

**My dark plugins development  
history ~ over 10 years ~**

Shougo

# Note

"My presentation may be the last.

So I need to speak the development history."

# Profile

- **Shougo** = 暗黒美無王
- **“Dark Vim Master”**  
(pronunciation: **Uncock Vim Awe**)
- **Text editor is the world**

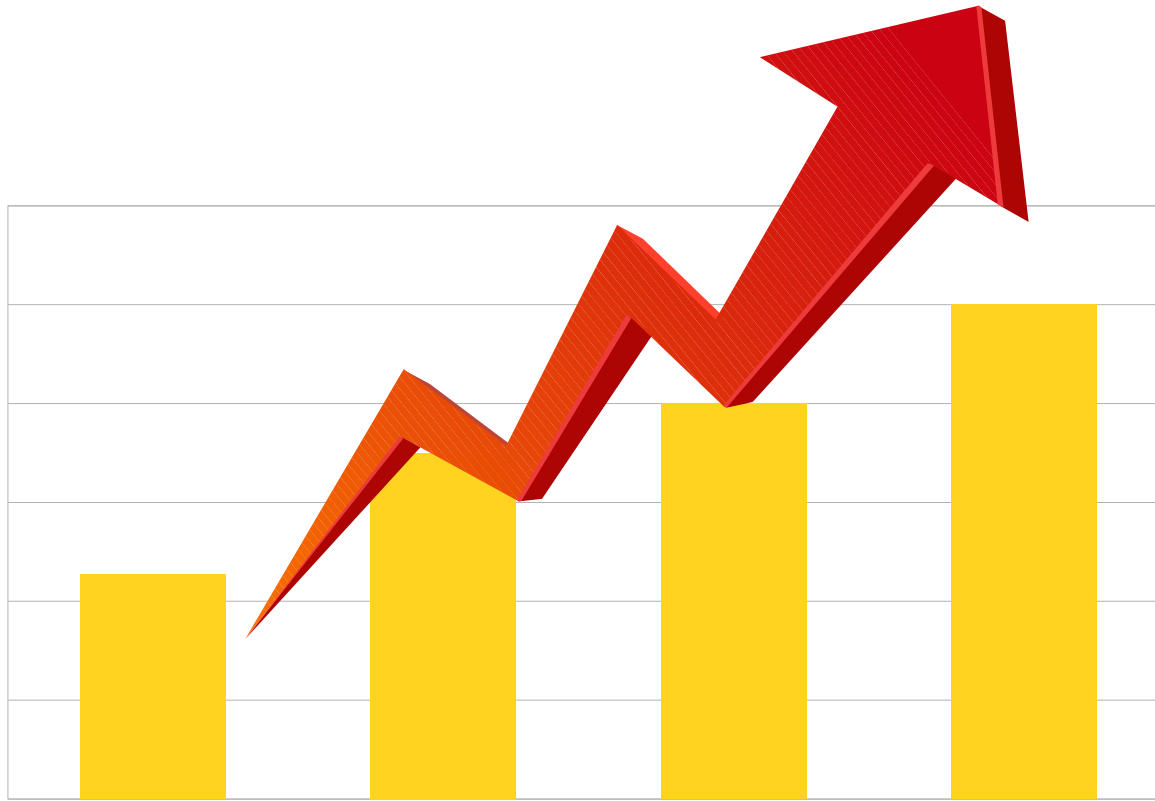


=



# VimConf 2019's theme







- “How to be more productive with Vim?”



# “more productive with Vim” is ...

- To develop dark plugins is my productive method!

Pinned Customize your pins

<p> <b>deoplete.nvim</b> <span style="float: right;">≡</span></p> <p>🚀 Dark powered asynchronous completion framework for neovim/Vim8</p> <p>● Python ★ 4.5k 🗨 257</p>	<p> <b>dein.vim</b> <span style="float: right;">≡</span></p> <p>⚡ Dark powered Vim/Neovim plugin manager</p> <p>● Vim script ★ 2.4k 🗨 137</p>
<p> <b>denite.nvim</b> <span style="float: right;">≡</span></p> <p>🌱 Dark powered asynchronous unite all interfaces for Neovim/Vim8</p> <p>● Python ★ 1.5k 🗨 196</p>	<p> <b>deol.nvim</b> <span style="float: right;">≡</span></p> <p>Dark powered shell interface for NeoVim and Vim8.</p> <p>● Vim script ★ 170 🗨 6</p>
<p> <b>deoppet.nvim</b> <span style="float: right;">≡</span></p> <p>The dark powered snippet plugin for neovim</p> <p>● Python ★ 141 🗨 1</p>	<p> <b>defx.nvim</b> <span style="float: right;">≡</span></p> <p>📁 The dark powered file explorer implementation</p> <p>● Python ★ 428 🗨 29</p>

# My theme is ...

- **My dark plugins development history over 10 years**
- **“Why does Shougo discard Vim script and choose Python3 instead?”**



# What is “dark plugins”?(1)

- I want to do everything in Vim(neovim) like Emacs
- It is not allowed by “:help design-not” in Vim

