

## 38 Mitgliederverwaltung

### 38.1 Anwendungsentwurf

Verbände, Vereine und ähnliche Organisationen haben auch für die betriebliche Praxis eine enorme Bedeutung. In der Firma Euromedia GmbH wird ein Leseclub organisiert. Dabei ist sowohl eine persönliche Mitgliedschaft als auch eine Firmenmitgliedschaft möglich.

Um die Vielzahl der anfallenden Daten besser in den Griff zu bekommen, bietet sich eine Anwendungsentwicklung mit einer Datenbank geradezu an. Dabei werden folgende Teilprozesse unterschieden:

- ❖ Schwerpunkt ist zunächst einmal die Pflege der Mitgliederdaten. Neben wichtigen Grunddaten (etwa Adressen, Telefonnummern etc.) empfiehlt sich oft eine Kategorisierung der Mitglieder (beispielsweise Art der Mitgliedschaft oder Zuordnung zu Regionen).
- ❖ Ein besonderes Problem bei einer Mitgliederverwaltung ist das Erstellen der Beitragsrechnungen sowie die Kontrolle der Zahlungseingänge (einschließlich des Mahnwesens). Zu berücksichtigen sind im Regelfall verschiedene Beitragsklassen.
- ❖ Schließlich werden verschiedene Listen für Auswertungszwecke benötigt sowie für Aussendungen aktuelle Etikettendrucke gewünscht. Mit einer gut organisierten Datenbanklösung dürfte es kein Problem sein, all dies per Knopfdruck zu erhalten.

Die folgende Übersicht gibt die Tabellen wieder, die unserer Anwendung zur Mitgliederverwaltung zugrundeliegen:

<b>Tabelle</b>	<b>Erläuterung</b>
<i>Personen</i>	Hier werden die Daten zu den persönlichen Mitgliedern gespeichert.
<i>Firmen</i>	Um Firmenmitgliedschaften zu verwalten, werden alle Firmendaten in einer gesonderten Tabelle gespeichert.
<i>Mitgliederinfo</i>	Alle Informationen, die die Mitgliedschaft betreffen, werden in der Tabelle <i>Mitgliederinfo</i> aufgenommen. Je nach Mitgliedstypus enthält die Datenbank entweder eine Personalnummer oder eine Firmennummer.
<i>Jahre</i>	Organisiert die Verwaltung nach Mitgliedschaftsjahren bzw. die dabei eventuell angefallenen Mahnungen.

<b>Tabelle</b>	<b>Erläuterung</b>
<i>Beiträge</i>	Für jedes Jahr werden alle Beitragskürzel und Beitragssummen verwaltet.
<i>Jahresbeiträge</i>	Die Tabelle enthält die Bewegungsdaten. Es werden für jedes Mitglied jedes Jahr der Beitrag bzw. das Zahlungsdatum verwaltet.
<i>Zeitschriften</i>	Im Abo über die Mitgliedschaft erhältliche Zeitschriften.
<i>Zeitschriften-zuordnung</i>	Für jedes Jahr und jedes Beitragskürzel die entsprechende Zuordnung. Damit ist sichergestellt, daß ein Mitglied mehrere Zeitschriften beziehen kann, umgekehrt eine Zeitschrift von mehreren Mitgliedern bezogen werden kann.
<i>Regionen</i>	Meeting-Regionen des Buchklubs werden hier aufgenommen.
<i>RegionenPlz</i>	Nebentabelle, um eine Zuordnung der PLZ zu den einzelnen Meeting-Regionen zu organisieren.

Interessant sind natürlich auch die existierenden Beziehungsmöglichkeiten. Dazu kann die folgende Abbildung als Orientierung dienen:

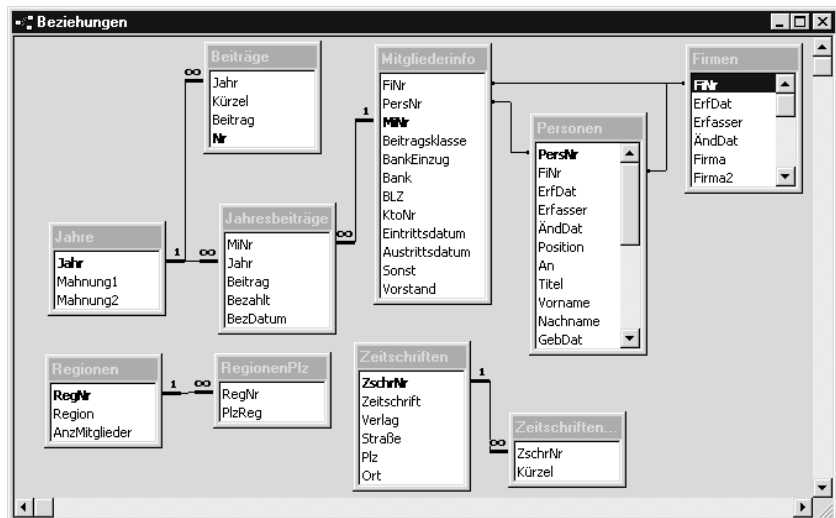


Bild 38.1: Beziehungen



*Es gibt Personen, die Firmen angehören, es gibt Personen, die Mitglieder sind, es gibt Firmen, die Mitglieder sind.*

Am Beispiel der Anwendung zur Mitgliederverwaltung können Sie folgende Besonderheiten des Arbeitens mit Accesskennen lernen:

- ❖ Ausgehend von erzeugten Berichten wird Ihnen gezeigt, wie Sie das Handling mit unterschiedlichen Berichten organisieren können. Ab einer bestimmten Anzahl wird der Aufruf von Berichten nämlich sehr umständlich. Abhilfe schafft hier ein besonderes Druckmenü. Wie Sie dies erzeugen, zeigen wir Ihnen mit dieser Anwendung *Mitgliederverwaltung*.
- ❖ Ein weiteres Problem, das Sie anhand der Mitgliederverwaltung kennenlernen können, ist die Möglichkeit, wie Sie über Programmierung einen bestimmten Datensatz hinzufügen können. Vielleicht gibt es auch bei Ihnen Anwendungen, wo Sie das gern nutzen möchten und so mitunter Zeit sparen und exakter arbeiten.
- ❖ Am Schluß gehen wir auf das Erstellen eines Suchfensters ein. Dazu benötigen Sie eine ausgefeilte VBA-Prozedur.

## 38.2 Anwendungsfunktionen und -prozesse



*Die folgende Lösung finden Sie auf Ihrer CD-ROM unter dem Namen VEREINSVERWALTUNG.MDB gespeichert. Nach dem Aufruf der Anwendung steht das folgende Hauptmenü zur Verfügung:*

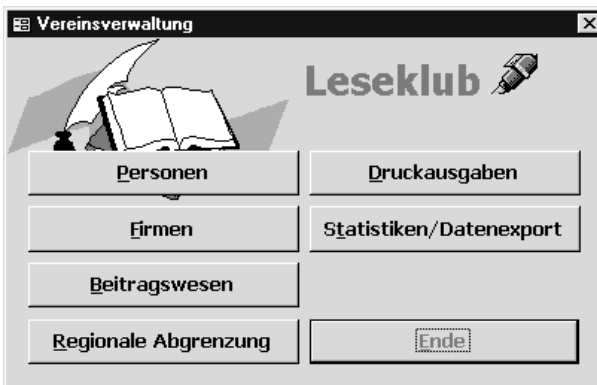


Bild 38.2: Hauptmenü

Die angezeigten Schaltflächen haben folgende Bedeutung:

- ...❖ *Personen*: Nach einem Mausklick erscheint eine Bildschirmmaske zum Erfassen/Verwalten der persönlichen Mitgliedschaften.
- ...❖ *Firmen*: Es wird eine besondere Maske für das Erfassen/Verwalten von Firmenmitgliedschaften bereitgestellt.
- ...❖ *Beitragswesen*: In diesem Bereich können die Daten erfasst und verwaltet werden, die mit der Beitragsverwaltung in Bezug stehen. Beispiele sind das Anlegen der Beiträge und Beitragsjahre sowie das Anzeigen der bezahlten/offenen Beiträge.
- ...❖ *Regionale Abgrenzung*: Zur Organisation dezentraler Veranstaltungen sind die Mitglieder bestimmten Regionen zugeordnet. Nach Aktivieren der Schaltfläche kann die Regionen-Zuordnung geändert und so neuen Anforderungen angepaßt werden.
- ...❖ *Druckausgaben*: Es werden verschiedene Möglichkeiten zur Druckausgabe bereitgestellt. Aus einem Dialogfeld können bestimmte Etiketten bedruckt oder ausgewählte Listen erzeugt werden (z. B. Mitgliederlisten).
- ...❖ *Statistiken/Datenexport*: Datenexport, vor allem in Word-Steuerdateien.

Die Schaltfläche *Ende* bewirkt den Ausstieg aus der Anwendung und aus Access selbst.

### 38.2.1 Persönliche Mitglieder erfassen und verwalten

Durch Mausklick auf die Schaltfläche *Personen* ergibt sich die folgende Bildschirmmaske für das Erfassen und Verwalten persönlicher Mitgliedschaften:

**Persönliche Mitgliedschaft**

PersNr: 7171 ErfDat: 01.05.96 Erfasser: KK ÄndDat: 01.05.99

Position: An: Herr Titel: Dr.

Nachname: Eifrig Vorname: Friedrich

Geburtsdat.: 12.03.62 ☒ Firmenmitgl.

