

Fifty Tips, Tricks & Tools

...in one talk

Sean Coates Fictive Kin
gimmebar.com

“Hey, did
you know that you
could use
{some tool}
to do that better?”



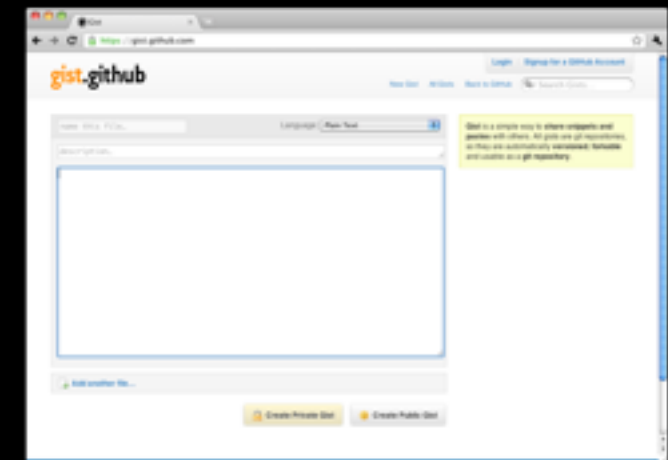


Photo: <http://flic.kr/p/82aL6y>



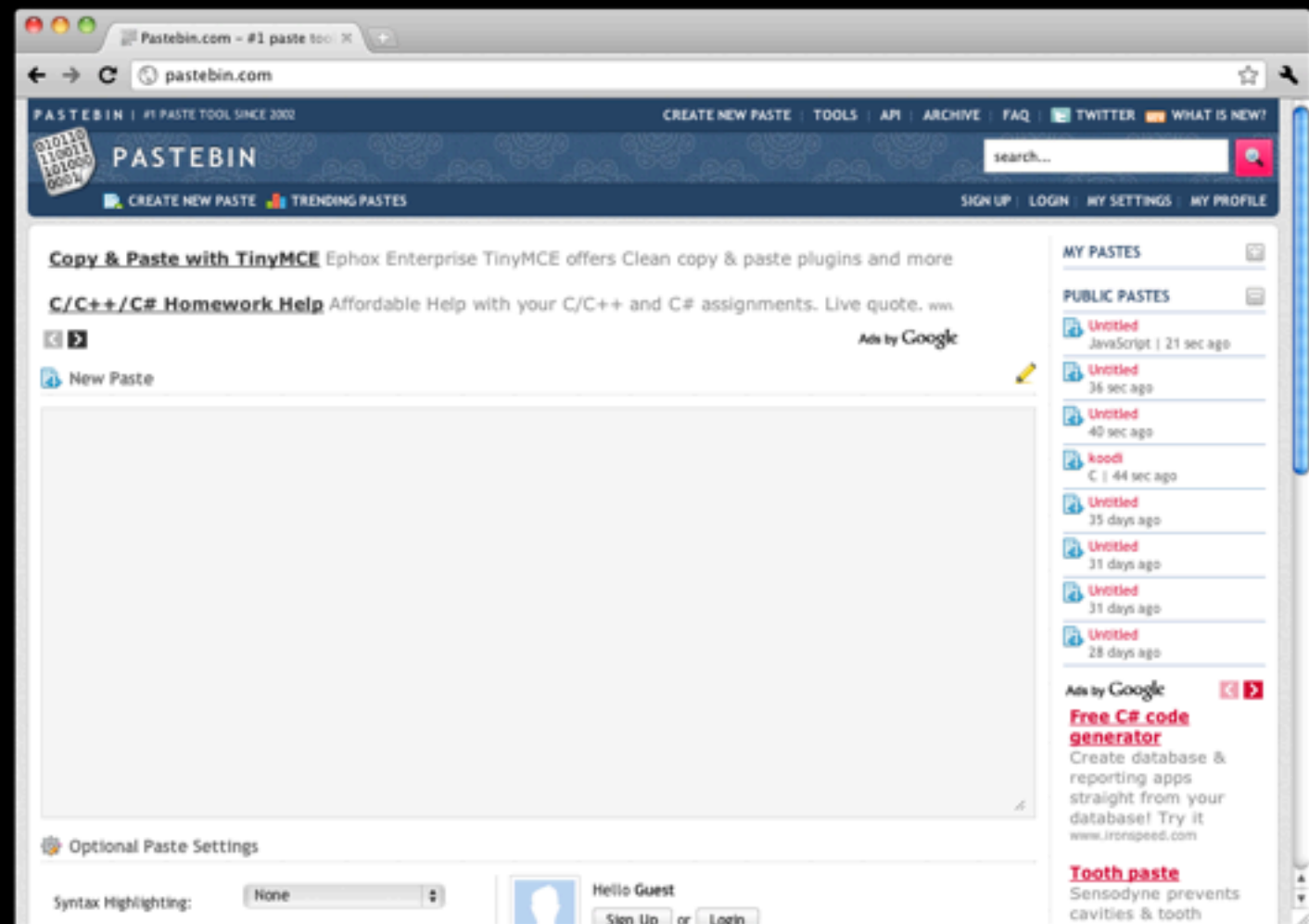
Services

Paste Bins



Paste Bins


- Common
- Useful for:
 - large dumps
 - source code sharing
 - bugs
 - diffs
 - etc.
- Lots of generic ones

