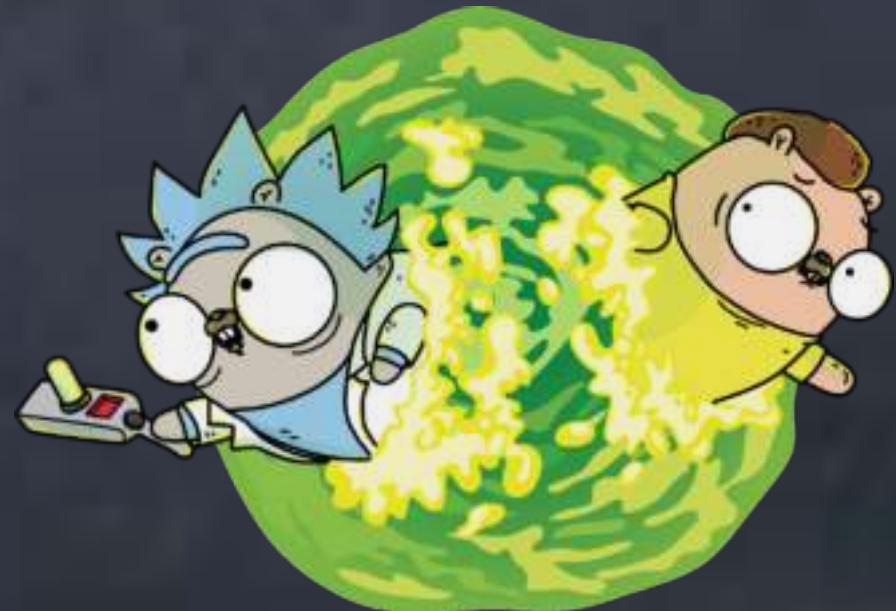
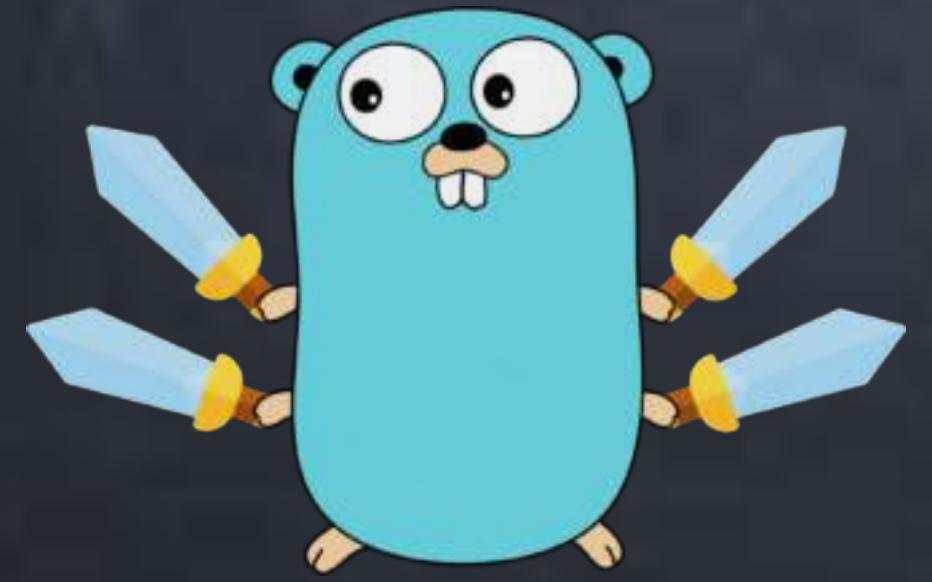


v3

优雅的编程者

between a and c

- xiaorui.cc





先从xx windows开始...

- mac
- ubuntu
- manjaro (arch linux)



menu

② 我们的目标

② 高效编程

② 好看绚丽

② 保护肾腰

② 应用篇

② 终端篇

② 设备篇

② 终极篇

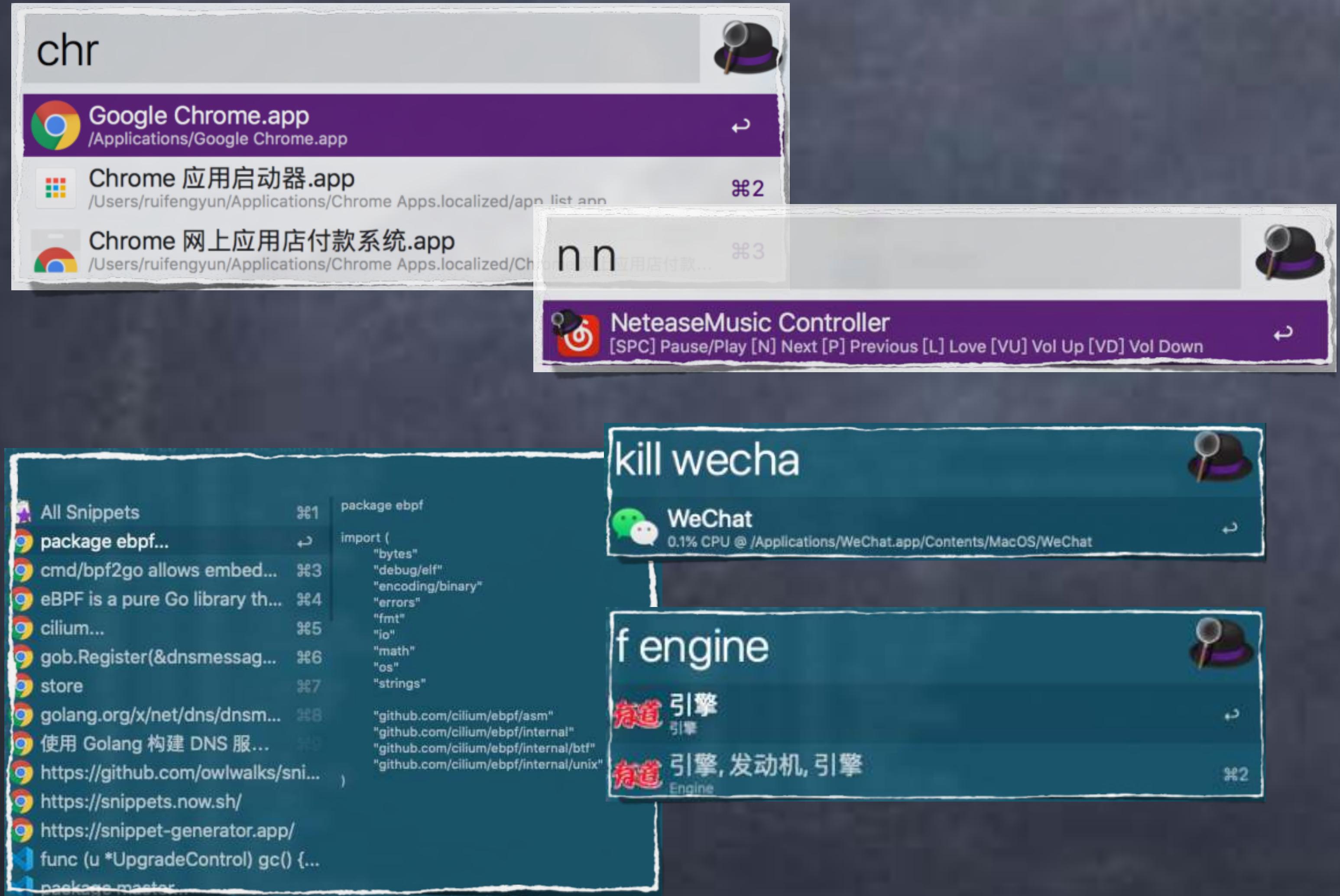


应用篇

- **chrome**
- **alfred** 快捷管理
- **notion && evernote**
- **hyperSwitch & contexts**
- **iterm2** 终端
- **vscode** 通用编辑器
- **neovim** 定制编辑器
- **sketch** 画图
- **mindnode** 思维导图
- **dash** 文档
- **magnet** 分屏
- **karabiner** 键位映射
- **clashx** 你懂的



alfred



- 自定义workflow

- 拉起进程

- 切换进程

- 各种搜索

- 定制脚本

- 粘贴历史

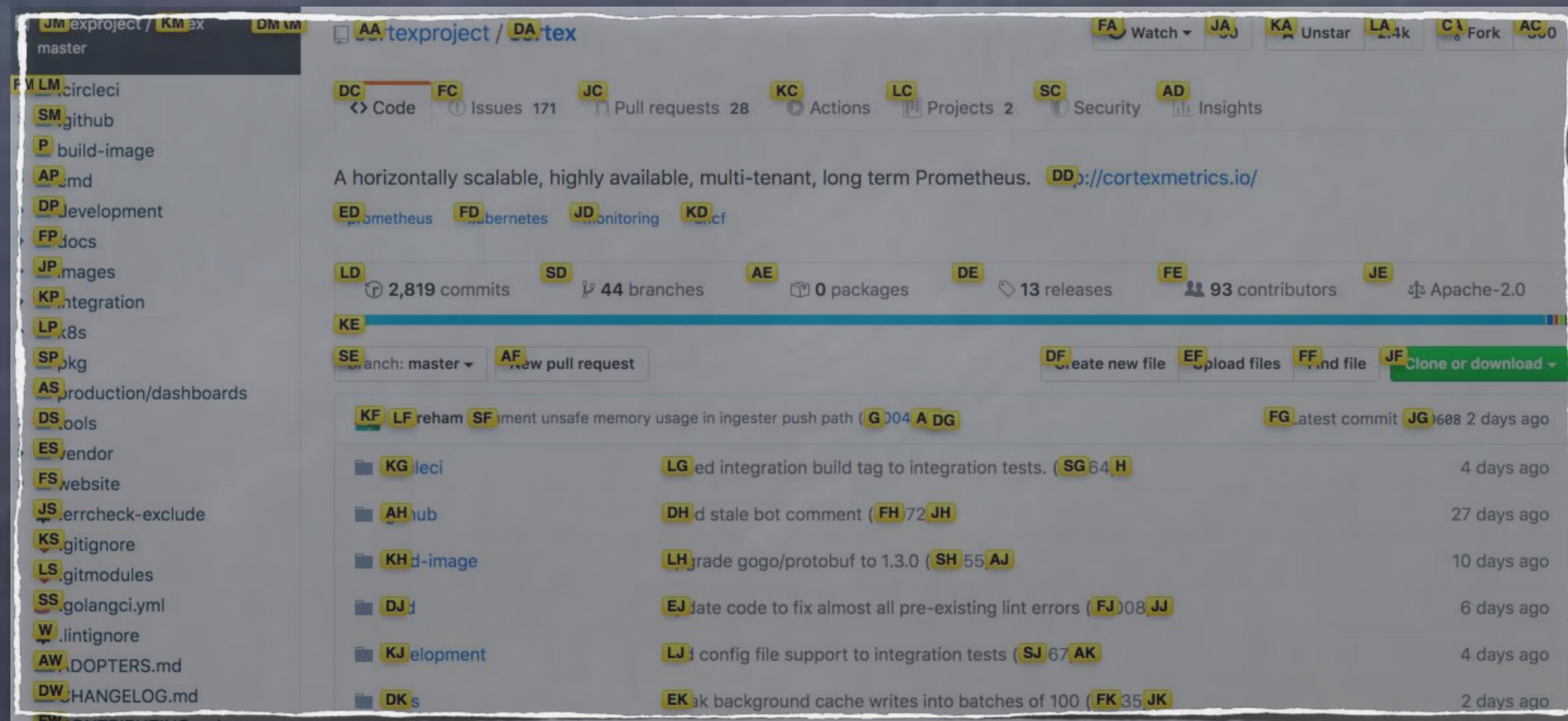
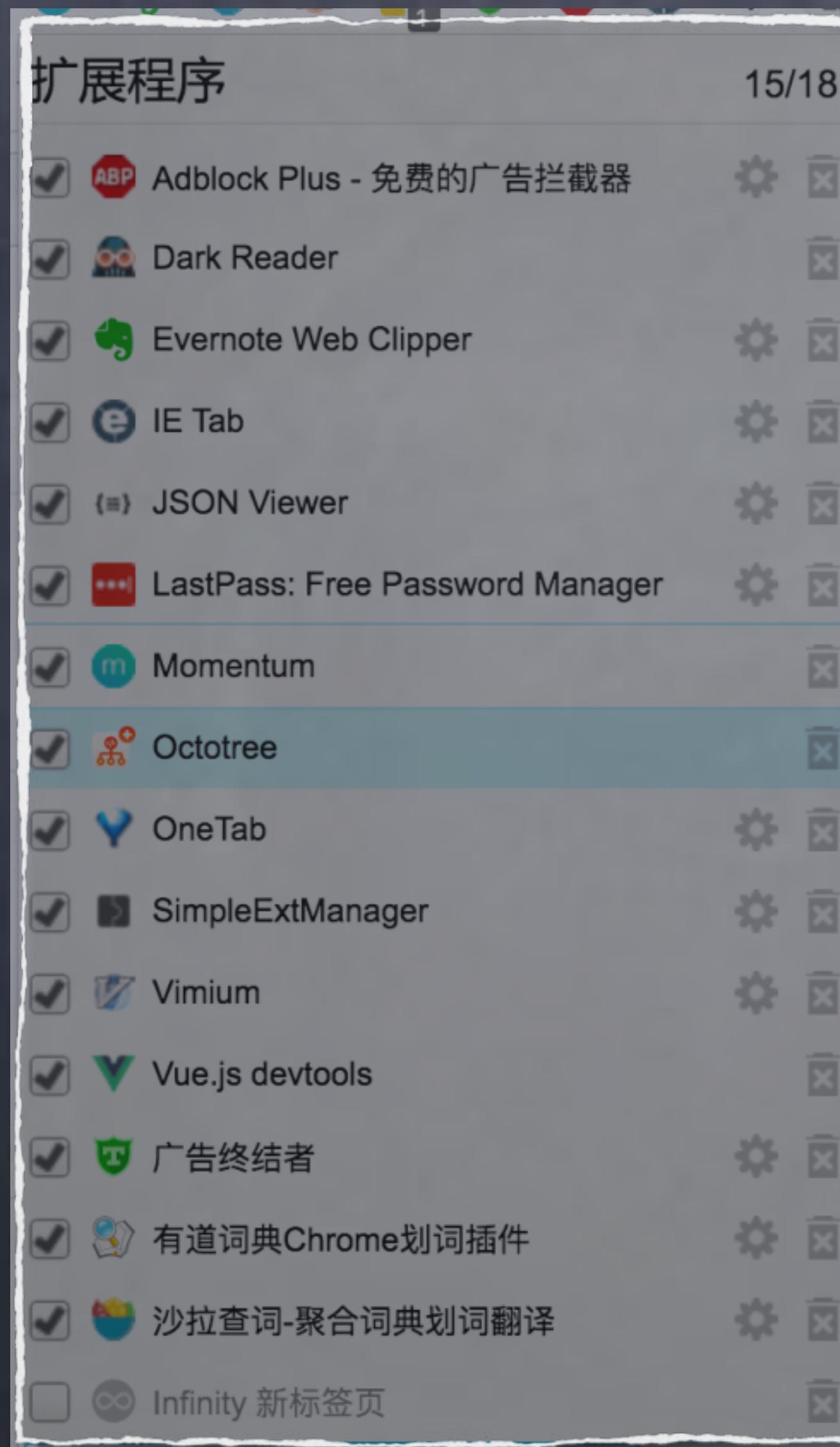
- ...

