

## Commands

ALIAS

ATTRIB

BATCH-files

BAT

BAC

BEEP

CD or CHDIR

4HELP

CDD

CLS

COPY

DEL

DESCRIBE

DIR

DIRS

EXCEPT

EXIT(/S)

EXT

FIND

FREE

GLOBAL

HELP

HISTORY

INDES(INDESI)

KILL

KD

LIST

LOAD

MAX

MD or MKDIR

MEMORY

MIN

MOVE

MOUSE

NEW

POPD

PS

PUSHD

PROMPT

RD or RMDIR

REN

REST

SET

STOP

TREE

TYPE

UPDATE

VER

WINBATCH

WHERE

YESTERDAY

## **Redirection and Devices.**

### **Execution of programs**

COM-files  
EXE-files  
BAT-files  
PIF-files  
WBT-files  
PBT-files

### **EDIT-keys**

### **MOUSE**

### **MULTIPLE COMMANDS**

### **SYSTEM VARIABLES**

### **COPYRIGHT**

**COPYRIGHT 1991 Hans Paijmans.**

**Sorry, this program is NOT public domain. But feel free to use it for non-commercial applications. Criminals, fascists, zionists and racists, or organizations of that ugly breed are NOT welcome to use my work and will be prosecuted if they do.**

## **MULTIPLE COMMANDS.**

You may enter more than one command on the command-line, if separated by a caret (^) between spaces. Note that the commandline may only be 80 characters wide!



## 4DOS

Command-line interpreter from 'JP Software'. Beats COMMAND.COM in about every aspect you might think of.

PSHELL was written to make the file-descriptions of 4DOS accessible to WINDOWS-users and to add a few other DOS-enhancements that 4DOS-users have become addicted to.

The maker of PSHELL is not in any way affiliated with 4DOS, other than as a very satisfied user.

**4HELP**

Executes 4DOS-help (if in path).

**ALIAS** [alias "expansion"]

Without parameters: displays list of aliases.

With parameters in the form ALIASNAME=EXPANSION creates an alias with that meaning. The double quote is obligatory if the expansion has spaces or slashes inbedded.

e.g.:

```
dw "dir *.bak /w
```

Pshell has room for 25 aliases of 40 chars each. The first part (the alias itself) can be no longer than 5 characters; the expansion is maximal 40 long.

Aliases consist of a single word and do not accept parameters.

They are expanded only if they are typed as a single word on the command-line after the prompt or read from a PBT-file.

Aliases beginning with % are treated as systemvariables (see below).

ATTRIB [+/-[R][S][H][A]] filespec

Displays and/or changes the attributes of the files.

BATCH-files.

Batch-files are supported by PSHELL; typical **4DOS**-commands are only possible if a 4DOS-shell is started as command-processor. See also **WINBATCH** - and the **EXEC**.-entry. BAT, PIF and WTB-files are all started by typing the first part of the filename.

PSHELL also supports a rudimentary batch-language of its own. Of course these files are ASCII-files too and their names have the PBT-suffix.

**4PSHELL.PBT** is a special batch-file that executes every time a copy of PSHELL is started.

Currently only the

ECHO on|off|message

PAUSE message

REM message

Parameters 1-9 (%1 .. %9)

IF (parameter == string) command

NOTE THE SPACES AROUND THE == SIGN!

Also EQ, NE, LT, GT, LE, GE as string operations.

EXISTS filename

GOTO label

:label

are supported.

Note that **SYSTEM VARIABLES** may be used after ECHO and IF-statements.

**NOTA BENE:** if a normal DOS-batch-file conforms to the PSHELL-syntax (that is a subset of DOS), you may use it as a PBT-file, just by adding an exclamation mark.  
So the command

SEARCH!

will execute the SEARCH.BAT inside PSHELL (without shelling to DOS) if no SEARCH.PBT is found. Of course PSHELL will shriek **\*\*\* Bad command** etc. if it encounters commands it doesn't recognize.

see also the command **BAT**.

BAT [ON] [OFF]

BAT ON causes all dos BAT-files to be executed as PBT-files (so no exclamation mark has to be added; see **BATCH**-files).

BAT OFF switches this option off. To execute a BAT-file as a PBT-file the ! has to be added to the filename.

