

# **Data Warehouse F&T**

Masterarbeit  
von  
Mario Johansson, B.Sc.

**Technische Universität Graz**

Fakultät für Informatik

Institut für Computer Grafik und Vision

Univ.-Prof. Dipl.-Ing. Dr. techn. Horst Bischof

Dipl.-Ing. Dr. techn. Josef Kolbitsch

© Copyright 2012, Mario Johansson  
Diese Arbeit ist in deutscher Sprache verfasst.

Graz, im Oktober 2012

# **Data Warehouse R&T**

Master's Thesis  
Of  
Mario Johansson, B.Sc.

**University of Technology Graz**

Department of Informatics

Institute for Computer Graphics and Vision

Univ.-Prof. Dipl.-Ing. Dr. techn. Horst Bischof

Dipl.-Ing. Dr. techn. Josef Kolbitsch

© Copyright 2012, Mario Johansson

Graz, October 2012

---

## EIDESSTATTLICHE ERKLÄRUNG

Ich erkläre an Eides statt, dass ich die vorliegende Arbeit selbstständig verfasst, andere als die angegebenen Quellen/Hilfsmittel nicht benutzt und die den benutzten Quellen wörtlich und inhaltlich entnommene Stellen als solche kenntlich gemacht habe.

Graz, am .....

.....

(Unterschrift)

## STATUTORY DECLARATION

I declare that I have authored this thesis independently, that I have not used other than the declared sources / resources, and that I have explicitly marked all material which has been quoted either literally or by content from the used sources.

.....

date

.....

(signature)

## **Danksagung**

An dieser Stelle möchte ich mich bei Herrn Dipl.-Ing. Dr. techn. Josef Kolbitsch herzlichst bedanken, der mir diese interessante Diplomarbeit möglich gemacht hat.

Ich möchte mich ebenfalls beim gesamten Team der Abteilung Business Solutions des Zentralen Informatikdienstes der TU Graz für die gute Zusammenarbeit bedanken.

In weiterer Folge bedanke ich mich bei meiner Mutter Dipl.-Ing. Barbara Johansson, die mir mein Studium an der TU Graz ermöglicht hat und mich während meiner gesamten Studienzeit tatkräftig unterstützt hat. Auch meinen Großeltern Dipl.-Ing. Horst und Gerlinde Lackner, sowie meinen Brüdern Markus und Timo möchte ich hiermit meinen Dank aussprechen.

Bei allen Bezeichnungen, die auf Personen bezogen sind, meint die Formulierung beide Geschlechter, unabhängig von der in der Formulierung verwendeten konkreten geschlechtsspezifischen Bezeichnung.

## Kurzfassung

Für das Berichtswesen an der TU Graz gibt es seit einigen Jahren das Produkt Oracle Discoverer. Da dieses System stark veraltet ist, ist es sinnvoll anzudenken neue Technologien zur Erstellung der benötigten Informationen heranzuziehen. Die Vorteile bringen einen besseren Support, sowie eine Vielzahl an neuen Möglichkeiten mit sich. Das Forschungs- & Technologie (F&T)-Haus ist ein Ersteller solcher analysebezogenen Abfragen. Da es angestrebt wird das alte Reporting-Werkzeug zu modernisieren, wird von der Fachabteilung eine neue Lösung für die Berichterstellung verlangt.

Anhand von zwei verschiedenen Ansätzen wird versucht eine für das F&T-Haus akzeptable Lösung zu finden, um weiterhin eine konsequente und qualitätssichere Berichtserstellung zu gewährleisten. Einerseits wird ein relationaler Ansatz implementiert, der stark dem Ursprungssystem ähnelt. Auf der anderen Seite wird versucht, mit einer zukunftssicheren Variante eines Data Warehouse die Anforderungen der Fachabteilung zu erfüllen.

Ziel der Arbeit ist es die beiden unterschiedlichen Ansätze zu implementieren und anschließend zu evaluieren. Anhand der Ergebnisse soll eine Empfehlung für das F&T-Haus getroffen werden, welches System die Arbeit des Oracle Discoverer schlussendlich weiterführen soll.

## **Abstract**

A product of the Oracle Company called Discoverer is used by the TU Graz for some years in terms of reporting. Since this product has become outdated, it is meaningful to start thinking of new technologies for creating the desired information. Advantages would be a better support and a handful of new features and opportunities. The Research & Technology House (R&T House) is a key user of such analysis-referred inquiries. Since it is aimed that the old reporting tool will be abolished, the department needs a new solution for creating reports.

According to the two different approaches it is tried to find a solution, which is acceptable for the R&T House, in order to ensure a further consistent and quality-safe report production. On the one hand a relational approach is implemented which resembles the original system. On the other hand it is tried to handle the specifications of the R&T House with a more future-proof approach of a Data Warehouse.

Goal of this thesis is to implement the two different solutions and to evaluate them afterwards. According to the results there should be a recommendation for the R&T House which system should replace the Oracle Discoverer.

# Inhaltsverzeichnis

|          |   |           |
|----------|---|-----------|
| <b>1</b> | <b>Einleitung .....</b>                                       | <b>1</b>  |
| 1.1      | Motivation .....  | 1         |
| 1.2      | Aufgabenstellung .....  | 1         |
| 1.3      | Aufbau und Gliederung der Arbeit .....                        | 2         |
| <b>2</b> | <b>Grundlagen.....</b>  | <b>3</b>  |
| 2.1      | Technische Grundlagen .....                                   | 3         |
| 2.1.1    | Operative Systeme .....                                       | 3         |
| 2.1.2    | Analytische Systeme.....                                      | 4         |
| 2.1.3    | Vergleich zwischen OLTP und OLAP.....                         | 7         |
| 2.2      | Data Warehouse .....  | 8         |
| 2.2.1    | Allgemein .....   | 8         |
| 2.2.2    | ETL Prozess .....   | 12        |
| 2.3      | Grundstruktur eines Data Warehouse .....                      | 17        |
| 2.3.1    | Star Schema .....   | 17        |
| 2.3.2    | Snow Flake Schema .....                                       | 19        |
| 2.4      | Reporting .....   | 20        |
| 2.5      | Ansatzarten von OLAP .....                                    | 24        |
| 2.6      | Modellierung eines Datawarehouse .....                        | 25        |
| 2.6.1    | Relationaler Modellierungsansatz .....                        | 25        |
| 2.6.2    | Dimensionaler Modellierungsansatz.....                        | 28        |
| <b>3</b> | <b>Ausgangssituation – Bestandsaufnahme .....</b>             | <b>31</b> |
| 3.1      | Historie .....  | 31        |
| 3.2      | Forderungen und Gebiete des F&T-Hauses.....                   | 32        |
| 3.2.1    | Abschlussarbeiten.....  | 32        |
| 3.2.2    | Technologietransfer .....                                     | 33        |
| 3.2.3    | Anforderungen .....   | 33        |
| 3.3      | Oracle Discoverer .....                                       | 35        |
| 3.4      | Oracle Warehouse Builder .....                                | 38        |
| 3.5      | Oracle Business Intelligence Enterprise Edition (OBIEE) ..... | 41        |
| 3.6      | Vorhandene Daten.....   | 42        |
| 3.6.1    | Abschlussarbeiten.....  | 43        |
| 3.6.2    | Technologietransfer .....                                     | 45        |

|          |  |           |
|----------|--|-----------|
| 3.7      | Vorhandene Abfragen.....   | 46        |
| 3.7.1    | Abschlussarbeiten - gr_20111125_Abschlussarbeit_extOE_alle.....            | 46        |
| 3.7.2    | Technologietransfer – gr_20111125_alle_GG_Koop_Name_Adresse_Laufzeit<br>48 |           |
| <b>4</b> | <b>Durchführung und Evaluierung der Lösungsansätze .....</b>               | <b>51</b> |
| 4.1      | Implementierung des relationalen Ansatzes.....                             | 51        |
| 4.1.1    | Importieren der Daten in das Data Warehouse .....                          | 52        |
| 4.1.2    | Aufbereiten der Daten mit OBIEE .....                                      | 56        |
| 4.2      | Implementierung des multidimensionalen Ansatzes.....                       | 60        |
| 4.2.1    | Aufbereitung der Daten für die Core Ebene .....                            | 60        |
| 4.2.2    | Besonderheit: Faktentabellen .....   | 64        |
| 4.2.3    | Aufbereitung der Daten für die Mart Ebene.....                             | 65        |
| 4.3      | Evaluierung.....   | 67        |
| 4.3.1    | User Interface .....   | 67        |
| 4.3.2    | Verständlichkeit und Lernfaktor.....                                       | 69        |
| 4.3.3    | Performance .....  | 69        |
| 4.3.4    | Zeitfaktor.....  | 69        |
| 4.3.5    | Funktionalität .....   | 70        |
| 4.3.6    | Zukunftsorientiertheit .....   | 70        |
| 4.3.7    | Wartbarkeit .....  | 70        |
| 4.3.8    | Implementierung .....  | 71        |
| 4.3.9    | Zusammenfassung .....  | 71        |
| <b>5</b> | <b>Empfehlung – Conclusio .....</b>  | <b>72</b> |
| 5.1.1    | Rückblick der Aufgabenstellung.....  | 72        |
| 5.1.2    | Lessons Learned .....  | 73        |
| 5.1.3    | Future Work.....   | 75        |
| 5.1.4    | Empfehlung.....  | 76        |
| 5.1.5    | Conclusio.....   | 78        |
|          | <b>Literaturverzeichnis .....</b>  | <b>79</b> |
|          | <b>Abbildungsverzeichnis .....</b>   | <b>82</b> |
|          | <b>Tabellenverzeichnis .....</b>   | <b>84</b> |
|          | <b>Abkürzungsverzeichnis .....</b>   | <b>85</b> |
|          | <b>Anhang.....</b>   | <b>86</b> |

# 1 Einleitung

## 1.1 Motivation

Durch zunehmende Datenmengen und Datenflüsse sind in Unternehmungen neue Methoden und Wege notwendig, um systematisch mit dieser Sachlage umzugehen. Zur Modernisierung gibt es durch den stetigen Fortschritt im Bereich der Technik und der Software immer neue Lösungsansätze. Damit werden die bestehenden Quellsysteme verbessert und optimiert.

Für Anwendungen im Bereich Jahresabschluss oder Bilanzen werden in der Regel Werkzeuge zur Berichtserstellung von den Unternehmen verwendet. Für eine schnellere und effektivere Verarbeitung werden vermehrt Datenbanksysteme eingesetzt. Diese scheitern sehr oft an den Aspekten der Zukunftsorientiertheit, oder der Erweiterbarkeit.

Deswegen werden neue Ansätze geschaffen. Einer dieser Gestaltungswege ist der Data-Warehouse (DWH) Ansatz. Durch spezielle Methoden versucht man die Verfügbarkeit, Funktionalität und die allgemeine Datenbasis zu verbessern. Dadurch lässt sich das Berichtswesen flexibler gestalten. Somit werden alte Quellsysteme in naher Zukunft modernisiert und für Unternehmen in Bezug auf Wartbarkeit und Erweiterbarkeit interessanter.

## 1.2 Aufgabenstellung

Ziel dieser Masterarbeit ist es, die bereits bestehende Reporting Lösung des Forschungs- & Technologie (F&T) Hauses der TU Graz mittels eines technologisch neueren Ansatz zu modernisieren. Anhand verschiedener Ansätze und eines abschließenden Vergleichs, soll es möglich sein eine Empfehlung für eine Lösung auszusprechen.

Das aktuelle System beruht auf dem Oracle Discoverer Plus. Dieses Produkt der Oracle Corporation wird firmenseitig nicht mehr unterstützt. Deshalb wird für den neuen Ansatz das Nachfolgeprodukt Oracle Business Intelligence (OBI) verwendet. Die neue Lösung ist bereits in der Abteilung Business Solutions des Zentralen Informatikdienstes (ZID) im Einsatz und alltagserprobt. Hinsichtlich des IST Zustands werden nun zwei Wege zur Erreichung einer Modernisierung getestet.

Einerseits wird versucht eine eins zu eins Kopie der Datenbankschemata im OBI zu implementieren. Dadurch wird getestet, ob durch eine reine Umstellung der Quelldaten bereits eine bessere Lösung erzielt werden kann.

Auf der anderen Seite wird das Quellsystem mit Hilfe eines DWH Ansatzes neu aufgebaut. Somit kann man in weiterer Folge auch die besonderen Funktionen von DWHs nutzen.

Nach der Pilotimplementierung werden beide Lösungsansätze miteinander verglichen. Durch unterschiedliche Gewichtung der Vergleichsaspekte ist es am Ende möglich eine klare Empfehlung für eine Lösung zu geben.

### **1.3 Aufbau und Gliederung der Arbeit**

Zunächst wird in Kapitel **2** auf die Grundlagen der eigentlichen Thematik eingegangen. Es findet eine Begriffsabgrenzung zwischen operativen und analytischen Systemen statt und spezielle Online Analytical Processing (OLAP) Technologien, insbesondere DWH, werden hier genau erklärt.

In Kapitel **3** wird eine Analyse der Ausgangssituation sowie eine Bestandsaufnahme des IST Zustandes durchgeführt, um die Komplexität der Problemstellung zu erfassen.

Danach wird in Kapitel **4** ein Vergleich der beiden Ansätze durchgeführt, um aufzuzeigen, wo die Stärken und Schwächen der jeweiligen Lösungen liegen.

Abschließend wird in Kapitel **5** ein Resümee über die Evaluierung der Ansätze gezogen und eine Empfehlung für die Implementierung gegeben.

Anschließend findet man die Verzeichnisse für Literatur, Abbildungen, Tabellen sowie Abkürzungen. Im Anhang befinden sich die vom F&T Haus benötigten Namenskonventionen.

## 2 Grundlagen

In diesem Kapitel wird der Unterschied zwischen operativen und analytischen Systemen erklärt. Danach werden spezielle Technologien von OLAP vorgestellt und erörtert. Zum Schluss wird auf die zwei ausgewählten Ansätze, zur Modellierung des SOLL Zustandes des Quellsystems, eingegangen.

### 2.1 Technische Grundlagen

Wenn man von Datenbanksystemen und Customer Relationship Management- (CRM-) Systemen spricht, hört man immer wieder die Begriffe analytische und operative Systeme bzw. OLTP (Online Transaction Processing) und OLAP (Online Analytical Processing). In den folgenden Kapiteln werden die technischen Grundlagen für das Verständnis von Transaktionsabläufen gegeben. Während in OLTP Verarbeitungsschritte eingesetzt werden, die Lesen als auch Schreiben können müssen, werden mittels OLAP aus den verarbeiteten Daten bereits Ergebnisse generiert. Bei OLTP, welches vorwiegend im operationalen Bereich Gebrauch findet, werden die Berechnung sofort durchgeführt, was in einer nahezu Echtzeitbereitstellung der Daten resultiert. OLAP wertet die Gemeinsamkeiten zwischen verschiedenen Datensätzen aus und leitet weitere Analyse-Schritte ein [Kal96 et al.].

#### 2.1.1 Operative Systeme

Bei Informationssystemen wie zum Beispiel SAP (Systeme, Anwendungen und Produkte in der Datenverarbeitung) wird transaktionsorientiert gearbeitet, da die Aufgabe die Abarbeitung von einigen Millionen Berechnungen ist. Es müssen Lese- und Schreibzugriffe auf die Datenbank in die richtige Abfolge gebracht werden. Dies ist ein zentrales Problem der meisten Anwendungen. Lösung dafür sind die sogenannten Transaktionen. Diese fassen einen Block von Datenbankoperationen zu einer kleineren atomaren Einheit zusammen. Bei Erfolg werden somit alle Anweisungen ausgeführt und bei einem Fehler alle verworfen.

Ein weiteres Problem ist die Sicherung der Konsistenz der Daten. Gleiche Daten müssen systemweit gleich bleiben. Wenn beispielsweise eine Eingabe geändert wird, muss sie an jeder anderen Stelle an der sie abgespeichert ist, ebenfalls geändert werden.

Daher entspricht die Organisation dieser Daten dem relationalen Modell. Dieses bringt eine hohe Performance beim Schreiben und Updaten mit sich. Des Weiteren muss die Integrität der Daten hier sichergestellt sein, welche durch den Rollback Mechanismus in den Datenbanksystemen gewährleistet wird. Solche OLTP Systeme sind aus vielen Tabellen aufgebaut, welche oft in dritter Normalform in den Systemen vorliegen [Kur99].

Bei DWH erfolgt ein schreibender Zugriff jedoch im Idealfall nur zu festgelegten Zeitpunkten. Damit stellt sich hier das Problem des konkurrierenden Zugriffes nicht.

Die Transaktionen bei OLTP als auch die Vorgänge sind für die eigentlichen Daten, welche ins DWH gespeichert werden, verantwortlich, wobei im Gegensatz zu OLAP Systemen nur der Zeitpunkt der Datenbankoperationen gespeichert wird und dem Datensatz als zusätzlicher Informationsgehalt hinzugefügt wird [Rud98].

## 2.1.2 Analytische Systeme

Unter OLAP werden Technologien zusammengefasst, welche die Ad-hoc-Analyse von multidimensionaler Information unterstützen. OLAP Systeme sollen in der Lage sein, zwischen Datensätzen Verbindungen zu finden, welche bei erstem Betrachten völlig unabhängig voneinander erscheinen [Kur99]. Auch das Berichtswesen hat sich der Zeit gemäß weiterentwickelt und daher reichen heutzutage rein statische OLTP generierte Berichte nicht mehr aus. Deshalb wurden eigene Datenbanken geschaffen, deren einziger Sinn die Analyse der Daten ist und damit OLAP Technologien möglich machen. Wohingegen bei OLTP Lese- und Schreibzugriffe notwendig sind, um die Verwaltung von aktuellen Daten zu gewährleisten, sind bei OLAP hauptsächlich Lesezugriffe notwendig. Die Daten für ein solches System sollten aggregiert vorliegen, da hiermit ein rasches Lesen sichergestellt wird. Daher muss als Datengrundlage für ein OLAP System stets ein DWH eingeplant werden um den vorherigen Punkt sicherzustellen.

Für eine deutliche Abgrenzung zwischen OLAP und OLTP stellt der Datenbanktheoretiker Codd 12 Regeln für die Anforderungen an ein OLAP System auf [Codd98].

1. Kontinuierliche Leistungsfähigkeit  
Eine schnelle flexible Leistung muss jederzeit vom DWH System gebracht werden.
2. Dynamische Speicherverwaltung dünnbesetzter Matrizen  
Falls es zu sehr sparsam besetzten Feldern in den einzelnen Spalten kommt, muss ein Weg gefunden werden, auch hier effizient Berechnungen durchzuführen.
3. Flexibles Berichtswesen  
In Bezug auf die Anwender sollte das Berichtswesen intuitiv und flexibel gestaltbar sein. Von einzelnen Snapshots der Datenbasis müssen ebenfalls brauchbare Berichte gebildet werden können.
4. Mehrbenutzer-Funktionalität  
Es müssen mehrere Benutzer gleichzeitig auf die Datenbasis zugreifen können.
5. Unbegrenzte Dimensionen und Aggregationsebenen  
Es soll keine Limitation bei der Erstellung von Dimensionen und Aggregationen geben.

6. Client-Server-Architektur  
In OLAP Architekturen muss eine Client Server Architektur verwendet werden, um die Basisdaten sinnvoll aufzuteilen.
7. Einheitlich strukturierte Dimensionen  
Es werden konforme Dimensionen angestrebt. Das bedeutet, dass diese gleich aufgebaut sein müssen. Dies erleichtert den Umgang mit den Werkzeugen und einen leichteren Datenzugriff.
8. Unbeschränkte dimensionsübergreifende Operationen  
Die Architektur muss mit unendlich vielen Dimensionen umgehen können.
9. Intuitive Datenanalyse  
Für ein Browsing in der Datenbank soll es mehrere Techniken zur Navigation geben.
10. Multidimensionale Sicht der Daten  
Eine gewisse multidimensionale Sicht muss auf die Daten möglich gemacht werden. Somit werden auch teils sehr individuelle Abfragen möglich.
11. Transparenz  
Die Grundfunktionalität darf für den Verwender nicht einsehbar sein. Das verhindert eine zu genaue Auseinandersetzung mit den Konzepten der Architektur.
12. Zugriffsmöglichkeiten  
Es muss zu jedem Zeitpunkt auf die für eine Abfrage relevanten Daten zugegriffen werden können.

Diese oben genannten Regeln hängen jedoch sehr stark vom jeweiligen Anwendungsgebiet ab. Deswegen haben einige Firmen aus der IT- und Beratungsbranche aufbauende Regeln aufgestellt [Cha98]. Damit gibt es zurzeit in etwa 50 Regeln, wobei unter dem Begriff FASMI von Pendse und Creeth eine zurzeit gültige Aussage spezifiziert wurde [Pen95].

1. Fast  
Es wird eine maximale Laufzeit von zwei Sekunden bei einfachen und bis zu 20 Sekunden bei sehr komplexen Abfragen erwartet.
2. Analysis  
Es müssen ad-hoc bei analytischen Abfragen Berechnungen mathematischer Form gebildet werden. Es muss ein Pool gewisser Formeln vorhanden sein, sodass ohne tiefere Programmierkenntnisse solche Abfragen vom Anwender durchgeführt werden können.

### 3. Shared

Es sollen die gängigen Sicherheitsanforderungen in Bezug auf ein DBMS eingehalten werden. Bei diversen schreibenden OLAP Funktionalitäten sollen auch Methoden wie zum Beispiel Query-processing implementiert werden.

### 4. Multidimensional

Dies ist eine grundlegende Anforderung an alle OLAP Produkte. Die Daten müssen in multidimensionaler Form vorliegen. Hauptfokus ist auf eine volle Kompatibilität der Hierarchie der Dimensionen zu legen.

### 5. Information

Es muss vom Produkt eine große Datenmenge gespeichert werden können, wobei hier nicht auf mögliche Ressourcenbelegung eingegangen wird. Es muss für den Anwender möglich sein alle relevanten Informationen aus dem DBMS zu beziehen.

Multidimensionale Datenanalyse ist eine weitere für OLAP gängige Bezeichnung. Dies beruht darauf, dass es OLAP möglich ist, Daten in mehr als 2 Dimensionen zu betrachten. Durch diese multidimensionale Struktur entstehen logisch unabhängige Dimensionen. Das Resultat ist ein Datenwürfel oder auch Hypercube, der beliebig viele Dimensionen besitzen kann. Da dahinter eine passende Datenstruktur liegen muss, gibt es mehrere Arten um eine Datenbank für OLAP zu erstellen. Die drei gängigsten und bekanntesten Ansätze sind ROLAP, MOLAP und HOLAP. In weiterer Folge gibt es noch den DOLAP und den FOLAP Ansatz. Durch die geringe Verbreitung der beiden letztgenannten Ansätze kommen diese aber nicht so stark zur Geltung und fallen im Großen und Ganzen nicht ins Gewicht.

### 2.1.3 Vergleich zwischen OLTP und OLAP

Tabelle 2.1 stellt die Hauptunterschiede zwischen OLTP und OLAP aufbereitet dar. [Kur99]

|                        | OLAP  | OLTP  |
|------------------------|---|---|
| Funktion               | Unterstützt Prozess der Entscheidungsfindung        | Datenverarbeitung basierend auf einem zuverlässigen Transaktionskonzept |
| Daten                  | Analytisch  | Operativ, aktuell, flüchtig   |
| Plattformen            | Verschiedene DB und OS                              | Heterogene Systemlandschaften   |
| Datenbank Design       | Objektorientiert                                    | Prozess- bzw. Transaktionsorientiert                                    |
| Datenbank Schema       | Multidimensional                                    | normalisiert  |
| Organisation der Daten | Anhand vorab definierter Dimensionen und Kennzahlen | hierarchisch  |
| Anschaffungskosten     | Minimal – moderat                                   | Moderat - teuer   |
| Amortisation           | Kurzfristig   | Langfristig   |
| Aktualität der Daten   | Nur historisch verdichtete Daten                    | Nur aktuelle Daten  |
| Datenvolumen           | Gigabyte bis mehrere Terabyte                       | Gigabyte  |
| Datenhaltung           | Auf unterschiedlichen Systemen gehalten             | Daten in unterschiedlichen Systemen verteilt                            |

Tabelle 2.1: Auszug der Unterschiede zwischen OLAP – OLTP [Kur99]

## 2.2 Data Warehouse

Der Bereich Data Warehouse wird in folgenden Sektionen näher erläutert. Daraufhin wird näher auf den Extract, Transform, Load Prozess (ETL) eingegangen.

### 2.2.1 Allgemein

Durch Einführung eines Data Warehouse wird durch Informationstechnologien ein besserer Support für Geschäftsprozesse erzielt, da die bereits zur Unterstützung eingesetzten Technologien oft nicht ausreichen, beziehungsweise teilweise veraltet sind. Durch operative Informationssysteme werden täglich anfallende Daten aus den Business Prozessen verarbeitet. Da diese Systeme jedoch sehr oft eine Datenerhaltung benutzen, ist es sehr schwer eine integrierte Sicht auf die Daten zu werfen. Als Abhilfe dafür und für eine permanente und schnelle Verfügbarkeit der Daten dieser Prozesse, sollte man diese Daten in einem Data Warehouse, wobei es sich um analytische Informationssysteme handelt, speichern, damit man später gewisse Analysetätigkeiten durchführen kann. In Abbildung 2.1 kann man den grundlegenden Aufbau eines analytischen Systems, angelehnt an Chamoni und Gluchowski, sehen. [Cham98]

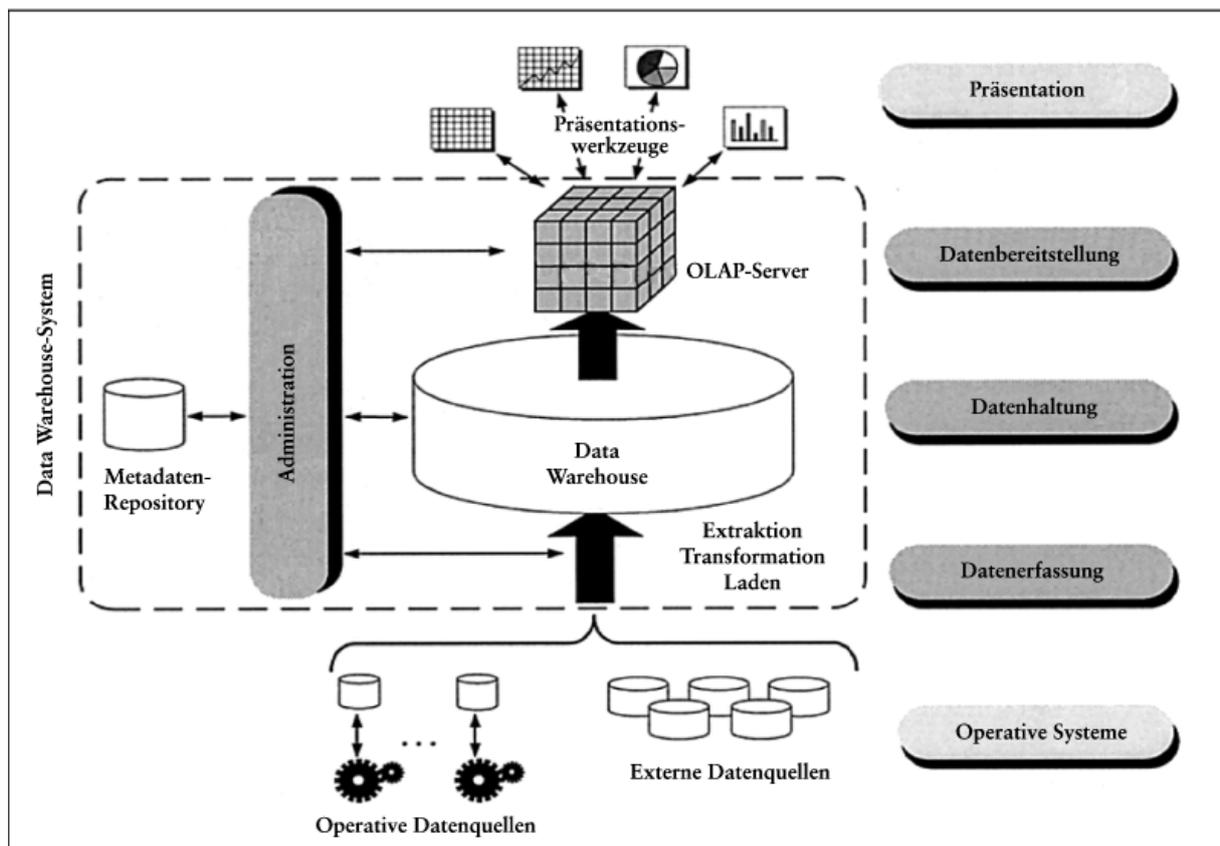


Abbildung 2.1: Informationssystem(analytisch) nach Chamoni und Gluchowski [Cham98]

Jedoch gibt es in der Literatur keine als richtig angesehene Definition des Begriffs Data Warehouse. Man verweist oft auf die meist gebräuchlichste und noch dazu eine der ersten Begriffsabgrenzungen:

„A data warehouse is a subject-oriented, integrated, nonvolatile, and time-variant collection of data in support of management’s decisions.” [Inm94, S. 34]

Bill Inmon, dem diese oben genannte Aussage zugrunde liegt, stellt vier zentrale Bedingungen an ein Datawarehouse, die im Folgenden kurz erläutert werden:

- **Time-variant (Zeitorientierung):**  
Veränderungen zeitlicher Ebene und andere Entwicklungen können mittels eines Data Warehouse durch Analysen sichtbar gemacht werden. Als Schlussfolgerung für die Einführung der Zeit-Dimension muss man die Tatsache sehen, dass man die Daten um diese auch qualitativ auswerten zu können über mehrere Jahre zwischenspeichern muss.
- **Non-volatile (Beständigkeit):**  
Die Daten, die durch Laden aus operativen Systemen geholt werden, werden lediglich ins Data Warehouse eingefügt, aber dauerhaft gespeichert. Die Verwender des DWH haben somit lediglich einen lesenden Zugriff auf diesen Datenbestand.
- **Integrated (Vereinheitlichung):**  
Die Daten, die sich in einem DWH befinden, liegen in integrierter und bereits normierter Form vor.
- **Subject-oriented (Themenorientierung):**  
Anstelle von Anwendungen wird Support anhand von Analyse und Prozessen zur Verfügung gestellt, wobei die Datensegmente nach den Themen einer Unternehmung strukturiert sind.

