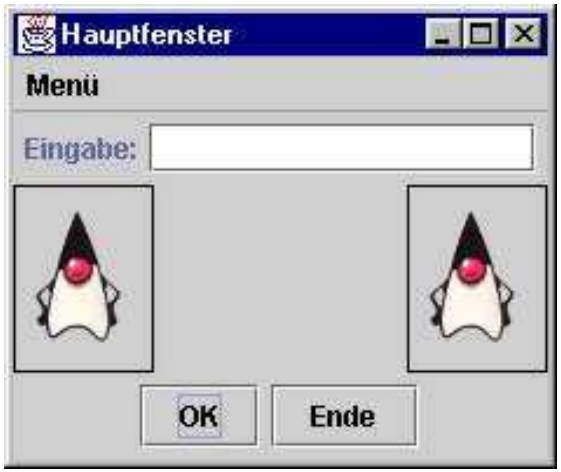
GUI-Programmierung mit JAVA-Swing

- Komponenten
- Layout
- Event Handling
- Imaging



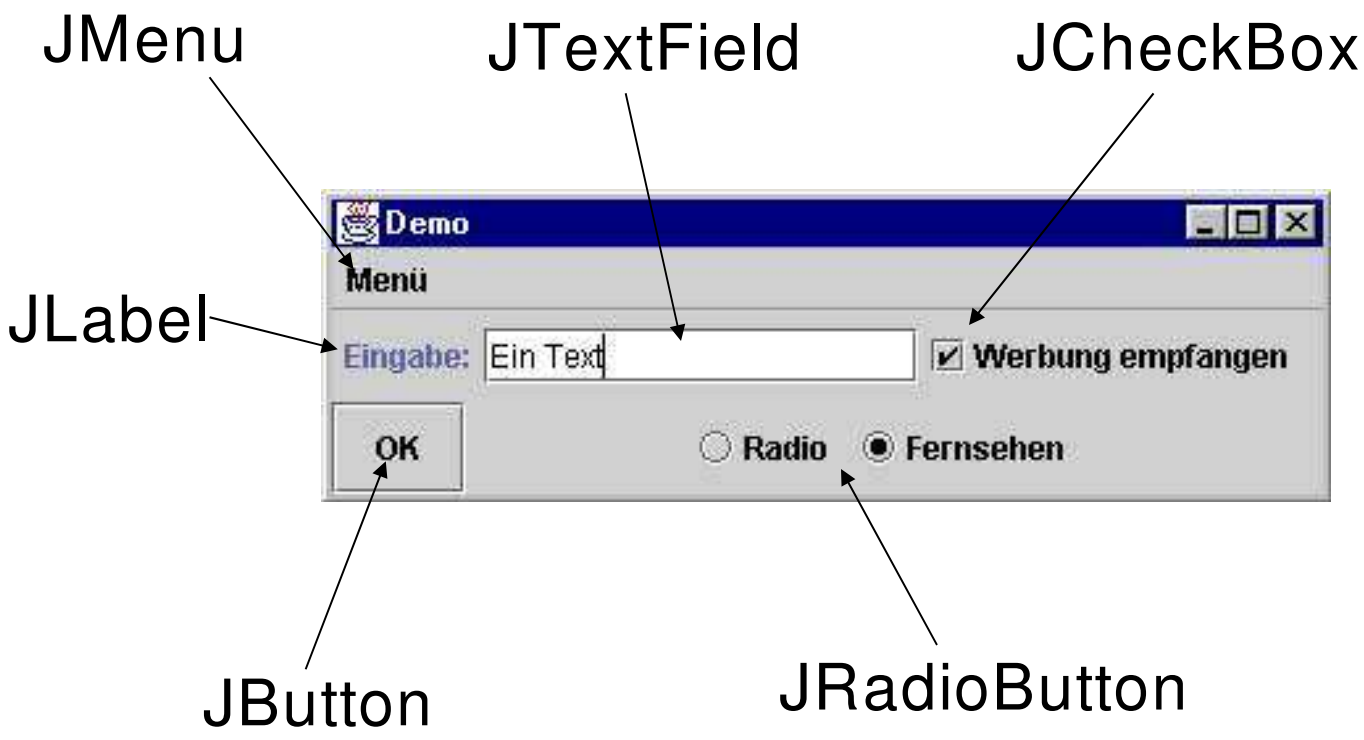
© 2001 Thomas Weiler

JAVA - Swing

- Bibliothek zur Erstellung grafischer Benutzerschnittstellen in JAVA
- Bietet Klassen für grafische Komponenten und zur Ergeignisbehandlung
- Weiterentwicklung des JAVA-AWT (Abstract Windowing Toolkit)
- Organisiert in der Paketstruktur javax.swing.*
- Beispielprogramme basieren auf JDK 1.4.2