BAT on its own returns ON or OFF.

**BAC source destination**

Copies new and younger files from source to destination.

**BEEP** [number of beeps]

BEEPS an integer number of beeps (default 1).

**CD** or **CHDIR** [directory]

Changes the default directory .

**CDD** [d:directory]

Changes both default drive and directory

## **CLS**

Guess what...

May be typed 'blind' when the PSHELL-window is scrolled up, to return quickly to the command-prompt.

## **COPY**

Copies files from one disk or directory to another. See also MOVE. Please note that none of the common switches (like in DOS: /A, /B or /V) are implemented. However; file-descriptions are carried with the files that are copied.

If during a COPY-operation the disk becomes full, you will be prompted for a new disk

**DEL filespec**

Deletes files from a directory

**DESCRIBE filespec** [stringargument]

Adds a description of up to 40 characters to a filename or directory name. The description is stored in the hidden DESCRIPT.ION-file in that directory and is identical to the 4DOS-file with that name.

A stringargument may be added, which will automatically update the description(s) of the file(s) in **filespec**. If spaces occur in the string, it should be preceded by a double quote.

## **DIR**

Shows directory with 4DOS file-descriptions. Wildcards \* and ? may be used to mask filenames.

Default order is sorted on name or according to the parameter [dirsrt] in PSHELL.INI..

### Parameters:

/w Displays filenames in columns.  
/a Displays fileattributes in the place of descriptions.  
/e Sort on order of entry (like DOS).  
/s Sort on size.  
/t Sort on date/time.  
/d Sort on description.  
/n Sort on name.

**DIRS**

Displays stack of directory names pushed by PUSHD.

## EDIT

Editing-keys are      CTRL-S and CTRL-D    or    Arrow left and Arrow-right,  
                          CTRL-E and CTRL-X    or    Arrow up and arrow down,  
                          CTRL-A and CTRL-F    or    Home and END.  
                          CTRL-V                or    INS (Toggles insert and overwrite).  
                          CTRL-G                or    DEL  
                          BACKSPACE (destructive).  
                          ESCAPE  
                          TAB  
                          F8  
                          F9

Arrow Up and Down cycle back and forth through the HISTORY-list and so allow repeating commands.

Home and End go to begin resp. end of the command line.

ESC cleans the command-line.

BACKSPACE and DEL delete respectively the character before and under the cursor.

TAB and F9 complete a command-line with the next file in a filelist, that has the same characters in its beginning as the partial filename (may be zero) in the command-line. The difference is that F9 first reads the default-directory, while TAB works with the last file-selection.

F8 cycles backward through the selection.

E.g.:

After DIR \PATH\\*.BAK you may use a command like

```
DEL \PATH\xyz<TAB>
```

to cycle through all files in \PATH, beginning with xyz and with the extension .BAK.

```
DEL xyz<F8>
```

however, reads your default-directory and cycles through all files that start with xyz.

Note that TAB only looks to the filenames, not to the preceding path. Therefore you will have to type the PATH when you use TAB for a different directory than the default.

See also **MOUSE.**

**EXCEPT (filespec)** command...

Executes command on all files except those between parentheses.

e.g.

EXCEPT (\*.EXE \*.COM) COPY \*.\* B:

which copies all files to B: except the EXE- and COM-files.

Between the filespecs between the parenthesis should be spaces.

N.B. Hidden or readonly files with those specifications become un-hidden and read-write after an EXCEPT operation!

## **EXEC**

Each directory in the path is checked for respectively COM, EXE, PBT,PIF, BAT, and WBT -files with the given name. If a file is found, it is executed.

N.B. When DOS-applications exit to the DOS-command line, WINDOWS immediatly snatches the screen back, obliterating any output of the program. These programs are best started from a batch-file with a PAUSE in its last line.

PSHELL has a rudimentary batch-language (see **BATCH**) with the PBT-extension for files. We suggest you look into the possibilities of **WINBATCH**, if you need batch-files for WINDOWS.

## **EXIT**

Exits from this instance of Pshell.

/S exits and saves the dimensions of the PSHELL-window, the prompt and the order in which the directory is displayed.

**EXT**

Without parameters: Gives the list with extension - application combinations.

