

Batch-Programmierung

Wikibooks.org

21. Juni 2012

On the 28th of April 2012 the contents of the English as well as German Wikibooks and Wikipedia projects were licensed under Creative Commons Attribution-ShareAlike 3.0 Unported license. An URI to this license is given in the list of figures on page 101. If this document is a derived work from the contents of one of these projects and the content was still licensed by the project under this license at the time of derivation this document has to be licensed under the same, a similar or a compatible license, as stated in section 4b of the license. The list of contributors is included in chapter Contributors on page 97. The licenses GPL, LGPL and GFDL are included in chapter Licenses on page 105, since this book and/or parts of it may or may not be licensed under one or more of these licenses, and thus require inclusion of these licenses. The licenses of the figures are given in the list of figures on page 101. This PDF was generated by the \LaTeX typesetting software. The \LaTeX source code is included as an attachment (`source.7z.txt`) in this PDF file. To extract the source from the PDF file, we recommend the use of <http://www.pdflabs.com/tools/pdftk-the-pdf-toolkit/> utility or clicking the paper clip attachment symbol on the lower left of your PDF Viewer, selecting Save Attachment. After extracting it from the PDF file you have to rename it to `source.7z`. To uncompress the resulting archive we recommend the use of <http://www.7-zip.org/>. The \LaTeX source itself was generated by a program written by Dirk Hünninger, which is freely available under an open source license from http://de.wikibooks.org/wiki/Benutzer:Dirk_Huenniger/wb2pdf. This distribution also contains a configured version of the `pdflatex` compiler with all necessary packages and fonts needed to compile the \LaTeX source included in this PDF file.

Inhaltsverzeichnis

1	Allgemeine Information	3
1.1	Batch-Programmierung	3
1.2	Wie erstelle ich eine Batchdatei?	4
1.3	Allgemeine Informationen	4
2	Wichtige DOS-Kommandos	5
2.1	assoc	5
2.2	attrib	5
2.3	cacls	5
2.4	cd / chdir	6
2.5	chcp	7
2.6	cls	8
2.7	comp oder fc	8
2.8	(x)copy/robocopy	8
2.9	date	8
2.10	del / erase	9
2.11	deltree	9
2.12	dir	10
2.13	edit	11
2.14	exit	11
2.15	find	11
2.16	format	11
2.17	help	11
2.18	ipconfig	11
2.19	keyb	12
2.20	loadhigh	12
2.21	md	12
2.22	more	12
2.23	move	13
2.24	netsh	13
2.25	path	13
2.26	prompt	13
2.27	rd / rmdir	13

2.28	ren oder rename	14
2.29	set	14
2.30	setlocal	14
2.31	start	14
2.32	taskkill	14
2.33	tasklist	15
2.34	time	15
2.35	title	15
2.36	type	16
2.37	Color	16
3	Batch Befehle	17
3.1	Vorsicht Leerzeichen!	17
3.2	@	17
3.3	:(Doppelpunkt)	18
3.4	CALL	19
3.5	CLS	21
3.6	CMD	21
3.7	COLOR	22
3.8	COMMAND	23
3.9	ECHO	24
3.10	FOR	26
3.11	GOTO	30
3.12	IF	30
3.13	PAUSE	33
3.14	PUSHD / POPD	34
3.15	REM	35
3.16	START	35
3.17	Variablen	37
4	Batch Operatoren	41
4.1	&	41
4.2	41
4.3	&&	42
4.4	42
4.5	<	43
4.6	>	43
4.7	>>	45
5	Erweiterungen unter Windows NT	47
5.1	Batchparameter	47

5.2	Stringmanipulation	47
5.3	Umgebungsvariablen	49
5.4	Die Kommandozeile lernt Rechnen	50
5.5	Benutzereingaben	51
5.6	Ausgaben in eine Variable umleiten	51
6	Programmierungshilfen	53
7	Netzwerkumgebung	65
7.1	ping	65
7.2	ipconfig	66
7.3	tracert	68
7.4	netstat	69
7.5	NET	69
8	Beispiele	75
9	Zusatz-Tools	89
10	Referenz	91
10.1	Liste aller internen und externen Befehle (entspricht der Ausgabe des Befehls <code>help</code> unter Windows XP)	91
11	WebLinks	95
11.1	Weblinks	95
12	Autoren	97
	Abbildungsverzeichnis	101
13	Licenses	105
13.1	GNU GENERAL PUBLIC LICENSE	105
13.2	GNU Free Documentation License	106
13.3	GNU Lesser General Public License	107

Kategorie: Buch mit Druckversion¹

¹ <http://de.wikibooks.org/wiki/Kategorie%3ABuch%20mit%20Druckversion>

1 Allgemeine Information

1.1 Batch-Programmierung

Batchprogramme (häufig mit Stapelverarbeitungsprogramme oder kurz Stapelprogramm übersetzt) sind meist kurze Dateien, die Befehle der Kommandozeile der Reihe nach abarbeiten. Batchprogrammierung erfüllte die grundlegenden Anforderungen an eine Programmiersprache wie z.B. `if`, `if not` und `while`. Da Batch sich in den Bereich Shells scripting einordnen lässt, kann man von einer *interpreterbasierten Skriptsprache* reden. Das heißt, dass die im Editor erstellte Textdatei nicht mit einem Compiler¹ einmal in Maschinsprache übersetzt und in einer ausführbaren Binärdatei im `*.exe` Format gespeichert, sondern bei jedem Aufruf durch einen Interpreter zur Laufzeit in ein für den Computer verständliches Format übersetzt wird. Ein Texteditor wie MS Notepad, den Windows von Haus aus mitbringt, reicht vollkommen aus um Batchprogramme zu schreiben. Batchprogrammierung ist speziell für die Steuerung des Betriebssystems gedacht, für die Entwicklung von Anwendungen ist sie nicht geeignet. Hierzu verwendet man Sprachen wie C/C++ und Java (nicht zu verwechseln mit JavaScript!!).

Viele wiederkehrende Installations- und Verwaltungsaufgaben, die man traditionell mit Tastatur und Maus ausführt, lassen sich mit Stapeldateien ausführen. Hat man genügend Kommentarzeilen in die Batchdatei eingefügt, genügt der Ausdruck der Stapeldatei meist als Dokumentation.

Kommandozeilenbefehle und Batchdateien sind keine veraltete Technologie. Auf Wunsch vieler Systemadministratoren hat Microsoft dafür gesorgt, dass Windows Server 2003 vollständig von der Kommandozeile installiert und administriert werden kann, ohne die Maus zu benutzen.

In diesem Buch geht es ausschließlich um die Erstellung von Batch-Dateien unter Windows und DOS. Andere Betriebssysteme bieten vergleichbare und wesentlich mächtigere Funktionen (wie z. B. die in der Linux-Welt weit verbreitete Bash oder AppleScript für den Mac).

¹ <http://de.wikibooks.org/wiki/Compiler>

Die Begriffe "Stapelprogramm" und "Stapelverarbeitung" rühren aus der Zeit her, als Permanentenspeicher auf Magnetplatten sehr teuer war. Zu dieser Zeit begab man sich mit einem Stapel Lochkarten aus Pappe, auf denen sich das Programm und die Daten in gestanzter Form befanden, zum Rechenzentrum und übergab dem Operator seinen Stapel. Die Stapel aller Benutzer wurden gesammelt und dann sequentiell an den Rechner übergeben. Nach einiger Zeit konnte man sich dann die (meist ausgedruckten) Ergebnisse abholen.

1.2 Wie erstelle ich eine Batchdatei?

Wie bereits gesagt, reicht ein einfacher Editor aus. Empfohlen kann man guten Gewissens den quelloffenen und kostenlosen Notepad++²-Editor, als bessere Alternative zu MS Notepad. Speichern Sie dann die Datei mit der Endung `*.bat` ab. Anschließend müssen Sie nur die Datei starten: Dies können Sie entweder direkt in Windows oder indem Sie in der DOS-Eingabeaufforderung in das entsprechende Verzeichnis wechseln und den Dateinamen eingeben.

Unter Windows NT-kompatiblen Betriebssystemen gibt es seit Windows 2000 auch Batchdateien mit der Endung `*.cmd`. Diese werden genau wie Batchdateien mit der Endung `*.bat` verarbeitet bzw. ausgeführt. Die Endung `*.cmd` wurde ursprünglich aus Kompatibilitätsgründen zu OS/2, einem Betriebssystem das MS ursprünglich in Kooperation mit IBM entwickelte, mit Windows NT 3.x eingeführt.

1.3 Allgemeine Informationen

- Allgemeine Informationen: Wikipedia: batch³
- Infos zur cmd.exe: Wikipedia: cmd.exe⁴
- Die für **Windows XP** verfügbaren Kommandos können auch im Hilfesystem von Windows direkt eingesehen werden:
 - START - Hilfe und Support
 - "Befehlszeilenreferenz A-Z" in das Suchfeld eintragen. (Für englisches Windows "Command-line reference A-Z")

2 <http://notepad-plus.sourceforge.net/de/site.htm>

3 <http://de.wikipedia.org/wiki/Stapelverarbeitung>

4 <http://de.wikipedia.org/wiki/cmd.exe>

2 Wichtige DOS-Kommandos

Man unterscheidet bei DOS-Befehlen grundsätzlich zwischen internen und externen Befehlen. Interne Befehle sind Bestandteil der Shell selbst, während externe Befehle als eigene Programme vorliegen. Die DOS-Befehle, **die interessant für die Batch-Programmierung sein könnten**, sind im Folgenden aufgelistet:

2.1 assoc

Mit Hilfe dieses Befehls kann man die Dateierweiterungen bearbeiten bzw. anzeigen lassen.

```
assoc [.dateiendung[=Dateibesreibung]]
```

2.2 attrib

Ändert das Dateiattribut. Möglich sind "schreibgeschützt" (+r/-r), "versteckt" (+h/-h) und "archiv" (+a/-a)

2.3 cacls

Ändert die Rechte (Access Control List) auf NTFS Partitionen oder zeigt diese an.

```
cacls *.* /E /T /G Bob:C
```

Fügt beispielsweise dem User Bob das Recht "Ändern" für alle Dateien und Ordner im aktuellen und allen Unterordnern hinzu. **Achtung!** Standardmäßig werden Rechte ersetzt!

2.4 cd / chdir

Über den `cd`- oder `chdir`-Befehl (change directory) kann das aktuelle Verzeichnis gewechselt werden. Beachte: Nach dem "`cd`" folgt ein Leerzeichen, das bei nachfolgendem `..` oder `\` je nach Betriebssystemversion weggelassen werden darf. Wird der Befehl ohne Parameter ausgeführt, so wird der Pfad des aktuellen Verzeichnisses ausgegeben.

relative Angaben:

Die Eingabe von "`cd ..`" wechselt in das nächsthöhere Verzeichnis ("`cd ..\..`" zwei Ebenen höher usw.).

Die Eingabe von "`cd ORDNERNAME`" wechselt in den Unterordner `ORDNERNAME`.

absolute Angaben:

Die Eingabe von "`cd \`" wechselt in das Hauptverzeichnis auf dem derzeitig befindlichen Laufwerk.

Die Eingabe von "`cd \ORDNERNAME`" wechselt absolut in den Ordner `ORDNERNAME` auf dem derzeitig befindlichen Laufwerk.

Die Eingabe von "`LAUFWERKSBUCHSTABE:`" (ohne vorangestelltes "`cd`") wechselt in das aktuelle Verzeichnis des gewählten Laufwerks `LAUFWERKSBUCHSTABE`.

Die Eingabe von "`cd LAUFWERKSBUCHSTABE:\ORDNERNAME`" wechselt absolut in den Ordner `ORDNERNAME` auf dem gewählten Laufwerk von `LAUFWERKSBUCHSTABE` jedoch nicht von einem Laufwerk auf ein anderes.

Pfadangaben mit Leerschlägen (Bsp "`C:\program files\`") müssen in Anführungs-/Schlusszeichen eingeschlossen sein.

Der Befehl "`cd /D %~dp0`" wechselt in den Ordner des Batch-Skripts. (sinnvoll bei Windows-Doppelklicks)

Wildcards:

Ist die Pfadangabe eindeutig kann der Ordnername durch Nutzen von Wildcards (z.b. `\win*` für `\windows` oder `\winnt`) abgekürzt werden. Dies ist jedoch im Batchbetrieb nicht ratsam.

Remotesysteme:

Pfade von nicht als Laufwerken verbundenen Remotesystemen können (Berechtigung vorausgesetzt) entweder über die Administrative Freigabe oder durch das Verbinden eines Netzlaufwerks angesprochen werden. Beispiel für administrative Freigabe: `\\[computername]\[laufwerk]$\[Weiterer Pfad]`

Pfad in Variable:

Der Befehl "`set PFAD=%cd%`" schreibt das aktuelle Verzeichnis in die Variable `%PFAD%`. (als letztes im Batch-Script mit "`cd %PFAD%`" wieder ins Startverzeichnis wechseln!)

2.5 chcp

Anzeige der aktuellen CodePage oder Setzen einer neuen CodePage (**change codepage**)

```
chcp [nnn]
```

Beispiele für nnn:

- 437 – Die ursprüngliche Zeichensatztabelle des IBM-PC
- 720 – Arabisches Alphabet
- 737 – Griechisches Alphabet
- 850 – westeuropäische Sprachen (DOS-Latin-1)
- 857 – Türkisches Alphabet

- 866 – Kyrillisches Alphabet

2.6 cls

Löscht den Bildschirminhalt (**clear screen**)

2.7 comp oder fc

Vergleicht den Inhalt zweier Dateien und stellt die Unterschiede dar (**compare / file compare**).

2.8 (x)copy/robocopy

Mit dem Befehl `copy` können eine Datei oder mehrere Dateien kopiert werden.

`xcopy` ist die erweiterte Version von `copy`, die mehr Optionen bietet. Diese ist nicht in allen Windows-Versionen verfügbar.

Das weitaus mächtigere `robocopy` ist ab Vista standardmäßig wieder verfügbar, bei Windows XP und 2000 fehlt dieses Programm zunächst. Microsoft stellt `robocopy` aber im Windows Server 2003 Resource Kit Tools 2003 auch für diese Betriebssystemversionen bereit.

Mit dem Befehl `copy /b "DATEI" + "DATEI" "NEUE DATEI"` kann man Dateien zusammenfügen.

2.9 date

Gibt das aktuelle Datum aus und ermöglicht dem Benutzer die Änderung des Datums. Wird der Befehl mit dem Parameter `/t` aufgerufen, so wird nur das aktuelle Datum ausgegeben. Um das Datum zu teilen, also z.B. nur das Jahr zu verwenden, gibt es folgende Syntax:

```
date:~6,4
```

Beispiel:

```
z:\>echo %date:~6,4%  
2010
```

Hier werden vom Datum die ersten 6 Zeichen weggelassen und dann vier Stellen angezeigt. Bei `time`¹ funktioniert das analog.

Bei einigen Betriebssystemversionen erfordert das Ändern des Systemdatums administrative Rechte.

XP: Die Ausgabe des Datumsformates ist abhängig von den Einstellungen in den Regions- und Sprachoptionen (Systemsteuerung)

2.10 del / erase

Löscht eine Datei, die sich im aktuellen Verzeichnis befindet. Als Parameter muss mindestens der Dateiname angegeben werden. Weitere Parameter sind optional:

- /p - sollen mehrere Dateien gelöscht werden, muss jede einzelne Löschung bestätigt werden.
- /f - auch schreibgeschützte Dateien werden gelöscht.
- /s - löscht auch Dateien, die sich in den Unterverzeichnissen befinden.
- /q - Unterbindet Bestätigung.

Beispiel:

```
erase H:\zulöschendedatei.bat
```

2.11 deltree

Löscht Verzeichnis(se) inkl. untergeordnete Dateien.

Unter Windows XP lautet der Befehl **rmdir**. Deltree kennt es nicht mehr. Anmerkung: `rmdir`/`rd` versteht keine Wildcards ('*' oder '?') im Verzeichnisnamen.

1 Kapitel 2.34 auf Seite 15

2.12 dir

Der Befehl `dir` (directory) zeigt alle Verzeichnisse und Unterverzeichnisse an. Verzeichnisse werden seit Windows 95 als Ordner bezeichnet, unterscheiden sich allerdings nicht von Verzeichnissen.

Der `dir`-Befehl hat zahlreiche Optionen:

- `/b` - Nur der Dateiname wird ausgegeben. Die Ausgabe aller weiteren Informationen wie beispielsweise Datum, Größe oder Datenträgerbezeichnung wird unterdrückt.
- `/c` - Die Dateigröße wird mit einem Punkt als Trennzeichen zwischen jeweils drei Dezimalstellen angegeben, also beispielsweise 343.232 (Standardeinstellung). Soll dies unterdrückt werden, geben Sie als Option `/-c` an.
- `/p` - Bei jedem Seitenumbruch wird die Ausgabe unterbrochen, bis der Anwender eine Taste drückt.
- `/on` - Gibt das Verzeichnis alphabetisch sortiert aus.
- `/w` - Mehrere Einträge in einer Zeile werden angezeigt.
- `/ah` - Versteckte Dateien werden angezeigt.
- `/a` - Alle versteckten- und System-Dateien werden angezeigt.
- `/s` - Zeigt auch den Inhalt der Unterverzeichnisse an.

Mit dem Befehl `dir [Dateiname]` ist es auch möglich, Dateien im aktuellen Verzeichnis zu suchen. Alternativ kann auch der Platzhalter `*` (bspw.: `dir *.*`) verwendet werden, um nach bestimmten Dateien zu suchen.

Mit `dir *` werden nur Objekte ohne Dateinamen-Erweiterung angezeigt. In der Regel haben Ordner keine Erweiterung und Dateien eine Erweiterung. In diesem Fall werden nur Ordner angezeigt. Ordner werden im DOS mit vor dem Ordnernamen gekennzeichnet. **2.12.1 Kombinationen**

<pre>dir /w/o/s/p durchsucht den aktuellen Pfad inklusive Unterordner nach einer Datei und stellt das Ergebnis seitenweise dar.</pre>
--

2.13 edit

Mit `edit` wird ein Editor geöffnet, zum Beispiel zum Bearbeiten von Batch²- oder Textdateien. **2.14 exit**

Beendet die Eingabeaufforderung/kehrt zu Windows zurück. **2.15 find**

Mit dem Befehl `find` kann - auch in mehreren Dateien - nach einer Zeichenfolge gesucht werden. Auch `find` kann mit Parametern gesteuert werden: `find /i` ignoriert Groß-/Kleinschreibung beim Suchbegriff `find /v` Zeigt alle Zeilen an, die die Zeichenfolge NICHT enthalten. `find /c` Zeigt nur die Anzahl der die Zeichenfolge enthaltenden Zeilen an. `find /n` Zeigt die Zeilen mit ihren Zeilennummern an.

2.16 format

Formatiert die Festplatte oder Diskette. Mit `format c:` wird die Festplatte `c:` formatiert. Mit `format a:` wird die Diskette im Laufwerk `a:` formatiert. Mit `format c: -s` bzw. `format a: -s` kann man die Festplatte/Diskette mit Systemdateien formatieren, so dass sie nachher bootfähig sind.

2.17 help

Die Hilfe-Funktion steht in allen Windows-Systemen (außer Win9X, d. h. 95/98/ME) zur Verfügung. Der allgemeine Befehl hierzu lautet `help`. Als Ausgabe wird eine Liste mit einem Teil der verfügbaren Kommandozeilen-Befehle ausgegeben. Diese Listen umfasst den Befehlsnamen (linke Spalte) und eine Beschreibung (rechte Spalte). Soll ein Befehl genauer erklärt werden, so lautet die Eingabe `help BEFEHLSNAME` oder auch `BEFEHLSNAME /?`. Diese Eingabe in die Kommandozeile fördert detaillierte Informationen und z.T. auch Beispiele zu einem Befehl zu Tage. Es werden jedoch nicht alle Befehle die es im DOS gibt angezeigt.

2.18 ipconfig

Informationen über die Windows-IP-Konfiguration
Der `ipconfig`-Befehl hat folgende Optionen:

² <http://de.wikibooks.org/wiki/Batch>

- /all - Zeigt zusätzliche Details
- /release - Gibt die zugeteilte IP-Adresse zurück
- /renew - Bezieht eine neue IP-Adresse

2.19 keyb

Lädt Tastatortreiber. `keyb gr,,c:\dos\keyboard.sys` lädt den deutschen Tastatortreiber (`keyboard.sys` muss sich im Verzeichnis `c:\dos\` befinden)

2.20 loadhigh

Lädt Programme in freie UMB Upper-Memory-Blocks

2.21 md

Der Befehl `md` (**m**ake **d**irectory) legt ein neues Verzeichnis an.

2.22 more

Mit der Ausgabeumleitung `| more` (bspw.: `type [Dateiname] | more`) wird bei Seitenwechsel zur Bestätigung aufgefordert.

Damit besteht die Möglichkeit über mehrere Seiten führende Dokumente anzuzeigen.

`more` ist der einzige Befehl, bei der auch in der DOS-Welt die sogenannte Pipe verwendet wird. Bei der Pipe handelt es sich um das Zeichen `|`. Mit diesem Zeichen ist es möglich, zwei Befehle miteinander auf eine spezielle Weise zu verbinden. Dabei wird der erste Befehl ausgeführt, seine Ausgaben werden jedoch nicht auf den Bildschirm geschrieben, sondern in einen Puffer. Der zweite Befehl wird auf eine Weise gestartet, dass er seine Eingaben nicht von der Tastatur, sondern von ebendiesem Puffer bekommt.

Mit `more` ist es ebenfalls möglich den Inhalt einer Datei an eine andere Datei anzuhängen.

So kann man zum Beispiel mit: `more "C:\eins.txt" >> "D:\zwei.txt"` den Inhalt der "eins.txt" an den Inhalt der "zwei.txt" anhängen. "eins.txt" wird dabei nicht verändert.

2.23 move

Verschieben von einer oder mehrerer Dateien von einem Verzeichnis in ein anderes.

oder

ein Verzeichnis umbenennen

```
move test test_neu
```

2.24 netsh

Ohne Parametereingabe ergibt sich ein neuer Kontext. Weitere Hilfe mit: ?

Beispiel : netsh interface dump > ipconf1.dat

Diese Zeile als Batch-Datei ausgeführt legt die momentane IP-Konfiguration in einer Datei namens ipconf1.dat ab. Umgekehrt kann aus einer Datei z. B. namens ipconf2.dat eine neue IP-Konfiguration geladen und übernommen werden:

```
netsh -f ipconf2.dat
```

So lässt sich eine IP-Konfiguration ohne Umherklicken in Systemmenüs ablegen und (modifiziert) wieder zurückholen.

2.25 path

Durch die Angabe eines Pfades wird das Betriebssystem auf der Kommandozeile dazu aufgefordert, neben dem aktuellen Verzeichnis auch in weiteren Verzeichnissen zu suchen. Wird der Befehl ohne Parameter ausgeführt, so werden alle aktuellen Pfade ausgegeben.

2.26 prompt

Legt das Aussehen der Eingabezeile fest.

2.27 rd / rmdir

Der Befehl `rd` oder `rmdir` (remove directory) löscht ein Verzeichnis, wenn dieses leer ist.

Parameter:

- /s Löscht einen ganzen Verzeichnisbaum mit Dateien
- /q Löscht ohne Nachfrage ganze Verzeichnisbäume. Kann in Batch-Dateien praktisch sein.

Der Verzeichnisname darf keine Wildcards enthalten ('*' oder '?').

```
rmdir /s /q C:\temp
```

2.28 ren oder rename

Datei(en) umbenennen.

