

# JAVA und die grafische Benutzeroberfläche

## Events, Mausereignisse

### Alfred Nussbaumer

Das Ereignismodell von JAVA wurde in PCNEWS-92 an Hand von Tastaturereignissen bereits kurz vorgestellt (vgl. [2]). Neben diesen gehören Mausereignisse und Ereignisse in Zusammenhang mit dem Anwendungsfenster zu den elementarsten Ereignissen. Sie werden in einfachen Beispielen in diesem Artikel besprochen.

### 1. Fensterereignisse (WindowEvents)

Fensterereignisse treten auf, wenn sich der Zustand eines Fensters ändert. Sie werden somit von einem Window-Event ausgelöst und entweder von einem WindowListener-Objekt oder von einem WindowAdapter-Objekt entgegen genommen. Das folgende Beispiel läuft im Gegensatz zu Applets (vgl. [2]) in einem eigenen Anwendungsfenster ab. Die Grundlage dafür bietet die Klasse JFrame, die die für Anwendungsfenster typische Umgebung liefert: Das Fenster hat auf eine bestimmte Position und Größe auf einem bestimmten Bildschirm, einen Eintrag in seiner Titelleiste und kann dem verwendeten Betriebssystem entsprechend beispielsweise verschoben, minimiert, maximiert, eingerollt, geschlossen,... werden. Die eigene Anwendung erweitert die Klasse JFrame und erbt somit seine Eigenschaften.

Im folgenden JFrame-Beispiel Draw1.java dient die Adapterklasse WindowAdapter dazu, das Fensterereignis „WINDOW\_CLOSING“ zu verarbeiten: In diesem Fall wird das laufende Programm durch den Aufruf der System-Methode exit() beendet.

#### Draw1.java

```
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;

public class Draw1 extends JFrame {

    public Draw1() {
        addWindowListener(new WindowAdapter() {
            public void windowClosing(WindowEvent evt) {
                System.exit(0);
            }
        });
    }

    public static void main(String[] args) {
        Draw1 proggi = new Draw1();
        proggi.setTitle("... ein leeres Fenster ...");
        proggi.setSize(400,300);
        proggi.setLocation(100,100);
        proggi.show();
    }
}
```

Die Adapterklasse WindowAdapter wird dem JFrame-Objekt im Konstruktor mit Hilfe der Window-Methode addWindowListener hinzugefügt. Im Hauptprogramm sorgt die Window-Methode show() dafür, dass das Fenster angezeigt wird. Die Methoden setSize() und setLocation() werden von der Oberklasse Component vererbt – sie legen die Fenstergröße und ihre Position fest. Die Methode setTitle() übergibt eine Zeichenkette an die Titelleiste des Anwendungsfensters.

Die Anwendung Draw1 verläuft unspektakulär: Das Fenster wird an der angegebenen Position mit einem 400 Pixel mal 300 Pixel großen Bereich ausgegeben. Es hat – abhängig vom verwendeten Betriebssystem – typische Fenstereigenschaften. Beim Schließen des Fensters wird die mit der Anwendung gestartete Java Virtual Machine gestoppt.

Die Fensterereignisse, die beispielsweise auftreten, wenn ein Fenster deaktiviert oder aktiviert wird, wenn es auf eine minimale Größe verkleinert oder wieder geöffnet wird, usw. werden in der Java-Dokumentation (vgl. [1]) detailliert beschrieben.

### 2. Mausereignisse

Mausereignisse treten auf, wenn eine bestimmte Maustaste gedrückt wurde, oder wenn die Maus über ein Objekt gezogen wird. Dazu kommen noch Ereignisse, die in Zusammenhang mit dem Mausrad auftreten. Die folgenden Interfaces und die dazu passenden Adapterklassen werden verwendet:

- mouseClicked(MouseEvent e) - wird bei einem Mausklick ausgelöst. Dies ist der Fall, wenn eine Maustaste gedrückt und anschließend wieder ausgelassen wurde.
- mouseEntered(MouseEvent e) – wird ausgelöst, wenn der Mauszeiger den Bereich eines Objekts erreicht
- mouseExited(MouseEvent e) – wird ausgelöst, wenn der Mauszeiger den Bereich eines Objekts verlässt
- mousePressed(MouseEvent e) – wird ausgelöst, wenn der Mauszeiger auf ein Objekt weist und die Maustaste gedrückt wird
- mouseReleased(MouseEvent e) – wird ausgelöst, wenn die gedrückte Maustaste wieder losgelassen wird.
- mouseDragged(MouseEvent e) – wird ausgelöst, wenn der Mauszeiger bei gedrückter Maustaste bewegt wird.
- mouseMoved(MouseEvent e) – tritt auf, wenn der Mauszeiger bewegt wird.
- mouseWheelMoved(MouseWheelEvent e) – tritt auf, wenn das Mausrad betätigt wird.

