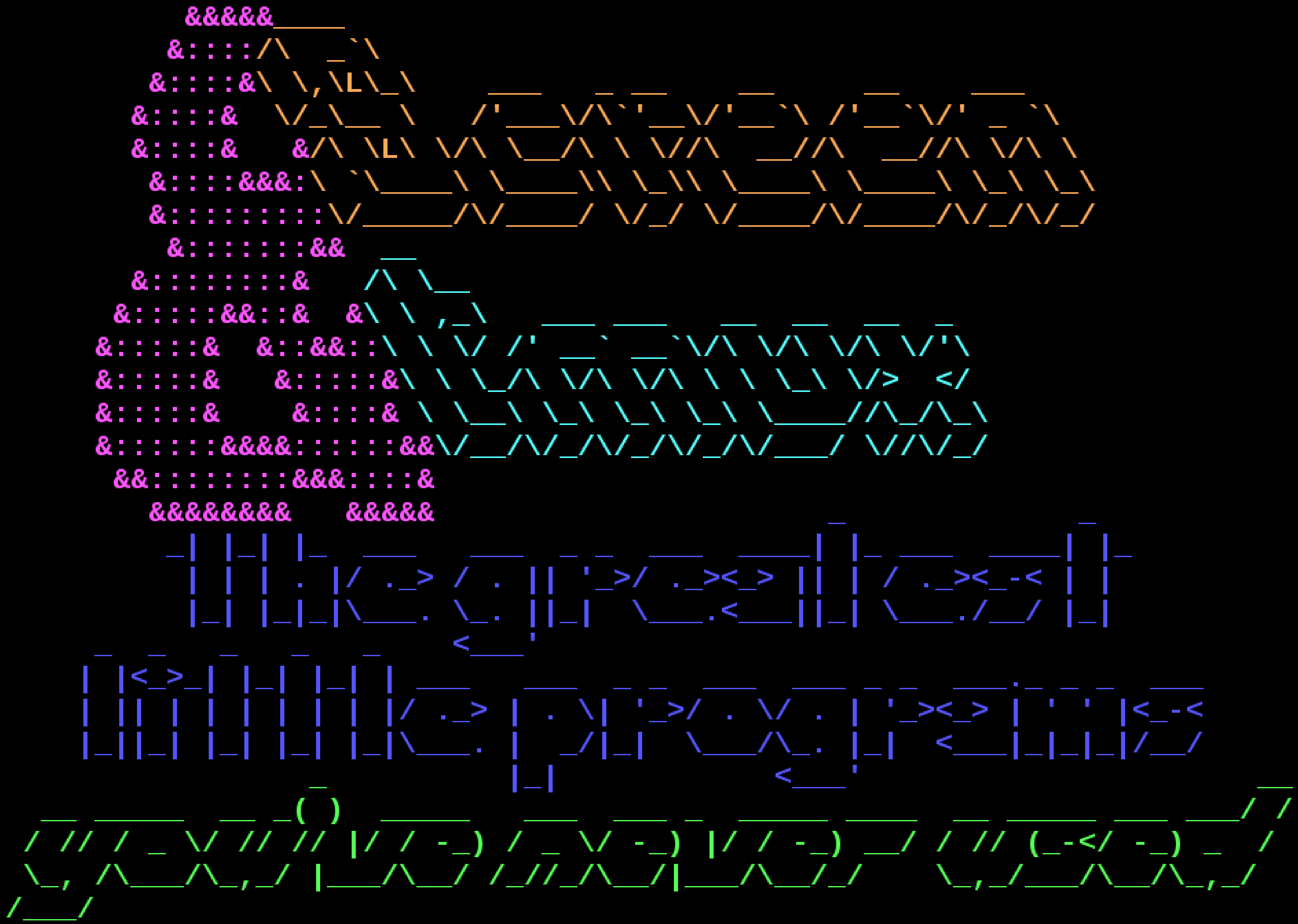
```
   _   _              _   
  /_\ | |__  ___ _  _| |_   _ __  ___ 
 / _ \| '_ \/ _ \ || |  _| | '  \/ -_)
/_/ \_\_.__/\___/\_,_|\__| |_|_|_\___|
```

* I am a Masters' student if Computer Science at USU

* I work as a software engineer at Spillman Technologies

* I have been a Screen user since 2006

```
 _____     _                   _____     _ _             
|  ___|_ _| | ___  _ __       |  ___|_ _| | | _____  _ __ 
| |_ / _` | |/ _ \| '__|  not | |_ / _` | | |/ / _ \| '__|
|  _| (_| | | (_) | |         |  _| (_| | |   < (_) | |   
|_|  \__,_|_|\___/|_|         |_|  \__,_|_|_|\_\___/|_|   
```


the greatest
little programs
you've never used

Either of these programs will make your computing time much more awesome

Try both tmux and Screen to find out which you prefer

The impression you will take away from this presentation is

"How did I ever get along without a terminal multiplexer in my life?"

```
   ____     __   __ _                             ___
  /___ \_  / / /\ \ \ |__  _   _   _ __ ___  _   ___  __/ _ \
 //  / (_) \ \/  \/ / '_ \| | | | | '_ ` _ \| | | \ \/ /\// /
/ \_/ / _   \  /\  /| | | | |_| | | | | | | | |_| |>  <   \/
\___,_\(_)   \/  \/ |_| |_|\__, | |_| |_| |_|\__,_/_/\_\  ()
                           |___/
```

A: To survive the worst thing that could befall a millenial

```
         _____                   __          __ _          __  __
        / ___/____     _      __/ /_  ____ _/ /( )_____   / /_/ /_  ___
        \__ \/ __ \   | | /| / / __ \/ __ `/ __/// ___/  / __/ __ \/ _ \
       ___/ / /_/ /   | |/ |/ / / / / /_/ / /_  (__  )  / /_/ / / /  __/
      /____/\____/    |__/|__/_/ /_/\__,_/\__/ /____/   \__/_/ /_/\___/

                     __    _          __           _____
                    / /_  (_)___ _   ____/ /__  ____ _/ /__ \
                   / __ \/ / __ `/  / __  / _ \/ __ `/ / / _/
                  / /_/ / / /_/ /  / /_/ /  __/ /_/ / / /_/
                 /_.___/_/\__, /   \__,_/\___/\__,_/_/ (_)
                         /____/
```

What is a Terminal Multiplexer?

```
                                                remoteserver
                                            +------------------+
                  It makes one xterm        | +--------------+ |
                     window act like        | |              | |
                          many xterm   /----->|     zsh      | |
                             windows  /     | |              | |
                                     /      | +--------------+ |
                     localhost      /       |                  |
                  +--------------+ /        | +--------------+ |
                  +--------------+/         | |              | |
                  |              |----------->|     ftp      | |
                  |    xterm     |\         | |              | |
                  |              | \        | +--------------+ |
                  +--------------+  \       |                  |
                                     \      | +--------------+ |
                                      \     | |              | |
                                       \----->|   nethack    | |
                                            | |              | |
                                            | +--------------+ |
                                            +------------------+
```

