

Besonderheiten Textdateien Windows → Linux



Besonderheiten Textdateien aus Windows

Zeilenumbrüche:

- Windows-Zeilenumbrüche auf Linuxsystemen
 - Zeilenumbrüche werden in der Windows- und der Linuxwelt anders dargestellt
 - Windows verwendet zwei Sonderzeichen, nämlich `\r\n`, die auch als CRLF bezeichnet werden.
 - Linux verwendet NUR ein Zeichen, nämlich `\n`, das auch als LF bezeichnet wird.
- Das Ergebnis dieser Unterschiede ist, dass Textdateien, die nicht für das jeweilige System erstellt wurden, den für das Zielsystem typischen Zeilenumbruch NICHT verwenden.
- Das Bearbeiten von Textdateien kann Probleme machen, wenn man diesen Umstand ignoriert.

Besonderheiten Textdateien aus Windows

Abhilfe Zeilenumbrüche:

- Vor der Verarbeitung eines Textfiles ist eine Analyse des Inhalts (inkl. enthaltener Sonderzeichen) empfehlenswert – Kommando od -c filename verwenden.
- Analyse der Sonderzeichen im Inhalt, Sonderzeichen beginnen immer mit einem \ od -c filename
- Reduziere CRLF zu LF mit sed

```
sed -r 's/\r//g' filename > filename_ohne_cr
```
- Alternativ mit Programm dos2unix

```
dos2unix filename  
(Original File wird überschrieben)
```

```
0000000 T e i l n e h m e r ; ; ; \r \n e  
0000020 r s t e l l t a m : 2 6 . 0  
0000040 3 . 2 0 1 5 1 6 : 4 2 ; ; ; \r  
0000060 \n ; ; ; \r \n-----
```

Windowsfile mit CRLF
filename

```
0000000 T e i l n e h m e r ; ; ; \n e  
0000020 s t e l l t a m : 2 6 . 0 3  
0000040 . 2 0 1 5 1 6 : 4 2 ; ; ; \n ;  
0000060 ; ; \n-----
```

Linuxfile mit LF
filename_ohne_cr

Besonderheiten Textdateien aus Windows

Leerzeilen mit Tabs:

- Textfiles enthalten oft Tabs als Trennzeichen, die aber bei der Anzeige des Textes mit cat oder less nicht dargestellt werden
- Leerzeilen sind oft mit Tabs belegt, die aber bei der Anzeige nicht erscheinen und die Zeile erscheint leer.
- Der Versuch, Leerzeilen mit folgendem sed –Kommando zu löschen, scheitert.

```
sed -r '/^$/d' filename
```

```
Teilnehmer  
erstellt am: 26.03.2015 16:42
```

```
Personenkz. Name Vornamen Stg./Org.
```

```
cat filename
```

```
0000000 T e i l n e h m e r \t \t \t \n e r  
0000020 s t e l l t a m : 2 6 . 0 3  
0000040 . 2 0 1 5 1 6 : 4 2 \t \t \n \t  
0000060 \t \t \n P e r s o n e n k z . \t N  
0000100 a m e \t V o r n a m e n \t S t g
```

```
od -c filename
```

Besonderheiten Textdateien aus Windows

Abhilfe Leerzeilen und Tabs:

Überprüfen des Inhalts einer „Leerzeile“ und mit sed Kommando ggf. anpassen durch Löschen überflüssiger Tabs

```
sed -r '/^(\t)*$/d' filename # entferne alle Tabs
```

```
Teilnehmer  
erstellt am: 26.03.2015 16:42  
Personenkz. Name      Vornamen  Stg./Org.
```

```
cat filename
```

```
0000000 T e i l n e h m e r \t \t \t \n e r  
0000020 s t e l l t a m : 2 6 . 0 3  
0000040 . 2 0 1 5 1 6 : 4 2 \t \t \t \n P  
0000060 e r s o n e n k z . \t N a m e \t  
0000100 V o r n a m e n \t S t g . / O r
```

```
od -c filename
```

Auch bei **Ankerverwendung** am Anfang oder am Ende einer Zeile kann dieses Problem auftreten!

Besonderheiten Textdateien aus Windows

Falsche Codepage im Textfile:

- Falsche Codierung, z.B.: keine dt. Umlaute oder ß erkannt
- Textfiles müssen an die Codierung des Zielsystems angepasst werden.
- Beispiel:

```
14/1/0307/074;Kr<E4>mer;Alexander Markus Jakob;SEBakk/VZ  
14/1/0307/084;<D6>ller;Michael;SEBakk/VZ
```

Besonderheiten Textdateien aus Windows

Abhilfe Falsche Codepage im Textfile :

- 1. Feststellen der aktuellen Codepage mit Kommando `file`:

```
file filename (liefert aktuelle codepage für File)
```

Ausgabe Beispiel:

```
filename: ISO-8859 text
```

- 2. Konvertieren des Files in die richtige Codepage

```
iconv -f ISO-8859-1 -t UTF8 filename > filename_con
```