

Einführung in MS Access

Teil 2: Datenmodellierung

Karel Štípek

Datenmodellierung

Der Entwurf der Tabelle tblPerson ist FALSCH

Warum? So eine schöne Tabelle! (Einige Felder sind hier wegen der Übersichtlichkeit ausgeblendet)

wir in einer Tabelle, die Attribute einer Firma in einer anderen Tabelle. Wie werde ich aber dann festlegen, welche Person bei welcher Firma arbeitet?

Die Lösung ist einfach und heißt

Das Ergebnis sehen Sie auf dem unteren Bild:

Der technische Namen des fremden Schlüssels setzt sich aus der Abkürzung für die Tabelle, wo er gespeichert ist und dem Namen der Tabelle, woher er kommt (dort ist er als primärer Schlüssel vorhanden). Diese Namenskonvention trägt der guten Übersichtlichkeit bei.

Der Felddatentyp für die Speicherung des fremden Schlüssels muss *Zahl/Long Integer* sein

So wird eine Datenbank gebaut

Der richtige Entwurf von Tabellen nach Prinzipien der Datenmodellierung ist der wichtigste Faktor für die gute Funktionalität einer Datenbankapplikation. Die Theorien, wie z.B. die so

	PERS_ID	Nachname	Vorname	Geburtsdatum	Firmenname	Firmentelefonnummer
▶	1	Schwarz	Alois	04.04.1954	Billa	123456
	2	Glock	Harald	10.10.1958	Merkur	654321
	3	Kranz	Johann	25.02.1953	SAP-CCC	666777
	4	Grünberg	Josef	05.06.1964	SAP-CCC	666777
	5	Lehner	Martin	03.04.1955	Billa	123456
	6	Korn	Peter	03.06.1958	SAP-CCC	666777
*	(AutoWert)					

Wir haben doch alles schön beisammen. Einige Angaben zur Person, einige zur Firma, wir können alle Mitarbeiter einer Firma selektieren und alles Mögliche und Unmögliche tun.

Es ist doch Einiges einzuwenden

Bevor Sie über folgende Fragen nachdenken, gehen Sie davon aus, dass eine solche Tabelle nicht sechs sondern sechs Millionen Datensätze hat. Die Millionen sind nicht selten, stellen Sie sich vor, wie viele Kunden, Artikel oder Bestellungen eine große Firma hat.

- Werden Sie den Firmennamen und die Firmentelefonnummer bei jeder dort angestellten Person neu eingeben wollen?
- Werden Sie bei der Änderung des Firmennamens oder der Firmentelefonnummer mehrere Tausend Einträge ändern wollen?
- Sehen Sie deutlich den Unterschied zwi-

Fremder Schlüssel

Sie wissen schon, was der primäre Schlüssel einer Tabelle ist. Es ist die Nummer, mit der jeder Datensatz eindeutig identifiziert wird. Wir nummerieren so die Personentabelle und auch die Firmentabelle. Und den primären Schlüssel der Firma speichern wir als ein neues Feld in die Personentabelle. Er ist zwar ein Schlüssel, der aber in der Personentabelle fremd ist, wird also als **fremder Schlüssel** bezeichnet.

Aha, dann muss ich aber, wenn ich eine Person eingabe immer in die Firmentabelle schauen, damit ich die Nummer weiß, die ich als fremden Schlüssel eintragen soll?

Natürlich nicht!

Die Verknüpfung der Tabellen wird über ein Nachschlagfeld organisiert, fast genau so, wie die Eingabe der Kategorie in unsere erste Ta-

genannte Normalformen würden den Rahmen dieses Kurses sprengen. Wir kommen in unserer Datenbank mit einfachen Fragen und Überlegungen auch ohne die exakte Theorie zum richtigen Ergebnis.

Entitäten

Zuerst müssen so genannte Entitäten gefunden werden. Die Entitäten sind Objekte der realen Welt, die aber nicht nur physikalische Gegenstände oder Lebewesen sein können, sondern auch z.B. Ereignisse, Vorgänge u.v.a.). Die Zusammenfassung aller Entitäten des gleichen Typs (ein so genannter Entitätstyp) wird in der Datenbank als eine Tabelle implementiert.

