

Hacking Vim to Pieces with Embedded Ruby

@cpatuzzo — **VimLondon** — 2013 - 03 - 26

I LOVE

VIM

efficient

customizable

extensible

powerful

open-source

sociable

bewildering

I LOVE

RUBY

expressive

flexible

mature

simplistic

beautiful

standard libraries

third-party libraries

How can we
bridge the gap?

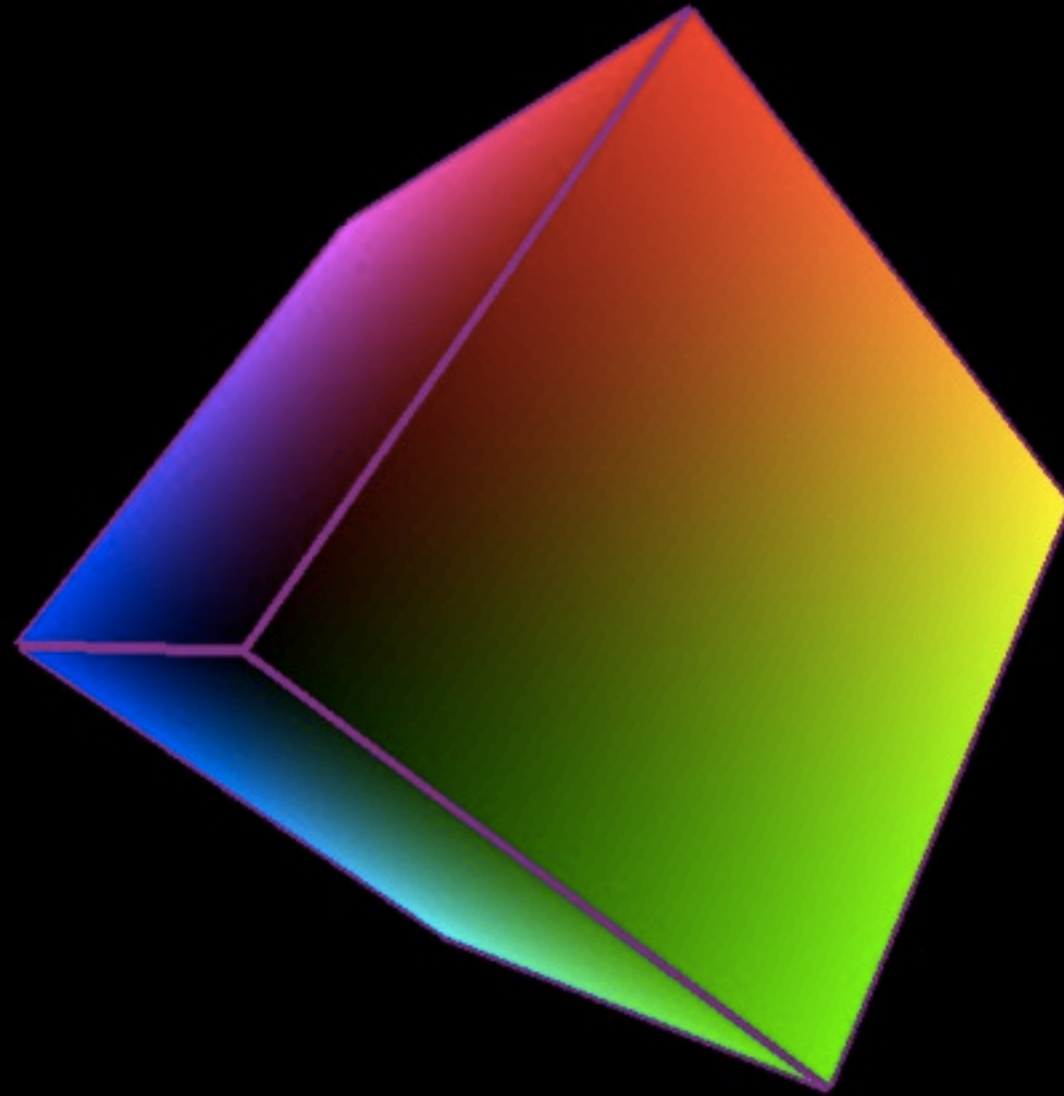
... and what
happens when
we do?

Some really bad things...

```
17 vim 0x00000001013299a8 rb_protect + 232 18 vim 0x00000001012d
f4 ex_ruby + 66 19 vim 0x00000001011ee1ba do_cmdline
5210 20 vim 0x0000000101253eda nv_colon + 173
21 vim 0x000000010124e33d normal_cmd + 3976
22 vim 0x00000001012e438c main_loop + 1130
23 vim 0x00000001012e396e main + 8294
24 libdyld.dylib 0x00007fff93c747e1 start + 0 25 ??? 0x0000000000000000
02 0x0 + 2
)
2013-03-15 14:50:47.214 vim[79356:707] _createMenuRef called with existing pr
cipal MenuRef already associated with menu
2013-03-15 14:50:47.216 vim[79356:707] (
0 CoreFoundation 0x00007fff8edd70a6 __exceptionPreprocess + 198
1 libobjc.A.dylib 0x00007fff8ce093f0 objc_exception_throw + 43
2 CoreFoundation 0x00007fff8edd6e7c +[NSException raise:format:] + 204
3 AppKit 0x00007fff90a14eac -[NSCarbonMenuImpl _createM
uRef] + 62
4 AppKit 0x00007fff90a147fb -[NSCarbonM
uImpl _instantiateCarbonMenu] + 143
5 AppKit 0x00007fff90
3743 -[NSApplication finishLaunching] + 778
6 GLUT 0x0000000101b2219d -[GLUTApplication run] + 114
+ 18469 12 vim 0x0000000101459945 vm_exec_cor
13 vim 0x000000010145e573 vm_exec + 1491
14 vim 0x000000010145ed92 eval_string_with_cref + 866
15 vim 0x00000001013299a8 rb_protect + 232
16 vim 0x00000001012dabf4 ex_ruby + 66
17 vim 0x00000001011ee1ba do_cmdline + 6210
18 vim 0x0000000101253eda nv_colon + 173
19 vim 0x0000000101
e33d normal_cmd + 3976
20 vim 0x00000001012e438c mai
loop + 1130
21 vim 0x00000001012e396e main + 8294
```