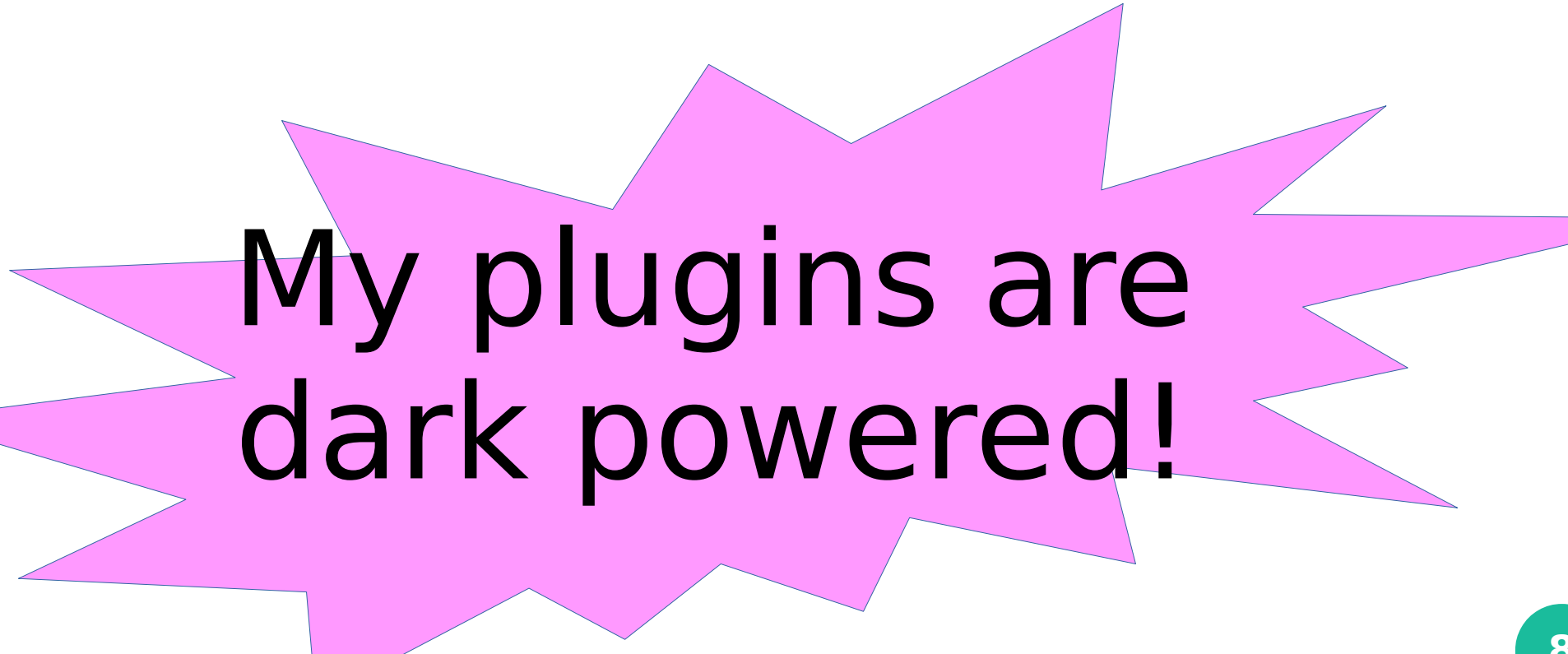
```
VIM IS... NOT
```

```
design-not
```

- ```
- Vim is not a shell or an Operating System. It does provide a terminal window, in which you can run a shell or debugger. E.g. to be able to do this over an ssh connection. But if you don't need a text editor with that it is out of scope (use something like screen or tmux instead).  
A satirical way to say this: "Unlike Emacs, Vim does not attempt to include everything but the kitchen sink, but some people say that you can clean one with it. ;-)"  
To use Vim with gdb see: http://www.agide.org and http://clewn.sf.net.  
- Vim is not a fancy GUI editor that tries to look nice at the cost of being less consistent over all platforms. But functional GUI features are welcomed.
```

# What is “dark plugins”?(2)

- **It is not the light way to use Vim**
- **I have defined it as the dark way**



**My plugins are  
dark powered!**



# Zero generation plugins

- **Autocomplpop.vim(acp.vim)**
- **FuzzyFinder.vim(fzf.vim)**
- **Shell.vim**

# Why did I use the plugins?

- **I used IDEs in Windows**
- **I wanted to use IDE features**

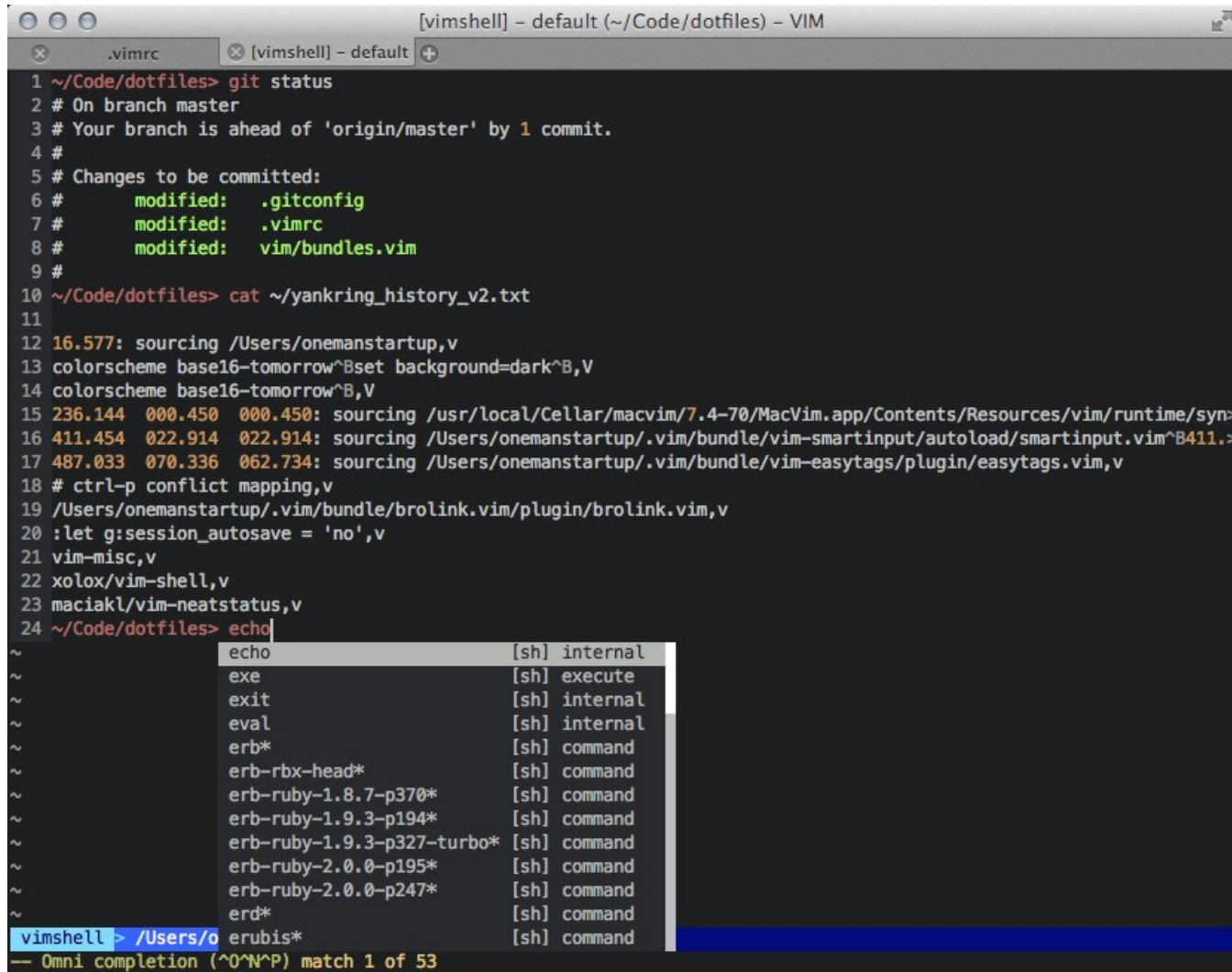
# Why did I quit to use zero generation plugins?

