

**Friedrich-Alexander-Universität  
Erlangen-Nürnberg**



*Data Warehouse  
zur Prognose von  
Studierendenzahlen  
in Lehrveranstaltungen*

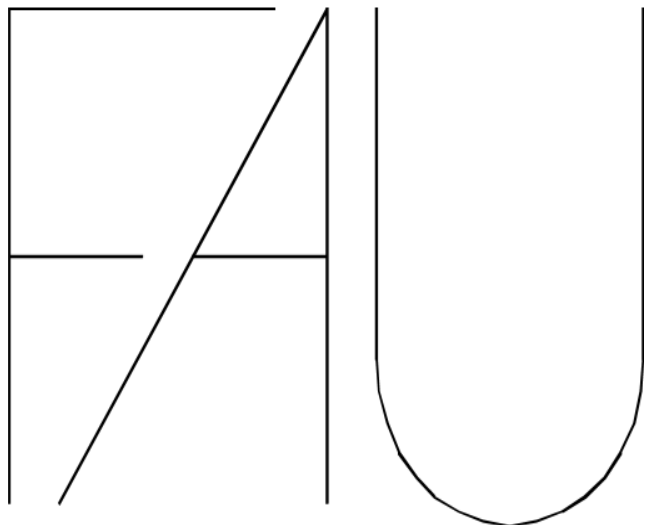
Studienarbeit

*André Kreuter*

Lehrstuhl für Informatik 6  
(Datenmanagement)

Department Informatik  
Technische Fakultät

Friedrich-Alexander-  
Universität  
Erlangen-Nürnberg





# Data Warehouse zur Prognose von Studierendenzahlen in Lehrveranstaltungen

Studienarbeit im Fach Informatik

vorgelegt von

**André Kreuter**

geb. 22.12.1984 in Castrop - Rauxel

angefertigt am

**Department Informatik  
Lehrstuhl für Informatik 6  
Datenmanagement  
Friedrich-Alexander-Universität Erlangen–Nürnberg  
(Prof. Dr. Klaus Meyer-Wegener)**

Betreuer: Prof. Dr. Richard Lenz  
Dipl.-Inf. Christoph Neumann

Beginn der Arbeit: 01.05.2008  
Abgabe der Arbeit: 28.02.2009



Ich versichere, dass ich die Arbeit ohne fremde Hilfe und ohne Benutzung anderer als der angegebenen Quellen angefertigt habe und dass die Arbeit in gleicher oder ähnlicher Form noch keiner anderen Prüfungsbehörde vorgelegen hat und von dieser als Teil einer Prüfungsleistung angenommen wurde. Alle Ausführungen, die wörtlich oder sinngemäß übernommen wurden, sind als solche gekennzeichnet.

Der Universität Erlangen-Nürnberg, vertreten durch die Informatik 6 (Datenmanagement), wird für Zwecke der Forschung und Lehre ein einfaches, kostenloses, zeitlich und örtlich unbeschränktes Nutzungsrecht an den Arbeitsergebnissen der Studienarbeit einschließlich etwaiger Schutzrechte und Urheberrechte eingeräumt.

Erlangen, den 28.02.2009

André Kreuter



# Kurzzusammenfassung

## Data Warehouse zur Prognose von Studierendenzahlen für Lehrveranstaltungen

Prognosen von Studierendenzahlen für Lehrveranstaltungen aufzustellen ist ein allgegenwärtiges Problem an der Friedrich-Alexander-Universität Erlangen-Nürnberg. Hierbei stellt sich die Frage nach der Bereitstellung von Räumen, Ausbauplanungen, Personaleinstellungen und gezielten Förderungen von Studierenden. Das Department Informatik ist ein Großdienstleister für exportierte Lehrveranstaltungen: Es bietet nicht nur den an der Technischen Fakultät angesiedelten Studiengängen die Möglichkeit, sich im Bereich der Informatik zu bilden und/oder weiterzubilden, sondern auch Studiengängen, die an anderen Fakultäten angesiedelt sind, beispielsweise an der Naturwissenschaftlichen Fakultät oder der Philosophischen Fakultät mit Fachbereich Theologie. Diese Studienarbeit beschäftigt sich mit der Problematik der Hochrechnung von Studierendenzahlen für Lehrveranstaltungen. Zur Analyse der Studierendenzahlen wird ein Data Warehouse eingesetzt, welches auf strukturellen Informationen der Universität basiert und das Verhalten der Studierenden berücksichtigt. Gleichzeitig muss für den Studierenden die Übersicht bereitgestellt werden, welche Fächer sie/er in der Informatik hören kann bzw. muss. Des Weiteren muss neuen Studiengängen, die Lehrangebote im Bereich der Informatik wahrnehmen wollen, innerhalb kurzer Zeit mitgeteilt werden können, welche Veranstaltungen fachlich und kapazitätsmäßig in Frage kommen.





# Inhaltsverzeichnis

<b>Abbildungsverzeichnis</b>	vii
<b>Tabellenverzeichnis</b>	ix
<b>1 Einleitung</b>	1
1.1 Problemdefinition	2
1.1.1 Allgemeine Problembeschreibung	2
1.1.2 Probleme im Detail	3
1.1.3 Veraltete Informationen	4
1.2 Motivation	5
1.3 Ziele	5
<b>2 Methodik</b>	7
<b>3 Vorläufersystem</b>	11
3.1 Wie wird das Problem aktuell gelöst?	11
3.1.1 Pflichtfach:	12
3.1.2 Wahlpflichtfach:	13
3.1.3 Nebenfach:	13
3.1.4 Schlüsselqualifikation:	13
3.2 Bewertung	14
3.3 Verbesserungspotential	15
<b>4 Datenquellen</b>	17
4.1 Bewertung der Datenquellen	18
4.2 Verwendete Datenquellen	22
4.3 Datenquelle Fachprüfungsordnungen	23
<b>5 Konzeptioneller Entwurf</b>	25
5.1 Anforderungsanalyse	25
5.2 Modellierung eines idealen Schemas	32
5.3 Vereinfachung des Schemas	35

5.4	Multidimensionales Model.....	40
<b>6</b>	<b>Systementwurf</b>	<b>49</b>
6.1	Basisdatenbank.....	49
6.1.1	Basisdatenbank ein Operational Data Store?.....	50
6.2	Logische Ebene.....	51
6.3	Datenbeschaffung.....	57
6.3.1	Campus IT Webservice.....	58
6.3.2	Selbstpflege.....	63
6.4	Data Warehouse System.....	66
6.4.1	Laden des Data Warehouse.....	70
6.5	Analytischer Prozess.....	70
6.5.1	Studierendenberatung bzgl. Lehrveranstaltung in der Informatik	71
6.5.2	Berichte an das Department.....	71
6.5.3	Studierendenanzahlen in Lehrveranstaltungen.....	71
<b>7</b>	<b>Technik</b>	<b>75</b>
7.1	Kommerzielle Data Warehouse System.....	75
7.1.1	Microsoft SQL Server mit der Business Intelligent.....	75
7.1.2	IBM DB2 mit InfoSphere Warehouse Enterprise Edition.....	76
7.2	Open Source Data Warehouse Systeme.....	77
7.2.1	Pentaho.....	77
7.2.2	JasperSoft.....	78
7.3	Die technische Umsetzung.....	78
7.3.1	Installation.....	78
7.3.2	Basisdatenbank: Schema anlegen.....	80
7.3.3	Laden der Daten.....	80
7.3.4	Pentaho Schema Workbench.....	82
7.3.5	Analyse am JasperSoft Server.....	83
7.3.6	IReport.....	84
<b>8</b>	<b>Evaluation</b>	<b>87</b>
8.1	Datenquellen.....	87
8.2	Konzeptioneller Entwurf.....	88
8.3	Systementwurf.....	90
8.4	Technik.....	92
8.5	Ziele erreicht?.....	93

<b>9</b>	<b>Zusammenfassung</b>	95
<b>A</b>	<b>Zusatzinformationen zu Kapitel ??</b>	97
	A.1 Standortproblematik .....	97
	A.2 Umfrage an die Studierende bzgl. ihrem Lehveranstaltungsbesuchverhalten	
97		
<b>B</b>	<b>Anleitungen</b>	99
	B.1 MySQL Administrator - Remote Zugriff einrichten .....	99
	B.2 Laden der Basisdatenbank mit JasperETL .....	100
	B.2.1 Input: Excel File .....	101
	B.2.2 Input: CSV File .....	101
	B.2.3 Input: DBS .....	101
	B.2.4 Verbindung .....	102
	B.3 Laden des Data Warehouses mit JasperETL .....	102
	B.4 Pentaho Schema Workbench, Schema Erstellung .....	104
	B.4.1 Dimension .....	104
	B.4.2 Table .....	105
	B.4.3 Join .....	105
<b>C</b>	<b>Dokumentation</b>	107
	C.1 Zuordnung Identifikationsnummern zur Abschlussart .....	107
	C.2 Zuordnung Identifikationsnummer zum Studiengang .....	108
	<b>Literaturverzeichnis</b>	113



# Abbildungsverzeichnis

1.1	William Somerset Maugham . . . . .	1
3.1	Tabellen in der Access Datenbank . . . . .	11
4.1	Taxonomie von Datenqualitätsmerkmalen . . . . .	19
5.1	Pflege der Dozenten mit Verantwortlichkeit . . . . .	28
5.2	USE-Case Abhängigkeiten Teil1 . . . . .	30
5.3	USE-Case Abhängigkeiten Teil2 . . . . .	31
5.4	FAU Struktur in E/R-Notation . . . . .	33
5.5	Department zu Jahrgang in E/R-Notation . . . . .	34
5.6	FPO, Module, Modulgruppen, Lerveranstaltungsblock in E/R-Notation	36
5.7	E/R-Diagramm der Basisdatenbank Teil1 . . . . .	37
5.8	E/R-Diagramm der Basisdatenbank Teil2 . . . . .	38
5.9	Optimiertes E/R-Diagramm der Basisdatenbank Teil1 . . . . .	41
5.10	Optimiertes E/R-Diagramm der Basisdatenbank Teil2 . . . . .	42
5.11	Dimensionsmodell . . . . .	47
6.1	Skizzierte Berechnung der Studierendenzahlen . . . . .	73
7.1	USE-Case der zu ladenden Daten . . . . .	80



# Tabellenverzeichnis

4.1	Datenquellenvergleich . . . . .	21
7.1	Data Warehouse Systemevergleich . . . . .	79





# 1 Einleitung

“Zukunft: Etwas, das die meisten Menschen erst lieben, wenn es Vergangenheit geworden ist.“ (William Somerset Maugham)



Abbildung 1.1: William Somerset Maugham

Dieses Zitat beschreibt auf seine Art die Bewertung der Data Warehouse Systeme, als sie entwickelt wurden. Die Idee des Data Warehouse existiert schon seit den späten sechziger Jahren. Zuerst wurde ihr wenig Beachtung geschenkt, doch seit den 90er Jahren sind Data Warehouse Systeme in vielen Bereichen nicht mehr weg zu denken. Sei es, um Firmen einen Wettbewerbsvorsprung durch Informationsfluss von regionalen Produktverkäufen zu verschaffen oder sei es in der Klimaforschung, große Datenmengen mittels eines Data Warehouse Systems zu analysieren.

Den Begriff Data Warehouse verbindet man mit dem Wort Informationsbereitstellung. Ziel ist es, eine große Menge an Daten viele Analysen laufen zu lassen. Dabei soll die Informationsbereitstellung zeitnah, fehlerfrei, flexibel, ergonomisch, effizient, effektiv und inspirativ erfolgen. [AB01]

Die effiziente und effektive Analyse von Daten wird heute nicht mehr nur in der Forschung an Universitäten benötigt, sondern auch in deren Verwaltung.

### 1.1 Problemdefinition

Die folgenden Abschnitte beschreiben die Schwierigkeit, die Studierendenzahlen einer Lehrveranstaltung im Voraus abzuschätzen. In Abschnitt 1.1.1 wird im Groben das Problem umrissen und in Abschnitt 1.1.2 sequenziell die einzelnen Ursachen der Probleme dargestellt. Abschließend wird in 1.1.3 noch diskutiert, worin die Schwierigkeit besteht, die Studierenden in Bezug auf Informatik-Lehrveranstaltungen gut zu beraten und der Studienkommission zu berichten, welche Studiengänge die Lehrveranstaltungen des Departements Informatik besuchen.

#### 1.1.1 Allgemeine Problembeschreibung

Das Department Informatik mit seinen zwölf Lehrstühlen und dem Fachbereich der Lehrerbildung, auch Didaktik der Informatik genannt, gehört zu den größten Departments der Universität Erlangen-Nürnberg. Die durch die Größe begründete Personalstärke bietet den heimischen Studiengängen, Informatik, Computational Engineering, Informations- und Kommunikationswissenschaft sowie Informatik Lehramt für Gymnasien ein ausgewogenes und vielseitiges Spektrum an Lehrveranstaltungen.

Durch die weitgehende Automatisierung vieler Fertigungsprozesse von Produkten und dem Ausbau von Informationsverarbeitungssystemen sind heute Informatikkenntnisse in vielen Ingenieurberufen essentiell geworden. Aus diesem Grund bietet das Department Informatik nicht nur Veranstaltungen für ihre eigenen Studiengänge an, sondern auch für externe Studiengänge wie z.B. den Studiengängen der Technischen Fakultät.

Es ist offensichtlich, dass moderne wissenschaftliche Versuchsaufbauten, bei denen Messinstrumente Unmengen von Messdaten liefern, nicht realisierbar wären, wenn diese Daten noch händisch verarbeitet werden müssten. Die Messwerte, resultierend aus Versuchen der Biologie, Chemie und Physik, werden heute mit Computern verarbeitet. Um die Daten an Computern effizient auswerten zu können, müssen die Naturwissenschaftler auch Grundkenntnisse der Informationsverarbeitung besitzen. Auch die Mathematik verwendet zum Lösen von schwierigen Fragenstellungen Cluster-Computer, um die Antworten schnell zu erhalten. Um diese aufwendigen Berechnungen anzustellen, müssen Kenntnisse der Hardwarearchitektur der Computer und die Fähigkeit, algorithmische Probleme in Programmiersprachen zu verfassen, vorhanden sein. Für diese Problematik der anderen Studiengänge, bietet die Informatik auch ihre Dienste im Sinne von Lehrveranstaltungen an.

Selbst zwei Studiengänge der Philosophischen Fakultät mit Fachbereich Theologie besuchen Lehrveranstaltungen der Informatik, Kulturgeographik- und Computerlinguistik-Studierende brauchen für ihre Themenbereiche die Unterstützung von Informationsverarbeitungssystemen. Aus diesem Grund besuchen diese beiden Studiengänge die Lehrveranstaltungen der "Grundlagen der Informatik". Andere Studiengänge dieser Fakultät können Lehrveranstaltungen der Informatik als Schlüsselqualifikation anerkennen lassen, z. B. die Buchwissenschaft.

Für viele externe Studiengänge gibt es spezielle Einführungsveranstaltungen, die weiter mit Speziallehrveranstaltungen der Informatik gekoppelt werden. Oft wird bei diesen Studiengängen nicht genau vorgeschrieben, was die Studierenden hören müssen und wann sie es hören. Somit kann eine genaue Berechnung der Besucherzahlen einer Lehrveranstaltung schlecht im Voraus erfolgen. Zusätzlich kommen noch feingranulare Probleme, die im nächsten Abschnitt erläutert werden.

### 1.1.2 Probleme im Detail

#### **Strukturelle Veränderung der Friedrich-Alexander-Universität Erlangen-Nürnberg**

Zum Wintersemester 2007/2008 hat sich die Friedrich-Alexander-Universität Erlangen-Nürnberg mit dem Beschluss des Senats vom 7. Februar 2007 ein neues Gesicht gegeben: Aus elf Fakultäten wurden fünf. Durch diese Zusammenlegung haben die Dekanate insgesamt mehr Räume zu verwalten. Die meisten Fakultäten sind nicht nur an einem Standort angesiedelt, sondern gleich an mehreren. So können jetzt, um eine optimale Lastenverteilung des vorhandenen Raumvorrats zu erreichen, die Lokationen innerhalb einer Fakultät gewechselt werden. Dies hat zur Folge, dass die Studierenden öfter pendeln müssen, was sich in der Motivation, eine Vorlesung zu besuchen, niederschlägt. Dieses Phänomen liegt besonders bei der Naturwissenschaftlichen Fakultät und bei der Philosophischen Fakultät mit Fachbereich Theologie vor. Bei diesen beiden Fakultäten müssen die Studierenden bereits für den Besuch der Hauptveranstaltungen pendeln. Wenn dann noch Informatikveranstaltungen, die an der Technischen Fakultät angeboten werden, hinzukommen, kann dies zu einem Interessenverlust bei den Studierenden führen.

#### **Studienbeiträge kollidieren mit den Veranstaltungen**

Durch die Einführung von Studienbeiträgen werden die Studierenden vor neue Probleme gestellt, die einen Einfluss auf den Besuch von Lehrveranstaltungen haben. Die meisten Studierenden der Friedrich-Alexander-Universität Erlangen-Nürnberg

kommen aus Bayern [EN09d], und sind dann für ihr Studium nach Erlangen oder Nürnberg gezogen. 76% der Studierenden leben nicht mehr zu Haus. Das heißt, sie müssen noch Miete für ihre Wohnung oder Wohnheimszimmer auf bringen. Erschwerend kommt hinzu, dass ca. 43% der Studierenden im Monat nur zwischen 200 bis 500 Euro zur Verfügung stehen, um ihren Lebensunterhalt zu bestreiten, [UEN09]. Hochgerechnet gehen 70% aller Studierenden für ihren Lebensunterhalt und die zu entrichtenden Studienbeiträge arbeiten. Da bestimmte Studiengänge ihre Klausuren in den Semesterferien schreiben lassen, was bei der Technischen Fakultät praktisch bei allen Studiengängen der Fall ist, sind die Studierenden fast gezwungen, während der Vorlesungszeit arbeiten gehen. Durch die Einführung der Bachelor- und Master-Studiengänge gibt es weniger Wahlmöglichkeit in Bezug auf die Lehrveranstaltungen, und somit ist die Möglichkeit, Veranstaltungen zu verschieben, marginal. Daher werden Veranstaltungen teilweise nicht besucht, weil die Studierenden ihren Lebensunterhalt verdienen müssen.

### **Skripte online, wieso Vorlesung?**

Das Verhalten der Studierenden ist ein entscheidender und nur schwer berechenbarer Aspekt bei der Bestimmung der Studierendenzahlen für Veranstaltungen. Durch die Veröffentlichung von Skripten und Foliensätzen zu Vorlesungen oder von Musterlösungen für Übungen ist die Hemmschwelle deutlich geringer geworden, eine Veranstaltung nicht zu besuchen und sich mit dem Inhalten im Selbststudium auseinanderzusetzen.

### **1.1.3 Veraltete Informationen**

Die in oben gezeigten vielseitigen Möglichkeiten führen zu einem organisatorischen Problem. Viele Studiengänge heißt auch viele Fachprüfungsordnungen und Studienführer. Meist wurde der Beschluss von anderen Studiengängen, etwas aus der Informatik zu hören, nicht mit dem Department Informatik genau koordiniert, und deswegen gibt es kein einheitliches Schema. So müssen alle FPOs<sup>1</sup> und Studienführer genau gesichtet werden und wenn es nötig ist, die alten Lehrveranstaltungen, die in den FPOs beschrieben sind, auf die neue Lehrveranstaltungen des Department Informatik anpassen. Es kommt momentan vor, dass in FPOs Lehrveranstaltungen stehen, die die Informatik nicht mehr anbietet, weil sie durch eine neue ersetzt wurden.

---

<sup>1</sup>Fachprüfungsordnung

## 1.2 Motivation

In den letzten Semestern gab es einen Umbruch an der Friedrich-Alexander-Universität Erlangen-Nürnberg. Nicht nur, dass sich die Universität selbst neu aufgestellt hat (aus elf Fakultäten wurden fünf), so hat auch der Bologna-Prozess Einzug gehalten. Diese Änderungen spiegeln sich nicht nur in neuen Prüfungsordnungen wider, sondern auch im Leben der Studierenden.

Das Department Informatik ist eines der Departments, die ihre Lehrveranstaltungen anderen Studienfächern am meisten öffnet. Nicht nur für die heimischen technischen Studienfächer werden die Vorlesungsangebote bereit gestellt, sondern auch Studienfächern und Studiengängen anderer Fakultäten, wie beispielsweise der Naturwissenschaftlichen Fakultät oder der Philosophischen Fakultät mit Fachbereich Theologie.

Durch die Vielfalt der einzelnen Studienfächer und Studiengänge, für die die Informatik ihre Veranstaltungen zugänglich macht, tritt bei den Dozenten immer zu Semesteranfang eine Frage auf; "Wie groß muss der Raum für meine Veranstaltung sein, damit er einerseits nicht zu groß ist und andererseits auch nicht zu klein ist, um allen Studierenden die Möglichkeit zu geben, einen Platz finden". Diese Frage ist so schwer zu beantworten, weil in den meisten Prüfungsordnungen nicht explizit steht, in welchem Semester die Veranstaltungen der Informatik gehört werden sollen.

Weiterhin muss der Exportbeauftragte des Departments Informatik der Kollegialen Leitung und der Studienkommission die Modulexport-Situation, also wer was wann besucht sowie die Kapazitäten, aufschlüsseln.

Die Motivation dieser Studienarbeit besteht darin, ein Data Warehouse zu entwickeln, das auf Basis der Informationen von vorhergehenden Semestern eine sichere Prognose erstellen kann. Diese Informationen bestehen daraus,

1. wie viele Studierende in eine Veranstaltung kommen und
2. für welche Veranstaltungen von welchem Studiengang besucht werden.

## 1.3 Ziele

Ziel dieser Arbeit ist die Entwicklung und Umsetzung eines Data Warehouses für:

- die effektive und effiziente Analyse zur Bestimmung der Studierendenanzahlen,
- für die Anmeldung der Veranstaltungsräume,

sowie:

- eine genau Untersuchung der Universitätsstruktur,
- eine Studienfach- und Studiengangabbildung,
- eine Abbildung der Verantwortlichen für einen Studiengang
- für die Entwicklung eines Schemas für die Basisdatenbank,

um schnelle Auskunft gegenüber den Studierenden und dem Department zu erteilen. Weiter soll diskutiert werden, ob der existierende Zustand für den Analyseprozess zweckdienlich ist oder ob durch Generalisierung oder Spezifizierung an dem Schema eine Anpassung erfolgen soll. Durch unterschiedliche Informationsquellen an Studierendenzahlen in Bezug auf Lehrveranstaltungen sollte für die Analyse die Aussagekraft der einzelnen Informationsquellen erörtert werden.

## 2 Methodik

Nach dem die Hauptziele feststehen, Bestimmung der Studierendenzahlen in Lehrveranstaltungen und Studierendenberatung für departmentsfremde Studiengänge bzw. Berichterstattung an das Department 1.2, müssen Informationen für die Bewältigung der Aufgabe gesammelt werden.

Vergleicht man die Studierendenzahlen, die an einer Lehrveranstaltung teilnehmen und die Studierendenzahlen, die eigentlich an einer Lehrveranstaltung teilnehmen sollten, so stellt man fest, dass diese Zahlen von einander abweichen. Es stellt sich daher die Frage, warum Studierende die Lehrveranstaltungen nicht besuchen. Um dieser Frage nach zu gehen wurden Studierende befragt. Bei der Umfrage nahmen zweiundfünfzig Studierende teil. Neben der Frage, warum die Studierende nicht in Lehrveranstaltungen gehen, wurden die Gründe noch thematisiert. Die Ergebnisse sind im Kaptiel 1.1 in der Problemdarstellung beschrieben. Nur die Hauptgründe wurden erwähnt. Weitere genannte Begründungen, die aber in den Persönlichkeitsbereich des Studenten bzw. Studentin fallen wurden außen vor gelassen.

Die Informationen, die für die Studierendenberatung und Berichterstattung ans Department Informatik benötigt wurden, werden in den Fachprüfungsordnungen gefunden. Hier musste keine Suche bzgl. Informationsfindung durchgeführt werden.

Die benötigten Informationen liegen nun vor. Jetzt stellt sich die Frage, woher sind diese Informationen bekommen und wie gut sind sie für die Ziele geeignet.

Die Fachprüfungsordnung findet man auf der Webseite des Prüfungsamtes. Somit war die Suche schnell erledigt

Die Informationen bezüglich Studierendenzahlen in Vorlesung liegen nicht vor. Es existieren vier Datenquellen, die die Studierendenzahlen in Lehrveranstaltungen indirekt zu einem gewissen Grad widerspiegelt.

Um die verschiedenen Datenquellen zu vergleichen, wurde die Taxonomie von Heinrichs [Hei02] verwendet. Anhand diesen Kriterien wurden die Datenquellen, die schlussendlich verwendet werden sollen ausgewählt. Dieser Prozess wird in Kaptitel 4 beschrieben.

Da die Datensätze nicht genau die Studierendenanzahlen in Lehrveranstaltungen wieder spiegeln und auch die Fachprüfungsordnung nicht unabhängig betrachtet werden kann, muss die Abhängigkeiten von den Datenquellen und den Fachprüfungsordnungen ermittelt werden. Diese Daten müssen konsistent ins Datenbanksystem eingepflegt werden. Die Abhängigkeiten wurden mittels USE-Case Diagrammen [MH05] aus der UML <sup>1</sup> beschrieben. Zum Zeichnen der USE-Case-Diagramme wird Visual Paradigm [Par09] verwendet. Nicht nur die Abhängigkeiten wurden betrachtet, sondern auch wie die Informationen verarbeitet werden, die der Exportbeauftragte für seine Arbeit benötigt. Zu finden ist das in Kapitel 5.1. Für die UML-Modulierung wurde Visual Paradigm verwendet.

Aus den Anforderungen wird dann ein E/R-Diagramm für das Datenbanksystem erstellt. Hier gehen die Informationen aus der Anforderungsanalyse aus Abschnitt 5.1 ein. Neben den Anforderungen wurden die Informationen für die Universitätsstruktur [EN09h] benötigt. Für die Entwicklung des E/R-Diagramm wurde Microsoft Visio [Gmb09d] verwendet mit der Zusatzerweiterung "Conceptual Database Modeling ERD / EERD". Dieser Schritt wird in Abschnitt 5.2 beschrieben.

Beim Abbilden der realen Universitätsstruktur ins E/R-Diagramm, kam es immer dazu, dass Redundanz oder zu viele Informationen auftraten, die später einen zusätzlichen Pflegeaufwand bedeuten würde. Aus diesem Grund, wird das E/R-Diagramm dahin gehend optimiert. Dies wird in Abschnitt 5.3 beschrieben.

Es soll ein Data Warehouse entwickelt werden. Für diesen Zweck, wird noch ein multidimensionales Schema für das Data Warehouse benötigt. Dieses wird auf Basis des optimierten E/R-Diagramms entwickelt. Für die Virtualisierung dieses Schemas wurde Microsoft Visio verwendet mit der Zusatzerweiterung "Conceptual Database Modeling ERD / EERD". 5.4

Basierend auf dem konzeptionellen Entwurf wird der logische Entwurf entwickelt. Hierfür muss das E/R-Diagramm und das multidimensionale Schema auf Relationen abgebildet werden. Zu erst wird diskutiert, welche Art von Datenbanksystem verwendet werden soll. Als Bewertungskriterien, werden die vorhandenen Datenbanksysteme verwendet. Das Verfahren, um das E/R-Diagramm aus Abschnitt 5.3 in Relationen umzusetzen, wurde aus dem Buch "Datenbanksysteme" [AK06] entnommen. Die Relationen werden erst als Text und dann symbolisch dargestellt.

Um an die Daten der Datenquellen, die in Kapitel 4 beschrieben werden, zu erlangen, wird ein Webservice verwendet. Mein Campus stellt einen Webservice zu Verfügung, der mit meinem Client abgerufen werden kann. Die Anfragen für den Webservice und welche Daten von ihm zurück gesendet werden, wird anhand des Datenbestandes

---

<sup>1</sup>Unified Modeling Language



---

von Mein Campus diskutier. Für die Entwicklung des Clients wurde Java mit der Entwicklungsumgebung Eclipse [ECL09] verwendet.

Wenn die Daten vorhanden sind, ist die Frage, in welcher Reihenfolge werden die Daten ins System geladen, zu beantworten. Hier wird eine Reihenfolge definiert, die sich auf die Problematik der Fremdschlüssel bezieht. Weiter wird unterschieden zwischen dem ersten Laden des Datenbanksystems und dem Nachladen des Systems.

Die Umsetzung in Relationen, der Webservice und das Beladen des Datenbanksystems werden in Kapitel 6 beschrieben.

Die Entscheidung, welches Datenbanksystem mit Data Warehouse verwendet wird, stand danach an. Diese Entscheidung wurde abgenommen, weil ein System vorgegeben wurde. Um mehrerer Möglichkeiten zu zeigen, wird ein Vergleich zwischen zwei kommerziellen und zwei Open Source Lösung beschrieben. Als Vergleichsparameter werden die positiven und negativen Eigenschaften, des vorgegeben System her ran gezogen.

Weiter wird das verwendete System beschrieben und nach positiven und negativen Merkmale der Lösung differenziert.

Die Technik wird in Kapitel 7 beschrieben.

Nach der technischen Umsetzung wird das Projekt kritisch evaluiert. Dabei werden die Vorteile, Nachteile sowie die Verbesserungsmöglichkeiten, die die Umsetzung dieses Projektes vereinfacht hätten, beschrieben. Dies wird in Kapitel 8 ausgeführt.



# 3 Vorläufersystem

## 3.1 Wie wird das Problem aktuell gelöst?

Die Probleme der Berechnung der Studierendenzahlen, die im Kapitel 1.1 erläutert wurden, sind nicht nur durch die Einführung der Bachelor-Studiengänge entstanden; die Problematik gab es bereits schon mit den alten Studiengängen. Zu den alten Studiengängen zählen beispielsweise die Magister- und Diplom-Studiengänge. Die Informatik bietet schon länger anderen Studiengängen die Möglichkeiten an, etwas aus ihrem Angebot an Lehrveranstaltungen zu besuchen.

Um diesem Problem Herr zu werden, hat der damalige Exportbeauftragte des Departments Informatik eine Access-Datenbank angelegt. Sie ist ziemlich einfach aufgebaut und erfüllt nur den Tatbestand der Beratung der Studierenden und der Berichterstattung an das Departement. Dabei wurden die Lehrveranstaltungen für Diplom-Studiengänge eingetragen. Über die Zeit, vermischte sich der Name Modul mit Lehrveranstaltung.

Die Datenbank besteht aus vier Tabellen, die die wichtigsten Informationen beinhalten. Die vier Tabellen sind Semester, Studiengang, Module, Studienprogramm.

Beschreibung der Tabellen:

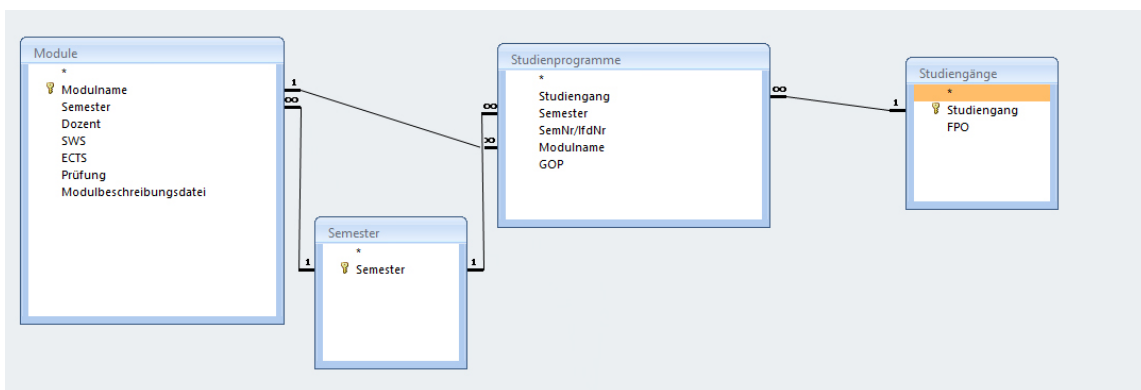


Abbildung 3.1: Tabellen in der Access Datenbank

Die Tabelle Semester besitzt das Attribut Semester, welches eine der folgenden Ausprägungen hat: Sommer- oder Wintersemester. Die Tabelle wird auch über das Attribut Semester identifiziert. Ein Schlüssel dient in einer relationalen Datenbank dazu, die Tupel einer Relation eindeutig zu identifizieren. Anschaulich kann man sich eine Relation als Tabelle vorstellen. Ein Schlüssel für eine solche Tabelle ist dann eine Gruppe von Spalten, die so ausgewählt wird, dass jede Zeile in dieser Gruppe eine einmalige Wertekombination hat.

Eine weitere Tabelle, Studiengang, besitzt als Primärschlüssel das Attribut Studiengang, welches den Namen eines Studiengangs repräsentiert. Das zweite Attribut der Tabelle ist ein Link, der zu der Fachprüfungsordnung eines Studiengangs zeigt. Der Link weist auf die Prüfungsordnungen, die sich in einem Verzeichnis auf dem lokalen Computer befinden. Die dritte Tabelle lautet Module, die - wie der Name schon sagt - Module beschreibt. Die Tabelle Module ist mit Vorsicht zu genießen. Das Wort Module spiegelt eher die Lehrveranstaltungen wieder. Module wurden bei der Einführung von den Bachelor- und Master-Studiengängen eingeführt. Sie definieren einen Lehrveranstaltungsblock, der über einen Themenbereich geht. Ein Modul hat mindestens eine Teilleistung. Teilleistungen decken je nach Anzahl der Teilleistungen einen gewissen Teil des gesamten Stoffinhaltes ab, deren Ende durch einen Leistungsnachweis geprüft wird. Identifiziert werden die Module Tabelle durch das Attribut Modulname. Die Tabelle hat weitere Attribute, um die Module zu beschreiben; dazu gehören folgende Attribute: Semester, Dozent, SWS <sup>1</sup>, ECTS <sup>2</sup>, Prüfungen und Modulbeschreibung. Tabelle vier lautet Studienprogramm, die neben den Fremdschlüsseln <sup>3</sup> Studiengang, Semester und Module, die auf die jeweiligen gleichnamigen Tabellen zeigen, aus den oben bereits erwähnten Tabellen besteht.

Für die schnelle Abfrage wurde ein Formular erstellt, bei dem man den Namen des Studiengangs eingeben muss. Daraufhin werden die unterschiedlichen Module angezeigt, die die Studierenden besuchen können. Zusätzlich werden die Informationen des Semesters, SWS, Prüfungen und ECTS ausgegeben. Beim Semester wird auch noch angeführt, welche Belegungsart es für den gewählten Studiengang gibt. Hierfür gibt es vier verschiedene Möglichkeiten, die im Folgenden näher erläutert werden.

#### 3.1.1 Pflichtfach:

In der Fachprüfungsordnung eines Studiengangs steht, dass der oder die Studierende die als Pflichtveranstaltung gekennzeichnete Veranstaltung im Studiumsverlauf besuchen

---

<sup>1</sup>Semesterwochenstunden

<sup>2</sup>European Credit Transfer System Punkte

<sup>3</sup>Ein Fremdschlüssel ist ein Sekundärschlüssel einer Relation, der in einer anderen Relation Primärschlüssel oder Schlüsselkandidat ist

muss. Oft gibt der Studienführer den Studierenden einen Vorschlag, in welchem Semester die Lehrveranstaltung besucht werden sollte.

### **3.1.2 Wahlpflichtfach:**

Das Wahlpflichtfach ist ein bestimmter Kanon von Lehrveranstaltungen eines Fachbereiches, die der oder die Studierende sich selbst aussuchen kann. Hier gibt es keine Unterscheidung, ob die Veranstaltungen aus dem eigenen Fachbereich kommen oder aus einem fremden stammen muss. Beispielsweise muss der Diplom-Informatiker ein Wahlpflichtfach für sein Vordiplom bestehen. Das Wahlpflichtfach ist in der Fachprüfungsordnung vorgeschrieben, das heißt, dass er ein Wahlpflichtfach belegen muss und es nicht weglassen kann. Der Studierende kann jetzt aussuchen, ob die Lehrveranstaltungen aus der Informatik kommt oder aus einem anderen Department wie der Physik.

### **3.1.3 Nebenfach:**

Das Nebenfach ist ähnlich aufgebaut wie das Wahlpflichtfach. Die Studierenden können hier auch aus einem Kanon an Lehrveranstaltung einen Fachbereich auswählen. Allerdings gibt es beim Nebenfach eine Beschränkung und zwar, dass hier keine Veranstaltungen gewählt werden dürfen, die aus dem eigenen Fachbereich kommen. Das Nebenfach ist in der Fachprüfungsordnung vorgeschrieben, das heißt, dass man ein Nebenfach belegen muss und es nicht weglassen kann.

