Teil C: Jenseits der Hamsterwelt

In diesem Teil geht es darum, Themen aufzugreifen, die sich mit der Hamsterwelt nicht zeigen lassen. Zum einen sind dies Java-Applets. Applets sind Multimediabereiche innerhalb einer Webseite, die mit Java verwaltet werden.. Zum andern ist dies die Einbindung einer Datenbank in ein Angebot auf dem Internet.

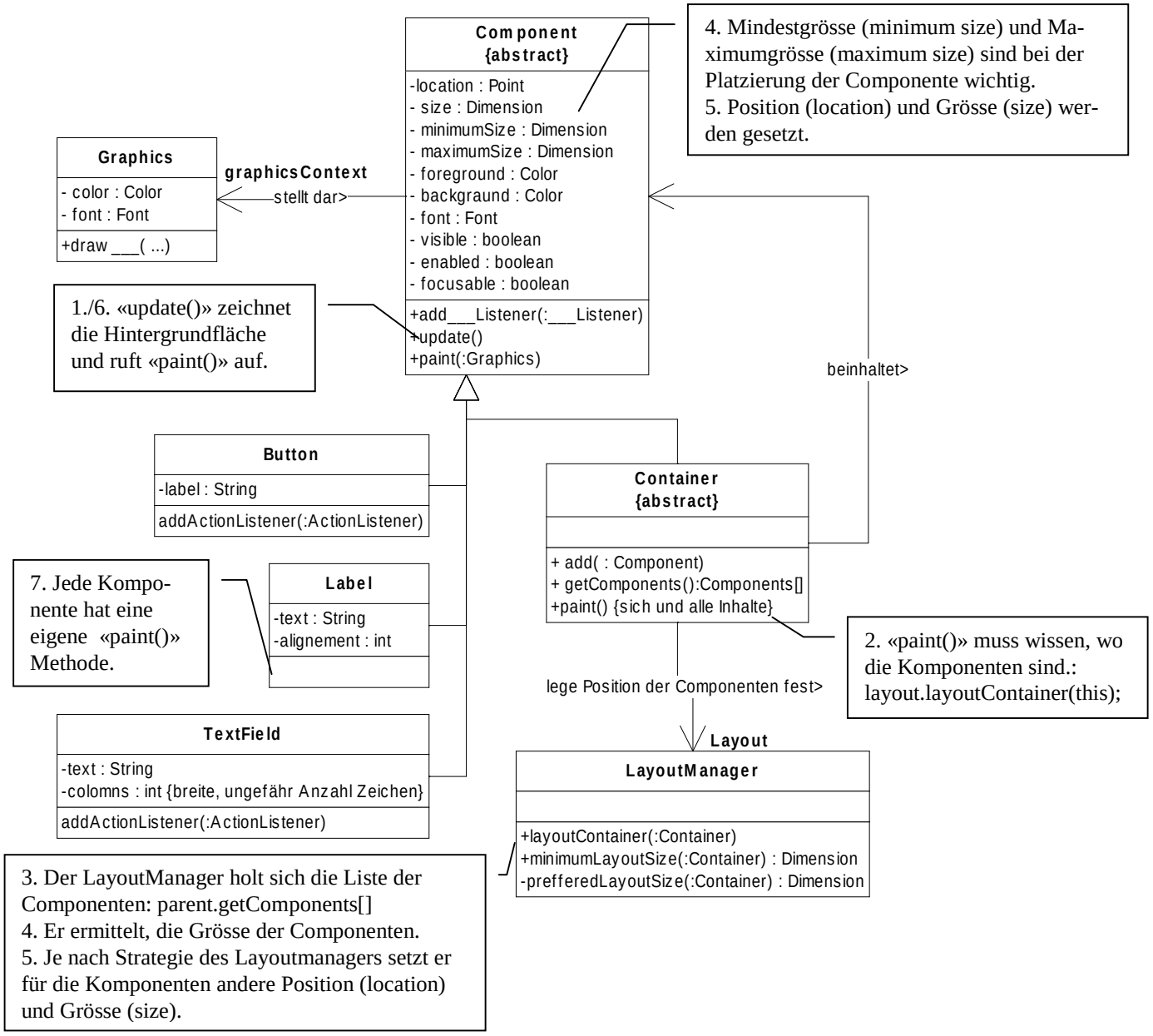
Inhaltsverzeichnis Teil C

Kapitel 1: Einfache Benutzeroberflächen

Für einfache Benutzeroberflächen stellt «Java 2» das Abstract Window Toolkit (AWT) zur Verfügung. Dieses soll hier am Beispiel einer einfachen Benutzeroberfläche erklärt werden.

Thema 1: Warum man eine Benutzeroberfläche sieht

Man sieht eine Benutzeroberfläche, weil das Programm diese auf den Bildschirm zeichnet. In diesen Zeichnungsvorgang können Sie anhand Ihres Programmcodes eingreifen.
Als Beispiel diene ein **Applet** (Fläche in einer Web-Seite, die von Java verwaltet wird), mit einem Text (Label), einem Eingabefeld (TextField) und einem Button. Ein Applet ist ein Unterklasse von Container und daher auch von Component und erbt alle Eigenschaften, die in Component oder Container eingetragen sind.
So funktioniert der Zeichnungsvorgang bei «Java 2-AWT». Er beginnt mit dem Aufruf der Methode «update()».



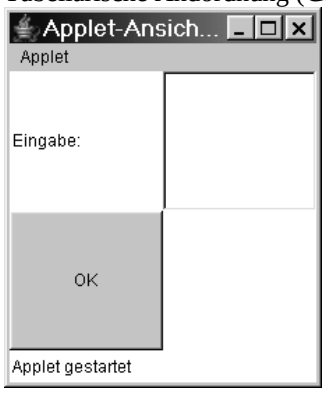
Das Designmuster «Composite-Component»

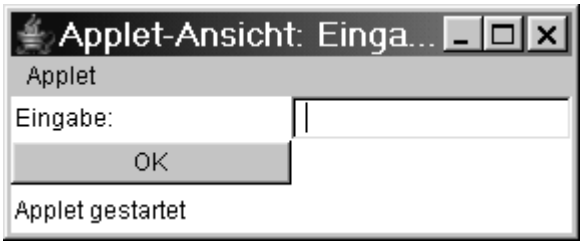
Wesentlich an diesem Vorgang ist das Designmuster «Composite-Component». Jeder **Container** (in umserem Fall das Applet), beinhaltet Komponenten. Das Programm kann beliebig viele Komponenten mit dem Befehl «add(:Component)» dem Container hinzugefügt werden. Beim Zeichnungsvorgang legt der Container die Position der Komponenten fest und fordert diese auf, sich auch zu zeichnen.
Als Komponenten kommen nicht nur einfache Komponenten wie Labels (Titel), Textfelder oder Buttons in Frage, sondern wiederum ganze Container. Das sind bei AWT **Panels** (Teilfenster).

Der LayoutManager als «strategy object»

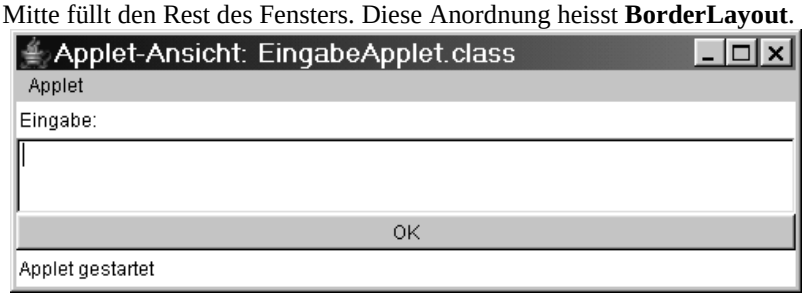
Die Strategie, wie der Container (Applet, Panel) mit Objekten gefüllt werden soll, bestimmt der LayoutManager. Die drei einfachsten Strategien sind:
Alles nebeneinander und wenn der Platz ausgeht auf die nächste Zeile wechseln, wie der Text in einer Textverarbeitung (**FlowLayout**):

Tabellarische Andordnung (**GridLayout**):





Anordnung der Componenten je eine Oben (NORTH), unten (SOUTH), links (WEST), rechts (EAST) und in der Mitte (CENTER). Die Objekte oben und unten nehmen sich die Höhe die sie brauchen, diejenigen, links und rechts die Breite, und dasjenige in der Mitte füllt den Rest des Fensters. Diese Anordnung heisst **BorderLayout**.