2.29 set

set setzt eine temporäre Umgebungsvariable oder gibt die Variablenwerte aus. Temporär bedeutet jeweils nur für die aktuelle cmd Session.

2.30 setlocal

setlocal Steuert die Begrenzung des Gültigkeitsbereiches von Änderungen. Wird mit endlocal beendet.

2.31 start

Öffnet neues Fenster für eigenes Programm.

Beispiel: (WinXP) zum öffnen eines neuen Konsolenfensters in dem eine Datei aufgerufen wird.

```
start "Name des neuen Fensters" /WAIT /D C:\temp /NORMAL call test.bat
```

/WAIT = Wartet bis Anwendung geschlossen wird.

/NORMAL = Startet Anwendung in der NORMAL-Prioritätsklasse.

/D = Danach folgt das Startverzeichnis

2.32 taskkill

Ab Windows XP kann man mit diesem Befehl einen Prozess beenden, beispielsweise wenn dieser nicht mehr reagiert. Unter Einbindung von /t werden auch alle untergeordneten Prozesse beendet, und wenn man /f hinzufügt, wird das Beenden erzwungen.

```
taskkill /f iexplore.exe /t
```

2.33 tasklist

Dieser Befehl listet alle aktiven Prozesse auf, die gerade auf diesem Computer laufen. Damit lässt sich zum Beispiel der Name eines zu beendenden Prozesses ermitteln (siehe unten: taskkill). Mit einem hinzugefügten /nh werden die Spaltenüberschriften ausgeblendet.

Der Befehl "tasklist" ist erst ab Windows XP verfügbar.

2.34 time

Gibt die aktuelle Zeit aus und ermöglicht dem Benutzer die Änderung der Uhrzeit. Wird der Befehl mit dem Parameter /t aufgerufen, so wird nur die aktuelle Zeit ausgegeben. time³ kann auch als Variable benutzt werden, so kann man zum Beispiel mit %time:~0,5% die ersten 5 Zeichen übernehmen.

Beispiele:

```
C:\>time
Aktuelle Zeit: 10:03:04,63
Geben Sie die neue Zeit ein:
c:\>echo %time%
10:03:04,63
c:\>echo %time:~0,5%
10:03
```

Bei einigen Betriebssystemversionen erfordert das Ändern der Systemzeit administrative Rechte.

2.35 title

Dieser Befehl ändert die Fensterüberschrift des Programmfensters.

```
title Beispiel
```

2.36 type

Gibt den (reinen Text-)Inhalt der angegebenen Datei (z. B. *.bat, *.cmd oder *.txt) aus.

2.37 Color

Gibt die Farbe an

Farbattribute werden durch ZWEI hexadezimale Ziffern angegeben - die erste bezieht sich auf den Hintergrund, die zweite auf den Vordergrund. Jede Ziffer kann einen der folgenden Werte annehmen:

0 = Schwarz	8 = Dunkelgrau
1 = Dunkelblau	9 = Blau
2 = Dunkelgrün	A = Grün
3 = Blaugrün	B = Zyan
4 = Dunkelrot	C = Rot
5 = Lila	D = Magenta
6 = Ocker	E = Gelb
7 = Hellgrau	F = Weiß

3 Batch Befehle

3.1 Vorsicht Leerzeichen!

Fehlplatzierte oder fehlende Leerzeichen können bei dem Programmieren einer Batch-Datei zu Fehlern führen. Bei den nachfolgenden Beispielen ist also auf die Setzung von Leerzeichen und auf entsprechende Bemerkungen genau zu achten. Scheinbar grundlose Abbrüche beim Ausführen einer Batch-Datei können ebenfalls fehlplatzierten oder fehlenden Leerzeichen geschuldet sein.

3.2 @

Schaltet die Ausgabe der Befehlszeile auf dem Bildschirm nur für den aktuellen Befehl aus und ist selbst kein eigener Befehl.

Syntax

```
@befehl
```

Stapelanweisung: Ab MS-DOS bis Windows 7

Beispiel:

Inhalt

```
echo Diese Zeile wird mit Befehlszeile ausgefuehrt...  
@echo und diese ohne!
```

Ausgabe

```
C:\>echo Diese Zeile wird mit Befehlszeile ausgefuehrt...  
Diese Zeile wird mit Befehlszeile ausgefuehrt...  
  
und diese ohne!
```

In Batch Files verhindert "@echo off" zu Beginn des Skriptes die Ausgabe aller (!) Befehlszeilen auf dem Bildschirm bis die Stapelverarbeitung beendet wird, abbricht oder mittendrin ein "@echo on" Befehl erfolgt, um z. B. Befehlszeilen

tatsächlich anzuzeigen und dann auszuführen. Kommentare (mit `:`¹ oder `REM`²) werden natürlich auch nicht angezeigt. Ist aber nur ein Nebeneffekt.

```
@echo off
REM Verhindert, dass dieser Kommentar angezeigt wird.
```

Ohne `@echo off`:

```
C:\>REM Dieser Kommentar wird so angezeigt
```

3.3 : (Doppelpunkt)

Sprungmarke für ein Unterprogramm bzw. eine Kommentarzeile.

Sprungmarken werden benötigt, wenn mittels der Batchdatei eine Bedingung überprüft und erfüllt bzw nicht erfüllt wird und entsprechend weiter verfahren werden soll.

Mit dem Batchbefehl `goto` wird die Sprungmarke angesprungen.

Anmerkung:

Der Doppelpunkt hat auch die Funktion der Manipulation von Variablen, wenn er direkt hinter einer Variablen steht. Siehe Kapitel "Variablen" in dieser Publikation.

Syntax

```
:NAMEDERSPRUNGMARKE
```

Sprungmarken eine beliebige Länge haben, unter MS-DOS und älteren Windows-Versionen werden allerdings nur die ersten 8 Zeichen beachtet, der Rest wird ignoriert. Kommen in einer Batch also `:Sprungmarke1` und `:Sprungmarke2` vor, so wird unter Umständen nur die erste beim Aufruf einer der Beiden gefunden. Also besser `:ziell` oder `:1st` schreiben. Groß- und Kleinschreibung wird nicht unterschieden.

Beispiel:

Inhalt

1 Kapitel 3.3 auf Seite 18
2 Kapitel 3.15 auf Seite 35

```
if exist C:\blabla.txt goto EDITBLA
goto END

:: Kommentarzeile, sofern es erforderlich ist, einen Kommentar zu schreiben
:: Zur Unterscheidung von Sprungmarken verwende ich zwei ":"
:EDITBLA
edit c:\blabla.txt

:END
```

Sofern die Datei `C:\blabla.txt` existiert, wird sie mit `edit` geöffnet, sonst wird das Unterprogramm übersprungen und die Batchdatei bei der Marke `:END` fortgesetzt, also beendet.

Seit die Befehlsweiterungen aktiviert sind, steht in Batchdateien die Sprungmarke `:EOF` zur Verfügung, welche sich unsichtbar am Ende der Batch-Datei befindet.

3.4 CALL

Mit `call` kann man eine andere Batch-Datei aufrufen. Sobald diese beendet wurde, wird die ursprüngliche Batchdatei weiter ausgeführt.

Beispiel:

```
REM Call.cmd
@echo off
echo Diese Batchdatei ruft eine andere auf.
call anderedatei.bat
pause

REM anderedatei.bat
@echo off
echo Die andere Datei
```

Die Ausgabe wenn man die Datei Call.cmd startet:

```
Diese Batchdatei ruft eine andere auf.
Die andere Datei
{Pause}
```

Wenn die Befehlsweiterungen aktiviert sind (Standard ab Windows 2000) kann man auch Sprungmarken aufrufen (und auch Parameter übergeben):

Beispiel:

```
@echo off
REM Diese Batchdatei ruft eine eigene Sprungmarke auf
echo Vor dem Sprung
call :sprungmarke
```

Batch Befehle

```
echo Nach dem Sprung
pause
goto end

:sprungmarke
echo Sprungmarke aufgerufen!
goto :eof
:: " :EOF" führt nicht zum unsichtbaren Ende der Batch-Datei, wie oben
  beschrieben,
:: sondern führt die Batch-Datei nach dem Aufruf der Sprungmarke fort

:end
exit
```

Die Ausgabe:

```
Vor dem Sprung
Sprungmarke aufgerufen!
Nach dem Sprung
```

Beispiel mit Parameter:

```
@echo off
REM Diese Batchdatei ruft eine eigene Sprungmarke auf
echo Vor dem Sprung
call :sprungmarke meinParameter
echo Nach dem Sprung
pause
goto end

:sprungmarke
echo Sprungmarke aufgerufen und Parameter %1 uebergeben!
goto :eof

:end
exit
```

Die Ausgabe:

```
Vor dem Sprung
Sprungmarke aufgerufen und Parameter meinParameter uebergeben!
Nach dem Sprung
```

```
goto :eof
```

Dieser Befehl springt automatisch zum Ende der Batchdatei (beendet die aktuelle Prozedur)

3.5 CLS

Mit `cls` (clear screen) wird der Bildschirm gelöscht.

Syntax

```
cls
```

Interner Befehl: Ab MS-DOS bis Windows 7

Beispiel:

Inhalt

```
@echo off
echo Hier schreibe ich jetzt ganz viel Text.
echo Hier kann ich z.B. hinschreiben, dass ich jemanden mag.
echo.
echo Aber den Text sieht man gleich sowieso nicht mehr... Hihi!
cls
echo War irgendetwas?
pause>nul
```

Ausgabe

```
War irgendetwas?
```

3.6 CMD

Das Programm, das es möglich macht, DOS-Befehle auf einem NT-System zu benutzen. CMD ist die NT-Entsprechung von COMMAND³.

Syntax:

```
cmd /a|/u|/q|/d|/e (ON|OFF) | /f (ON|OFF) | /v (ON|OFF) | /c
befehl | /s | /k | /y
```

Konsole: Ab Windows NT 4.0 bis Windows NT 6.1 (Windows 7)

Beispiel:

Inhalt

```
cmd
```

Ausgabe (unter XP)

3 Kapitel 3.8 auf Seite 23

```
C:\>cmd
Microsoft Windows XP [Version 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.
```

```
C:\>
```

Ausgabe (unter Vista)

```
C:\>cmd
Microsoft Windows [Version 6.0.6001]
Copyright (c) 2006 Microsoft Corporation. Alle Rechte vorbehalten.
```

```
C:\>
```

Ausgabe (unter Win 7)

```
C:\>cmd
Microsoft Windows [Version 6.1.7600]
Copyright (c) 2009 Microsoft Corporation. Alle Rechte vorbehalten.
```

```
C:\>
```

Ausgabe (unter Win 8 Beta)

```
C:\>cmd
Microsoft Windows [Version 6.2.8250]
Copyright (c) 2012 Microsoft Corporation. Alle Rechte vorbehalten.
```

```
C:\>
```

3.7 COLOR

Mit dem Befehl COLOR kann man die Vorder- und Hintergrundfarbe verändern. Die COLOR Werte bestehen aus zwei HEX-Werten.

0 = Schwarz

1 = Dunkelblau

2 = Dunkelgrün

3 = Blaugrün

4 = Dunkelrot

5 = Lila

6 = Ocker

7 = Hellgrau

8 = Dunkelgrau

9 = Blau

A = Grün

B = Zyan

C = Rot

D = Magenta

E = Gelb

F = Weiß

Der Befehl

```
COLOR 2F
```

ergibt z. B. einen grünen Hintergrund mit weißer Schrift.

3.8 COMMAND

Startet einen neuen Kommandointerpreter. Dieser kann mit `exit` wieder beendet werden.

Syntax:

```
command Laufwerk:Pfad Gerät /e /l /u /P /MSG /LOW (/Y  
(/c|/k) Befehl)
```

Konsole: Ab MS-DOS bis Windows XP

Beispiel:

Inhalt

```
command
```

Ausgabe (unter Windows 95)

```
C:\WINDOWS>command
```

```
Microsoft (R) Windows 95  
(C) Copyright Microsoft Corp 1981-1996.
```

```
C:\WINDOWS>
```

3.9 ECHO

Gibt einen Text aus oder schaltet die Befehlszeilen an/aus. Wenn ein Text ausgegeben wird, können dort auch Variablen⁴ angezeigt werden, wie z. B. die Variable `%ver%` (in Windows XP `%os%`).

Syntax:

```
echo text|ON|OFF oder alternativ echo. [text]
```

Interner Befehl: Ab MS-DOS bis Windows NT 5.1 (XP)

Beispiel:

Inhalt

```
@echo off
echo Die aktuelle Datei heißt %0.
echo Die aktuelle Version Ihrer Befehls-Konsole oder -OS heißt %ver%
```

Ausgabe

```
Die aktuelle Datei heißt beispiel.bat.
Die aktuelle Version Ihrer Befehls-Konsole oder -OS heißt Windows NT
```

Mit `echo.` können Sie zudem leere Zeilen ausgeben. **Beispiel:**

Inhalt

```
@echo off
echo Jetzt gibt es 3 Leere Zeilen zu sehen!
echo.
echo.
echo.
echo So! Da waren sie.
```

Ausgabe

```
Jetzt gibt es 3 Leere Zeilen zu sehen!
```

```
So! Da waren sie.
```

4 Kapitel 3.17 auf Seite 37

3.9.1 Benutzereingaben mit ECHO

Mit Hilfe des `echo`-Befehls können Sie auch in einem Skript Benutzereingaben simulieren, indem Sie den Pipe-Operator `|` verwenden.

Beispiel: Uhrzeit anzeigen ohne Nachfrage

Das normale Verhalten des `time`-Befehls ist, die aktuelle Zeit der verwendeten Systemuhr anzuzeigen und in der nächsten Zeile die Eingabe einer neuen Uhrzeit zu erwarten. Drückt man auf Enter, bleibt die Systemzeit unverändert. Will man die Zeit nur anzeigen lassen (z. B. in einer Batchdatei vor und nach einer Befehlsfolge, um zu messen, wie lange der PC dafür braucht), lässt sich die Betätigung der Enter-Taste durch einen entsprechenden `echo`-Befehl ersetzen.

```
echo.|time
```

Über den Pipe-Mechanismus lässt sich darüber hinaus die Zeile „Geben Sie die neue Uhrzeit ein:“ unterdrücken:

```
echo.|time|find /v "neue"
```

Dabei ist die Groß-/Kleinschreibung von "neue" zu beachten oder der Schalter `/I` zu verwenden, denn `find` ist case-sensitiv! Diese Beispiele dienen allerdings nur zur Demonstration, denn die Zeitausgabe wäre auch ohne Pipes (aber erst ab Windows 2000) möglich mit:

```
time /t
```

oder einfach:

```
echo %time%
```

Sehr nützlich ist das `echo`-Piping auch zur Übergabe von Benutzereingaben, welche von einzelnen Befehlen abgefragt werden.

Beispiel: Überprüfung einer Festplatte

```
chkdsk c: /f /r
```

kann (da es sich beim Laufwerk C um das Systemlaufwerk handelt) erst nach einem Systemstart ausgeführt werden. Normalerweise müsste der Benutzer deswegen den Systemstart durch Eingabe von "Y" bestätigen. Diese Aktion kann man in einem Batch so abbilden:

```
echo y | chkdsk c: /f /r
```

3.10 FOR

Ermöglicht die Schleifenbearbeitung.

Syntax:

```
for Variable in Satz do Befehl [Parameter]
```

Interner Befehl: Ab MS-DOS bis Windows NT 6.1 (Windows 7)

Beispiel:

Zeigt alle Dateien im Verzeichnis *%temp%* an. Es werden nur Dateien, keine Verzeichnisse angezeigt. Um Verzeichnisse anzuzeigen siehe Liste der FOR-Optionen unten. Der Parameter */R* bewirkt, dass alle Unterverzeichnisse mit einbezogen werden (Rekursive Schleife).

ACHTUNG:

Die Variable darf nur aus einem Buchstaben bestehen! *%t* ist erlaubt, *%test* nicht! Bei der Verwendung mehrerer Befehle muss zwischen "DO" und der Klammer "(" ein Leerzeichen sein.

Falsch

```
for Variable in Satz do(
```

RICHTIG

```
for Variable in Satz do (
    Befehl1
    Befehl2
)
```

Inhalt

```
@echo off
for /R %temp% %%f in (*.*) do (
    echo %%f
)
REM Den Befehl könnte man auch einzilig schreiben.
pause
```

Ausgabe

(Alle Temp-Dateien)
Bitte beliebige Taste drücken...

Zählschleifen

Mit solchen Schleifen kann man Aktionen eine bestimmte Anzahl oft ausführen.
Dazu muss man den Parameter /L angeben.

Syntax: for /L {Variable} IN (Startzahl, Schrittweite,
Endzahl) DO (Aktion)

```
REM Schreibe Text 5 Mal
for /L %%N IN (1, 1, 5) DO echo Nummer %%N
```

Ausgabe:

```
C:\>for /L %%N IN (1, 1, 5) DO echo Nummer %%N
```

```
C:\>echo Nummer 1
Nummer 1
```

```
C:\>echo Nummer 2
Nummer 2
```

```
C:\>echo Nummer 3
Nummer 3
```

```
C:\>echo Nummer 4
Nummer 4
```

```
C:\>echo Nummer 5
Nummer 5
```

verschachtelte Zählschleife:

```
for /L %%N IN (1, 1, 5) DO
for /L %%N IN (1, 1, %%N) DO echo Nummer %%N
```

Ausgabe:

```
C:\>for /L %%N IN (1 1 5) DO (for /L %N IN (1 1 %N) DO echo Nummer %N )
```

```
C:\>(for /L %%N IN (1 1 1) DO echo Nummer %N )
```

```
C:\>echo Nummer 1
Nummer 1
```

```
C:\>(for /L %%N IN (1 1 2) DO echo Nummer %N )
```

```
C:\>echo Nummer 1  
Nummer 1
```

```
C:\>echo Nummer 2  
Nummer 2
```

```
C:\>(for /L %%N IN (1 1 3) DO echo Nummer %N )
```

```
C:\>echo Nummer 1  
Nummer 1
```

```
C:\>echo Nummer 2  
Nummer 2
```

```
C:\>echo Nummer 3  
Nummer 3
```

```
C:\>(for /L %%N IN (1 1 4) DO echo Nummer %N )
```

```
C:\>echo Nummer 1  
Nummer 1
```

```
C:\>echo Nummer 2  
Nummer 2
```

```
C:\>echo Nummer 3  
Nummer 3
```

```
C:\>echo Nummer 4  
Nummer 4
```

```
C:\>(for /L %%N IN (1 1 5) DO echo Nummer %N )
```

```
C:\>echo Nummer 1  
Nummer 1
```

```
C:\>echo Nummer 2  
Nummer 2
```

```
C:\>echo Nummer 3  
Nummer 3
```

```
C:\>echo Nummer 4  
Nummer 4
```

```
C:\>echo Nummer 5  
Nummer 5
```

Weitere Möglichkeiten der FOR-Schleife:

syntax-FOR-Files


```
FOR %%parameter IN (set) DO command
```

syntax-FOR-Files-Rooted at Path

```
FOR /R [[drive:]path] %%parameter IN (set) DO command
```

syntax-FOR-Folders

```
FOR /D %%parameter IN (folder_set) DO command
```

syntax-FOR-List of numbers

```
FOR /L %%parameter IN (start,step,end) DO command
```

syntax-FOR-File contents

```
FOR /F ["options"] %%parameter IN (filename_set) DO command
```

```
FOR /F ["options"] %%parameter IN ("Text string to process") DO command
```

syntax-FOR-Command Results

```
FOR /F ["options"] %%parameter IN ('command to process') DO command
```

Beispiel: Sucht im Ordner C:\Windows\Temp rekursiv nach Dateien mit dem Namen //temp.dat// und gibt die Liste aus. Die Option "token=*" ist notwendig, damit die Ausgabe zeilenweise gelesen wird und auch Pfade mit enthaltenem Leerzeichen ausgegeben werden können.

```
for /F "tokens=*" %%f in ('dir /S /b C:\Windows\Temp\temp.dat') do (
echo "%%f".
)
```

Zählvariablen in Zeichenketten einbetten

Um die Zählvariable %%f (%f auf der Kommandozeile) in einer Zeichenfolge zu verwenden, wird einfach die Variable in dem String eingebettet

```
FOR %%f IN (A B C D E) DO (  
    echo mitten%%fdrinnen  
)
```

3.11 GOTO

Mit dem Batchbefehl `goto` wird eine Sprungmarke : (s.o.) angesprungen.

Syntax

```
goto NAMEDERSPRUNGMARKE
```

Beispiel

Siehe unter (Doppelpunkt)⁵.

3.12 IF

Der `IF` Befehl ermöglicht eine einfache Verzweigung und wird oft zusammen mit dem `GOTO`⁶ Befehl eingesetzt. `IF` ermöglicht hierbei sowohl die Prüfung auf eine Gleichheit als auch auf das Vorhandensein von Dateien.

Beispiel 1:

```
@echo off  
IF exist c:\temp\my.log echo.>c:\temp\my.log  
echo.Log Datei erstellt>>c:\temp\my.log
```

Beispiel 1 prüft ob eine Logdatei vorhanden ist und erstellt ggf. eine Neue.

Beispiel 2:

```
@echo off  
IF "%COMPUTERNAME%" == "Bastie" GOTO WAHR  
REM hier landet man wenn der if-Ausdruck falsch ist
```

5 Kapitel 3.3 auf Seite 18

6 Kapitel 3.11 auf Seite 30

```

GOTO WEITER
:WAHR
REM hier landet man wenn der if-Ausdruck wahr ist
  echo Willkommen Zuhause
  REM Jetzt wird der if Zweig verlassen
  GOTO WEITER

:WEITER
echo.Have a nice Day!

```

Beispiel 3:

```

IF "%COMPUTERNAME%" == "Bastie" (
  echo Willkommen zu Hause!
) ELSE (
  echo Du bist auf Computer: %COMPUTERNAME%
)
echo. Schönen Tag noch!

```

Beachten Sie, bei der Prüfung von Umgebungsvariablen niemals

```
IF %Umgebungsvariable% == Prüfwert ...
```

zu schreiben, wenn die Umgebungsvariable nicht gesetzt ist; Sie erhalten sonst einen Syntaxfehler. Der Parameter /i unterbindet eine Differenzierung der Groß-/Kleinbuchstaben.

ACHTUNG:

Bei der Verwendung mehrerer Befehle muss zwischen *Bedingung* und der Klammer "(" ein Leerzeichen sein.

Falsch

```
IF Bedingung(
```

Richtig

```
IF Bedingung (
  Befehl1
  Befehl2
)
```

3.12.1 Syntax Vergleiche

IF <NOT>Variable1==Variable2

IF %Variable% EQU %Variable2% (Befehl) An die Stelle von EQU kann jede der Optionen gesetzt werden.

NOT Der Befehl wird nur ausgeführt, wenn die Bedingung NICHT Wahr ist. Optional.

== ist gleich

EQU ist gleich

NEQ nicht gleich

LSS kleiner als

LEQ kleiner als oder gleich

GTR größer als

GEQ größer als oder gleich

3.12.2 Hinweis zu UND bzw. ODER Verknüpfung

Eine UND bzw. ODER Verknüpfung von zwei Bedingungen scheint nicht direkt möglich zu sein. Beim Vergleichen von Strings hilft es aber eventuell wenn man die beiden Strings miteinander verkettet.

