



Arbeitsplan-Generierung für Sportinstruktoren am Institut für Sporterziehung und Sport

Seminararbeit

eingereicht bei
Prof. Dr. Andreas Meier
Betreuer: Daniel Wismer
Lehrstuhl für Wirtschaftsinformatik
Wirtschafts- und Sozialwissenschaftliche Fakultät
Universität Fribourg

von
Roger Nufer

Meierhofstrasse 15
CH-8108 Dällikon
076 525 98 29
roger.nufer@unifr.ch
Studentennr.: 00-202-341

Studienrichtung: Wirtschaft
4. Semester
Februar 2002

1 Einleitung

1.1 Problemstellung

Ist-Zustand:

Als Ausgleich zum akademischen Alltag, bietet die Universität Fribourg ein breitgefächertes Programm sportlicher Aktivitäten an. Verschiedene Kurse, zu verschiedenen Tageszeiten und an verschiedenen Orten werden von ausgebildeten Instruktoren betreut und geleitet. Das Institut für Sporterziehung und Sport der Universität ist Organisator dieser Kurse. Jeder

Instruktor führt Buch über seine geleisteten Arbeitsstunden und vermerkt diese auf einem vorgedruckten Arbeitsreport, welchen er Ende Semester beim Institut einreicht. Dieses nimmt die Lohnabrechnung vor und übergibt diese schliesslich der Universitätsverwaltung, welche die Transaktion in Gang setzt. Bis anhin verursachte vor allem die Erstellung dieser vorgedruckten Arbeitsrapporte erheblichen Aufwand. Jeder einzelne Rapport wurde auf dem Excel konzipiert und als einzelnes Excel Dokument abgespeichert. Alle Kurse, welche sich durch eines der Merkmale Sportart, Tag, Zeit, Ort oder Instruktor unterscheiden, erhielten ihren eigenen Arbeitsreport. So kamen 120 einzeln gespeicherte Excel Dateien zusammen.

SEMESTRE D'HIVER 2001 - 2002

RAPPORT D'ACTIVITE

Discipline :
 Entraîneur :
 Jour : vendredi
 Horaire :
 Durée :

Nbre d'heures à payer :

Dates	Participants	Remarques
26.10.01		
02.11.01		
09.11.01		
16.11.01		
23.11.01		
30.11.01		
07.12.01		
14.12.01		
21.12.01		
11.01.02		
18.01.02		
25.01.02		
01.02.02		
08.02.02		
TOTAL :		Moyenne participants/heure :

N.B. Si vous faites remplacer, veuillez indiquer par qui.
 Si vous ne donnez pas le cours, expliquez pourquoi.

VOEUX, SUGGESTIONS :

Abb. 1: Vorlage Arbeitsreport

Jedes neue Semester mussten die Rapporte modifiziert werden und allfällige Änderungen im Personalbereich sowie veränderte Kursanordnungen von Hand abgeändert werden. Am meisten Aufwand jedoch bereitete die laufende Anpassung der Semesterdaten. Für jeden Wochentag wurde eine Liste mit den relevanten Semesterdaten, an welchen der Kurs stattfand, erstellt und danach mit "copy-paste" in den einzelnen Rapporten eingefügt.

1.2 Zielsetzung

Ziel dieser Seminararbeit ist es, eine relationale Datenbank zu entwerfen und zu implementieren. Diese erlaubt dann, die Arbeitsrapporte einfacher und mit geringerem Aufwand zu erstellen und zu modifizieren. Das Hauptziel besteht aus der Minimierung des Arbeitsaufwandes. Die Ausbaufähigkeit der Datenbank und die Minimierung des benötigten Speicherplatzes sind Unterziele.

1.3 Vorgehensweise

In einer Datenanalyse werden zunächst alle vorhandenen Daten aufgelistet und die Beziehungen dieser Daten untereinander erläutert. In einem nächsten Schritt wird durch das Entitäten-Relationen Modell (ERM) die Grundstruktur der künftigen Datenbank definiert. Abschliessend wird das ER-Modell mit Hilfe von Transformationsregeln in ein relationales Datenbankschema überführt und anhand der Normalformen kontrolliert. Die Implementation wird mit der Datenbanksoftware FileMaker Pro 5.5 durchgeführt.

2 Datenanalyse

Es bestehen folgende Informationssachverhalte:

Insgesamt sind 67 Instruktoren am Institut für Sporterziehung und Sport angestellt. Diese Instruktoren teilen sich auf 42 Disziplinen auf. Wesentlich für die Arbeitsrapporte sind lediglich die Disziplin, der Instruktor mit Name und Vorname, der Tag samt Semesterdaten, an welchen der Kurs stattfindet, die Zeit und Dauer und schliesslich der Ort.

Die Beziehungen zwischen den vorhandenen Daten lassen sich wie folgt zusammenfassen:

Ein Instruktor kann mehrere Sportarten betreuen. Eine Disziplin kann von mehreren Instruktoren geleitet werden. Die Kurse können an verschiedenen Tagen und zu verschiedenen Zeiten stattfinden. Sie können von verschiedenen und gleichzeitig mehreren Instruktoren geleitet werden.

3 Datenmodellierung

Über die Zeit wachsen Datenbanken um ein Vielfaches ihrer Ursprungsgrösse. Verantwortlich dafür sind Erweiterungen, das Hinzufügen von zusätzlichen Informationssachverhalten und Aggregationen von verschiedenen Datenbanken zu unternehmensweiten Datenbanksystemen. Dies macht die Architektur der Datenbank bereits in ihrer Entstehungsphase zu einem äusserst wichtigen Punkt. Eine einfach und klar strukturierte Datenbank ist auch nach Erweiterungen und Aggregationen noch überschaubar und nutzbar.

Weiter ist die Wahl der Art der Datenbank von Bedeutung. Es gibt verschiedene Datenbanktypen, welche auf verschiedenen Modelltypen, wie zum Beispiel Netzwerkmodelle oder hierarchische Datenmodelle, aufbauen. In der Praxis dominiert jedoch die relationale Datenbank, die Informationen in Tabellenform (Relation) darstellt. Vorteile der relationalen Datenbank sind unter anderem die einfache Handhabung und die bereits bestehenden, deskriptiven Abfragesprachen, welche es auch dem Nicht-Spezialisten ermöglichen, die Datenbank sinnvoll zu nutzen. Bei den anderen Datenbanktypen sind Programmierkenntnisse Voraussetzung. Aus diesem Grund und dank der Flexibilität einer Tabellendarstellungsform erfreut sich die relationale Datenbank zunehmender Beliebtheit.¹

3.1 Erstellen eines Entitäten-Beziehungsmodells (ERM)



Als ersten Schritt einer Datenbankmodellierung wird ein Entwurf eines Datenbankschemas erstellt. Dieser beschreibt einen bestimmten Ausschnitt aus der realen Welt mit Hilfe eines Modells. Durch die Modellbildung wird dieser „Weltausschnitt“ vereinfacht, diskretisiert, idealisiert, andererseits aber auch für eine systematische Darstellung zugänglich gemacht. Dazu ist eine Methode erforderlich, welche übersichtlich und problembezogen die Elemente dieses Ausschnitts beschreibt. Das Modell soll jedoch unabhängig von technischen Randbedingungen sein. Eine Entwurfsmethode, welche diese Bedingungen erfüllt, ist das ER-Modell (Entity-Relationship-Modell), welches auf dem relationalen Ansatz von Codd (1970) basiert und von Chen (1976) entwickelt wurde.²

Nun wird also die Realität abstrahiert, indem den Elementen dieses Ausschnittes die Bezeichnung Entität oder Beziehung zugeordnet wird.

Def.: „Eine Entität (entity) ist ein individuelles Exemplar von Elementen der realen oder der Vorstellungswelt.“³

Oder nach Chen (1976): „Eine Entität (entity) ist ein „Ding“, welches eindeutig identifizierbar ist.“⁴

Def.: „Eine Beziehung (relationship) ist eine Zuordnung zwischen Entitäten.“⁵ Chen (1976)

Die einzelnen Entitäten und Beziehungen werden dann zu Entitätsmengen (Symbol: ) bzw. Beziehungsmengen (Symbol: ) zusammengefasst, wobei diese Mengen gleiche oder ähnliche Merkmale (Attribute), aber unterschiedliche Merkmalswerte aufweisen.

Schliesslich werden diese Entitäten und Beziehungen in einem sogenannten Entitätenblockdiagramm grafisch dargestellt.

¹ vgl. Meier 01, S.125

² vgl. Zehnder 98, S.61 f

³ Zehnder 98, S.63

⁴ Zehnder 98, S.68

⁵ Zehnder 98, S.68

Zur Beschreibung einer Beziehung zwischen zwei Entitätsmengen wird eine gerichtete Assoziation festgelegt.