Die Klasse MouseEvent enthält dabei Methoden wie getButton(), getPoint(), getX() oder getY(). Alle Objekte und Methoden können in der JAVA-Dokumentation ([1]) genau nachgelesen werden.

### 3. Beispiel: Punkte mit der Maus setzen

Mit einem Mausklick sollen auf einem leeren Rechteck kleine ausgezeichnete Kreise gezeichnet werden. Außerdem sollen die Koordinaten des Mauszeigers beim Klick ausgegeben werden. Die Ausgabe aller Objekte erfolgt mit Hilfe eines so genannten Containers, der im vorliegenden Fall den gesamten Inhalt des Fensters („ContentPane“) aufnimmt. (Darin unterscheiden sich JFrame-Anwendungen gründlich von Frame-Anwendungen.)

#### Draw2.java

```
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;

public class Draw2 extends JFrame implements MouseListener {

    private int xpos;
    private int ypos;

    public Draw2() {
        addWindowListener(new WindowAdapter() {
            public void windowClosing(WindowEvent evt) {
                System.exit(0);
            }
        });
        Container c = getContentPane();
        c.setLayout(new FlowLayout());
        addMouseListener(this);
    }

    public void mouseClicked(MouseEvent e) {
        xpos = e.getX();
        ypos = e.getY();
        repaint();
    }

    public void mouseEntered(MouseEvent e) {
    }

    public void mouseExited(MouseEvent e) {
    }

    public void mousePressed(MouseEvent e) {
    }

    public void mouseReleased(MouseEvent e) {
    }

    public void paint(Graphics g) {
        g.setColor(Color.white);
        g.fillRect(0,0,400,300);
    }
}
```



# Webdesign-Tipps

Nachlese vom Clubabend am 3.3. 2005

Franz Fiala

WebWord

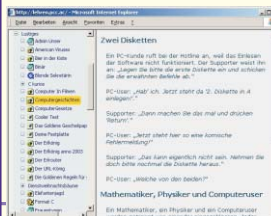
Anhang, Seite 1



Tipps für die Vorbereitung von Word-Dokumenten für die Publikation als Html-Datei. Unter Ausnutzung von Frames sowie Master- und Unterdokumenten können mit wenig Nacharbeit navigierbare Html-Dokumente hergestellt werden.

<http://demo.pcc.ac/> -> **Webword**

Baumdarstellung



Internet-Publikation ohne Programmieren. Dieses ASPX-Programm kombiniert einen kommerziellen Modul (für Clubmitglieder lizenziert) mit einem Zusatzprogramm, das jegliche Programmierung bei der Herstellung von baumartigen Verzeichnissen entbehrlich macht.

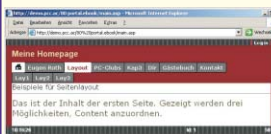
<http://demo.pcc.ac/> -> **Downloads -> tv**

Das Programm erfordert einen Internet Information Server mit installiertem Dot-Net-Framework (ist für die Verwendung am Clubweb-space geeignet). Eine Datenbank ist nicht notwendig. Die Struktur ergibt sich durch die Anwendung einiger weniger Regeln, wie zum Beispiel: Ordner und Dateien verbergen oder zeigen, Reihenfolge einstellen, Symbole wählen, Ordner für Quellkodendarstellung, Ordner für Bilderdarstellung, Autostartdokumente.

**Beispielinstallationen** <http://pcc.ac/>, <http://lehren.pcc.ac/>, <http://demo.pcc.ac/>, <http://fiala.cc/-/tv/>

(Bastel-)Portal

Anhang, Seite 15



Das ASP-Portal wird durch eine Access-Datenbank gesteuert. Es wird im Rahmen eines ASP-Seminars hergestellt und kann als Ausgangspunkt für eigene Entwicklungen verwendet werden. Die Datenbank steuert Navigation, Layout und

Content eines Webs.

<http://demo.pcc.ac/> -> **Downloads -> Portal**

Features

Zweistufiges Menü mit horizontaler Anordnung; Menüeintrag als Text oder Button; Userverwaltung (Vorname, Zuname, E-Mail, Username, Passwort, Berechtigung); Seiten und Inhalte können bestimmten Usern gezeigt werden; Jede Seite mit Zugriffszähler, der verborgen werden kann; Gästebuch mit Freischaltung durch Administrator; Antwortseite; Mitlaufende Uhrzeit auf jeder Seite; Inhalt einer Seite kann aus der Datenbank oder aus einer Datei kommen; Eine Seite kann aus einem einzelnen Text stammen oder aus mehreren Texten zusammengesetzt sein, wobei eine zwei- oder dreispaltige Anordnung gewählt werden kann; Ein Text kann mehrmals (auf derselben oder auf verschiedenen Seiten) verwendet werden Administrator kann alle beteiligten Datenbanken online editieren; Es kann mehrere Administratoren geben; Hilfsprogramm zur Darstellung von Tabellen.

