

Programmiermethodik

Übung 6

Sommersemester 2011
Fachgebiet Software Engineering

Andreas Koch
andreas.koch@cs.uni-kassel.de

Agenda

- **Entwicklung von grafischen Oberflächen**
 - Mock-Ups
 - GUI Builder
 - Swing
 - SWT
 - Demo
- **Praktische Übung: Erstellen eines Wizard Login Screens**

Entwicklung von grafischen Oberflächen – Mock-Ups

- Bevor man anfängt zu coden: „Mock-Ups“
- In der Realität: Designer != Entwickler
- Erstellung von Mock-Ups kostet deutlich weniger Zeit
- Erleichtert die Implementierung, weil man schon weiß was herauskommen soll



Quelle: <http://media.konigi.com/notebook/iphone-mockup.jpg>

Entwicklung von grafischen Oberflächen – GUI Builder I

- **Problem bei grafischen Oberflächen: Aufwendig in der Entwicklung**
- **Abhilfe sollen GUI-Builder schaffen (WYSIWYG Prinzip)**
- **Kommerzielle Tools (SWT/Swing):**
 - Jigloo
 - ...
- **Freie Tools (SWT/Swing):**
 - Window Builder Pro
 - Visual Editor (Eclipse 3.2, veraltet, wird nicht mehr gepflegt)
- **Generieren (meist hässlichen) SWT/Swing Code**

Entwicklung von grafischen Oberflächen – GUI Builder II

The screenshot displays the Eclipse IDE interface with several GUI development tools overlaid:

- Jigloo:** The top-most window, titled "Java - testApps/src/apps/NewSingleFrameApplication.java - Eclipse SDK", showing a menu bar and toolbar.
- Visual Editor:** The middle window, titled "Java - SimpleSWTBrowser.java - Eclipse SDK - /Projekte/Java", showing a browser-like GUI with buttons and a URL field.
- Window Builder Pro:** The bottom-most window, titled "Layout Assistant", showing a visual design area with a grid and several "New JButton" components. It includes a "Swing Controls" palette on the left and a "Layout Assistant" dialog with "Anchors" and "Insets" settings.

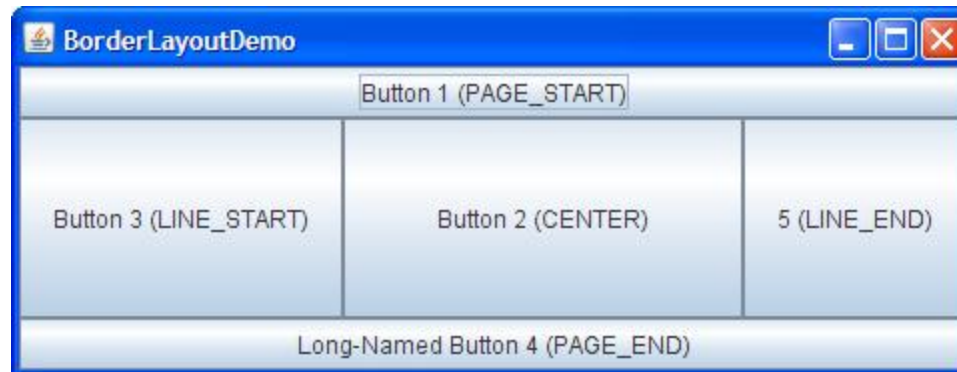
Red arrows point from the labels on the right to the corresponding windows in the screenshot.

Entwicklung von grafischen Oberflächen – Swing I

- **Swing ist seit Java 1.2 Bestandteil der Java Runtime**
- **Baut auf dem Abstract Window Toolkit (AWT) auf**
- **Swing Komponenten werden direkt von Java gerendert**
 - Funktioniert auf allen Plattformen
 - Sieht überall gleich aus
 - NICHT nativ
- **Verschiedene Look&Feels**
 - Windows
 - Linux
 - Mac
 - ...