Def.: „Eine Assoziation (EM1, EM2) legt fest, wie viele Entitäten aus EM2 einer Entität aus EM1 zugeordnet sein können.“⁶

Es existieren folgende Assoziationstypen⁷:

<u>Assoziationstypen:</u>	
Typ 1:	„genau ein“
Typ c:	„kein oder ein“
Typ m:	„mehrere“
Typ mc:	„kein, ein oder mehrere“

Abb. 2: Assoziationstypen

Der letzte Schritt zur Fertigstellung des Entitätenblockdiagrammes beinhaltet die Ausstattung der Entitätsmengen und Beziehungsmengen mit ihren dazugehörigen Attributen (Merkmale).

Def.: Ein Attribut ist die Beschreibung einer bestimmten Eigenschaft der Entitäten einer Entitätsmenge und definiert (implizit) den Wertebereich der zugeordneten Datenwerte.⁸

Mit Hilfe dieser Attribute werden Identifikationsschlüssel für die Entitätsmengen wie auch Beziehungsmengen definiert.

Def.: „Ein Identifikationsschlüssel ist ein Schlüssel, dessen Wert jede Entität einer Entitätsmenge eindeutig identifiziert.“⁹

Somit sind alle Bauteile unseres ERM, welches unseren ausgewählten Ausschnitt aus der realen Welt auf formale Art darstellt, spezifiziert.

3.2 Entitäten

Der eben beschriebene Vorgang wird jetzt auf das konkrete Problem angewendet.

Folgende Entitäten (mit zugehörigen *Attributen*; **Identifikationsschlüssel**) sind vorhanden:

- DISZIPLIN (**Sportart**)
- KURS (**K#**, *Zeit*, *Dauer*, *Ort*)
- INSTRUKTOR (**P#**, *Name*, *Vorname*)
- WOCHE (**Wochentag**)
- SEMESTER (**Datum**)

⁶ Zehnder, S.65

⁷ Meier 01, S.18

⁸ vgl. Zehnder 98, S.70

⁹ Zehnder 98, S.42 ff

3.3 Beziehungen

Zwischen den ENTITÄTEN bestehen folgende **Beziehungen** (mit den dazugehörigen Assoziationen):

- Eine DISZIPLIN **wird** in mehreren KURSEN **betrieben**, ein KURS beinhaltet jedoch nur eine DISZIPLIN. (m:1)
- Ein INSTRUKTOR kann mehrere KURSE **leiten** und ein KURS kann von mehreren INSTRUKTOREN geleitet werden. (m:m)
- Ein KURS **findet** nur an einem WOCHEntag **statt**, während an einem WOCHEntag mehrere KURSE stattfinden können. (1:m)
- Ein Datum **im Semester** besitzt nur einen WOCHEntag, jedoch besitzt jeder WOCHEntag mehrere SEMESTERdaten. (1:m)

3.4 ER-Modell

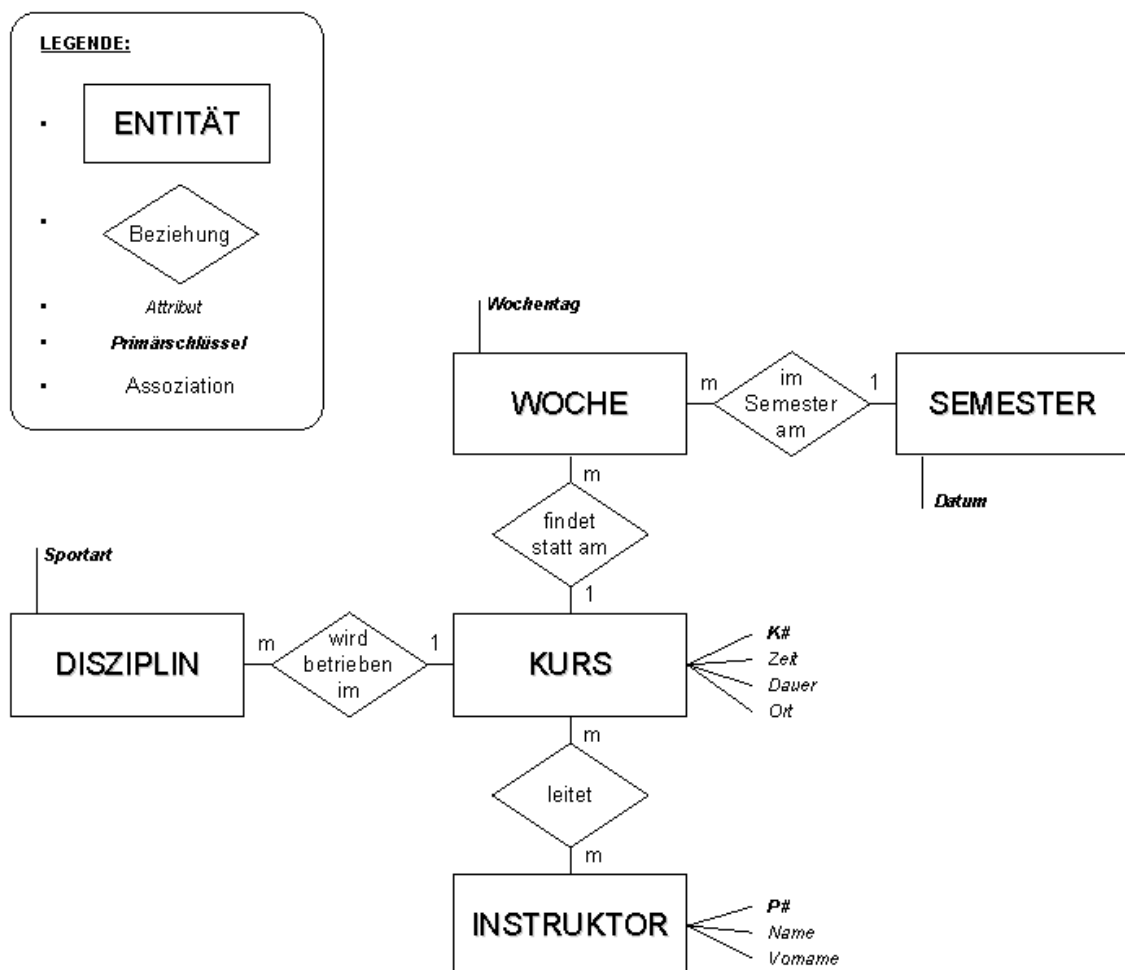


Abb. 3: Entitäten-Beziehungsmodell

3.5 Überführen des Entitäten-Beziehungsmodells in ein relationales Datenbankschema

Eine relationale Datenbank stellt Daten in Tabellenform dar (Relation=Tabelle). Aus diesem Grund wird das bis jetzt noch von technischen Randbedingungen unabhängige ERM unter Einsatz von Abbildungsregeln in ein relationales Datenbankschema überführt. Ein relationales Datenbankschema enthält die Definition der Tabellen, der Merkmale und der Primärschlüssel. Die folgenden **Abbildungsregeln**¹⁰ schreiben im Wesentlichen vor, wie die einzelnen Entitäts- und Beziehungsmengen als Tabellen darzustellen sind. An Stelle der Überführung des ERM durch Abbildungsregeln könnte auch ein äquivalentes Verfahren, das Normalisieren der Relationen, verwendet werden, um zu einem konsistenten relationalen Datenbankschema zu gelangen.¹¹

Regel 1 (Entitätsmengen)

„Jede Entitätsmenge *muss* als eigenständige Tabelle mit einem eindeutigen Primärschlüssel definiert werden. Als Primärschlüssel der Tabelle dient entweder der entsprechende Schlüssel der Entitätsmenge oder ein Schlüsselkandidat. Die übrigen Merkmale der Entitätsmengen gehen in die korrespondierenden Attribute der Tabelle über.“

Auf Grund dieser Regel, werden die Entitäten (vgl. 3.2) in eigenständige Tabellen überführt. Es entstehen folgende Tabellen, wobei die Entität zum Tabellennamen wird und die Attribute in der obersten Zeile zu stehen kommen. Die **Primärschlüssel** erscheinen fett gedruckt.

Sportart
Basketball
Fussball
Tennis
...

P#	Name	Vorname
1	Meier	Ludwig
2	Dupont	Francois
3	Bertagia	Giovanni
...

K#	Zeit	Dauer	Ort
1	20.00-22.00	2h	St. Croix
2	18.30-20.00	1.5h	Pérolles
3	12.00-13.45	1.75h	Miséricorde
...

Wochentag
Montag
Dienstag
Mittwoch
...

Datum
18.03.02
19.03.02
20.03.02
...

