

Vorkurs in Informatik: Tag 3

Eine Einführung in die Programmierung mit Groovy

Emanuel Giger, Giacomo Ghezzi, Michael Würsch, and
Harald Gall

University of Zurich, Switzerland



University of Zurich
Department of Informatics



software evolution & architecture lab

Ablauf

1. Tag: Grundlagen
2. Tag: Software Engineering by Example
3. Tag: Einführung in die Programmierung

Ablauf: Tag 1

09:30 bis 12:00

- Was ist ein Computer?
- Wie ist ein Computer aufgebaut?
- Das Rechnen mit Wahrheitswerten
- Zahlensysteme
- Wie bringe ich den Computer dazu, für mich Probleme zu lösen?

13:00 bis 16:00

- Eine Einführung in die Programmierung mit Scratch

Ablauf: Tag 2

09:30 bis 12:00

- Eine Einführung in das systematische Entwickeln von Software (aka. Software Engineering)
- Beginn Gruppenarbeiten: Ein kleines eigenes Projekt mit Scratch

13:00 bis 16:00

- Fortsetzung vom Morgen

Ablauf: Tag 3

09:30 bis 12:00

- Kurzpräsentationen der Gruppenarbeiten vom Vortag
- Eine Einführung in die Programmierung mit Groovy