Selbstverständlich kann ein Programmierer weitere Layoutmanager selber programmieren. Meist verwendet verwendet er jedoch ein Hilfprogramm mit grafischer Unterstützung beim definieren der Positionen der Elemente, welches im Hintergrund mit dem sehr flexiblen aber komplexen **GridBagLayout** arbeitet.

Grafische Darstellungen der Komponenten

Einfache Komponenten wie Buttons, Labels, Textfelder delegieren bei AWT das Zeichnen der Komponenten ans Betriebssystem. Dieses zeichnet und verwaltet eine betriebssystemspezifische Komponente (Peer) an der gewünschten Stelle.
Die grafische Darstellungen eines Containers kann man bestimmen, indem man die Methode **«paint(:Graphics)»** des Containers überschreibt. Das Argument dieser Methode ist eine Referenz auf die Grafik des Containers. Mit Methoden wie «drawString("Ein Text",10,50);» kann man Texte, Linien, Rechtecke, Kreise, Vielecke, etc. zeichnen.
Codebeispiel:

```
class MeinDiagramm extends Panel
{
  public void paint(Graphics g)
  {
    g.drawRect(30,20,10,100);
    g.drawString("Ein Text",30,50);
  }
}
```

Die KLassenbibliothek **«Swing»** nutzt diese Möglichkeit. Dort sind alle Componenten Container und können deshalb nach belieben auf allen Plattformen gleich und nach belieben dargestellt werden. Die Hamsterwelt ist mit Swing geschrieben.

Aufgabe: Applet definieren und anzeigen

Definieren Sie ein Applet und fügen Sie diesem ein paar Komponenten hinzu. Verwenden Sie verschiedene LayoutManager. Den Aufbau des Applet bestimmen Sie in der Methode init:

```
public void init()
{
  Button okButton = new Button();
  TextField eingabeFeld = new TextField();
  Label feldTitel = new Label();

  okButton.setLabel("OK");
  feldTitel.setText("Eingabe:");
  setLayout(new BorderLayout());
  eingabeFeld.setColumns(20);
  add(feldTitel,BorderLayout.NORTH);
  add(eingabeFeld);
  add(okButton,BorderLayout.SOUTH);
}
```

Aufgabe: Container selbst bemalen

Definieren Sie eine Unterklasse von Panel. Überschreiben Sie in dieser, Ihrer Unterklasse die Methode «paint(:Graphics)». Verwenden Sie in Ihrer «paint()»-Methode ein paar draw-Befehle.
Instantiieren Sie in Ihrem Applet ein Objekt dieser Klasse («MeinDiagramm») und plazieren Sie es in der Mitte des Applet. Verwendung Sie dazu das BorderLayout. Im Codebeispiel weiter oben wurde das «eingabeFeld» in der Mitte plaziert.

Aufgabe: Container in Container plazieren

In der vorangegangenen Aufgabe haben Sie bereits Container (Panel) in einen Container (Applet) paziert. Verwenden Sie diesmal ein leeres Panel und fügen Sie diesem Komponenten hinzu.
Codebeispiel:

```
Panel mitte = new Panel();
mitte.add(feldTitel);
mitte.add(eingabeFeld);
add(mitte,BorderLayout.CENTER);
```

Aufgabe: Diagramm für Ablauf beim Zeichnen des Bildschirms

Erstellen Sie ein Sequenzdiagramm oder ein Kollaborationsdiagramm, mit dem Sie den Ablauf beim Zeichnen eines Applets mit drei einfachen Komponenten aufzeichnen.

Merken Sie sich

- Applets sind Java-Teile einer Web-Seite:
- Panels sind Teilfenster, in diesen könne Sie zeichnen, indem sie «paint(Graphics g)» überschreiben

```
public void paint(Graphics g)
{g.drawString("Hallo",10,20);}
```

• Container (Applets, Panel) beinhalten Komponenten nach dem Composite-Component-Muster
• Die Position der Komponenten bestimmt der LayoutManager als «strategy-object»
• Label, TextField und Button sind einfache Komponenten von AWT und werden vom Betriebssystem gezeichnet.

Thema 2: Auf ein Ereignis reagieren

Vielleicht haben Sie sich gewundert, wie Sie dem Computer beibringen können, dass nach dem Klicken auf Ok etwas passiert.

Ereignisbehandlung in «Java 2»

Jede Benutzeraktion (Mausklick, Tastatureingabe, Mausbewegung) bezeichnet man als Ereignis (Event). Sie können auf elementare Ereignisse (Mausklick an beliebigen Ort, Tastatureingabe ohne jeden Bezug), sogenannte «low level events» reagieren. Sie können aber auch, was bequemer ist, auf sinnvolle Ereignise reagieren (Benutzer hat Button geklickt, Benutzer verlässt Textfeld). Dies sind sogenannte semantische Ereignisse. Wir beschränken uns hier auf letztere.
Die Ereignisbehandlung in «Java 2» folgt der Observer-Logik. Sie brauchen einen Beobachter (Observer) für den OK-Button. In der Regel meldet man das Panel oder Applet, das den Button beinhaltet als Observer an: Der Observer für einfache Aktionen des Benutzers heisst **ActionListener**.

```
  okButton.addActionListener(this);
```

Ein Observer, der als Action Listener funktionieren soll, muss die Schnittstelle «ActionListener» implementieren, da man nur «ActionListener» als solche beim Button anmelden kann. Praktisch bedeutet dies zwei Schritte:
1. Muss das Panel oder Applet die Schnittstelle deklarieren

```
class MeinApplet extends Applet implements ActionListener
```

2. Muss man die Methode «actionPerformed()» implementieren.

```
public void actionPerformed(ActionEvent ae)
{

}
```

In dieser Methode plaziert man das, was ausgeführt werden soll, wenn der Button geklickt worden ist.
Das Objekt vom Typ «ActionEvent» enthält Informationen über das Ereignis. Zum Beispiel die Komponente, die das Ereignis ausgelöst hat.

```
  Componetn c = ae.getSource();
```

Oder die Beschriftung dieses Objektes.

```
  String cmd = ae.getActionCommand();
```

Applet und Panel als Mediator (Gruppenleiter)

In der Regel steuert man von einem Panel aus die Komponenten, die sich im Panel befinden. Das Panel meldet sich dann bei allen Komponenten, die ein Ereignis auslösen als ActionListener an.
Dies erlaubt, auf einfache Art, die Eingaben zu prüfen und den Benutzer auf Eingabefehler und Eingabemöglichkeiten Aufmerksam zu machen.
Dieses Design-Muster heisst Mediator (Gruppenleiter), weil das Panel eine Gruppe von Komponenten leitet.

