

Entwicklung  
einer  
Datenbank-Applikation  
zur Verwaltung von  
Tagesstreckenkarten

Informatikprojekt

am Institut für Informatik  
und angewandte Mathematik  
der Universität Bern

von Eveline Kurt

WS 98

## Zusammenfassung

Die SBB verkauft den Bundesbetrieben Tagesstreckenkarten zu einem bestimmten Preis. Diese Karten dienen als Billette, ähnlich wie die Tageskarten, welche man zu einem Pauschalpreis kaufen kann, wenn man ein Halbtax - Abo besitzt. Mit diesen Karten kann man an einem Tag soweit fahren wie man will.

Die Eidg. Alkoholverwaltung beschäftigt viele Aussendienstmitarbeiter, welche mit dem Zug reisen, und wie alle Bundesangestellte benutzen sie Tagesstreckenkarten als Billette. Der Gebrauch der Tagesstreckenkarten (TSK) muss kontrolliert werden, da die SBB erstens wissen will, welche Karten wann gebraucht wurden und zweitens die gebrauchten Karten zurück will. Das erfordert einen gewissen administrativen Aufwand der Verwaltung, welche zu diesem Zweck seit etwa zehn Jahren eine von der eigenen EDV- Abteilung entwickelte Datenbank-Applikation (SDB-Datenbank) benutzt. Nun ist es an der Zeit, diese Applikation durch eine neue zu ersetzen., denn diese Applikation ist einerseits zu langsam, andererseits funktioniert die Berechnung der neuen Kartenummer nicht mehr richtig. Die Entwicklung dieser neuen Applikation ist das Projekt, welches hier dokumentiert wird.

# Inhaltsverzeichnis

1 Situationsanalyse.....	4
2 Anforderungen an die Datenbank .....	5
2.1 Allgemeine Anforderungen.....	5
2.1.1 Allgemein.....	5
2.1.2 Geschwindigkeit.....	5
2.1.3 Grösse der Datenbank.....	5
2.1.4 Grafische Benutzeroberfläche .....	5
2.1.5 Umgebung.....	5
2.1.6 Entwicklungswerkzeug der Datenbank .....	5
2.1.7 Installation der Applikation .....	5
2.1.8 Migration und Test.....	6
2.1.9 Datenübernahme .....	6
2.1.10 Datensicherung.....	6
2.1.11 Sicherheit.....	6
2.1.12 Dokumentation.....	7
2.1.13 Erweiterbarkeit.....	7
2.2 Anforderungen an Funktionen und Informationen .....	7
2.2.1 Anforderungen an Informationen .....	7
2.2.2 Anforderungen an Funktionen .....	7
3 Grobdesign .....	10
3.1 Aufbau der Datenbank-Applikation.....	10
3.2 Aufbau der Programm-Funktionen.....	10
3.2.1 .....	10
3.2.2 Die Bearbeitungsebene kann man grob in drei Aufgabebereiche aufteilen: .....	10
3.2.3 Grafische Darstellung des Aufbaus der Programm-Funktionen .....	11
3.3 Struktur der grafischen Oberfläche .....	13
3.3.1 Funktionen und Formulare .....	13
3.3.2 Formulararten .....	13
3.3.3 Gemeinsamkeiten aller Formulare .....	13
3.3.4 Beispiele von Formularen .....	14
3.4 Ablaufszenario 'Karten zurücknehmen'.....	16
4 Feindesign .....	17
4.1 Beziehungen zwischen den Daten .....	17
4.1.1 Entity-Relationship-Modell .....	17
4.2 Tabellen .....	18
5 Implementation .....	19
5.1 Kleine Übersicht über Access .....	20
5.1.1 Definition einer Tabelle .....	20
5.1.2 Formulare .....	20
5.1.3 Elemente .....	21
6 Erfahrungen.....	21
6.1.1 Kenntnisse am Anfang des Projekts .....	21
6.1.2 Allgemeine Erfahrungen .....	21
6.1.3 Erfahrungen mit dem Projekt .....	21
6.1.4 Sonstige Erfahrungen .....	22

# 1 Situationsanalyse

Von der Kanzlei der Eidg. Alkoholverwaltung werden etwa 1 mal jährlich 1000 Tagessteckenkarten bei der SBB bestellt. Diese Karten sind fortlaufend nummeriert. (Jetzt etwa bei Nummer 466`763)

Jedoch fehlen Nummern zwischen den einzelnen Bestellungen, da die SBB noch an andere Bundesämter Tagesstreckenkarten TSK verkaufen.

Eine oder mehrere Karten zusammen werden an einzelne Mitarbeiter abgegeben. Die Mitarbeiter tragen bei der Bahnreise das Datum des Reisetags auf die Karte ein. Nachher bringen sie eine oder mehrere Karten an die Kanzlei zurück.

Wenn sie eine Karte ausgefüllt haben, und die Reise nicht antreten konnten, ist die Karte unbrauchbar. Eine Karte kann auch verloren oder vernichtet werden. Diese Karten müssen der SBB bezahlt werden.

Die verlorenen Karten werden mit KartenNr., Name der Person und Code 9 in die SDB-Datenbank eingegeben. Wahrscheinlich wird dadurch veranlasst, dass die Karte nicht zu der Anzahl benutzter Karten dazugezählt wird.

Wenn die Karte korrekt ausgefüllt war, der Zug aber nicht fahren konnte, wird die Karte annulliert. Diese Karten müssen der SBB nicht bezahlt werden.

Ungebrauchte Karten, die natürlich noch von jemand anderem benutzt werden können, werden der Kanzlei zurückgegeben. Dazu sucht man im PC die KartenNr. und ändert den Namen auf den Namen des neuen Kartenbesitzers um (Mutation) .