### **3.1.4 Schlüsselqualifikation:**

Nicht nur die neue Bachelor-Studiengänge, sondern auch die alten Magister- oder Diplom-Studiengänge erlauben es, Zusatzleistungen als Schlüsselqualifikation anrechnen zu lassen. Diese Leistungen sind freiwillig und werden im Abschlusszeugnis vermerkt.

Wenn man kein Modul von einem Studiengang auslesen möchte, sondern ein Modul für den angegebenen Studiengang hinzufügen möchte, wird dies durch einen Klick direkt unter dem letzten Eintrag realisiert. Wie bei der Bedienung eines Tabellenkalkulationsprogramm, ist es möglich, neue Module aus der Informatik bei anderen Studiengängen einzutragen, sofern schon die Fremdschlüssel existieren.

Wenn die Fremdschlüsseigenschaften nicht existieren, müssen die einzelnen Tabellen mit SQL <sup>4</sup> hergeholt werden. Beispiel:

```
SELECT * FROM module;
```

Werden die Informationen bezüglich der Tabelle "Module" angezeigt, dann ist es möglich, die Inhalte wie in einem Tabellenkalkulationsprogramm zu ändern oder Änderungen mit SQL-Befehlen durchzuführen. Das Einpflegen neuer Module ist damit kein Problem.

In einem zweiten Formular gibt es die Möglichkeit, eine komplette Übersicht über die Studiengänge mit den kombinierbaren Informatik-Modulen einzusehen. Tabellarisch untereinander werden alle Module aufgeführt, die das Department Informatik anbietet. In den Spalten werden die Studienfächer aufgelistet, welche anderen Studiengänge Module des Departments Informatik besuchen müssen bzw. besuchen können. „Besuchen können“ ist so zu verstehen, dass es für verschiedene Studiengängen unterschiedliche Möglichkeiten gibt, Lehrveranstaltungen zu besuchen. Dies sind wieder die vier Varianten, die oben aufgeführt wurden. Ein direktes Einfügen von neuen Studiengängen oder Modulen ist nicht möglich.

Durch Analyse der Fachprüfungsordnung konnten die Informationen Studienfachsemester und Belegungsreihenfolge ermittelt werden. Diese Informationen wurden im Schnitt aus den Informatik-Modulen (Y-Achse in der Tabelle) und den nicht-Informatik-Studiengängen (X-Achse in der Tabelle) eingetragen. Dargestellt wird es wie folgt: Studiengangsfachsemester/Belegungsmöglichkeitenreihenfolge. Eine Aktualisierung ist bei dieser Anfrage nicht möglich. Diese Informationen müssen in den Basistabellen eingetragen werden. Dies bedeutet, dass die angezeigte Tabelle ein Produkt aus verschiedenen Tabellen der Datenbank ist. Um Veränderungen vorzunehmen, müssen diese in den Tabellen in der Datenbank geschehen.

## 3.2 Bewertung

Der vorhandene Ansatz löst die Problemstellung nur zum Teil. Sie gibt dem Exportbeauftragten die Übersicht, was ein Studierender mit dem Studiengang X in der Informatik besuchen kann oder muss. Durch die vorgefertigte Maske ist es leicht, die Exportinformationen für einen Studiengang zu erhalten. Die Pflege der Datensätze ist nicht automatisiert. Alle Daten müssen manuell aus den Fachprüfungsordnungen extrahiert und eingepflegt werden. Problematisch sind dabei die Aktualisierungen, weil Veränderungen nicht automatisiert eingepflegt werden. Für diese Auskünfte muss man die ganzen Fachprüfungsordnungen der Universität nach Änderungen durchsuchen.

---

<sup>4</sup>Structured Query Language

Wie die Verteilung der Studierenden auf die Lehrveranstaltungen erfolgt, ist aus dieser Access-Datenbank nicht ersichtlich. Des Weiteren ist nicht ersichtlich, wie viele Studierende eine Lehrveranstaltung besuchen. Durch die Betrachtung einer Zeile kann man aufsummieren, wie viele Studiengänge ein Modul besuchen. Daraus kann man schließen, wie groß der Raum, der für die aus den Modulen resultierenden Lehrveranstaltungen sein muss. Eine genaue Anzahl kann daraus nicht gefolgert werden, weil die Datenbank keine Daten liefert, wie viele Studierende in einem Studiengang sich in einem Fachsemester befinden. Auch eine abstrakte Variante, wie viele Studierende ein Studiengang in einem Fachsemester hat, geht aus den Datensätzen nicht her vor.

Die Definition von Modulen wird bei der Access-Datenbank nicht eingehalten. Es werden Lehrveranstaltungen als Module ausgegeben bzw. Module werden mit Lehrveranstaltungen gleichgesetzt. Module können nach Definition der Bachelor-Studiengänge mehrere Lehrveranstaltungen in sich vereinen.

## 3.3 Verbesserungspotential

Ein Großteil der Informationen, die in der Access-Datenbank vorhanden sind, werden immer gebraucht. So bleibt immer die Suche in den Fachprüfungsordnungen nach Veränderung. Veränderungen können sein, dass ein Studiengang mehr aus der Informatik hören möchte oder auch nur ein anderes Modul für seinen Vorlesungskanon haben möchte. Eine Vereinfachung ist auch nicht in Sicht, weil die Fachprüfungsordnung nur als PDF und in gedruckter Form heraus gegeben werden. Eine andere Informationsquelle für diese Daten liegt nicht vor.

Eine wichtige Verbesserung wäre die Genauigkeit der Vorhersage, wie viele Studierende in einem Semester eine Veranstaltung eines Moduls besuchen werden. Um diese Berechnungen durchzuführen, werden Studierendenzahlen von vorhergegangenen Semestern benötigt, aus denen durch Analysemethoden Studierendenzahlen für die anstehenden Semester errechnet werden können. Für die Analyse werden die Studierendenzahlen für Lehrveranstaltungen benötigt, die nicht selbst angefertigt werden müssen, da diese Informationen in unterschiedlichen Systemen vorliegen. Dadurch, dass diese Daten schon in elektronischer Form vorliegen, können die Daten auch automatisch zu dem Analysesystem transferiert werden, so dass potentielle Fehlermöglichkeiten, die durch manuelles Handeln (Human Factor) bedingt sind, verringert werden.

Eine Anpassung an die neuen Bachelor- und Masterstudiengänge sollte durchgeführt werden. Die Anpassung der Module ist notwendig, und die Module sollten als Module verstanden werden.

Eine Übersicht der Auswahlmöglichkeiten von Modulen eines Studiengangs sollte verfügbar sein, ohne die gesamte Tabelle durchsuchen zu müssen. Diese Module könnte man als Modulgruppen zusammenfassen.

Die aktuellste Version der Fachprüfungsordnungen liegt auf der Webseite des Prüfungsamtes. Um einen aktuellen und direkten Zugriff auf die Daten zu haben, sollten die Links auf die Webseite zeigen statt auf die abgespeicherten FPOs auf dem Access-Datenbank-Rechner. Wenn eine FPO verändert wird, wird das durch einen HTML-Fehler beim Aufruf mitgeteilt, weil die Dateinamen der Fachprüfungsordnungen eine Versionsnummer haben.



## 4 Datenquellen

Genauere Informationen über Studierendenzahlen, die eine Lehrveranstaltung besuchen, sind nicht einfach zu erhalten. Es gibt zwar ein paar Informationsquellen, die aber jeweils nur eine Teilmenge der Studierenden auflisten, die eine Lehrveranstaltung besuchen. Dabei ist zu beachten, dass dem Exportbeauftragten keine dieser Datenquellen direkt zur Verfügung steht. Daher ist er auf externe Datenlieferanten angewiesen. Bezüglich datenschutzrechtlichen Gründen sei angemerkt, dass die Datenquellen nur anonymisierte Zahlen an das Analysesystem versenden und keine einzelnen Studierendennamen oder Matrikelnummern. Um datenschutzrechtliche Probleme zu vermeiden, wurde der ganze Datentransfer mit dem Datenschutzbeauftragten der Universität Erlangen-Nürnberg diskutiert, der aber keine Bedenken in Bezug auf dieses Projekt hat.

Nachfolgend werden die einzelnen Datenquellen aufgelistet:

### **a) Datenbank der Studentenkazlei**

Die Studentenkazlei ist die zentrale Verwaltungsstelle der Studierendendaten. Sie verwaltet nicht nur die Anschriftendaten der Studierenden und vergibt die Benutzerkennungen, sondern listet auch noch die Zahlen der Studierenden in den einzelnen Studiengängen auf. Die Studierendenzahlen eines Studiengangs werden sogar in Fachsemester unterschieden. Durch zusätzliche Informationen, die sich in der Fachprüfungsordnung finden lassen, ist ersichtlich, wie viele Studierende eine Veranstaltung besuchen müssten. Die Fachprüfungsordnung beschreiben, welches Modul in welchem Semester gehört werden muss bzw. kann. Manchmal ist die Modul-Semester-Zuordnung ausgelagert.

### **b) Lehrveranstaltungsevaluationsprogramm**

Gegen Ende jedes Semesters werden die Lehrveranstaltungen der Technischen Fakultät durch die Studierenden evaluiert. Dafür werden vom Lehrpersonal Transaktionsnummern [EN09b] (TANs) an die Studierenden ausgegeben. Mit diesen Transaktionsnummern können die Studierenden die Bewertungsunterlagen online ausfüllen. Wenn der

Bewertungsbogen ausgefüllt ist, wird dies registriert. Durch die Anzahl der Registrierungen bekommt man - jedenfalls theoretisch - die Anzahl der Studierenden, die eine Veranstaltung besucht haben.

### **c) StudOn vom Institut für Lerninnovation**

StudOn[EN09g] ist eine Onlineplattform, die als multimediale Lernplattform dienen soll. Die Studierenden bekommen, wenn die entsprechenden Lehrstühle StudOn nutzen, dort ihre Vorlesungsskripte. Die Benutzer von StudOn müssen sich persönlich mittels Matrikelnummer und Benutzerkennung bei StudOn anmelden. Um an die bereitgestellten Informationen und Materialien heranzukommen, müssen die Studierenden im System der Veranstaltung beitreten. Somit bekommt man die Zahl der Studierenden, die sich für diese Vorlesung die Unterlagen besorgen.

### **d) Campus IT die Prüfungsanmeldung**

Durch die Entwicklung eines Online-Prüfungsanmeldesystem[EN09f] bekommt man die Studierendenzahlen, die an eine Leistungsnachweis in einer Teilleistung eines Moduls teilgenommen haben. Zusätzlich erhält man Informationen über die Anzahl der Studierenden die Teilleistung bestanden haben, einen Fehlversuch hatten oder die Teilleistung endgültig nicht bestanden haben.

## **4.1 Bewertung der Datenquellen**

Es stehen vier völlig verschiedene und unabhängige Datenquellen zur Auswahl, die auch unterschiedliche Informationen bereitstellen. Des Weiteren sind Studierendenzahlen von bestimmten Datenquellen aussagekräftiger als andere. Aus diesem Grund werde ich uns die Quellen unter dem Aspekt Informationsgehalt und Aussagekraft noch genauer besprechen.

Um die Datenquellen in Bezug auf deren Datenqualität zu vergleichen, wird in dem Buch "Data Warehouse Systeme"beschriebene Taxonomie von Datenqualitätsmerkmalen herangezogen. Abbildung 4.1

Hierbei wird jede Eigenschaft außer Schlüsseleindeutigkeit und referenzieller Integrität diskutiert. Diese Eigenschaften werden von den Datenlieferanten anders vergeben, als sie für meine Betrachtung nötig sind. Auch die Relevanz an Daten wird hier nicht berücksichtigt, weil nur relevante Daten bzw. Datenquellen betrachtet werden.

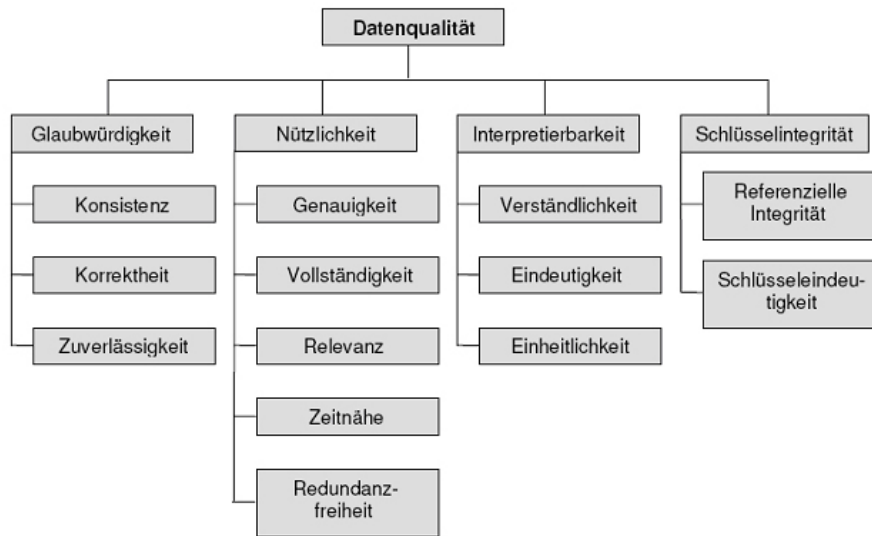


Abbildung 4.1: Taxonomie von Datenqualitätsmerkmalen

Alle vier Datenquellen erfüllen das Kriterium der Interpretierbarkeit. Deswegen wird in den einzelnen Unterpunkten nicht mehr drauf eingegangen.

#### a) Datenbank der Studentenkanzlei (Stud-Kan)

Die Datenbank der Studentenkanzlei bietet uns die vollständigen und genauen Informationen, wie viele Studierende sich gerade in einem Fachsemester eines speziellen Studiengangs befinden. Diese Daten liegen auf der Webseite der Friedrich-Alexander-Universität Erlangen-Nürnberg. Aktuell sind die Daten nicht, weil die Datenblätter zu den Studierendenanzahlen nur jedes Wintersemester angefertigt werden. Genauere Daten würde man erhalten, wenn die Studentenkanzlei öfter die Daten den Exportbeauftragten zuschicken würde. Da dies manuell passiert, würde das einen zusätzlichen Aufwand mit sich bringen. Neben dem Vorteil, dass die Studentenkanzlei die Studierendenanzahlen korrekt, konsistent und zuverlässig ermitteln kann, hat diese Datenquelle auch einen Nachteil. Sie sagt nichts darüber aus, wie viele Studierende wirklich eine Lehrveranstaltung besuchen. Mittels der Zusatzinformation aus dem Studienführer oder Fachprüfungsordnung eines jeweiligen Studiengangs ist nachschlagbar, in welchem Semester ein Studierender eine Veranstaltung besuchen sollte. Im Studienführer steht diese Information allerdings als Vorschlag, beispielsweise soll im Studiengang Informatik, die Lehrveranstaltung "Konzeptionelle Modellierung" im zweiten Semester gehört werden. Ein weiterer Schwachpunkt ist die fehlende Information, ob die Stu-

dierenden die Lehrveranstaltung besuchen oder sich den Lerninhalt zu Hause selbst beibringen.

Jeder Studierende ist nur einmal im System vorhanden und somit besteht kein Problem bezüglich der Redundanz.

### **b) Lehrveranstaltungsevaluationsprogramm (EVA-Sys)**

Das Lehrveranstaltungsevaluationsprogramm ist hervorragend geeignet, um Studierendenzahlen zu analysieren, weil jeder Studierende einer Vorlesung eine Transaktionsnummer bekommt und an der Evaluation mitmachen kann. So ist es angedacht. Basierend auf der Anzahl der Stimmen bei der Evaluation der letzten Jahre können Hochrechnungen für die kommenden Jahre anfertigt werden. Leider gibt eine Diskrepanz zwischen dem geplanten Vorgehen und der Wirklichkeit. Es nimmt in Veranstaltungen nicht jeder Studierende eine Transaktionsnummer mit und somit ist der Datenbestand unvollständig. Des Weiteren füllen nicht alle Studierende, die eine Transaktionsnummer haben, auch das Onlineformular aus. Damit sind die Daten unvollständig. Auch besteht die Möglichkeit, dass Studierende zwei Transaktionsnummern mitnehmen und somit die abgegebene Formularanzahl verfälschen, womit die Korrektheit der Daten nicht mehr gewährleistet ist. Mit dieser Inkorrektheit geht auch eine Unzuverlässigkeit in Bezug auf die Konsistenz und Genauigkeit der Daten von Jahr zu Jahr einher. Die Informationen könnten auch manuell ermittelt und immer zu festen Zeiten an den Export beauftragten übermittelt werden. Die Evaluation der Lehrveranstaltungen sollte immer am Ende der Vorlesungszeit stattfinden. Damit würde sich zwei Mal im Jahr die Notwendigkeit der Datenübermittlung ergeben.

### **c) StudOn vom Institut für Lerninnovation**

StudOn bietet die genaue Anzahl der Studierenden, die eine Vorlesung besuchen, weil jeder, der die Prüfung ablegen will, die Kursunterlagen braucht, die er nur von StudOn bekommen kann. Dabei muss unterschieden werden, wer die Vorlesung besucht und wer die Prüfung ablegen möchte. Der anzunehmende Fall ist hier, dass fast alle Studierende, die eine Lehrveranstaltung besuchen, auch einen Leistungsnachweis erhalten wollen. Eine weitere Unschärfe ist die, dass die Unterlagen möglicherweise von einem befreundeten Kommilitonen besorgt werden und somit nicht alle, die offiziell die Veranstaltung besuchen, auch registriert sind. Das größere Problem an StudOn ist aber, dass sich nicht alle Lehrstühle an StudOn beteiligen. Der größere Teil der Lehrstühle stellt die Informationen für ihre Veranstaltungen noch auf ihrer eigenen Webseite bereit. Somit sind die Daten dieser Datenquelle weder korrekt noch genau oder vollständig und zuverlässig. Der Datentransfer würde per Datenblatterfolgen, die

dann zu einem festen Zeitpunkt oder nach gefragten Zeitpunkt übermittelt werden. Im StudOn-System muss man sich mit seiner Benutzerkennung anmelden und deswegen ist die Anzahl der Anmeldungen einer Lehrveranstaltung redundanzfrei.

#### d) Campus IT die Prüfungsanmeldung (CIT)

Die Mein Campus-Software ersetzt für die Bachelor-Studierende das Prüfungsamt mit der Anmeldung sowie die Notenverbuchung und deckt damit vollständig die Studierenden der Bachelor- und Master-Studiengänge ab. Jeder Studierende taucht nur einmal im System auf und somit sind die Daten redundanzfrei. Aus diesen Gründen ist Mein Campus eine zuverlässige, korrekte und konsistente Datenquelle bezüglich der Information, wie viele Studierende einen wie auch immer gearteten Leistungsnachweis wollen. Des Weiteren stellt Mein Campus die Information bereit, welcher Anteil der Studierenden eine Prüfung bestanden, nicht bestanden und endgültig nicht bestanden hat. Der Unterschied zwischen nicht bestanden und endgültig nicht bestanden ist, dass bei nicht bestanden ein weitere Versuch möglich ist, den Leistungsnachweis zu erwerben. Dies bedeutet, dass die Studierenden, die nicht bestanden haben, noch einmal die Veranstaltung besuchen können. Mein Campus identifiziert aber nicht die Studierenden, die eine Vorlesung besuchen, aber keinen Leistungsnachweis haben wollen. Das Campus IT-Team[EN09a], die Entwickler von Mein Campus, stellen eine Schnittstelle zur Verfügung. Somit ist die Möglichkeit gegeben, dass man immer wieder auf die Daten zugreifen kann.

Übersicht der betrachteten Datenqualitätsmerkmale: 4.1

	Stud-Kan	EVA-Sys	StudOn	CIT
Korrektheit	JA	NEIN	NEIN	JA
Konsistenz	JA	NEIN	NEIN	JA
Zuverlässigkeit	JA	NEIN	NEIN	JA
Vollständigkeit	JA	NEIN	NEIN	JA
Genauigkeit	JA	NEIN	NEIN	JA
Zeitnahe	jährlich	semesterlich	JA	JA
Redundanzfrei	JA	NEIN	JA	JA

Tabelle 4.1: Datenquellenvergleich

### 4.2 Verwendete Datenquellen

Jede der Datenquellen ist anders aufgebaut und hat mehr oder weniger Relevanz für die Ermittlung der Studierendenzahlen. Für die Vorausberechnung der Studierendenzahlen ist Mein Campus durch die Aufteilung in "bestanden", "nicht bestanden" und "endlich nicht bestanden" aussagekräftig. Es sind nicht nur die Prüfungsanmeldezahlen von Studierenden der letzten Jahre vorhanden, sondern es liegen auch Datensätze vor, wie viele aus dem letzten Semester die Lehrveranstaltung ein weiteres Mal besuchen können. Die Problematik, dass nicht jeder Studierende einen Leistungsnachweis haben will, ist mit den Bachelor-Studiengängen marginal geworden, weil jedes Fach, durch einen benoteten oder unbenoteten Leistungsnachweis, geprüft wird. Ein weiterer positiver Aspekt bei der Verwendung von Mein Campus ist, dass eine kostenlose Internetschnittstelle angeboten wird. Dadurch sind keine Mitarbeiter auf Seiten des Campus IT-Teams mehr involviert. Wegen dieser Vorteile wird Mein Campus als Datenquelle ausgewählt.

Die Datenbank der Studentenkazlei bietet wie mein Mein Campus zuverlässige Daten mit dem Unterschied, dass die Daten nicht auf eine Lehrveranstaltung begrenzt sind, sondern sich auf einen ganzen Studiengang beziehen. In den Studienführern der einzelnen Studiengänge steht, welche Fächer in welchem Semester besucht werden müssen. Dieser Zusatzaufwand der Sichtung der Studienführer fällt also trotzdem an, unabhängig davon, ob die Datenquelle verwendet wird oder nicht. Allerdings sind diese Informationen sowieso schon vorhanden, da der Exportbeauftragte ggf. Studierende beraten muss, welche Module oder Lehrveranstaltungen diese anhören können. Dass eine Anzahl der Studierenden nicht zu den Vorlesungen kommt und sich den Stoff zu Hause alleine beibringt, ist zunächst nicht interessant, weil grundsätzlich für die Hörsaalreservierung jeder Studierende mit einberechnet werden muss, ob er nun die Lehrveranstaltung besucht oder ihr fernbleibt. Diese Aussage wird später noch einmal aufgegriffen.

Nach dem Gespräch mit dem Verantwortlichen von Campus IT wurde dem Exportbeauftragten die Möglichkeit gegeben, diese Daten auch über die kostenlose Internetschnittstelle zu erhalten. Die Server von Mein Campus beinhalten auch immer die aktuellen Daten der Studierendenzahlen für einen Studiengang, aufgeschlüsselt nach Fachsemestern. Dadurch wird für den Datentransport kein zusätzlicher Aufwand von der Studentenkazlei benötigt.

Durch die Automatisierung und jeder Zeit verfügbaren Datenübermittlung und durch die genaueren Informationen, werden diese Datenquelle verwendet.

Das Lehrveranstaltungsevaluationsprogramm hat den entscheidenden Vorteil, dass hierüber ein Zugang zu der tatsächlichen Anzahl an Studierenden zu bekommen ist, die eine Lehrveranstaltung besuchen. Nur ist dies leider nicht die Wirklichkeit, da nicht

jeder Studierende eine Transaktionsnummer aus einer Veranstaltung für das System mitnimmt und weiterhin nicht jeder Studierende, der eine Transaktionsnummer mitnimmt, die Lehrveranstaltung evaluiert. Daher spiegelt die Anzahl der ausgehenden Transaktionsnummern nicht die wirkliche Anzahl der Besucher einer Veranstaltung wider. Des Weiteren gibt es eine feste Zahl, wie viele die Transaktionsnummer mitnehmen oder nicht. Diese prozentuale Transaktionsnummermitnehmerzahl variiert von Semester und Lehrveranstaltung, und somit ist die Anzahl nicht berechenbar. Da die Informationen, die geliefert werden, auch nur händisch eingetragen können, ist diese Datenquelle nicht geeignet.

Das StudOn-Portal vom Institut für Lerninnovation bietet eine fast genaue Anzahl an Studierenden, die eine Veranstaltung besuchen. Leider benutzen, wenn man sich nur das Departement Informatik betrachtet, noch zu wenige Lehrstühle das Angebot von StudOn. Dadurch würden Informationen nur für wenige Lehrveranstaltungen vorliegen. Wenn mehrere Lehrstühle das StudOn-Portal verwenden würden, wäre es eine Möglichkeit, diese Informationen in der Zukunft miteinzubringen. Zudem gibt es das Problem, dass die Informationen von StudOn über die Studierendenanzahlen nur per Datenblatt erhalten werden können und somit noch eine manuelle Eingabe vonnöten ist. Aus diesen Gründen sollte StudOn für die Zukunft im Hinterkopf behalten werden, aber für die ersten Berechnungen wird StudOn durch die geringe Datenanzahl außer Acht gelassen.

## 4.3 Datenquelle Fachprüfungsordnungen

Eine feststehende Datenquelle ist die Webseite des Prüfungsamtes. Auf der Webseite stehen die Fachprüfungsordnungen aller Fakultäten und gegebenenfalls auch die Studienführer für die Studiengänge, in denen die Module nur spartanisch aufgeführt sind. Dadurch, dass es sich um Gesetzestexte handelt, erfüllt der Text die Bedingungen der Konsistenz, Korrektheit und der Zuverlässigkeit. Die Fachprüfungsordnung beschreibt genau und vollständig die Anforderungen an die Studierenden, wenn sie ein Fach studieren wollen. Änderungen müssen sofort immer bekannt gemacht werden, damit die Studierenden informiert werden können. Die Texte sind in juristischer Fachsprache verfasst und somit nicht ganz leicht verständlich sowie schwierig zu erlernen. Der große Vorteil der juristischen Sprache ist es hingegen, dass die Texte eindeutig sind. Wegen dieser Kriterien gehören die Fachprüfungsordnungen auch zu dem Datenbestand, auch wenn sie leider nicht automatisierbar sind.





# 5 Konzeptioneller Entwurf

Durch die Problemdefinition aus Kapitel 1.1 ist bekannt, welche zwei Hauptziele zu realisieren sind. Das erste ist die Berechnung der Studierendenzahlen einer Lehrveranstaltung in der Informatik und das zweite ist der Informationsservice des Exportbeauftragten für die Studierenden anderer Studiengänge, die eine Lehrveranstaltung der Informatik hören möchten oder müssen. Diese Informationen liegen nicht explizit vor, sondern müssen errechnet werden. Da die Studierenden von unterschiedlichen Departements kommen und unterschiedliche Studiengänge besuchen, die verschiedene Prüfungskriterien haben, muss die Umwelt der Studierenden abgebildet werden, um die oben erwähnten Ziele zu erreichen. Aus diesem Grund muss untersucht werden, welche Fakten für die zwei Hauptziele zusätzlich relevant sind und in welcher Abhängigkeit sie mit den Hauptzielen stehen.

## 5.1 Anforderungsanalyse

Zur Bestimmung der Studierendenzahlen in Lehrveranstaltungen werden weitere Informationen benötigt. Es muss einkalkuliert werden, welches Semester für die Analyse betrachtet wird. Von Jahr zu Jahr bzw. von Semester zu Semester ändern sich die Studierendenzahlen in den Lehrveranstaltungen. Die Analysen sind daher immer von Semestern abhängig, somit muss der zeitliche Zusammenhang stimmen.

Semester ist definiert durch eine Jahresangabe und mit einem xS Kürzel. Das xS-Kürzel betitelt die Einteilung in Sommersemester (SS) bzw. Wintersemester (WS).

Die Studierenden müssen auch noch abgebildet werden. Von den Datenquellen, die zur Verfügung stehen, werden beispielsweise die Jahrgangsinformationen mitgeteilt. Ein Jahrgang beschreibt die Anzahl von Studierenden in einem Semester eines Studiengangs. Der Jahrgang bildet eine solide Grundlage für die Berechnungen, die durchzuführen sind. Veränderungen in Jahrgangsinformationen haben keine Abhängigkeiten, da sie keine direkte Abhängigkeit von der Zeit oder vom Studiengang haben. Es könnte nur neue Einträge für einen Jahrgang geben, die in die Datenbank einfach einzupflegen sind.

Die Studiengänge spielen eine wichtige Rolle, weil unterschiedliche Studiengänge eine unterschiedliche Anzahl von Lehrveranstaltungen besuchen. Außerdem können sich in einer Vorlesung mehr Studiengänge befinden. Diese Information ist entscheidend für die spätere Analyse. Zugleich wird auch die Information, wie viele Studierende sich in einem Jahrgang befinden, von einer Datenquelle geliefert.

Unterschieden wird auch zwischen den alten Studiengängen wie Magister und Diplom sowie den neuen Bachelor- und Master-Studiengängen. In den alten Studiengängen gab es eine Lehrblocksveranstaltung, die mindestens eine Lehrveranstaltung beinhaltet hat, aber auch mehrere Lehrveranstaltungen beinhalten konnte, wie z. B. eine Vorlesung und eine Übung. Bei den Bachelor- und Master-Studiengängen werden stattdessen Module angeboten. Module fassen eine Menge von Teilleistungen zu einem bestimmten Themengebiet zusammen.

Sowohl Lehrveranstaltungsblock als auch Module werden durch einen dem Lehrinhalt nahen Namen definiert. Weiterhin besitzen beide ein Punktesystem. Die Module besitzen die European Credit Transfer System-Punkte (ECTS), die den Arbeitsaufwand diese Module beschreiben. Für den gleichen Zweck besitzen die Lehrveranstaltungsblöcke Leistungspunkte. Auch können beide im Sommer- oder Wintersemester angeboten werden. Wichtig für die Berechnung ist die Information, ob die Module eine GOP (Grundlagen- und Orientierungsprüfung) erfordern. Diese Module können nämlich nur einmal wiederholt werden.

Teilleistungen, die in Module unterteilt sind, sind nicht als Lehrveranstaltungen zu sehen. Teilleistungen sind Leistungsnachweise über eine oder mehrer Lehrveranstaltungen. Teilleistungen können Scheine sein, die für Übungen oder Praktika ausgestellt werden, oder ein Leistungsnachweis nach einer Vorlesung.

Eine Lehrveranstaltung unterteilt sich in drei unterschiedliche Typen, die die Universität im Angebot hat. Es gibt Vorlesungen, Übungen und Praktika.

Wenn man bedenkt, dass die Studierenden auch Lehrveranstaltungsblöcke im Sinne der alten Studiengänge bzw. Modulgruppen im Sinne der neuen Studiengänge wählen können, müssen diese auch betrachten. Besonders im Hauptstudium oder in der Masterphase wählen Studierende aus verschiedenen Bereichen Lehrveranstaltungen, um ihr Studium individuell zu gestalten. Somit müssen die Modulgruppen als Teilmengen, aus denen sich die Studierenden für eine Lehrveranstaltung entscheiden können, betrachtet werden.

Es gibt viele verschiedene Möglichkeiten, Lehrveranstaltungen zu besuchen. Oben wurde schon thematisiert, dass die Vielfältigkeit eines Studiengangs Probleme herbeiführen kann. Die Information, wie und welche Möglichkeiten es gibt, etwas zu belegen, liegt in der Fachprüfungsordnung des einzelnen Studiengangs. Sie gibt Auskunft über die Lehrveranstaltungsblöcke bzw. Module, die abgelegt werden müssen und stellt zusätzlich

Informationen bereit, welche Wahlmöglichkeiten es gibt. Die Fachprüfungsordnungen liegen immer beim Prüfungsamt auf der Webseite. Somit sollte man die Information der Homepage-Adresse aufnehmen, um auf die aktuellen Prüfungsordnungen zurückgreifen zu können.

Die Fachprüfungsordnung beschreibt einen Studiengang. Hierbei ist zu beachten, dass es immer mehrere Fachprüfungsordnungen zu einem Studiengang geben kann. Es können in einem Studienfach unterschiedliche Fachprüfungsordnungen aktuell sein. Ziehe man hier als Beispiel nur die Informatik heran: hier hat sowohl die Bachelor-Fachprüfungsordnung Gültigkeit als auch die Diplom-Fachprüfungsordnung. Somit muss man darauf achten, in welchem Semester ein Studierender mit dem Studium angefangen hat. Weiterhin gibt es Studienfächer, die im Sommer- und im Wintersemester anfangen, und dabei jeweils mit einem anderen Modulkanon arbeiten, so dass sie zwei Fachprüfungsordnungen haben. Fachprüfungsordnungen werden auch immer wieder aktualisiert, weswegen es nötig ist, den Zusammenhang zwischen den alten und neuen Fachprüfungsordnungen zu dokumentieren.

Ein weiterer wichtiger Punkt ist die Betrachtung des Studiengangs an sich bzw. an welchem Department ein Studiengang untergebracht ist. Hieraus wird ersichtlich, welcher Studiengang, aus dem ein Modul bzw. eine Lehrveranstaltung angeboten wird, nicht zu dem Department gehört. Wenn dies der Fall ist, werden die Exportleistungen eines Departments angezeigt.

Zusätzlich ist die Information wichtig, an welcher Fakultät ein Department angeordnet ist, um daraus auf die Studiengänge zu schließen. Hieraus ergibt sich die Information, ob Studierende für den Besuch von Lehrveranstaltungen pendeln müssen. Auch weiterhin dient dieser Einstieg als Navigationshilfe für Suchanfragen.

Zeitliche Veränderungen wird es immer wieder geben, allein durch veränderte Anforderungen an Berufen oder Weiterentwicklungen in den Spezialgebieten. Somit müssen immer wieder Veränderungen eingepflegt werden. Dafür wird die Hilfe der Verantwortlichen für Teilleistungen, Module und Studiengängen benötigt. Die Verantwortlichen sind meist auch die Dozentinnen und Dozenten an der Universität und gehören einem Lehrstuhl an. Somit werden die Informationen direkt von den Verantwortlichen ins System eingepflegt. Der Veränderungsaufwand wird im USE-Case 5.1 gezeigt. Darauf zu achten ist auch, dass sich jeder einzelne angesprochene Punkt ändern kann; durch Abhängigkeiten untereinander können dann mehrere Komponenten betroffen sein. Als gutes Beispiel dient die Modulgruppe. Ändert sich eine Fachprüfungsordnung in Bezug auf eine Modulgruppe, dann muss der Verantwortliche für einen Studiengang diese Informationen an den Exportbeauftragten herantragen. Als erstes wird der Exportbeauftragte die neue Fachprüfungsordnung aufnehmen und beim Studiengang eintragen. Wenn sich außer dem Namen der Modulgruppe etwas ändert, müssen auch Module angepasst werden, die zur veränderten Modulgruppe gehören. Daraus resultie-

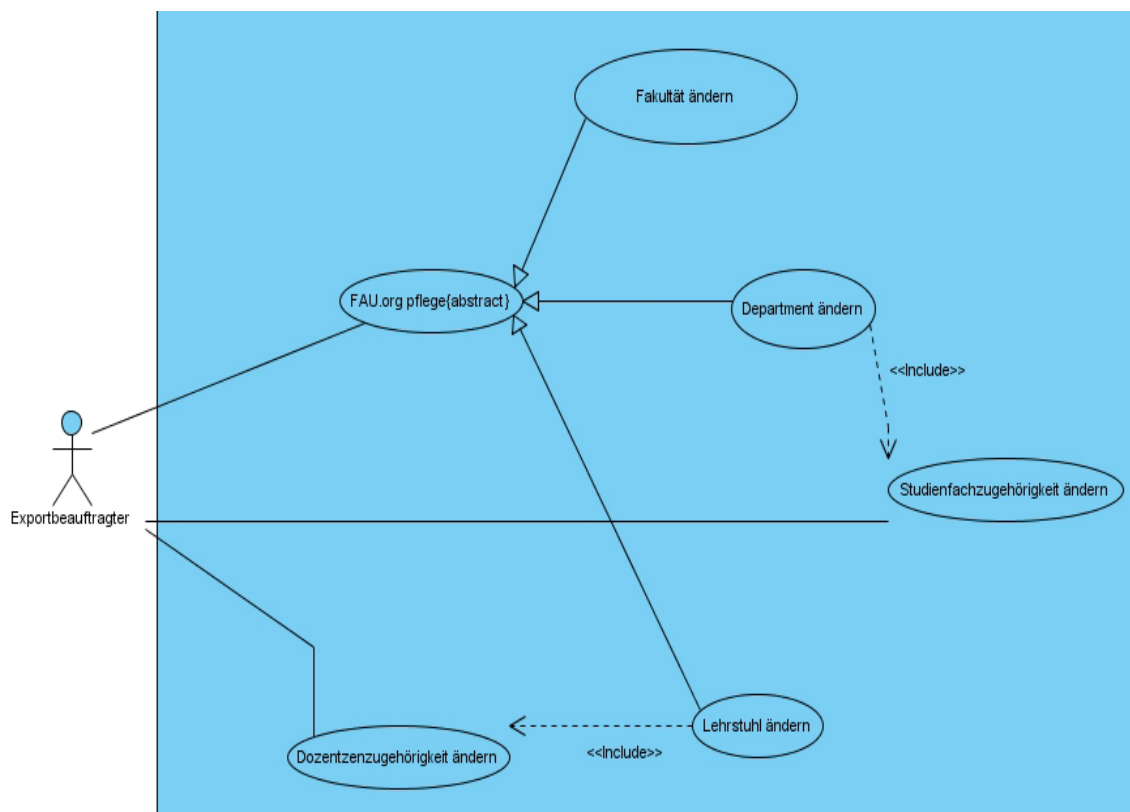


Abbildung 5.1: Pflege der Dozenten mit Verantwortlichkeit

ren Veränderungen bei den Modulen, die Veränderungen bei Teilleistungen mit sich ziehen können. Entweder wird ein neues Modul hinzugefügt oder ein altes modifiziert. Ein Modul besteht aus mindestens einer Teilleistung. Bei neuen Modulen werden neue Teilleistungen eingetragen und bei bestehenden Modulen kommen neue Teilleistungen hinzu oder es werden alte modifiziert. Eine Teilleistung bezieht sich auch wieder auf eine oder mehrere Lehrveranstaltungen; somit müssen neue Lehrveranstaltungen eingetragen bzw. die alten modifiziert werden. Beschrieben wird das in Abbildung 5.2.