Umwandlung von Text in Zahl und Zahl in Text

Ein elementares Problem in «Java 2» besteht darin, die Eingaben, welche ein String sind, in Zahlen zu verwandeln und vorgegebene Zahlen zur Anzeige (Ausgabe) wiederum in Text.
Die Umwandlung von Zahlen in Text ist sehr einfach. Man verknüpft einen leeren String mit der Zahl:

```
label.setText(""+25);
```

Oder in der Hamsterwelt, als Beispiel

```
  Hamster h = new Hamster();
  h.init(3,4,2,10);
  h.schreib(""+getReihe());
```

Die Umwandlung von Text in Zahlen ist etwas schwieriger. Im Beispiel wird der String, den getText() zurückgibt in einen Integerwert übersetzt.

```
  int i = Integer.parseInt(textField.getText() );
```

Entsprechend die Übersetzung von Zahlen mit Komma:

```
double d = Double.parseDouble(textField.getText() );
```

Merken Sie sich

• Panel (Mediator) als ActionListener definieren:	… `implements ActionListener`
• Zu Erledigendes in Methode für Ereignisbehandlung	… `actionPerformed(ActionEvent ae)`
• Mediator bei Komponente anmelden:	`okButton.addActionListener(this);`
• Zahl in String (Text) verwandeln	`label.setText(""+(3*4) );`
• Text in int verwandeln	`Integer.parseInt("2");`
• Text in double verwandeln	`Double.parseDouble( "2.5");`

Kapitel 2: Datenverwaltung via Internet

Eine Internetanwendung, die auch Daten speichert (z. B. Gästebuch, FAQ) kann recht komplex sein. Anhand einer weniger verbreiteten, aber sehr einfach zu handhabenden Entwicklungsumgebung (4th-Dimension) erarbeiten wir uns die Grundtechniken und Grundbegriffe. Dabei gehen wir auf viele technische Fragen, wie Datenschutz und Datensicherheit, um nur die wichtigsten zu nennen, nicht ein. Datenschutz wird durch Kennwörter und sichere Verbindungen gewährleistet. Datensicherhiet durch Transaktionen, welche sich beim Scheitern als ganzes rückabwickeln lassen, sowie Backups (mehrfaches systematisches Speichern der Daten) gewährleistet.

Thema 3: Architekturen

Typische Lösung

Unter Architektur versteht man den groben Aufbau einer Applikation, die aus mehrern Komponenten besteht
Eine typische Internetanwendung besteht in der Regel aus vier Teilen.
Dem **Web-Browser**, der HTML-Seiten interpretiert.
Dem **Web-Server**, der dem Web-Browser die HTML-Seiten liefert.
Einer **strukturierten Programmiersprache**, um die Abläufe zu automatisieren.
Ein **Datenbankmanagementsystem**, um die Datenablage zu organisieren.
Hier zwei gängige Lösungsansätze:

Web-Browser	**Web-Server**	**Programmiersprache (Server side scripting)**	**Datenbankmanagementsystem**
beliebig	Internet Information Server (IIS) von Windows	Active Server Pages (ASP)	MS Access
beliebig	Apache (Open Source Server für Unix)	Personal Home Page (PHP)	MySQL
beliebig	4th-Dimension mit integriertem Apache-Web-Server und eigener strukturierter Programmiersprache		

Einfache Lösungen

Für einfachere Aufgaben wird oft auf ein Datenbankmanagementsystem verzichtet. Die Programmiersprache verwaltet die Daten direkt. Die Speicherung der Daten in Dateien und das Wiederauslesen der Daten aus den Dateien auf dem Server wird von der Programmiersprache bewältigt. Damit lassen sich zum Beispiel mit PHP einfache Gästebücher realiseiren.

4th-Dimension, all-in-one-Lösung

Wir lösen die Aufgabe mit **4th-Dimension**. Diese Entwicklungsumgebung bietet alles in einer Applikation: Web-Server Verwaltung, Programmiersprache, Datenbankmanagement. Damit sparen wir Zeit und können dennoch alle entscheidenden Aspekte kurz anschauen.

Vor- und Nachteile der verschiedenen Lösungsansätze

Die All-in-one Lösung 4th-Dimension hat den Vorteil, dass sich kleinere Projekte schnell und einfach realisieren lassen. Aber auch grosse und komplexe Projekte sind mit verhältnis mässig wenig Aufwand realisierbar. Anbieter der All-in-one-Lösung bewältigen die technologischen Risiken weitgehend und geben dem Software-Entwickler ein mächtiges, leicht beherrschbares Instrument in die Hand.
All-in-one-Lösungen haben natürlich auch Nachteile. Insbesondere ist der Entwickler vom Hersteller der Lösung abhängig. Stellt der Entwickler der All-in-one-Lösung sein Produkt ein, gehen die eigenen Investitionen verloren. Dies betrifft nicht nur den geschaffenen Code, sondern auch das Know-how der Mitarbeiter.
Keine Angst bezüglich Einstellung der Weiterentwicklung einer Lösung besteht in der Regel bei breit abgestützten Open-Source-Angeboten (Apache-PHP-MySQL z. B.). Auch Lösungen der Firma Microsoft werden aus diesem Grund bevorzugt, weil diese Firma aufgrund ihrer Grösse und Marktmacht die Fortentwicklung der Produkte eher gewährleistet.
Server-Verwaltung und Programmierung sind bei diesen Lösungen aber viel aufwändiger zu lernen.

4th-Dimension installieren und starten

4th-Dimension gibt es zur Zeit in zwei aktuellen Versionen (2003 und 2004), die sich für den Anfänger kaum unterscheiden. Aufgrund des einfacheren Installationsvorganges und der Unterstützung einer grösseren Zahl von Betriebssystemen stütze ich mich hier auf die Version 2003.
Die Applikation finden Sie als Archiv unter ftp://ftp.ajar.ch/4d_products/deutsch/2003/win/4D_2003.4.exe

Für einefache Übungen reicht die Demoversion vollauf. Wollen Sie etwas vertiefter lernen, erhalten Sie eine Schullizenznummer gratis.
Zuerst gilt es eine neue leere Struktur (Applikation) anzulegen.

Geben Sie den Namen der Applikation ein. Bei meinem Beispiel ist dies WebBeispiel. 4th-Dimension legt für das Programm, die Programmressourcen und die Daten je eine Datei auf ihrem Computer an. Jetzt kann es richtig los gehen.

Merken Sie sich

- Die Wahl der richtigen Software-Architektur ist wichtig.
- Eine Internetanwendung besteht meist aus Browser, Server, strukturierten Programmiersprachen und Datenbankmanagementsystem
- Einfachere Aufgaben lassen sich ohne Datenbankmanagementsystem realisieren
- Open-Source Lösungen und Lösungen von Microsoft sind beliebt wegen deren Verlässllichkeit
- 4th-Dimension ist einfach zu lernen und dennoch ausbaubar

Thema 4: Aufbau der Datenbank festlegen

Eine Datenbank besteht aus mehreren Tabellen (Listen mit Daten, wie sie auch in Excel verwaltet werden könen). In diesen werden die Daten aufgelistet.
Beispiel: Ich möchte, dass die Teilnehmer eines Kurse eine Rückmeldung machen können, wie sie den Kurstag erlebt haben. Sie sollen das Datum, ihren Namen, Positives, Negatives und Anregungen in die Liste eintragen können.

Das Datenbankentwicklungssystem 4th-Dimension

Software-Entwicklungsumgebungen sind mächtige Instrumente mit sehr vielen Funktionen. Man verliert in Ihnen schnell einmal den Überblick. 4th Dimension ist wie folgt aufgebaut:
Es gibt 3 als Mudus bezeichnete Umgebungen: Design, Benutzer, Runtime. Wir arbeiten in der Designumgebung. In dieser gibt es 2 Hauptfenster: Struktur und Explorer.
Menü **Modus**