In unserem ersten Beispiel erkennen wir zwei Entitäten

- Person

	FIRMA_ID	Firmenname	Firmentelefonnummer
▶	2	Merkur	654321
	3	SAP-CCC	666777
*	(AutoWert)		

Datensatz: 1 von 2

	PERS_ID	Nachname	Vorname	PERS_FIRMA_ID
▶	2	Glock	Harald	Merkur
+	3	Kranz	Johann	SAP-CCC
+	4	Grünberg	Josef	SAP-CCC
+	6	Korn	Peter	SAP-CCC
*	(AutoWert)			0

Datensatz: 1 von 4

schen SAP-CCC und SAP- CCC? Im zweiten Fall ist ein Leerzeichen mehr im Namen, das kann beim Tippen leicht passieren. Was ist die Folge? Wenn Sie alle Mitarbeiter von SAP-CCC suchen, finden Sie die SAP- CCC nicht!

So geht's einfach nicht! Oder nur schwierig, langsam und fehleranfällig.

Was ist die richtige Lösung?

Die Personen und die Firmen müssen wir voneinander trennen. Die Angaben (in Weiterem Attribute genannt) zu einer Person speichern

belle. Sie werden die Firmen aus ihren Namen auswählen, nach der Auswahl wird aber nicht der Name, sondern der Schlüssel der Firma als fremder Schlüssel in die Personentabelle gespeichert. In der Datenblattansicht selbst wird der Name der Firma, nicht der numerische Schlüssel angezeigt.

Wenn der Firmenname geändert wird, wird die Änderung nur einmal in der Firmentabelle durchgeführt, alle (egal wie viele) Einträge in der Personentabelle werden automatisch geändert.

- Firma

Zwischen den Entitäten existieren Beziehungen, die man mit folgenden Aussagen beschreiben kann:

- **Eine Firma kann mehrere Personen beschäftigen.**
- **Eine Person kann bei einer Firma angestellt werden.**

Beziehung 1:N

Die Beziehung wird durch die Häufigkeit der Auftritte des Schlüsselwertes in beiden Entitäten charakterisiert.

Auf der Firmenseite kommt jeder Firmenschlüssel nur **einmal** vor.

Auf der Personenseite kann jeder Firmenschlüssel **mehrmals** vorkommen, weil mehrere Personen bei der gleichen Firma arbeiten können.

Solche Situation wird als die **Beziehung 1:N** bezeichnet.

Beziehung M:N

Die Struktur des Datenmodells, besonders die Problematik der Beziehungen hängt stark von der Aufgabenstellung ab. Die Beziehung 1:N ist in Ordnung, wenn es darum geht zu den Personen die Information über **nur eine Firma** (die aktuelle) zu speichern. Für folgende Aufgaben reicht es aber nicht:

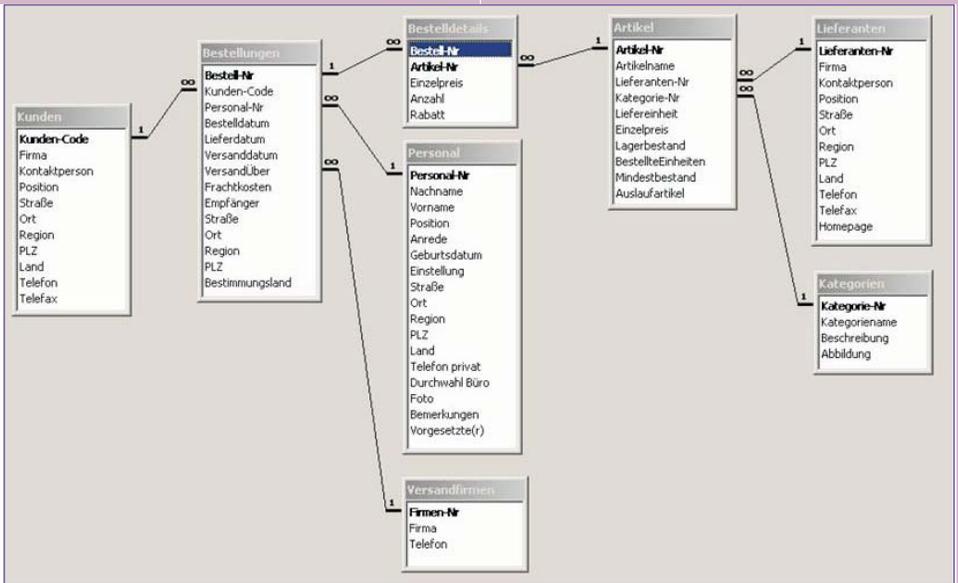
- Eine Person kann für mehrere Firmen gleichzeitig tätig sein (z.B. Berater)
- Sie wollen den ganzen Lebenslauf einer Person speichern, dann brauchen Sie auch meistens die Möglichkeit, mehrere Firmen einer Person zuzuordnen.