Straße: Schnellweg 3 Metall AG  
Einkauf  
Eisenweg 3  
D-70563 Stuttgart

Land: D Plz: 70563 Ort: Stuttgart

Anrede: Sehr geehrter Herr Doktor Privattelefon:

Telefon: Mobil: Telefax:

Branche:

MiNr: 20000 Beitragsklasse: DD

Eintrittsdatum: 14.09.91 ☐ BankEinzug

Austrittsdatum: Bank: BLZ: KtoNr:

Sonst: ☐ Vorstand

Buttons: Bearbeiten, Neu, Löschen, Suche, Beitrag, Firmenmitgl., Schließen

Navigation: 1 von 4

Alphabetische Suche: A Ä B C D E F G H I J K L M N O Ö P Q R S T U Ü V W XYZ Alle

Bild 38.3: Persönliche Mitgliedschaften

Über diese Bildschirmmaske können Sie gezielt die Daten der persönlichen Mitglieder des Leseklubs von Euromedia verwalten. Dazu müssen Sie unter Umständen zunächst über die Schaltflächen am Rand rechts unten einen gewünschten Datensatz, den Sie bearbeiten wollen, aktivieren. Die Datensatznavigation erfolgt in allen Fenstern über die Navigationsflächen. Diese Navigationsschaltflächen werden im Bearbeitungsmodus allerdings ausgeblendet, um sicherzustellen, daß nicht zu einem anderen Datensatz gewechselt werden kann, obwohl der aktuelle noch nicht gesichert ist.



Bild 38.4: Benutzerdefinierte Navigationsschaltflächen

Die Bedeutung und Anwendung der am rechten Rand der Bildschirmmaske angebotenen Schaltflächen zeigt die folgende Zusammenstellung:

- ❖ *Bearbeiten*: Sie aktivieren den Bearbeitungsmodus, mit dem ein Datensatz zur Editierung freigegeben wird. So kann eine Adresse geändert oder beispielsweise eine neue Telefonnummer erfaßt werden.
- ❖ *Neu*: Anlegen eines neuen Mitglieds. Einige Felder wie *PersNr* und *ErfDat* enthalten bereits standardmäßig Einträge. In wichtigen Feldern wie beispielsweise *Nachname* und *Beitragsklasse* ist die Eingabe obligatorisch.
- ❖ *Löschen*: Ermöglicht das Löschen des aktiven Datensatzes. Es wird dazu zunächst eine Sicherheitsabfrage gestellt, ob der markierte Datensatz tatsächlich gelöscht werden soll.
- ❖ *Suche*: Wenn Sie auf diese Schaltfläche klicken, erscheint ein Dialogfeld zur Eingabe der Suchkriterien. Es kann entweder über die Mitgliedsnummer oder über Namens- und Adressenteile gesucht werden. Ist im Feld *MiNr* etwas eingetragen, kann in den restlichen Feldern keine Eingabe erfolgen und umgekehrt. Hinweis: Wie Sie dieses Problem programmtechnisch lösen, zeigen wir Ihnen am Ende des Kapitels.
- ❖ *Beitrag*: Mit dieser Schaltfläche gelangen Sie zur Beitragsverwaltung. Dort wird automatisch jenes Mitglied angezeigt, das zuletzt im Formular *Personenmitglieder* aktiv war.
- ❖ *Firmenmitglieder*: Durch den Wechsel zur Maske *Firmenmitgliedschaften* kann flexibel ein Bezug zwischen den persönlichen und den institutionellen Mitgliedschaften hergestellt werden.
- ❖ *Schließen*: Durch Klicken auf *Schließen* wird das aktuelle Formular verlassen und zum Hauptmenü zurückgekehrt.

Weitere Schaltflächen können – wie Ihnen beim Testen der Anwendung sicher schon aufgefallen ist – nach einer Aktivierung der zuvor beschriebenen Schaltflächen anfallen. Die möglichen Schaltflächen haben folgende Bedeutung:

- ...❖ *Abbrechen*: Diese Schaltfläche taucht auf, wenn die Schaltfläche *Neu* oder *Bearbeiten* aktiviert wurde. Sie können damit die Dateneingabe abbrechen bzw. eine vorgenommene Korrektur von Daten verwerfen.
- ...❖ *Übernehmen*: Hiermit kann eine Änderung oder Neuerfassung bestätigt werden. Bei der Übernahme eines neuen Datensatzes wird gefragt, ob die Person/Institution den Beitrag für das lfd. Jahr bereits bezahlt hat. Damit kann eine Integration zur Beitragsverwaltung vorgenommen werden.
- ...❖ *Firma*: Das aktuelle Mitglied kann einer Firma zugeordnet werden. Sofern die Zuordnung zu einer Firma schon vorgenommen wurde, ist diese im folgenden Listenfeld bereits als Vorgabe angezeigt.

Schließlich ist auch noch eine Filterung von Datensätzen möglich. Dazu dienen die folgenden Schaltflächen:



Bild 38.5: Filterschaltflächen

Sie haben eine einfache über Makros gelöste Variante dieser Filterschaltflächen bereits in der Anwendung ADRESS4 kennengelernt.

- ...❖ Eine Filterung von Datensätzen wird automatisch aufgehoben, wenn über das Suchfenster nach einem Namen gesucht wird.
- ...❖ Als Orientierung für die Benutzung wird der gefilterte Buchstabe immer rechts von den Schaltflächen angezeigt.
- ...❖ Sofern nach einem Namen über das Suchfenster gesucht wird, wird dem Benutzer durch ein »?» angezeigt, daß ein Filter aktiv ist, der durch Klicken auf *Alle* wieder aufgehoben werden kann.
- ...❖ Ist kein Filter aktiv, wird ein Bindestrich angezeigt.

### 38.2.2 Firmenmitgliedschaften erfassen und verwalten

Gerade bei Vereinen und Verbänden ist es üblich, bestimmte Gruppen von Mitgliedern zu bilden. Im Beispielfall der Firma Euromedia GmbH ist die Unterscheidung zwischen persönlicher und Firmenmitgliedschaft von besonderer Bedeutung.

Um das folgende Dialogfeld für das Erfassen und Verwalten von Firmenmitgliedschaften aufzurufen, aktivieren Sie im Hauptmenü der Mitgliederverwaltung die Schaltfläche *Firmen*:

**Firmenmitglieder**

FINr: 7378 ErfDat: 12.12.94 Erfasser: 28 ÄndDat: 16.03.97

Firma: Metall AG Branche:

Firma2:  Anz: Firma

Abteilung: Einkauf Anrede: Sehr geehrte Damen und Herren,

Anspr.:

Straße/Pf.: Eisenweg 3

Land: D Plz: 70563 Ort: Stuttgart

Telefon: 0711/123456 Telefax: 0711/1234567

MiNr: 20004 Beitragsklasse: EE

BankEinzug ☐

Eintrittsdatum: 02.09.90 Bank:

Austrittsdatum:  BLZ:  KtoNr:

Sonst:

Bearbeiten Neu Löschen Suche Beitrag Personenmitgl. Menü

1 von 3

A Ä B C D E F G H I J K L M N O Ö P Q R S T U Ü V W XYZ Alle -

Bild 38.6: Firmenmitgliedschaften

Die hier angezeigten Schaltflächen haben die gleiche Bedeutung wie bei der Verwaltung der persönlichen Mitgliedschaften. Der einzige Unterschied ist die Schaltfläche *Personenmitglieder*; hierüber kann zu den Personenmitgliedern gewechselt werden.

### 38.2.3 Wie kann eine Beitragsverwaltung erfolgen?

Grundsätzlich soll eine Mitgliederverwaltung nicht nur der Pflege der Adressen dienen, sondern natürlich auch die Ermittlung und Verwaltung der Beiträge sowie die Abwicklung der Zahlungsvorgänge übernehmen.

Wird im Hauptmenü zur Mitgliederverwaltung die Schaltfläche *Beitragswesen* angeklickt, ergibt sich auf dem Bildschirm das folgende Formular:

**Beitragsverwaltung**

**Persönliches Mitglied**

Dr. Friedrich Eifrig  
Metall AG

Einkauf  
Eisenweg 3  
D-70563 Stuttgart

**Firmenmitglied**

MiNr: 20000 Beitragsklasse: 00

Eintritt: 14.09.91 Austritt:

BankEinzug ☐

Bank:

BLZ:  KtoNr:

Sonst:

Suche Stammdaten Rechnung Schließen

1993 Jahr: 1999

1999	100,00	<input type="checkbox"/> Bezahl, am	<input type="text"/>
1998	100,00	<input type="checkbox"/> Bezahl, am	<input type="text"/>

1 von 7

A Ä B C D E F G H I J K L M N O Ö P Q R S Sch St T U Ü V W XYZ Alle -

Bild 38.7: Beitragsverwaltung

Die Suchfunktion kann auch hier über die Schaltfläche *Suche* aufgerufen werden. Über die Schaltfläche *Stammdaten* gelangen Sie zu den Stammdaten des angezeigten Mitgliedes. Das Programm unterscheidet zwischen Personen- und Firmenmitgliedschaften und wechselt automatisch zum richtigen Formular. Eine Beitragsaufstellung kann über die Schaltfläche *Rechnung* ausgedruckt werden.

Über die weiteren Schaltflächen können Jahresbeiträge generiert, widerrufen, Vereinsjahre und Beiträge definiert sowie offene Posten angezeigt werden.

## 38.3 Druckmenü anlegen

Vereinfachen können Sie sich den Aufruf von Berichten, wenn Sie diese in ein Druckmenü einbinden. Dies finden Sie in den meisten Anwendungen zu diesem Buch realisiert.

### 38.3.1 Menüformular aufbauen

Als Basis für das Druckmenü dient ein Formular. Nach der Auswahl eines Berichts aus einer Optionsgruppe oder einem Listefeld kann die komfortable Auswahl zwischen den bereitgestellten Druckvarianten realisiert werden. Zusätzlich bietet sich ergänzend jeweils eine besondere Schaltfläche an, um etwa die Seitenansicht zu aktivieren oder den Druckbefehl unmittelbar auszulösen.