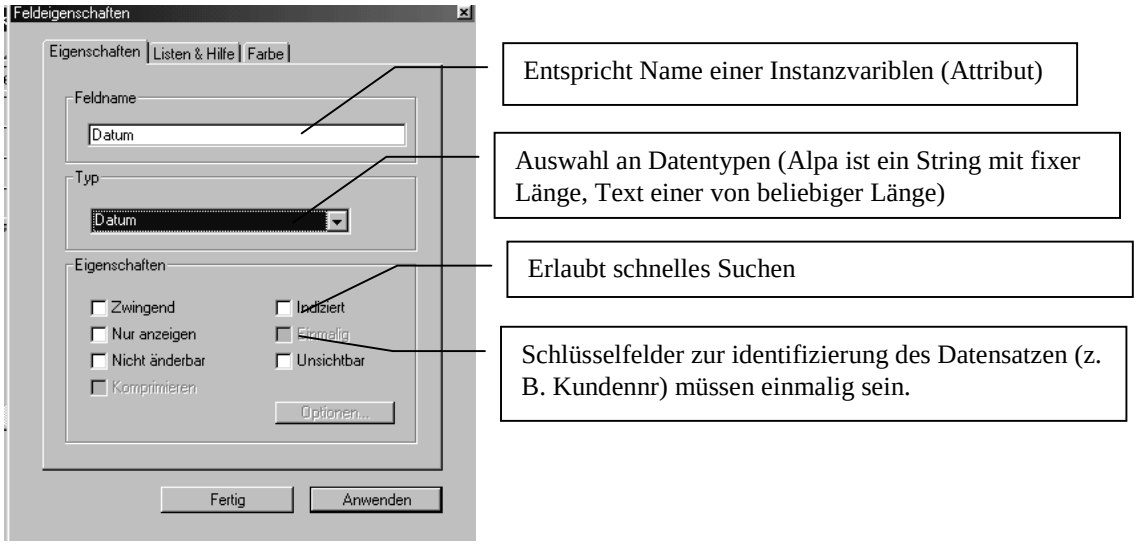
Desing (Entwicklungsumgebung)
Menü **Design**, früher Menü **Werkzeuge**
Datenbankstruktur (Aufbau der Datenbank)
Explorer (Hauptelemente des Datenbankmanagementsystems)
Methoden, Formulare
Toolbox (weitere Elemente)
Menüs, Auswahllisten, Hilfetexte, Anwender
Benutzer (Datenbankverwaltung ohne fertiges Programm)
Runtime (Fertiges Programm auf der Basis der Datenbank)

Struktur Fenster, Hauptelemente

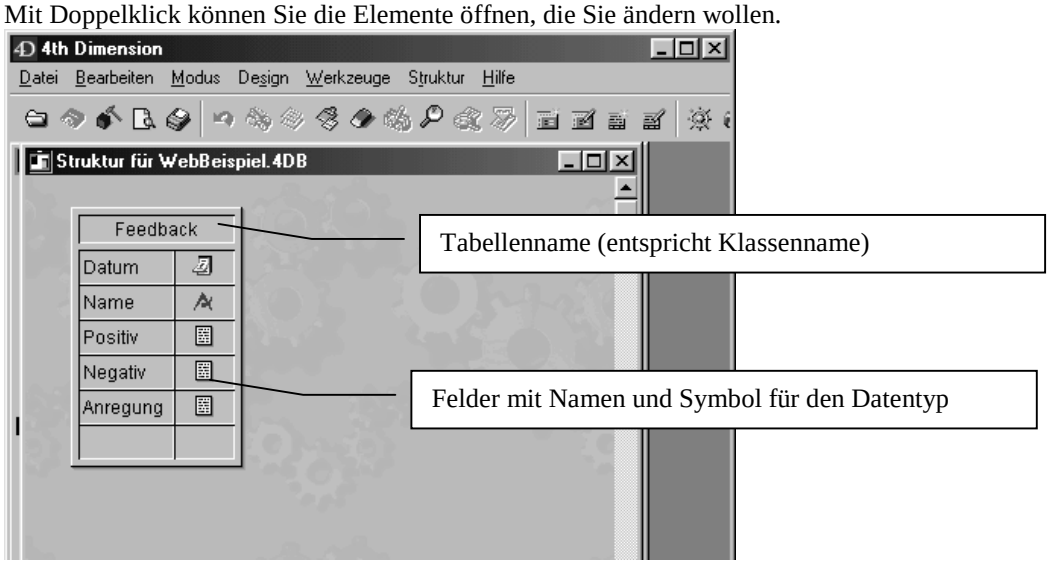
(Im Fenster Struktur) Wählen Sie im **Kontextmenü «Neue Tabelle»**. Eine Tabelle hat grosse Ähnlichkeit mit einer Klasse.

	Datenbank	**Klassen und Objekte**
Hauptstrukturelement (Entität genannt)	Tabelle	Klasse
Einteilung	Felder mit Namen	Attribute (Instanzvariablen mit Namen)
Objekte mit Daten	Datensatz	Objekt
Beziehungen zwischen Hauptelementen (Entitäten)	Relationen	Assoziationen
Operationen	Allgemeine Datenbankfuntkionen	Spezifische Operationen sind in der

		Klasse definiert
Bezugnahme	Über Schlüsselfelder	Objekte haben eine Identität
Existenz	Permanent gespeicherte Daten	Zur Laufzeit
Datentypen	Beschränkte Anzahl Typen. BLOB (Binary Large Objects) erlauben es aber beliebige Daten zu speichern.	Frei definerbare eigene und vorgegebene Typen



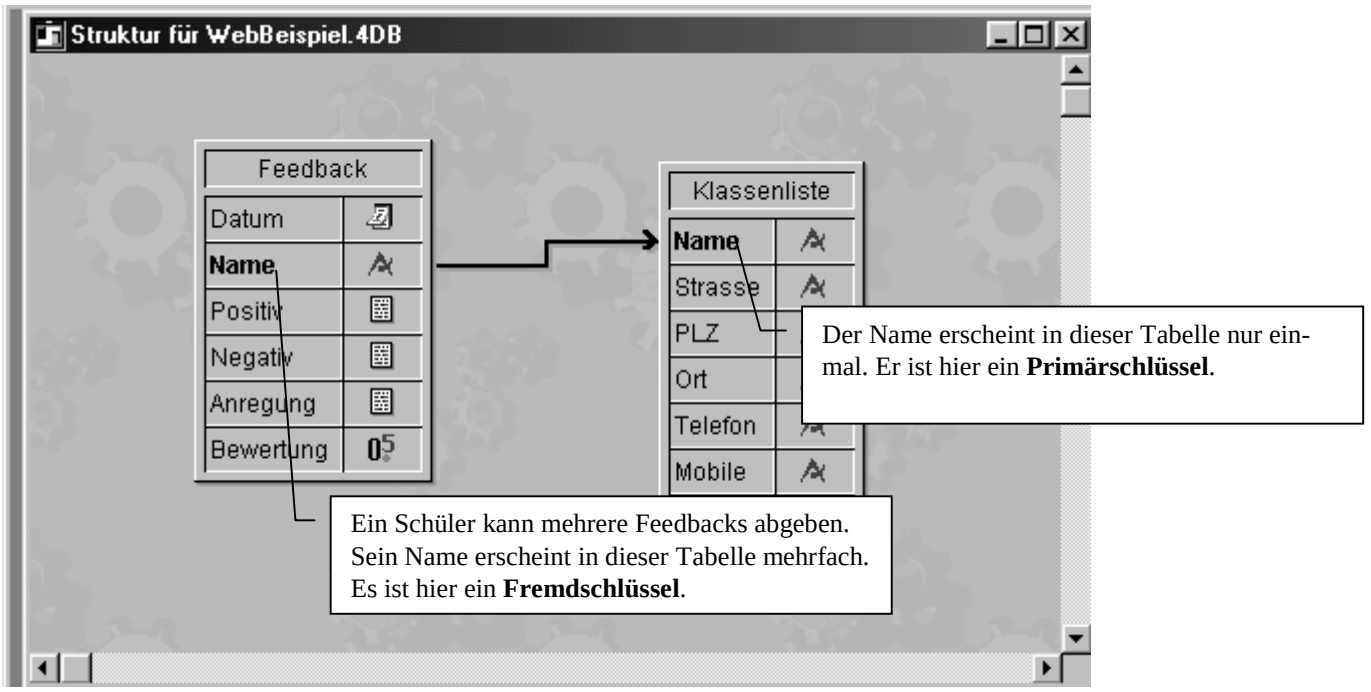
Mit Doppelklick können Sie die Elemente öffnen, die Sie ändern wollen.