Beispiel

```
set A=true
set B=false
if "%A%;%B%"=="true;true" (
    echo A und B sind beide TRUE
) else (
    echo entweder A oder B sind nicht = TRUE
)
```

Als Workaround können mehrere aufeinanderfolgende IFs zu einer UND bzw. ODER Verknüpfungen kombiniert werden. Bei einer ODER Verknüpfung wird der Code ausgeführt, sobald eine der Bedingungen wahr ist. Wenn alle Bedingungen geprüft wurden, und keine erfolgreich war, werden die Befehle im ELSE Zweig ausgeführt.

```

set A=true
set B=false
if "%A%"=="true" goto :WAHR // Diese Zeile ist doch erfüllt, also sollte der in
:WAHR springen
if "%B%"=="true" goto :WAHR
REM keine der Bedingungen ist zu :WAHR gesprungen, wir sind also im ELSE Zweig
REM hier waere :FALSCH

echo Weder A noch B ist TRUE
goto :eof

:WAHR
echo A oder B ist TRUE

```

Für ein UND wird in den ELSE Zweig gesprungen (:FALSCH) sobald eine der Bedingungen **nicht** zutrifft. Nur wenn alle Bedingungen zutreffen wird der Code ausgeführt.

```

set A=true
set B=false
if "%A%"NEQ"true" goto :FALSCH
if "%B%"NEQ"true" goto :FALSCH

REM wird sind durch die IFs gekommen, also hat keine der Bedingungen
angeschlagen.
REM hier waere die :WAHR Sprungmarke

echo A und B sind beide TRUE
goto :eof

:FALSCH
echo A oder B (oder beide) sind FALSE

```

Diese beiden Beispiele lassen sich einfach durch Kopieren der "IF.." Zeile um beliebig viele Bedingungen erweitern. Ein Mischen von UND und ODER Verknüpfungen ist leider nicht ohne weiteres möglich.

3.13 PAUSE

Unterbricht die Abarbeitung der Batchdatei und wartet auf einen Tastendruck.

Syntax

```
pause
```

Interner Befehl: Ab MS-DOS bis Windows NT 5.1 (XP)

Beispiel 1:

Inhalt

```
@echo off
echo Willkommen zur Batchdatei %0 !!!
echo.
echo Die Batchdatei wird auf einer neuen Seite fortgeführt.
pause
cls
echo Hier fängt meine Batchdatei an...
pause
```

Ausgabe

Willkommen zur Batchdatei beispiel.bat !!!

Die Batchdatei wird auf einer neuen Seite fortgeführt.
Bitte beliebige Taste drücken...

(Neue Seite)

Hier fängt meine Batchdatei an...
Bitte beliebige Taste drücken...

Beispiel 2:

Inhalt

```
@echo off
echo Diese Zeile wird angezeigt anstelle von "Bitte beliebige Taste drücken..."
pause > NUL
```

Ausgabe

Diese Zeile wird angezeigt anstelle von "Bitte beliebige Taste drücken..."

Um eine Pause für eine bestimmte Zeitdauer vorzugeben, kann der ping-Befehl⁷ „missbraucht“ werden. Das folgende Beispiel erzeugt eine Pause von etwa 10 Sekunden (nämlich 11 minus 1); durch die Ausgabeumleitung >nul wird jegliche Meldung unterdrückt.

```
ping -n 11 localhost >nul
```

3.14 PUSH / POP

pushd wechselt zum angegebenen Pfad und speichert den aktuellen Pfad bis zum Aufruf von popd.

7 Kapitel 7.1 auf Seite 65

popd wechselt zum gespeicherten Pfad.

Die Befehle können geschachtelt werden.

Syntax

```
pushd pfad
```

```
popd
```

Beispiel:

```
C:\WINDOWS>pushd c:\temp  
C:\temp>pushd c:\  
C:\>popd  
C:\temp>popd  
C:\WINDOWS>
```

3.15 REM

REM leitet einen Kommentar ein. Die Zeile wird ignoriert, beachten Sie jedoch, dass REM von einem Leerzeichen / Tabulator gefolgt werden muss. Alternativ dazu werden häufig auch Sprungmarken eingesetzt, da hier nur ein statt vier Zeichen verwendet werden muss. Häufig wird jedoch die Sprungmarke zur besseren Übersicht doppelt hintereinander geschrieben ::

Beispiel:

```
REM kill iexplore.exe  
:kill iexplore.exe  
::kill iexplore.exe
```

3.16 START

Startet ein Programm.

Syntax:

```
START ["Titel"] [/D <Pfad>] [/I] [/MIN] [/MAX]  
[/SEPARATE | /SHARED] [/LOW | /NORMAL | /HIGH |  
/REALTIME] [/WAIT] [/B] [Befehl/Programm] [Parameter]
```

Optionen:

- "Titel" Der Titel des neuen Fensters.
- /D <Pfad> Startverzeichnis
- /I Die neue Umgebung soll die dem CMD.EXE beim Aufruf übergebene sein und nicht die aktuelle Umgebung.
- /MIN Startet das Fenster minimiert.
- /MAX Startet das Fenster maximiert.
- /SEPARATE Startet 16-Bit-Windows-Programm in separatem Speicherbereich.
- /SHARED Startet 16-Bit-Windows-Programm in gemeinsamen Speicherbereich.
- /LOW Startet Anwendung in IDLE-Prioritätsklasse.
- /NORMAL Startet Anwendung in der NORMAL-Prioritätsklasse.
- /HIGH Startet Anwendung in der HIGH-Prioritätsklasse.
- /REALTIME Startet Anwendung in der REALTIME-Prioritätsklasse.
- /WAIT Startet die Anwendung und wartet auf das Ende.
- /B Startet die Anwendung ohne ein neues Fenster zu öffnen. Die Anwendung ignoriert STRG+C. Wenn die Anwendung nicht selbständig STRG+C überprüft, ist STRG+UNTB die einzige Möglichkeit, um die Anwendung abzubrechen.
- /? Gibt die Hilfe aus.

Hier eine Liste der nützlichen (System)Programme, die man so ausführen kann:

regedit.exe = neuer Registrierungseditor

(regedt32.exe = älterer Registrierungseditor)

explorer.exe = Windows Ordner Explorer

taskmgr.exe = Windows Taskmanager

taskeng.exe = Aufgabenplanungsmodul

calc.exe = Taschenrechner

mshta.exe = Scripthost für HTA (HTML) Scripting

W/Script.exe = Beides Bestandteil des Windows Scripthostes für JS und VBS

iexplore.exe = Microsoft Windows Internet Explorer, Standardbrowser für Windows

firefox.exe = Mozilla Firefox, Internetbrowser

dialer.exe = Windows-Hilfeprogramm für DFÜ Einwahlverbindungen

Notepad.exe = Standard-Textbearbeitungsprogramm von Microsoft

cmd.exe = Microsoft Windows Befehlsprozessor für Batch

winword = Microsoft Word

3.17 Variablen

Es gibt zweierlei Arten von Variablen: Scriptvariablen und Systemvariablen. Die Scriptvariablen werden innerhalb von Schleifen eingesetzt, beginnen mit % und einem frei wählbaren Zeichen, oder einer Zeichenfolge. Systemvariablen müssen von % begonnen und von % abgeschlossen werden. Sie sind fest definiert. Die Variable %0% hat stets den Namen der aktuellen Datei. Die Variable %ver% hat immer den Namen der aktuellen Betriebssystemversion. Unter Windows XP muss %os% statt %ver% verwendet werden.

Gibt man unter Windows Vista und 7 den Befehl echo %os% ein, so gibt der Computer die Version des Systems aus, auf dem das System basiert. Unter Windows Vista und Windows 7 gibt es einen Befehl namen ver, mit dem man die Version direkt angezeigt bekommt. Es gibt zwar keine direkte Variable bei Vista und Windows 7, die das Betriebssystem definiert, jedoch kann man das mit folgendem Trick umgehen:

```
@echo off          ' Autobefehlsanzeige ausgeschaltet
ver > TMP.dat      ' speichert die Ausgabe des Befehls in der Datei TMP.dat
Set /p ver= < TMP.dat ' definiert die Variable des Betriebssystems. (NUR
TEMPORÄR !)
echo %ver%         ' gibt nun die definierte Variable aus.
pause>nul         ' pause
```

Nützliche Stringoperation für Variablen:

Teilstring

Syntax

```
var:~n,m
```

Ergibt den Teilstring von var, beginnend mit dem n-ten Zeichen (von links) und einer Länge von m Zeichen. Gezählt wird ab 0, d.h. das erste Zeichen hat die Position 0 und nicht 1. Werden negative Werte verwendet, so wird vom Ende des Strings /von rechts nach links) gezählt.

Beispiel:

Inhalt:

```
set str=myfile.bat
set name=%str:~0,6%
echo %name%
```

In diesem Beispiel wird der Teilstring von str vom ersten Zeichen an mit einer Länge von 6 Zeichen ausgegeben.

Ausgabe:

```
myfile
```

Stringsstitution

Syntax

```
var:str1=str2
```

Mithilfe des Syntax var:str1=str2 kann die Zeichenkette str1 des Inhalts der Variable var durch str2 ersetzt werden.

Beispiel:

Inhalt:

```
set str="mycommand /p /m file"
echo %str%
set str=%str:/p /m=/t %
echo %str%
```

In diesem Beispiel werden die Optionen /p /m des Befehls mycommand durch die Option /t ersetzt.

Ausgabe:

```
"mycommand /p /m file"
"mycommand /t file"
```

Benutzereingaben in Variablen speichern

Um eine Benutzereingabe in eine Variable speichern zu können wird die Option /p benötigt

Beispiel:

```
set /p EINGABE=Variable Eingeben:  
echo %EINGABE%
```

Mit Variablen Rechnen

Rechnungen mit Variablen sind erst mit der Option /a möglich

Beispiel:

```
set WERT1=2  
set WERT2=21  
set /a WERT1 = WERT1 * WERT2  
echo %WERT1%
```


4 Batch Operatoren

4.1 &

Befehlsverkettung: mehrere Batch-Befehle in einer Zeile können hintereinander ausgeführt werden

Syntax

```
befehl1 & befehl2
```

Stapelanweisung: Ab ? bis Windows 7

Beispiel:

Inhalt

```
echo Hallo! & echo und Tschüss!
```

Ausgabe

```
Hallo!  
und Tschüss!
```

4.2 |

Befehlsverkettung mittels "pipe": der zweite Befehl bekommt die Ausgabe des ersten Befehls als Eingabe

Syntax

```
befehl1 | befehl2
```

Stapelanweisung: Ab DOS (nur für more als zweiter Befehl); allgemein verwendbar ab WindowsNT bis Windows 7

Beispiel:

Inhalt

```
type test.txt | more
```

Ausgabe

```
Zeile 1  
Zeile 2  
...  
Zeile 24  
-- More --
```

(nach Drücken einer beliebigen Taste werden die nächsten Zeilen angezeigt)

4.3 &&

bedingte Befehlsverkettung: der zweite Befehl wird nur ausgeführt, wenn der erste Befehl erfolgreich war

Syntax

```
befehl1 && befehl2
```

Stapelanweisung: Ab ? bis Windows 7

Beispiel:

Inhalt

```
copy test.txt test2.txt && echo Kopieren erfolgreich!
```

Ausgabe

```
Kopieren erfolgreich!
```

(falls test.txt existiert und nach test2.txt kopiert werden konnte)

4.4 ||

bedingte Befehlsverkettung: der zweite Befehl wird nur ausgeführt, wenn der erste Befehl fehlschlug

Syntax

```
befehl1 || befehl2
```

Stapelanweisung: Ab ? bis Windows 7

Beispiel:

Inhalt

```
copy test.txt test2.txt || echo Fehler beim Kopieren!
```

Ausgabe

Fehler beim Kopieren!

(falls test.txt nicht existiert oder test2.txt nicht geschrieben bzw. überschrieben werden konnte)

4.5 <

Umleitung der Eingabe

Syntax

```
befehl < file
```

Beispiel:

Inhalt

```
echo Hallo! > tmp.txt  
set /P v= < tmp.txt  
echo %v%
```

Ausgabe

Hallo!

(set /P v= würde eine Zeile vom Bediener erwarten; durch die Umleitung wird diese Zeile von der Datei tmp.txt gelesen, die zuvor mittels Umleitung der Ausgabe (siehe unten) befüllt wurde.)

4.6 >

Umleitung der (Standard-)Ausgabe zu einem anderen Ziel. Wenn das Ziel eine Datei ist, wird diese neu angelegt (falls die Datei schon existiert, wird sie zuvor gelöscht) Es kann aber auch nach NUL(Die Ausgabe verschwindet) oder CON(Bildschirm) umgeleitet werden

Syntax

```
befehl > file
```

oder auch mit vorrangestellter Umleitung (praktisch wenn bei der Ausgabe einzelne " vorkommen)

```
> file befehl
```

Beispiel:

Inhalt

```
echo Hallo! > tmp.txt  
type tmp.txt
```

Ausgabe

```
Hallo!
```

Neben der Standard-Ausgabe (**stdout**) gibt es noch eine Fehlerausgabe (**stderr**). Normalerweise landen die Standard-Ausgabe und die Fehler-Ausgabe am Bildschirm, wodurch der Unterschied nicht auffällt. Wenn die Standard-Ausgabe auf eine Datei umgelenkt wird, landen Fehlerausgaben immer noch auf dem Bildschirm, was oft erwünscht ist. Falls nicht, kann auch die Fehler-Ausgabe in eine Datei umgeleitet werden, und zwar mit `2>` .

Beispiel:

```
dir >stdout.txt 2>stderr.txt
```

Schreibt den Output des `dir` Befehles in die Datei *stdout.txt*, da kein Fehler aufgetreten ist.

```
dir DieseDateiExistiertNicht.txt >stdout.txt 2>stderr.txt
```

Schreibt die Fehlermeldung des `dir` Befehles in die Datei *stderr.txt*, da die Datei *DieseDateiExistiertNicht.txt* nicht existiert und somit ein Fehler auftritt.

Wenn die Fehlerausgabe in die gleiche Datei wie die Standardausgabe umgeleitet werden soll, hängt man noch `2>&1` an.

Beispiel:

```
type EineDatei.txt > UmgeleiteteAusgabe.txt 2>&1
```

Der Inhalt der Datei *EineDatei.txt* wird in die Datei *UmgeleiteteAusgabe.txt* kopiert (wenn *EineDatei.txt* existiert).

```
type EineDateiDieNichtExistiert.txt > UmgeleiteteAusgabe.txt 2>&1
```

Wenn die Datei *EineDateiDieNichtExistiert.txt* nicht existiert, dann wird die Fehlermeldung nicht im Dos-Fenster (**stdout**) ausgegeben, sondern die Ausgabe wird auch in die Datei *UmgeleiteteAusgabe.txt* geschrieben.

4.7 >>

Umleitung der (Standard-)Ausgabe mit Anhängen des Textes (falls die Datei schon existiert; sonst wird die Datei wie bei > (siehe oben) angelegt)

Syntax

```
befehl >> file
```

Stapelanweisung: Ab ? bis Windows 7

Beispiel:

Inhalt

```
echo Hallo! > tmp.txt
echo Haalooo!! >> tmp.txt
type tmp.txt
```

Ausgabe

```
Hallo!
Haalooo!!
```

Hier gilt analoges für die Fehler-Ausgabe (siehe Operator '>' oben).

5 Erweiterungen unter Windows NT

Unter Windows NT, Windows 2000 und Windows XP stehen einige weitere Befehle zur Verfügung, zum Teil sind dort auch erweiterte Optionen für Standard-Befehle enthalten.

5.1 Batchparameter

Die Parameter lassen sich jetzt folgendermaßen erweitern (für 1 setze man den jeweiligen Parameter ein):

```
%~1  Anführungszeichen (") werden entfernt
%~f1 vollständige Pfadbezeichnung
%~d1 Laufwerksbuchstabe
%~p1 Pfad (ohne Laufwerksbuchstabe)
%~n1 Dateiname
%~x1 Dateinamenserweiterung
%~s1 Pfad nur mit kurzen Verzeichnis/Dateinamen
([http://de.wikipedia.org/wiki/8.3 8.3-Konvention])
%~a1 Dateiattribute
%~t1 Datums- und Uhrzeitangaben der Datei (Geändert am)
%~z1 Größe der Datei
```

Die Angaben können auch kombiniert werden, z.B. `%~dp0` ist Laufwerksbuchstaben + Pfad der aktuellen Batchdatei.

5.2 Stringmanipulation

Mit folgender Syntax lassen sich Teile aus einer Zeichenkette extrahieren:

```
set str=Hallo
set str=%str:~1%
echo.%str%
```

Die angegebene positive Zahl gibt an, wie viele Zeichen links übergangen werden sollen. Dieser Ausdruck liefert daher den Teilstring "allo" von "Hallo".

```
set str=Hallo
set str=%str:~-4%
echo.%str%
```

Wird eine negative Zahl eingesetzt, so gibt ihr Betrag an, wieviele Zeichen von rechts genommen werden sollen. Dieser Ausdruck liefert daher ebenfalls "allo".

Über die Angabe einer zweiten Zahl kann definiert werden, was mit dem Rest nach Auswertung der ersten Zahl entsprechend obiger Regeln geschehen soll. Während die erste Zahl die Zeichenkette links beschneidet, beschneidet die zweite Zahl sie rechts und zwar nach folgenden Regeln:

```
set str=Hallo
set str=%str:~1,2%
echo.%str%
```

Eine positive zweite Zahl gibt an wie viele Zeichen ab links vom Rest genommen werden sollen. Dieser Ausdruck liefert daher "al".

```
set str=Hallo
set str=%str:~1,-2%
echo.%str%
```

Wird eine negative zweite Zahl eingesetzt, so gibt ihr Betrag an, wie viele Zeichen vom Rest ab rechts übergangen werden sollen. Dieser Ausdruck liefert daher ebenfalls "al".

Möchte man beispielsweise nur einen Teil der Zeichenkette ab links, so ist nach diesen Regeln als erste Zahl eine Null anzugeben:

```
set str=Hallo
set str=%str:~0,2%
echo.%str%
```

Dieser Ausdruck liefert daher "Ha".

Mehr zu Stringmanipulation¹

5.3 Umgebungsvariablen

Unter DOS und Windows 9x/ME speichern alle Anwendungen und Skripte ihre Umgebungsvariablen in der gleichen Umgebung. Unter Windows NT-basierten Betriebssystemen ist dies etwas anders:

Zunächst gibt es einen systemweiten Bereich. Dieser enthält Variablen, die für alle Benutzer und alle Anwendungen zugänglich sind. Zusätzlich hat jeder Benutzer seinen eigenen Variablenraum. Jede Anwendung, die nun vom System gestartet wird, erhält einen eigenen Variablenraum, der als Kopie aus dem der aufrufenden Anwendung erzeugt wird. Ändert das Programm etwas (Wertänderung, neue Variable), wirkt sich dies nicht auf parallel gestartete Anwendungen oder das System aus. Lediglich Prozesse, die von der Anwendung gestartet werden erhalten eine Kopie des Anwendungsvariablenraumes. Dadurch ist es nicht möglich, dass Anwendungen Umgebungsvariablen setzen, die dann anschließend in einer Batchdatei verwertbar sind.

Beispiel:

```
@echo off
set a=2
echo %a%
command /c: set a=1
echo %a%
```

Unter Dos würde die Ausgabe lauten:

```
2
1
```

Unter Windows NT/2000/XP:

```
2
2
```

¹ http://www.dostips.com/DtTipsStringManipulation.php#_Toc135152735

Das heißt nicht, dass unter Windows NT und Nachfolgern der Befehl `set a=1` keinen Effekt hätte, sondern dieser wird in einer geschützten Umgebung ausgeführt, die dann außerhalb dieser Umgebung keinen Effekt mehr hat. Für die aufgerufene Variante von `command` ist `a` nach dem `set`-Befehl sehr wohl gleich 1.

Noch eine Änderung Variablen betreffend.²

5.4 Die Kommandozeile lernt Rechnen

Seit Windows NT 4 kann man mit dem `set` Befehl auch Rechnen.

Ein Beispiel:

```
set /a 1+2
3
```

Man kann das Ergebnis auch in Variablen schreiben:

```
set /a n=5+5
10
echo %n%
10
```

- CMD nimmt folgendes entgegen (Auszug aus der Hilfe):

<code>()</code>	- Gruppierung
<code>! ~ -</code>	- unäre Operatoren
<code>* / %</code>	- arithmetische Operatoren (<code>%</code> bedeutet Modulo)
<code>+ -</code>	- arithmetische Operatoren
<code><< >></code>	- logische Verschiebung
<code>&</code>	- bitweise UND
<code>^</code>	- bitweise exklusives ODER
<code> </code>	- bitweise ODER
<code>= *= /= %= += -=</code>	- Zuordnung

- Mit Klammern:

```
set /a (1+1)*(4-3)
2
```

² http://de.wikibooks.org/wiki/Batch-Programmierung:_Programmierungshilfen#Probleme_mit_Variablen

5.5 Benutzereingaben

Seit Windows 2000 kann man mit dem `set` Befehl auch Eingaben abfragen:

Syntax:

```
set /p {variablenname}={Angezeigter Text}
```

5.5.1 Beispiel

```
@echo off
set /p var=Eingabe:
echo Sie haben %var% eingegeben.
pause
```

Ausgabe:

```
Eingabe: Test
Sie haben Test eingegeben.
{Pause}
```

Prüfung ob der Benutzer Text eingegeben hat:

```
@echo off
set /p var=Eingabe:
if not defined var (
    echo Bitte geben Sie etwas ein!
    set var=ichts
    REM Könnte auch was ganz anderes sein
)
echo Sie haben "%var%" eingegeben.
pause
```

Ausgabe:

```
Eingabe: {leer}
Bitte geben Sie etwas ein!
Sie haben "ichts" eingegeben.
{pause}
```

5.6 Ausgaben in eine Variable umleiten

Eine Möglichkeit der Umleitung einer Ausgabe besteht über den `set /p` Befehl:

Syntax:

```
ECHO {Befehl}> {Dateiname}  
SET /P {Variablenname}=<{Dateiname}
```

5.6.1 Beispiel

```
@ECHO OFF  
ECHO %date% > _.txt  
SET /P tmp=<_.txt  
ECHO Das Datum heute ist: %tmp%
```

- siehe auch Net-Befehle³

3 Kapitel 7.5 auf Seite 69

6 Programmierungshilfen

6.0.2 Ändern des Editors zum Bearbeiten von Batchdateien

Wenn man im Windows Explorer mit der Rechten-Maus-Taste (RMT) auf eine *.bat klickt, so werden einem die Befehle *Öffnen* und *Bearbeiten* angeboten.

Öffnen: führt die Batchdatei aus. Mit dem Befehl

Bearbeiten: wird die Batchdatei in den Texteditor *notepad.exe* zum Bearbeiten geöffnet.

Auch wenn Notepad zum Bearbeiten von Batchdateien ausreicht, so möchte man häufig doch die Batchdateien mit einem anderen, komfortableren Editor bearbeiten, der z. B. Syntaxhervorhebung (Syntaxhighlighting) beherrscht.