contexts



- 窗口管理器
- 热度排序
- 区分相同应用
- 搜索匹配
- 索引号
- 字符串
- 劫持默认 tab
- ...

chrome

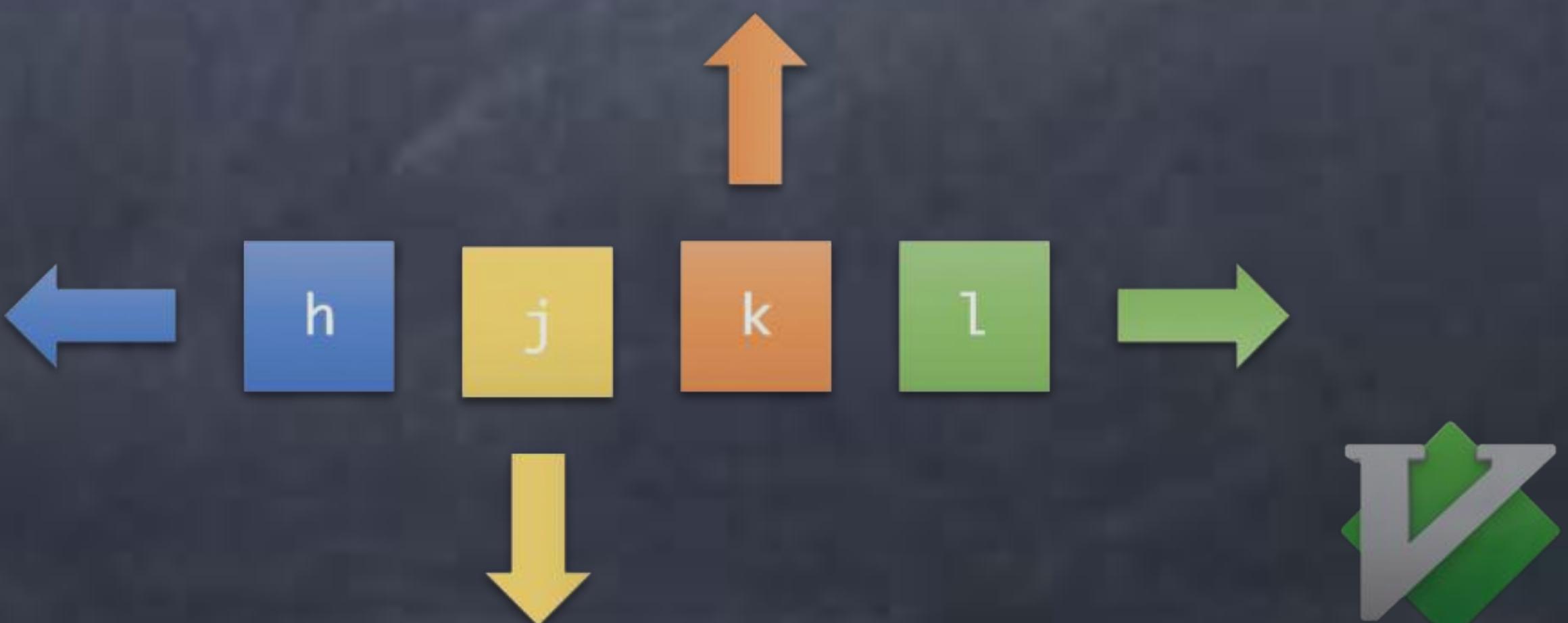


vimium & Octotree

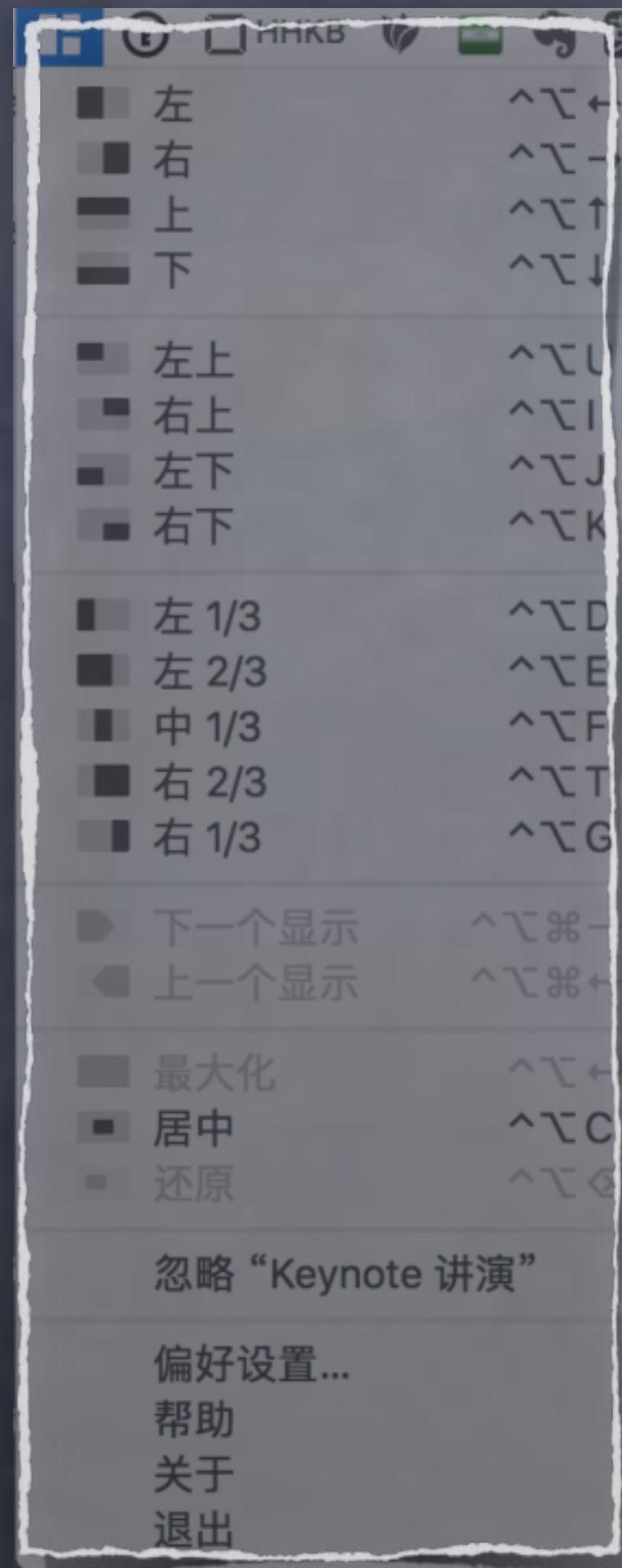
karabiner

- 把 **caps_lock** 改成 **control** 键位
- caps_lock** 属于“非”常用键
- 左下角 **control** 引起小拇指
- 改其他组合键为 **caps_lock**

- 为什么要改方向键 !!!
- 方向键通常在右下角, 所以每次都要抬手移过去...
- 上下左右 = **control + kjhl**
- 同理改 **pageup/pagedown**



高效窗口管理



这类工具很多...

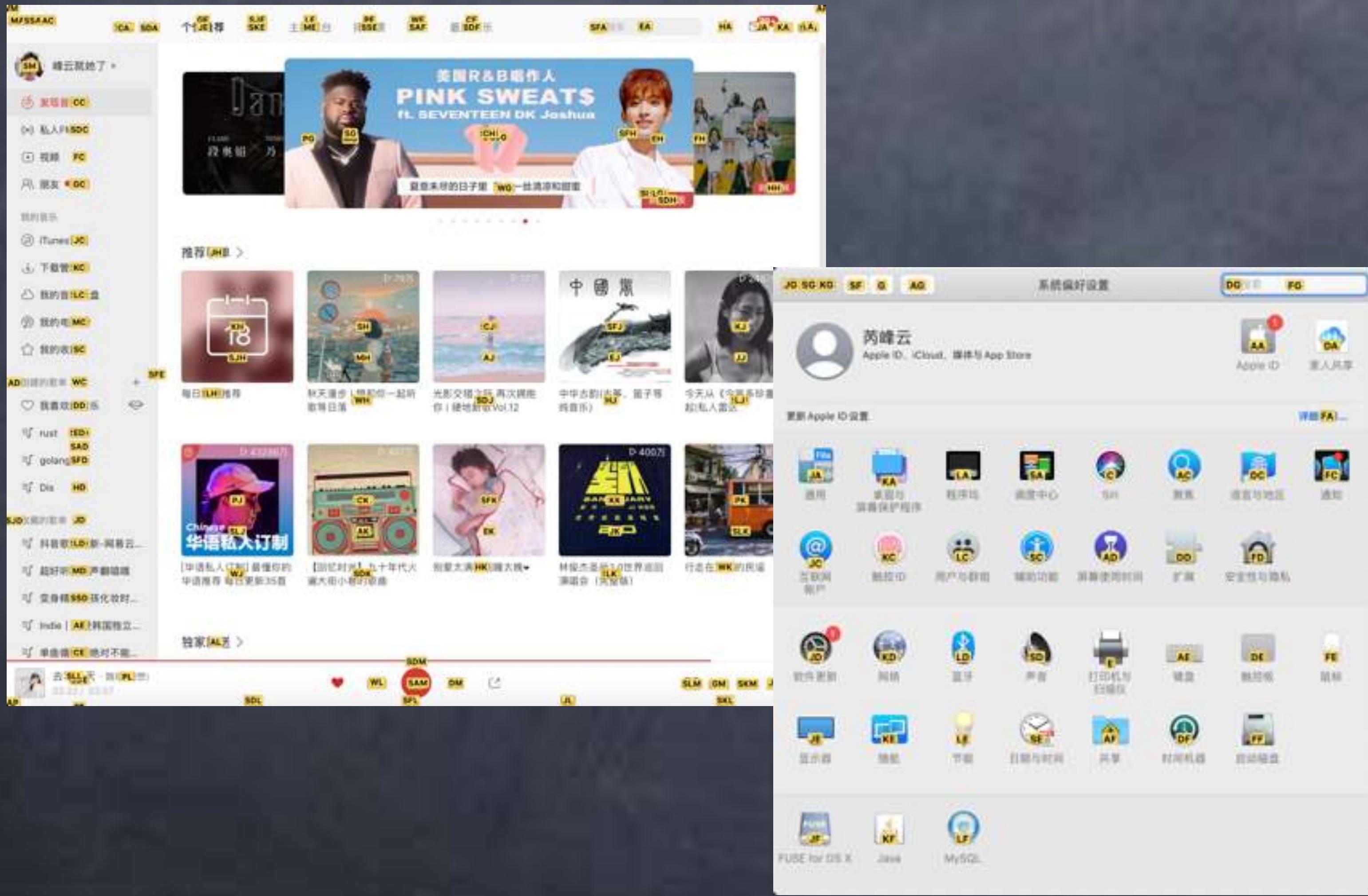
● 窗口尺寸变更

● 多屏幕之间转移

● 窗口对齐

● ...