Clubabende

<http://pcc.ac?id=treffen>

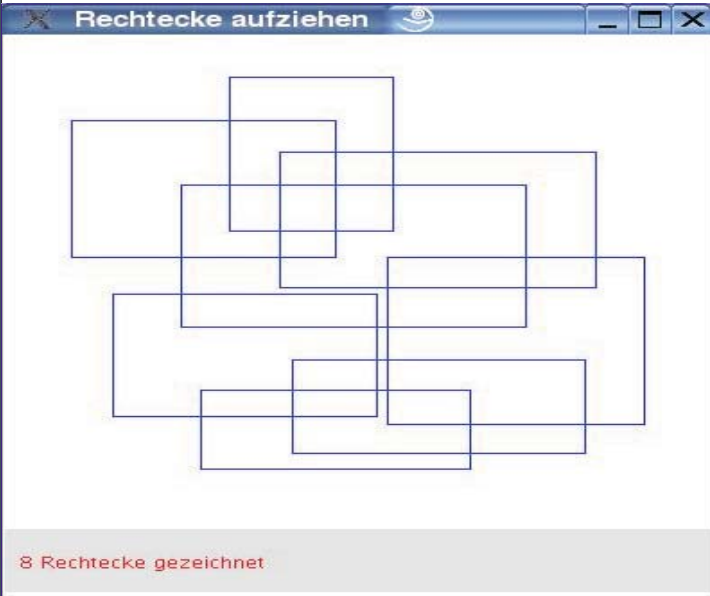
Wir laden alle PCNEWS-Leser ein, unsere Clubabende, jeweils am ersten Donnerstag im Monat in Wien-Ottakring zu besuchen. Anmeldung nicht erforderlich. Termine siehe <http://pcc.ac?id=termine> oder <http://pcnews.at?id=regeltermine&n=1>. Über den Newsletter PCNINFO (<http://pcc.ac?id=pcninfo>) erhalten Sie eine Einladung mit Hinweisen auf das Vortragsthema.

Bilder vom Clubabend

• <http://bildarchiv.pcc.ac?datum=2005.03.03>

```
public static void main(String[] args) {
    Draw3 proggi = new Draw3();
    proggi.setTitle("Rechtecke aufziehen");
    proggi.setSize(400,450);
    proggi.setLocation(100,100);
    proggi.show();
}
```

Im vorliegenden Beispiel werden die Daten in einzelnen Arrays mit einer feststehenden Größe gespeichert. Aus diesem Grund wird vor jeder Aktion überprüft, ob der Speicherplatz noch ausreicht. Die Gesamtzahl aller bereits gezeichneten Rechtecke wird deshalb in einer Statuszeile ausgegeben:



## 5. Aufgaben, Ausblick

Zeichenprogramme unterscheiden sich von den hier vorgestellten Beispielen gewaltig. Dennoch können einige überschaubare Aufgaben bewältigt werden:

1. Weitere Grafikobjekte (Linien, Kreise, Polygone, etc.) sind zu verwenden. Die Auswahl der Objekte kann beispielsweise über einen Tastaturcode erfolgen...
2. Grafikobjekte sollen im Nachhinein ausgewählt und beispielsweise gelöscht werden können.
3. Verschiedene Grafikobjekte, Zeichenfarben, Pinselstärken und andere Eigenschaften (beispielsweise geschlossene Objekte ausfüllen) sollen über ein Menü-System zur Auswahl gestellt werden. Die gewählten Eigenschaften sollen (in Kurzform) in der Statuszeile ausgegeben werden.
4. Bestimmte Eigenschaften (Pinselauswahl, Zeichenfarbe, etc.) sollen mit Hilfe von Schaltflächen zur Auswahl angeboten werden. Diese Schaltflächen bilden zusammen eine Reihe von Tools entlang einer senkrechten Bildschirm-Fensterreihe.

Nicht alle Aktionen, die „mit der Maus ausgeführt“ werden sind an MouseEvents gebunden. In diesem Artikel haben wir das Bewegen der Maus, das Ziehen und das Drücken der Maustasten behandelt, die mit den JAVA-Interfaces `MouseListener` und `MouseMotionListener` entgegengenommen werden. Für den Klick auf eine Schaltfläche verwenden wir beispielsweise einen `ActionListener`, einen `ItemListener` für die Auswahl eines Auswahlkästchens oder einen `AdjustmentListener` für die Position eines Scrollbalkens. Beispiele dazu sollen zu einem späteren Termin hier vorgestellt werden.

## 6. Literatur, Weblinks

- [1] <http://java.sun.com/j2se/1.4.2/docs/index.html> (Dokumentation aller verfügbaren Packages)
- [2] PCNEWS 92, S. 24 („JAVA und die grafische Benutzeroberfläche“)
- [3] Herbert Schildt, „Java 2 Ent-Packt“, mitp-Verlag
- [4] Christian Ullenboom, „Java ist auch eine Insel“, Galileo Computing
- [5] <http://www.gymmelk.ac.at/nus/informatik/wpf/JAVA> (Unterrichtsbeispiele zum Programmieren mit JAVA)