Um einen anderen Editor (z. B. Syn¹) zu verwenden, muss man in der Registry an der Stelle:

```
HKEY_CLASSES_ROOT\batfile\shell\edit\command
```

den Standard Wert

```
(Standard) = %SystemRoot%\System32\notepad.exe %1
```

auf den Startbefehl des entsprechenden Editors ändern. Z. B.

```
HKEY_CLASSES_ROOT\batfile\shell\edit\command | (Standard) =  
c:\Programme\Editor\syn\syn.exe %1
```

1 <http://syn.sourceforge.net/>

6.0.3 Datum und Uhrzeit anzeigen

Für Log-Dateien ist es wichtig, dass man die Logeinträge mit Datum und Uhrzeit versehen kann:

```
echo %date:~0% - %time:~0,8% Uhr
```

Ergebnis: **11.09.2010 - 15:59:53 Uhr**

Hierbei steht ":~0,8" für die Angabe der Stellen. Mit "0,8" wird angegeben, dass die Ausgabe der Zeit bei Position 0 beginnen soll und insgesamt 8 Stellen beinhalten soll. Die maximale Stellenanzahl ist 11 (0,11).

Beim Datum ist diese Angabe hier nicht nötig, da dieses standardmäßig im dd.mm.yyyy-Format ausgegeben wird. Wer aber nur das Jahr haben will, kann "%date:~4%" eingeben und erhält damit die letzten 4 Zeichen. Für ein sortiergerechtes Datum in der Umgebungsvariablen sortdate sorgt z. B.

```
set SORTDATE=%date:~4%-%date:~3,2%-%date:~0,2%
echo %SORTDATE%
```

Wert der Umgebungsvariable: **2009-04-20**

Beachte:

In einer Batchdatei kann die Verwendung von %DATE% und insbesondere von %TIME% dazu führen, daß sich die ausgegebene Uhrzeit nicht aktualisiert.

Hierzu folgendes Beispiel:

```
@echo off
echo ## Die aktuelle Zeit ist: %TIME%
echo ## bitte ca. 5 Sec. warten ...
ping -n 5 localhost>NUL
echo ## jetzt sollten ca. 5 Sec. vergangen sein, TIME liefert %TIME%, das ist
    noch OK
echo ## doch in der FOR Schleife wird bereits die alte Zeit verwendet.
for /L %%N IN (0, 1, 3) DO (
    echo %time%
    pause
)
echo ## und dies bleibt für jede Ausgabe innerhalb der FOR-Schleife so.
echo.
echo ## Auch in z.&nbsp;B. IF-Schleifen ist das so.
```

```
if TRUE==TRUE (
    echo 1. Zeit in der If Schleife: %TIME%
    echo Warte ca. 5 Sec.
    ping -n 5 localhost >NUL
    echo 2. Zeit in der If Schleife: %TIME%
    echo Warte nochmals ca. 5 Sec.
    ping -n 5 localhost >NUL
    echo 3. Zeit in der If Schleife: %TIME%
)
echo ## Dabei ist es bereits: %TIME%
pause
```

Damit `%DATE%` und `%TIME%` die richtigen Werte ausgeben, muss unbedingt die verzögerte Erweiterung von Umgebungsvariablen mit dem Befehl `SETLOCAL2 ENABLEEXTENSIONS` aktiviert werden.

Hier das korrekte Beispiel:

```
@echo off
SetLocal EnableDelayedExpansion
echo ## Die aktuelle Zeit ist: %time:~0,8%
echo ## bitte ca. 5 Sec. warten ...
ping -n 5 localhost>NUL
echo ## jetzt sollten ca. 5 Sec. vergangen sein, TIME liefert %TIME%, das ist OK
echo ## Jetzt gibt auch die FOR Schleife die korrekte Zeit aus.
for /L %%N IN (0, 1, 3) DO (
    echo !TIME!
    pause
)
echo.
echo ## Auch in z.&nbsp;B. IF-Schleifen ist es jetzt richtig.
if TRUE==TRUE (
    echo 1. Zeit in der If Schleife: %TIME%
    echo Warte ca. 5 Sec.
    ping -n 5 localhost >NUL
    echo 2. Zeit in der If Schleife: %TIME%
    echo Warte nochmals ca. 5 Sec.
    ping -n 5 localhost >NUL
    echo 3. Zeit in der If Schleife: %TIME%
)
echo ## Es ist jetzt: %TIME%
EndLocal
pause
```

6.0.4 Ausgaben besser anzeigen

Wenn man nicht die Ausgabe von Befehlen per `@echo off` "Ausblendet" kann man am besten das Prompt ändern, sodass man besser erkennen kann, was passiert:

```
@prompt -$G
```

Der Prompt ist dann ->

6.0.5 Unterroutinen und Unterprogramme

Unterroutinen kann man mittels **goto** oder **call** und Unterprogramme mit Hilfe von **call** realisieren.

```
call:unterroutine Hallo
echo Fertig!
goto:eof

:unterroutine
    echo Übergebener Parameter an Unterroutine: %1
goto:eof
```

Beachten Sie, dass Sie beim Aufruf von Unterroutinen per `call` Probleme mit Filehandles bekommen können. Dies liegt darin begründet, dass ein Aufruf per `call` als Aufruf eines Unterprogramms interpretiert wird, während es sich bei `goto` stets um Unterroutines handelt.

Anmerkung: `goto:eof` ist eine Spezialmarke mit der Sie stets zum Ende Ihres Skriptes (bzw. Ihrer Unterroutine) springen

6.0.6 Benutzereingaben mittels "set /P"

```
@echo off
    set /P w= [i]nstallieren / [d]eInstallieren?
    REM die option /I beim if bewirkt, dass nicht
    REM zwischen Gross und Kleinschreibung
    REM unterschieden wird.
    if /I "%w%"=="i" goto Install
    if /I "%w%"=="d" goto Deinstall
    echo Fehler: [%w%]
goto ende

:Install
    echo "installieren" ausgewählt
goto ende

:Deinstall
    echo "deInstallieren" ausgewählt
goto ende
```

```
:ende
  echo.
  pause
```

6.0.7 stdout in Umgebungsvariable speichern

Falls man den **stdout** in einer Umgebungsvariable speichern möchte, muss man das komplizierter umsetzen. Es gibt zwei verschiedene Möglichkeiten dies anzugehen. `befehl | set /P variable=` funktioniert nämlich nicht. Stattdessen braucht man:

```
BEFEHL > temp.txt
set /p BefehlOutput= < temp.txt
del temp.txt
```

Oder:

```
FOR /F %%i IN ('BEFEHL') DO set BefehlOutput=%%i
```

Oder mit "usebackq"-Option:

```
FOR /F "usebackq" %%i IN ('BEFEHL') DO set BefehlOutput=%%i
```

Die Zeichenkette zwischen den einfachen Anführungszeichen wird dabei als Befehlszeile betrachtet und von einer untergeordneten CMD.EXE ausgeführt. `%BefehlOutput%` kann nun beliebig gebraucht werden.

Beispiel:

Code:

```
@echo off
FOR /F %%i IN ('CD') DO set verzeichnis=%%i
echo %verzeichnis%
```

Ausgabe:

```
C:\Programme\Batch
```

Vorsicht ist geboten, bei Befehlen, welche mehrzeilige Ausgaben produzieren und bei solchen, welche in ihrer Ausgabe auch Leerzeichen enthalten können. Da das Standardtrennzeichen ein Blank ist muss man, wenn man nicht will, dass die Variable nur bis zum Blank gefüllt wird, das Standardtrennzeichen verändern. `FOR /F "delims=" %%i IN ('CD') DO set verzeichnis=%%i` entfernt jede Art von Trennzeichen. Bei Befehlen, welche mehrzeilige Ausgaben zur Folge haben, bleibt jeweils die letzte Zeile in der Variablen erhalten.

6.0.8 Dateien und Verzeichnisse auflisten

Hier ist ein Beispiel, in dem alle Dateien, auf welche die Filterbedingung zutrifft, aufgelistet werden. Ausserdem werden die Dateianzahl und die Dateigrößen zusammenaddiert.

```
@echo off
set Filter=*. *
set /A DateiAnzahl=0
set bytes=0

for /R %pfad% %%f in (%Filter%) do (
    set /A DateiAnzahl += 1
    echo %%f - %%~zfBytes
    set /A bytes=bytes+%%~zf
)

echo.
echo %~dp0%Filter%
echo Es sind %DateiAnzahl% Dateien vorhanden.
echo Alle Dateien zusammen: %bytes%Bytes
set /A kbytes=bytes/1024
echo umgerechnet sind das %kbytes% KBytes
echo.
pause
```

6.0.9 Pause

Oft ist es hilfreich, dass ein nach dem Beenden des Batch-Programms das Eingabeaufforderungsfenster offen bleibt. So kann man Ausgaben nachlesen oder evtl. aufgetretene Fehler entdecken. Nun könnte man einfach am Ende eine **pause** einfügen. Dabei kann der User einfach das Fenster schließen oder ENTER drücken. Man kann aber auch einfach eine zeitliche Pause mit **ping** realisieren:

```
@echo off
echo Ich schlieÙe gleich.
```

```
@ping localhost -n 2 >NUL
```

Dabei kann man die Zeit mit dem Parameter `-n` variieren.

Bei installiertem `ResourceKit`³ steht der Befehl `"sleep"` zur Verfügung welcher dieselbe Funktionalität (zeitliche Pause) bietet.

6.0.10 Minimiert ausführen

Hin und wieder ist es sinnvoll, dass die Batchdatei minimiert ausgeführt wird (z. B. eine Login-Batch-Datei). Es ist möglich, dass man die Batchdatei normal startet und sie sich selber minimiert ausführt. Der Nachteil ist allerdings, dass sich kurzzeitig ein Eingabeaufforderungs-Fenster öffnet.

```
@echo off
if not "%1"==" " goto %1

start /MIN cmd.exe /C "%~nx0 begin"
goto:eof

:begin
    echo Hallo, ich laufe minimiert!
    pause
goto:eof
```

Noch eine Konstruktion ganz ohne Labels, nach diesem Newsgroup-Beitrag⁴

```
@set !=||(set !=1&start "%~dpnx0" /min cmd /c %0 %*&set !=&goto :eof)
```

Anmerkung: Sollte die Command-Processor-Option `"DelayedExpansion"` in der Registry aktiviert sein (siehe unter Hilfe `"cmd /?"`), lässt sich ein `"!"` als Variablenname **nicht** verwenden. In diesem Fall -bzw. sinnvollerweise immer- den Variablennamen ändern auf `x` oder `y` oder `#` oder `@....`

Beispiel:

-
- 3 http://www.google.de/url?sa=t&source=web&ct=res&cd=1&ved=0CAgQFjAA&url=http%3A%2F%2Fwww.microsoft.com%2Fdownloads%2Fdetails.aspx%3FFamilyID%3D9D467A69-57FF-4AE7-96EE-B18C4790CFFD&rct=j&q=ms+resource+kit&ei=SEtGS8H6FYfM-Qai2tRt&usg=AFQjCNEU130wz1C_161SaIBsG99a7Lccrg
 - 4 <http://groups.google.co.uk/group/alt.msdos.batch.nt/msg/d11e951a822bc420?dmode=source>

```
@set #=|| (set #=1&start "%~dpnx0" /min cmd /c %0 %*&set #=&goto :eof)
```

6.0.11 Mittels `start /LOW` die Priorität festlegen

Manchmal ist es hilfreich, wenn die Batchdatei mit einer niedrigen Priorität läuft. Das kann man mittels `start /LOW` erreichen. Weitere Optionen sind **NORMAL**, **HIGH**, **REALTIME**, **ABOVENORMAL** und **BELOWNORMAL**. Das Beispiel zeigt, wie eine Batchdatei quasi sich selber in die niedrige Priorität versetzen kann. In dem Fall klappt es allerdings nur, wenn beim ersten Start kein Parameter übergeben wurde.

```
@echo off
if "%1"==" " (
    start /WAIT /LOW /B cmd.exe /V /C %~s0 weiter_machen
    goto:eof
)
echo Jetzt laufe ich mit niedriger Priorität!
echo Überprüfe es im Taskmanager!
pause
```

Funktionsweise: Das Prinzip ist eigentlich ganz einfach. Wenn kein Parameter übergeben wird, wird angenommen, daß die Batchdatei zum ersten mal gestartet wurde. Die `if "%1"==" "` Bedingung ist also erfüllt. Mittels `start` wird dann dieselbe Batchdatei mit veränderter Priorität gestartet, allerdings mit einem angehängten Parameter **weiter_machen** (Könnte auch irgendwas anderes sein!) Somit ist beim nächsten Aufruf die `if "%1"==" "` Bedingung nicht mehr erfüllt und der normale Teil der Batchdatei wird abgearbeitet.

Wenn man der Batchdatei einen Parameter übergeben möchte (z. B. ein Dateiname o.ä.) muss man alle Parameter verschieben:

```
@echo off
if "%2"==" " (
    start /WAIT /LOW /B cmd.exe /V /C %~s0 %1 weiter_machen
    goto:eof
)
echo Jetzt laufe ich mit niedriger Priorität!
echo Nun kann [%1] 'bearbeitet' werden...
pause
```

Anmerkungen: Normalerweise könnte man statt `%~s0` auch `"%~0"` bei der `cmd.exe` Zeile verwenden. Allerdings klappt das nicht richtig, wenn Leerzeichen in der

Batch Datei vorhanden sind. Mit %~s0 wird der komplette Pfad zur Batchdatei als "Kurznamen" angegeben. In dem Pfad kommt dann keine Leerzeichen vor.

6.0.12 Probleme mit Variablen

Wenn man sich die Hilfeseiten zu **set** (mittels **set /?**) durchliest, stößt man auf das Thema *verzögerte Erweiterung von Variablen*. Das will ich hier mal anhand von Beispielen erklären:

Das Problem

```
set test=1
if "%test%"=="1" (
    set test=2
    echo Wert von 'test' im IF-Block: %test%
)
echo Wert von 'test' nach IF-Block: %test%
```

Man sollte meinen, dass der Wert von %test% in beiden Ausgaben **2** ist. Doch *leider* ist es nicht so. Denn innerhalb des IF-Blocks wird das Neusetzen der Variable test von 1 auf 2 noch nicht aktiv und somit ist das Ergebnis **Wert von 'test' im IF-Block: 1** Erst nach dem IF-Block ist der Wert aktualisiert: **Wert von 'test' nach IF-Block: 2**

Lösung: cmd.exe /V

In einer Batchdatei die mit **cmd /V** gestartet wurde, werden Variablen innerhalb von Befehlsblöcken aktualisiert. Jedoch kann man sie nicht gewohnt mit **%test%** ansprechen, sondern mit **!test!**

```
@echo off
if "%1"==" " (
    start /WAIT /B cmd.exe /V /C "%~0" machen!
    goto:eof
)
set test=1
if "%test%"=="1" (
    set test=2
    echo Wert von 'test'-Prozent in dem IF-Block: %test%
    echo Wert von 'test'-Ausrufezeichen in dem IF-Block: !test!
)
echo Wert von 'test' nach dem IF-Block: %test%
```

pause

6.0.13 Ausgaben/Fehler unterdrücken

Manchmal möchte man per Batch ein Programm starten aber es soll dabei keine Ausgabe gemacht werden. Das ist recht einfach:

```
MeinProgramm.exe >NUL
```

Es könnte aber sein, dass evtl. Fehler dennoch ausgegeben werden. Das liegt daran, daß die Programme in dem Fall auf **stderr** statt **stdout** schreiben. Um auch in dem Fall die Ausgabe zu unterdrücken, kann man mit einem zusätzlichen **2>&1** die Ausgaben von stderr auf stdout umleiten. Da stdout dann nach **NUL** verschoben wird, sieht man absolut nichts:

```
MeinProgramm.exe >NUL 2>&1
```

6.0.14 Professionelle Message-Fenster erzeugen

Bisher konnte man in Batch keine Fenster erzeugen, bzw. nur in Windows XP mithilfe des Windows Nachrichtendienstes. Doch der ist für Anwendungen viel zu unpraktisch, da man nicht einmal den Fenstertitel bestimmen kann und außerdem ist dieser unter Vista oder Windows 2000 nicht verfügbar.

Ich habe jedoch ein kleines Schlupfloch gefunden, mit dem man dennoch Fenster erzeugen kann:

```
@echo off
Echo msgbox"Text",0,"Fenstername" >Test.vbs
ping localhost -n 3 -w 1000 >NUL
start Test.vbs
pause
```

Erklärung: Man erzeugt hier mithilfe des Umleitungssymbols (Eng.: pipe) > einen Temporären VBScript, der in der Lage ist, ein Messagefenster zu erzeugen. Dies sorgt in Anwendungen für mehr Übersicht und Professionalität.

6.0.15 Falls das nicht funktioniert

Falls statt eures Textes die Meldung :

"Der Zugriff auf den Windows ScriptHost ist auf diesem Computer deaktiviert"... erscheint, dann ist das Öffnen von VBscripts und JScripts aus Sicherheitsgründen verboten.

Um das zu ändern, öffnet ihr den Windows Registrierungseditor (regedit.exe) und löscht NUR folgenden Wert :

HKey_Local_Machine\Software\Microsoft\WindowsScriptHost\Enabled

dann dürfte es funktionieren. **Erstellt vor der Änderung eine Sicherheitskopie eurer Werte !!!**

6.0.16 Status über bearbeitete Zeilen ausgeben

Häufig bearbeitet man eine Liste von Objekten mit einer FOR-Schleife. Damit man darüber informiert ist wie weit die Bearbeitung bereits fortgeschritten ist, kann man die Anzahl der Bearbeiteten Objekte im titel der DOS-Box ausgeben lassen.

Schritt 1 - Ermitteln / zählen der zu bearbeitenden Zeilen: dies geht am besten mit folgender FOR-Schleife:

```
FOR /F "eol=# tokens=1,2,3 " %A IN ('find /c ";"liste.txt') DO echo %C Zeilen
```

Wobei das ";" ein Zeichen sein muss das in jeder zu verarbeitenden Zeile vorkommt. Dies ist am schnellsten. Will man wirklich jede Zeile zählen unabhängig vom Inhalt, dann kann man auch folgenden Befehl verwenden:

```
FOR /F "delims=" %%A IN ('findstr /N .* "liste.txt"') DO echo %%A Zeilen
```

Jetzt bedarf es noch des SET-Befehl zum Berechnen der aktuellen Zeile und setLocal EnableDelayedExpansion, dann könnte das Script folgendermaßen aussehen:

```
@echo off
setLocal EnableDelayedExpansion
set COUNT=0
set COUNTMAX=0
```

```
FOR /F "eol=# tokens=1,2,3 " %%A IN ('find /c ";"liste.txt') DO set
COUNTMAX=%%C

for /f "eol=# tokens=1 delims=;" %%j in (liste.txt) do (
set /A COUNT +=1
title %0 - !COUNT! Zeilen von !COUNTMAX! bearbeitet
echo Tue etwas mit dem Token %%j
REM der ping wird nur zur Verzögerung ausgeführt damit man der Titel Zeile
besser verfolgen kann.
ping -n 1 localhost >NUL
)
```

7 Netzwerkumgebung

7.1 ping

Kann verwendet werden, um die Verbindung zu einem anderen Rechner zu testen. Wenn die Gegenseite antwortet, werden Informationen über die IP-Nummer und zur Verbindungsqualität angezeigt. Zu beachten ist dabei, dass aus einer ausbleibenden Antwort nicht zwangsläufig auf einen missglückten Verbindungsversuch geschlossen werden darf: manche Administratoren unterbinden diese Art Test aus Sicherheitsgründen.

Beispiel

```
C:\>ping de.wikibooks.org
```

Ausgabe

```
Ping rr.knams.wikimedia.org [145.97.39.155] mit 32 Bytes Daten:

Antwort von 145.97.39.155: Bytes=32 Zeit=67ms TTL=57
Antwort von 145.97.39.155: Bytes=32 Zeit=35ms TTL=57
Antwort von 145.97.39.155: Bytes=32 Zeit=34ms TTL=57
Antwort von 145.97.39.155: Bytes=32 Zeit=39ms TTL=57

Ping-Statistik für 145.97.39.155:
    Pakete: Gesendet = 4, Empfangen = 4, Verloren = 0 (0% Verlust),
    Ca. Zeitangaben in Millisek.:
        Minimum = 34ms, Maximum = 67ms, Mittelwert = 43ms

C:\>
```

Hinweis

Dies lässt sich durch einen Trick mit dem find Befehl auch automatisieren:

```
ping de.wikibooks.org | find "TTL" || goto hostnotfound
```

7.2 ipconfig

- Durch den Parameter */help* erhält man kurze Hilfe.
- Der Aufruf des Befehls *ipconfig* ohne Parameter teilt die eigene IP-Adresse mit.
- Durch den Befehl *ipconfig* lässt sich die IP-Adresse von Netzwerkverbindungen (Ethernetkarten) verwalten.

Beispiel

```
C:\>ipconfig
```

Ausgabe

```
Windows-IP-Konfiguration
```

```
Ethernetadapter VMware Network Adapter VMnet8:
```

```
Verbindungsspezifisches DNS-Suffix:  
IP-Adresse. . . . . : 192.168.0.1  
Subnetzmaske. . . . . : 255.255.255.0  
Standardgateway . . . . . : 192.168.0.11
```

```
Ethernetadapter VMware Network Adapter VMnet1:
```

```
Verbindungsspezifisches DNS-Suffix:  
IP-Adresse. . . . . : 192.168.2.1  
Subnetzmaske. . . . . : 255.255.255.0  
Standardgateway . . . . . : 192.168.2.11
```

```
Ethernetadapter LAN-Verbindung 3:
```

```
Verbindungsspezifisches DNS-Suffix: myhost.local  
IP-Adresse. . . . . : 192.168.1.1  
Subnetzmaske. . . . . : 255.255.255.0  
Standardgateway . . . . . : 192.168.1.11
```

```
C:\>
```

7.2.1 ipconfig /all

Eine detailliertere Auskunft erhält man mit dem Parameter */all*

Beispiel

```
C:\>ipconfig /all
```

Ausgabe

```
Windows-IP-Konfiguration

    Hostname. . . . . : cestmoi
    Primäres DNS-Suffix . . . . . : mypc.local
    Knotentyp . . . . . : Unbekannt
    IP-Routing aktiviert. . . . . : Nein
    WINS-Proxy aktiviert. . . . . : Nein
    DNS-Suffixsuchliste . . . . . : mypc.local
                                   mypc.local

Ethernetadapter VMware Network Adapter VMnet8:

    Verbindungsspezifisches DNS-Suffix:
    Beschreibung. . . . . : VMware Virtual Ethernet Adapter for
VMnet8
    Physikalische Adresse . . . . . : 01-23-45-67-89-AB
    DHCP aktiviert. . . . . : Nein
    IP-Adresse. . . . . : 192.168.0.1
[ ... ]
```

7.2.2 ipconfig /renew

Sollten die Netzwerkkonfigurationen nicht mehr aufzufinden sein, dann hilft oft der Parameter `/renew`

Beispiel

```
C:\> ipconfig /renew
```

7.2.3 ipconfig /flushdns

Situation

Eine Homepage wird nicht angezeigt, stattdessen wird eine Fehlermeldung angezeigt, wie z.B. *"Die Website könnte vorübergehend nicht erreichbar sein, versuchen Sie es bitte später nochmals."*

Als Test wird der Server angepingt, und es kommt eine Fehlermeldung

Zeitüberschreitung der Anforderung.

Der Administrator kümmert sich um das Problem und sagt, der Server sei wieder OK, aber noch immer kommen Fehlermeldungen und Homepages können nicht angezeigt werden. Dann kann es daran liegen, dass der DNS-Cache noch die Fehlmeldung gespeichert hat, und dieser muss geleert werden. Anstelle den Rechner neu zu starten gibt man folgendes ein

```
C:\>ipconfig /flushdns

Windows-IP-Konfiguration
Der DNS-Auflösungscache wurde geleert.

C:\>
```

7.3 tracert

Mit tracert kann man die Route vom eigenen PC zur Zieladresse verfolgen, und evtl. Rückschlüsse ziehen, ob und warum eine Verbindung hängt.