In meiner Liste habe ich das Datum als Datum festgelegt, den Namen als alphanumerisches Feld (dies bedeutet, dass Buchstaben und Zahlen gespeichert werden können) und die anderen Felder als Text. Das heisst, dort kann fast beliebig viel Text eingegeben werden.

Beziehungen (Relationen)

Falls mehrere Tabellen vorhanden sind können Beziehungen (die Entsprichung zu Assoziationen) durch ziehen mit der Maus in Beziehung zu einander gesetzt werden.



In «4th-Dimension» erstellt man die Beziehung, indem man vom Feld mit dem Fremdschlüssel zum Feld mit dem Primärschlüssel zieht.
Das Feld Name ist zwar ein einfaches Beispiel für einen Primärschlüssel. In der Praxis wird man aber andere Lösungen bevorzugen, denn der Name hat folgende Nachteile:
1. Gibt es oft mehrere Personen mit dem gleichen Namen. Ein Primärschlüssel darf aber nur einmal vorkommen.
2. Ändern Personen ihren Namen durch Heirat. Primärschlüssel sollten aber nie ändern.
3. Besteht die Gefahr, dass der Name bei der ersten Erfassung fehlerhaft eingegeben wurde.
In der Praxis hat sich deshalb druchgesetzt, dass man als Primärschlüssel Felder mit rein technischen Informationen verwendet, z. B. eine Schülernummer.

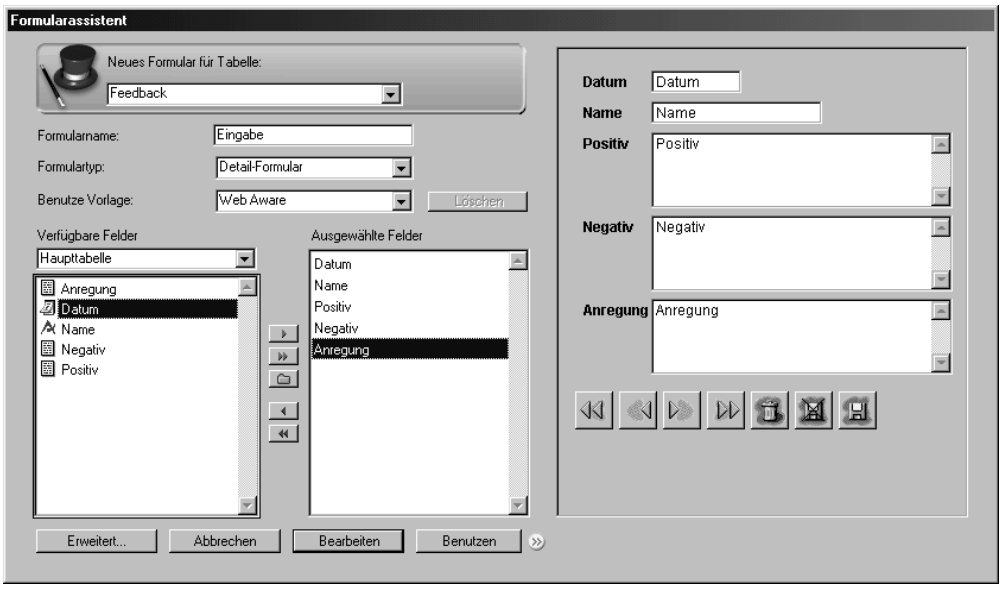
Merken Sie sich
- Eine Datenbank hat eine **Struktur** (einen Aufbau), der den Klassen und ihren Assoziationen ähnlich ist.
- Eine Datenbank besteht aus **Tabellen**
- Tabellen sind eingeteilt in **Felder**, die einen **Feldnamen** und einen **Dateentyp** haben
- Tabellen enthalten als Objekte **Datensätze**, auf die über den **Primärschlüssel** Bezug genommen wird.
- Ein **Datenbankmanagementsystem** erleichtert die Benutzung einer Datenbank und vielfach auch die Erstellung von fertigen Programmen, die auf der Datenbank aufbauen.

Thema 5: Benutzeroberfläche definieren

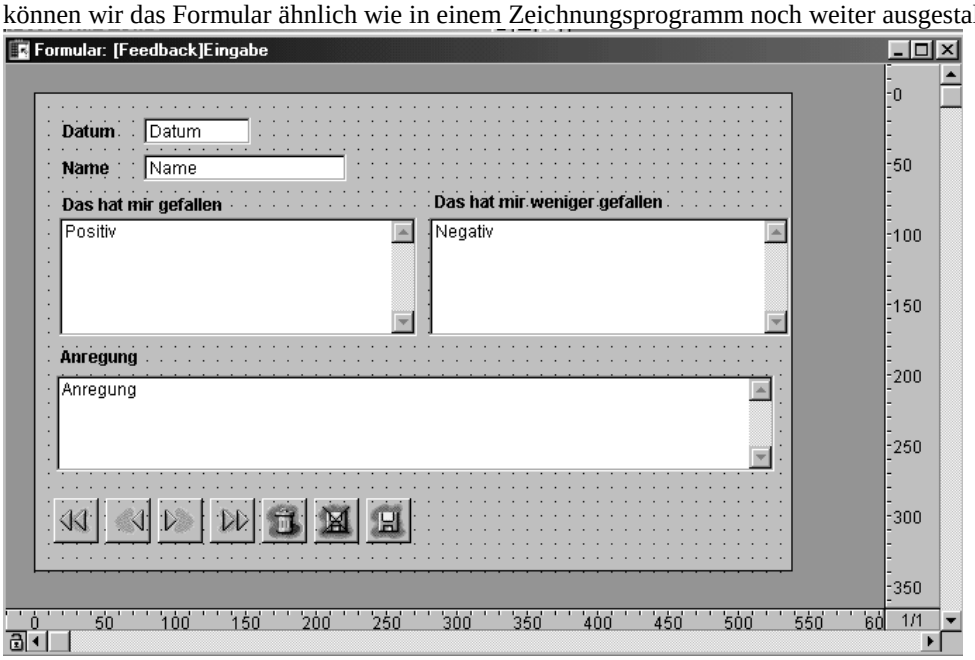
Eine Benutzeroberfläche besteht aus einer Benutzerführung (Menüstruktur), den Auflistungen (in 4th-Dimenstion Ausgabeformulare genannt), den Eingabemasken (Eingabeformuleren) und den programmierten Abläufen, die das ganze Steuern.

Eingabemasken und Listen definieren

«4th-Dimension» hilft uns, Eingabe- und Ausgabeformulare schnell zu erstellen.
Wir wählen im Menü Design, den Befehl Neues Formular und stellen das Formular zusammen. Wichtig: Es soll ein Detail-Formular aufgrund der Vorlage **Web-Aware** sein.



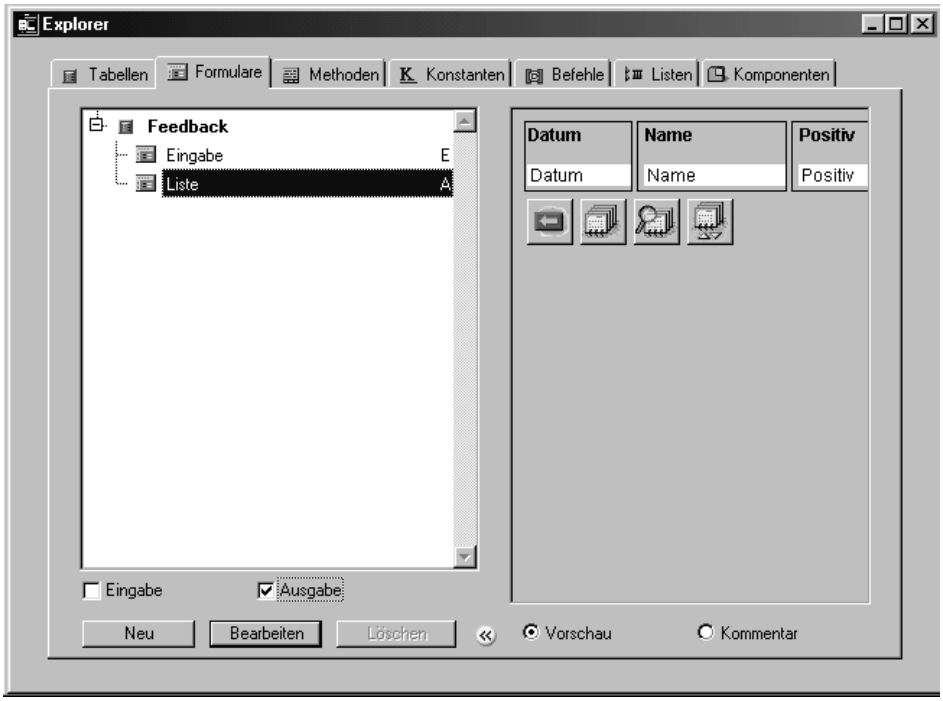
Sind die Felder, die im Formular erscheinen sollen bestimmt, so klicken wir auf Bearbeiten. Im nächsten Dialog können wir das Formular ähnlich wie in einem Zeichnungsprogramm noch weiter ausgestalten.