Im Beispielsfall können Sie die Auswahl des Berichts aus einem Listefeld treffen.

Zusätzlich soll für den Fall, daß Adreßetiketten als Auswahl getroffen werden, die Möglichkeit eingebaut werden, eine Auswahloption bereitzustellen, die festlegt, ob alle Adreßetiketten oder nur jene ausgewählter Regionen gedruckt werden. Beispiel:

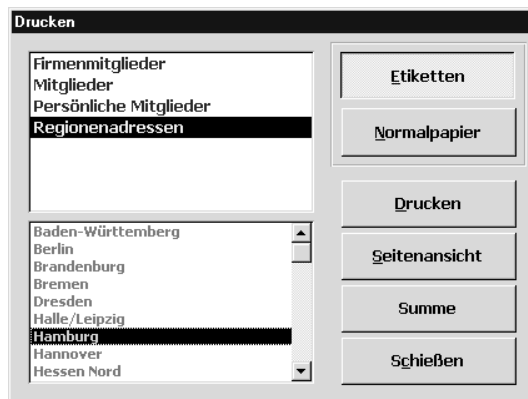


Bild 38.8: Druckmenü



Diese Lösung ist auf verschiedene Weise möglich. Neben einer Makro-Lösung können Sie auch VBA alternativ als Werkzeug wählen. Der Weg über eine VBA-Programmierung wurde in der vorliegenden Mitgliederverwaltung gewählt.

### 38.3.2 Druckmenü als VBA-Lösung

Zunächst zu den Anforderungen, die an das zu erstellende Druckmenü und die damit verbundenen Funktionen gestellt werden:

- ...✚ Die Liste der in das Druckmenü aufgenommenen Berichte soll einfach dynamisch erweiterbar sein.
- ...✚ Die Berichte sollen in zwei Gruppen, nämlich jene für den Ausdruck von Adreßetiketten und jene für Normalpapier, geteilt werden und über diese Gruppe aufrufbar sein.
- ...✚ Jeder Bericht soll sowohl sofort zum Ausdruck gebracht als auch in der Seitenansicht angezeigt werden können.

Zur Realisierung der gestellten Anforderungen werden im Formularentwurf des Druckmenüs folgende Objekte benötigt:

- ...✚ Ein Listenfeld, das dynamisch mit den in der Datenbank gespeicherten Berichten gefüllt wird.
- ...✚ Ein zweites Listenfeld, das zur Auswahl der Region verwendet wird, wenn ein Bericht ausgewählt wird, der die Möglichkeit der Einschränkung auf eine Region vorsieht.
- ...✚ Ein Optionsfeld mit zwei Umschaltflächen, die der Auswahl der Berichte für Normalpapier und für Etiketten dienen. Nach dem Klicken auf eine der beiden Schaltflächen wird das Listenfeld mit den jeweiligen Berichten gefüllt.
- ...✚ Schaltflächen für Druck und Seitenansicht.
- ...✚ Ergänzend findet sich eine Schaltfläche *Summe* für Auswertungen (Neuberechnung der Mitgliederzahlen). Diese Schaltfläche ist immer dann verfügbar, wenn ein Bericht mit der Möglichkeit der regionalen Abgrenzung ausgewählt ist. Ergebnis der Ausführung ist eine Übersicht, die anzeigt, wie viele Adressen/Mitglieder auf jede Region entfallen.
- ...✚ Schließlich findet sich eine Schaltfläche für die Rückkehr zum Hauptmenü.

#### Berichtsnamen einlesen

Wichtig ist, daß beim Öffnen des Formulars die entsprechenden Berichte auch zur Verfügung stehen. So müssen Sie für das Ereignis *Beim Öffnen* mittels einer Ereignisprozedur für das Formular sicherstellen, daß das Listen-

feld *Abfrage* mit den Berichtsnamen gefüllt ist. Der Aufbau der Funktion ist relativ einfach: Sie erstellen eine Funktion, die die Berichtsnamen in einer Zeichenkette zurückliefert, wobei die einzelnen Namen durch ein Semikolon voneinander getrennt werden:

```
Private Sub Form_Open(Cancel As Integer)
    Me!Abfrage.RowSource = EtikettenBerichte()
End Sub
```

Die beschriebene Funktion entspricht genau der Syntax, wie es ein Listensfeld als Datensatzherkunft benötigt, wenn die Eigenschaft *Herkunftstyp* auf *Wertliste* eingestellt ist.

Da im Optionsfeld des Druckmenüs standardmäßig die Berichte für Etiketten ausgewählt sind, wird die folgende Funktion *EtikettenBerichte()* zunächst verwendet, um die Berichtsnamen einzulesen.

```
Public Function EtikettenBerichte()
    Dim db As Database
    Dim berichte As Container, bernamen As Document
    Dim i As Integer
    Dim liste As String
    liste = ""
    Set db = CurrentDb
    Set berichte = db.Containers("Reports")
    For i = 0 To berichte.Documents.Count - 1
        Set bernamen = berichte.Documents(i)
        If Left$(bernamen.Name, 9) = "Etiketten" Then
            liste=liste & Mid$(bernamen.Name,10) & ";"
        End If
    Next
    EtikettenBerichte = liste
    db.Close
End Function
```

Zur Erläuterung der Funktion:

- ...❖ Die Auflistung *Reports* wird dazu verwendet, auf die einzelnen in der Datenbank gespeicherten Berichte zuzugreifen. Jedes Dokument dieser Auflistung repräsentiert einen Bericht, auf dessen Name über die gleichnamige Eigenschaft des Dokuments zugegriffen werden kann.

- ❖ In einer Schleife werden alle Berichte durchlaufen. Jene Berichte, deren Name mit »Etiketten« beginnt, werden dem Funktionsergebnis hinzugefügt. Zuvor wird der Name aber jeweils um diesen Namensteil »Etiketten« gekürzt.

Damit ein neuer Bericht in das Druckmenü automatisch übernommen wird, muß sein Name also lediglich mit »Etiketten...« beginnen. Standardberichte – das sind jene, die angezeigt werden, wenn im Druckmenü *Normalpapier* ausgewählt ist – müssen mit »Druck...« beginnen. Sie werden über die Funktion `StandardBerichte()` eingelesen.

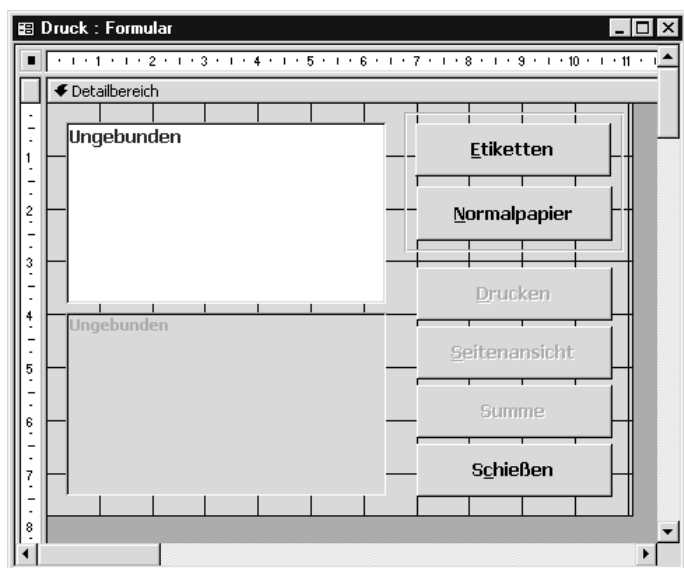


Bild 38.9: Druckmenü im Entwurf

Um eine gezielte Auswahl von Berichten zu realisieren, wird eine Optionsgruppe mit dem Namen *Typ* angelegt, die – wie bereits erwähnt – zwei Umschaltflächen enthält:

- ❖ Mit *Etiketten* werden die Berichte für den Etikettendruck ausgewählt. Diese Schaltfläche mit dem Optionswert 1 innerhalb der Optionsgruppe ist die Standardauswahl beim Öffnen des Druckmenüs.
- ❖ *Normalpapier* dient der Auswahl der Standardberichte. Diese Schaltfläche besitzt innerhalb der Optionsgruppe den Optionswert 2.

Nach Aktualisierung der Optionsgruppe wird eine Ereignisprozedur aufgerufen, die einerseits die Anzeige innerhalb des Listenfeldes aktualisiert und andererseits das Druckmenü wieder in eine Art »Grundstellung« bringt, indem bestimmte Schaltflächen deaktiviert werden. So werden z. B. die Schaltflächen *Drucken* und *Seitenansicht* erst aktiviert, nachdem ein Bericht im Listenfeld ausgewählt worden ist.



Bild 38.10: Optionsgruppe im Formularentwurf

```
Private Sub Typ_AfterUpdate()
    If Typ = 1 Then
        Me!Abfrage.RowSource = EtikettenBerichte()
    Else
        Me!Abfrage.RowSource = StandardBerichte()
    End If
    Me!Abfrage = Null
    Me!Drucken.Enabled = False
    Me!Seitenansicht.Enabled = False
    If Me!Summe.Enabled = True Then Me!Summe.Enabled =
False
    If Me!Regionen.Enabled = True Then
        Me!Regionen = Null
        Me!Regionen.Enabled = False
        Me!Regionen.ForeColor = 12632256
    End If
End Sub
```

Zur Erläuterung der vorstehenden Ereignisprozedur:

Die wichtigste Aufgabe dieser Ereignisprozedur ist es, je nach Auswahl innerhalb der Optionsgruppe die korrekten Berichte in das links von den Umschaltflächen befindliche Listefeld mit dem Namen *Abfrage* zu stellen. Dies geschieht wie beim Öffnen des Berichtes, indem der Eigenschaft *RowSource* (Datensatzherkunft) eine Textliste mit den Berichtsnamen

zugewiesen wird. Das kann dann mittels einer einfachen Auswahlabfrage (If...Then...Else-Auswahl) bewirkt werden:

- ...❖ Wenn in der Optionsgruppe *Typ* die Umschaltfläche mit dem Optionswert 1 angeklickt wird, sollen alle Etikettenberichte im aktuellen Formular dem Listenfeld mit dem Namen *Abfrage* zugewiesen werden.
- ...❖ Beim Anklicken der zweiten Umschaltfläche sollen alle Standardberichte dem Listenfeld zugeordnet werden.