What is a Terminal Multiplexer?

```
                                  remoteserver
                              +------------------+
        It also keeps your    | +--------------+ |
    programs on the server    | |    ^    ^    | |
         running after the    | |              | |
      xterm window dies...    | |    \____/    | |
                              | +--------------+ |
       localhost              |                  |
    +--------------+          | +--------------+ |
    +--------------+          | |    ^    ^    | |
    |    X   X     |          | |              | |
    |    _____     |          | |    \____/    | |
    |   /  U  \    |          | +--------------+ |
    +--------------+          |                  |
                              | +--------------+ |
    ...or after the network   | |    ^    ^    | |
     connection goes out...   | |              | |
                              | |    \____/    | |
 ...or when the power goes!   | +--------------+ |
                              +------------------+
```

What is a Terminal Multiplexer?

```
    You can also share your       remoteserver
    work with another person  +------------------+
                              | +--------------+ |
       localhost              | |              | |
    +--------------+          | |     zsh      | |
    +--------------+          | |              | |
    |     My       |          | +--------------+ |
    |    Urxvt     |--------->|                  |
    |              |          | +--------------+ |
    +--------------+          | |              | |
                              | |     ftp      | |
                              | |              | |
    +--------------+          | +--------------+ |
    +--------------+          |                  |
    |     Your     |          | +--------------+ |
    |    PuTTY     |--------->| |              | |
    |              |          | |   nethack    | |
    +--------------+          | |              | |
                              | +--------------+ |
    They can see what you     +------------------+
    see, or they can take
     over from you if you
            need to leave
```

```
  ___
 / __| __ _ _ ___ ___ _ _
 \__ \/ _| '_/ -_) -_) ' \
 |___/\__|_| \___\___|_||_|
```

The first version of Screen was released in 1987...

...it may be older than some of you who may use it

KONAMI
CONTRA
ZELDA II
The Adventure of LINK
Another journey of ultimate challenge in the fantasy world of Hyrule.
Nintendo ENTERTAINMENT SYSTEM
ARCADE
SEGA
For 1 or 2 players
DOUBLE DRAGON
Mike Tyson
PUNCH-OUT!
• Fight Tyson in The Dream Bout.
• Challenge boxers from around the globe.
• Win and become W.V.B.A. Champ.
Nintendo ENTERTAINMENT SYSTEM

```
  ___                      
 / __| __ _ _ ___ ___ _ _  
 \__ \/ _| '_/ -_) -_) ' \ 
 |___/\__|_| \___\___|_||_|
```

The grandpappy of all of the muxers

Features:

* Nethack mode!
* Widely available on POSIX systems
* Serial console support
* Multi-user support

```
  ___
 / __| __ _ _ ___ ___ _ _
 \__ \/ _| '_/ -_) -_) ' \
 |___/\__|_| \___\___|_||_|
```

Screen is the inspiration for many X11 tiling WMs such as ratpoison, stumpwm, dwm, and xmonad

The latest version is 4.2.1

Protip search for 'GNU Screen', or you're gonna have a bad time

```
 _
| |_ _ __ ___  _   ___  __
| __| '_ ` _ \| | | \ \/ /
| |_| | | | | | |_| |>  <
 \__|_| |_| |_|\__,_/_/\_\
```

tmux was first released in 2009 under a BSD license

It aims to be a modern alternative to Screen

(read: it's <u>not</u> a drop-in replacement)

```
 _
| |_ _ __ _  _ __ __
|  _| '  \ || \ \ /
 \__|_|_|_\_,_/_\_\
```

The brash, young upstart with lots of mindshare

Features:

* Actively maintained
* Simple configuration
* More flexible process organization
* Easily scriptable from the command-line

```
 _
| |_ _ __ _  ___ __
|  _| '  \ || \ \ /
 \__|_|_|_\_,_/_\_\
```

If you're new to all of this, tmux is the best place to start

The latest version is 1.9a

```
   __                           _          __
  / /  ___    ___ _ __ __ _ ___| |__    _ / /
 / /  / _ \  / __| '__/ _ `/ __| '_ \  (_) |
/ /__|  __/ | (__| | | (_| \__ \ | | | _| |
\____/\___|  \___|_|  \__,_|___/_| |_| (_) |
                                          \_\
```

Now what?

```
 ___
/ __| __ _ _ ___ ___ _ _
\__ \/ _| '_/ -_) -_) ' \
|___/\__|_| \___\___|_||_|
```

List all Screen sessions:

```
$ screen -ls
```

Reconnect to a detached session:

```
$ screen -r [session name]
```

```
  _
 | |_ _ __ ___  _   ___  __
 | __| '_ ` _ \| | | \ \/ /
 | |_| | | | | | |_| |>  <
  \__|_| |_| |_|\__,_/_/\_\
```

List all tmux sessions:

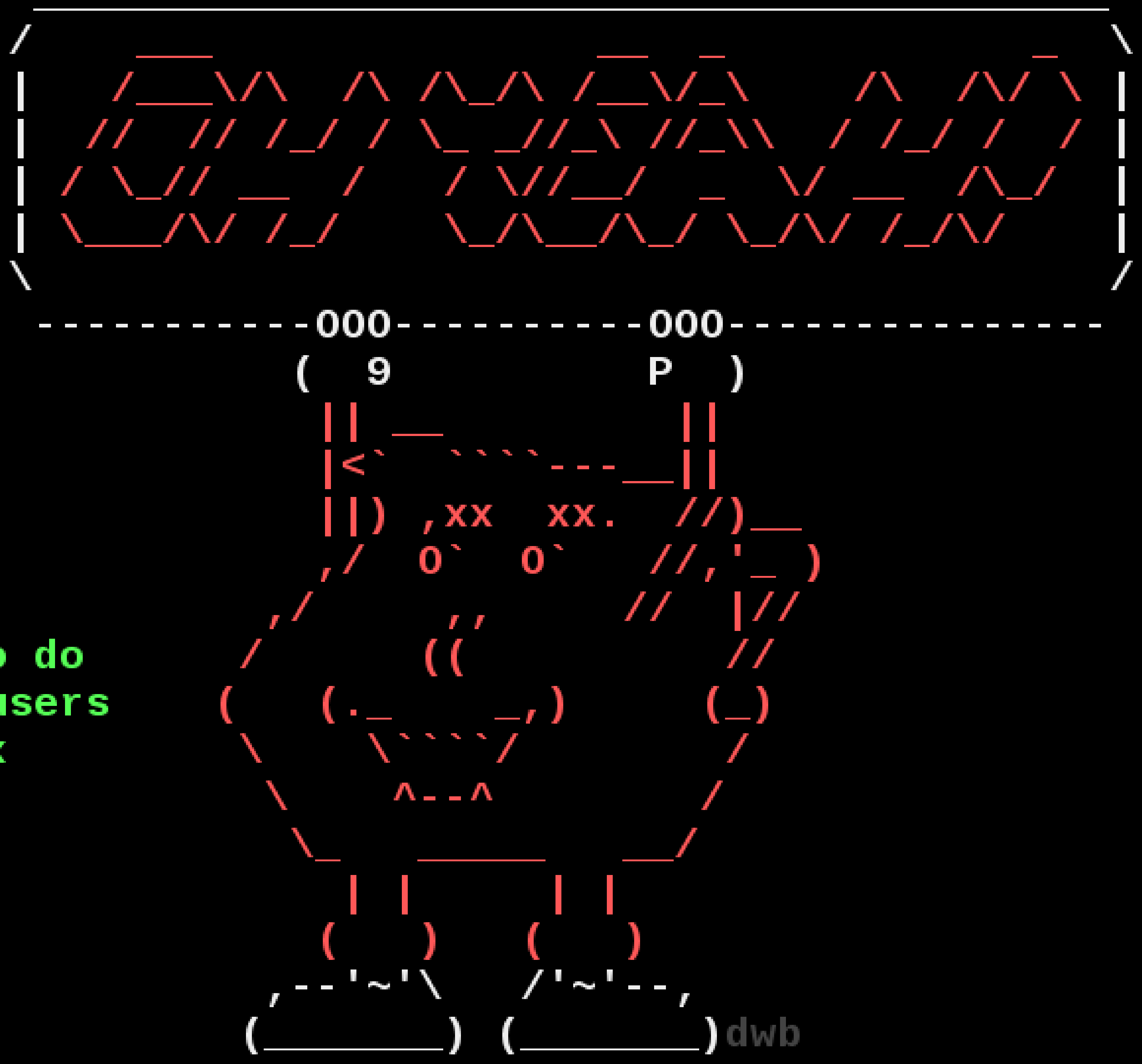
```
    $ tmux list-sessions
```

Reconnect to a detached session:

```
    $ tmux attach [-t session name]
```

Congratulations!!

You now know enough to do
what draws 99% of mux users
to Screen and tmux



```
   ___             __ _                       _   _
  / __\___  _ __  / _(_) __ _ _   _ _ __ __ _| |_(_) ___  _ __
 / /  / _ \| '_ \| |_| |/ _` | | | | '__/ _` | __| |/ _ \| '_ \
/ /__| (_) | | | |  _| | (_| | |_| | | | (_| | |_| | (_) | | | |
\____/\___/|_| |_|_| |_|\__, |\__,_|_|  \__,_|\__|_|\___/|_| |_|
                        |___/
```

WEAPONS
LIFE
ENERGY
M.BUSTER
BRIGHT
TOAD
DRILL
PHARAOH
RING
DUST
DIVE
SKULL
R.COIL
R.MARINE
R.JET
WIRE
BALLOON
E 08
09

```
 ___ _        _     _   _    _                __ _        _   
| __(_)_ _ __| |_  | |_| |_ (_)_ _  __ _ ___  / _(_)_ _ __| |_ 
| _|| | '_(_-<  _| |  _| ' \| | ' \/ _` (_-< |  _| | '_(_-<  _|
|_| |_|_| /__/\__|  \__|_||_|_|_||_\__, /__/ |_| |_|_| /__/\__|
                                   |___/                        
```

The most important configuration item is the
escape key or prefix key

The defaults are:

```
 _______  ____      _______  ____
||ctrl ||||a ||    ||ctrl ||||b ||
||_____||||__||    ||_____||||__||
|/_____\||/__\|    |/_____\||/__\|
```

in Screen   in tmux

But the problem is...

* ctrl + a moves the shell cursor to the beginning of the line, and increments numbers in Vim

* ctrl + b moves the shell cursor back one word, and scrolls Vim's buffer back a page

```
                 _             _         _ _
       ___ ___  | |__  __ _ __(_)__ __ _| | |_  _
 _ _ _(_-</ _ \ | '_ \/ _` (_-< / _/ _` | | | || |
(_|_|_)__/\___/ |_.__/\__,_/__/_\__\__,_|_|_|\_, |
                                             |__/

                _______  _________
               ||ctrl ||||space  ||
               ||_____||||_______||
               |/_____\||/_______\|
```

(denoted as ^@ in configuration files)

```
  ___
 / __| __ _ _ ___ ___ _ _
 \__ \/ _| '_/ -_) -_) ' \
 |___/\__|_| \___\___|_||_|
```

It's configuration file is named ~/.screenrc

```
# replace ctrl-a with ctrl-space
escape ^@@

# don't use visual bell; print the ASCII bell char 0x07
vbell off
vbell_msg Ding!

# play a bell even when the beeping window isn't visible
bell_msg 'Ding! %n^G'
msgwait 2

# better error messages
nethack on

# suppress the startup message
startup_message off

# put a caption on bottom line
caption always "%{= WK}%-w%{=b kW}%n %t%{= WK}%+w %-=%{= Wk}%H%{= WK} %l %c"
```

```
 _
| |_ _ __  _  ___ __
|  _| '  \| || \ \ /
 \__|_|_|_|\_,_/_\_\
```

It's configuration file is named ~/.tmux.conf

```
# rebind escape key to ctrl-space
unbind C-b
set -g prefix C-Space

# use Vi keybindings in copy-mode
set -g mode-keys vi

# statusline
set -g window-status-current-attr reverse
set -g status-right \
    "#[fg=brightyellow,bg=black,bold]#h#[default] \
    #(cut -d' ' -f 1-3 /proc/loadavg) #[fg=black,bright]%H:%M"
```

The cool things...

The statusbar clock helps you to not accidentally work too late

Keeps your connection alive by defeating the SSH idle timeout

Which wouldn't even have bothered you anyway

BECAUSE YOUR PROGRAM IS RUNNING INSIDE A MUXER!!!

Like Vim, Screen and tmux have a
command-line mode which is introduced with a colon character

```
 _
(_)
 _
(_)
```

However, in our case, this mode is introduced by

```
 _______  _________     ____
||ctrl ||||space   ||   ||: ||
||_____||||________|| + ||__||
|/_____\||/________\|   |/__\|
```

All of the commands which may appear in your configuration file may be specified at runtime

This allows you to try out new configuration settings interactively...

...as well as use commands for which you don't have/want a binding

```
dP     dP   88888888b dP         888888ba  dP
88     88   88        88         88    `8b 88
88aaaaa88a a88aaaa    88        a88aaaa8P' 88
88     88   88        88         88        dP
88     88   88        88         88
dP     dP   88888888P 88888888P  dP        oo
```

Oh noes! I can't remember which key does what!