With parameters : adds new extension-application combination to the list or removes them.

e.g.

```
EXT DOC E:\WORD\WINWORD
```

NOTA BENE: Parameters and PSHELL.INI discontinued after version 2.08. Using the first 25 extensions from WIN.INI after that.

## **FIND**

FIND *filespec string* [/I] [/F]

Searches in *filespec* for the occurrence of *string*. If found, it returns on default both the filename and the line in which the string occurs (limited to 80 characters).

If *string* has embedded spaces, it should be surrounded by double quotes.

/I Ignore case

/F Return filename only.

Example

```
GLOBAL FIND *.TXT "Sincerely yours" /I
```

To find and display all TXT-files in all subdirectories, which have the words *Sincerely yours* in them.

**GLOBAL command [filespec]**

Executes an internal command like DEL or COPY for all directories under the default directory.

For more information, select the Edit menu command name.

**FREE** [drive:]

Shows free space on disk.

**HELP**

Starts this HELP-file.

**HELP keyword**

Starts helpfile on specific keyword.

## **HISTORY**

Displays list with last 20 commands. May be edited with the EDIT-keys.

**INDES [NOT] "string" command filespec**  
**INDESI**

OK, so I made this one up. So what?

**INDES** means **In DESCRIPTION** and is followed by a string between double quotes, followed by a normal command like **DIR filename** or **REN** or whatever. It checks if the string occurs in the **description** of the files and ignores them if the string does not occur. Alternatively you can use **INDES NOT "string"**, which ignores files when the string **DOES** occur in the description. Not bad eh?

**INDES** is case-sensitive. **INDES** has a companion command, **INDESI**, which ignores case.

## **LIST or TYPE**

Display ASCII-files at the terminal. Wildcards are allowed.

/c Adds CR/LF to files that only have LF's.

/s Suppresses weird characters (e.g. for scanning an executable).

/w trims bit 8 (for WORDSTAR files).

**LOAD** filename [parms]

Loads an executable and minimizes it as an ICON.

**KD** dirname

Deletes and removes all files and directories under dirname, inclusive dirname itself.

**KILL** processnumber.

Kills the window with that processnumber.

NOTA BENE: The processnumbers are not static, but change after minimizing, maximizing and other actions. Always do a PS immediatly before using KILL, MIN, MAX or REST.

**MAX** processnumber

Maximizes the window with that processnumber.

NOTA BENE: The processnumbers are not static, but change after minimizing, maximizing and other actions. Always do a PS immediatly before using KILL, MIN, MAX or REST.

**MD** or **MKDIR** directory

Creates new directory.

## **MEMORY**

Displays free memory (RAM).

**MIN**

Minimizes the window with that processnumber.

NOTA BENE: The processnumbers are not static, but change after minimizing, maximizing and other actions. Always do a PS immediatly before using KILL, MIN, MAX or REST.

**MOVE**

Works like COPY, but removes original files.

## **MOUSE**

The mouse in PSHELL may be used to click with the left button on a string somewhere in PSHELL's window. That string will be inserted in the command-line, complete with slashes and points (only spaces will separate string). Useful to pick up a long path after a WHERE-command and put it in the command-line, or to start applications from a directory.

Singleclicking the left button will insert the word pointed at in the command-line.

Doubleclicking the left button will select the string and immediately launch it.

Singleclicking the right button will insert spaces in the command-line

Doubleclicking the right button will also launch the command-line.

Clicking the left-button on the **commandline** moves the cursor to that place.

clicking the left- and right button on the **prompt** cycles up and down through the historylist.

**NEW** [ambig.filename]

Shows files newer than 'yesterday' where 'yesterday' is the previous date on which NEW was used. Yesterday may be reset using **YESTERDAY** and is saved in the PSHELL.INI file.

**POPD**

Pops a directory from the directorystack, which was put there by PUSHD

**PS**

Gives a list of active windows with processnumbers.

NOTA BENE: The processnumbers are not static, but change after minimizing, maximizing and other actions. Always do a PS immediatly before using KILL, MIN, MAX or REST.

## PROMPT

Like its DOS counterpart, PROMPT lets you change the prompt. Without parameters the prompt defaults to \$p\$.g.

