flexibel durchgeführt werden.
Mit Hilfe von Sichten (Views) und Tabellen kann genau die Information dargestellt werden, die benötigt wird.
Für die Unterstützung des Kommunikationsprozesses kann man Diagramme entweder ausdrucken oder über HTML auf einem Web Server speichern und somit anderen zugänglich machen. Auch wird der Datenaustausch per Email mit Outlook unterstützt.

Beispiel: Planung der Softwareauswahl mit Hilfe eines „Grantt Charts“

	Task Name	Duration	Start	Finish	Pre	Resource Names
	Market analysis	21 days?	Mon 2/10/03	Mon 3/10/03		Sarah[50%],Frank
	Chose candidates	9 days	Tue 3/11/03	Fri 3/21/03	1	Steve
	Evaluate candidates	15 days	Mon 3/31/03	Fri 4/18/03	2	Tom
	Prototyping	15 days	Mon 4/21/03	Fri 5/9/03	3	Linda
	Evaluate close candidates	6 days	Mon 5/12/03	Mon 5/19/03	4	Sarah
	Chose Product	7 days	Tue 5/20/03	Wed 5/28/03	5	Christina

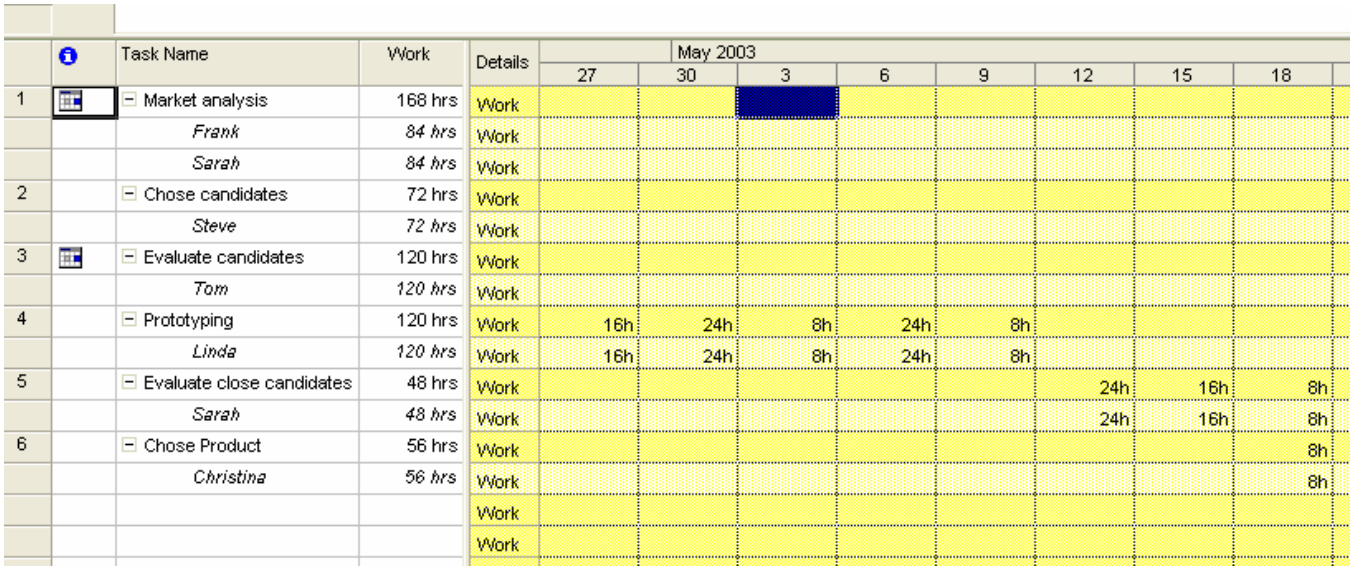


Bilder aus Microsoft Office 2000 [6]

Anhand der Planung und den Lohnkosten kann dann eine Arbeitszeitkostenrechnung durchgeführt werden, mit deren Hilfe die voraussichtlichen Lohnkosten eines Teilprojektes offensichtlich werden.

	Resource Name	Type	Material Label	Initials	Group	Max. Units	Std. Rate	Ovt. Rate	Cost/Use	Accrue At	Base Calendar
	Frank	Work		F		100%	$20.00/hr	$23.00/hr	$0.00	Prorated	Standard
	Steve	Work		S		100%	$21.00/hr	$24.00/hr	$0.00	Prorated	Standard
	Tom	Work		T		100%	$26.00/hr	$29.00/hr	$0.00	Prorated	Standard
	Linda	Work		L		100%	$17.00/hr	$20.00/hr	$0.00	Prorated	Standard
	Sarah	Work		S		100%	$24.00/hr	$27.00/hr	$0.00	Prorated	Standard
	Christina	Work		C		100%	$21.00/hr	$24.00/hr	$0.00	Prorated	Standard

Der Stundenplan könnte wie folgt aussehen:



	Task Name	Work	Details	May 2003							
				27	30	3	6	9	12	15	18
1	Market analysis	168 hrs	Work								
	Frank	*84 hrs*	Work								
	Sarah	*84 hrs*	Work								
2	Chose candidates	72 hrs	Work								
	Steve	*72 hrs*	Work								
3	Evaluate candidates	120 hrs	Work								
	Tom	*120 hrs*	Work								
4	Prototyping	120 hrs	Work	16h	24h	8h	24h	8h			
	Linda	*120 hrs*	Work	16h	24h	8h	24h	8h			
5	Evaluate close candidates	48 hrs	Work						24h	16h	8h
	Sarah	*48 hrs*	Work						24h	16h	8h
6	Chose Product	56 hrs	Work								8h
	Christina	*56 hrs*	Work								8h
			Work								
			Work								

Es gibt zudem noch eine Vielzahl weiterer Features und Planungstools, die hier allerdings nicht alle vorgestellt werden sollen.

MS Project eignet sich nicht nur zum Aufbau des Data Warehouses, sondern auch für die Durchführung der regelmäßig anfallenden Wartungsarbeiten wie Updates, Sicherungsarbeiten. Die Planung der Ressourcen sowie Aufstellung der zeitlichen Abläufe erfolgt dann analog. Die so erstellten Pläne sind jedoch nicht beliebig feingranular gestaltbar, ein gewisser Handlungsspielraum ist notwendig und unvermeidlich um auf Änderungen der Umwelt angemessen reagieren zu können.

Abschließend ist anzumerken, dass die o.g. Tools einem die Planung nicht abnehmen können. Vielmehr können sie den Planungs - und Ausführungsprozess durch Kalkulationen und grafische Tools unterstützen. So werden getroffene Entscheidungen übersichtlich darstellbar, was die Möglichkeit zur Fehlerkorrektur und globalem Verständnis gibt. Außerdem eröffnet sich auch für den einzelnen Mitarbeiter die Struktur der Gesamtplanung, was der Motivation und dem Zusammengehörigkeitsgefühl dient.

Quellen

[1] Bauer, A. and Günzel, H. (eds.), Data-Warehouse-Systeme, dpunkt.verlag, 2001
[2] Agosta, L., The Essential Guide to Data Warehousing, Prentice Hall, 2000
[3] Stella Gatziu, Athanasios Vavouras , Data Warehousing: Concepts and Mechanisms
[4] Stefan Conrad, Kai-Uwe Sattler, Vorlesung Data-Warehouse-Technologien
[5] ARIS Toolset 6.0 (C) IDS Scheer AG
[6] MS Project 2000 (C) Microsoft Corp.
[7] Quick Preview MS Project 2002
[8] Jakubczik und Skubch, Business Process, 1994