- **The plugins are not customizable fully**
- **Does not exist plugins for me**
- **I can create the plugins!**

# 1<sup>st</sup> generation plugins

- **vimshell.vim (2009)**
- **neocomplcache.vim (2009)**
- **neosnippet.vim (2012)**

# Vimshell.vim



```
[vimshell] - default (~/.Code/dotfiles) - VIM
.vimrc [vimshell] - default
1 ~/Code/dotfiles> git status
2 # On branch master
3 # Your branch is ahead of 'origin/master' by 1 commit.
4 #
5 # Changes to be committed:
6 #   modified:   .gitconfig
7 #   modified:   .vimrc
8 #   modified:   vim/bundles.vim
9 #
10 ~/Code/dotfiles> cat ~/yankring_history_v2.txt
11
12 16.577: sourcing /Users/onemanstartup,v
13 colorscheme base16-tomorrow^Bset background=dark^B,V
14 colorscheme base16-tomorrow^B,V
15 236.144 000.450 000.450: sourcing /usr/local/Cellar/macvim/7.4-70/MacVim.app/Contents/Resources/vim/runtime/syn>
16 411.454 022.914 022.914: sourcing /Users/onemanstartup/.vim/bundle/vim-smartinput/autoload/smartinput.vim^B411.>
17 487.033 070.336 062.734: sourcing /Users/onemanstartup/.vim/bundle/vim-easytags/plugin/easytags.vim,v
18 # ctrl-p conflict mapping,v
19 /Users/onemanstartup/.vim/bundle/brolink.vim/plugin/brolink.vim,v
20 :let g:session_autosave = 'no',v
21 vim-misc,v
22 xolox/vim-shell,v
23 maciakl/vim-neatstatus,v
24 ~/Code/dotfiles> echo
~ echo [sh] internal
~ exe [sh] execute
~ exit [sh] internal
~ eval [sh] internal
~ erb* [sh] command
~ erb-rbx-head* [sh] command
~ erb-ruby-1.8.7-p370* [sh] command
~ erb-ruby-1.9.3-p194* [sh] command
~ erb-ruby-1.9.3-p327-turbo* [sh] command
~ erb-ruby-2.0.0-p195* [sh] command
~ erb-ruby-2.0.0-p247* [sh] command
~ erd* [sh] command
~ erubis* [sh] command
vimshell > /Users/o
-- Omni completion (^O^N^P) match 1 of 53
```

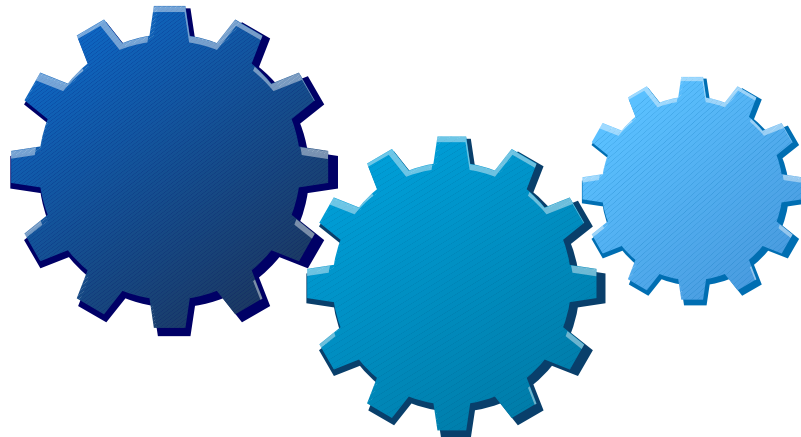
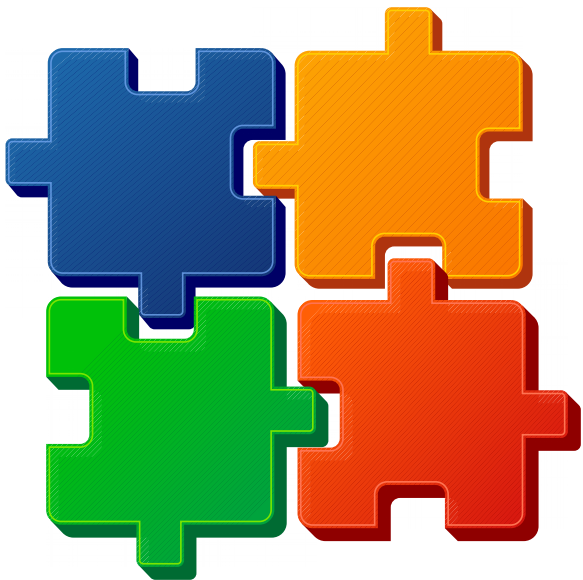
# Neocomplcache.vim

```
call neocomplcache#set_variable_pattern('g:neocomplcache_keyword_patterns', 'java',
    '\<\u\w*\%(\.\w*\%(())\?\)\?\)*\|[@]\?\h\w*\%(\s\?()\?\|<\)\?')
ca
cabbrev ; like "abbreviate" but for..and-line mode c [V] command , 'javascript,actions
cabclear ; clear all abbreviations ..and-line mode c [V] command
caddbuffer ; add errors from buffer c [V] command , 'awk',
caddexpr ; add errors from expr c [V] command
caddfile ; add error message to cur..quickfix list c [V] command , 'haskell,int-ghci',
call ; call a function c [V] command
catch ; part of a try command c [V] command , 'ml,ocaml,int-ocaml
Capture command ... c [V] command
call( [B] neocomplcache.v , 'erlang,int-erl',
cache [B] neocomplcache.v
case [B] neocomplcache.v , 'html,xhtml,xml,max
Caching [B] neocomplcache.v num:]-]*="\"%([\" ]*\"
Camel [B] neocomplcache.v , 'css',
cached [B] neocomplcache.v
call neocomplcache#set_variable_pattern('g:neocomplcache_keyword_patterns', 'tags',
    '\^[^|][^/[.blank:]*')
```



