HOWTO zu Unix-Entitäten

(C) 2006-2024 T.Birnthaler/H.Gottschalk <howtos(at)ostc.de>
OSTC Open Source Training and Consulting GmbH
http://www.ostc.de

$Id: unix-entities-HOWTO.txt,v 1.12 2025/02/23 20:14:55 tsbirn Exp $

Dieses Dokument beschreibt die Entitäten des Betriebssystems UNIX.

INHALTSVERZEICHNIS

1) Namen unter UNIX
2) Anmerkungen
3) Beispiele für Namen

---

1) Namen unter UNIX

UNIX ist nicht grafisch orientiert, sondern kommando-orientiert. D.h. man benötigt ständig "Namen" für irgendwelche "Dinge", die man verwenden möchte.

Ein Problem unter UNIX ist dabei, herauszubekommen, wofür ein Name (z.B. "xyz") steht. Da Namen fast immer klein geschrieben werden (außer bei Variablen), Dateinamen-Extensions analog Windows (z.B. ".exe", ".com", ".txt") kaum verwendet werden und die UNIX-Namen anfangs noch recht ungewohnt sind, dauert es eine Weile, bis man zurecht kommt (und vor allem eine gewisse "Intuition" entwickelt).

Ein NAME kann im Prinzip für eine der folgenden Entitäten stehen:

Entität	Bedeutung
DATEI	Zusammenfassung beliebiger Daten/Informationen (z.B. "gedicht", Text, Skript, Programm, Datenliste, ...)
VERZ	Liste von Datei- und Verzeichnis-Namen (z.B. "/tmp")
VAR	Variable (Parameter), enthält einen "Wert" (z.B. $PATH, $HOME)
BUILTIN	In Shell direkt eingebautes Kommando (z.B. "cd", "pwd", "set")
ALIAS	Kommando-Abkürzung, Miniskript (z.B. c=clear, ll="ls -l")
FUNKTION	Analog Alias, aber leistungsfähiger (Parameter möglich)
KMDO	Ausführbares Binär-Programm (z.B. "ls", "ps")
SKRIPT	Meist eine Zusammenfassung von Shell-Kommandos
GERÄT	DATEI für Hardware, Zugriff meist eingeschränkt (device)

Je nach Entitäts-Typ sind UNTERSCHIEDLICHE Kommandos notwendig, um die entsprechende Entität zu verwenden bzw. zu bearbeiten:

Operation	Datei	Verzeichnis	Variable	Alias
Anlegen	touch DATEI vi DATEI... :wq	mkdir VERZ	VAR="Text"	alias \ NAME="ERSATZ"
Löschen	rm DATEI	rmdir VERZ rm -r VERZ	unset VAR VAR=	unalias NAME
Anzeigen	more DATEI less DATEI cat DATEI vi DATEI	ls VERZ ll VERZ ls -R VERZ ls -ld VERZ	echo $VAR env (TEIL!) set (ALLE!)	alias NAME alias (ALLE!)

Öffnen	KMDO DATEI vi DATEI	---	---	---
Ausführen	chmod a+x DATEI PATH=$PATH:... DATEI ./DATEI /PFAD/ZU/DATEI	cd VERZ	$VAR	NAME
Kopieren	cp DATEI NEU	cp -r VERZ NEU	NEU=$VAR	alias NEU=NAME
Verschieben	mv DATEI VERZ	mv VERZ VERZ2	---	---
Umbenennen	mv DATEI NEU	mv VERZ NEU	Kop.+Löschen	Kop.+Löschen
2. Namen erzeugen	ln DATEI NEU ln -s DATEI NEU	(geht nicht!) ln -s VERZ NEU	---	alias NEU=NAME

2) Anmerkungen

* "rmdir" funktioniert nur bei LEEREM Verzeichnis.

* "-r/-R" bedeutet "recursive/rekursiv", d.h. ganzen Dateibaum bearbeiten.

* Von Benutzern können keine Hardlinks auf Verzeichnisse angelegt werden (UNIX selbst legt aber die Hardlinks "." + ".." für jedes Verzeichnis autom. an).

* Variablen werden meist GROSS geschrieben, damit sie leicht erkennbar sind (nur eine Konvention, kein Muss!).

* "type KMDO" (bzw. "which KMDO", "whence KMDO" oder "whereis KMDO") liefert den Typ (Builtin, Alias, Funktion, Binärprogramm) und den Ort (Pfad) bzw. die Definition von "KMDO".

* "ls -l DATEI" liefert Informationen über eine DATEI (Zugriffsrechte, Anzahl Hardlinks, Besitzer, Besitzer-Gruppe, Grösse, Datum der letzten Änderung).

* "ls -ld VERZ" liefert Informationen über ein VERZEICHNIS (Zugriffsrechte, Anzahl Hardlinks, Besitzer, Besitzer-Gruppe, Grösse, Datum letzte Änderung). (ohne "-d" (directory) wird der INHALT des Verzeichnisses angezeigt!)

3) Beispiele für Namen

Name	Beschreibung
RANDOM $RANDOM	Variable (Gross)
/tmp /tmp/	Verzeichnis (evtl. am Suffix "/" erkennbar, d.h. "/tmp/")
.alias	(Versteckte) Datei (kommen fast nur in Heimat-Verz. vor)
/	Wurzel-Verz. des Dateisystems (ohne Laufwerk-Buchstabe!)
~	Heimat-Verz. des aktuellen Benutzers ("/home/USERNAME")
.	Aktuelles Verz. (von "pwd" oder "echo $PWD" angezeigt)
..	Übergeordnetes Verz., Eltern-Verz., Stamm-Ordner
ls	Kommando (Programm-Datei, über Suchpfad PATH gefunden)
/bin/ls	Programm-Datei (in "/bin" liegen nur Programme)
/sbin/dump	Programm-Datei (in "/sbin" liegen nur Programme)
/usr/bin/vi	Programm-Datei (in "/usr/bin" liegen nur Programme)
/usr/sbin/vipw	Programm-Datei (in "/usr/sbin" liegen nur Programme)
/etc/profile	System-Konfigurations-Datei ("/etc" enthält nur solche)

```
| ll              | Alias für "ls -l"                                          |
| dir             | Alias für "ls -l" oder Binärprogramm "/usr/bin/dir"        |
+-----------------+------------------------------------------------------------+
```