Abb. 4: Umsetzung Abbildungsregel 1

¹⁰ Die folgenden vier Abbildungsregeln wurden nach (Meier 01, S. 25 ff) angewendet.

¹¹ vgl. Zehnder 98, S. 77

Wenden wir uns jetzt den Beziehungsmengen zu.

Regel 2 (Beziehungsmengen)

„Jede Beziehungsmenge *kann* als eigenständige Tabelle definiert werden, wobei die Identifikationsschlüssel der zugehörigen Entitätsmengen als Fremdschlüssel in dieser Tabelle auftreten müssen. Der Primärschlüssel der Beziehungsmengentabelle kann der aus den Fremdschlüsseln zusammengesetzte Identifikationsschlüssel sein oder ein anderer Schlüsselkandidat, z.B. in Form eines künstlichen Schlüssels. Weitere Merkmale der Beziehungsmenge erscheinen als zusätzliche Attribute in der Tabelle.“

Bei dieser Regel handelt es sich per Definition um eine *Kann*-Regel. Es besteht also kein unbedingter Zwang sie anzuwenden. Eher dient sie als Grundlage für die folgenden zwei Regeln, welche die Beziehungsmengen in Relationen überführen. Diese Regeln stützen sich auf die Spezifikation der Assoziationen der Beziehungen zwischen den Entitäten.

Regel 3 (komplex-komplexe Beziehungen)

„Jede komplex-komplexe Beziehung (m:m; mc:m; m:mc; mc:mc) *muss* als eigenständige Tabelle definiert werden. Dabei treten mindestens die Identifikationsschlüssel der zugehörigen Entitätsmengen als Fremdschlüssel auf. Der Primärschlüssel der Beziehungsmengentabelle ist entweder der aus den Fremdschlüsseln zusammengesetzte Identifikationsschlüssel oder ein anderer Schlüsselkandidat. Die weiteren Merkmale der Beziehungsmenge gehen in Attribute der Tabelle über.“ Fremdschlüssel erscheinen unterstrichen.

leitet

<u>K#</u>	<u>P#</u>
1	2
2	1
3	3
...	...

Wie aus Abb.3 ersichtlich, besteht in unserem Fall lediglich eine Beziehung, die dieser Kategorie zugeordnet werden kann. Somit muss die Beziehung „leitet“ als eigenständige Tabelle, mit dem aus den beiden Fremdschlüsseln (unterstrichen) K# und P# zusammengesetzten Primärschlüssel, dargestellt werden.

Abb. 5: Umsetzung Abbildungsregel 3

Regel 4 behandelt die einfach-komplexen Beziehungen. Dieser Gruppe gehören die restlichen Beziehungsmengen unseres ERM an.

Regel 4 (einfach-komplexe Beziehungen)

„Eine einfach-komplexe Beziehungsmenge *kann* ohne eine eigenständige Beziehungsmengentabelle durch die beiden Tabellen der zugeordneten Entitätsmengen ausgedrückt werden. Dazu wird in der Tabelle mit der einfachen Assoziation (d.h. mit Assoziationstyp 1 oder c) ein Fremdschlüssel auf die referenzierte Tabelle mit eventuell weiteren Merkmalen der Beziehungsmenge geführt.“

Somit werden die Beziehungsmengen „wird betrieben in“, „findet statt am“ und „im Semester am“ mit Hilfe der Tabellen der zugehörigen Entitätsmengen ausgedrückt, wobei die Primärschlüssel der Entitäten mit einfachen Assoziationen in den Tabellen der Entitäten mit komplexen Assoziationen als Fremdschlüssel auftauchen. „Sportart“ und „Wochentag“ werden der Tabelle KURS angehängt, während die Tabelle SEMESTER ebenfalls den Fremdschlüssel „Wochentag“ erhält.

K#	Zeit	Dauer	Ort	Sportart	Wochentag
1	20.00-22.00	2h	St. Croix	Basketball	Montag
2	18.30-20.00	1.5h	Pérolles	Fussball	Mittwoch
3	12.00-13.45	1.75h	Miséricorde	Spinning	Dienstag
...

Datum	Wochentag
18.03.02	Montag
19.03.02	Dienstag
20.03.02	Mittwoch
...	...

Abb. 6: Umsetzung Abbildungsregel 4

Hiermit ist die Überführung des ERM in ein relationales Datenbankschema vollzogen und wir erhalten als vorläufiges Resultat die Relationen:

- DISZIPLIN
- INSTRUKTOR
- WOCHE
- "leitet"
- KURS
- SEMESTER

3.6 Kontrolle anhand von Normalformen und Integritätsbedingungen

Normalformen:

Datenstrukturen können mit verschiedenen Verfahren modelliert werden. Das relationale Modell ist ein solches. Wie das vorhin angewendete Verfahren des ERM und das Anschliessende „Überführen mit Hilfe von Abbildungsregeln“ besteht eine gleichwertige Entwurfsmethode, die Normalisation. Die Normalisation ist eine Technik, welche entwickelt wurde um sicher zu gehen, dass eine Datenstruktur effizient ist. Sie stammt von Codd und wurde speziell für den Gebrauch an relationalen Systemen entwickelt.¹²

Auch heute wird sie noch von etlichen Datenarchitekten verwendet. Die Normalisation verfolgt das gleiche Ziel wie die „Überführung mit Hilfe von Abbildungsregeln“. Es soll ein relationales Datenbankschema resultieren, welches frei von redundanten Informationen ist.¹³

¹² vgl. Mayne/ Wood 83, S. 61

¹³ Eine ausführliche Anleitung zur Normalisation findet sich in (Kudlich 88, S.75).

Def.: „Ein Merkmal einer Tabelle ist *redundant*, wenn einzelne Werte dieses Merkmals innerhalb der Tabelle *ohne Informationsverlust weggelassen* werden können.“¹⁴

Bei Tabellen mit redundanter Information können sogenannte Mutationsanomalien auftreten. Als solche sind zu nennen: die Einfüge-, Lösch- und Änderungsanomalie. Während bei einer Einfügeanomalie keine neuen Tupel eingefügt werden können, gehen bei einer Löschanomalie auf Grund eines Löschvorgangs ungewollt Informationen verloren. Bei einer Änderungsanomalie verursachen Änderungen, die eigentlich nur einen einzigen Sachverhalt betreffen, einen Änderungsnotstand an zusätzlichen Stellen der Tabelle.

Da mit beiden Methoden der gleiche Endzustand angestrebt wird, kann die Eine benutzt werden um die richtige Anwendung der Anderen zu kontrollieren. Aus diesem Grunde, wird nun anhand der Normalformen (NF) kontrolliert, ob die Abbildungsregeln sauber angewandt und Redundanzen und Anomalien vermieden wurden.¹⁵

Im vorliegenden Fall genügt es, wenn man sich auf die ersten drei Normalformen (NF) konzentriert.

Zuerst werden nun die ersten drei **Normalformen**¹⁶ definiert um dann die einzelnen Relationen, welche wir im relationalen Datenbankschema erhalten haben, zu kontrollieren. Ist jede Tabelle in 3. NF (siehe folgende Definition) und bestehen keine zusammengesetzten, sich überlappende Schlüssel, so wurden allfällige Redundanzen vollständig eliminiert.

1. Normalform (1.NF)

„Eine Tabelle ist in *erster Normalform*, falls die Wertebereiche der Merkmale *atomar* sind. Die erste Normalform verlangt, dass jedes Merkmal Werte aus einem unstrukturierten Wertebereich bezieht. Somit dürfen keine Mengen, Aufzählungstypen oder Wiederholungsgruppen in den einzelnen Merkmalen vorkommen.“

Da in keiner der Tabellen Aufzählungen oder Wiederholungen im Wertebereich der Merkmale auftauchen, ist jede Tabelle in erster Normalform!

2. Normalform (2.NF)

„Eine Tabelle ist in *zweiter Normalform*, wenn sie in erster Normalform ist und wenn jedes Nichtschlüsselmerkmal von jedem Schlüssel voll *funktional abhängig* bleibt.“

Def.: Funktionale Abhängigkeit bedeutet, dass jeder Wert des Primärschlüssels eindeutig einen Wert der Merkmale bestimmt.

¹⁴ Meier 01, S. 34

¹⁵ vgl. Meier 01, S. 35f

¹⁶ Die folgenden Definitionen der Normalformen 1-3 finden sich in (Meier 01, S. 36ff).

Volle funktionale Abhängigkeit besagt, dass nur ein zusammengesetzter Schlüssel ein Nichtschlüsselmerkmal eindeutig bestimmen darf. Kann ein Teil des Schlüssels ein Nichtschlüsselmerkmal eindeutig bestimmen, besteht keine volle funktionale Abhängigkeit.¹⁷

In diesem Fall bestimmt jeder Wert der Primärschlüssel die Merkmalswerte eindeutig. Da bei der einzigen Tabelle („leitet“) mit einem zusammengesetzten Schlüssel keine weiteren Nichtschlüsselmerkmale vorkommen, sind alle Tabellen laut Definition in zweiter Normalform.