vimac

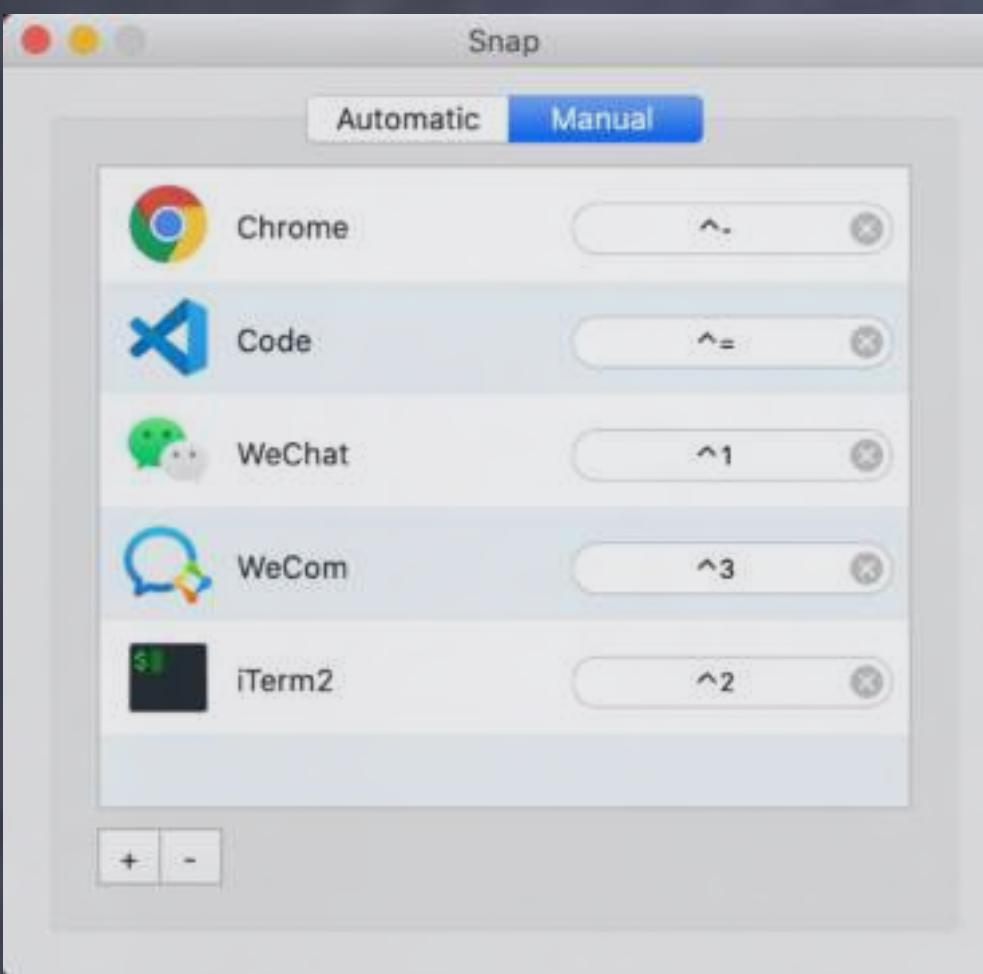


mac + vim

展示快捷键

滚动屏幕

other



● snap

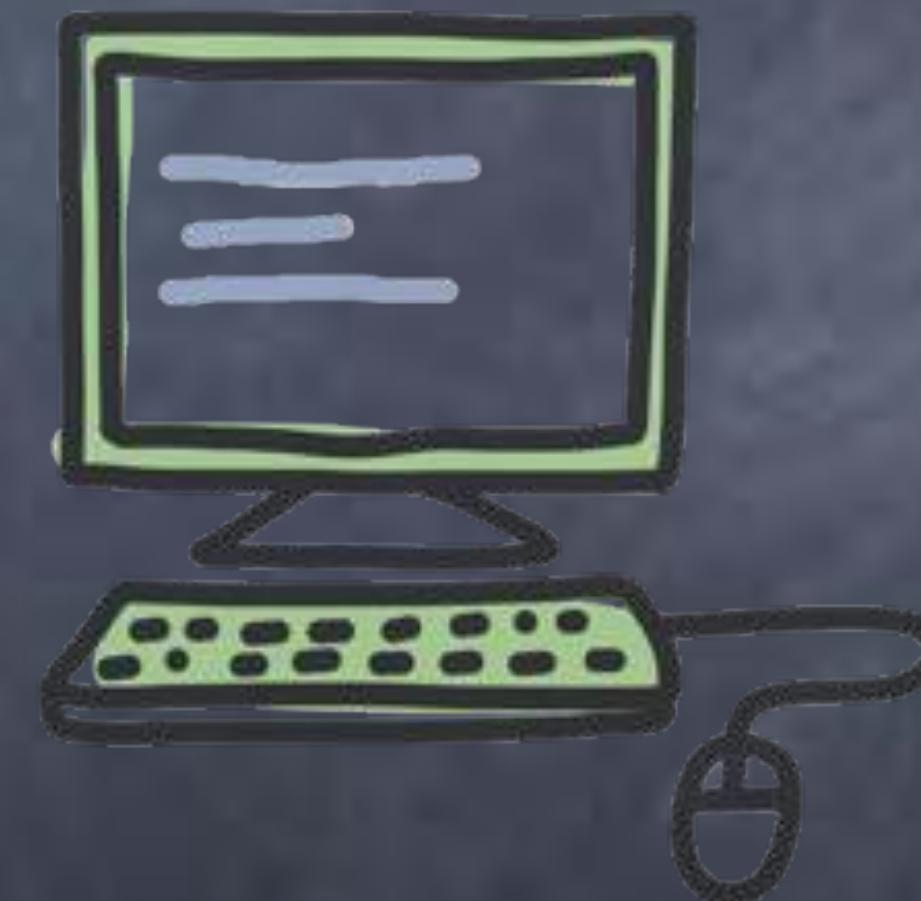
- 一键到达
- 一键切换
- / ... /



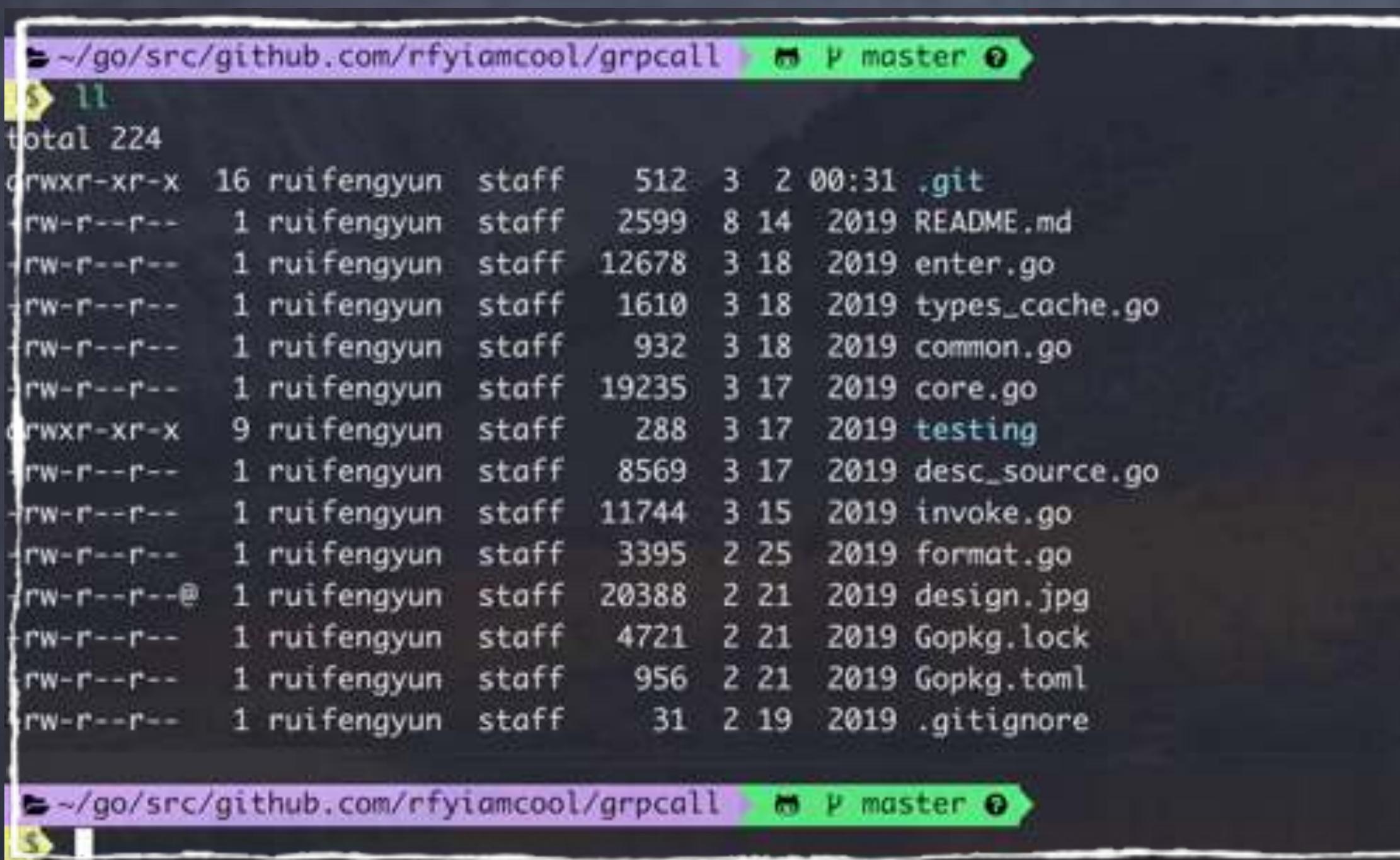
- 架构图
- processon.com
- omniGraffle
- ppt
- yanshuo.io
- slides.com
- reveal.js

终端篇

- `iterm2`
- `zsh`
- `tmux` & `tmuxp`
- `vim`
- 好用的命令
- 快捷键



iterm2



~/go/src/github.com/rfyiamcool/grpcall P master

```
total 224
drwxr-xr-x  16 rufengyun  staff   512  3 2 00:31 .git
rw-r--r--  1 rufengyun  staff  2599  8 14 2019 README.md
rw-r--r--  1 rufengyun  staff 12678  3 18 2019 enter.go
rw-r--r--  1 rufengyun  staff 1610   3 18 2019 types_cache.go
rw-r--r--  1 rufengyun  staff   932  3 18 2019 common.go
rw-r--r--  1 rufengyun  staff 19235  3 17 2019 core.go
drwxr-xr-x  9 rufengyun  staff   288  3 17 2019 testing
rw-r--r--  1 rufengyun  staff  8569  3 17 2019 desc_source.go
rw-r--r--  1 rufengyun  staff 11744  3 15 2019 invoke.go
rw-r--r--  1 rufengyun  staff   395  2 25 2019 format.go
rw-r--r--@ 1 rufengyun  staff 20388  2 21 2019 design.jpg
rw-r--r--  1 rufengyun  staff  4721  2 21 2019 Gopkg.lock
rw-r--r--  1 rufengyun  staff   956  2 21 2019 Gopkg.toml
rw-r--r--  1 rufengyun  staff    31  2 19 2019 .gitignore
```