# 1<sup>st</sup> generation features

- **Custimizable**
- **Many functions**
- **Pluggable**





# 2<sup>nd</sup> generation plugins

- **unite.vim (2010)**
- **vimfiler.vim (2010)**
- **neobundle.vim (2011)**
- **neocomplete.vim (2013)**

# 2<sup>nd</sup> generation features

- **Use if\_lua**
- **Plugins combination**



**I choosed if\_lua because...**



**Vim script is  
too Slow!!!**

# Plugins combination examples

- **Unite.vim + vimfiler.vim**
- **Neocomplete.vim + neosnippet.vim**
- **Unite.vim + neobundle.vim**

# 3<sup>rd</sup> generation plugins

- **deoplete.nvim (2015)**
- **denite.nvim (2016)**
- **deol.nvim (2016)**
- **dein.vim (2015)**

# 3<sup>rd</sup> generation features

- **Use Vim8/neovim features**
- **Use Python3 instead**
- **Asynchronous**



# Use Vim8/neovim features

- **:terminal**
- **Job APIs**
- **Remote plugins**
- **winid**
- **And more...**
  
- **No more Vim 7!**

**I have chosen Python3 because...**



**Vim script is  
too Slow!!!**



# Python3 vs if\_lua

if\_lua



90%



10%

Python3



python™



# For example:



slow!

A screenshot of a search result for 'unite.vim slow'. The search bar at the top contains 'unite.vim slow'. Below the search bar, there are navigation links: 'すべて', '動画', 'ニュース', '画像', 'ショッピング', and 'もっと見る 設定'. The search results show approximately 1,080,000 items in 0.41 seconds. A hint suggests filtering results by Japanese language. The first search result is a GitHub issue titled 'file\_rec/async! still slow - Issue #890 - Shougo/unite.vim - GitHub'. The second result is another GitHub issue titled 'file\_rec/async! extremely slow - Issue #723 - Shougo/unite.vim ...'. Below these are two Reddit posts: 'What is your Unite trick? : vim - Reddit' and 'Unite is really slow compared to Ctrl-P : vim'. The date for the second Reddit post is 2015年11月19日. The date for the first Reddit post is 2015年11月19日. The date for the second Reddit post is 2018年8月2日. The date for the third Reddit post is 2016年6月12日. The date for the fourth Reddit post is 2015年9月30日. The search results are from www.reddit.com.

# 4<sup>th</sup> generation plugins

- **deoplete.nvim (2019, ver. 5.1~)**
- **denite.nvim (2019, ver. 3.0~)**
- **deoppet.nvim (2017)**
- **defx.nvim (2018)**

# 4<sup>th</sup> generation features

- **Type definitions**
- **Non default key-mappings**
- **Remove configuration variables**

mappings



variables



# Why is type definition important?

- **It can detect some bugs**
- **Detect bugs is important**

# Why non default key-mappings?(1)

<https://github.com/Shougo/defx.nvim/issues/5>

## Default key mappings #5

 Closed kutsan opened this issue on 2 Feb 2018 · 11 comments



kutsan commented on 2 Feb 2018 · edited ▾



I am sorry Shougo but I would like to say your default key mappings of choices were horrible for vimfiler. I know you can change it but it's not a great experience for first-time users. Like, who would have thought to press `Shift + K` to [make a new directory](#)? There is no mnemonic and it's not intuitional. So, if you don't mind, can we discuss this before its implementation? They should be easy to remember and well-thought. For example, `cf`, `[c]reate [f]ile` or `cd`, `[c]reate [d]irectory` something like that. Although this can be more intuitive with `dd`, `cut file or directory under cursor` and `yy`, `copy file or directory under cursor`.

