

Graphische Benutzungsoberflächen mit Java

Einführung in NetBeans

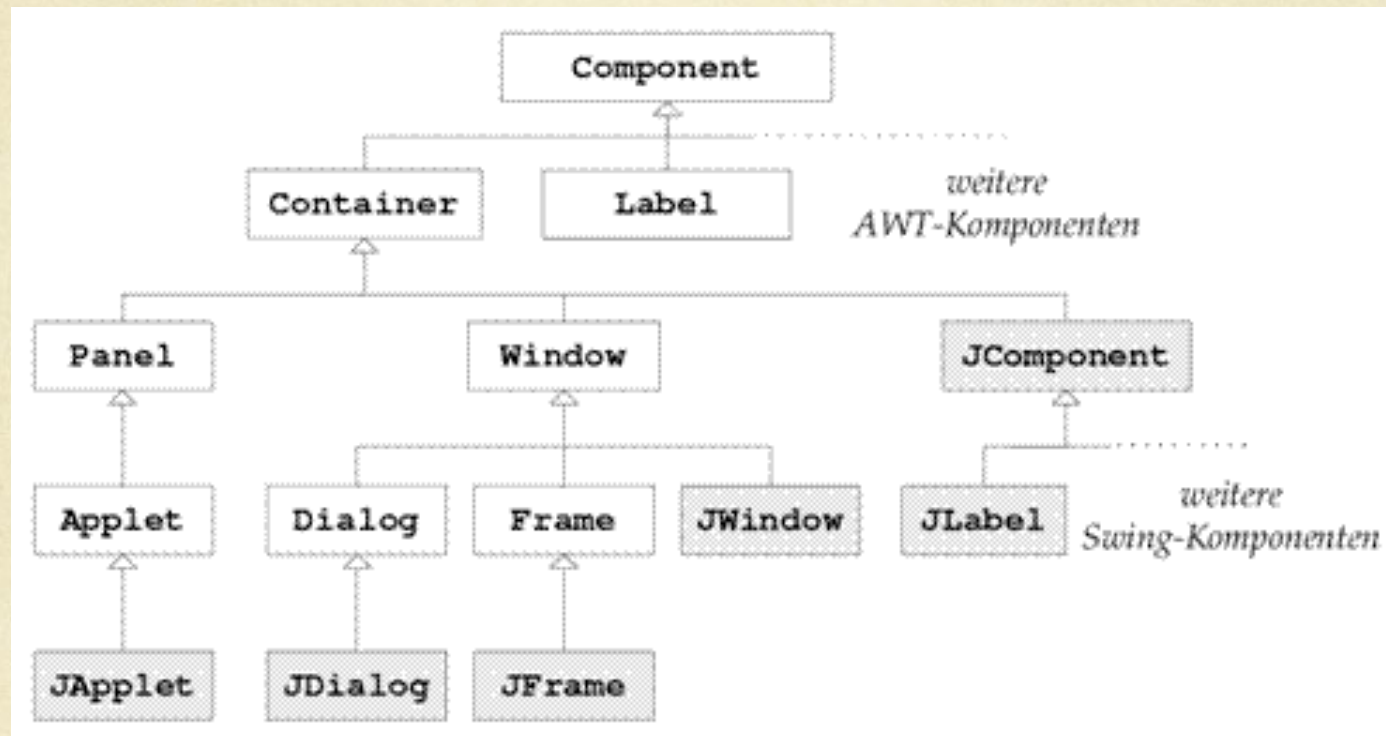
Graphische Benutzungsoberflächen

- Interaktion mit dem Programm
- Datenein- und -ausgabe
- Rückmeldungen über Verarbeitung der Daten
- Visualisierung von Daten
- optisch ansprechende Gestaltung der Benutzungsschnittstelle

Graphische Benutzungsoberflächen in Java

- Rückgriff auf Betriebssystem
- umfangreiche Klassenbibliotheken verfügbar, AWT und Swing
- Erzeugung von Oberflächen, indem Komponenten auf dem Bildschirm arrangiert werden

AWT- und Swing-Klassenhierarchie



Quelle: RATZ, Grundkurs Programmieren in Java, HANSER Verlag 2010

Java-GUI-Basisklassen

Grundkomponenten

einfache
Oberflächen-
Elemente, z.B.:

- Beschriftungen,
- Knöpfe,
- Auswahlfelder,
- Klapptafeln

Container

spezielle
Komponenten
, die selbst
wieder
Komponenten
enthalten
können

Layout- Manager, Farben, Fonts

Klassen,
deren Objekte
für die
Anordnung
und
Gestaltung der
einzelnen
Komponenten
zuständig sind

Ereignisse, Listener

Klassen für die
Erzeugung und
Verarbeitung von
Ereignissen, die
für die Interaktion
der Komponenten
mit dem Anwender
benötigt werden

Basisklasse JFrame

- JFrame ist Basisklasse für Fenster in Java (Swing-Komponente)
- abgeleitet von Klasse Frame (AWT-Komponente)
- darauf können Komponenten platziert werden
- zur Reaktion auf Ereignisse

NetBeans

- Projekt erstellen, das verschiedene Klassen enthalten kann
- verschiedene Arten von Projekten: Java, Java Web, BlueJ (NetBeans 6.9, nicht mehr in NetBeans 7)
- jedes Projekt braucht eine Main Class (Klasse, die eine main-Routine enthält) die beim Starten des Projektes aufgerufen wird
- für Projekte mit graphischer Oberfläche sollte das eine von JFrame abgeleitete Klasse sein

NetBeans – erste Schritte

- neues Projekt erstellen – hier „set main class“ deaktivieren!
- im Projekt neue Klasse erstellen (new File → Swing GUI Forms → JFrame Form)
- beim ersten Start des Projektes wird die main class ausgewählt
- Wechsel zwischen Quelltextansicht und Designansicht bei Programmerstellung möglich (für GUI-Klassen)

NetBeans – erste Schritte

- Quelltextansicht zeigt automatisch generierten Quelltext: Klassendeklaration, Konstruktor, main-Methode
- weiterer generierter Quelltext ist versteckt und kann durch Klick auf „+“ angezeigt werden
- grau hinterlegter Quelltext kann nicht editiert werden, Bsp.: Hinzufügen eines Buttons mit der Maus fügt nicht editierbaren Quelltext hinzu

NetBeans – erste Schritte

- Komponenten hinzufügen in Designansicht aus der Palette
- Konfiguration der Komponenten im Fenster „Properties“ oder über rechten Mausklick auf die Komponente
- Verknüpfung mit Ereignissen und Aktionen: Auswahl des Ereignisses der Komponente erzeugt automatisch ActionListener und Methodenrumpf, der dann um die gewünschten Befehle ergänzt werden muss

Ereignisverarbeitung

- beteiligt: Quelle, Ereignis, Empfänger
- Ereignisquelle: Komponenten der graphischen Oberfläche
- Ereignis: Aktion, das durch Benutzer beim Arbeiten mit graphischer Oberfläche ausgelöst wird
- Ereignisempfänger (event listener): Objekte, die auf Ereignisse „lauschen“ und reagieren
- Ereignisempfänger müssen bei der Ereignisquelle registriert werden – Delegation Event Model (ermöglicht Trennung von Oberfläche und eigentlicher Ereignisverwaltung)

Ereignisverarbeitung

- Erzeugen eines Oberflächen-Projektes, hinzufügen einer JFrame-Klasse
- Hinzufügen eines Buttons
- Quelltext enthält Konstruktor für Oberfläche, der Fenster erstellt und Button einbaut und eine main-Methode, die ein Objekt der Oberflächenklasse erzeugt

Ereignisverarbeitung

- Ereignisquelle: Button
- Ereignisempfänger: muss erzeugt und beim Button registriert werden
- auszuführendes Ereignis muss programmiert werden
- in NetBeans: Rechtsklick auf Ereignisquelle, dann Event → action → actionPerformed
- im Quelltext: in `initComponents ()` wurde zum Button ein `ActionListener ()` als anonyme Klasse hinzugefügt und ein Methodenrahmen zur Programmierung des Ereignisses wurde erstellt