3. Normalform (3.NF)

„Eine Tabelle ist in *dritter Normalform*, wenn sie in zweiter Normalform ist und kein Nichtschlüsselmerkmal von irgendeinem Schlüssel *transitiv* abhängig ist.“

Def.: Transitiv abhängig bedeutet, über Umwege abhängig zu sein. Sie lässt sich am einfachsten anhand einer Grafik erklären:¹⁸

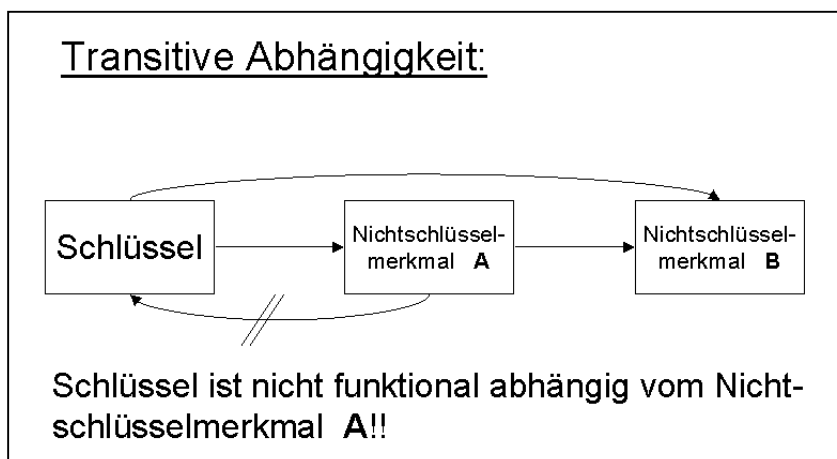


Abb. 7: Transitive Abhängigkeit (vgl. Meier 01, S. 40)

Einzig die Tabelle „Kurs“ könnte eine transitive Abhängigkeit beinhalten während sich alle anderen Tabellen in 3. Normalform befinden. Tatsächlich lässt sich in der Relation „Kurs“ folgenden Sachverhalt feststellen:

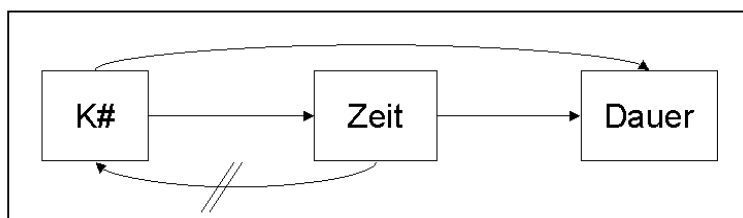


Abb. 8: Transitive Abhängigkeit der Dauer von K# (vgl. Meier 01, S. 40)

¹⁷ vgl. Meier 01, S. 37f

¹⁸ vgl. Meier 01, S. 39f

Durch Zerlegung wird die Relation von der transitiven Abhängigkeit befreit, indem das redundante Merkmal „Dauer“ zusammen mit der „Zeit“ als eigenständige Tabelle DURATION geführt wird. Die „Zeit“ bleibt als Fremdschlüssel in der Resttabelle KURS bestehen.

K#	Zeit	Ort	Sportart	Wochentag
1	20.00-22.00	St. Croix	Basketball	Montag
2	18.30-20.00	Pérolles	Fussball	Mittwoch
3	12.00-13.45	Miséricorde	Spinning	Dienstag
...

Zeit	Dauer
20.00-22.00	2h
18.30-20.00	1.5h
12.00-13.45	1.75h
...	...

Abb. 9: Tabellen in 3. Normalform

Da dieses Problem jedoch auch durch eine Rechenoperation gelöst werden kann, belassen wir der Einfachheit halber das Merkmal „Dauer“ in der Tabelle KURS. Wie im nächsten Kapitel ersichtlich, unterstützt FileMaker solche Operationen bestens.

Integritätsbedingungen:

Def.: Integrität oder Konsistenz bedeutet Widerspruchsfreiheit von Datenbeständen. Eine Datenbank ist integer oder konsistent, falls die gespeicherten Daten fehlerfrei erfasst sind und den gewünschten Informationsgehalt korrekt wiedergeben.¹⁹

Eine relationale Datenbank muss gewisse strukturelle Integritätsbedingungen erfüllen. Es wird nun getestet, ob diese Bedingungen bei der vorliegenden Datenbank erfüllt sind.

Die *Eindeutigkeitsbedingung* besagt, dass jede Tabelle einen Identifikationsschlüssel besitzt, der jedes Tupel in der Tabelle auf eindeutige Art bestimmt. Wie aus dem relationalen Datenbankschema ersichtlich, erfüllt jede Tabelle diese Bedingung.

Die *Wertebereichbedingung* fordert, dass die Merkmale einer Tabelle nur Datenwerte aus einem vordefinierten Wertebereich annehmen dürfen. Da die exakte Überprüfung dieser Bedingung komplex und mit viel Aufwand verbunden ist, wird auf die genaue Ausführung hier verzichtet. Die letzte Bedingung (*referentielle Integritätsbedingung*), die eine relationale Datenbank erfüllen muss, sagt aus, dass jeder Wert eines Fremdschlüssels auch effektiv als Schlüsselwert in der referenzierten Tabelle existieren muss. Auch diese letzte Bedingung erfüllt das vorliegende Schema und so besitzen wir nun ein konsistentes relationales Datenbankschema (siehe Abb.10), welches darauf wartet mit geeigneter Software implementiert zu werden.²⁰

¹⁹ vgl. Meier 01, S. 44

²⁰ vgl. Meier 01, S. 44ff

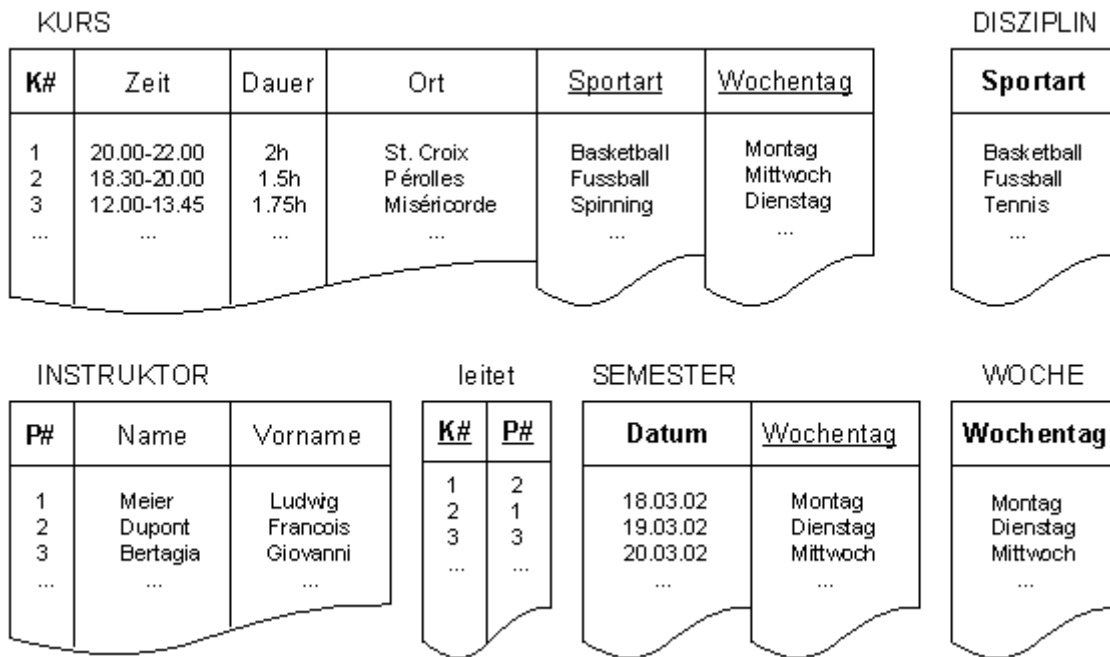


Abb. 10: Finales relationales Datenbankschema

4 Implementierung

4.1 FileMaker

Viele erfahrene Benutzer von FileMaker rühmen die Vorteile, die diese Datenbanksoftware mit sich bringt. Es ist die meist geschätzte Datenbanksoftware für Macintosh und die am zweit meisten verwendete für Windows. Diese Begeisterung für FileMaker mag etwas rätselhaft erscheinen, wenn man noch nie mit diesem Programm gearbeitet hat. Die Vorteile lassen sich vereinfacht in zwei Worten zusammenfassen: sauber und einfach. Die Benutzeroberfläche ist laufend vereinfacht worden und erlaubt es dem Benutzer, bereits nach einer kurzen Einführung, mit diesem Programm zu arbeiten (FileMaker besteht seit ca. 1985). Die Kompatibilität der Benutzung auf Macintosh und Windows zählt zu den weiteren grossen Stärken von FileMaker. Die Einfachheit hemmt zwar etwas die Mächtigkeit der Software, aber was nützt eine Datenbank, welche mit einer leistungsstarken Software aufgebaut wurde, wenn sie nur von Spezialisten bedient werden kann?²¹

In den folgenden Kapiteln werden des Öfteren englische Begriffe vorkommen. Es wird teilweise auf Anführungs- und Schlusszeichen verzichtet.

