



OOP mit Java und BlueJ im Informatikunterricht

Lehrer-Fortbildung

06.03.2006

Ralf Punkenburg (Menzel-Oberschule)

Alexander Dietz (Humboldt-Oberschule)

Mathias Müller (Camille-Claudel Oberschule)

MODELLIEREN und PROGRAMMIEREN

Verschiedene Wege zum Ziel

- Algorithmik \rightarrow OOA \rightarrow OOP
- OOA \rightarrow Algorithmik \rightarrow OOP
- (OOA \leftrightarrow OOP \leftrightarrow Algorithmik) von Anfang an



Übersicht

1. Was ist BlueJ?
2. Probleme mit Java und BlueJ's „Lösungen“
3. Probleme mit Entwicklungsumgebungen und BlueJ's „Lösungen“
4. Tagesplanung (Beispiele und Konzepte)
5. Probleme mit BlueJ



1. Was ist BlueJ?

- integrierte Java-Umgebung, die speziell zu Lehrzwecken entworfen wurde
- Ziel: „Objects first!“ -- Schüler sollen ohne Ballast „in Klassen und Objekten denken“
→ Objekte werden „greifbar“
- plattformunabhängig und kostenlos
- entwickelt an der Monash University, AUS
von Barnes/Kölling

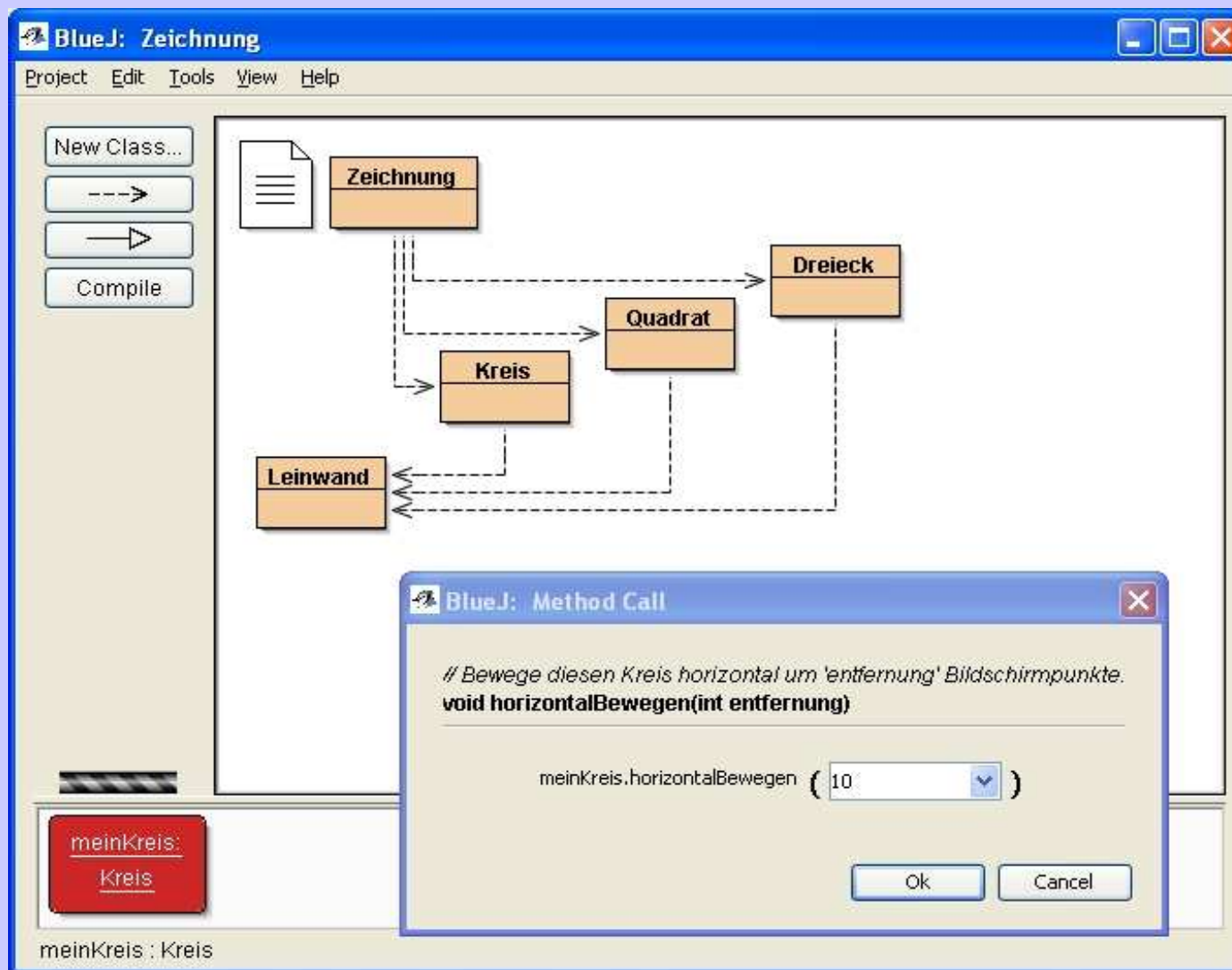


2. Probleme mit Java und *BlueJ's „Lösungen“*

```
- public static void main (String[]  
  args) {  
    Kreis meinKreis = new Kreis(50,11,5);  
    meinKreis.horizontaleBewegen(10);...
```

+ BlueJ:

Objekte interaktiv erzeugen und über Methoden direkt
manipulieren



The screenshot shows the BlueJ IDE interface. The main window, titled "BlueJ: Zeichnung", contains a class diagram with the following classes and relationships:

- Zeichnung**: The root class, with dashed arrows pointing to **Kreis**, **Quadrat**, and **Dreieck**.
- Kreis**: Has a dashed arrow pointing to **Quadrat** and another pointing to **Leinwand**.
- Quadrat**: Has a dashed arrow pointing to **Leinwand**.
- Dreieck**: Has a dashed arrow pointing to **Leinwand**.
- Leinwand**: The canvas class, which is the target of all drawing actions.