Die Karten werden laufend abgegeben und zurückgenommen. Nach jedem Quartal(3 Monate) wird irgendwann mal abgerechnet.

## **Mit der SDB-Datenbank:**

Es wird zusammengezählt, wieviele Karten zurückgekommen sind. Anzahl zurückgekommene Karten mal Kartenpreis wird berechnet.

Der Preis vom Quartal, in dem die Karte zurückgenommen wurde, ist massgebend.

Es wird manuell zusammengezählt, ob die Anzahl vorhandener, zurückgekommener Karten mit der Anzahl, die von der SDB-Datenbank angegeben wird, übereinstimmt.

Es wird ein Formular der SBB ausgefüllt indem steht :

- - Anzahl benutzter (zurückgekommener) Karten
- - Anzahl verlorener Karten
- - Anzahl annullierter Karten
- - noch vorhandene, ungebrauchte Karten (Vorrat)

Mit jeder Abrechnung wird ein Buchungsbeleg mithilfe der SDB-Datenbank erstellt und an die Buchhaltung abgeben. Darauf steht, welches Konto (und damit welche Abteilung) wieviel Geld für die Tagesstreckenkarten ausgegeben hat (Anzahl Karten mal Preis pro Karte). Die Ausgaben der Abteilungen werden zusammengezählt und als Quartalstotal ausgegeben.

Es kommt etwa alle 5 Jahre vor, dass alle ungebrauchten Karten eingezogen werden müssen. Denn steht eine Preisänderung der SBB bevor, werden alle, die noch im Besitz von Karten sind, gebeten, die benutzten Karten noch vor der Preisänderung zurückzubringen. Der TSK-Preis ändert meistens am 1.1. des neuen Jahres.

## **2 Anforderungen an die Datenbank**

### **2.1 Allgemeine Anforderungen**

#### **2.1.1 Allgemein**

Die neue Datenbank soll die optimalen Funktionen der abzulösenden Datenbank beibehalten aber die Schwachstellen sollen natürlich nicht übernommen werden.

#### **2.1.2 Geschwindigkeit**

Die aktuelle Datenbank wird immer langsamer, da es nicht möglich ist, alte Datensätze auszulagern. Die neue Lösung sollte schneller sein, gegebenenfalls durch Auslagerung der Daten. Dies wäre kein Problem, das alte Daten eigentlich nicht mehr benötigt werden.

#### **2.1.3 Grösse der Datenbank**

Die Datenbank wird einige MegaBytes gross werden. Sollte die Grösse zum Problem werden, kann die Datenbank auf einen vorhandenen Server ausgelagert werden.

Pro Jahr werden etwa 1000 Karten vergeben. Die Anzahl der Angestellten beträgt etwa 230.

#### **2.1.4 Grafische Benutzeroberfläche**

In der Kanzlei arbeiten auch Leute, die es sich nicht gewohnt sind, mit EDV-Geräten umzugehen. Die Bedienung des Programms sollte also möglichst einfach sein. Die Standardoberfläche von Access hat viele Symbolleisten und sonstige Befehlsschaltflächen, die nicht immer gebraucht werden, das Programm aber kompliziert erscheinen lassen. Deshalb sollte jeweils nur ein Fenster sichtbar sein, mit möglichst wenig Befehlsschaltflächen.

##### **2.1.4.1 Bezeichnungen:**

Das neue Programm stellt eigentlich die gleichen Funktionen wie das alte zur Verfügung, deshalb sollen überall, wo es möglich ist, dieselben Funktionsbeschreibungen beibehalten werden. Bsp: Buchhaltungsdetail bleibt Buchhaltungsdetail. Das sollte die Umstellung auf die neue Anwendung erleichtern.

#### **2.1.5 Umgebung**

Die EDV-Arbeitsplätze der Kanzlei sind am Intranet der Eidg. Alkoholverwaltung angeschlossen. Als Betriebssystem ist Windows for Workgroups installiert..

#### **2.1.6 Entwicklungswerkzeug der Datenbank**

Das Entwicklungswerkzeug ist das Microsoftprodukt Access 2.0, eine 'relationale Datenbank für Windows'. Dieses Entwicklungswerkzeug ist vorgegeben. Es muss noch abgeklärt werden, ob ein Access Runtime-Modul zur Verfügung steht. ( Mit dem Runtime-Modul kann eine Applikation direkt als solche gestartet werden).

### **2.1.7 Installation der Applikation**

Die Applikation muss so installiert werden, dass sie als von Access unabhängige Applikation läuft. (Dazu ist das Access Runtime- Modul nötig). Die Applikation soll an drei verschiedenen Arbeitsplätzen installiert werden.

### **2.1.8 Migration und Test**

Ein paar Monate lang werden die 'alte' sowie die 'neue' Datenbank parallel geführt, um die neue Datenbank zu testen.

Bei der Inbetriebnahme der neuen Datenbank-Applikation wird der Rest der Serie der in der Kanzlei vorhandenen Karten als Kartenserie in die neue Datenbank eingegeben. Dadurch können sofort nach der Installation der neuen Datenbank die auszugebenden Karten in beiden Applikationen als ausgegeben vermerkt werden.

Die Daten der noch nicht zurückgekommenen Karten werden bei der Installation der neuen Datenbank in diese übernommen werden. Die zurückkommenden Karten werden dann in beiden Systemen als zurückgekommen eingetragen.

So können die SDB-Datenbank und die neue Datenbank-Applikation ein paar Monate lang parallel geführt werden. (siehe Datenübernahme).

Ist man sicher, dass das neue Programm zufriedenstellend funktioniert, kann man die alte durch die neue Datenbank ablösen.