Die Anforderungen, die Inmon an ein Data Warehouse stellt, besinnen sich im Grunde lediglich nur auf die Datenbasis, wie Wieken anmerkt. Aus dem Betrachtungswinkel der Entwickler und Fachinformatiker sind allgemeine Anforderungen im Allgemeinen viel größer und spezifischer [Wie99, S. 16f]. Auch nach Günzel und Bauer ist die Aussage von Inmon sehr eingeschränkt nutzbar, da sie von Grund aus Gebiete der Anwendung und Vorgehensweisen verbietet [Bau04]. Die technische Durchsetzbarkeit, die oft sehr schwer ist, kann mit Hilfe eines speziellen Konzeptes, des Data Marts, implementiert werden [Bau04, S. 59ff].

Im Grunde genommen handelt es sich bei einem Data Mart um eine Subgruppe innerhalb eines Data Warehouse. Somit wird ein Teil aus dem gesamten Gebilde herausgenommen und in einer eigenen Datenbank gespeichert, wobei man hier berücksichtigen muss, alle

relevanten Daten für eine bestimmte Abfrage mitzuspeichern. Mittels solcher Data Marts kann man Verbesserungen in Bezug auf die Laufzeit der Abfragen erzielen, da hier mit einer weit kleineren Datenmenge gearbeitet wird [Ana97, S 70ff.].

Nachfolgend sind die elementaren Vorteile dieser Methode von Bauer und Günzel [Bau04, S.60] zusammengefasst:

- Datenschutz durch Subdaten
- Drastische Verringerung des anfallenden Datenvolumens
- Aufteilung bezüglich der Last
- Unabhängigkeit in Bezug auf Update Vorgänge im Data Warehouse
- Performance Gewinn durch Aggregationen der Daten

Der Grundgedanke liegt darin, das Datenmodell zu vereinfachen sowie das Volumen der anfallenden Daten zu reduzieren. Man unterscheidet bei der Modellierung einerseits zwischen abhängigen Data Marts, und auf der anderen Seite zwischen unabhängigen.

Wenn eine Datenbasis zur Verfügung steht, bilden sich so genannte abhängige Data Marts, welche einen kleinen Ausschnitt eines Data Warehouse darstellen, und welche bereits aggregiert sein können. Da es sich hierbei um Momentaufnahmen von den Realtime Daten handelt, stimmen diese jederzeit in Struktur als auch im Inhalt mit den Daten des Data Warehouse überein. Zur Erstellung von Data Marts kann man einige jeweils produktspezifische Vorgänge nutzen, welche das Erstellen für den Anwender sehr einfach machen. Graphisch beschrieben wird der Ablauf der Bildung von abhängigen Data Marts in Abbildung 2.2, in Anlehnung an Bauer und Künzel [Bau04, S. 61].

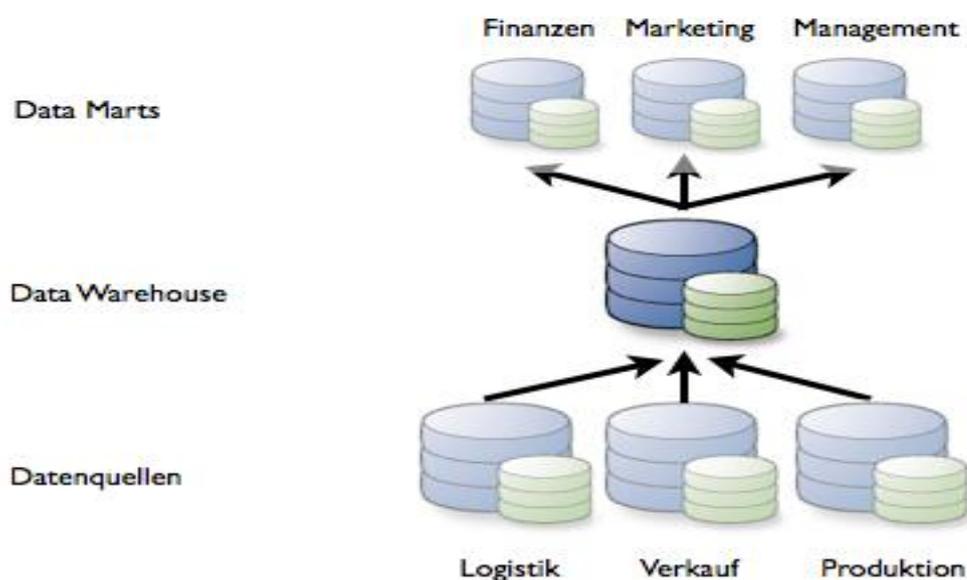


Abbildung 2.2: Architektur von abhängigen Data Marts [Bau04, S. 61]

Falls keine allgemeine Datenbasis zur Verfügung steht, entstehen sogenannte unabhängige Data Marts, wobei diese Daten aus verschiedenen Data Warehouse Systemen stammen. Historisch gesehen hat dies den Sinn beispielsweise jeder Tochter einer Unternehmung ein eigenes Data Warehouse zur Verfügung zu stellen, damit diese recht schnell gute und brauchbare Ergebnisse im Reporting liefern können. Falls die Komplexität der Abfragen jedoch ansteigt und konzernübergreifende Informationen gewünscht werden, wird es mit diesem Konzept etwas schwierig. Jedoch kann durch einige Regeln der Transformation eine übergreifende Abfrage kalkuliert werden [Bau04, S. 62ff]. Diese unabhängigen Data Marts liegen als genormte Datenquellen vor dem eigentlichen Datawarehouse. In Abbildung 2.3 wird nun die Architektur von solch unabhängigen Data Marts in Anlehnung an Bauer und Künzel [Bau04, S. 63] dargestellt.

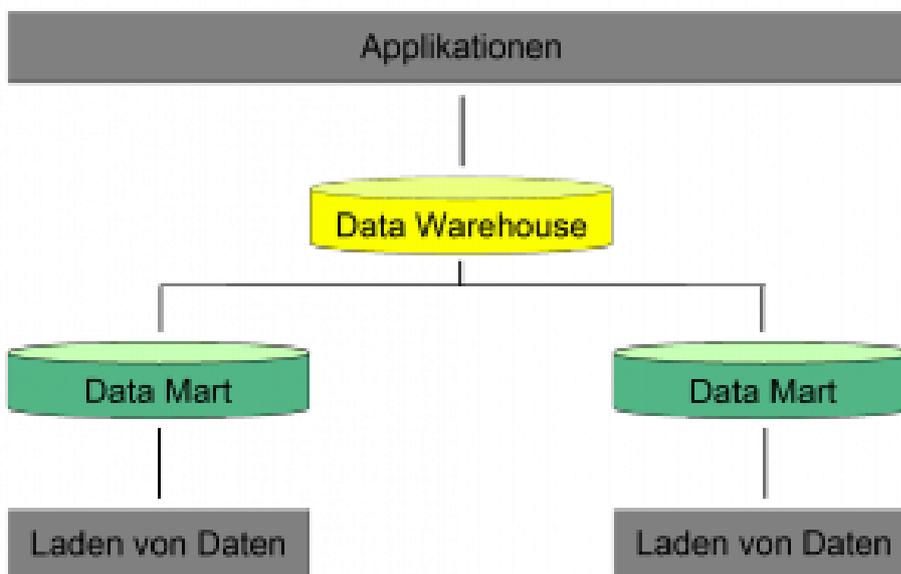


Abbildung 2.3: Architektur von unabhängigen Data Marts [Bau04, S. 63]

## 2.2.2 ETL Prozess

Die Migration der Daten ist ein essenzieller Schritt für ein Data Warehouse. Dieser Migrationsschritt wird auch Extract Load Transform Prozess, kurz ETL genannt. Die benötigten Metadaten werden in korrekter Struktur und Qualität mittels ETL Prozess bereitgestellt. Damit dies dem Anwender nicht zusätzliche Zeitressourcen kostet, sind diese Funktionalitäten oft in eigenen ETL Tools bereitgestellt [Wie99]. Kurz [Kur99] zufolge wird der gesamte ETL Vorgang kurz in einige Arbeitsschritte unterteilt. Der ETL Prozess wird in zwei allgemeine Phasen unterteilt:

- **Definitionsphase**

Diese ist dafür verantwortlich, alle Objekte festzulegen, wobei im ersten Schritt alle operativen Datensysteme, die als Quellen eingetragen wurden analysiert werden. Nach diesem Schritt werden nun Transformationen durchgeführt und abschließend eine ETL Routine für die Beladung erstellt. Diese Routine verarbeitet aufgrund von Transformationsregeln die Objekte, die zuvor eingelesen wurden und legt neue Objekte für die Staging Ebene an.

- **Ausführungsphase**

In dieser Phase sind die allgemeinen technischen Abläufe des konkreten ETL Prozesses festgelegt. Die Durchführung erfolgt aufgrund von in der Definitionsphase vorgegebenen Beladungsroutinen, Transformationseglern und Beschreibung der Metadaten.

In den Abbildungen 2.4 und 2.5 kann man die Abläufe der Defintions- bzw. Ausführungsphase in Anlehnung an Kurz [Kur99] betrachten.

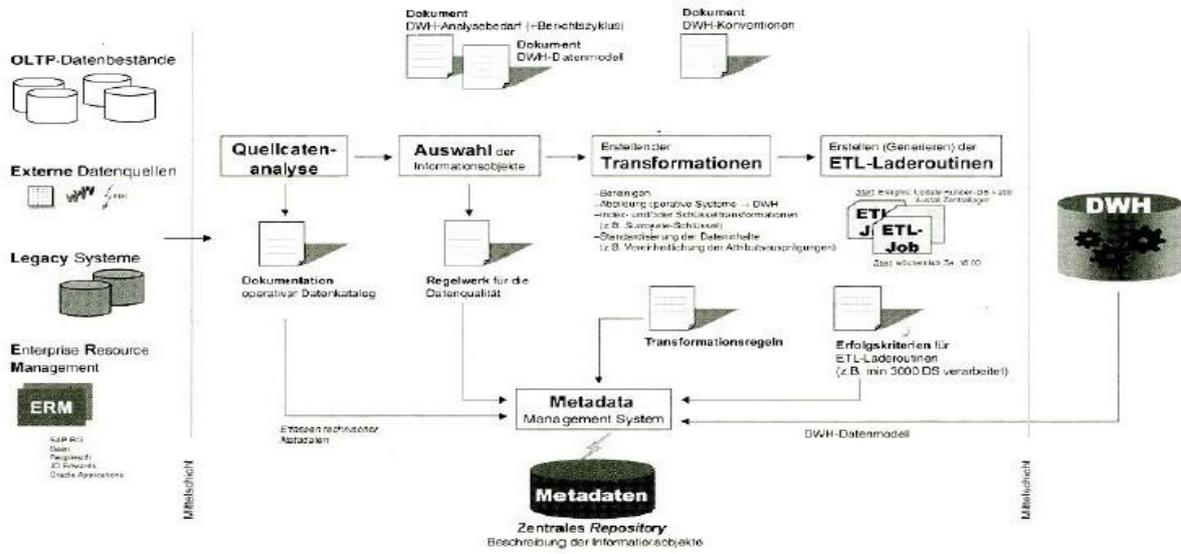


Abbildung 2.4: Definitionsphase im ETL Prozess [Kur99]

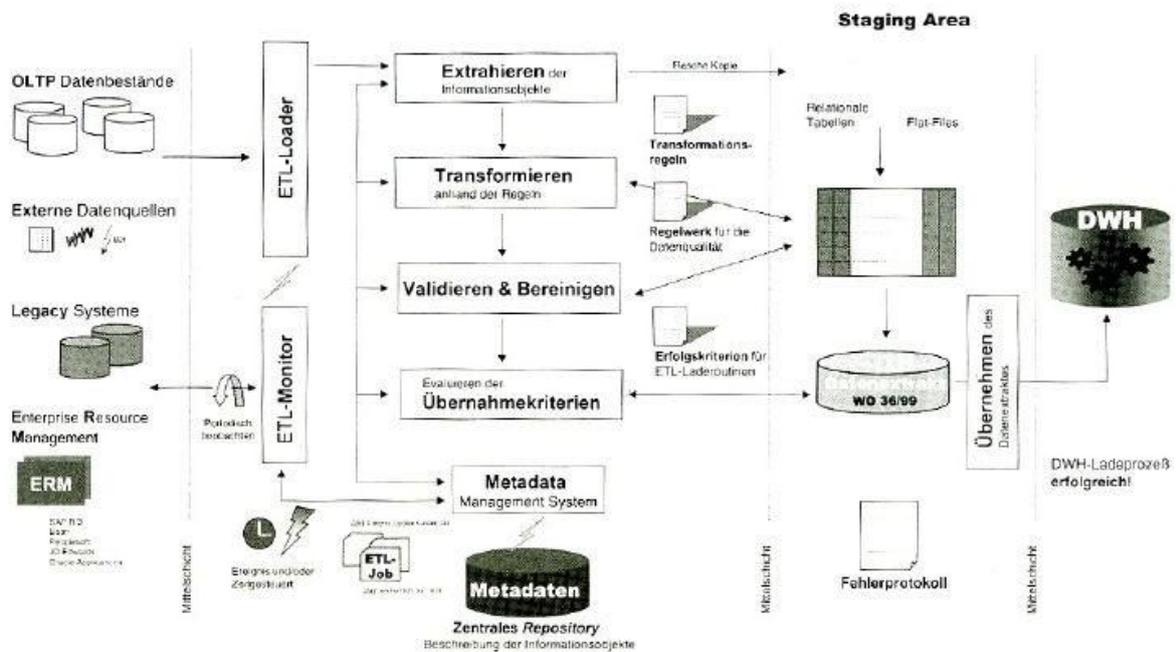


Abbildung 2.5: Ausführungsphase im ETL Prozess [Kur99]

### 2.2.2.1 ETL Prozess 1/3 : Extraktion

Mit der Datenextraktion wird der Auslesevorgang, der gewählten und bis zum Extraktionszeitpunkt veränderten Daten beschrieben. Oft kommt es vor, dass es sich hierbei um sehr große Mengen von Daten handelt. Deshalb kann es sinnvoll sein, die zu bearbeitenden Segmente zusammenzufassen, beziehungsweise zu komprimieren [Kur99]. Man kann zwischen einem periodisch kompletten Abzug der Daten, einem periodischen Abzug der geänderten Daten und einer einfachen Protokollierung aller Daten unterscheiden [Wie99, S. 190f].

Bei einem kompletten Abzug der Daten werden aus den Datenbanken der operativen Systeme spezifische Objekte geladen. Die hierbei große Datenmenge macht diesen Ansatz aus technischer Sicht zum schwierigsten und anspruchsvollsten. Jedoch bedeutet dies durch den kompletten Abzug der Daten auch den einfachsten Lösungsweg.

Wenn man sich nun auf die periodisch geänderten Daten bezieht, kann man eine Senkung der Gesamtdatenmenge erreichen. Falls sich ein Wert zwischen zwei Iterationen mehrmals geändert hat wird nur der letzte und somit aktuelle Wert berücksichtigt und in das DWH transferiert.

Bei der dritten Möglichkeit, der Protokollierung aller Änderungen, werden sämtliche Veränderungen protokolliert und danach in das DWH eingefügt [Wie99].

Um den Extraktionsprozess zu starten gibt es nach Bauer und Günzel einige verschiedene Zeitpunkte, die sie nach Kimball et al. [Kim98] aufgestellt haben.

- **Sofort**  
Wenn es notwendig ist, die Daten immer auf einem bestimmten Stand der Aktualität zu halten, ist es möglich nach jeder Form einer Änderung den Extraktionsprozess zu starten. Damit wird der Stand des DWH mit dem Stand der operativen Datenbank auf den gleichen Level gebracht.
- **Anfragegesteuert**  
Um die Extraktion zu starten, ist es notwendig eine bestimmte Anfrage auszuführen.
- **Periodisch**  
Hier wird in zeitlich gleich versetzten Intervallen eine Extraktion gestartet. Man muss hierbei jedoch die Aktualität der Daten überprüfen und aus welchem Kontext diese stammen. Daten, die sich häufiger ändern müssen anders behandelt werden, als solche, die sich nur sehr selten ändern.

- Ereignisgesteuert  
Durch einen gewissen Trigger, sei es ein Ereignis, eine Abfrage aus einer operativen Datenbank oder ein bestimmtes Zeitintervall, wird hier eine Extraktion der Daten durchgeführt. Um den Begriff „ereignisgesteuert“ abzugrenzen, können auch die Punkte periodisch und anfragegesteuert in diese Kategorie übernommen werden.

### 2.2.2.2 ETL Prozess 2/3: Transformation

Um die Daten an die Anforderungen der eigentlichen Anwendung und des DWHs anzupassen [Bau04], müssen mittels Transformation der Daten einzelne oder mehrere Felder aus den operativen Datenbanksystemen in äquivalente Felder des Data Warehouse transformiert werden und somit Schemata und Datenqualität angepasst werden [Wie99]. Wenn man Glück hat, kann man die anfallenden Werte eins zu eins übernehmen, was den einfachsten Fall darstellt. Falls dies aus einem oder mehreren Gründen jedoch nicht möglich ist, muss man entsprechende Transformationen vornehmen, um Konflikte in den Schemata zu vermeiden.

Die generelle Vorgangsweise in der Transformationsphase beginnt mit einem Zusammenführen bzw. Join der Daten aus den operativen Strukturen. Diese werden danach in einer virtuellen Tabelle abgelegt, wobei es sich hierbei je nach ETL Werkzeug um eine virtuelle oder „echte“ Datei handeln kann. Danach werden auf diese Tabellen spezifische Transformationsregeln angewandt, um die Daten für das DWH aufzubereiten. Nach der Aufbereitung werden die Ergebnisdaten in sogenannte Staging Tabellen geladen die dann ins eigentliche Data Warehouse eingebunden werden [Wie99, S. 196f].

Da die Daten aus den operativen Systemen meist aus Quellen stammen, die in heterogener Form vorliegen, muss man ein genormtes Format anstreben. Für eine erfolgreiche Datenmigration sind nach Kimball et al. [Kim98] meistens einige Transformationen notwendig. Diese sind unter anderem:

- Datumsangaben  
Diese müssen vereinheitlicht werden um erfolgreich ins DWH geladen werden zu können. Zum Beispiel wäre eine einheitliche Form des Datums in der Form DDMMYYYY erstrebenswert.
- Maßeinheiten  
Es sollte automatisch eine Umrechnung erfolgen um ein einheitliches Format zu garantieren wie zum Beispiel eine Umrechnung von Zentimeter in Meter.
- Zusammenfassung und Trennung von bestimmten Attributen  
Aus bestimmten Feldern kann man mehrere Metainformationen extrahieren. Beispielsweise könnte man „Postleitzahl – Ort“ in zwei extra Spalten aufteilen.

- Anpassung von spezifischen Datentypen  
Hierbei ist gemeint, dass man zum Beispiel Längenangaben im Typ Number speichert oder Datumsangaben im Typ Date.
- Konvertierung von bestimmten Kodierungen  
Es soll eine Vereinheitlichung von unterschiedlichen Kodierungen getroffen werden. Am Beispiel des Geschlechts würde dies bedeuten, dass 1 und f weiblich bzw. 2 und m männlich bedeuten würde.

### 2.2.2.3 ETL Prozess 3/3: Laden

Die nun mithilfe des Transformationsschrittes aufbereiteten Daten werden mit Hilfe der Ladephase direkt in das DWH eingelesen. Laut Wies [Wie99] wird bereits erwartet, dass die Daten ohne Problem geladen werden können, da die Form der Daten sich mit der Struktur der Zieltabellen im DWH bereits deckt. Durch die hohe Auslastung, aufgrund des großen Datenaufkommens, die während der Ladephase auftritt funktionieren einige Systeme während der Ausführung teilweise eingeschränkt oder werden sogar für kurze Zeit gesperrt [Bau04]. Somit hat dieser Vorgang auf alle involvierten Komponenten direkt oder indirekt Auswirkungen. Bei der Planung des Zeitpunktes eines Ladevorgangs sollte man berücksichtigen, dass die Systeme nicht schon von Benutzern vorbelastet sind. Deshalb empfiehlt sich ein Zeitpunkt meist nachts oder früh morgens.

Während des Vorgangs muss man konkret zwischen zwei Szenarien unterscheiden:

- Vollständige Ladung
- Aktualisierung

Bei der vollständigen Ladung, wird das DWH zur Gänze neu gefüllt, während bei der Aktualisierung nur neue Änderungen aktualisiert werden müssen. Bei beiden Verfahren wird zusätzlich zur Ladung ins DWH noch eine Historie der Daten angelegt. Daher werden die zu aktualisierenden Daten in einer weiteren Datenbank für Historisierung gespeichert [Inm02, S. 34].

In weiterer Folge können bei der Beladung durchaus Fehler auftreten, die aber vom ETL Tool abgefangen werden müssen um daraufhin maßgebende Schritte einzuleiten. Eine Möglichkeit des Umganges mit Fehlern wäre es zum Beispiel anstatt der zu schreibenden Werte eine Standard Konfiguration zu laden [Ana97, S. 40ff].

## 2.3 Grundstruktur eines Data Warehouse

Die Beschreibung einer Datenbank wird auch Datenbankschema genannt. Dieses wird während der Design-Phase spezifiziert und legt fest, wie die Datenbank auf einem Medium gehandhabt wird. OLTP Datenbankschemata sind vor allem für die Aufzeichnung von Geschäftstransaktionen optimiert, und verwenden zahlreiche Normalisierungsformen. Unter Normalisierung versteht man das Zerlegen von Attributen in kleinere Teile, wenn die Relationen ungenügend sind. Normalisierung hat einige Vorteile im Zuge der Datenintegrität innerhalb einer OLTP Umgebung.

Beim Datawarehouse werden oft viele nicht normalisierte Schemata verwendet, um einen schnellen Zugriff zu den Daten zu sicherzustellen, ohne dabei viele Joins auszuführen. Im Grunde genommen gibt es zwei sehr weit verbreitete Typen von Schemata für DWs:

- Star Schema
- Snow Flake Schema

Bei diesen beiden Schemata werden die Daten als eigenständige Kette repräsentiert, wobei jede Dimension einem Themenbereich zugeordnet ist.

### 2.3.1 Star Schema

Der Name des Star-Schemas kommt davon, da in einem „Entity-Relationship“ Diagramm, ein einziges Objekt in der Mitte mit allen weiteren Objekten verbunden ist und dabei gleichzeitig aussieht wie ein Stern [Pet94]. Diese zentrale Tabelle nennt sich Faktentabelle. Diese beinhaltet alle Primary Keys, mit jeweils einer Spalte für jede Dimension. Sie kann auch weitere Spalten für numerische Fakten oder andere Informationen mit sich führen.

Für jede Dimension gibt es eine Dimensionstabelle mit einem generierten Key und einem Ebenen Indikator, der angibt auf welcher Ebene der jeweilige Record liegt. In Abbildung 2.6 kann man ein Beispiel eines Star Schemas sehen. Hier kann man die Dimensionen erkennen und sehen, dass die Foreign Keys der Faktentabelle auf die Primary Keys der Dimensionstabellen joinen.

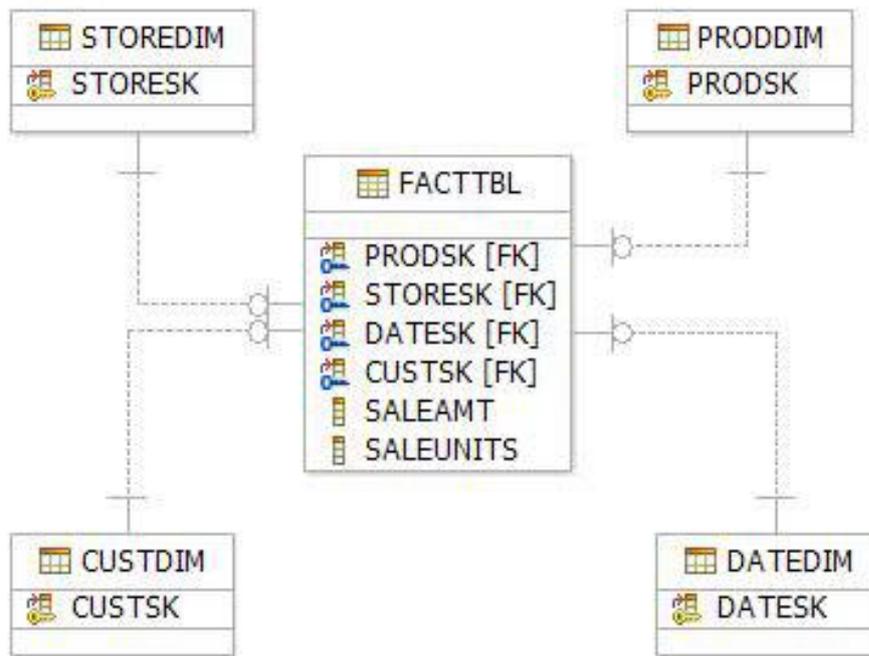


Abbildung 2.6: Typisches Star Schema [Pet94]

Weil man beim Star Schema einen Key für jede Dimension hat, muss man einen Key generieren. Ein Ansatz zur Erstellung eines Keys, ist aus allen bestehenden Keys, durch Aneinanderhängen, einen Key zu bekommen. Dies führt jedoch zu sehr langen Keys. In der Praxis hat es sich bewährt, anstatt Keys Namen zu geben, einfache Nummern zu verwenden. Es soll möglich sein, den kleinsten möglichen Key zu generieren, der die einzelnen Records eindeutig voneinander trennen kann, wobei sich hier vor allem Integer Größen eignen. Diese Methode der Generation resultiert in der besten Performance, da die Größe eines Integer klein ist und die Wartung dieser Keys sehr gering im laufenden Alter des DWHs ausfällt.

### 2.3.1.1 Vorteile

- Das Star Schema ist sehr leicht verständlich, da die Daten rund um die Themengebiete strukturiert sind.
- Durch die denormalisierte Form kann man leicht in die Daten suchen.
- Es sind, um Informationen über alle Spalten zu erhalten, nur wenige Joins nötig.
- Die Wartung des DWHs fällt sehr gering aus, da die Keys aus Integer bestehen.

### 2.3.1.2 Nachteile

- Durch die denormalisierte Form kommt es zu vielen Redundanzen.
- Die Darstellung erfordert mehr Speicher im Gegensatz zum Snow Flake Schema.
- Konsistenzprobleme entstehen durch das Problem der Redundanzen, falls die Daten aktualisiert werden.

### 2.3.2 Snow Flake Schema

Das Snowflake Schema ist im Grunde eine Erweiterung des Star Schemas, wobei hier jeder Punkt des Sterns in weitere Punkte aufteilt wird. In diesem Schema sind die Dimensionstabellen, im Gegensatz zum Starschema, mehr normalisiert. Der Vorteil des Snow Flake Schemas ist vor allem die Erhöhung in der Abfrage Performance, da weniger Daten in den Datenbanken gespeichert werden müssen.