13:00 bis 16:00

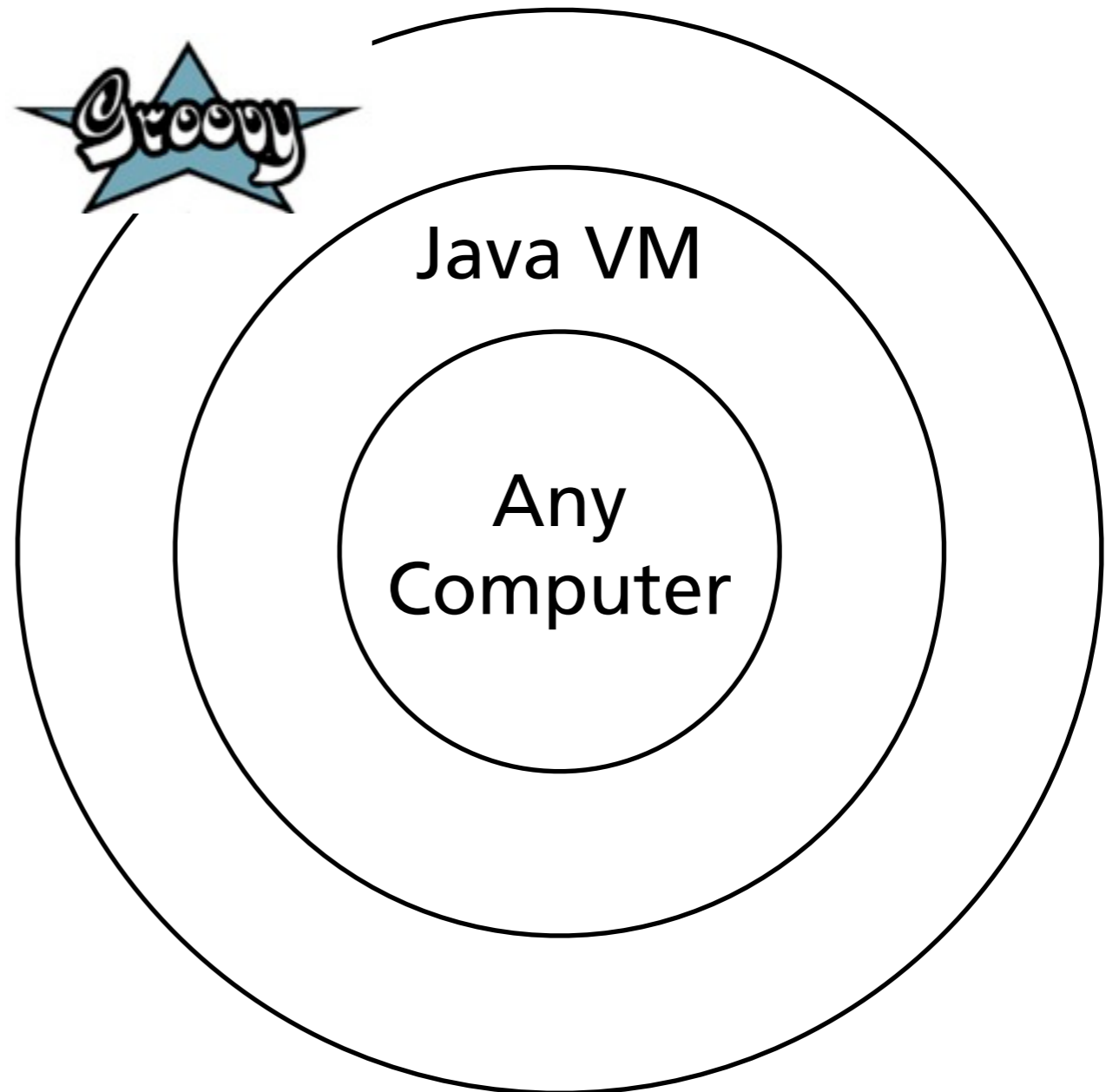
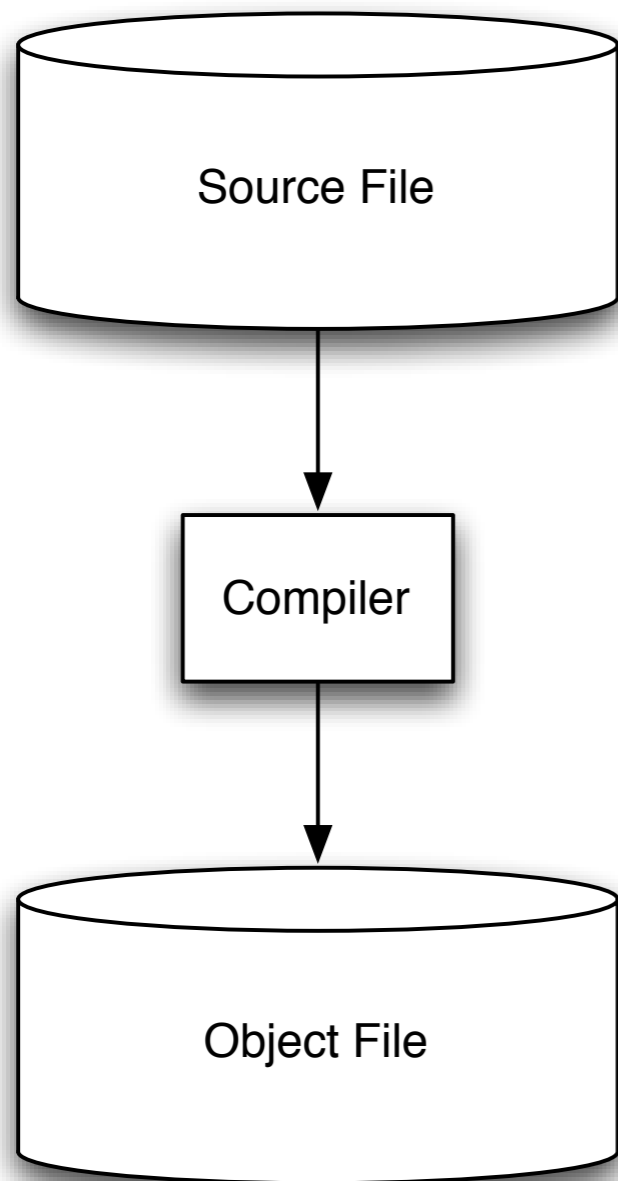
- Fortsetzung vom Morgen

Zielsetzung

Die wichtigsten Konzepte der Programmierung von Scratch auf eine "richtige" Programmiersprache übertragen.