Beispiel

```
C:\>tracert de.wikibooks.org
```

Ausgabe

```
Routenverfolgung zu rr.knams.wikimedia.org [145.97.39.155] über maximal 30
Abschnitte:
 1  <1 ms    <1 ms    <1 ms    123.456.789.1
 2  24 ms    18 ms    19 ms    adslgw01-fra4.rm-com.net [217.173.128.45]
 3  139 ms   35 ms    36 ms    bbcr01-ams.titannetworks.nl [217.173.143.17]
 4  40 ms    *        43 ms    xsr03.asd002a.surf.net [195.69.144.34]
 5  41 ms    34 ms    35 ms    AZ-500.XSR01.Amsterdam1A.surf.net [145.145.80.21]
 6  39 ms    38 ms    37 ms    KNCSW001-router.Customer.surf.net
 [145.145.18.158]
 7  41 ms    43 ms    37 ms    gi0-24.csw2-knams.wikimedia.org [145.97.32.29]
 8  38 ms    38 ms    36 ms    rr.knams.wikimedia.org [145.97.39.155]
Ablaufverfolgung beendet.
C:\>
```


7.4 netstat

Dieser Befehl zeigt alle offenen TCP/IP/ICMP/UDP (usw...) Verbindungen an.

```
netstat
```

Mit der Erweiterung `-p` Protokoll lassen sich auch nur Verbindungen zum Entsprechenden Protokoll auslesen(meist TCP):

```
netstat -p TCP
```

7.5 NET

NET ist ein Tool zur Überwachung und Manipulation von Netzwerkaktivitäten sowie Windows-Diensten. Es hat einen recht großen Funktionsumfang, von dem ich hier jedoch nur einige kurz vorstelle.

7.5.1 NET USE - Netzlaufwerke verbinden

NET USE kann genutzt werden um eine Verbindung zu Netzwerk- oder Samba-Freigaben herzustellen. Üblicherweise wird hierbei ein Laufwerksbuchstabe verwendet, dies ist jedoch nicht zwingend.

Syntax:

```
NET USE x: \\name.oder.ip.des.rechners\freigabe
```

Wenn für die Verbindung die Eingabe von Zugangsdaten erforderlich ist können diese auch mit angegeben werden:

```
NET USE x: /USER:benutzer \\name.oder.ip.des.rechners\freigabe "passwort"
```

Wenn für einen Rechner Zugangsdaten benötigt werden und diese nicht oder falsch angegeben wurden, so wird der Benutzer nach diesen gefragt. Es kann auch nur ein Benutzer angegeben werden indem das Passwort durch einen Asterisk (*) ersetzt wird.

Die Eingabe von NET USE ohne Parameter zeigt bereits verbundene Laufwerke an. Bei Windows-Freigaben wird auch ein Status angezeigt, bei Samba-Freigaben jedoch nicht.

7.5.2 NET SEND - Nachrichten an andere Rechner senden

Mit NET SEND können kurze Nachrichten an andere Rechner geschickt werden. Der Empfänger sieht diese dann normalerweise in einer Message-Box. Die Benutzung empfiehlt sich nur bedingt, da moderne Firewalls solche Nachrichten normalerweise blocken (siehe Hinweis unten).

Syntax:

```
NET SEND Empfänger Nachricht
```

Empfänger kann dabei sein:

- IP-Adresse
- Domain
- Domain mit Benutzerangabe (/domain:name)
- Rechnernamen im lokalen Netzwerk
- Benutzername im lokalen Netzwerk
- Asterisk für alle erreichbaren (lokalen) Rechner

***Hinweis:** Da NET SEND leider sehr oft für Spam missbraucht wird/wurde empfiehlt es sich es abzuschalten. Dies geschieht über den Windows-Service "Nachrichtendienst". Dadurch können weder Nachrichten gesendet noch empfangen werden. (NET STOP Messenger)*

7.5.3 NET START

Startet Windows-Dienste

Syntax:

```
NET START Dienstname (Entweder "Messenger" oder "Nachrichtendienst", je nach  
Regions- / Spracheinstellungen)
```

Die Eingabe von NET START ohne Parameter zeigt die aktuell gestarteten Windows-Dienste an.

7.5.4 NET STOP

Stoppt Windows-Dienste

Gleiche Syntax wie NET START

7.5.5 NET VIEW

Zeigt erreichbare Computer des lokalen Netzwerks, bzw. deren Freigaben an.

Syntax:

```
NET VIEW [\\computername]
```

Einfache Eingabe von NET VIEW zeigt alle Computer an, wenn ein Computername mit angegeben wird werden die Freigaben angezeigt. Dies funktioniert nur bei Windows-Rechnern. UNIX-Rechner mit Samba-Freigaben werden zwar in der Übersicht angezeigt, die Freigaben können aber nicht abgefragt werden.

7.5.6 NET USER

Mit diesem Befehl können Benutzerkonten verwaltet werden. Dazu sind Administratorrechte erforderlich.

Syntax:

```
NET USER
```

Dieser Befehl allein zeigt nur alle Benutzer des Rechners an. (2000) (XP)

```
Net User <name>
```

Durch diesen Befehl wird die Einstellung des Benutzerkontos des Benutzers <name> angezeigt. (2000) (XP)

```
Net User <name> <password> /add
```

Dieser Befehl fügt einen Benutzer hinzu. (2000) (XP)

```
Net User <name> /delete
```

Und dieser Befehl wiederum löscht einen Benutzer. (2000) (XP)

```
Net User <name> <passwort>
```

Dieser Befehl verändert das Passwort des Benutzers. Wenn anstatt des passworts ein Stern * angegeben wird, dann wird das neue Passwort abgefragt. (2000) (XP)

```
Net User <name> /expires:never
```

Damit wird eingestellt, dass das Passwort des Benutzers <name> niemals abläuft. (2000) (XP)

```
Net User <name> /active:Yes  
Net User <name> /active:No
```

Damit wird das entsprechende Benutzerkonto aktiviert oder deaktiviert. (2000) (XP)

```
Net User <name> /passwordchg:Yes  
Net User <name> /passwordchg:No
```

Damit wird eingestellt, ob der Benutzer sein eigenes Passwort verändern darf. (2000) (XP)

```
Net User <name> /comment:"Herr Maier vom Verkauf"
```

Damit wird der Benutzerkommentar verändert. (2000) (XP)

```
Net User <name> /Fullname:"Karl-Otto Maier-Müller"
```

Damit wird der komplette Benutzername verändert. (2000) (XP)

```
Net User <name> /expires:2009-12-31
```

Damit wird eine Zeit vorgegeben, nach der der Benutzer sein Passwort ändern muss. (2000) (XP)

```
Net User <name> /profilepath="D:\Maier"  
Net User <name> /scriptpath="D:\Maier"  
Net User <name> /homedir="D:\Maier"  
Net User <name> /homedirreq=Yes  
Net User <name> /homedirreq=No
```

Damit wird eingestellt, wo sich die Benutzerdaten befinden. Der Ort der Benutzerdaten wird in der Umgebungsvariable *USERPROFILE* gespeichert. (2000) (XP)

```
Net User <name> /countrycode=049
```

Damit wird die Ländereinstellung verändert. (2000) (XP)

```
Net User <name> /workstations:*
```

Damit wird es dem Benutzer erlaubt, sich von anderen Rechnern aus an diesem Rechner anzumelden. Anstatt * kann hier auch eine Liste von Computernamen angegeben werden. Dann darf sich der Benutzer nur von diesen Computern anmelden. (2000) (XP)

Hinweis: Diese Arbeitsvorgänge gehen auch mit der Windows Benutzerkontensteuerung. Diese kann erreicht werden durch [Start] [Einstellungen] [Systemsteuerung] [Benutzer]

Hinweis: Der Name des aktuell angemeldeten Benutzers ist in der Umgebungsvariable *UserName*.

Hinweis: Hilfe zum Befehl *Net User* erhält man durch Eingabe von *Net User /help*

Beispiel: Um den Benutzer *Otto* mit dem Passwort *Frieda* einzurichten, kann folgende Batch-Datei ausgeführt werden:

```
net user Otto Frieda /add  
net user Otto /expires:never  
net user Otto /comment:"Default USER from This Computer"  
net user Otto /active:Yes  
net user Otto
```


8 Beispiele

8.0.7 Regedit.exe

Eine kleine Batchdatei, um mit regedit.exe direkt einen bestimmten REG-Zweig anzuspringen:

```
set LastKey="HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows
  NT\CurrentVersion\Hotfix"

set
  Key=HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Applets\Regedit
reg add %Key% /f /v "LastKey" /d %LastKey%
start regedit.exe
```

Erläuterung:

- **LastKey** ist der Reg-Schlüssel, der als Letztes angezeigt wurde und dieser wird beim Start von regedit direkt angesprungen, diesen Schlüssel kann man nach Belieben anpassen.
- **Key** ist der Reg-Schlüssel in dem sich der Wert "**LastKey**" befindet.

LastKey wird mittels **reg add** in die Registry eingespielt und danach wird regedit gestartet. Das Ganze funktioniert allerdings nur, wenn regedit.exe noch nicht läuft.

Und hier noch eine kleine Batchdatei, um mit regedit.exe Werte aus der Registry auszulesen:

```
set
Key="HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Applets\Regedit"
regedit.exe /e c:\temp\inhalt.reg %Key%
```

Achtung! Die Datei inhalt.reg ist im Unicode-Format. Ein Tipp dazu: Der type-Befehl kann Unicode lesen.

8.0.8 TaskList-Abfrage

Eine Routine um das Laufen eines Prozesses mittels **tasklist.exe** festzustellen. In diesem Beispiel wird geprüft, ob **Prozess.exe** läuft.

```
@echo off
set ProgEXE=Prozess.exe
set tempdatei=%temp%\TempDatei.txt

tasklist /FI "IMAGENAME eq %ProgEXE%" /FO CSV>%tempdatei%

for /F "skip=2 tokens=1 delims=" %%f in (%tempdatei%) do (
    echo gefiltert: [%%f]
    if "%%f"=="%ProgEXE%" echo Prozess %ProgEXE% läuft
    goto ende
)
echo Prozess %ProgEXE% läuft nicht!
goto ende

:ende
del %tempdatei%
```

Alternativ, ohne obige temporäre Datei 'tempdatei' zu benötigen, kann man beispielsweise unter Windows XP folgendes verwenden:

```
TaskList /FI "IMAGENAME eq %ProgEXE%" 2>NUL | Find "%ProgEXE%" >NUL
IF ERRORLEVEL == 1 (
    ECHO NOT Running.
    GOTO Ende
) ELSE (
    ECHO Running.
    GOTO Ende
)
```

8.0.9 Dienste einrichten

```
@echo off
prompt -$G
echo.
echo ### Manuelle Dienste ###
echo.
call:M helpsvc "Hilfe und Support"
call:M ImapiService "IMAPI-CD-Brenn-COM-Dienste"

echo.
echo ### nicht benötigte Dienste Deaktivieren ###
echo.
call:D cisvc "Indexdienst"
call:D ERSvc "Fehlerberichterstattungsdienst"

echo.
echo ### Automatische Dienste ###
echo.
call:A CryptSvc "Kryptografiedienste"
call:A MSIServer "Windows Installer"
```



```
echo.
echo.
echo Fertig!
echo.
pause
goto:eof

:A
echo %2 auf Automatisch
set Parameter=auto
net start %1
goto machen

:M
echo %2 auf Manuell
set Parameter=demand
net stop %1
goto machen

:D
echo %2 auf Deaktiviert
set Parameter=disabled
net stop %1
goto machen

rem Systemstart                Gerätetreiber, der vom Startladeprogramm geladen wird.
rem system                      Gerätetreiber start während der Kernelinitialisierung.
rem auto                        automatischer start, nach systemstart, vor Benutzeranmeldung.
rem demand                      Dienst, der manuell gestartet werden muss.
rem disabled                    Dienst deaktivieren

:machen
    echo on
    sc config %1 start= %Parameter%
    @echo off
    echo.
    echo.
goto:eof
```

8.0.10 Windows XP SP2: installierte Patches/Updates auflisten

Zeigt unter Windows XP mit Service Pack 2 alle zusätzlich installierten Patches/Updates an.

```
@echo off

set RegHotfixListe=HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows
NT\CurrentVersion\Hotfix

set RegHotfixInfo=HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Updates\Windows XP\SP3
```

```
for /F "delims=\ tokens=7" %%f in ('reg query "%RegHotfixListe%") do (
    set HotFixNr=%%f
    CALL :Anzeigen
)

echo.
echo.
pause
goto:eof

:Anzeigen
    reg query "%RegHotfixListe%\%HotFixNr%" /v Installed | FIND "0x1">NUL
    if "%errorlevel%"=="0" (
        set installiert=ja
        call:KB_Info
    ) ELSE (
        set installiert=nein
    )

    echo %HotFixNr% - Ist Installiert: %installiert%
    if not "%InstalledBy%"==" " echo %HotFixNr% installiert von %InstalledBy%
    if not "%InstalledDate%"==" " echo %HotFixNr% am %InstalledDate%
    echo.
goto:eof

:KB_Info
    set InstalledBy=
    for /F "tokens=3" %%c in ('reg query "%RegHotfixInfo%\%HotFixNr%" /v
    InstalledBy 2^>NUL ^| FIND "InstalledBy"') do (
        set InstalledBy=%%c
    )
    set InstalledDate=
    for /F "tokens=3" %%c in ('reg query "%RegHotfixInfo%\%HotFixNr%" /v
    InstalledDate 2^>NUL ^| FIND "InstalledDate"') do (
        set InstalledDate=%%c
    )
goto:eof
```

8.0.11 Printdateien direkt an den Drucker senden

Sendet einzelne Druckdateien an im Netzwerk freigegebene Drucker. Eignet sich besonders bei Standard Drucksprachen wie Postscript und HPGL/2.

```
@echo off
rem "pl.bat" sendet einzelne Druckdateien an bel. Drucker direkt.
rem eignet sich besonders bei Standard Drucksprachen wie Postscript und HPGL/2.
```

```
rem Laesst sich in Windows bequem einbinden, z.B rechte Maustaste - Senden an
Plotter

:voreingestellt
  set server=192.168.0.5
  rem server --- IP von Printserver/oder Name von Freigaberechner ---
  set druckerbeschreibung=192.168.0.5 Freigabename EPSON
  rem druckerbeschreibung --- damit der Drucker leichter identifiziert werden
kann ---
  set drucker=EPSONNT
  rem drucker --- das muss der Freigabename im Netzwerk/auf dem Freigaberechner
sein ---

  if '%1' == 'drucker01' goto drucker01
  if '%1' == 'drucker02' goto drucker02
  if '%1' == 'canon' goto canon
goto datei

:drucker01
  shift
goto datei

:drucker02
  set server=42.1.1.240
  set druckerbeschreibung=HP DJ 755CM im Sekretariat
  set drucker=hpdj755-tcp
  shift
goto datei

:canon
  set server=192.168.0.1
  set druckerbeschreibung=Canon s/w Laser im EG
  set drucker=CanonLBP
  shift
goto datei

:datei
  if '%~1' == '' goto aufruf
goto plot

:plot
  echo.
  echo Dateiname      : %1
  echo Druckserver   : %server%
  echo Freigabename: %drucker%
  echo Beschreibung: %druckerbeschreibung%
  echo.
  print /d:\\%server%\\%drucker% '%~1'
goto:eof

:aufruf
echo.
echo   aufruf      : pl [plotter] Datei
echo   plotter    : drucker01, drucker02, canon, (voreingestellt ist drucker01)
echo.
```

Anmerkung zur Nutzung dieses Scriptes:

- Der Printbefehl schickt die Datei OHNE Interpretation direkt an den Drucker. Eine Bilddatei im JPEG-Format würde statt der Grafik also lediglich einen haufen wirrer Zeichen ergeben.
- Der Drucker muss in der Lage sein, selbst ASCII-Zeichen zu verarbeiten und aus zu geben. Dies ist bei älteren Druckern (vor ca. 1996 sicher) immer der Fall. Heute beherrschen meist nur noch PostScript-fähige Drucker diese Fähigkeit. "Billigen" GDI-Druckern muss man etwas auf die Sprünge helfen mit dem verwendeten "Druckprozessor" im Windows-Druckertreiber.
- Zum Umschalten zunächst die Druckerübersicht öffnen - Zu finden unter Start|Drucker und Faxgeräte
- Dann die Eigenschaften des gewünschten Druckers aufrufen (Rechtsklick|Eigenschaften)
- Auf dem Reiter "Erweitert" den Button "Druckprozessor..." klicken
- In dem erscheinenden Dialogfeld gibt es 2 Felder mit Einträgen. Im rechten Feld (Standarddatentypen) findet sich neben verschiedenen anderen auch der Eintrag "Text". Diesen markieren und dann alles mit OK wieder schließen.

Der Druckprozessor von Windows ist nun in der Lage, Ausgaben entsprechend zu Interpretieren und dem GDI-Drucker als Grafik zu liefern. Es können aber nur stur 80 Zeichen (maximal) auf ca. 60 Zeilen verteilt werden. Manchmal sind es bis zu 66 Zeilen.

8.0.12 Copy

Kopieren von Dateien.

Der Befehl

```
copy test.txt c:\temp
```

kopiert die Datei *test.txt* aus dem aktuellen Ordner in den Ordner *c:\temp*. Der Befehl

```
copy c:\test.txt c:\temp
```

kopiert die Datei *test.txt* aus dem Ordner *c:* in den Ordner *c:\temp*

Datei- und Pfadnamen mit Sonderzeichen (hier das Leerzeichen sowie `()[]{}^=;!'+,`~`)` müssen beim Copy=Kommando und den meisten anderen

Kommandos in doppelten Anführungszeichen (") angegebenen werden, zum Beispiel:

```
copy f:\Beispiel.cmd "c:\Dokumente und Einstellungen\Siegfried\Eigene
Dateien\"
```

Wie man sieht, kann man beide Schreibweisen mischen. Dagegen können sich mehrfach vorhandene Anführungszeichen wieder aufheben. Deshalb müssen vom Benutzer angegebene Anführungszeichen erst entfernt werden, bevor man in der eigenen Batchdatei neue setzt:

```
@echo off
echo Kopiert eine Datei in das eigene Benutzerprofil.
rem Die Anführungszeichen in der folgenden Zeile werden nur benötigt,
rem damit zwischen dem Doppelpunkt und der Eingabe ein Abstand entsteht.
set /p name="Bitte geben Sie einen Dateinamen an: "
rem In der folgenden Zeile werden alle Anführungszeichen entfernt, da
rem nach dem Gleichheitszeichen kein Ersatztex folgt, siehe auch set/?.
rem Das Set-Kommando kommt sowohl beim Variablennamen als auch beim
rem Inhalt ohne Anführungszeichen aus.
set name=%name:="%
rem An dieser Stelle sollte erst geprüft werden, ob die Datei überhaupt
rem existiert, aber das würde den Rahmen hier sprengen.
copy "%name%" "%HOMEDRIVE%%HOMEPATH%\Eigene Dateien"
```

Theoretisch könnte man auch mit den kurzen Dateinamen (8+3) arbeiten, aber diese sind nicht eindeutig und können von Laufwerk zu Laufwerk unterschiedlich sein. Daher besser nicht verwenden!

Kombiniert in der gegebenen Reihenfolge zwei oder mehrere durch "+" verbundene Textdateien in einer neuen Gesamtdatei.

In diesem Zusammenhang muss auf die Bedeutung der Parameter "/A" (für ASCII-Format, der Standardwert) und "/B" (für Binär-Format) eingegangen werden, die jedem Dateinamen in obigem Beispiel mit einem Leerzeichen Abstand vorangehen können. DOS-Textdateien enthalten ein Dateiendezeichen (Dezimal 26, Hexadezimal 1A). Beim Kopiervorgang darf in der Gesamtdatei nur als letztes Zeichen ein Dateiendezeichen erscheinen, sonst würde der Text nach der ersten Endmarkierung nicht mehr angezeigt. "copy" kopiert ohne das abschließende Zeichen und fügt als letztes wieder eines zur Gesamtdatei hinzu.

Möchte man Dateien mit binären Inhalten zusammenfügen (z.B. PostScript-Dateien, die auch binäre Abschnitte enthalten können), so würde ohne Parameter "/B" ebenfalls nur bis zu einem zufällig vorkommenden Dateiendezeichen kopiert und

damit das Ergebnis unbrauchbar.

```
copy /b "seite 1.ps" + /b "seite 2.ps" "Neue Datei.ps"
```

Als Zieldatei sollte man keine der Quelldateien verwenden, da möglicherweise sonst der Inhalt überschrieben wird.

8.0.13 Message-Ping

Mit Hilfe des folgenden Scripts kann man Windows um eine nützliche Funktionalität erweitern. Mit Hilfe von **mping** erhält man eine Nachricht, sobald der gepingte Computer wieder erreichbar ist.

Um das Script global zu verwenden, muss es z.B. als "mping.cmd" in "**Windows-Installationsordner\System32**" gespeichert werden und kann von nun an z.B. mit "mping google.com" aufgerufen werden.

```
@echo off

:: wenn kein remote Host angegeben ist, dann gehe zu :Syntax
IF [%1]==[] GOTO Syntax
IF [%1]==[/?] GOTO Syntax
IF [%1]==[/h] GOTO Syntax
IF [%1]==[/help] GOTO Syntax

:mainloop
:: pinge den Host 1 mal und leite die Ausgabe nach 'nul' um (keine sichtbare
  Ausgabe)
ping %1 -n 1 >nul

:: speichere den Errorlevel in die Variable 'status'
set status=%errorlevel%

:: sende eine Nachricht sobald der remote Host online ist
if %status%==0 (msg %username% %1 online && exit 0)

:: minimalistische Fortschrittsanzeige
(Set /P i=.) < NUL

:: sleep workaround (wer eine bessere, einfachere, genauere & universellere
  Möglichkeit kennt bitte korrigieren)
:: pinge den lokalen Computer 10 mal
ping localhost -n 10 >nul

:: gehe zurück zum anfang
goto mainloop

:Syntax
echo.
```

```

echo Syntax: %~n0 [Hostname oder IP]
echo.
echo mping
echo.
echo Pingt einen offline Host und gibt eine Meldung
echo aus, sobald er wieder online ist
echo.

```

8.0.14 Defrag mit Endlos-Schleife

Das eingebaute Defrag-Tool erlaubt ja immer nur die gleichzeitige Defragmentierung eines Laufwerks. Außerdem sind oftmals mehrere Durchgänge nötig, bis das Laufwerk wirklich vollständig defragmentiert ist. Mit dieser kleinen Batch Datei werden alle Partitionen immer wieder hintereinander defragmentiert. Wenn man wieder arbeiten will, kann man einfach das Eingabefenster schließen. Auch bei laufendem defrag passiert dabei nichts, weil defrag im Hintergrund die aktuelle Arbeit sauber beendet.

Damit aber nicht defekte Laufwerke einfach blind defragmentiert werden, habe ich eine etwas aufwendige Prozedur eingebaut, die die Laufwerke mit chkdsk erst überprüft. Wenn dabei ein Fehler auftritt, wird mit fsutil das dirty Bit gesetzt. Das bewirkt, dass beim nächsten Systemstart die Fehler behoben werden. Außerdem werden die Partitionen in dem loop ausgelassen, welche schon als dirty markiert sind.