²¹ vgl. Hester 00, S. 7f

4.2 Einführung

Es wird versucht, einen kurzen Einblick in die Welt des FileMaker zu geben, welcher es dem Leser vereinfachen soll, die nachfolgenden Erklärungen zu verstehen. Im Folgenden werden die Begriffe File, Relation und Tabelle synonym verwendet.

FileMaker arbeitet mit verschiedenen Modi (modes), welche im "View menu" gewechselt werden können.

- Browse mode, welcher dazu dient Daten einzugeben und Einträge anzuschauen
- Find mode, der es erlaubt Einträge schnell zu lokalisieren und aufzulisten
- Layout mode, in welchem das Erscheinungsbild der Daten verändert werden kann
- Preview mode, welcher die Daten, wie sie beim Ausdruck erscheinen, darstellt

In einem geöffneten File kann zwischen diesen Modi, je nach Bedarf gewechselt werden.

In der Arbeitsleiste am linken Rand wird die Gesamtzahl der Records (Einträge) angezeigt, wobei im L-förmigen Kästchen die Nummer des aktuellen Records erscheint. Durch Klicken auf das Buchsymbol kann ein anderer Tupel (Eintrag) angezeigt werden.

Im Layout Modus wird definiert, welche der Felder, die ein File besitzt, gezeigt werden sollen und welche nicht. FileMaker erlaubt es also eigene Layouts zu kreieren. Diese können in drei verschiedenen Formen dargestellt werden.

- Form View, zeigt nur einen Eintrag (Record) des aktuellen Files aufs Mal
- List View, zeigt die aktuellen Einträge in Form einer Liste. Eintrag unter Eintrag
- Table View zeigt viele Einträge in Tabellenform

FileMaker speichert jegliche Änderungen sofort. Es ist daher ratsam, vor grossen Modifikationen Backup's zu machen.²²

4.3 Dokumentation der Implementierung

Im 5. Kapitel wird vermehrt auf die Benutzeroberfläche eingegangen. Die Implementierungserklärungen in diesem Kapitel werden kurz gehalten und nur die wichtigsten Schritte erläutert.

1. Zuerst wird für jede der Relationen, welche im relationalen Datenbankschema ermittelt wurden, ein einzelnes FileMaker File kreiert. In File menu>Define Fields, werden die Merkmale der einzelnen Relationen definiert. Hier werden bereits die Wertebereiche, welche die Merkmalswerte (Attributwerte) annehmen können berücksichtigt. So wird zum Beispiel zwischen "Text", "Time" oder "Number" unterschieden. Zusätzlich wird der Tabelle DISZIPLIN noch ein künstlicher Schlüssel (S#) angehängt, damit allfällige Namensänderungen von Sportarten einfacher umgesetzt werden können.

²² vgl. FMP 5.5 Getting Started.pdf 01, S. 41ff

Die Tabelle WOCHE wird zwecks Vereinfachung weggelassen und als Merkmal in der Relation KURS geführt.

Ein Knackpunkt stellt das Attribut Dauer aus der Tabelle KURS dar. Es handelt sich bei diesem Field um ein "Calculation Field", welches die Differenz von Beginn und Ende des Kurses in 45-Minuten Lektionen umwandelt und als eine ganze Zahl mit einem Bruch darstellt. Diese Darstellungsform wird gewählt, um die nachfolgende Lohnabrechnung zu vereinfachen.²³

2. Definition der Beziehungen zwischen den Relationen: Unter File menu>Define Relationships werden Beziehungen des aktuellen Files zu anderen Relationen spezifiziert

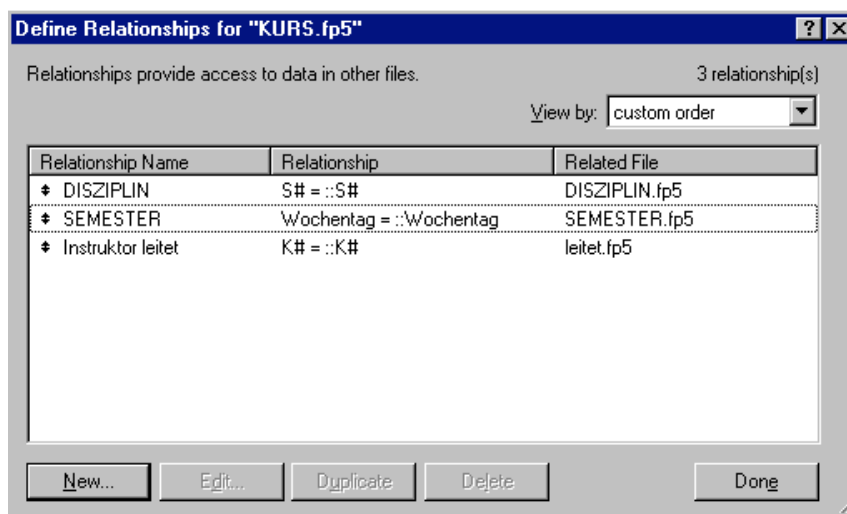


Abb. 11: Beziehungsdefinitions Assistent (FileMaker, File menu>Define Relationships)

(siehe Abb.11, Beispiel KURS). FileMaker unterscheidet zwischen "Master" und "Related Files". Im "Master File" können Daten des "Related File" dargestellt werden. Umgekehrt jedoch nicht! In diesem

Beispiel ist KURS das "Master File", welches Einträge der Files DISZIPLIN und SEMESTER zeigt.²⁴

Ein etwas komplizierterer Fall ist die letzte Beziehung "Instruktor leitet". Da das File KURS dazu benutzt wird, das Layout für den Arbeitsrapport zu erstellen, müssen in dieser Relation alle benötigten Daten dargestellt werden können. Dazu gehören auch die Vor- und Nachnamen der Instruktoren. FileMaker kann jedoch nur Merkmalswerte direkt relationierter Files anzeigen. Wie aus dem ERM ersichtlich, befinden sich diese Daten jedoch in der Tabelle INSTRUKTOR, welche durch die Beziehungstabelle "leitet" vom KURS getrennt ist.

²³ vgl. FMP 5.5 Getting Started.pdf 01, S. 51f

²⁴ vgl. FMP 5 User's Guide.pdf 99, Chapter 8 S. 9ff

Das Problem wird gelöst, indem in der Tabelle "leitet" zwei neue Fields: „Instruktor Vorname“ und „Instruktor Name“, erstellt werden (siehe Abb.12), die als "Calculation Fields" die Beziehung von INSTRUKTOR zu

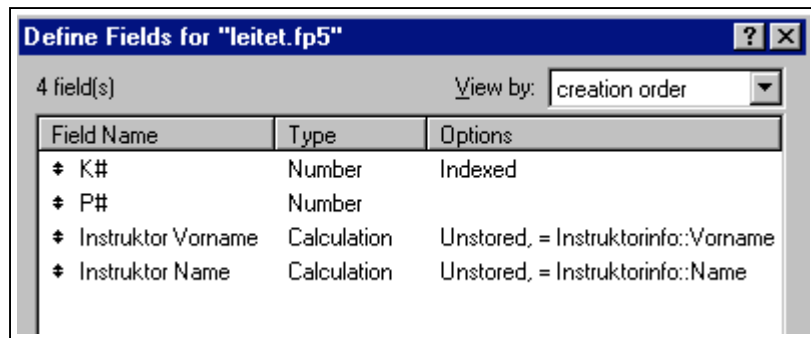


Abb. 12: „Field-Definitions“ Assistent (FileMaker, File menu>Define Fields)

"leitet" mit der Beziehung von "leitet" zu KURS (Instruktor leitet) verbindet. Mit Hilfe dieser Überbrückung können die Merkmalswerte der Attribute Vorname und Name der Tabelle INSTRUKTOR in der Relation KURS dargestellt werden²⁵