Auch wird eine generell höhere Leistung durch Joinen von kleinen normalisierten Tabellen erreicht. Beim Star Schema muss man auf große denormalisierte Tabellen joinen. Die Granularität der Dimensionen wird ebenfalls durch die Normalisierung erreicht. Aber durch das hohe Tabellenaufkommen im Snow Flake Schema, erfordert bereits eine einfache Suche in den Daten, ein Joinen der Tabellen, um alle gewünschten Informationen zu erhalten.

Falls die Datenspeicherung geringe Redundanz hat, erreicht man mit diesem Schema nur sehr geringe Leistung für aggregierte Abfragen. In Abbildung 2.7 kann man ein typisches Snow Flake Schema betrachten.

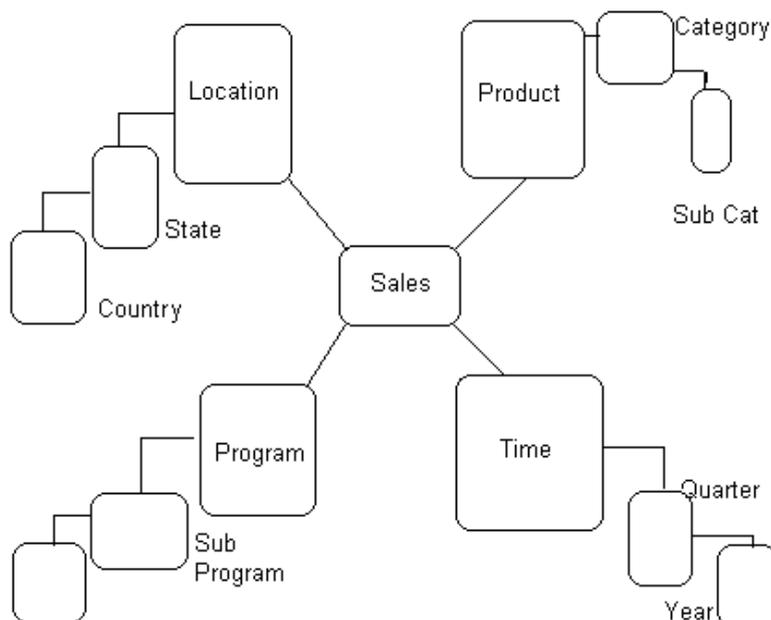


Abbildung 2.7: Typisches Snow Flake Schema [Pet97]

## 2.4 Reporting

Um einen Gewinn aus den im Data Warehouse abgespeicherten Informationen zu beziehen, ist es notwendig bestimmte Anwendungen einzusetzen, um in weitere Folge Abfragen und spezifische Analysen umzusetzen. Mittels solcher Anwendungen, auf die in den folgenden Unterkapiteln genauer eingegangen wird, ist man im Besitz mächtiger Analysewerkzeuge die jeweils für bestimmte Gebiete und Bereiche eingesetzt werden, weshalb es auch sinnvoll ist eine weitere Einteilung je nach Klassifizierung zu treffen. Analyse Tools werden oft unter dem Begriff OLAP zusammengefasst [Tan97]. Dieser Begriff bietet keine genaue Aussage über das Verfahren, jedoch hat sich OLAP vermehrt für die multidimensionale Analyse bewährt.

Es werden von Tanler fünf Funktionen erwähnt, welche auf jeden Fall in ein Analyse Tool integriert sein müssen [Tan97]:

- **Formatierung**  
Es wird eine Logik für die Anwendung benötigt, welche es ermöglicht eine standardisierte Speicherung wie beispielsweise als XML Datei sowie eine geeignete Formatierung durchzuführen.
- **Abfrage**  
Es sollen von der Logik spezifische SQL Abfragen erzeugt werden.
- **Anzeige**  
Am Ende sollen die Daten in formatierter Form als Abfrage oder Bericht für den Endnutzer sichtbar gemacht werden.
- **Verarbeitung**  
Die Ergebnisse einer Abfrage sollen von der Logik der Anwendung nach spezifischen Kriterien analysiert werden.
- **Anwendungsschnittstelle**  
Es soll dem Endnutzer möglich sein eigene Eingaben wie Text oder Zahlen an integrierte Funktionen der Applikationslogik weiterzugeben.

Um diese Werkzeuge zu separieren, gibt es eine grobe Einteilung in vier Bereiche, die aber nicht sehr stark ausgeprägt ist, da man in Wirklichkeit sehr oft Mischformen zwischen den einzelnen Verfahren benötigt.

Bei genauerer Betrachtung der heutzutage am Markt befindlichen Analysetools kann man erkennen, dass meistens ein bestimmtes Verfahren sehr stark ausgeprägt wird, aber um

dessen Durchführung zu verwirklichen wird dieser Prozess durch andere Verfahren unterstützt.

### **2.4.1.1 Berichtswerkzeuge und Abfragewerkzeuge**

Die Funktionalität dieser Werkzeuge ist in etwa die gleiche, wie jene, die in den relationalen Datenbanken von Haus aus implementiert ist. Daher zählen Berichts- und Abfragewerkzeuge zu den am einfachsten gestalteten Tools.

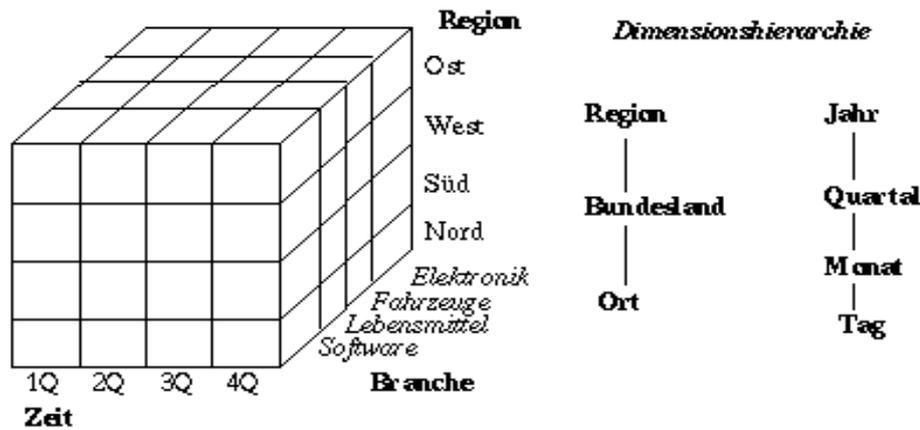
Diese Werkzeuge sollen die Abfragen Erstellung allerdings erleichtern, indem sie beispielsweise Query by Example anstatt SQL verwenden. In weiterer Folge soll es dem Endbenutzer möglich sein, die Ausgabe der Daten einer Abfrage so zu gestalten, wie er es sich es wünscht. Somit liegt der Hauptfokus dieser Anwendungen in der Erstellung von Listen, Berichten oder einfachen Zählungen.

Am meisten verbreitet sind sogenannte Ad-hoc Abfragen, die vom Benutzer spontan durchgeführt werden, um Daten zu bestimmten Zeitpunkten zu bestimmen. Es gibt aber auch die Möglichkeit, diese Berichte automatisch zu erstellen. Für Unternehmen bedeutet das zum Beispiel monatlich einen gewissen Statusbericht durchzuführen.

### **2.4.1.2 Multidimensionale Analyse**

Hierbei handelt es sich um ein sehr tiefgehendes Verfahren zur Analyse im OLAP. Es wird hier zwischen den eingesetzten Datenbankmanagementsystemen (DBMS) unterschieden. In Bezug auf Kapitel 2.1 handelt es sich bei einem multidimensionalen DMBS um MOLAP und bei einem relationalen um ROLAP.

Die eigentlichen Daten werden hier in einem Cube oder Hypercube gespeichert. Als Beispiel für einen Cube mit drei Dimensionen dient Abbildung 2.8. Die Kanten des Cubes sind die eigentlichen verwendeten Dimensionen. In jeder einzelnen Zelle des Cubes ist eine Information gespeichert, in der sich die jeweiligen Dimensionen überschneiden. Damit sollte laut Theorie eine sinnvolle Aggregation der Daten und eine Summenbildung möglich sein. Die einzelnen Operationen eines Hypercubes wurden bereits in Kapitel 2.1 erklärt.



**Abbildung 1: Mehrdimensionale Datenstruktur ( aus [Rah97] )**

Abbildung 2.8: Beispielhafter Aufbau eines Hypercubes nach Inmon [Inm02]

Der Vorteil bei der multidimensionalen Analyse ist jener, dass die Verfahren in beliebiger Reihenfolge angewandt werden können. Auch die Auswahl der Dimensionen ist beliebig und die Integrität der Berechnungen wird dabei nicht verfälscht. Es werden durch die multidimensionale Analyse sehr flexible Abfragen möglich wie zum Beispiel:

- Was ist unser teuerstes Produkt?
- Welche Studenten haben im Jahr 2007 Softwareentwicklung – Wirtschaft studiert?

Dazu wird zusätzlich eine weitere Ebene für die Berechnungen benötigt. Wenn man die einzelnen Zellen des Cubes als Datenträger betrachtet, der verdichtete Werte aus dem DWH enthält, dann erfordern solche komplexen Abfragen weitere Aggregationen.

### 2.4.1.3 Statistische Analyse

Diese Werkzeuge dienen dazu, sehr große Datenmengen auf einige Charakteristika zu reduzieren. Hier kommen Formeln wie zum Beispiel Mittelwert oder Median zum Tragen. Diese werden vor allem bei Marktanalysen eingesetzt, denn hier ist oft die gängige Frage:

Was wäre wenn?

Die Modelle zeichnen also alternative Szenarien ab die zukünftig eintreten könnten. Es muss bei diesem Analysetool stark zwischen dem Analysten und dem Anwender der Abfrage unterscheiden werden. Im Regelfall gibt es nur ein paar Analysten, welche Abfragen erstellen und sehr viele Anwender, die diese Tools dann nutzen, um Prognosen zu machen.

#### **2.4.1.4 Data Mining**

Diese Technik bedient sich vieler Funktionalitäten der statistischen Analyse. Diese werden dann noch mit komplexen Funktionen erweitert. Es kommen hier erweiterte Algorithmen aus den Bereichen neuronale Netze und Mustererkennung zum Einsatz um gewisse Vorhersagen treffen zu können. Diese Applikationen laufen vermehrt im Hintergrund und richten sich nicht wie die statistische Analyse an den Benutzer. Data Mining steckt eigentlich noch in den Kinderschuhen, jedoch gibt es auf diesem Gebiet bereits sehr viele Projekte und Forschungsaktivitäten. Auf Unternehmungsebene kommt Data Mining aber nur für große Unternehmen in Betracht [Jia2000].

## 2.5 Ansatzarten von OLAP

Nachfolgend werden einige weit verbreitete Ansätze zur Modellierung von OLAP Systemen angeführt.

### 2.5.1.1 ROLAP

ROLAP ist der am meisten verbreitetste Ansatz wobei das „R“ für relational steht. Da sich Technologien für Datenbanken, die relational sind, durchgesetzt haben, ist dies auch der Grund warum ROLAP der am weitesten verbreitetste Ansatz ist. Hier gibt es einen zentralen Server, der sowohl auf die Metadaten der Datenbank, als auch auf das DWH zugreift. Für Abfragen und die Datenaufbereitung wird SQL herangezogen und dadurch, dass ROLAP auf RDBMS basiert, ist es in der Lage mit großen Mengen von Daten umzugehen. Es gibt auch eine Art Cache, in der OLAP spezifische Abfragen, die zum Teil sehr oft verwendet werden, im Vorhinein ausgeführt werden. [Kur99] [Kot98]

### 2.5.1.2 MOLAP

Das „M“ in MOLAP steht für Multidimensional. Es wurde als Alternative zu ROLAP entwickelt. Obwohl sowohl MOLAP als auch ROLAP dazu entwickelt wurden, um mit Hilfe eines multidimensionalen Datenmodells Analysen durchzuführen, unterscheidet sich MOLAP von ROLAP dadurch, dass es eine Vorberechnung benötigt, welche als Processing bekannt ist. Die meisten dieser Ansätze speichern diese Daten in einem multidimensionalen Array und nicht in einer relationalen Datenbank wie ROLAP. Es gibt zahlreiche Methoden und Algorithmen die ein effektives Speichern mit einer MOLAP Lösung ermöglichen. Die Hauptvorteile einer MOLAP Lösung sind z.B. schnelle Abfragen-Leistungen, oder durch den Einsatz von Kompressionstechniken kleinere Speichergrößen. [Kur99]

### 2.5.1.3 HOLAP

HOLAP ist eine Kombination aus ROLAP und MOLAP und ist demzufolge „Hybrides“ OLAP. Damit ist es möglich einen Teil der Daten in einen MOLAP Speicher zu sichern und den anderen Teil in einen ROLAP Speicher, wodurch die Vorteile beider Ansätze genutzt werden können. Es ist jedoch von Produkt zu Produkt unterschiedlich inwieweit der Designer des Cubes Kontrolle über die Partitionierung hat. [Kur99]

Damit OLAP sich innerhalb der benötigten Datenstrukturen navigieren kann, benötigt man einige Funktionen die dies ermöglichen. Zu den Standard Funktionen eines OLAP Systems, mit denen die eigentlichen OLAP Abfragen gemacht werden, gehören:

- **Pivot**  
Hier werden die Daten aus verschiedenen Perspektiven betrachtet, d.h. der Datenwürfel wird gedreht (virtuell).
- **Drill-Down**  
Dies bedeutet das Hinzufügen von Informationen zu einem Report. Das kann innerhalb einer Dimension passieren, als auch durch das Hinzufügen von Informationen aus anderen Dimensionen.
- **Roll-Up**  
Ist das genaue Gegenteil zu Drill-Down.
- **Drill-Across**  
Hiermit wird das Austauschen der X und Y Achsen im Report bezeichnet.
- **Dice**  
Durch das Setzen von Filtern wird aus dem Hypercube eine gewisse Teilmenge herausgenommen. Das Ergebnis ist wiederum ein Datenwürfel, der jedoch schneller analysiert werden kann.

Zu den Standard Funktionen eines OLAP Systems kommen noch die zum Teil produktspezifischen Funktionen wie zum Beispiel Extended Drill-Down oder Drill-Through. Diese Funktionen sind dazu da, um die Analyse noch genauer zu gestalten und dadurch die Auswertungen präziser zu machen. [Kur99] [Swi2000]

## 2.6 Modellierung eines Datawarehouse

Im Grunde gibt es zwei Ansatzarten, um ein Datawarehouse zu implementieren. In den folgenden zwei Unterkapiteln wird einerseits der relationale Ansatz nach Bill Inmon und andererseits der dimensionale Ansatz näher erklärt. Der relationale Ansatz wird auch top-down Ansatz genannt. Der dimensionale Ansatz wurde von Ralph Kimball eingeführt und wird als bottom-up Ansatz verstanden. Es gibt in weiterer Folge noch andere Ansatzmöglichkeiten für die Modellierung eines DWH, wie zum Beispiel Imhoff et al. (2003) mit einem Ansatz, der die meisten Vorteile aus dem relationalen und dem dimensionalen Ansatz vereint [Pon01].

### 2.6.1 Relationaler Modellierungsansatz

Der relationale Ansatz deckt die komplette DWH Architektur ab. Die Idee im Hintergrund ist ein zentrales Data Warehouse, oder auch Enterprise DWH genannt, das gebaut wird um alle Daten in einem integrierten und normalisierten Format zur Verfügung zu stellen. Diese

Zentrale leitet gewünschte Daten dann an die jeweiligen Data Marts weiter, wobei der Hauptteil der Analysearbeit in den Data Marts geschieht. Ausgewählte Anwender können auch direkt auf das zentrale DWH zugreifen und somit alle korrelierte Daten anzusehen. In Abbildung 2.9 kann man die Architektur eines solchen Ansatzes erkennen [Inm02].

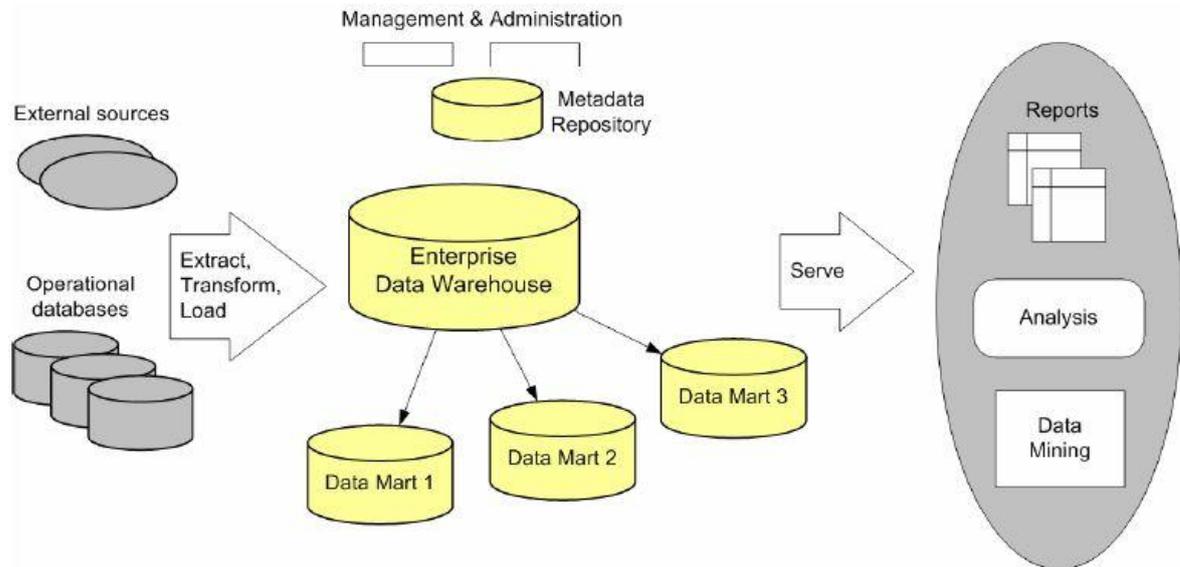


Abbildung 2.9: Architektur eines Data Warehouse nach relationalem Ansatz [Inm02]

Hierbei wird das zentrale DWH als Quelle von integrierten und hoch qualitativen Daten angesehen. Die Qualität der Daten wird durch den ETL Prozess in der Staging Ebene garantiert, da hier die Daten von Duplikaten bereinigt werden und einige spezifische Methoden angewandt werden, um möglicherweise korrupte Daten zu beseitigen. Für die Datenspeicherung wird immer eine relationale Datenbank herangezogen, wobei hier das Datawarehouse nach der dritten Normalform normiert ist, um Integrität und Konsistenz der Daten sicherzustellen. Die Qualität der Daten muss beim Laden in die Data Marts nicht sichergestellt werden, da die Data Marts in Relation mit dem zentralen DWH stehen. Weiters ist es sehr einfach Zieldaten in die Marts zu laden, da es nur eine einzige zentrale Quelle gibt. Anders als die zentrale DWH Komponente können die Data Marts mit Technologien wie DBMS gebaut werden [Inm02].

Die Marts werden meist abteilungsspezifisch verwendet. Das bedeutet, dass ein Data Mart zu jeweils einer Abteilung eines Unternehmens gehört. Ein weiterer Faktor bei der Verwendung des relationalen Ansatzes ist die zukunftsorientierte Entwicklung. Inmon nennt dies das „day 1- day n phenomen“. Das bedeutet, dass man bei Bedarf ohne Probleme weitere Data Marts sowie Themengebiete zum Data Warehouse hinzufügen kann.

### 2.6.1.1 Vorteile

Einige der grundlegenden Vorteile der relationalen Lösung sind:

- Das Enterprise DWH verfügt von Natur aus über eine Basisarchitektur und besteht nicht aus mehreren verschiedenen Data Marts [Pon01].
- Das Enterprise DWH garantiert die Qualität der Daten in den Data Marts [Inm02].
- Das Enterprise DWH garantiert sogenannte „Kernregeln“ und Kontrolle für das Data Warehouse [Pon01].

### 2.6.1.2 Nachteile

Der Hauptkritikpunkt bei der Verwendung des relationalen Ansatzes liegt darin, dass im Enterprise DWH als Datenbankstruktur die dritte Normalform verwendet wird. Der Vorteil von 3NF ist die Eigenschaft bestimmte Business Regeln, wie zum Beispiel Multiplikationen auf Beziehungen zwischen Objekten, auf die Daten anzuwenden. Aber das erfordert sehr hohe Komplexität was der eigentliche Kritikpunkt ist. Laut Kimball [Kim02] resultieren folgende Probleme aus dem Ansatz der 3NF:

- Die normalisierte Datenbankstruktur für alle Daten verschiedener Quellen zu entwickeln, kann sehr viel Zeit und Ressourcen in Anspruch nehmen. Das nimmt den eigentlichen Fokus vom Design der Data Marts, die die Hauptschnittstelle der Benutzer des DWHs sind.
- Das Enterprise DWH und die normalisierte Datenbank sind für die Anwender zugänglich. Dies ist ein Problem, auch wenn die Strukturen oft zu komplex sind um sie zu verstehen.
- Die Daten müssen im eigentlichen Sinne zwei Mal extrahiert und geladen werden. Anfangs in das Enterprise Data Warehouse und danach in die einzelnen Data Marts. Dadurch resultieren höhere Entwicklungszeiten sowie Wartungskosten

Auch die abteilungs-basierte Struktur der Data Marts wird als Nachteil angesehen. Wenn man verschiedene Datenformate für die verschiedenen Marts nutzt, benötigt man dafür mehr Arbeit und es kann zu Unstimmigkeiten führen, falls Abteilungen übergreifende Analysen machen wollen [Kimball, 2002]. Abschließend kann man sagen, dass der top-down Ansatz für die Modellierung eines DWH eine höhere Fehleranfälligkeit mit sich bringt als der bottom-up Ansatz [Pon01].

### 2.6.2 Dimensionaler Modellierungsansatz

Im Gegensatz zum relationalen Ansatz geht der dimensionale Ansatz einen ganz anderen Weg. Hierbei arbeitet man in der Regel nahezu „geradeaus“. Daraus resultieren schnelle Resultate, und Performance. Hier gibt es kein zentrales DWH, sondern man entwickelt iterativ Data Marts, die virtuell ein Data Warehouse formen [Kim02].

In Abbildung 2.10 kann man die Architektur eines DWHs beim dimensionalen Ansatz erkennen. Die Staging Ebene ist ähnlich wie beim relationalen Ansatz, jedoch wird beim dimensionalen Ansatz auf die Normierung der Datenbankstruktur verzichtet [Kim02]

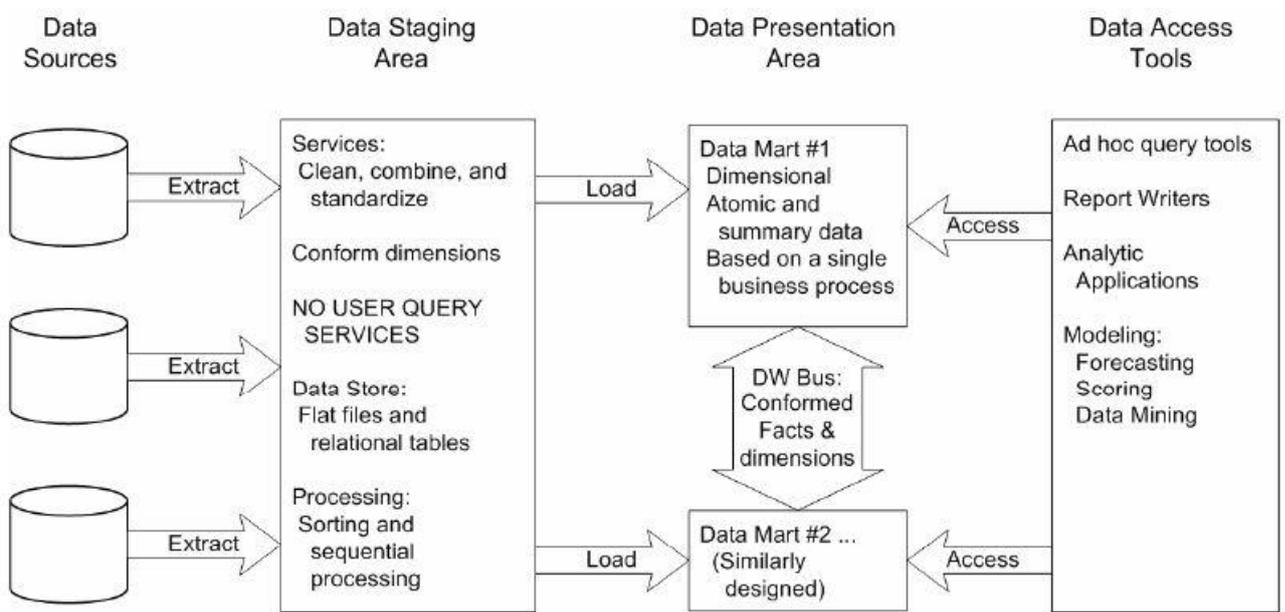


Abbildung 2.10: Architektur eines Data Warehouse nach dimensionalem Ansatz [Kim02]

Durch den ETL Prozess in der Staging Ebene werden die Data Marts der Präsentationsschicht geladen und aktualisiert. Hier sind die Data Marts so gebaut, dass sie Geschäftsprozesse der Organisation beschreiben können. Das bedeutet, ein Data Mart enthält Daten zu einem spezifischen Prozess.

Ein wichtiger Aspekt bei der dimensionalen Modellierung ist, dass die Data Marts die Daten in multidimensionaler Form speichern. Dies entspricht dem Cube Ansatz und falls relationale Datenbanken als Datenspeicher verwendet werden, empfiehlt sich die Verwendung des Star Schemas. Das Snowflake Schema sollte vermieden werden, da es das Schema unnötigerweise komplex macht und die Performance im Gegensatz zum Star Schema schlechter ist. Der dimensionale Ansatz setzt auch voraus, dass die Data Marts detaillierte Daten enthalten, weil dies der beste Weg ist, um es dem Anwender zu ermöglichen jederlei Art von Abfragen oder Analysen durchzuführen. Aggregierte Daten können ebenfalls in den Marts vorhanden sein um die Leistung zu steigern, aber Detailinformationen sind nötig, um auch mit ad-hoc Abfragen umzugehen [Kim02].

Damit man Abfragen auch übergreifend ausführen kann, ist es nötig, dass die Data Marts kompatibel zueinander sind. Dies hat auch den Sinn virtuell das Data Warehouse zu formen. Diese Kompatibilität wird durch so genannte „konformen“ Fakten und Dimensionen sichergestellt. Konforme Dimensionen kann man meist dadurch schaffen, indem man gemeinsame Dimensionstabellen zwischen den Star Schemata der Data Marts nutzt. Durch die Sicherstellung, dass mehrfache Definitionen oder Datentypen in den Data Marts die gleichen sind, erhält man konforme Fakten. Falls dies jedoch nicht möglich ist muss die Bezeichnung geändert werden, da man hier nicht vom selben Objekt spricht [Kim02].

### **2.6.2.1 Vorteile**

Laut Ponniah und Kimball sind die Vorteile der dimensionalen Lösung [Pon01] [Kim02]:

- Schnellere und leichtere Implementierung der Projekte
- Geringeres Fehlerrisiko verglichen mit dem top-down Ansatz
- Hauptaugenmerk ist der Performancegewinn

Hier ist hinzuzufügen, dass dieser Ansatz sehr weit verbreitet und in der Praxis bereits erprobt ist.

### **2.6.2.2 Nachteile**

Durch die Verwendung von verschiedenen Data Marts wird dem dimensionalen Ansatz oft nachgesagt, dass es sich hierbei nicht um ein richtiges DWH handle, da der Nutzen von Data Marts abteilungsspezifisch ist. Auch der Tatsache der mehrfachen Speicherung der Daten in übergreifenden Data Marts wird nachgesagt, dass diese zu inkonsistenten Ergebnissen führen kann [Inm99].

Auch aufgrund der Data Mart Definition und des relationalen Ansatzes wird diese Vorgehensweise kritisiert. Es scheint, dass die Marts nicht dazu geschaffen sind konforme Fakten und Dimensionen zu haben, sodass die Fehlerrate sehr hoch ansteigen kann. Wenn man die Data Marts nicht sehr genau plant, wird dies zu Inkompatibilitätsproblemen führen [Pon01].