Entwicklung von grafischen Oberflächen – Swing II

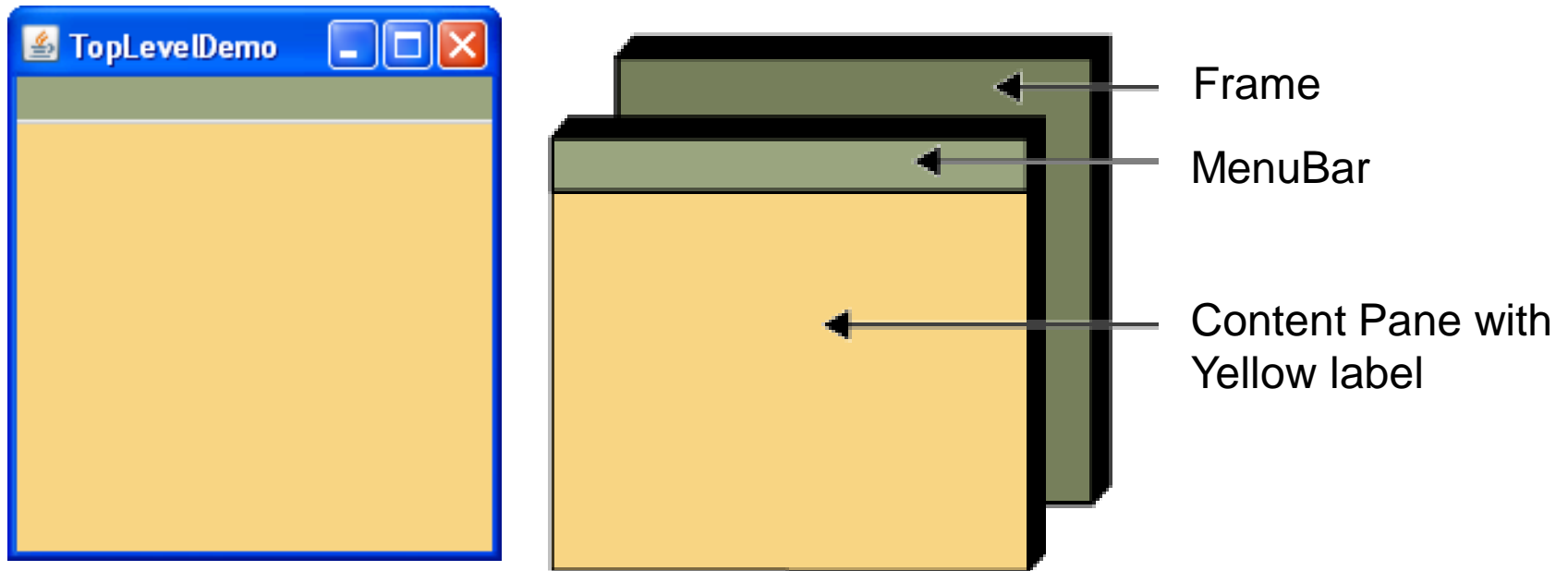
- Referenz: <http://java.sun.com/docs/books/tutorial/uiswing/>
- Einstieg: JFrame (Top-Level Container)
 - Repräsentiert ein Fenster
 - Kann Inhalt aufnehmen
 - Standard Layoutmanager: BorderLayout



- `jFrame.getContentPane()` liefert Container

Entwicklung von grafischen Oberflächen – Swing III

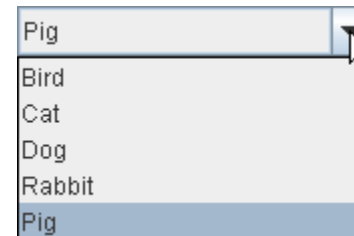
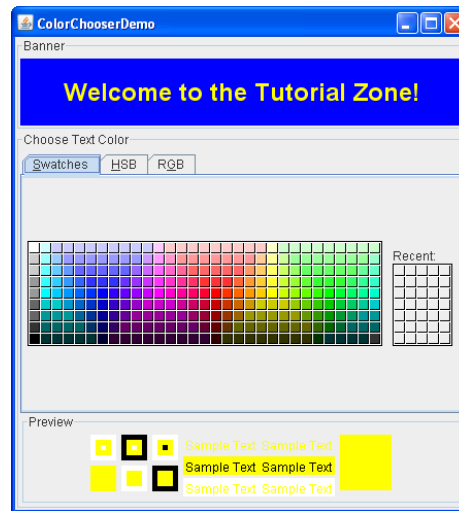
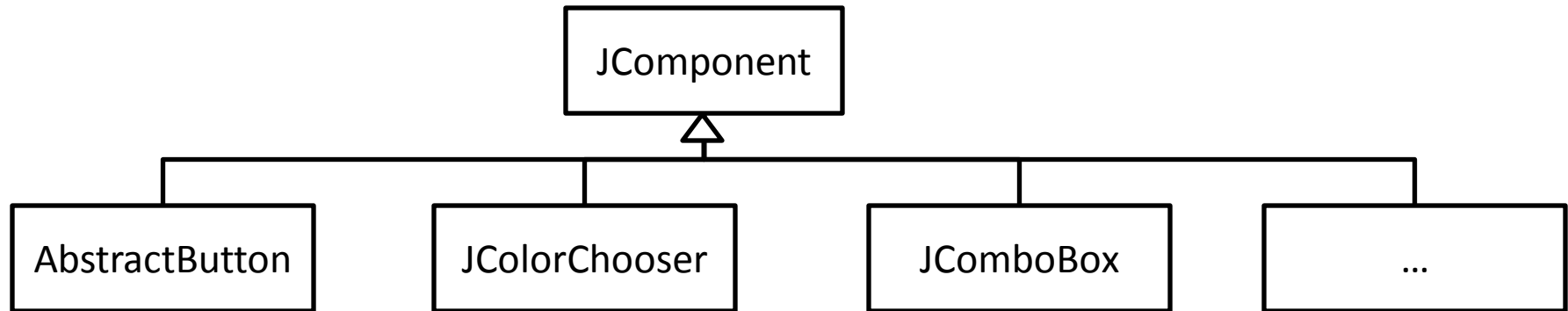
- Aufbau eines JFrame