# Why non default key-mappings?(2)

- **No best default key mappings**
- **Hard to define defaults**
- **Conflict with other plugins/config**

**Default mappings  
are not needed!**

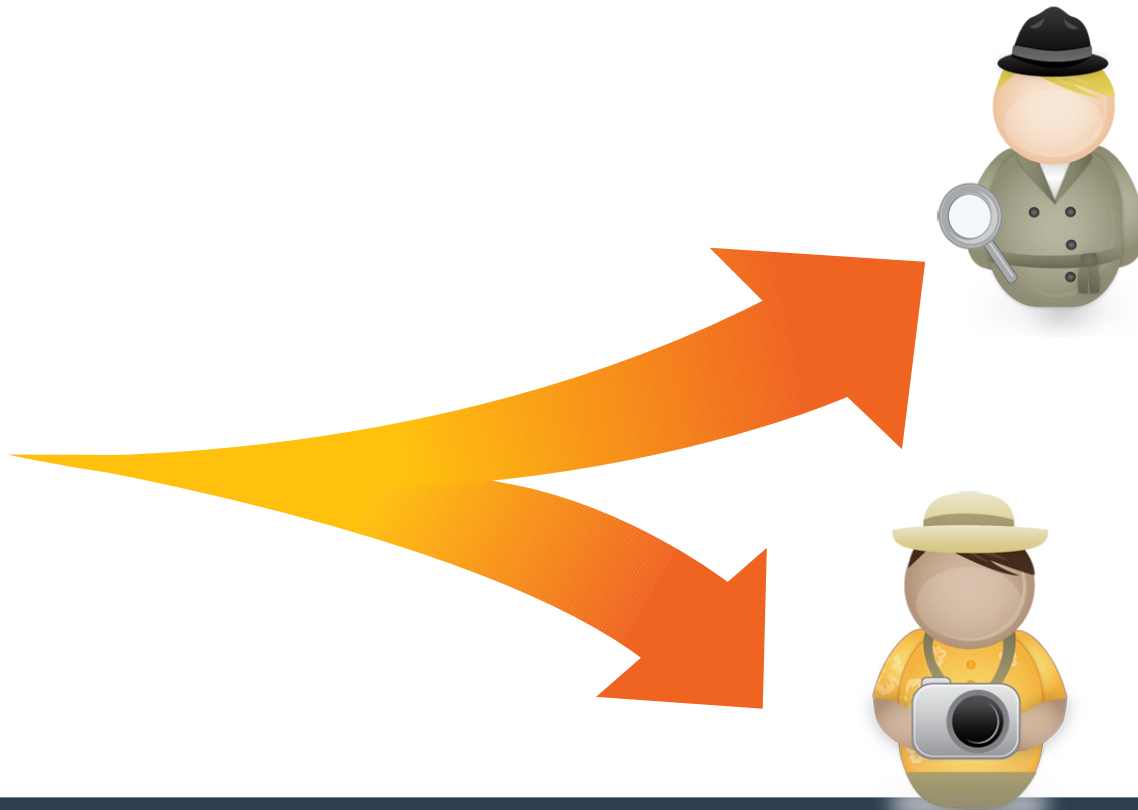
# Why I have removed configuration variables?

- **Functions can keep interfaces in multiple plugins**
- **Example:**
  - call `defx#custom#option('_', {})`**
  - call `denite#custom#option('_', {})`**



# Let's create your plugins!

- **This is just my history.**
- **You can choose other selection!**



# Your Vim is your text editor!



# The extra: denite.nvim version 3

Shougo

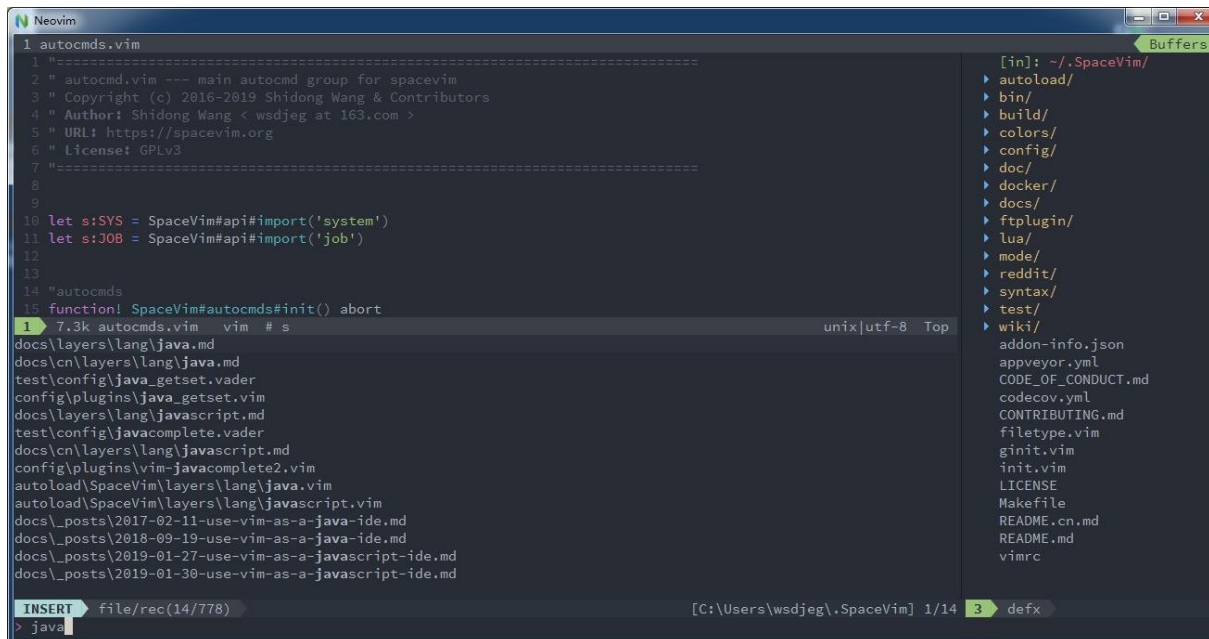
# Denite.nvim big changes

- Many issues are reported after upgrade

The screenshot shows a GitHub pull request for the repository `Shougo / denite.nvim`. The pull request is titled "[RFC] Replace UI #624" and is in a "Merged" state. It was merged by Shougo on 1 Jun, merging 144 commits from the `ui` branch into the `master` branch. The repository has 46 unwatchers, 1,483 stars, and 196 forks. The pull request has 197 conversations, 144 commits, 0 checks, and 65 files changed. The pull request is currently closed. The main comment from Shougo, dated 24 Apr, mentions "Fix #509 #625" and has 4 likes. A comment from Milly, dated 26 Apr, references this pull request and mentions "Feature request: Customize default-action per kind. #628". Another comment from Shougo, dated 6 May, mentions "force-pushed the ui branch from 5bc367e to 1de25c3 on 6 May" and asks "Please test the branch.". A comment from Shougo, dated 6 May, references this pull request and mentions "WIP: basic coroutine and subprocess support #393". The right sidebar shows settings for reviewers, assignees, labels, projects, milestones, and notifications.

