**tmux的使用方法和个性化配置**

[home](http://mingxinglai.com/cn/) [about](http://mingxinglai.com/cn/about/) [publication](http://mingxinglai.com/cn/publication/) [guestbook](http://mingxinglai.com/cn/guestbook/) [vitae](http://mingxinglai.com/cn/vitae/) [categories](http://mingxinglai.com/cn/categories/) [tags](http://mingxinglai.com/cn/tags/) [links](http://mingxinglai.com/cn/links/) [subscribe](http://mingxinglai.com/cn/feed.xml)

#介绍

[tmux](https://wiki.freebsdchina.org/software/t/tmux)是一个优秀的终端复用软件，即使非正常掉线，也能保证当前的任务运行，这一点对于 远程SSH访问特别有用，网络不好的情况下仍然能保证工作现场不丢失!此外，tmux完全使用键盘 控制窗口，实现窗口的切换功能。

简单地说，tmux对于我主要有两个功能（这应该也是tmux的主要功能）:

1. split窗口。可以在一个terminal下打开多个终端，也可以对当前屏幕进行各种split，即可以 同时打开多个显示范围更小的终端。
2. 在使用SSH的环境下，避免网络不稳定，导致工作现场的丢失。想象以下场景， 你在执行一条命令的过程中，由于网络不稳定，SSH连接断开了。这个时候，你就不知道之前 的那条命令是否执行成功。如果此时你打开了很多文件，进入了较深层次的目录，由于网络 不稳定，SSH连接断开。重新连接以后，你又不得不重新打开那些文件，进入那个深层次的 目录。如果使用了tmux，重新连接以后，就可以直接回到原来的工作环境，不但提高了工作 效率，还降低了风险，增加了安全性。

#安装

sudo apt-get install tmux

安装完成后输入命令tmux即可打开软件，界面十分简单，类似一个下方带有状态栏的终端控制台； 不出意外，这时候你会跟我第一次一样，觉得tmux没什么牛逼的。

#会话，窗口，面板

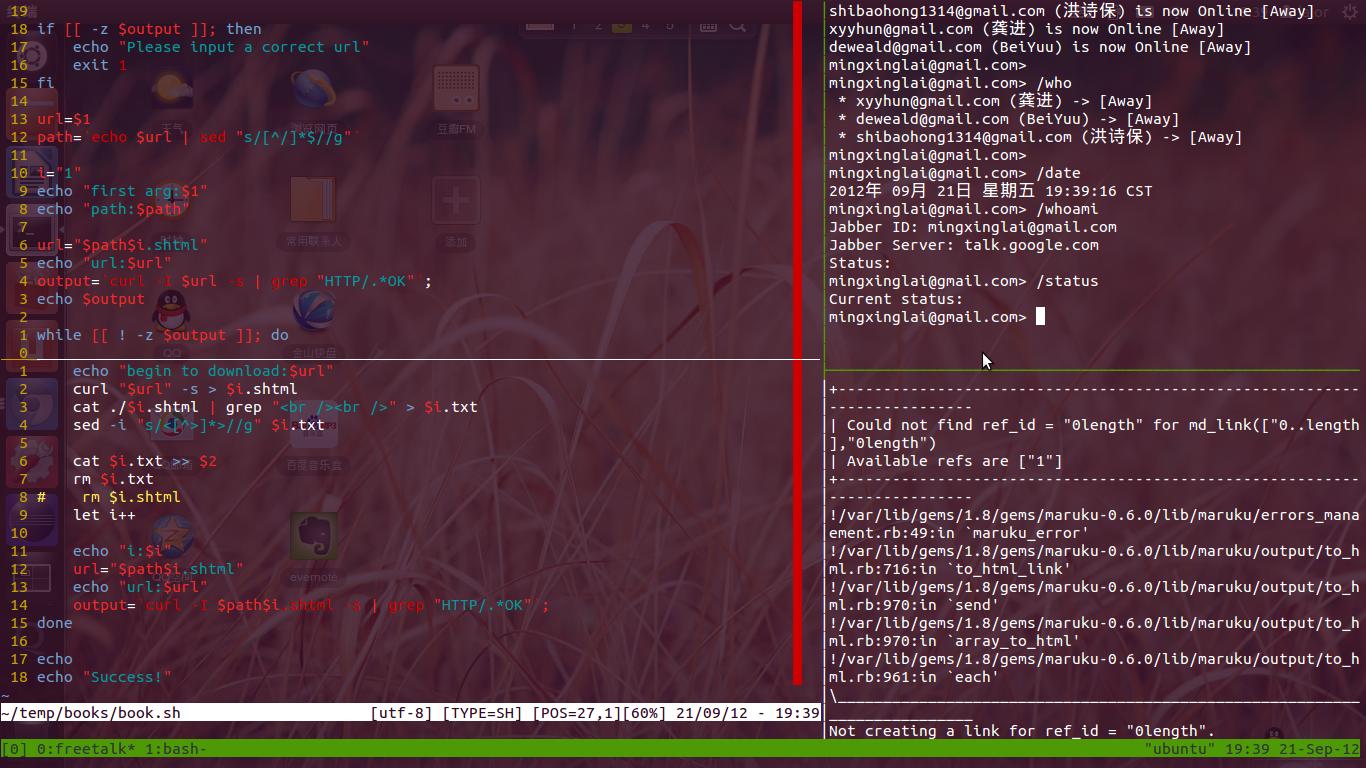
根据tmux的定义，在开启了tmux服务器后，会首先创建一个会话，而这个会话则会首先创建一个 窗口，其中仅包含一个面板；也就是说，这里看到的所谓终端控制台应该称作tmux的一个面板， 虽然其使用方法与终端控制台完全相同。