Prompt accepts a string, which may end with a number of \$(character) combinations.

These characters are

\$	The \$-sign itself.
_	The underscore inserts a carriage/linefeed
d	The current date
e	The ASCII escape character.
g	The > sign
l	The < sign
n	default drive letter
P	Path in uppercase
p	path in lower case
s	The space character

A string may be placed before the first \$-character.

Example

**PROMPT This is an idiotic prompt\$\_t\$\$P\$g**

will create the following prompt:

**This is an idiotic prompt**

**12:02:11 D:\PSHELL>**

**RD** or **RMDIR** directory

Removes directory.

**REN** filespec filespec

Renames files. 4DOS-descriptions are preserved with the new filename. Ambiguous filenames (\* and ?) are allowed. Descriptions are carried over.

**PUSHD** directory

Pushes the current directory on a stack and changes to another directory.

**REST**

Restores the window with that processnumber.

NOTA BENE: The processnumbers are not static, but change after minimizing, maximizing and other actions. Always do a PS immediatly before using KILL, MIN, MAX or REST.

## **SET**

Displays settings of DOS environment.  
Does not *SET* it!

**STOP**

Exits from WINDOWS. Asks for confirmation first (answer 'Y' in caps). WINDOWS will check for open files in other applications.

**STOP!**

(with exclamation mark) does NOT ask for confirmation when closing Windows.

**TREE** [directory]

Displays all directories under the current directory

**UPDATE source destination**

Copies younger files from source to destination, BUT ONLY if destination already exists.

**VER**

Displays the version numbers of PSHELL, MS-DOS and MS-WINDOWS.

filespec

Ambiguous file-specification (uses \* and/or ?).

filename

Unambiguous filename.

## **SYSTEM VARIABLES**

Pshell knows a number of system variables. They have the general form %qxyz (the percent is obligatory).

%\_BAT ON or OFF according to the last **BAT**-command.  
%\_CWD current drive and directory in the format d:\directory  
%\_DATE current date in the format yyyy-mm-dd  
%\_YESTERDAY the yesterday-variable (see commands).  
%\_TIME current time in the format hh:mm:ss  
%\_DISK current disk as character (e.g. C )  
%\_DOW day of the week (mon, tue, wed etc.)  
%\_SYSDIR Windows systemdirectory (e.g. C:\WINDOWS\SYSTEM ).  
%\_WINMODE returns ENHANCED, STANDARD or REAL.

Aliases that consist of % followed by no more than four characters are treated as system variables.

System variables are used in IF- and ECHO-statements.

## **WINBATCH**

Files with WBT-extension should adhere to the WINBATCH-syntax from Morris Wilson. If it does, and if it is in the path, and if the WINBATCH.EXE is in the path too, it will perform as a 'normal' batchfile.

WINBATCH is available as shareware from any good BBS. It will keep nagging you for registering every first time you use it in a Windows-session. Apart from that it is one of the finest extensions for Windows I have ever seen.

**WHERE** filespec

Relatively slow filefind, but it'll get there. Searches the default-directory and all directories below for the filespec given.

If you have more harddisks, this would be a sensible PBT-file:

```
WHEREIS.PBT
```

```
PUSHD c:\
WHERE %1
CD D:\
WHERE %1
POPD
```

Which pushes the current drive and directory, changes to the root of C, does a WHERE in the C-drive, changes to the root of D:, searches the D:=drive and returns to the original drive and directory.

YESTERDAY [yyyy:mm:dd]

YESTERDAY on it own shows the last date **NEW** was used. If followed by a date, that date is inserted in YESTERDAY.

NOTA BENE: no check is done on the correctness of the datestring. It is your own responsibility to use the correct format yyyy:mm:dd!!!

## REDIRECTION

Sorry: only redirection by the > and >> signs is supported, and that only in PSHELL-output. I feel that either your commands run under COMMAND.COM (or 4DOS), in which case the DOS-shell will take care of those details, or else they run under WINDOWS, with its wonderful data exchange.

Somebody from Finland asked for the

### **COPY CON Filename**

possibility. I use it myself all the time under DOS for short batch-files and the like. So the CON: device (with or without colon) is recognized by PSHELL.