Never fear, because

```
 _______  _________      ____
||ctrl ||||space   ||   ||? ||
||_____||||________|| + ||__||
|/_____\||/________\|   |/__\|
```

puts a handy reference into your active window pane

```
__      ___         _               __  __
\ \    / (_)_ _  __| |_____ __ __  |  \/  |__ _ _ _  __ _ __ _ ___ _ _ ___
 \ \/\/ /| | ' \/ _` / _ \ V  V /  | |\/| / _` | ' \/ _` / _` / -_) '_(_-<
  \_/\_/ |_|_||_\__,_\___/\_/\_/   |_|  |_\__,_|_||_\__,_\__, \___|_| /__/
                                                         |___/
```

A new window running your preferred shell may be created with

```
 _______  _________     ____
||ctrl ||||space  ||   ||c ||
||_____||||_______|| + ||__||
|/_____\||/_______\|   |/__\|
```

This window is destroyed when the contained program exits

```
You have a few tools to help you keep track of all these windows


                           _______  _________     ____
                          ||ctrl ||||space   ||   ||" ||
Menu of windows as a list ||_____||||________|| + ||__||
                          |/_____\||/________\|   |/__\|


 _______  _________     ____
||ctrl ||||space   ||   ||w ||
||_____||||________|| + ||__|| idem.
|/_____\||/________\|   |/__\|
```

You have a few tools to help you keep track of all these windows

```
                     _______  _________    ____
                    ||ctrl |||||space  ||   ||' ||
Select window by title ||_____|||||________|| + ||__||
                    |/_____\||/________\|   |/__\|
```

Select a window by title or content

```
 _______  _________    ____
||ctrl |||||space  ||   ||f ||
||_____|||||________|| + ||__|| fnmatch() pattern
|/_____\||/________\|   |/__\|
```

As you open and close windows you will destroy the nice monotonic ordering you began with

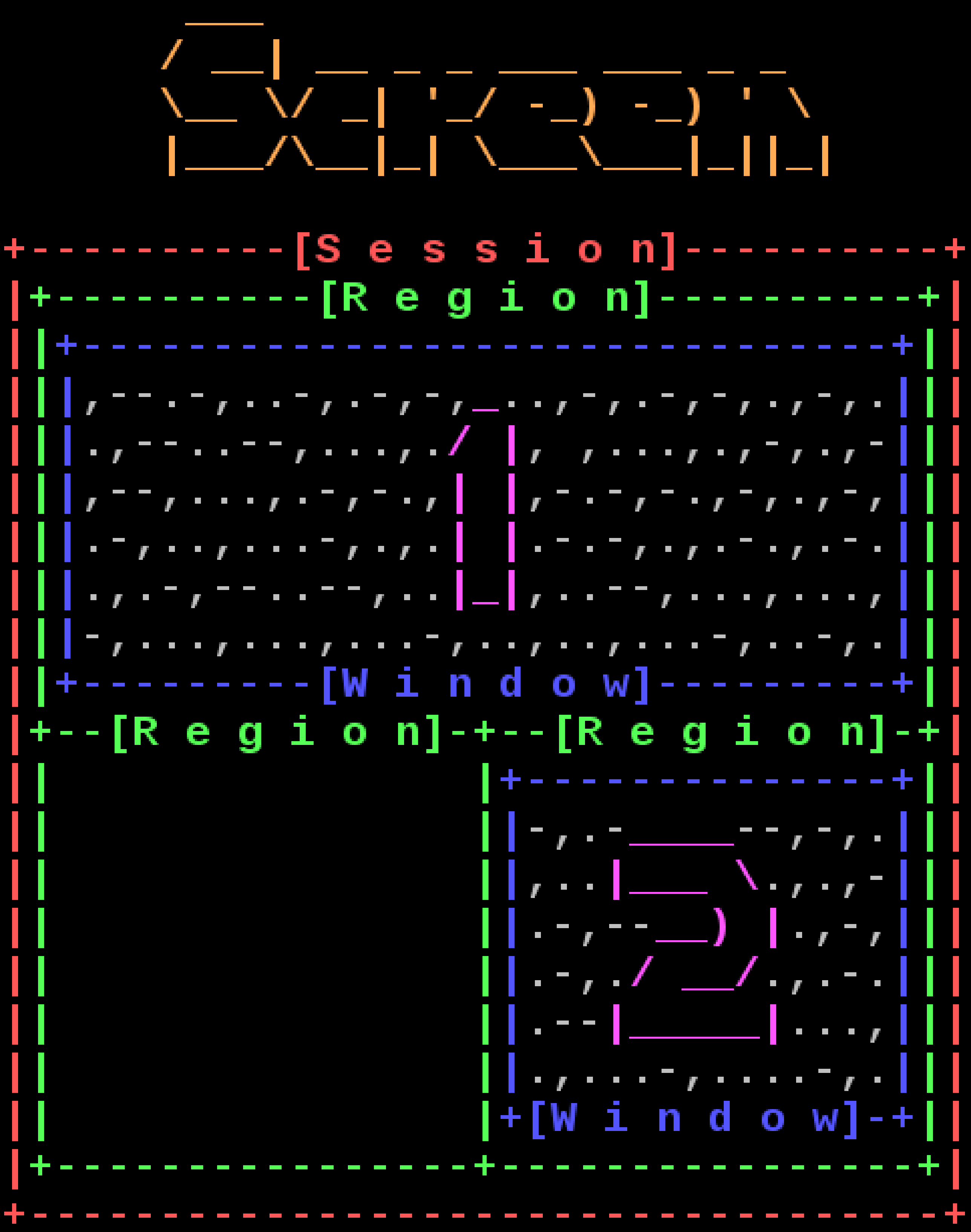
If you have become accustomed to a particular ordering, (or if you are particularly OCD), you may wish to re-assign the windows' numbers