© 2001 Thomas Weiler

Komponenten



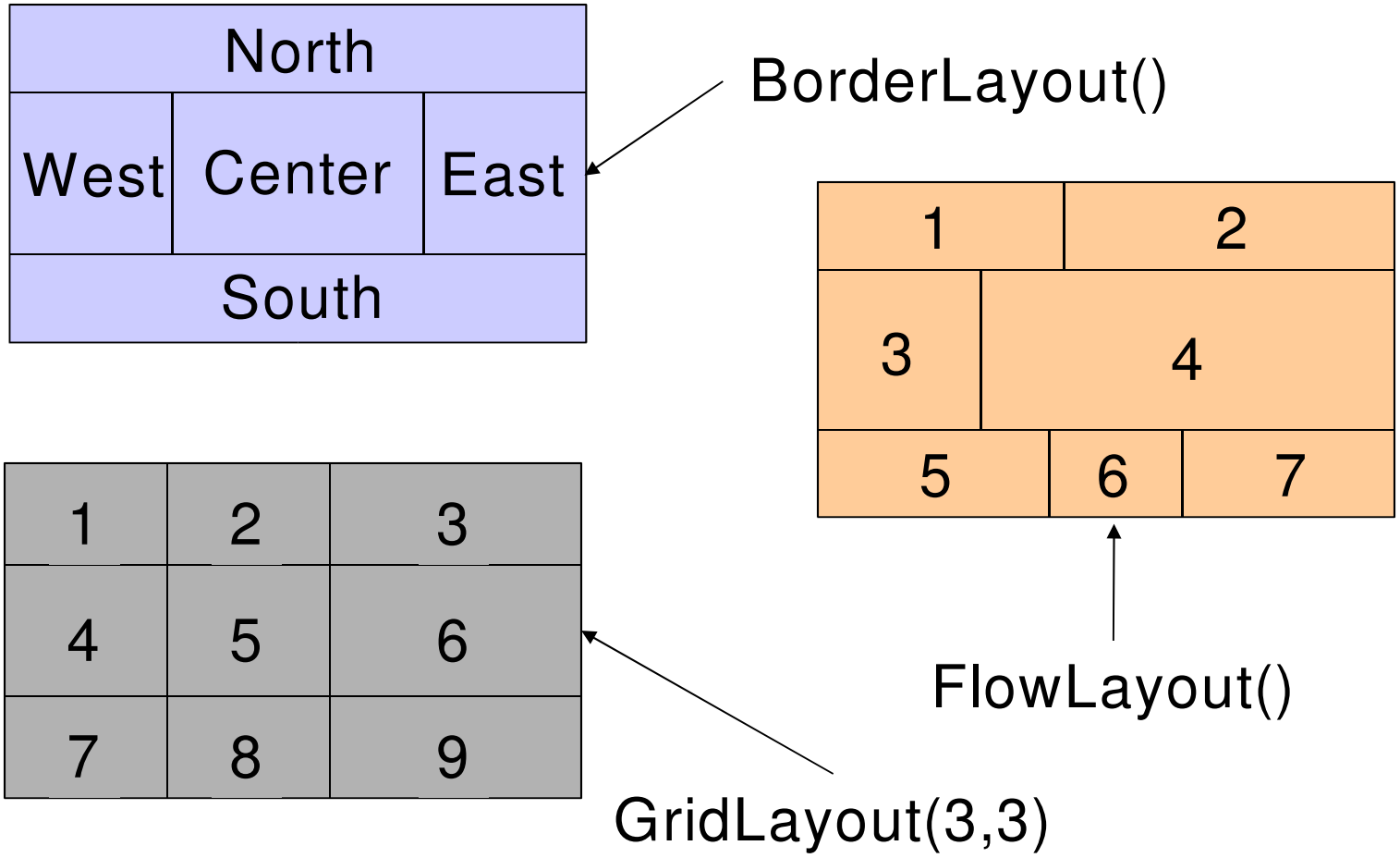
© 2001 Thomas Weiler

Beispielprogramm

- Benutzer kann einen Text eingeben
- Bei Auswahl von OK wird die Eingabe in einem Dialog angezeigt
- Bei Auswahl von Ende wird Programm nach Abfrage beendet
- Diese Aktionen lassen sich auch über Menü aufrufen
- Bei Anklicken eines Bildes ändert es sein Aussehen