3. In der Tabelle SEMESTER sollen alle Daten, an denen Kurse stattfinden, mit den dazugehörigen Wochentagen erscheinen. Diese Daten sollen dann im Layout Arbeitsrapport, im File KURS, für jeden einzelnen Wochentag aufgelistet werden. Eine Beziehung zwischen KURS und SEMESTER über den Schlüssel Wochentag wurde bereits definiert. Dies weicht etwas von der ursprünglichen Lösung des ERM ab, ist jedoch in der Praxis einfacher realisierbar. Die Daten der einzelnen Wochentage werden durch ein Portal im File KURS abgebildet.²⁶

Um die gewünschten Daten zu erhalten, kann in FileMaker ein Skript konzipiert werden, welches mehrere Befehle hintereinander ausführt. Im File SEMESTER wird Scripts

```

Delete All Records [No dialog]
Set Field [„Rep“, „1“]
Loop
  New Record/Request
  Set Field [„Datum“, „GetRepetition(Anfang;Rep)“]
  Loop
    Exit Loop If [„Datum = GetRepetition(Ende;Rep)“]
    New Record/Request
    Insert from last Record [Select, „Datum“]
    Set Field [„Datum“, „Datum + 1“]
  End Loop
  Exit Loop If [„Last(Anfang) = GetRepetition(Anfang;Rep)“]
  Set Field [„Rep“, „Rep + 1“]
End Loop

```

Abb. 13: Skript „Semesterdaten“

menu>ScriptMaker aufgerufen und mit Hilfe des Skript Assistenten das erforderliche Skript "Semesterdaten" (siehe Abb.13) erstellt.

Bei Ausführung dieses Skripts werden zuerst alle alten Einträge gelöscht. Danach wird das Feld "Rep" auf Eins gesetzt, welches die Anzahl der Repetitionen, die der äussere Loop durchläuft, zählt. Der äussere Loop wird gestartet und ein neuer Eintrag wird eruiert.

²⁵ vgl. FMP 5 User's Guide.pdf 99, Chapter 8 S. 15

²⁶ vgl. FMP 5 User's Guide.pdf 99, Chapter 6 S. 19

“Set Field” setzt das Datum ein, welches in der Spalte "Anfang" an der “Rep-ten” Stelle steht. Das heisst, beim Ersten Eintrag der Spalte steht der Counter (Rep) auf 1, also wird der erste Eintrag eingesetzt. Somit wird für jeden Eintrag der Spalte "Anfang" eine Repetition durchgeführt. Der innere Loop füllt nun das Intervall zwischen dem Anfangsdatum und dem Enddatum mit den dazwischenliegenden Daten. Jeder neue Eintrag entsteht aus dem vorherigen Datum, zu welchem Eins dazu gezählt wird. Der Prozess soll gestopt werden, wenn das Datum dem Datum entspricht, welches in der "Rep-ten" Zeile der Spalte "Ende" eingesetzt wurde. Die Nummer im Feld "Rep" wird dann um Eins erhöht und der ganze Loop erneut ausgeführt, bis die Nummer der letzten Zeile der Spalte "Anfang" der Anzahl durchlaufener Repetitionen entspricht.²⁷

Somit werden mit diesem Skript in der Tabelle SEMESTER alle Daten aufgeführt, welche zwischen den von Hand eingesetzten Vorlesungsanfang und Vorlesungsenddaten liegen.

Um das Wechseln zwischen den verschiedenen Layouts und die Aktivierung des Skripts zu vereinfachen, wurden noch zusätzlich Buttons erstellt.²⁸

4. Die grundlegendsten Bestandteile der relationalen Datenbank sind somit erstellt und es können nun verschiedenen Layouts kreiert werden. Im nächsten Kapitel werden die Layouts eingehender erklärt.

5 Benutzerinformationen

In Kapitel 5 soll der Benutzer mit der erstellten Datenbank vertraut gemacht werden.

5.1 Erstellen des Arbeitsrapportes

Wie in der Einleitung erwähnt, erfordert vor allem das Modifizieren der Semesterdaten in den Arbeitsrapporten enormen Aufwand. In diesem Teil wird nun diese Modifikation mit Hilfe der erstellten Datenbank, Schritt für Schritt beschrieben. Es soll möglich sein anhand dieser kleinen Anleitung die gewünschten Arbeitsrapporte zu erstellen. Wir gehen davon aus, dass keine weiteren Veränderungen durchgeführt werden müssen. Für weitere Modifikationen konsultiere Kapitel 5.2.

²⁷ vgl. FMP 5 User's Guide.pdf 99, Chapter 10

²⁸ vgl. FMP 5.5 Getting Started.pdf 01, S. 74

1. Öffnen Sie die 5 Files des Ordners FileMaker. Sie sehen nun die 5 Relationen, KURS, INSTRUKTOR, SEMESTER, DIZIPLIN und „leitet“, deren Kreierung im vorhergehenden Kapitel beschrieben worden sind. Überdecken sich die Fenster, so kann unter dem "Window menu" zwischen den einzelnen Files gewechselt werden.
2. Klicken sie auf das Fenster Semester. Der Titelbalken erscheint nun blau und das File ist aktiv. Klicken sie in der rechten oberen Ecke auf das Fenster Symbol, um das Fenster auf dem ganzen Bildschirm anzuzeigen.
3. Klicken sie auf das "Pop-up menu" in der Arbeitsleiste am linken Rand, welches die verfügbaren Layouts zeigt. Gehen Sie zum Layout Vorlesungsdaten.
4. Klicken sie unter die rot unterstrichenen Titel der Spalten Anfang und Ende. Wie in der

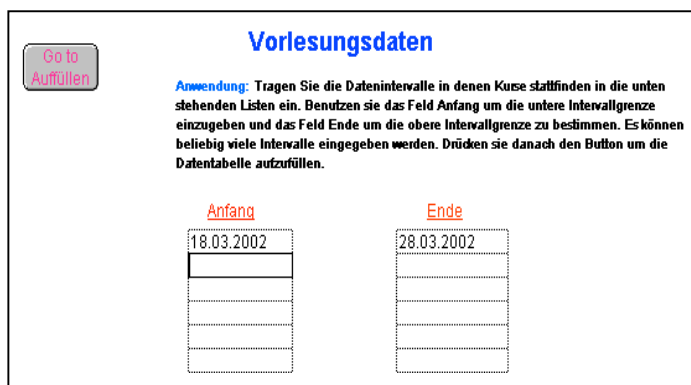


Abb. 14: Layout „Vorlesungsdaten“ im File SEMESTER

Abbildung zu sehen, erscheinen zwei netzlinienartige Tabellenspalten. Geben Sie nun die Anfangs- und Enddaten der Zeitperioden ein, in denen Vorlesungen bzw. Kurse stattfinden. Sind in einem Intervall nur einzelne Tage vorlesungsfrei, können diese

ignoriert werden. Später werden sie einfach einzeln gelöscht. Sonntage werden nicht berücksichtigt. Haben sie alle Intervalle eingegeben, so klicken sie auf den rosa Button „Go to Auffüllen“.

5. Im Layout "Auffüllen", wir befinden uns immer noch im File SEMESTER, erscheinen die vorhin eingesetzten Daten oben Rechts. Allerdings sind sie hier nicht mehr veränderbar. Um sie wieder zu verändern, klicken sie den rosa Button „Go to Semesterdaten“ und wiederholen sie Punkt 4. Stimmen die Daten, so klicken sie auf den roten Button „Auffüllen“. Die zwischen den eingegebenen



Abb. 15: Layout „Auffüllen“ im File SEMESTER

Intervallsgrenzen liegenden Daten werden nun in der untenstehenden Liste eingefüllt. Falls vorhin einzelne vorlesungsfreie Tage ignoriert wurden, können sie nun in der Tabelle SEMESTER einzeln gelöscht werden. Drücken sie dazu den violetten Button „Go to Semester“. Die Daten sind nun wie gewünscht bereitgestellt und die Relation Kurs kann über die in Kapitel 4 definierte Beziehung „SEMESTER“ auf diese Daten Rückgriff nehmen.