## 3 Ausgangssituation – Bestandsaufnahme

In diesem Kapitel wird eine Bestands- bzw. Momentaufnahme der vorhandene Daten und Abfragen, die dem F&T-Haus zur Verfügung stehen durchgeführt. Es wird auch detailliert auf die Forderungen, die an die Umsetzung gestellt wurden, eingegangen sowie eine kurze Einführung in das alte bereits verwendete System „Oracle Discoverer“ gegeben. Anhand dieser Ausgangssituation soll verstanden werden, warum es für die F&T Abteilung förderlich ist, eine State of the Art Lösung zu verwenden und welche Vorteile daraus resultieren.

### 3.1 Historie

Im Allgemeinen gibt es mehrere Lösungen und Ansätze, um Daten für weitere Verwendung aufzubereiten. Die Rohdaten liegen meist in historisierter Form in Datenbanksystemen vor. Um aus diesen Fakten Informationen zu gewinnen bedarf es einiger Aufbereitungs- bzw. Analysearbeit. Diese Arbeit wird generell von sogenannten BI- oder Reporting-Tools übernommen, welche es dem Anwender einfach machen, schnell und präzise die gewünschten Metadaten zu ihrer Abfrage zu erhalten.

An der Technischen Universität Graz gibt es mehrere Abteilungen, an denen Reportingtools zum Einsatz kommen. Eine davon ist das F&T-Haus. Hier beschäftigt man sich vor allem damit, herauszufinden, welche Kooperationspartner und externen Organisationen an Abschlussarbeiten oder auch an Forschungsprojekten der TU Graz beteiligt waren. Da für diese Analysen oft nur eine begrenzte Zeit zur Verfügung steht, verwendet man in dieser Abteilung seit einigen Jahren das von der Firma Oracle entwickelte Tool Discoverer. In Kapitel 3.3 wird näher auf dieses Analysewerkzeug eingegangen und es werden die Möglichkeiten, die der Endnutzer damit hat aufgezeigt.

Im Jahr 2009 wurde an der TU Graz durch den Zentralen Informatikdienst (ZID) ein Projekt gestartet, das den Einsatz eines zentralen Data Warehouse für die TU Graz mit sich führte. Zunächst wurden im DWH nur studienrelevante Daten erfasst, damit die Studienabteilung mit der Analyse der Daten wichtige Informationen schnell durch den Einsatz eines Analysetools abrufen kann.

Die hierbei verwendete Software kommt ebenfalls aus dem Hause Oracle und ist ein Zusammenspiel des Oracle Warehouse Builders und der Oracle BI Suite, mit welcher nach Archivierung der Daten, im DWH ad-hoc Analysen über die Data Marts ausgeführt werden können.

Die eigentliche Idee ist es, den Stand der vorhandenen Daten, der vom F&T-Haus benötigt wird, in das Data Warehouse zu transferieren und die vorhandene Software Discoverer zu modernisieren. Der Grund dafür ist, dass diese „veraltete“ Software seitens Oracle nicht mehr unterstützt wird und im eigentlichen Sinne nicht mehr verwendet werden sollte. Die BI Suite ist der direkte Nachfolger des Discoverer und bringt nicht nur die Grundfunktionalitäten

der alten Software mit sich, sondern auch eine Vielzahl neuer Features, welche es dem Anwender noch einfacher machen, schnell und effektiv Reporte zu erstellen, und diese anschließend für die jeweilige Fachabteilung aufzubereiten.

Damit dieses Vorhaben strukturiert und erfolgreich ablaufen kann, hat das F&T-Haus einige wichtige Informationen zur Verfügung gestellt. Ein wichtiger Punkt hierbei sind die eigentlichen Anforderungen der Abteilung, die in Kapitel 3.2 genau beschrieben werden. Grundsätzlich wurden zwei verschiedene Lösungen für den Übergang vom Discoverer zur BI Suite implementiert.

Einerseits wurde die komplette Struktur der vorhandenen Daten eins zu eins relational im DWH implementiert und für die BI Suite zur Verfügung gestellt. Auf der anderen Seite wurden die Daten strukturiert zu logischen Dimensionen zusammengefasst und ein multidimensionaler Ansatz im Data Warehouse der TU Graz geschaffen.

In dieser Masterarbeit werden im Kapitel 4 die beiden Ansätze strukturiert und im Detail miteinander verglichen. Anhand dieser Evaluierung erkennt man, welche Wege der Datenaufbereitung für die vorhandenen F&T Daten am sinnvollsten sind, und warum der andere Lösungsweg scheitert bzw. nicht wirklich anzustreben ist.

## **3.2 Forderungen und Gebiete des F&T-Hauses**

Da sich das F&T-Haus der TU Graz unter anderem mit dem Analysewesen der Bereiche Abschlussarbeiten und Technologietransfer beschäftigt und es sich bei diesen Gebieten um einen Zusammenschluss großer und komplexer Datenmengen handelt, muss eine angestrebte Reporting-Lösung auf jeden Fall damit umgehen können. Im Anschluss werden nun die Hauptarbeitsgebiete des F&T-Hauses kurz erklärt.

### **3.2.1 Abschlussarbeiten**

In diesem Themengebiet werden sämtliche Analysen, die im weitesten Sinne etwas mit Abschlussarbeiten zu tun haben getätigt. Es muss möglich sein detaillierte Informationen zu jeder einzelnen in der Datenbank gespeicherten Arbeit abzurufen und diese Fakten anschließend für weiterverarbeitende Zwecke aufzubereiten. Auf die vorhandenen Daten aus diesem Bereich wird in Kapitel 3.4 näher eingegangen.

### 3.2.2 Technologietransfer

Hierbei handelt es sich um ein Gebiet, in dem man herausfinden will, welche Kooperationspartner (intern oder extern) es zu bestimmten einschlägigen Projekten gibt. Es handelt sich hierbei rein um Projekte aus dem Bereich Forschung. Da es oft sehr viele Partner bei den einzelnen Vorhaben gibt, ist ein Analysetool zur Auswertung essenziell, da man mit einer großen Menge manuell nicht zeiteffektiv arbeiten könnte.

### 3.2.3 Anforderungen

Seitens des F&T-Hauses gibt es für die Umsetzung der Implementierung einige Vorgaben, die auf alle Fälle zu berücksichtigen sind.

- Problemloser Umstieg für Mitarbeiter

Die Mitarbeiter im Bereich Reportingwesen und Abfragen sollen ohne Probleme vom Discoverer in das neue System umsteigen können. Da sie mit der alten Lösung bereits vertraut sind, muss das BI Suite Vorgehen so gestaltet werden, dass es weiterhin möglich ist, rasch und effektiv Berichte für die Fachabteilungen zu erstellen. Dies ist ein wichtiger Faktor, da es bei Unklarheiten mit der neuen Lösung eventuell zu Stehzeiten innerhalb des Reportingwesens kommen kann und somit die Zeitkonstante innerhalb des F&T-Hauses nicht mehr eingehalten werden kann.

- Einschulung für Anwender

Da ein reibungsfreier Umstieg natürlich nicht eins zu eins ohne einiges an Vorwissen geschehen kann, muss auf jeden Fall für eine Einschulung bzw. Tutorien - Informationen seitens des ZID gesorgt werden. Die Einschulungsphase bedarf nicht allzu vieler Zeit, da die Grunddaten erhalten bleiben. Es werden auf der Online Plattform der TU Graz Schulungsmaterialien zur Verfügung gestellt und Workshops mit den Mitarbeitern des Bereichs Reporting durchgeführt.

- Flächendeckende Bereitstellung der benötigten F&T Daten

Die Daten der Quellsysteme für die Bereiche Abschlussarbeiten und Technologietransfer müssen ohne Lücken in das DWH System transferiert werden. Es sollen somit keine Detailinformationen verloren gehen und eine eventuelle spätere Nachbereitung minimal gehalten werden. Dazu muss man zunächst das Quellsystem des Discoverer analysieren und die vorhandenen Daten transparent machen, damit man weiß, welche Rohdaten für einen reibungslosen Umstieg benötigt werden.

- Namenskonvention

Da die Quelldaten in oft sehr minimal benannten Datenbankschemen vorliegen, sollte man für eine Vereinfachung der Operabilität der Endnutzer eine anforderungsbasierende Namensgebung anstreben. Die Benennung muss mit der Fachabteilung abgesprachen werden und nach deren Wunsch angepasst werden.

- Transparenz

Die Technikkomponente hinter der Analyseanwendung soll für den Nutzer nicht zugänglich gemacht werden. Die Mitarbeiter sollen sich rein auf die Aufgabe der Reporterstellung konzentrieren und brauchen somit zu den Datenbankschemen und Businessmodellen keinen konkreten Zugriff.

- Performance

Die Leistung der neuen Anwendung soll nicht schlechter sein als die des Discoverer. Die Dauer der Analysen und die allgemeine Zeit beim Erstellen einer Abfrage sollen nicht mehr Ressourcen in Anspruch nehmen als bereits vorhanden. Erstrebenswert ist ein Wert kleiner gleich des IST Wertes.

- Wartbarkeit

Falls es im Laufe der Zeit den Wunsch gibt neue Themengebiete oder Detailinformationen in das System zu integrieren, soll dies ohne große Komplexität möglich sein. Das DWH soll zukunftsorientiert sein und stets die Möglichkeit bieten neuen Kontext zu integrieren ohne dabei große Kosten zu verursachen.

- Ergebnis Übereinstimmung

Die Ergebnisse aus randomisiert ausgewählten bestehenden Abfragen, die bereits im Discoverer vorhanden sind, müssen sich mit den Ergebnissen des neuen Systems decken und somit einen Datenverlust ausschließen.

Neben den Grundanforderungen seitens des F&T-Hauses bedarf es natürlich einer genauen Vorgehensweise in Bezug auf die Implementierung. Zusätzlich ist es notwendig bei der Implementierung im DWH für spätere Wartbarkeit bereits zukunftsorientiert zu denken und Schnittstellen für weitere Daten offen zu lassen.

### 3.3 Oracle Discoverer

Die zurzeit vom F&T-Haus verwendete Lösung für das Berichtswesen ist der Oracle Discoverer. Hierbei handelt es sich um ein auf ROLAP basierendes Analysewerkzeug. Dieses Produkt bietet OLAP basierende Analysen für Online-Systeme und Data Warehouse Lösungen. Um die Schemata der Datenbanken richtig zu verbinden wird eine Metaebene verwendet um diese den Business Ansprüchen anzupassen. Den Anwendern ist es möglich komplexe Abfragen und berichtsübergreifende Reporte ganz ohne Programmierkenntnisse durchzuführen. In Abbildung 3.1 bekommt man einen Eindruck vom Discoverer. Darin kann man den Output eines Berichtes, sowie die verwendeten Tabellenelemente und Filter erkennen.

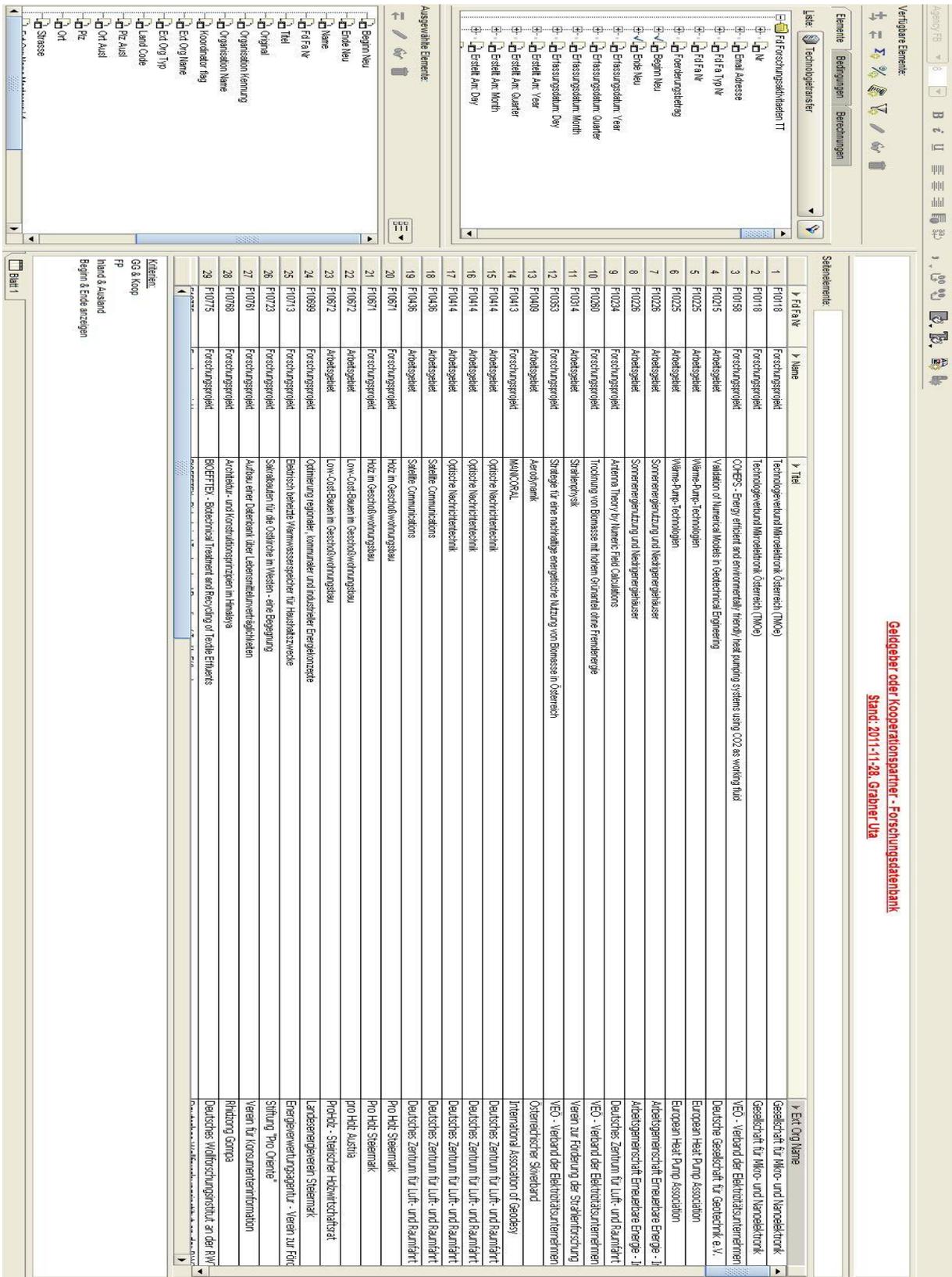


Abbildung 3.1: Discoverer User Interface

Der Discoverer ist deshalb sehr interessant, da es durch das Produkt möglich ist ohne Kenntnisse der Datenbankstrukturen und deren Konzepte, und ohne Fachwissen für online Systeme bzw. DWHs Abfragen mühelos zu erstellen. Die reinen Queries über die Daten

ermöglichen es Analysen und Berichte über multidimensionale Strukturen durchzuführen ohne SQL erlernt zu haben. Es stehen außerdem zahlreiche Standard-Templates zur Verfügung um das Erstellen solcher ad-hoc Abfragen schnell zu starten.

Eine weitere Vereinfachung aus Sicht des Endbenutzers ist der eingebaute Formel Editor. Damit ist es möglich durch einfaches Klicken bzw. Drag and Drop mathematisch sehr anspruchsvolle Formeln für das Analysieren der Daten zu erstellen. Damit wird auch das eigentliche Filtern der Metadaten sehr angenehm gestaltet, da man mit wenigen Mausklicken präzise und detaillierte Kriterien schaffen kann. Diese Formeln und Filter kann man als privilegierter Nutzer direkt online in den Endbenutzer-Layer abspeichern und somit anderen Anwendern zur Verfügung stellen. In Abbildung 3.2 bekommt man einen Eindruck vom Formel Editor des Discoverer anhand des Erstellens einer Gruppensumme.

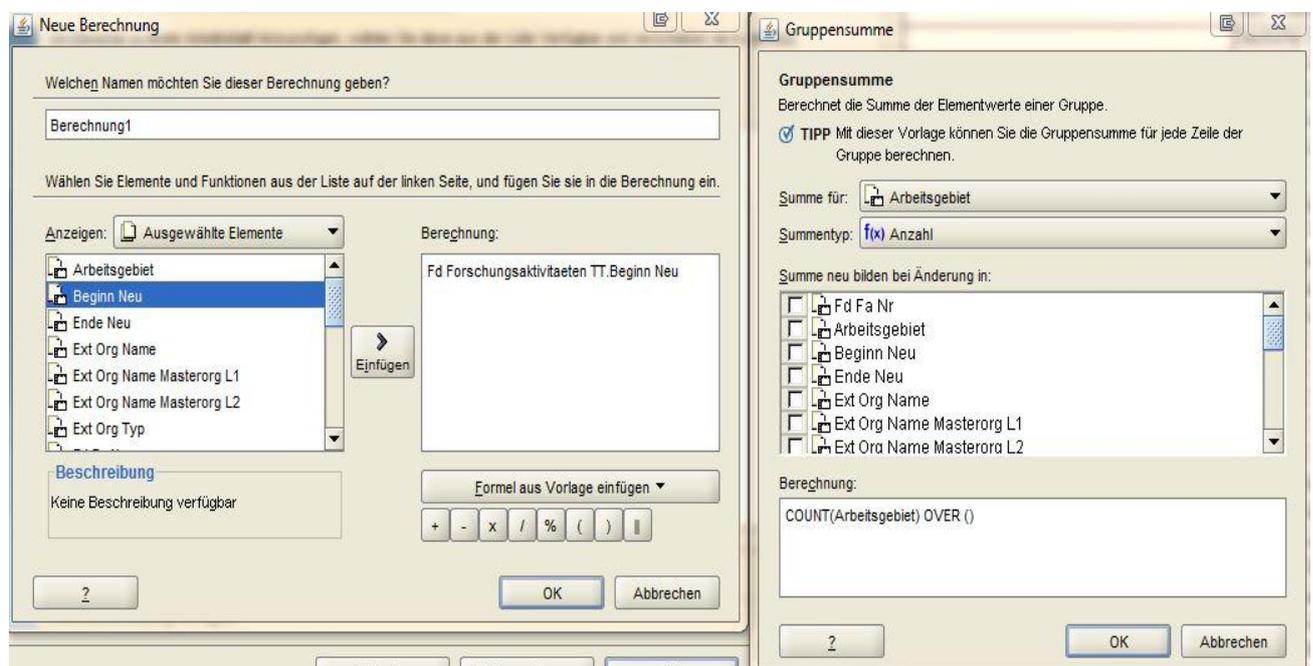


Abbildung 3.2: Formel Editor des Discoverer

Der Discoverer kann in allen Umgebungen eingesetzt werden und das mit allen Daten. Eine Integration in bestehende Enterprise Systeme ist ohne Probleme möglich, da das Tool mit allen populären Datenbanken kompatibel ist. In weiterer Folge wird heterogener Open Database Connectivity Zugriff auf weniger kritische Datenspeicher und eine einfache Schemadefinition sichergestellt, um lokale Kopien der offline Abfragen zu sichern.

Da es heutzutage aber bessere, schnellere und effizientere Lösungen gibt, um diese Funktionalitäten abzudecken, eignet sich der Discoverer nicht besonders für diese Thematik. Seitens Oracle wird die Software nicht mehr unterstützt, da die BI Suite der direkte Nachfolger des Discoverer ist. Es wird eine Lösung geboten, den Discoverer Datenbestand in das neue System zu migrieren, aber davon ist erfahrungsgemäß abzuraten, da es zu



Seit Version 9 des OWB wird Change Data Capture (CDC) angeboten, um die Zeit beim Extrahieren aus den Quellsystemen gering zu halten, indem man nur Veränderungen der Daten in das DWH lädt und somit die vollständige Extraktion der Daten unterbindet. Der eigentliche Zugriff auf diese Veränderungen wird per SQL realisiert [Bae03].

CDC wird hier synchron zur Verfügung gestellt und die Tabellen mit so genannten Triggern ausgestattet. Es gibt sogenannte Änderungstabellen, in die Änderungen an Daten kopiert werden. Somit ist es möglich nur relevante Daten abzuarbeiten und damit werden die weiteren Schritte beim ETL Prozess ressourcenschonender ausgeführt. Falls auf einem operativen System jedoch eine Vielzahl an schreibenden Operationen ausgeführt werden, könnte das für das System eine große Last mit sich tragen, darum kann es beim Verwenden von Triggern zu deutlich größeren Last bezüglich der Rechenzeit kommen. Mit Version 10 des OWB wird auch asynchrones CDC angeboten. Hier verzichtet man auf den Einsatz von Triggern und beobachtet nur die Logdateien.

Damit erspart man sich große Änderungen am System der Quelldaten. Änderungen werden, wenn sie erkannt werden, zum Beispiel durch Oracle Streams direkt in das DWH eingeschleust. Seitens Oracle wird zusätzlich eine schnellere Transformationsbearbeitung angeboten. Anstatt Ergebnisse aus Zwischenschritten zu speichern, gibt es nun Tabellenfunktionen, die Abhilfe schaffen. Diese Funktionen werden direkt in SQL Abfragen eingebunden und können in den meisten gängigen Programmiersprachen definiert werden. Der Output solcher Funktionen kann direkt in bestehende Tabellen eingefügt werden. Ohne solche Verfahren muss man jedes einzelne Ergebnis in einer Hilfstabelle zwischenspeichern, was dahingehend resultiert, dass man für n Zwischenschritte auch n Hilfstabellen benötigt. Wenn man aber Funktionen verwendet, kann man diese direkt aufeinanderfolgend ausführen. Es bildet sich somit eine Kette von Funktionen welche als Input jeweils den Output einer vorhergehenden Funktion verwenden. Mit diesem Verfahren kann eine sehr große Leistungssteigerung durchgeführt werden [Rit04]. Abbildung 3.4 zeigt den groben Unterschied eines Systems mit Tabellenfunktionen und eines ohne dieses Verfahren auf.

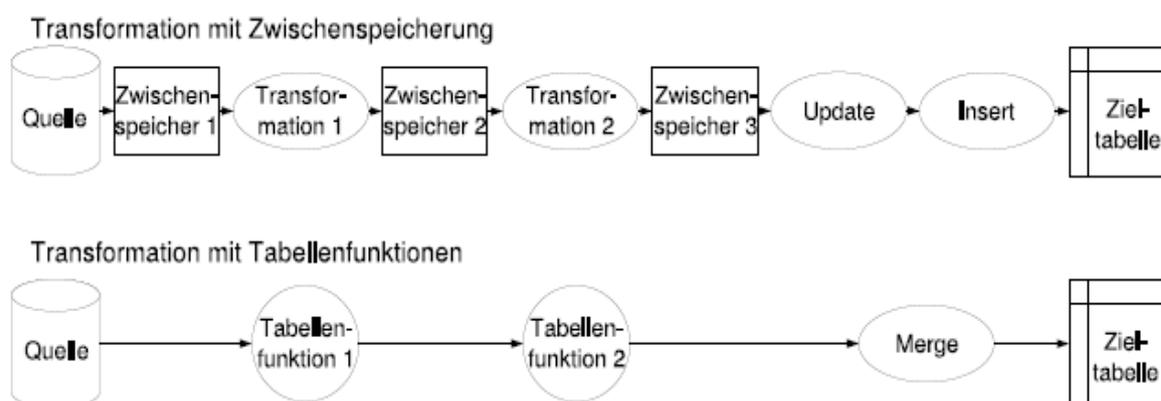


Abbildung 3.4: Vergleich eines Systems mit Tabellenfunktion und eines Systems ohne [Rit06a]

Eine weitere Beschleunigung hinsichtlich des Einfügens der Metadaten in das Data Warehouse, wird durch die eingeführte Merge Funktionalität erreicht. Falls man in einem DWH Daten korrigieren bzw. kleine Änderungen vornehmen muss, wird durch eine Operation des Einfügens sichergestellt, dass die eigentlichen Veränderungen am Datensatz richtig erkannt werden und unnötige Duplikate vermieden werden. Normalerweise wären dabei eine aneinandergefügte Reihe von Update und Insert Operationen notwendig. Das Merge Kommando fügt diese Schritte in nur einem zusammen.

Durch eine spezielle Funktion des DBMS kann man einzelne Tabellen partitionieren. Diese Partitionierung erfolgt aufgrund der Attributwerte. Somit wird eine Tabelle in mehrere einzelne Tabellen unterteilt und es ist möglich diese auf verschiedenen Datenträgern zu speichern. Damit lässt sich eine Leistungssteigerung hinsichtlich der Abfragegeschwindigkeit erzielen. Es können auch Real Time Partitionen durch den Einsatz von Partition Exchange Loading (PEL) erstellt werden. Die Vorgehensweise ist hier so, dass PEL eine temporäre Partition zum Arbeiten zur Verfügung stellt, die jederzeit mit einer beliebigen anderen Tabellenpartition ausgetauscht werden kann. In Oracle basierenden DBMS kann eine Tabelle in bis zu 64000 Partitionen aufgeteilt werden. Wenn man nun jeden Tag die Partition austauscht, kann man über 170 Jahre an Daten historisieren. Dies würde dem Ansatz eines Real Time DWH sehr nahe kommen und der Vorteil von PEL ist, dass beim Tauschen der Partitionen die Berechnung der Indizes nicht erneut erfolgen muss [Mac04].

Die oben genannten Funktionen sind die Hauptvorteile von OWB. Während des ETL Prozesses werden große Mengen an Metadaten mitgespeichert. Weiters werden vom OWB auch noch eventgesteuerte Transformationen eingeführt. Erst wenn ein spezieller Trigger ausgeführt wird, wie zum Beispiel eine Funktion durch das User Interface, werden auf die Virtual Targets die Transformationen ausgeführt. Damit ist es möglich auch anderen Applikationen die Techniken des OWB zur Verfügung zu stellen. In Abbildung 3.5 kann man die allgemeine Umsetzung solcher Techniken erkennen.

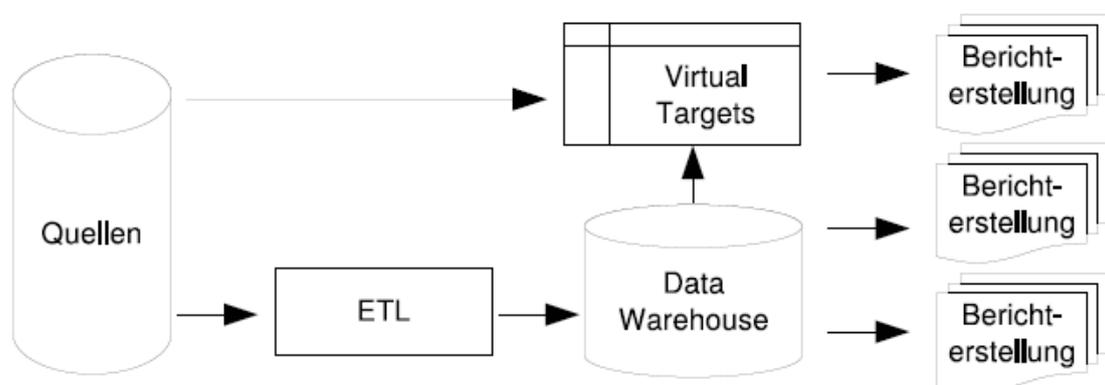


Abbildung 3.5: Anfragegesteuerte Transformation [Dij05]

### 3.5 Oracle Business Intelligence Enterprise Edition (OBIEE)

Die Daten, die mittels OWB in das Data Warehouse gebracht wurden, stehen nun für Analysetätigkeiten und Abfragen zur Verfügung. Um die Analysen für den Anwender so einfach und komfortabel wie möglich zu gestalten verfügt die TU Graz über ein spezielles Tool für diese Anforderungen. Mit OBIEE ist es möglich, die Daten aus dem DWH strukturiert auszulesen und multidimensionale Abfragen zu erstellen. Im Grunde basiert OBIEE auf folgenden Elementen:

- Ad-hoc Analysen für Reporting Zwecke
- Interaktive Dashboards
- Enterprise Reporting
- Scorecards
- Multidimensionales OLAP

Das eigentliche System besteht aus besteht aus Systemkomponenten und Java Komponenten. Als Systemkomponenten sind folgende fünf Hauptkomponenten zu nennen:

- Oracle BI Server
- Oracle BI Java Host
- Oracle BI Presentation Server
- Oracle BI Cluster Controller
- Oracle BI Scheduler

Diese Anwendungen sind in Java geschrieben und können direkt vom Java Applikationsserver der Oracle Weblogic ausgeführt werden. Damit man nun Analysen über die Metadaten durchführen kann, muss man diese zunächst vom DWH in ein eigens für die Bearbeitung eingerichtetes Repository transferieren. Genauer wird in Kapitel 4 beschrieben. In Abbildung 3.6 kann man ein beispielhaftes OBIEE Repository betrachten.

