Objects first!
OOP mit Java und BlueJ

IBBB-Tagung zur Schulinformatik

08.03.2007

Ralf Punkenburg (Robert-Jungk-Oberschule Berlin)
Alexander Dietz (Humboldt-Oberschule Berlin)

Übersicht

1. Modellieren und Programmieren im Anfangsunterricht
2. Was ist BlueJ?
3. Probleme mit Java und BlueJ´s „Lösungen“
4. Probleme mit Entwicklungsumgebungen und BlueJ´s „Lösungen“
5. Beispiele und Konzepte im Workshop
6. Probleme mit BlueJ

1. MODELLIEREN und PROGRAMMIEREN

Verschiedene Wege führen zum Ziel

1. Algorithmik -> OOM -> OOP
2. OOM -> Algorithmik -> OOP
3. (OOM ↔ OOP ↔ Algorithmik) von Anfang an

Anfangsunterricht

- **These**: BlueJ von Anfang an ist zu komplex
- **Folgerung**: Anfangsunterricht mit einer (möglichst objektorientierten) Minisprache/Arbeitsumgebung, z. B. Greenfoot, Niki für Java, EOS, Java-Kara im Umfang von ca. 18 Stunden

2. 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

3. Probleme mit Java und *BlueJ´s „Lösungen“*

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

+ BlueJ:

  Objekte lassen sich interaktiv erzeugen und über Methoden direkt manipulieren

BlueJ: Zeichnung

Project Edit Tools View Help

New Class...

Compile

Zeichnung

Dreieck

Quadrat

Kreis

Leinwand

meinKreis: Kreis

meinKreis : Kreis

BlueJ: Method Call

// Bewege diesen Kreis horizontal um 'entfernung' Bildschirmpunkte.

void horizontalBewegen(int entfernung)

meinKreis.horizontalBewegen (10)