Eine kleine Routine war auch nötig, damit Netzwerklaufrwerke ausgelassen werden.

```

@echo off
cd /d C:\

:loop
  for %i in (c,d,e,f,g,h,i,j,k,l,m,n,o,p,q,r,s,t,u,v,w,x,y,z) do (
    if exist %i:\NUL call:defrag %i
  )
  echo.
  echo pause...
  ping localhost -n 10 >NUL
goto loop

:defrag
  call:check %1
  if "%ok%"=="nein" (
    echo Laufwerk %1: wird ausgelassen
    echo.
    goto:eof
  )
  echo.
  echo.
  echo chkdsk %1: /v

```

```
chkdsk %1:
if %errorlevel% NEQ 0 (
    if %errorlevel% NEQ 2 (
        echo %1: chkdsk Fehler!!!
        echo Errorlevel: %errorlevel%
        echo.
        echo setzte dirty-Bit:
        fsutil dirty set %1:
        echo.
        echo Versuche Reparatur
        chkdsk %1: /v /x
        if %errorlevel% EQU 2 (
            echo Fehler bereinigt!
        ) else (
            echo Errorlevel: %errorlevel%
            echo.
            goto:eof
        )
    )
)
echo defrag %1: -f
defrag %1: -f
if %errorlevel% NEQ 0 (
    pause
    exit
)
rem pause...
ping localhost -n 1 >NUL
goto:eof

:check
set ok=nein
net use | find /i "%1:" >NUL
if not errorlevel 1 (
    echo %1: ist ein Netzlaufwerk
    goto:eof
)
fsutil dirty query %1:
for /F "tokens=5" %%f in ('fsutil dirty query %1:') do (
    if not "%%f"=="NICHT" (
        rem Laufwerk ist als fehlerhaft markiert
        goto:eof
    )
)
set ok=ja
goto:eof
```

Als Alternative gibt es das Open Source Program **JkDefrag**¹, welches auch alle Laufwerke nacheinander defragmentiert.

¹ <http://www.kessels.com/JkDefrag/>

8.0.15 Dateiinformationen anzeigen

Dieses Batch-Skript zeigt Informationen über die Datei, die als Parameter übergeben wird, an. (beispielsweise per Drag&Drop auf die Batch-Datei)

```
@echo off
echo Attribute.....: %~a1
echo Laufwerk.....: %~d1
echo kompletter Pfad.....: %~f1
echo Dateiname.....: %~n1
echo Ordnerpfad.....: %~p1
echo kompletter Pfad (kurze Form): %~s1
echo Aenderungsdatum.....: %~t1
echo Dateierweiterung.....: %~x1
echo Dateigroesse.....: %~z1
pause
```

Hinweis: Falls die Attribute des eigenen Batch-Skripts ausgelesen werden sollen, so muss der Parameter Null verwendet werden (Beispielsweise %~p0). Eine Kombination ist auch möglich, z.B: %~dp0 (Laufwerk + Pfad)

8.0.16 Alle verfügbaren PCs im aktuellen Netzwerk suchen

Dieses Batch-Skript pingt alle IP-Adressen eines IP-Bereichs an und zeigt dann die PCs an, die verfügbar sind.

```
@echo off

if exist ips.txt del ips.txt > nul

echo Bitte geben Sie den ersten Teil der IP ein:
set /p ip=

for /L %%N IN (1, 1, 255) DO (
    echo Die IP: %ip%%N wird nun angepingt
    ping %ip%%N -n 1 -w 1 | find "TTL" && echo %ip%%N >> ips.txt
)

cls
type ips.txt
pause >nul
```

Hinweis: Die Datei ips.txt wird nicht gelöscht, damit man sie für weitere Zwecke benutzen kann.

8.0.17 Hosts Datei mit einem Aliasnamen und der aktuellen IP-Adresse aktualisieren

Häufig werden Webserver oder ähnliche Programme unter Angabe eines Aliasnamens installiert, damit sie leicht auf einen anderen Rechner verschoben werden können, falls der ursprüngliche Rechner defekt ist. Oder die Applikationläuft unter VM-Ware auf einen Laptop bei dem sich die IP-Adressen häufig ändern. Dann kann mit dem Skript der Hostnamenalias mit der aktuellen IP-Adresse aktualisiert werden. Idealerweise wird das Script automatisch beim Start von Windows oder beim Anmelden des Anwenders ausgeführt.

```
@echo off
set HOST_ALIAS=myserver myserver.mydomain.org
set HOST_FILE=%SystemRoot%\System32\drivers\etc\hosts
set TEMP_FILE=%TEMP%\hosts.tmp

REM vorhandenen Eintrag entfernen und den Rest in TEMP-Datei schreiben
type %HOST_FILE% | find /v "myserver" > %TEMP_FILE%

REM Neuen Eintrag wieder hinzufügen
for /F "tokens=15" %i in ('ipconfig ^| find "IP Ad"') do set IP=%i
echo %IP% %HOST_ALIAS%>>%TEMP_FILE%
echo ## Added %IP% %HOST_ALIAS% to the hostsfile

REM Temp-File nach %SystemRoot%\System32\drivers\etc\hosts kopieren
REM Die Datei Hosts darf nicht mit "move" verschoben werde, sondern muss kopiert
  werden, da sonst der Hostname nicht gefunden werden kann.
copy /y %TEMP_FILE% %HOST_FILE%
del /F /Q %TEMP_FILE%
```

8.0.18 Windowsversion herausfinden

Mit dieser Batchdatei findet man heraus, welche Windowsversion verwendet wird. Das ist nützlich, wenn man mit bestimmten Pfaden arbeitet, die je nach Betriebssystem verschieden sind. Hier wird das Ergebnis mithilfe einer VBScript MsgBox ausgegeben.

```
@echo off

VER |find /i "Windows 95" >NUL
IF NOT ERRORLEVEL 1 GOTO 9598ME
VER |find /i "Windows 98" >NUL
IF NOT ERRORLEVEL 1 GOTO 9598ME
VER |find /i "Windows Millennium" >NUL
IF NOT ERRORLEVEL 1 GOTO 9598ME
VER | find "XP" > nul
IF %errorlevel% EQU 0 GOTO XP
VER | find "2000" > nul
```

```
IF %errorlevel% EQU 0 GOTO 2000
VER | find "NT" > nul
IF %errorlevel% EQU 0 GOTO NT
VER | find "Microsoft Windows [Version 6.1" > nul
IF %errorlevel% EQU 0 GOTO Sieben
VER | find "Microsoft Windows [Version 6" > nul
IF %errorlevel% EQU 0 GOTO Vista
VER | find "Microsoft Windows [Version 5" > nul
IF %errorlevel% EQU 0 GOTO 2003
goto unknown
goto end

:unknown
echo MsgBox "Betriebssystem unbekannt", VbInformation + VbOKOnly, "Ihr
  Betriebssystem" >%TEMP%\os.vbs
goto end

:9598ME
echo MsgBox "Sie haben 95, 98 oder ME", VbInformation + VbOKOnly, "Ihr
  Betriebssystem" >%TEMP%\os.vbs
goto end

:NT
echo MsgBox "Sie haben Windows NT", VbInformation + VbOKOnly, "Ihr
  Betriebssystem" >%TEMP%\os.vbs
goto end

:2003
echo MsgBox "Sie haben Windows 2003", VbInformation + VbOKOnly, "Ihr
  Betriebssystem" >%TEMP%\os.vbs
goto end

:2000
echo MsgBox "Sie haben Windows 2000", VbInformation + VbOKOnly, "Ihr
  Betriebssystem" >%TEMP%\os.vbs
goto end

:XP
echo MsgBox "Sie haben Windows XP", VbInformation + VbOKOnly, "Ihr
  Betriebssystem" >%TEMP%\os.vbs
goto end

:Vista
echo MsgBox "Sie haben Windows Vista oder Windows 7 RC", VbInformation +
  VbOKOnly, "Ihr Betriebssystem" >%TEMP%\os.vbs
goto end
:Sieben
echo MsgBox "Sie haben Windows 7 ", VbInformation + VbOKOnly, "Ihr
  Betriebssystem" >%TEMP%\os.vbs
goto end

:end
START %TEMP%\os.vbs
```

8.0.19 TEMP-Verzeichnis löschen

Im %TEMP%-Verzeichnis sammeln sich mit der Zeit viele Dateien und Unterverzeichnisse an. Mit diesem Skript werden alle auf einmal gelöscht:

```
@echo off

call:clean %TEMP%
IF NOT "%TEMP%" == "%TMP%" (
    call:clean %TMP%
)

pause
goto:eof

:clean
del /q "%~1\*.*"
FOR /D %%D IN ("%~1\*") DO (
    rmdir /s /q "%%D"
)
goto:eof
```

(Wahrscheinlich erhält man ein paar Fehler, wenn z.B. Dateien geöffnet sind)

Neue Version

Eine Version für Win XP/2000/WinVista/Win7

```
Echo off
del /f /s /q "%HOMEPATH%\Lokale Einstellungen\Temp\*.*"
del /f /s /q "%HOMEPATH%\Lokale Einstellungen\Temporary Internet Files\*.*"
del /f /s /q "%HOMEPATH%\Lokale Einstellungen\Verlauf\*.*"
del /f /s /q "%windir%\Temp\*.*"
del /f /s /q "%windir%\Prefetch\*.*"
del /f /s /q "%windir%\Temp\*.*"
rmdir /s /q "%HOMEPATH%\Lokale Einstellungen\Temp\"
rmdir /s /q "%windir%\Prefetch\"

diskperf -n
ipconfig /flushdns
```

Einfach den Text kopieren in eine Textdatei und zur .bat machen.

9 Zusatz-Tools

Viele Aufgaben sind mit der Batchprogrammierung bzw. mit Windows-Scripting alleine kaum oder nur sehr umständlich zu lösen. So kann das Umwandeln von Dateien (Textersetzungen) aufgrund der nicht ganz sauberen Behandlung von Sonder- bzw. Operationszeichen zur Qual werden. Erstaunlich für ein Betriebssystem, welches seit nun mehr über *10 Jahren* - mit Windows 95 - diese Restriktionen offiziell beseitigt hat.

Oft können dafür OpenSource-Tools wie SFK ("Swiss File Knife")¹ und Sed² sehr erfolgreich in die Batch-Programmierung eingebunden werden. Für das Verpacken und Komprimieren von Dateien lässt sich das freie 7-Zip³ gut über die Kommandozeile steuern. Das GPL-Programm Gnuplot eignet sich hervorragend für die Visualisierung von Daten im CSV-Format und lässt sich einfacher und schneller scripten als MS Excel oder OO.org Calc. Zum Herunterladen von Dateien und Webseiten kann Wget verwendet werden. Auch für Batch-Dateien gibt es diverse kostenlose Batch-Compiler, die die Datei in eine ausführbare .exe-Datei umwandeln und den Code vor Manipulation und/oder Copyright-Verletzungen schützen. Für gehobeneren Scripting-Ansprüche und komplexere Vorhaben mit einer längeren Lebensdauer sollten dezidierte Scripting-Sprachen z.B. Awk, Perl⁴, Python, Ruby oder Tcl in Erwägung gezogen werden, für die glücklicherweise das GNU-Projekt *freie* und quelloffene Alternativen bietet.

Eine kostenlose Alternative zum Batching bietet Microsoft selbst auf seiner Homepage mit dem Tool *PowerShell*.

Weitere Tools sind im Abschnitt "WebLinks" aufgelistet!

1 <http://swissfileknife.sourceforge.net/>

2 <http://de.wikipedia.org/wiki/Sed%20%28Unix%29>

3 <http://www.7-zip.org/>

4 <http://de.wikibooks.org/wiki/Perl>

10 Referenz

10.1 Liste aller internen und externen Befehle (entspricht der Ausgabe des Befehls `help` unter Windows XP)

ASSOC	Zeigt Dateierweiterungszuordnungen an bzw. ändert sie.
AT	Legt eine Zeit fest, zu der Befehle und Programme auf diesem Computer ausgeführt werden.
ATTRIB	Zeigt Dateiattribute an bzw. ändert sie.
BREAK	Schaltet die erweiterte Überprüfung für STRG+C ein bzw. aus.
CACLS	Zeigt Datei-ACLs (Access Control List) an bzw. ändert sie.
CALL	Ruft eine Batchdatei aus einer anderen Batchdatei heraus auf.
CD	Zeigt den Namen des aktuellen Verzeichnisses an bzw. ändert diesen.
CHCP	Zeigt die aktive Codepagenummer an bzw. legt diese fest.
CHDIR	Zeigt den Namen des aktuellen Verzeichnisses an bzw. ändert diesen.
CHKDSK	Überprüft einen Datenträger und zeigt einen Statusbericht an.
CHKNTFS	Zeigt die Überprüfung des Datenträgers beim Start an bzw. verändert sie.
CLS	Löscht den Bildschirminhalt.
CMD	Startet eine neue Instanz des Windows-Befehlsinterpreters.
COLOR	Legt die Hintergrund- und Vordergrundfarben für die Konsole fest.
COMP	Vergleicht den Inhalt zweier Dateien oder Sätze von Dateien.
COMPACT	Zeigt die Komprimierung von Dateien auf NTFS-Partitionen an bzw. ändert diese.
CONVERT	Konvertiert FAT-Volumes in NTFS. Das aktuelle Laufwerk kann nicht konvertiert werden.
COPY	Kopiert eine oder mehrere Dateien an eine andere Stelle.
DATE	Zeigt das Datum an bzw. legt dieses fest.
DEL	Löscht eine oder mehrere Dateien.

DIR	Listet die Dateien und Unterverzeichnisse eines Verzeichnisses auf.
DISKCOMP	Vergleicht den Inhalt von zwei Disketten.
DISKCOPY	Kopiert den Inhalt von einer Diskette auf eine andere Diskette.
DOSKEY	Bearbeitet Befehlseingaben, ruft Windows-Befehle zurück und erstellt Macros.
ECHO	Zeigt Meldungen an bzw. schaltet die Befehlsanzeige ein oder aus.
ENDLOCAL	Beendet den lokalen Gültigkeitsbereich von Umgebungsänderungen in einer Batchdatei.
ERASE	Löscht eine oder mehrere Dateien.
EXIT	Beendet das Programm CMD.EXE (Befehlsinterpreter).
FC	Vergleicht zwei oder mehr Sätze von Dateien und zeigt die Unterschiede an.
FIND	Sucht eine Zeichenkette in einer oder mehreren Datei(en).
FINDSTR	Sucht Zeichenketten in Dateien.
FOR	Führt einen angegebenen Befehl für jede Datei in einem Dateiensatz aus.
FORMAT	Formatiert einen Datenträger für die Verwendung mit Windows.
FTYPE	Zeigt die Dateitypen an, die bei den Zuordnungen für die entsprechenden Dateierweiterungen verwendet werden bzw. ändert sie.
GETMAC	Zeigt unter Windows XP SP2 die Mac-Adresse an.
GOTO	Setzt den Windows-Befehlsinterpreter auf eine markierte Zeile in einem Batchprogramm.
GRAFTABL	Ermöglicht Windows, Sonderzeichen im Grafikmodus anzuzeigen.
HELP	Zeigt Hilfeinformationen zu Windows-Befehlen an.
IF	Verarbeitet Ausdrücke in einer Batchdatei abhängig von Bedingungen.
LABEL	Erstellt, ändert oder löscht die Bezeichnung eines Volumens.
MD	Erstellt ein Verzeichnis
MKDIR	Erstellt ein Verzeichnis.
MODE	Konfiguriert ein Systemgerät.
MORE	Zeigt Ausgabe auf dem Bildschirm seitenweise an.
MOVE	Verschiebt ein oder mehrere Dateien von einem Verzeichnis in ein anderes.

NET USE	Verbindet einen Computer mit einer freigegebenen Ressource oder trennt die Verbindung und zeigt Informationen über die Verbindungen eines Computers an. Der Befehl steuert außerdem ständige Netzwerkverbindungen.
PATH	Legt den Suchpfad für ausführbare Dateien fest oder zeigt diesen an.
PAUSE	Hält die Ausführung einer Batchdatei an und zeigt eine Meldung an.
POPD	Wechselt zu dem Verzeichnis, das durch PUSHHD gespeichert wurde.
PRINT	Druckt eine Textdatei.
PROMPT	Ändert die Eingabeaufforderung.
PUSHD	Speichert das aktuelle Verzeichnis, und wechselt dann zu einem anderen Verzeichnis.
RD	Entfernt ein Verzeichnis.
RECOVER	Stellt lesbare Daten von einem beschädigten Datenträger wieder her.
REM	Leitet Kommentare in einer Batchdatei bzw. CONFIG.SYS ein.
REN	Benennt eine Datei bzw. Dateien um.
RENAME	Benennt eine Datei bzw. Dateien um.
REPLACE	Ersetzt Dateien.
RMDIR	Löscht ein Verzeichnis.
SET	Setzt oder löscht die Umgebungsvariablen bzw. zeigt sie an.
SETLOCAL	Beginnt den lokalen Gültigkeitsbereich von Umgebungsänderungen in einer Batchdatei.
SHIFT	Verändert die Position ersetzbarer Parameter in Batchdateien.
SORT	Sortiert die Eingabe.
START	Startet ein eigenes Fenster, um ein bestimmtes Programm oder einen Befehl auszuführen.
SUBST	Weist einem Pfad einen Laufwerksbuchstaben zu.
TASKLIST	Zeigt alle zurzeit laufenden Aufgaben inklusive der Dienste an.
TASKKILL	Bricht einen laufenden Prozess oder eine Anwendung ab oder beendet ihn bzw. sie.
TIME	Zeigt die Systemzeit an bzw. legt sie fest.
TITLE	Legt den Fenstertitel für das Eingabeaufforderungsfenster fest.
TREE	Zeigt die Ordnerstruktur eines Laufwerks oder Pfads grafisch an.
TYPE	Zeigt den Inhalt einer Textdatei an.

VER	Zeigt die Windows-Version an.
VERIFY	Legt fest, ob überwacht werden soll, ob Dateien korrekt auf den Datenträger geschrieben werden.
VOL	Zeigt die Datenträgervolumenbezeichnung und die Seriennummer an.
XCOPY	Kopiert Dateien und Verzeichnisbäume.

11 WebLinks

11.1 Weblinks

- <http://www.heise.de/ct/03/16/136/default.shtml>¹ c't 16/2003, S. 136: Windows-Befehlszeile
- <http://www.ss64.com/nt/index.html>² NT/XP Batch Befehle (en)
- <http://www.admini.de>³ Umfangreiche Informationen über Batchprogrammierung unter NT/2000/XP
- <http://www.fpschultze.de>⁴ Batch Scripting Site mit vielen Beispielen
- <http://sven-of-nine.de/site/doku.php/faq:batch>⁵ Umfangreiche Funktionssammlung für Batchprogrammierer
- <http://www.knowware.de/?cat=6.1&book=batch>
 - <http://download.knowware.de/batch.pdf>⁶ Leicht verständliche Einführung
- http://publibz.boulder.ibm.com/cgi-bin/bookmgr_-OS390/BOOKS/IEA2B510/CCONTENTS?SHELF=IEA2BK32&DN=SA22-7598-01&DT=20010626133938⁷ englischsprachiges JCL Benutzerhandbuch
 - <http://www.plogmann.net/w/2/54/index.htm>⁸ - Optionen des Befehls XCOPY
- <http://www.antonis.de/dos/#download/>⁹ Diverse Beispielskripte
- <http://www.easy-coding.de/shellscripting.html?s=5fb492fd4b515b0c880795a6a7d92ef5c4aed29c>¹⁰ Forum über Shellscripting

1 <http://www.heise.de/ct/03/16/136/default.shtml>

2 <http://www.ss64.com/nt/index.html>

3 <http://www.admini.de>

4 <http://www.fpschultze.de/>

5 <http://sven-of-nine.de/site/doku.php/faq:batch>

6 <http://download.knowware.de/batch.pdf>

7 http://publibz.boulder.ibm.com/cgi-bin/bookmgr_-OS390/BOOKS/IEA2B510/CCONTENTS?SHELF=IEA2BK32&DN=SA22-7598-01&DT=20010626133938

8 <http://www.plogmann.net/w/2/54/index.htm>

9 <http://www.antonis.de/dos/#download/>

10 <http://www.easy-coding.de/shellscripting.html?s=5fb492fd4b515b0c880795a6a7d92ef5c4aed29c>

- <http://home.mnet-online.de/horst.muc/maind.htm>¹¹ Eine Sammlung von Zusatztools für Batch-Skripter
- http://www.axel-hahn.de/axel/page_compi/bat_index.htm¹² Viele weitere Informationen und Tools zum Thema
- http://www.script-example.com/themen/Windows_Batch.php¹³ Batch Dateien erstellen für absolute Beginner

11.1.1 Tutorials

- Batch for Runaways Part I¹⁴ - Tutorial Batch I
- Batch for Runaways Part II¹⁵ - Tutorial Batch II
- Batch for Runaways Part III¹⁶ - Tutorial Batch III
- Die For-Schleife¹⁷ - Tutorial "FOR"
- Variablen¹⁸ - Tutorial Umgebungsvariablen

11.1.2 Referenzen

- Microsoft Technet¹⁹ - Windows Server 2008 - Command Reference (en)
- Microsoft Technet²⁰ - Windows Vista - Command-Line Tools Technical Reference (en)
- Microsoft Technet²¹ - Windows Server 2003 - Befehlszeilenreferenz (de)
- Microsoft Technet²² - Windows XP - Command-line reference A-Z (en)

11 <http://home.mnet-online.de/horst.muc/maind.htm>

12 http://www.axel-hahn.de/axel/page_compi/bat_index.htm

13 http://www.script-example.com/themen/Windows_Batch.php

14 http://www.administrator.de/Workshop_Batch_for_Runaways_-_Part_I_-_Beispiel_FindLongPath.Bat_Bedenklich_lange_Pfade_finden.html

15 http://www.administrator.de/Workshop_Batch_for_Runaways_-_Part_II_-_Ein_bisschen_Handwerkszeug.html

16 http://www.administrator.de/Workshop_Batch_for_Runaways_-_Part_III_-_Datums_und_Zeitvariablen_im_Batch.html

17 http://www.administrator.de/Tutorial_zur_FOR-Schleife.html

18 http://www.administrator.de/Wie_Arbeite_ich_mit_Batch_UmgebungsVariablen%3F_Erstellung_Umgang_Erweiterungen_Ver%C3%A4nderungen.html

19 <http://technet.microsoft.com/en-us/library/cc754340.aspx>

20 <http://technet.microsoft.com/en-us/library/cc722159.aspx>

21 <http://technet.microsoft.com/de-de/library/cc785423.aspx>

22 <http://technet.microsoft.com/en-us/library/bb490890.aspx>

12 Autoren

Edits	User
2	Alizdair ¹
1	AquariaNR ²
2	Autor ³
2	Azaël ⁴
7	Bastie ⁵
13	Batchscripter93 ⁶
2	Blingnue ⁷
1	Boern99 ⁸
10	Codejunkie ⁹
1	CorneliusWasmund ¹⁰
1	Creibens ¹¹
7	Daniel B ¹²
1	Der Leo ¹³
20	Dirk Huenniger ¹⁴
1	DonQuichot ¹⁵