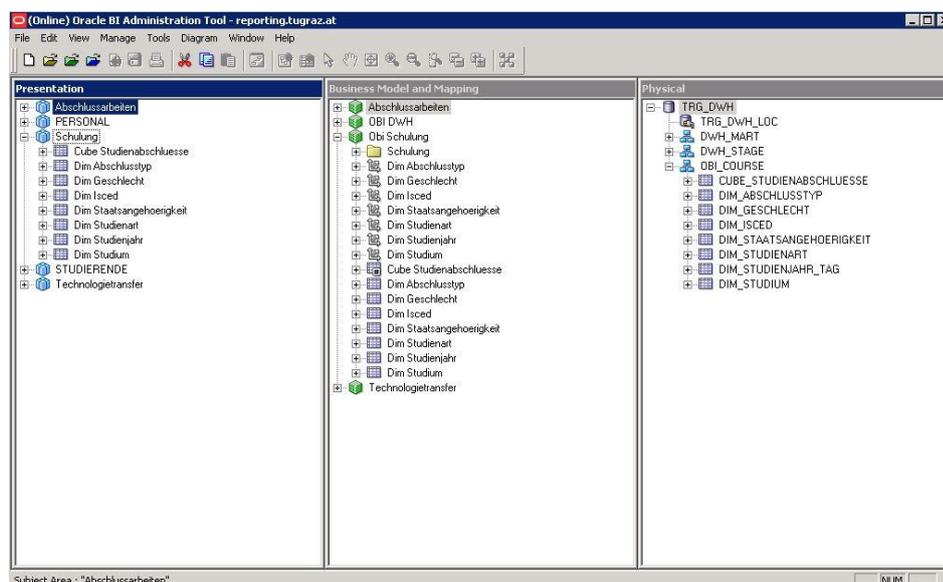


Abbildung 3.6: OBIEE Repository

Nach dem Einrichten des Repositories kann man zu den erstellten Themengebieten nun Analysen und Berichte erstellen. Dies erfolgt mittels der Oracle Weblogic und dem Tool BI Answers. Es steht eine Vielzahl von bereits eingebauten Funktionen zur Verfügung, welche man aber bei Bedarf verändern kann. Ein weiterer Vorteil gegenüber dem Discoverer ist es, dass man im Repository definieren kann welche Informationen des Business Modells in der Präsentationsschicht angezeigt werden und welche für den Anwender transparent sind. Eine kleine Abfrage mittels BI Answers findet man in Abbildung 3.7.

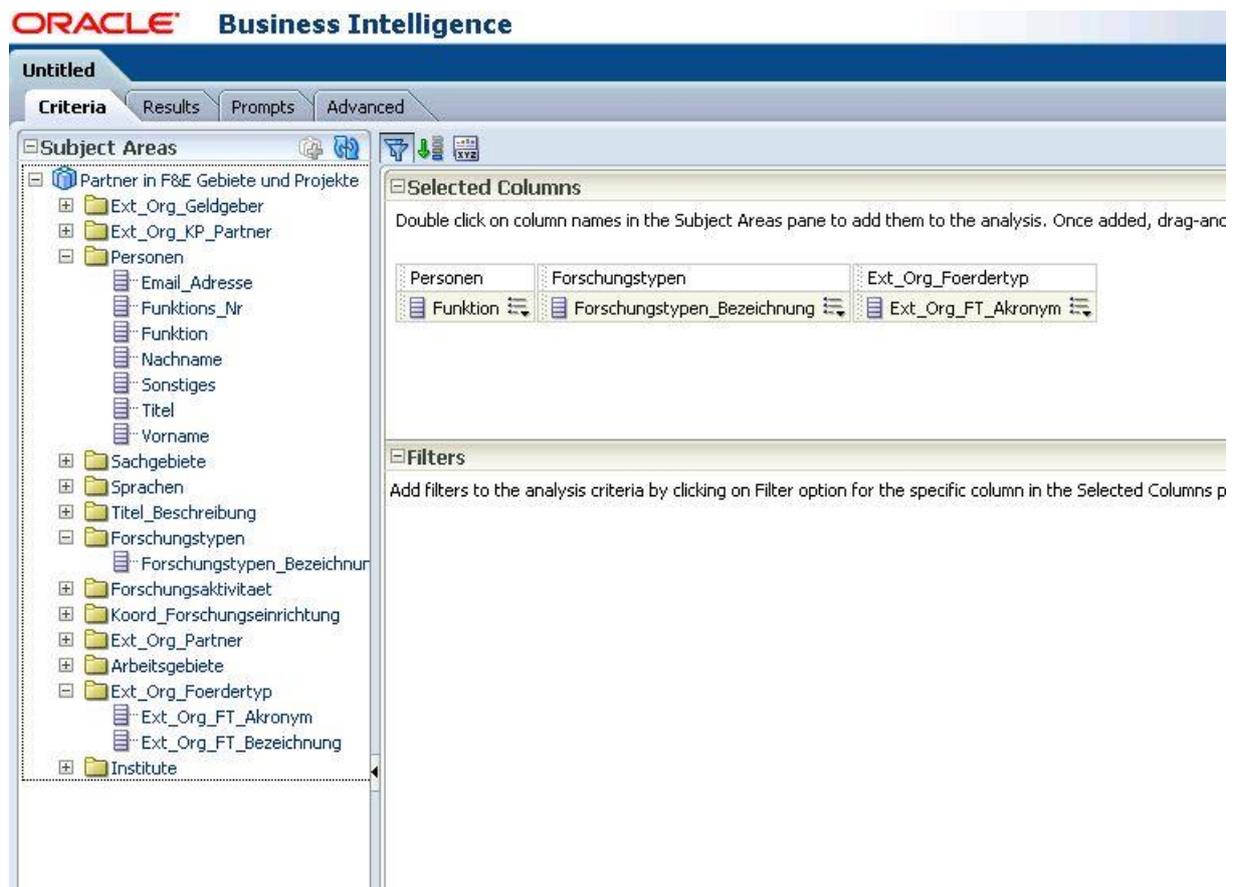


Abbildung 3.7: BI Answers mit Beispielabfrage

### 3.6 Vorhandene Daten

Teil der Bestandsaufnahme ist eine umfassende Analyse der vom F&T-Haus benötigten Ausgangsdaten, mit der die Fachabteilung ihre Berichte durchführt. Dazu wurden alle Quelldatenbanken des Discoverer in Betracht gezogen und die einzelnen Schemata auf deren Wichtigkeit und Verwendung geprüft. In den nun folgenden Sektionen werden die vom F&T-Haus verwendeten Datenbankschemen und Tabellen aufgezeigt. Für die verwendeten Gebiete Abschlussarbeiten und Technologietransfer stammen die Quelldaten aus den Datenbanken des Systems TU Graz Online.

### 3.6.1 Abschlussarbeiten

Im Bereich der Abschlussarbeiten gibt es eine Vielzahl von verwendeten Tabellen, die nötig sind um detaillierte Informationen zu bekommen und ein präzises Berichtswesen zu ermöglichen, welche in Tabelle 3.1 ersichtlich sind.

|                            |
|----------------------------|
| Laender 1                  |
| Adressen 1                 |
| Organisationen 1           |
| Personen 1                 |
| St Personen 1              |
| St Personen Alle V         |
| St Studien 1               |
| Fd Sachgebiete 1           |
| Fd Sprachen 1              |
| Dd Abs Pers Typen 1        |
| Dd Autoren 1               |
| Dd Personen 1              |
| Dd Personen Typen 1        |
| Dd Sachgebiete 1           |
| Dd Schlagworte 1           |
| Dd Titel Kurzfassungen 1   |
| Dd Werke 1                 |
| Dd Werke Aufträge          |
| Dd Werk Ext Organisationen |
| Dd Auft Branchen 1         |
| Dd Auftr Groessen 1        |
| Dd Auftr Standorte 1       |
| Dd Auftr Typen 1           |
| Dd Auftraggeber 1          |
| Pv Abs Typen 1             |
| Pu Abschluesse V           |
| Pu St Adressen V           |
| Zugangs Typen 1            |

|                              |
|------------------------------|
| Postleitzahlen Orte 1        |
| Pu Uni Studien V             |
| Externe Organisationen       |
| Externe Org Typen            |
| Externe Personen 2           |
| Extpers Anschriften 2        |
| Identitaeten 2               |
| Identitaeten aktiv           |
| Cis Abs Aktiv                |
| Cis_Abs_Aktiv_Pers_Adr       |
| Abs Werke                    |
| Abs Werke Ext Organisationen |
| Abs Autoren V                |
| Abs Betreuer V               |
| Abs Koop V                   |
| Abs Titel Kurzfassungen      |
| Abs Werke Sachgebiete V      |

Tabelle 3.1: IST-Stand der Tabellen für den Bereich Abschlussarbeiten

Diese Tabellen sind Bestandteil der vom Discoverer verwendeten Datenbank. Für eine Umsetzung in das BI Suite System müssen aber nicht alle Daten transferiert werden. Nach einer Analyse des F&T-Hauses werden die mit „Dd“ gekennzeichneten Tabellen für einen reibungsfreien Betrieb nicht benötigt. Hierbei handelt es sich um ein veraltetes Schema, welches nun nicht mehr genutzt wird. Historisch gesehen handelt es sich bei diesen Tabellen meist um sogenannte „User defined Views“. Das bedeutet, dass aus mehreren Datenbankschemen einzelne Metadaten ausgewählt werden und mittels Merge-Kommando zu einem View zusammengebaut werden. Für die Implementierung des neuen BI Ansatzes liegen die Daten noch auf den Datenbank Servern auf und somit müssen keine neuen Views angelegt werden sondern es ist möglich die bereits bestehenden zu verwenden. Da bei den Berichten, die seitens des F&T-Hauses erstellt werden aber nicht alle dieser Informationen benötigt werden, fallen darauffolgend noch einige Tabellen aus dem System um die Komplexität des neuen Systems einzuschränken. Detaillierte Informationen über die Implementierung findet man in Kapitel 4.

### 3.6.2 Technologietransfer

Auch im Bereich des Technologietransfers gibt es einige verwendete Datenbankschemen und daraus resultierende Tabellen, die für Analysen vom F&T-Haus verwendet werden, welche in Tabelle 3.2 aufgezeigt werden.

|                                      |
|--------------------------------------|
| Fd Forschungsaktivitaeten TT         |
| Fd Forschungsaktivitaet Typen TT     |
| Fd Fa Subtyp Inter TT                |
| Fd Fa Subtypen TT                    |
| Fd Forschungseinrichtungen Fa TT     |
| Fd Personen Fa TT                    |
| Fd Personen TT                       |
| Fd Pers Funktionen TT                |
| Fd Fa Sachgebiete TT                 |
| Fd Sachgebiete TT                    |
| Fd Schlagworte TT                    |
| Fd Titel Beschreib Fa TT             |
| Fd Sprachen TT                       |
| Fd Geldgeber TT                      |
| Fd Gg Foerdertyp TT                  |
| FD Kooperationspartner TT            |
| Organisationen TT                    |
| Externe Organisationen TT            |
| Externe Org Typen TT                 |
| Ext Org Foerdertypen TT              |
| Ext Org Kontaktpersonen TT           |
| Adressen TT                          |
| Postleitzahlen Orte TT               |
| Laender TT                           |
| Lander Gruppen TT                    |
| Laender Gruppen Laneder TT           |
| FD_Kooperationspartner und Geldgeber |
| FD_Arbeitsgebiete zu FProjekte       |

|                         |
|-------------------------|
| FD_Projektleiter        |
| FD_Personen_Sachgebiete |
| PU PERSONEN ORG ADR V 2 |

Tabelle 3.2: IST-Stand der Tabellen für den Bereich Technologietransfer

Diese Tabellen und Views werden für die Erstellung von Berichten im Bereich Technologietransfer verwendet. Auch all diese Daten sind noch auf dem Datenbank Server vorhanden und somit ist eine Weiterverwendung garantiert. Einige dieser Tabellen, die zum selben Kontext gehören lassen sich als Vereinfachung in eigenen Views zusammenfassen wie zum Beispiel: Laender TT, Laender Gruppen TT und Laender Gruppen Laender TT. Die zuletzt genannte Tabelle stellt hierbei eine Intersektion bzw. Überschneidung zwischen den Ländern und den Ländergruppen dar. Somit würde es sich eigenen diese Informationen der Einfachheit halber in einem eigenen View zusammen zu fassen. Damit ist auch das Implementieren der Dimensionen in der multidimensionalen DWH Lösung um einiges angenehmer als ein Arbeiten mit einer Vielzahl von komplex verbundenen Tabellen.

### 3.7 Vorhandene Abfragen

Da der Oracle Discoverer bereits mehrere Jahre im Einsatz ist, gibt es in der online Datenbank bereits eine große Anzahl von Berichten, die mit den Gebieten Abschlussarbeiten und Technologietransfer zu tun haben. An dieser Stelle werden nun die zwei größten und umfassendsten Analysen, die im Discoverer vorhanden sind, näher betrachtet und anhand dieser die Bestandsaufnahme weitergeführt. Anhand dieser beiden Abfragen soll nach der Implementierung der BI Suite Lösung ein qualitätssichernder Vergleich durchgeführt werden, um die Implementierung anschließend zu kontrollieren.

#### 3.7.1 Abschlussarbeiten - gr\_20111125\_Abschlussarbeit\_extOE\_alle

Der Zweck dieser Abfrage ist es, alle vorhandenen Abschlussarbeiten mit Kooperationspartnern aufzuzeigen. Diese Analyse wurde mit den Daten vom Stand 28. November 2011 durchgeführt. Die gewählten Daten für die Analyse sind folgende Informationen aus den Tabellen:

- Adressen 1
  - Strasse Hausnummer
  - Postleitzahlen Ausland
  - Ort Ausland
- Laender 1
  - Name
- Organisationen 1
  - Kennung

- Name
- Postleitzahlen Orte 1
  - Postleitzahl
  - Ort
- Pu Abschluesse V
  - Abschlussartname
  - Abschlussartkb
- Externe Organisationen
  - Name
  - Name Masterorg L1
  - Name Masterorg L2
- Externe Org Typen
  - Name
- Abs Titel Kurzfassungen
  - Titel
- Abs Autoren V
  - Vorname
  - Nachname
  - Titel
- Abs Betreuer V
  - Betreuer Typ Name
  - Nachname
  - Vorname
  - Titel
- Abs Werke
  - Nr
  - Erscheinungsjahr

Anhand dieser Informationen wird das Grundgerüst des späteren Berichtes gebildet. Die Ergebnisse werden nach dem Erscheinungsjahr der Arbeiten sortiert. Des Weiteren gibt es noch einige Kriterien bzw. Filter die auf die Analyse angewendet werden, welche in Abbildung 3.8 dargestellt werden.

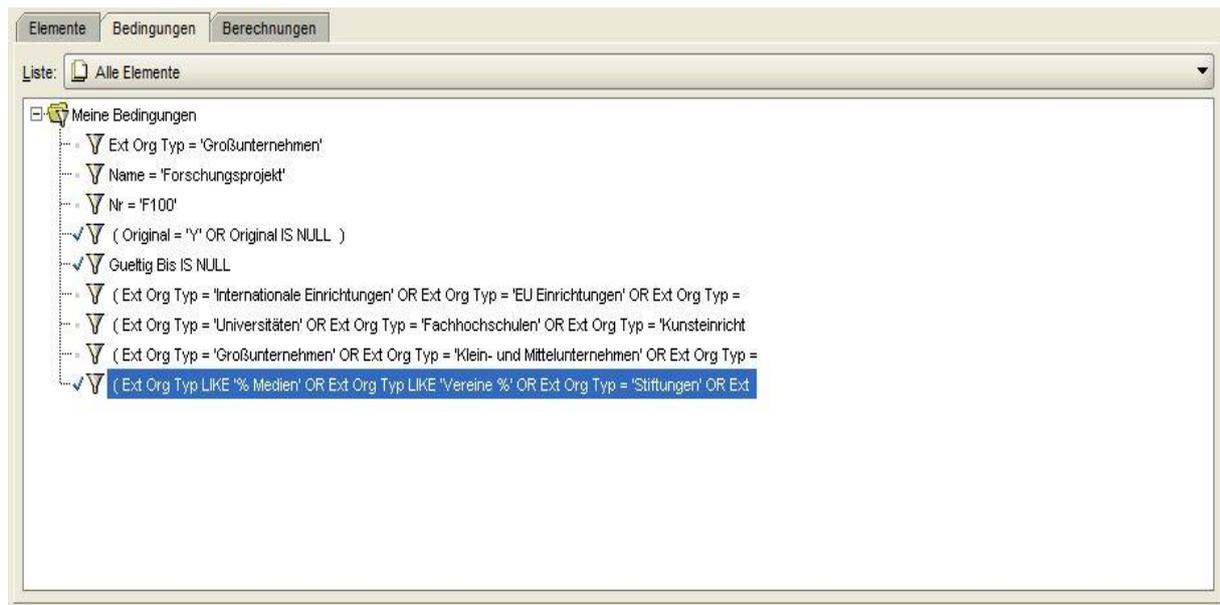


Abbildung 3.8: Filter für die Analyse der Abschlussarbeiten

- Es sollen nur jene Arbeiten angezeigt werden, bei denen eine externe Organisation vorhanden ist.
- Die Abschlusstyp Kurzbezeichnung soll entweder konform mit „M“ oder „P“ sein.
- Das Erscheinungsjahr wird auf Jahre zwischen 2004 und 2009 eingeschränkt.

Spezielle Berechnungen oder Formeln sind in dieser ad-hoc Analyse nicht verwendet worden. Diese Abfrage stellt in gewisser Hinsicht eine allumfassende Analyse des Abschlussarbeiten-Bereiches dar und mit dem Austausch der Filter bzw. Bedingungen kann man mithilfe dieser weit komplexere und nahezu alle Arten von Analysen tätigen.

### 3.7.2 Technologietransfer – gr\_20111125\_alle\_GG\_Koop\_Name\_Adresse\_Laufzeit

Diese Analyse wird verwendet um zum Stichtag 29. November 2011 alle Kooperationspartner und Geldgeber sowie deren Anschrift abzufragen. Es ist wichtig für die TU Graz von ihren Partnern Daten zu haben für spätere wirtschaftliche Zwecke zu haben. Die Informationen die aus den Datenbankschemen und Tabellen benötigt werden sind folgende:

- Fd Forschungsaktivitaeten TT
  - Beginn Neu
  - Ende Neu
- Fd Forschungsaktivitaet Typen TT
  - Name
- Fd Titel Beschreib Fa TT
  - Fd Fa Nr

- Titel
- Original
- Organisationen TT
  - Organisation Kennung
  - Organisation Name
- Fd Forschungseinrichtungen Fa TT
  - Koordinator Flag
- FD\_Kooperationspartner und Geldgeber
  - Ext Org Name
  - Ext Org Typ
  - Land Code
  - Plz ausl
  - Ort ausl
  - Plz
  - Ort
  - Strasse
  - Ext Org Masterorg L1
  - Ext Org Masterorg L2
  - Gueltig Bis
  - Typ
- FD\_Arbeitsgebiete zu FProjekte
  - Arbeitsgebiet

Anhand dieser Daten ist es möglich Auskunft über die Adressen der Kooperationspartner bzw. der Geldgeber zu kommen, die in weitere Folge vom F&T-Haus gebraucht werden um Zusammenhänge zwischen den einzelnen Partnern aufzuzeigen. Die Filter, die für die Analyse angewandt wurden, kann man in Abbildung 3.9 erkennen.



Abbildung 3.9: Filter für Technologietransfer

- Das Projekt muss aktuell noch laufen (Gueltig = null).
- Um Duplikate auszublenden muss das Originalflag auf „Y“ gesetzt sein wenn es mehrere Sprachversionen des Titels gibt. Ansonsten reicht ein Setzen auf „null“.
- Der Typ der Kooperationspartner/Geldgeber muss einer der folgenden sein:
  - Medien
  - Vereine
  - Stiftungen
  - Gesetzliche Interessensvertretungen
  - Nicht bekannt
  - Andere Organisationen

Auch hier gibt es keine speziellen Formeln oder Berechnungen. Bei dieser Abfrage handelt es sich um eine Hauptanalyse aus dem Bereich Technologietransfer, aus dem alle weiteren für den Bereich relevanten Analysen abgeleitet werden können.

## 4 Durchführung und Evaluierung der Lösungsansätze

In diesem Kapitel werden nun im Detail die beiden Lösungsansätze für die Aufbereitung der Daten des F&T-Hauses aufgezeigt. Es wird zunächst auf die Implementierung im Sinne des relationalen Ansatzes eingegangen. Im Anschluss daran findet man die Entwicklung des multidimensionalen Ansatzes mittels DWH. Schlussendlich werden die Erkenntnisse und Ergebnisse der beiden Methoden miteinander verglichen, um Stärken und Schwächen der jeweiligen Ansätze aufzuzeigen.

### 4.1 Implementierung des relationalen Ansatzes

Ziel des relationalen Ansatzes ist es, durch manuelle Modifikation der Quelldaten ein Gebilde zu schaffen, welches ohne eigentlichen DWH Charakter die Daten der Ursprungssysteme abbilden und verarbeiten kann. Hierzu muss man die Bestandsaufnahme aus Kapitel 3.6 heranziehen. Um die im Kapitel 3.6 genannten Informationen und Tabellen in den relationalen Ansatz zu überführen, muss im Quellsystem TU Graz Online analysiert werden, welche Tabellen und Views verwendet werden sollen. Nach einer genauen Betrachtung werden für den Bereich Abschlussarbeiten und den Bereich Technologietransfer die in den Tabellen 4.1 und 4.2 verwendeten Views herangezogen.

|                           |
|---------------------------|
| TUG_New_Laender           |
| TUG_New_Adressen          |
| TUG_New_Plz_Orte          |
| TUG_New_Externe_Org       |
| TUG_New_Externe_Org_Typen |
| TUG_New_Organisationen    |
| Co_Abs_Autoren            |
| Stud_Pu_Abschluesse       |
| Stud_In_St_Adressen       |
| St_Person                 |
| Co_Abs_Betreuer           |
| TUG_New_Externe_Personen  |
| Co_Abs_Werke              |
| PV_Abs_Typen              |
| Fodok_Fd_Sprachen         |

Tabelle 4.1: Verwendete Views für den Bereich Abschlussarbeiten

Diese Views stammen vorwiegend aus den Datenbankschemen „Co\_Abs“, „Stud“ und „TUG\_New“, mit Ausnahme des Views Fodok\_fd\_Sprachen welcher aus dem Schema „Fodok“ stammt.

|                                     |
|-------------------------------------|
| TUG_New_Ext_Org_Foerdertyp          |
| Fodok_FD_GG_Foerdertyp              |
| Fodok_Fd_Geldgeber                  |
| TUG_New_Organisationsen             |
| Fodok_Koop_Partner_Geldgeber        |
| Fodok_Fd_Fa_Sachgebiete             |
| Fodok_Fd_Sachgebiete                |
| Fodok_Fd_Kooperationspartner        |
| Fodok_Fd_Forschungseinrichtungen_Fa |
| Fodok_Forschungsaktivitaeten        |
| Fodok_Arb_Gebiete                   |
| Fodok_Forschungseintr_Typen         |
| Fodok_Fd_Personen_Fa                |
| Fodok_Fd_Personen                   |
| Fodok_Titel_Beschreibungen          |
| Fodok_Fd_Sprachen                   |

Tabelle 4.2: Verwendete Views für den Bereich Technologietransfer

Im Bereich Technologietransfer ist das vorherrschende Datenbankschema „Fodok“, da dies sämtliche forschungsrelevanten Daten beinhaltet.

#### 4.1.1 Importieren der Daten in das Data Warehouse

Nachdem die Grunddaten erfasst wurden, muss man diese, um in weiterer Folge damit Berechnungen und Abfragen tätigen zu können mittels ETL Prozess in das DWH transferieren. Dies geschieht mit Hilfe des Oracle Warehouse Builders.

Um die entsprechenden Quellen zu definieren, muss man mittels der Importfunktion des OWB einen Metadatenimport der Daten des TU Graz Online durchführen. Damit werden die benötigten Views und Tabellen für das ETL Tool sichtbar. In Abbildung 4.1 kann man einen Ausschnitt eines solchen Import-Vorganges betrachten.

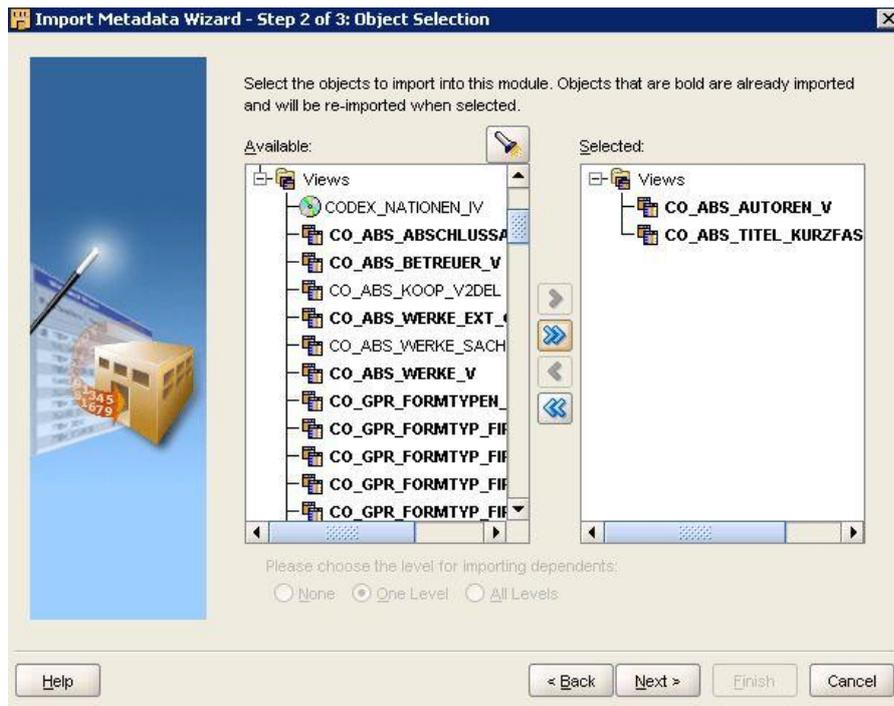


Abbildung 4.1: Import-Tool des OWB

Nach einem erfolgreichen Import scheinen die Metadaten geordnet nach Tabellen und Views im Kontextmenü des OWB auf. Um diese Informationen nun weitergehend im Warehouse Builder nutzen zu können, muss man sogenannte Mappings kreieren. Mit Hilfe solcher Mappings werden die importierten Metadaten in virtuelle, vom OWB genutzte, Tabellen geladen. Dies hat den Vorteil, dass die Metadaten gekapselt und getrennt von der Quellsystem-Ebene werden. Ein weiterer Pluspunkt ist eine schnellere Rechenleistung, da es beim Übertragen der Metadaten zu einem gewissen Overhead kommen kann. Somit ist das Verwenden solcher virtueller Tabellenstrukturen ein großer Geschwindigkeitsvorteil. Für jede einzelne zu verwendende Tabelle muss nun ein Mapping in das DWH System durchgeführt werden. In Tabelle 4.3 und 4.4 werden die Mappings in den Stage Bereich des DWH aufgelistet.

|                              |
|------------------------------|
| MAP_TO_CO_ABS_AUTOREN        |
| MAP_TO_CO_ABS_BETREUER       |
| MAP_TO_CO_ABS_TITEL_KURZFASS |
| MAP_TO_CO_ABS_WERKE          |
| MAP_TO_FODOK_FD_SPRACHEN     |
| MAP_TO_PV_ABS_TYPEN          |
| MAP_TO_STUD_IN_ST_ADRESSEN   |
| MAP_TO_STUD_PU_ABSCHLUESSE   |
| MAP_TO_STUD_ST_PERSON        |

|                                 |
|---------------------------------|
| MAP_TO_TUG_NEW_ADRESSEN         |
| MAP_TO_TUG_NEW_EXTERNE_ORG      |
| MAP_TO_TUG_NEW_EXTERNE_ORG_TYP  |
| MAP_TO_TUG_NEW_EXTERNE_PERSONEN |
| MAP_TO_TUG_NEW_LAENDER          |
| MAP_TO_TUG_NEW_ORGANISATIONEN   |
| MAP_TO_TUG_NEW_PLZ_ORTE         |

Tabelle 4.3: Mappings für den Stage Bereich für Abschlussarbeiten

|                               |
|-------------------------------|
| TO_FODOK_FD_FA_SACHGEBIETE    |
| TO_FODOK_FD_GELDGEBER         |
| TO_FODOK_FD_GG_FOERDERTYP     |
| TO_FODOK_FD_KOOPPARTNER       |
| TO_FODOK_FD_PERSONEN          |
| TO_FODOK_FD_PERSONEN_FA       |
| TO_FODOK_FD_SACHGEBIETE       |
| TO_FODOK_FD_SPRACHEN          |
| TO_FODOK_FD_TITEL_BESCHREI    |
| TO_FODOK_FD_FORSCHUNGS_TYPEN  |
| TO_FODOK_FD_FORSCHUNGSAKTIV   |
| TO_FODOK_FD_FORSCHUNGSEINR_FA |
| TO_FODOK_FD_KOOP_PARTNER      |
| TO_FODOK_FD_ARB_GEBIETE       |
| TO_TUG_NEW_EXT_ORG_FOERDERTYP |
| TO_TUG_NEW_ORGANISATIONEN     |

Tabelle 4.4: Mappings für den Stage Bereich für Technologietransfer

Durch Ausführung dieser Mappings werden die Informationen in den Metadaten Views für das DWH zur Verfügung gestellt. Ein solches Mapping wird in Abbildung 4.2 beschrieben. Es handelt sich hierbei um das Mapping für den View „Fodok:\_fd\_Personen“. Hier kann man erkennen, dass die Ursprungsinformationen aus dem View eins zu eins in die virtuelle Tabelle geladen werden. Zusätzlich wird ein neues Feld in die Tabelle Namens „SDAT“ eingeführt. Darin wird der Zeitpunkt der Beladung (Systemzeit) festgehalten und steht somit für weitere Bearbeitungsschritte zur Verfügung.