6. Durch Klicken auf das Fenster KURS, wechseln wir die Relation. In der Tabelle KURS gehen sie mit Hilfe des "Pop-up menu" zum Layout KURS. Sie sehen nun alle Kurse in Tabellenform. Am linken Rand zeigt „Records:....“, wieviele Einträge insgesamt in der

ARBEITSRAPPORT
Wintersemester 2011/2012

Spezial: Bachelorbil
Freiwillig: Fremdspr Doppel
Nachmittag: Montag
Zeit: 20.00 bis 22.00
Lektionen: 2 219 (1 Lektion = 50 Min)

Lektionen Total:

Datum	Anr.	Fehlzeiten	Bemerkungen
14.01.2012			
25.01.2012			
04.02.2012			
15.02.2012			
22.02.2012			
29.02.2012			
06.03.2012			
13.03.2012			
20.03.2012			
27.03.2012			
03.04.2012			
10.04.2012			
17.04.2012			
24.04.2012			
01.05.2012			
08.05.2012			
15.05.2012			
22.05.2012			
29.05.2012			
05.06.2012			
12.06.2012			
19.06.2012			
26.06.2012			
03.07.2012			
10.07.2012			
17.07.2012			
24.07.2012			
31.07.2012			
07.08.2012			
14.08.2012			
21.08.2012			
28.08.2012			
04.09.2012			
11.09.2012			
18.09.2012			
25.09.2012			
02.10.2012			
09.10.2012			
16.10.2012			
23.10.2012			
30.10.2012			
06.11.2012			
13.11.2012			
20.11.2012			
27.11.2012			
04.12.2012			
11.12.2012			
18.12.2012			
25.12.2012			
01.01.2013			
08.01.2013			
15.01.2013			
22.01.2013			
29.01.2013			
05.02.2013			
12.02.2013			
19.02.2013			
26.02.2013			
05.03.2013			
12.03.2013			
19.03.2013			
26.03.2013			
02.04.2013			
09.04.2013			
16.04.2013			
23.04.2013			
30.04.2013			
07.05.2013			
14.05.2013			
21.05.2013			
28.05.2013			
04.06.2013			
11.06.2013			
18.06.2013			
25.06.2013			
02.07.2013			
09.07.2013			
16.07.2013			
23.07.2013			
30.07.2013			
06.08.2013			
13.08.2013			
20.08.2013			
27.08.2013			
03.09.2013			
10.09.2013			
17.09.2013			
24.09.2013			
01.10.2013			
08.10.2013			
15.10.2013			
22.10.2013			
29.10.2013			
05.11.2013			
12.11.2013			
19.11.2013			
26.11.2013			
03.12.2013			
10.12.2013			
17.12.2013			
24.12.2013			
31.12.2013			
07.01.2014			
14.01.2014			
21.01.2014			
28.01.2014			
04.02.2014			
11.02.2014			
18.02.2014			
25.02.2014			
03.03.2014			
10.03.2014			
17.03.2014			
24.03.2014			
31.03.2014			
07.04.2014			
14.04.2014			
21.04.2014			
28.04.2014			
05.05.2014			
12.05.2014			
19.05.2014			
26.05.2014			
02.06.2014			
09.06.2014			
16.06.2014			
23.06.2014			
30.06.2014			
07.07.2014			
14.07.2014			
21.07.2014			
28.07.2014			
04.08.2014			
11.08.2014			
18.08.2014			
25.08.2014			
01.09.2014			
08.09.2014			
15.09.2014			
22.09.2014			
29.09.2014			
06.10.2014			
13.10.2014			
20.10.2014			
27.10.2014			
03.11.2014			
10.11.2014			
17.11.2014			
24.11.2014			
01.12.2014			
08.12.2014			
15.12.2014			
22.12.2014			
29.12.2014			
05.01.2015			
12.01.2015			
19.01.2015			
26.01.2015			
02.02.2015			
09.02.2015			
16.02.2015			
23.02.2015			
01.03.2015			
08.03.2015			
15.03.2015			
22.03.2015			
29.03.2015			
05.04.2015			
12.04.2015			
19.04.2015			
26.04.2015			
03.05.2015			
10.05.2015			
17.05.2015			
24.05.2015			
31.05.2015			
07.06.2015			
14.06.2015			
21.06.2015			
28.06.2015			
05.07.2015			
12.07.2015			
19.07.2015			
26.07.2015			
02.08.2015			
09.08.2015			
16.08.2015			
23.08.2015			
30.08.2015			
06.09.2015			
13.09.2015			
20.09.2015			
27.09.2015			
04.10.2015			
11.10.2015			
18.10.2015			
25.10.2015			
01.11.2015			
08.11.2015			
15.11.2015			
22.11.2015			
29.11.2015			
06.12.2015			
13.12.2015			
20.12.2015			
27.12.2015			
03.01.2016			
10.01.2016			
17.01.2016			
24.01.2016			
31.01.2016			
07.02.2016			
14.02.2016			
21.02.2016			
28.02.2016			
06.03.2016			
13.03.2016			
20.03.2016			
27.03.2016			
03.04.2016			
10.04.2016			
17.04.2016			
24.04.2016			
01.05.2016			
08.05.2016			
15.05.2016			
22.05.2016			
29.05.2016			
05.06.2016			
12.06.2016			
19.06.2016			
26.06.2016			
03.07.2016			
10.07.2016			
17.07.2016			
24.07.2016			
31.07.2016			
07.08.2016			
14.08.2016			
21.08.2016			
28.08.2016			
04.09.2016			
11.09.2016			
18.09.2016			
25.09.2016			
02.10.2016			
09.10.2016			
16.10.2016			
23.10.2016			
30.10.2016			
06.11.2016			
13.11.2016			
20.11.2016			
27.11.2016			
04.12.2016			
11.12.2016			
18.12.2016			
25.12.2016			
01.01.2017			
08.01.2017			
15.01.2017			
22.01.2017			
29.01.2017			
05.02.2017			
12.02.2017			
19.02.2017			
26.02.2017			
05.03.2017			
12.03.2017			
19.03.2017			
26.03.2017			
02.04.2017			
09.04.2017			
16.04.2017			
23.04.2017			
30.04.2017			
07.05.2017			
14.05.2017			
21.05.2017			
28.05.2017			
04.06.2017			
11.06.2017			
18.06.2017			
25.06.2017			
02.07.2017			
09.07.2017			
16.07.2017			
23.07.2017			
30.07.2017			
06.08.2017			
13.08.2017			
20.08.2017			
27.08.2017			
03.09.2017			
10.09.2017			
17.09.2017			
24.09.2017			
01.10.2017			
08.10.2017			
15.10.2017			
22.10.2017			
29.10.2017			
05.11.2017			
12.11.2017			
19.11.2017			
26.11.2017			
03.12.2017			
10.12.2017			
17.12.2017			
24.12.2017			
31.12.2017			
07.01.2018			
14.01.2018			
21.01.2018			
28.01.2018			
04.02.2018			
11.02.2018			
18.02.2018			
25.02.2018			
03.03.2018			
10.03.2018			
17.03.2018			
24.03.2018			
31.03.2018			
07.04.2018			
14.04.2018			
21.04.2018			
28.04.2018			
05.05.2018			
12.05.2018			
19.05.2018			
26.05.2018			
02.06.2018			
09.06.2018			
16.06.2018			
23.06.2018			
30.06.2018			
07.07.2018			
14.07.2018			
21.07.2018			
28.07.2018			
04.08.2018			
11.08.2018			
18.08.2018			
25.08.2018			
01.09.2018			
08.09.2018			
15.09.2018			
22.09.2018			
29.09.2018			
06.10.2018			
13.10.2018			
20.10.2018			
27.10.2018			
03.11.2018			
10.11.2018			
17.11.2018			
24.11.2018			
01.12.2018			
08.12.2018			
15.12.2018			
22.12.2018			
29.12.2018			
05.01.2019			
12.01.2019			
19.01.2019			
26.01.2019			
02.02.2019			
09.			

Änderungen, welche die Records (Einträge) betreffen, werden im Browse Modus durchgeführt (View menu>Browse Mode). Da die Tabellensicht (View menu>View as Table) grosse Übersichtlichkeit bietet, werden allfällige Modifikationen vorzugshalber in dieser Sicht vollzogen. Änderungen und Einträge in der Relation SEMESTER wurden im vorhergehenden Kapitel behandelt.

- DISZIPLIN

In der DISZIPLIN-Tabelle lässt sich der Schlüssel (S#), wie auch die Sportart verändern. Klicken sie dazu in das zu ändernde Feld und nehmen sie die Veränderung vor.

Um neue Einträge einzufügen, wählen sie unter Records menu New Record. Duplikationen von Einträgen und deren Löschung finden sich ebenfalls in diesem Menü. Bei einem neuen Eintrag erhöht sich der Schlüssel um eins.

- INSTRUKTOR

Analog zu DISZIPLIN.

- „leitet“

Hier lässt sich nur die Verknüpfung der Instruktoren mit den zu leitenden Kursen ändern. Leitet also ein Instruktor einen neuen Kurs, ist die Modifikation hier vorzunehmen. Alle anderen Werte ändern sich nach Modifikation der Kurs- und Personalnummer automatisch.

- KURS

Werden neue Kurse deklariert, werden sie hier mit den Informationssachverhalten K#, Sportart (bestimmt S# automatisch), Wochentag, Beginn, Ende (Dauer wird automatisch ausgerechnet) und Ort eingegeben. Gleiches gilt für Änderungen.

