

The Korn Shell: Don't Worry, it's in There

Parts of this presentation borrowed without permission from *The Korn Shell: Command and Programming Language* by Bolsky and Korn.

What's in There

- Functions
- Aliases
- History
- Partial File Name Completion
- Command Line Editing
- Arrays
- Tilde Expansion
- Job Control (if OS supported)

MICROCOMPUTER
DIVISION

QUALITY • PEOPLE • PERFORMANCE

Agenda

- Korn Shell Features
- Korn Shell Pitfalls
- Questions and (hopefully) Answers

Pitfalls

- Path

A ":" at the end or the beginning or "::" means current directory search.

- I/O redirection

Order dependent: `cmd > file 2>&1 ; cmd 2>&1 > file`

- Exported variables

- Exported aliases and functions

- Tracked aliases

- IFS

MICROCOMPUTER
DIVISION

QUALITY • PEOPLE • PERFORMANCE

Examples of environment variable assignments:

```
VAR="some text and a $ANOTHERVAR"
VAR2='text $ANOTHERVAR'
var="different from VAR"
export var=3
typeset -ix intvar=7
X[2*(i+1)]=6
```

Examples of aliases:

```
alias ll="ls -l"
alias vim="vi [Mm]akefile"
alias sh="ksh "
alias phone="$EDITOR $HOME/p/phone.list"
alias l="fg %1"
alias cmd=". $SHELLDIR/cmd; cmd"
```

Examples of Functions:

```
ls() {
if [ $# -gt 0 ] ; then
    /bin/ls $@
else
    /bin/ls -CpqF
fi
}
```

```
chuid() {
typeset OLDPS1="$PS1"
export PS1="# "
cuid
PS1="$OLDPS1"
}
```

Accessing History:

41\$ history

```
32      ftp adagio
33      ll
34      cd ..
35      ps2c adagio/kshrc.dep adagio/profile.dep adagio/setpath adagio/setterm
36      ll
37      exit
38      vi .profile
39      ls -l .profile
40      co -l .profile .shell/profile.dep
41      history
```

```
42$ r          # Repeats last command
42$ r 38        # Repeats command number 38
42$ r ft        # Repeats command starting with the string "ft"
42$ r old=new   # Repeats last command replacing "old" with "new"
```

Exiting History:

```
42$ <ESC>/adagio
42$ ps2c adagio/kshrc.dep adagio/profile.dep adagio/setpath adagio/setterm
42$ ftp adagio
```

Korn Shell extensions or changes:

```
/      Searches backwards instead of forwards (n continues backwards search)
v      Drops you into "vi"
*      Causes an * to append to current word and file name generation attempted
[c]_   Recalls [c] word of previous command and inserts it (no c last word)
#      Comments lines and inserts it into the history
=      Lists filenames that match current word if appended with *
@letter Alias list is searched for _letter alias and alias executed
```

Shell Builtins:

```
$30 pwd
/mnt/qbarnes
$31 cd /etc
$32 cd -
/mnt/qbarnes
$33 cd -
/etc
```

```
$35 cd /np.gone/devel/qbarnes/src/sys/sys
$35 cd qbarnes dweber
/np.gone/devel/dweber/src/sys/sys
```