Bei diesen ganzen Veränderungsmaßnahmen sollte man bedenken, dass auch die Verantwortlichen für die Module, Teilleistungen oder Lehrveranstaltungen angepasst werden müssen.

Ein ähnlicher Veränderungs- bzw. Einfügeaufwand ist zu leisten, wenn ein neuer Studiengang eingerichtet wird. So müssen neben den Modulgruppen, wie oben beschrieben wurde, die einzelnen Module eingetragen werden, wenn sie noch nicht vorhanden sind. Wenn neue Module eingefügt werden, müssen auch die Teilleistungen für die Module erstellt werden, weil sie immer in Abhängigkeit von den Modulen existieren. Zu beachten ist bei den Teilleistungen, dass es ggf. die Lehrveranstaltungen, auf die sie verweisen, bereits gibt. Wenn dies der Fall ist, muss nur die Verlinkung eingetragen werden. Wenn die Lehrveranstaltung aber nicht vorhanden ist, muss sie noch erstellt werden. Ein neuer Studiengang wird immer einem Department zugeordnet. Diese Informationen müssen zusätzlich eingetragen werden. Es wird auch einen oder mehrere Verantwortliche für einen Studiengang geben, von denen die Informationen benötigt werden.

Wenn sich Teilleistungen oder Lehrveranstaltungen ändern, ist der Aufwand geringer. Bei einer Veränderung muss der Name verändert werden, und wenn es die abhängige Lehrveranstaltung trifft, muss auch die Lehrveranstaltung angepasst werden. Eine Teilleistung kann nicht ohne ein Modul erstellt werden, weil Teilleistung eine Untermenge eines Moduls ist. Wenn eine Lehrveranstaltung verändert wird, müssen ihre Werte und eventuell die Angaben der Dozenten, die für die Lehrveranstaltung verantwortlich sind, verändert werden. Beim Einfügen verhält es sich bei einer Lehrveranstaltung ähnlich. Es müssen die Daten der Lehrveranstaltung eingetragen werden und ggf. der verantwortliche Dozent.

Ein anderer Aspekt ist, dass sich die Universitätsstruktur in Form der Fakultäten der Departments oder der Lehrstühle ändert. Die Fakultäten werden sich nicht so schnell ändern und die Departments auch eher nicht. Es kann trotzdem nicht ausgeschlossen werden. Wenn eine neue Fakultät, Department oder Lehrstuhl eingefügt wird, müssen nur die Verbindungen zur höheren Ebene eingetragen werden. Die oberste Ebene ist die Fakultät die Departments und die Departments bestehen wiederum aus Lehrstühlen. Jede untere Ebene besitzt eine Verbindung zur oberen Ebene. Ein Lehrstuhl gehört zu einem Department und ein Department gehört zu einer Fakultät.

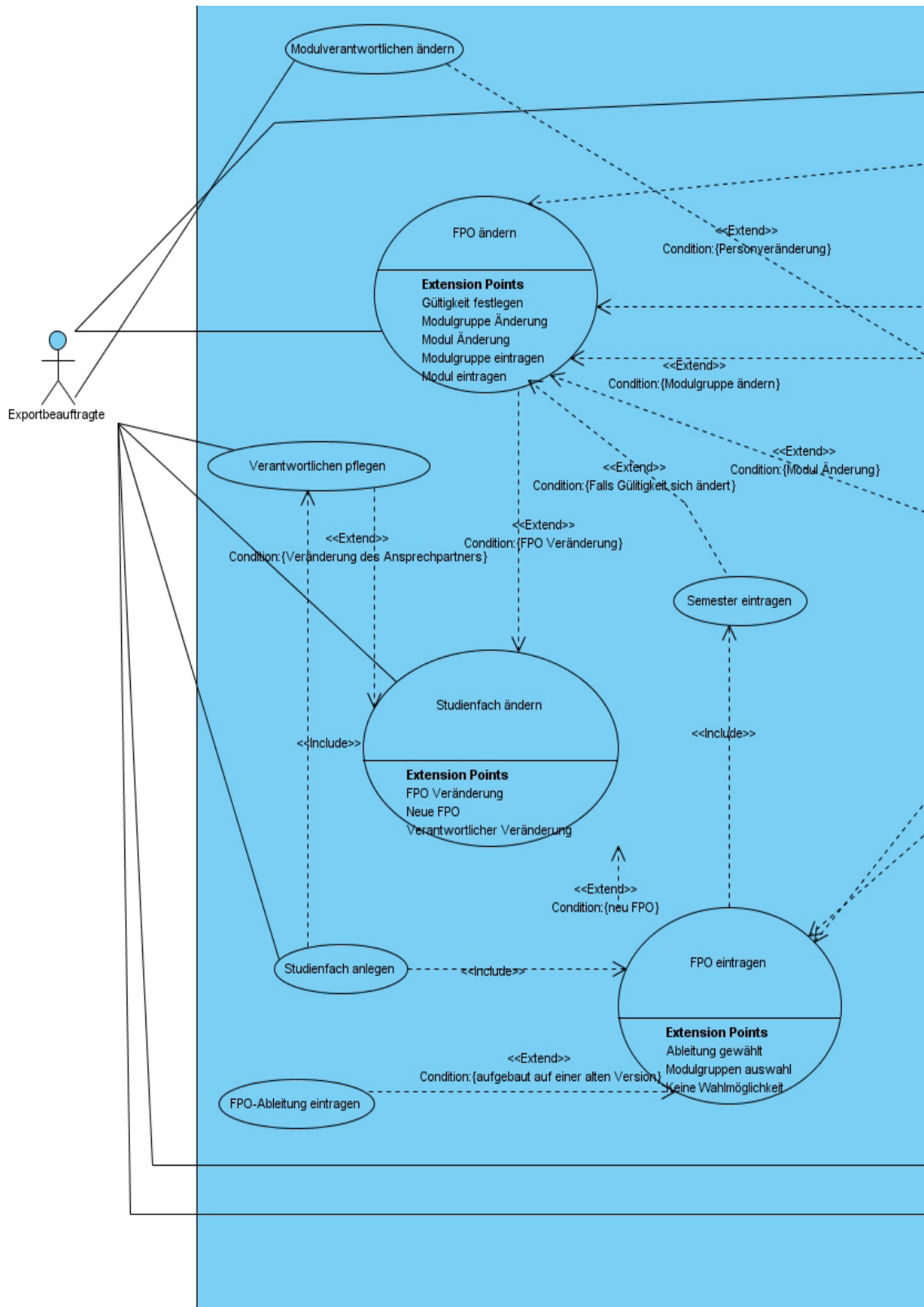


Abbildung 5.2: USE-Case Abhängigkeiten Teil1

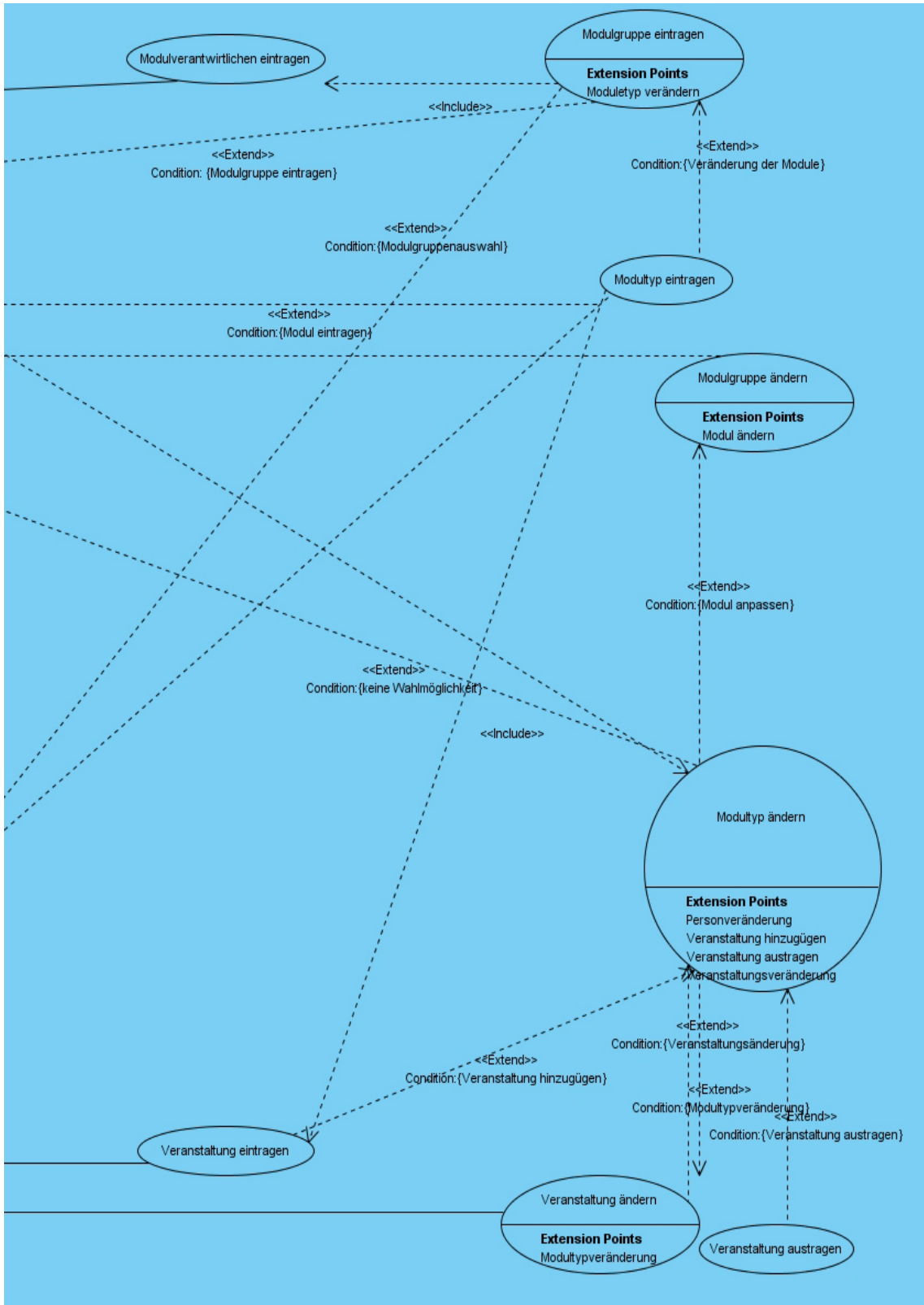


Abbildung 5.3: USE-Case Abhängigkeiten Teil2

Durch Modifikationen von Lehrstühlen oder Arbeitsplatzwechseln von Dozenten innerhalb der Universität muss die Zugehörigkeit der Dozenten an die veränderte Situation angepasst werden. D.h. die neue Telefonnummer und Raumnummer sind zu berücksichtigen, um die Person zu erreichen.

## 5.2 Modellierung eines idealen Schemas

Betrachtet man den Text der Anforderungsanalyse genauer, so tauchen spezielle Wörter öfter auf als andere. Diese Wörter sind folgende:

Fakultät	Fachprüfungsordnung
Department	Modulgruppe
Lehrstuhl	Modul
Dozent	Lehrveranstaltungsblock
Jahrgang	Teilleistung
Semester	Lehrveranstaltung
Studiengang	

Diese Wörter ergeben nach der Anforderungsanalyse unsere Entities<sup>1</sup> für das Entity-Relation E / R-Diagramm.

Gehen wir als erstes zu der Navigationshilfe der Universitäts-Struktur der Universität Erlangen-Nürnberg. Auf höchster Ebene haben wir die Fakultät, die durch einen eindeutigen Namen beschrieben wird. Zu einer Fakultät gehören immer mehrere Departements. Als Beispiel soll nur die Technische Fakultät erwähnt sein, der fünf Departements zugeordnet sind.

Ein Department hat einen Namen und zu jedem Department gehört auch immer genau eine Fakultät. Weiterhin besitzt ein Department immer einen oder mehrere Lehrstühle.

Lehrstühle gehören immer zu einem Department. Lehrstühle besitzen einen fachspezifischen Namen. An einem Lehrstuhl arbeiten immer mehrere Mitarbeiter, die meist Dozenten sind. Dozenten haben ein Büro am Lehrstuhl und somit haben sie eine Raumnummer und eine Telefonnummer, unter der sie zu erreichen sind. Ein Mitarbeiter besitzt einen Vornamen und einen Nachnamen. Abbildung 5.4

---

<sup>1</sup>Gegenstände sind wohl unterscheidbare physisch oder gedanklich existierende Konzepte der zu modellierenden Welt. [AK06]



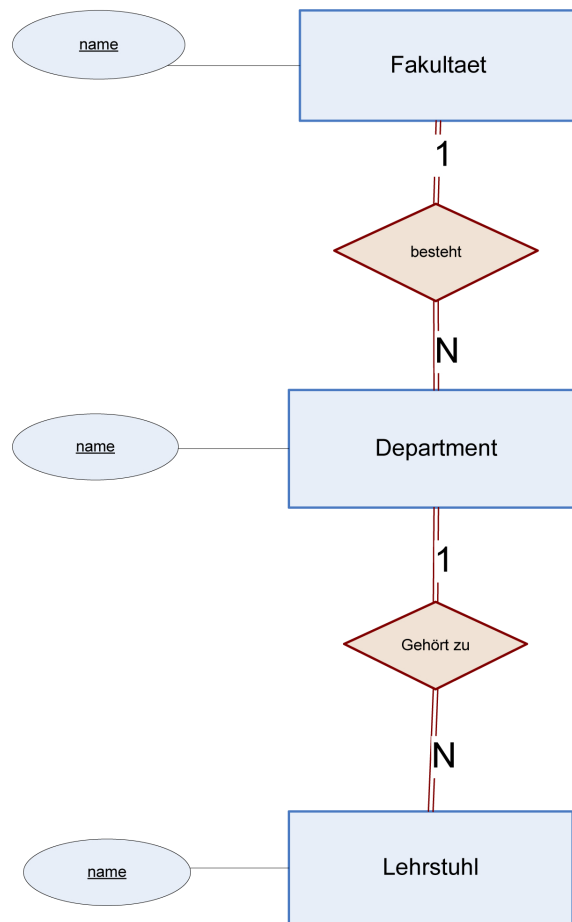


Abbildung 5.4: FAU Struktur in E/R-Notation

An einem Departement können mehrere Studiengänge angesiedelt sein. Studiengänge besitzen einen Namen und ein Kürzel für die Darstellung in Vorlesungsverzeichnissen. Jeder Studiengang gehört immer nur zu einem Department, obwohl manche Studiengänge Module aus mehreren Departments integriert haben. Abbildung 5.5

Zu jedem Studiengang gibt es mindestens eine Fachprüfungsordnung und diese Fachprüfungsordnung ist genau einem Studiengang zugeordnet. Die Fachprüfungsordnung besitzt einen eindeutigen Namen und eine Internet-Adresse vom Prüfungsamt, auf der sie publiziert ist. In ihr wird festgehalten, zu welchem xS man ein Studium nach dieser Prüfungsordnung anfangen kann. Bei Fachprüfungsordnungen wird auch vermerkt, ab welchem Semester sie gültig sind bzw. seit wann sie keine Gültigkeit mehr haben. Neuere Fachprüfungsordnungen können auf älteren Fachprüfungsordnungen basieren.

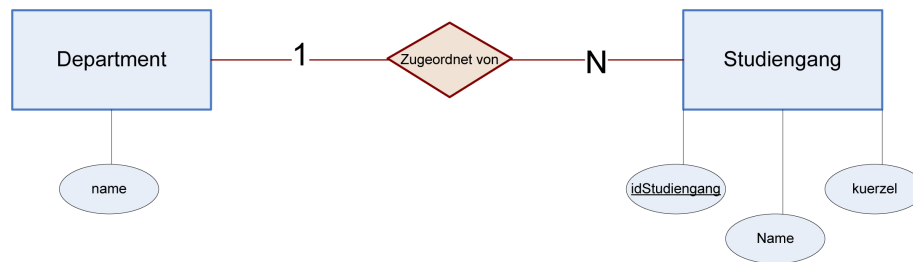


Abbildung 5.5: Department zu Jahrgang in E/R-Notation

In den Fachprüfungsordnungen stehen auch die Module, die zu einer bestimmten Semesterzahl gehört werden können. Weiterhin können auch in einer Fachprüfungsordnung in verschiedenen Semestern Wahlpflichtfächer vorgeschrieben sein, die als Modulgruppe angeordnet sind. Eine Modulgruppe kann zu einer Fachprüfungsordnung gehören und hat einen dem Lehrstoff entsprechenden Namen. Zusätzlich hat sie auch eine gewisse ECTS-Punkteanzahl.

Modulgruppen können aus einem oder mehreren Modulen bestehen. Module können zu mehreren Modulgruppen gehören oder auch direkt in Fachprüfungsordnungen verankert sein. Ein Modul hat neben seinem Namen ein xS-Kürzel, in welchem Semester dieses angeboten wird. Zusätzlich unterscheidet man noch, ob das Modul eine Grundlagen- und Orientierungsprüfung ist oder nicht. Weiterhin hat es einen Wert in Form von ECTS-Punkten.

Lehrveranstaltungsblöcke gehören mindestens zu einer Fachprüfungsordnung. Ähnlich wie bei Modulen für die Bachelor-Studiengänge gilt selbiges für die alten Magister- und Diplom-Studiengänge. Allerdings haben sie hier weniger Eigenschaften. Eine Lehrblockveranstaltung hat einen Namen und Leistungspunkte. Abbildung 5.6

Für einige Studiengänge ist es möglich, Module als Schlüsselqualifikation anerkennen zu lassen. Für jeden Studiengang gibt es viele Module die als Schlüsselqualifikation anerkannt werden.

Zu jedem Modul gibt es mindestens eine Teilleistung. Eine Teilleistung gehört genau zu einem Modul. Eine Teilleistung wird durch sein Modul identifiziert und hat eine eigene ECTS-Anzahl. Ohne ein Modul gibt es keine Teilleistung.

Geprüfte Teilleistungen unterscheiden zwischen Studierendenzahlen, die bestanden haben, die nicht bestanden haben und die endgültig durchgefallen sind.

Aus den Teilleistungen gehen die Lehrveranstaltungen hervor. Jede Teilleistung besteht aus einer Veranstaltung; dies gilt auch für die Lehrveranstaltungsblöcke. Jede Lehrveranstaltung gehört wiederum zu einem Lehrveranstaltungsblock bzw. zu einer

Teilleistung. Eine Lehrveranstaltung kann entweder eine Vorlesung, eine Übung oder eine Teilleistung sein, die jeweils einen Namen trägt.

Jede Lehrveranstaltung wird pro Semester evaluiert. Bei dieser Evaluation nimmt eine bestimmte Anzahl von Studierenden in einem Semester teil.

Für jedes Modul und für jede Teilleistung muss es immer genau einen verantwortlichen Dozenten geben. Ein Dozent kann aber für mehrer Module und/oder Teilleistungen verantwortlich sein. Für einen Studiengang kann es mehrere Verantwortliche geben, die dann eine bestimmte Rolle erfüllen.

Ein Jahrgang wird aus einem Studiengang und einem Semester beschrieben. Einen Studierenden eines Studiengangs kann man mit Hilfe seines Semesters einem Jahrgang zuordnen. Jeder Jahrgang ist auch mehreren Semestern zugeordnet. Ein Jahrgang ist existenziell abhängig von Semester und Jahrgang.

Die Gesamtübersicht über das E/R-Diagramm ist hier zu finden 5.7 5.8.

## 5.3 Vereinfachung des Schemas

Nach der Betrachtung des idealen Schemas sind immer wieder Anomalien im Schema aufgetreten, die wieder zu Veränderungen im E/R-Diagramm geführt haben.

Als eins der ersten Probleme wurde identifiziert, dass es Dozenten gibt, die keinem Lehrstuhl zugeordnet sind. Als Beispiel kann das Department Informatik dienen. Die Professur für Didaktik der Informatik ist keinem Lehrstuhl zugeordnet. Dieses Problem wurde durch die Führung eines imaginären Lehrstuhls gelöst.

Ein großes Problem ist auch die Zuordnung der Verantwortlichen von Studiengängen, Modulen und Teilleistung. Es ist nicht immer möglich, sofort festzustellen, wer zuständig ist, wenn ein Studiengang, Modul oder Teilleistung angelegt wird. Somit wird im Folgenden nicht mehr davon ausgegangen, dass immer einen Ansprechpartner für Studiengänge, Module und Teilleistungen existiert. Weiterhin gibt es Module, die immer neue Veranstalter haben, weil beispielsweise Grundmodule rotationsmäßig von verschiedenen Lehrstühlen angeboten werden. So kann es sein, dass die Verantwortlichen von Teilleistungen oft wechseln, und die Verantwortlichen auch nicht sofort die Information weitergeben.

Weitere Probleme sind bei den Schlüsselkandidaten <sup>2</sup> aufgetaucht. Es kann bei vielen Entities dazu kommen, dass Schlüsselkandidaten nicht eindeutig sind. Besonders bei

---

<sup>2</sup>Eine minimale Menge von Attributen, deren Werte das zugeordnete Entity eindeutig innerhalb aller Entities seines Typs identifizieren nennt man Schlüssel. [AK06] Schlüsselkandidat ist eine Möglichkeit für den Fall, dass es mehrer Schlüssel geben sollte.

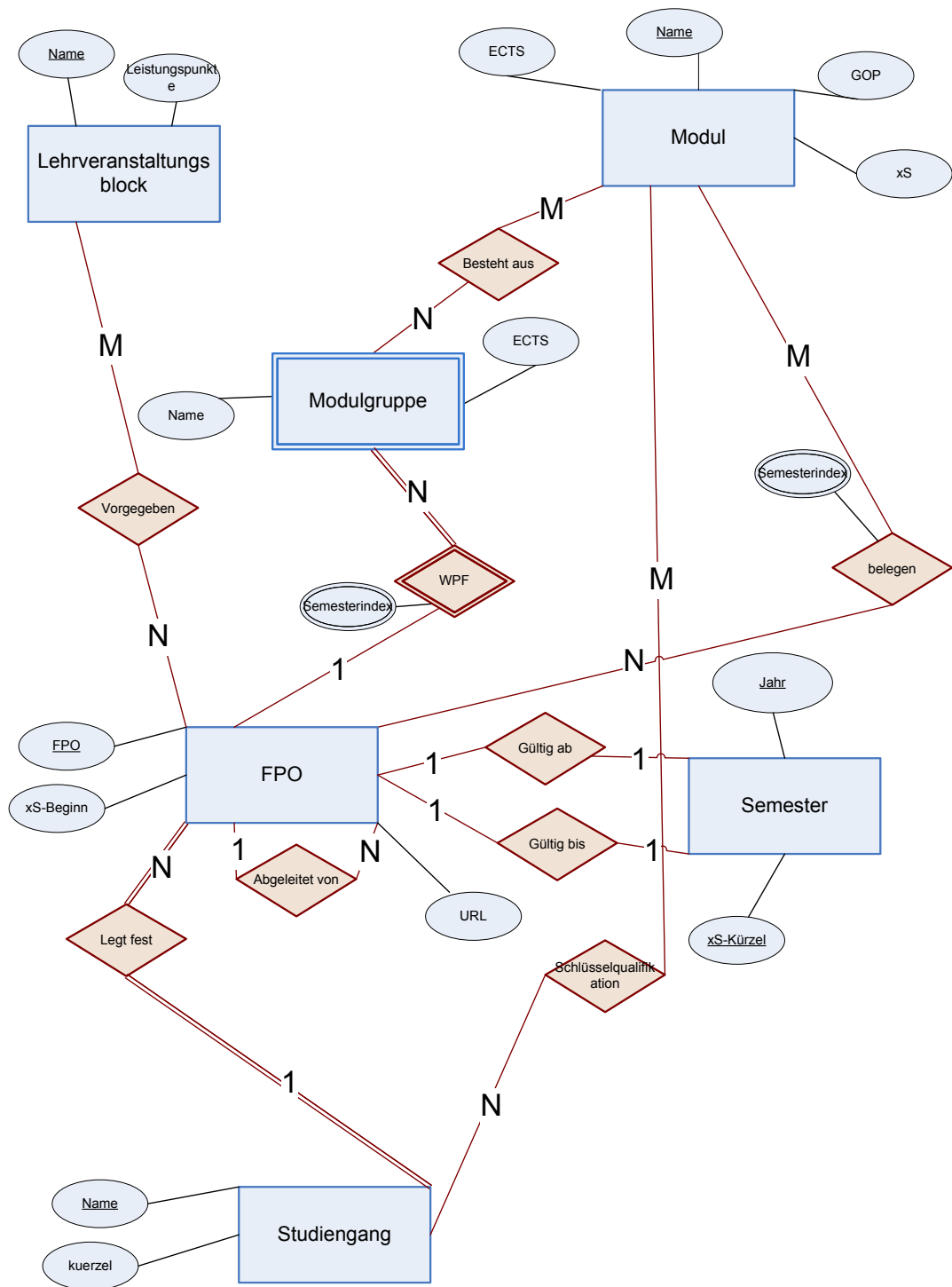


Abbildung 5.6: FPO, Module, Modulgruppen, Lerveranstaltungsblock in E/R-Notation

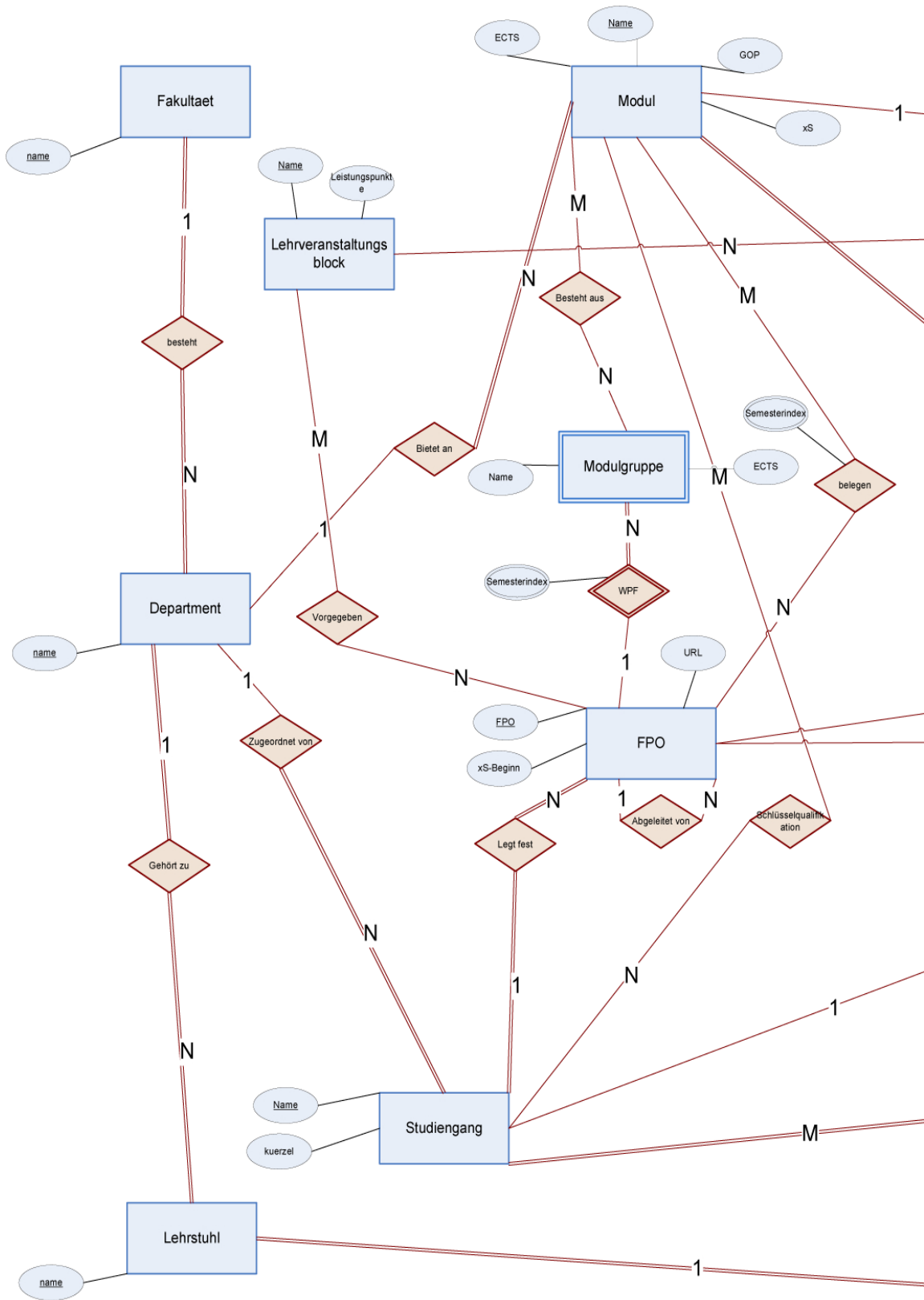


Abbildung 5.7: E/R-Diagramm der Basisdatenbank Teil I

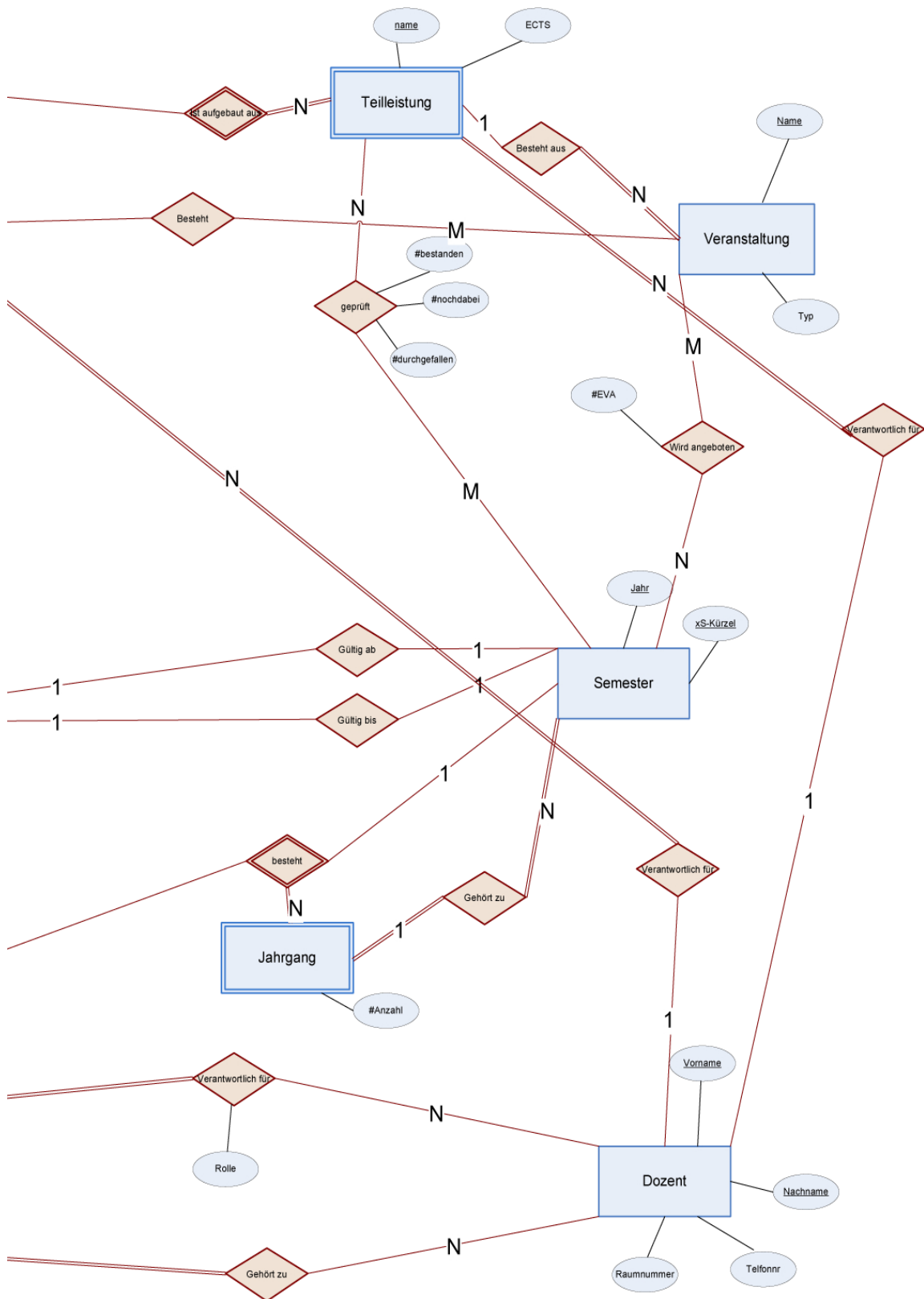


Abbildung 5.8: E/R-Diagramm der Basisdatenbank Teil2

Lehrveranstaltungen, Teilleistungen und Modulen tritt dieses Phänomen auf. Aus diesem Grund wurden für alle Entities außer für die Entity Fachprüfungsordnung Surrogate eingeführt. Eine Fachprüfungsordnung ist immer eindeutig, weil nach dem Namen der Fachprüfungsordnung das Datum der Erstellung miteinfließt. Die neuen Surrogate sehen wie folgt aus: Der Entityname wird mit einem Präfix 'id' versehen und bildet damit den neuen Schlüsselkandidaten der jeweiligen Entity. Man beachtet auch die Entity Dozenten. Hier hätte man auch Personalnummer nehmen können, aber diese Daten liegen nicht vor. Die Surrogate sind fortlaufende Zahlen.

Eine weitere Veränderung äußert sich darin, dass die Lehrblockveranstaltungen aus dem E/R-Diagramm entfernt werden. Bei Beibehaltung würde ein paralleler Weg zu den Lehrveranstaltungen aufgebaut werden. Es existiert ein Weg für die alten Studiengänge wie Magister und Diplom über Lehrveranstaltungsblöcke, und für die neuen Studiengänge wie Bachelor über die Module. In erster Line ist dies nicht problematisch. In der Praxis gestaltet es sich allerdings so, dass neue Lehrveranstaltungen alte Lehrveranstaltungen vom Stoffinhalt abdecken, und somit Studierende der neuen Studiengänge, aber auch der alten Studiengänge in der gleichen Lehrveranstaltung sitzen. Wenn dies der Fall ist, müssten die Datensätze sowohl für die Lehrveranstaltungsblöcke als auch für die Module vorhanden sein. Dies würde eine enorme Redundanz in der Datenbank verursachen. Die Lehrveranstaltungsblöcke müssten manuell herausgesucht werden, weil keine der Datenquellen, die zur Verfügung stehen, diese Daten liefern kann. Somit wäre der Pflegeaufwand, die Lehrveranstaltungsblöcke aus dem Univis <sup>3</sup>[eG09] zu suchen, sehr hoch. Um dieses Problem zu umgehen, wurden Lehrveranstaltungsblöcke entfernt. Damit diese Informationen nicht verloren gehen, mussten diese Informationen anderweitig untergebracht werden. Dies geschieht bei der Entity Modul. Wie in der Anforderungsanalyse schon erklärt wurde, haben Module ähnliche Attribute wie Lehrveranstaltungsblöcke. Aus dem Attribut ECTS-Punkte <sup>4</sup> -werden bei Lehrveranstaltungsblöcken die Leistungspunkte eingetragen. Das Attribut GOP <sup>5</sup> wird allerdings für Lehrveranstaltungen nicht genutzt. Somit würde man viele Null-Werte bekommen. Dieser Nachteil wiegt den Pflegeaufwand von Lehrveranstaltungsblöcken nicht auf.

Als die letzte Veränderung, werden nur die neuen Studiengänge zu betrachten und nicht mehr die alten Magister- und Diplom-Studiengänge. Die alten Studiengänge laufen momentan aus. Es werden schon keine Grundstudiumslehrveranstaltungen mehr für diese Studiengänge angeboten. Somit sind die Informationen, die man für die beiden Hauptziele braucht, nicht mehr so wichtig, dass es den Aufwand rechtfertigen würde, die für diese Studiengänge benötigten Daten zu besorgen.