Ereignisverarbeitung

Generierter Quelltext für Button mit Namen „Farbwechselknopf“ mit eingefügtem Befehl zum Ändern der Hintergrundfarbe:

```
private void FarbwechselknopfActionPerformed(java.awt.event.ActionEvent
    evt) {

    // TODO add your handling code here:

        float zufall1 = (float) Math.random();

        float zufall2 = (float) Math.random();

        float zufall3 = (float) Math.random();

        Color farbe = new Color (zufall1, zufall2, zufall3);

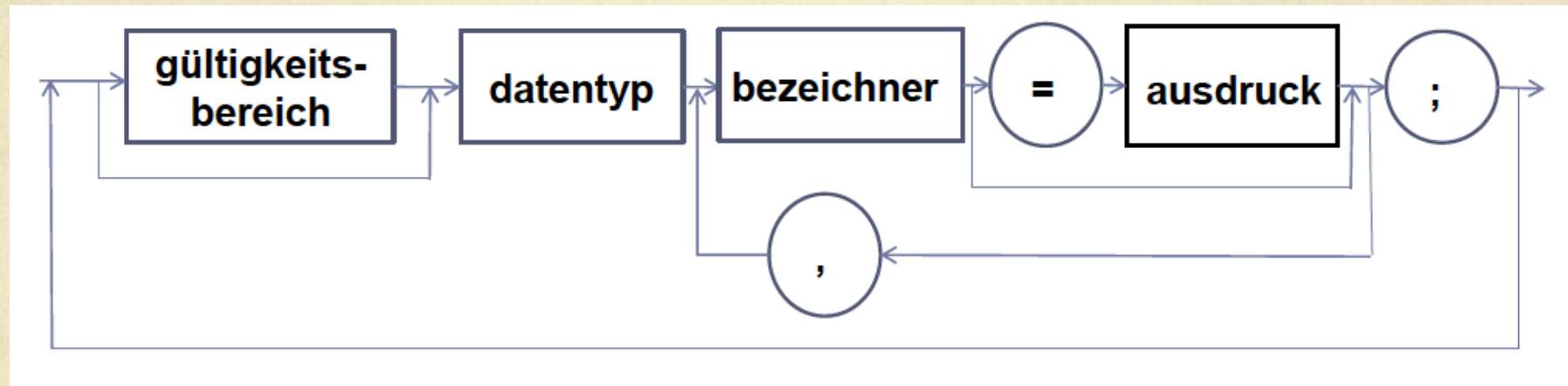
        setBackground(farbe);

}
```

Datentypen

- logischer Typ: boolean
- Ganzzahltypen: int, byte, short, und long
- Gleitpunkttypen: float und double
- Zeichentyp: char
- Zeichenketten sind in Java Objekte und keine Felder von char, sie sind nicht nullterminiert und besitzen auch kein spezielles Byte für die Länge

Deklaration



Beispiele:

```
private int x;  
private int y = 12;  
public double wert;  
private String name;
```


Java Grundlagen

- Unterscheidung zwischen Groß- und Kleinschreibung
- objektorientiert, Klasse erstellen und Methoden für die Klasse schreiben
- Klassen können Instanzvariablen besitzen (Gültigkeit global für die Klasse, evtl. darüber hinaus)
- Konstruktoren sind spezielle Methoden, die bei der Instanziierung von Objekten der Klasse aufgerufen werden, dort können Instanzvariablen initialisiert werden

Java Grundlagen

- Fokus Algorithmierung: Klasse dient nur als Rahmen für die Methoden, Methoden enthalten die Algorithmen
- Fokus Objektorientierung: mehrere Klassen entwerfen und Interaktion zwischen

NetBeans Features

- automatisch generierter Quelltext
- Autovervollständigung beim Tippen von Befehlen
- Syntaxfehler werden sofort am Zeilenanfang durch rotes ! gekennzeichnet – Mausbewegung auf Symbol zeigt Fehlermeldung an
- wenn verfügbar, werden beim Klick auf das Symbol Lösungsvorschläge angeboten, wie z.B. das hinzufügen der erforderlichen Pakete beim Verwenden spezieller Klassen

Quellen und weiterführende Informationen

- Ratz, D.; Grundkurs Programmieren in Java, HANSER-Verlag
- Kempe, Tepaße (Hrsg.); Softwareentwicklung mit Greenfoot und BlueJ, Schöningh-Verlag
- Java Handbuch in deutsch
<http://www.dpunkt.de/java/index.html>