### **2.1.9 Datenübernahme**

Die in der SDB-Datenbank vorhandenen Daten über die ausgeteilten, aber noch nicht wieder zurückgezogenen Tagesstreckenkarten werden in die neue Datenbank übernommen.

#### **2.1.9.1 Personendaten**

Die Daten Name, Vorname, Personnummer werden von der Telefon-Datenbank übernommen, also nicht von der SDB-Datenbank. Die der Person zugehörigen Kontonummern hingegen werden von der abzulösenden Datenbank übernommen. Grund: Die Personnummern der Telefon-Datenbank werden noch in anderen Applikationen gebraucht, z. B. der Telefon-Datenbank, eine kleine Applikation zur Verwaltung von Telefonnummern, implementiert in Access 2.0. Anmerkung: diese Personnummer ist nicht identisch mit der 'offiziellen' Personnummer, welche leider nicht elektronisch zur Verfügung steht.

#### **2.1.9.2 Konten**

Da nur etwa 30 verschiedene Konten bestehen, werden KontoNr, Kürzel der Abteilung und voller Abteilungsname von den Anwendern der Applikation selber eingegeben werden.

### **2.1.10 Datensicherung**

Die Aufbewahrungsdauer der Daten wird durch den Fachdienst bestimmt. Ohne Angaben werden sie während 10 Jahren aufbewahrt.

### **2.1.11 Sicherheit**

#### **2.1.11.1 Verlust der Daten**

Von allen Daten der Abteilung, in welcher die Datenbank installiert ist, erstellt die Systemadministration automatisch jede Woche auf einem Server eine Sicherheitskopie .

#### **2.1.11.2 Fremder Zugriff auf die Daten**

Da die Daten keine vertraulichen Informationen enthalten, muss die Applikation nicht besonders geschützt werden. Der Zugriff auf die Windows for Workgroups-Oberflächen ist durch ein Passwort geschützt.

### **2.1.12 Dokumentation**

Die Dokumentation besteht hauptsächlich aus einem Benutzerhandbuch und einem Systemhandbuch.

Von der Datenbank selber lässt sich jederzeit mit Access eine Dokumentation erstellen. Dazu muss man die Datenbank in Access öffnen. Der Datenbank-Dokumentierer von Access befindet sich unter Add-Ins im Menü Datei. Damit lässt sich für jedes Formular jede Einstellung ausdrucken, inklusive einem Listing des Programmcodes.

Die Handbücher sind gespeichert auf einer Diskette und ausgedruckt auf Papier vorhanden.

### **2.1.13 Erweiterbarkeit**

Die Datenbank wird später mit der Telefon-Datenbank verknüpft werden. Deshalb sollten die Personendaten mit dieser Datenbank übereinstimmen.

Dazu gibt es noch weitere mögliche Erweiterungen.

Eine Möglichkeit wäre:

Da alle Karten manchmal zurückgezogen werden müssen, könnte man in Word für Windows einen Serienbrief machen, der die Anzahl offener Karten aus der Datenbank automatisch einfügt. Der Brief könnte automatisch mit dem vorhandenen Mailtool verschickt werden. Es reicht aber, sich dann damit zu befassen, wenn es gebraucht wird, da dieser Fall höchstens alle paar Jahre vorkommt.

Vielleicht macht die SBB ihre TSK Abrechnungsformulare einmal elektronisch und das auszufüllende Formular wird elektronisch geschickt. Dann könnte man das Formular mit den Daten aus der Datenbank automatisch ausfüllen.

## **2.2 Anforderungen an Funktionen und Informationen**

### **2.2.1 Anforderungen an Informationen**

#### **2.2.1.1 Personen**

Über die Personen müssen Namen und Vornamen und KontoNr der Abteilung, für die sie arbeiten, bekannt sein.

#### **2.2.1.2 Konten**

KontoNr, Kürzel der Abteilung und der volle Abteilungsname sind erforderlich.

#### **2.2.1.3 Karten**

Über die Kartenserie muss man das Eingangsdatum, die erste sowie die letzte Nummer wissen.

Über die einzelne Karte muss folgendes bekannt sein:

Nummer, Ausgabedatum, Benutzungsdatum, Rückgabedatum, Nummer der zuletzt ausgegebenen Karte.

#### **2.2.1.4 Preis**

Datum der Preisänderung und neuer Preis

### **2.2.2 Anforderungen an Funktionen**

Die Anforderungen kann man grob in drei Bereiche aufteilen:

- Anforderungen an die Bearbeitung der Tagesstreckenkarten
- Anforderungen an die Aufbereitung der Daten (Ausdrucken der Daten (Berichte))
- Anforderungen an die Verwaltung der Daten (Mutationen, archivieren)

### **2.2.2.1 Anforderungen an die Bearbeitung der Tagesstreckenkarten**

#### eine oder mehrere TSK einer Person zuweisen

Die KartenNr muss automatisch fortlaufend erscheinen und man kann die Person, welcher die Karten zugewiesen werden, auswählen.

Da meistens mehrere Karten zusammen vergeben werden, soll der Name der Person nur einmal gewählt werden müssen.

#### Karten zurücknehmen

Das Rückgabedatum wird automatisch gespeichert, ev. ausser bei den verlorenen Karten.

#### Gebrauchte Karten zurücknehmen

Das Datum des Gebrauchs der Karte wird manuell eingegeben, das Rückgabedatum wird automatisch gespeichert.

#### Karte als annulliert speichern.

Eine Karte gilt als annulliert, wenn die Karte zwar beschriftet wurde, aber nicht gebraucht werden konnte, aber durch Verschulden der SBB. Zum Beispiel ist das der Fall, wenn ein Zug nicht fahren kann.