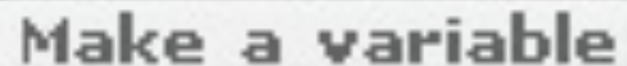
Groovy - Eine dynamische Sprache für die JVM



Groovy: Variablen

Man kann Werte einer Variablen zuweisen, und sie später wiederverwenden.

Scratch:



Make a variable



variable

Groovy:

```
x = 1  
println x
```

```
x = -3.1499392  
println x
```

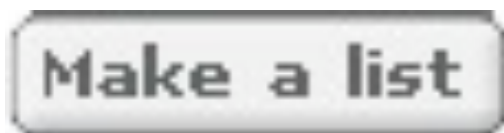
```
x = false  
println x
```

```
x = "Hi"  
println x
```


Groovy: Listen

Listen werden dazu verwendet, um eine Menge von Werten geordnet zu speichern.

Scratch:

A grey Scratch block with rounded corners and a drop shadow, containing the text "Make a list".An orange Scratch block with rounded corners and a drop shadow. It contains the text "add apple to my list" with a small downward-pointing triangle at the end of "my list".

Groovy:

```
myList = [1776, -1, 33, 99, 0, 928734928763]
println myList[0]
```

```
myList.add(10)
```

Groovy: Decisions

if-else Statements werden dazu verwendet, Entscheidungen zu treffen.

Scratch:



Groovy:

```
amPM = Calendar.getInstance().get(Calendar.AM_PM)
if (amPM == Calendar.AM) {
    println("Good morning")
} else {
    println("Good evening")
}
```

Groovy: Loops

Loops (Schleifen) werden dazu verwendet, bestimmte Anweisungen mehrmals auszuführen.

Scratch:



Groovy:

```
for (i=0; i < 10; i++) {...}
```

```
len = 10
```

```
for (i=0; i < len; i++) {...}
```

```
for (i in 0..10) {...}
```

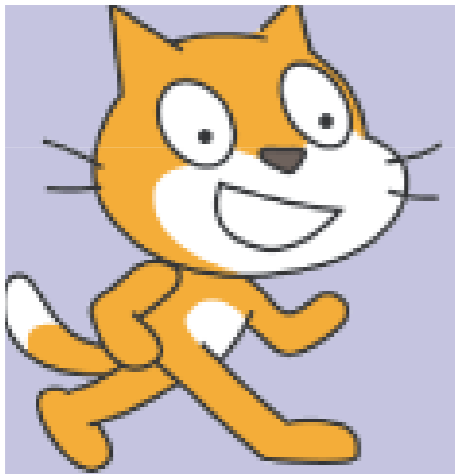
```
i = 0
```

```
while (i < 10){...}
```

Groovy: Klassen / Objekte

Klassen beschreiben ein Konzept, also eine Menge zusammengehöriger Attribute und Verhalten. Objekte sind Instanzen einer Klasse.

Scratch:



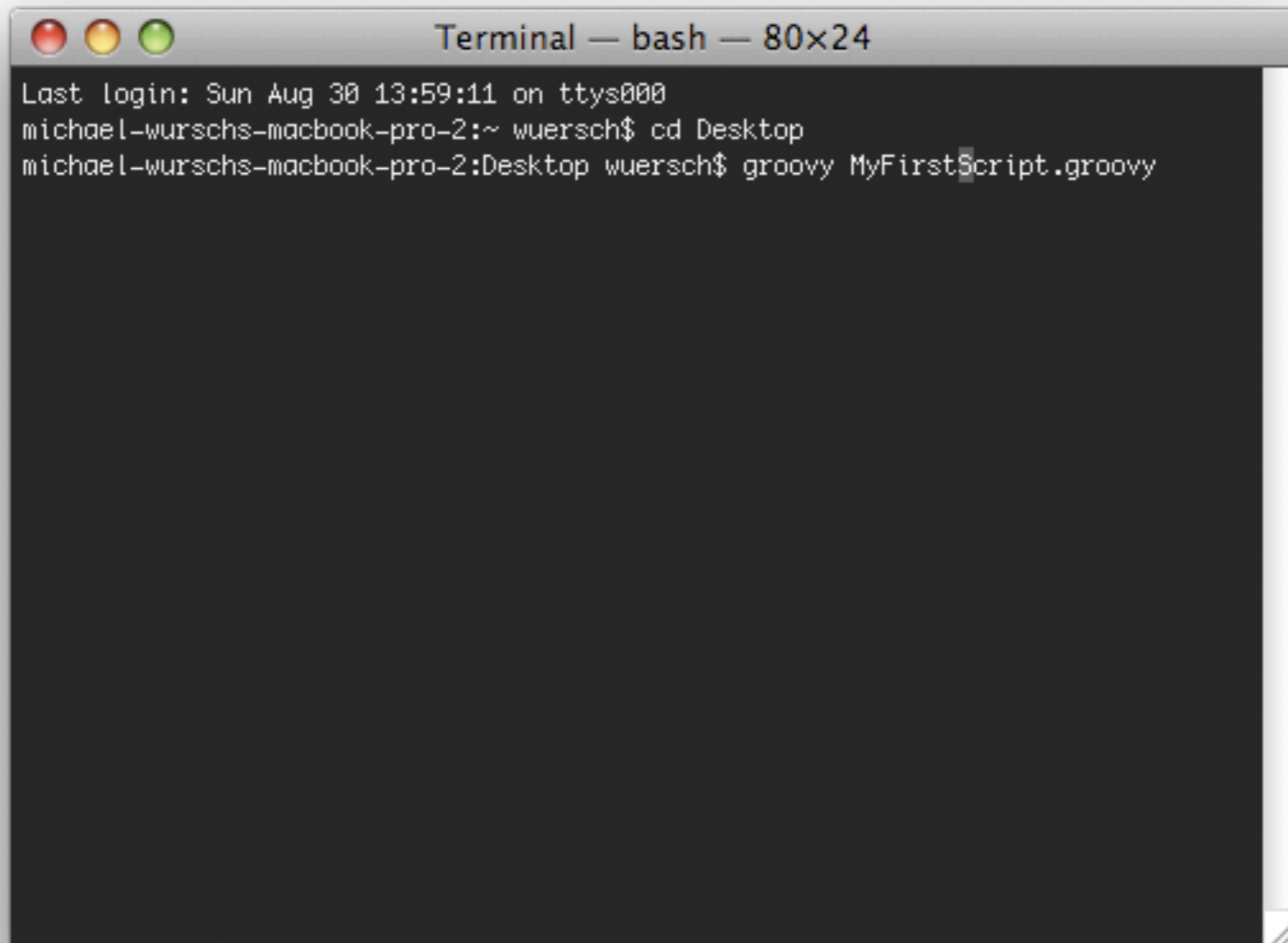
Groovy:

```
class CatSprite {  
    def name  
  
    def width  
    def height  
  
    def moveTo(x, y) {...}  
}
```

```
myCat = new CatSprite(name:"Minka", width:10, height:10)  
myCat.moveTo(200, 200)
```

```
myBigCat = new CatSprite(name:"Bruce", width:20, height:20)  
myBigCat.moveTo(100, 50)
```

Let's get Groovy!

A terminal window titled "Terminal — bash — 80x24" with three window control buttons (red, yellow, green) in the top-left corner. The terminal content shows a login message, a directory change to Desktop, and the execution of the Groovy command.

```
Terminal — bash — 80x24
Last login: Sun Aug 30 13:59:11 on ttys000
michael-wurschs-macbook-pro-2:~ wuersch$ cd Desktop
michael-wurschs-macbook-pro-2:Desktop wuersch$ groovy MyFirstScript.groovy
```