Wenn Sie die oben erklärte Beziehung 1:N einfach umdrehen, könnten Sie einer Person mehrere Firmen zuordnen, nicht aber gleichzeitig einer Firma mehrere Personen.

Die Beziehung, **wo auf beiden Seiten die Schlüsselwerte mehrmals vorkommen**, wird als die **Beziehung M:N** bezeichnet. Sie lässt sich nur mit Hilfe einer dritten Entität, einer so genannter **Beziehungsentität** realisieren.

Wir benennen die Entität *Anstellung*. Die entsprechende Tabelle *tblAnstellung* wird folgende Felder haben:

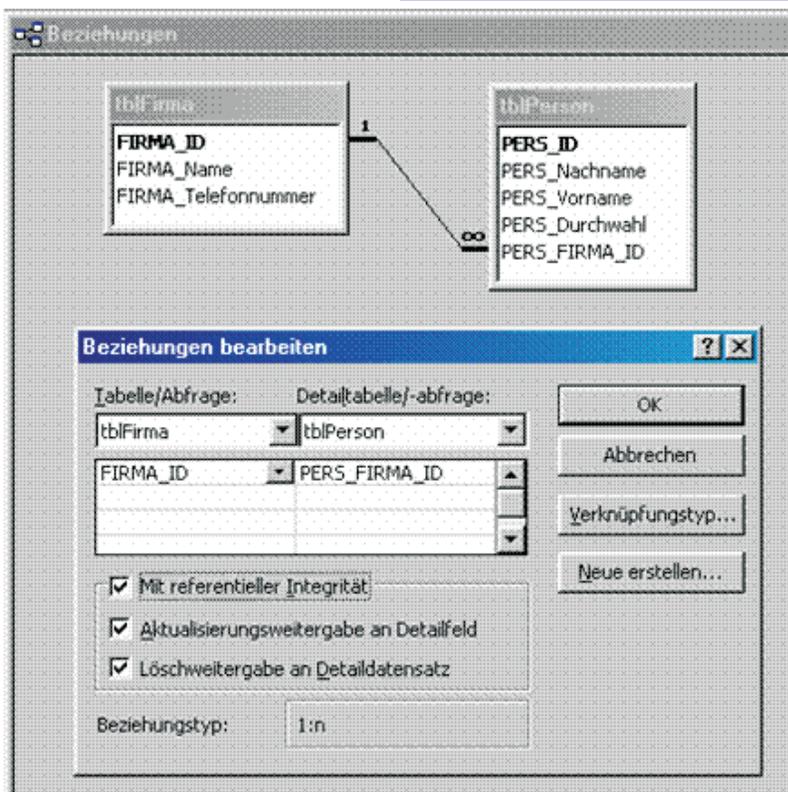
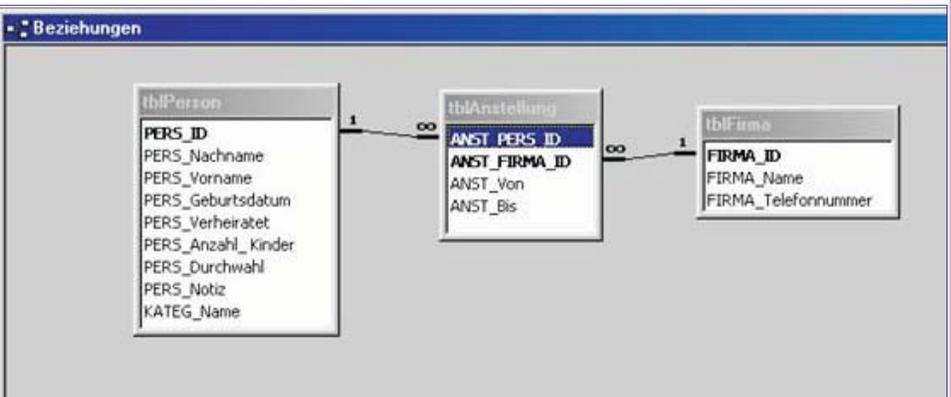
ANST_PERS_ID	Prim.Schlüssel der Person
ANST_FIRMA_ID	Prim.Schlüssel der Firma
ANST_Von	Anfang der Anstellung
ANST_Bis	Ende der Anstellung



