

Programmieren I

Dr. Klaus Höppner

Hochschule Darmstadt – Wintersemester 2009/2010

1/6

JAR-Dateien

2/6

JAR-Dateien

Die Java-JDK bietet ein Archiv-Format, um mehrere Dateien zu einer Datei zusammen zu packen. Dieses heißt JAR. Für das Komprimieren wird der ZIP-Algorithmus verwendet (daher kann man JAR-Dateien auch in .zip umbenennen und wie jede andere ZIP-Datei öffnen).

Syntax: `jar [ctx][v]f datei.jar [dateien]`

Hierbei wird `c` zum Erzeugen (*create*) einer Datei mit den angegebenen Dateien benutzt, `t` zum Auflisten (*table*) der enthaltenen Dateien, `x` zum Auspacken (*extract*) der Dateien. Wird zusätzlich die Option `v` verwendet, so werden beim Ausführen zusätzliche Informationen (*verbose*) angezeigt.

3/6

Verwenden von JAR-Dateien im Classpath

Der Vorteil von JAR-Dateien liegt darin, dass diese im CLASSPATH verwendet werden können, und dann in dem JAR-Archiv eingepackte Class-Dateien von der JVM gefunden werden.

Befindet sich in einer Datei `myclasses.jar` z. B. eine Datei `Main.class`, so kann man diese mit dem Befehl

```
java -jar myclasses.jar Main
```

oder

```
java -cp myclasses.jar Main
```

ausgeführt werden. Hierbei können in der Klasse `Main` weitere Klassen verwendet werden, die sich innerhalb der JAR-Datei befinden.

4/6

Das Manifest

In der JAR-Datei befindet sich automatisch eine Manifest-Datei, die zusätzliche Informationen aufnehmen kann, nämlich:

- Einen Entry-Point, d. h. die Klasse, die beim Starten von `java -jar datei.jar` ausgeführt werden soll, sowie
- einen Classpath, der beim Ausführen berücksichtigt werden soll.

Eine Manifest-Datei wird beim Erzeugen des Archivs über die Option `m` angegeben, z. B.

```
jar cfm archiv.jar manifest.txt dateien
```

5/6

Beispiel

Erstellt man folgende Manifest-Datei:

```
Manifest-Version: 1.0  
Main-Class: app.MyApp  
Class-Path: MyUtils.jar
```

und fügt diese dem JAR-Archiv mit der Option `m` beim Erstellen hinzu, so wird bei

```
java -jar archiv.jar
```

die statische Methode `main` aus der Klasse `MyApp` im Paket `app` innerhalb des Archivs verwendet, Klassen werden zusätzlich auch in dem Archiv `MyUtils.jar` gesucht.

6/6