-
- 1 <http://de.wikibooks.org/w/index.php?title=Benutzer:Alizdair>
 - 2 <http://de.wikibooks.org/w/index.php?title=Benutzer:AquariaNR>
 - 3 <http://de.wikibooks.org/w/index.php?title=Benutzer:Autor>
 - 4 <http://de.wikibooks.org/w/index.php?title=Benutzer:Aza%C3%ABl>
 - 5 <http://de.wikibooks.org/w/index.php?title=Benutzer:Bastie>
 - 6 <http://de.wikibooks.org/w/index.php?title=Benutzer:Batchscripter93>
 - 7 <http://de.wikibooks.org/w/index.php?title=Benutzer:Blingnue>
 - 8 <http://de.wikibooks.org/w/index.php?title=Benutzer:Boern99>
 - 9 <http://de.wikibooks.org/w/index.php?title=Benutzer:Codejunkie>
 - 10 <http://de.wikibooks.org/w/index.php?title=Benutzer:CorneliusWasmund>
 - 11 <http://de.wikibooks.org/w/index.php?title=Benutzer:Creibens>
 - 12 http://de.wikibooks.org/w/index.php?title=Benutzer:Daniel_B
 - 13 http://de.wikibooks.org/w/index.php?title=Benutzer:Der_Leo
 - 14 http://de.wikibooks.org/w/index.php?title=Benutzer:Dirk_Huenniger
 - 15 <http://de.wikibooks.org/w/index.php?title=Benutzer:DonQuichot>

- 1 E^(nix)¹⁶
- 2 Emes2k¹⁷
- 1 Enomil¹⁸
- 1 F. F. Fjodor¹⁹
- 2 Fehlerkorrektur²⁰
- 2 Flominator²¹
- 21 Genrich²²
- 4 Global667²³
- 1 Haddock²⁴
- 1 Hardy42²⁵
- 5 Heuler06²⁶
- 4 Hjherbert²⁷
- 2 Jaho4896²⁸
- 2 Jan²⁹
- 1 Joddel³⁰
- 7 Johanna31³¹
- 17 Juetho³²
- 1 Klartext³³
- 16 Klaus Eifert³⁴
- 1 Kleidersack³⁵

-
- 16 <http://de.wikibooks.org/w/index.php?title=Benutzer:E%5E%28nix%29>
 - 17 <http://de.wikibooks.org/w/index.php?title=Benutzer:Emes2k>
 - 18 <http://de.wikibooks.org/w/index.php?title=Benutzer:Enomil>
 - 19 http://de.wikibooks.org/w/index.php?title=Benutzer:F._F._Fjodor
 - 20 <http://de.wikibooks.org/w/index.php?title=Benutzer:Fehlerkorrektur>
 - 21 <http://de.wikibooks.org/w/index.php?title=Benutzer:Flominator>
 - 22 <http://de.wikibooks.org/w/index.php?title=Benutzer:Genrich>
 - 23 <http://de.wikibooks.org/w/index.php?title=Benutzer:Global667>
 - 24 <http://de.wikibooks.org/w/index.php?title=Benutzer:Haddock>
 - 25 <http://de.wikibooks.org/w/index.php?title=Benutzer:Hardy42>
 - 26 <http://de.wikibooks.org/w/index.php?title=Benutzer:Heuler06>
 - 27 <http://de.wikibooks.org/w/index.php?title=Benutzer:Hjherbert>
 - 28 <http://de.wikibooks.org/w/index.php?title=Benutzer:Jaho4896>
 - 29 <http://de.wikibooks.org/w/index.php?title=Benutzer:Jan>
 - 30 <http://de.wikibooks.org/w/index.php?title=Benutzer:Joddel>
 - 31 <http://de.wikibooks.org/w/index.php?title=Benutzer:Johanna31>
 - 32 <http://de.wikibooks.org/w/index.php?title=Benutzer:Juetho>
 - 33 <http://de.wikibooks.org/w/index.php?title=Benutzer:Klartext>
 - 34 http://de.wikibooks.org/w/index.php?title=Benutzer:Klaus_Eifert
 - 35 <http://de.wikibooks.org/w/index.php?title=Benutzer:Kleidersack>

- 1 Komikaa³⁶
- 6 Kronos86m³⁷
- 7 MF-Warburg³⁸
- 5 MichaelFrey³⁹
- 2 MichaelFreyTool⁴⁰
- 7 Michfrm⁴¹
- 11 Mjchael⁴²
- 1 Nomike⁴³
- 3 Ocrho⁴⁴
- 2 Pc-world⁴⁵
- 1 Peperkorn⁴⁶
- 1 Prince Kassad⁴⁷
- 1 Produnis⁴⁸
- 6 Prog⁴⁹
- 6 Reseka⁵⁰
- 3 Robert⁵¹
- 10 Robidog⁵²
- 25 Schneijo⁵³
- 1 Se4598⁵⁴
- 1 Stefan Majewsky⁵⁵

-
- 36 <http://de.wikibooks.org/w/index.php?title=Benutzer:Komikaa>
 - 37 <http://de.wikibooks.org/w/index.php?title=Benutzer:Kronos86m>
 - 38 <http://de.wikibooks.org/w/index.php?title=Benutzer:MF-Warburg>
 - 39 <http://de.wikibooks.org/w/index.php?title=Benutzer:MichaelFrey>
 - 40 <http://de.wikibooks.org/w/index.php?title=Benutzer:MichaelFreyTool>
 - 41 <http://de.wikibooks.org/w/index.php?title=Benutzer:Michfrm>
 - 42 <http://de.wikibooks.org/w/index.php?title=Benutzer:Mjchael>
 - 43 <http://de.wikibooks.org/w/index.php?title=Benutzer:Nomike>
 - 44 <http://de.wikibooks.org/w/index.php?title=Benutzer:Ocrho>
 - 45 <http://de.wikibooks.org/w/index.php?title=Benutzer:Pc-world>
 - 46 <http://de.wikibooks.org/w/index.php?title=Benutzer:Peperkorn>
 - 47 http://de.wikibooks.org/w/index.php?title=Benutzer:Prince_Kassad
 - 48 <http://de.wikibooks.org/w/index.php?title=Benutzer:Produnis>
 - 49 <http://de.wikibooks.org/w/index.php?title=Benutzer:Prog>
 - 50 <http://de.wikibooks.org/w/index.php?title=Benutzer:Reseka>
 - 51 <http://de.wikibooks.org/w/index.php?title=Benutzer:Robert>
 - 52 <http://de.wikibooks.org/w/index.php?title=Benutzer:Robidog>
 - 53 <http://de.wikibooks.org/w/index.php?title=Benutzer:Schneijo>
 - 54 <http://de.wikibooks.org/w/index.php?title=Benutzer:Se4598>
 - 55 http://de.wikibooks.org/w/index.php?title=Benutzer:Stefan_Majewsky

- 1 Stefan wichmann⁵⁶
- 1 StefanTiran⁵⁷
- 2 Sys.Bak⁵⁸
- 6 ThePacker⁵⁹
- 1 Theres no global warming⁶⁰
- 10 Timon.Freitag⁶¹
- 1 Tschäfer⁶²
- 2 Uwe.kausch⁶³
- 2 WeißNix⁶⁴
- 1 Wolfgangbeyer⁶⁵
- 3 Worker⁶⁶

-
- 56 http://de.wikibooks.org/w/index.php?title=Benutzer:Stefan_wichmann
 - 57 <http://de.wikibooks.org/w/index.php?title=Benutzer:StefanTiran>
 - 58 <http://de.wikibooks.org/w/index.php?title=Benutzer:Sys.Bak>
 - 59 <http://de.wikibooks.org/w/index.php?title=Benutzer:ThePacker>
 - 60 http://de.wikibooks.org/w/index.php?title=Benutzer:Theres_no_global_warming
 - 61 <http://de.wikibooks.org/w/index.php?title=Benutzer:Timon.Freitag>
 - 62 <http://de.wikibooks.org/w/index.php?title=Benutzer:Tsch%C3%A4fer>
 - 63 <http://de.wikibooks.org/w/index.php?title=Benutzer:Uwe.kausch>
 - 64 <http://de.wikibooks.org/w/index.php?title=Benutzer:Wei%C3%9FNix>
 - 65 <http://de.wikibooks.org/w/index.php?title=Benutzer:Wolfgangbeyer>
 - 66 <http://de.wikibooks.org/w/index.php?title=Benutzer:Worker>

Abbildungsverzeichnis

- GFDL: Gnu Free Documentation License. <http://www.gnu.org/licenses/fdl.html>
- cc-by-sa-3.0: Creative Commons Attribution ShareAlike 3.0 License. <http://creativecommons.org/licenses/by-sa/3.0/>
- cc-by-sa-2.5: Creative Commons Attribution ShareAlike 2.5 License. <http://creativecommons.org/licenses/by-sa/2.5/>
- cc-by-sa-2.0: Creative Commons Attribution ShareAlike 2.0 License. <http://creativecommons.org/licenses/by-sa/2.0/>
- cc-by-sa-1.0: Creative Commons Attribution ShareAlike 1.0 License. <http://creativecommons.org/licenses/by-sa/1.0/>
- cc-by-2.0: Creative Commons Attribution 2.0 License. <http://creativecommons.org/licenses/by/2.0/>
- cc-by-2.0: Creative Commons Attribution 2.0 License. <http://creativecommons.org/licenses/by/2.0/deed.en>
- cc-by-2.5: Creative Commons Attribution 2.5 License. <http://creativecommons.org/licenses/by/2.5/deed.en>
- cc-by-3.0: Creative Commons Attribution 3.0 License. <http://creativecommons.org/licenses/by/3.0/deed.en>
- GPL: GNU General Public License. <http://www.gnu.org/licenses/gpl-2.0.txt>
- LGPL: GNU Lesser General Public License. <http://www.gnu.org/licenses/lgpl.html>
- PD: This image is in the public domain.
- ATTR: The copyright holder of this file allows anyone to use it for any purpose, provided that the copyright holder is properly attributed. Redistribution, derivative work, commercial use, and all other use is permitted.

- **EURO:** This is the common (reverse) face of a euro coin. The copyright on the design of the common face of the euro coins belongs to the European Commission. Authorised is reproduction in a format without relief (drawings, paintings, films) provided they are not detrimental to the image of the euro.
- **LFK:** Lizenz Freie Kunst. <http://artlibre.org/licence/lal/de>
- **CFR:** Copyright free use.
- **EPL:** Eclipse Public License. <http://www.eclipse.org/org/documents/epl-v10.php>

Copies of the GPL, the LGPL as well as a GFDL are included in chapter Licenses⁶⁷. Please note that images in the public domain do not require attribution. You may click on the image numbers in the following table to open the webpage of the images in your webbrowser.

⁶⁷ Kapitel 13 auf Seite 105

13 Licenses

13.1 GNU GENERAL PUBLIC LICENSE

Version 3, 29 June 2007

Copyright © 2007 Free Software Foundation, Inc.
http://fsf.org/

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed. Preamble

The GNU General Public License is a free, copyleft license for software and other kinds of works.

The licenses for most software and other practical works are designed to take away your freedom to share and change the works. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change all versions of a program—to make sure it remains free software for all its users. We, the Free Software Foundation, use the GNU General Public License for most of our software; it applies also to any other work released this way by its authors. You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for them if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs, and that you know you can do these things.

To protect your rights, we need to prevent others from denying you these rights or asking you to surrender the rights. Therefore, you have certain responsibilities if you distribute copies of the software, or if you modify it: responsibilities to respect the freedom of others.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must pass on to the recipients the same freedoms that you received. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

Developers that use the GNU GPL protect your rights with two steps: (1) assert copyright on the software, and (2) offer you this License giving you legal permission to copy, distribute and/or modify it.

For the developers' and authors' protection, the GPL clearly explains that there is no warranty for this free software. For both users' and authors' sake, the GPL requires that modified versions be marked as changed, so that their problems will not be attributed erroneously to authors of previous versions.

Some devices are designed to deny users access to install or run modified versions of the software inside them, although the manufacturer can do so. This is fundamentally incompatible with the aim of protecting users' freedom to change the software. The systemic pattern of such abuse occurs in the area of products for individuals to use, which is precisely where it is most unacceptable. Therefore, we have designed this version of the GPL to prohibit the practice for those products. If such problems arise substantially in other domains, we stand ready to extend this provision to those domains in future versions of the GPL, as needed to protect the freedoms of users.

Finally, every program is threatened constantly by software patents. States should not allow patents to restrict development and use of software on general-purpose computers, but in those that do, we wish to avoid the special danger that patents applied to a free program could make it effectively proprietary. To prevent this, the GPL assures that patents cannot be used to render the program non-free.

The precise terms and conditions for copying, distribution and modification follow. TERMS AND CONDITIONS 0. Definitions.

"This License" refers to version 3 of the GNU General Public License.

"Copyright" also means copyright-like laws that apply to other kinds of works, such as semiconductor masks.

"The Program" refers to any copyrightable work licensed under this License. Each licensee is addressed as "you," "Licensees" and "recipients" may be individuals or organizations.

To "modify" a work means to copy from or adapt all or part of the work in a fashion requiring copyright permission, other than the making of an exact copy. The resulting work is called a "modified version" of the earlier work or a work "based on" the earlier work.

A work "based on" means either the unmodified Program or a work covered by the Program.

To "propagate" a work means to do anything with it that, without permission, would make you directly or secondarily liable for infringement under applicable copyright law, except executing it on a computer or modifying a private copy. Propagation includes copying, distribution (with or without modification),

making available to the public, and in some countries other activities as well.

To "convey" a work means any kind of propagation that enables other parties to make or receive copies. Mere interaction with a user through a computer network, with no transfer of a copy, is not conveying.

An interactive user interface displays "Appropriate Legal Notices" to the extent that it includes a convenient and prominently visible feature that (1) displays an appropriate copyright notice, and (2) tells the user that there is no warranty for the work (except to the extent that warranties are provided), that licensees may convey the work under this License, and how to view a copy of this License. If the interface presents a list of user commands or options, such as a menu, a prominent item in the list meets this criterion. 1. Source Code.

The "source code" for a work means the preferred form of the work for making modifications to it. "Object code" means any non-source form of a work.

A "Standard Interface" means an interface that either is an official standard defined by a recognized standards body, or, in the case of interfaces specified for a particular programming language, one that is widely used among developers working in that language.

The "System Libraries" of an executable work include anything, other than the work as a whole, that (a) is included in the normal form of packaging a Major Component, but which is not part of that Major Component, and (b) serves only to enable use of the work with that Major Component, or to implement a Standard Interface for which an implementation is available to the public in source code form. A "Major Component", in this context, means a major essential component (kernel, window system, and so on) of the specific operating system (if any) on which the executable work runs, or a compiler used to produce the work, or an object code interpreter used to run it.

The "Corresponding Source" for a work in object code form means all the source code needed to generate, install, and (for an executable work) run the object code and to modify the work, including scripts to control those activities. However, it does not include the work's System Libraries, or general-purpose tools or generally available free programs which are used unmodified in performing those activities but which are not part of the work. For example, Corresponding Source includes interface definition files associated with source files for the work, and the source code for shared libraries and dynamically linked subprograms that the work is specifically designed to require, such as by intimate data communication or control flow between those subprograms and other parts of the work.

The Corresponding Source need not include anything that users can regenerate automatically from other parts of the Corresponding Source.

The Corresponding Source for a work in source code form is that same work. 2. Basic Permissions.

All rights granted under this License are granted for the term of copyright on the Program, and are irrevocable provided the stated conditions are met. This License explicitly affirms your unlimited permission to run the unmodified Program. The output from running a covered work is covered by this License only if the output, given its content, constitutes a covered work. This License acknowledges your rights of fair use or other equivalent, as provided by copyright law.

You may make, run and propagate covered works that you do not convey, without conditions so long as your license otherwise remains in force. You may convey covered works to others for the sole purpose of having them make modifications exclusively for you, or provide you with facilities for running those works, provided that you comply with the terms of this License in conveying all material for which you do not control copyright. Those thus making or running the covered works for you must do so exclusively on your behalf, under your direction and control, on terms that prohibit them from making any copies of your copyrighted material outside their relationship with you.

Conveying under any other circumstances is permitted solely under the conditions stated below. Sublicensing is not allowed; section 10 makes it unnecessary. 3. Protecting Users' Legal Rights From Anti-Circumvention Law.

No covered work shall be deemed part of an effective technological measure under any applicable law fulfilling obligations under article 11 of the WIPO copyright treaty adopted on 20 December 1996, or similar laws prohibiting or restricting circumvention of such measures.

When you convey a covered work, you waive any legal power to forbid circumvention of technological measures to the extent such circumvention is effected by exercising rights under this License with respect to the covered work, and you disclaim any intention to limit operation or modification of the work as a means of enforcing, against the work's users, your or third parties' legal rights to forbid circumvention of technological measures. 4. Conveying Verbatim Copies.

You may convey verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice; keep intact all notices stating that this License and any non-permissive terms added in accord with section 7 apply to the code; keep intact all notices of the absence of any warranty; and give all recipients a copy of this License along with the Program.

You may charge any price or no price for each copy that you convey, and you may offer support or warranty protection for a fee. 5. Conveying Modified Source Versions.

You may convey a work based on the Program, or the modifications to produce it from the Program, in the form of source code under the terms of section 4, provided that you also meet all of these conditions:

a) The work must carry prominent notices stating that you modified it, and giving a relevant date. b) The work must carry prominent notices stating that it is released under this License and any conditions added under section 7. This requirement modifies the requirement in section 4 to "keep intact all notices". c) You must license the entire work, as a whole, under this License to anyone who comes into possession of a copy. This License will therefore apply, along with any applicable section 7 additional terms, to the whole of the work, and all its parts, regardless of how they are packaged. This License gives no permission to license the work in any other way, but it does not invalidate such permission if you have separately received it. d) If the work has interactive user interfaces, each must display Appropriate Legal Notices; however, if the Program has interactive interfaces that do not display Appropriate Legal Notices, your work need not make them do so.

A compilation of a covered work with other separate and independent works, which are not by their nature extensions of the covered work, and which are not combined with it such as to form a larger program, in or on a volume of a storage or distribution medium, is called an "aggregate" if the compilation and its resulting copyright are not used to limit the access or legal rights of the compilation's users beyond what the individual works permit. Inclusion of a covered work in an aggregate does not cause this License to apply to the other parts of the aggregate. 6. Conveying Non-Source Forms.

You may convey a covered work in object code form under the terms of sections 4 and 5, provided that you also convey the machine-readable Corresponding Source under the terms of this License, in one of these ways:

a) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by the Corresponding Source fixed on a durable physical medium customarily used for software interchange. b) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by a written offer, valid for at least three years and valid for as long as you offer spare parts or customer support for that product model, to give anyone who possesses the object code either (1) a copy of the Corresponding Source for all the software in the product that is covered by this License, on a durable physical medium customarily used for software interchange, for a price no more than your reasonable cost of physically performing this conveying of source, or (2) access to copy the Corresponding Source from a network server at no charge. c) Convey individual copies of the object code with a copy of the written offer to provide the Corresponding Source. This alternative is allowed only occasionally and noncommercially, and only if you received the object code with such an offer, in accord with subsection 6b. d) Convey the object code by offering access from a designated place (gratis or for a charge), and offer equivalent access to the Corresponding Source in the same way through the same place at no further charge. You need not require recipients to copy the Corresponding Source along with the object code. If the place to copy the object code is a network server, the Corresponding Source may be on a different server (operated by you or a third party) that supports equivalent copying facilities, provided you maintain clear directions next to the object code saying where to find the Corresponding Source. Regardless of what server hosts the Corresponding Source, you remain obligated to ensure that it is available for as long as needed to satisfy these requirements. e) Convey the object code using peer-to-peer transmission, provided you inform other peers where the object code and Corresponding Source of the work are being offered to the general public at no charge under subsection 6d.

A separable portion of the object code, whose source code is excluded from the Corresponding Source as a System Library, need not be included in conveying the object code work.

A "User Product" is either (1) a "consumer product", which means any tangible personal property which is normally used for personal, family, or household purposes, or (2) anything designed or sold for incorporation into a dwelling. In determining whether a product is a consumer product, doubtful cases shall be resolved in favor of coverage. For a particular product received by a particular user, "normally used" refers to a typical or common use of that class of product, regardless of the status of the particular user or of the way in which the particular user actually uses, or expects or is expected to use, the product. A product is a consumer product regardless of whether the product has substantial commercial, industrial or non-consumer uses,

unless such uses represent the only significant mode of use of the product.

"Installation Information" for a User Product means any methods, procedures, authorization keys, or other information required to install and execute modified versions of a covered work that is transferred to the recipient in perpetuity or for a fixed term (regardless of how the transaction is characterized), the Corresponding Source conveyed under this section must be accompanied by the Installation Information. But this requirement does not apply if neither you nor any third party retains the ability to install modified object code in the User Product (for example, the work has been installed in ROM).

If you convey an object code work under this section in, or with, or specifically for use in, a User Product, and the conveying occurs as part of a transaction in which the right of possession and use of the User Product is transferred to the recipient in perpetuity or for a fixed term (regardless of how the transaction is characterized), the Corresponding Source conveyed under this section must be accompanied by the Installation Information. But this requirement does not apply if neither you nor any third party retains the ability to install modified object code in the User Product (for example, the work has been installed in ROM).

The requirement to provide Installation Information does not include a requirement to continue to provide support service, warranty, or updates for a work that has been modified or installed by the recipient, or for the User Product in which it has been modified or installed. Access to a network may be denied when the modification itself materially and adversely affects the operation of the network or violates the rules and protocols for communication across the network.

Corresponding Source conveyed, and Installation Information provided, in accord with this section must be in a format that is publicly documented (and with an implementation available to the public in source code form), and must require no special password or key for unpacking, reading or copying. 7. Additional Terms.

"Additional permissions" are terms that supplement the terms of this License by making exceptions from one or more of its conditions. Additional permissions that are applicable to the entire Program shall be treated as though they were included in this License, to the extent that they are valid under applicable law. If additional permissions apply only to part of the Program, that part may be used separately under those permissions, but the entire Program remains governed by this License without regard to the additional permissions.

When you convey a copy of a covered work, you may at your option remove any additional permissions from that copy, or from any part of it. (Additional permissions may be written to require their own removal in certain cases when you modify the work.) You may place additional permissions on material, added by you to a covered work, for which you have or can give appropriate copyright permission.

Notwithstanding any other provision of this License, for material you add to a covered work, you may (if authorized by the copyright holders of that material) supplement the terms of this License with terms:

a) Disclaiming warranty or limiting liability differently from the terms of sections 15 and 16 of this License; or b) Requesting preservation of specified reasonable legal notices or author attributions in that material or in the Appropriate Legal Notices displayed by works containing it; or c) Prohibiting misrepresentation of the origin of that material, or requiring that modified versions of such material be marked in reasonable ways as different from the original version; or d) Limiting the use for publicity purposes of names of licensors or authors of the material; or e) Declining to grant rights under trademark law for use of some trade names, trademarks, or service marks; or f) Requiring indemnification of licensors and authors of that material by anyone who conveys the material (or modified versions of it) with contractual assumptions of liability to the recipient, for any liability that these contractual assumptions directly impose on those licensors and authors.

All other non-permissive additional terms are considered "further restrictions" within the meaning of section 10. If the Program as you received it, or any part of it, contains a notice stating that it is governed by this License along with a term that is a further restriction, you may remove that term. If a license document contains a further restriction but permits relicensing or conveying under this License, you may add to a covered work material governed by the terms of that license document, provided that the further restriction does not survive such relicensing or conveying.

If you add terms to a covered work in accord with this section, you must place, in the relevant source files, a statement of the additional terms that apply to those files, or a notice indicating where to find the applicable terms.

Additional terms, permissive or non-permissive, may be stated in the form of a separately written license, or stated as exceptions; the above requirements apply either way. 8. Termination.