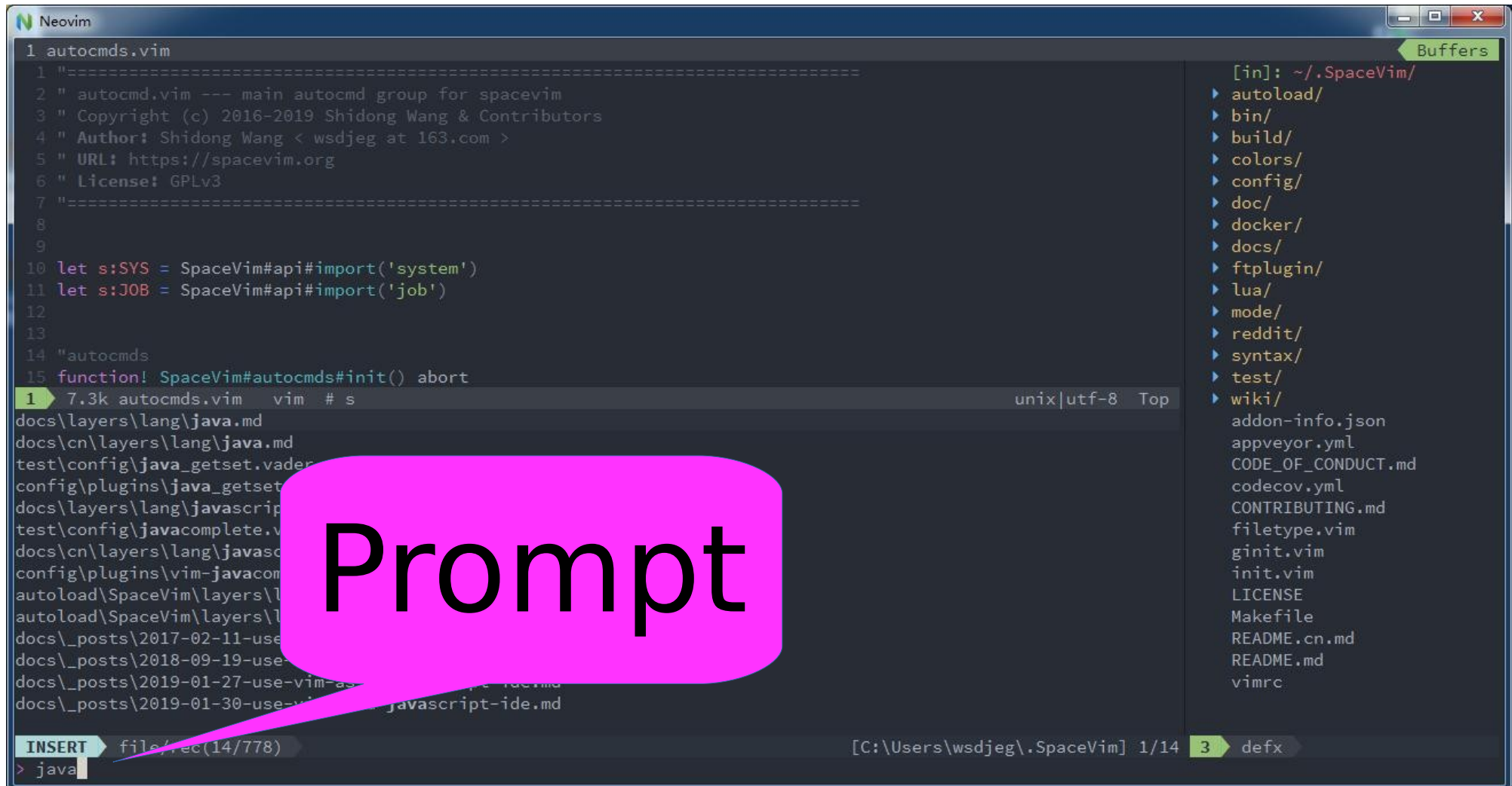
# Denite.nvim version 2

- **Steal your input**
- **Define original mappings**
- **Similar to ctrlp.vim**



The screenshot shows the Neovim editor interface. The main window displays the contents of the file `autocmds.vim`. The file contains several lines of Vim script, including comments and function definitions. The current line is `15 function! SpaceVim#autocmds#init() abort`. The status bar at the bottom indicates the current file is `file/rec(14/778)` and the cursor is at line 3, column 3, with the text `defx` under the cursor. On the right side, there is a file explorer window titled "Buffers" showing a directory tree for `~/SpaceVim/`. The tree includes folders like `autoload/`, `bin/`, `build/`, `colors/`, `config/`, `doc/`, `docker/`, `docs/`, `ftplugin/`, `lua/`, `mode/`, `reddit/`, `syntax/`, `test/`, and `wiki/`, along with various files like `addon-info.json`, `appveyor.yml`, `CODE_OF_CONDUCT.md`, `codecov.yml`, `CONTRIBUTING.md`, `filetype.vim`, `ginit.vim`, `init.vim`, `LICENSE`, `Makefile`, `README.cn.md`, `README.md`, and `vimrc`.