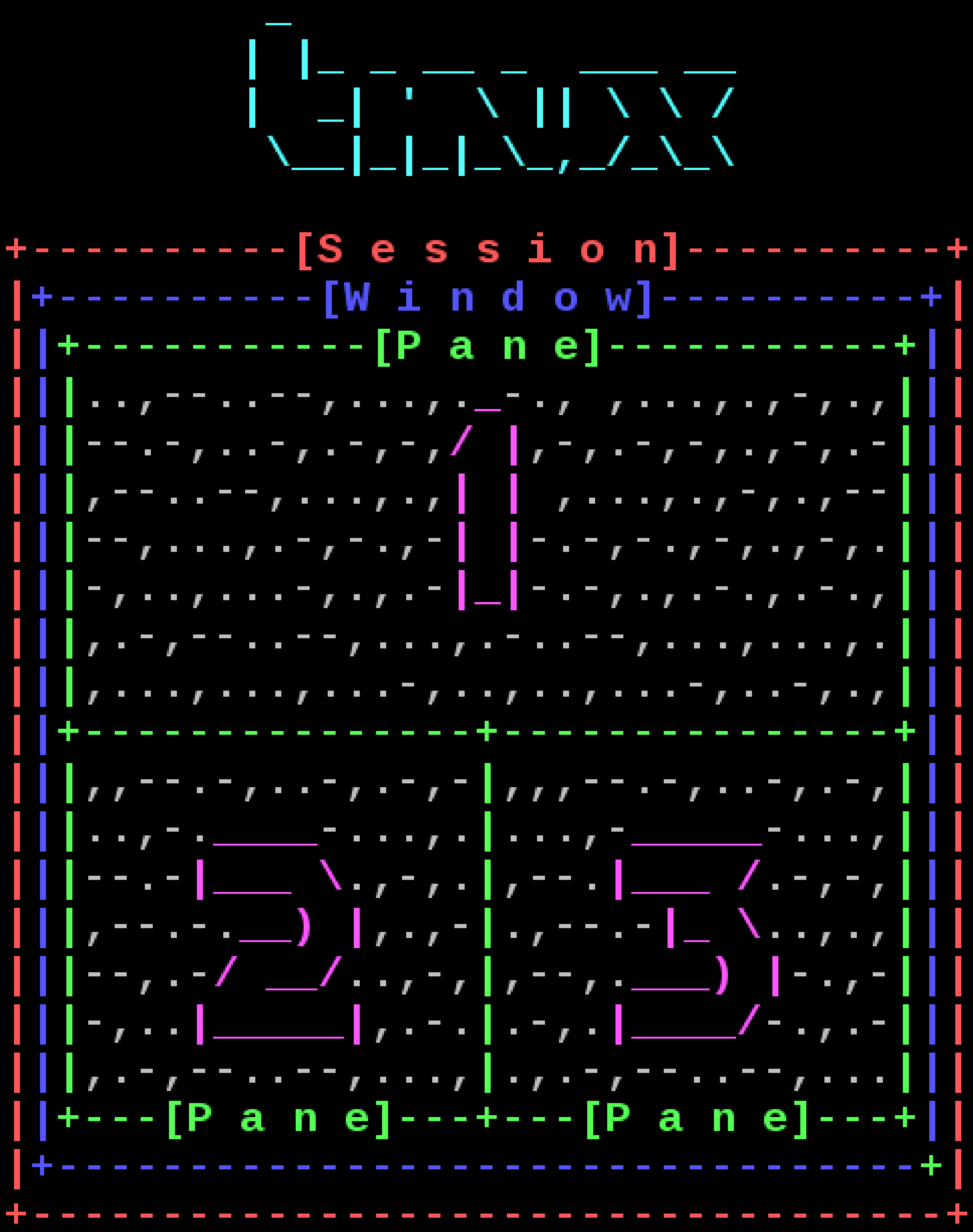
```
 _______   _________     ____
||ctrl ||||space  ||   ||: ||
||_____||||_______|| + ||__|| number N
|/_____\||/_______\|   |/__\|

 _______   _________     ____
||ctrl ||||space  ||   ||: ||
||_____||||_______|| + ||__|| move-window -t N
|/_____\||/_______\|   |/__\|
```

For much of the time you may be content to switch between full screen applications

However, it is nice to see two things at one time

Both Screen and tmux can divide the screen amongst
your running applications, but they each take their own approach


Screen
[S e s s i o n]
[R e g i o n]
1
[W i n d o w]
[R e g i o n]
[R e g i o n]
2
[W i n d o w]


Tmux
[S e s s i o n]
[W i n d o w]
[P a n e]
1
2
3
[P a n e]
[P a n e]

```
  ___
 / __| __ _ _ ___ ___ _ _
 \__ \/ _| '_/ -_) -_) ' \
 |___/\__|_| \___\___|_||_|
```

You begin with one window in fullscreen mode

From there you may split each window into two regions

```
 _______  _________   ____
||ctrl ||||space  ||   ||S ||
||_____||||________|| + ||__|| Split region horizontally
|/_____\||/________\|   |/__\|

                          _______  _________   ____
                         ||ctrl ||||space  ||   ||| ||
Split region vertically  ||_____||||________|| + ||__||
                         |/_____\||/________\|   |/__\|
 ___
/ __| __ _ _ ___ ___ _ _
\__ \/ _| '_/ -_) -_) ' \
|___/\__|_| \___\___|_||_|
```

The following configuration items allow you to use
vi-inspired keybindings to navigate between split windows

Add these lines to your ~/.screenrc:

```
# use vi-like motion keys to move between split regions
bind j focus down
bind k focus up
bind h focus left
bind l focus right

# make the active region the only region
bind o only

# close the active region (the program remains running)
bind C remove
```

```
 _
| |_ _ __ _  ___ __
|  _| '  \ || \ \ /
 \__|_|_|_\_,_/_\_\
```

The arrangement of panes is governed by the active layout

You may cycle through possibile layouts with

```
 ________  __________    __________
||ctrl ||||space   ||   ||space   ||
||_____||||________|| + ||________||
|/_____\||/________\|   |/________\|
```

```
                                           ________  __________    ____
                                          ||ctrl ||||space   ||   ||" ||
Split the current pane into a new shell   ||_____||||________|| + ||__||
                                          |/_____\||/________\|   |/__\|
```

```
 ________  __________    ____
||ctrl ||||space   ||   ||! ||
||_____||||________|| + ||__|| Break the current pane into its own window
|/_____\||/________\|   |/__\|
 _
| |_ _ __ _  ___ __
|  _| '  \ || \ \ /
 \__|_|_|_|\_,_/_\_\
```

The following makes tmux behave more like vi:

```
# Vi directions to navigate between panes
unbind-key j
bind-key j select-pane -D

unbind-key k
bind-key k select-pane -U

unbind-key h
bind-key h select-pane -L

unbind-key l
bind-key l select-pane -R
```

tmux remembers how your windows were split into panes between disconnecting and reconnecting

As of the latest version, Screen can do so as well, through its new layout commands

```
   ___                                    _
  / __\___  _ __  _   _ _ __   __ _ ___| |_ __ _
 / /  / _ \| '_ \| | | | '_ \ / _` / __| __/ _` |
/ /__| (_) | |_) | |_| | |_) | (_| \__ \ || (_| |
\____/\___/| .__/ \__, | .__/ \__,_|___/\__\__,_|
           |_|    |___/|_|
```

Have you used one of those new-fangled terminal emulators

which features a <u>searchable scrollback</u> buffer?