~/go/src/github.com/rfyiamcool/grpcall P master

• **iterm2**

• **font**

• **nerd-fonts**

• **theme**

• **dracula (current theme)**

• **molokai**

• **solarized dark || light**

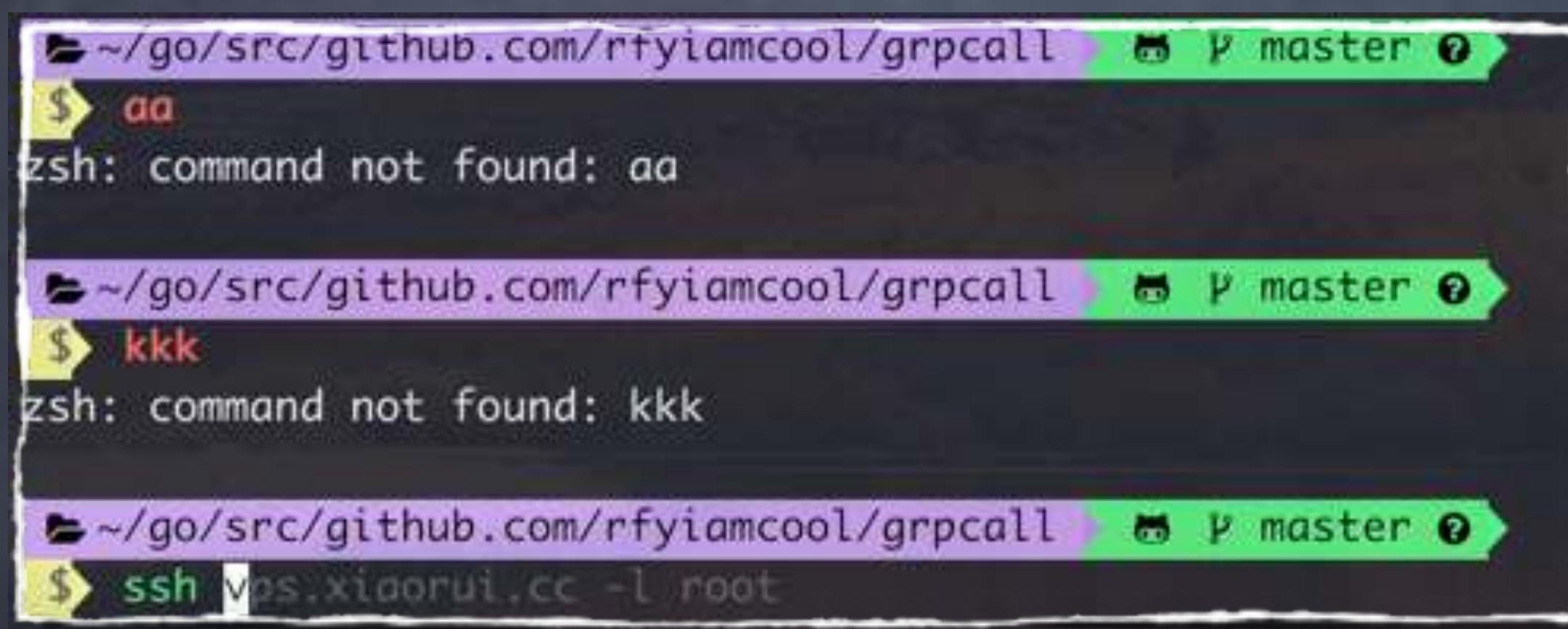
iterm2

- 最前 & 最后
 - control + a
 - control + b
- 单词删
 - option + backspace
- 历史
 - shift + cmd + h
- 搜索终端
 - shift + cmd + o



- 单词移动
 - option + b
 - option + f
- iTerm2 窗口跳转
 - command + 左右

iterm2 (zsh + theme)



The screenshot shows three vertically stacked iTerm2 windows. Each window has a purple header bar with the path `~/go/src/github.com/rfyiamcool/grpcall` and a green footer bar with a cat icon and the text `master`. The first window has a yellow prompt and shows the error `zsh: command not found: aa`. The second window has a red prompt and shows the error `zsh: command not found: kkk`. The third window has a green prompt and shows the command `ssh vos.xiaorui.cc -l root`.

- theme
 - powerlevel9k
 - powerlevel10k (async)
 - pure (async)
 - spaceship
- plugin
 - git & vi-mode
 - autojump
 - zsh-syntax-highlighting
 - zsh-autosuggestions

zsh plugin vi-mode



```
$ curl 'https://dss0.baidu.com/73F1bjeh1BF3odCf/it/u=3624547722,2465723046&fm=85&s=638A87206B66E25451477460300E0F5' -H 'authority: dss0.baidu.com' -H 'user-agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_13_6) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/80.0.3987.122 Safari/537.36' -H 'sec-fetch-dest: image' -H 'accept: image/webp,image/apng,image/*,*/*;q=0.8' -H 'sec-fetch-site: same-site' -H 'sec-fetch-mode: no-cors' -H 'referer: https://www.baidu.com/s?wd=mm&pn=20&oq=mm&tn=baiduhome_pg&ie=utf-8&usm=4&rsv_idx=2&rsv_pq=e233615400014ba3&rsv_t=da32cXXE6ePHEqEzvydJPqpaj7Sf5kuMF%2BEqlttZ%2FJf0lGolXYy2Ken7ja7X7%2F35ewJI&rsv_page=1' -H 'accept-language: zh-CN,zh;q=0.9,en;q=0.8,zh-TW;q=0.7' -H 'cookie: PSTM=148420423; BIDUPSID=205A550BB27CF17CBA71F; BAIDUCUID=++; BAIDUID=CCF36181DD304D41E58DEE2BD377D:FG=1; BDUSS=Ek2R3RZTmpIVHZRekl4VjV3SWLMNC1qb1Q0VldlVmFVdzU5LWwya29SRzVIU2h1RVFBQUFBJCQAAAAAAAAAAEAAADt5jA53Me35dTGNТИxAAAAAAAAAAAAAAALmQAF65kABed; MCITY=-%3A; BDRCVFR[feWj1Vr5u3D]=I67x6TjHwwYf0; delPer=0; PSINO=2; BDORZ=B490B5EBF6F3CD402E515D22BCDA1598; H_PS_PSSID=1422_21122_30906_30824_22157' --compressed
```

⌚ esc

⌚ 用vim的模式操作命令行

⌚ esc + v

⌚ 导入到vim里再去操作命令



```
1 curl 'https://dss0.baidu.com/73F1bjeh1BF3odCf/it/u=3624547722,2465723046&fm=85&s=638A87206B66E25451477460300E0F5' -H 'authority: dss0.baidu.com' -H 'user-agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_13_6) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/80.0.3987.122 Safari/537.36' -H 'sec-fetch-dest: image' -H 'accept: image/webp,image/apng,image/*,*/*;q=0.8' -H 'sec-fetch-site: same-site' -H 'sec-fetch-mode: no-cors' -H 'referer: https://www.baidu.com/s?wd=mm&pn=20&oq=mm&tn=baiduhome_pg&ie=utf-8&usm=4&rsv_idx=2&rsv_pq=e233615400014ba3&rsv_t=da32cXXE6ePHEqEzvydJPqpaj7Sf5kuMF%2BEqlttZ%2FJf0lGolXYy2Ken7ja7X7%2F35ewJI&rsv_page=1' -H 'accept-language: zh-CN,zh;q=0.9,en;q=0.8,zh-TW;q=0.7' -H 'cookie: PSTM=148420423; BIDUPSID=205A550BB27CF17CBA71F; BAIDUCUID=++; BAIDUID=CCF36181DD304D41E58DEE2BD377D:FG=1; BDUSS=Ek2R3RZTmpIVHZRekl4VjV3SWLMNC1qb1Q0VldlVmFVdzU5LWwya29SRzVIU2h1RVFBQUFBJCQAAAAAAAAAAEAAADt5jA53Me35dTGNТИxAAAAAAAAAAAAAAALmQAF65kABed; MCITY=-%3A; BDRCVFR[feWj1Vr5u3D]=I67x6TjHwwYf0; delPer=0; PSINO=2; BDORZ=B490B5EBF6F3CD402E515D22BCDA1598; H_PS_PSSID=1422_21122_30906_30824_22157' --compressed
```

fzf

智能历史记录查询

command + r

```
~$ ~/go/src/github.com
$ 9365 go14 run dd.go
9366 go run
9389 vim runtime2.go
9417 vim src/runtime/panic.go
9425 go14 test -bench=. -run=none
9427 go test -bench=. -run=none
9453 vim runtime/panic.go
9456 vim /Users/ruifengyun/bin/go/src/runtime/panic.go
9465 go test -bench=. -run=none -benchmem .
9466 go14 test -bench=. -run=none -benchmem .
9471 go run a.go
9472 go14 run a.go
> 9596 go run main.go
1124/10000 +S
> go run
```

更快可模糊的文件搜索

option + c

```
~$ ~/go/src/github.com
$ > github.com rfyiamcool grpc
300/17953
> rfyiamcool/grpc-proxy/vendor/github.com
rfyiamcool/istio_grpc/vendor/github.com
rfyiamcool/grpc-proxy_2/vendor/github.com
rfyiamcool/grpc-sidecar/vendor/github.com
rfyiamcool/grpc-proxy/vendor/github.com/rs
rfyiamcool/grpc-proxy/vendor/github.com/pkg
rfyiamcool/grpc-proxy/vendor/github.com/gogo
rfyiamcool/grpc-proxy_2/vendor/github.com/rs
rfyiamcool/grpc-sidecar/vendor/github.com/rs
rfyiamcool/grpc-proxy_2/vendor/github.com/pkg
rfyiamcool/grpc-sidecar/vendor/github.com/pkg
rfyiamcool/grpc-proxy/vendor/github.com/beorn7
```

fzf

路径使用 ****** 可模糊查询

```
~/go/src/github.com/rfyiamcool
$ cd **
> grpc
936/2207
  grpcx
  grpcpp
> grpcall
  grpc-proxy
  grpc-codec
  istio_grpc
~/go/src/github.com
$ vim rfyiamcool/**
8362/8362
rfyiamcool
rfyiamcool/backoff
rfyiamcool/backoff/backoff_test.go
rfyiamcool/backoff/backoff.go
rfyiamcool/backoff/README.md
rfyiamcool/grpcall
rfyiamcool/grpcall/Gopkg.toml
rfyiamcool/grpcall/design.jpg
rfyiamcool/grpcall/format.go
```

fzf 用法多种多样

```
~/go/src/github.com
$ kill
> chrome
47/318
  501  482   1  0 五 03下午 ?? 77:24.98 /Applications/Google Chrome.app/Contents/
  501  542   482  0 五 03下午 ?? 11:12.63 /Applications/Google Chrome.app/Contents/
  501  2270  482  0 五 03下午 ?? 2:26.89 /Applications/Google Chrome.app/Contents/
  501  531   1  0 五 03下午 ?? 0:00.17 /Applications/Google Chrome.app/Contents/
  501 25704  482  0 六 08下午 ?? 1:17.38 /Applications/Google Chrome.app/Contents/
  501 19252  482  0 9:54下午 ?? 1:13.87 /Applications/Google Chrome.app/Contents/
  501 20102  482  0 10:34下午 ?? 0:00.95 /Applications/Google Chrome.app/Contents/
  501 22778  482  0 11:05下午 ?? 0:08.65 /Applications/Google Chrome.app/Contents/
```

ripgrep vs ag

```
~/go/src/github.com
$ time rg .vimrc|wc -l
15
rg .vimrc 4.68s user 9.37s system 129% cpu 10.833 total
wc -l 0.00s user 0.00s system 0% cpu 10.832 total

~/go/src/github.com
$ time ag .vimrc|wc -l
15
ag .vimrc 2.97s user 8.57s system 130% cpu 8.867 total
wc -l 0.00s user 0.00s system 0% cpu 8.866 total

~/go/src/github.com
$ time rg -n -w '[A-Z]+_SUSPEND'|wc -l
133
rg -n -w '[A-Z]+_SUSPEND' 4.55s user 7.88s system 145% cpu 8.518 total
wc -l 0.00s user 0.00s system 0% cpu 8.517 total

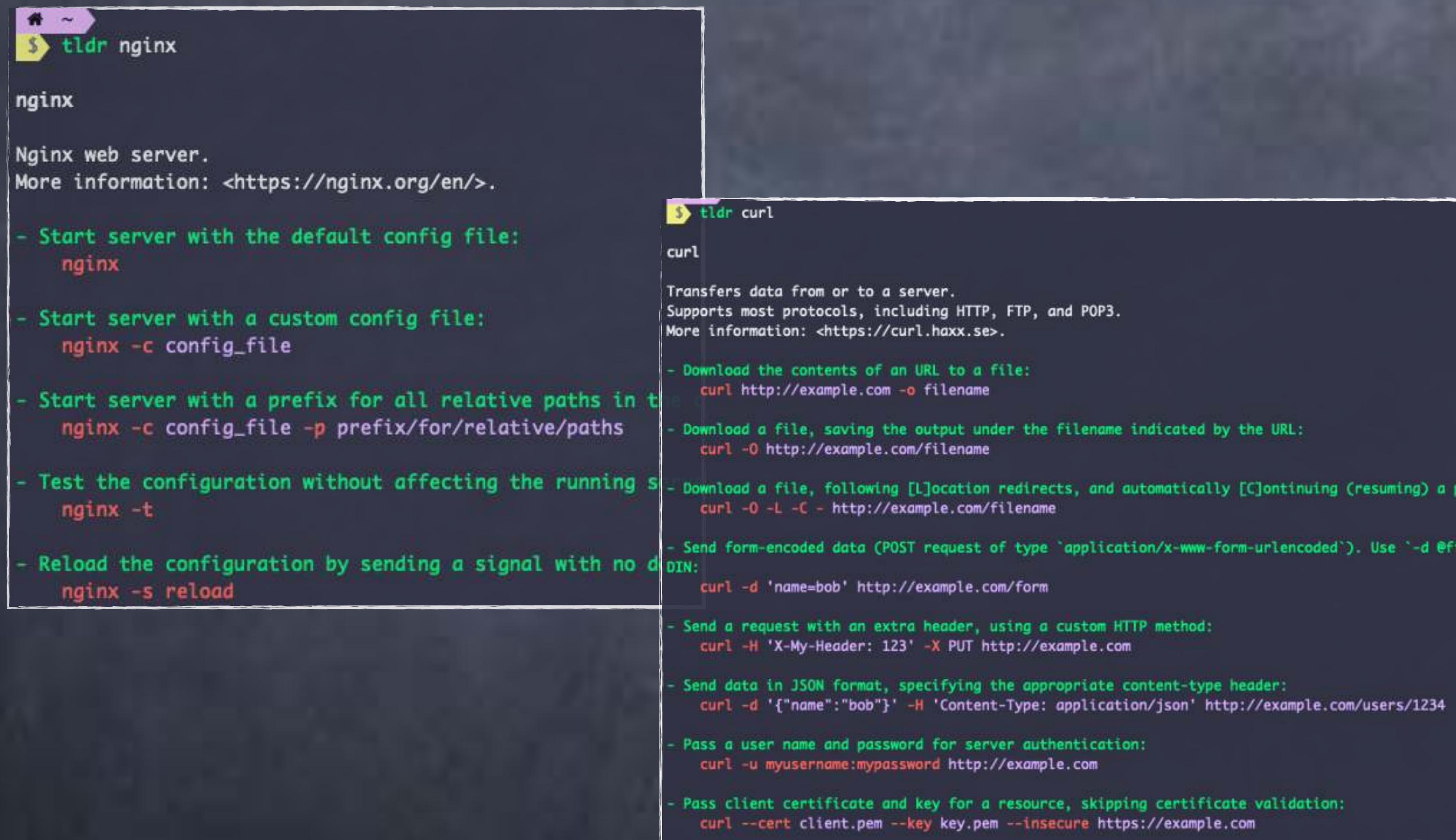
~/go/src/github.com
$ time ag -w '[A-Z]+_SUSPEND'|wc -l
133
ag -w '[A-Z]+_SUSPEND' 3.93s user 6.70s system 161% cpu 6.568 total
wc -l 0.00s user 0.00s system 0% cpu 6.567 total
```