# Denite.nvim version 2



```
1 autocmds.vim
2 " =====
3 " autocmd.vim --- main autocmd group for spacevim
4 " Copyright (c) 2016-2019 Shidong Wang & Contributors
5 " Author: Shidong Wang < wsdjeg at 163.com >
6 " URL: https://spacevim.org
7 " License: GPLv3
8 " =====
9
10 let s:SYS = SpaceVim#api#import('system')
11 let s:JOB = SpaceVim#api#import('job')
12
13
14 "autocmds
15 function! SpaceVim#autocmds#init() abort
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000
```

7.3k autocmds.vim vim # s unix|utf-8 Top

docs\layers\lang\java.md  
docs\cn\layers\lang\java.md  
test\config\java\_getset.vader  
config\plugins\java\_getset  
docs\layers\lang\javascript  
test\config\javacomplete.v  
docs\cn\layers\lang\javasc  
config\plugins\vim-javacom  
autoload\SpaceVim\layers\  
autoload\SpaceVim\layers\  
docs\\_posts\2017-02-11-use  
docs\\_posts\2018-09-19-use  
docs\\_posts\2019-01-27-use-vim-as  
docs\\_posts\2019-01-30-use-vim-javascript-ide.md

INSERT file:///ec(14/778) [C:\Users\wsdjeg\.SpaceVim] 1/14 3 defx

> java

Buffers

- [in]: ~/.SpaceVim/
- ▶ autoload/
- ▶ bin/
- ▶ build/
- ▶ colors/
- ▶ config/
- ▶ doc/
- ▶ docker/
- ▶ docs/
- ▶ ftplugin/
- ▶ lua/
- ▶ mode/
- ▶ reddit/
- ▶ syntax/
- ▶ test/
- ▶ wiki/

addon-info.json  
appveyor.yml  
CODE\_OF\_CONDUCT.md  
codecov.yml  
CONTRIBUTING.md  
filetype.vim  
ginit.vim  
init.vim  
LICENSE  
Makefile  
README.cn.md  
README.md  
vimrc

Prompt

# Prompt UI's disadvantages

- **Ignore user's configuration**
- **Hard to work asynchronously**
- **Cannot combine with other plugins**
  
- **They fit with my policy???**

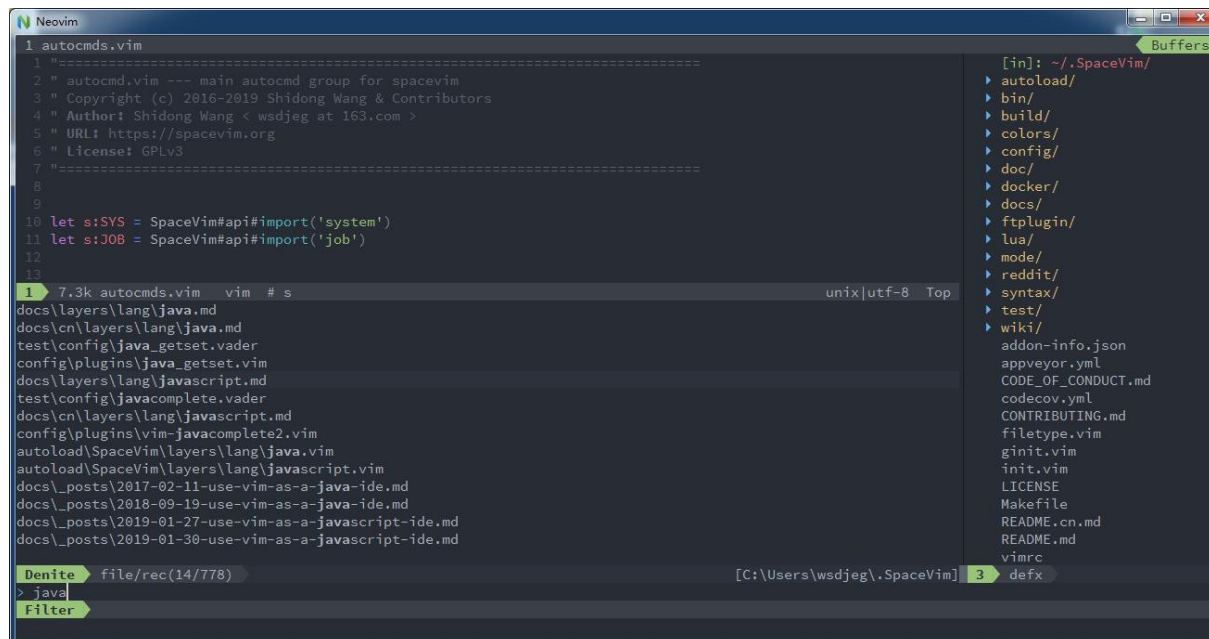
# My plugin development policy

- **Customizable**
- **Combine with other plugins**
- **Asynchronous if possible**



# Denite.nvim version 3

- Replace prompt.nvim to normal buffer
- Edit buffer instead of command line



```
1 autocmds.vim
2 "-----
3 " autocmd.vim --- main autocmd group for spacevim
4 " Copyright (c) 2016-2019 Shidong Wang & Contributors
5 " Author: Shidong Wang <wsdjeg at 163.com >
6 " URL: https://spacevim.org
7 " License: GPLv3
8 "-----
9
10 let s:SYS = SpaceVim#api#import('system')
11 let s:JOB = SpaceVim#api#import('job')
12
13
14 7.3k autocmds.vim vim # s unix|utf-8 Top
docs\layers\lang\java.md
docs\cn\layers\lang\java.md
test\config\java_getset.vader
config\plugins\java_getset.vim
docs\layers\lang\javascript.md
test\config\javacomplete.vader
docs\cn\layers\lang\javascript.md
config\plugins\vim-javacomplete2.vim
autoload\SpaceVim\layers\lang\java.vim
autoload\SpaceVim\layers\lang\javascript.vim
docs\_posts\2017-02-11-use-vim-as-a-java-ide.md
docs\_posts\2018-09-19-use-vim-as-a-java-ide.md
docs\_posts\2019-01-27-use-vim-as-a-javascript-ide.md
docs\_posts\2019-01-30-use-vim-as-a-javascript-ide.md

Denite file/rec(14/778) [C:\Users\wsdjeg\.SpaceVim] 3 defx
> java
Filter
```