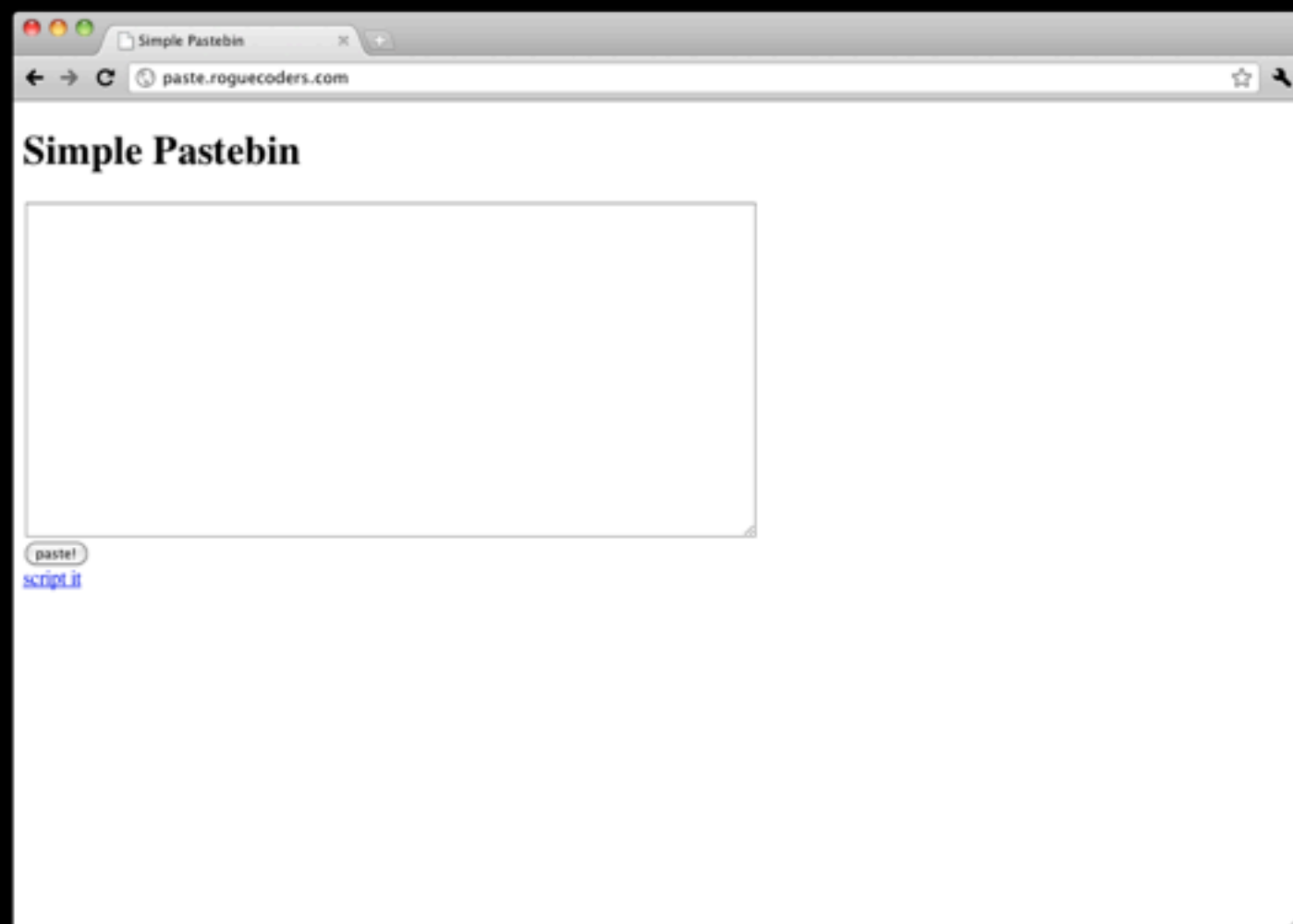


<http://pastebin.com/>



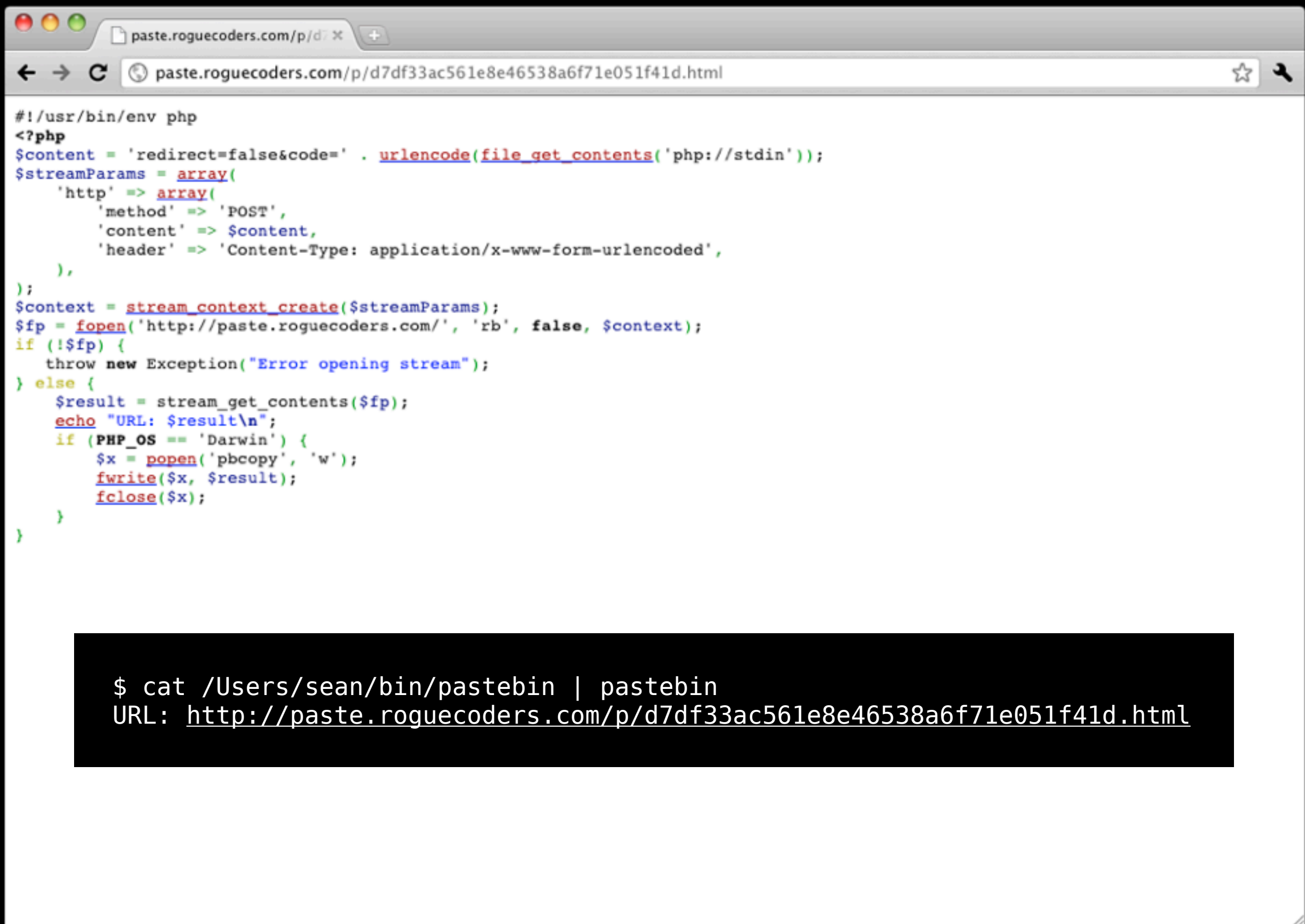
Mine

- There are many like it, but this one is mine.
- ~~micro~~ nano API: 
- “secret” URLs




<http://paste.roguecoders.com/>

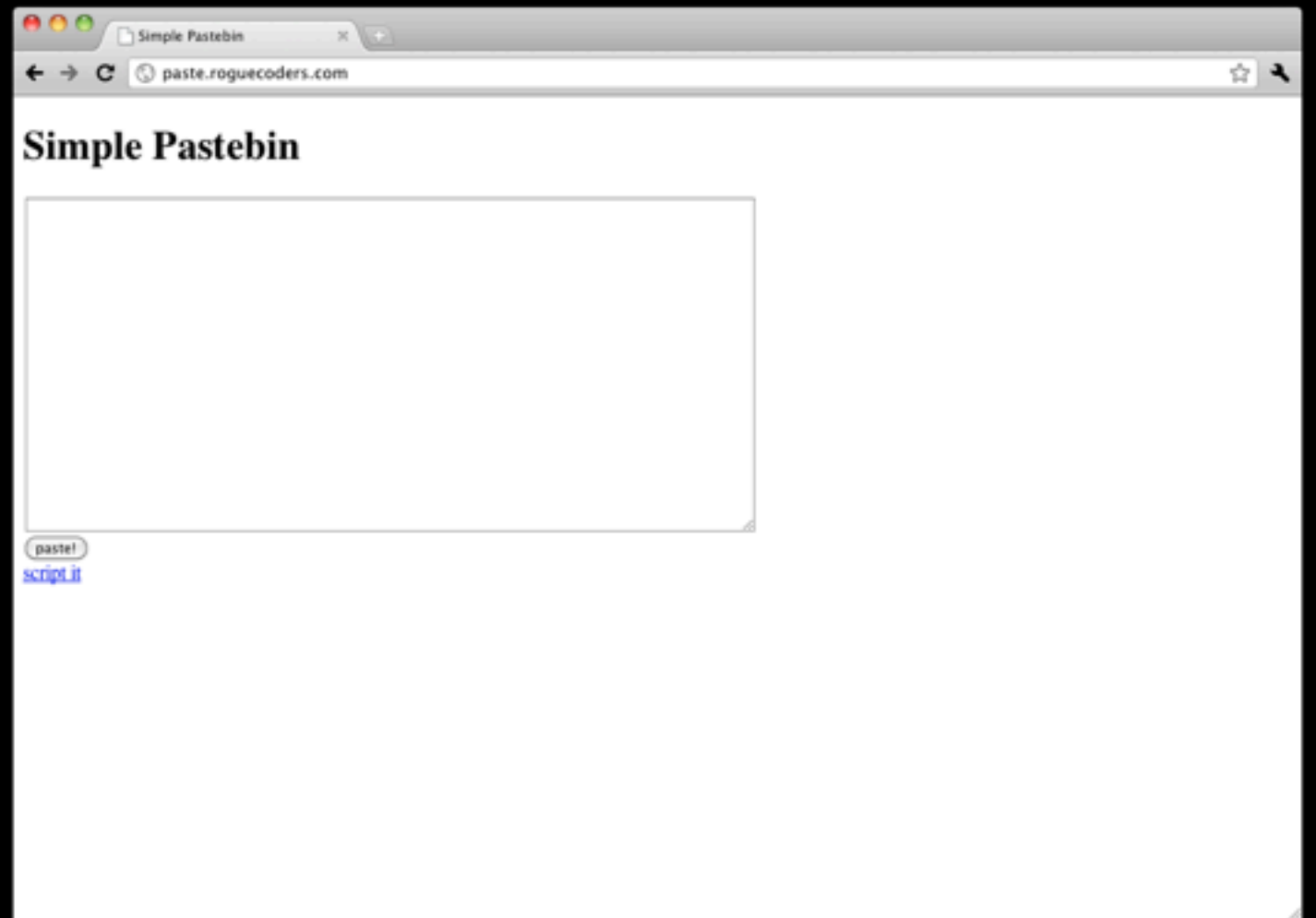




```
$ cat /Users/sean/bin/pastebin | pastebin
URL: http://paste.roguecoders.com/p/d7df33ac561e8e46538a6f71e051f41d.html
```


Mine

- There are many like it, but this one is mine.
- ~~micro~~ nano API: 
- “secret” URLs

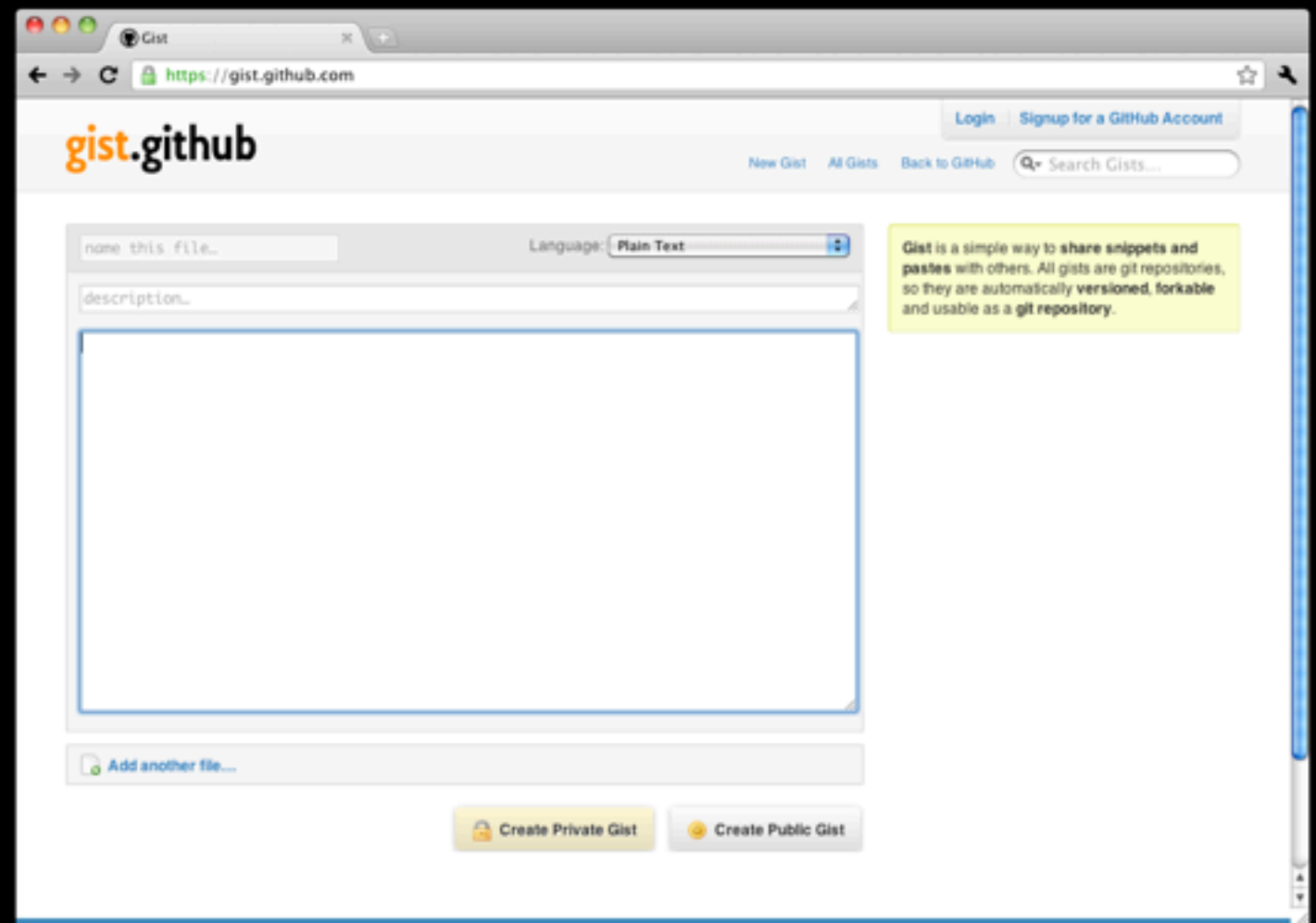


<http://paste.roguencoders.com/>



Gist (GitHub)