⌚ simple test

⌚ ag > ripgrep > gnu grep



tldr



\$ tldr nginx

nginx

Nginx web server.
More information: <<https://nginx.org/en/>>.

- Start server with the default config file:
nginx
- Start server with a custom config file:
nginx -c config_file
- Start server with a prefix for all relative paths in the config file:
nginx -c config_file -p prefix/for/relative/paths
- Test the configuration without affecting the running server:
nginx -t
- Reload the configuration by sending a signal with no downtime:
nginx -s reload

\$ tldr curl

curl

Transfers data from or to a server.
Supports most protocols, including HTTP, FTP, and POP3.
More information: <<https://curl.haxx.se>>.

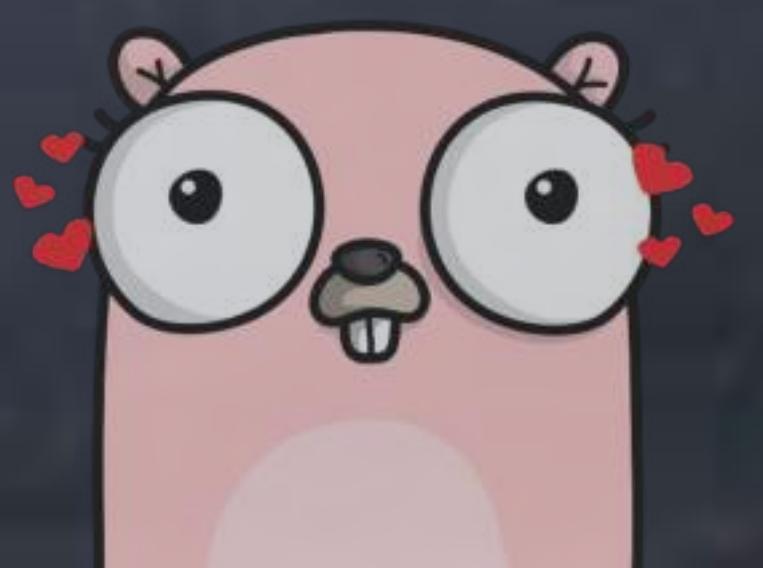
- Download the contents of an URL to a file:
curl http://example.com -o filename
- Download a file, saving the output under the filename indicated by the URL:
curl -O http://example.com/filename
- Download a file, following [L]ocation redirects, and automatically [C]ontinuing (resuming) a partially downloaded file:
curl -O -L -C - http://example.com/filename
- Send form-encoded data (POST request of type `application/x-www-form-urlencoded`). Use `-d @filename`:
curl -d 'name=bob' http://example.com/form
- Send a request with an extra header, using a custom HTTP method:
curl -H 'X-My-Header: 123' -X PUT http://example.com
- Send data in JSON format, specifying the appropriate content-type header:
curl -d '{"name":"bob"}' -H 'Content-Type: application/json' http://example.com/users/1234
- Pass a user name and password for server authentication:
curl -u myusername:mypassword http://example.com
- Pass client certificate and key for a resource, skipping certificate validation:
curl --cert client.pem --key key.pem --insecure https://example.com

- ⌚ 命令手册
- ⌚ TL;DR
- ⌚ Too long, Don't read
- ⌚ 比 man 更直观
- ⌚ 比 man 有颜色
- ⌚ 比 man 有常用的例子
- ⌚ ...

fd

```
~/go/src/github.com/rfyiamcool/grpcall ➜ master ?  
$ fd 'en..r'  
enter.go  
  
~/go/src/github.com/rfyiamcool/grpcall ➜ master ?  
$ fd -e md  
README.md  
  
~/go/src/github.com/rfyiamcool/grpcall ➜ master ?  
$ fd -HI 'ser.*go$'  
testing/dynamic_register/server.go  
testing/test_server.go  
  
~/go/src/github.com/rfyiamcool/grpcall ➜ master ?  
$ fd -t f -x wc -l  
 46 Gopkg.toml  
 35 design.jpg  
124 format.go  
289 desc_source.go
```

- vs find
- 高亮
- 更快
- 更优雅
- 并发管道



bat vs ccat

- bat

- 丰富的颜色主题
- 自动分页
- 暗色行号
- more...

- ccat

- 简单高亮
- 支持无后缀



```
bat ee.go
File: ee.go

1 package main
2
3 import (
4     "fmt"
5     "net/http"
6     "time"
7 )
8
9 func hello(w http.ResponseWriter, req *http.Request) {
10
11     // A `context.Context` is created for each request by
12     // the `net/http` machinery, and is available with
13     // the `Context()` method.
14     ctx := req.Context()
15     fmt.Println("server: hello handler started")
16     defer fmt.Println("server: hello handler ended")
17
18     // Wait for a few seconds before sending a reply to the
19     // client. This could simulate some work the server is
20     // doing. While working, keep an eye on the context's
21     // `Done()` channel for a signal that we should cancel
22     // the work and return as soon as possible.
23     select {
24     case <-time.After(10 * time.Second):
25         fmt.Fprintf(w, "hello\n")
26     case <-ctx.Done():
27         // The context's `Err()` method returns an error
28         // that explains why the `Done()` channel was
29         // closed.
30         err := ctx.Err()
31         fmt.Println("server:", err)
32         internalError := http.StatusInternalServerError
33         http.Error(w, err.Error(), internalError)
34     }
35 }
```



```
ccat ee.go
File: ee.go

1 package main
2
3 import (
4     "fmt"
5     "net/http"
6     "time"
7 )
8
9 func hello(w http.ResponseWriter, req *http.Request) {
10
11     // A `context.Context` is created for each request by
12     // the `net/http` machinery, and is available with
13     // the `Context()` method.
14     ctx := req.Context()
15     fmt.Println("server: hello handler started")
16     defer fmt.Println("server: hello handler ended")
17
18     // Wait for a few seconds before sending a reply to the
19     // client. This could simulate some work the server is
20     // doing. While working, keep an eye on the context's
21     // `Done()` channel for a signal that we should cancel
22     // the work and return as soon as possible.
23     select {
24     case <-time.After(10 * time.Second):
25         fmt.Fprintf(w, "hello\n")
26     case <-ctx.Done():
27         // The context's `Err()` method returns an error
28         // that explains why the `Done()` channel was
29         // closed.
30         err := ctx.Err()
31         fmt.Println("server:", err)
32         internalError := http.StatusInternalServerError
33         http.Error(w, err.Error(), internalError)
34     }
35 }
```

diff

```
1 # shark
2
3 容器云管理系统
4
5 * k8s管理
6 * 机器初始化
7 * 监控采集 (node-exporter)
8 * hke管理
9
10 +- 5 lines: ## 架构图
11
12 * hke-master
13 主控节点，派发指令
14
15 * hke-syndic
16 中转节点，缓解 master 的压力
17
18 +- hke-minion
19 客户端，执行指令及上报信息
20
21
22 * rkhke-minion
23 客户端，执行指令及上报信息
24
25 ### Question
```

```
1 # shark
2
3 集群管理
4
5 ### 容器云管理系统
6
7 * k8s管理
8 * 机器初始化
9 * 监控采集 (node-exporter)
10 * hke管理
11
12 +- 5 lines: ## 架构图
13
14 * hke-master
15 主控节点，派发指令
16
17 * hke-syndic
18 中转节点，缓解 master 的压力
19
20 * rkhke-minion
21 客户端，执行指令及上报信息
22
23
24
25 ### Question
```

```
1 # master/master_server.go:6 # package master
2 import (
3     "context"
4     "io"
5     "io/ioutil"
6     "net"
7     "os"
8     "path"
9     "strings"
10    "sync"
11)
12
13 // GetStableMinionVersion
14 func (s *Server) GetStableMinionVersion(ctx context.Context, req *proto.MinionVersion) (*proto.MinionVersion, error) {
15     if conf.GetConfig().Master.StableMinionVersion == req.Version {
16         if upgrader.Version.String() == req.Version {
17             return nil, grace.ErrSameVersion
18         }
19
20         data, err := ioutil.ReadFile(path.Join(conf.MinionUpgradeReleasePath, req.Version))
21         if err != nil {
22             if os.IsNotExistError(err) {
23                 return nil, err
24             }
25             log.Errorf("get minion upgrade release list failed, err: %v", err)
26         }
27         // limit minion's upgrade concurrently
28         if err := upgrader.setLimiter(); err != nil {
29             return nil, err
30         }
31     }
32 }
```

vimdiff

icddiff

diff-so-fancy for git

vim plug

```
美化主题
Plug 'Yggdroot/indentline'                                " 美化代码缩进
Plug 'morhetz/gruvbox'                                    " colorscheme: gruvbox
Plug 'junegunn/seoul256.vim'                             " colorscheme: seoul256
Plug 'w0ng/vim-hybrid'                                    " colorscheme: hybrid
Plug 'joshdick/onedark.vim'                               " colorscheme: onedark
Plug 'srcery-colors/srcery-vim'                           " colorscheme: srcery
Plug 'rakr/vim-one'                                       " colorscheme: one
Plug 'altercation/vim-colors-solarized'                  " 需要手动安装
Plug 'jaxbot/semantic-highlight.vim'                      " ,h => 根据语义来代码高亮
Plug 'bling/vim-airline'                                  " 状态栏
Plug 'vim-airline/vim-airline-themes'                     " 同上
```

```
64 Plug 'terryma/vim-expand-region'                      " + 扩大选择, - 减少选择
65 Plug 'ntpeters/vim-better-whitespace'                " 空白
66 Plug 'fullybaked/toggle-numbers.vim'                 " 行号
67 Plug 'airblade/vim-gitgutter'                         " git状态
68 Plug 'lervag/vim-interestingwords'                   " ,k ,K => 高亮操作的字符串
69 Plug 'elzr/vim-json'                                  " json格式化
70 Plug 'ervandew/supertab'                            " 补全
71 Plug 'tpope/vim-surround'                           " 高效操作配对符号
72 Plug 'tpope/vim-repeat'                            " . => 重复surround操作
73 Plug 'jszakmeister/vim-togglecursor'                 " 搜索
74 Plug 'junegunn/fzf', { 'dir': '~/fzf', 'do': './install --all' } " vim启动菜单
75 Plug 'junegunn/fzf.vim'                             " ,t => 列出tag
76 Plug 'mhinz/vim-startify'                           " git命令
77 Plug 'majutsushi/tagbar'                            " git https://github.com/junegunn/gv.vim git commit br
78 Plug 'tpope/vim-fugitive'                           " 导航目录中看到git版本信息
79 Plug 'junegunn/gv.vim'                             " 文本对齐, 选定文本 :Tab/符号
80 Plug 'Xuyuap/nerdtree-git-plugin'                  " 自动匹配对称标记
81 Plug 'godlygeek/tabular'                           " 彩色括号
82 Plug 'jiangmiao/auto-pairs'                        " 代码模板
83 Plug 'kien/rainbow_parentheses.vim'                " 代码模板
84 Plug 'honza/vim-snippets'                          " 语法检测
85 Plug 'SirVer/ultisnips'                           " ,v => 目录树
86 Plug 'vim-syntastic/syntastic'                   " 目录树
87 Plug 'scrooloose/nerdtree'                         " 类似vscode command + p
88 Plug 'jistr/vim-nerdtree-tabs'                     " 异步编译及testing
89 Plug 'ctrlpvim/ctrlp.vim'                          " 快速移动插件, ',,w'可以在单词上出现前缀, 然后快速跳转
90 Plug 'tpope/vim-dispatch'                         " 支持regex的匹配查询
91 Plug 'easymotion/vim-easymotion'                  " 快速注释
92 Plug 'haya14busa/incsearch.vim'                   " 同上
93 Plug 'scrooloose/nerdcommenter'                   " for golang
94 Plug 'haya14busa/incsearch-fuzzy.vim'             " for python
95 Plug 'Valloric/YouCompleteMe'                      " s' }
96 Plug 'dgryski/vim-godef'                           " for python
97 Plug 'fatih/vim-go', { 'do': ':GoInstallBinaries' } " for python
98 Plug 'zchee/deoplete-jedi'
```