Datenmodell der Beispieldatenbank nordwind.mdb

tblGeschenkeFalsch : Tabelle						
	Nachname	Vorname	Firmenname	2003	2004	2005
▶	Schwarz	Alois	Billa	Wein + Karte	Karte	Karte + Buch
	Glock	Harald	Merkur	Buch + Karte	Buch, Karte	Karte, Buch
	Kranz	Johann	SAP-CCC	Wein	Wein	Karte
	Grünberg	Josef	SAP-CCC	Karte	Wein	Karte
*						

Tabelle zur Verteilung von Weihnachtsgeschenken



Die Felder ANST_Von und ANST_Bis gehören weder zur Person noch zur Firma – es sind Attribute der Beziehungsentität.

Das Beziehungsfenster

Die mit dem Nachschlage-Assistenten erstellten Tabellenbeziehungen können im Beziehungsfenster (Menü *Extras->Beziehungen*) um die referentielle Integrität ergänzt werden. Sie garantiert die Konsistenz der Datenbank.

Die Beziehung 1:N schaut im Beziehungsfenster so aus:

Symbol 1 bezeichnet die 1-Seite, das „Unendlich“ Symbol die N-Seite der Beziehung.

Klicken Sie mit der rechten Maustaste auf die Verbindungslinie und wählen „*Beziehung bearbeiten*“ aus.

Mit dieser Einstellung werden

- nach der Änderung des primären Schlüssels der Firma auch der fremde Schlüssel in der Personentabelle geändert
- beim Löschen einer Firma auch gleichzeitig alle ihr zugeordneten Personen gelöscht.

Die Beziehung M:N schaut im Beziehungsfenster so aus:

Aus dem Bild ist ersichtlich, dass die Beziehungsentität die Beziehung M:N eigentlich in zwei Beziehungen 1:N umwandelt.

Übungen

1. Wie wird die Definition einer Anstellung eingeschränkt, wenn Sie in der Tabelle `tblAnstellung` die Kombination der Felder `ANST_PERS_ID` und `ANST_FIRM_ID` als primären Schlüssel definieren?

2. Auf dem ersten Bild auf dieser Seite ist das Datenmodell der Beispieldatenbank `Nordwind.mdb`, die mit MS Access mitgeliefert wird. Finden Sie heraus, zwischen welchen Entitäten eine Beziehung 1:N und wo eine M:N zustande kommt und warum.

Warum wird der `Einzelpreis` sowohl in der Tabelle `Artikel` als auch in der Tabelle `BestellDetails` gespeichert? Ist es nicht eine redundante Datenhaltung?

3. Eine Applikation für die Verteilung der Weihnachtsgeschenke speichert alle Daten in einer Tabelle (siehe Bild vorige Seite).