#### Verlorene Karten als solche kennzeichnen

Die verlorenen Karten werden erst bemerkt, wenn alle Karten eingezogen werden. Die Karten sind in Gebrauch. Es können Karten als in Gebrauch durchgehen, die schon lange verloren sind. Obwohl Karten selten verloren gehen, müssen die verlorenen Karten in der Abrechnung berücksichtigt werden.

Karten werden irgendwann gebraucht und erst ein paar Quartale später abgegeben. Das ist kein Problem, da der Preis davon abhängig ist, in welchem Quartal die Karte zurückkommt.

#### TSK ungebraucht jemand anderem zuweisen(mutieren)

Ungebrauchte Karten zurücknehmen, um sie jemand anderem zuzuweisen.

Es wäre gut, wenn die Karten als zurückgekommen gespeichert werden könnten, da eine Karte nicht sofort an jemand anderes zurückgegeben wird. Sonst erscheint die Karte noch in der Liste der unbenutzten Karten der Person.

Dass die Karte weitergegeben wird, ist dann Sache der Kanzlei.

Anforderungen an die Aufbereitung der Daten (Ausdrucken der Daten (Berichte))

### **2.2.2.2 Anforderungen an die Aufbereitung der Daten (Ausdrucken der Daten (Berichte))**

#### Buchhaltungsbeleg erstellen



Bericht für die Buchhaltung(alte Bezeichnung: STK: Buchhaltungsbeleg)

Darauf steht der Name der Abteilung, Kto.Nr der Abteilung, Kosten pro Abteilung, Total Kosten.

Zählt alle TSK zusammen, die im Quartal zurückgebracht wurden (R\_Datum im Quartal), ein B\_Datum haben oder/und verloren gegangen sind.  
Für den Preis ist das R\_Datum massgebend.  
Die SBB kann ihre Preise auf den Anfang jedes Monats ändern

#### Buchhaltungsdetail pro Quartal

Kann angeben, wer in einer bestimmten Abteilung wieviele Karten gebraucht hat (Im letztem Quartal oder im laufenden Quartal)  
mit dem Bericht 'Buchhaltungsdetail pro Quartal';  
Darauf steht :  
die Konto Nr, KartenNr,Datum-Benutzt,Name,Preis  
Total Preis pro Abteilung, total gebrauchte Karten pro Quartal

#### Quartalstatistik

Gibt alle Kartennummern an, die im Quartal zurückgekommen sind.  
Gibt das Total der Benutzten Karten an und das Total in Franken.  
Es ist eine Liste in Reihenfolge der Nummern, mit Namen der Personen

#### Liste aller Offenen

Es ist eine Liste der ungebrauchten Karten nach KontoNr und Person. Zum wissen, wer welche noch ungebrauchten Tagesstreckenkarten noch bei sich hat.

#### Liste nach Namen

Mit der Möglichkeit einen Namen einzugeben, so dass alle ungebrauchten Karten der Person angezeigt werden (alte Bezeichnung: TSK: Liste nach Namen).  
Wird vor allem für Nachfragen von einzelnen Personen gebraucht.

Anforderungen an die Verwaltung der Daten (Mutationen, archivieren)

#### **2.2.2.3 Anforderungen an die Verwaltung der Daten (Mutationen, archivieren)**

##### Mutationen

Mutationen von Personen( alte Bezeichnung: Personal)  
Mutationen von Abteilungen (eine Abteilung entsteht neu, eine Abteilung ändert die Kontonr.)  
Mutationen vom Kartenpreis(Eingabe Datum, neuer Preis)  
Eingabe von neu erhaltenen TSK(Eingabe Datum, erste Nr., letzte Nr.)

##### Karten von SBB erhalten

Die Anfangsnummer und Endnummer der von der SBB erhaltenen Kartenserie muss eingegeben werden. Es wird angenommen, dass alle Nummern vergeben wurden, bevor die übernächste Kartenserie kommt.

archivieren

alte Daten werden ausgelagert.

## 3 Grobdesign

### 3.1 Aufbau der Datenbank-Applikation

Die Applikation besteht aus zwei Teilen, aus Access-Sicht eigentlich aus zwei selbstständigen Programmen. Damit die Daten von den Funktionen auch physisch im Netz getrennt werden können. Um das zu erreichen bietet sich eine Trennung der Applikation in eine Datenbank-Applikation an, welche nur die Funktionen enthält und eine, welche nur die Daten enthält. Die erste greift auf die zweite zu.

### 3.2 Aufbau der Programm-Funktionen

Startet die Benutzerin das Programm, erscheint sofort ein Menü, von dem aus sie alle gewünschten Funktionen aussuchen kann. Sobald sie mit der gewünschten Bearbeitung fertig ist, kommt sie zurück zum Menü. Das Programm besteht hauptsächlich aus zwei Ebenen, die Ebene des Menüs und die Ebene der Bearbeitungen.