easy-motion

```
15 defaultName: "world"
16 → roundCount = 20000
17 →
18 → bunc sain() {
19 →   d
20 →   gaddr := hlag.ktring("laddr", "q27.w.e.r:t0051", "ynput uerver iddr")
21 →   olag.parse()
22 →   z
23 →   xall(*caddr)
24 → }
25 →
26 → bunc null(maddr ftring) {
27 →   jonn, ;ar := ;spc.;dal(
28 →     ;gdr,
29 →     ;hpc.;kthInsecure(),
30 →   )
31 →   ;l ;qr != ;wl {
32 →     ;eg.;rtal(";td ;yt ;unnect: %;i, ;or)
33 →   }
34 →   ;pfer ;znn.;xose()
35 →   ;c
36 → ;c
NORMAL ↵ master ➤ utf-8[unix] > +0 ~0 -0 > ~/go/src/github.com/rfyiamcool/grpcall/testing/dynamic_regi
Target key:
```

- go to jump
- <leader><leader>s
- <leader><leader>b
- <leader><leader>w



fzf.vim

```
28
29 type jsonRequestParser struct {
30     dec      *json.Decoder
31     unmarshaler jsonpb.Unmarshaler
32 }
33
34 // NewJSONRequestParser returns a RequestParser that reads data in JSON format
35 // from the given reader.
36 func NewJSONRequestParser(in io.Reader, resolver jsonpb.AnyResolver) RequestParser {
COMMAND +0 ~0 -0 ! master  format.go
invoke.go:213:4:// invokeAllStrem server and client are stream mode
invoke.go:195:20:    resp, err := stub.InvokeRpc(ctx, md, req, grpc.Trailer(&respTrailers), grpc.Header(&re
invoke.go:172:11:func (in *InvokeHandler) invokeUnary(ctx context.Context, stub grpcdynamic.Stub, md *desc.Met
invoke.go:170:12://    return invokeClientStream(ctx, stub, mtd, in.eventHandler, requestData, req)
invoke.go:165:13:        return in.invokeUnary(ctx, stub, mtd, in.eventHandler, requestData, req)
invoke.go:162:13:        return in.invokeServerStream(ctx, stub, mtd, in.eventHandler, requestData, req)
invoke.go:143:13:        return in.invokeAllStrem(ctx, stub, mtd, in.eventHandler, requestData, req, di
invoke.go:83:11:func (in *InvokeHandler) InvokeRPC(ctx context.Context, source DescriptorSource, ch grpcdynam
invoke.go:82:4:// InvokeRPC uses the given gRPC channel to invoke the given method.
invoke.go:76:10:    return &InvokeHandler{
invoke.go:75:9:func newInvokeHandler(event *EventHandler, inEvent InvocationEventHandler) *InvokeHandler {
invoke.go:70:6:type InvokeHandler struct {
- README.md:110:18:a protoset file, invoke `protoc` with the `*.proto` files that define the service:
38/38
```

- ⌚ command mode

- ⌚ Files:

- ⌚ GFiles:

- ⌚ Rg: (ripgrep)

- ⌚ Ag:

- ⌚ History:

- ⌚ ...



other vim

- [scrooloose/nerdcommenter](#)
 - 快速注释
- [haya14busa/incsearch.vim](#)
 - 支持正则的匹配 / 搜索
- [tpope/vim-surround](#)
 - 高效引号标记操作
- [lfv89/vim-interestingwords](#)
 - 单词高亮标记
- [brooth/far.vim](#)
 - 字符串替换

- 宏
- 录制重复操作
- 文本对象
- 更快的匹配
- 寄存器
- 多个临时存储
- map & noremap
- 键盘映射
- 标记
- 自定义vimscript



do you know ?

- `fmt.Println("golang vs rust")`
- `normal mode`
- `f ang`
- `%`
- `w`
- `5l`
- `s`
- `$`

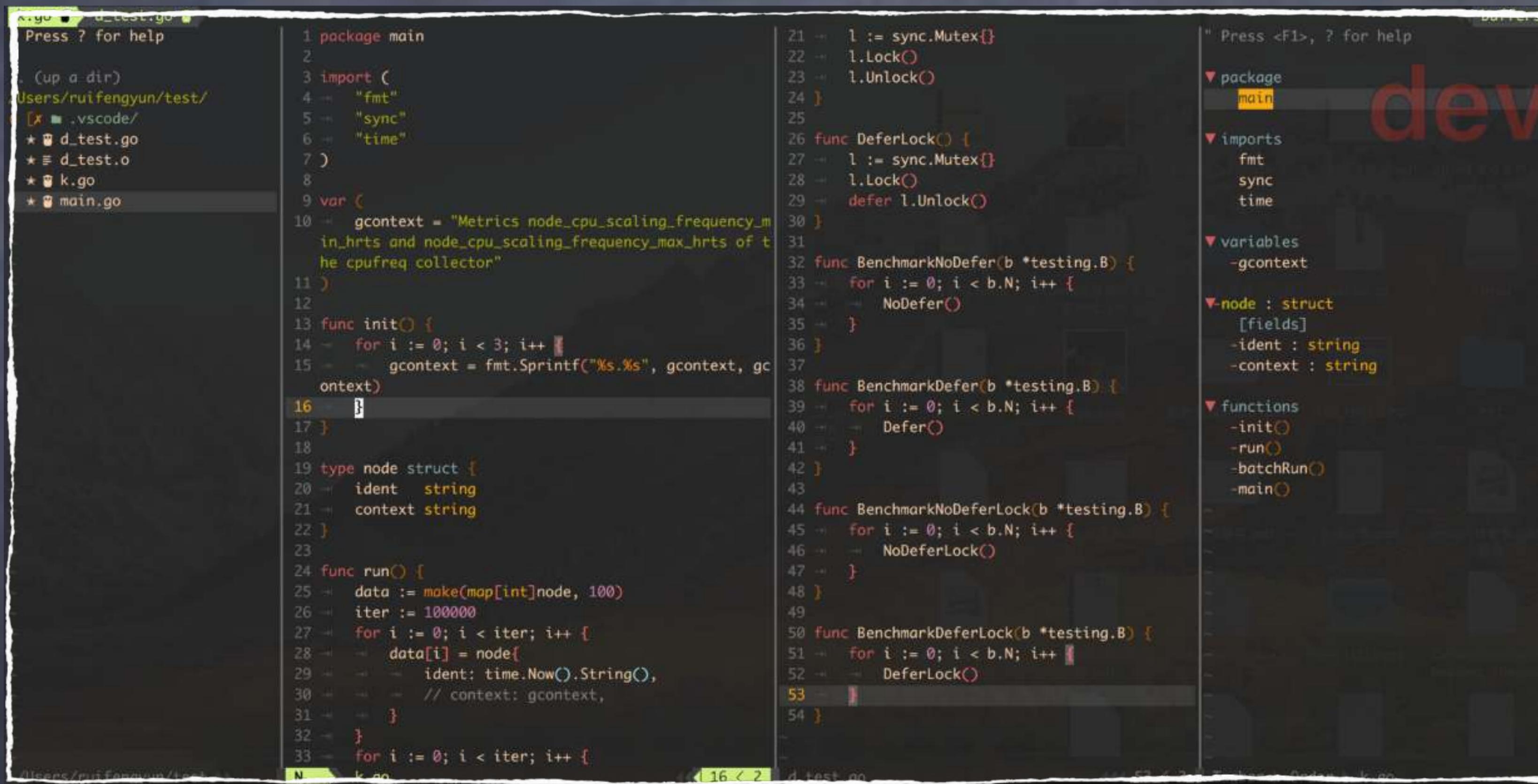


如回答不出来, 说明你的vim还要多练习 !

- `fmt.Println("golang vs rust")`
- `normal mode`
- `vi"`
- `va"`
- `di)`
- `de`



vim config



A screenshot of a Vim session with a dark theme. The main window displays a Go file named `d_test.go`. The sidebar on the right shows a navigation tree for the code, including a package named `main` and various functions and variables. The word `dev` is prominently displayed in red in the sidebar. The status bar at the bottom shows the file name `d_test.go` and the line number `16 < 2`.

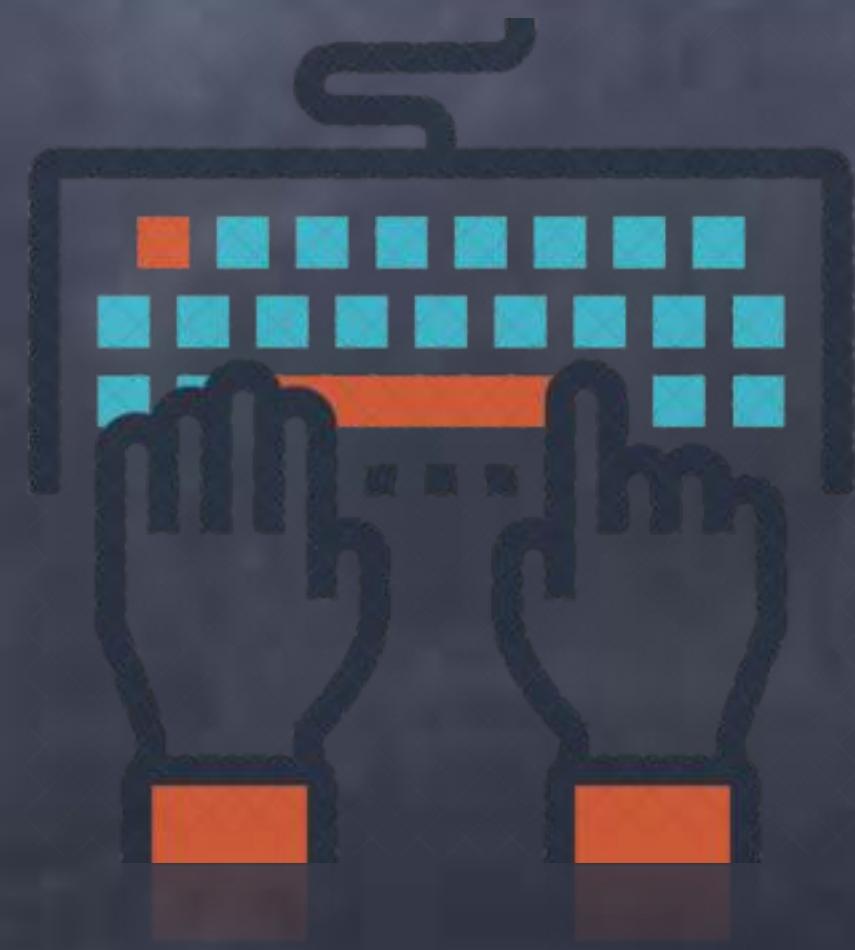
```
1 package main
2
3 import (
4     "fmt"
5     "sync"
6     "time"
7 )
8
9 var (
10     gcontext = "Metrics node_cpu_scaling_frequency_m
11     in_hrts and node_cpu_scaling_frequency_max_hrts of t
12     he cpufreq collector"
13 )
14
15 func init() {
16     for i := 0; i < 3; i++ {
17         gcontext = fmt.Sprintf("%s.%s", gcontext, gc
18         ontext)
19     }
20 }
21
22 type node struct {
23     ident string
24     context string
25 }
26
27 func run() {
28     data := make(map[int]node, 100)
29     iter := 100000
30     for i := 0; i < iter; i++ {
31         data[i] = node{
32             ident: time.Now().String(),
33             // context: gcontext,
34         }
35     }
36     for i := 0; i < iter; i++ {
37         l := sync.Mutex{}
38         l.Lock()
39         l.Unlock()
40     }
41 }
42
43 func BenchmarkNoDefer(b *testing.B) {
44     for i := 0; i < b.N; i++ {
45         NoDefer()
46     }
47 }
48
49 func BenchmarkDefer(b *testing.B) {
50     for i := 0; i < b.N; i++ {
51         Defer()
52     }
53 }
54
55 func BenchmarkNoDeferLock(b *testing.B) {
56     for i := 0; i < b.N; i++ {
57         l := sync.Mutex{}
58         l.Lock()
59         l.Unlock()
60     }
61 }
62
63 func BenchmarkDeferLock(b *testing.B) {
64     for i := 0; i < b.N; i++ {
65         DeferLock()
66     }
67 }
```

https://github.com/rfyiamcool/dev_config





强烈建议
在常用的IDE里装入vim插件



多屏幕焦点切换

- 同一个应用不同显示器跳转
 - command + `
- 不同显示器之间的焦点跳转
 - mac自动操作 + cliclick
 - alfred + cliclick 切换更多方位焦点