tmux使用C/S模型构建，主要包括以下单元模块：

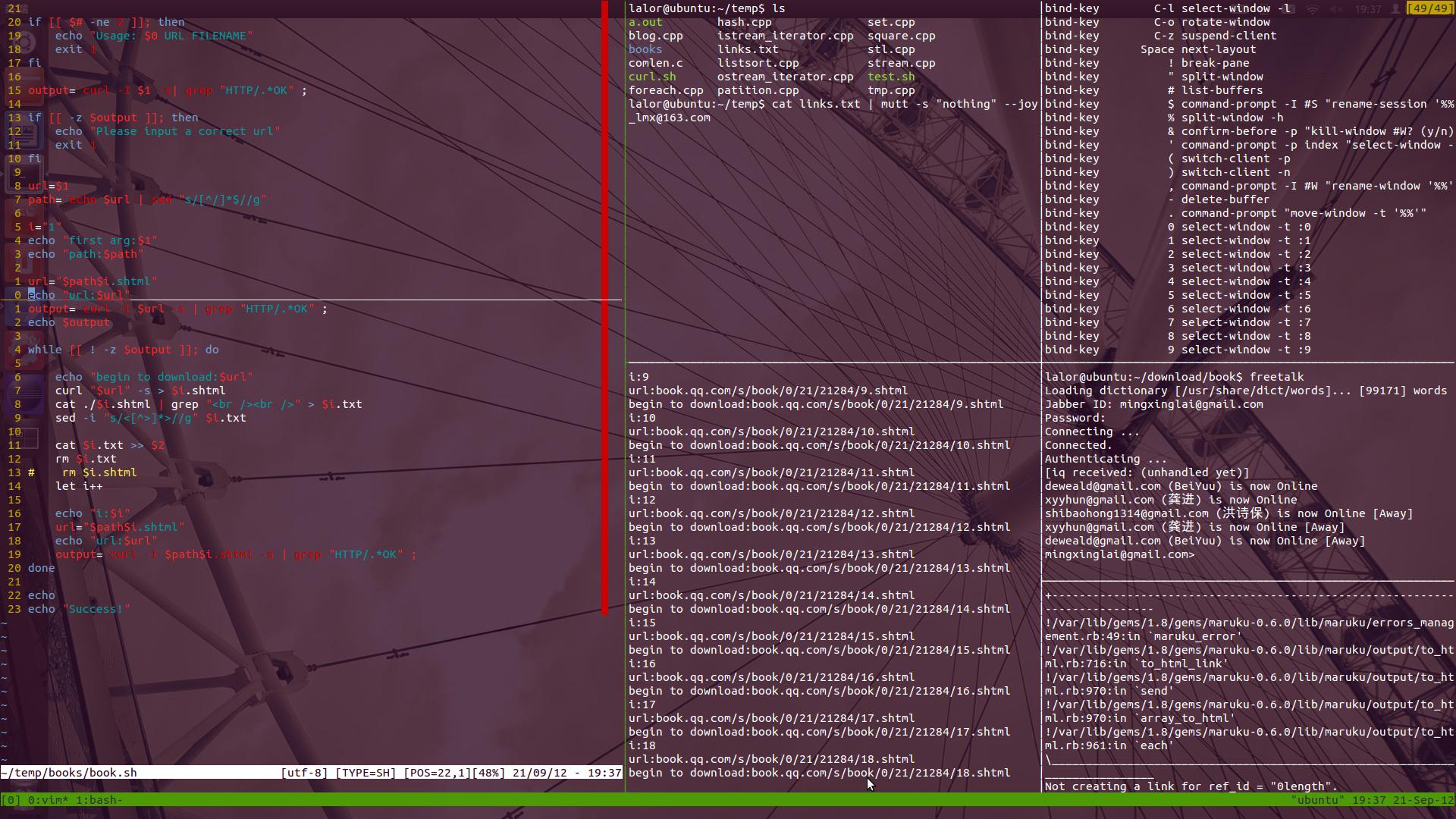
* server服务器。输入tmux命令时就开启了一个服务器。
* session会话。一个服务器可以包含多个会话
* window窗口。一个会话可以包含多个窗口。
* pane面板。一个窗口可以包含多个面板。