- Was ist der größte Nachteil dieser Lösung, wenn mit der Datenbank die Sekretärin selbst arbeiten soll?
- Entwerfen Sie das ordentliche Datenmodell für diesen Zweck.

Datenmodell der CD-Evidenz

Nach dem schwierigen aber wichtigen Kapitel über die Datenmodellierung werden wir uns jetzt nur mehr unserem praktischen Ziel widmen – dem Aufbau der Datenbank für die CD-Evidenz.

Aufgabenstellung

Unsere Datenbank soll die Informationen über die CDs und ihre Inhalte speichern.. Die CDs sollen vordefinierten Kategorien zugeordnet werden. Für jede CD werden die Namen aller Lieder (entsprechen den Spuren auf der CD – es können auch Musik- oder Textstücke sein) und die Namen der Interpreten gespeichert.

Für die Eingabe und Anzeige der Daten werden Formulare erstellt. Es soll möglich sein, eine CD nach dem Namen oder Kategorie zu suchen und den gesamten Inhalt in einem Formular anzuzeigen.

Es werden auch zwei Berichte implementiert.

- Die Gesamtübersicht aller CDs nach Kategorien.

- Der Inhalt der ausgewählten CD mit allen Liedern und Interpreten.

Alle Funktionen des Programms werden aus einem Hauptformular über die Schaltflächen aufgerufen. Dieses Formular wird automatisch angezeigt, wenn die Datenbank geöffnet wird.

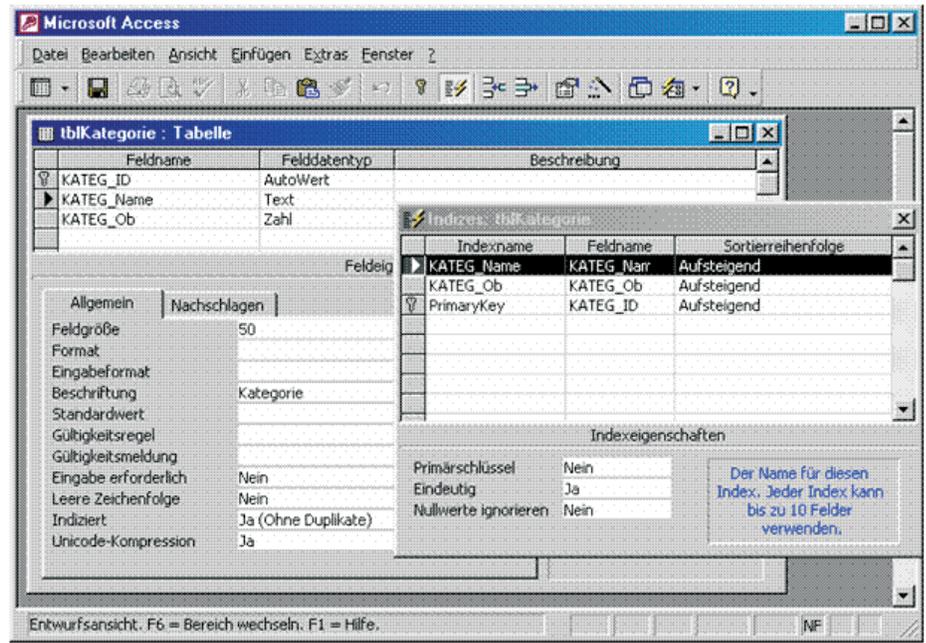
Basis-Entitäten

Der erste Schritt beim Entwurf des Datenmodells ist die Definition der notwendigen Entitäten und ihrer Attributen. Jede Entität wird in der Datenbank in Form einer Tabelle dargestellt.

Die Attribute werden die Datenfelder der Tabelle bilden. Wir werden richtig unterscheiden müssen, welche Attribute zu welcher Entität zugeordnet werden.