Some awesome things...

```
1 red
2 green
3 blue
4 yellow
5 white
6 black
```



```
demo [utf-8,unix]
[:ruby require 'scene'; Scene.display
```


EXPERIMENTAL



THIS IS NOT:

> proven to work

> software engineering advice

THIS IS:

- > good fun for hackers
- > mostly unexplored

**It's worth
mentioning...**

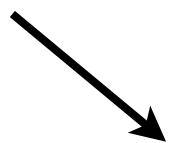


```
:python print "Sup, VimLondon?"
```

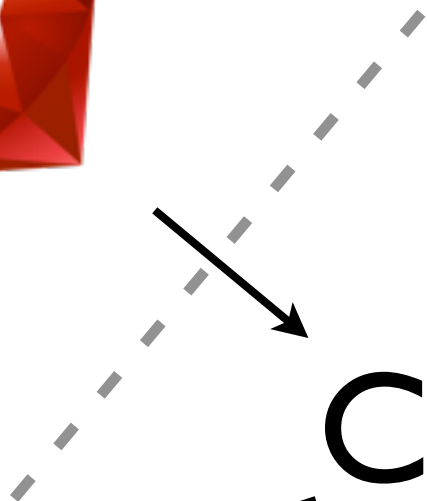
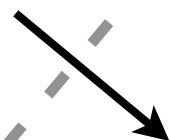
So let's get
started...

github.com/tuzz/
vim-experiments

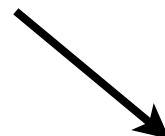
Diagrams



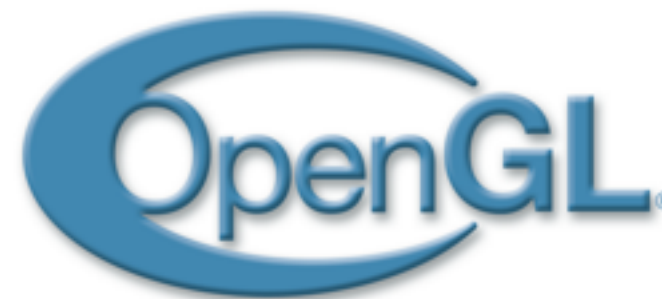
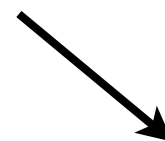
Detached



Child



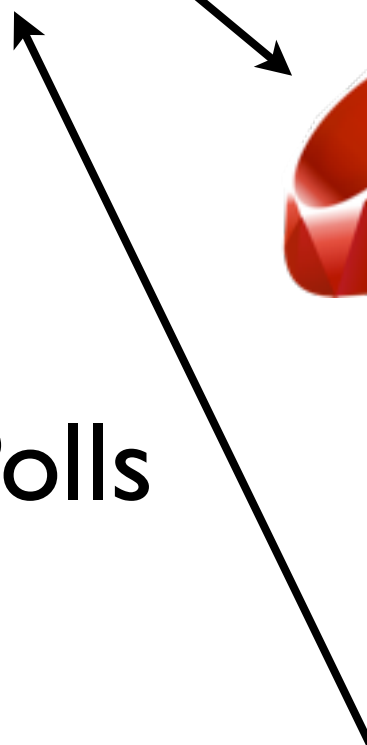
C



OpenGL
Experiment

(simplified)

Polls

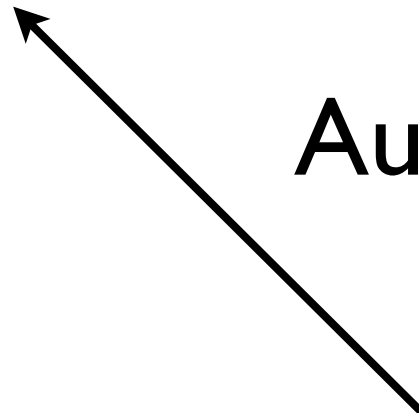


Remote
Control



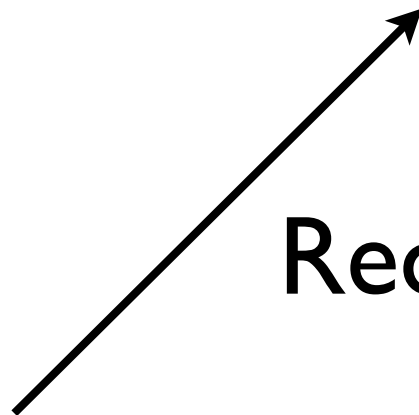
Server

Autocmd



vimremote

Request



(simplified)

THANK
YOU

@cpatuzzo

chris@patuzzo.co.uk

<https://github.com/tuzz/vim-experiments>

Questions?