- Ties into GitHub
- Forkable, iterative:
- More collaborative




<http://gist.github.com/>



memusg -- Measure memory

https://gist.github.com/526585

☆ 🔍



[Login](#) | [Signup for a GitHub Account](#)

[New Gist](#) | [All Gists](#) | [Back to GitHub](#) |


gist: 526585

[download](#) [fork](#)

Description: memusg -- Measure memory usage of processes

Public Clone URL: [git://gist.github.com/526585.git](https://gist.github.com/526585.git)

Embed All Files: [show embed](#)

 **netj** (owner)

Revisions

8ee17e

netj

October 15, 2010

590293

netj

August 16, 2010

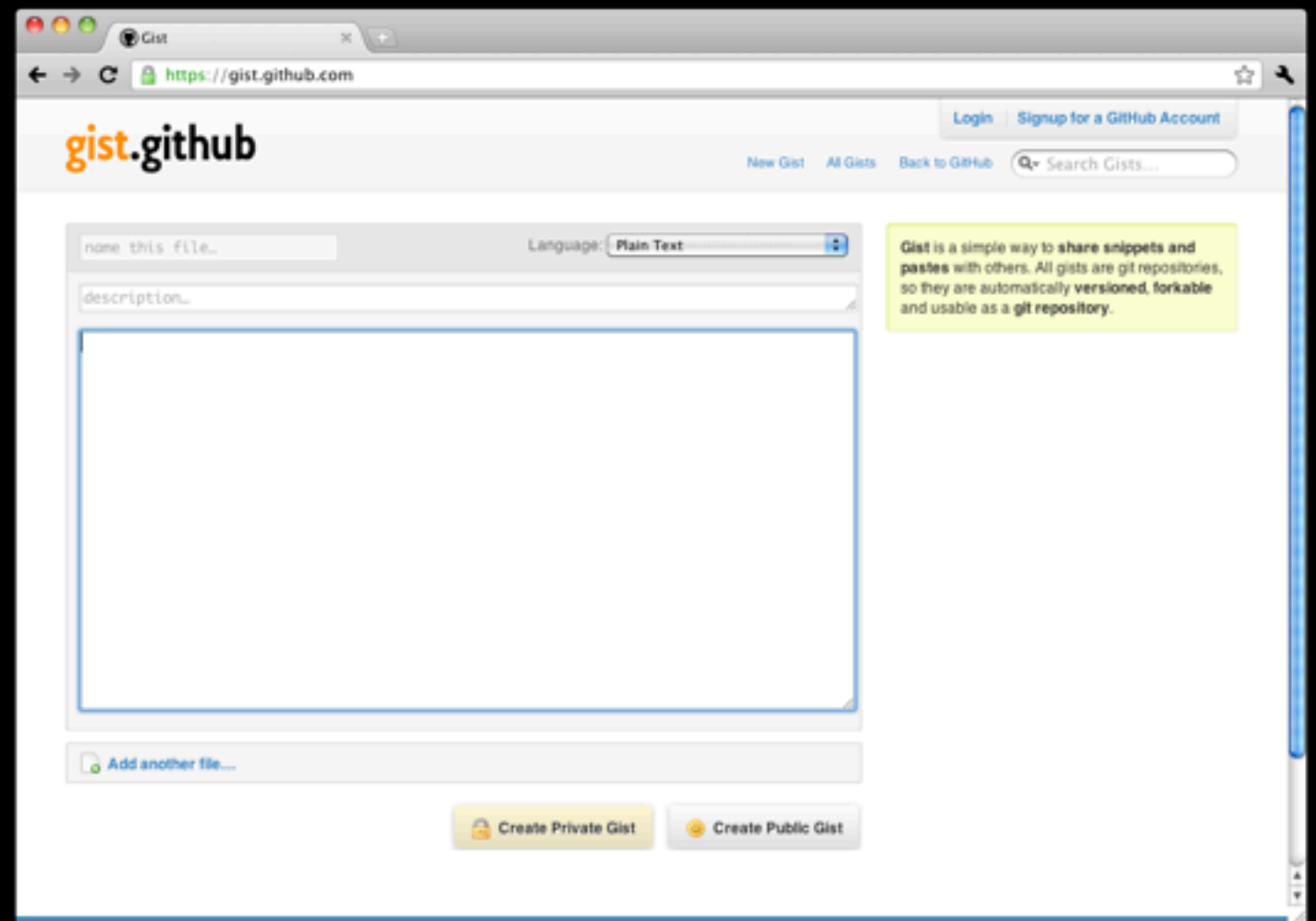
memusg # [embed](#) [raw](#)

```
1  #!/usr/bin/env bash
2  # memusg -- Measure memory usage of processes
3  # Usage: memusg COMMAND [ARGS]...
4  #
5  # Author: Jaeho Shin <netj@sparcs.org>
6  # Created: 2010-08-16
7  set -um
8
9  # check input
10 [ $# -gt 0 ] || { sed -n '2,/^\#$/ s/^\# //p' <"$0"; exit 1; }
11
12 # TODO support more options: peak, footprint, sampling rate, etc.
13
14 pgid=`ps -o pgid= $$`
15 # make sure we're in a separate process group
16 if [ x$pgid = x$(ps -o pgid= $(ps -o ppid= $$)) ]; then
17     cmd=
18     set -- "$0" "$@"
19     for a; do cmd+="'${a//\"/\"/\'/\'\\\\\"}' "; done
20     exec bash -i -c "$cmd"
21 fi
22
23 # detect operating system and prepare measurement
24 case `uname` in
25     Darwin|*BSD) sizes() { /bin/ps -o rss= -g $1; } ;;
26     Linux) sizes() { /bin/ps -o rss= -g $1; } ;;
```

Monday, March 7, 2011

Gist (GitHub)

- Ties into GitHub
- Forkable, iterative:
- More collaborative



<http://gist.github.com/>

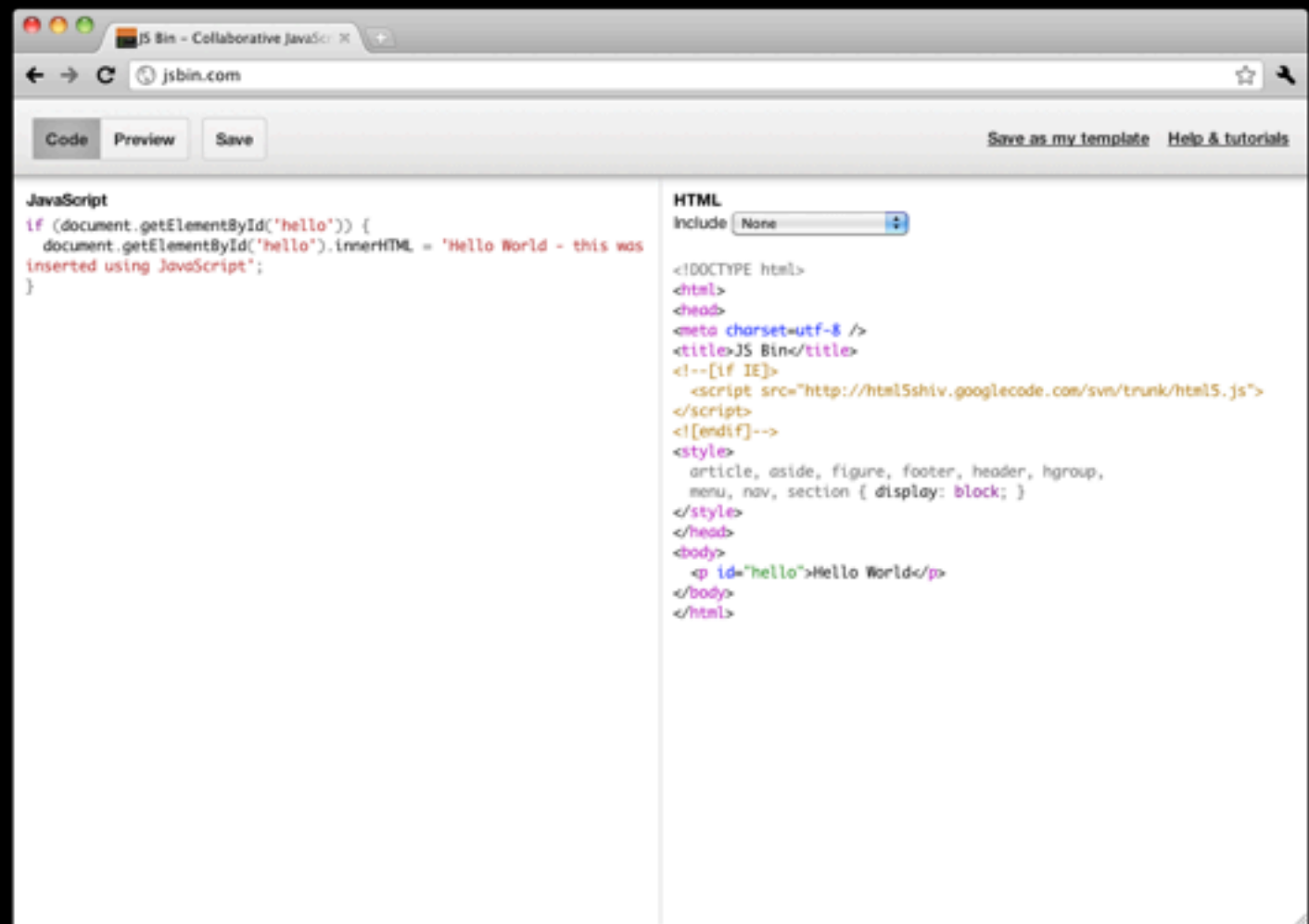


Interactive Paste Bins



JS Bin

- Paste bin that understands HTML and JavaScript
- Includes libraries
- Code available on GitHub:
 - github.com/remy/jsbin
- Great for sharing browser-specific problems
- Simple and straightforward