⇒ Nach jeder Änderung sollen die Einträge neu geordnet werden (Records>Sort..)

Fazit: Auch wenn in den Tabellen mehrere Informationssachverhalte gezeigt werden, können nur diejenigen Werte verändert werden, welche im relationalen Datenbankschema Bestandteil der Relation sind.

Beispiel: Es wird ein neuer Instruktor (Lars Stäuber P11) eingestellt. Er übernimmt den bereits bestehenden Kurs 2 (Fussball am Mittwoch im Pérolles von 18.30-20.00), der auf Grund mangelnder Nachfrage in ein Badminton (S6) Kurs umgewandelt wird, von Francois Dupont (P2). Dieser leitet dafür den neu ins Sortiment aufgenommene Kurs 11 (Curling am Freitag in der Patinoire von 10.00-12.00).

DISZIPLIN: Curling wird als neue Sportart #11 in der Tabelle aufgenommen.

INSTRUKTOR: Lars Stäuber erhält als neuen Record in Tabelle Instruktor die P# 11.

„leitet“: P2 leitet neu K11. P2 leitet K2 wird gelöscht. Neuer Eintrag P11 leitet K2.

KURS: Neuer Eintrag des Kurses 11 mit S11 (=Curling), Freitag, 10.00-12.00 in der Patinoire. Änderung K2 erhält neu S6 statt S2 (=Fussball).

5.3 Suchen und Finden (Datenbankabfragen)

Eine Datenbank ist eine Sammlung von Einträgen. Manchmal will man jedoch nicht mit allen Einträgen arbeiten, sondern nur mit einer Auswahl von Einträgen. Diese Auswahl wird durch Kriterien eingeschränkt. Zum Beispiel könnte man daran interessiert sein, welche Instrukturen Basketball leiten. FileMaker besitzt den bereits erwähnten "Find mode", in welchem solche Abfragen einfach durchgeführt werden können. Hier ein paar Beispiele möglicher Abfragen und wie man am schnellsten zum gewünschten Ergebnis kommt.

1 Welche Instrukturen leiten Basketball?

In Tabelle KURS View menu>Find Mode. Eingabe in Feld Sportart=Basketball, klicke in der sich am linken Rand befindenden Arbeitsleiste „Find“. Es erscheinen die gefundenen Einträge. Die gleiche Antwort würde auch die Frage: Wann finden Basketballkurse statt? eruieren. Um wieder alle Einträge zu sehen, Records>Show All Records.

2 Welche Kurse finden am Montag statt?

KURS, Eingabe in Wochentag=Montag, Find

3 Welche Kurse finden im Péroilles statt?

KURS, Eingabe in Ort=Péroilles, Find

4 Zeige alle Montage im Semester.

SEMESTER, Eingabe in Wochentag=Montag, Find

5 Welche Kurse oder Disziplin gibt Francois Dupont?

„leitet“, Eingabe in Name=Dupont und in Vorname=Francois, Find

6 Von wem wird K2 geleitet?

„leitet“, Eingabe in K#=2, Find

Hierbei handelt es sich um einfache Abfragen. FileMaker kann auch nach mehreren Kriterien suchen. Zum Beispiel nach allen Instrukturen, welche Fussball leiten und deren Vorname Andreas ist. Dazu geben sie bei der Tabelle KURS für Sportart=Fussball ein und beim "Vornamen" Andreas, dann wieder "Find" anklicken. Diese Suche wird auch AND search genannt.

Manchmal braucht man jedoch Einträge, welche mehrere Kriterien in einem einzelnen Feld erfüllen können. Wie zum Beispiel die Suche nach allen Instruktoren die Fussball oder Basketball leiten. Erste Eingabe ins Feld Sportart=Fussball, dann Requests menu>Add new Request, zweiter Eintrag Sportart=Basketball und schliesslich „Find“: OR search.³⁰

⇒ Um wieder alle Einträge sichtbar zu machen (Records>Show All Records).

Wie in diesem Kapitel gezeigt wurde, sind die Abfragen in FileMaker sehr einfach zu handhaben. Dies ist ein weiterer grosser Vorteil, den FileMaker gegenüber viel mächtigerer Datenbanksoftware hat.

6 Schlussbemerkung

Nach anfänglicher Skepsis, ob mein , nur in Vorlesungen des ersten Jahres Wirtschaft erworbenes Wissen über Datenbanken ausreichen wird, um die gestellte Aufgabe zu bewältigen, wurde das Interesse bezüglich Datenbanken und ihrer Möglichkeiten zunehmend grösser. Besonders im Arbeitsschritt der Datenbankmodellierung wurde mir bewusst, wie schwierig es ist, gelernte Theorie in die Praxis umzusetzen. So wurde dann gemeinsam mit dem Assistenten nach möglichen Lösungen gesucht und schliesslich der richtige Weg eingeschlagen. Wie wichtig eine saubere Konzeption ist wurde in der Implementierung offensichtlich, welche bis auf ein paar kleine Haken problemlos vor sich ging. Besonders die Implementierungssoftware FileMaker Pro 5.5 hat diese Aufgabe enorm vereinfacht, wenn nicht gar für mich erst lösbar gemacht.

Die Arbeit an diesem Projekt war für mich sehr lehrreich und spannend. Ich hoffe, damit dem Institut für Sporterziehung und Sport eine Lösung zu präsentieren, welche es auch in zukunft anwenden kann. Die Praxisbezogenheit war und ist eines der Hauptkriterien für die Wahl dieses Themas.

³⁰ vgl. FMP 5.5 Getting Started.pdf 01, S. 47

7 Abbildungsverzeichnis

Abb. 1:Vorlage Arbeitsrapport.....	1
Abb. 2:Assoziationstypen.....	4
Abb. 3:Entitäten-Beziehungsmodell.....	5
Abb. 4:Umsetzung Abbildungsregel 1	6
Abb. 5:Umsetzung Abbildungsregel 3	7
Abb. 6:Umsetzung Abbildungsregel 4	8
Abb. 7:Transitive Abhängigkeit (vgl. Meier 01, S. 40).....	10
Abb. 8:Transitive Abhängigkeit der Dauer von K# (vgl. Meier 01, S. 40)	10
Abb. 9:Tabellen in 3. Normalform.....	11
Abb. 10:Finales relationales Datenbankschema	12
Abb. 11:Beziehungsdefinitions Assistent (FileMaker, File menu>Define Relationships).....	14
Abb. 12:„Field-Definitions“ Assistent (FileMaker, File menu>Define Fields).....	15
Abb. 13:Skript „Semesterdaten“	15
Abb. 14:Layout „Vorlesungsdaten“ im File SEMESTER.....	17
Abb. 15:Layout „Auffüllen“ im File SEMESTER	17
Abb. 16:Layout „Arbeitsrapport 1“ im File KURS.....	18

8 Literaturverzeichnis

FileMaker Pro 5.5, Electronic Documentation \ FMP 5 User's Guide.pdf. Santa Clara (CA), 1999

FileMaker Pro 5.5, Electronic Documentation \ FMP 5.5 Getting Started.pdf. Santa Clara (CA), 2001

Hester, Nolan; FileMaker Pro 5. CampusPress Paris Berkeley (CA), 2000

Kudlich, Hermann; Datenbank-Design. Springer-Verlag Wien New York, 1988

Mayne, Alan/Wood, Michael; Introducing Relational Database. NCC Publications Manchester, 1983

Meier, Andreas; Relationale Datenbanken. Springer-Verlag Berlin Heidelberg, 4. Auflage, 2001

Zehnder, Carl August; Informationssysteme und Datenbanken. B. G. Teubner Stuttgart, 6.Auflage, 1998

Inhaltsverzeichnis

1	Einleitung	1
1.1	Problemstellung	1
1.2	Zielsetzung	2
1.3	Vorgehensweise	2
2	Datenanalyse	2
3	Datenmodellierung	2
3.1	Erstellen eines Entitäten-Beziehungsmodells (ERM)	3
3.2	Entitäten	4
3.3	Beziehungen	5
3.4	ER-Modell	5
3.5	Überführen des Entitäten-Beziehungsmodells in ein relationales Datenbankschema	6
3.6	Kontrolle anhand von Normalformen und Integritätsbedingungen	8
4	Implementierung	12
4.1	FileMaker	12
4.2	Einführung	13
4.3	Dokumentation der Implementierung	13
5	Benutzerinformationen	16
5.1	Erstellen des Arbeitsrapportes	16
5.2	Neue Einträge und Modifikationen	18
5.3	Suchen und Finden (Datenbankabfragen)	20
6	Schlussbemerkung	21
7	Abbildungsverzeichnis	22
8	Literaturverzeichnis	22