CD-Packung

Die erste Entität, die wir brauchen werden, ist die CD selbst. Wir werden aber nicht die einzelne Scheibe, sondern die gekaufte Packung als eine Entität betrachten. Oft befinden sich näm-



Beschreibung	Feldname	Felddatentyp	Bemerkung
nichtnotwendig	KATEG_ID	Autowert	Primärer Schlüssel
Kategorie	KATEG_Name	Text	Länge 50 Zeichen, keine Duplikate möglich

Beschreibung	Feldname	Felddatentyp	Bemerkung
nichtnotwendig	INTR_ID	Autowert	Primärer Schlüssel
Interpret	INTR_Name	Text	Länge 255 Zeichen, keine Duplikate
Geburtsdatum	INTR_Geburtsdatum	Datum/Uhrzeit	
Geburtsort	INTR_Geburtsort	Text	Std.Länge 50 Zeichen
Geburtsland	INTR_Geburtsland	Text	Std.Länge 50 Zeichen
Telefon	INTR_Telefon	Text	Std.Länge 50 Zeichen
E-Mail	INTR_Email	Text	Std.Länge 50 Zeichen
Notiz	INTR_Notiz	Memo	ein beliebig langer Text

Beschreibung	Feldname	Felddatentyp	Bemerkung
nichtnotwendig	LIED_ID	Autowert	Primärer Schlüssel
Lied	LIED_Titel	Text	Länge 255 Zeichen, keine Duplikate

Beschreibung	Feldname	Felddatentyp	Bemerkung
nichtnotwendig	CDP_ID	Autowert	Primärer Schlüssel
Titel	CDP_Titel	Text	max. Länge 255 Zeichen
Kategorie	CDP_KATEG_ID	Zahl/Long Int.	Nachschlagen aus Tabelle <code>tblKategorie</code>
Anzahl der CDs	CDP_Anzahl_CD	Zahl/Integer	
Ausgabejahr	CDP_Ausgabejahr	Zahl/Integer	
IdentifikationsNr	CDP_IdentNr	Text	Std.Länge 50 Zeichen

lich mehrere CDs in einer Packung. Deswegen benennen wir auch die Entität CD-Packung.

Die Tabelle `tblCdPackung` wird folgende Datenfelder enthalten:

Im Feld `IdentifikationsNr` wird die eindeutige Bezeichnung der CD-Packung gespeichert, die an der Verpackung angedruckt ist. Es handelt sich eigentlich um keine richtige Nummer, weil es auch Buchstaben enthalten kann. Das entsprechende Datenfeld wird also den Felddatentyp `Text` haben, damit auch Buchstaben eingegeben werden können.

Kategorie

Die Tabelle `tblKategorie` wird ähnliche Funktion haben, wie in der ersten Datenbank mit den Personendaten. Wir machen sie aber diesmal etwas ordentlicher:

- mit dem richtigen primären Schlüssel vom Typ `Autowert`
- mit einem zusätzlichen Feld `Ordnungsbegriff`. In dieses Feld werden wir eine `Zahl` eintragen

und die Datensätze nach diesem Feld sortieren. Das ermöglicht uns, die Kategorien in einer vordefinierten Reihenfolge anzeigen zu lassen.

Die Duplikate in einem anderen Datenfeld als dem primären Schlüssel vermeiden Sie dadurch, dass Sie für die letzten zwei Felder in der Eigenschaft `Indiziert` den Eintrag `Ja (Ohne Duplikate)` auswählen.

Alle für diese Tabelle definierten Indizes können Sie dann über die Schaltfläche mit dem Blitzsymbol in einem neuen Fenster anzeigen lassen.

Übung

- Öffnen Sie die Datenbank `CD_Evidenz.mdb`
- Erstellen Sie die Tabelle `tblKategorie` und befüllen Sie sie mit den passenden Einträgen für die CD-Evidenz (Klassik, Rock, Country, ...). Definieren Sie je nach Ihren Vorlieben die gewünschte Sortierung mit dem Feld `KATEG_Ob`

- Erstellen Sie die Tabelle `tblCdPackung` und definieren Sie das Feld `Kategorie` als ein Nachschlagefeld. Lassen Sie das Feld `KATEG_ID` als fremden Schlüssel speichern und das Feld `KATEG_Name` bei der Eingabe anzeigen.