Die Grundeinstellung des Formulars wird durch folgende Schritte wiederhergestellt:

1. Die Auswahl aus dem Listenfeld *Abfrage* wird gelöscht, d. h. auf Null gesetzt. Anweisung: `Me!Abfrage = Null`
2. Die Schaltflächen *Drucken* und *Seitenansicht* werden deaktiviert, damit sie erst gedrückt werden können, wenn ein Bericht ausgewählt ist. Anweisungen:

```
Me!Drucken.Enabled = False
Me!Seitenansicht.Enabled = False
```

3. Ist die Schaltfläche *Summe* aktiviert, wird sie ebenfalls deaktiviert. Anweisung:

```
If Me!Summe.Enabled = True Then Me!Summe.Enabled = False
```

4. Ist das Listenfeld zur Auswahl von Regionen aktiviert, wird es ebenfalls deaktiviert. Da ein deaktiviertes Listenfeld nicht die »Farbe verliert«, wird die Textfarbe auf Grau verändert. Anweisungen:

```
If Me!Regionen.Enabled = True Then
    Me!Regionen = Null
    Me!Regionen.Enabled = False
    Me!Regionen.ForeColor = 12632256
End If
```



Um die Farbe über VBA umzustellen, müssen Sie den internen Farbcode einer Farbe eingeben. Stellen Sie dazu temporär die gewünschte Zielfarbe für das Objekt ein, und lesen Sie den Farbcode aus dem Eigenschaftsfenster ab.

Nach Aktualisierung der Listenfeldes werden die Schaltflächen *Druck* und *Seitenansicht* aktiviert sowie gegebenenfalls das Zusatzlistenfeld *Regionen* aktiviert oder deaktiviert wird. Dazu dient die folgende Ereignisprozedur, die dem Listenfeld *Abfrage* für das Ereignis *Nach Aktualisierung* zugeordnet wird:

```

Private Sub Abfrage_AfterUpdate()
    If Me!Drucken.Enabled = False Then
        If Not IsNull(Me!Abfrage) Then
            Me!Drucken.Enabled = True
            If Not IsNull(Me!Abfrage) Then
                Me!Seitenansicht.Enabled = True
            ElseIf Me!Drucken.Enabled = True Then
                If IsNull(Me!Abfrage) Then Me!Drucken.Enabled =
False
                If IsNull(Me!Abfrage) Then
                    Me!Seitenansicht.Enabled = False
                End If
            If Me!Abfrage = "Regionenadressen" Then
                With Me
                    !Regionen.Enabled = True
                    !Regionen.ForeColor = 255
                    !Drucken.Enabled = False
                    !Seitenansicht.Enabled = False
                    !Summe.Enabled = True
                End With
            ElseIf Me!Regionen.Enabled = True Then
                With Me
                    !Regionen = Null
                    !Regionen.Enabled = False
                    !Regionen.ForeColor = 12632256
                    !Summe.Enabled = False
                End With
            End If
        End Sub

```

In der ersten Auswahlabfrage wird geprüft, ob Werte in dem Listenfeld *Abfrage* vorhanden sind. Wenn dieses nicht auf Null steht, werden die Schaltflächen *Drucken* und *Seitenansicht* aktiviert.

In dem *ElseIf*-Abschnitt werden demgegenüber die Schaltflächen *Drucken* und *Seitenansicht* deaktiviert, wenn nach Aktualisierung des Listenfeldes *Abfrage* in dem Feld keine Werte enthalten sind.

In der zweiten Auswahlabfrage wird geprüft, ob im Listenfeld *Abfrage* »Regionenadressen« ausgewählt ist. Ist dies der Fall, wird das Listenfeld *Regionen* eingeblendet, die Vordergrundfarbe verändert und die Schaltfläche *Summe* aktiviert.

In einem nächsten Schritt können Sie nun einstellen, daß ausgewählte Berichte zunächst in der Seitenansicht angezeigt werden. Dies ermöglicht die folgende Ereignisprozedur, die bei der Schaltfläche *Seitenansicht* dem Ereignis *Beim Klicken* zugeordnet wird:

```
Private Sub Seitenansicht_Click()  
Dim bername As String  
    If IsNull(Me!Abfrage) Then Exit Sub  
    Select Case Me!Typ  
        Case 1  
            bername = "Etiketten" & Me!Abfrage  
        Case 2  
            bername = "Druck" & Me!Abfrage  
    End Select  
    Me.Visible = False  
    DoCmd.OpenReport bername, acPreview  
  
End Sub
```

In der Prozedur wird auf ausgewählte Berichtsnamen zugegriffen. Er wird in der Variablen *bername* gespeichert, die als String deklariert ist.

In einem ersten Schritt wird zunächst abgefragt, ob im Listefeld *Abfrage* eine Auswahl getroffen ist. Ist dies nicht der Fall, wird die Prozedur verlassen.

Über eine *Case*-Struktur wird danach abgefragt, ob es sich um einen Etikettendruck oder um sonstige Druckausgaben handelt. Danach wird der Berichtsname aus der Auswahl im Listefeld *Abfrage* und dem Vorsatz »Etiketten« oder »Druck« zusammengesetzt. Sie erinnern sich, daß der betreffende Namensteil beim Einlesen der Berichtsnamen abgeschnitten worden ist, und nun wieder ergänzt werden muß, bevor der Bericht geöffnet werden kann.

```
Select Case Me!Typ  
    Case 1  
        bername = "Etiketten" & Me!Abfrage  
    Case 2  
        bername = "Druck" & Me!Abfrage  
End Select
```

Abschließend wird der Bericht mit *DoCmd.OpenReport* in der Seitenansicht (mittels Konstante *acPreview*) geöffnet.

```
DoCmd.OpenReport bername, acPreview
```

In ähnlicher Weise kann eine Ereignisprozedur für den direkten Ausdruck angelegt werden. Die folgende Ereignisprozedur ist der Schaltfläche *Drucken* beim Ereignis *Beim Klicken* zugeordnet:

```
Private Sub Drucken_Click()
Dim bername As String, text As String, weiter As Byte
    If IsNull(Me!Abfrage) Then Exit Sub
    Select Case Me!Typ
        Case 1
            bername = "Etiketten" & Me!Abfrage
            text = "Legen Sie bitte die Etiketten in den
Drucker ein und drücken Sie anschließend <OK>."
        Case 2
            bername = "Druck" & Me!Abfrage
            text = "Prüfen Sie, ob der Drucker einsatz-
bereit ist und drücken Sie anschließend <OK>."
    End Select
    weiter = MsgBox(text, vbYesNo + vbInformation,
"Druck")
    If weiter = vbYes Then
        DoCmd.OpenReport bername, acNormal
        MsgBox "Druckausgabe ist erfolgt", 64, "Druck"
    End If
End Sub
```

Die zuvor dargestellte Ereignisprozedur, die beim Klicken auf die Schaltfläche *Drucken* aufgerufen wird, prüft ebenfalls auf Etikett oder Normalpapier und setzt den entsprechenden Berichtsnamen zusammen. Nach einer Benutzerabfrage erfolgt der Ausdruck, wenn diese vom Benutzer mit »Ja« beantwortet worden ist.

## 38.4 Datensatz über Programmierung hinzufügen

Es wurde im Verlauf der Anwendungen bereits deutlich: Die Daten werden intern in den Tabellen der Datenbank abgelegt. Sinnvollerweise erfolgt die Erfassung und Pflege dieser Datensätze in einem Formular. Nun kann sich das folgende Problem ergeben: Wenn in einem Formular ein neuer Datensatz angelegt wird, kann es gewünscht sein, dazu korrespondierend in einer ganz anderen Tabelle ebenfalls einen Datensatz anzulegen. Durch reine Formulareingabe kann dies nicht immer erfolgen, denn dazu müßte die andere Tabelle in das Formular integriert sein.

Der zuvor allgemein beschriebene Anwendungswunsch läßt sich mit der vorliegenden Anwendung realisieren. Im Beispiel der Vereinsverwaltung muß



nämlich am Beginn eines jeden Vereinsjahres in der Tabelle *Jahresbeiträge* für jedes Mitglied ein Datensatz angelegt werden, welcher den Mitgliedsbeitrag in seiner Höhe und schließlich das Datum des Zahlungseinganges speichert.

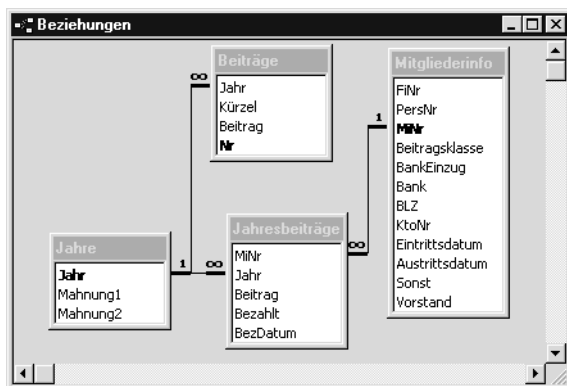


Bild 38.11: Modell der Beitragsverwaltung

Damit dieser Prozeß unterjährig nicht wiederholt gestartet werden muß – was natürlich möglich wäre –, soll sofort bei Neuanlage eines Mitgliedes dieser Datensatz in der Tabelle *Jahresbeiträge* generiert werden. Zusätzlich soll die Abfrage erfolgen, ob der Mitgliedsbeitrag bei Anmeldung bereits gezahlt worden ist. Der Zahlungseingang ist dann gegebenenfalls zu buchen.