Beim Listenformular gehen wir analog vor. Statt Detailformular wählen wir Listenformular. Beachten Sie die vorbildliche Dialoggestaltung von 4th-Dimension. Sie können die Feldliste mit Klick auf die entsprechenden Buttons, Doppelklick auf die auszuwählenden Felder oder Drag-and-drop zusammenstellen oder verändern.

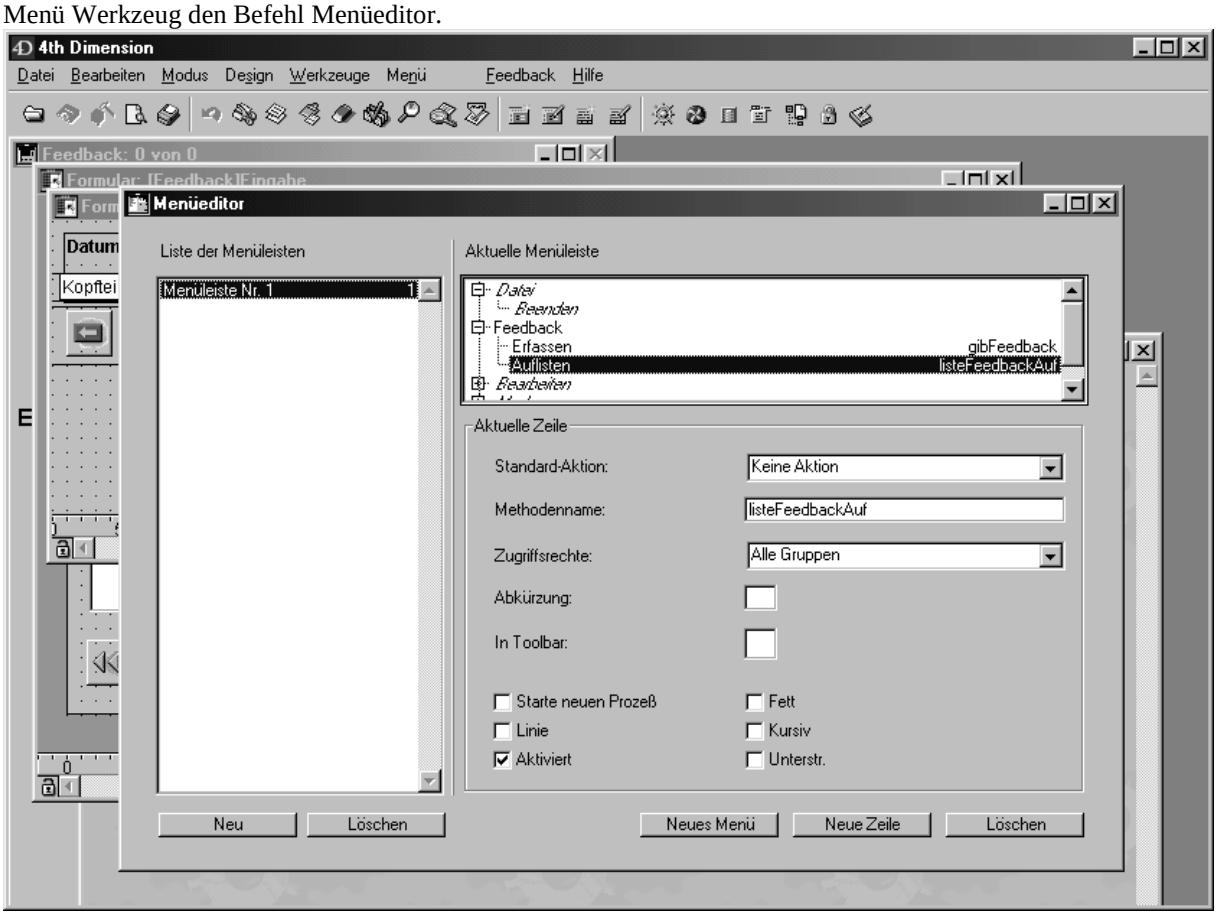
Eingabe- und Ausgabeformulare festlegen

Mit dem Werkzeug Explorer können wir das standardmässige (default-mässige) Eingabe- und Ausgabeformular festlegen. Ein Ausgabeformular dient in «4th-Dimension» der Auflistung von Datensätzen, dem Überblick, das Eingabeformular dient dem Eingeben und Ändern. Andere Datenbankentwicklungssysteme (z. B. Acces) machen diese Unterscheidung nicht. Daten lassen sich dort auch ohne Umwege in der Listenform erfassen und ändern.



Navigationsmenü zusammenstellen

Unsere Formulare müssen jetzt noch in eine Menüstruktur eingebunden werden. Dazu bietet 4th-Dimension im Menü Werkzeug den Befehl Menüeditor.