That's a nice feature which Screen has already been providing

for a really, really long time

under the guise of Copy & Paste mode

```
 _______  _________     ____
||ctrl ||||space   ||   ||[ ||
||_____||||_______|| + ||__||
|/_____\||/_______\|   |/__\|
```

Search for scrolled-off text with vi-like keybindings

```
/pattern

?pattern
```

Delimit a region of text by pressing Enter at both ends

The text which you selected is placed into a buffer

(which are analaogus to one of Vim's registers, or a clipboard)

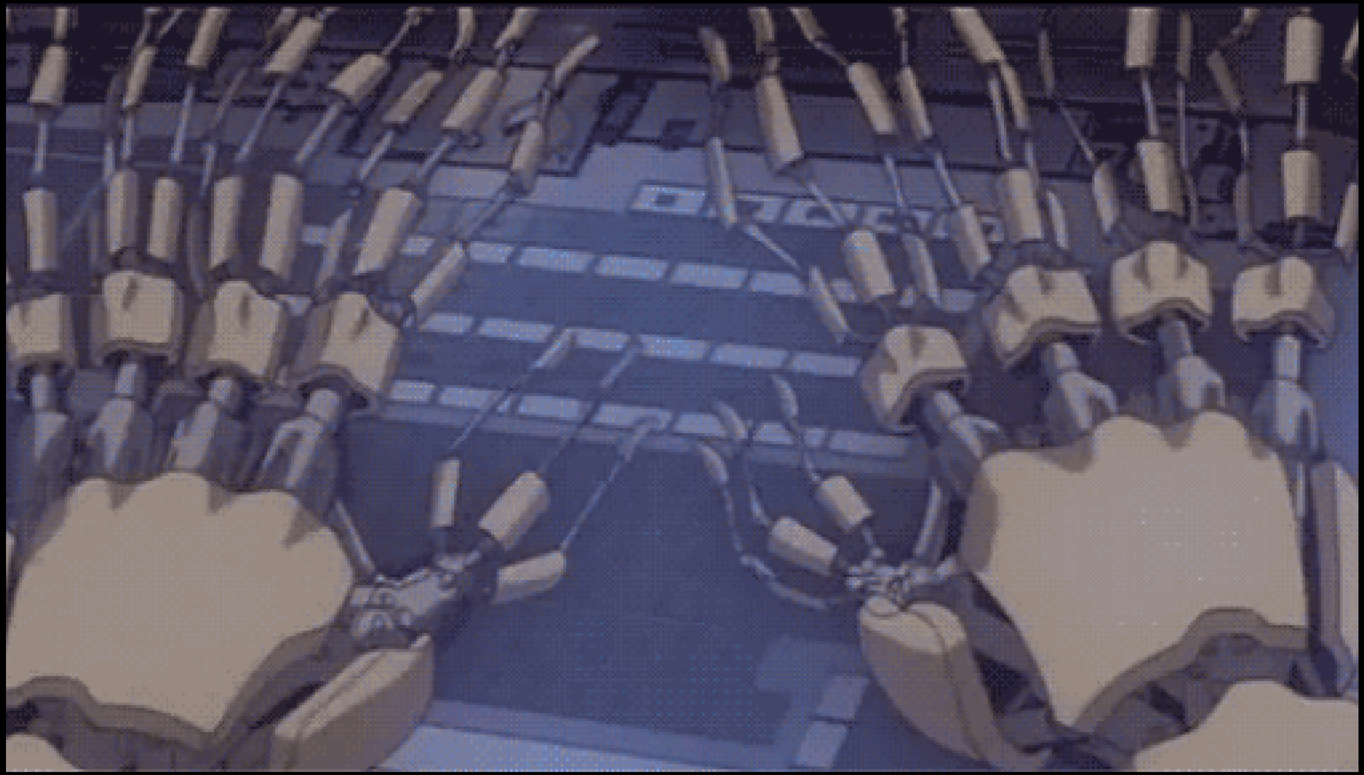
Paste the buffer with

```
 _______  _________     ____
||ctrl ||||space  ||   ||] ||
||_____||||_______|| + ||__||
|/_____\||/_______\|   |/__\|
```

Pasted text is sent into the program through the keyboard

and so appears to a program as though you just typed it in...

...<u>really</u> fast



Copy & paste works identically in tmux for the most part

except tmux keeps each copied selection in a stack of buffers

You can view and paste each historical selection with

```
 _______  _________    ____
||ctrl ||||space  ||   ||= ||
||_____||||_______|| + ||__||
|/_____\||/_______\|   |/__\|
 __    __ _                           _       _
/ / /\ \ \ |__   ___   __      ____ _| |_ ___| |__   ___  ___
\ \/  \/ / '_ \ / _ \  \ \ /\ / / _` | __/ __| '_ \ / _ \/ __|
 \  /\  /| | | | (_) |  \ V  V / (_| | || (__| | | |  __/\__ \_ _ _
  \/  \/ |_| |_|\___/    \_/\_/ \__,_|\__\___|_| |_|\___||___(_|_|_)
```

Quis custodiet ipsos custodes?
-- Juvenal

Suppose you're compiling a ginormous program

cough Firefox *cough*

But you don't want to sit and watch it grind away for hours...

```
                                         -----
       -----------       -----------    #-....#####                   --------
       |.........|       |.........|    #|...|    #                   |%[.+(?|
       |.........|       |.........|    #--|--    ### ----------    ##+...%.%|
       |.........|       |..<......-#   ####        # |........|    # |..?.%?|
       |....>.....#     #-.........|#      ###      ##.........|    # --------
       --|------ -#     #|......^..|#      # #        |........|    #
         #        #     #-----------#      # ###      -.----|---    #
         ###      #     #####       #      #   #       #%   #       #
           #      #        #        #      #   ###      #   #       #
           ###    #        #        #      #     #      ##0 #       #
          ####    #        #        #------#     ###      # #       #
  --------|--#########     #        #|.....#       #      # ###     #
  |.........|     #  -.----#        #|?...|        ##############   #
  |..........##### ##|....-#        #-.....#########    --.-----.---#
  |.........|     # #|....|          ------             |....@.....|#
  -----------      ##|>....######                       |.....d....|#
                    #.....|                             |....%.....-#
                    #------                             ------------

[Fadein the Skirmisher]     St:18/02 Dx:16 Co:13 In:8 Wi:9 Ch:9  Lawful
Dlvl:3  $:215 HP:31(31) Pw:11(11) AC:5  Exp:4 T:1668
```