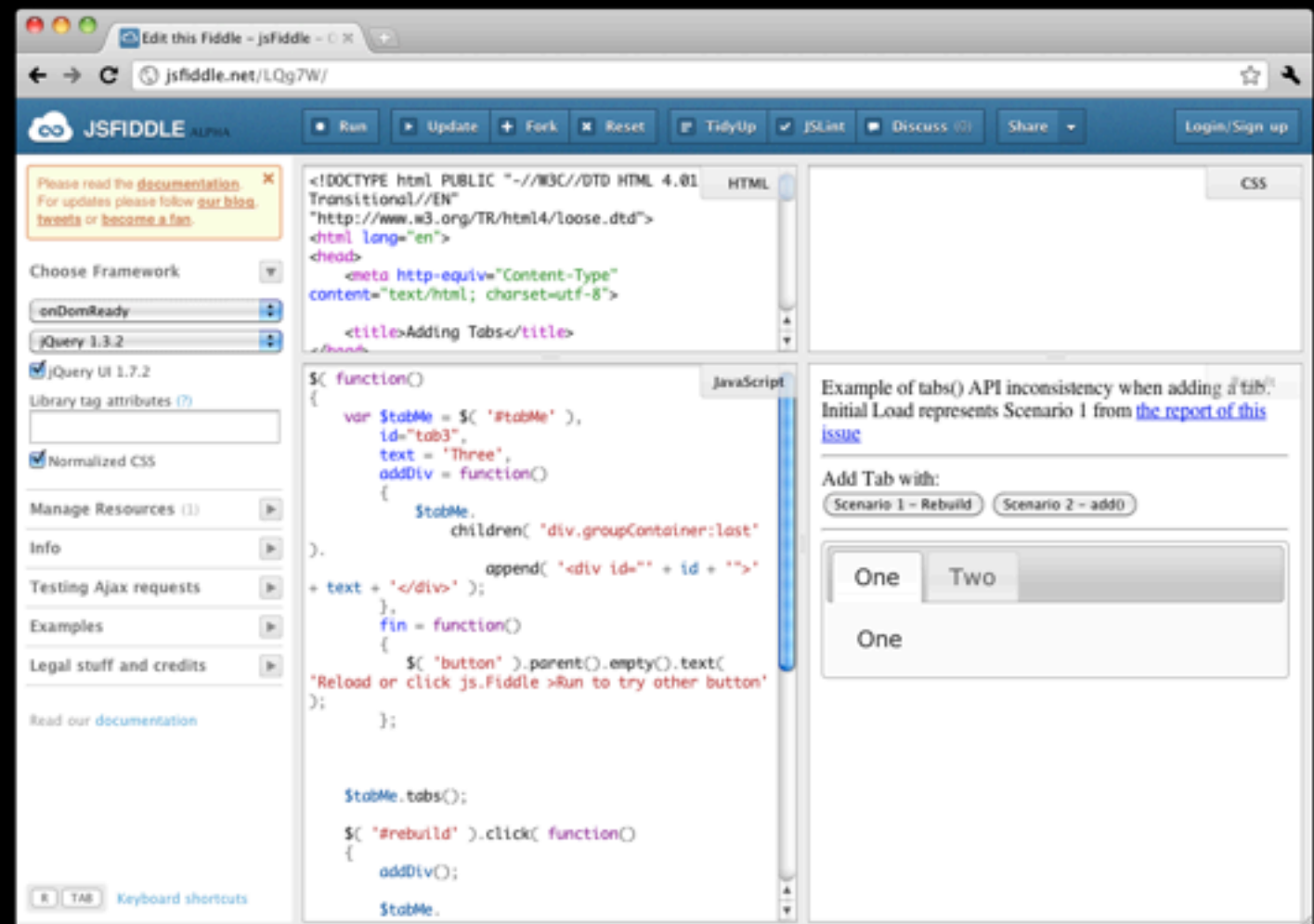


<http://jsbin.com/>



JS Fiddle

- HTML, CSS, JS
- More complex, but nicer integration for non-simple
- Easy pull-in for external URLs
- Larger framework support
- Versioning, easy sharing, forking

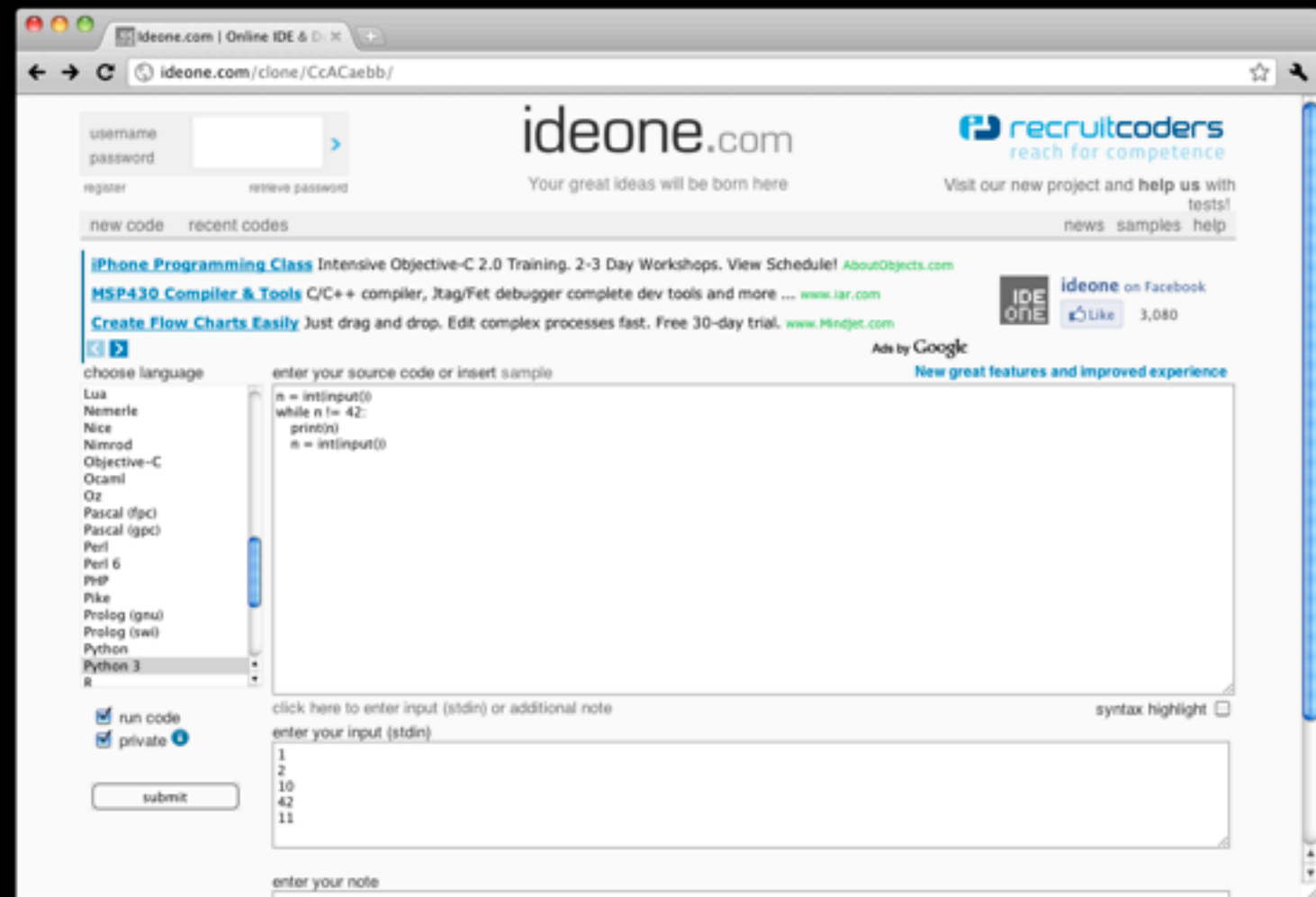


<http://jsfiddle.net/>



IDE One

- Online code compiling and execution
- Not just JavaScript: PHP, Python, Perl, Ruby, Java, C, and a whole bunch of others
- Source, input, output
- Great for posting questions on (e.g.) stackoverflow



<http://ideone.com/>



Ideone.com | Online Python3 x

ideone.com/CcACaebb/

MSP430 Compiler & Tools

C/C++ compiler, Jtag/Fet debugger complete dev tools and more ... [www.lar.com](#)

Instant FTP Site

Upload, Download, Share Large Files Free Trial. No Credit Card Required [www.egnyte.com/Free-Trial](#)

IDE ONE

ideone on Facebook

Like 3,080

Ads by Google

language: Python 3 (python-3.1.2)

date: 395 days 20 hours ago

link: <http://ideone.com/CcACaebb> [Tweet](#)

visibility: private

source code

clone

download

ShareThis

report bug / make suggestion

```
1 n = int(input())
2 while n != 42:
3     print(n)
4     n = int(input())
```

clone

download

input / output

show all

hide all

upload with new input

#1: [hide](#) [clone](#) [input](#) 395 days 20 hours ago

result: **success** time: 0.02s memory: 7388 kB returned value: 0

input:

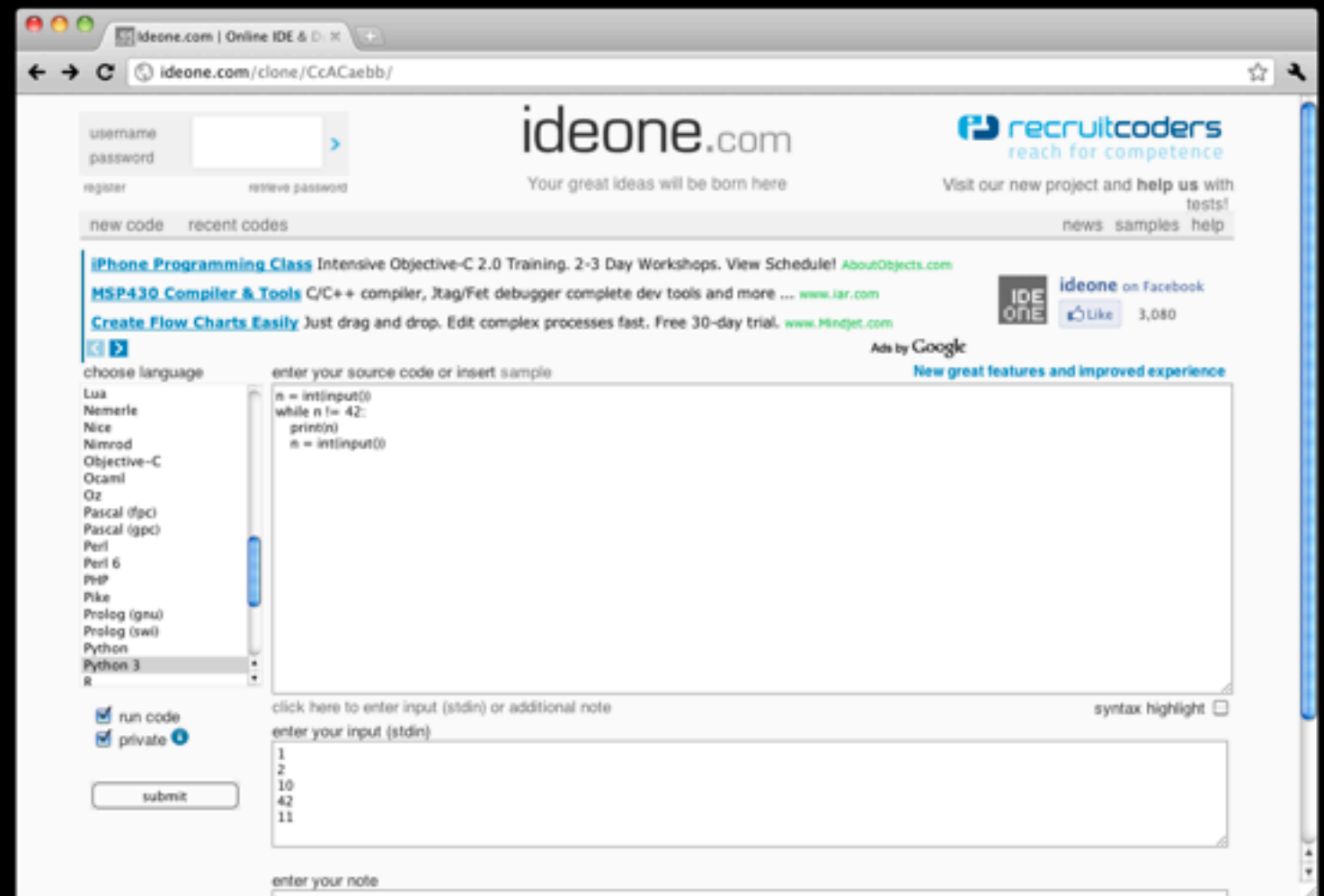
```
1
2
10
42
11
```

output:

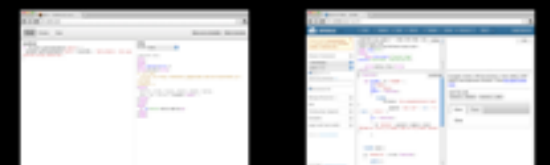
```
1
2
10
```


IDE One

- Online code compiling and execution
- Not just JavaScript: PHP, Python, Perl, Ruby, Java, C, and a whole bunch of others
- Source, input, output
- Great for posting questions on (e.g.) stackoverflow



<http://ideone.com/>

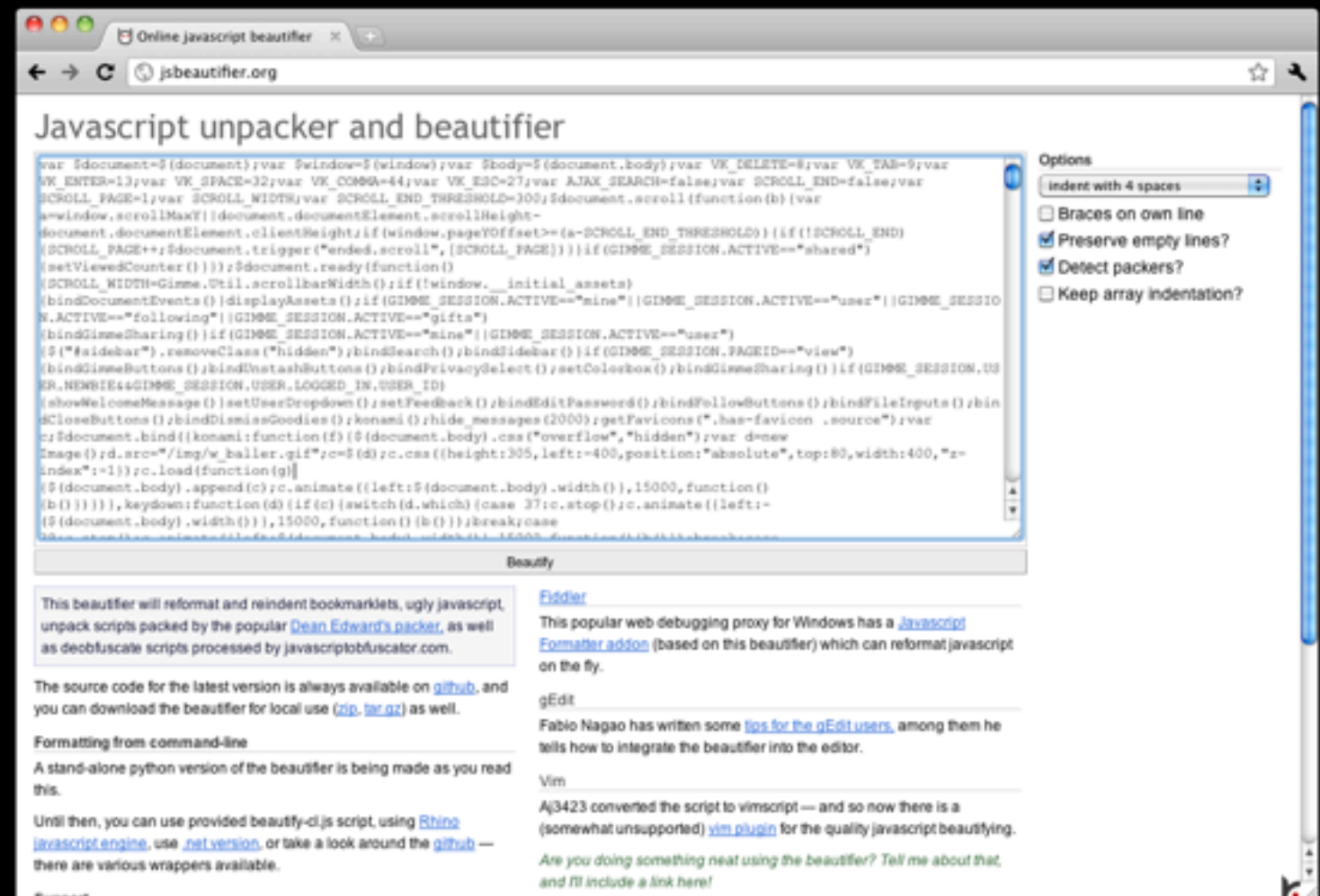


JavaScript Helpers



JS Beautifier

- Pretty printer for JS
- Useful for de-minification (will not rename variables)
- Helps spot syntax problems
- Formats JSON documents to make them readable
- Python version coming
- JS version uses Rhino