On the left side of the IDE, there are buttons for "New Class...", "Compile", and a "meinkreis: Kreis" button. The status bar at the bottom left shows "meinkreis: Kreis".

A "BlueJ: Method Call" dialog box is open in the foreground, displaying the following code:

```
// Bewege diesen Kreis horizontal um 'entfernung' Bildschirmpunkte.  
void horizontalBewegen(int entfernung)
```

The dialog shows the method call `meinkreis.horizontalBewegen (10)` with a dropdown menu set to "10". The "Ok" and "Cancel" buttons are visible at the bottom right of the dialog.

- Ein-/Ausgabe

```
import java.io.*;
class Fahrsccheinautomat
{
public static void main(String[] args)
throws IOException
{
int a, b, c;
BufferedReader din = new BufferedReader(
new InputStreamReader(System.in));
System.out.println("Bitte Ticketpreis in Cent eingeben: ");
a = Integer.parseInt(din.readLine());
System.out.println("Bitte Geldbetrag in Cent einwerfen: ");
b = Integer.parseInt(din.readLine());
c = a - b;
System.out.println("Restbetrag: "+c);
```

+ BlueJ: Parameter durch interaktive Methodenaufrufe direkt ein- und ausgegeben.



The screenshot shows the BlueJ IDE interface. The main window is titled "BlueJ: FSA1" and contains a class diagram with a class named "Fahrsccheinautomat". A dialog box titled "BlueJ: Methodenaufruf" is open, showing the method signature `void muenzEinwurf(int wert)` and a call site `fahrsche1.muenzEinwurf ([dropdown])`. The dialog also contains explanatory comments in German.

BlueJ: FSA1
Projekt Bearbeiten Werkzeuge Ansicht Hilfe

Neue Klasse...
--->
->
Übersetzen

Fahrsccheinautomat

BlueJ: Methodenaufruf

*// Methode zum Verarbeiten der eingeworfenen Münzen (in Cent);
// nur wenn ein Fahrscchein gewählt worden ist, werden Münzeinwürfe akz...
// wenn ausreichend Münzen eingeworfen worden sind, wird der Vorgang ...
// @param wert Wert der eingeworfenen Münze*

void muenzEinwurf(int wert)

fahrsche1.muenzEinwurf ([dropdown])

Ok Abbrechen

fahrsche1:
Fahrsccheinaut



3. Probleme mit Entwicklungsumgebungen

- zu komplex
- nicht objektorientiert
- GUI-orientiert
- kosten viel Geld
- Arbeiten ohne
Entwicklungs-
umgebung?

```
Java - Screen.java - Eclipse SDK
File Edit Source Refactor Navigate Search Project Run Window Help
Screen.java
import javax.microedition.lcdui.*;

/**
 * Die Screen-Klasse stellt die grafische Oberfläche dar
 * und sorgt für das Game-Rendering
 * Darstellbar sind alle Klassen die von Displayable erben,
 * also Canvas, Alert, List, TextBox und Form
 */
@author LK-INF
@author (Last changed by: $eller$)
@version $Revision: 1.0 $
*/
public class Screen extends GameCanvas implements Runnable, Player

private Sprite auto, explosion;
private Vector explosions;
private TiledLayer landscape;
private LayerManager layerManager;
private boolean isRunning, showHelp, showHighscore, gameOver;
private boolean soundOn = false;
private AufDerFlucht midlet;
private int zaehler, width, height, zeilen, spalten, tileHeight;
private Image splashScreen, lkwbild, muenzebild, herzbild;
```



BlueJ ...

- ist objektorientiert
- ist für den Anfangsunterricht konzipiert
- visualisiert Klassenstrukturen und Objekte
- ermöglicht einen interaktiven Umgang mit Objekten (Erzeugen und Methodenaufrufe)
- erleichtert die Konzentration auf Fachklassen
- ist einfach zu installieren, bedienen und zu erlernen
- benötigt wenig Speicher und ist kostenlos

Beispiele und *Konzepte*

1. Figuren (*Klasse, Objekt, Methode, Attribute*)
2. Zeichnung (*Quelltext, Java-Syntax, Compilieren*)
3. Fahrscheinautomat (*Klassendefinition, Klassenbeziehungen, Objektkommunikation, Algorithmik, Dokumentation*)

Beispiele und Konzepte

4. Verwaltung von Medien (*Sammlung von
Objekten, Ausnahmebehandlung,
Vererbung*)



Probleme mit BlueJ

- Programmierung außerhalb von BlueJ (Schüler wollen/sollen „echte“ Java-Programme schreiben)
- OOM ist eingeschränkt (z. B. keine Unterscheidung zwischen Assoziation und Aggregation)
- es gibt keinen visuellen GUI-Builder
- BlueJ ist manchmal ganz schön langsam
- ...