Get the slides at http://unnovative.net   (52/86)

```
 __  __          _ _             ___ _ _
|  \/  |___ _ _ (_) |_ ___ _ _  / __(_) |___ _ _  __ ___
| |\/| / _ \ ' \| |  _/ _ \ '_| \__ \ | / -_) ' \/ _/ -_)
|_|  |_\___/_||_|_|\__\___/_|   |___/_|_\___|_||_\__\___|
```

```
 _______  _________       ____
||ctrl ||||space  ||     ||_ ||
||_____||||_______||  +  ||__||
|/_____\||/_______\|     |/__\|

(the default is 30 seconds;
can be changed by :set silencewait N)
```

```
 _______  _________       ____
||ctrl ||||space  ||     ||: ||
||_____||||_______||  +  ||__|| set monitor-silence N
|/_____\||/_______\|     |/__\|
 __  __          _ _               _      _   _      _ _
|  \/  |___ _ _ (_) |_ ___ _ _    /_\  __| |_(_)_ _(_) |_ _  _
| |\/| / _ \ ' \| |  _/ _ \ '_|  / _ \/ _|  _| \ V / |  _| || |
|_|  |_\___/_||_|_|\__\___/_|   /_/ \_\__|\__|_|\_/|_|\__|\_, |
                                                          |__/
```

```
  _______   _________     ____
||ctrl  ||||space    ||   ||M ||
||______||||_________|| + ||__||
|/______\||/_________\|   |/__\|
```

```
  _______   _________     ____
||ctrl  ||||space    ||   ||: ||
||______||||_________|| + ||__|| set monitor-activity on
|/______\||/_________\|   |/__\|
 ___              _  __ _        _       _   _       _ _
/ __|_ __  ___ __(_)/ _(_)__    /_\  __| |_(_)_ _(_) |_ _  _
\__ \ '_ \/ -_) _| |  _| / _|  / _ \/ _|  _| \ V / |  _| || |
|___/ .__/\___\__|_|_| |_\__| /_/ \_\__|\__|_|\_/|_|\__|\_, |
    |_|                                                 |__/
```

```
 _______  _________      ____
||ctrl ||||space  ||    ||: ||
||_____||||_______|| +  ||__|| set monitor-content pattern
|/_____\||/_______\|    |/__\|
```

The pattern is matched against content on the terminal by the fnmatch(3) function...

...which matches the the same as the familiar shell wildcard

```
                 _       _                       _ _      _          _
  /\/\   __ _| | _(_)_ __   __ _  (_) |_  | | __ _ ___| |_
 /    \ / _` | |/ / | '_ \ / _` | | | __| | |/ _` / __| __|
/ /\/\ \ (_| |   <| | | | | (_| | | | |_  | | (_| \__ \ |_
\/    \/\__,_|_|\_\_|_| |_|\__, | |_|\__| |_|\__,_|___/\__|
                           |___/
```

Why don't you take a picture, it'll last longer

How would you prove that you have ascended NetHack?

scrot?

Alt + PrintScr?

Your phone?

```
  ___
 / __| __ _ _ ___ ___ _ _
 \__ \/ _| '_/ -_) -_) ' \
 |___/\__|_| \___\___|_||_|
```

Grab a screencap of window N into hardcopy.n with

```
 _______  _________    ____
||ctrl ||||space   ||  ||H ||
||_____||||________|| +||__|| (not the default setting)
|/_____\||/________\|  |/__\|
```

By default, screen captures are written into Screen's cwd

You may change that default with

```
 _______  _________    ____
||ctrl ||||space   ||  ||: ||
||_____||||________|| +||__|| set hardcopydir dir
|/_____\||/________\|  |/__\|
```

(Obviously, you'd keep this setting in your ~/.screenrc)

```
  _
 | |_ _ __ _  ___ __
 |  _| '  \ || \ \ /
  \__|_|_|_\_,_/_\_\
```

This is, unfortunately, not quite as simple to achieve in tmux,
but the mechanism is much more flexible

Put this into your ~/.tmux.config:

```
bind H capture-pane \; save-buffer -b 0 ~/.tmux-hardcopy \; delete-buffer -b 0
```

Now you can save an image of the active pane into ~/.tmux-hardcopy with

```
  _______  _________      ____
||ctrl  ||||space   ||   ||H ||
||______||||________|| + ||__||
|/______\||/________\|   |/__\|
  ___
 / __| __ _ _ ___ ___ _ _
 \__ \/ _| '_/ -_) -_) ' \
 |___/\__|_| \___\___|_||_|
```

Log files may be written into the <u>window's</u> cwd with a name like screenlog.<u>n</u>

This may be toggled with

```
 _______    _________      ____
||ctrl ||||space   ||    ||L ||
||_____||||________|| +  ||__|| (not the default setting)
|/_____\||/________\|    |/__\|
 _
| |_ _ __ _  ___ __
|  _| '  \ || \ \ /
 \__|_|_|_\_,_/_\_\
```

Again, this isn't out-of-the-box as in Screen,
but is much more flexible

```
 _______  _________     ____
||ctrl ||||space  ||   ||: ||
||_____||||_______|| + ||__|| pipe-pane -t n 'shell command'
|/_____\||/_______\|   |/__\|
```

shell command might be as simple as `cat > tmuxlog.0`

Or it may be as awesome as `base64 | rot13 | gzip -c > srsly3ncrypted.0.gz`

```
   ____           _ __  __
  /  _/__    ___ (_) /_/ /  ___ ____  _______ ____ ___
 _/ // _ \  / -_) / __/ _ \/ -_) __/ / __/ _ `(_-</ -_) _ _
/___/_//_/  \__/_/\__/_//_/\__/_/    \__/\_,_/___/\__(_|_|_)
```

Logging a window copies any output generated <u>after</u> logging begins...

...it does not include what is already on the window :(

```
 __ _                _                 _____ _
/ _\ |__   __ _ _ __(_)_ __   __ _    /__   (_)_ __ ___   ___
\ \| '_ \ / _` | '__| | '_ \ / _` |     / /\/ | '_ ` _ \ / _ \
_\ \ | | | (_| | |  | | | | | (_| |    / /  | | | | | | |  __/
\__/_| |_|\__,_|_|  |_|_| |_|\__, |    \/   |_|_| |_| |_|\___|
                             |___/
```

but i thought
sharing was caring

Both Screen and tmux facilitate multiple clients connecting to the same session

```
  ___
 / __| __ _ _ ___ ___ _ _
 \__ \/ _| '_/ -_) -_) ' \
 |___/\__|_| \___\___|_||_|
```

Connect to a Screen session in multi-user mode:

```
$ screen -x [-r session name]
```

```
  ___
 / __| __ _ _ ___ ___ _ _
 \__ \/ _| '_/ -_) -_) ' \
 |___/\__|_| \___\___|_||_|
```

Screen has a rich set of commands relating
to controlling access to various aspects of a session

* Multiple user accounts may connect to the same session
* Users can view windows independently
* Users can be denined view access from certain windows
* Users may be restricted from providing input
* Users may not be able to enter Screen commands

```
 _
| |_ _ __ _  ___ __
|  _| '  \ || \ \ /
 \__|_|_|_\_,_/_\_\
```

Connect to a session:

```
    $ tmux attach -t session name
```

You may cycle through active sessions with

```
 _______  _________     ____       ____
||ctrl ||||space  ||   ||( ||     ||( ||
||_____||||_______|| + ||__|| and ||__||
|/_____\||/_______\|   |/__\|     |/__\|
 _
| |_ _ __ _  ___ __
|  _| '  \ || \ \ /
 \__|_|_|_\_,_/_\_\
```