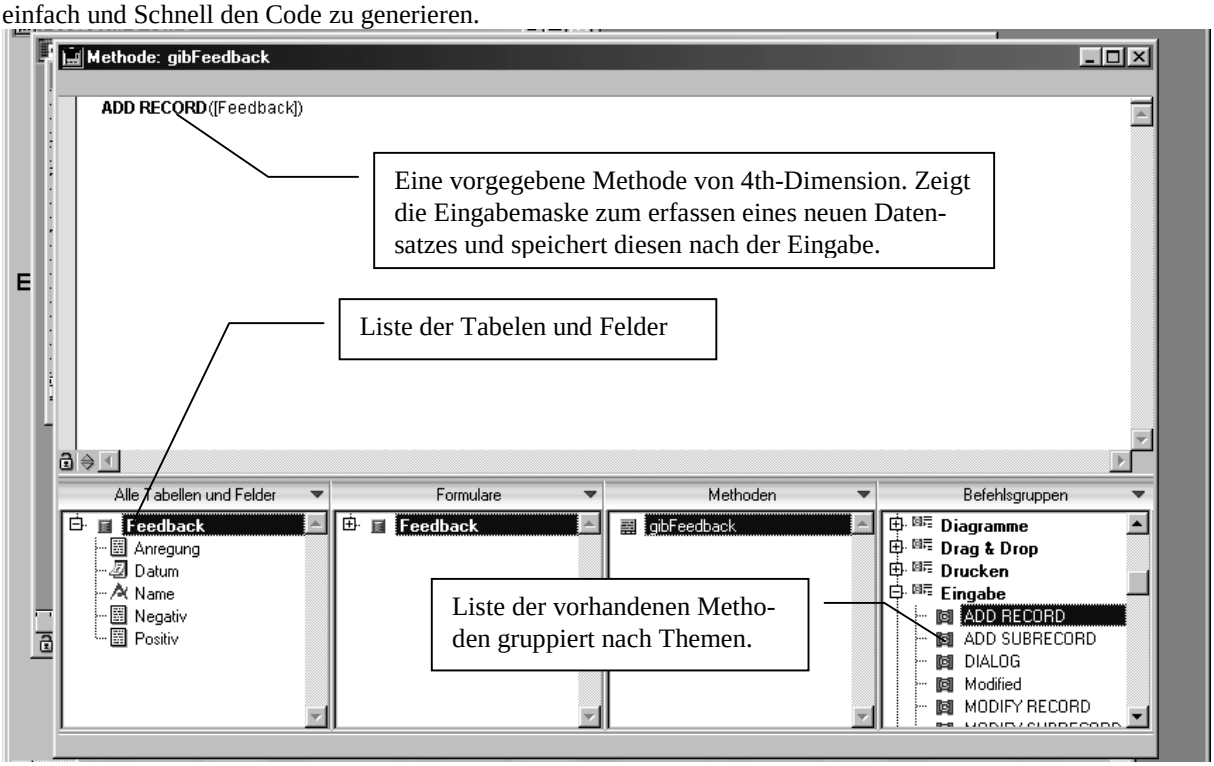


In der «Menüleiste Nr 1» fügen wir das neue Menü «Feedback» ein und als erste Menüzeile Eingabe. Bei Wahl dieser Menüzeile soll die Methode «gibFeedback» aufgerufen werden.
Einer weiteren Menüzeile weisen wir den Aufruf der Methode «listeFeedbackAuf» zu.

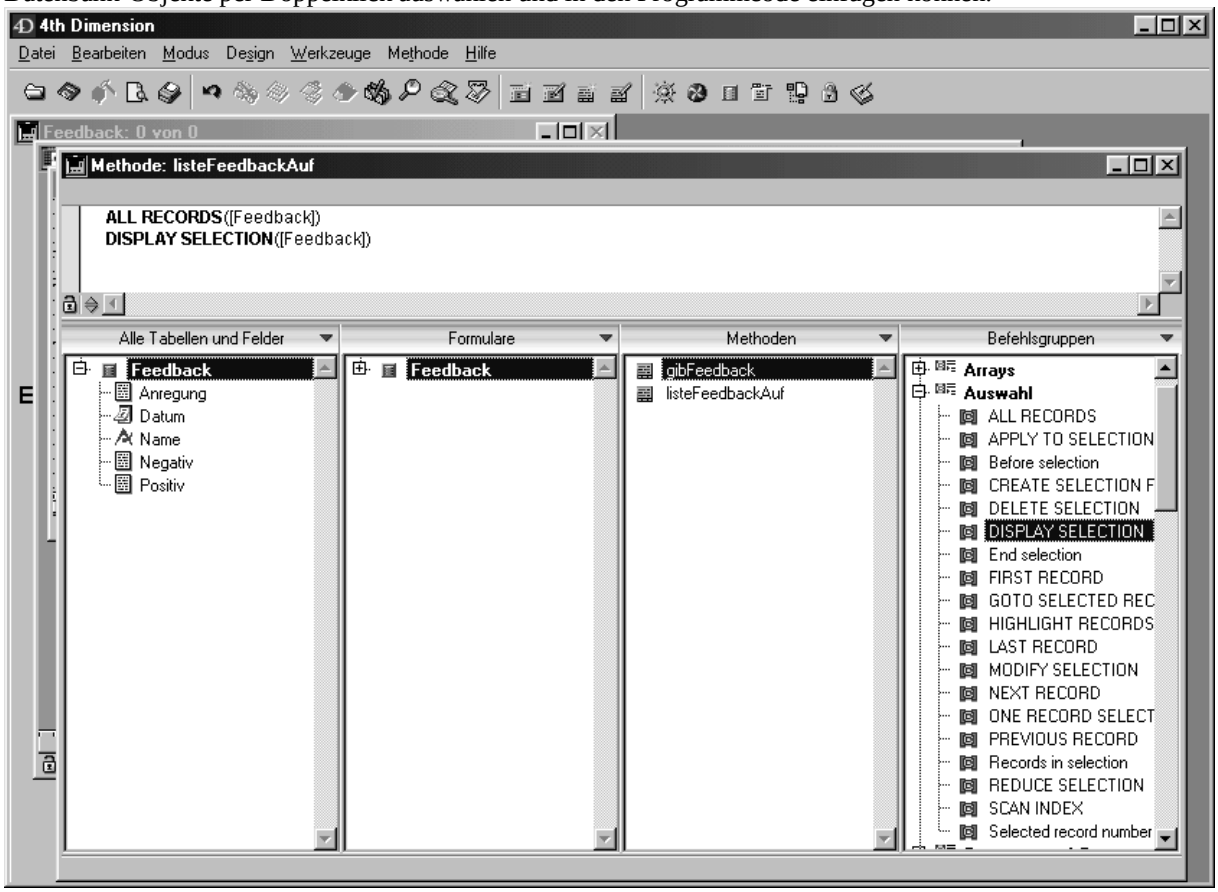
Abläufe programmieren

Die beiden Methoden, die im Menü aufgerufen werden, müssen wir noch programmieren. 4th-Dimension und andere Datenbankmanagementsysteme bieten dazu zahlreiche massgeschneiderte Methoden an.

Im Menü Design finden wir den Befehl neue Methode. Programme wie «4th-Dimension» bieten Hilfen, um einfach und Schnell den Code zu generieren.



Die Methode «gibFeedback» besteht nur aus einem Befehl. In Tabelle Feedback soll ein neuer Datensatz (Eintrag) erstellt werden. Beachten Sie, dass die Befehle thematisch geordnet sind und sie sowohl Befehle als auch Datenbank-Objekte per Doppelklick auswählen und in den Programmcode einfügen können.



Diese Methode befielt zuerst alle Datensätze auszuwählen und diese Auswahl (Selection) anzuzeigen (to display) und nicht etwa zu ändern (to modify).
Beachten Sie auch, dass 4th-Dimension, wie auch die meisten der ähnlichen Produkte keinen objektorientierten Code verwenden. Die Objekte der Datenbank werden als Argumente den Methoden mitgegeben.

Prototyping

Mit Tools wie 4th-Dimension läst sich sehr schnell eine Benutzeroberfläche aufbauen. Dadurch kann im Gespräch mit dem Anwender anhand eines Prototyps, der noch nicht funktionsfähig ist, eine Bedürnisanalyse durchgeführt werden. Diesen Vorgehen nennt sich Prototyping.

Merken Sie sich

- Eine Benutzeroberfläche besteht aus einer **Menüstruktur** und Fenstern.
- Es gibt **Übersichtsfenster** mit Listen (Ausgabeformulare) und **Detailfenster** (Eingabefenster).
- Menüs rufen **programmierte Abläufe** auf, welche die Fenster zeigen und Eingaben verarbeiten.