© 2001 Thomas Weiler

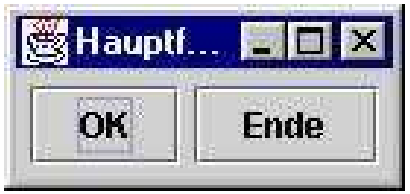
Layout



© 2001 Thomas Weiler

Quelltext - JButton

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

public class Hauptfenster extends JFrame {

    public Hauptfenster(String title) {
      super(title);

      JButton okKnopf = new JButton("OK");
      JButton endeKnopf = new JButton("Ende");
      JPanel sueden = new JPanel();
      sueden.setLayout(new FlowLayout());
      sueden.add(okKnopf);
      sueden.add(endeKnopf);
      getContentPane().add(sueden, BorderLayout.SOUTH);
    }

    public static void main(String args[]) {
      Hauptfenster hauptfenster = new Hauptfenster("Hauptfenster");
      hauptfenster.pack();
      hauptfenster.show();
    }
}
```



© 2001 Thomas Weiler

Quelltext – JLabel und JTextField

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

public class Hauptfenster extends JFrame {
    private JTextField textfeld;

    public Hauptfenster(String title) {
        [...]
        textfeld = new JTextField(15);
        JPanel norden = new JPanel();
        norden.setLayout(new FlowLayout());
        norden.add(new JLabel("Eingabe:"));
        norden.add(textfeld);
        getContentPane().add(norden, BorderLayout.NORTH);
    }

    [...]

}
```

© 2001 Thomas Weiler

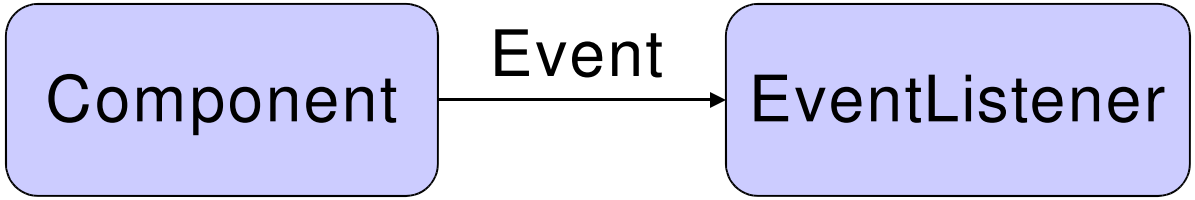
Quelltext - JMenu

```
public class Hauptfenster extends JFrame {
    private JTextField textfeld;

    public Hauptfenster(String title) {
        [...]
        JMenu menue = new JMenu("Menü");
        JMenuItem ersterEintrag = new JMenuItem("Eingabe anzeigen...");
        JMenuItem zweiterEintrag = new JMenuItem("Programm beenden...");
        menue.add(ersterEintrag);
        menue.add(zweiterEintrag);

        JMenuBar menueLeiste = new JMenuBar();
        menueLeiste.add(menue);
        setJMenuBar(menueLeiste);
    }

    [...]

}
```

© 2001 Thomas Weiler

Event Handling

- Komponente löst ein Event aus
- EventListener bearbeitet Event



- Verschiedene Arten von Listeners
  - ActionListener
  - WindowListener
  - MouseListener
  - FocusListener

© 2001 Thomas Weiler

Quelltext - ActionListener

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

public class Hauptfenster extends JFrame implements ActionListener {
    public Hauptfenster(String title) {
        [...]
        okKnopf.setActionCommand("ok");
        okKnopf.addActionListener(this);
        endeKnopf.setActionCommand("ende");
        endeKnopf.addActionListener(this);

        ersterEintrag.setActionCommand("menue1");
        ersterEintrag.addActionListener(this);
        zweiterEintrag.setActionCommand("menue2");
        zweiterEintrag.addActionListener(this);
    }

    public void actionPerformed(ActionEvent e) {
    }

    [...]
}
```

© 2001 Thomas Weiler

Quelltext - actionPerformed

```
public void actionPerformed(ActionEvent e) {
        if (e.getActionCommand().equals("ok")) {
            zeigeEintrag();
        }
        if (e.getActionCommand().equals("ende")) {
            beendeProgramm();
        }
        if (e.getActionCommand().equals("menue1")) {
            zeigeEintrag();
        }
        if (e.getActionCommand().equals("menue2")) {
            beendeProgramm();
        }
}
```

© 2001 Thomas Weiler

Quelltext - Hilfsprozeduren