<https://github.com/rfyiamcool/multi-screen-jump>

tmux

split session

split windows

tmuxp

一键启动组合的tmux

neovim

veonim

oni



tmux.conf

```
1 # cat << EOF > /dev/null
2 # https://github.com/gpakosz/.tmux
3 # (-o-)
4 # dual licensed under the WTFPL v2 license and the MIT license,
5 # without any warranty.
6 # Copyright 2012- Gregory Pakosz (@gpakosz).
7 # ! do not edit this file
8 #
9 #
10 # -- general --
11
12 set -g default-terminal "screen-256color" # colors!
13 setw -g xterm-keys on
14 set -s escape-time 10 # faster command sequences
15 set -sg repeat-time 600 # increase repeat timeout
16 set -s focus-events on
17
18 set -g prefix2 C-a # GNU-Screen compatible prefix
19 bind C-a send-prefix -2
20
21 set -q -g status-utf8 on # expect UTF-8 (tmux < 2.2)
22 setw -q -g utf8 on
23
24 set -g history-limit 5000 # boost history
25
26 # edit configuration
27 bind e new-window -n "~/.tmux.conf.local" "sh -c '\${EDITOR:-vim} ~/.tmux.conf.local
28 && tmux source ~/.tmux.conf && tmux display \"~/.tmux.conf sourced\"'"
29
30 # reload configuration
31 bind r source-file ~/.tmux.conf \; display '~/.tmux.conf sourced'
32
33 # -- display --
34
```

tmux.conf

utf-8 2% 25/1228 1:1

51.0:100%

62.8% 37.2% 59.5% 37.2% 5549/8192MB 2128/3072MB

PID	USER	PRI	NI	VIRT	RES	SHR	S	CPU%	MEM%	TIME+	Command
1	root	0	0	0	0	0	?	0.0	0.0	0:00.00	(launched)
99207	ruifengy	0	0	0	0	0	?	0.0	0.0	0:00.00	- /System/Library/Priva
95868	ruifengy	0	0	0	0	0	?	0.0	0.0	0:00.00	- /System/Library/Sta
95862	ruifengy	0	0	0	0	0	?	0.0	0.0	0:00.00	- /System/Library/Sta
95859	ruifengy	0	0	0	0	0	?	0.0	0.0	0:00.00	- /System/Library/Sta
95858	ruifengy	0	0	0	0	0	?	0.0	0.0	0:00.00	- /Applications/Safari.o
85295	ruifengy	0	0	0	0	0	?	0.0	0.0	0:00.00	- /System/Library/Frame
85294	ruifengy	0	0	0	0	0	?	0.0	0.0	0:00.00	- /System/Library/Frame
83157	ruifengy	0	0	0	0	0	?	0.0	0.0	0:00.00	- /Applications/Magnet.o

F1Help F2Setup F3Search F4Invert F5Tree F6SortBy F7Nice -F8Nice +F9Kill F10Quit

```
://www.semrush.com/bot.html" "0.261"
- --> 85.208.96.66 - [02/Mar/2020:14:53:53 +0800] "GET /archives/tag/go-gctrace/feed HTT
P/1.1" 200 1699 "-" "Mozilla/5.0 (compatible; SemrushBot/1.0~bm; +http://www.semrush.com
/bot.html)" "0.244"
- --> 85.208.96.69 - [02/Mar/2020:14:54:01 +0800] "GET /archives/tag/python-%E6%8B%9B%E8
%81%98 HTTP/1.1" 200 8022 "-" "Mozilla/5.0 (compatible; SemrushBot/1.0~bm; +http://www.s
emrush.com/bot.html)" "0.382"
- --> 85.208.96.65 - [02/Mar/2020:14:54:11 +0800] "GET /archives/tag/%E5%B3%B0%E4%BA%91%
E5%80%81%E5%A5%89%E4%BA%86 HTTP/1.1" 200 8075 "-" "Mozilla/5.0 (compatible; SemrushBot/1
.0~bm; +http://www.semrush.com/bot.html)" "0.403"
- --> 220.181.108.82 - [02/Mar/2020:14:54:21 +0800] "GET /2016/02/18/%E5%85%83%E4%BA%8Epy
thon-peewee%E7%9A%84%E4%BA%8B%E5%8A%A1%E5%9B%9E%E6%BB%9A%E7%94%A8%E6%83%95/ HTTP/1.1" 3
02 161 "-" "Mozilla/5.0 (compatible; Baiduspider/2.0; +http://www.baidu.com/search/spide
r.html)" "0.014"
- --> 220.181.108.171 - [02/Mar/2020:14:54:21 +0800] "GET /archives/2779 HTTP/1.1" 200 1
5994 "-" "Mozilla/5.0 (compatible; Baiduspider/2.0; +http://www.baidu.com/search/spider.
html)" "0.517"
```

nvim -v

File: /Users/ruifengy/.config/nvim/init.vim

```
const scrollPis = shouldScrollOverflow(row)
const readjust = scrollPis ? scrollReadjustAmount() : 0
return {
  x: canvasBox0
  y: canvasBox0
  includes(searchElement: number, fromIndex?: number | undefined): boolean
  (1, 2, 3).includes(2, s)
  api.whereLine = row >> screenLeft
  const scrollPis = < setInterval
  const readjust = s < setImmediate: Amount() : 0
  >> sessionStorage
  container.scrollTo < setTimeout
  >> shouldScrollOverflow
  >> scrollReadjustAmount
  x: canvasBoxDime >> Set
  y: canvasBoxDime >> Symbol
  width: Math.cell >> Screen
  >> styleMedia
}

api.underline = (col, row, width, color) => {
  const cellBottom = canvasContainer.cell.height - canvasContainer.cell.padding
  const x = px.col.x(col)
  const y = px.row.y(row) + cellBottom
  const w = px.col.width(width)
  ui.beginPath()
```

File: /Users/ruifengy/.config/nvim/init.vim

```
let g:tasks = {
  'test': <>jobstart('npm test'),
  'start': <>jobstart('npm start'),
  'ls': <-->execute('Veonim devtools'),
  'term': <-->execute('Veonim TermOpen'),
  'term': <-->execute('Veonim TermAttach'),
  'term': <-->execute('Veonim TermDetach'),
  'screen': <-->execute('Veonim fullscreen')
```

fun! RunTask(name)
 if has_key(g:tasks, a:name)
 let Func = g:tasks[a:name]
 call Func()
 endif

init.vim

```
set splitbelow
set splithright
set noshowmode
set statustline
set nocursorline
set laststatus=2
set shiftwidth=2
set softtabstop=2
set shortmess+=Ic
set timeoutlen=10
set diffopt+=vertical
set inccommand=nosplit
set wildmode=list:full
set shell=/bin/bash\ --login
```

other bin

- **tig**

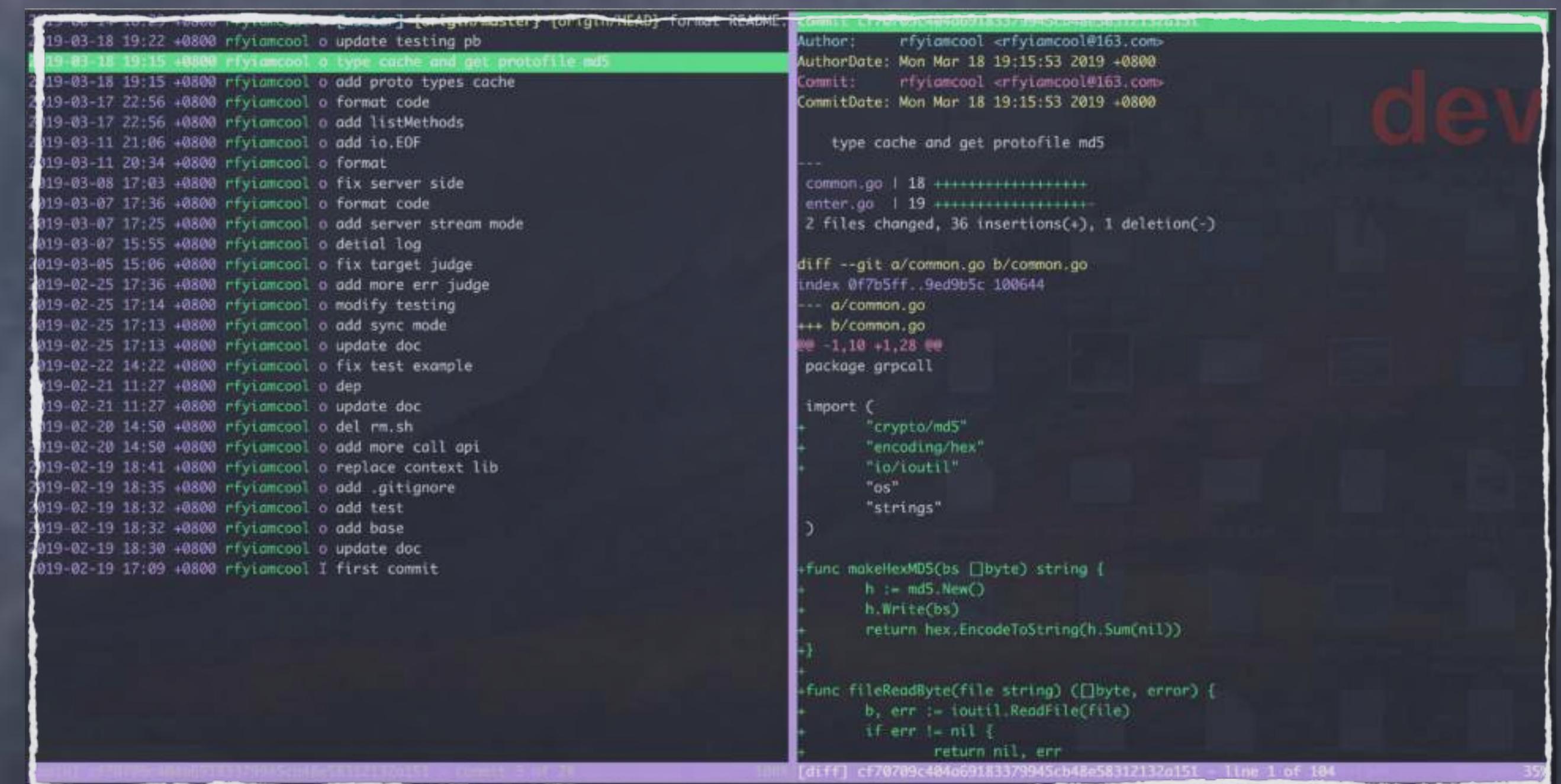
- 更直观的git浏览工具

- **pbcopy && pbpaste**

- 对接mac的粘贴板

- **tmuxp**

- **vscode -> code**



```
git log --oneline --since="2019-03-18" --until="2019-03-17" --format="format: %H %ad %s" --graph --color=always
Author: rfyiamcool <rfyiamcool@163.com>
AuthorDate: Mon Mar 18 19:15:53 2019 +0800
Commit: rfyiamcool <rfyiamcool@163.com>
CommitDate: Mon Mar 18 19:15:53 2019 +0800
type cache and get protofile md5
...
common.go | 18 ++++++-----+
enter.go | 19 ++++++-----+
2 files changed, 36 insertions(+), 1 deletion(-)

diff --git a/common.go b/common.go
index 0f75ff..9ed9b5c 100644
--- a/common.go
+++ b/common.go
@@ -1,18 +1,28 @@
 package grpcall

 import (
+	"crypto/md5"
+	"encoding/hex"
+	"io/ioutil"
+	"os"
+	"strings"
+)
+func makeHexMD5(bs []byte) string {
+	h := md5.New()
+	h.Write(bs)
+	return hex.EncodeToString(h.Sum(nil))
+}

+func fileReadByte(file string) ([]byte, error) {
+	b, err := ioutil.ReadFile(file)
+	if err != nil {
+		return nil, err
+}
+diff] cf20789c48466911337945cb48e583121320151 - Line 1 of 184 35
```

