

## 1.6 Anwendung/Wildcards

### 1.6.1 Mehrere Dateien auf einmal ansprechen

Häufig möchte man mit einem Befehl mehrere Dateien auf einmal bearbeiten. So kann es sein, dass man im Verzeichnis 100 MP3-Dateien liegen hat, die zusammen viel Platz einnehmen und die man alle auf einmal löschen möchte. Nun kann man sie alle von Hand löschen:

```
Terminal
schueler@debian964:~$ rm sinfonie_alt.mp3
schueler@debian964:~$ rm sinfonie_neu.mp3
schueler@debian964:~$ rm kantate_bach.mp3
schueler@debian964:~$ rm air.mp3
C:\> del sinfonie_alt.mp3
C:\> del sinfonie_neu.mp3
C:\> del kantate_bach.mp3
C:\> del air.mp3
```

Die allermeisten Befehle aber erlauben es, mehrere Dateinamen als Parameter anzunehmen:

```
Terminal
schueler@debian964:~$ rm sinfonie_alt.mp3 sinfonie_neu.mp3 \
> kantate_bach.mp3 air.mp3
C:\> del sinfonie_alt.mp3 sinfonie_neu.mp3 kantate_bach.mp3 air.mp3
```

An dieser Stelle setzt die Wildcard-Expansion von Pfadnamen ein:

Mit `sinfonie_???.mp3` erreicht man alle Dateien, deren Namen mit `sinfonie_` beginnen, dann drei beliebige Zeichen haben und auf `.mp3` enden. Das Fragezeichen ist also ein Joker (engl. *wildcard*), der auf jedes andere Zeichen passt. Pro Zeichen braucht man also ein Fragezeichen, weil jedes Fragezeichen nur ein einziges Zeichen ersetzt (wie im Kartenspiel). Mit dem folgenden Befehl können also zwei der MP3-Dateien gelöscht werden:

```
Terminal
schueler@debian964:~$ rm sinfonie_???.mp3
C:\> del sinfonie_???.mp3
```

Mit `*.mp3` erreicht man alle Dateien, deren Namen mit beliebigen Zeichen beginnen und auf `.mp3` enden. Der Stern ist also ein Joker, der auf jede beliebige Zusammenstellung von Zeichen passt. Deshalb reicht für mehrere Zeichen ein Stern. Der Stern passt übrigens auch auf eine leere Zeichenkette. So kann man nun alle MP3-Dateien löschen:

```
Terminal
schueler@debian964:~$ rm *.mp3
C:\> del *.mp3
```

Ein entscheidender Unterschied zwischen den Linux-Shells und dem Standard-Kommando-Interpreter von Windows liegt darin, dass die Linux-Shells diese Ersetzung auch für andere Programme nutzbar machen (auch für unsere selbstgeschriebenen): Wenn man eine Kommandozeile mit Wildcards aufruft, werden die Wildcards automatisch ersetzt. Beim Standard-Kommando-Interpreter von Windows funktioniert das leider nur mit den eingebauten Befehlen wie `copy`, `del` usw.

Wichtig ist, dass man mit den Wildcards nur auf existierende Dateien und Verzeichnisse prüfen kann. Deshalb kann man mit `mkdir ?` nicht mehrere Verzeichnisse anlegen.

### 1.6.2 Ausführlicher Modus der Bash

Bei der Bash kann man mit dem Befehl `set -x` sichtbar machen, was passiert. Jeder Befehl wird dann vor der Ausführung angezeigt:

```

Terminal
schueler@debian964:~$ set -x
schueler@debian964:~$ cp echo *.mp3 /tmp
+ cp sinfonie_alt.mp3 sinfonie_neu.mp3 kantate_bach.mp3 air.mp3 /tmp

```

Mit dem Befehl `set +x` macht man den vorherigen `set`-Befehl wieder rückgängig.

### 1.6.3 Nichts gefunden

Wenn zu einem Wildcard-Ausdruck kein Eintrag gefunden wird, wird es nicht expandiert. Stattdessen bleibt der Wildcard-Ausdruck im Befehl:

```

Terminal
schueler@debian964:~$ echo *.mp3
sinfonie_alt.mp3 sinfonie_neu.mp3 kantate_bach.mp3 air.mp3
schueler@debian964:~$ echo *.mp4
*.mp4

```

### 1.6.4 Versteckte Dateien

Unter Linux gelten Dateien und Verzeichnisse, deren Namen mit Punkt beginnen, als versteckt. Sie werden durch normale Befehle nicht angezeigt, und bearbeitet. Insbesondere Konfigurationsdateien sind meistens versteckt.

Daher werden sie auch durch Wildcard-Expansion nicht mit expandiert. Das kann man jedoch ändern mit dem Befehl `shopt -s dotglob`. Mit dem Befehl `shopt -u dotglob` kann man das bisherige Verhalten wieder einschalten.

### 1.6.5 Maskierung

Auf der Kommandozeile haben einige Zeichen eine besondere Bedeutung:

- Leerzeichen, Tabulator und Zeilenumbruch trennen Dateinamen
- Dollarzeichen kennzeichnen Variablennamen
- Stern und Fragezeichen sind Wildcards

Was macht man nun, wenn ein Dateiname ein solches Zeichen enthalten soll? Man macht es unwirksam, indem man den ganzen Dateinamen in Hochkommata setzt:

```

Terminal
schueler@debian964:~$ ls 'die drei ???'.mp3
C:\> dir 'die drei ???'.mp3

```

Dieses Unwirksammachen nennt man *Maskierung*.

Eine andere Möglichkeit ist es, jedem Zeichen mit besonderer Bedeutung einen Backslash voranzustellen:

```

Terminal
schueler@debian964:~$ ls die\ drei\ \?\?\?.mp3
C:\> ls die\ drei\ \?\?\?.mp3

```

Es gibt noch eine dritte Möglichkeit der Maskierung, nämlich Anführungszeichen. Das Anführungszeichen maskiert Leerzeichen und weitere Trennzeichen, es maskiert Wildcards – es maskiert aber kein Dollarzeichen. Das heißt, Variablen werden trotzdem erkannt:

```

Terminal
schueler@debian964:~$ echo "$USER - Musik - Verzeichnis"
schueler - Musik - Verzeichnis
schueler@debian964:~$ echo '$USER - Musik - Verzeichnis'
$USER - Musik - Verzeichnis

```