#初体验

我们先来看一张效果图，再来学习tmux的配置和使用我一般都是这么用，因为我的笔记本 只有13寸:-)



当然，做为一个准IT男，我还有一个很大的屏幕，所以我偶尔也这么用：



你所看到的只是X个会话中的一个会话，且是该会话下面的Y个窗口中的一个窗口。是的，你在一 个窗口里就可以拥有这么多个终端，这就是tmux的主要功能。好了，下面是tmux的常用按键，你 先自己玩一下再往下看吧。

#常用按键

这里需要说明一点的是，tmux的任何指令，都包含一个前缀，也就是说，你按了前缀(一组按键， 默认是Ctrl+b)以后，系统才知道你接下来的指令是发送给tmux的。

* C-b ? 显示快捷键帮助
* C-b C-o 调换窗口位置，类似与vim 里的C-w
* C-b 空格键 采用下一个内置布局
* C-b ! 把当前窗口变为新窗口
* C-b “ 横向分隔窗口
* C-b % 纵向分隔窗口
* C-b q 显示分隔窗口的编号
* C-b o 跳到下一个分隔窗口
* C-b 上下键 上一个及下一个分隔窗口
* C-b C-方向键 调整分隔窗口大小
* C-b c 创建新窗口
* C-b 0~9 选择几号窗口
* C-b c 创建新窗口
* C-b n 选择下一个窗口
* C-b l 切换到最后使用的窗口
* C-b p 选择前一个窗口
* C-b w 以菜单方式显示及选择窗口
* C-b t 显示时钟
* C-b ; 切换到最后一个使用的面板
* C-b x 关闭面板
* C-b & 关闭窗口
* C-b s 以菜单方式显示和选择会话
* C-b d 退出tumx，并保存当前会话，这时，tmux仍在后台运行，可以通过tmux attach进入 到指定的会话

#配置

我们先来看一下几个配置，这些配置才是我离不开tmux的原因:-)。tmux的配置文件是 ~/.tmux.conf，这个文件可能不存在，你可以自己新建。下面开始配置，首先，有没有 觉得tmux的前缀按起来太不方便了，ctrl与b键隔得太远，很多人把它映射成C+a，也就 是在配置文件(~/.tmux.conf)中加入下面这条语句：

#设置前缀为Ctrl + a

set -g prefix C-a

与此同时，取消默认的前缀按键：

#解除Ctrl+b 与前缀的对应关系

unbind C-b

配置完以后，重启tmux起效，或者先按C+b，然后输入：，进入命令行模式， 在命令行模式下输入：

source-file ~/.tmux.conf

你也可以跟我一样，在配置文件中加入下面这句话，以后改了只需要按前缀+r了。

#将r 设置为加载配置文件，并显示"reloaded!"信息

bind r source-file ~/.tmux.conf \; display "Reloaded!"