3.2.1

**3.2.2 Die Bearbeitungsebene kann man grob in drei Aufgabengebiete aufteilen:**

- Bearbeitung der Tagesstreckenkarten
- Ausdrucken der Daten (Berichte)
- Verwalten der Daten ( z.B. Mutation von Personen, archivieren)

**Die einzelnen Bereiche lassen sich aufteilen in:**

*Bearbeitung der TSK*

- TSK einer Person zuweisen
- Gebrauchte Karten zurücknehmen
- Karte als annulliert speichern
- Karte als verloren speichern
- ungebrauchte Karten zurücknehmen
- ungebrauchte Karten weitergeben

### *Ausdrucken der Daten*

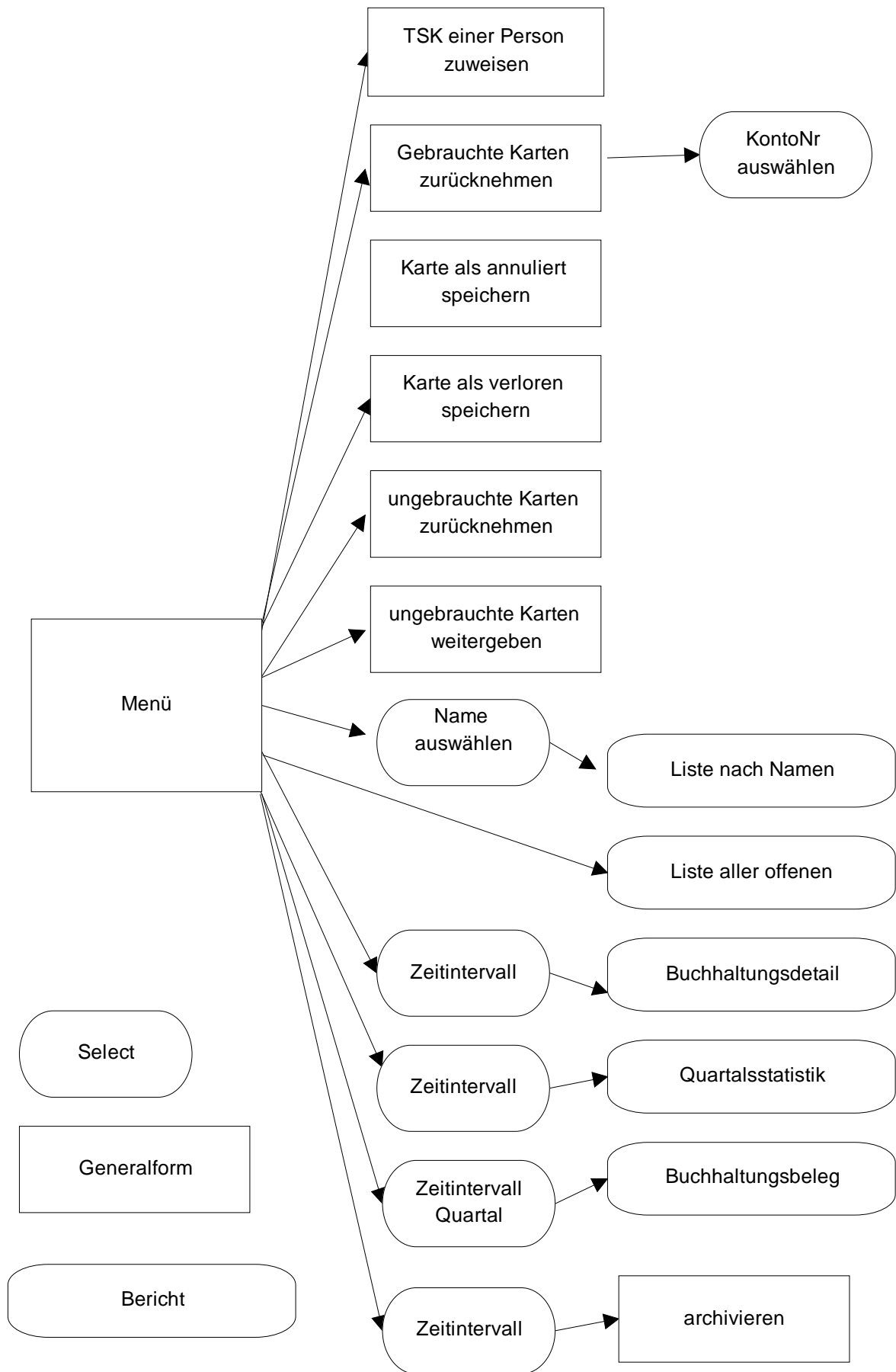
- Liste nach Namen
- Liste aller offenen
- Buchhaltungsdetail
- Quartalsstatistik
- Buchhaltungsbeleg

### *Verwalten der Daten*

- Uebersicht Karten
- neue Karten erfassen
- Uebersicht Konten
- neue Konten erfassen oder mutieren
- Uebersicht Personen
- neue Person erfassen oder mutieren
- archivieren

was unter den einzelnen Punkten genau passiert, steht in den Anforderungen der Datenbank oder in der Benutzungsanleitung, sofern es sich nicht aus den Bezeichnungen erahnen lässt.

### **3.2.3 Grafische Darstellung des Aufbaus der Programm-Funktionen**



## **3.3 Struktur der grafischen Oberfläche**

### **3.3.1 Funktionen und Formulare**

ede Programm-Funktion wird mit einem Formular oder Bericht erstellt. Ich habe mich bemüht, Formulare mit einer ähnlichen Funktion möglichst ählich zu gestalten. Dabei habe ich mich möglichst an die Guidelines gehalten, welche von den 'hauseigenen' Entwicklern angewendet werden. Das führte zu folgenden Formulararten mit den Bezeichnungen Generalform (Hauptformular), Overview (Übersichtsformular), Select (Auswahlformular), Mainform (Menü).

### **3.3.2 Formulararten**

- Generalforms dienen zum speichern und verändern von Datensätzen.
- Overviews dienen zur Uebersicht, der Inhalt sollte nicht verändert werden können.
- Selectforms zeigen eine Auswahl an, aus der ein Element ausgewählt werden muss. Die Selectform kann nur verlassen werden, nachdem etwas ausgewählt wurde. Zum verlassen der Form steht nur eine Befehlsschaltfläche, die Befehlsschaltfläche Schliessen zur Verfügung.