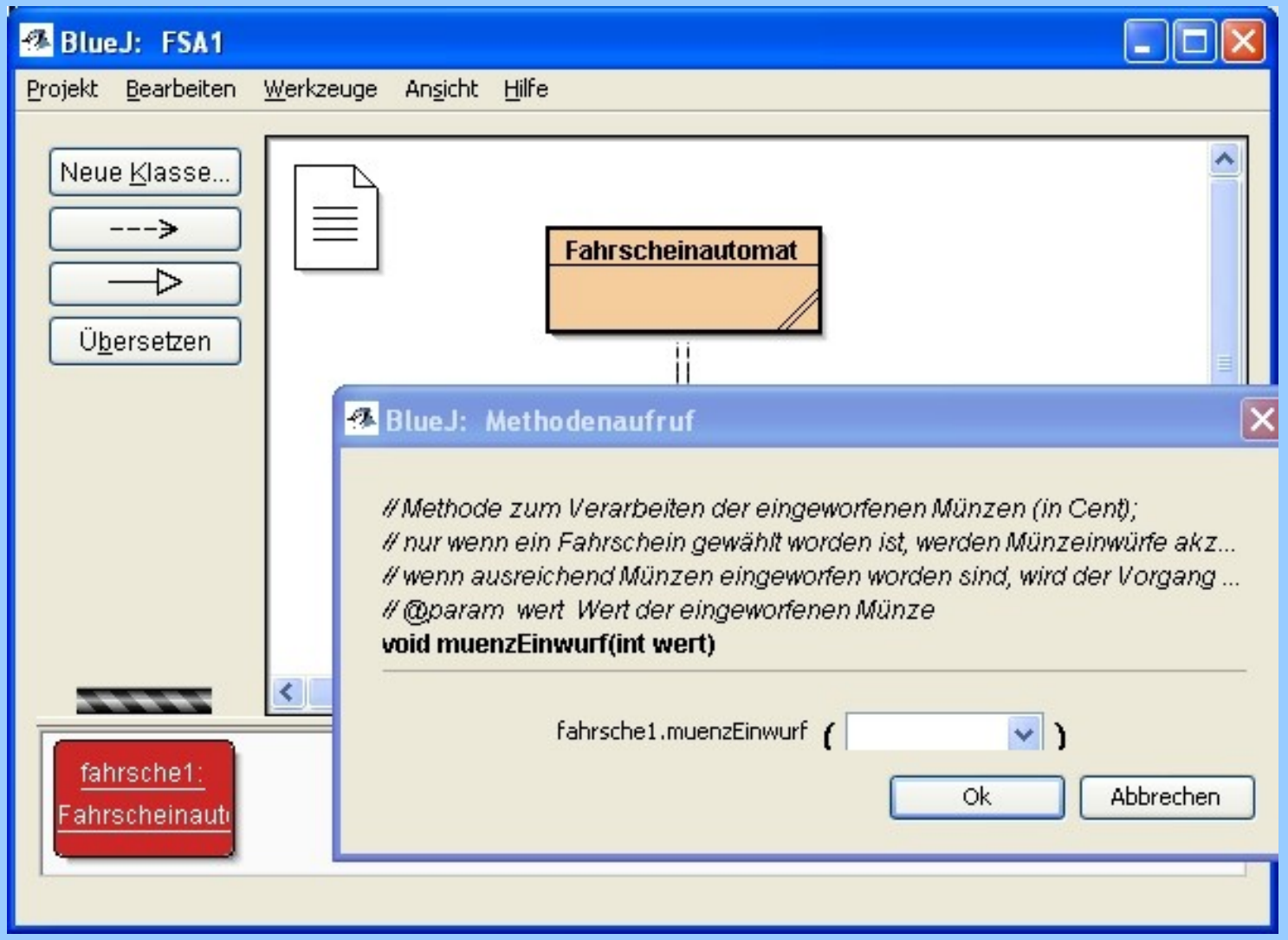
Ok Cancel

- Ein-/Ausgabe

```
import java.io.*;
class Fahrscheinautomat
{
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);
```

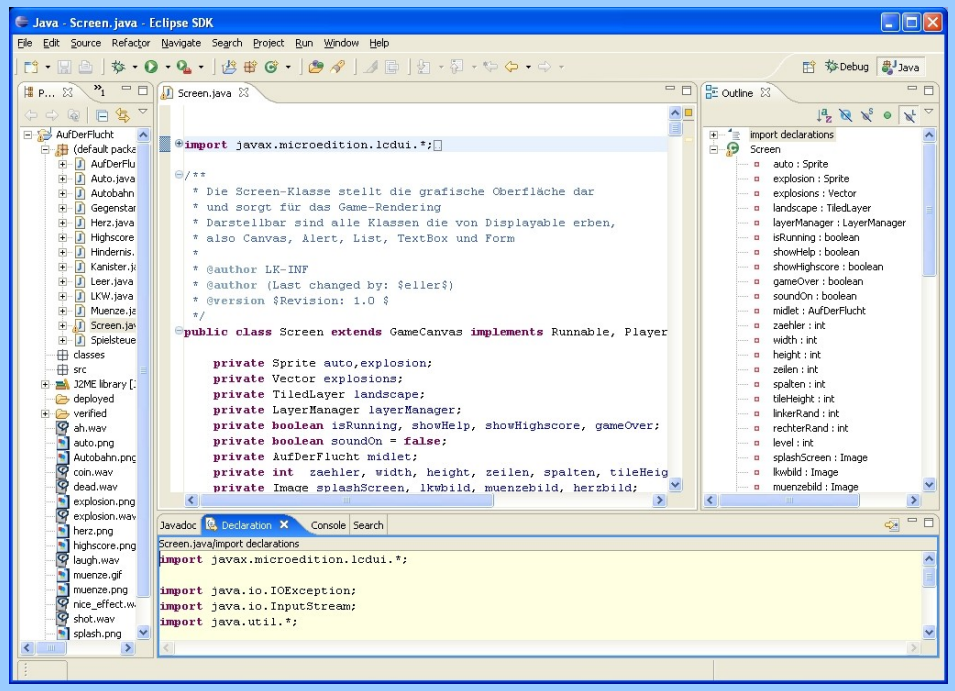
+ BlueJ:
Parameter werden durch interaktive Methodenaufrufe direkt ein- und ausgegeben.


BlueJ


BlueJ: FSA1
Projekt Bearbeiten Werkzeuge Ansicht Hilfe
Neue Klasse...
Übersetzen
Fahrscheinautomat
BlueJ: Methodenaufruf
// Methode zum Verarbeiten der eingeworfenen Münzen (in Cent);
// nur wenn ein Fahrschein 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 ( )
Ok
Abbrechen
fahrsche1:
Fahrscheinaut

4. Probleme mit Entwicklungsumgebungen

- zu komplex
- nicht objektorientiert
- GUI-orientiert
- kosten viel Geld



- Arbeiten ohne Entwicklungsumgebung führt zu Einschränkungen

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

5. **Beispiele** und Konzepte

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

6. 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
- …