alias	# Lists all aliases
alias -t	# Lists all tracked aliases
unalias name	# Removes alias
jobs [-l]	# Lists all current jobs
print	# Output mechanism for ksh
print -s "string"	# Places "string" into history file
print -n n	# Places "string" to file descriptor "n"
read name	# Place input into "name" var
read -p	# Read from input pipe
read -s	# read from history
typeset -L var=val	# Left justify output
typeset -L3 var=val	# Left justify output with field with 3 chars
typeset -Z	# Right justify fill with leading zeros
set	# Sets options and values
unset [-f] name	# Removes name (-f if a function)
whence file	# displays path of file (like "which")
type file	# Long version of "whence"

```
# $Header$

#
# Aliases
#
alias sh="ksh "
alias bsh="/bin/sh "
alias h="fc -l"
alias md="mkdir"
alias rd="rmdir"
alias l="ls"
alias ll="ls -l"
alias lr="ls -lR"
alias la="ls -la"
alias lt="ls -lt"
alias m="more"
alias j="jobs"
alias vim="vi [Mm]akefile"
alias vt="vi -t"
alias vm="vmail -inc"
alias sigh="echo Sigh!"
alias phone="$EDITOR $HOME/p/phone.list"
alias bones="ls -l $HACKHOME/bones*"

if [ -f $HOME/.noteshost ]
then
    alias N="autoseq"
else
    alias N="box -q $SHELL nf"
fi

if [ -d $HOME/notes ]
then
    alias qbn="notes $HOME/notes/qbn"
else
    alias qbn="echo Wrong machine."
fi

#
# Functions
#

ls() {
if [ $# -gt 0 ] ; then
    /bin/ls $@
else
    /bin/ls -CpqF
fi
}

#
# Read system dependent configuration.
#

if [ -r $SHELLDIR/envfile.dep ]
then
    . $SHELLDIR/envfile.dep
fi
```


.profile Mon Dec 19 11:30:17 1988 1

```
# $Header: .profile,v 1.3 88/12/18 22:03:22 qbarnes Exp $

# Location of Shell's files.

[ -d $HOME/.shell ] && SHELLDIR=$HOME/.shell
export SHELLDIR

#
# Try to determine if we are Bourne or Korn shell setting 'iskorn'.
#

[ -n "$PS3" ] && iskorn=1 || iskorn=0; export iskorn
#if [ -n "$PS3" ]
#then
#     iskorn=1
#else
#     iskorn=0
#fi
#export iskorn

#
# Set up envirmment vars we need.
# Path is environment dependent; place it in a separate file.
#

[ -r $SHELLDIR/setpath ] && . $SHELLDIR/setpath

CDPATH=":$HOME:/usr"
CERASE="^H"
CKILL="^U"
CINTR="^C"
TERMINFO=/u/qbarnes/terminfo
LESS=emq
EXINIT="set autoindent redraw nowrapscan"
e=$ENVFILE
p=$HOME/.profile

export LOGNAME MAIL CDPATH CERASE CKILL CINTR TERMINFO LESS EXINIT e p

if [ iskorn -eq 1 ]
then
    export  ENVFILE=$SHELLDIR/envfile \
            HISTSIZE=250 \
            EDITOR='whence vi' \
            OLDPWD=$HOME
    export  ENV="{ENVFILE[(_$-=1)+(_=0)-(_$-!=_{$-%%*i*})]}"

    # Set mail check if we're on our mail host.

    [ -f $HOME/.mailhost ] && export MAILCHECK=0

    # Figure out host's simple name set HOST for PS1.

    if [ -x "`whence hostname`" ]
    then
        HOST="`hostname`"
    elif [ -x "`whence uname`" ]
    then
        HOST="`uname -n`"
    fi

    HOST=${HOST%%.*}
    export HOST=${HOST:="unknown"}
fi
```

```
#
# Set up terminal stuff and set TERM variable.
#

stty erase $CERASE kill $CKILL intr $CINTR

[ -r $SHELLDIR/setterm ] && . $SHELLDIR/setterm

if [ iskorn -eq 1 ]
then
    # If we have tput available, use it to set columns and lines.

    if [ -n "`whence tput`" ]
    then
        export COLUMNS=`tput cols` \
            LINES=`tput lines`
    fi

    # Don't let a ^D kill us, track everything, and make sure the
    # job monitor is on.

    set -o ignoreeof -o trackall -o monitor
fi

# Give us rwxr-xr-x.

umask 022

#
# Set up system dependent profiling stuff.
#

[ -r $SHELLDIR/profile.dep ] && . $SHELLDIR/profile.dep
```