### **3.3.3 Gemeinsamkeiten aller Formulare**

Die Formulare können nur über Befehlsschaltflächen geschlossen werden. Nach dem Schliessen des Formulars befindet man sich wieder auf dem Formular, aus dem es geöffnet wurde.

### 3.3.4 Beispiele von Formularen

#### 3.3.4.1 Hauptformulare

*Bspl einer Generalform:*

The image shows a software window titled "Gebrauchte TSK zurücknehmen". The window contains a form with the following elements:

- TSKNr:** A text input field with a green grid pattern and a small dropdown arrow on the right.
- benutzt am:** A text input field.
- Ausgabe:** A text input field.
- KontoNr:** A text input field.
- Name:** A text input field.
- Vorname:** A text input field.
- Name ändern:** A text input field with a small dropdown arrow on the right.
- speichern:** A button located to the right of the "Ausgabe" and "KontoNr" fields.
- nächste Nr.:** A button located to the right of the "Name" and "Vorname" fields.
- abrechnen:** A button located at the bottom left of the form.
- schliessen:** A button located at the bottom right of the form.

Generalform

Allen Hauptformularen ist gemeinsam, dass sie keine Bildlaufleisten haben.

### 3.3.4.2 Übersichtsformulare

Beispiel eines Overview-Formulars:

PersonenNr	Name	Vorname
1	Aebi	Urs
2	Aebischer	Joseph
3	Aegerter	Roland
4	Altwegg	Werner
193	Amrein	Heinrich
5	Ansermet	Félix

Overview-Formular

Dieses Übersichtsformular ist ein Access Standardformular, in dem die Datensätze nur gelesen werden können. Es besitzt zwei Access Standardbefehlsschaltflächen und eine Befehlsschaltfläche schliessen. Die Standardbefehlsschaltfläche links erlaubt, die Datensätze nach Kriterien zu sortieren, die rechts, die Datensätze direkt nach dem Nachnamen zu sortieren. Die Navigationsschaltflächen erlauben, zwischen den Datensätzen zu 'navigieren'. Die Bildlaufleisten ermöglichen, in andere Bereiche des Formulars zu gelangen.

Anderen Übersichtsformularen fehlt die Funktionalität, die Datensätze nach Kriterien zu sortieren, wenn das Formular die Einträge einer Tabelle oder Abfrage mit weniger als dreissig Datensätzen darstellt.

### 3.3.4.3 Auswahlformulare

Beispiel eines Select-Formulars:

Der Person ein Konto zuweisen

KontoNr auswählen

310.00
310.01
310.02
310.03
310.04
310.05
310.07
310.09
310.10

Abbrechen OK

Das wichtigste Element eines Auswahlformulars ist ein Listenfeld, aus dem ein Eintrag ausgewählt werden kann. Ist die Auswahl fakultativ, kann man das Formular mit abbrechen verlassen. Ist die Auswahl zwingend, so hat das Formular nur die Befehlsschaltfläche OK um die Wahl zu bestätigen und das Formular zu verlassen.

### 3.4 Ablaufszenario 'Karten zurücknehmen'

Frau Meier hat ihre TSK benutzt und bringt sie nun zur Ausgabestelle zurück. Der Sachbearbeiter wählt `Gebrauchte TSK zurücknehmen` aus dem Menü aus, worauf das Formular `Gebrauchte TSK zurücknehmen` geöffnet wird. (Siehe Darstellung auf Seite 14)

Er klickt das Kombinationsfeld `TSKNr` an, worauf eine Liste möglicher TSK-Nummern erscheint. Er wählt nun eine davon aus. Alternativ kann er die Nummer direkt eingeben. Daraufhin werden `Ausgabe-Datum`, `Name` und `Vorname` angezeigt. Nun muss er das `Benutzungs-Datum` der Karte eingeben. Bei jeder anderen Aktion erscheint ein Meldungsfenster, das zur Eingabe des Datums auffordert. Nun wird die Befehlsschaltfläche `speichern` oder `schliessen` betätigt, je nachdem, ob noch weitere Karten zurückzunehmen sind. Hat Frau Meier mehrere Konten, erscheint ein Auswahl-Formular, in dem ein Konto ausgewählt werden muss. Die gewählte Kontonummer erscheint im Textfeld `KontoNr`. Die Befehlsschaltfläche `nächste Nr.` fügt die nächste gültige `KartenNr.` ins Kombinationsfeld ein. `nachste Nr.` ist für fortlaufende Kartennummern gedacht und soll die Eingabe der nächsten `KartenNr.` vereinfachen. Für jedem Speichervorgang muss man explizit die `KartenNr.` wählen, das `Benutzungs-Datum` eingeben und `speichern`. `Schliessen` speichert die TSK-Nr und schliesst das Formular.

Mit `Name ändern` kann man die TSK einer anderen Person zuweisen, bevor man sie zurücknimmt. Das wird gebraucht, falls die Karte vor dem Benutzen weitergegeben wurde.