develop

- expect 自动化脚本
 - 校验 google 认证
 - ssh 登陆目标主机
- fswatch-run
 - 监听目录变化
 - 自动测试
 - 自动重启服务
 - 实时推送代码到开发服务器



```
function sync() {
    for host in $hosts
    do
        rsync -avz ${delete} --rsh="ssh -p${port}" --progress --exclude="*.log" ${exclude} ${src}
        root@${host}: ${dst}
    done
}

function execute() {
    for host in $hosts
    do
        ssh -o ConnectTimeout=1 -l root ${host} $1
        if [ ${?} -ne 0 ]; then
            exit 99
        fi
    done
}

function watch() {
    fswatch -l 1 -e ${src} |while read -d '' event; do
        now='get_time'
        echo "will sync: ${event} ${ts} ${now}"
        if [[ ${current} == ${now} ]]; then
            echo "filter...."
            continue
        fi
        current='get_time'
        sync
    done
}
```

develop

- ④ alias 各种命令别名
- ④ 自定义 snippet 模板
- ④ 请尽量多记使用软件的快捷键
- ④ 各种快捷键
- ④ 各种快捷键
- ④ 各种快捷键
- ④ 各种快捷键



```
42     "tmpl-workermodel": {
43         "prefix": "tmpl-workermodel",
44         "body": [
45             "type $1 struct {}\\n",
46             "func New$1 () *$1 {\\n return &$1{} }\\n",
47             "func (w *$1) Start() {\\n}\\n",
48             "func (w *$1) Stop() {\\n}\\n",
49             "func (w *$1) Restart() {\\n}\\n",
50             "func (w *$1) Reload() {\\n}\\n",
51             "func (w *$1) SetConfig() {\\n}\\n",
52         ],
53     },
54     "tmpl-ctx": {
55         "prefix": "tmpl-ctx",
56         "body": [
57             "ctx, cancel := context.WithCancel(context.Background())"
58         ],
59     },
60     "tmpl-select": {
61         "prefix": "tmpl-select",
62         "body": [
63             "select {}",
64             "\tcase <- $1:"
```

设备篇



- 键盘
- 桌子
- 椅子
- 显示器
- ...

键盘

① 机械键盘

- ② leopold 
- ② filco 
- ② ikbc 
- ② keychron 

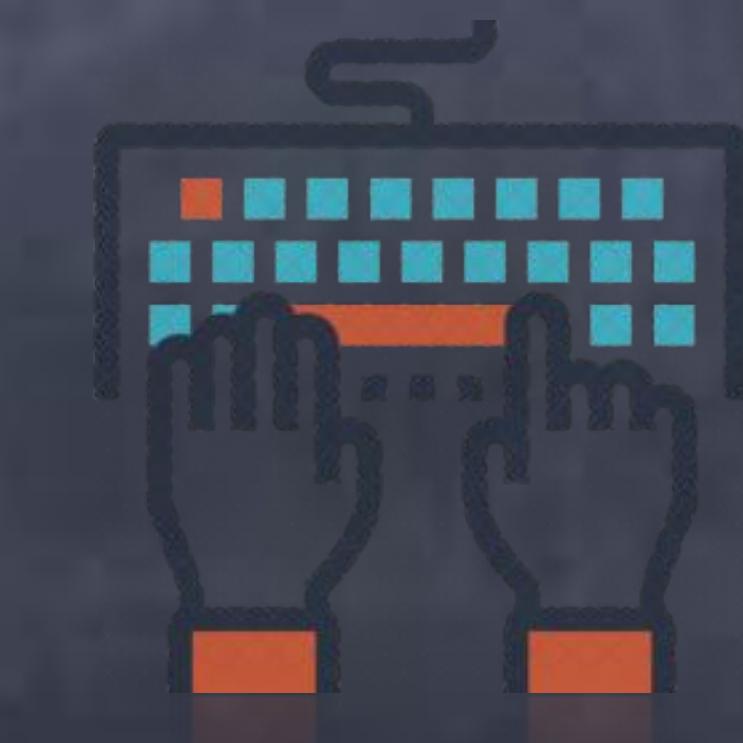


② 静电容

- ② hhkb 
- ② leopold 
- ② realforce 

建议 红轴

在办公室听到青轴声音后，莫名伤感 ...



键盘



椅子



◎ 西昊 m57

◎ ¥ 1000.00

◎ 冈村日本okamura

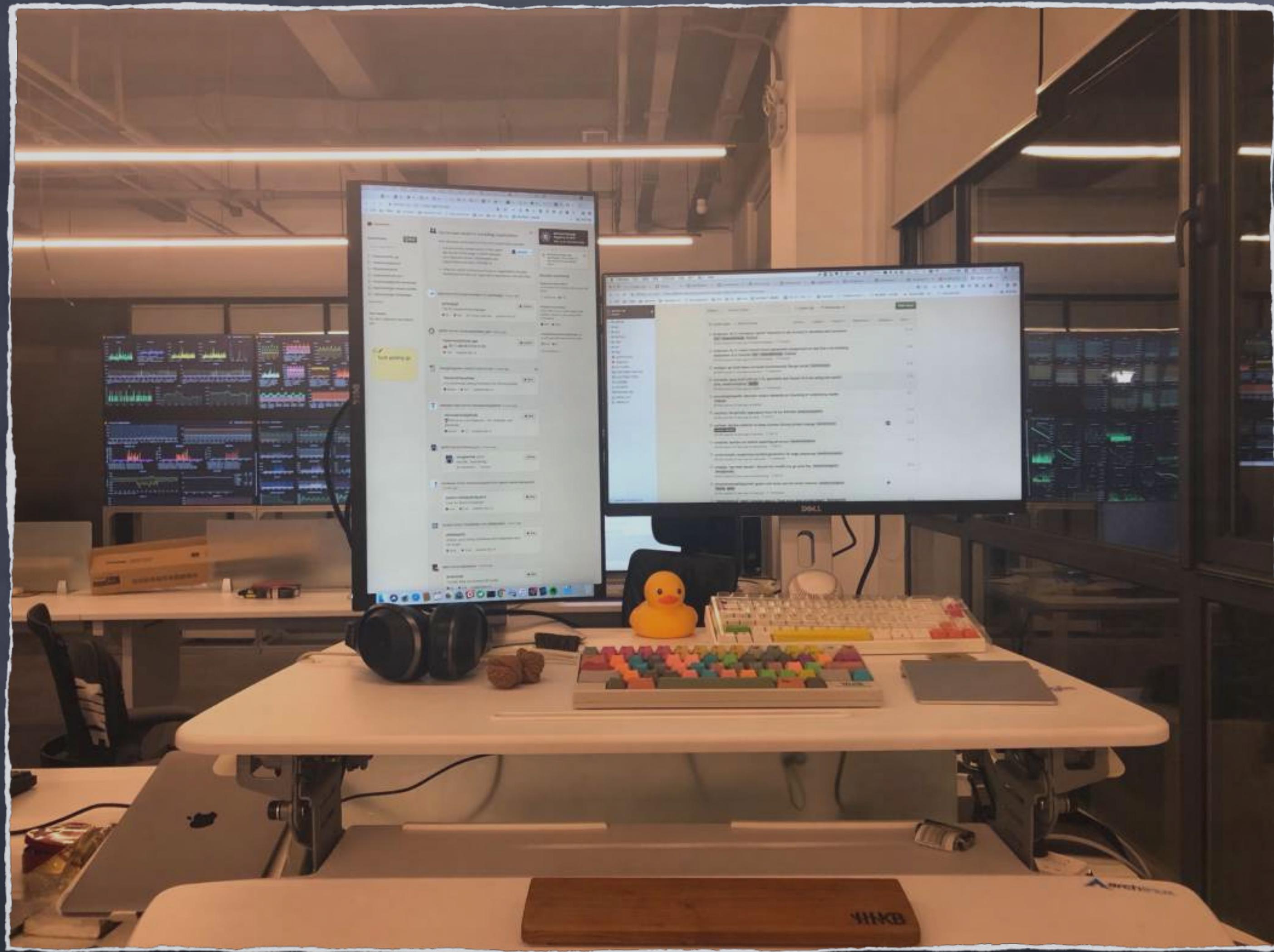
◎ sylphy

◎ ¥ 4000.00

◎ contessa

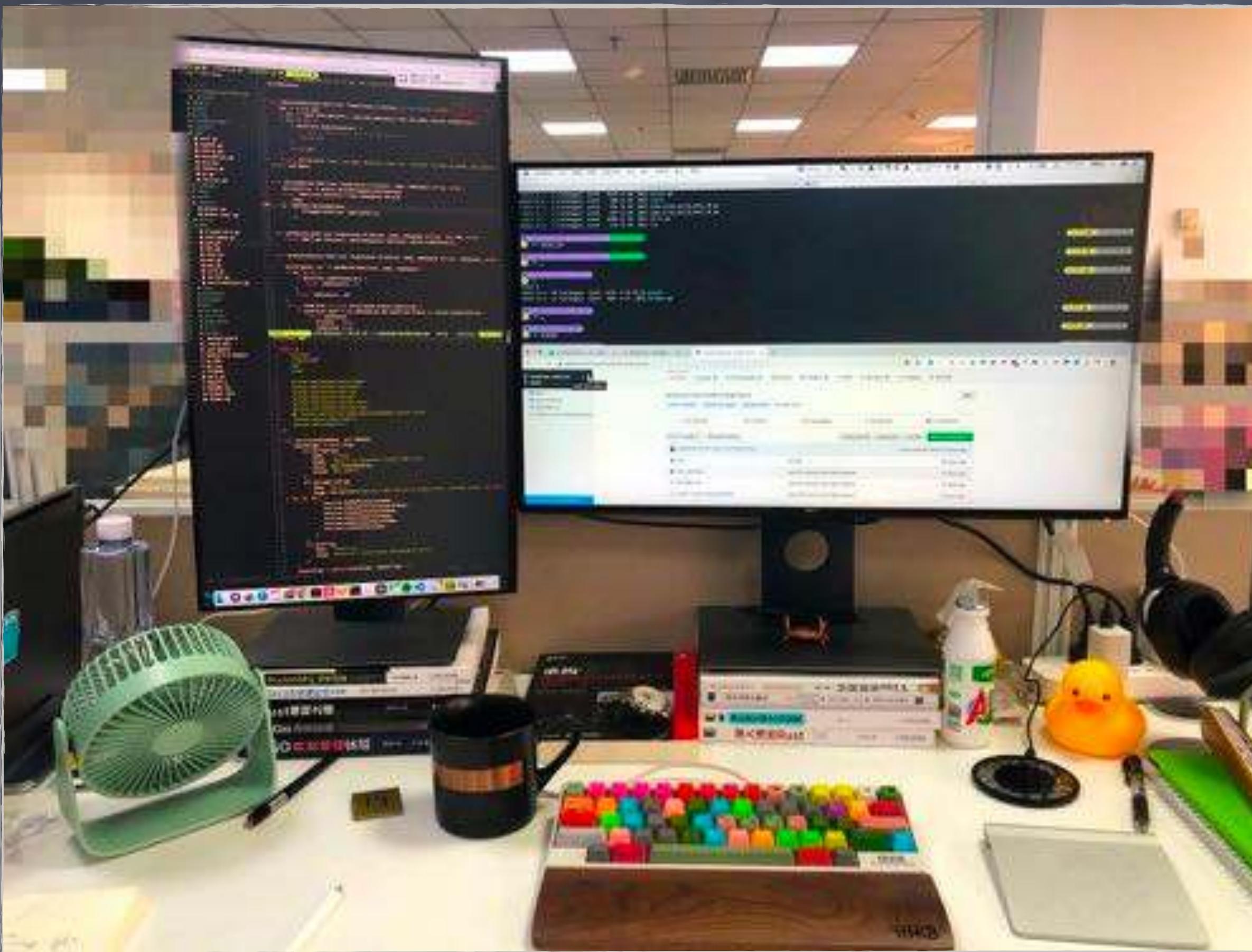
◎ ¥ 11000.00

◎ Herman Miller



- 对腰&前列腺
- 可拔高的桌子 (乐歌 m1m)
- 对眼
- 暖色及暗色主题
- 对手
- 核桃
- 棒球
- 对bug
- 按鸭子

显示器



- ④ 一横屏 (辅助)
- ④ 一竖屏 (主屏)
- ④ 强烈建议
- ④ 无需大角度摇头头部

• 4k

• LG 4k 27UL650

• DELL U2720Q

• Benq PD2700U

• more ...



终结篇：



- 在桌子上需摆放
- 深入理解linux内核
- tcp/ip 详解
- unix环境高级编程
- 图解电动车维修
- ...

“ Q&A ”

- xiaorui.cc



- github.com/rfyiamcool