Jahresbeiträge : Tabelle					
	MINr	Jahr	Beitrag	Bezahlt	BezDatum
▶	20000	1998	100,00	Nein	
	20000	1999	100,00	Nein	
	20001	1998	50,00	Ja	01.05.98
	20001	1999	50,00	Nein	
	20002	1998	69,00	Ja	01.05.98
	20002	1999	69,00	Nein	
	20003	1998	69,00	Ja	01.05.98
	20003	1999	69,00	Nein	
	20004	1998	110,00	Nein	
	20004	1999	110,00	Nein	
	20005	1998	110,00	Ja	01.05.98
	20005	1999	110,00	Nein	
	20006	1998	100,00	Ja	01.05.98
	20006	1999	100,00	Nein	
*				Nein	
Datensatz: 1 von 14					

Bild 38.12: Generierte Datensätze für Jahresbeitrag

Auch zur Lösung dieses Problems bietet sich der Weg über die VBA-Programmierung an. Im Beispielfall wurde eine Subprozedur mit dem Namen `NeuerBeitrag()` angelegt. Den Aufbau dieser Prozedur sollen Sie im folgenden genauer kennenlernen.

Ausgangspunkt der Anwendung ist die Prüfung, ob die Tabelle mit den Beiträgen für das gewünschte Jahr bereits Einträge enthält. Ist dies nicht der Fall, wird der Vorgang abgebrochen, da die Höhe des zuzuweisenden Betrages noch nicht feststeht. Dazu ist die Tabelle *Beiträge* zu öffnen und die Anzahl der Datensätze zu zählen. Dies wird mit folgenden Befehlen realisiert:

```
Private Sub NeuerBeitrag()
Dim db As Database, liste As Recordset
Dim Summe As Single, bez As Byte
    Set db = CurrentDb
    'Beitragshöhe feststellen
    Set liste = db.OpenRecordset("Beiträge",
dbOpenSnapshot)
    If liste.RecordCount < 1 Then
        MsgBox "Es sind noch keine Jahresbeiträge fest-
gelegt.", vbCritical, "Fehler"
        liste.Close
        db.Close
        Exit Sub
    Else
        liste.FindFirst "[Jahr] = '" & Year(Date) & "'
And [Kürzel] = '" & Me!Beitragsklasse & "'"
        If liste.NoMatch Then
            MsgBox "Es konnte der Beitragsgruppe kein Bei-
trag zugeordnet werden.", vbCritical, "Fehler"
            liste.Close
            db.Close
            Exit Sub
        Else
            Summe = liste!Beitrag
        End If
    End If
End Sub
```

Im nächsten Schritt wird in der Tabelle *Beiträge* nach dem entsprechenden Jahr und dem Eintrag für die bei der Anlage des Mitgliedes ausgewählte Beitragsklasse gesucht:

1. Dazu wird die Methode *MoveFirst* verwendet. Sie sucht jenen Datensatz, bei dem sowohl Kalenderjahr als auch Beitragsgruppe übereinstimmen.

2. Wird kein entsprechender Datensatz gefunden, ist entweder die Beitragsgruppe unzulässig, oder die Beitragshöhen für diese Gruppe und dieses Jahr sind noch nicht erfolgt. Dies kann dann vom Formular *Beitragswesen* aus nachgeholt werden.
3. Wird ein gültiger Jahresbeitrag gefunden, wird dieser in der Variablen *Summe* zwischengespeichert.

Da nun der fällige Mitgliedsbeitrag erhoben ist, kann der Datensatz an die Tabelle *Jahresbeiträge* angefügt werden. Dafür können Sie die Methode `AddNew` nutzen; hiermit wird an einer Tabelle ein neuer Datensatz angehängt.

```
Set liste = db.OpenRecordset("Jahresbeiträge",
dbOpenDynaset)
liste.AddNew
liste!MiNr = Me!MiNr
liste!Jahr = Trim(Str(Year(Date)))
liste!Beitrag = Summe
bez = MsgBox("Wurde der Mitgliedsbeitrag für das lau-
fende Jahr bei der Anmeldung bereits entrichtet?", 36,
"Beitrag")
If bez = vbYes Then
    liste!Bezahlte = True
    liste!BezahlteDatum = Date
End If
liste.Update
liste.Close
db.Close
End Sub
```

4. Im Feld *Mitgliedsnummer* wird jene aus dem entsprechenden Feld des Formulars eingetragen.
5. Dem Feld *Jahr* wird das aktuelle Jahr zugeordnet. Da das Feld *Jahr* in der Tabelle vom Typ *Text* definiert ist, muß das Ergebnis der Funktion `Year(Date)`, welche das aktuelle Jahr als Zahl zurückgibt, mit der Funktion `Str()` in einen Text umgewandelt werden. Access stellt allerdings bei der Umwandlung einer Zahl in einen Text mit der erwähnten Funktion immer ein Leerzeichen voran. Deshalb muß dieses mit der Funktion `Trim()` abgeschnitten werden.
- ❖ Dem Feld *Beitrag* wird die zuvor erhobene Höhe des Mitgliedsbeitrages zugeordnet.
- ❖ Nun wird vom Benutzer abgefragt, ob der Beitrag bei Anmeldung bereits entrichtet worden ist. Ist dies der Fall, wird der Beitrag als bezahlt gekennzeichnet sowie das aktuelle Datum als Datum des Zahlungseinganges eingetragen.
- ❖ Mit der Methode `Update` wird der neue Datensatz endgültig gespeichert.



Ohne diese Anweisung würden die Einträge verlorengehen.

... Die Variablen der Datenzugriffsobjekte werden geschlossen.

Die erläuterte Subprozedur wird von einer Ereignisprozedur aufgerufen, die beim Klicken auf die Schaltfläche *Übernehmen* ausgeführt wird.

Da diese sowohl bei einer Neuanlage als auch dann, wenn ein bestehender Datensatz nur editiert worden ist, zum Speichern der Eingaben verwendet wird, muß im Code für diese beiden Fälle eine genaue Unterscheidung getroffen werden.

Zu diesem Zweck wurde im Formular das verborgene Steuerelement *NeuMarkierung* angelegt. Bei Neuanlage eines Datensatzes wird in dessen Beschriftung »Neu« eingetragen. Dieser Eintrag wird ausgelesen und anschließend geändert. Der zugehörige Code:

```
If Me!NeuMarkierung.Caption = "Neu" Then
    Me!NeuMarkierung.Caption = "Alt"
    Set db = CurrentDb
    Set liste = db.OpenRecordset("Mitgliederinfo",
dbOpenTable)
    With liste
        .Index = "PrimaryKey"
        .Seek "=", Me!MiNr
        .Edit
        !PersNr = Me!PersNr
        .Update
        .Close
    End With
    db.Close
    NeuerBeitrag
End If
```



*Noch ein kurzer Hinweis zur Methode Seek: Bei der Neuanlage eines Datensatzes wird im Feld PersNr der Tabelle Mitgliederinfo nicht automatisch der betreffende Eintrag hinzugefügt. Dies liegt daran, daß das Formular Personenmitglieder auf einer Abfrage basiert, die auf den Tabellen Personen und Mitgliederinfo aufbaut. Das Feld PersNr aus der Tabelle Mitgliederinfo fehlt dabei und muß daher noch gefüllt werden. Bei Übernahme des Datensatzes wird in der Tabelle Mitgliederinfo mittels der Methode Seek die Mitgliedsnummer des neuen Datensatzes gesucht und so das Feld PersNr ergänzt.*

## 39 Literatur-/Bücherverwaltung

*Wer regelmäßig Fachliteratur und Fachzeitschriften liest, weiß, wie schwer es ist, nach ein paar Monaten eine bestimmte Textstelle wiederzufinden. Vielfach besteht der Wunsch, das vorhandene Material an Literatur (Büchern, Zeitschriftenaufsätzen, Prospekten) besser in den Griff zu bekommen. Auch hier kann eine Access-Lösung helfen.*

### 39.1 Anwendungsentwurf

Eine Literaturdatenbank dürfte bei der Suche nach der Stecknadel im Heuhaufen wertvolle Dienste leisten. Die typischen Anforderungen an eine solche Lösung sind:

- ...❖ Durch die Angabe der Kenndaten wie Autor oder Titel sowie durch Eingabe von Stichworten sollen die archivierten Bücher oder Zeitschriften schnell gefunden werden.
- ...❖ Das gezielte Suchen nach Dokumenten über verschiedene Schlagwörter steht im Mittelpunkt des Retrieval-Prozesses. Dazu müssen der Deskriptor oder die Deskriptoren am Computerbildschirm eingegeben werden.
- ...❖ Bei einer Volltextrecherche kann nach jedem Wort gesucht werden, das innerhalb der Dokumente auftaucht. Im anderen Fall müssen die den Dokumenten zugeordneten Suchbegriffe eingegeben werden.
- ...❖ Nach Auslösen des Recherchevorganges werden die »Trefferanzahl« sowie die Volltexte bzw. die bibliographischen Angaben zu den gefundenen Dokumenten am Bildschirm angezeigt (Titel, Autor, Standort). Im Bedarfsfall ist auch ein Ausdruck der Suchergebnisse bzw. bei Volltextspeicherung aller oder bestimmter Dokumente möglich.
- ...❖ Integrierend soll auch die Ausleihe mitverwaltet werden. Damit ist jederzeit ein Überblick gegeben, in wessen Besitz sich gerade ein bestimmtes Buch oder eine Zeitschrift befindet.



Die folgende Lösung finden Sie auf Ihrer CD-ROM unter dem Namen LITERATURVERWALTUNG.MDB gespeichert. Grundlage sind die folgenden sieben Tabellen.