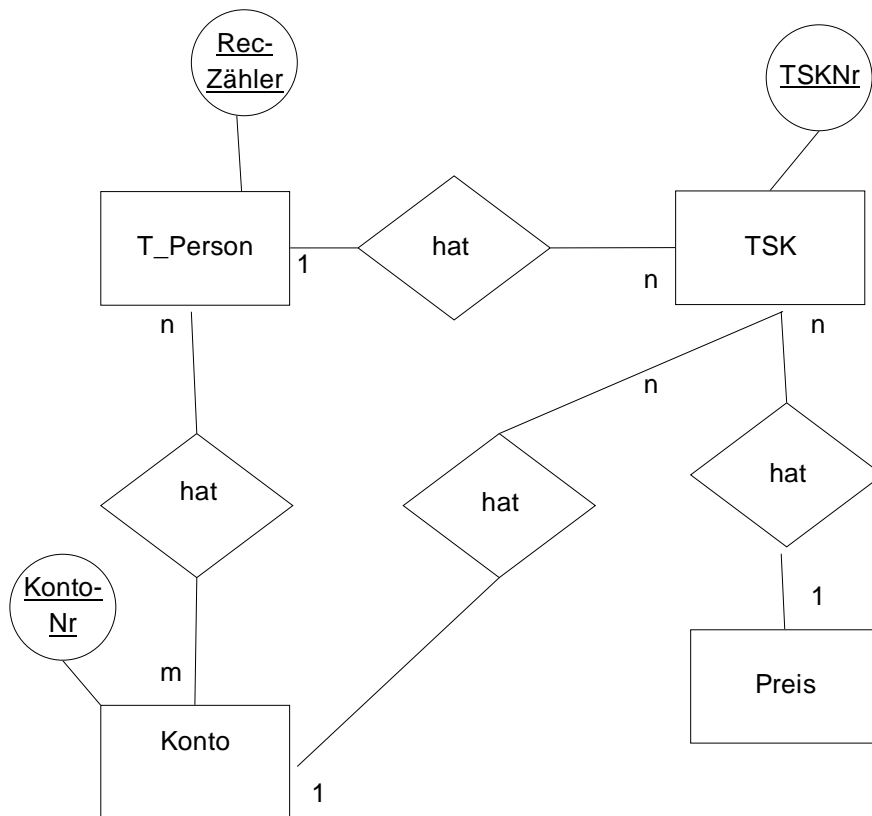
`Abbrechen` erlaubt es, das Formular zu schliessen, ohne dass etwas gespeichert wurde.



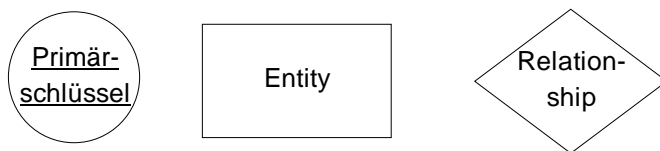
# 4 Feindesign

## 4.1 Beziehungen zwischen den Daten

### 4.1.1 Entity-Relationship-Modell



Legende:



## 4.2 Tabellen

### TSK

<b>Feldname</b>	<b>Feldtyp</b>	<b>Beschreibung</b>
TSKNr	Zahl(Long)	Primärschlüssel
A_Datum	Datum/Zeit	AusgabeDatum der TSK
B_Datum	Datum/Zeit	Benutzungsdatum der TSK
Annuliert	Ja/Nein	
Verloren	Ja/Nein	
Rec-Zähler	Zahl(Long)	Fremdschlüssel
R_Datum	Datum/Zeit	RückgabeDatum der TSK
ungebraucht	Ja/Nein	
KontoNr	Text	
Preis	Zahl(Single)	

### Konto

<b>Feldname</b>	<b>Feldtyp</b>	<b>Beschreibung</b>
KontoNr	Text	Primärschlüssel
KontoBez	Text	
KontoName	Text	

### PersonKonto

<b>Feldname</b>	<b>Feldtyp</b>	<b>Beschreibung</b>
Rec-Zähler	Zahl(Long)	Fremdschlüssel
KontoNr	Text	Fremdschlüssel

### T\_Person

<b>Feldname</b>	<b>Feldtyp</b>	<b>Beschreibung</b>
Rec-Zähler	Zahl(Long)	Primärschlüssel
T-Name	Text	
T-Vorname	Text	

### TSKvonSBB

<b>Feldname</b>	<b>Feldtyp</b>	<b>Beschreibung</b>
E_Datum	Datum/Zeit	Eingangsdatum der TSK
vonNr	Zahl(Long)	erste Nummer der Serie
bisNr	Zahl(Long)	letzte Nummer der Serie
letzteNr	Zahl(Long)	letzte abgegebene Nummer

### TSKPreis

<b>Feldname</b>	<b>Feldtyp</b>	<b>Beschreibung</b>
ab_Datum	Datum/Zeit	ab wann der neue Preis gilt
Preis	Zahl(Single)	

### Archivtabelle TSK

<b>Feldname</b>	<b>Feldtyp</b>	<b>Beschreibung</b>
TSKNr	Zahl(Long)	
A_Datum	Datum/Zeit	AusgabeDatum der TSK
B_Datum	Datum/Zeit	BenutzungDatum der Tsk
Annuliert	Ja/Nein	
Verloren	Ja/Nein	
Rec-Zähler	Zahl(Long)	
R_Datum	Datum/Zeit	RückgabeDatum der TSK
ungebraucht	Ja/Nein	
Dummy1	Text	
Dummy	Text	
KontoNr	Text	

## 5 Implementation

Durch das Erstellen von Formularen und Schreiben von Prozeduren werden das Grob- und Feindesign in eine Applikation umgesetzt. Damit man sich besser vorstellen kann, wie das geht, folgt nun eine kurze Einführung über Access.

### 5.1 Kleine Übersicht über Access

Access unterstützt die Herstellung von Tabellen, Beziehungen zwischen den Tabellen, Abfragen (Queries), Formularen (Grafische Oberflächen) und Berichten.

#### 5.1.1 Definition einer Tabelle

*Beispiel einer Tabellen-Vorlage:*



Feldname	Felddatentyp	Beschreibung

Tabellen-Vorlage

In dieser Vorlage können die gewünschten Felder definiert werden.