```
public void zeigeEintrag() {
    JOptionPane.showMessageDialog(this,
            "Ihre Eingabe war: "+textfeld.getText(),
            "Information",
            JOptionPane.INFORMATION_MESSAGE);
}

public void beendeProgramm() {
    if (JOptionPane.showConfirmDialog(this,
                        "Möchten Sie wirklich aufhören",
                        "Frage",
                        JOptionPane.YES_NO_OPTION,
                        JOptionPane.QUESTION_MESSAGE)
        == JOptionPane.YES_OPTION) {
            System.exit(0);
    }
}
```

© 2001 Thomas Weiler

EventListener vs. EventAdapter

- EventListener ist Interface
  - Implementierende Klasse muß alle Methoden implementieren

2. EventAdapter ist abstrakte Klasse
   - Unterklasse redefiniert nur die notwendigen Methoden
   - Adapter gibt es nicht für alle Events

© 2001 Thomas Weiler

java.awt.event.WindowListener

```
public interface WindowListener {
    public void windowClosing(WindowEvent e);
    public void windowClosed(WindowEvent e);
    public void windowOpened(WindowEvent e);
    public void windowIconified(WindowEvent e);
    public void windowDeiconified(WindowEvent e);
    public void windowActivated(WindowEvent e);
    public void windowDeactivated(WindowEvent e);
}
```

© 2001 Thomas Weiler

java.awt.event.WindowAdapter

```
public abstract class WindowAdapter {
    public void windowClosing(WindowEvent e) {
    }
    public void windowClosed(WindowEvent e) {
    }
    public void windowOpened(WindowEvent e) {
    }
    public void windowIconified(WindowEvent e) {
    }
    public void windowDeiconified(WindowEvent e) {
    }
    public void windowActivated(WindowEvent e) {
    }
    public void windowDeactivated(WindowEvent e) {
    }
}
```

© 2001 Thomas Weiler

Eigener WindowAdapter

```
import java.awt.event.WindowAdapter;

public class MyWindowAdapter extends WindowAdapter {
    public void windowClosing(WindowEvent e) {
        System.exit(0);
    }
}
```

```
addWindowListener(new MyWindowAdapter());
```

© 2001 Thomas Weiler

Anonyme Klasse

- Anonyme Klasse ist Klasse ohne Name
- Wird innerhalb einer anderen Klasse definiert und verwendet

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

© 2001 Thomas Weiler

Quelltext - WindowAdapter

```
public class Hauptfenster extends JFrame implements ActionListener {
    [...]
    public Hauptfenster(String title) {
        [...]
      addWindowListener(new WindowAdapter() {
                    public void windowClosing(WindowEvent e) {
                                    System.exit(0);
                    }
                });
    } // Konstruktor

    [...]

} // Klasse
```

Anonyme Klasse

© 2001 Thomas Weiler

Quelltext – Bild (1)

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

public class Bild extends JPanel {
    private Image grafik1 = Toolkit.getDefaultToolkit().getImage("T1.gif");
    private Image grafik2 = Toolkit.getDefaultToolkit().getImage("T2.gif");
    private Image grafikAktuell = grafik1;

    public Bild() {
    }

    public Dimension getPreferredSize() {
            return new Dimension(60,80);
    }

}
```

© 2001 Thomas Weiler

Quelltext – Bild (2)

```
public class Bild extends JPanel {

    [...]

    public void paintComponent(Graphics g) {
        super.paintComponent(g); //Hintergrund zeichnen
        g.drawRect(0,0, 59,79);
        g.drawImage(grafikAktuell, 0, 0, this);
    }

}
```

© 2001 Thomas Weiler

Quelltext – Bild (3)

```
public class Bild extends JPanel {
    [...]

    public Bild() {
        addMouseListener(new MouseAdapter() {
            public void mouseClicked(MouseEvent e) {
                if (grafikAktuell == grafik1) {
                    grafikAktuell = grafik2;
                } else {
                    grafikAktuell = grafik1;
                }
                repaint();
            }
        });
    }

    [...]
}
```

© 2001 Thomas Weiler

Quelltext - Hauptfenster

```
public class Hauptfenster extends JFrame implements ActionListener {
    [...]

    public Hauptfenster(String title) {
        [..]
        getContentPane().add(new Bild(), BorderLayout.EAST);
        getContentPane().add(new Bild(), BorderLayout.WEST);
    }
}
```

© 2001 Thomas Weiler

Weitere Möglichkeiten von Swing

- Applets
- Native language support
- Look & Feel
- Tooltips
- Drag & Drop

© 2001 Thomas Weiler

Dokumentation

- Folien dieses Vortrags und Beispielprogramme unter

  http://programmierung.informatik.rwth-aachen.de

  unter Globalübung
- JDK 1.4.2 API Dokumentation

  http://java.sun.com/j2se/1.4.2/docs/api/index.html
- JAVA-Tutorial von SUN

  http://java.sun.com/docs/books/tutorial/index.html

© 2001 Thomas Weiler

Fragen ?

© 2001 Thomas Weiler