---

<sup>3</sup>Universität Informationssystem

<sup>4</sup>European Credit Transfer System

<sup>5</sup>Grundlagen und Orientierungsprüfung

Das folgende E/R-Diagramm 5.9 5.10. zeigt, wie sich die technisch begründeten Änderungen auswirken.

## 5.4 Multidimensionales Model

Nachdem das konzeptionelle Schema für die Basis Datenbank beschrieben worden ist, wird nun das multidimensionale Schema für das Data Warehouse System beschrieben.

Um das Data Warehouse zu beschreiben, arbeitet man an zwei Zielen. Ein Ziel ist es, die Studierendenzahl in Lehrveranstaltungen zu bestimmen. Aus Kapitel 4 ist auch schon bekannt welche Daten zur Verfügung stehen. Dies sind die Prüfungsinformationen der Studierenden und die Information, wie viele Studierende in einem Jahrgang eines Studiengangs sind.

Diese beiden Angaben, Jahrgang und Prüfungsinformationen, haben eine gleiche Abhängigkeit, die Zeit. Der Jahrgang hängt außerdem von dem Studiengang ab, und die Prüfungsinformationen benötigen zu der Zeit noch die geprüfte Teilleistung. Somit ist klar, dass die Zeit eine eigene Dimension sein muss.

Im OLAP-Themenbereich wird unter einer Dimension eines Raums eine ausgewählte Entität verstanden, mit der eine Auswertungssicht eines Anwendungsbereichs definiert wird und die der eindeutigen, orthogonalen Strukturierung dient. [AB01]

Aus dem E/R-Diagramm wurden vier Dimensionen entwickelt. Die erste Dimension stellt die Zeit dar, die mit der Entität Semester korreliert ist. Die zweite Dimension steht für die Lehrveranstaltung und die dritte Dimension für Dozent. Diese drei Dimensionen kommen mit nur einer Klassifikationshierarchie und Klassifikationsstufe aus. Diese Dimensionen besitzen jeweils keinen zusammenfassenden Sammelbegriff und keine Unterstrukturen. Die vierte Dimension erweist sich dafür als komplexer:

Betrachtet man das E/R-Diagramm 5.7, so geht von der Entität Fakultät eine Zusammenhangskette aus. Der FAU <sup>6</sup>-Aufbau, auch kurz genannt dimOrg besitzt mehrere Klassifikationshierarchien und Klassifikationsstufen. Die oberste Klassifikationsstufe bedeutet die Fakultät, gefolgt vom Department. Ab dieser Stufe bieten sich nun drei alternative Pfade an:

1. Nach Department folgt der Lehrstuhl. Bei diesem Pfad werden die einzelnen Lehrstühle betrachtet.
2. Nach Department folgt das Modul. Module werden immer von einem Department angeboten. Unter Modul kommt dann die Klassifikationsstufe Teilleistung.

---

<sup>6</sup>Friedrich-Alexander-Universität Erlangen-Nürnberg



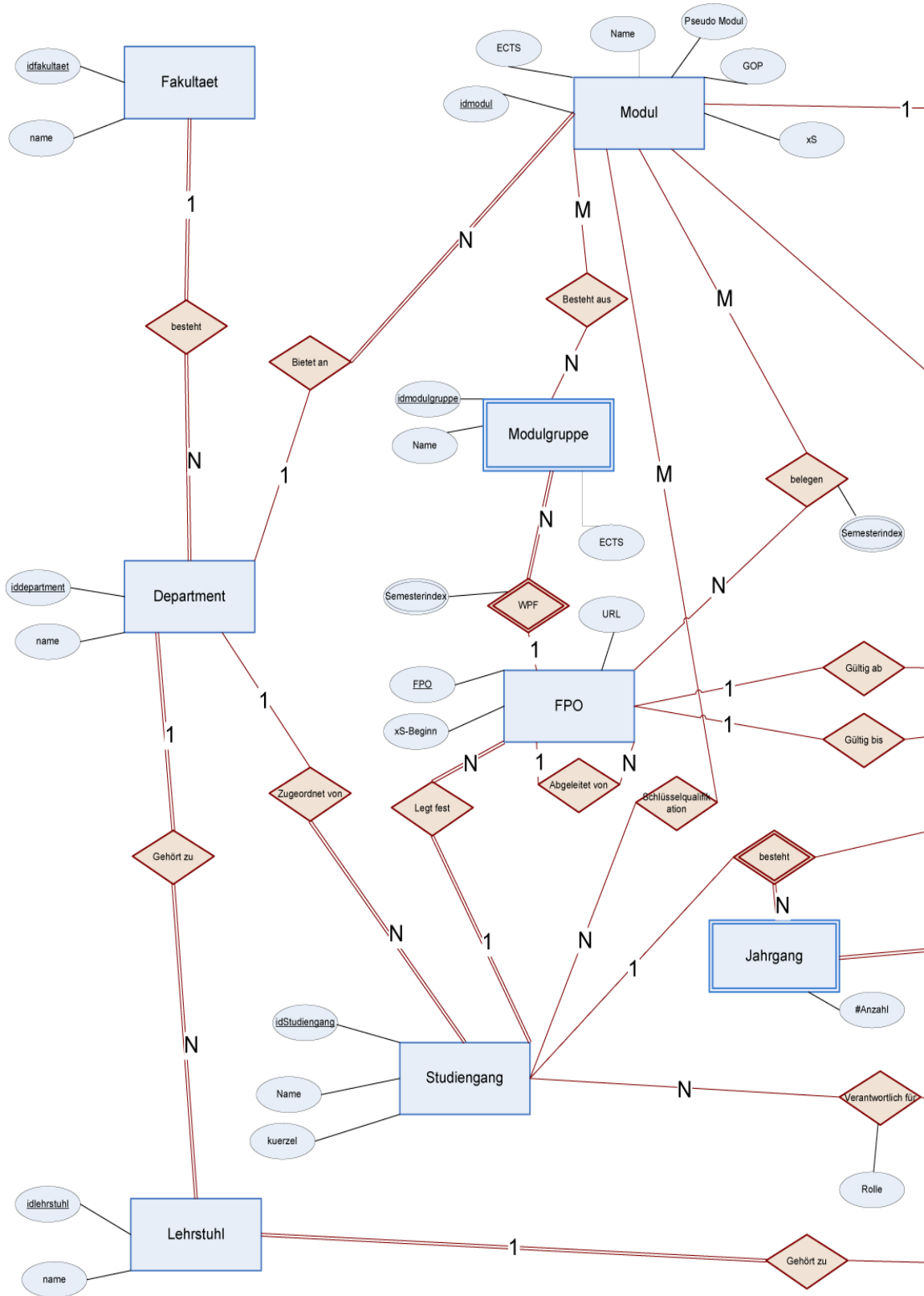


Abbildung 5.9: Optimiertes E/R-Diagramm der Basisdatenbank Teil1

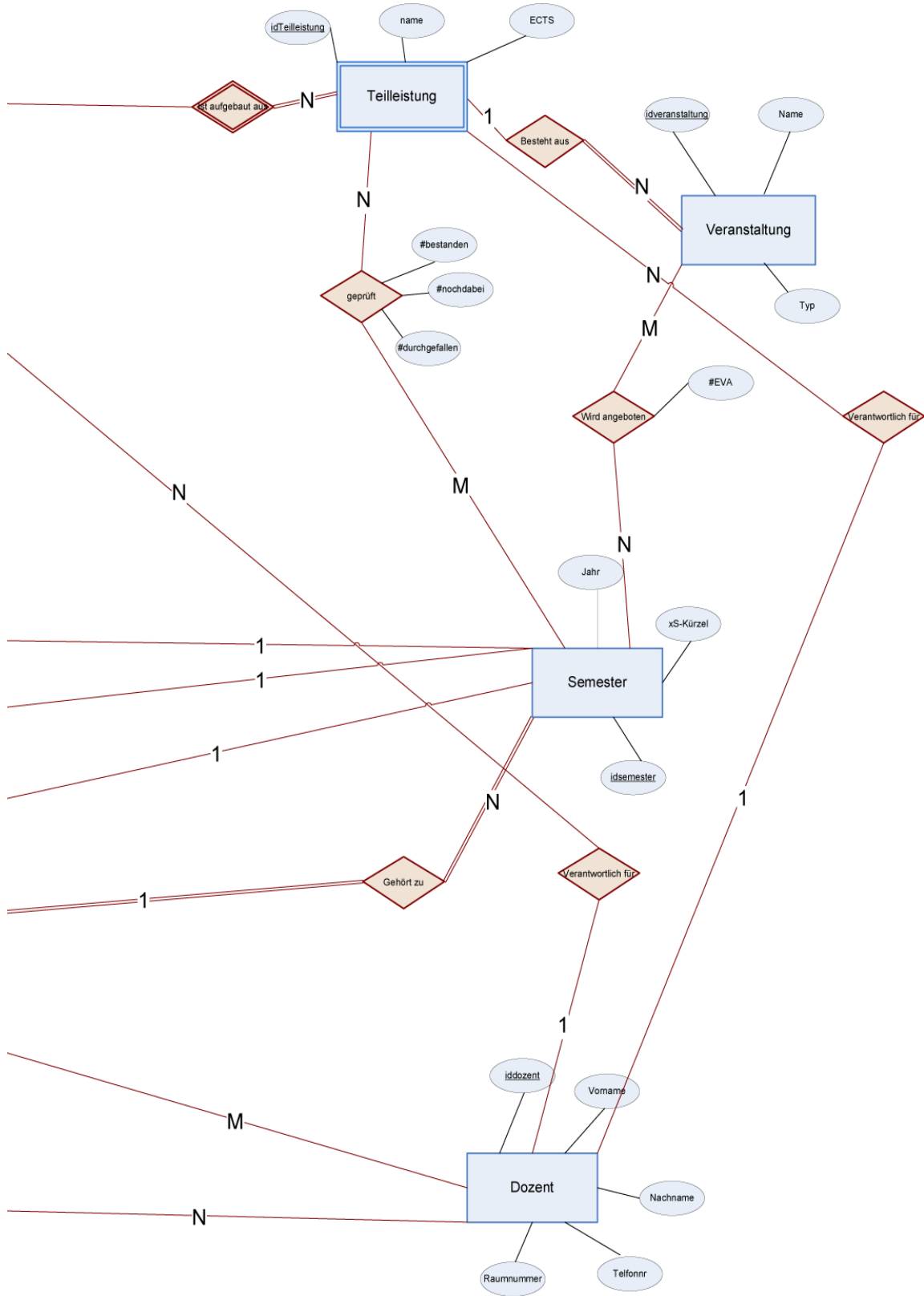


Abbildung 5.10: Optimiertes E/R-Diagramm der Basisdatenbank Teil2

Jedes Modul hat mindestens eine Teilleistung.

3. Nach dem Department kommt als kleinere Klassifikationsstufe Studiengang. Departments bieten Studiengänge an. Auch ist ein Studiengang immer einem Department zugeordnet. Jeder Studiengang besitzt eine Fachprüfungsordnung. In den Fachprüfungsordnungen werden Modulgruppen spezifiziert. Jede Modulgruppe gehört zu einer Fachprüfungsordnung.

Diese drei Dimensionen spannen Würfel für unterschiedliche Faktentabellen auf. In den Faktentabellen wird oft von den Primärschlüsseln von Ausgangsidentitäten gesprochen. Es sei hier noch darauf hingewiesen, dass bis auf die Identität FPO <sup>7</sup> immer ein Surrogat eingefügt wurde.

Hier kommen wieder die Datenquellen ins Spiel. Im Kapitel Datenquellen 4 wurde beschrieben, dass das Campus IT-Team die Studierendenzahlen einer Teilleistung bzw. eines Moduls liefern kann; aufgeschlüsselt in die Kategorien bestanden, nicht bestanden und endgültig nicht bestanden. Diese Informationen sind Grundlage für die erste Faktentabelle, die nur noch durch die Primärschlüssel von Semester und Teilleistung als Fremdschlüssel ergänzt wird. Die beiden Fremdschlüssel bilden für die Faktentabellen den Primärschlüssel. Zusätzlich wird vermerkt, in welchem Semester die Prüfung gestellt wurde. Auch werden die Daten der Prüfung festgehalten. So besitzt die Tabelle die Informationen, ob die Teilleistung eine Prüfung war oder ob sie zusätzlich noch eine GOP war, und wie viele ECTS sie mit sich bringt. Für diese Faktentabelle werden zwei Dimensionen benötigt. Der Würfel für diese Dimensionen wird einmal von dimSemester aufgespannt, da das Semester - also wann eine Leistung erbracht wurde - benötigt wird, und von der Dimension dimOrg, weil der Teilleistungsname für die Informationen relevant ist, die sich in der Dimension Teilleistung befindet.

Die zweite Faktentabelle heißt factEVA und beinhaltet die Anzahl der abgegebenen Stimmen, die bei der Evaluation von Lehrveranstaltungen abgegeben wurden. Mit den Primärschlüsseln aus den Tabellen Semester und Lehrveranstaltung bilden sie den Primärschlüssel der Faktentabelle. Weitere Informationen, die die Faktentabelle beinhaltet, sind der Name der Lehrveranstaltung und deren Typ. Des Weiteren beinhaltet sie die Informationen, in welchem Semester evaluiert wurde. Der Würfel für die Faktentabelle factEVA wird durch die zwei Dimensionen aufgespannt. Wie bei factCIT wird das Semester benötigt, um herauszufinden, in welchem Semester die Evaluation stattgefunden hat. Des Weiteren werden die Lehrveranstaltungen benötigt, die evaluiert werden. Diese Information wird aus der Dimension dimOrg bereitgestellt, die einen Pfad zur Klassifikationsstufe Lehrveranstaltungen besitzt.

---

<sup>7</sup>Fachprüfungsordnung

Die nächste Faktentabelle, die für die Analyse benötigt wird, ist die factJG. JG, dies steht für Jahrgang, und die Tabelle beinhalten die Information, wie viele Studierende eines Studiengangs im gleichen Fachsemester sind. Für die Fachsemester wird wieder die Dimension dimSem benötigt, um den Zeitbezug zu erhalten. Für den Studiengang wird die Dimension dimOrg verwendet, die einen Pfad zum Studiengang besitzt. Da die Jahrgangs- und Semester-Informationen aus den zwei Dimensionen kommen, beinhaltet factJG die Fremdschlüssel von Jahrgang und Semester, die zusammen den Primärschlüssel stellen. Somit befinden sich in dieser Faktentabelle noch die Information vom Fachsemester und Namen des Studiengangs.

Diese drei Faktentabellen sind dafür da, dass am Ende Analysen bezüglich der Fragestellung, wie viele Studierende in eine Lehrveranstaltung gehen, erstellt werden können. Zusätzlich werden noch weitere Informationen benötigt, beispielsweise welche Lehrveranstaltungen zu einer Teilleistung gehören. Nicht zu vergessen sind auch die Verantwortlichen für die Module, Teilleistungen und Lehrveranstaltungen.

Der Informationsaustausch zwischen Lehrveranstaltungen und Teilleistungen wird durch die Faktentabelle facTeilVer realisiert. Die Tabelle hält die Primärschlüssel für Lehrveranstaltung und Teilleistung bereit. Weitere Attribute umfassen den Namen von der Teilleistung, das Kürzel der Lehrveranstaltung sowie die ECTS-Punkte und die Zugehörigkeit der Teilleistung eines Moduls. Für diesen Würfel werden die Dimensionen dimVeranstaltung und dimOrg benötigt. In dimVeranstaltung befindet sich die Klassifikationsstufe Veranstaltung und in dimOrg gibt es einen Pfad zur Klassifikationsstufe Teilleistung.

Als weiterer wichtiger Punkt in der Anforderungsanalyse stehen die Verantwortlichen. Diese Tabelle factVerSF beinhaltet als Fremdschlüssel die Primärschlüssel vom Studiengang und den verantwortlichen Dozenten. Dabei wird vermerkt, welche Rolle der Verantwortliche hat. Als weitere Attribute verfügt die Tabelle über Vor- und Nachname des Dozenten, Raumnummer an der Universität und Telefonnummer des Dozenten. Dazu kommen noch die Informationen des betreffenden Studiengangs, z.B. der Name desselben. Die Dimensionen dimDozent und die dimOrg spannen die den Daten-Würfel. In dimOrg liegt einen Pfad zur Klassifikationsstufe Studiengang vor.

Ein weiterer Verantwortlicher wird für die Teilleistungen definiert. Dafür wird die Faktentabelle factTeilVer angelegt, die als Primärschlüssel die beiden Fremdschlüssel von Dozent und Teilleistung bekommt. Ihre weiteren Attribute sind Vor- und Nachname des Dozenten, Raumnummer an der Universität und Telefonnummer des Dozenten. Dazu kommen noch die Informationen der betreffenden Teilleistungen, beispielsweise die Namen der Teilleistungen und ihre ECTS-Punkteanzahl. Die Dimensionen, die den Daten-Würfel aufspannen, sind die dimDozent und die dimOrg. In dimOrg gibt es einen Pfad zur Klassifikationsstufe Teilleistung.

Der letzte Verantwortliche bezeichnet denjenigen für die Module. Dieser ist der Ansprechpartner für Veränderungen in Modulen. Hierfür existiert ein Würfel mit der Faktentabelle factVerMod erstellt. Als Primärschlüssel dienen die Fremdschlüssel von Dozent und Modul. Die weiteren Attribute nennen sich Vor- und Nachname des Dozenten, Raumnummer an der Universität und Telefonnummer des Dozenten. Für die Module beinhaltet die Tabelle Eigenschaften des Moduls wie ECTS-Punkte, GOP und Pseudomodul. Als würfelaufspannende Dimensionen werden dimDozent und dimOrg verwendet. Die letztere besitzt die Klassifikationsstufe Modul.

Dozenten gehören immer auch einem Lehrstuhl an. Aus diesem Grund wird auch die Dozentenzugehörigkeit analysiert. Für diesen Fall gibt es die Faktentabelle factVerLS, die als Primärschlüssel die Fremdschlüssel von Dozent und Lehrstuhl besitzt. Weitere Attribute sind Vor- und Nachname des Dozenten, Raumnummer an der Universität und Telefonnummer des Dozenten, und vom Lehrstuhl kommen Attribute wie der Name des Lehrstuhls und welchem Department er angehört. Die verwendeten Dimensionen, um den Würfel aufzuspannen, sind dimDozent und der Klassifikationspfad von dimOrg zu Lehrstuhl.

Um Studierenden bei ihrer Wahl von Modulgruppen zu helfen, werden die Informationen benötigt, die Auskunft darüber geben, welches Modul sich in einer Modulgruppe befindet. Dafür gibt es die Faktentabelle factModModgr. Sie besitzt als Primärschlüssel die Fremdschlüssel von Modul und Modulgruppe. Von Modul kommen weitere Attribute wie der Name des Moduls, ECTS-Punkteanzahl, in welchem xS das Modul angeboten wird, ob es eine GOP ist und ob es sich um ein Pseudomodul handelt. Die Modulgruppe besitzt folgende Attribute: Name der Modulgruppe, ihre ECTS-Punkteanzahl und in welchem xS mit dieser Modulgruppe angeboten wird. Für diesen Würfel gibt es nur eine Dimension, und zwar die dimOrg, die in parallelen Klassifikationspfaden sowohl Modulgruppe als auch Modul beinhaltet.

Die Fachprüfungsordnung stellt der letzte Analyseblock dar. Für den Exportbeauftragten erweisen sich drei Fragestellungen als wichtig:

1. Welche Module schreibt eine Fachprüfungsordnung vor?
2. Wie lange ist bzw. war eine Fachprüfungsordnung gültig?
3. Von welcher Fachprüfungsordnung ist die vorliegende eventuell abgeleitet?

Für Punkt eins gibt es die Würfel mit Faktentabelle factFPOMod. Die Faktentabelle hat als Primärschlüssel die Fremdschlüssel von Modul und Fachprüfungsordnung. Zusätzlich besitzt die Faktentabelle die Informationen eines Moduls, beispielsweise Name des Moduls, ECTS-Punkteanzahl, in welchem xS das Modul angeboten wird, ob

es sich um eine GOP oder um ein Pseudomodul handelt. Von der Fachprüfungsordnung kommen folgende Attribute: xS für die Entscheidung, ob sie für einen WS- oder SS-Studienanfang gültig ist. Weiterhin noch ein Semesterindex zur Definition, in welchem Semester die Fachprüfungsordnung gültig ist. Der Würfel wird nur von der Dimension dimOrg aufgespannt.

Punkt zwei behandelt die Problematik, dass sich Fachprüfungsordnungen gleicher Studiengänge überschneiden können, und somit muss der Exportbeauftragte wissen, ab wann und bis wann eine Fachprüfungsordnung ihre Gültigkeit besitzt. Hierfür gibt es die Faktentabelle factGültigkeit. Sie besitzt als Primärschlüssel den Fremdschlüssel des Namen einer Fachprüfungsordnung und verfügt als weitere Attribute die Werte, ab wann die Fachprüfungsordnung in Kraft getreten ist oder eventuell wann sie ausläuft. Weiter besitzt sie die genaue Webadresse, auf der die Fachprüfungsordnung auf der Webseite des Prüfungsamtes zu finden ist, und ob man nach dieser Fachprüfungsordnung im Sommer- oder im Wintersemester beginnen kann. Da sich diese Faktentabelle nur auf eine Dimension bezieht, wird für den Würfel nur die Dimension dimOrg verwendet, die einen Pfad zur Klassifikationsstufe Fachprüfungsordnung besitzt.

Wenn eine neue Fachprüfungsordnung erstellt wird und eine alte dafür ausläuft, kommt es immer wieder zu Überschneidungen, welche Module bzw. Lehrveranstaltungen gehört werden müssen. So kann es sein, dass im Semester eines Studiengangs Studierende in einer Lehrveranstaltung sitzen, die nicht alle die gleiche Fachprüfungsordnung haben. Dies passiert, wenn die Lehrveranstaltungen in den Fachprüfungsordnungen eines Studiengangs verschoben werden. Für diese Informationen gibt es die Faktentabelle factAbl. Sie besteht aus dem Primärschlüssel der Fachprüfungsordnung und besitzt zusätzlich nur den Namen der abgeleiteten Fachprüfungsordnung. Wie bei der Faktentabelle factGültigkeit wird hier nur die Dimension dimOrg verwendet, weil diese die Klassifikationsstufe Fachprüfungsordnung besitzt.

Die letzte Faktentabelle lautet factSQ wobei SQ für Schlüsselqualifikation steht. Mit dieser Faktentabelle kann der Exportbeauftragte die angebotenen Schlüsselqualifikationen für den betreffenden Studiengang herausfinden. Als Primärschlüssel besitzt factSQ die Fremdschlüssel von Studiengang und Modul. Aufgespannt wird der Datenwürfel von der Dimension dimOrg. Diese Dimension beinhaltet die Informationen bezüglich Module und Studiengang.

Die folgende Grafik beschreibt die Dimension mit den einzelnen Faktentabellen.

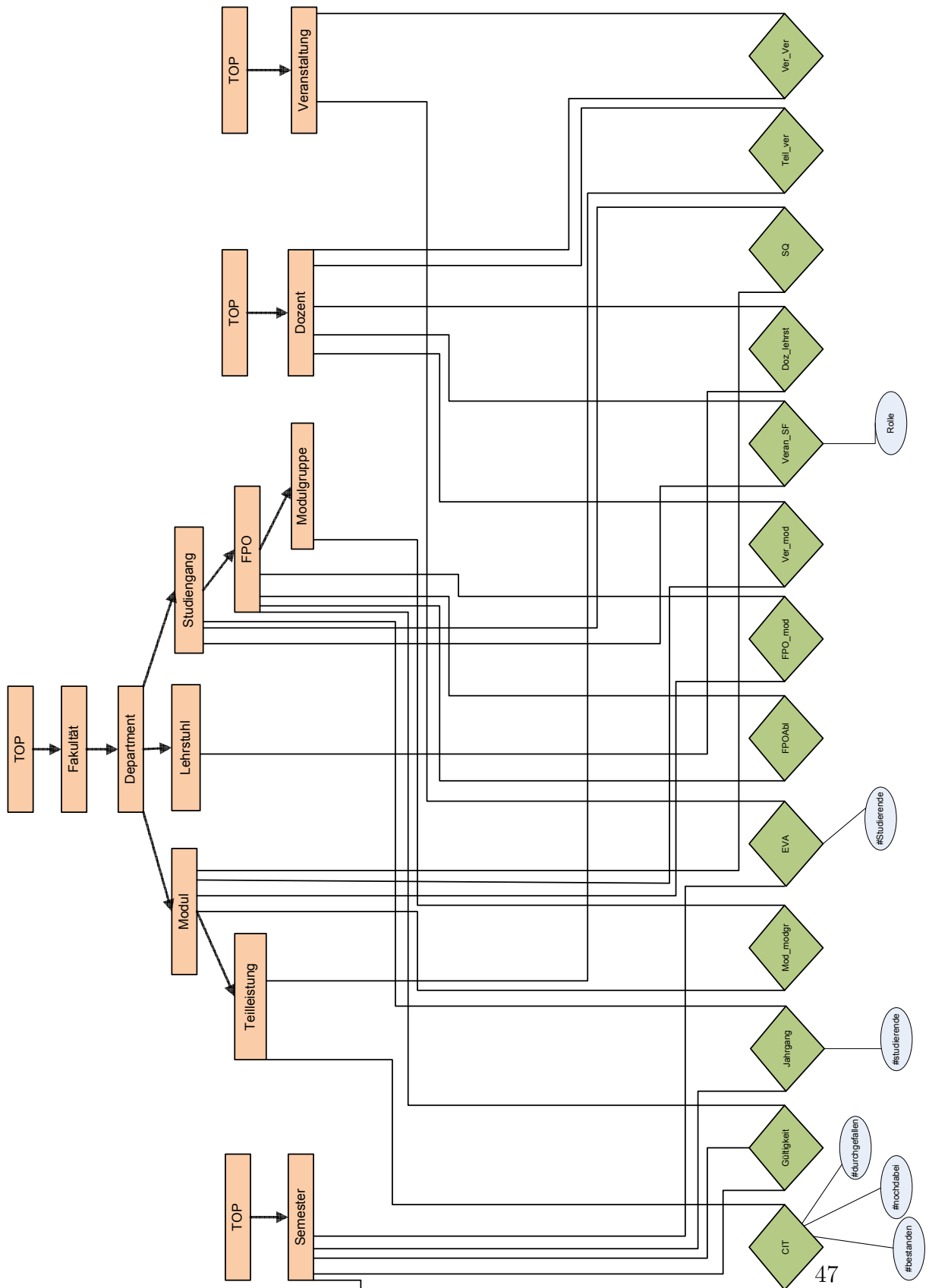


Abbildung 5.11: Dimensionsmodell





# 6 Systementwurf

## 6.1 Basisdatenbank

Trotz des zusätzlichen Aufwands für den Aufbau und die Pflege wurde eine relationale Basisdatenbank als Fundament für das Data Warehouse eingerichtet. Durch die Verwendung von zwei Datenquellen hätten die Daten vor dem Einfügen ins Data Warehouse System bereinigt sein müssen. Durch eine Basisdatenbank kann das Data Warehouse System direkt beladen werden, weil die Bereinigung der Daten schon beim Einfügen in die Basisdatenbank geschieht. Somit können Analysen gefahren werden, während derer die Basisdatenbank sich aktualisiert. Es gibt keine Blockade durch Sperrungen. Ein weiterer Vorteil einer Basisdatenbank ist, dass sie für keine Analyse optimiert wurde, sondern eher auf die Anwendungsneutralität zugeschnitten ist. Ins Data Warehouse System muss auch nicht immer die gesamte Datenmenge der Ausgangstabelle eingefügt werden. Vielleicht wird nur ein aktueller Auszug aus dem Datenbestand benötigt. Die Basisdatenbank dient als Sammel Speicher für alle anfallenden Daten. Dabei steht es frei, später weitere Datenquellen zu erschließen. Ein Nachteil ist, dass eine zweite Datenbank für die Basisdaten bereit stehen muss, und die Daten, wenn sie sich im Data Warehouse und in der Basisdatenbank befinden, doppelt vorhanden sind. Dies bedeutet, dass doppelter Speicherplatz zur Verfügung stehen muss.

Mit einer zentralen Datenbasis, die dem Lehrstuhl direkt zur Verfügung steht, können weitere Data Warehouse Systeme entwickelt werden, um zum Beispiel die Akzeptanz von Wahlveranstaltungen zu überprüfen und dann das Lehrangebot anzupassen.

Es wurde oben erwähnt, dass eine relationale Datenbank verwendet wird. Der Grund dafür ist, dass alle für das Projekt zugänglichen Basisdatenbanksysteme relationale Datenbanksysteme sind. Die Betrachtung wurde soweit eingeschränkt, dass die Hersteller der Data Warehouse Systeme fast immer ein relationales Datenbanksystem als bevorzugtes Basisdatenbanksystem verlangen. Es geht meist soweit, dass die Hersteller der Data Warehouse Systeme einen oder zwei spezielle Anbieter von relationalen Datenbanken vorschreiben.

### 6.1.1 Basisdatenbank ein Operational Data Store?

Ein Operational Data Store wird als Vorstufe eines Data Warehouse Systems bezeichnet. Daten werden auf Transaktionsbasis von verschiedenen operativen Quellsystemen gesammelt und für Anwendungen, beispielsweise für Analyse-Systeme, wieder bereitgestellt.

Wie nach Andreas Bauer und Holger Günzel im Buch "Data Warehouse System" beschrieben [AB01], folgt die eingesetzte Basisdatenbank der damit verbundenen Eigenschaft, nämlich der Klasse zwei der Operational Data Stores.

Die zweite Klasse wird wie folgt beschrieben: Datenänderungen werden gesammelt, integriert, transformiert und in gewissen Zeitabständen in den ODS <sup>1</sup> übertragen, um im Data Warehouse ausgewertet zu werden zu können. Die im ODS integrierten Daten werden in das Data Warehouse übertragen. [AB01]

Dieses Vorgehen bildet den Datenfluss dieses Projektes ab. Erst werden Daten gesammelt, d.h. sie werden den Datenquellen entnommen, die dann transformiert und auf das Schema der Basisdatenbank angepasst werden. Transformationen müssen vorgenommen werden, weil die Daten aus den Datenquellen nicht das gleiche Format vorweisen, wie es in der Basisdatenbank benötigt wird. Das Informationsupdate aus den Datenquellen wird immer vor den Vorlesungszeiträumen etabliert. In periodischen Zeitabständen wird dann die komplette Basisdatenbank mit neuen Informationen gefüllt. Durch Abhängigkeiten, die zwischen Fremdschlüsselbeziehungen bestehen, ist dies notwendig. Die Fremdschlüsselabhängigkeiten werden im Folgenden verdeutlicht.

Das periodische Nachladen der Basisdatenbank besitzt zwei Abhängigkeiten. Die erste Abhängigkeit ist, dass die Informationen, wie viele Studierende in einer Lehrveranstaltung sein sollten, schon vor Semesterbeginn vorliegen sollten. Die zweite Abhängigkeit ist, ab wann die Daten in den Datenquellen für ein Semester vorliegen. Betrachtet man die Informationen bezüglich eines Jahrgangs, so liegen diese Daten ziemlich exakt in der Mitte eines Semesters vor. Am Anfang der Vorlesungszeit kann man die Zahlen nur eingeschränkt verwenden, weil es immer noch Fluktuationen durch ausländische Studierende, die sich erst später immatrikulieren können, oder durch Studienplatzwechsler, geben kann. Bei den Prüfungsdaten ist die Vorausssehbarkeit, wann die Daten vorliegen, noch schwieriger einzuschätzen. Hier gibt es Abhängigkeiten, wann die Prüfungen sind: entweder am Ende einer Vorlesungszeit oder am Anfang der nächsten Vorlesungszeit. Des Weiteren weiß man nicht, wie ausgelastet ein Lehrstuhl während der Prüfungszeit ist, und wie viel Zeit er benötigt, die Klausuren zu korrigieren, bevor sie dann ins System eingetragen werden.

---

<sup>1</sup>Operational Data Store

Aus dem oben Beschriebenen ergibt sich folgendes zeitliches Problem: Wann sollen bzw. können die Daten von den Quellen abgeholt werden und in die Basisdatenbank integriert werden? Der beste Zeitpunkt ist in der Mitte eines Semesters, weil dann die Prüfungsergebnisse fast vollständig vorliegen, und die Jahrgangsinformationen korrekt sein sollten.

## 6.2 Logische Ebene

Nach der Entscheidung eine relationale Basisdatenbank einzusetzen, wird ein logisches Schema auf Basis des E/R-Diagramms benötigt. Hier wird das technische E/R-Diagramm 5.9 verwendet, um aus ihm die Relationen zu generieren. Dabei wird immer am Ende eines Attributs vermerkt, welches Attribut ein Schlüsselkandidat sein kann. Markiert werden diese Attribute mit dem Zeichen <CK>. CK steht für den englischen Ausdruck Candidate Key. Die Schlüsselkandidaten werden nochmals erwähnt, weil im Kapitel 5.3 diskutiert wurde, bei einfacherer Verarbeitung Surrogate für die Primärschlüssel zu verwenden. Bei vielen Relationen ist der Name zwar schon eindeutig, deswegen ist oft der Name auch der Schlüsselkandidat, aber dies trifft nicht für jede Relation zu. Für die Primärschlüssel gilt immer, dass sie nicht Null sein dürfen und auch eindeutig sein müssen.

