

II.4.3 Modularität und Pakete

Montag, 26. November 2018 11:00

- SW sollte modular aufgebaut (\Rightarrow bessere Entwicklung, Übersicht, Änderbarkeit)
- Klassen werden in Pakete (packages) zusammengefasst
- Pakete werden zu Modulen zusammengefasst (Seit Java 9)

Bsp: Element und Liste sollten im gleichen Paket listen stehen.

Paketstruktur $\hat{=}$ Ordnerstruktur

⇒ Liste.java,
Element.java sollten
Dateien in Ordner listen
sein.

Wenn man am Anfang
der Klasse kein

```
package ... ;
```

schreibt, dann gehört die
Klasse zu Standardpaket
ohne Namen.

Verwendung von Klassen
aus anderen Paketen:

- Angabe des Paketnamens
vor dem Klassenname oder
- import

Reihenfolge:

erst package,
dann import

Wenn Test.java im aktuellen Verzeichnis steht und dieses hat Unterverzeichnis listen:

```
javac listen/Liste.java
```

```
javac Test.java
```

Paket

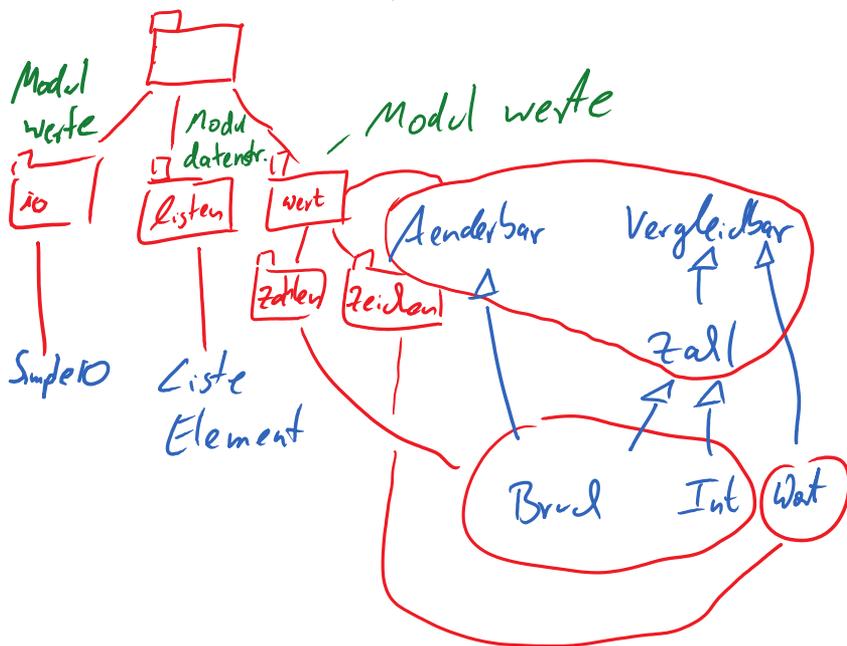
- exportiert die Komponenten, die mit public oder protected gekennzeichnet sind
- import ggf Komponenten aus anderen Paketen

Paket java.lang wird automatisch immer importiert.

Schnittstellendoku. für Pakete kann ebenfalls mit

Javadoc erstellt werden.

javac -d doc
listen
⋮



Modul user

test

• Pakethierarchie:

wert hat Unterverzeichnis

Zahlen, Schreibweise in Java:

wert.Zahlen

• Man kann nicht automatisch vom "Ober-" auf "Unterpaket" zugreifen (und auch nicht umgekehrt).

• Man sollte Prog. so strukturieren, dass Änderungs-

aufwand auf 1 Paket
beschränkt bleibt.

Seit Java 9: Module
zur Zusammenfassung von
Paketen

Javadoc kann auch Module dokumentieren:

```
javadoc -d doc --module-source-path src  
--module datenstr,werte,user
```