关于前缀，很多人都喜欢改成Ctrl+a，不过我个人更喜欢Ctrl+x，如果你是vim用户，你一定懂 的。还有就是面板的切换很不方便，需要先按前缀，再按方向键，还记得vim里面怎么切换各个 面板的吗？[tmux也可以](http://liuerfire.is-programmer.com/posts/32150.html)，因为它支持映射。把前缀映射改成Ctrl+x，再加入如下几条语句， 现在切换窗口就和vim一摸一样了，顿时觉得亲切了很多。

#up

bind-key k select-pane -U

#down

bind-key j select-pane -D

#left

bind-key h select-pane -L

#right

bind-key l select-pane -R

上面的最后一条语句会更改C-x l的功能，我挺喜欢这个功能的，因为我们很时候都是在两个窗 口或这两个面板中切换，所以我又加入如下语句

#select last window

bind-key C-l select-window -l

现在我的l键可不能随便按了，Ctrl+x l是切换面板，Ctrl+x Ctrl+l切换窗口，Ctrl+l清屏。

**复制模式copy-mode**

* 前缀 [ 进入复制模式
* 按 space 开始复制，移动光标选择复制区域
* 按 Enter 复制并退出copy-mode。
* 将光标移动到指定位置，按 PREIFX ] 粘贴

如果把tmux比作vim的话，那么我们大部分时间都是处于编辑模式，我们复制的时候可不可以像 vim一样移动呢？只需要在配置文件(~/.tmux.conf)中加入如下行即可。

#copy-mode 将快捷键设置为vi 模式

setw -g mode-keys vi

#会话

* C-x s 以菜单的方式查看并选择会话
* C-x :new-session 新建一个会话
* C-x d 退出并保存会话
* 终端运行 tmux attach 返回会话

#[命名会话](https://github.com/xuxiaodong/tmuxen)

tmux new -s session

tmux new -s session -d #在后台建立会话

tmux ls #列出会话

tmux attach -t session #进入某个会话

#使当前pane 最大化

从我用vim开始，就相信这世界上有些人比你自己还了解你自己想要什么，所以乔老爷子才能做 出让世人尖叫的产品，所以我经常浏览高手的.vimrc文件，我用了tmux不到3天就发现，它怎么 不能最大化当前pane?就像vim里的ZommWinPlugin.vim插件一样，你按下某键，当前窗口就最大 化了，获得最大可视界面。当目前为止，就还没有发现我有需求而别人没有的，同样，已有前人 找到的解决办法，方法如下：

# zoom pane <-> window

#http://tmux.svn.sourceforge.net/viewvc/tmux/trunk/examples/tmux-zoom.sh

bind ^z run "tmux-zoom"

##

#滚屏

滚屏要进入copy-mode，即前缀+[，然后就可以用上下键来滚动屏幕，配置了vi快捷键模式，就 可以像操作vi一样来滚动屏幕，非常的方便。退出直接按‘q’键即可。

#其他配置

下面是从[别人](https://github.com/xuxiaodong/tmuxen)那里摘录过来的配置，反正我是用不着，留着以后参考吧，或许屏幕前的你 有需要呢。

#

# author : Xu Xiaodong <xxdlhy@gmail.com>

# modified : 2012 Jul 22

#

#-- base settings --#

set -g default-terminal "screen-256color"

set -g display-time 3000

set -g escape-time 0

set -g history-limit 65535

set -g base-index 1

set -g pane-base-index 1

#-- bindkeys --#

# prefix key (Ctrl+a)

set -g prefix ^a

unbind ^b

bind a send-prefix

# split window

unbind '"'

# vertical split (prefix -)

bind - splitw -v

unbind %

bind | splitw -h # horizontal split (prefix |)

# select pane

bind k selectp -U # above (prefix k)

bind j selectp -D # below (prefix j)

bind h selectp -L # left (prefix h)

bind l selectp -R # right (prefix l)

# resize pane

bind -r ^k resizep -U 10 # upward (prefix Ctrl+k)

bind -r ^j resizep -D 10 # downward (prefix Ctrl+j)

bind -r ^h resizep -L 10 # to the left (prefix Ctrl+h)

bind -r ^l resizep -R 10 # to the right (prefix Ctrl+l)