*Für die Umsetzung vom E/R-Diagramm wurden die Vorgehensweisen aus den Büchern "Datenbanksysteme"[AK06] und "Grundlagen von Datenbanksystemen"[SBN02] verwendet.*

Bei der Abbildung des E/R-Diagramms wird mit der Fakultät begonnen. Sie ist der wichtigste Einstiegspunkt für die spätere Navigation im Schema und bildet durch ihre Einfachheit einen guten Einstieg. Die Fakultät wird durch ihr Surrogat Primärschlüssel `idfakultaet` und den Schlüsselkandidaten `name`, welcher den Fakultätsnamen repräsentiert, beschrieben. Der Fakultätsname darf auch nicht Null sein, weil jede Fakultät an der Universität Erlangen-Nürnberg einen Namen besitzt.

```
fakultaet(idfakultaet:integer, name:string)
```

```
name NOT NULL
name <CK>
```

Den Fakultäten sind die Departements angegliedert. Die Relation `Department` hat ein Surrogat namens `idDepartment`, der Primärschlüssel ist. Des Weiteren besitzt `Department` ein Attribut namens `name`, welches den Namen des Departments betitelt. Der `name` ist auch Schlüsselkandidat. Aus diesem Grund darf auch er nicht Null sein. Als Fremdschlüssel von Fakultät besitzt `Department` noch ein Attribut `idfakultaet`, das auf den Primärschlüssel der Relation `Fakultät` zeigt. Jedes `Department` gehört zu

einer Fakultät und kann klar zugeordnet werden. Deswegen gibt es den Fremdschlüssel zur Fakultät.

```
department(iddepartment:integer, name:string,  
           idfakultaet[fakultaet]:integer)
```

```
(name, idfakultaet[fakultaet]) NOT NULL  
name <CK>
```

Betrachtet man die Universitätsstruktur noch ein Stück näher, dann stehen als nächstkleinere Einheiten unter den Departements die Lehrstühle. Der Lehrstuhl hat als Primärschlüssel das Surrogat idLehrstuhl. Als weiteres Attribut und Schlüsselkandidaten besitzt es das Attribut Name, das den Namen des Lehrstuhls wiedergibt. Ein Lehrstuhl hat immer einen Namen und darf deswegen nie Null sein. Jeder Lehrstuhl ist auch genau einem Departement zugeordnet. Somit besitzt die Relation Lehrstuhl einen Fremdschlüssel zu Department.

```
lehrstuhl(idlehrstuhl:integer, name:string,  
          iddepartment[department]:integer)
```

```
(name, iddepartment[department]) NOT NULL  
name <CK>
```

Die Relation Dozent wird als Nächstes abgebildet. Neben dem Surrogat idDozent, der den Primärschlüssel für die Relation stellt, gibt es hier eine Besonderheit. Der Schlüsselkandidat besteht bei dieser Relation aus drei Teilen, dem Vor- und Nachnamen des Dozenten sowie der Raumnummer, die auf das Büro des Dozenten verweist. Die Attribute Vor- und Nachnamen und Raumnummer sollten durch die Schlüsselkandidateneigenschaft nicht Null sein und sollten zusammen eindeutig sein. Des Weiteren besitzt die Relation noch das Attribut Telefonnummer. Jeder Dozent ist an einem Lehrstuhl beschäftigt. Somit besitzt die Relation einen Fremdschlüssel von der Relation Lehrstuhl auf idLehrstuhl.

```
dozent(iddozent:integer, vorname:string, nachname:string,  
       telefonnr:integer, raumnr:integer, idlehrstuhl[lehrstuhl]:integer)
```

```
(vorname, nachname, raumnr, telefonnr,  
 idlehrstuhl[lehrstuhl]) NOT NULL  
(vorname, nachname, raumnummer) UNIQUE  
(vorname, nachname, raumnummer) <CK>
```

Studiengang ist die nächste betrachtete Relation. Sie besitzt wie alle zuvor beschriebenen Relationen ein Surrogat als Primärschlüssel mit namens idStudiengang. Der Name eines Studiengangs wird im Attribut Name verbucht und kann kein Schlüsselkandidat

sein. Es gibt beispielsweise zwei Studiengänge Informatik. Der eine Studiengang endet mit dem Abschluss Diplom und der andere mit Bachelor. Auch unterscheiden sich die Studiengänge der Bachelor-Informatik darin, ob der Studienbeginn im Sommersemester oder im Wintersemester war. Viele Studiengänge werden noch abgekürzt, aber dieses Attribut kann Null sein, weil es nicht jeden Studiengang betrifft. Ein Studiengang wird immer von einem Department angeboten. Deswegen hat die Relation Studiengang einen Fremdschlüssel von der Relation Department, der nicht Null werden darf.

```
studiengang(idstudienfach:integer, name:string, kuerzel:string,
            iddepartment[department]:integer)
```

```
(name, iddepartment[department]) NOT NULL    name <CK>
```

Die Lehrveranstaltung besitzt wie die anderen Relationen auch einen Surrogatprimärschlüssel namens idVeranstaltung. Ein weiteres Attribut ist der Typ der Lehrveranstaltung. Eine Veranstaltung ist immer ein Typ, beispielsweise Vorlesung, Übung oder Praktikum. Jede Lehrveranstaltung sollte auch einen Ansprechpartner haben. Dies wird über eine Fremdschlüsselbeziehung zu Dozent realisiert. Auch hat eine Lehrveranstaltung einen Namen, der als Schlüsselkandidat gelten kann. Ein weiterer Fremdschlüssel zeigt auf die Teilleistung, in der die Lehrveranstaltung eingeordnet ist. Eine Lehrveranstaltung gehört immer zu einer Teilleistung.

```
veranstaltung(idveranstaltung:integer, name:string, typ:string,
              iddozent[dozent]:integer, idteilleistung[teilleistung]:integer)
```

```
(name, idteilleistung[teilleistung]) NOT NULL
name <CK>
```

Die Zeit wird durch die Relation Semester ausgedrückt. Das Semester hat einen Surrogatprimärschlüssel namens idSemester. Die weiteren Attribute Jahr und xS sind die Schlüsselkandidaten. Aus diesem Grund dürfen diese Attribute nicht Null sein. Der Verbund aus beiden Attributen soll eindeutig sein. Mit xS und dem Jahr kann genau ein Semester adressiert werden.

```
semester(idsemester:integer, jahr:integer, xS:string)
```

```
(jahr, xS) NOT NULL
(jahr, xS) <CK>
```

Die größte Relation stellt die Relation Modul dar. Modul besitzt einen Surrogatprimärschlüssel namens idModul. Ein Modul hat auch einen Namen, welcher der

Schlüsselkandidat ist. Weiterhin hat Semester die Attribute ECTS<sup>2</sup>, GOP<sup>3</sup>, Pseudomodul und xS, von denen Pseudomodul und xS nicht Null sein dürfen. Semester besitzt auch Fremdschlüssel. Ein Modul wird immer von einem Department angeboten, somit besitzt die Relation Modul einen Fremdschlüssel zum Department mit Namen idDepartment. In der Anforderungsanalyse wird beschrieben, dass Modul einen Ansprechpartner haben soll. Deswegen besitzt Modul einen Fremdschlüssel zu Dozent mit dem Namen idDozent.

```
modul(idmodul:integer, name:string, ects:integer, gop:boolean,  
      pseudomodul:boalean, xS:string, iddozent[dozent]:integer,  
      iddepartment[department]:integer)
```

```
(name, pseudomodul, xS, iddepartment[department]) NOT NULL  
name <CK>  
pseudomodul NOT NULL  
(Wenn nur Bachelor betrachtet wird, ist NULL erlaubt)
```

Die artverwandte Relation zu Modul ist Modulgruppe. Sie besitzt einen Primärschlüssel namens idModulgruppe. Wie bei Modul hat auch Modulgruppe den Namen als Schlüsselkandidaten. Als weiteres Attribut besitzt die Relation Modulgruppe ECTS. Einen Fremdschlüssel von FPO gibt es mit dem Namen FPOnamen. Ein weiterer Fremdschlüssel ist idr\_semidx\_modgr, der auf die Relation r\_semidx\_modgr zeigt. Dieser Fremdschlüssel gibt an, in welchem Semester die Ablegung einer Modulgruppe erlaubt ist.

```
modulgruppe(idmodulgruppe:integer, name:string, ects:integer,  
            fponame[fpo]:string, idr_semidx_modgr[r_semidx_modgr])  
(name, idr_semidx_modgr[r_semidx_modgr]) NOT NULL  
name <CK>
```

Im System gibt es auch schwache Entitäten. Die Relation Jahrgang besteht aus einem zusammengesetzten Primärschlüssel, nämlich aus Fremdschlüsseln von Semester und Studiengang mit den Namen idSemester und idStudiengang. Ein weiteres Attribut der Relation ist die Anzahl, die mitteilt, wie viele Studierende in einem Jahrgang sind. Bei dieser Relation gibt es keinen Schlüsselkandidaten.

```
jahrgang(idsemester[semester]:integer, idstudiengang[studiengang]:integer,  
anzahl:integer)
```

```
anzahl NOT NULL
```

---

<sup>2</sup> European Credit Transfer System Punkte

<sup>3</sup> Grundlagen und Orientierungsprüfung

Die zweite schwache Identität ist Teilleistung. Teilleistung besitzt einen Primärschlüssel aus idTeilleistung, gepaart mit dem Fremdschlüssel aus Modul idModul. Des Weiteren besitzt diese Relation einen Attributnamen Name, der nicht Null sein darf, und das Attribut ECTS.

```
teilleistung(idteilleistung:integer, idmodul[modul]:integer,
             name:string, ects:integer)
```

name NOT NULL

Als Entität die kein Surrogat aufweisen kann ist Entität FPO<sup>4</sup> betrachten. FPO hat als Primärschlüssel ihren Namen, der eindeutig ist, weil in den Namen ein Datum miteinfließt, und kein Studiengang zwei FPOs an einem Tag erlassen wird. Die weiteren Attribute der Relation sind ECTS und URL<sup>5</sup>. URL darf auch nicht Null sein, weil immer ein Link für die FPO beim Prüfungsamt für eventuell notwendige Recherchen hinterlegt sein muss. Auch besitzt FPO drei Fremdschlüssel, zwei zur Relation Semester, die jeweils den Beginn bzw. das Ende der Gültigkeit der FPO anzeigt. Eine FPO muss immer einen Startgültigkeitszeitpunkt haben. Somit muss dieser Fremdschlüssel immer ein Wert haben. Der dritte Fremdschlüssel zeigt auf sich selbst. Falls eine neue FPO von einer alten abgeleitet ist, wird dies vermerkt. Der letzte Fremdschlüssel, idstudienfach, gibt die Zugehörigkeit zum Studiengang an.

```
fpo(fponame:string, idsemester1[semester]:integer,
    idsemester2[semester]:integer, xS:string, url:string,
    fponame[fpo]:string, idstudienfach[studiengang]:integer)
```

(idsemester1[semester], url) NOT NULL

Mit den bis jetzt beschriebenen Relationen sind alle Entitäten sowie die 1:1- und 1:N-Beziehungen abgedeckt, allerdings nicht die N:M-Beziehungen. Für die N:M-Beziehungen werden weitere Relationen benötigt.

Die erste Relation befasst sich mit der N:M-Beziehung zwischen FPO und Modul. Der Name der Relation ist r\_fpo\_mod. Sie besitzt als Primärschlüssel die Fremdschlüssel von FPO und Modul, also FPOname und idModul. Des Weiteren existiert das Attribut idr\_semidx\_mod, deren Fremdschlüssel auf die Relation r\_semidx\_mod zeigt. Diese Relation gibt an, in welchem Semester dieses Modul abgelegt werden muss. Deswegen darf auch idr\_semidx\_mod nicht Null sein.

```
r_fpo_mod(idmodul[modul]:integer, fpo[name]:string,
          idr_semidx_mod[r_semidx_mod]:integer)
```

---

<sup>4</sup>FPO

<sup>5</sup> Uniform Resource Locator <sup>6</sup>

```
idr_semidx_mod[r_semidx_mod] NOT NULL
```

Die nächste Relation gibt an, welche Module zu welcher Modulgruppe gehören. Die Relation heißt `r_mod_modgr` und besitzt als Primärschlüssel die Fremdschlüssel `idModul` und `idModulgruppe` von `Modul` und `Modulgruppe`.

```
r_mod_modgr(idmodulgruppe[modulgruppe]:integer,  
idmodul[modul]:integer)
```

Ein Studiengang kann mehrere Verantwortliche haben. Aus diesem Grund muss dafür eine eigene Relation aufgestellt werden. Ein Dozent kann für mehrere Studiengänge verantwortlich sein. Somit stellt diese Situation eine N:M-Beziehung dar. Die Relation, die die Verantwortlichen für den Studiengang realisiert, heißt `r_sfver`. Sie besitzt als Primärschlüssel die Fremdschlüssel von `Dozent` und `Studiengang`, `idDozent` und `idStudiengang`. Da es für einen Studiengang mehrere Verantwortliche geben kann, die unterschiedliche Aufgaben haben, gibt es ein weiteres Attribut namens `Rolle`. Dieses Attribut beschreibt die jeweilige Aufgabe des Verantwortlichen.

```
r_sfver(idstudiengang[studiengang]:integer, iddozent[dozent]:integer,  
rolle:string)
```

In der Anforderungsanalyse wurde beschrieben und in der Definition der Informationsquellen weiter ausgeführt, dass die Informationen vorliegen, wie viele Studierende eine Teilleistung bestanden, nicht bestanden oder endgültig nicht bestanden haben. Für diese Information wird eine Relation zwischen Teilleistung und Semester angelegt. Von Semester und Teilleistung stammt auch der zusammengesetzte Primärschlüssel aus `idSemester` und `idTeilleistung`. Als weitere Attribute hat die Relation die wichtigen Informationen, nämlich die Studierendenzahlen, aufgegliedert nach bestanden, nicht bestanden und endgültig nicht bestanden. Für die drei Attribute gilt, dass sie nicht Null sein dürfen. Der Name der Relation ist `r_cit`.

```
r_cit(idmodul[modul]:integer, idsemester[semester]:integer,  
bestanden:integer, durchgefallen:integer, nochdabei:integer)
```

```
(bestanden, durchgefallen, nochdabei) NOT NULL
```

Die nächste Relation berücksichtigt den Fall, dass in der Zukunft noch die Evaluationsdaten als Datenquelle hinzukommen. Eine Evaluation wird jedes Semester in jeder Lehrveranstaltung des Departements Informatik abgehalten. Somit besteht die N:M-Beziehung zwischen Lehrveranstaltung und Semester. Die beiden Fremdschlüssel bilden den zusammengesetzten Primärschlüssel für die Relation `r_eva`. Als weiteres Attribut besitzt die Relation die Anzahl der Studierenden, die sich an der Evaluation beteiligt haben.



```
r_eva(idveranstaltung[veranstaltung]:integer,  
idsemester[semester]:integer, anzahl:integer)
```

Eine weitere N:M-Relation ist die Schlüsselqualifikationsrelation. Sie beschreibt welche Module als Schlüsselqualifikation für einen Studiengang angerechnet werden können. Als Primärschlüssel besitzt diese Relation die Fremdschlüssel von Studiengang und Modul. Der Name der Relation ist r\_sq.

```
r_sq(idstudiengang[studiengang]:integer],idmodul[modul]:integer)
```

Jetzt sind alle Entities und Relationships ausgedrückt, aber es fehlen noch zwei Hilfsrelationen. Für die Relation Modulgruppe und die M:N-Relation zwischen Modul und FPO wird jeweils ein eigener Semesterindex benötigt. Dieser ist im E/R-Diagramm als mehrwertiges Attribut vermerkt.

Der Semesterindex für die Modulgruppe besitzt als Primärschlüssel idr\_semidx\_modgr und die Relation selbst heißt r\_semidx\_modgr. Als Attribute besitzt die Relation noch Jahr und xS, welche im Zusammenspiel das Jahr und das Semester beschreibt. Diese beiden Attribute dürfen nicht Null sein.

```
r_semidx_modgr(idr_semidx_modgr:integer, jahr:integer, xS:string)
```

```
(jahr, xS) NOT NULL
```

Der zweite Semesterindex für die Verbindung zwischen FPO und Modul ist fast analog aufgebaut. Die Relation heißt r\_semidx\_mod, besitzt als Primärschlüssel idr\_semidx\_mod und zwei weitere Attribute namens Jahr und xS, die nicht Null sein dürfen.

```
r_semidx_mod(idr_semidx_mod:integer, jahr:integer, xS:string)  
(jahr, xS) NOT NULL
```

Bei den Schlüsselkandidaten wird keine Garantie gegeben, dass sie sicher als Primärschlüssel einsetzbar sind. Dieses zu überprüfen, würde viel Zeit in Anspruch nehmen, und im schlimmsten Fall bei der Befüllung der Basisdatenbank auffallen. Aus diesem Grund wurden für alle Entities Surrogate eingefügt, um dieses Problem zu vermeiden.

## 6.3 Datenbeschaffung

In Kapitel Datequellen 4 wurden die folgenden zwei Quellen beschrieben: die Jahrgangsinformationen aus der Studentenkazlei und die Prüfungsdaten von Mein Campus. Nach den ersten Gesprächen mit dem Leiter von Campus IT wurde angeboten, auch die bei Campus IT vorliegenden Jahrgangsinformationen zur Verfügung stellen. Dies

wurde angenommen, um eine automatisierte und jederzeit abrufbare Datenquelle zu haben, was die Studentenkazlei nicht bieten konnte.

Die Frage war nun, wie man die Daten aus den Systemen von Campus IT auf den Rechner, auf dem das Data Warehouse läuft, bekommt. Mein Campus bot an, dass einer seiner Entwickler eine Webschnittstelle programmiert, die sowohl die Prüfungsinformationen als auch die Jahrgangsinformationen liefern könnte. Dieses Angebot wurde angenommen.

### 6.3.1 Campus IT Webservice

Nach der Evaluation vom Serviceangebot von Mein Campus war bekannt, welche Daten Mein Campus verwalten muss, um Prüfungen zu verbuchen, eine Spezifikation einer Java Schnittstelle dem Mein Campus-Team übergeben. Einige dieser Datensätze sind interessant, dass diese Informationen auch für das Data Warehouse interessant sind. Um die Noten für Teilleistungen zu verbuchen, muss im System die Information verfügbar sein, um welche Teilleistung es sich handelt. Somit müssen diese Informationen im System von Mein Campus auch zu finden sein.

Ein Modul besteht aus einer oder mehreren Teilleistungen. In den FPOs wird verlangt, dass Module bestanden werden müssen. Aus diesem Grund muss Mein Campus die Module auch abbilden, weil Mein Campus wissen muss, welche Teilleistung zu welchem Modul gehört. Diese Informationen sind für das Data Warehouse sehr interessant, und dieser Fakt wurde als eigene Anfrage an den Webservice spezifiziert. Die Problematik bei der Sache ist, dass Module und Teilleistungen nicht zusammen in die Basisdatenbank eingepflegt werden können. Somit müssten diese Informationen manuell von einer Person eingegeben werden. Diese Methode würde viel Zeit in Anspruch nehmen. Deswegen wurden zwei Datentransfer-Objekte spezifiziert, eins für Module und eins für Teilleistungen.

Das Datentransfer-Objekt für den Austausch der Module besteht aus dem Namen des Moduls, dessen ECTS-Punkten, dem Anfragejahr und xS <sup>7</sup>. Weiter soll die interne Modulnummer von Mein Campus verwendet werden, um sie im Data Warehouse als Primärschlüssel einzusetzen. Der Vorteil dieser Vorgehensweise wäre, dass keine weitere Transformation der Daten für den Primärschlüssel notwendig wären. Mein Campus kann allerdings keine Auskunft geben, ob ein Modul ein Pseudo-Modul ist. Ein Pseudo-Modul wurde eingeführt, um die alten Studiengänge abzubilden. Da diese Studiengänge aber nicht mehr betrachtet werden, ist diese Information nicht wichtig. Der Webservice gibt noch die Information zurück, in welchem Studiengang das Modul angeboten wird.

---

<sup>7</sup>x steht für W(inter) oder S(Sommer)

`Modul_Daten_Transfer_Objekt(Name, Nr, Anfragejahr, AnfragesS,  
Studiengang, ECTS)`

Für das Teilleistung Data Transferobjekt sieht der Datencontainer ziemlich ähnlich aus wie für die Module. Für das Data Warehouse wird der Name und das dazugehörige Modul benötigt. Dazu sollte es wieder die von Mein Campus verwendete Teilleistungsnummer geben, um sie im Data Warehouse als Primärschlüssel zu verbuchen.

`Teilleistung_Daten_Transfer_Objekt(Name, Nr, Modulname, ECTS)`

Die Separierung zwischen Modulen und Teilleistungen ist auf Seiten von Mein Campus nicht möglich bzw. nur mit einem großen Aufwand realisierbar. Aus diesem Grund gibt es eine Anfrage, die Module und Teilleistung zurückgibt.

Die Anfrage, um die Module und Teilleistungen zu bekommen, heißt `getModules(Studiengang, Abschluss)` und besitzt zwei Parameter, die den Studiengang und Studienfach als Integerwerte spezifizieren. Die Tabelle für den Studiengang bzw. der Abschluss in Integerwerten befindet sich im Anhang C. Mit den zwei Parametern müssen dann systematisch die Information für alle Studiengänge mit den unterschiedlichen Abschlüssen beim Webservice erfragt werden. Nach der Anfrage wird vom Webservice eine lange Liste in XML<sup>8</sup>-ähnlichen Format zurückgeliefert. Es gibt Anfangs- und Endzeichen, die beschreiben, um welchen Inhalt es sich handelt. Der Listeninhalt ist wie folgt aufgebaut: Als erstes werden die Informationen zu einem Modul ausgegeben, beispielsweise der Modulname. Direkt danach folgen die Teilleistungen mit ihren Attributen, die zu dem Modul gehören.

Allerdings liefert diese Anfrage an den Webservice bis jetzt noch Fehler. Zwar werden die Module und Teilleistungen ausgegeben, allerdings stimmen die Modulnummern und die Teilleistungsnummern nicht überein. Eine Teilleistungsnummer sieht bei Mein Campus folgendermaßen aus: Der erste Teil der Teilleistungsnummer ist die Modulnummer. Als Suffix wird eine Ziffer angehängt, die die einzelnen Teilleistungen durchnummeriert. Eine Darstellung der Modul und Teilleistungsnummer am Beispiel des Studiengangs Informatik mit Abschluss Bachelor ist im Internet zu finden <sup>9</sup>. Ein weiteres Problem ist, das seltsame Anomalien auftreten. So taucht bei den Aufrufparametern Studiengang Informatik und Abschluss Bachelor die Bachelorarbeit zweimal mit unterschiedlichen Modulnummern auf.

Eine weitere Schnittstelle wurde spezifiziert, um die Informationen der Studiengänge zu erhalten. Für die Datentransfer-Objekte wurden folgende Inhalte spezifiziert: Im Datentransfer-Objekt soll sich der Name des Studiengangs befinden, sein Kürzel und die von Mein Campus vergebene Studiengangidentifikationsnummer. Warum wieder die Studiengangidentifikationsnummer von Mein Campus verwendet werden

---

<sup>8</sup>Extensible Markup Language

<sup>9</sup>[www.cit.uni-erlangen.de/fileadmin/CIT/ModelleBA/20070823\\_BSc\\_Informatik.png](http://www.cit.uni-erlangen.de/fileadmin/CIT/ModelleBA/20070823_BSc_Informatik.png)[EN09e]

soll, hat wieder den Grund, eine Transformation bezüglich des Primärschlüssels von Studiengang zu sparen. Hier gab es keine Einwände vom Campus IT-Team.

`Studiengang_Daten_Transfer_Objekt(Name, Nummer, Kürzel)`

Der Aufruf an den Webservice sieht wie folgt aus: `getAllCourses()`. Bei dem Aufruf müssen keine Parameter übergeben werden und man erhält alle Studiengänge der Universität Erlangen-Nürnberg. Die Ausgabe erfolgt wieder in einem XML-ähnlichen Format.

Nun werden noch die Jahrgangsinformationen benötigt. Hierbei ist zu beachten, dass es zwei verschiedene Zeitabhängigkeiten gibt. Die erste ist das Fachsemester der Studierenden. Dies bedeutet, wie viele Semester ein Studierender in einem Studiengang bereits absolviert hat. Das erste Semester wird auch mit Eins beschrieben. Das Fachsemester ist allerdings wieder relativ. Zurzeit ist die Anzahl der Studierenden mit Fachsemester fünf im Studiengang Informatik eine andere, als wenn die Anfrage vor einem Jahr erstellt wurde. Somit muss die Abfragejahr und AbfragexS bei der Anfrage mit verarbeitet werden. Am Ende kommen zu dem Abfragejahr, AbfragexS und Fachsemester noch der Studiengang und die Anzahl der Studierenden im Fachsemester und Studiengang der gestellten Anfrage. Für die Studierendenzahlen greift Mein Campus auf die ZUV-Datenbank zu, die dann auch die Informationen der alten Studiengänge wie Magister und Diplom, beinhaltet. Somit muss noch der Abschluss zum Studiengang im Datentransfer-Objekt definiert werden.

`Jahrgang_Daten_Transfer_Objekt(Abfragejahr, AbfragexS, Studiengang, Abschluss, Fachsemester, Anzahl)`

Der Aufruf für dieses Datentransfer-Objekt lautet `getStudyYearInformation(Abfragejahr, AbfragexS, Studiengang, Abschlussart)`. Warum der Aufruf diesen Aufrufparameter hat, wurde bereits oben mit der Zeitrelativität des Fachsemesters beschrieben. Der Webservice antwortet darauf mit einer Liste in XML-ähnlichem Format.

Es ist mühsam, für jeden Studiengang und für jeden Abschluss die Anfrage manuell zu stellen. Deswegen ist eine weitere Anfrage formuliert worden, um alle Studiengänge mit allen deren Abschlüssen auf einmal zu bekommen. Das Datentransfer-Objekt sieht fast gleich aus wie für den einzelnen Fall, nur wird hier eine Liste von dem Jahrgang Datentransfer-Objekt zurückgegeben. Campus IT konnte diese Anfrage ebenfalls umsetzen.

Um alle Studiengänge abzufragen, gibt es einen neuen Aufruf mit Namen `getStudyYearInformationFromAll(Abfragejahr, AbfragexS)`. Die Parameter der Anfrage haben den gleichen Grund wie bei `getStudyYearInformation`. Das Fachsemester ist immer relativ von der Abfragezeit. Die Abfragezeit wird durch das Abfragejahr und das AbfragexS spezifiziert. Die Rückantwort vom Webservice ist wieder in XML-ähnlicher Form. Hier werden jetzt systematisch alle Studiengänge mit ihren Fachsemestern und mit dem

jeweiligen Abschluss ausgeben. Sind alle Fachsemester eines Studiengangs ausgegeben, fängt ein neuer Studiengang an.

Nun fehlt nur noch die Spezifikation der Prüfungsleistung, die für die Analysen benötigt wird. Dafür wird ein neues Datentransfer-Objekt erstellt. Da bekannt ist, welches Modul gewünscht ist, braucht man erst die Modulnummer des Moduls, welches abgefragt werden soll. Hier gibt es ein Problem. Bei der Angabe muss die Modulnummer genommen werden, die Mein Campus verwendet. Woher ist diese bekannt? Zu finden ist die Nummer in der Relation Modul in der Datenbank. Dafür muss die Modultabelle geladen sein. In dieser Relation werden alle Module mit den Modulnummern von Campus IT gespeichert. Dies ist der Grund, warum die Modulnummer von Campus IT auch im zu entwickelnden Data Warehouse verwendet werden soll. Der Zeitpunkt der Anfrage muss noch spezifiziert werden, aus diesem Grund liefert das Datentransfer-Objekt auch das Abfragejahr und das AbfragexS zurück. Es werden aber noch die Teilleistungen der Module benötigt. Somit gibt die Anfrage auch noch die Teilleistungen eines Moduls mit aus. Als letztes werden die drei Werte bestanden, nicht bestanden und endgültig nicht bestanden ausgegeben. Eine weitere Ausgabe ist der Datenpool. Der Pool gibt an, in welchem Datenbankschema die Daten bei Mein Campus liegen. Sie sind nach Fakultäten getrennt.

```
CIT_Daten_Transfer_Objekt(Modulnummer, Abfragejahr, AbfragexS,  
    Teilleistungen Daten Transfer Objekt(Teilleistungsnummer,  
    Fachsemester, Anzahl bestanden, Anzahl nicht bestanden,  
    Anzahl endgültig nicht bestanden, Pool))
```

Die Anfrage für die Prüfungsinformationen an den Webservice lautet `getModuleExaminationEleByYear(Abfragejahr, AbfragexS, Modulnummer, Pool)`. Die Attribute dieser Anfrage bestehen u.a. aus dem Zeitpunkt, an dem die Anfrage gestellt wurde. Hier hat man die Abhängigkeit mit den Prüfungen. Es werden Prüfungen jedes Semester oder alle zwei Semester angeboten. Weiter muss die Modulnummer angegeben werden, von der man die Information haben will. Hier ist auch wieder die Modulnummer von Mein Campus gefragt. Diese Nummer kann man in der Relation Modul nachschauen. Zum Schluss muss man den Datenpool von mein Campus angeben. Für die jetzige Anfrage steht immer "Tech" für die Technische Fakultät drin. Wenn das Data Warehouse für andere Studiengänge erweitert wird, muss man die anderen Datenpools verwenden.<sup>10</sup>

Die Ausgabe des Webservice ist auch hier in XML-ähnlicher Form. Als Kopf der Anfrage werden die Modulnummer, das Abfragejahr und AbfragexS ausgegeben. Darunter werden die Teilleistungen ausgegeben mit den drei Prüfungsinformationen. Zusätzlich wird noch der abgefragte Datenpool ausgegeben.

---

<sup>10</sup>[http://www.cit.uni-erlangen.de/dokumente-und-materialien/bachelorstudiengaenge/\[EN\]](http://www.cit.uni-erlangen.de/dokumente-und-materialien/bachelorstudiengaenge/[EN])

Wie bei den Jahrgangsinformationen wäre es von Vorteil, wenn die Informationen nicht Modul für Modul anfragen und an den Webservice schicken müsste. Besonders beim ersten Laden der Basisdatenbank des Data Warehouse wäre es vorteilhaft, alle Informationen auf einmal ins System zu bekommen. Deswegen wurde auch an das CIT<sup>11</sup> Datentransfer-Objekt eine Sammelanfrage gestellt, die vom Campus IT-Team leider nicht ganz realisiert werden konnte. Ihr Datenbankschema erlaubt es nicht, alle Module auszulesen. Als Gegenangebot folgte, dass jetzt alle Module eines Studiengangs mit Abschluss und deren Module, ausgegeben werden. Die Ausgabe wird jetzt eine Liste von CIT-Datentransfer-Objekten sein.

Die Anfrage für die Sammelanfrage sieht wie folgt aus `getStudyInformationFromAll(Studiengang, Abschluss, Abfragejahr, AbfragexS, Pool)`; Studiengang und Abschluss werden angegeben, um die Module zu erhalten, die der angegebene Studiengang besitzt. Die Begründung, warum Abfragejahr, AbfragexS und der Datenpool drinstehen, ist die gleiche wie bei `getModuleExaminationEleByYear`.

Der Rückgabewert des Webservices sieht ähnlich aus wie bei `getModuleExaminationEleByYear`. Die Ausgabestruktur ist gleich, nur werden jetzt Modul für Modul eines Studiengangs ausgegeben. Die Darstellung ist in einer XML-ähnlichen Form gehalten.

### **Bewertung**

Die Lösung ist in Bezug auf die Verfügbarkeit sehr gut. Wenn der Webservice in einer finalen Version vorliegt, läuft er automatisch ab, außer es gibt Wartungsarbeiten auf den Servern. Allerdings variiert die Antwortzeit extrem. Wenn die gesamten Jahrgangs- oder die Prüfungsinformationen gesucht werden, kann dies schon einige Zeit dauern. Verschlimmert wird dies noch, wenn viele Nutzer, sowohl Studierende als auch Dozenten, auf dem System arbeiten. Dies alles hatte zur Folge, dass der Timeout für die Verbindung zwischen Webservice und Client auf zehn Minuten erhöht werden musste. Die Rückgabe in XML-ähnlichem Format erzeugt einmalig zusätzliche Arbeit, weil die Informationen extra in ein CSV umgeschrieben werden. Dafür muss ein Übersetzer programmiert werden, dann läuft auch dies automatisch ab. Die Aufteilung zwischen Modulen und Teilleistungen ist nicht gerade vorteilhaft für das entwickelte Basisdatenbankschema, weil Module und Teilleistungen getrennt sind. Es werden vom Webservice viele Informationen zusammen geliefert, die getrennt einfacher zu bewältigen gewesen wären. Dann wären Fragen feingranularer und die Systeme von Mein Campus wären nicht so stark belastet worden. Um Informationen von Teilleistungen zu erhalten, muss man immer mit Modulen arbeiten. Wenn jetzt viele verschiedene Teilleistungsinformationen benötigt werden, müssen viele Module angefasst werden, von denen alle Teilleistungen ausgegeben werden. Wenn man sich

---

<sup>11</sup>Campus IT

allerdings die Alternative anschaut, dass diese Informationen nur einmal im Semester und als Papier bzw. in PDF-Form zu erhalten gewesen wären, wäre dies weitaus mehr Arbeit gewesen als die Lösung vom Campus IT-Team.

### 6.3.2 Selbstpflege

Betrachtet man das E/R-Diagramm 5.9 noch einmal, so sieht man jetzt, dass viele Informationen über den Webservice geliefert werden. Darunter fallen Studiengänge, Module, Teilleistungen, Jahrgänge, Jahrgangszahlen und Prüfungsinformationen. Es gibt allerdings weitaus mehr Informationen, die in die Basisdatenbank müssen und die nicht automatisch geladen werden können.

Betrachtet man den linken Rand des E/R-Diagramms 5.9, dann sieht man die Entities Fakultät, Department und Lehrstuhl. Sie gehören noch zu den Entities, die weniger Arbeit machen. Die Universitätsstruktur ändert sich nicht so häufig, und die Fakultätenanzahl ist sehr überschaubar. Der Erstellungsaufwand wird somit gering sein. Ähnliches gilt für die Departments. Zwar gibt es schon deutlich mehr Departments als Fakultäten, aber ihre Anzahl ist momentan unter 30. Somit sind der Erstellungsaufwand und der Wartungsaufwand zwar höher als bei der Fakultät, aber noch angemessen. Für Lehrstühle gibt es eine Einschränkung. Das momentane Interesse an Lehrstühlen ist nur an den Lehrstühlen der Informatik vorhanden, weil die Analyse die Informationen nur für die Lehrveranstaltungen braucht, die auch vom Departement Informatik angeboten werden. Da die Anzahl der Informatik-Lehrstühle konstant ist und sich nicht semesterweise ändert, ist der Erstellungs- und Pflegeaufwand gering.

Eben wurden schon die Lehrveranstaltungen erwähnt. Hier besteht mehr Arbeitsaufwand. Das Department Informatik bietet über hundert verschiedene Lehrveranstaltungen an; seien es Vorlesungen, Übungen, Praktika oder Repetitorien. Für die manuelle Erstellung ist dies mit einem zeitlichen Aufwand verbunden. Diese Informationen stehen alle im Univis und müssen herausgesucht und eingetragen werden. Das größere Problem ist der Pflegeaufwand. Die Lehrveranstaltungen verändern sich oft. Es kann auch sein, dass sich beispielsweise nur der Name ändert. Aber selbst diese kleine Änderung muss neu eingepflegt werden. Zusätzlich ist darauf zu achten, dass eine Lehrveranstaltung nicht doppelt eingetragen wird. Hier taucht ein Problem auf, dass die Primärschlüssel Surrogate sind. Die Basisdatenbank würde hier nicht merken, wenn eine Lehrveranstaltung mit gleichen Namen, aber mit unterschiedlichem idmodul eingetragen wird. Anders herum kann es sein, dass sich der Inhalt einer Lehrveranstaltung geändert hat, aber der Name gleich geblieben ist. Dies ist somit eintragbar. Des Weiteren muss immer die Information vorliegen, zu welcher Teilleistung die Lehrveranstaltung gehört. Damit kann man dann in den Teilleistungen die idteilleistung

heraussuchen und eintragen. Weiterhin gibt es auch einen Verantwortlichen für die Lehrveranstaltung, der aus der Relation Dozent gesucht werden muss.

Die gerade angesprochenen Dozenten sind auch nicht automatisch einpflegbar. Da meine Arbeit nur auf die Lehrveranstaltungen des Departements Informatik beschränkt ist, müssen nur die Mitarbeiter der Informatik und die externen Dozenten eingepflegt werden. Diese Informationen befinden sich im Univis mit Raumnummer und Telefonnummer. Der Pflegeaufwand ist hierbei wieder höher. Es kann sein, dass Dozenten nur für eine Lehrveranstaltung angestellt sind, und somit, wenn die Vorlesung wieder angeboten wird, ein neuer Dozent eingegeben werden muss. Auch ist die Fluktuation bei den Mitarbeitern sehr groß. Der Zeitraum, in dem Doktoranden angestellt sind, ist meist kurz und endlich. Diese Problematik gilt auch für die Verantwortlichen von den Modulen.

Es gibt auch noch ein weiteres Informationsdefizit bezüglich des Dozenten und seiner Position als Verantwortlicher für einen Studiengang. In den FPOen ist nicht vermerkt, wer für die Entwicklung oder Kontrolle von Studiengängen verantwortlich ist. Um diese Information zu erhalten, muss bei den Departements, die die Studiengänge anbieten, nachgefragt werden. Dies kann einiges an Zeit und Nerven kosten, zumal meist mehrere Personen für unterschiedliche Bereiche verantwortlich sind.

Die Entity Semester ist dagegen sehr angenehm. Die Werte für Semester kann man leicht mit einem Tabellenkalkulationsprogramm erzeugen und dann in Semester einlesen. Auch ist das Updaten dieser Relation sehr leicht. Der Pflegeaufwand ist genau so wie der Erstellungsaufwand, da nur die letzten Werte betrachtet werden müssen, die in Semester verbucht sind. Diese müssen überprüft und eventuell korrigiert werden.

Den größten Aufwand bereiten die Relationen um FPO, Modulgruppe, Modul und die Relationen zwischen den Entities. Die FPOen liegen auf der Webseite des Prüfungsamtes. Die Gültigkeit von Zeitpunkt x bis Zeitpunkt y oder die Zuordnung zum Studiengang oder den Beginn des xS muss aus der FPO heraus gelesen werden. Eine FPO hat kein einheitliches Format, wodurch Zeit gespart werden könnte, wenn man nur diese Stellen lesen müsste. Somit ist hier viel Zeit zu veranschlagen, denn die Universität Erlangen-Nürnberg hat über 100 FPOen.

Modulgruppen haben das gleiche Problem. Wie die Modulgruppen heißen und welche Module sie beinhalten, wird in der FPO niedergeschrieben. Hier heißt es wieder die FPOen lesen, um die Modulgruppennamen zu erfahren, wenn es welche gibt. Meist bieten die Studienführer der einzelnen Studiengänge noch mehr Informationen. Allerdings sind die Studienführer auf den Webseiten der Departements verteilt, die dann extra gesucht werden müssen. Aus welchen Modulen eine Modulgruppe besteht, ist aber im Studienführer zu finden. Modulgruppen können bzw. müssen ab einem



bestimmten Semester belegt werden oder können nur in bestimmten Semestern begonnen werden. Diese Informationen befinden sich auch in den FPOen.

Für die definierte Relation Modul bekommen liegen alle Informationen, die die Relation Modul betreffen. Allerdings fehlen die Informationen, von welchen Studiengängen sie belegt werden müssen oder können und in welchem Semester die Module gehört werden. Diese Informationen stehen dann wieder in der FPO und im Studienführer.

Die FPOen stehen im Bereich Modul, Modulgruppen und Jahrgang stark im Vordergrund und aus ihnen müssen sehr viele Informationen gezogen werden. Für das genaue Lesen der FPOs ist der Zeitaufwand sehr hoch. Dies gilt besonders für das erste Einpflegen der Daten, weil die Universität Erlangen-Nürnberg mehr als 100 FPOs besitzt. Ein weiterer Pflegeaufwand der vorhandenen FPOs ist nicht erforderlich, weil es keine Erneuerungen gibt. Allerdings kommt es vor, dass neue FPOs verabschiedet werden und so alle Daten für die neuen FPOs gefunden werden müssen.

Die Relation Evaluation bleibt leer, weil die Datenquelle nicht erschlossen ist.