- Tragen Sie einige Datensätze ein, testen Sie dabei die Funktionalität des Felds `Kategorie`. Sie werden sehen, dass die Sortierung nach Ihrem Ordnungsbegriff noch nicht funktioniert. Diese Funktionalität werden wir noch später bei den Abfragen ergänzen.

Interpret

Warum wird der Interpret als eine eigene Entität modelliert und nicht als Attribut der CD-Packung gespeichert? Der Grund ist der, dass es CDs gibt, die mehrere Interpreten enthalten. Wenn wir den Interpreten in einem Feld der Tabelle `tblCdPackung` speichern würden, könnten wir dann nicht mehrere Interpreten zu einer CD-Packung angeben...

Beim Entwurf dieser Entität werden wir eine Vereinfachung akzeptieren. Wir werden als den Interpreten eventuell auch eine Gruppe speichern. Die Felder, die in dem Fall nicht sinnvoll ausgefüllt werden können, lassen wir einfach leer.

Die Tabelle `tblInterpret` kann folgende Datenfelder enthalten:

Lied

Wie schon oben erwähnt, wird diese Entität Lied genannt, weil die meisten CD-Inhalte Lieder sind. In Wirklichkeit werden wir aber darunter die Spuren auf der CD verstehen. Wir brauchen nichts anderes speichern als primären Schlüssel und den Titel. Deswegen wird die Struktur der Tabelle `tblLied` nur aus zwei Datenfeldern bestehen.

Übung

Erstellen Sie die Tabellen `tblInterpret` und `tblLied` und befüllen Sie sie mit einigen Datensätzen.

Beziehungsentität – Daten werden verknüpft

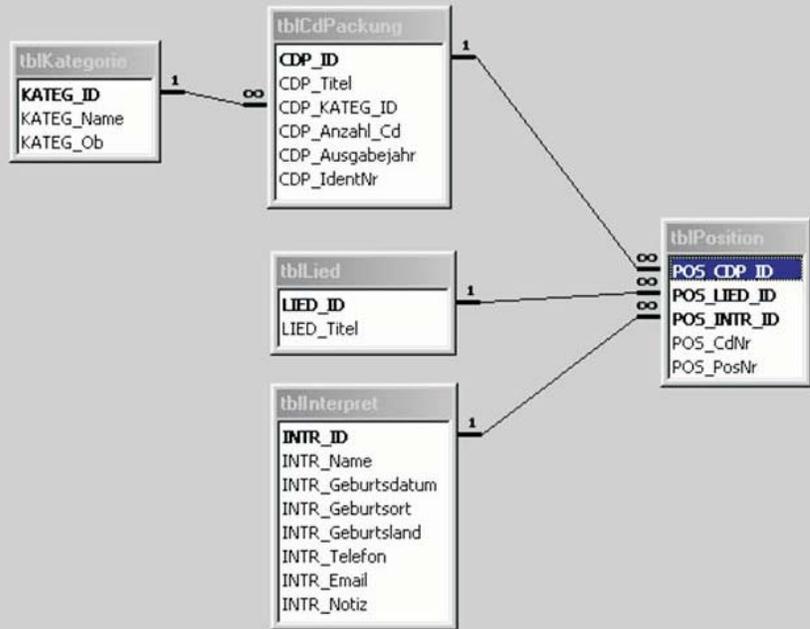
Wir haben jetzt Tabellen für unsere Basis-Entitäten fertig. Wir können in der Tabelle `tblCdPackung` die Informationen über die CD-Packung speichern, die Tabelle `tblKategorie` gibt uns die erlaubten Werte für die Kategorie vor. Die Tabellen `tblInterpret` und `tblLied` enthalten die Daten über die Interpreten und die Lieder.

Wir müssen jetzt die Beziehungen zwischen den Daten erstellen. Es ist nicht ganz trivial darauf zu kommen, wie das am besten zu machen ist.