<http://jsbeautifier.org/>



Online javascript beautifier

jsbeautifier.org

Javascript unpacker and beautifier

```
var $document=$(document);var $window=$(window);var $body=$(document.body);var VK_DELETE=8;var VK_TAB=9;var
VK_ENTER=13;var VK_SPACE=32;var VK_COMMA=44;var VK_ESC=27;var AJAX_SEARCH=false;var SCROLL_END=false;var
SCROLL_PAGE=1;var SCROLL_WIDTH;var SCROLL_END_THRESHOLD=300;$document.scroll(function(b){var
a=window.scrollMaxY||document.documentElement.scrollHeight-
document.documentElement.clientHeight;if(window.pageYOffset>=(a-SCROLL_END_THRESHOLD)){if(!SCROLL_END)
{SCROLL_PAGE++;$document.trigger("ended.scroll",[SCROLL_PAGE])}if(GIMME_SESSION.ACTIVE=="shared")
{setViewedCounter()}};$document.ready(function()
{SCROLL_WIDTH=Gimme.Util.scrollbarWidth();if(!window.__initial_assets)
{bindDocumentEvents();displayAssets();if(GIMME_SESSION.ACTIVE=="mine"||GIMME_SESSION.ACTIVE=="user"||GIMME_SESSIO
N.ACTIVE=="following"||GIMME_SESSION.ACTIVE=="gifts")
{bindGimmeSharing();if(GIMME_SESSION.ACTIVE=="mine"||GIMME_SESSION.ACTIVE=="user")
{$("#sidebar").removeClass("hidden");bindSearch();bindSidebar();if(GIMME_SESSION.PAGEID=="view")
{bindGimmeButtons();bindUnstashButtons();bindPrivacySelect();setColorbox();bindGimmeSharing();if(GIMME_SESSION.US
ER.NEWBIE&&GIMME_SESSION.USER.LOGGED_IN.USER_ID)
{showWelcomeMessage();setUserDropDown();setFeedback();bindEditPassword();bindFollowButtons();bindFileInputs();bin
dCloseButtons();bindDismissGoodies();konami();hide_messages(2000);getFavicons(".has-favicon .source");var
c;$document.bind({konami:function(f){$(document.body).css("overflow","hidden");var d=new
Image();d.src="/img/w_baller.gif";c=$(d);c.css({height:305,left:-400,position:"absolute",top:80,width:400,"z-
index":-1});c.load(function(g){
{$(document.body).append(c);c.animate({left:$(document.body).width()},15000,function()
{b()}})},keydown:function(d){if(c){switch(d.which){case 37:c.stop();c.animate({left:-
$(document.body).width()},15000,function(){b()}};break;case
39:c.stop();c.animate({left:$(document.body).width()},15000,function(){b()}};break;case
```

Beautify

Options

indent with 4 spaces

☐ Braces on own line

☒ Preserve empty lines?

☒ Detect packers?

☐ Keep array indentation?

This beautifier will reformat and reindent bookmarklets, ugly javascript, unpack scripts packed by the popular [Dean Edward's packer](#), as well as deobfuscate scripts processed by [javascriptobfuscator.com](#).

The source code for the latest version is always available on [github](#), and you can download the beautifier for local use ([zip](#), [tar.gz](#)) as well.

Formatting from command-line

A stand-alone python version of the beautifier is being made as you read this.

Until then, you can use provided beautify-cl.js script, using [Rhino javascript engine](#), use [.net version](#), or take a look around the [github](#) — there are various wrappers available.

Support

[Fiddler](#)

This popular web debugging proxy for Windows has a [Javascript Formatter addon](#) (based on this beautifier) which can reformat javascript on the fly.

[gEdit](#)

Fabio Nagao has written some [tips for the gEdit users](#), among them he tells how to integrate the beautifier into the editor.

[Vim](#)

Aj3423 converted the script to vimscript — and so now there is a (somewhat unsupported) [vim plugin](#) for the quality javascript beautifying.

Are you doing something neat using the beautifier? Tell me about that, and I'll include a link here!

Monday, March 7, 2011

Online javascript beautifier

jsbeautifier.org

Javascript unpacker and beautifier

```
var $document = $(document);
var $window = $(window);
var $body = $(document.body);
var VK_DELETE = 8;
var VK_TAB = 9;
var VK_ENTER = 13;
var VK_SPACE = 32;
var VK_COMMA = 44;
var VK_ESC = 27;
var AJAX_SEARCH = false;
var SCROLL_END = false;
var SCROLL_PAGE = 1;
var SCROLL_WIDTH;
var SCROLL_END_THRESHOLD = 300;
$document.scroll(function (b) {
    var a = window.scrollMaxY || document.documentElement.scrollHeight - document.documentElement.clientHeight;
    if (window.pageYOffset >= (a - SCROLL_END_THRESHOLD)) {
        if (!SCROLL_END) {
            SCROLL_PAGE++;
            $document.trigger("ended.scroll", [SCROLL_PAGE])
        }
    }
})
if (typeof document.activeElement == "object") {
```

Options

indent with 4 spaces

☐ Braces on own line

☒ Preserve empty lines?

☒ Detect packers?

☐ Keep array indentation?

Beautify

This beautifier will reformat and reindent bookmarklets, ugly javascript, unpack scripts packed by the popular [Dean Edward's packer](#), as well as deobfuscate scripts processed by [javascriptobfuscator.com](#).

The source code for the latest version is always available on [github](#), and you can download the beautifier for local use ([zip](#), [tar.gz](#)) as well.

Formatting from command-line

A stand-alone python version of the beautifier is being made as you read this.

Until then, you can use provided beautify-cl.js script, using [Rhino javascript engine](#), use [.net version](#), or take a look around the [github](#) — there are various wrappers available.

Support

Fiddler

This popular web debugging proxy for Windows has a [Javascript Formatter addon](#) (based on this beautifier) which can reformat javascript on the fly.

gEdit

Fabio Nagao has written some [tips for the gEdit users](#), among them he tells how to integrate the beautifier into the editor.

Vim

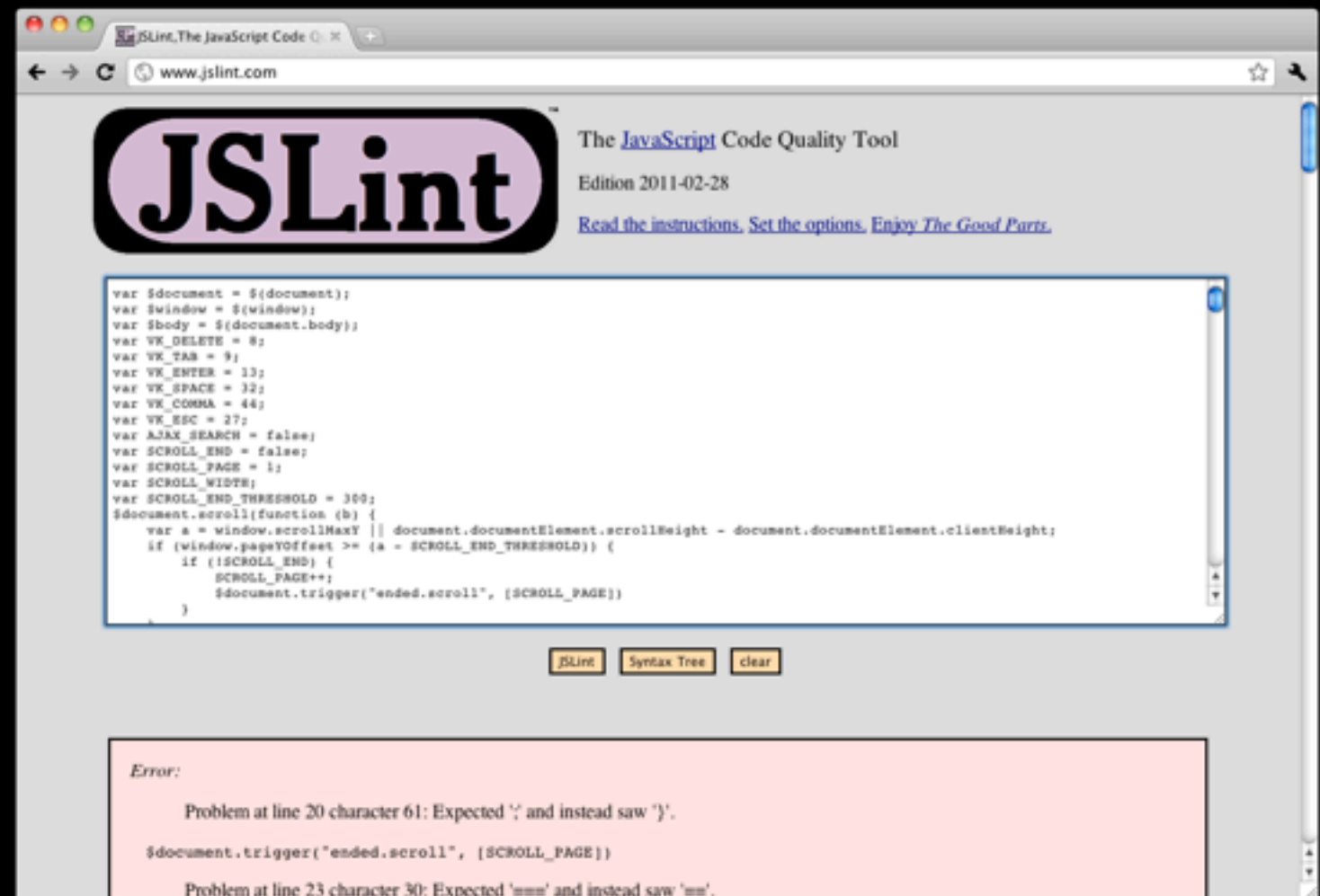
Aj3423 converted the script to vimscript — and so now there is a (somewhat unsupported) [vim plugin](#) for the quality javascript beautifying.

Are you doing something neat using the beautifier? Tell me about that, and I'll include a link here!