Schaut man sich die Informationen an, die Mein Campus erhalten liefert, wird man feststellen, dass nicht alle Informationen vorliegen, die spezifiziert wurden. So fehlt bei Modul beispielsweise die Informationen, ob es ein GOP-Modul ist oder ein Pseudomodul. Die Information für ein Pseudomodul muss nicht beschafft werden, weil das System nur noch für die Bachelor-Studierenden entwickelt ist. Die GOP-Informationen muss man beim Studieren der FPO mit herausziehen.

Beim Studiengang fehlt auch noch die Information, zu welchem Department ein Studiengang gehört. Diese Information erhält man auch durch die FPOen, weil der Name des anbietenden Departements enthalten ist.

Aus Erfahrung kann man nur die Relation Schlüsselqualifikation befüllen. Da Schlüsselqualifikationen nicht genau in FPOs spezifiziert sind, muss der Exportbeauftragte entscheiden, welcher Studiengang welches Modul als Schlüsselqualifikation hören kann.

### **Vorgehensweise**

#### **Erstes Laden**

Semester sollte als Erstes geladen werden, weil man Rückantworten vom Webservice auf die Fremdschlüssel von Semester Bezug nehmen. Jetzt sollten alle Informationen, die der Webservice liefert, in die Basisdatenbank geladen werden. Sie sollten ohne Abhängigkeiten von anderen Relationen einfügbar sein. Danach sollte man die Universitätsstruktur befüllen, d.h., dass Fakultät, Department und Lehrstuhl in dieser Reihenfolge geladen werden sollten, um alle Fremdschlüssel gleich laden zu können. Nun sollte man das Univis konsultieren und damit die Lehrveranstaltungen und Do-

zenten füllen. Zum Schluss kommen dann die gelesenen FPOen, um die Informationen für FPO, Modulgruppe und Modul zu erhalten. Dabei sollte nicht außer Acht gelassen werden, die Zusatzinformationen, die noch bei Modul und in den Relationen benötigt werden, mitzunehmen. Ist dies alles erledigt, kann man bei den Departments wegen des Verantwortlichen für den Studiengang nachfragen.

### **Sukzessives Updaten**

Bei den meisten Relationen muss man sich umhören, falls Veränderungen eintreten. Gibt es eine neue FPO, so wissen es der Fakultätsrat und der Senat der Universität. Bei Änderungen der Verantwortlichen liegt diese Information nur beim entsprechenden Lehrstuhl, der die Lehrveranstaltung anbietet, vor. Aus diesem Grund ist es angebracht, vor jedem Semester die Basisdatenbank auf den neuesten Stand zu bringen. Die Durchführung der Updates funktioniert bis auf die Daten von Mein Campus alles manuell. Die Mein Camus-Informationen sollten einmal im Semester abgerufen werden. Möglichst in der Mitte, weil dann Prüfungsdaten komplett vorliegen sollten. Wenn es notwendig ist, kann man auch zwischendrin die Daten von Mein Campus auf den neuesten Stand bringen, weil die Daten automatisch geladen werden können. Wenn die Datenbank geupdatet wird, sollte es im Normalfall keine Rolle spielen, welche Relation zuerst auf den neuesten Stand gebracht wird, weil normalerweise die Module schon vorliegen. Um sicherzugehen, ist zu empfehlen, dass die Reihenfolge vom ersten Laden eingehalten wird.

## **6.4 Data Warehouse System**

Nachdem die Daten in der Basisdatenbank vorliegen, müssen die Daten jetzt noch in die Tabellen des Data Warehouse übertragen werden. Es muss jetzt entschieden werden, welche Art von Datenbank unter dem Data Warehouse liegen soll. Die Möglichkeiten sind, die Daten in einer relationalen Datenbank zu speichern oder in einer multidimensionalen. Die Vorteile einer multidimensionalen Datenbank sind, dass der Analyseprozess sehr effektiv auf die Daten zugreifen kann, weil die Daten in aggregierten Datenwürfel vorliegt. Die relationale Datenbanklösung ist für die Speicherung einer hohen Anzahl von Datensätzen das idealste. Eine weitere Lösung wäre eine Mischform aus beiden zu nehmen. Die Auswahl wurde durch das einzusetzende Programm (JasperSoft 7) erledigt. Das Data Warehouse wird auf einer relationalen Datenbanksystem basieren.

Für das Data Warehouse werden dann Relationen benötigt, die sich aus dem multidimensionalen Schema ableiten lassen. Im Grunde sind die Relationen vom Data Warehouse ähnlich den Tabellen in der Basisdatenbank aufgebaut. Der größte Unterschied ist, dass es keine Fremdschlüsselbeziehungen zwischen Relationen im Data Warehouse mehr gibt. Die Daten werden eins zu eins aus der Basisdatenbank geladen

und somit wird die Konsistenz behalten. Weiter wird auf das Data Warehouse nur lesend zugegriffen, d.h. der Datenbestand im Data Warehouse wird nicht verändert.

Trotzdem gibt es neue Fremdschlüsselbeziehungen. Um die Dimensionshierarchien aufzubauen, werden neue Fremdschlüssel benötigt. Nach dem multidimensionalen Datenschema aus Kapitel 5.4 existiert eine Dimension, die aus mehreren Relationen aus der Basisdatenbank besteht. Umgesetzt wird hier das SSnowflakeSchema, d.h., dass die Tabellen einzeln betrachtet werden und nicht in eine neue riesige Relation immigriert werden. Der Grund, warum snowflake eingesetzt wird, wird in Kapitel 7 beschrieben. Um am Ende diese Dimension richtig mit Verbundoperationen zusammensetzen zu können, wird eine Fremdschlüsselbeziehung benötigt. Etwas weiter unten wird beschrieben, dass die Abhängigkeiten eins zu eins aus der Basisdatenbank übernommen werden. Dies ist auch richtig, allerdings aus technischen Gründen, die in Kapitel 7 beschrieben werden.

Die Relationen für die Dimensionen:

```
dimSemester(idsemester:integer, jahr:integer, xS:string)
```

```
dimVeranstaltung(idveranstaltung:integer, name:string, typ:string,  
  iddozent:integer, idteilleistung:integer)
```

```
dimDozent(iddozent:integer, vorname:string, nachname:string,  
  telefonnr:integer, raumnr:integer, idlehrstuhl:integer)
```

Dies sind die Relationen für die Dimensionen, die jeweils eine komplette Dimension abdecken. Bei der Dimension dimOrg, die die Organisationsstruktur der Universität abbildet sieht es anders aus.

```
dimFakultaet(idfakultaet:integer, name:string)
```

```
dimDepartment(iddepartment:integer, name:string,  
  idfakultaet[dimFakultaet]:integer)
```

Ab hier gibt es parallel Hirachien.

1. dimLehrstuhl(idlehrstuhl:integer, name:string,  
 iddepartment[dimDepartment]:integer)
2. dimModul(idmodul:integer, name:string, ects:integer, gop:boolean,  
 pseudomodul:boolean, xS:string, iddozent:integer,  
 iddepartment[dimDepartment]:integer)  
 dimTeilleistung(idteilleistung:integer, idmodul[dimModul]:integer,  
 name:string, ects:integer)

```
3. dimStudienfach(idstudienfach, name:string, kuerzel:string,
  iddepartment[dimDepartment])
  dimfpo(fponame, idsemester:integer, idsemester:integer, xS:string,
  url:string, fponame:string,
  idstudienfach[dimStudienfach]:integer)
  modulgruppe(idmodulgruppe:integer, name:string, ects:integer,
  fponame[dimfpo]:string, idr_semidx_modgr:integer)
```

Eins bis drei stellen die unterschiedlichen parallele Hierarchien dar, die jeweils von ihrer höheren Klassifikationsstufe dimDepartment ausgehen.

Nachdem alle Dimensionen spezifiziert sind, folgen die einzelnen Datenwürfel. Hier wird schnell eine Besonderheit dieses Modells 5.11 sichtbar: Es gibt wenige Faktentabellen, die ein Faktenattribut besitzen.

Der erste Datenwürfel wird von den Dimensionen dimOrg und dimSemester aufgespannt. Die Faktenrelation heißt factCIT und ist für die Analyse der Studierenden da, die an einer Teilleistung in einem Semester teilgenommen haben. Als Faktenattribut wird noch angegeben, wie viele Studierende bestanden (bestanden), nicht bestanden (nochdabei) haben und endgültig nicht bestanden (durchgefallen) haben.

```
factCIT(idSemester[dimSemester]:integer, idTeilleistung[dimOrg]:integer,
  bestanden:integer, nochdabei:integer,
  durchgefallen:integer)
```

Die zweite Faktenrelation, die Faktenattribute besitzt, ist die Relation factEVA. Diese Relation wird leer bleiben, weil es keine Datenquelle gibt, die die Anzahl der abgegeben Evaluationen einer Lehrveranstaltung wieder gibt. Es könnte allerdings für die Zukunft interessant sein. Als Faktenattribut agiert die Anzahl der abgegeben Evaluationen. Der Datenwürfel von factEVA wird durch die Dimension dimOrg und dimSemester aufgespannt.

```
factEVA(idSemester[dimSemester]:integer, idLehrveranstaltung[dimOrg]:integer,
  anzahl:integer)
```

Die dritte Relation, die ein Faktenattribut enthält, ist die Faktenrelation factVerSF. Diese Relation kümmert sich um den Verantwortlichen für die Studiengänge bzw. Studienfächer. Die Relation besitzt als Faktenattribut die Rolle, welche der Verantwortliche einnimmt. Der Datenwürfel factVerSF wird von den Dimensionen dimDozent und dimOrg aufgespannt.

```
factVerSF(idDozent[dimDozent]:integer, idStudienfach[dimOrg]:integer,
  rolle:string)
```

Ein weitere wichtige Relation für die Analyse ist die Faktenrelation factJG. Diese Faktenrelation betitelt die Jahrgangsinformationen, die für die Analyse später wichtig sein wird. Als Faktenattribut besitzt sie die Anzahl der Studierenden, die in einem Semester eines bestimmten Studiengangs sind. dimOrg und dimSemester spannen diesen Datenwürfel auf.

```
factJG(idSemester[dimSemester]:integer, idStudiengach[dimOrg:integer],
anzahl:integer)
```

Die folgenden Faktenrelationen besitzen keine Faktenattribute mehr.

Die nächste Relation heißt factFPOableit. Sie beschreibt, welche FPO von einer anderen abgeleitet ist. Aufgespannt wird der Datenwürfel nur von der Dimension dimOrg.

```
factFPOableit(fponame1[dimOrg]:string, fponame2[dimOrg]:string)
```

Für die Analyse der Verbindung zwischen der FPO und den Modulen gibt es die Faktenrelation factFPOMod. Dieser Datenwürfel wird auch nur durch die Dimension dimOrg aufgespannt.

```
factFPOMod(idModul[dimOrg:integer], fponame[dimOrg]:string)
```

Wie lange eine FPO gültig ist, beschreibt die Faktenrelation factFPOGuelte, die von den Dimensionen dimSemester und FPO aufgespannt wird.

```
factFPOGuelte(idSemester[dimSemester]:integer, fponame[dimOrg]:string)
```

Die Schlüsselqualifikationen werden durch die factSQ abgebildet. Sie beschreibt den Zusammenhang zwischen Studiengang und Modulen. Der Datenwürfel für factSQ wird von einer Dimension, nämlich dimOrg, aufgespannt.

Welche Module sich in einer Modulgruppe befinden, kann durch den Datenwürfel factModModgr ermittelt werden. factModModgr wird durch die Dimension dimOrg aufgespannt.

```
factModModgr(idModul[dimOrg]:integer, idModulgruppe[dimOrg]:integer)
```

Es folgen nun noch die Verantwortlichen für Studiengang bzw. Studienfach, Modul und Teilleistung. Die Relationen heißen factVerSF (Verantwortlicher für Studiengang bzw. Studienfach), factVerMod (Verantwortlicher für Modul) und factVerTeil (Verantwortlicher für Teilleistung). Die drei Datenwürfel werden jeweils durch die Dimensionen dimDozent und dimOrg aufgespannt.

```
factVerSF(idDozent[dimDozent]:integer, idStudienfach[dimOrg])
```

```
factVerMod(idDozent[dimDozent]:integer, idModul[dimOrg])
```

`factVerTeil(idDozent[dimDozent]:integer, idTeilleistung[dimOrg]:integer)`

### 6.4.1 Laden des Data Warehouse

Das Laden der Tabellen erfolgt eins zu eins aus der Basisdatenbank in das Data Warehouse System. Durch Nutzung des snowflake-Schemas, befinden sich keine zusammengesetzten Tabellen im Data Warehouse System. Somit kann jede einzelne Tabelle direkt von der Basis Datenbank ins Data Warehouse geladen werden. Ein Vorteil bei der Verwendung einer Basisdatenbank ist, dass nicht immer alle Datensätze ins Data Warehouse geladen werden müssen. Bei manchen Tabellen macht dies mehr oder weniger Sinn. Veranschaulichen kann man dies am Beispiel der Relationen Modul, Teilleistung und Veranstaltung. Vom Webservice von Mein Campus werden alle Informationen über Module und Teilleistungen von der ganzen Universität geliefert. Momentan besteht das Interesse nur an den Modulen, die vom Departement Informatik angeboten werden. So können für die Analyse nur die Module und Teilleistungen, die vom Departement Informatik angeboten werden, geladen werden. Mit weniger Inhalt erhöht sich die Geschwindigkeit, weil weniger Daten ausgewertet werden müssen bzw. eine geringere Anzahl an Datensätzen, die für die Analyse benötigt werden, ermittelt werden muss.

## 6.5 Analytischer Prozess

Den analytischen Prozess muss man in die zwei Hauptaufgaben unterteilen. Die Bestimmung der Studierendenzahlen für Lehrveranstaltungen ist eine Analyse, die nicht nur durch die Qualität der vorhandenen Daten bestimmt, sondern vom Verhalten der Studierenden selbst beeinflusst wird. Das Verhalten der Studierenden einzubeziehen, wird durch Zurückblicken auf vergangene Jahre versucht, aber es gibt immer eine Unschärfe dabei. Anders sieht es beim zweiten Ziel aus, also bei der Beratung der Studierenden und bei der Berichterstattung an's Department. Hier müssen sorgfältig die FPOen angeschaut werden, um die Daten richtig in die Basisdatenbank und dann weiter ins Data Warehouse zu bekommen.

Beim analytischen Prozess in Bezug auf Studierendenzahlen muss zwei Fakten Genüge geleistet werden. Auf der einen Seite gibt es die genauen Studierendenzahlen, die Jahrgangsinformationen und die Prüfungsinformationen, und auf der anderen Seite steht das Verhalten der Studierenden. Das Verhalten von den Studenten und Studentinnen vorzusagen, ist nicht einfach. In Kapitel 1.1 wurde dies schon ansatzweise thematisiert. Somit werden die Zahlen immer einer gewissen Schwankung unterliegen.

### **6.5.1 Studierendenberatung bzgl. Lehrveranstaltung in der Informatik**

Der einfache und auch genaue Weg ist die Auswahl der Studiengänge und die dazu möglichen Module aus der Informatik. Hierfür muss die Anfrage auf den betreffenden Studiengang gehen, um die Module herauszufinden, die dieser Studiengang insgesamt absolvieren muss. Diese Menge muss verglichen werden mit den Modulen, die die Informatik anbietet. Werden die Module ausgegeben, die in beiden Mengen vorhanden sind, dann bekommt man die Module, die ein anderer Studiengang in der Informatik hören kann. Wenn die Arbeit beim Auslesen der FPO gewissenhaft und in regelmäßigen Zeitabständen immer wieder kontrolliert wird, bekommt man ein genaues und exaktes Ergebnis. Der Exportbeauftragte kann mit einer einfachen Anfrage, die Informationen, die ein Studierender braucht, schnell und korrekt weiter geben.

### **6.5.2 Berichte an das Department**

Die Berichterstattung, welche Studiengänge eine Lehrveranstaltung bzw. ein Modul besuchen, kann durch die genaue Auswertung der FPO auch sicher gestellt werden. Das Department interessiert, von welcher Fakultät die Studierenden kommen, die eine Lehrveranstaltung bzw. ein Modul besuchen. Natürlich ist auch die Anzahl der Studiengänge von Interesse. Diese Informationen liegen in der FPO der Studiengänge. Der Weg der Anfrage sieht wie folgt aus: Beginnend mit der Lehrveranstaltung, über die Teilleistung und das dazugehörige Modul. Beim Modul muss man nur noch betrachten, welche FPOen dieses Modul vorschreiben oder in welchen Modulgruppen es sich befindet. Aus dieser Information kann man dann auf den Studiengang schließen. Diese Informationen können schnell aus der Datenbank gelesen werden. Allerdings setzt diese Anfrage eine gute und konsequente Bearbeitung der FPO voraus.

### **6.5.3 Studierendenanzahlen in Lehrveranstaltungen**

So genau wie die Analyse bezüglich der Auffindung von exportierten Lehrveranstaltung ist, so unberechenbar ist die Anzahl der Studierenden, die in eine Lehrveranstaltung kommen werden. Für diese Analyse werden Informationen benötigt, wie z.B. die Anzahl der Studierenden eines Studiengangs, die in dem Fachsemester sind, in der ein Modul angeboten wird. Hier werden nicht nur die aktuellen Informationen benötigt, sondern auch die Informationen von den letzten zwei Jahren verwendet. Die letzten zwei Jahren sind relevant, um das Verhalten der Studierenden abzuschätzen. Weiter in die Vergangenheit zu blicken, ist nicht optimal, weil sich die Studiensituationen zwischen den Jahren verändert. Beispielsweise sei die Wohnheimsplatzsituation erwähnt.

### Die 100%

Als Erstes muss die Anzahl der Studierenden, die eine Teilleistung bzw. Lehrveranstaltung eigentlich besuchen müssen, ermittelt werden. Diese Anzahl setzt sich zusammen aus den Jahrgangszahlen der Studiengänge, die die Lehrveranstaltung nach ihrer FPO besuchen müssen, plus die Anzahl der Studierenden, die die Prüfung im letzten Semester nicht bestanden haben. Zu den "nicht bestanden-Kandidaten" zählen auch die Studierenden, die durch Krankheit nicht an der Prüfung teilnehmen konnten.

### Wie viele Studierende waren wirklich da?

Nun muss berechnet werden, wie viele Studierende denn tatsächlich eine Lehrveranstaltung besucht haben. Die aufsummierte Studierendenzahl einer Prüfungsleistung (bestanden, nicht bestanden, endgültig nicht bestanden) ergibt eine gute Einschätzung. Das sind die Studierenden, die in der Lehrveranstaltung sitzen hätten sollen und somit auch ein Platz bekommen sollten. Von dieser Anzahl berechnet man zehn Prozent heraus und addiert diese Zahl zu den Studierenden, die geprüft wurden, dazu. Diese zehn Prozent kümmert sich um die Studierenden, die vergessen haben, sich für die Prüfung anzumelden oder aus irgendeinem Grund nicht geprüft werden müssen, wie beispielsweise die Studierenden, die sich eine Lehrveranstaltung als Schlüsselqualifikation anrechnen lassen wollen und somit nicht bei den Prüfungsdaten auftauchen. Ein kleiner Puffer nach oben ist besser als am Ende einen Raum zu haben, in dem nicht alle Studierenden einen Platz finden.

Für die Vorausberechnung braucht man das Verhältnis der Studierenden, die in den Lehrveranstaltungen sein müssten (100%), und die Studierenden, die wirklich in der Lehrveranstaltung waren. Für die Berechnung der Studierenden, die wirklich da waren, werden die Prüfungsinformationen verwendet, die direkt vor der neuen Vorlesungszeit erhalten werden. Um einen aussagekräftigen Wert zu erhalten, werden Werte von den zwei Semestern bzw. zwei Jahren (davon abhängig, ob die Lehrveranstaltung semesterlich oder jährlich angeboten wird) zuvor, die zeitlich direkt vor der Analyse liegen, berechnet. Die beiden Verhältnisse, die sich daraus ergeben, müssen noch gewichtet werden. Das zeitlich nähere Semester bzw. Jahr vor der zu analysierenden Lehrveranstaltung ist aussagekräftiger als das vor zwei Semestern bzw. zwei Jahren. Der zeitlich frühere Jahrgang wird dabei mit 38% gewichtet und der zeitlich jüngere mit 62%. Die unterschiedliche Gewichtung der beiden Jahre, setzt sich aus Tests verschiedener Lehrveranstaltungen zusammen. Aus diesen beiden Werten wird dann der arithmetische Mittelwert gebildet.

Der arithmetische Mittelwert wird nun verwendet, um ihn mit den Jahrgangsinformationen aus dem aktuellen Semester zu verrechnen. So werden die Jahrgangsinformationen der Studiengänge, die eine Lehrveranstaltung besuchen müssen, zusammen



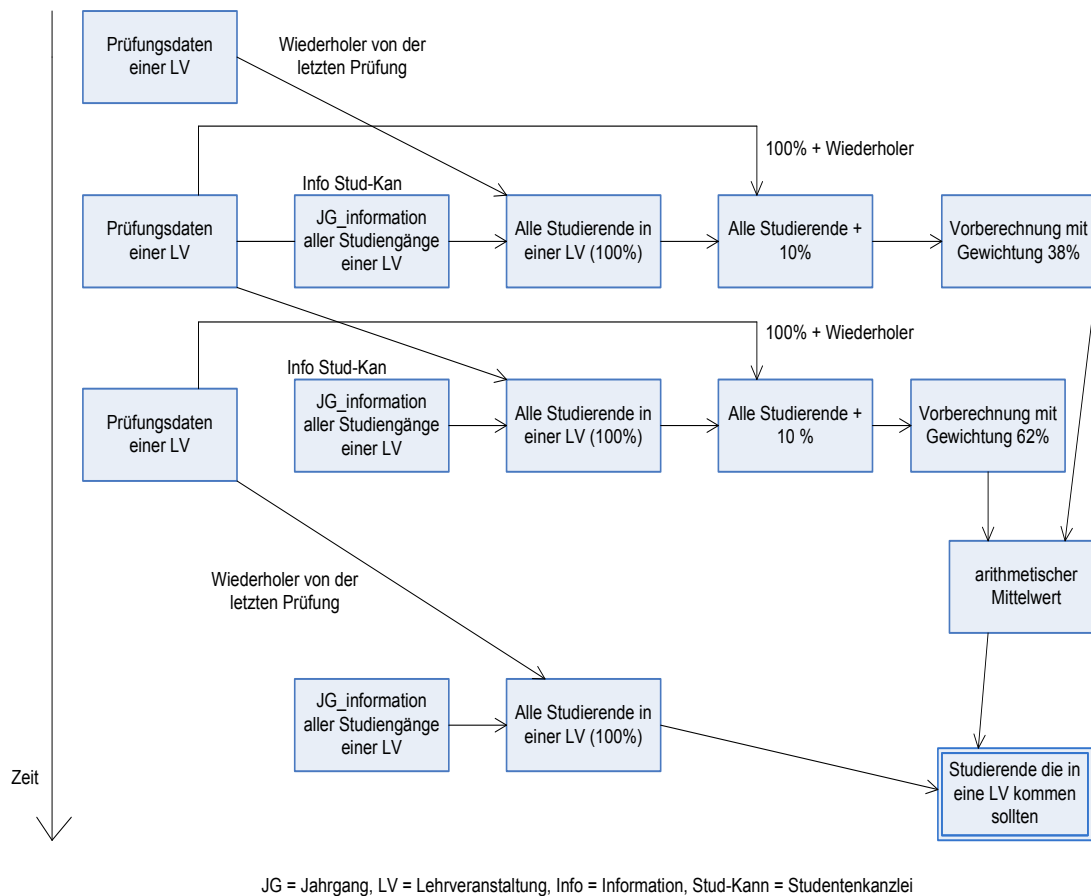


Abbildung 6.1: Skizzierte Berechnung der Studierendenzahlen

addiert, und dies ist Anzahl der Studierenden, die eine Lehrveranstaltung besuchen müssten. Dieser Wert wird mit dem arithmetischen Mittel der zwei vorangegangenen Jahren verrechnet. Dies spiegelt die Studierendenzahl derer wieder, die an einer Lehrveranstaltung teilnehmen sollten. Für die aktuelle Berechnung liegen die Jahrgangsinformationen schon vor der Vorlesungszeit vor. Hierbei gibt es immer noch Fluktuationen wie im weiter oben beschriebenen Fall der ausländischen Studierenden. Dieser Anteil liegt im Normalfall unter drei Prozent der Besucher einer Lehrveranstaltung. Die Berechnung wird in Abbildung 6.1 skizziert.

Eine exakte Vorhersage ist schwer zu erzeugen, weil noch einige Faktoren mitspielen, die nicht direkt vorhersehbar sind. Dazu gehört der folgende Fall: Gibt es zeitliche Lücken im Stundenplan vor oder nach den betrachteten Lehrveranstaltungen, kann es sein, dass nicht mehr alle Studierende in diese Lehrveranstaltung gehen. Wechseln

Dozenten, so kann es auch sein, dass es Schwankungen in der Anzahl der Studierenden gibt, die die Lehrveranstaltung besuchen, weil der neue Dozent die Lehrveranstaltung in den Augen der Studierenden besser oder schlecht hält. Auch wird die Besucherzahl davon abhängen, ob eine Abgabe von Hausaufgaben am gleichen Tag oder nächsten ist. Es gibt noch weitere Faktoren, die in Kapitel 1.1 thematisiert sind. Selbst dann sind vermutlich nicht alle Gründe abgedeckt.

# 7 Technik

[ER01] Nachdem jetzt die Planungen abgeschlossen sind, steht die Entscheidung an, welche Software für die Basisdatenbank und das Data Warehouse-System verwendet werden soll. Für die Entwicklung gibt es zwei Richtungen: Die eine besteht darin, kommerzielle Produkte zu verwenden. Dabei wird hier genauer der Microsoft SQL-Server mit der Business Intelligent und das gleiche Produkt der IBM, die DB2-Datenbank und der IBM InfoSphere Warehouse Enterprise Edition Processor Value Unit, geprüft. Microsoft bietet dieses Paket mit der Basisdatenbank und dem Data Warehouse System komplett an. Bei der IBM sind diese beiden Produkte getrennt. Als Alternative werden auch noch zwei Open Source Data Warehouse-Projekte betrachtet, die durch ihren kostenlosen Erwerb einen Vorteil haben. Wichtig zu erwähnen ist hierbei, dass die Open Source-Projekte nicht an eine bestimmte Basisdatenbank gekoppelt sind. Die betrachteten Produkte sind Business Intelligent-Produkte von Jasper und Pentaho.

Bei der Betrachtung soll jetzt nicht auf jede Feinheit eingegangen werden oder die Analyse-Geschwindigkeit der Produkte verglichen werden. Es geht darum, welche Gründe es gibt, ein Produkt zu verwenden, wenn man sich die Hauptcharakteristika der verschiedenen Anbieter anschaut, und wie diese auf die Ziele passen. Als Hauptigenschaften werden dabei folgende Punkte gewählt: Dokumentation, Flexibilität, Stabilität der Software, Kosten, Update-Fähigkeiten, Installation und Benutzerempfinden bei der Benutzung. Dabei wird die Beschreibung vom Open Source Produkt JasperSoft detaillierter sein, weil dieses schlussendlich benutzt wurde.

Die Datenbanksysteme wurde alle bis auf JasperSource nur kurz getestet. Jasper Source wurde zur Verwendung vorgegeben.

## 7.1 Kommerzielle Data Warehouse System

### 7.1.1 Microsoft SQL Server mit der Business Intelligent

Mit dem Microsoft SQL-Server [MyS09] verhält es sich ähnlich wie mit der IBM-Lösung. Ein Unterscheid zwischen den Systemen ist, dass sich der Microsoft SQL-Server nicht

auf Linux- und Unix-Betriebssystem installieren lässt. Durch die direkte Verwendung vom Betriebssystem Windows können kleine Updates über Windows Update installiert werden. Versionsupdates sind auch hier kostenpflichtig. Die Installation läuft auch wieder einfach und ohne Probleme ab. Der Installer vom SQL-Server stellt bei der Installation ein paar Fragen und installiert sich zum größten Teil alleine. Zur Stabilität ist zu sagen, dass, wenn das System richtig konfiguriert ist, das System gut durchläuft. Es kommt zu keinen Abstürzen, und wenn ein Fehler auftritt, wird er per Fehlermeldung abgefangen. Über sinnvolle Informationen bei den Fehlermeldungen kann man sich jetzt streiten. Dokumentationsmaterial stellen Microsoft und Drittanbieter zur Verfügung. Ein kleines Begleitheft wird beim Kauf zur Verfügung gestellt. Auch gibt es im Internet Communities bei denen man sich Hilfe holen kann, wenn Probleme auftauchen sollten. Wer sich mit Windows und den Office-Produkten von Microsoft auskennt, wird sich auch mit der Bedienung des SQL-Servers anfreunden. Wenn man sich die Standardversion vom SQL Server besorgt, ist man mit ca. 5.600 Euro dabei.

### 7.1.2 IBM DB2 mit InfoSphere Warehouse Enterprise Edition

Die Installation der IBM<sup>1</sup> Produkte kann einfach sein, indem man den Autoinstaller, der nur mit ein paar Fragen die Installation begleitet, benutzt. Komplizierter ist die Installation, wenn man die sie händisch durchführt. Die zweite Variante sollte man bevorzugen, wenn man ein System auf Leistung und Sicherheit trimmen will. Die Installation kann sowohl unter Windows, Linux und Unix durchgeführt werden. Es gibt auch Versionen, die speziell auf 64-Bit-Systeme ausgelegt sind. Wenn eine neue Version des Datenbanksystems oder InfoSphere [Gmb09b] herauskommt, kann diese ohne große Schwierigkeiten eingespielt werden. Das Produktiv-System muss dafür allerdings kurze Zeit offline gehen. Es müssen aber keine Sicherungen von den Datenbankinhalten getätigt werden. Updates sind jedoch meist nicht kostenlos und müssen erworben werden. Die instabilste Phase des Projektes ist bei der Installation, wenn sie per Hand durchgeführt wird. Danach laufen die Systeme stabil. Treten Fehler auf, so werden sie oft durch sinnvolle Fehlermeldungen dargestellt. Als Nachteil ist zu sehen, dass diese Lösung stark auf die Basisdatenbank DB2 [Gmb09a] zugeschnitten ist, und es ist empfehlenswert, diese zu benutzen. Informations- und Dokumentationsmaterial wird beim Kauf der Softwareprodukte mitgeliefert, und reicht dieses nicht aus, gibt es auch spezielle Benutzerhandbücher von der IBM, die allerdings extra käuflich zu erwerben sind. Auch gibt es viele Dokumentationen, die nicht von Mitarbeitern der IBM oder Tochterunternehmen geschrieben sind. Diese können im Buchhandel erworben werden. Da IBM mit DB2 stark am Markt vertreten ist, gibt es einige Internet-Communities, bei denen man sich Hilfe holen kann, ohne etwas zu bezahlen. Bezahlen ist hierbei ein Stichwort. Die Produkte sind nicht ganz billig. Für

---

<sup>1</sup>International Business Machines Corporation

die Basisdatenbank braucht man eine Version von DB2, die man als Mehrbenutzer für ca. 2.500 Euro bekommt, dazu kommt noch InfoSphere, das etwa 1.000 Euro kostet. Die Bedienung ist sehr angenehm, und man findet alles da, wo man es bei den Software-Produkten erwartet.

## 7.2 Open Source Data Warehouse Systeme

Open Source Data Warehouse Systeme besitzen nicht wie die kommerziellen Produkte eine direkte Verbindung zu ihrer Basisdatenbank. Sie ist oft frei wählbar, aber es gibt auch Einschränkungen.

### 7.2.1 Pentaho

Pentaho [Int09] stellt eine eigene Business Intelligent mit Data Warehouse zur Verfügung. Darunter fallen Programme zum Laden der Basisdatenbank, für den ETL-Prozess und für den Analyseschritt. Die Business Intelligent liefert standardmäßig keine Basis Datenbank mit. Man merkt aber schon, dass die Favoritendatenbank MySQL [Gmb09c] ist. Es gibt auf der Webseite von Pentaho eine vorgefertigte Version, die angepasst auf das MySQL-Datenbank-System ist. Postgres und andere auch kommerzielle Datenbanksystemhersteller kann man mit Pentaho auch benutzen. Allerdings ist die Konfiguration etwas zeitaufwendig, wenn man mit den Datenbanksystemen nicht vertraut ist. Die Installation der Programme funktioniert sowohl unter Windows, Linux und Unix. Für manche Programme muss allerdings das Java Runtime Environment installiert sein. Zum größten Teil laufen die Anwendungen stabil, aber es kann auch sein, dass Fehlermeldungen auftauchen, die besagen, dass ein Fehler aufgetreten ist, allerdings nicht, warum der Fehler auftritt und was die Ursache des Fehlers ist. Kleine Verbesserungen werden nicht zum Update angeboten. Man bietet gleich komplette neue Versionen an. Somit müssen Projekte immer abgespeichert werden und in die neue Version immigriert werden. Dafür sind die neuen Versionen kostenlos. Bei der Bedienung kann Pentaho mit den kommerziellen Produkten nicht mithalten. Es kann sein, dass die in der Enterprise Version, die kostenpflichtig ist, anders aussieht. Die kostenpflichtige Version wurde aber nicht getestet. Bei der Dokumentation sieht es bei weitem anders aus als bei den kommerziellen Produkten. Es gibt eine Hauptdokumentation, die von Pentaho auch kostenlos zur Verfügung gestellt wird. Weiter gibt es kleine Communities, die Hilfestellung geben. Das Problem hierbei ist, dass sich in der Open Source Welt ein paar wenige Produkte behauptet haben und es somit viele Data Warehouse Projekte gibt, die jeweils einen gewissen Anhängerkreis haben. Die Kosten für dieses Produkt sind gleich null, wenn man nicht die Enterprise Version nimmt.

## 7.2.2 JasperSoft

Jasper [Cor09a] stellt genau so wie Pentaho nur die Business Intelligent mit dem Data Warehouse zur Verfügung. Jasper läuft unter Windows und Linux. Die Installation läuft nicht so einfach vonstatten. Jasper gibt keine Basisdatenbank vor, aber der Hersteller rät, nur MySQL oder Postgres zu verwenden. Allerdings bringt der JasperServer schon MySQL mit. Selbst wenn man das mitgebrachte MySQL verwendet, lässt sich Jasper nicht so einfach installieren. Es müssen erst die Gruppe und der Benutzer "mysqlängelegt werden. Wenn man selbst MySQL installiert, gibt es das Problem, dass die Kommunikation zur Datenbank per Hand hergestellt werden muss. Bei den Basiseigenschaften wie dem Laden der Basisdatenbank laufen die Jasper Programm sehr stabil, und wenn es Fehler gibt, sind sie auch gut beschrieben. Allerdings beim ETL-Prozess kann es schon sein, dass ein Prozess mit einem Java Stacktrace abbricht und man nicht den Fehler ermitteln kann. Wie bei Pentaho werden keine inkrementellen Updates bereitgestellt. Neue Versionen erscheinen bei Jasper so schnell, dass in einem Quartal drei Versionen aktuell waren. Das Problem dabei ist, dass, wenn man die mitgelieferte Datenbank verwendet, immer erst die Datenbank komplett gesichert werden muss, und sie bei der neuen Version als Backup wieder einspielen muss. Benutzerfreundlichkeit in den Programmen ist zum größten Teil da. Problematisch ist nur, wenn Menüs in einer anderen Version in einem anderen Bereich untergebracht sind, als man sie erwartet. Von Jasper gibt es eine kostenpflichtige Version, die vermutlich weitere Optionen, die besonders beim ETL-Prozess sehr hilfreich wären, erweitern. Dokumentation wird von Jasper für die Grundfunktionalitäten zur Verfügung gestellt. Ein Problem ist hier, dass Jasper auf ein Tool, das "SSchema Workbench", zurückgreifen muss, und wie dieses Tool mit den anderen Jasper-Tools zusammenspielt, wird nicht beschrieben. Von Jasper ist die Standardausstattung kostenlos.

Tabelle 7.1 veranschaulicht den Vergleich der Systeme.

## 7.3 Die technische Umsetzung

Viele Probleme, die bei Jasper aufgetaucht sind, wurden erst im Laufe der Zeit entdeckt. Eingesetzt wurde als Produktiv-System JasperSoft. Die einzelnen Funktionsweisen und Fall- Beispiele werden im Anhang erklärt.