- Wenn wir den Schlüssel der CD-Packung in den Tabellen mit den Interpreten und den Liedern als fremden Schlüssel speichern, können wir damit die Situation abbilden, wenn in einer CD-Packung mehrere Interpreten und mehrere Lieder sind.

- Es wird aber nicht möglich sein, dass ein Interpret oder ein Lied in mehreren CD-Packungen enthalten sind und das wäre ein wesentlicher Mangel.

- Ein Lied kann auch unterschiedliche Interpreten auf unterschiedlichen CDs haben, daran wäre auch nicht gedacht. Aus dem Grund



tblPosition

Beschriftung	Feldname	Feldtyp
CD-Packung	POS_CDP_ID	Zahl/Long Integer
Interpret	POS_INTR_ID	Zahl/Long Integer
Lied	POS_LIED_ID	Zahl/Long Integer
CD-Nummer	POS_CDNr	Zahl/Integer
Position	POS_PosNr	Zahl/Integer

können wir die Interpreten und Lieder nicht fix miteinander verknüpfen.

Wir brauchen daher eine neue Entität, die jede Ausprägung einer Beziehung zwischen der CD-Packung, Interpret und Lied speichern kann. Diese Beziehung entspricht einer Position auf der CD, deswegen wird die Beziehungsentität Position genannt und mit der Tabelle `tblPosition` realisiert.

Struktur der Tabelle tblPosition

Wenn die Tabelle die Beziehungen zwischen drei Tabellen speichern soll, muss sie die primären Schlüssel der Tabellen als fremde Schlüssel enthalten. Diese drei Felder werden gemeinsam den primären Schlüssel der Tabelle `tblPosition` bilden.

Die Beziehungsentität hat auch eigene Attribute, die sonst in keiner anderen Tabelle gespeichert werden können. Es sind in unserem Fall die Nummer der CD in der Packung und die Nummer der Position (Spur) auf der CD.

Übung

- Erstellen Sie die Tabelle `tblPosition`
- Definieren Sie die ersten drei Felder als Nachschlagfelder mit dem Nachschlage-Assistenten
- Ändern sie nachträglich die Namen der Nachschlagfelder, damit sie unseren Namenskonventionen entsprechen.
- Probieren Sie die Funktionalität der Nachschlagfelder in der Tabelle `tblPosition` aus.
- Ergänzen Sie die Beziehungen und referentielle Integrität über Menü `Extras->Beziehungen`

- Löschen Sie einen Datensatz in der Tabelle `tblCdPackung`, `tblInterpret` oder `tblLied` und beobachten Sie die Veränderungen in der Tabelle `tblPosition`.

Beziehungen und referentielle Integrität

Nachdem wir die Beziehungen zwischen den Tabellen und referentielle Integrität im Beziehungsfenster definiert haben, ist das Datenmodell der CD-Evidenz komplett fertig.

Datenmodell ist nicht immer gleich

Die Datenmodellierung hat bestimmte Grundsätze, aber keine genauen Vorgaben für den Aufbau der Entitäten und Beziehungen. Die Struktur hängt stark von der Aufgabenstellung ab. Für unseren Zweck wird es reichen, wenn wir als sowohl eine Einzelperson als auch eine Gruppe in der Interpretentabelle speichern. Uns geht es in erster Reihe darum, die Übersicht über vorhandenen CDs, Lieder und Interpreten zu haben.

Übung

Überlegen Sie, welche Entitäten und Beziehungen Sie zusätzlich brauchen werden, wenn Sie in Ihrer Datenbank auch folgende Daten und Zusammenhänge speichern möchten. Wann brauchen Sie keine neue Entität sondern nur ein neues Attribut zu einer bestehenden Entität?

- Die Spieldauer eines Lieds.
- Der komplette Text des Liedes. Welcher Feldtyp ist dafür geeignet?
- Sie wollen für das Geburtsland nur aus einer vordefinierten Menge von Werten auswählen.
- Autoren des Textes und Autoren der Musik zu einem Lied.
- Verleih von CDs an Bekannte – nutzen Sie dafür die schon existente Personentabelle.