Monday, March 7, 2011

JSLint

- Checks the syntax of your JS
- Insults you when you do something stupid (you idiot)
- Really helpful for debugging JS's sometimes-hard-to-predict behaviour
- Annoying when you are slightly smarter than it is (-:
- Also: Crockford



<http://jslint.com/>



WonderProxy

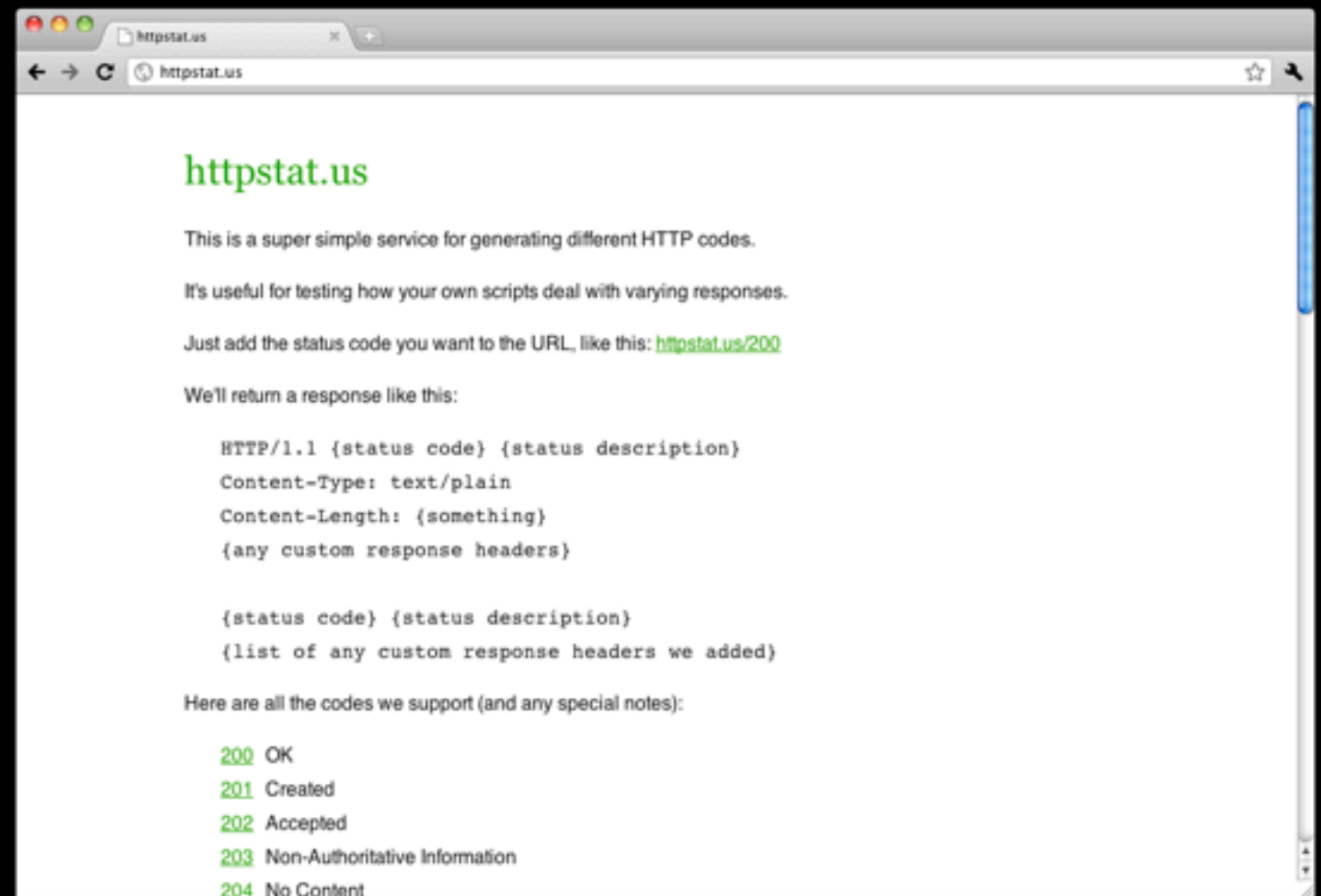
- Geolocation testing
- 37 Locations
- 25 Countries
- 5 Continents
- Find Paul Reinheimer and ask him about it



<http://wonderproxy.com/>

httpstat.us

- Quick Reference for HTTP status codes
- Build a RESTishful API!
- Test pages for each code
 - <http://httpstat.us/200>
 - <http://httpstat.us/404>
 - (etc.)



<http://httpstat.us/>

The image features the word "Unix" in a bold, white, sans-serif font. It is centered between two vertical bars of a dark teal color. The bars are positioned on either side of the text, creating a symmetrical design. The entire composition is set against a solid black background.

Unix

/etc/hosts

(or %SystemRoot%\system32\drivers\etc\hosts)

- You probably already know about this:

```
$ cat /etc/hosts | grep habari.local
```

```
127.0.0.1 habari.local
```

```
$ ping habari.local
```

```
PING habari.local (127.0.0.1): 56 data bytes
```

/etc/hosts

(or %SystemRoot%\system32\drivers\etc\hosts)

- But you can also name interfaces:

```
$ cat /etc/hosts | grep 'www.local'  
64.15.79.182 eth0-0.local www.local
```

- Useful for daemon configuration:

```
<VirtualHost www.local:80>  
    ServerName www.seancoates.com  
    RedirectMatch ^/(.*)$ http://seancoates.com/$1  
</VirtualHost>
```

netstat

- Find out which ports and IPs a process is bound to
- On Linux (and elsewhere with different arguments):

```
# netstat -tln | grep mysqld
tcp        0      0 127.0.0.1:3306
0.0.0.0:*               LISTEN      31506/
mysqld
```

- (Bound to 127.0.0.1 (localhost) on port 3306)

top & atop

- High-level system overview

```

root@iconoclast: ~ — ssh — bash — 80x50 — 3
top - 17:30:47 up 435 days, 13:02, 1 user, load average: 0.11, 0.12, 0.21
Tasks: 215 total, 1 running, 214 sleeping, 0 stopped, 0 zombie
Cpu(s): 0.3%us, 0.3%sy, 0.0%ni, 99.3%id, 0.0%wa, 0.0%hi, 0.0%si, 0.0%st
Mem: 2061408k total, 1803320k used, 258088k free, 523524k buffers
Swap: 6064496k total, 2781524k used, 3282972k free, 231340k cached

  PID USER      PR  NI  VIRT  RES  SHR  S  %CPU  %MEM    TIME+  COMMAND
 3328 couchdb  20   0 146m 9204 1208 S   0   0.4 328:12.75 beam.smp
11720 zabbix  25   5 5904 544 488 S   0   0.0 63:12.81 zabbix_agentd
15074 root    20   0 2580 1236 904 R   0   0.1 0:00.04 top
23760 phergie  20   0 31968 9272 3676 S   0   0.4 217:07.14 php
   1 root    20   0 1764 124 72 S   0   0.0 0:13.33 init
   2 root    15  -5     0     0     0 S   0   0.0 0:00.01 kthreadd
   3 root    RT  -5     0     0     0 S   0   0.0 0:10.92 migration/0
   4 root    15  -5     0     0     0 S   0   0.0 42:37.12 ksoftirqd/0
   5 root    RT  -5     0     0     0 S   0   0.0 0:00.00 watchdog/0
   6 root    RT  -5     0     0     0 S   0   0.0 0:12.21 migration/1
   7 root    15  -5     0     0     0 S   0   0.0 42:14.77 ksoftirqd/1
   8 root    RT  -5     0     0     0 S   0   0.0 0:00.00 watchdog/1
   9 root    15  -5     0     0     0 S   0   0.0 0:20.54 events/0
  10 root    15  -5     0     0     0 S   0   0.0 0:39.26 events/1
  11 root    15  -5     0     0     0 S   0   0.0 0:00.00 khelper
  12 root    RT  -5     0     0     0 S   0   0.0 0:00.00 kstop/0
  13 root    RT  -5     0     0     0 S   0   0.0 0:00.00 kstop/1
  14 root    15  -5     0     0     0 S   0   0.0 0:00.00 kintegrityd/0
  15 root    15  -5     0     0     0 S   0   0.0 0:00.00 kintegrityd/1
  16 root    15  -5     0     0     0 S   0   0.0 0:04.98 kblockd/0
  17 root    15  -5     0     0     0 S   0   0.0 0:10.11 kblockd/1
  18 root    15  -5     0     0     0 S   0   0.0 0:00.00 kacpid
  19 root    15  -5     0     0     0 S   0   0.0 0:00.00 kacpid_notify

```

```

root@iconoclast: ~ — ssh — bash — 80x50 — 3
ATOP - iconoclast 2011/03/04 17:31:17 10 seconds elapsed
PRC | sys 0.07s | user 0.22s | #proc 213 | #zombie 0 | #exit 0 |
CPU | sys 1% | user 2% | irq 0% | idle 195% | wait 2% |
cpu | sys 0% | user 1% | irq 0% | idle 97% | cpu000 w 2% |
cpu | sys 0% | user 1% | irq 0% | idle 99% | cpu001 w 0% |
CPL | avg1 0.07 | avg5 0.11 | avg15 0.20 | csw 5971 | intr 2336 |
MEM | tot 2.0G | free 245.9M | cache 226.0M | buff 511.4M | slab 103.2M |
SWP | tot 5.8G | free 3.1G | vmcom 2.6G | vmlim 6.8G |
DSK | sda busy 2% | read 14 | write 10 | avio 8 ms |
NET | transport | tcpi 69 | tcpi 60 | udpi 0 | udpo 0 |
NET | network | ipi 69 | ipo 60 | ipfrw 0 | deliv 69 |
NET | tun0 0% | pcki 3 | pcko 3 | si 0 Kbps | so 0 Kbps |
NET | eth0 0% | pcki 69 | pcko 55 | si 14 Kbps | so 8 Kbps |
NET | lo ---- | pcki 2 | pcko 2 | si 0 Kbps | so 0 Kbps |

  PID  SYSCPU  USRCPU  VGROW  RGROW  RDDSK  WRDSK  ST  EXC  S  CPU  CMD  1/1
 3328  0.00s  0.09s   OK    OK    OK    OK  --  -  S  1%  beam.smp
23760  0.00s  0.06s   OK    OK    OK    OK  --  -  S  1%  php
4606  0.00s  0.03s   OK    OK    OK    OK  --  -  S  0%  trac.fcgi
30271  0.01s  0.02s   OK    OK    OK    OK  --  -  S  0%  php
15078  0.01s  0.01s   OK    OK    OK    OK  --  -  R  0%  atop
12922  0.02s  0.00s   OK    OK    OK    OK  --  -  S  0%  snmpd
11720  0.02s  0.00s   OK    OK    OK    OK  --  -  S  0%  zabbix_agentd
7721  0.01s  0.00s   OK    OK    OK    OK  --  -  S  0%  ctrlproxy
14514  0.00s  0.01s   OK    OK    OK    4K  --  -  S  0%  apache2
31506  0.00s  0.00s   OK    4K    48K   60K  --  -  S  0%  mysqld
14621  0.00s  0.00s   OK   12K    OK    4K  --  -  S  0%  apache2
14623  0.00s  0.00s   OK  2244K   36K    OK  --  -  D  0%  apache2
14924  0.00s  0.00s  140K   32K    OK    OK  --  -  S  0%  sshd
15046  0.00s  0.00s   OK   36K    4K   12K  --  -  S  0%  sftp-server
741  0.00s  0.00s   OK    OK    OK    OK  --  -  S  0%  klogd