### 7.3.1 Installation

Wie in Abschnitt 7.2.2 beschrieben, verfügt das Paket von JasperSoft gleichzeitig über die Basisdatenbank von MySQL und eine Webdarstellung über Tomcat. Die Installati-

	Microsoft	IBM	Pentaho	JasperSoft
Installation	einfach	einfach	A.N.	A.N.
Flexibilität der BD	keine	gering	hoch	hoch
Zuverlässigkeit	hoch	hoch	mittelmäßig	mittelmäßig
Stabilität	hoch	hoch	NEIN	JA
Kosten	hoch	hoch	keine	keine
Updatefähigkeit	ja/k.V.	ja/k.V.	ja	ja
Benutzerfreundlichkeit	sehr gut	sehr gut	gut	gut

A.N. = Anleitung Notwendig

BD = Basisdatenbank

k.V. = keine Versionsupdates

Andere Menüführung zwischen Version behindert die Arbeit

Tabelle 7.1: Data Warehouse Systemevergleich

on beinhaltet ein paar Voraussetzungen, auch wenn man das Gesamtpaket installiert. So müssen die Ports 8080, 3306 und 8000 frei sein. Wenn MySQL mitinstalliert werden soll, muss es die Benutzergruppe und den Benutzer mysql geben bzw. diese müssen eingerichtet werden. Die Installation kann allerdings auf Windows, Linux und Unix aus der Kommandozeile oder grafisch installiert werden. Für eine bessere Bedienung und für eine einfachere Eingabe empfiehlt es sich, noch das Administrationstool von MySQL zu installieren.

Um den Jasper zu starten, empfiehlt es sich, einen neuen Benutzer zu erstellen. Dieser Benutzer startet dann den MySQL-Server, Tomcat-Server, eine Java-Instanz und den JasperSoft-Server. Bei dem Shutdown-Skript von JasperSoft werden diese Programme nicht immer ordentlich herunter gefahren. Dieses Problem löst man, beispielsweise unter Linux, mit `pkill -u Benutzername`.

Nach der Installation sollte man die Administrationsprogramme von MySQL installieren, beispielsweise MySQL Administrator. Unter Linux bekommt man diese durch die Updateprogramme der einzelnen Linux-Hersteller oder bezieht das Programm direkt von der Webseite von MySQL. Es ist allerdings nicht möglich, per Kommandozeile sowohl unter Windows als auch unter Linux mit dem Befehl `MySQL` zuzugreifen. Hier muss man explizit angeben, dass man per Protokoll TCP<sup>2</sup> auf den MySQL Server zugreifen will.

<sup>2</sup>Transmission Control Protocol

### 7.3.2 Basisdatenbank: Schema anlegen

Am einfachsten geht die Arbeit mit dem MySQL Administrator auf der Basisdatenbank. Mit diesem Programm kann man vieles verwalten. Neben den Serverinformationen, laufenden Diensten und Benutzerverwaltungen kann man auch die Tabellen für die einzelnen Schemata anlegen. Die Funktionsweise ist relativ leicht: Neue Tabelle anlegen, Namen vergeben, Attribute mit Wertebereich angeben, Primärschlüssel definieren und die Fremdschlüssel eintragen. Danach steht die Tabelle. Es werden dafür keine SQL-Befehle benötigt, aber man kann sie verwenden, wenn man will. Die Bedienung ist einfach. Mit großen Hauptmenüs, die die wichtigsten Eigenschaften der Datenbank beschreiben, ist das Gesuchte schnell gefunden. Allerdings muss man, wenn sich der MySQL Administrator nicht auf dem Datenbankserver befindet, die Benutzer für die Fernverwaltung freischalten. Die Freischaltung kann man nur am Server aktivieren.

### 7.3.3 Laden der Daten

Vorab zeigt Abbildung 7.1, welche Informationen in die Basisdatenbank geladen werden sollen.

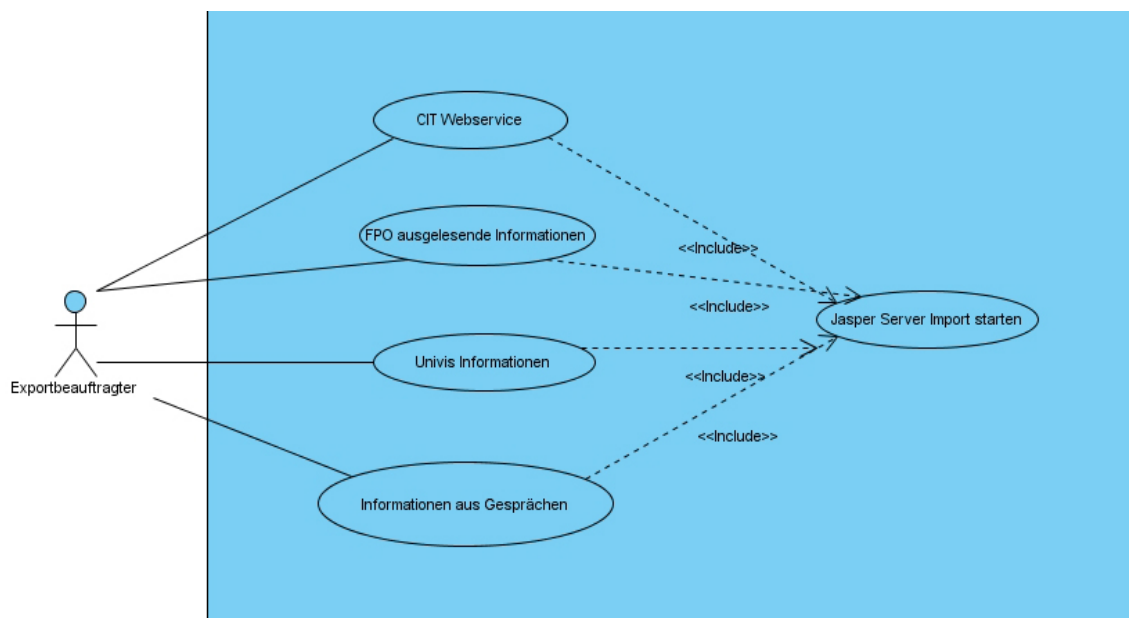


Abbildung 7.1: USE-Case der zu ladenden Daten



Das einfachste Laden ist, wenn man in MySQL Query Browser verwendet. Wählt man die gewünschte Tabelle an der rechten Seite im Menü, so kann man die Tabelle wie mit Tabellenkalkulationsprogramm füllen. Dieses Tool bietet auch an, die Tabellen per txt<sup>3</sup>-, csv<sup>4</sup>- oder Excel Datei zu füllen.

JasperSoft bietet darüber hinaus ein eigenes Programm, um Daten in die Basisdatenbank einzupflegen oder das Data Warehouse zu laden. Das von Jasper entwickelte Programm heißt JasperSoftETL. Es werden verschiedene Symbole bereitgestellt, die unterschiedliche Bedeutungen haben. Es gibt Symbole, die Eingabedokumente und Symbole, die Datenbankaktionen darstellen. Für das Laden des Data Warehouses werden auch Transformationssymbole angeboten. Durch Hineinziehen der Symbole auf die Arbeitsfläche kann man diese benutzen. Den Transfer der Daten kann man mittels Ziehen einer Verbindungslinie zwischen dem Eingangsfilesymbol und dem Ausgangsfilesymbol realisieren. Dieser Datentransfer kann auf zwei Arten realisiert werden. Die eine Möglichkeit ist, dass der Datentransfer manuell per Anwender gestartet wird. Die andere Möglichkeit ist, dass das System auf ein Event wartet, wie beispielsweise, dass sich das Inputfile verändert. Dann können sofort die Daten in die Basis-Datenbank bzw. in das Data Warehouse geladen werden. Diese Variante funktioniert sehr gut, wenn man die Basisdatenbank füllen will. Problematischer ist der Ladeprozess des Data Warehouse aus der Basis Datenbank. Für das Laden gibt es in JasperSoftETL die ELT-Symbole. Diese Symbole sind immer auf die speziellen Datenbanksystemhersteller zugeschnitten. Allerdings können die datenbankspezifischen Symbole nur unter sich verwendet werden. Somit können nur zwei Operationen abgebildet werden. Das erste ist das direkte Laden von einer Tabelle der Basisdatenbank in das Data Warehouse und das Laden aus mehreren Datenbanken in das Data Warehouse. Beim letzten Schritt kann noch vorab eine Transformation durchgeführt werden, um die Tabellen bezüglich der verlangten Attribute in der Data Warehouse-Tabelle zu vereinen.

Beim Laden der Basisdatenbank kommt es zu keinen Problemen. Wenn beim Einfügen der Daten Duplikate erkannt werden, besteht die Möglichkeit, das Laden rückgängig zu machen oder dass die Duplikate nicht in die Basisdatenbank geladen werden. Bei beiden Möglichkeiten werden die Informationen, welche Duplikate beim Ladeprozess die Probleme verursachen, im Ladebericht vermerkt.

Eine andere Methode, um den Duplikaten Herr zu werden, besteht darin, JasperETL zu sagen, dass er beim Einlesen gewisse Zeilen auslassen soll. Die Zeilen können am Anfang oder am Ende einer Datei sein: Auch kann man Spalten, die man nicht benötigt, auslassen.

---

<sup>3</sup>Textdatei

<sup>4</sup>Colon Separated Values

Des Weiteren kann eingestellt werden, mit welchem Seperator die einzulesende Datei bearbeitet werden soll. Um die Arbeit zu erleichtern, gibt es die Möglichkeit, das Einladen per Haken auf CSV-Daten zu trimmen.

Beim Laden des Data Warehouses gibt es mehr Probleme. So ist es nicht möglich, mit der Version von JasperETL die Tabellen vom Data Warehouse mit den Tabellen der Basisdatenbank per Datenbankschema zu trennen. So wurde eine Präfix-Notation für die Tabellennamen eingeführt, um die Tabellen von der Basisdatenbank des Data Warehouses schnell unterscheiden zu können. Der Präfix für die Basisdatenbank ist `basis` und für das Data Warehouse ist es `dwh`. Ein weiteres wichtiges Problem ist, dass es nicht möglich ist, Duplikate sicher zu erkennen. Durch die Tabellendefinition gibt es Attribute, die eindeutig sein müssen, z.B. der Primärschlüssel. Wenn jetzt eine Tabelle sukzessive nachgeladen wird und ein Tupel doppelt enthalten ist, bricht der Vorgang mit einem Java Error ab, und die Ladetransaktion bleibt bis zum Fehler in der Datenbank, was zu Problemen führen kann. Auch ist nur ein komplettes Laden einer Tabelle bzw. einer Transformation aus verschiedenen Tabellen der Basisdatenbank möglich, d.h. man muss immer die komplette Basisdatenbanktabelle in das Data Warehouse laden. Somit tritt der Duplikatenfehler oft auf. Eine Lösung des Problems wäre, die Data Warehouse-Tabellen vor dem Laden immer komplett zu leeren.

Ein weiterer Kritikpunkt ist, dass kleine Veränderungen am Schemata der Datenbank zur kompletten Neubeschreibung der Ladeprozesse führt. Bei dem Ladeprozess wird die Datenbank per JDBC Treiber mit JasperETL verbunden. Danach muss man JasperSoftETL sagen, dass er die Informationen der Tabellen in einem Datenbankschema erfassen soll. Hierbei werden dann auch alle Attribute erfasst. Danach können die Tabellen für den Ladeprozess verwendet werden. Werden jetzt Veränderungen am Tabellenschema durch geführt, so bekommt JasperSoftETL nur mit, dass sich etwas verändert hat, und blockiert die Operationen. Was sich genau verändert hat, bekommt es aber nicht mit. So muss das Schema einer Datenbank neu geladen werden, was dazu führt, dass die alten definierten Ladeprozesse auf ungültigen Tabelleninformationen arbeiten und so nicht mehr funktionsfähig sind.

Vorteile beim Programm sind, dass alle Symbole leicht mit Ziehen auf die Arbeitsfläche verwendet werden können. Der Transfer ist auch mit ein paar Handgriffen eingerichtet, wenn man weiß, in welchem Menü man gucken muss. Problematisch ist, dass auch schnell neue Versionen herauskommen und jede Version kleine Veränderungen im Menü besitzt. Da heißt es oftmals umdenken.

### 7.3.4 Pentaho Schema Workbench

Es wird hier JasperSoft-Software beschrieben. Es stellt sich die Frage, warum man jetzt ein Produkt von Pentaho braucht. Der JasperServer braucht eine XML Beschreibung

von den Dimensionen und Fakten des Data Warehouses. JasperSoft besitzt kein eigenes Programm dafür. Aus diesem Grund muss auf das Produkt Schema Workbench von Pentaho zurück gegriffen werden. Die Beschreibung wird in XML-ähnliche Syntax angefertigt. JasperSoft bietet kein Programm an, das diese Aufgabe übernimmt. Aus diesem Grund wird auf die Schema Workbench von Pentaho zurück gegriffen. JasperSoft bietet einen Link auf die Seite von Pentaho zu Schema Workbench an. Nach dem Download stellt man fest, dass die Schema Workbench nicht funktionsfähig ist. Es muss erst ein JDBC-Treiber für die eingesetzte Datenbank installiert werden. Diesen Treiber erhält man beispielsweise, wenn man die komplette Business Suit von Pentaho installiert. Dann kann man den Treiber von einer Anwendung zur Schema Workbench ziehen. Nach der Installation des Treibers, wird eine Verbindung zu dem Datenbankserver aufgebaut. Das Programm ist ziemlich einfach gestrickt, um Datenwürfel zu bauen. Es gibt die Möglichkeit, Dimensionen global zu definieren, und sie dann in einzelnen Datenwürfeln zu verwenden. Das Programm ist einfach und übersichtlich aufgebaut. Allerdings fehlt zu diesem Programm die Dokumentation. JasperSoft bietet nur eine Kurzbeschreibung dazu, die für die richtige Verwendung nicht ausreicht. Ähnlich sieht es bei Pentaho aus, allerdings bekommt man eine genauere und detailliertere Dokumentation, wenn man sich die Schema Workbench kauft. Hier soll auch eine aktuellere Schema Workbench-Version zu finden sein, die als frei zum Download zur Verfügung steht. Durch Fehler in der Parallelhierarchie `dimOrg` wurde aus `dimOrg` drei Dimensionen erstellt. Die Daten der Dimension sind im Snowflake-Schema entwickelt, und es müssen auch keine Daten dupliziert werden. Die neuen Dimensionen heißen `dimOrg-Lehrstuhl`, `dimOrg-Modgr` und `dimOrg-Teilleistung`.

Snowflake wurde eingesetzt, um eine größere Transparenz zu erhalten, weil beim Erlernen des Programmes Schema Workbench, die Dimension oft verändert wurde. Auch werden Anfragen meist nur auf einer Klassifikationsstufe durchgeführt, was den Aufwand von wenigen Verbundoperationen rechtfertigt.

Ein schöner Vorteil ist, dass in der Schema Workbench gleich MDX-Anfragen [Cor09b] modelliert werden können. Dadurch besteht die Möglichkeit, gleich das Schema im Programm zu testen. Durch die Verbindung zur Datenbank kann die Anfrage gleich die Werte liefern. Veränderungsfehler können so schnell behoben werden. Da das Programm ziemlich einfach entwickelt ist, besteht die Ausgabe aus reiner Faktenangabe ohne grafische Unterstützung, was zu Folge hat, dass die Übersicht leidet.

### 7.3.5 Analyse am JasperSoft Server

Der JasperSoft-Server gehört zu den Komponenten, die am Anfang als Bündel installiert worden sind. Nach dem Anmelden über den Webbrowser als Administrator,

wird dem Benutzer das Wurzelverzeichnis dargestellt. Zu sehen sind Unterpunkte, wie beispielsweise Analyse Components oder "Reports". Weiterhin kann man noch Bilder für die Analysen hinterlegen oder Dateien hochladen, die als Zusatzinformationen bearbeitet werden können. Für die Administratoren ist hauptsächlich die Analyse Components am wichtigsten. Hier wird beschrieben, welches Datenbanksystem verwendet und welches Schema auf der Datenbank benutzt wird. Dazu muss angegeben werden, welcher Art der Treiber zur Datenbank ist: beispielsweise kann der JDBC-Treiber für ein Datenbanksystem verwendet werden. Des Weiteren muss auch das multidimensionale Schema eingebracht werden, was zuvor mit der Schema Workbench erstellt wurde. Ist dies erbracht worden, kann man nun eine Analyse (Report) erzeugen.

Für die Analyse wird auch beim JasperServer MDX-Anfrage verwendet. Allerdings werden die Antworten durch grafische Unterstützung in Form von Pivot-Tabellen deutlich besser dargestellt, als bei der Schema Workbench. Des Weiteren können aus den erfragten Daten auch Grafiken wie beispielsweise Säulen oder Tortengrafiken erstellt werden.

Dazu muss man sagen, dass sich der Server etwas instabil bei Veränderungen verhalten kann. Wenn viele Veränderungen an den Datenbankverbindungen oder am multidimensionalen Schema vorgenommen werden, reagiert der Server ziemlich hart, weil er dies nicht verarbeiten kann, und stürzt ab. Danach kann man nur den Server herunterfahren und neu starten. Dies ist besonders problematisch, wenn mehrere Benutzer gleichzeitig auf dem Server arbeiten.

Da JasperSoft in relativ kurzer Zeit komplette neue Versionen herausbringt, müssen die erstellten Analysen exportiert und in der neuen Version importiert werden. Dies geschieht nicht automatisch, sondern muss von Hand durchgeführt werden, wenn kein eigenes ausführbares Skript zur Verfügung steht. Wenn allerdings mit dem Bündel von JasperSoft gearbeitet wird, in dem sich auch die MySQL-Datenbank befindet, muss man den Datenbankinhalt auch sichern und wieder zurückspielen. Im JasperSoftserver müssen auch die Verbindungen zu dem Datenbankschema nach der Installation einer neuen Version vom JasperSoftServer komplett neu eingerichtet werden. Dies nimmt einige Zeit in Anspruch. Durch das Einspielen einer neuen Version geht bei geübten Personen ca. eine Stunde Zeit verloren. Das wirkt abschreckend, den JasperSoftServer immer aktuell zu halten.

### 7.3.6 IReport

IReport ist ein extra Programm, das JasperSoft anbietet, um Analysen zu modellieren und sie in einem guten Design in gedruckter Form oder als PDF zu präsentieren. Die Anfragesprache ist SQL oder MDX, um Daten vom JasperServer zu bekommen. IReport baut eine Verbindung zum JasperServer und von dort eine direkte Verbindung zur

Datenbank auf. Neben den Anfragen wird hier auf das Aussehen der wiedergegebenen Daten sehr viel Wert gelegt. So gibt es Schablonen für Firmenlogos oder Farbtabelle, die beurteilen, welche Farben nebeneinander gut passen. IRport ist das Programm, das sein Aussehen am meisten im Versuchverlauf verändert hat. Zwischen den Versionen 3.0.0. und 3.0.1 wurde eine komplett neue Benutzeroberfläche entwickelt, die mit der alten keine Ähnlichkeit mehr hat.



## 8 Evaluation

Die Frage, wie viele Studierenden in einer Lehrveranstaltung sitzen, ist auf den ersten Blick eigentlich leicht zu beantworten. Auf den zweiten Blick wird diese einfache Frage ziemlich undurchschaubar, weil diese Information nirgends ansatzweise zu finden ist.

Stand der Lösung des Problems: Bei der Beschreibung des aktuellen Standes wurde nur auf die Informationen eingegangen, die die Ziele dieser Arbeit genau beschreiben. Es wurde nicht auf die Funktionalität und Benutzerfreundlichkeit von Microsoft Access eingegangen oder den Vergleich zu anderen Datenbanksystemen. Dies wurde außer Acht gelassen, weil es hier darum ging, das System in Bezug auf die Ziele der Arbeit zu analysieren. Ein weiterer Grund dafür ist, dass auf Microsoft Access-Basis kein Data Warehouse entwickelt werden kann, um die Daten zu analysieren. Da die eine Hauptaufgabe war, ein Data Warehouse zu entwickeln, war der Einsatz von Microsoft Access nicht mehr möglich. Somit wurde auf weitere Analysen, die nicht direkt mit den Zielen der Berechnung der Studierendenanzahlen oder des Tatbestands der Auskunft verzichtet.

Des Weiteren wird in einem Abschnitt ein Problem bezüglich der Wahlkurse dargestellt. Diese Problematik wurde extra in dieses Kapitel gestellt, um die Probleme bezüglich der Abschätzung der Studierendenanzahlen in Modulgruppen zu thematisieren. Diese werden im darauf folgendem Kapitel thematisiert. Gleichzeitig stellt dies auch einen Ansatz dar, warum die Access-Datenbank dafür nicht ausgelegt wurde, die Studierendenanzahlen zu verwalten.

### 8.1 Datenquellen

Beim Betrachten der vier verwendeten Datenquellen fällt auf, dass die Zahl der Studierenden, die eine Lehrveranstaltung besuchen, nicht erfasst werden kann. Jede der Datenquellen hat Vor- und Nachteile, die abgewogen werden, aber selbst wenn alle Quellen zur Verfügung stehen, bekommt man noch nicht einmal dafür eine exakte Aussage über die Anzahl.

Ein optimales Verfahren wäre, wenn die Dozenten die Studierenden in den Lehrveranstaltungen zählen würden und die Anzahl jedes Mal notieren würden. Hier würden

die Dozenten nicht mitspielen, weil dies zu viel Zeit kosten würde. Eine weitere Möglichkeit wäre, dass sich Studierende an der Tür zum Raum der Lehrveranstaltung mit ihrer Mensakarte identifizieren müssen und die Anzahl der Studierenden automatisch vermerkt würde. Problematisch bei dieser Lösung ist, dass der Datenschutz nicht gewährleistet wäre. Diese Lösung wäre weiterhin sehr kostspielig, da alle Räume der Universität mit dem entsprechenden elektronischen System ausgestattet werden müssten.

Für die Zukunft wäre eine Datenquelle von Vorteil. Wenn es eine Momentaufnahme einer Lehrveranstaltung geben würde, könnte man mit dieser Information die Analyse der Studierendenzahlen für das kommende Semester viel genauer durchführen. Diese Daten müssten allerdings dann auch per Web- oder Datenbankinterface im Internet zur Verfügung stehen, um weitere manuelle Arbeit zu vermeiden.

Trotz der genauen Daten kann es sein, dass es nicht möglich sein wird, die Daten genau vorzusagen. Da die Studierenden Menschen sind und es viele Gründe geben kann, nicht zu einer Lehrveranstaltung erscheinen, ist eine Vorhersage schwierig.

## 8.2 Konzeptioneller Entwurf

Die Zielsetzung, nämlich die Entwicklung eines Uni-Schemas, war klar formuliert, aber die Umsetzung stellte sich allgemein als schwierig dar. Welche Informationen für die Studiengänge und für die Studierendenzahlen benötigt werden, war vergleichsweise gering, aber die Abhängigkeiten führten zu Schwierigkeiten. Wenn sich eine Lehrveranstaltung verändert, zieht das eine ganze Menge Veränderungen mit sich. So können Teilleistungen betroffen sein und somit auch Module. Von Modulen kann es in drei Richtungen weiter gehen, entweder zur Modulgruppe, zur Fachprüfungsordnung oder zum Jahrgang. Hierbei ist es egal, ob man direkt bei der Lehrveranstaltung etwas ändert oder weiter hinten im Weg, es kann immer zur Folge haben, dass man den ganzen Weg abarbeiten muss. Dies macht die Pflege nicht einfacher.

Um die gerade erwähnten Informationen alle zu erhalten, sind auch viele Quellen nötig. So muss neben der Recherche in der Fachprüfungsordnung viel durch Erfragen bei Lehrstühlen geschehen. Hier sei nur beispielsweise die Buchwissenschaft zu erwähnen, deren Studierende in Informatik Lehrveranstaltungen als Schlüsselqualifikationen hören kann. Die Informationen sind nirgendwo gelistet.

Die Informationsrecherche würde stark vereinfacht werden, wenn beispielsweise alle Informationen zentral gespeichert werden würden und jeder Studiengang seine Information, beispielsweise über Schlüsselqualifikationen, auflisten würde. In vielen Fachprüfungsordnungen werden Module nicht genau spezifiziert. Es kommt oft vor, dass die Module erst im jeweiligen Studienführer genau definiert werden. Somit müssen



Informationen in mehreren Dateien zusammengesucht werden, die an unterschiedlichen Orten gespeichert sind. Die Fachprüfungsordnung befindet sich auf der Seite des Prüfungsamtes, aber die Studienführer stehen auf den verschiedenen Seiten der Departments. Bezüglich Schlüsselqualifikationen muss nachgefragt werden.

Durch die hohe Zahl an Informationswegen kann es vorkommen, dass sich leichte Fehler in die Datenbank einschleichen, weil ein genaues Auslesen von nicht maschinen lesbaren Dokumenten erfolgen muss. Eine Verbesserung wäre, wenn die Fachprüfungsordnungen einem gleichem Schema folgen würden oder wenigstens die Informationen alle auf einer Seite zur Verfügung stehen würden. Für die Fachprüfungsordnungen ist dies der Fall. [EN09c]. Eine Möglichkeit um das maschinelle lesen zu gewährleisten wäre, die Fachprüfungsordnungen in eine feste XML-Struktur zu bringen, aber es müsste erst geprüft werden, ob diese Möglichkeit juristisch zulässig ist. Aus einem XML-File könnten die Daten schnell und einfach automatisch ausgelesen werden.

Ein weiterer hoher Pflegeaufwand sind die Verantwortlichen für die Teilleistungen, Module und Studiengänge zu Pflegen. Diese Beziehung wurde immer als schön zu haben definiert. Da durch, dass die Ansprechpartner, nirgends offiziell geführt werden, ist die Suche auf Nachfragen bei den einzelnen Departments. Diese Definition, dass nicht unbedingt ein Ansprechpartner gefunden werden muss, führt wohl über kurz oder lange dazu, dass die Verantwortlich nicht mehr gepflegt werden oder nur noch dann, wenn sie gerade gebraucht werden. Hier hätte man rigider vorgehen müssen und ein muss daraus machen sollen. Dies hätte zu massiven manuellen Pflegeaufwand geführt. Dies hätte nur erleichtert werden können, wenn eine Schnittstelle zum Univis, da wäre.

Ein weiterer hoher Pflegeaufwand sind die Verantwortlichen für die Teilleistungen, Module und Studiengänge. Diese Beziehung wurde immer als schön zu haben definiert. Da durch, dass die Ansprechpartner nirgends offiziell geführt werden, geschieht die Suche durch Nachfragen bei den einzelnen Departments. Diese Definition, dass nicht unbedingt ein Ansprechpartner gefunden werden muss, führt wohl über kurz oder lange dazu, dass die Verantwortlich nicht mehr gepflegt werden bzw. nur noch dann, wenn sie gerade gebraucht werden. Hier hätte man rigider vorgehen müssen und ein Muss daraus machen sollen. Dies hätte jedoch zu massivem manuellen Pflegeaufwand geführt, der nur dadurch erleichtert hätte werden können, wenn eine Schnittstelle zum Univis existieren wäre.

Bei der Betrachtung des Departments Informatik müsste man im E/R-Diagramm nicht die Fakultäten und andere Departments außer dem der Informatik selbst pflegen. Dies sind die Überreste der Idee, das Projekt für alle Departments und alle Module auszuweiten. Trotz der engen Betrachtung bietet die Darstellung den Fakultät und Department den Vorteil einer kompletten Navigation von der höchsten Universitäts-ebene an. Für diese Darstellung eignet sich das multidimensionale Schema gut. Wird

das Projekt irgendwann universitätsweit verwendet, bräuchte man das Schema der Basisdatenbank und das multidimensionale Schema nicht anpassen.

Die Einführung von Surrogaten für die Primärschlüssel ist in manchen Fällen nicht so richtig begründet, weil oft eine Kombination aus dem jeweiligen Namen und eines weiteren Attributs, einer Entität, eindeutig beschreiben würde. Für das Einfügen der Daten, die von Mein Campus geliefert werden, ist die Einführung der Surrogate allerdings sehr gut geeignet. Mein Campus hat für ihre Tabellen eigene Surrogate, die somit gleich einlesbar sind. Für die Kompatibilität zwischen den manuell eingegebenen Daten und von Mein Campus gelieferten Daten kann dies zu weniger Fehlern führen, falls beim manuellen Eingeben von Daten im Namen Fehler passieren. Dieser Fall wäre für die Suche nach dem Namen wäre schlecht, aber die Fremdschlüsselbeziehungen würden funktionieren.

Die Entscheidung, die alten Studiengänge wie Diplom oder Magister nicht als extra Entität einzuführen, war angemessen, um Duplikate zu vermeiden. Viele Veranstaltungen werden nur noch für den Bachelor-Studiengang angeboten, und die alten Studiengänge müssen an diesen teilnehmen. Somit müssten die Lehrveranstaltungen einmal für die alten Studiengänge und einmal für die neuen Studiengänge eingetragen werden. Auf diese Weise sich würde der Pflegeaufwand erhöhen, weil die Lehrveranstaltungen manuell eingepflegt werden müssten.

Das multidimensionale Modell ist sehr ungewöhnlich in Bezug auf die Faktentabellen. Es gibt nur zwei Fakten, die eigene Attribute haben. Die anderen besitzen keine eigenen Attribute. Wie bereits im E/R-Diagramm sichtbar wurde, ist die Dimension DimOrg durch ihren Zusammenhang zu den verschiedenen Departments, Studiengängen und Modulen und den daraus definierte Entitäten sehr groß. Aus diesem Grund besitzen die anderen Dimensionen wenige Klassifikationsstufen. Hier wurde auf die Zugehörigkeit bei den Dimensionen geachtet, d.h. wer von wem abhängig ist.

Weiter kann man zum konzeptionellen Entwurf sagen, dass die Universitätsstruktur im Groben komplett abgedeckt ist. Es fehlen eher Kleinigkeiten, wie beispielsweise, dass Dozenten, die keinen Lehrstuhl angehören, an einem Pseudolehrstuhl arbeiten. Eine Lösung dieses Problems würde die Komplexität des E/R-Diagramm erhöhen, weil beispielsweise die Dozenten den Räumen zuordnet und die Räume den Lehrstühlen, wenn man diese Bindung nicht verlieren will. Der extra Aufwand und die geringe Zusatzinformation, rechtfertigen es nicht, diese Änderungen ein zuführen.

### 8.3 Systementwurf

Der Anfang des Systementwurfs beschäftigt sich als erstes mit dem Thema, ob eine relationale Basisdatenbank eingesetzt wird. Hier hätte man noch diskutieren können,

ob vielleicht eine objektorientierte Basisdatenbank die bessere Wahl gewesen wäre. Da bei der Umsetzung keine anderen Datenbanksysteme außer dem relationalen zur Auswahl standen, wurde hier nicht weiter auf andere Arten eingegangen.

Als kleiner Exkurs wird auf die Operational Data Store eingegangen, um zu klassifizieren, welches System für die Basisdatenbank entwickelt wird. Hier wird für die Einordnung ein kurzer Auszug aus dem Buch "Data Warehouse Systeme" gegeben. Es wird nur die Klassifikationsstufe beschrieben. Für nähere Informationen wird auf das Buch verwiesen. Durch die Klassifizierung der Basisdatenbank wird eine kleine Einführung gegeben, warum eine Basisdatenbank verwendet werden soll und dass Data Warehouse nicht direkt vom Produktivsystem beladen wird.

Für die Relationen textuell immer genau beschrieben, wie die Relationen vom E/R-Diagramm bzw. von multidimensionalen Schemata übersetzt werden. Dazu gibt es immer noch eine schematische Relationsbeschreibung. Dieser Prozess muss genau überwacht werden; es können hier schnell Fehler einschleichen, weil die Primärschlüssel fast alle mit dem Präfix id anfangen. So kann es leicht passieren, dass ein falscher Fremdschlüssel gesetzt wird.

Der Mein Campus-Webservice ist die zentrale Informationsquelle für das Projekt. Die Daten wären hier in einer Ausgabe ohne XML-ähnliche Syntax deutlich besser gewesen als in der angebotenen Form.

Selbstpflge der Daten ist eine zentrale, aber auch kritische Angelegenheit. In dem konzeptionellen Entwurf wurde am Anfang des Kapitels schon darauf hingewiesen, dass die Suche der Verantwortlichen durch eine nicht totale Partizipation dazu führen kann, dass die Verantwortlichen für Module, Teilleistungen und Studiengang nicht aktuell gehalten werden. Dies bedeutet mehr Aufwand, wenn ein Verantwortlicher gesucht wird; diese Anfrage taucht aber auch relativ selten auf. Diese Nachlässigkeit kann bei den Fachprüfungsordnungen nicht toleriert werden. Wird dies vergessen, sind keine genauen Anfragen für alle Ziele nicht mehr möglich. Dieser extreme Aufwand der Aktualisierung der Fachprüfungsordnungen sollte mindestens einmal jedes Jahr durchgeführt werden. Man sollte darauf hinwirken, dass Fachprüfungsordnungen ein einheitliches Format bekommen, was allerdings aus juristischen Gründen nicht einfach so möglich sein wird. Informationen über die Fakultät, das Department oder Dozenten können auch durch das Univis erlangt werden, welches dafür in Erwägung gezogen werden sollte. Hierfür müsste man sich mit den Betreibern des UnivIS. Dies würde auch einiges an manueller Arbeit sparen.

Der Ladeprozess besitzt eine festgeschriebene Reihenfolge, wenn man die Fremdschlüsselbeziehungen nicht nachträglich einpflegen will. Die Reihenfolge muss gelernt werden, was sehr negativ ist. Bei große Relationen wie Modul und Veranstaltung wäre die Pflege der Fremdschlüssel sehr aufwendig.

Die Analyse ist eine der schwierigsten Angelegenheiten des ganzen Projektes. Die Ziele der Studierendenberatung und des Auskunft Gebens an das Department sind eine exakte Angelegenheit, die durch SQL- bzw. MDX-Anfragen schnell und effizient gelöst werden können. Vorausgesetzt, die Basisdaten sind richtig aufgearbeitet. Bei der Abschätzung der Studierendenanzahlen in Lehrveranstaltungen wird es schon weitaus schwieriger. Hier liegen nie konkrete Daten vor. Selbst wenn eine Lehrveranstaltung abgeschlossen ist, liegen keine Daten vor, wie viele Studierende in der Lehrveranstaltung waren. Somit muss abgeschätzt werden. Die Analyse basiert dabei auf indirekten Werten. Die Analyse muss auch nach einem Jahr angepasst werden, wenn man mehr Erfahrung mit den Bachelor-Studiengängen hat. Der derzeitige Vorschlag ist nur auf die letzten zwei Jahren, als die Bachelor-Studiengänge eingeführt wurden, angepasst. Wenn sich der Aufbau nochmal ändert, könnte die jetzige Analyse hinfällig sein. Dies muss im Auge behalten werden. Jede Analyse muss sich mit dem Studierendenverhalten auseinandersetzen. Es gibt zwar Fakten wie die Prüfungsanzahl oder Jahrgangszahlen, aber das heißt beispielsweise nicht gleich automatisch, dass, wer geprüft wurde, eine Vorlesung besucht hat. Somit können nur Schätzwerte berechnet werden.

### 8.4 Technik

Es gibt viele verschiedene Datenbanksysteme und dazu auch verschiedene Data Warehouse-Systeme. Für den allgemeinen Test in Kapitel 7, wurden die vier Systeme ausgewählt, die gerade zur Verfügung standen. Die Tests, die durchgeführt wurden, sind sehr auf die Problematiken angepasst, die bereits bei der JasperSoft-Lösung aufgetaucht sind. Diese Lösung wurde vorgegeben. Somit konnten an Hand der Fehler die anderen Systeme getestet werden. Für allgemeine Vergleiche sollte dieser Test nicht herangezogen werden. Die an das Problem angepassten Tests machen einen Vergleich der Datenbanksysteme allerdings sehr gut möglich. Ein Problem kann nur sein, dass die speziellen Fehler der anderen Datenbanksysteme nicht auffallen werden.

Die Beschreibung über die Produkte von JasperSoft sind allesamt subjektiv nach der Installation und der Benutzung entstanden. JasperSoft-Produkte bieten von allen Produkten in Bezug auf die Basisdatenbank den besten Komfort, was die Entscheidung befürwortet, dieses Produkt zu verwenden. Doch stellen die Anwendungen für das Data Warehouse die Nerven auf eine große Probe. Fehlermeldungen, die nichts aussagen, und Programmführungen, die sich nach jeder neuen Version ändern, sollten abhalten, diese JasperSoft Data Warehouse-Lösung zu benutzen.

## 8.5 Ziele erreicht?

Für die Antworten wird vorausgesetzt, dass die Technik funktioniert und die Informationen im Data Warehouse-System stimmen.

Das Ziel, für die Beratung der Studierenden und die Berichterstattung an das Department Informatik Daten bereitzustellen, können exakt erfüllt werden. Es tauchen keine Abweichungen oder Unschärfen auf, wenn die Voraussetzung stimmt.

Das Ziel, die Studierendenzahl in Lehrveranstaltungen abzuschätzen, kann man nicht mit 100 Prozent erfüllen. Selbst wenn man aus den letzten Jahren die genaue Anzahl an Studierenden in Lehrveranstaltungen hätte, führt dies nicht dazu, dass man genaue Voraussagen treffen kann. Von Semester zu Semester ändern sich die Bedingungen für die Studierenden, und diese Bedingungen müsste man immer berücksichtigen, was einfach nicht geht. Somit ist eine genaue Vorhersage nicht möglich, nur eine annähernde.