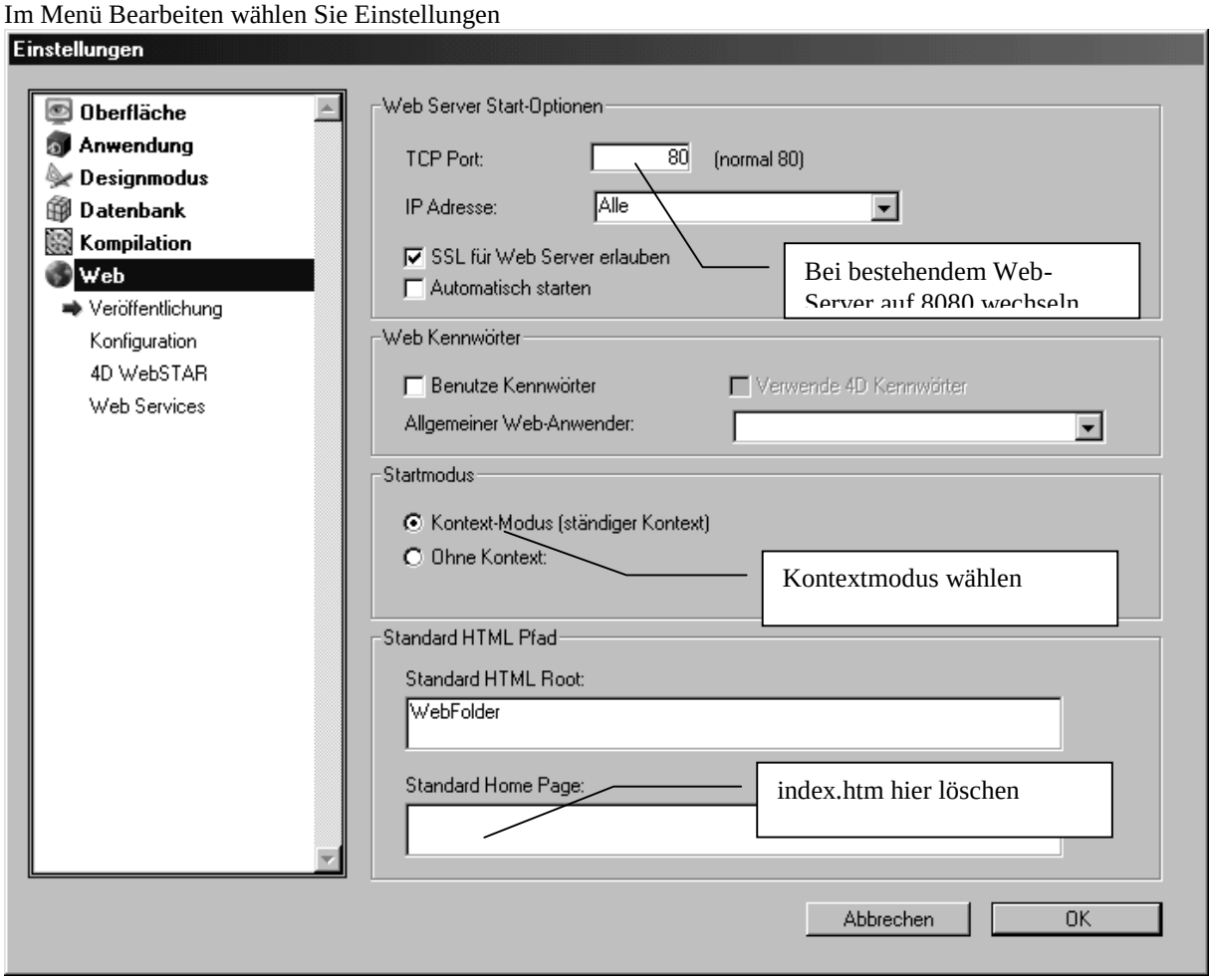
Thema 6: Web-Server in Betrieb nehmen

Die Verwaltung eines Web-Servers ist eine aufwändige Angelgenheit. Bei 4th-Dimension sind aber nur drei Dinge zu tun:

- Bestimmen, dass die Website über die Menüs von 4th-Dimension navigierbar sein soll (Kontext-Modus).
- Standard Homepage löschen
- Webserver starten.

Kontext-Modus, Standard-Homepage

Im Menü Bearbeiten wählen Sie Einstellungen

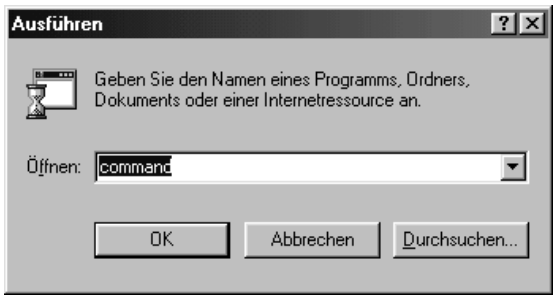


Web-Server starten

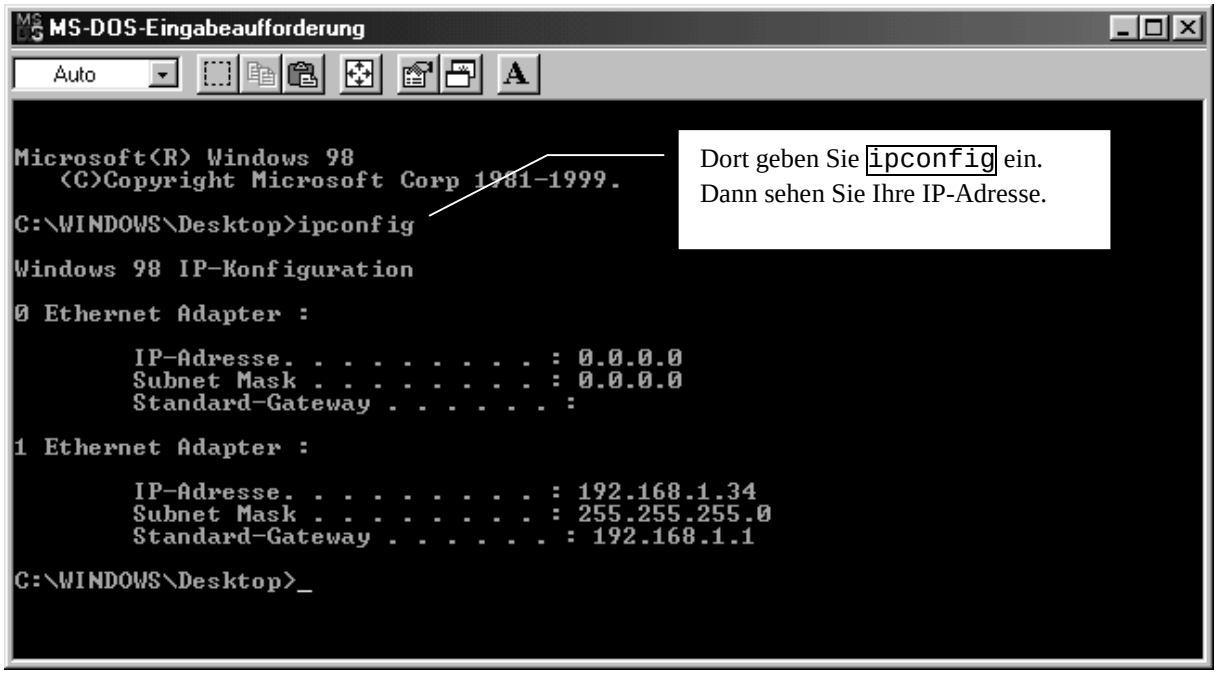
Wählen Sie im Menü Modus den Befehl Benutzer/Runtime modus. Anschliessend könenn Sie im Menü Webserver den Server starten. Dieser läuft in der gratis Schülerversion während einer Stunde.
Während dieser Stunde kann über die IP-Nummer von jedem Browser aus problemlos auf Ihre Datenbank zugegriffen werden.

IP-Nummer des eigenen Computers ermitteln

Unter Windows wählen Sie Start - Ausführen und geben als Programm Command ein.



Das DOS-Fenster öffnet sich.



Im Browser würden Sie im obenstehenden Fall als Adresse:
`http://192.168.1.34`
eingeben.
Auf Ihrem eigenen Computer können Sie auch
`http://localhost`
oder
`http://127.0.0.1`
eingeben.

Falls schon ein Web-Server auf Ihrem PC läuft

Geben Sie bei den Einstellung des Web-Server in 4D als **Port** `8080` ein, statt des normalen `80` für Webserver. Beim Aufruf im Browser müssen Sie auch die `8080` eingeben: Zwischen IP-Adresse und der Portnummer (8080) muss ein Doppelpunkt stehen.
`http://192.168.1.34:8080`

Merken Sie sich

- Um eine Datenbank ins Internet zu stellen braucht es Web-Server, HTML-Formulare, eine Programmiersprache und ein Datenbankmanagementsystem.
- Es gibt dazu viele Lösungsmöglichkeiten, nicht alle wird es auch in 5 Jahren noch geben.

Stichwortverzeichnis