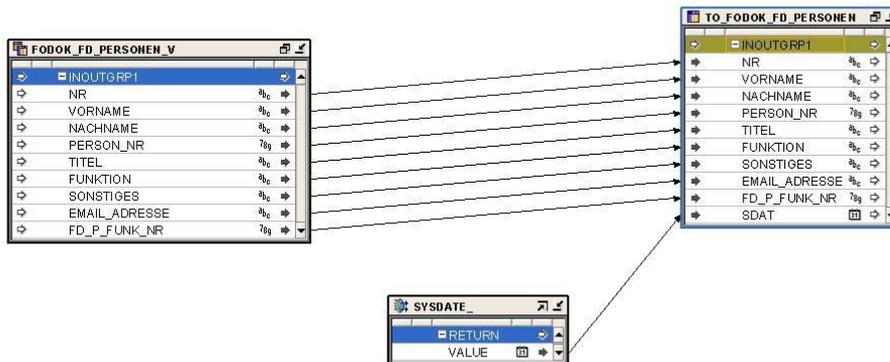


Abbildung 4.2: Mapping für den View "Fodok\_Fd\_Personen"

Ein wichtiges Kriterium bei der Beladung im Stage Bereich ist, dass die Metadaten mittels Truncate/Insert eingefügt werden. Damit wird sichergestellt, dass bei einer erneuten Beladung und Ausführung des Mappings die vorhandenen Daten in der virtuellen Zieltabelle verworfen werden und nur die neuen Daten Zugang in das DWH finden. Nach dem Erstellen der Mappings müssen diese noch ausgeführt werden. Um eine effiziente Arbeit mit aktuellen Daten zu garantieren, werden so genannte Prozessflüsse erstellt. Damit ist es möglich mehrere Mappings hintereinander auszuführen und bei möglicherweise auftretenden Komplikationen oder Fehlern einen Fehlerbericht an den Anwender zu übermitteln. Einen Aufbau eines beispielhaften Prozessflusses kann man in Abbildung 4.3 betrachten. Hier wird je nach Status der Exekution ein Bericht an den Initiator des Events geschickt.

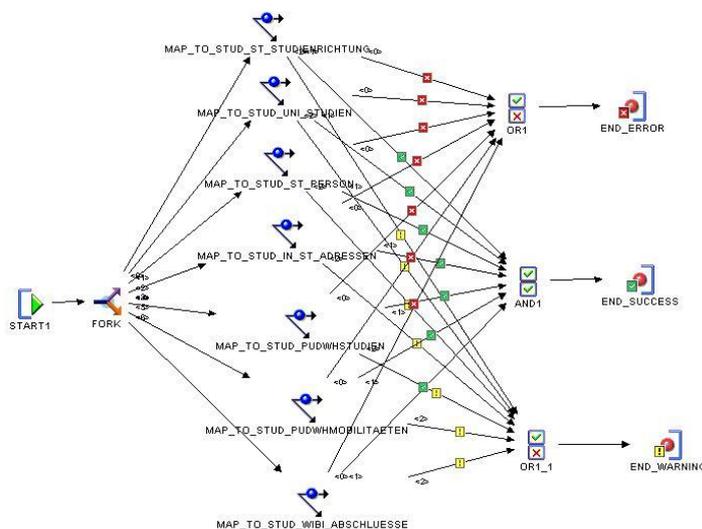


Abbildung 4.3: Beispielhafter Prozessfluss

Das Ausführen von Prozessflüssen ist jederzeit ohne Limitierungen automatisch möglich. In der implementierten Lösung wurde für die Ausführung ein Zeitpunkt um 01:00 nachts gewählt, da hier nahezu keine Anwender im DWH Betrieb anzufinden sind. Wie in Kapitel 2 erklärt kann es sein, dass während der Rechenzeit der Prozessflüsse andere Applikationen vom System gesperrt werden, was mit diesem Ausführungszeitpunkt umgangen wird.

#### 4.1.2 Aufbereiten der Daten mit OBIEE

Nach der Abarbeitung der Prozessflüsse sind die Daten im DWH verfügbar. Damit man weitergehend im OBIEE Repository auf die Daten zugreifen kann, muss man spezielle Rechte für den OBIEE setzen. Das Repository muss über „SELECT“ Rechte zum Auswählen der Daten aus dem DWH verfügen. In Abbildung 4.4 ist eine solche Rechteübergabe dargestellt.

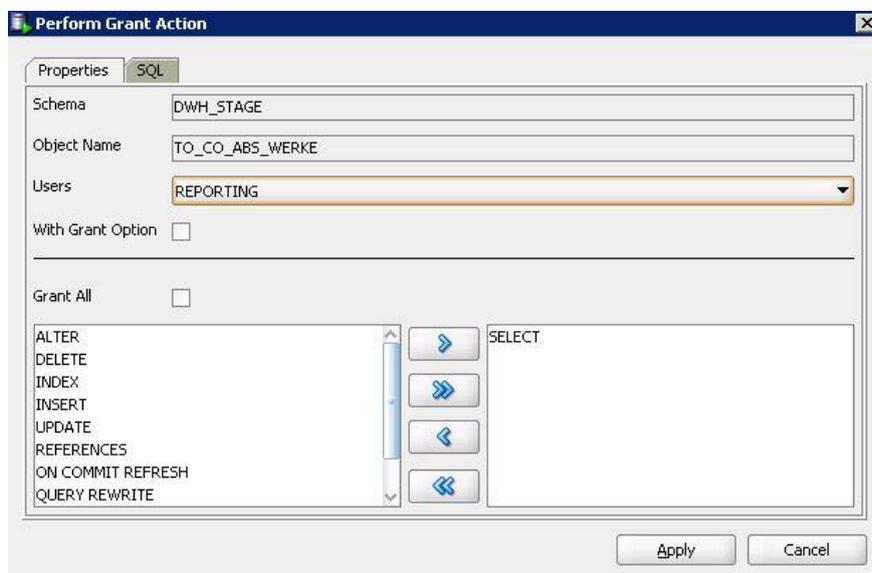


Abbildung 4.4: Setzen der SELECT Rechte für das OBIEE Repository

In den vorangegangenen Schritten wurden lediglich die Metadaten aus dem Quellsystem extrahiert und in den Stage Bereich des DWH transferiert. Um eine relationale Lösung zu implementieren muss man nun anhand der physischen Basis der Daten Verknüpfungen mit den einzelnen Tabellen herstellen.

Da eine schnelle Verbindung der Tabellen der einzelnen Themengebiete nicht möglich ist muss man sogenannte Faktentabellen herstellen. Diese Tabellen wären wie in Kapitel 2.3.1 beschrieben im Star Schema das zentrale Element und somit eine Schnittstelle aller Daten. Für die Bereiche Abschlussarbeiten und Technologietransfer wurde jeweils eine Faktentabelle kreiert.

- FODOK\_TECHNOLOGIETRANSFER
  - FD\_FA\_NR
  - GG\_EXT\_ORG\_NR
  - KP\_EXT\_ORG\_NR
  - KP\_NR
  - ORG\_NR
  -
- CO\_ABS\_ABSCHLUSSARBEITEN
  - AUTOR\_NR
  - BETREUER\_NR
  - EXT\_ORG\_NR
  - THESIS\_NR

Diese beiden Tabellen sind die zentralen Tabellen der Implementierung und vom Business Layer des OBIEE aus gesehen Faktentabellen. Diese beiden Tabellen müssen dieselben Prozesse durchlaufen, wie bereits oben angeführt, um für das Repository sichtbar zu werden.

Nun müssen die DWH Daten der Stage Ebene in das Repository geladen werden. Realisiert wird diese Transformation durch ein spezielles Import Tool des OBIEE welches in Abbildung 4.5 ersichtlich ist.

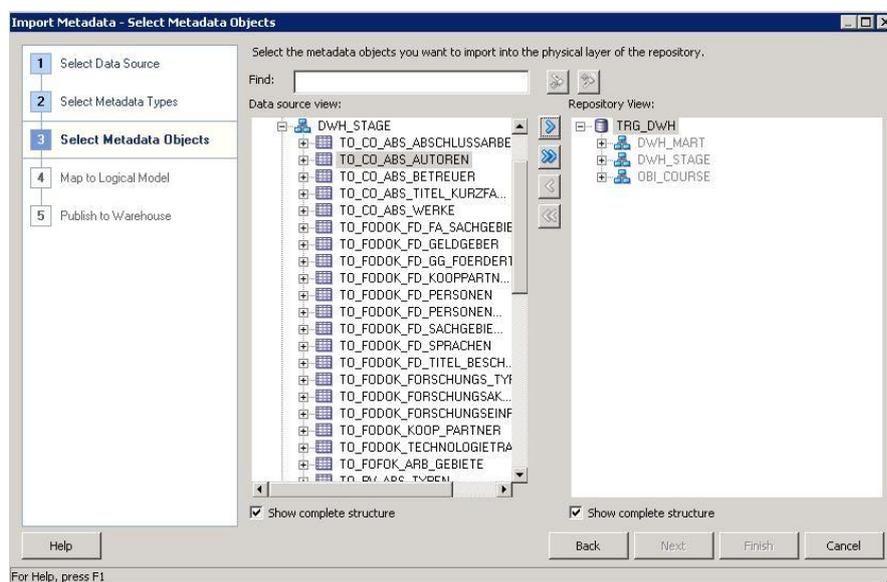


Abbildung 4.5: Metadatenimport mittels OBIEE Import-Tool

Nach dem Import der Metadaten kann man mit der Modellierung des relationalen Ansatzes beginnen. Um die Verbindungen oder Joins zwischen den einzelnen Tabellen richtig herzustellen, muss auf jeden Fall das Quellsystem einschlägig analysiert und verstanden worden sein.

Im Bereich der Abschlussarbeiten beginnt die Modellierung mit der zentralen Faktentabelle, die eine Kombination aus Daten in allen ausgeprägten Formen der Möglichkeiten mit sich bringt. Ausgehend davon werden für die Faktentabelle folgende Joins modelliert, die in Tabelle 4.5 aufgezeigt werden.

|                                |   |
|--------------------------------|---|
| TO_CO_ABS_AUTOREN (NR)         | TO_CO_ABS_ABSCHLUSSARBEITEN (AUTOR_NR)    |
| TO_CO_ABS_BETREUER (NR)        | TO_CO_ABS_ABSCHLUSSARBEITEN (BETREUER_NR) |
| TO_TUG_NEW_EXTERNE_ORG (NR)    | TO_CO_ABS_ABSCHLUSSARBEITEN (EXT_ORG_NR)  |
| TO_TUG_NEW_ORGANISATIONEN (NR) | TO_CO_ABS_ABSCHLUSSARBEITEN (ORG_NR)      |
| TO_CO_ABS_WERKE (NR)           | TO_CO_ABS_ABSCHLUSSARBEITEN (THESES_NR)   |

Tabelle 4.5:Join Modellierungen der Faktentabelle

Die Modellierung der Joins wird direkt am Physical-Layer des OBIEE vorgenommen. Damit werden diese initiierten Verbindungen in späterer Folge direkt in das Business Modell und den Präsentation Layer übernommen. Eine Join Modellierung mittels OBIEE wird in Abbildung 4.6 dargestellt.

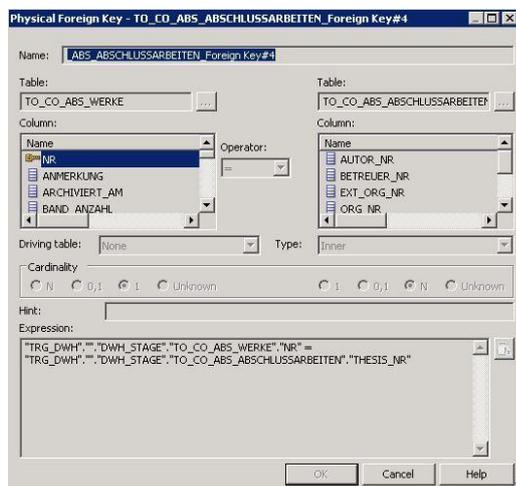


Abbildung 4.6: Modellierung eines Joins im OBIEE

In Bezug auf die Anforderungen des F&T-Hauses müssen nun ausgehend von der Faktentabelle alle weiteren benötigten Tabellen miteinander in Relation gebracht werden. Die hierfür notwendigen Joins werden in Tabelle 4.6 zusammengefasst.

|   |   |
|---|---|
| TO_STUD_PU_ABSCHLUESSE<br>(ST_STUDIUM_NR) | TO_CO_ABS_AUTOREN (ST_STUDIUM_NR)         |
| TO_STUD_ST_PERSON (NR)                    | TO_CO_ABS_AUTOREN (ST_PERSON_NR)          |
| TO_STUD_IN_ST_ADRESSEN(ST_PERSON_NR)      | TO_CO_ABS_AUTOREN(ST_PERSON_NR)           |
| TO_TUG_NEW_EXTERNE_PERSONEN (NR)          | TO_CO_ABS_BETREUER (PERSON_NR)            |
| TO_FODOK_FD_SPRACHEN (NR)                 | TO_CO_ABS_TITEL_KURZFASS<br>(SPRACHE_NR)  |
| TO_CO_ABS_TITEL_KURZFASS<br>(ABS_WERK_NR) | TO_CO_ABS_WERKE (NR)                      |
| TO_PV_ABS_TYPEN (NR)                      | TO_CO_ABS_WERKE (PV_ABS_TYP_NR)           |
| TO_TUG_NEW_PLZ_ORTE (NR)                  | TO_TUG_NEW_ADRESSEN (PLZ_NR)              |
| TO_TUG_NEW_LAENDER (NR)                   | TO_TUG_NEW_ADRESSEN (LAND_NR)             |
| TO_TUG_NEW_EXTERNE_ORG_TYP (NR)           | TO_TUG_NEW_EXTERNE_ORG<br>(EXT_ORGTYP_NR) |
| TO_TUG_NEW_ADRESSEN (NR)                  | TO_TUG_NEW_EXTERNE_ORG (ADR_NR)           |

Tabelle 4.6: Weitere Joins für Abschlussarbeiten

Für den Bereich Technologietransfer ist im Grunde genommen derselbe Ablauf wie für das Themengebiet der Abschlussarbeiten notwendig. Alle benötigten Joins ausgehend von der Faktentabelle werden in Tabelle 4.7 dargestellt.

|  |   |
|--|---|
| TO_FODOK_FD_KOOPPARTNER (NR)                 | TO_FODOK_TECHNOLOGIETRANSFER<br>(KP_NR)       |
| TO_FODOK_FD_GELDGEBER (NR)                   | TO_FODOK_TECHNOLOGIETRANSFER<br>(GG_NR)       |
| TO_TUG_NEW_ORGANISATIONEN (NR)               | TO_FODOK_TECHNOLOGIETRANSFER<br>(ORG_NR)      |
| TO_FODOK_FORSCHUNGSAKTIV (NR)                | TO_FODOK_TECHNOLOGIETRANSFER<br>(FD_FA_NR)    |
| TO_FODOK_KOOP_PARTNER (FD_FA_NR)             | TO_FODOK_TECHNOLOGIETRANSFER<br>(FD_FA_NR)    |
| TO_FODOK_FD_FA_SACHGEBIETE<br>(FD_FA_NR)     | TO_FODOK_TECHNOLOGIETRANSFER<br>(FD_FA_NR)    |
| TO_FODOK_FD_SACHGEBIETE (NR)                 | TO_FODOK_FD_FA_SACHGEBIETE<br>(FD_SACHGEB_NR) |
| TO_TUG_NEW_EXTERNE_ORG (NR)                  | TO_FODOK_FD_GELDGEBER (EXT_ORG_NR)            |
| TO_FODOK_FD_GG_FOERDERTYP<br>(FD_GELDGEB_NR) | TO_FODOK_FD_GELDGEBER (NR)                    |

|                                       |  |
|---------------------------------------|--|
| TO_TUG_NEW_EXT_ORG_FOERDERTYP (NR)    | TO_FODOK_FD_GG_FOERDERTYP (EXT_ORG_FT_NR)  |
| TO_FODOK_FD_PERSONEN (NR)             | TO_FODOK_FD_PERSONEN_FA (FD_PERSON_NR)     |
| TO_FODOK_FD_SPRACHEN (NR)             | TO_FODOK_FD_TITEL_BESCHREI (FD_SPRACHE_NR) |
| TO_FODOK_FORSCHUNGS_TYPEN (NR)        | TO_FODOK_FORSCHUNGSAKTIV (FD_FA_TYP_NR)    |
| TO_FODOK_FD_TITEL_BESCHREI (FD_FA_NR) | TO_FODOK_FORSCHUNGSAKTIV (NR)              |
| TO_FODOK_FD_PERSONEN_FA (FD_FA_NR)    | TO_FODOK_FORSCHUNGSAKTIV (NR)              |
| TO_FODOK_FORSCHUNGSEINR_FA (FD_FA_NR) | TO_FODOK_FORSCHUNGSAKTIV (NR)              |
| TO_FODOK_ARB_GEBIETE (FD_FA_NR)       | TO_FODOK_FORSCHUNGSAKTIV (NR)              |

Tabelle 4.7: Joins für den Bereich Technologietransfer

Mit Hilfe dieser Verbindungen wird der relationale Ansatz der Implementierung aufgebaut. Nach der Modellierung des physischen Layers wird der Kontext nun getrennt nach Themengebiet in die Business Modelle übertragen. Daraus resultieren jeweils ein Business Modell für den Bereich Abschlussarbeiten und Technologietransfer. Durch die Bereitstellung der Business Daten im Präsentationsbereich des Repository werden die Metadaten nun für die Weblogic sichtbar und es ist nun möglich Analysen und Abfragen zu den Themengebieten durchzuführen.

## 4.2 Implementierung des multidimensionalen Ansatzes

Bei der Durchführung des multidimensionalen Lösungsansatzes zur Analyse der F&T Daten muss man wie in Kapitel 4.1 beschrieben ein Importieren der benötigten Metadaten in die Stage Ebene des DWH durchführen. Die hierbei benötigten Daten aus dem Quellsystem sind analog zu denen, die in der relationalen Lösung gebraucht wurden. Nach der Ausführung der Mappings unterscheidet sich die Vorgehensweise der multidimensionalen Lösung im Vergleich zur relationalen.

### 4.2.1 Aufbereitung der Daten für die Core Ebene

Die virtuellen Tabellen im Stage Bereich werden im nächsten Schritt zu logisch verknüpften Dimensionen zusammengefasst und in die Core Ebene des DWH gebracht. Hierzu muss man zunächst überlegen, welche Dimensionen für eine weitere Strukturierung der Lösung notwendig sind.

Die gewählten Dimensionen für die Themengebiete Abschlussarbeiten und Technologietransfer werden in den Tabellen 4.8 und 4.9 aufgelistet.

|                          |
|--------------------------|
| D_AUTOREN                |
| D_BETREUER               |
| D_EXTERNE_ORGANISATIONEN |
| D_ORGANISATIONEN         |
| D_WERKE                  |

Tabelle 4.8: Dimensionen für den Bereich Abschlussarbeiten

|                        |
|------------------------|
| D_FORSCHUNGSAKTIVITAET |
| D_GELDGEBER            |
| D_KOOP_PARTNER         |
| D_ORGANISATION         |
| D_SACHGEBIETE          |
| D_PARTNER              |

Tabelle 4.9: Dimensionen für den Bereich Technologietransfer

Diese Dimensionen fassen alle elementaren Sachgebiete der Themenbereiche zusammen und somit ist es für spätere umfassende Analysen möglich einen sehr hohen Detaillevel zu erreichen. Es werden aber nicht mehr alle Informationen aus den Stage Tabellen benötigt. Deshalb ist eine zukunftsorientierte Planung sehr wichtig, die bei der Eingrenzung der benötigten Informationen hilfreich ist. Die Dimensionstabellen werden im Core Bereich des DWH erstellt. Somit ist es wiederum notwendig die Daten der Stage Ebene korrekt auf die virtuellen Dimensionstabellen der Core Ebene zu mappen. In Abbildung 4.7 ist ein Dimensionsmapping für die Dimension Organisation zu sehen, da dieses Mapping in beiden Themenbereichen genutzt wird.

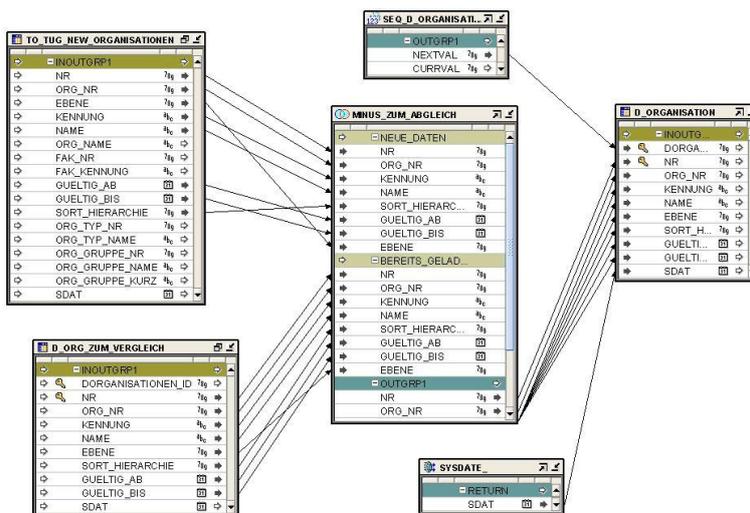


Abbildung 4.7: Mapping für die Dimension Organisation

Auf eine Dimension werden alle aus den Ursprungstabellen relevanten Informationen verlinkt, sodass diese einen Zusammenschluss bilden. In weiterer Folge ist es wichtig in dieser Phase für die Dimensionstabelle einen eindeutigen Primary Key zu definieren, der in weiterer Folge noch gebraucht wird. Auch Unique Keys müssen unbedingt definiert werden, um die Eindeutigkeit der Datensätze festzustellen (Kapitel 2.3.1). Da Prozessflüsse meist täglich durchgeführt werden und somit täglich die Daten aktualisiert werden, wäre es aus Performance Gründen nicht empfehlenswert alle Daten einer Dimensionstabelle einfach zu löschen und anschließend neu einzufügen. Die Beladung der Dimensionen basiert auf Update/Insert. Zusätzlich wird bei einer Beladung mit den bereits bestehenden Daten ein so genannter „Minus-Abgleich“ durchgeführt. Dies bedeutet, dass aus den neuen und alten Daten eine Schnittmenge gebildet wird und nur die Daten außerhalb der Schnittmenge neu in das DWH eingefügt werden. Somit erspart man sich sehr viel Rechenzeit, da die Ausführung eines Dimensionsmappings sehr viel Zeit in Anspruch nehmen kann.

Eine ausführliche Auflistung der verwendeten Stage-Tabellen pro Dimension ist anschließend zu finden:

- D\_AUTOREN
  - TO\_CO\_ABS\_AUTOREN
  - TO\_STUD\_PU\_ABSCHLUESSE
  - TO\_STUD\_IN\_ST\_ADRESSEN
- D\_BETREUER
  - TO\_CO\_ABS\_BETREUER
  - TO\_TUG\_NEW\_EXTERNE\_PERSONEN
- D\_EXTERNE\_ORGANISATIONEN
  - TO\_TUG\_NEW\_LAENDER
  - TO\_TUG\_NEW\_ADRESSEN

- TO\_TUG\_NEW\_POSTLEITZAHLEN\_ORTE
- TO\_TUG\_NEW\_EXTERNE\_ORG\_TYPEN
- TO\_TUG\_NEW\_EXTERNE\_ORG
- D\_WERKE
  - TO\_CO\_ABS\_WERKE
  - TO\_CO\_ABS\_TITEL\_KURZFASSUNGEN
  - TO\_FODOK\_FD\_SPRACHEN
  - TO\_PV\_ABS\_TYPEN
- D\_FORSCHUNGSAKTIVITAET
  - TO\_FODOK\_FD\_FORSCHUNGSAKTIV
  - TO\_FODOK\_FORSCHUNGS\_TYPEN
  - TO\_FODOK\_FD\_SPRACHEN
  - TO\_FODOK\_FD\_TITEL\_BESCHREI
  - TO\_FODOK\_FD\_PERSONEN\_FA
  - TO\_FODOK\_FD\_PERSONEN
- TO\_FODOK\_FORSCHUNGSEINR\_FA
  - TO\_FODOK\_ARBGEBIETE
- D\_GELDGEBER
  - TO\_FODOK\_FD\_GELDGEBER
  - TO\_FODOK\_FD\_GG\_FOERDERTYP
  - TUG\_NEW\_EXT\_ORG\_FOERDERTYP
- D\_KOOP\_PARTNER
  - TO\_FODOK\_FD\_KOOPPARTNER
- D\_SACHGEBIETE
  - FODOK\_FD\_SACHGEBIETE
- D\_PARTNER
  - TO\_FODOK\_FD\_PARTNER

Wenn ein Dimensionsmapping aus mehr als einer Stage-Tabelle besteht, ist es oft notwendig innerhalb des OWB Joins durchzuführen. Hierbei kann man mittels einer eingebauten Funktionalität das Join-Kriterium manuell festlegen. In weiterer Folge muss man, wenn mehrere Datensätze verwendet werden, oft Union Operationen ausführen, um die Daten aus zwei oder mehreren verschiedenen Tabellen zusammenzuführen. In Abbildung 4.8 kann man die beispielhafte Einsetzung eines Joiner-Tools erkennen, um 2 Ursprungstabellen logisch miteinander zu verknüpfen.