## 9 Zusammenfassung

Prognosen von Studierendenzahlen für Lehrveranstaltungen aufzustellen ist jedes Semester ein neues Problem der einzelnen Departments der Friederich-Alexander-Universität Erlangen-Nürnberg. Dieses Problem soll durch eine Analyse der Studierendenzahlen in einem Data Warehouse erleichtert werden.

Als ersten Schritt der Umsetzung, mussten die Gründe und Abhängigkeiten untersucht werden, warum Studierende eine Lehrveranstaltung besuchen sollten. Diese Aussage lässt keine Rückschlüsse darauf zu, wie viele Studierende am Ende wirklich an einer Lehrveranstaltung teilnehmen. Die Gründe dafür und die damit zusammenhängenden Abhängigkeiten haben dazu geführt, dass eine Abbildung der Universitäts-Struktur nötig wurde, um zu verstehen, warum genau die Studierende eine Lehrveranstaltung besuchen. Hier raus entstand das Schema, dass die Realität abbildet. Beim aufstellen dieses Schemas fiel auf, dass durch die Abbildung der neuen (Bachelor bzw. Master) und alten (Magister bzw. Diplom) Studiengänge viel redundante Daten in die Datenbank stehen. Aus diesem Grund und dass die alten Studiengänge fast ausgelaufen sind, wurden nur noch die neuen Studiengänge betrachtet. Dies führte zu Abweichungen vom Schema der realen Welt.

Da es keine genaue Datensätze von Studierendenzahlen in Lehrveranstaltungen an der Universität gespeichert bzw. erfasst sind, müssen dies Informationen anderweitig ermittelt werden. Dies führt auch wieder zur eine Anpassung des Schema. Es blieb nichts anderes übrig auf Datensätze zurück zugreifen, die fast die Anzahl der Studierenden in Lehrveranstaltungen wieder spiegelt, beispielsweise Anzahl der Studierenden, die eine Prüfung in einem Fach ablegen. Vier verschiedene Datenquellen standen zur Auswahl, allerdings eigneten sich aufgrund von Aussagekraft und Genauigkeit nur zwei Datenquellen. Für die zwei nicht verwendeten Quellen ist es möglich, das konzeptionelle Schema anzupassen, damit die Informationen, der nicht benutzen Datenquellen, später eingebunden werden können.

Für die weitere Entwicklung des Projektes, stand die Wahl des zu benutzenden Datenbanksystems mit Data Warehouses an. Da nur relationale Datenbanksysteme zur Wahl standen und die Data Warehouse Systeme auf relationale Speicherung deren Tabellen ausgelegt waren, wurde das Schema der optimierten reale Welt auf Relationen umgesetzt. Gleichzeitig erfolgte die Schnittstellenspezifikation für die Datenquellen.

Um an die Informationen der Datenquellen zu gelangen, wurde ein Webinterface zur Verfügung gestellt.

Für die Umsetzung des Projektes, wurden die Anwendungen von JasperSoft vorgeschrieben. Allerdings haben diese Programme auch ihre Schwächen. Deshalb wurden andere Datenbankhersteller mit Data Warehouse Systemen getestet und verglichen. Dazu gehörten sowohl kommerzielle als auch Open Source Lösungen. Hierbei lagen die kommerziellen Systeme in Benutzerfreundlichkeit und Funktionalität vorne. Schlussendlich begann die Umsetzung des Projektes mittels JasperSoft. Die positiven und negativen Eigenschaften, die während der Umsetzung aufgetreten sind, wurden dargestellt.

Um die einzelnen Schritte der Umsetzung zu bewerten, wurde die einzelnen Schritte noch mal extra bewertet. Als Bewertungskriterien dienten die Verbesserungsmöglichkeiten. Dabei wurde nicht außer acht gelassen, welche Arbeitsschritte positiv verlaufen sind oder bei welchen Arbeitsschritten Probleme aufgetaucht sind.



# Anhang A

## Zusatzinformationen zu Kapitel ??

### A.1 Standortproblematik

Im Abschnitt ?? wurde die Problematik der Standortwechsel in einer Fakultät und zwischen Fakultäten angesprochen. Um diese Problematik zu veranschaulichen folgt ein Stundenplan der Buchwissenschaft (BuWi) aus dem dritten Semester (Bild ??) und ein Stundenplan der Linguistischen Informatik (LI) aus dem ersten Semester (Bild ??). Der Stundenplan der BuWi aus dem dritten Semester verdeutlicht den Standortwechsel an der Phil<sup>1</sup> Fakultät. Hier wird am Mittwoch zwischen zwei Standorten in der Innenstadt, Kollegienhaus und Bismarckstraße, und dem Erlanger-Süden gewechselt. Nebenbei sei erwähnt, dass die Studiengänge der Phil aus zwei Studienfächer bestehen und ein zweites Fach hier nicht aufgeführt wurde. Somit ist anzunehmen, dass ein weiterer Standortwechsel in einer Woche an der Tageordnung liegen. Bei dem 2-Fach-Informatik Bachelor ist dies garantiert, weil die Informatik im Süden von Erlangen liegt und jedes andere Fach, welches mit dem 2-Fach-Informatik-Bachelor kombiniert werden kann, in der Innenstadt statt findet.

### A.2 Umfrage an die Studierende bzgl. ihrem Lehrveranstaltungsbesuchverhalten

Um das Verhalten von Studierenden bzgl. der Regelmäßigkeit ihrer Besuche an Lehrveranstaltungen zu analysieren, wurde eine Umfrage an Studierenden durchgeführt die am Department Informatik Lehrveranstaltungen besuchen. Dabei sind ausschließlich Studierende befragt worden, die den Abschluss Bachelor erhalten werden. Bei der Umfrage haben 53 Studierende teilgenommen aus unterschiedlichen Studiengängen. Hier bei waren 24 Studierende aus dem Department Informatik befragt

---

<sup>1</sup>Philosophische Fakultät mit Fachbereich Theologie

worden. Der Rest nahm an den exportierten Lehrveranstaltungen des Departments Informatik teil.

1. Welches Studienfach studierst du?
2. Besuchst du regelmäßig alle Lehrveranstaltungen, die dir dein FPO vorschreibt?
3. Musst du zwischen den Lehrveranstaltungen die Standorte wechseln?
  - a) Welche Standorte betrifft es?
  - b) Ändert sich die Motivation Lehrveranstaltungen zu besuchen und wie wirkt sich dies aus?
4. Welche Gründe gibt es, warum du Lehrveranstaltungen besuchen bzw. nicht besuchen würdest?

# Anhang B

## Anleitungen

Beschreibung der verwendeten Zusatzprogramme für die Umsetzung des Projektes.

### B.1 MySQL Administrator - Remote Zugriff einrichten

Wenn der Datenbankserver nicht direkt auf dem Computer / Server installiert ist von dem aus der MySQL Administrator benutzt wird, muss als erstes ein Administratorkonto vom Datenbankserver für die Verwaltung von einem Fremdcomputer frei geben werden.

```
mysql> GRANT ALL PRIVILEGES ON *.* TO 'monty'@'localhost'  
-> IDENTIFIED BY 'some_pass' WITH GRANT OPTION;  
mysql> GRANT ALL PRIVILEGES ON *.* TO 'monty'@'%'  
-> IDENTIFIED BY 'some_pass' WITH GRANT OPTION;  
mysql> GRANT RELOAD,PROCESS ON *.* TO 'admin'@'localhost';  
mysql> GRANT USAGE ON *.* TO 'dummy'@'localhost';
```

Das "%" nach dem "@" bedeutet, dass der Benutzer monty von jedem Rechner aus dem angeschlossenen Netzwerk auf den Rechner zugreifen darf.

Die Installation vom MySQL Workbenche läuft ohne Probleme ab, wenn man die Bildschirmansagen durch geht. Wenn man das Programm installiert hat startet man MySQL Administrator und baut eine Verbindung zum Datenbankserver auf. Das Fenster verlangt per Promteingabe Adresse, Port, Benutzername und das dazugehörige Passwort.

## B.2 Laden der Basisdatenbank mit JasperETL

Neben dem Laden der Basisdatenbank mit dem von MySQL zur Verfügung gestellten MySQL-Loader kann man auch das Produkt JasperETL verwenden.

Nach dem Download von der Webseite, muss JasperETL nur noch entpackt und gestartet werden. Eine Installation ist nicht notwendig. Nach dem Starten muss man ein neues Projekt erstellen mit Namen und mit welcher Programmiersprache man arbeiten möchte. Es stehen Perl und Java zu Auswahl. Nach der Erstellung eines neuen Projekts wird nach einer E-Mailadresse für die Registrierung gefragt. Man kann eine imaginäre Adresse angeben.

Im Programm sollte man dann als erstes einen Businessplan erstellen. Dieser dient dem Zweck, seinem Chef, Team oder jemand anderen, der für diese Projekt arbeitet oder verantwortlich ist zu zeigen, was wie erstellt werden soll. Der Businessplan hat für den eigentlichen ETL-Prozess keine Auswirkung.

Im nächsten Schritt wird dann eine Job-Design erstellt. Dafür muss zuerst eine Datenbankverbindung etabliert werden. Dafür klickt man auf Metadata (am linken Rand) und dann auf "Db Connections". Mit Rechtsklick erstellt man eine neue Verbindung zu einem Datenbanksystem, in dem man alle Daten zu der Netzverbindung und der eingesetzten Datenbank angegeben hat. Wenn das neue Symbol der Datenbank auftaucht klickt man mit der rechten Maustaste auf das Symbol und ermittelt das Schema der Datenbank. Nach diesem Schritt kann man wieder zum Job-Design zurück kehren. Nach dem Öffnen eines neuen Jobs erscheinen auf der rechten Seite Symbole, die für bestimmte Operationen und für verschiedene Prozesse eingesetzt werden können. Jetzt gibt es unterschiedliche Möglichkeiten des Datenspeicherungsformat in dem unsere Daten sein können. Entweder sind die Daten schon in einer anderen Datenbank, in einem Excel-File oder in einer CSV-Datei. Die Bilderstrecke unten zeigt alle drei Möglichkeiten. Ausgangs Objekt

Egal woher wir die Daten nehmen, wir brauchen eine Datenbank, in die wir die extrahierten Informationen speichern. Dies ist eine Tabelle einer Datenbank, zu der wir oben eine Verbindung aufgebaut haben. Als erstes sucht man auf der rechten Seite des Programmfensters, den Reiter "Databases". Wenn man darauf geklickt hat, tauchen neue Unterordner auf, die beschriftet sind mit Datenbankherstellernamen. Jetzt muss man sich den Hersteller heraus suchen, der das Datenbanksystem vertreibt, in dem wir die Informationen abspeichern wollen. Auf dem Bildschirm werden jetzt die Symbole angezeigt, wofür es unterschiedliche Operationen gibt. Wir wählen nun von MySQL tMySQLOutput aus und setzen es in das JopDesign - Fenster. Mit Doppelklick kommen wir auf die Eigenschaften des Ouput-Operation. Hier wählt man jetzt die Datenbankverbindung aus und die Tabelle in die die Operationen gespeichert

werden. Nach dem wir jetzt die Speicher-Datenbank ausgewählt haben brauchen wir unsere Eingabe.

### **B.2.1 Input: Excel File**

Wir gehen zum rechten Bildschirmrand und klicken auf "File", dabei erscheint unter den Symbolen das tFileInputExcel. Dieses zeihen wir nun in das JobDesign. Mit Doppelklick klicken wir auf das Symbol, um an die Eigenschaften zu kommen. Nun erscheint unten im Fenster ein neues Menü. Hier muss man als erstes den Pfad und den Dateinamen angeben, an dem die Datei liegt. Dabei ist der lokale PC gemeint. Danach muss man die Arbeitsmappe in der Excel Datei auswählen. Danach kann man noch unterscheiden, ab wann Kopf- und Fußzeilen anfangen bzw. enden und ab wann und bis zu welcher Zeile die Daten eingelesen werden sollen. Um sich die Arbeit zu erleichtern, kann man unten noch zwischen "Build inüüd "Repository"wählen. Wenn man "Repository"wählt, und dann die Tabelle der Datenbanktabelle auswählt, in der man die Informationen speichern will, kann man gleich die Spalteninformationen mit dem Input abgleichen.

### **B.2.2 Input: CSV File**

Wir gehen zum rechten Bildschirmrand und klicken auf "File", dabei erscheint unter den Symbolen das tFileDelimited. Dieses zeihen wir nun in das JobDesign. Mit Doppelklick klicken wir auf das Symbol, um an die Eigenschaften zu kommen. Nun erscheint unten im Fenster ein neues Menü. Man kann nun einstellen wie sich die Operation verhalten soll. Als erstes muss man unterscheiden, ob die eingelesene Datei auch eine CSV - Datei ist. Wenn ja, sollte man diese Option bestätigen. Jetzt wählt man den Pfad und den Dateinamen, an dem die Datei liegt. Dabei ist der lokale PC gemeint. Zum Schluss werden die Zeilen- und Spaltenterminatoren angeben.

### **B.2.3 Input: DBS**

Als erstes sucht man auf der rechten Seite des Programmfensters den Reiter "Databases". Wenn man darauf geklickt hat, tauchen neue Unterordner auf, die beschriftet sind mit Datenbankherstellernamen. Jetzt muss man sich den Hersteller aussuchen, der das Datenbanksystem erstellt hat, in dem wir die Informationen abspeichern wollen. Auf dem Bildschirm werden jetzt die Symbole angezeigt, wofür es Operationen gibt. Wir wählen nun tMySQLInput aus und setzen es in das JobDesign - Fenster. Mit Doppelklick kommen wir auf die Eigenschaften des Input-Operation. Hier wählt

man jetzt die Datenbankverbindung aus und die Tabelle in die die Operationen herausgelesen werden.

### **B.2.4 Verbindung**

Wenn dies passiert ist, muss noch eine Verbindung von den Inputsymbolen zu der Outputdatabase erstellt werden. Dafür klickt man mit Rechts auf das Inputsymbol. Dann erscheint ein neues Menü, bei dem wir auf Zeile klicken und dann auf Main und ziehen die Line auf das Outputsymbol. Nun ist die Verbindung hergestellt. Schlussendlich geht man unten auf den Reiter SStartünd betätigt den Button SStart”.

## **B.3 Laden des Data Warehouses mit JasperETL**

Nach dem Download von der Webseite[Cor09a], muss JasperETL nur noch entpackt und gestartet werden. Eine Installation ist nicht notwendig. Nach dem Starten muss ein neues Projekt erstellt werden mit Namen und mit welcher Programmiersprache man arbeiten möchte. Es stehen Perl und Java zu Auswahl. Nach der Erstellung eines neuen Projekts wird nach einer E-Mailadresse für die Registrierung gefragt. Man kann eine imaginäre Adresse angeben.

Im Programm sollte man dann als erstes einen Businessplan erstellen. Dieser dient dem Zweck, seinem Chef, Team oder jemand anderes der für diese Projekt arbeitet oder Verantwortlich ist zu zeigen, was wie erstellt werden soll. Der Businessplan hat für den eigentlichen ETL-Prozess keine Auswirkung.

Im nächsten Schritt soll dann eine Job-Design erstellt werden. Dafür muss erst mal eine Datenbankverbindung etabliert werden. Wenn beim Laden der Basisdatenbank schon JasperETL-Verwendet hat, brauch man gegenenfalls keine neue Datenbankverbindung. Ist dies der Fall, dann kann man mit dem nächsten Abschnitt weiter arbeiten. Dafür klickt man auf Metadata (am linken Rand) und da auf ”Db Connections”. Mit Rechtsklick erstellt man eine neue Verbindung zu einer Datenbank, in dem man alle Daten zu der Netzverbindung und der eingesetzten Datenbank angegeben hat. Wenn das neue Symbol der Datenbank auftaucht, klickt man mit der rechten Maustaste auf das Symbol und ermittelt das Schema der Datenbank. Nach diesem Schritt kann man wieder zum Job-Design zurück kehren.

Nach dem Öffnen eines neuen Jobs erscheint auf der rechten Seite Symbole, die für bestimmte Operationen für verschiedene Prozesse eingesetzt werden können. Wir suchen bei den Symbolen die ELT Symbole. Nicht wundern über den Namen ELT, Japsper hat nur die Reihenfolge der Wörter umgedreht. Angekommen bei den

ELT Operationen, wählt man nun die Operation für die eingesetzte Datenbank aus. Achtung, die ELT Operationen können mit keinen anderen Operation vermischt werden, auch nicht mit anderen ELT Operationen für andere Datenbankanbieter.

Als erstes setzt man die Inputsymbole. Ein Inputsymbol steht für eine Tabelle aus dem Basisdatenstamm. Je nach Ladeprozess sind 1 bis N Inputsymbole setzbar. Nach dem ein Inputsymbol gesetzt wurde, muss unten im Component Reiter noch von "Built in auf "Repository" gewechselt werden. Dies bedeutet, dass man jetzt auf die oben angelegte Datenbank zugreifen möchte und eine bestimmte Tabelle wählt.

Jetzt braucht man ein ELTMapper für den Prozess. Man wählt diesen auch von dem Menü auf der rechten Seite, passend zu der Datenbank der gesetzten Inputsymbole. Ist der ELTMapper gesetzt, muss man bei ihm auch von "Built in auf "Repository" wechseln und dabei die Inputdatenbankschema wählen.

Zum Schluss braucht man noch die Outputsymbole, die einem die Datenbank angeben, in die das Endprodukt des Mappers speichert werden soll. Auch hier wieder, dass passende Symbol des Datenbankanbieters wählen, wie für den ELTMapper und dem Inputsymbol. Nun wechselt man im Reiter "Component" von "Built in auf "Repository" und wählt die gewünschte Tabelle aus. Ich gehe hier davon aus, dass die Tabelle schon in der Datenbank angelegt ist.

Um Verbindungen zwischen Input, Mapper und Output zu erstellen, klickt man dazu nun auf Input- bzw. Mappersymbol mit rechts und da auf Link, was einem dann nur nach "new Output (Table)" führt. Wichtig ist, dass beim Inputsymbol der exakte Name der Inputtabelle gewählt wird und beim Mapper der exakte Name der Outputtabelle, die jeweils in der Datenbank vorhanden sind. Passiert dies nicht, wird es bei einem "Run" zu Metadaten-Problemen führen.

Jetzt kümmern wir uns um den Mapper und klicken mit Doppelklick auf den Mapper. Es öffnet sich ein Fenster, dessen rechte Seite ein Fenster enthält, das die Tabellenelemente der Zieldatenbank enthält. Es werden zusätzlich die Typen und die Primärschlüssel angegeben. Auf der linken Seite kann man mit dem Pluszeichen die verbundenen Inputtabellen hinzu fügen. Ein Join in mehrere Tabellen realisiert man, indem man ein Element einer Inputtabelle auf eine andere Inputtabelle zieht. Dabei kann man dann noch unterscheiden welche Operation darauf ausgeführt werden soll, beispielsweise "oder" oder "■". Ein einfacher Weg ist jetzt den Knopf "Auto map" zu drücken. Hierbei versucht JasperETL mittels Namensgleichheit die Verbindungen zu ziehen. Verbindungen werden als gelbe Linien zwischen den Inputtabelle(n) unter Outputtabelle angezeigt. Wenn man der Automatik nicht vertraut oder sie nicht korrekt arbeitet, kann man von Elementen der Inputtabelle(n) zu der Outputtabelle manuell überziehen. Es wird dann eine Verbindungslinie gezogen. Sehr wichtig ist, dass man darauf achtet, dass die Typinformation in der Inputtabelle mit den Typinformationen der Outputtabelle übereinstimmen. Dies kann man unten im Fenster vergleichen. Achtung,

wenn man hier etwas ändert, ändert man dies auch in der Datenbank. Nun kann man das Fenster schließen.

Jetzt geht man auf den unteren Reiter SStarteüind drückt start, nun wird der Prozess gestartet. Man kann mehrere Tabellen in einem Job beschreiben. Ich rate davon ab, wenn ein Job einen Fehler hat werden alle nicht durchgeführt und die Fehler geben auch keinen Anhaltspunkt welcher Mapper von einem Fehler betroffen ist.

## B.4 Pentaho Schema Workbench, Schema Erstellung

Das Programm Schema Workbench ist das einzige Programm, das nicht von Jasper-Forge selbst angeboten wird. Allerdings wird das Endprodukt, ein JRXML-File, das die Kuben beschreibt, für den JasperServer benötigt, für die Multidimensionalen Anfragen. Für einfache SQL Anfragen wird diese Darstellung nicht benötigt.

Anmerkung: Vorab, das Wort Kubus wird hier nicht gleich gesetzt mit drei Dimensionen! Schema Workbench wird von der Firma Pantaho hergestellt. Das Programm kann man hier herunterladen. Man sollte bedenken, dass die Software ohne Java-Treiber für Datenbanken geliefert wird. Um dieses Problem zu lösen, lohnt es sich noch ein anderes Programm von Pantaho herunter zu laden und den Treiber in den Unterordner von Schema Workbench mit dem Namen "Driverßu kopieren. JDBC -Treiber

Nach den Vorbereitungen kann man das Programm starten. Der erste Schritt ist, auf "Toolsßu klicken und da auf "Preferences". Hier gibt man jetzt den JDBC-Treiber, den Netzwerkpfad mit Port und Datenbankschema z. B. Fix, Benutzername und Passwort ein. Die Verbindung zur Datenbank ist nicht notwendig, aber erleichtert die Angabe der Tabellen und das Programm überprüft, ob sie überhaupt in dem Datenbanksystem existieren und ob die Spaltentypen übereinstimmen.

### B.4.1 Dimension

Zunächst muss man sich die Faktentabellen anschauen. Gibt es für mehrere Kuben, bei denen die Dimensionen gleich sind. Wenn ja, sollte man sie außerhalb der Kuben definieren und dann nur in den Kuben verwenden. Um eine Dimension zu definieren, klick man auf das Symbol "Add Dimension". Nun wurde auf der linken Navigationsseite ein Symbol dazu gefügt, welche den Unterpunkt "Hierarchy", der wiederum den Unterpunkt Table hat. Direkt unter der Dimension kann man unterschiedliche parallele Hierarchy aufbauen. Wie das klassische Zeitbeispiel: Man baut die Granularität Jahr, Montag, Tag auf oder Jahr und Wochen. Die Granularitäten in einer Hierarchy setzt



man mit dem Button **Add level** in die oberen Werkzeugleiste. Jedes Granulat muss zu einem Attribut in der Tabelle gehören.

## B.4.2 Table

Zu jeder Hierarchie gehört auch noch eine Tabelle oder ein Join aus Tabellen. Bei einer Tabelle muss ein Tabellennamen angegeben werden. Wenn eine Verbindung zu einem Datenbanksystem steht, gibt das System eine Auswahlmöglichkeit der vorhandenen Tabellen in dem Datenbankschema an.

## B.4.3 Join

Um ein Join zu realisieren, muss man mit Doppelklick auf das Tabellensymbol klicken. Danach erscheinen die Auswahlmöglichkeiten **Table** oder **Join**. Bei **Table** geht man vor wie oben, bei Auswahlmöglichkeit **Join** erscheint ein Submenu. Wenn diese ausgeklappt wird, erscheinen zwei neue Tabellensymbole, die wie im Abschnitt davor beschrieben, bestückt werden können. Beim Joinsymbol muss man von beiden Tabellen noch die Attribute auswählen, worüber der Join ausgeführt werden soll. Man kann mehrere Joins hintereinander angeben. Kubus erstellen

Zum Erstellen eines Kubus, klickt man auf **add cube**. Gleich wird ein Symbol für ein Kubus mit Unterordner, in der sich eine Tabelle befindet, erzeugt. Diese Tabelle wird für die Faktentabelle benötigt. Jetzt hat man zwei Möglichkeiten: Entweder man definiert für den Kubus eine eigene Dimension, dies geschieht, wenn man den Button **add dimension** in dem Kubussymbol drückt ( vorgehen wie oben), oder man benutzt eine Dimension, die man vorab definiert hat. Für die zweite Variante, benutzt man den **add Dimension usage** aus der Kopfleiste. Bei dieser Vorgehensweise muss bei dem Symbol unter **SSource** die globale Dimension angegeben werden. Weitere Funktionen

Zusätzlich kann man noch weitere aggregierte Messfelder angeben für Wert. Beispielsweise summiert oder aufgezählt. Weiterhin kann man auch noch Namensräume festlegen. Auch hat man die Möglichkeit Benutzer definierte Funktion zu schreiben, die dann allerdings in Java geschrieben werden müssen und in die class Datei eingebunden werden muss.



# Anhang C

## Dokumentation

Informationen für die Abfrage am Webservice, der Schnittstelle zu Mein Campus.

### C.1 Zuordnung Identifikationsnummern zur Abschlussart

Kirchl. Prüfung	04
Staatsexamen	08
Staatsexamen (S)	09
Diplom Univ.	11
Master Comp. Eng.	12
Master Univ.	13
Bachelor Univ.	14
Bakkalaureus	15
Master/Orient.Sem.	16
Comput. Eng.	17
LA Hauptschulen	22
LA Realschulen	23
LA Gymnasium	25
LA Berufsschulen	27
ERW.PRF. Grundschulen	31
ERW.PRF. Realschulen	33
ERW.PRF. Gymnasien	34
ERW.PRF. Berufsschulen	36
Ergänzungsprüfung	38
Zertifikat Grundschulen	41
Zertifikat Hauptschulen	42
Zertifikat Realschulen	43
Zertifikat Gymnasien	45

Abschluß nicht FAU 90  
sonst. Ergänzungsprüfung 93  
Abschluß im Ausland 96  
keine Abschlußprüfung 97  
Teilstudienplatz 98  
Magister/Aufbaustudium 92  
Magisterprüfung 02  
Master of Science w. H. 67  
Promotion 06  
LA Grundschulen 21  
LA Grundschulen NEU 71  
LA Hauptschulen NEU 72  
LA Realschulen NEU 73  
LA Gymnasium NEU 75  
LA Berufsschulen NEU 77  
ERW.PRF. Hauptschulen 32  
Pool 99  
Erwpr Gymnasien-neu 85  
Erwpr Grundschulen-neu 81  
Erwpr Hauptschulen-neu 82  
Erwpr Realschulen-neu 83  
Erwpr Berufsschulen-neu 87  
B.Sc. 55  
Magisterprüfung (2HF) 03  
B.A. (1 Fach) 51  
Bachelor of Education 56 Master of Arts 61  
Master of Science 65  
Bachelor of Arts (2 Fäch)52

## **C.2 Zuordnung Identifikationsnummer zum Studiengang**

996 Uni Bayreuth  
997 Akademie Nürnberg  
998 FHS Nürnberg  
999 Uni Bamberg  
598 Christliche Archäologie  
290 Fak. fremdes Nebenfach

207 Medizin Klinik  
303 Humanbiologie  
307 Medizin (PJ)  
108 Medizinische Physik  
185 Zahnmedizin  
132 Psychologie  
147 Sozialkunde  
184 Wirtschaftswissenschaften  
561 Alte Geschichte  
562 Mittlere Geschichte  
563 Ur- und Frühgeschichte  
565 Bayerische und fränkische Landesgeschichte  
567 Osteuropäische Geschichte  
572 Didaktik der Sozialkunde  
573 Didaktik der Geschichte  
608 Buchwissenschaft  
570 Neuere und Neueste Geschichte  
532 Psychogerontologie  
081 Indoiranistik  
084 Italienisch  
085 Japanologie  
095 Latein  
122 Semitische Philologie  
574 Assyriologie  
575 Anglistik  
576 Galloromanische Philologie  
577 German.u.Deutsche Philologie  
578 Iberoromanische Philologie  
579 Indogermanistik  
589 Didaktik der französischen Sprache und Literatur  
590 Didaktik der englischen Sprache und Literatur  
592 Didaktik der deutschen Sprache und Literatur  
593 Griechische Philologie  
594 Lateinische Philologie  
596 Lateinische Philologie des Mittelalters  
604 Vergleichende Literaturwissenschaft  
605 Mittel- u. Neulatein  
606 Slavische Philologie  
607 Neuere deutsche Literaturwissenschaft  
551 Anglistik: Literaturwiss.  
584 Nordische Philologie  
555 Amerikan.: Literaturwiss.

556 Angl./Amerik.:Kulturwiss.  
582 Neuere dt. Lit.geschichte  
128 Physik  
118 Technomathematik  
276 Wirtschaftsmathematik  
126 Pharmazie  
400 Molecular Science  
559 Didaktik der Geographie  
595 Geographie  
597 Kulturgeographie und Area Studies  
148 Sozialwissenschaften  
181 Wirtschaftspädagogik  
182 Internationale Betriebswirtschaftslehre (International Business)  
183 Internationale Betriebswirtschaftslehre (International Business)  
277 Wirtschaftsinformatik  
530 Kommunikationswissenschaft  
609 Internationale Wirtschafts- und Entwicklungspolitik  
202 Fertigungstechnik  
248 Elektrotechnik, Elektronik und Informationstechnik  
250 Informatik/Wirtschaftsinformatik  
233 Chemie- und Bioingenieurwesen  
204 Mechatronik  
348 Elektrotechnik u. Informationstechnik  
178 Advanced Materials and Processes  
249 Systeme der Informations- und Multimediatechnik  
011 Arbeitslehre  
090 Hauptschuldidaktiken  
091 Kunst  
098 Sport  
361 Qualif.des Beratungslehrers  
533 Didaktik der Arbeitslehre  
560 Didaktik des Deutschen als Zweitsprache  
566 Landes- und Volkskunde  
601 Sportpädagogik  
603 Musikpädagogik  
602 Didaktik der bildenden Kunst  
116 Grundschulpädagogik  
152 Religionspädagogik und Did. d. ev. Rel.unterrichts  
088 Didaktik des kath. Religionsunterrichts  
134 Psychologie  
549 Ethik der Textkulturen  
801 Zulassungsfreies Fach 1

802 Zulassungsfreies Fach 1

625 Kulturgeographie

626 Life Science Engineering

569 Wirtschaftswissenschaften / Staatswissenschaften

136 Internationales Wirtschaftsrecht

179 Wirtschaftsingenieurwesen

300 Molekulare Medizin

039 Geowissenschaften

611 Kulturgeographie (Zweifach)

612 American Literary and Cultural Studies: USA, Canada and the Caribbean





# Literaturverzeichnis

- [AB01] ANDREAS BAUER, Holger G. ; ZIMPFER, Ursula (Hrsg.): *Data Warehouse Systeme Architektur Entwicklung Anwendung*. dpunkt.verlag, 2001
- [AK06] A. KEMPER, A. E. ; ROTH, Margit (Hrsg.): *Datenbanksystem Die Einführung*. Oldenburg, 2006
- [Cor09a] CORPORATION, JasperSoft: *JasperServer Informationswebseite*. JasperSoft Corporation. <http://www.jaspersoft.com/>. Version: Februar 2009
- [Cor09b] CORPORATION, Jupitermedia: *MDX Tutorial*. Jupitermedia Corporation. <http://www.databasejournal.com/features/mssql/article.php/1495511/MDX-at-First-Glance-Introduction-to-SQL-Server-MDX-Essentials.htm>. Version: Februar 2009
- [ECL09] ECLIPSECON: *Eclipse*. ECLIPSECON. <http://www.eclipse.org/>. Version: Februar 2009
- [eG09] eG, Config I.: *UnivIS Informationswebseite*. Config Informationstechnik eG. <http://www.config.de/UnivIS/>. Version: Februar 2009
- [EN] ERLANGEN-NÜRNBERG, Friedrich-Alexander-Universität: *Modulpläne aller Studiengänge der Universität Erlangen-Nürnberg*. Friedrich-Alexander-Universität Erlangen-Nürnberg. <http://www.cit.uni-erlangen.de/dokumente-und-materialien/bachelorstudiengaenge>
- [EN09a] ERLANGEN-NÜRNBERG, Friedrich-Alexander-Universität: *Campus IT Informationswebseite*. Friedrich-Alexander-Universität Erlangen-Nürnberg. <http://www.cit.uni-erlangen.de/>. Version: Februar 2009
- [EN09b] ERLANGEN-NÜRNBERG, Friedrich-Alexander-Universität: *Evaluationssystem der Friedrich-Alexander-Universität Informationswebseite*. Friedrich-Alexander-Universität Erlangen-Nürnberg. <http://eva.uni-erlangen.de/>. Version: Februar 2009
- [EN09c] ERLANGEN-NÜRNBERG, Friedrich-Alexander-Universität: *Fachprüfungsordnungen der Friedrich-Alexander-Universität Erlangen-*

- Nürnberg. Friedrich-Alexander-Universität Erlangen-Nürnberg. <http://www.uni-erlangen.de/universitaet/organisation/recht/studiensatzungen/>. Version: Februar 2009
- [EN09d] ERLANGEN-NÜRNBERG, Friedrich-Alexander-Universität: *Herkunft der Studierenden an der Universität Erlangen-Nürnberg*. Friedrich-Alexander-Universität Erlangen-Nürnberg. [www.uni-erlangen.de/universitaet/statistik/studierende/bundeslaendern.pdf](http://www.uni-erlangen.de/universitaet/statistik/studierende/bundeslaendern.pdf). Version: Februar 2009
- [EN09e] ERLANGEN-NÜRNBERG, Friedrich-Alexander-Universität: *Informatik Modulplan*. Friedrich-Alexander-Universität Erlangen-Nürnberg. [www.cit.uni-erlangen.de/fileadmin/CIT/ModelleBA/20070823\\_BSc\\_Informatik.png](http://www.cit.uni-erlangen.de/fileadmin/CIT/ModelleBA/20070823_BSc_Informatik.png). Version: Februar 2009
- [EN09f] ERLANGEN-NÜRNBERG, Friedrich-Alexander-Universität: *Mein Campus Informationswebseite*. Friedrich-Alexander-Universität Erlangen-Nürnberg. <https://www.campus.uni-erlangen.de/>. Version: Februar 2009
- [EN09g] ERLANGEN-NÜRNBERG, Friedrich-Alexander-Universität: *StudOn Informationswebseite*. Friedrich-Alexander-Universität Erlangen-Nürnberg. <http://www.studon.uni-erlangen.de/studon/>. Version: Februar 2009
- [EN09h] ERLANGEN-NÜRNBERG, Friedrich-Alexander-Universität: *Universitäts Struktur*. Friedrich-Alexander-Universität Erlangen-Nürnberg. <http://www.uni-erlangen.de/universitaet/>. Version: Februar 2009
- [ER01] ERHARD RAHM, Theo H. ; ERHARD RAHM, Theo H. (Hrsg.): *Datenbanksysteme Konzepte und Techniken der Implementierung*. Bd. Auflage 4. Springer-Verlag, 2001. – 4–8 S.
- [Gmb09a] GMBH, IBM D.: *IBM DB2 Informationswebseite*. IBM Deutschland GmbH. <http://www-01.ibm.com/software/de/data/>. Version: Februar 2009
- [Gmb09b] GMBH, IBM D.: *IBM Infosphere Informationswebseite*. IBM Deutschland GmbH. <http://www-01.ibm.com/software/de/data/infosphere/>. Version: Februar 2009
- [Gmb09c] GMBH, Microsoft D.: *Microsoft SQL Server Informationswebseite*. Microsoft Deutschland GmbH. <http://www.microsoft.com/germany/sql/2008/default.aspx>. Version: Februar 2009
- [Gmb09d] GMBH, Microsoft D.: *Microsoft VISIO Informationswebseite*. Microsoft Deutschland GmbH. <http://office.microsoft.com/de-de/visio/FX010857981031.aspx>. Version: Februar 2009

- [Hei02] HEINRICHS h.: *Datenqualitätsmanagement in Data Warehouse-Systeme*, Carl-von-Ossietzky-Universität Oldenburg, Diss., 2002
- [Int09] INTERNATIONAL, Citadel: *Pentahol Informationswebseite*. Citadel International. <http://www.pentaho.com/>. Version: Februar 2009
- [MH05] MARTIN HEINZ, Elisabeth Kapsammer und Werner R. Gerti Kappel K. Gerti Kappel ; AUFLAGE, 3. (Hrsg.): *UML Work Objektorientierte Modellierung mit UML 2*. dpunkt.verlag, 2005
- [MyS09] MYSQL: *MySQL-Downloadwebseite*. MySQL. <http://dev.mysql.com/downloads/>. Version: Februar 2009
- [Par09] PARADIGM, Visual: *Visual Paradigm Informationswebseite*. Visual Paradigm. <http://www.visual-paradigm.com/>. Version: Februar 2009
- [SBN02] SHAMKANT B. NAVATHE, Ramez E. ; ANTJE KLATT, Helge S. (Hrsg.): *Grundlagen von Datenbanksystem*. Bd. 3. Pearson Studium, 2002
- [UEN09] UNIVERSITÄT ERLANGEN-NÜRNBERG, Wirtschafts und Sozialwissenschaftliche Fakultät d.: *Entrichtung und Verwendung der Studienbeiträge an der Friedrich-Alexander-Universität Erlangen-Nürnberg*. Friedrich-Alexander-Universität Erlangen-Nürnberg. [http://www.soziologie.wiso.uni-erlangen.de/publikationen/berichte/b\\_08-01.pdf](http://www.soziologie.wiso.uni-erlangen.de/publikationen/berichte/b_08-01.pdf). Version: Februar 2009