- `jFrame.getContentPane().add(yellowLabel, BorderLayout.CENTER)`

Entwicklung von grafischen Oberflächen – Swing IV

- Alles (bis auf die Top-Level Container) leitet von JComponent ab

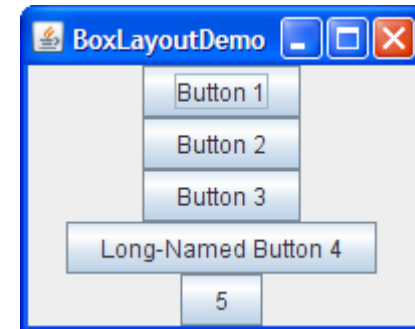
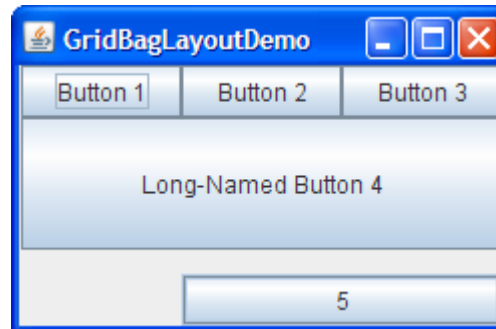


Entwicklung von grafischen Oberflächen – Swing V

- **JComponent stellt alle grundlegenden Features zur Verfügung**
 - Tooltips
 - Rahmen
 - Look&Feel
 - Layout
 - DnD
 - ...

Entwicklung von grafischen Oberflächen – Swing VI

- **Absolute Positionierung vs. LayoutManager**
- **LayoutManager**
 - Positionierung
 - Verhalten bei Größenänderung des Fensters
- **Immer einen LayoutManager benutzen!**
- **Beispiele**
 - BorderLayout
 - BoxLayout
 - GridBagLayout



Entwicklung von grafischen Oberflächen – Swing VI

- **Verwendung von LayoutManagern**

- Layout wählen (<http://java.sun.com/docs/books/tutorial/uiswing/layout/visual.html>)
 - `BoxLayout boxLayout = new BoxLayout(jComponent, BoxLayout.PAGE_AXIS)`
- LayoutManager auf Container anwenden
 - `jComponent.setLayout (boxLayout)`
- Kinder hinzufügen
 - `jComponent.add(new Label(`

Entwicklung von grafischen Oberflächen – SWT

- **Standard Widget Toolkit (SWT)** <http://www.eclipse.org/swt>
- **Wurde 2001 von IBM für Eclipse entwickelt**
- **NICHT Bestandteil der Java Runtime**
 - Bibliothek (inkl. nativen Bestandteilen) müssen mit ausgeliefert werden
- **Abstrahiert von nativer grafischer Benutzeroberfläche**
 - Einmal coden, überall nativ laufen lassen (theoretisch)
- **(Unter Windows) deutlich schneller als Swing**
- **Im Gegensatz zu Swing „schwergewichtig“, wegen der Verwendung von nativen Komponenten (statt sie selbst zu zeichnen)**

SWT – Tutorials und hilfreiche Links

- **SWT**

- <http://www.eclipse.org/articles/article.php?file=Article-Understanding-Layouts/index.html>
- <http://www.eclipse.org/swt/widgets/>
- <http://zetcode.com/tutorials/javaswttutorial/>
- <http://www.vogella.de/articles/SWT/article.html>

- **WindowBuilder Pro**

- **Update Site:** <http://dl.google.com/eclipse/inst/d2wbpro/latest/3.6>
- <http://code.google.com/javadevtools/wbpro/index.html>

Entwicklung von grafischen Oberflächen

- Demo: „Celsius – Fahrenheit converter“
- Umrechnungsformel:
 - Celsius in Fahrenheit = $((T_{\text{Celsius}} \times 9) / 5) + 32$
 - Fahrenheit in Celsius = $(T_{\text{Fahrenheit}} - 32) \times 5 / 9$

