

tools

Nguyen Tun Anh

March 27, 2015

Shell

slide

zsh

slide

- Better interactive usage
- Better completion
- Useful config in oh-my-zsh

Some useful keys

slide

- `Ctrl-r` to search backward
- `Alt-.` to repeat last argument of last command
- Editing keys: `Ctrl-w`, `Alt-f`, `Alt-b`
- `Alt-l` to `ls`
- `gnome-terminal`: hide the menu, otherwise many M- combos are taken

Prompt

slide

Configure prompt to show more information

Customized shortcuts for better editing

slide

```
bindkey -e "\eu" backward-word
bindkey -e "\er" emacs-forward-word
```

```
bindkey -e "\ed" backward-kill-word
bindkey -e "\ef" kill-word
```

```
bindkey -e "\eh" beginning-of-line
bindkey -e "\eH" end-of-line

bindkey -e "\ei" up-history
bindkey -e "\ek" down-history

bindkey -e "\e\t" menu-complete
bindkey -e "\t" menu-complete
bindkey -e "\en" menu-complete

bindkey -e "\e " insert-last-word
```

Aliases

slide

- Better default

```
alias ls='ls -aCFGlh --color=auto'
```

- Search processes by name (check if something is actually running)

```
alias pp='ps -ef | grep'
pp twistd
```

- Search processes listening on port (check if the server is ok)

```
alias sk='sudo netstat -ntlp | grep'
sk 8092
```

Simple shell functions

slide

- Search packages

```
function pa {
    apt-cache search --names-only $1 | grep $1
}
```

```
pa bpython
```

- Simple server for a static site

```
function server {
    local port="${1:-1111}"
    gnome-open "http://localhost:${port}/"
    python -m SimpleHTTPServer $port
}

cd server-root
server # or server 8080
```

Useful programs

slide

Quick directory switching with autojump

slide

```
sudo apt-get install autojump
```

In `.profile` or `.zshrc`

```
if [ -f /usr/share/autojump/autojump.sh ]; then
    . /usr/share/autojump/autojump.sh
fi
```

Switching between frequently used directories

```
cd ~/Programming/cogini/click-vagrant/clickserver
cd ~/Programming/cogini/bidandbuy-vagrant/bidandbuy
j cli
j bid
```

Copying with rsync

slide

- Local/remote
- Incremental
- Compression
- Generally better than cp/scp

```
alias rs='rsync --progress -rv'
alias rsl='rsync --progress -rv --inplace' # Local
alias rsn='rsync --progress -rvz'        # Network
```

```
rsn bnb-prod2:~/bidandbuy/web/html/protected/dump.sql ./
rsn ~/Programming/cogini/phyaura/phyauraehr/ phyaura@dev3:~/phyaura.cogini.com/

# Set up a quick-n-dirty git repo
rsn ~/Programming/cogini/phyaura/phyauraehr/.git/ phyaura@dev3:~/phyaura.git/
git clone phyaura@dev3:~/phyaura.git
```

Port forwarding with ssh

slide

Access a local port on the remote machine

```
ssh -L [local_bind_address]:local_port:remote_host:remote_port
# Remote MySQL
ssh -L :3306:dev1.bidandbuy.com.tw:3306 bnb-dev1
```

Let a remote machine access a port on your machine

```
ssh -R [remote_bind_address]:remote_port:localhost:local_port
# Make a local server accessible on a remote machine
ssh -R :8080:localhost:9022 click-dev1
gnome-open http://dev1.clickmedix.com:8080
```

Add `-f -N` if you want it to run in the background

Search in file name or content

slide

- The better way, if you use Emacs
 - M-x find-grep
 - M-x find-name-dired
- Or use `grep`, `find`, `locate`

Persistent session with tmux

slide

- Disconnection-safe

```
tmux new-session -s ubolonton
```

```
tmux attach-session -t ubolonton
```

- Split screen into smaller "windows": edit code and see logs at the same time

- Multiple "tabs"
- Pair programming without sharing monitor
- Configure shortcuts and other options with `.tmux.conf`

xbindkeys

Assign system-wide shortcuts to shell command

- Custom volume control
- `xdotool`:
- Getting the mouse out of the way
- Rudimentary window management, app switching, fullscreen toggling
- `conkeror`:
- Control web music player without switching to the browser
- Switch to browser's mail tab from anywhere
- `emacsclient`:
- Anything you want

git

slide

- Easier-to-read log (or just use `magit`): put in `~/.gitconfig`
- Manage config files and other personal stuff

```
cd ~  
git clone https://github.com/ubolonton/dotfiles  
ln -s ~/dotfiles/.zshrc ~/.zshrc
```