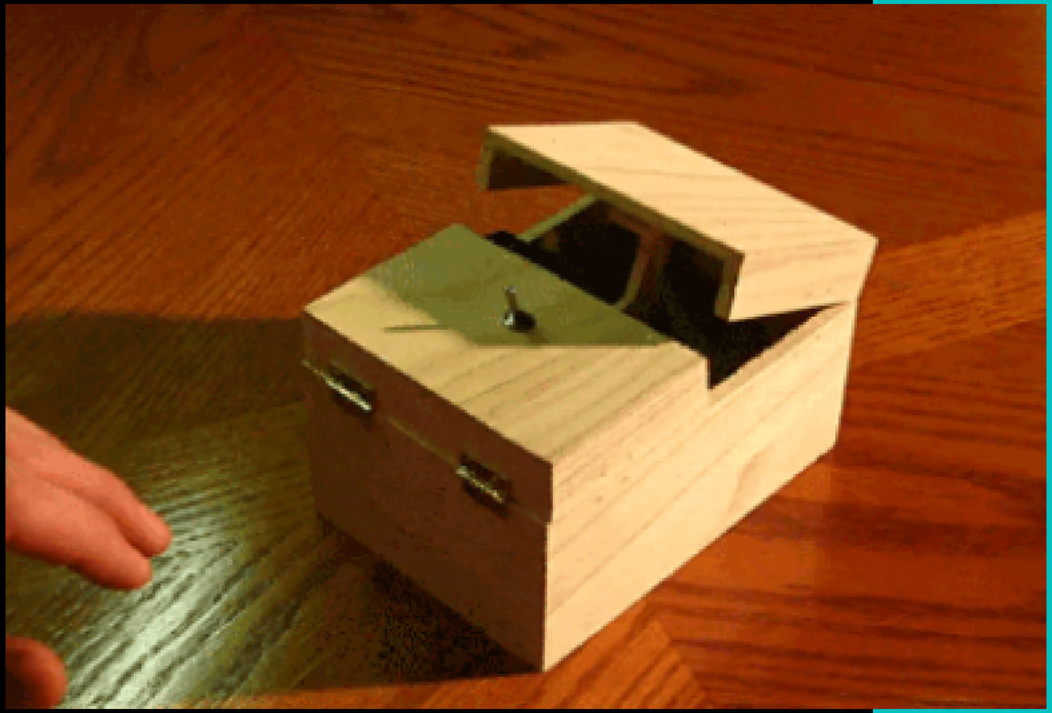
Multi-user is one area where tmux still lags a bit

* Multiple clients may connect to one session from
  the same user account
  (there is a project called "wemux" that seeks to addres this)

* Each client are constrained to view the same window

* The visible area is governed by the smallest client

```
    _    _                     _                                _      
   / \  | |_ __ ___   ___  ___| |_   _ __ ___   __ _  __ _(_) ___ 
  / _ \ | | '_ ` _ \ / _ \/ __| __| | '_ ` _ \ / _` |/ _` | |/ __|
 / ___ \| | | | | | | (_) \__ \ |_  | | | | | | (_| | (_| | | (__ 
/_/   \_\_|_| |_| |_|\___/|___/\__| |_| |_| |_|\__,_|\__, |_|\___|
                                                     |___/        
```



Your muxer can serve as a keyboard macro for the console

Just bind a string to a convenient sequence

Open a root shell:

```
:bind s screen sudo -i
```

Launch a new instance of Vim:

```
:bind v screen vim
```

Too lazy to remember your password?

```
:bind ^P stuff 123456^M
```

(Don't actually do that last one, okay?)

It's really nice to be able to use on-screen text

as input to a program

(xclip(1) users, amirite?)

```
   ______            __
  / ____/___  ____  / /
 / /   / __ \/ __ \/ /
/ /___/ /_/ / /_/ / /
\____/\____/\____/_/

    __
   / /_  _________ _      __________  _____
  / __ \/ ___/ __ \ | /| / / ___/ _ \/ ___/
 / /_/ / /  / /_/ / |/ |/ (__  )  __/ /
/_.___/_/   \____/|__/|__/____/\___/_/

   __       _      __
  / /______(_)____/ /_______
 / __/ ___/ / ___/ //_/ ___(_)
/ /_/ /  / / /__/ ,< (__  )
\__/_/  /_/\___/_/|_/____(_)
```

0. Copy a bit of text straight off the screen

1. Bind a key to open a browser and go right to the URL in the paste buffer

2. Another binding uses the paste buffer as a search query

Place this simple shell wrapper named lynx+ under your $PATH:

```
    #!/bin/bash
    case $1 in
        go)
            exec lynx "$(head -1 ~/.mux-exchange)"
            ;;
        search)
            exec lynx https://duckduckgo.com?q="$(cat ~/.mux-exchange)"
            ;;
    esac
```

```
  ___
 / __| __ _ _ ___ ___ _ _
 \__ \/ _| '_/ -_) -_) ' \
 |___/\__|_| \___\___|_||_|
```

Given this stanza in ~/.screenrc:

```
    # store copy buffer in ~/.mux-exchange
    bufferfile $HOME/.mux-exchange

    # launch lynx with URL in the paste register
    bind B eval "writebuf" "screen lynx+ go"

    # launch lynx with the search term in the paste register
    bind ^B eval "writebuf" "screen lynx+ search"
```

You can now look up a keyword or URL in a keystroke!

```
 _
| |_ _ __ _  ___ __
|  _| '  \ || \ \ /
 \__|_|_|_\_,_/_\_\
```

The same procedure is configured a bit differently here

Add this to your ~/.tmux.conf:

```
# Copy the selection to an exchange file
bind-key -t vi-copy Enter copy-pipe "cat > ~/.mux-exchange"

# launch $browser with url stored in paste register
bind B new-window "lynx+ go"

# launch $browser with search term stored in paste register
bind C-B new-window "lynx+ search"
```

Your muxer can also fill the role of ClusterSSH by multiplexing your input across multiple windows

`:at <windowName> <command>`

Run <command> in each window with a particular name

These windows do not need to be active or displayed

```
 ____ ____ ____ ____ ____ _  _ ____ ____ ____ ___ _ ____ _  _
 [__  |    |__/ |___ |___ |\ | |    |___ |__]  |  | |  | |\ |
 ___] |___ |  \ |___ |___ | \| |___ |___ |     |  | |__| | \|
:set-window-option synchronize-pane
```

Each pane in the window gets your input

(with the drawback that you must be able to fit them all on one window)

```
 _____     _       _
|_   _| __(_)_ __ | | ___
  | || '__| | '_ \| |/ _ \
  | || |  | | |_) | |  __/
  |_||_|  |_| .__/|_|\___|
            |_|

 _____ _                    _
|_   _| |__  _ __ ___  __ _| |_
  | | | '_ \| '__/ _ \/ _` | __|
  | | | | | | | |  __/ (_| | |_
  |_| |_| |_|_|  \___|\__,_|\__|

 _____    _        _
|_   _|__| |_ _ __(_)___
  | |/ _ \ __| '__| / __|
  | |  __/ |_| |  | \__ \
  |_|\___|\__|_|  |_|___/
```

Type once, run everywhere!