You may not propagate or modify a covered work except as expressly provided under this License. Any attempt otherwise to

propagate or modify it is void, and will automatically terminate your rights under this License (including any patent licenses granted under the third paragraph of section 11).

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, you do not qualify to receive new licenses for the same material under section 10.9. Acceptance Not Required for Having Copies.

You are not required to accept this License in order to receive or run a copy of the Program. Ancillary propagation of a covered work occurring solely as a consequence of using peer-to-peer transmission to receive a copy likewise does not require acceptance. However, nothing other than this License grants you permission to propagate or modify any covered work. These actions infringe copyright if you do not accept this License. Therefore, by modifying or propagating a covered work, you indicate your acceptance of this License to do so. 10. Automatic Licensing of Downstream Recipients.

Each time you convey a covered work, the recipient automatically receives a license from the original licensors, to run, modify and propagate that work, subject to this License. You are not responsible for enforcing compliance by third parties with this License.

An "entity transaction" is a transaction transferring control of an organization, or substantially all assets of one, or subdividing an organization, or merging organizations. If propagation of a covered work results from an entity transaction, each party to that transaction who receives a copy of the work also receives whatever licenses to the work the party's predecessor in interest had or could give under the previous paragraph, plus a right to possession of the Corresponding Source of the work from the predecessor in interest, if the predecessor has it or can get it with reasonable efforts.

You may not impose any further restrictions on the exercise of the rights granted or affirmed under this License. For example, you may not impose a license fee, royalty, or other charge for exercise of rights granted under this License, and you may not initiate litigation (including a cross-claim or counterclaim in a lawsuit) alleging that any patent claim is infringed by making, using, selling, offering for sale, or importing the Program or any portion of it. 11. Patents.

A "contributor" is a copyright holder who authorizes use under this License of the Program or a work on which the Program is based. The work thus licensed is called the contributor's "contributor version".

A contributor's "essential patent claims" are all patent claims owned or controlled by the contributor, whether already acquired or hereafter acquired, that would be infringed by some man-

ner, permitted by this License, of making, using, or selling its contributor version, but do not include claims that would be infringed only as a consequence of further modification of the contributor version. For purposes of this definition, "control" includes the right to grant patent sublicenses in a manner consistent with the requirements of this License.

Each contributor grants you a non-exclusive, worldwide, royalty-free patent license under the contributor's essential patent claims, to make, use, sell, offer for sale, import and otherwise run, modify and propagate the contents of its contributor version.

In the following three paragraphs, a "patent license" is any express agreement or commitment, however denominated, not to enforce a patent (such as an express permission to practice a patent or covenant not to sue for patent infringement). To "grant" such a patent license to a party means to make such an agreement or commitment not to enforce a patent against the party.

If you convey a covered work, knowingly relying on a patent license, and the Corresponding Source of the work is not available for anyone to copy, free of charge and under the terms of this License, through a publicly available network server or other readily accessible means, then you must either (1) cause the Corresponding Source to be so available, or (2) arrange to deprive yourself of the benefit of the patent license for this particular work, or (3) arrange, in a manner consistent with the requirements of this License, to extend the patent license to downstream recipients. "Knowingly relying" means you have actual knowledge that, but for the patent license, your conveying the covered work in a country, or your recipient's use of the covered work in a country, would infringe one or more identifiable patents in that country that you have reason to believe are valid.

If, pursuant to or in connection with a single transaction or arrangement, you convey, or propagate by procuring conveyance of, a covered work, and grant a patent license to some of the parties receiving the covered work authorizing them to use, propagate, modify or convey a specific copy of the covered work, then the patent license you grant is automatically extended to all recipients of the covered work and works based on it.

A patent license is "discriminatory" if it does not include within the scope of its coverage, prohibits the exercise of, or is conditioned on the non-exercise of one or more of the rights that are specifically granted under this License. You may not convey a covered work if you are a party to an arrangement with a third party that is in the business of distributing software, under which you make payment to the third party based on the extent of your activity of conveying the work, and under which the third party grants, to any of the parties who would receive the covered work from you, a discriminatory patent license (a) in connection with copies of the covered work conveyed by you (or copies made from those copies), or (b) primarily for and in connection with specific products or compilations that contain the covered work, unless you entered into that arrangement, or that patent license was granted, prior to 28 March 2007.

Nothing in this License shall be construed as excluding or limiting any implied license or other defenses to infringement that may otherwise be available to you under applicable patent law. 12. No Surrender of Others' Freedom.

If conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot comply with both conditions, you may still satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not convey it at all.

13.2 GNU Free Documentation License

Version 1.3, 3 November 2008

Copyright © 2000, 2001, 2002, 2007, 2008 Free Software Foundation, Inc. <<http://fsf.org/>>

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed. 0. PREAMBLE

The purpose of this License is to make a manual, textbook, or other functional and useful document "free" in the sense of freedom: to assure everyone the effective freedom to copy and redistribute it, with or without modifying it, either commercially or noncommercially. Secondly, this License preserves for the author and publisher a way to get credit for their work, while not being considered responsible for modifications made by others.

This License is a kind of "copyleft", which means that derivative works of the document must themselves be free in the same sense. It implements the GNU General Public License, which is a copyleft license designed for free software.

We have designed this License in order to use it for manuals for free software, because free software needs free documentation: a free program should come with manuals providing the same freedoms that the software does. But this License is not limited to software manuals; it can be used for any textual work, regardless of subject matter or whether it is published as a printed book. We recommend this License principally for works whose purpose is instruction or reference. 1. APPLICABILITY AND DEFINITIONS

This License applies to any manual or other work, in any medium, that contains a notice placed by the copyright holder saying it can be distributed under the terms of this License. Such a notice grants a world-wide, royalty-free license, unlimited in duration, to use that work under the conditions stated herein. The "document", below, refers to any such manual or work. Any member of the public is a licensee, and is addressed as "you". You accept the license if you copy, modify or distribute the work in a way requiring permission under copyright law.

For example, if you agree to terms that obligate you to collect a royalty for further conveying from those to whom you convey the Program, the only way you could satisfy both those terms and this License would be to refrain entirely from conveying the Program. 13. Use with the GNU Affero General Public License.

Notwithstanding any other provision of this License, you have permission to link or combine any covered work with a work licensed under version 3 of the GNU Affero General Public License into a single combined work, and to convey the resulting work. The terms of this License will continue to apply to the part which is the covered work, but the special requirements of the GNU Affero General Public License, section 13, concerning interaction through a network will apply to the combination as such. 14. Revised Versions of this License.

The Free Software Foundation may publish revised and/or new versions of the GNU General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies that a certain numbered version of the GNU General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that numbered version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of the GNU General Public License, you may choose any version ever published by the Free Software Foundation.

If the Program specifies that a proxy can decide which future versions of the GNU General Public License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Program.

Later license versions may give you additional or different permissions. However, no additional obligations are imposed on any author or copyright holder as a result of your choosing to follow a later version. 15. Disclaimer of Warranty.

THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION. 16. Limitation of Liability.

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MODIFIES AND/OR CONVEYS THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. 17. Interpretation of Sections 15 and 16.

Examples of suitable formats for Transparent copies include plain ASCII without markup, Texinfo input format, LaTeX input format, SGML or XML using a publicly available DTD, and standard-conforming simple HTML, PostScript or PDF designed for human modification. Examples of transparent image formats include PNG, XCF and JPG. Opaque formats include proprietary formats that can be read and edited only by proprietary word processors, SGML or XML for which the DTD and/or processing tools are not generally available, and the machine-generated HTML, PostScript or PDF produced by some word processors for output purposes only.

The "Title Page" means, for a printed book, the title page itself, plus such following pages as are needed to hold, legibly, the material this License requires to appear in the title page. For works in formats which do not have any title page as such, "Title Page" means the text near the most prominent appearance of the work's title, preceding the beginning of the body of the text.

The "publisher" means any person or entity that distributes copies of the Document to the public.

A section Entitled XYZ means a named subunit of the Document whose title either is precisely XYZ or contains XYZ in parentheses following text that translates XYZ in another language. (Here XYZ stands for a specific section name mentioned below, such as Acknowledgements, "Dedications", Endorsements, or "History".) To "Preserve the Title" of such a section when you modify the Document means that it remains a section Entitled XYZ according to this definition.

The Document may include Warranty Disclaimers next to the notice which states that this License applies to the Document. These Warranty Disclaimers are considered to be included by reference in this License, but only as regards disclaiming warranties; any other implication that these Warranty Disclaimers may have is void and has no effect on the meaning of this License. 2. VERBATIM COPYING

You may copy and distribute the Document in any medium, either commercially or noncommercially, provided that this License, the copyright notices, and the license notice saying this License applies to the Document are reproduced in all copies, and

If the disclaimer of warranty and limitation of liability provided above cannot be given local legal effect according to their terms, reviewing courts shall apply local law that most closely approximates an absolute waiver of all civil liability in connection with the Program, unless a warranty or assumption of liability accompanies a copy of the Program in return for a fee.

END OF TERMS AND CONDITIONS How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively state the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.> Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program. If not, see <<http://www.gnu.org/licenses/>>.

Also add information on how to contact you by electronic and paper mail.

If the program does terminal interaction, make it output a short notice like this when it starts in an interactive mode:

<program> Copyright (C) <year> <name of author> This program comes with ABSOLUTELY NO WARRANTY; for details type 'show w'. This is free software, and you are welcome to redistribute it under certain conditions; type 'show c' for details.

The hypothetical commands 'show w' and 'show c' should show the appropriate parts of the General Public License. Of course, your program's commands might be different; for a GUI interface, you would use an "about box".

You should also get your employer (if you work as a programmer) or school, if any, to sign a "copyright disclaimer" for the program, if necessary. For more information on this, and how to apply and follow the GNU GPL, see <<http://www.gnu.org/licenses/>>.

The GNU General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Lesser General Public License instead of this License. But first, please read <<http://www.gnu.org/philosophy/why-not-lgpl.html>>.

that you add no other conditions whatsoever to those of this License. You may not use technical measures to obstruct or control the reading or further copying of the copies you make or distribute. However, you may accept compensation in exchange for copies. If you distribute a large enough number of copies you must also follow the conditions in section 3.

You may also lend copies, under the same conditions stated above, and you may publicly display copies. 3. COPYING IN QUANTITY

If you publish printed copies (or copies in media that commonly have printed covers) of the Document, numbering more than 100, and the Document's license notice requires Cover Texts, you must enclose the copies in covers that carry, clearly and legibly, all these Cover Texts: Front-Cover Texts on the front cover, and Back-Cover Texts on the back cover. Both covers must also clearly and legibly identify you as the publisher of these copies. The front cover must present the full title with all words of the title equally prominent and visible. You may add other material on the covers in addition. Copying with changes limited to the covers, as long as they preserve the title of the Document and satisfy these conditions, can be treated as verbatim copying in other respects.

If the required texts for either cover are too voluminous to fit legibly, you should put the first ones listed (as many as fit reasonably) on the actual cover, and continue the rest on adjacent pages.

If you publish more than 100 copies of the Document numbering more than 100, you must either include a machine-readable Transparent copy along with each Opaque copy, or state in or with each Opaque copy a computer-network location from which the general network-using public has access to download using public-standard network protocols a complete Transparent copy of the Document, free of added material. If you use the latter option, you must take reasonably prudent steps, when you begin distribution of Opaque copies in quantity, to ensure that this Transparent copy will remain thus accessible at the stated location until at least one year after the last time you distribute an Opaque copy (directly or through your agents or retailers) of that edition to the public.

It is requested, but not required, that you contact the authors of the Document well before redistributing any large number of copies, to give them a chance to provide you with an updated version of the Document. 4. MODIFICATIONS

You may copy and distribute a Modified Version of the Document under the conditions of sections 2 and 3 above, provided that you release the Modified Version under precisely this License, with the Modified Version filling the role of the Document, thus licensing distribution and modification of the Modified Version to whoever possesses a copy of it. In addition, you must do these things in the Modified Version:

- * A. Use in the Title Page (and on the covers, if any) a title distinct from that of the Document, and that of previous versions (which should, if there were any, be listed in the History section of the Document). You may use the same title as a previous version if the original publisher of that version gives permission.
- * B. List on the Title Page, as authors, one or more persons or entities responsible for authorship of the modifications in the Modified Version, together with at least five of the principal authors of the Document (all of its principal authors, if it has fewer than five), unless they release you from this requirement.
- * C. State on the Title page the name of the publisher of the Modified Version, as the publisher.
- * D. Preserve all the copyright notices of the Document.
- * E. Add an appropriate copyright notice for your modifications adjacent to the other copyright notices.
- * F. Include, immediately after the copyright notices, a license notice giving the public permission to use the Modified Version under the terms of this License, in the form shown in the Addendum below.
- * G. Preserve in that license notice the full lists of Invariant Sections and required Cover Texts given in the Document's license notice.
- * H. Include an unaltered copy of this License.
- * I. Preserve the section entitled "History", Preserve its Title, and add to it an item stating at least the title, year, new authors, and publisher of the Modified Version as given on the Title Page. If there is no section entitled "History" in the Document, create one stating the title, year, authors, and publisher of the Document as given on its Title Page, then add an item describing the Modified Version as stated in the previous sentence.
- * J. Preserve the network location, if any, given in the Document for public access to a stamped copy of the Document, and likewise the network locations given in the Document for previous versions if it was based on. These may be placed in the "History" box. You may omit a network location for a work that was published at least four years before the Document itself, or if the original publisher of the version it refers to gives permission.
- * K. For any section entitled "Acknowledgements" or "Dedications", Preserve the Title of the section, and preserve in the section all the substance and tone of each of the contributor acknowledgements and/or dedications given therein.
- * L. Preserve all the Invariant Sections of the Document, unaltered in their text and in their titles. Section numbers or the equivalent are not considered part of the section titles.
- * M. Delete any section entitled "Endorsements". Such a section may not be included in the Modified Version.
- * N. Do not retitle any existing section to be entitled "Endorsements" to conflict in title with any Invariant Section.
- * O. Preserve any Warranty Disclaimers.

If the Modified Version includes new front-matter sections or appendices that qualify as Secondary Sections and contain no material copied from the Document, you may at your option designate some or all of these sections as invariant. To do this, add their titles to the list of Invariant Sections in the Modified Version's license notice. These titles must be distinct from any other section titles.

13.3 GNU Lesser General Public License

GNU LESSER GENERAL PUBLIC LICENSE

Version 3, 29 June 2007

Copyright © 2007 Free Software Foundation, Inc. <<http://fsf.org/>>

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

This version of the GNU Lesser General Public License incorporates the terms and conditions of version 3 of the GNU General Public License, supplemented by the additional permissions listed below. 0. Additional Definitions.

As used herein, "this License" refers to version 3 of the GNU Lesser General Public License, and the "GNU GPL" refers to version 3 of the GNU General Public License.

"The Library" refers to a covered work governed by this License, other than an Application or a Combined Work as defined below.

An "Application" is any work that makes use of an interface provided by the Library, but which is not otherwise based on the Library. Defining a subclass in a class defined by the Library is deemed a mode of using an interface provided by the Library.

A "Combined Work" is a work produced by combining or linking an Application with the Library. The particular version of the Library with which the Combined Work was made is also called the "Linked Version".

The "Minimal Corresponding Source" for a Combined Work means the Corresponding Source for the Combined Work, excluding any source code for portions of the Combined Work that, considered in isolation, are based on the Application, and not on the Linked Version.

You may add a section entitled "Endorsements", provided it contains nothing but endorsements of your Modified Version by various parties—for example, statements of peer review or of the text has been approved by an organization as the authoritative definition of a standard.

You may add a passage of up to five words as a Front-Cover Text, and a passage of up to 25 words as a Back-Cover Text, to the end of the list of Cover Texts in the Modified Version. Only one passage of Front-Cover Text and one of Back-Cover Text may be added by (or through arrangements made by) any one entity. If the Document already includes a cover text for the same cover, previously added by you or by arrangement made by the same entity you are acting on behalf of, you may not add another, but you may replace the old one, on explicit permission from the previous publisher that added the old one.

The author(s) and publisher(s) of the Document do not by this License give permission to use their names for publicity or to assert or imply endorsement of any Modified Version. 5. COMBINING DOCUMENTS

You may combine the Document with other documents released under this License, under the terms defined in section 4 above for modified versions, provided that you include in the combination all of the Invariant Sections of all the original documents, unmodified, and list them all as Invariant Sections of your combined work in the license notice, and that you preserve all their Warranty Disclaimers.

The combined work need only contain one copy of this License, and multiple identical Invariant Sections may be replaced with a single copy. If there are multiple Invariant Sections with the same name but different contents, make the title of each such section unique by adding at the end of it, in parentheses, the name of the original author or publisher of that section if known, or else a unique number. Make the same adjustment to the section titles in the list of Invariant Sections in the license notice of the combined work.

In the combination, you must combine any sections entitled "History" in the various original documents, forming one section entitled "History"; likewise combine any sections entitled "Acknowledgements", and any sections entitled "Dedications". You must delete all sections entitled "Endorsements". 6. COLLECTIONS OF DOCUMENTS

You may make a collection consisting of the Document and other documents released under this License, and replace the individual copies of this License in the various documents with a single copy that is included in the collection, provided that you follow the rules of this License for verbatim copying of each of the documents in all other respects.

You may extract a single document from such a collection, and distribute it individually under this License, provided you insert a copy of this License into the extracted document, and follow this License in all other respects regarding verbatim copying of that document. 7. AGGREGATION WITH INDEPENDENT WORKS

A compilation of the Document or its derivatives with other separate and independent documents or works, in or on a volume of a storage or distribution medium, is called an aggregate if the copyright resulting from the compilation is not used to limit the legal rights of the compilation's users beyond what the individual works permit. When the Document is included in an aggregate, this License does not apply to the other works in the

aggregate which are not themselves derivative works of the Document.

If the Cover Text requirement of section 3 is applicable to these copies of the Document, then if the Document is less than one half of the entire aggregate, the Document's Cover Texts may be placed on covers that bracket the Document within the aggregate, or the electronic equivalent of covers if the Document is in electronic form. Otherwise they must appear on printed covers that bracket the whole aggregate. 8. TRANSLATION

Translation is considered a kind of modification, so you may distribute translations of the Document under the terms of section 4. Replacing Invariant Sections with translations requires special permission from their copyright holders, but you may include translations of some or all Invariant Sections in addition to the original versions of these Invariant Sections. You may include a translation of this License, and all the license notices in the Document, and any Warranty Disclaimers, provided that you also include the original English version of this License and the original versions of those notices and disclaimers. In case of a disagreement between the translation and the original version of this License or a notice or disclaimer, the original version will prevail.

If a section in the Document is entitled "Acknowledgements", "Dedications", or "History", the requirement (section 4) to Preserve its Title (section 1) will typically require changing the actual title. 9. TERMINATION

You may not copy, modify, sublicense, or distribute the Document except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, or distribute it is void, and will automatically terminate your rights under this License.

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, receipt of a copy of some or all of the same material does not give you any rights to use it. 10. FUTURE REVISIONS OF THIS LICENSE

The Free Software Foundation may publish new, revised versions of the GNU Free Documentation License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns. See <http://www.gnu.org/copyleft/>.

Each version of the License is given a distinguishing version number. If the Document specifies that a particular numbered version of this License or any later version applies to it, you have the option of following the terms and conditions either of

that specified version or of any later version that has been published (not as a draft) by the Free Software Foundation. If the Document does not specify a version number of this License, you may choose any version ever published (not as a draft) by the Free Software Foundation. If the Document specifies that a proxy can decide which future versions of this License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Document. 11. RELICENSING

"Massive Multiauthor Collaboration Site" (or "MMC Site") means any World Wide Web server that publishes copyrightable works and also provides prominent facilities for anybody to edit those works. A public wiki that anybody can edit is an example of such a server. A "Massive Multiauthor Collaboration" (or "MMC") contained in the site means any set of copyrightable works thus published on the MMC site.

"CC-BY-SA" means the Creative Commons Attribution-Share Alike 3.0 license published by Creative Commons Corporation, a not-for-profit corporation with a principal place of business in San Francisco, California, as well as future copyleft versions of that license published by that same organization.

Incorporate" means to publish or republish a Document, in whole or in part, as part of another Document.

An MMC is eligible for relicensing if it is licensed under this License, and if all works that were first published under this License somewhere other than this MMC, and subsequently incorporated in whole or in part into the MMC, (1) had no cover texts or invariant sections, and (2) were thus incorporated prior to November 1, 2008.

The operator of an MMC Site may republish an MMC contained in the site under CC-BY-SA on the same site at any time before August 1, 2009, provided the MMC is eligible for relicensing. ADDENDUM: How to use this License for your documents

To use this License in a document you have written, include a copy of the License in the document and put the following copyright and license notices just after the title page:

Copyright (C) YEAR YOUR NAME. Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.3 or any later version published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts. A copy of the license is included in the section entitled "GNU Free Documentation License".

If you have Invariant Sections, Front-Cover Texts and Back-Cover Texts, replace the "with ... Texts." line with this:

with the Invariant Sections being LIST THEIR TITLES, with the Front-Cover Texts being LIST, and with the Back-Cover Texts being LIST.

If you have Invariant Sections without Cover Texts, or some other combination of the three, merge those two alternatives to suit the situation.

If your document contains nontrivial examples of program code, we recommend releasing these examples in parallel under your choice of free software license, such as the GNU General Public License, to permit their use in free software.

4. Combined Works.

You may convey a Combined Work under terms of your choice that, taken together, effectively do not restrict modification of the portions of the Library contained in the Combined Work and reverse engineering for debugging such modifications, if you also do each of the following:

- * a) Give prominent notice with each copy of the Combined Work that the Library is used in it and that the Library and its use are covered by this License.
- * b) Accompany the Combined Work with a copy of the GNU GPL and this license document.
- * c) For a Combined Work that displays copyright notices during execution, include the copyright notice for the Library among these notices, as well as a reference directing the user to the copies of the GNU GPL and this license document.
- * d) Do one of the following: o) Convey the Minimal Corresponding Source under the terms of this License, and the Corresponding Application Code in a form suitable for, and under terms that permit, the user to recombine or relink the Application with a modified version of the Linked Version to produce a modified Combined Work, in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source. o 1) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (a) uses at run time a copy of the Library already present on the user's computer system, and (b) will operate properly with a modified version of the Library that is interface-compatible with the Linked Version.
- * e) Provide Installation Information, but only if you would otherwise be required to provide such information under section 6 of the GNU GPL, and only to the extent that such information is necessary to install and execute a modified version of the Combined Work produced by recombining or relinking the Application with a modified version of the Linked Version. (If you use option 4d, the Installation Information must accompany the Minimal Corresponding Source and Corresponding Application Code. If you use option 4d1, you must provide the Installation Information in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.)

5. Combined Libraries.

You may place library facilities that are a work based on the Library side by side in a single library together with other library facilities that are not Applications and are not covered by this License, and convey such a combined library under terms of your choice, if you do both of the following:

- * a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities, conveyed under the terms of this License.
- * b) Give prominent notice with the combined library that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

6. Revised Versions of the GNU Lesser General Public License.

The Free Software Foundation may publish revised and/or new versions of the GNU Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library as you received it specifies that a certain numbered version of the GNU Lesser General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that published version or of any later version published by the Free Software Foundation. If the Library as you received it does not specify a version number of the GNU Lesser General Public License, you may choose any version of the GNU Lesser General Public License ever published by the Free Software Foundation.

If the Library as you received it specifies that a proxy can decide whether future versions of the GNU Lesser General Public License shall apply, that proxy's public statement of acceptance of any version is permanent authorization for you to choose that version for the Library.