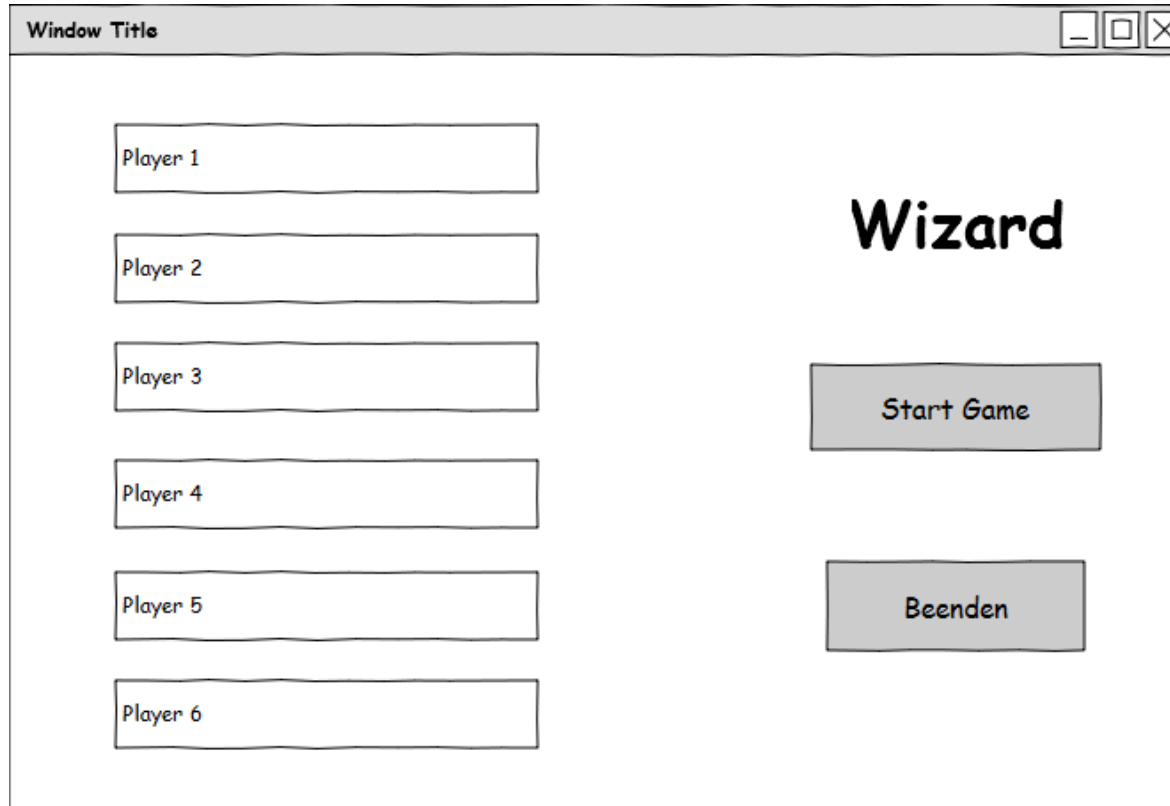
- Mock-Up

The mock-up shows a window titled "Temperature Converter" with standard window controls (minimize, maximize, close). The interface includes:

- A label "Value:" followed by a text input field containing the text "value to convert".
- A horizontal container with two radio buttons:
 - The first radio button is selected and labeled "Convert to Fahrenheit".
 - The second radio button is unselected and labeled "Convert to Celsius".
- A label "Converted:" followed by a text input field containing the text "converted value".

Praktische Übung

- Entwicklung eines Wizard Login Screens
- Mock-Up



The image shows a mock-up of a wizard login screen within a window titled "Window Title". The window has standard minimize, maximize, and close buttons in the top right corner. On the left side, there are six text input fields, each containing the text "Player 1" through "Player 6" respectively. On the right side, the word "Wizard" is displayed in a large, bold font. Below the title, there are two buttons: "Start Game" and "Beenden".

- Zeigt das Ergebnis einem Betreuer

Praktische Übung - Alternative

- Entwicklung eines erweiterten Wizard Login Screens
- Mock-Up

The mock-up shows a window titled "Window Title" with standard window controls (minimize, maximize, close). The main content area is divided into two sections. On the left, there are three text input fields, each containing the text "Player 1", "Player 2", and "Player 3" respectively. Below these fields is a button labeled "Add New Player". On the right, the word "Wizard" is displayed in a large, bold font. Below "Wizard" are two buttons: "Start Game" and "Beenden".

Ende

Schönes WE!