<b>Tabelle</b>	<b>Zweck</b>
<i>tblBücher</i>	Die Daten zu den Büchern, die zu verwalten sind, werden in dieser Tabelle organisiert.
<i>tblThemen</i>	Die Tabelle dient der Spezifizierung der Themen, zu denen die Bücher archiviert werden. Mit dieser Nebentabelle wird einerseits die Bedienung der Datenbank erleichtert, andererseits sorgt sie dafür, daß für gleiche Themenbereiche immer die gleichen Schlagwörter benutzt werden.
<i>tblAutoren</i>	Die Stammdaten der Autoren werden hiermit verwaltet. Darüber hinaus soll eine systematische Erfassung und Pflege erfolgen.
<i>tblBuchautoren</i>	Da ein Autor mehrere Bücher verfassen kann, umgekehrt ein Buch auch von mehreren Autoren verfaßt werden kann, wird die Tabelle <i>Buchautoren</i> eingefügt.
<i>tblZitate</i>	Zitate, die für ein Buch charakteristisch sind, sollen einer besonderen Tabelle zugeordnet werden.
<i>tblAusleihe</i>	Um immer einen Überblick zu haben, wer welches Buch wann ausborgt und wieder zurückgebracht hat, werden alle Ausleihvorgänge in dieser Tabelle vermerkt.
<i>Switchboard Items</i>	Diese Tabelle wird vom System automatisch angelegt, wenn mit dem Übersichts-Manager eine Menüführung für die Datenbank erstellt wird. Diese Tabelle enthält die Übersichtseinträge, die zur Steuerung der Menüs/Übersichten benötigt werden.

Als Hintergrundinformation ist eine Analyse der zwischen den Tabellen bestehenden Beziehungen interessant. Einen Überblick gibt die folgende Abbildung:

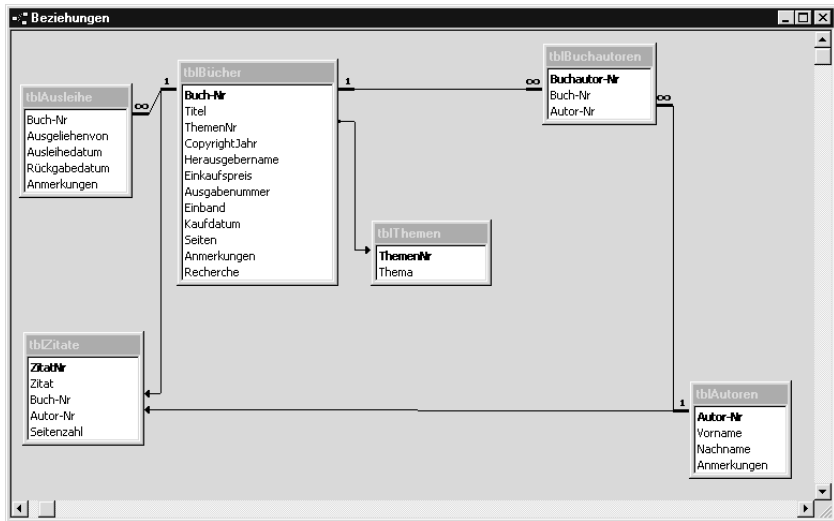


Bild 39.1: Beziehungen der Anwendung Literaturverwaltung

Die Abbildung zeigt:

- ❖ Kern der Datenbank ist die Tabelle *tblBücher*. Ein Buch kann von mehreren Buchautoren verfaßt werden. Deshalb besteht eine 1:n-Beziehung zwischen den Tabellen *tblBücher* und *tblBuchautoren*.
- ❖ Ein Autor kann mehrfach als Buchautor vorkommen. Deshalb besteht eine 1:n-Beziehung zwischen den Tabellen *tblAutoren* und *tblBuchautoren*.
- ❖ Die Tabelle *tblThemen* ist eine Nebentabelle zur Tabelle *tblBücher*. Die Verknüpfung wird über das Feld *ThemenNr* hergestellt.
- ❖ Einem Autor können mehrere Zitate zugeordnet werden. Eine Verknüpfung erfolgt von der Tabelle *tblAutoren* über das Feld *Autor-Nr* zur Tabelle *tblZitate*. Ebenso besteht eine 1:n-Beziehung zwischen den Tabellen *tblBücher* und *tblZitate*. Die Tabelle *tblZitate* fungiert als eine Zwischentabelle zur Auflösung der m:n-Beziehung zwischen *tblAutoren* und *tblBücher*, ebenso wie die Tabelle *tblBuchautoren*.
- ❖ Die Tabelle *tblAusleihe* bezieht sich auf die Bücher. Da ein Buch öfter ausgelohnt werden kann, besteht zwischen den Tabellen *tblBücher* und *tblAusleihe* eine 1:n-Beziehung.

Damit die Anwendung komfortabel bedient werden kann, stehen für alle Aufgaben Formulare zur Verfügung. Diese seien in der nachfolgenden Übersicht kurz vorgestellt. Außerdem wurde mit dem Übersichts-Manager – so wie in Kapitel 21 beschrieben – eine Menüführung für die gesamte Anwendung erstellt.

Beim Öffnen der Anwendung wird automatisch das Formular *frmÜbersicht* geöffnet. Dies wird über die Startoptionen bewerkstelligt.

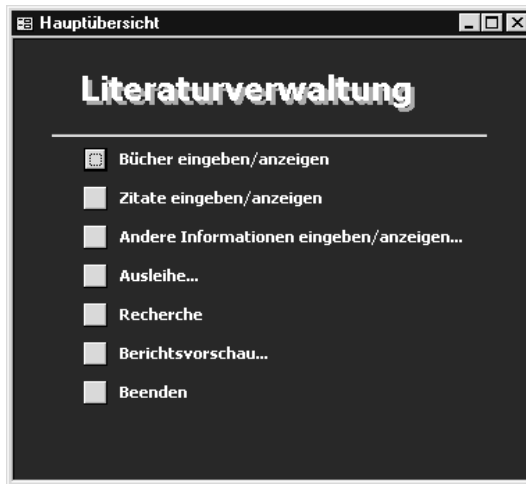


Bild 39.2: Hauptübersicht

Die Hauptübersicht gliedert die Datenbank in folgende Bereiche:

- ...❖ Eingeben neuer bzw. Anzeige erfaßter Bücher
- ...❖ Eingabe von Zitaten
- ...❖ Anzeige ausgewählter Informationen
- ...❖ Aufruf einer Recherchefunktion (Volltextrecherche)
- ...❖ Berichtsvorschau

Der Erfassung der Bücher liegt zunächst das folgende Formular zugrunde:

Bild 39.3: Formular frmBücher



Damit im Formular *frmBücher* das Listenfeld *Thema* gefüllt wird, müssen Themen im Formular *frmThemen* erfasst werden.

Bild 39.4: Formular *frmThemen*

Da Bücher nichts ohne ihre Autoren sind, müssen auch diese erfasst und gewartet werden. Dies geschieht über das Formular *frmAutoren*.

Bild 39.5: Formular *frmAutoren*

Um immer zu wissen, wer sich welches Buch gerade ausborgt hat bzw. bei wem man ein vermisstes Buch wiederfinden kann, gibt es die beiden Formulare *frmNeueAusleihe* zur Erfassung der Ausleihen sowie *frmZurZeitAusgeliehen*, um eine Übersicht der »im Umlauf befindlichen« Bücher zu erhalten.

Buch	ausgeliehen von	von	bis	Anmerkungen
Diamanten	Herbert Eifrig	03.04.99		
Kosten senken mit dem Einkauf	Hans Lustig	05.05.99		
So planen Sie Ihre Karriere	Gerhard Müller			
Diamanten	Gerhard Funkel			

Ausleihe	
Buch-Nr:	Information als Führungsmit
Ausgeliehen von	Helmut Maier
Ausleihedatum	15.05.99
Rückgabedatum	13.06.99
Anmerkungen	Er benötigt es für ..

Bild 39.6: Formulare *frmZurZeitAusgeliehen* und *frmNeueAusleihe*

Um die gespeicherten Informationen auch zu Papier bringen zu können, können von der Übersicht *Berichtsvorschau...* aus verschiedene Berichte aufgerufen werden – beispielsweise der Bericht *Titel nach Autoren*.

Autorennamen	Titel	Herausgebername	Einkaufspreis	K
Becker, Hetz-Joachim	Controller und Controlling	expert - Taylorix	49,00 DM	
	So planen Sie Ihre Karriere	Schmitt & Jungmann	22,95 DM	
Callahan, Laura	Techniken des Tai Chi	Becker und Severin	25,95 DM	
	Dirk Luchte	GGG&G Publishing	23,50 DM	
Fuller, Andrew	So planen Sie Ihre Karriere	Schmitt & Jungmann	22,95 DM	
Grundwald, Herbert				

Bild 39.7: Bericht »Titel nach Autoren«

## 39.2 Besondere VBA-Lösungen in der Anwendung

### 39.2.1 Automatische Datensatzsuche im Formular

Access stellt seit der Version 7.0 über den Steuerelementassistenten für Kombinationsfelder eine Möglichkeit zur Verfügung, auf Formularebene nach der Auswahl aus einem Listenfeld direkt zum entsprechenden Datensatz zu gelangen.

Bild 39.8: Kombinationsfeld zur Datensatzsuche

Während eine solche Anwendung in früheren Versionen nur mit Tricks oder Programmierung bewerkstelligt werden konnte, erstellt jetzt der Assistent den entsprechenden Programmcode für Sie automatisch.

Im Formular *frmBücher* wird ein solches Kombinationsfeld eingefügt, um schnell ein bestimmtes Buch auswählen zu können. Damit sich das Feld von jenen etwas abhebt, die der Dateneingabe dienen, wird es im Formularkopf platziert.