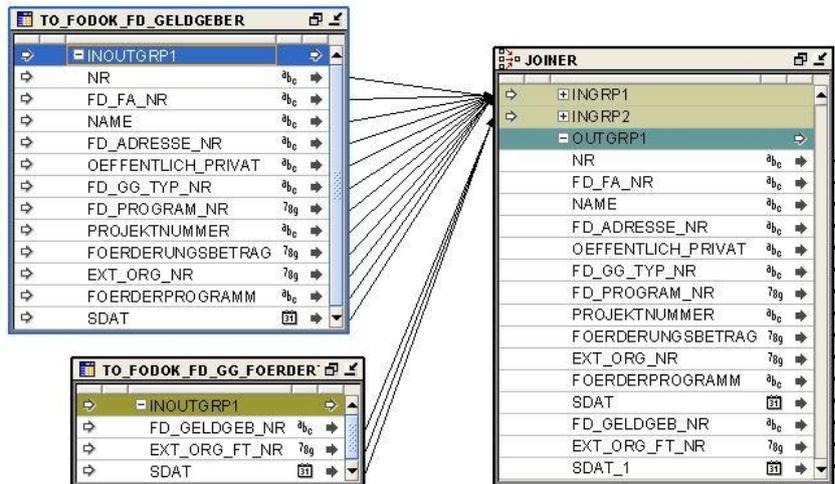


Abbildung 4.8: Join Operation

### 4.2.2 Besonderheit: Faktentabellen

Eine Besonderheit bei den Mappings stellen die Faktentabellen für das hier implementierte Star Schema dar. Es gilt zu unterscheiden ob es sich um ein Dimensions- oder Faktentabellenmappig handelt. Bei den Tabellen F\_Abschlussarbeiten und F\_Forschungsaktivitaeten handelt es sich um die zentralen Stellen des Star-Schemas. Diese Faktentabellen bestehen aus einem Primary Key und einem Unique Key, der sich wiederum aus Foreign Keys der verwendeten Dimensionstabellen zusammensetzt. Im Mapping dieser Tabellen werden die Foreign Keys via Key Lookup der Tabellen der Dimensionen herbeigerufen und ermöglichen somit ein Mapping für die Faktentabelle. In Abbildung 4.9 wird Aufbau der Faktentabelle der Abschlussarbeiten dargestellt.

| Table Details: TRG_DWH_CORE.F_ABSCHLUSSARBEITEN |         |             |         |
|---|---------|-------------|---------|
| Name  | Columns | Constraints | Indexes |
| PK_ABSCHLUSSARB...                              |         | Primary Key |         |
| UK_ABSCHLUSSARB...                              |         | Unique Key  |         |
| FK_DAUTOREN                                     |         | Foreign Key |         |
| FK_DBETREUER                                    |         | Foreign Key |         |
| FK_DEXTERNE_ORG...                              |         | Foreign Key |         |
| FK_ORGANISATION                                 |         | Foreign Key |         |
| FK_WERKE  |         | Foreign Key |         |

| Name               | Type        | Local Columns  | Reference           |
|--------------------|-------------|--|---------------------|
| PK_ABSCHLUSSARB... | Primary Key | FABSCHLUSSARBEITE...   |                     |
| UK_ABSCHLUSSARB... | Unique Key  | DWERKE_ID<br>DBETREUER_ID<br>DAUTOREN_ID<br>DORGANISATION_ID<br>DEXTERNE_ORGANISA... |                     |
| FK_DAUTOREN        | Foreign Key | DAUTOREN_ID  | TRG_DWH_CORE.D_A... |
| FK_DBETREUER       | Foreign Key | DBETREUER_ID   | TRG_DWH_CORE.D_B... |
| FK_DEXTERNE_ORG... | Foreign Key | DEXTERNE_ORGANISA...   | TRG_DWH_CORE.D_E... |
| FK_ORGANISATION    | Foreign Key | DORGANISATION_ID   | TRG_DWH_CORE.D_O... |
| FK_WERKE           | Foreign Key | DWERKE_ID  | TRG_DWH_CORE.D_W... |

Abbildung 4.9: Aufbau der Tabelle F\_Abschlussarbeiten

### 4.2.3 Aufbereitung der Daten für die Mart Ebene

Nach erfolgreicher Durchführung der Dimensionsmappings ist der nächste und letzte Schritt die Verknüpfung der Dimensionen zu einem OLAP Würfel. Mittels eines von OWB zur Verfügung gestellten Assistenten kann ein Cube erstellt werden. Dieser beinhaltet die Primary Keys der einzelnen Dimensionen, um so dimensionsübergreifende Abfragen möglich zu machen. In weiterer Folge kann man im Cube so genannte Measures oder Kennzahlen definieren. Diese werden dimensionsübergreifend online berechnet. In der hier implementierten multidimensionalen Lösung wurden keine Kennzahlen definiert. Die Datenspeicherung innerhalb des Würfels wird mit einem ROLAP Ansatz realisiert und die Speicherung des Würfels ist in der Mart Ebene des DWH, wo die einzelnen Dimensionsinformationen zu den Data Marts zusammen getragen werden.

Zunächst müssen die Daten aus den Dimensionstabellen via Mapping auf die konkreten Dimensionen übertragen werden. Bei der Implementierung des DWH der TU Graz werden für jede Dimension mindestens zwei Ebene definiert:

- ALLE
- EBENE1

Bei „ALLE“ handelt es sich um eine übergeordnete Ebene, in der nur die wichtigsten Informationen der gesamten Dimension vorkommen. In der „EBENE1“ werden danach die Detailinformationen aggregiert. Per LOAD Kommando wird bei der Ausführung des Mappings der Datenbestand geladen.

Um schlussendlich die Befüllung der Daten in den Hypercube zu realisieren, wird von der in der Core Ebene liegenden Faktentabelle ein Mapping direkt auf den Cube durchgeführt. Somit werden im Cube alle möglichen Verknüpfungen zu den konkreten Dimensionsdetails gespeichert und während der Analyse mit Hilfe einer Lookup Funktion die Daten aufgerufen.

Nach Ausführung dieser elementaren Schritte müssen wie in Kapitel 4.1.2 die Rechte für das Repository des OBIEE gesetzt werden. Der Import der Metadaten läuft analog zu Kapitel 4.1.2 ab. Der Unterschied ist nun, dass man sich nicht mehr um die Strukturierung des physischen Layers kümmern muss. Die Verknüpfungen sind bereits während der Arbeit mit dem OWB erfolgt. Es müssen noch ein Businessmodell für das jeweilige Themengebiet erstellt werden und abschließend steht im Präsentationsmodell, das für die Analysen zugängliche Modell zur Verfügung. In Abbildung 4.10 kann man mit einem Blick auf das Business Modell der Abschlussarbeiten erkennen, dass man durch die Modellierung mit dem OWB ein Star Schema erstellt hat.

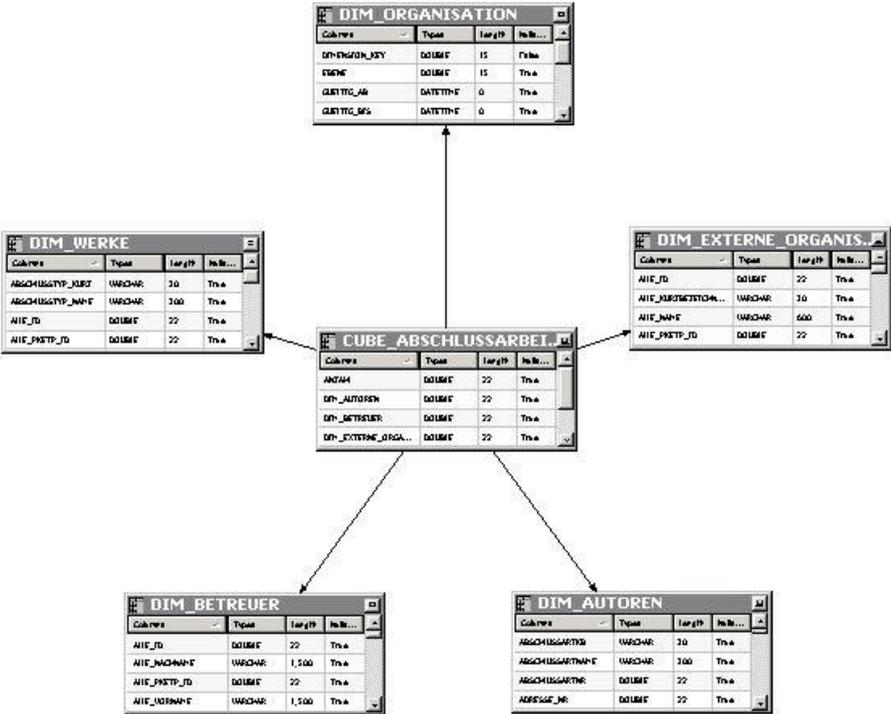
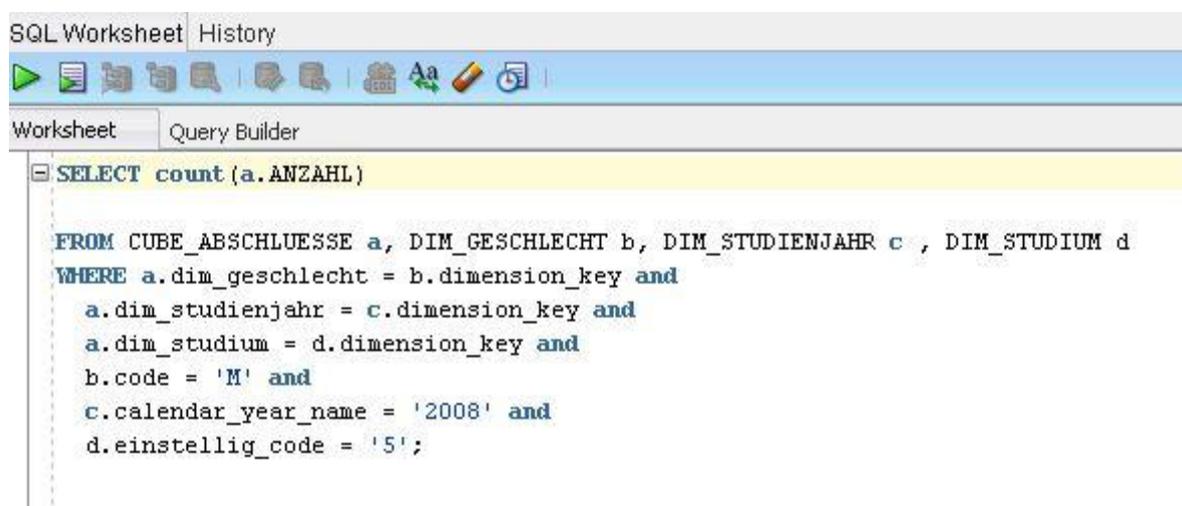


Abbildung 4.10: Business Modell - Star Schema

## 4.3 Evaluierung

In Bezug auf die Evaluierung der beiden oben angeführten Lösungsansätze, muss man zunächst für die Richtigkeit und Vollständigkeit der Implementierungen sorgen.

Hierzu wurden die in Kapitel 3.7 beschriebenen Abfragen in den Lösungen nachgebildet und die Daten verglichen. Hierbei konnten bei der Abfrage bezüglich des Technologietransfers 47 Einträge durch beide Lösungsansätze erzielt werden. Auch die Anzahl der Resultate bei den Abschlussarbeiten deckt sich mit der Lösung des Discoverer. Auch bei SQL Abfragen direkt in den Datenbanken stellen sich deckende Ergebnisse dar. In Abbildung 4.11 ist eine beispielhafte SQL Abfrage zur Qualitätskontrolle dargestellt.

The image shows a screenshot of a software interface for an SQL worksheet. At the top, there is a title bar with 'SQL Worksheet' and 'History'. Below it is a toolbar with various icons for execution, saving, and editing. The main area is divided into 'Worksheet' and 'Query Builder' tabs. The 'Worksheet' tab is active, displaying a SQL query in a text editor. The query is as follows:

```
SELECT count(a.ANZAHL)
FROM CUBE_ABSCHLUESSE a, DIM_GESCHLECHT b, DIM_STUDIENJAHR c , DIM_STUDIUM d
WHERE a.dim_geschlecht = b.dimension_key and
      a.dim_studienjahr = c.dimension_key and
      a.dim_studium = d.dimension_key and
      b.code = 'M' and
      c.calendar_year_name = '2008' and
      d.einstellig_code = '5';
```

Abbildung 4.11: SQL Abfrage zur Qualitätsüberprüfung

Wenn man nun die Gegenüberstellung des relationalen und des multidimensionalen Ansatzes in Betracht zieht, muss man, um eine Empfehlung daraus zu ziehen einige Faktoren beachten.

### 4.3.1 User Interface

Obwohl bei beiden Lösungsansätzen dasselbe Reporting Werkzeug zum Einsatz kommt, gibt es hinsichtlich der Verwendung für die Endanwender einige Differenzen. Da beim relationalen Ansatz die Rohdaten im Prinzip nicht grundlegend aufbereitet werden, wird in der Weblogic kein DWH Gefühl erzeugt. Den Anwendern des F&T-Hauses kommt dies allerdings zu Gute, da die Struktur der Daten denen ähnelt, die bereits im Discoverer vorhanden sind. In Abbildung 4.12 wird das User Interface der relationalen Lösung dargestellt.



Abbildung 4.12: User Interface OBIEE relational

Diese Baumstruktur und Darstellung der einzelnen Bereiche des Themengebietes sind für den Anwender leicht verständlich, da dieser bereits mit dem Discoverer gearbeitet hat. In der multidimensionalen Lösung muss man zunächst wissen in welcher Dimension die eigentliche Information verpackt ist. Dies stellt sich etwas unübersichtlich dar, wie in Abbildung 4.13 zu sehen ist.



Abbildung 4.13: User Interface OBIEE multidimensional

Somit ist für den Anwender aus Betrachtung des User Interface die relationale Lösung sinnvoller, da diese weniger Einarbeitungszeit in Anspruch nimmt.

### **4.3.2 Verständlichkeit und Lernfaktor**

Hinsichtlich der Verständlichkeit der Lösungen kann ebenfalls das relationale Modell punkten. Diese Implementierung ist analog zum Oracle Discoverer aufgebaut und somit benötigen die Mitarbeiter des F&T-Hauses wenig bis keine Einarbeitungszeit. Hinsichtlich der multidimensionalen Lösung gibt es einige Sonderfeatures, die aber von der Fachabteilung nicht verwendet oder gebraucht werden.

Die Einarbeitungszeit in die DWH Lösung beträgt länger, da erst die grundlegenden Kenntnisse über den Aufbau und Ablauf eines DWH benötigt werden, um in weiterer Folge effizient Analysen erstellen zu können.

### **4.3.3 Performance**

Für die Evaluierung der Performance wurden wiederum die beiden Abfragen aus Kapitel 3.7 herangezogen. Während in der relationalen Lösung die Abfragen deutlich schneller als im Discoverer durchgeführt werden konnten, kann die Berechnung im DWH Ansatz einiges an Zeit dauern.

Dies liegt daran, dass sich aus den Strukturen der Daten kein richtiges Star Schema bilden lässt. Die Dimensionen sind zwar durch ein zentrales Element miteinander verbunden, jedoch ähnelt dieses Gebilde einem umgekehrten Star Schema. Das bedeutet, dass die eigentlichen Daten aus der Faktentabelle gelesen werden und nicht aus den Dimensionen.

Dadurch „explodiert“ die zentrale Tabelle förmlich, da es bei der Konstruktion durch Join Elemente zu kartesischen Produkten kommt. Damit muss auch in der Datenbank einiges mehr an Rechenzeit aufgewendet werden um ein Ergebnis zu erzielen.

Die Berechnungen erstrecken sich oft über mehrere Minuten und somit wäre dieser Punkt bereits ein Ausschlusskriterium für die multidimensionale Lösung.

### **4.3.4 Zeitfaktor**

Für die Implementierung der beiden Lösungen ist natürlich der Zeitfaktor nicht zu verachten. Während die relationale Lösung eine schnelle Möglichkeit bietet um die Daten zu strukturieren und für Abfragen verfügbar zu machen, muss in der DWH Lösung ein kompletter Prozess durch das gesamte Data Warehouse vollzogen werden.

Somit sind die Zeitkosten bei der Implementierung einer DWH Lösung natürlich um einiges höher als die der rein relationalen Implementierung. Dazu kommen noch die Vorarbeiten im Sinne des Planens der Dimensionen und Cubes, was auch einen erheblichen Arbeitsaufwand darstellt.

#### **4.3.5 Funktionalität**

In Bezug auf die durch die Lösungen gebotenen Funktionalitäten kann die multidimensionale Lösung beeindrucken. Während in der relationalen Lösung nur dieselben Funktionalitäten wie beim Discoverer zur Verfügung stehen, ist es beim DWH Ansatz beispielsweise möglich Kennzahlen einzuführen, um oft verwendete Berechnungen online durchzuführen.

In dieser Lösung wurden keine Kennzahlen implementiert, da das F&T-Haus diese nicht benötigt, aber für andere Anwendungsgebiete wäre diese Funktion durchaus interessant. Auch durch die Eigenschaft des OLAP Würfels entstehen viele Möglichkeiten der Datenverarbeitung. Man kann beispielsweise durch den Einsatz von anderer BI Software die Daten für den Bereich Data Mining aufbereiten oder andere Vorteile daraus ziehen, was in der relationalen Lösung keinesfalls möglich ist.

#### **4.3.6 Zukunftsorientiertheit**

Während die relationale Lösung nur einen schnellen Lösungsweg bietet, ist die multidimensionale Lösung sehr zukunftsorientiert gedacht. DWH ist generell ein Begriff der in Zukunft immer mehr zu hören sein wird und viele Unternehmen setzen bereits darauf.

Nicht nur die Historisierung der Daten, sondern auch die erweiterte Funktionalität durch den OLAP Ansatz garantieren eine Weiterentwicklung in der Zukunft. Es ist auch ein Ziel der TU Graz, die Daten der Institute in einem zentralen DWH zusammenzutragen. Dies würde viele Wege der Informationsbeschaffung ersparen. Im Vergleich dazu ist der relationale Ansatz nur eine rein kundenoptimierte Lösung.

#### **4.3.7 Wartbarkeit**

Um auch in weiterer Folge Daten in die beiden implementierten Ansätze zu integrieren, muss eine leichte Wartbarkeit realisierbar sein. Im Betracht auf den relationalen Ansatz kann man die Daten sehr schnell neu strukturieren und somit neue Metadaten in das System integrieren.

Beim DWH Ansatz ist eine Erweiterbarkeit noch besser realisierbar. In der Theorie eines DWH wird bereits ein großes Hauptaugenmerk auf die Wartbarkeit gelegt. Die Daten können, da diese meistens bereits im DWH befindlich sind schnell und einfach in die

Dimensionen integriert werden. Zwar ist eine Implementierung dieses Ansatzes mit mehr Zeitkosten verbunden, aber durch die steigende Funktionalität wäre dies gerechtfertigt.

#### **4.3.8 Implementierung**

Bei der Implementierung der Ansätze erkennt man sofort einen großen Unterschied. Während man bei der Implementierung des relationalen Ansatzes ein sehr breit gefächertes Wissen über das zu schaffende Quell System benötigt, kann man mit dem DWH Ansatz ohne Wissen über die Strukturen ein äquivalentes System nachbilden.

Das Wissen über das Ausgangssystem bekommt man aber vermehrt durch Analyse der Daten. Hierbei kann man aber schon beim Design potentielle Fehler erkennen und somit für weitere Schritte ausschließen. Bei der DWH Alternative werden fehlerhafte Daten erst nach der Erstellung sichtbar und man muss um auf ein richtiges Ergebnis zu kommen den kompletten DWH Ansatz von vorne implementieren.

Darum ist es besser sich mit dem Quellsystem zu beschäftigen, was dem relationalen Ansatz in diesem Punkt zu Gute kommt.

#### **4.3.9 Zusammenfassung**

Abschließend kann man sagen, dass der relationale Ansatz im Gegenzug für zu einem DWH System für das F&T-Haus besser geeignet ist. Auch wenn man in Zukunft eventuell eine multidimensionale Lösung anstreben will, muss zunächst eine Lösungsvariante für das Performance Problem gefunden werden. Wenn man diesen Faktor beheben kann, ist es möglich noch effizientere Berichte für die Fachabteilungen zu erstellen und man würde auf ein zukunftsorientiertes System setzen.

## 5 Empfehlung – Conclusio

In diesem Kapitel werden nun die Erkenntnisse, die aus der Ausgangssituation und der Evaluierung gezogen wurden, zusammengefasst und strukturiert ausgewertet. Anhand der Auswertungen soll es möglich sein eine konkrete Empfehlung für das F&T-Haus zu tätigen. Damit wird erreicht, dass die Fachabteilung mit den bestmöglichen Mitteln ihr Berichtswesen fortsetzen kann und mit der ausgewählten Lösung ein sinnvoller Ansatz für zukünftige Planungen zur Verfügung gestellt wird.

In weiterer Folge wird auch auf zukunftsorientierte Themen im Bereich der Wissenschaft eingegangen. Auch die Erkenntnisse, die während der Erstellung der Lösungsansätze aufgekommen sind werden in diesem Kapitel detailliert zusammengefasst.

### 5.1.1 Rückblick der Aufgabenstellung

In Betracht auf die vom F&T-Haus geforderten Details kann man sagen, dass die Umsetzung ein deutlicher Erfolg war. Alle geforderten Aufgabenstellungen konnten mithilfe des relationalen Ansatzes abgedeckt werden.

Da das Zielsystem dem Quellsystem stark ähnelt, ist für einen problemlosen Umstieg der Anwender gesorgt. Die Mitarbeiter können dadurch ohne Zeitverlust die Berichterstellung fortsetzen und es kommt beim Betrieb zu keinerlei Stehzeiten.

Seitens des Zentralen Informatikdienstes (ZID) gab es für die Stakeholder des Systems bereits zwei umfassende Einschulungen. Hierbei wurden grundlegende Basis-Skills für die Verwendung des relationalen Ansatzes geschaffen und auf die Fragen der Mitarbeiter eingegangen. Das System, das seit Mai 2012 in Produktionsbetrieb ist, weist keine Schwächen gegenüber dem Discoverer auf und erfreut sich einer großen Partizipation.

Durch die Evaluierung wurde eine qualitätssichernde Maßnahme hinsichtlich der Datengleichheit durchgeführt. Dadurch kann man feststellen, dass es keinen Datenverlust durch die Transferierung der Daten in das neue System gibt.

Die Namenskonvention wurde zugunsten des F&T-Hauses auf der Präsentationsschicht angepasst, um für die Mitarbeiter noch verständlichere Navigation durch das System zu bieten. Die genauen Anforderungen an die Umbenennung der geforderten Felder wurden durch die Fachabteilung festgelegt und sind im Anhang dieser Arbeit zu finden.

Dadurch, dass die eigentliche Business Komponente und sämtliche Technik für die Datenaufbereitung im OBIEE Repository realisiert wird, wird eine Transparenz für das F&T-Haus geschaffen. Die Mitarbeiter sind nur in der Lage mit den Daten zu arbeiten, jedoch nicht diese zu ändern. Für gewünschte Änderungen sind die Fachabteilungen des ZID verantwortlich.

In Bezug auf die Leistung des Systems wurde eine deutliche Verbesserung im Vergleich zum Quellsystem Discoverer erreicht. Die Abfragegeschwindigkeit ist deutlich höher und die Anzahl der zu verarbeitenden Daten wurde durch die Aufstockung des Repositories seitens Oracle merkbar erhöht.

Auch die Wartbarkeitsanforderungen an den relationalen Ansatz wurden zur Zufriedenstellung des F&T-Hauses durchgeführt. Es ist durch den ZID innerhalb kurzer Zeit möglich neue Daten in das System zu integrieren und einen Zusammenschluss der Quelldateien zu erhalten.

Somit wurden alle Anforderungen an das von der Fachabteilung gewünschte System gedeckt, sodass auch in Zukunft ein reibungsloser Betrieb seitens des F&T-Hauses möglich ist.

### **5.1.2 Lessons Learned**

Im Laufe der Implementierungsarbeit sind einige Erkenntnisse in Bezug auf die Modellierung von Datenbankstrukturen in relationalen und multidimensionalen Ansätzen aufgekommen. Bei einer Konzeption eines Systems in einem relationalen Umfeld muss man auf jeden Fall eine gewisse Vorkenntnis im Bereich des Quellsystems besitzen.

Wenn keine Vorbildung über das Ursprungssystem vorhanden ist, muss man durch einschlägige Recherche selbst die Informationen zusammentragen, die für die erfolgreiche Implementierung des Ansatzes benötigt werden. Dies hat allerdings den Vorteil, dass man nach Verständnis des Systems eine schnelle Implementierung durchführen kann. Auch Fehler, die sich im Quellsystem befinden können, werden in der Analysephase aufgedeckt. Somit ist es dem Modellierer möglich diese zu beheben und einen weiteren Leistungsvorteil in Hinsicht auf das ursprüngliche System zu bieten.

Andererseits ermöglicht der relationale Ansatz eine schnelle Realisierung der Lösung, falls wenig Zeit für die Implementierung vorhanden ist. Auch das Nachtragen von Daten in das Zielsystem ist ohne weitere Probleme möglich und die Produktionskosten und der Zeitfaktor sind hierbei sehr gering, was wiederum einer schnellen Lösung zu Gute kommt.

Hinsichtlich des multidimensionalen Ansatzes kommt es bei der Implementierung zu einigen Problemen. In der Theorie sollte es ohne Probleme möglich sein, ein effizientes zentrales System mit viel Funktionalität zu konstruieren. Allerdings kommt es durch die Tatsache, dass die Ursprungsdaten sehr verschachtelt sind, zu Einschränkungen. Dies beginnt bereits bei der Definition der Dimensionen.

Da für das F&T-Haus eine einfache Handhabung oberste Priorität hat, muss man sich sorgfältig logische Dimensionen überlegen, die für die Fachabteilung zu verwenden sind. In

der Praxis der Implementierung führt dies jedoch zu sehr großen Dimensionen mit sehr viel Detailinformationen.

Die Abfragegeschwindigkeit wird auch durch diesen Faktor verlangsamt. Wenn man verschiedenste Tabellen für die Zusammenfassung zu einer Dimension verwendet, kommt es aufgrund der Join Operationen zu sehr vielen kartesischen Produkten. Dadurch wird der Datenbestand der Faktentabelle von einem Bruchteil an Informationen auf ein Zehntausendfaches erhöht. Dies resultiert in einer geringen Performance der multidimensionalen Lösung und daher ist für den konkreten Datenbestand von dieser Lösung abzuraten.

Hinsichtlich der Wartbarkeit, ist der multidimensionale Ansatz sicherlich zukunftsorientiert. Durch den Umstand, dass die Implementierung weiterer Daten aber einige Zeit in Anspruch nimmt, sind hierbei keine schnellen Lösungswege realisierbar.

Ein weiterer Punkt ist, dass mit noch mehr Datenaufkommen möglicherweise noch mehr kartesische Produkte entstehen und somit der Ansatz noch ineffizienter gegenüber der relationalen Lösung wird.

Die Tatsache, dass bei den vorhandenen Metadaten keine Star-Schema gemäße Modellierung möglich ist, rundet die negative Erkenntnis des multidimensionalen Ansatzes ab. Dieser hier entstehende verkehrte Stern widerspricht der eigentlichen Modellierung eines DWH und somit kann keine performante Lösung erzielt werden.

Schlussendlich gilt zu betrachten, dass bei einer Implementierung immer als erstes eine Analyse durchgeführt werden muss. Aufgrund dieser lassen sich dann die weiteren Teilschritte einer Implementierung planen und in Zuge dieser durchführen. Mit dieser Vorgehensweise kann man im Vorhinein Fehler ausschließen und andere Probleme vermeiden. Eine beispielhafte Vorgehensweise ist in Abbildung 5.1 ersichtlich.

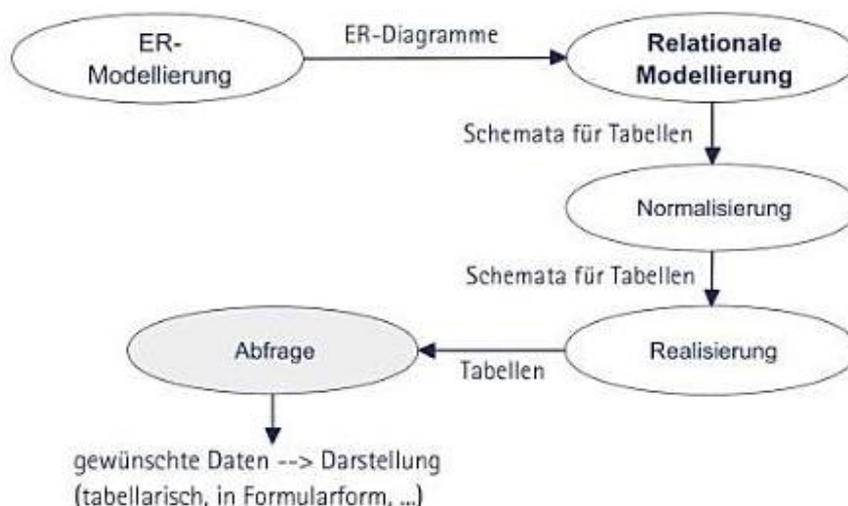


Abbildung 5.1: Vorgehensweise bei einer relationalen Implementierung [Sig12]

### 5.1.3 Future Work

In Betracht auf den relationalen Teil der Implementierung kann man für die Zukunft nur mäßige Entwicklungen erwarten. Der eigentliche Ansatz basiert auf seit Jahren in Verwendung befindlichen Datenbankwissen und SQL Abfragen. Man kann in näherer Zeit zwar mit der ein oder anderen Neuerung im Bereich relationaler Datenmodelle rechnen, jedoch wird dies bei weitem keine bahnbrechenden Erkenntnisse mit sich bringen, da man hier vermehrt auf bewährte Ansätze setzt.

Im Bereich der multidimensionalen DWH orientierten Lösung sieht die Zukunftsentwicklung deutlich fortschrittlicher aus. Da dieser Ansatz „jünger“ ist als ein rein relationaler, kommen hier laufend neue Erkenntnisse, die die Entwicklung multidimensionaler Lösungen fördern.