# swap pane

# swap with the previous pane (prefix Ctrl+u)

bind ^u swapp -U

# swap with the next pane (prefix Ctrl+d)

bind ^d swapp -D

# misc

# select the last pane (prefix e)

bind e lastp

# select the last window (prefix Ctrl+e)

bind ^e last

# kill pane (prefix q)

bind q killp

# kill window (prefix Ctrl+q)

bind ^q killw

# copy mode

# enter copy mode (prefix Escape)

bind Escape copy-mode

# paste buffer (prefix Ctrl+p)

bind ^p pasteb

# select (v)

bind -t vi-copy v begin-selection

# copy (y)

bind -t vi-copy y copy-selection

# zoom pane <-> window

#http://tmux.svn.sourceforge.net/viewvc/tmux/trunk/examples/tmux-zoom.sh

bind ^z run "tmux-zoom"

# app

# htop (prefix !)

bind ! splitw htop

# man (prefix m)

bind m command-prompt "splitw 'exec man %%'"

# perl func (prefix @)

bind @ command-prompt "splitw 'exec perldoc -t -f %%'"

# perl var (prefix \*)

bind \* command-prompt "splitw 'exec perldoc -t -v %%'"

# perl doc (prefix %)

bind % command-prompt "splitw 'exec perldoc -t %%'"

# ruby doc (prefix /)

bind / command-prompt "splitw 'exec ri %%'"

# reload config (prefix r)

bind r source ~/.tmux.conf \; display "Configuration reloaded!"

#-- statusbar --#

set -g status-utf8 on

set -g status-interval 1

set -g status-keys vi

setw -g mode-keys vi

setw -g automatic-rename off

#-- colorscheme --#

#https://github.com/daethorian/conf-tmux/blob/master/colors/zenburn.conf

# modes

setw -g clock-mode-colour colour223

setw -g mode-attr bold

setw -g mode-fg colour223

setw -g mode-bg colour235

# panes

set -g pane-border-bg colour234

set -g pane-border-fg colour234

set -g pane-active-border-bg colour232

set -g pane-active-border-fg colour232

# statusbar

set -g status-justify centre

set -g status-bg colour235

set -g status-fg colour248

set -g status-attr dim

#[default]» #[fg=colour187]#S #[default]• #[fg=colour187]w#I.p#P#[default]"

set -g status-left "

set -g status-left-attr bright

set -g status-left-length 20

#[fg=colour174]#(/home/xiaodong/bin/uptime) #[default]• #[fg=colour174]#(cut -d ' ' -f 1-3 /proc/loadavg)"

set -g status-right "

set -g status-right-attr bright

set -g status-right-length 80

setw -g window-status-current-fg colour223

setw -g window-status-current-bg colour237

setw -g window-status-current-attr bold

setw -g window-status-current-format "#I:#W#F"

#setw -g window-status-alert-attr bold

#setw -g window-status-alert-fg colour255

#setw -g window-status-alert-bg colour160

# messages

set -g message-attr bold

set -g message-fg colour223

set -g message-bg colour235

set -g visual-activity on

#快速启动tmux

如果觉得每次都要打开tmux，然后在打开几个窗口和面板很麻烦，那么下面这个脚本你一定会 喜欢。参考[这里](https://github.com/xuxiaodong/tmuxen/blob/master/tmuxen)

#!/bin/sh

#

# name : tmuxen， tmux environment made easy

# author : Xu Xiaodong xxdlhy@gmail.com

# license : GPL

# created : 2012 Jul 01

# modified : 2012 Jul 02

#

cmd=$(which tmux) # tmux path

session=codefun # session name

if [ -z $cmd ]; then

echo "You need to install tmux."

exit 1

fi

$cmd has -t $session

if [ $? != 0 ]; then

$cmd new -d -n vim -s $session "vim"

$cmd splitw -v -p 20 -t $session "pry"

$cmd neww -n mutt -t $session "mutt"

$cmd neww -n irssi -t $session "irssi"

$cmd neww -n cmus -t $session "cmus"

$cmd neww -n zsh -t $session "zsh"

$cmd splitw -h -p 50 -t $session "zsh"

$cmd selectw -t $session:5

fi

$cmd att -t $session

exit 0

如果还有其他需要，可以去[这里](http://linuxtoy.org/archives/scripting-tmux.html)看看，当然了，最全的参考文档，就是man tmux。 happy tmux!