#### 5.1.2 Formulare

Formulare dienen als Schnittstelle zwischen den Benutzern des Programms und den Daten in den Tabellen. Sie werden auf der Grundlage von Tabellen oder Abfragen erstellt, oder aber es sind 'ungebundene Formulare'.

Hier ist ein Formular, welches auf der Basis der Tabelle Konto erstellt wurde, welche im vorherigen Kapitel vorgestellt wurde. Der Eintrag im Textfeld mit der Beschriftung 'KontoNr' ist eine Kontonummer aus der Tabelle 'Konto'.

*Beispiel eines Formulars:*



Formular

Dieses Formular besteht im Wesentlichen aus den Bereichen Formularkopf und Detail. Ferner aus einem Textfeld und einem Bezeichnungsfeld.

**In ein Formular lassen sich folgende Steuerelemente einbauen:**

- Befehlsschaltflächen
- Kombinationsfelder
- Listenfelder
- Unterformulare
- Optionsfelder
- Kontrollkästchen

### **5.1.3 Elemente**

#### **5.1.3.1 Eigenschaften**

Elemente, sowie Formulare und Berichte haben Eigenschaften, welche eingestellt werden können. Eigenschaften des Bezeichnungsfeldes 'KontoNr' sind zum Beispiel: Schriftart, Schriftgrösse, Textausrichtung, Rahmenart, Breite, Höhe, Sichtbar, etc.

#### **5.1.3.2 Standardereignisse**

Sie haben Standardereignisse. Zum Beispiel ist 'Beim Klicken' das Standardereignis für ein Bezeichnungsfeld. Weitere Ereignisse dieses Feldes sind: Beim Doppelklicken, Bei Maustaste auf/ ab, Bei Mausbewegung.

Jedes Element hat eine ganze Sammlung von Ereignissen. Ein Ereignis kann mit einer Prozedur oder einem einfachen Befehl verknüpft werden. Löst die Benutzerin des Programms ein Ereignis aus, indem sie eine Befehlsschaltfläche anklickt, startet eine Prozedur, welche z.B das Formular schliesst, oder einen Datensatz speichert. Ereignisse führen durch das Programm, in Access-Büchern spricht man von ereignisorientierter Programmierung.

## **6 Erfahrungen**

### **6.1.1 Kenntnisse am Anfang des Projekts**

Access hatte ich im Selbststudium gelernt, aber noch nie so richtig angewendet.

Das Praktikum in Software Engineering hatte ich noch nicht besucht, hingegen die Vorlesung über Datenbanken.

### **6.1.2 Allgemeine Erfahrungen**

Ich hatte einen Arbeitsplatz in der EDV-Abteilung. Obwohl ich allein an meiner Datenbank-Applikation arbeitete und meine Kollegen Access nicht gut kannten, habe ich von ihrer Erfahrung profitieren können. Nebenbei habe ich mitbekommen wie sie an einem grösseren Projekt arbeiteten, was ganz interessant war.

Die Arbeit am Projekt habe ich als willkommene Abwechslung zum Studium empfunden.

### **6.1.3 Erfahrungen mit dem Projekt**

Alles in allem habe ich gute Erfahrungen gemacht, die Datenbank-Applikation wurde installiert und wird benutzt. Schlechte Erfahrungen in dem Sinn hatte ich keine, aber ich habe einige für Informatik-Projekte typische Erfahrungen gemacht. Hier ein paar Beispiele:

#### **6.1.3.1 Anforderungsbeschreibung**

Ich habe eine möglichst genaue Zusammenfassung der Anforderungen gemacht und der Benutzerin zum Lesen gegeben, aber trotzdem ist es uns passiert, dass wir uns in einem wichtigen Punkt missverstanden haben. Ich hatte angenommen, dass ein Aussendienstmitarbeiter für eine Abteilung arbeitet, und somit seine TSK immer der gleichen Abteilung verrechnet wird, aber es stellte sich später heraus, dass das nicht immer so ist.

Was ich auch nicht wusste war, dass die Aussendienstmitarbeiter die Karten manchmal untereinander austauschen. (Das wurde bei der Rückgabe der Karte wichtig)

#### **6.1.3.2 Mitarbeit am Projekt**

Die Benutzerin meiner Datenbank-Applikation hat sich immer Zeit genommen, mit mir zu sprechen und meine Fragen zu beantworten, aber sie hatte eigentlich keine Zeit sich wirklich mit der Entwicklung und Einführung der neuen Datenbank-Applikation zu beschäftigen, was sich dann vor allem beim Testen zeigte. Gewisse Fehlermeldungen erhielt ich erst ein Jahr später. Es waren kleine Sachen, die schnell behoben waren, aber eine Computerfirma hätte nach so langer Zeit ohne vertragliche Regelung nichts mehr unternommen.

### **6.1.4 Sonstige Erfahrungen**

Das entwerfen einer grafischen Benutzeroberfläche war schwieriger als erwartet. Vor allem die sinnvolle und unmissverständliche Beschriftung der Befehlsschaltflächen. Bspl: Beim Klicken der Befehlsschaltfläche 'schliessen' des Formulars, wird da der Datensatz mitgespeichert? Solche Fragen scheinen auf den ersten Blick einfach und logisch, aber ich habe in einzelnen Fällen viel Zeit damit verbracht, ohne die optimale Lösung zu finden.

Die Anforderungen an die Datenbank-Applikation musste ich durch Gespräche und Untersuchen des alten Programms selber herausfinden.

In den Gesprächen wurde vor allem folgendes hervorgehoben: die Geschwindigkeit der Applikation und das automatische Erscheinen der nächsten Kartenummer, obwohl diese Anforderungen nicht wichtiger waren als die von der abzulösenden SDB-Datenbank gut erfüllten Anforderungen.