Ein wichtiger Aspekt zukunftsorientierter DWH System ist vor allem der Umgang mit noch größeren Datenmengen, als bisher möglich. Da die Größe an Daten geometrisch zunimmt, muss man Lösungen entwickeln um nicht nur mit einer großen Flut an Metadaten umzugehen, sondern auch um variable Lösungswege dafür zu schaffen. Vor allem große Unternehmen sollten auf diese Technologien setzen um in weiterer Folge Kosten und Zeit zu sparen.

Auch im Bereich Persistenz werden DWHs in Zukunft großen Einzug erhalten. Wenn man beispielsweise Amazons Kundenrating heranzieht, kann dieses mit Hilfe einer DWH Struktur noch genauere Vorhersagen tätigen und schneller arbeiten, da bereits durch verschiedenste Mobile Applikationen Daten der Nutzer erfasst werden.

Hinsichtlich der Entwicklung der sozialen Netzwerke birgt der multidimensionale Ansatz große Chancen, um mit den nahezu explodierenden Größen der Daten umzugehen. Auch der mobile Betrieb erfreut sich wachsender Kundschaft. Somit muss auch mit Daten die in Echtzeitanwendungen vorkommen umgegangen werden können. Auf diesen Gebieten gibt es bereits einige erste Forschungen.

Durch die wachsende Leistung der Hardware hinsichtlich Multi-Core oder der zunehmenden Größe des verfügbaren Speichers, kann man heutzutage mehr Daten verarbeiten als jemals zuvor. In Bezug auf diese Tatsachen werden die DWH Strukturen überarbeitet um noch effizienteres Arbeiten zu ermöglichen. Auch die Kosten für die Anschaffung und Implementierung solcher Systeme werden immer geringer, da durch die Entwicklung sehr kostengerecht gearbeitet werden kann.

Während man vor ein paar Jahren DWHs noch als Nischenprodukt ansah, konnte mit der Bewegung der Technologien Hadoop und NoSQL ein Einzug in den Alltagsmarkt vollbracht werden. Da immer mehr Anwender in Betrieben multidimensionale Strukturen für die Sicherung ihrer Daten nutzen, kommt auch beispielsweise das Controlling eines Unternehmens mit DWHs in Kontakt.

Unternehmen die in der Erstellung von DWH Systemen tätig sind, setzen vermehrt in ihrem Produktportfolio auf die Entwicklungsabteilungen für die Warehouse Lösungen. Somit wird der Trend in den folgenden Jahren vermehrt in die Entwicklung solcher Systeme gehen. In Abbildung 5.2 ist ein Technologiechart der Firma Ventana dargestellt.

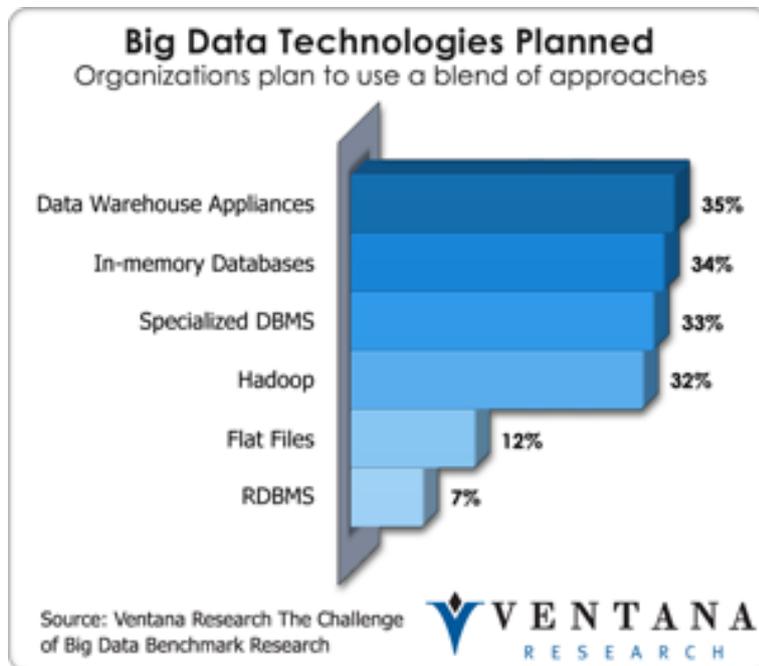


Abbildung 5.2: Technologiechart der Firma Ventana [Smi12]

Auch die Tatsache, dass die Rohdaten nicht in strukturierter Form vorliegen müssen, erleichtert den Einstieg auf ein DWH System und somit wird in der Zukunft vor allem für unverbundene Datenmenge ein Data Warehouse System die beste Lösung darstellen.

#### 5.1.4 Empfehlung

Anhand der Anforderungen des F&T-Hauses, die in Kapitel 3 dargestellt wurden, und des Ergebnisses der Evaluierung ist klar eine Empfehlung für den relationalen Ansatz auszusprechen.

Da dieser dem Quellsystem sehr stark ähnelt und man somit den Anwendern die Einarbeitungszeit so kurz als möglich macht ist diese Variante für die Fachabteilung sehr sinnvoll. Ein anderes Qualitätskriterium ist die mit dem System erzielte Leistung. Da beim multidimensionalen Ansatz durch die hohe Aggregation der Daten sehr lange Rechenzeiten aufkommen, ist somit klar eine Entscheidung gegen dieses Verfahren zu treffen.

Durch die Implementierung des relationalen Ansatzes ist es dem F&T-Haus auch nach einiger Zeit möglich, relevante Daten in das System zu migrieren und das ohne einen damit verbundenen Mehraufwand für die implementierenden Organisationen zu schaffen. In

weiterer Folge ist zu sagen, dass auch der Zeitfaktor der Entwicklung eine erhebliche Rolle spielt.

So kann man, wenn gefordert, ein neues System schnell und mühelos schaffen, wenn man Kenntnisse im Grundsystem hat. Auch die Verständlichkeit des relationalen Ansatzes garantiert für die Anwender des F&T-Hauses eine schnelle und korrekte Aufrechterhaltung des Berichtswesens.

Somit ist klar deutlich, dass mit Hilfe des relationalen Ansatzes alle Forderungen der Fachabteilung vollständig erfüllt sind und dieser im Vergleich mit der multidimensionalen Lösung besser abschneidet, vor allem aufgrund der Leistung des Systems.

Das F&T-Haus hat dadurch eine technologisch hochwertige Lösung als Ersatz für den in Verwendung befindlichen Oracle Discoverer gefunden. Da dies die letzte Instanz war, die Gebrauch von diesem Instrument macht, ist es nun möglich das Tool im Herbst 2012 vom Betrieb zu nehmen und nur mehr den relationalen Ansatz als Ausgangsbasis für die Erstellung der Berichte und Analysen heranzuziehen.

### 5.1.5 Conclusio

Zusammenfassend ist zu erwähnen, dass bei einer Implementierung im relationalen Stil das Grundsystem ohne einschlägige Analysen nicht nachgebildet werden kann. Aus dem Kontext der Future Work lässt sich ableiten, dass es in Zukunft möglicherweise Wege und Ideen geben wird um die in dieser Arbeit thematisierte Problematik zu realisieren.

Es können beispielsweise durch neue Entwicklungen der OWB Tools Möglichkeiten geschaffen werden, um gezielt mit Gebilden, für die kein gemeinsames Star Schema existiert, umgehen zu können.

Wenn man dieses Ziel erreichen würde, wäre die multidimensionale Lösung der relationalen auf jeden Fall vorzuziehen, da hier eine Vielzahl von weiterführenden Funktionen zur Verfügung steht. Außerdem würde man hier mit einer „State of the Art“-Lösung arbeiten, was einen weiteren Einsatz eines DWH rechtfertigen würde.

Zum Abschluss ist zu erwähnen, dass mit dieser Arbeit für das F&T-Haus ein neuartiges und vielversprechendes System erstellt wurde, welches auf jeden Fall mit dem Oracle Discoverer konkurrieren kann. Die Fähigkeiten der Implementierung übertreffen wie in den Vorbedingungen festgelegt die Funktionen der alten Software und somit ist der relationale Ansatz ein würdiger Nachfolger.

Anhand der Evaluierung der Ergebnisse wurde eine Entscheidung für das bessere System getroffen, welches mit Sicherheit für die nächsten Arbeitsschritte des F&T-Hauses Einsatz finden wird und auch benutzerfreundlich zu bedienen ist.

## Literaturverzeichnis

- [Ana97] Anahory, S., Murray, D. 1997, Data Warehouse / Planung, Implementierung und Administration, Bonn
- [Bae03] Baer, H., 2003, On-Time Data Warehousing with Oracle10g – Information at the Speed of your Business
- [Bau04] Bauer, A., Güunzel, H., 2004, Data Warehouse Systeme - Architektur, Entwicklung, Anwendung, 2. Auflage, Heidelberg
- [Cham98] Chamoni, P., Gluchowski, P., 1998, Analytische Informations-systeme, Berlin
- [Codd98] Codd, E.F., Codd, S.B., Salley, C.T., 1998, Providing OLAP (Online Analytical Processing) to User-analysts: An IT Mandate
- [Dij05] Dijcks, J.-P., 2005, OracleBI Warehouse Builder 10g Release 2 - An overview
- [Inm94] Inmon, W.H, Hackathorn, R. D., 1994, Using the Data Warehouse. John Wiley and Sons, Inc., New York
- [Inm99] Inmon, W.H., 1999, Data Mart Does Not Equal Data Warehouse, DM Direct
- [Inm02] Inmon, W. H., 2002, Building the Data Warehouse. John Wiley and Sons, Inc., New York
- [Jan96] Jahnke, B., Groffmann, H.-D., Kruppa , S., 1996, On-Line Analytical Processing, in Wirtschaftsinformatik 38, S. 312-324
- [Jia2000] Jiawei, H., Kamber, M., 2000, Data Mining – Concepts and Techniques, Morgan Kaufmann Publishers
- [Kal99] Kalakota, R., Whinston, A.B., 1996, Frontiers of Electronic Commerce, Verlag Addison Wesley
- [Kim02] Kimball, R., Ross, M., 2002, The Data Warehouse Toolkit, 2nd ed., USA, John Wiley & Sons Inc. 436 p. ISBN 0-471-20024-7
- [Kim98] Kimball, R., Reeves, L., Ross, M., Thornwaite, W., 1998, The Data Warehouse Lifecycle Toolkit, New York

- [Kot98] Kotidis, Y., Roussopoulos, N., 1998, An Alternative Storage Organization for ROLAP Aggregate Views Based on Cubetrees
- [Kur99] Kurz, A., 1999, Data Warehousing Enabling Technologies
- [Mac04] Machin, I., 2004, Optimizing Oracle Warehouse Builder Performance
- [Pen95] Pendse, N., Creeth, R., 1995, The OLAP-Report: Succeeding with On-Line Analytical Processing, Vol.1, Business Intelligence
- [Pet94] Peterson, S., 1994, Stars: A Pattern Language for Query Optimized Schema". Sequent Computer Systems WhitePaper
- [Pon01] Ponniah, P., 2001, Data Warehousing Fundamentals, USA. John Wiley & Sons Inc. 516 p. ISBN 0-471-41254-6
- [Rit04] Rittman, M., 2004, Streamlining Oracle 9i ETL with Pipelined Table Functions
- [Rit06a] Rittman, M., 2006, Implementing Real-Time Data Warehousing Using Oracle 10g
- [Rit06b] Rittman, M., 2006, An Introduction to Oracle Warehouse Builder 10g
- [Rud98] Rudin, K., 1998, Transaction Processing Today,  
<http://www.dbmsmag.com/9801d13.html>, Abruf: August 19.07.2012
- [Sig12] Sigmund, M., 2012, Erstellung eines eBooks zum Thema „Normalisierung“, [http://winfwiki.wifom.de/index.php/Erstellung\\_eines\\_eBooks\\_zum\\_Thema\\_%E2%80%9ENormalisierung%E2%80%9C](http://winfwiki.wifom.de/index.php/Erstellung_eines_eBooks_zum_Thema_%E2%80%9ENormalisierung%E2%80%9C) Abruf: 19.07.2012
- [Smi12] Smith, M., 2012, Hortonworks Leads a Fast and Growing Herd of Hadoop, <http://marksmith.ventanaresearch.com/2012/06/22/hortonworks-leads-a-fast-and-growing-herd-of-hadoop/> Abruf: 18.07.12
- [Swi2000] Swift, R.S., 2000, Accelerating Customer Relationships, Verlag Prentice Hall
- [Tan97] Tanler, R., 1997, The Intranet Data Warehouse, Tools and Techniques for Building an Intranet-Enabled Data Warehouse

- [Wie99 ]        Wieken, J.-H., 1999, Der Weg zum Data Warehouse / Wettbewerbsvorteile durch strukturierte Unternehmensinformationen. Addison-Wesley-Longman, München

## Abbildungsverzeichnis

|  |    |
|--|----|
| Abbildung 2.1: Informationssystem(analytisch) nach Chamoni und Gluchowski [Cham98]....           | 8  |
| Abbildung 2.2: Architektur von abhängigen Data Marts [Bau04, S. 61].....                         | 10 |
| Abbildung 2.3: Architektur von abhängigen Data Marts [Bau04, S. 63].....                         | 11 |
| Abbildung 2.4: Definitionsphase im ETL Prozess [Kur99].....                                      | 13 |
| Abbildung 2.5: Ausführungsphase im ETL Prozess [Kur99].....                                      | 13 |
| Abbildung 2.6: Typisches Star Schema [Pet94].....  | 18 |
| Abbildung 2.7: Typisches Snow Flake Schema [Pet97].....  | 19 |
| Abbildung 2.8: Beispielhafter Aufbau eines Hypercubes nach Inmon [Inm02] .....                   | 22 |
| Abbildung 2.9: Architektur eines Data Warehouse nach relationalem Ansatz [Inm02] .....           | 26 |
| Abbildung 2.10: Architektur eines Data Warehouse nach dimensionalem Ansatz [Kim02]....           | 28 |
| Abbildung 3.1: Discoverer User Interface .....   | 36 |
| Abbildung 3.2: Formel Editor des Discoverer.....   | 37 |
| Abbildung 3.3: Architektur des Oracle Warehouse Builders [Rit06b].....                           | 38 |
| Abbildung 3.4: Vergleich eines Systems mit Tabellenfunktion und eines Systems ohne [Rit06a]..... | 39 |
| Abbildung 3.5: Anfragegesteuerte Transformation [Dij05].....                                     | 40 |
| Abbildung 3.6: OBIEE Repository.....   | 41 |
| Abbildung 3.7: BI Answers mit Beispielabfrage.....   | 42 |
| Abbildung 3.8: Filter für die Analyse der Abschlussarbeiten .....                                | 48 |
| Abbildung 3.9: Filter für Technologietransfer .....  | 50 |
| Abbildung 4.1: Import-Tool des OWB .....   | 53 |
| Abbildung 4.2: Mapping für den View "Fodok_Fd_Personen".....                                     | 55 |
| Abbildung 4.3: Beispielhafter Prozessfluss.....  | 55 |
| Abbildung 4.4: Setzen der SELECT Rechte für das OBIEE Repository .....                           | 56 |
| Abbildung 4.5: Metadatenimport mittels OBIEE Import-Tool.....                                    | 57 |
| Abbildung 4.6: Modellierung eines Joins im OBIEE .....   | 58 |
| Abbildung 4.7: Mapping für die Dimension Organisation .....                                      | 62 |
| Abbildung 4.8: Join Operation .....  | 64 |
| Abbildung 4.9: Aufbau der Tabelle F_Abschlussarbeiten.....                                       | 64 |
| Abbildung 4.10: Business Modell - Star Schema .....  | 66 |

---

|  |    |
|--|----|
| Abbildung 4.11: SQL Abfrage zur Qualitätsüberprüfung .....                         | 67 |
| Abbildung 4.12: User Interface OBIEE relational .....                              | 68 |
| Abbildung 4.13: User Interface OBIEE multidimensional .....                        | 68 |
| Abbildung 5.1: Vorgehensweise bei einer relationalen Implementierung [Sig12] ..... | 74 |
| Abbildung 5.2: Technologiechart der Firma Ventana [Smi12] .....                    | 76 |

## Tabellenverzeichnis

|   |    |
|---|----|
| Tabelle 2.1: Auszug der Unterschiede zwischen OLAP – OLTP [Kur99] .....       | 7  |
| Tabelle 3.1: IST-Stand der Tabellen für den Bereich Abschlussarbeiten .....   | 44 |
| Tabelle 3.2: IST-Stand der Tabellen für den Bereich Technologietransfer ..... | 46 |
| Tabelle 4.1: Verwendete Views für den Bereich Abschlussarbeiten.....          | 51 |
| Tabelle 4.2: Verwendete Views für den Bereich Technologietransfer .....       | 52 |
| Tabelle 4.3: Mappings für den Stage Bereich für Abschlussarbeiten .....       | 54 |
| Tabelle 4.4: Mappings für den Stage Bereich für Technologietransfer .....     | 54 |
| Tabelle 4.5:Join Modellierungen der Faktentabelle.....                        | 58 |
| Tabelle 4.6: Weitere Joins für Abschlussarbeiten .....                        | 59 |
| Tabelle 4.7: Joins für den Bereich Technologietransfer .....                  | 60 |
| Tabelle 4.8: Dimensionen für den Bereich Abschlussarbeiten .....              | 61 |
| Tabelle 4.9: Dimensionen für den Bereich Technologietransfer .....            | 61 |

## Abkürzungsverzeichnis

|       |  |
|-------|--|
| BI    | Business Intelligence                                      |
| Bzw.  | beziehungsweise  |
| CDC   | Change Data Capture  |
| CRM   | Customer Relationship Management                           |
| DBMS  | Datenbankmanagementsystem                                  |
| DWH   | Data Warehouse   |
| ETL   | Extract Transform Load                                     |
| OBIEE | Oracle Business Intelligence Enterprise Edition            |
| OLAP  | Online Analytical Processing                               |
| OLTP  | Online Transaction Processing                              |
| OWB   | Oracle Warehouse Builder                                   |
| PEL   | Partition Exchange Loading                                 |
| SAP   | Systeme, Anwendungen und Produkte in der Datenverarbeitung |
| SQL   | Structured Query Language                                  |
| TU    | Technische Universität                                     |
| z.B.  | zum Beispiel   |
| ZID   | Zentraler Informatikdienst                                 |

## **Anhang**

|   |           |
|---|-----------|
| <b>Anhang 1: Namenskonventionen für OBIEE .....</b> | <b>87</b> |
|---|-----------|

**Anhang 1: Namenskonventionen für OBIEE**

| ALT  | NEU                             |
|--|---------------------------------|
|  TO_CO_ABS_BETREUER           | <b>Betreuer</b>                 |
| BETREUER_TYP_NAME  | <b>Betreuer_Typ_Bezeichnung</b> |
| NACHNAME   | <b>Betreuer_Nachname</b>        |
| TITEL  | <b>Betreuer_Titel</b>           |
| VORNAME  | <b>Betreuer_Vorname</b>         |
|  |                                 |
|  TO_CO_ABS_WERKE              | <b>Werke</b>                    |
| ERSCHEINUNGSJAHR   | <b>Werke_Erscheinungsjahr</b>   |
| NR   | <b>Werke_Nr</b>                 |
|  |                                 |
|  TO_TUG_NEW_EXTERNE_ORG      | <b>Ext_Org</b>                  |
| NAME   | <b>Ext_Org_Name</b>             |
| NAME_MASTERORG_L1  | <b>Ext_Org_Abt 1</b>            |
| NAME_MASTERORG_L2  | <b>Ext_Org_Abt 2</b>            |
|  |                                 |
|  TO_TUG_NEW_EXTERNE_ORG_TYP | <b>Ext_Org_Typ</b>              |
| NAME   | <b>Ext_Org_Typ_Bezeichnung</b>  |
|  |                                 |
|  TO_TUG_NEW_LAENDER         | <b>Laender</b>                  |
| NAME   | <b>Laender_Name</b>             |
|  |                                 |
|  TO_TUG_NEW_ORGANISATIONEN  | <b>Institute</b>                |
| KENNUNG  | <b>Institute_Kennung</b>        |
| NAME   | <b>Institute_Name</b>           |
|  |                                 |
|  TO_TUG_NEW_PLZ_ORTE        | <b>Ext_Org_PLZ_Orte</b>         |
| ORT  | <b>Ext_Org_Orte_AT_DE</b>       |

|  |                                |
|--|--------------------------------|
| POSTLEITZAHL   | <b>Ext_Ort_PLZ_AT_DE</b>       |
|  |                                |
|  TO_CO_ABS_AUTOREN          | <b>Autoren</b>                 |
| GESCHLECHT   | <b>Autoren_Geschlecht</b>      |
| MATRIKELNUMMER   | <b>Autoren_Matrikelnr</b>      |
| NACHNAME   | <b>Autoren_Nachname</b>        |
| TITEL  | <b>Autoren_Titel</b>           |
| VORNAME  | <b>Autoren_Vorname</b>         |
|  |                                |
|  TO_STUD_PU_ABSCHLUESSE     | <b>Abschluesse</b>             |
| ABSCHLUSSARTKB   | <b>AbschlussartKB</b>          |
| ABSCHLUSSARTNAME   | <b>Abschlussartbezeichnung</b> |
| <b>„Pv Abs Typen“ kommt noch</b>   |                                |
|  |                                |
|  TO_CO_ABS_TITEL_KURZFASS | <b>Abschlussarbeiten</b>       |
| ORIGINAL_SPRACHE_FLAG  | <b>Abs_Original_Sprache</b>    |
| SCHLAGWOERTER  | <b>Abs_Schlagwoerter</b>       |
| SPRACHE_NR   | <b>Abs_Sprache_Nr</b>          |
| TITEL  | <b>Abs_Titel</b>               |
|  |                                |
|  TO_FODOK_FD_SPRACHEN     | <b>Sprachen</b>                |
| NAME   | <b>Sprachen_Bezeichnung</b>    |
|  |                                |
|  TO_TUG_NEW_ADRESSEN      | <b>Ext_Org_Adressen</b>        |
| ORT_AUSLAND  | <b>Ext_Org_Ort_Ausland</b>     |
| POSTLEITZAHL_AUSLAND   | <b>Ext_Org_PLZ_Ausland</b>     |
| STRASSE_HAUSNUMMER   | <b>Ext_Org_Strasse_Hausnr</b>  |

|  |                 |
|--|-----------------|
|  TO_FODOK_FD_SPRACHEN | <b>Sprachen</b> |
|--|-----------------|

|  |                                    |
|--|------------------------------------|
| NAME   | <b>Sprachen_Bezeichnung</b>        |
|  |                                    |
|  TO_FODOK_FD_TITEL_BESCHREI   | <b>Titel_Beschreibung</b>          |
| FD_FA_NR   | <b>Forschungsaktivitaet_Nr</b>     |
| FD_SPRACHE_NR  | <b>Sprache</b>                     |
| KURZFASSUNG_NEU_TMP  | <b>Kurzfassung</b>                 |
| ORIGINAL   | <b>Original</b>                    |
| SCHLAGWORTE  | <b>Schlagworte</b>                 |
| TITEL  | <b>Forschungsaktivitaet_Titel</b>  |
|  |                                    |
|  TO_FODOK_FORSCHUNGS_TYPEN    | <b>Forschungstypen</b>             |
| NAME   | <b>Forschungstypen_Bezeichnung</b> |
|  |                                    |
|  TO_FODOK_FORSCHUNGSEINR_FA | <b>Koord_Forschungseinrichtung</b> |
| KOORDINATOR_FLAG   | <b>Koordinator_Flag</b>            |
|  |                                    |
|  TO_FODOK_KOOP_PARTNER      | <b>Ext_Org_Partner</b>             |
| BEZEICHNUNG  | <b>Ext_Org_Partne_Bezeichnung</b>  |
| EMAIL  | <b>Ext_Org_Partner_Email</b>       |
| EXT_ORG_NAME   | <b>Ext_Org_Partner_Name</b>        |
| EXT_ORG_NAME_MASTERORG_L1  | <b>Ext_Org_Partner_Abt1</b>        |
| EXT_ORG_NAME_MASTERORG_L2  | <b>Ext_Org_Partner_Abt2</b>        |
| EXT_ORG_TYP  | <b>Ext_Org_Partner_Typ</b>         |
| EXT_ORG_TYP_KURZBEZ  | <b>Ext_Org_Typ_Kurzbez_</b>        |
| GUELTIG_BIS  | <b>Ext_Org_Partner_Gueltig_bis</b> |
| LAND_CODE  | <b>Ext_Org_Partner_Land_Code</b>   |
| ORT  | <b>Ext_Org_Partner_Ort</b>         |
| ORT_AUSL   | <b>Ext_Org_Partner_Ort_Ausl</b>    |
| PLZ  | <b>Ext_Org_Partner_PLZ</b>         |
| PLZ_AUSL   | <b>Ext_Org_Partner_PLZ_Ausl</b>    |

|   |  |
|---|--|
| STRASSE   | <b>Ext_Org_Partner_Strasse</b>           |
| TEL_NR  | <b>Ext_Org_Partner_Tel_Nr</b>            |
| TYP   | <b>Typ_KP_GG</b>                         |
|   |  |
|  <b>TO_FODOK_ARB_GEBIETE</b>       | <b>Arbeitsgebiete</b>                    |
| TITEL   | <b>AG_Titel</b>                          |
| FD_FA_NR_UNTER  | <b>AG_Nummer</b>                         |
|   |  |
|  <b>TO_TUG_NEW_ORGANISATIONEN</b>  | <b>Institute</b>                         |
| FAK_KENNUNG   | <b>Fak_Kennung</b>                       |
| GUELTIG_AB  | <b>Inst_Gueltig_Ab</b>                   |
| GUELTIG_BIS   | <b>Inst_Gueltig_Bis</b>                  |
| KENNUNG   | <b>Inst_Kennung</b>                      |
| NAME  | <b>Inst_Bezeichnung</b>                  |
|   |  |
|  <b>TO_FODOK_FORSCHUNGSAKTIV</b> | <b>Forschungsaktivitaet</b>              |
| AENDERUNGSDATUM   | <b>Aenderungsdatum</b>                   |
| BEGINN_NEU  | <b>Beginn</b>                            |
| EMAIL_ADRESSE   | <b>Email_Adresse</b>                     |
| ENDE_NEU  | <b>Neu</b>                               |
| ERFASSUNGSDATUM   | <b>Erfassungsdatum</b>                   |
| ERSTELLT_AM   | <b>Erstellt_Am</b>                       |
| ERSTELLT_VON  | <b>Erstellt_Von</b>                      |
| EXT_FDFMT_NR  | <b>Ext_FDFMT_Nr</b>                      |
| EXT_FDFMT_NR_B  | <b>Ext_FDFMT_Nr_B</b>                    |
| FD_FA_GEAENDERT_AM  | <b>Forschungsaktivitaet_Geaendert_Am</b> |
| FOERDERUNGSBETRAG   | <b>Foerderungsbetrag</b>                 |
| GEAENDERT_AM  | <b>Geaendert_Am</b>                      |
| GEAENDERT_VON   | <b>Geaendert_Von</b>                     |
| SORTIERUNG  | <b>Sortierung</b>                        |

|   |                                    |
|---|------------------------------------|
| URL   | Url                                |
|   |                                    |
|  TO_FODOK_FD_PERSONEN      | <b>Personen</b>                    |
| EMAIL_ADRESSE   | <b>Email_Adresse</b>               |
| FD_P_FUNK_NR  | <b>Funktions_Nr</b>                |
| FUNKTION  | <b>Funktion</b>                    |
| NACHNAME  | <b>Nachname</b>                    |
| SONSTIGES   | <b>Sonstiges</b>                   |
| TITEL   | <b>Titel</b>                       |
| VORNAME   | <b>Vorname</b>                     |
|   |                                    |
|  TO_FODOK_FD_GELDGEBER     | <b>Ext_Org_Geldgeber</b>           |
| FD_PROGRAM_NR   | <b>Programm_Nr</b>                 |
| FOERDERPROGRAMM   | <b>Foerderprogramm</b>             |
| FOERDERUNGSBETRAG   | <b>Foederungsbetrag</b>            |
| NAME  | <b>GG_Name</b>                     |
| PROJEKTNUMMER   | <b>GG_Projektnummer</b>            |
|   |                                    |
|  TO_FODOK_FD_KOOPPARTNER | <b>Ext_Org_KP_Partner</b>          |
| KONTAKTPERSON   | <b>Ext_Org_KP_Kontaktperson</b>    |
| KOORDINATOR_FLAG  | <b>Ext_Org_KP_Koordinator_Flag</b> |
| NAME  | <b>Ext_Org_KP_Name</b>             |