Zur Lösung wird der Steuerelement-Assistent wie gewohnt gestartet. Im ersten Dialog ist folgende Option zu wählen:

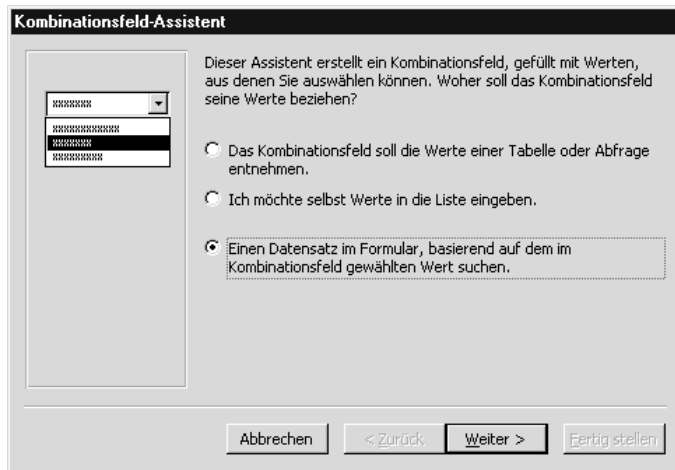


Bild 39.9: Option zur Datensatzsuche

Im weiteren Verlauf unterscheidet sich die Erstellung des Kombinationsfeldes nicht von einem gewöhnlichen. Werden die Felder des Formulars angezeigt, wählen Sie jenes aus, nach dem Sie suchen möchten. Den Rest erledigt der Assistent für Sie. Testen Sie die Lösung: Wählen Sie nun ein Buch aus der Liste aus – voilà, schon sind Sie beim entsprechenden Datensatz.

Was ist intern geschehen? Der Assistent fügt automatisch das jeweilige Schlüsselfeld als verborgene gebundene Spalte hinzu. Dieses stellt das wirkliche Suchkriterium dar. Verwenden Sie in Kombinationsfeldern nach Möglichkeit Felder, die Ihnen eindeutige Werte als Suchkriterium liefern. Beispiel: Im Formular *frmBücher* wurde das Feld *Titel* gewählt, und nicht ein Feld wie beispielsweise *Herausgeber*. Denn bei mehrmaligem Vorkommen identischer Werte finden Sie diese auch mehrmals untereinander in der Auswahlliste. Sie müssen dann unter Umständen mehrmals »probieren«, um zu Ihrem Buch zu gelangen. Sie können sich aber in der Auswahlliste auch mehrere Spalten anzeigen lassen, um sofort zu einer eindeutigen Auswahl zu kommen (beispielsweise *Herausgeber* und *Titel*).



*Der Steuerelement-Assistent stellt die beschriebene Möglichkeit nur zur Verfügung, wenn das Formular auf einer Tabelle basiert, nicht jedoch, wenn eine Abfrage als Basis für das Formular dient. Man kann sich folgendermaßen behelfen: Ändern Sie kurzfristig die Datenherkunft des Formulars in der Weise, daß das Formular auf der Haupttabelle Ihrer Abfrage basiert, und starten Sie den Assistenten. Anschließend machen Sie die Änderung wieder rückgängig. Der Suchvorgang funktioniert dann wunschgemäß.*

Intern legt der Assistent folgenden VBA-Code zur Datensatzsuche an:

```
Sub Buchsuche_AfterUpdate()  
    Me.RecordsetClone.FindFirst "[Buch-Nr] = " &  
    Me![Buchsuche]  
    Me.Bookmark = Me.RecordsetClone.Bookmark  
End Sub
```

Zur Erläuterung der Anweisungen:

- ❖ Mit `RecordsetClone` verweisen Sie auf ein Recordset, welchem genau die im aktiven Formular angezeigten Datensätze zugrundeliegen. Sie können damit auf diese Daten im Recordset zugreifen, diese jedoch nicht verändern.
- ❖ Die Methode `FindFirst` wird verwendet, um genau jenen Datensatz zu suchen, der im Kombinationsfeld ausgewählt worden ist. Im Beispielfall wird im Feld *Buch-Nr* nach der im Kombinationsfeld *Buchsuche* selektierten Nummer gesucht. Da im Listefeld genau jene Datensätze angezeigt werden, welche auch dem Formular zugrundeliegen, kann eine Überprüfung unterbleiben, ob die Suche erfolgreich gewesen ist. Sie muß es aus obiger Prämisse heraus immer sein.
- ❖ Jeder Datensatz eines gebundenen Formulars erhält beim Öffnen ein eindeutiges Lesezeichen (*Bookmark*). Der entsprechende Datensatz wird angezeigt, indem das *Bookmark* des im `RecordsetClone` gefundenen Datensatzes mit dem *Bookmark* des Formulars synchronisiert wird.

## 39.2.2 Volltextsuche – eine nützliche Hilfe für das

### Auffinden von Literatur

Die vorliegende Anwendung verfügt als ein besonderes Leistungsmerkmal über eine Volltextsuche. Die Volltextsuche ist ein recht hilfreiches Feature, das in Literaturdatenbanken zur Verfügung gestellt werden kann. Der Suchbegriff wird in diesem Fall quer über den gesamten Datenbestand gesucht, das Suchergebnis in einer Liste ausgegeben.

Bei der Lösung dieses Problems ist die bisher erläuterte einfache Suchen-Funktion überfordert. Access stellt die Möglichkeit der Volltextsuche nicht direkt zur Verfügung, einen Teil der in der Praxis benötigten Funktionalität können Sie aber ab der Version 7.0 mit dem formularbasierten Filter abdecken. Er verbirgt sich hinter dem Symbol mit dem Trichter und dem Formular in der Symbolleiste. Zur Bedienung verwendet er einfach die Felder und das Layout Ihres erstellten Formulars. Aber auch dies bringt nicht in allen Punkten ein befriedigendes Ergebnis.

Wir haben in der Anwendung Literaturverwaltung eine Art der Volltextsuche implementiert, die den Anforderungen einer solchen noch besser gerecht wird. Zur Recherche über den erfaßten Literaturbestand dient das folgende Formular:

Bild 39.10: Formular für Volltextsuche

Die Anwendung funktioniert so:

- ❖ Nach Eingabe des Suchbegriffes wird der Suchvorgang gestartet. Es werden all jene Datensätze im Formular angezeigt, in denen der Suchbegriff an beliebiger Stelle vorkommt.
- ❖ Zum leichteren Auffinden des Suchergebnisses werden jene Felder, in denen die Suche erfolgreich gewesen ist, durch rote Textfarbe hervorgehoben. Dies ist vor allem dann hilfreich, wenn das Suchergebnis aus mehreren Datensätzen besteht. Navigiert man zwischen diesen Datensätzen, wird die Aufmerksamkeit sofort auf den gesuchten Begriff gelenkt.

### Formular zur Volltextsuche erstellen:

Um das obige Formular zu erstellen, ist zunächst ein ungebundenes Textfeld zur Eingabe des Suchbegriffes anzulegen:

Bild 39.11: Hauptformular für Recherche

Das Textfeld erhält den Namen *Suchbegriff*, mit dem später auf dieses verwiesen werden kann. Zusätzlich wird eine Schaltfläche benötigt, um die Suche zu starten.

### Unterformular für das Suchergebnis erstellen:

Für die Anzeige der Suchergebnisse soll ein Unterformular erzeugt werden. Als Basis dafür kann das Formular *frmBücher* verwendet werden. Dieses wird kopiert und unter dem Namen *frmRechercheUF* gespeichert. Es muß in folgender Weise lediglich leicht modifiziert werden:

- ❖ Das Unterformular zur Anzeige der Autoren wird durch ein direktes Autorenfeld ersetzt, damit auch dieses in die Volltextsuche einbezogen werden kann.
- ❖ Die enthaltenen Ereignisprozeduren werden nicht benötigt, da sie auf ein Formular zur Dateneingabe abzielen. Sie können gelöscht werden. Öffnen Sie dazu die Ansicht *Code*, und entfernen Sie alle Prozeduren.
- ❖ Da das Formular nur zur Anzeige des Suchergebnisses und nicht zur Datenerfassung oder -wartung bestimmt ist, werden die Eigenschaften
  - ❖ *Bearbeitungen zulassen*

- ...❖ Löschen zulassen
  - ...❖ Anfügen zulassen und
  - ...❖ Daten eingeben
- jeweils auf *Nein* gestellt.

Hier das gewünschte Ergebnis:

Bild 39.12: Entwurf des Recherche-Unterformulars

Abfrage für das Formular *frmRechercheUF* erstellen:

Das zentrale Element der Volltextsuche ist eine Abfrage, die dem Unterformular zugrundeliegt. Für jedes der für die Volltextsuche in Frage kommenden Felder (das können alle Felder des Dynasets sein) wird mit der Funktion `InStr()` untersucht, ob der Suchtext innerhalb des Textes vorkommt.

`InStr([Start, ]Zeichfolge1, Zeichfolge2[, Vergleich])`

Kommt der Suchtext innerhalb des Feldes vor, liefert die Funktion die Position, anderenfalls gibt sie die Zahl 0 zurück. Nun werden alle Datensätze durch die Abfrage ausgewählt, bei denen die Funktion `InStr()` für zumindest eines der Datenfelder einen Wert zurückgibt, der größer als 0 ist.

Für die Abfrage werden folgende Tabellen benötigt:

- ...❖ *tblBücher* ist die Haupttabelle der Abfrage, da nach Büchern gesucht werden soll.

- ...❖ *tblThemen* wird benötigt, um auch im Text des Feldes *Thema* zu suchen, da in der Tabelle *tblBücher* lediglich die *ThemenNr* enthalten ist.
- ...❖ *tblBuchautoren* und *tblAutoren* werden benötigt, damit Namen der Autoren und Anmerkungen zu ihrer Person ebenfalls in die Suche einbezogen werden.