# Denite.nvim version 3

```
1 autocmds.vim
1 "-----
2 " autocmd.vim --- main autocmd group for spacevim
3 " Copyright (c) 2016-2019 Shidong Wang & Contributors
4 " Author: Shidong Wang < wsdjeg at 163.com >
5 " URL: https://spacevim.org
6 " License: GPLv3
7 "-----
8
9
10 let s:SYS = SpaceVim#api#import('system')
11 let s:JOB = SpaceVim#api#import('job')
12
13
1 7.3k autocmds.vim vim #
docs\layers\lang\java.md
docs\cn\layers\lang\java.md
test\config\java_getset.vader
config\plugins\java_getset.vim
docs\layers\lang\javascript.md
test\config\javacomplete.vader
docs\cn\layers\lang\javascript.md
config\plugins\vim-javacomplete2.vim
autoload\SpaceVim\layers\lang\java.vim
autoload\SpaceVim\layers\lang\javascript
docs\_posts\2017-02-11-use-vim-as-a-java
docs\_posts\2018-09-19-use-vim-as-a-java
docs\_posts\2019-01-27-use-vim-as-a-i
docs\_posts\2019-01-30-use-vim-as

Denite file/rec(14/778)
> java
Filter
```

Normal buffer

Edit buffer

# New UI advantages

- **Use user's configuration**
- **Easy to work asynchronously**
- **Can combine with other plugins**
  
- **The features fit with my policy!**

# Denite.nvim version 3 Demo

# The extra2: My development note

Shougo

# My development style

- <https://employment.en-japan.com/engineerhub/entry/2018/08/28/110000>

エンジニアHub Powered by En エン転職

若手Webエンジニアのための情報メディア

エン転職TOP > エンジニアHub > deoplete.nvim開発者、Shougoが考える「Vimの限界ギリギリ」

2018-08-28

## deoplete.nvim開発者、Shougoが考える「Vimの限界ギリギリ」を攻めるプラグイン開発の裏側

多くのエンジニアに愛されるテキストエディタ『Vim』のプラグインを作り続けるShougoさん。Vimmerならば一度は使ったことがある、というほどに広く知られたプラグインを作り続ける理由を聞きました。

418

B!ブックマーク

35

シェア

ツイート



「あなたにとって、テキストエディタとは何ですか？」

こう聞かれたとき、あなたは何と答えるでしょうか。開発に不可欠な大事なツール、あるいは、ただの道具——。きっと十人十色の回答が返ってくるでしょう。

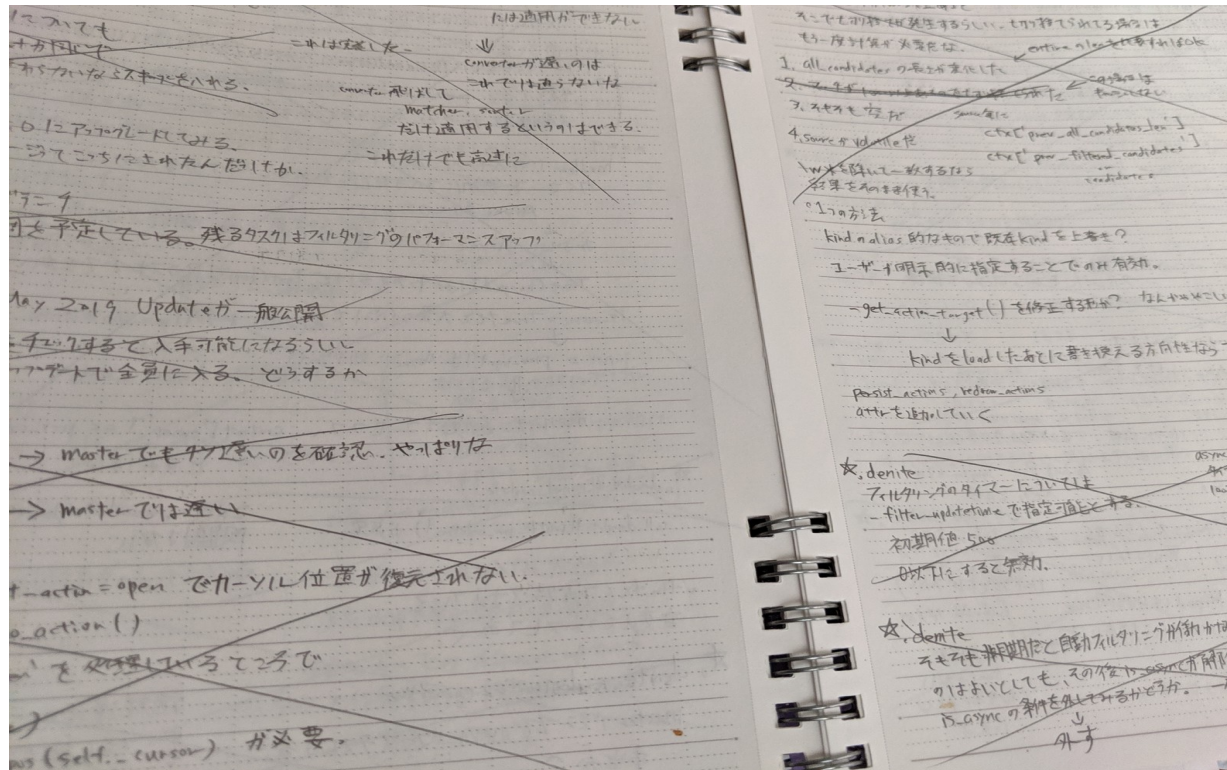
この問いに対し、「テキストエディタとは世界」と迷いなく答えるエンジニアがいま

# I don't use memo plugins!

- **Memolist.vim**
- **QfixHowm**
- **Vimwiki**
- **...**

# I use paper note for development

- Here





# I use the paper note for...

- **(Finished)task list**
- **Technology notes**
- **Survey for issues**

# Why do I use paper note?

- **Not needed PC!**
- **Free style**



# The note is the strongest Vim for me!

- **Thank you!**

# End

- **Any questions?**