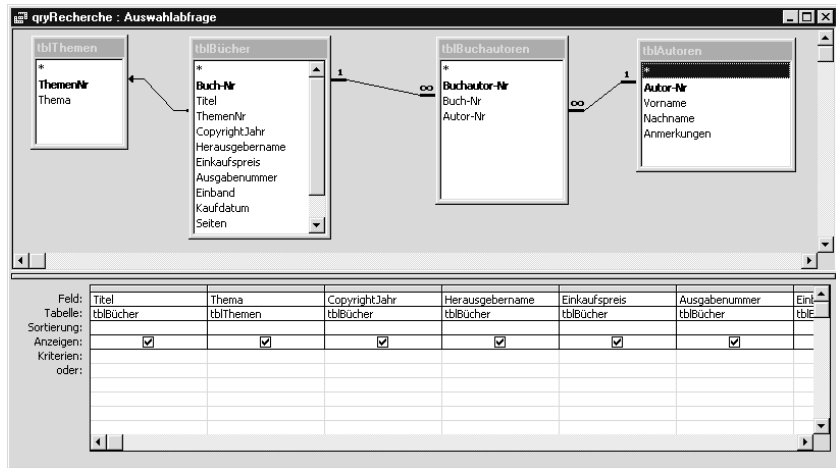


Bild 39.13: Basis für die Recherche-Abfrage

Nachdem alle Felder für die Anzeige im Formular in den QBE-Bereich übernommen worden sind, werden sie nun alle ein zweites Mal in Kombination mit der Funktion `InStr()` ausgewählt:

`InStr({Name des Feldes},{Suchbegriff im Hauptformular})`

Mit dem Ausdruck `[Formulare]![Recherche]![Suchbegriff]` wird auf den Inhalt des zuvor angelegten ungebundenen Textfeldes verwiesen. Kommt dieser Text im Feld vor, so ist die Bedingung »größer 0«, die für jeden Ausdruck definiert wird, wahr.

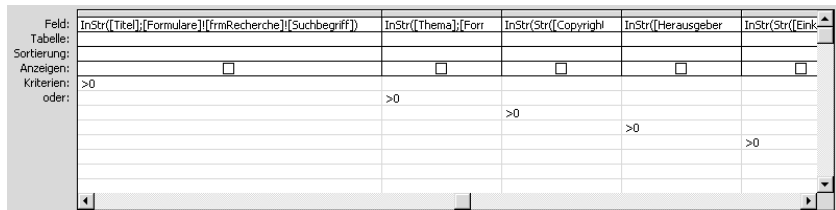


Bild 39.14: Abfragefelder mit Bedingungen für die Recherche

Diese Ausdrücke dienen nur der Auswahl der Datensätze durch die Abfrage und werden nicht als Teil des Abfrageergebnisses benötigt. Daher wird bei ihnen das Kästchen *Anzeigen* deaktiviert.



### Starten des Suchvorganges:

Beim Öffnen des Formulars *Recherche* werden noch keine Datensätze angezeigt. Nach der Eingabe eines Suchbegriffes wird das Unterformular nicht automatisch neu berechnet. Dies könnte durch Drücken der Taste [F9] bewerkstelligt werden. Damit dies für den Benutzer übersichtlicher gelöst ist, haben wir in das Formular eine Schaltfläche mit der Aufschrift *Suche Starten...* integriert. Diese kann beispielsweise mit folgender einfacher Ereignisprozedur hinterlegt werden:

```
Private Sub Suche_Click()  
    Me.Refresh  
    me!Suchbegriff.SetFocus  
End Sub
```

Mit der Anweisung `Me.Refresh` wird eine Aktualisierung der Formularanzeige erzwungen, was in diesem Fall dem Druck auf die Taste [F9] gleichkommt. Anschließend wird der Fokus wieder auf das Feld *Suchbegriff* gelegt, damit sofort ein neuer Suchbegriff eingegeben werden könnte.



*Anstelle der Anweisung `Me.Refresh` kann auch `SendKeys "{F9}"` verwendet werden, um direkt den Tastendruck zu simulieren.*

*Wenn Sie nur die Anzeige im Unterformular aktualisieren möchten, um z. B. den Fokus des aktuellen Datensatzes im Hauptformular nicht zu verlieren, können Sie einen »Refresh« im Unterformular vornehmen, indem Sie der Eigenschaft *Datenherkunft* (`RecordSource`) des Unterformulars einen neuen Wert zuweisen. Dadurch wird eine Neuberechnung des Dynasets erzwungen. Um einen Refresh durchzuführen, weisen Sie der Eigenschaft *Datenherkunft* einfach sich selber zu.*

```
Me!MeinUnterformular.Form.RecordSource =  
Me!MeinUnterformular.Form.RecordSource
```

.....

### Suchergebnis hervorheben:

Um das Suchergebnis noch anschaulicher zu machen, werden die Felder des Suchergebnisses, in denen der Suchbegriff enthalten ist, durch rote Textfarbe hervorgehoben. Dabei muß berücksichtigt werden, daß die Anzeige bei jedem Datensatz aktualisiert wird, wenn das Suchergebnis aus mehr als einem Datensatz besteht. Es bietet sich für die Umsetzung eine Ereignisprozedur für das Ereignis *Beim Anzeigen* des Formulars *frmRechercheUF* an. Dieses Ereignis tritt ein, wenn zwischen den Datensätzen gewechselt wird.

Für jedes Datenfeld muß folgender Code durchlaufen werden, der wiederum die Funktion `InStr()` verwendet. Dabei steht 255 für den Farbcode Rot und 0 für Schwarz.

```
If InStr({Name des Feldes}, {Suchbegriff}) > 0 Then
    {Name des Feldes}.ForeColor = 255
Else
    {Name des Feldes}.ForeColor = 0
End If
```

Die einfachste Variante wäre es, diesen Code hintereinander zu kopieren und jeweils den Namen eines Datenfeldes einzutragen. Die elegantere Methode ist jedoch, für diesen wiederkehrenden Codeteil eine Subprozedur anzulegen, die jeweils mit einem anderen Feldnamen als Übergabeparameter aufgerufen wird:

```
Private Sub Form_Current()
    Farbanzeige ("Titel")
    Farbanzeige ("Thema")
    Farbanzeige ("Herausgebername")
    Farbanzeige ("Einband")
    Farbanzeige ("Einkaufspreis")
    Farbanzeige ("BücherAnmerkungen")
    Farbanzeige ("Autor")
    Farbanzeige ("AutorenAnmerkungen")
    Farbanzeige ("Ausgabenummer")
    Farbanzeige ("CopyrightJahr")
    Farbanzeige ("Kaufdatum")
    Farbanzeige ("Seiten")
End Sub
```

Rufen Sie die Subprozedur `Farbanzeige()` für jedes Feld auf, das Sie in die Volltextsuche integriert haben.

In der Subprozedur `Farbanzeige()` wird einer Objektvariablen vom Typ `Control` das Steuerelement mit dem beim Funktionsaufruf übergebenen Namen zugeordnet. Diese Variable wird nun verwendet, um auf das jeweilige Steuerelement zu verweisen.

Auf den Suchbegriff im Hauptformular kann anstatt mit der vollen Syntax `Forms!Recherche!Suchbegriff` mit der abgekürzten Form `Parent!Suchbegriff` Bezug genommen werden. Dies ist vergleichbar mit der Verwendung von `Me`, um auf das aktuelle Formular zu verweisen.



Verwenden Sie die *Parent* (Hauptobjekt)-Eigenschaft, um von einem Unterformular auf das Hauptformular zu verweisen. Sie können das Hauptformular dann problemlos umbenennen, alle Verweise funktionieren weiterhin.

```
Private Sub Farbanzeige(FELDNAME As String)
Dim FELD As Control
    Set FELD = Me(FELDNAME)
    If InStr(FELD, Parent!Suchbegriff) > 0 Then
        FELD.ForeColor = 255
    Else
        FELD.ForeColor = 0
    End If
End Sub
```

Zum einfacheren und übersichtlicheren Verweis auf ein Steuerelement wird eine entsprechende Variable vom Typ `Control` deklariert. Dieser Variablen wird jeweils das entsprechende Steuerelement, dessen Name beim Prozeduraufruf mitübergaben worden ist, zugewiesen.

Nach der Überprüfung, ob der Suchbegriff aus dem Hauptformular im Feld enthalten ist, erfolgt die Einstellung der *Textfarbe* (`ForeColor`). Kommt der Text vor, wird der Text in Rot, sonst in Schwarz angezeigt.

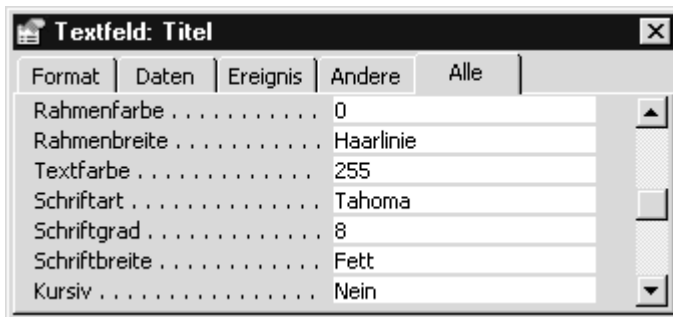


Bild 39.15: Textfarbe einstellen



Den Zahlenwert für eine Farbe erhalten Sie, indem Sie ein Steuerelement in der jeweiligen Farbe einfärben und anschließend den Wert im Eigenschaftsfenster der Zeile *Textfarbe* entnehmen.

Anschließend binden Sie das Formular *frmRecherche* über den Übersichtsmanager in die Hauptübersicht ein.