```



```
root@iconoclast: ~ — ssh — bash — 80x50 — 3
bash root@icon...st: ~ — ssh bash
top - 17:30:47 up 435 days, 13:02, 1 user, load average: 0.11, 0.12, 0.21
Tasks: 215 total, 1 running, 214 sleeping, 0 stopped, 0 zombie
Cpu(s): 0.3%us, 0.3%sy, 0.0%ni, 99.3%id, 0.0%wa, 0.0%hi, 0.0%si, 0.0%st
Mem: 2061408k total, 1803320k used, 258088k free, 523524k buffers
Swap: 6064496k total, 2781524k used, 3282972k free, 231340k cached

PID USER PR NI VIRT RES SHR S %CPU %MEM TIME+ COMMAND
3328 couchdb 20 0 146m 9204 1208 S 0 0.4 328:12.75 beam.smp
11720 zabbix 25 5 5904 544 488 S 0 0.0 63:12.81 zabbix_agentd
15074 root 20 0 2580 1236 904 R 0 0.1 0:00.04 top
23760 phergie 20 0 31968 9272 3676 S 0 0.4 217:07.14 php
1 root 20 0 1764 124 72 S 0 0.0 0:13.33 init
2 root 15 -5 0 0 0 S 0 0.0 0:00.01 kthreadd
3 root RT -5 0 0 0 S 0 0.0 0:10.92 migration/0
4 root 15 -5 0 0 0 S 0 0.0 42:37.12 ksoftirqd/0
5 root RT -5 0 0 0 S 0 0.0 0:00.00 watchdog/0
6 root RT -5 0 0 0 S 0 0.0 0:12.21 migration/1
7 root 15 -5 0 0 0 S 0 0.0 42:14.77 ksoftirqd/1
8 root RT -5 0 0 0 S 0 0.0 0:00.00 watchdog/1
9 root 15 -5 0 0 0 S 0 0.0 0:20.54 events/0
10 root 15 -5 0 0 0 S 0 0.0 0:39.26 events/1
11 root 15 -5 0 0 0 S 0 0.0 0:00.00 khelper
12 root RT -5 0 0 0 S 0 0.0 0:00.00 kstop/0
13 root RT -5 0 0 0 S 0 0.0 0:00.00 kstop/1
14 root 15 -5 0 0 0 S 0 0.0 0:00.00 kintegrityd/0
15 root 15 -5 0 0 0 S 0 0.0 0:00.00 kintegrityd/1
16 root 15 -5 0 0 0 S 0 0.0 0:04.98 kblockd/0
17 root 15 -5 0 0 0 S 0 0.0 0:10.11 kblockd/1
18 root 15 -5 0 0 0 S 0 0.0 0:00.00 kacpid
19 root 15 -5 0 0 0 S 0 0.0 0:00.00 kacpid_notifier
```

elapsed 0

2%

2%

0%

2336

03.2M

6.8G

8 ms

0

69

Kbps

Kbps

Kbps

1/1

gi

agentd

xy

server

1.4

root@iconoclast: ~ — ssh — bash — 80x50 — 3

bash

root@icon...st: ~ — ssh

bash

ATOP - iconoclast

2011/03/04 17:31:17

10 seconds elapsed

PRC	sys	0.07s	user	0.22s	#proc	213	#zombie	0	#exit	0
CPU	sys	1%	user	2%	irq	0%	idle	195%	wait	2%
cpu	sys	0%	user	1%	irq	0%	idle	97%	cpu000 w	2%
cpu	sys	0%	user	1%	irq	0%	idle	99%	cpu001 w	0%
CPL	avg1	0.07	avg5	0.11	avg15	0.20	cs	5971	intr	2336
MEM	tot	2.0G	free	245.9M	cache	226.0M	buff	511.4M	slab	103.2M
SWP	tot	5.8G	free	3.1G			vmcom	2.6G	vmlim	6.8G
DSK	sda	busy	2%	read	14	write	10	avio	8 ms	
NET	transport	tcpi	69	tcpo	60	udpi	0	udpo	0	
NET	network	ipi	69	ipo	60	ipfrw	0	deliv	69	
NET	tun0	0%	pcki	3	pcko	3	si	0 Kbps	so	0 Kbps
NET	eth0	0%	pcki	69	pcko	55	si	14 Kbps	so	8 Kbps
NET	lo	----	pcki	2	pcko	2	si	0 Kbps	so	0 Kbps

PID	SYSCPU	USRCPU	VGROW	RGROW	RDDSK	WRDSK	ST	EXC	S	CPU	CMD	1/1
3328	0.00s	0.09s	0K	0K	0K	0K	--	-	S	1%	beam.smp	
23760	0.00s	0.06s	0K	0K	0K	0K	--	-	S	1%	php	
4606	0.00s	0.03s	0K	0K	0K	0K	--	-	S	0%	trac.fcgi	
30271	0.01s	0.02s	0K	0K	0K	0K	--	-	S	0%	php	
15078	0.01s	0.01s	0K	0K	0K	0K	--	-	R	0%	atop	
12922	0.02s	0.00s	0K	0K	0K	0K	--	-	S	0%	snmpd	
11720	0.02s	0.00s	0K	0K	0K	0K	--	-	S	0%	zabbix_agentd	
7721	0.01s	0.00s	0K	0K	0K	0K	--	-	S	0%	ctrlproxy	
14514	0.00s	0.01s	0K	0K	0K	4K	--	-	S	0%	apache2	
31506	0.00s	0.00s	0K	4K	48K	60K	--	-	S	0%	mysqld	
14621	0.00s	0.00s	0K	12K	0K	4K	--	-	S	0%	apache2	
14623	0.00s	0.00s	0K	2244K	36K	0K	--	-	D	0%	apache2	
14924	0.00s	0.00s	140K	32K	0K	0K	--	-	S	0%	sshd	
15046	0.00s	0.00s	0K	36K	4K	12K	--	-	S	0%	sftp-server	
741	0.00s	0.00s	0K	0K	0K	8K	--	-	S	0%	kiourald	

top & atop

- High-level overview

```
root@iconoclast: ~ — ssh — bash — 80x50 — 33
bash root@icon...st: ~ — ssh bash
top - 17:30:47 up 435 days, 13:02, 1 user, load average: 0.11, 0.12, 0.21
Tasks: 215 total, 1 running, 214 sleeping, 0 stopped, 0 zombie
Cpu(s): 0.3%us, 0.3%sy, 0.0%ni, 99.3%id, 0.0%wa, 0.0%hi, 0.0%si, 0.0%st
Mem: 2061408k total, 1803320k used, 258088k free, 523524k buffers
Swap: 6064496k total, 2781524k used, 3282972k free, 231340k cached

  PID USER      PR  NI  VIRT  RES  SHR  S  %CPU  %MEM    TIME+  COMMAND
 3328 couchdb  20   0 146m 9204 1208 S   0   0.4 328:12.75 beam.smp
11720 zabbix  25   5 5904 544 488 S   0   0.0 63:12.81 zabbix_agentd
15074 root    20   0 2580 1236 904 R   0   0.1 0:00.04 top
23760 phergie  20   0 31968 9272 3676 S   0   0.4 217:07.14 php
   1 root    20   0 1764 124 72 S   0   0.0 0:13.33 init
   2 root    15  -5     0     0   0 S   0   0.0 0:00.01 kthreadd
   3 root    RT  -5     0     0   0 S   0   0.0 0:10.92 migration/0
   4 root    15  -5     0     0   0 S   0   0.0 42:37.12 ksoftirqd/0
   5 root    RT  -5     0     0   0 S   0   0.0 0:00.00 watchdog/0
   6 root    RT  -5     0     0   0 S   0   0.0 0:12.21 migration/1
   7 root    15  -5     0     0   0 S   0   0.0 42:14.77 ksoftirqd/1
   8 root    RT  -5     0     0   0 S   0   0.0 0:00.00 watchdog/1
   9 root    15  -5     0     0   0 S   0   0.0 0:20.54 events/0
  10 root    15  -5     0     0   0 S   0   0.0 0:39.26 events/1
  11 root    15  -5     0     0   0 S   0   0.0 0:00.00 khelper
  12 root    RT  -5     0     0   0 S   0   0.0 0:00.00 kstop/0
  13 root    RT  -5     0     0   0 S   0   0.0 0:00.00 kstop/1
  14 root    15  -5     0     0   0 S   0   0.0 0:00.00 kintegrityd/0
  15 root    15  -5     0     0   0 S   0   0.0 0:00.00 kintegrityd/1
  16 root    15  -5     0     0   0 S   0   0.0 0:04.98 kblockd/0
  17 root    15  -5     0     0   0 S   0   0.0 0:10.11 kblockd/1
  18 root    15  -5     0     0   0 S   0   0.0 0:00.00 kacpid
 19 root    15  -5     0     0   0 S   0   0.0 0:00.00 kacpid_notify
```

```
root@iconoclast: ~ — ssh — bash — 80x50 — 33
bash root@icon...st: ~ — ssh bash
ATOP - iconoclast 2011/03/04 17:31:17 10 seconds elapsed
PRC | sys 0.07s | user 0.22s | #proc 213 | #zombie 0 | #exit 0 |
CPU | sys 1% | user 2% | irq 0% | idle 195% | wait 2% |
cpu | sys 0% | user 1% | irq 0% | idle 97% | cpu000 w 2% |
cpu | sys 0% | user 1% | irq 0% | idle 99% | cpu001 w 0% |
CPL | avg1 0.07 | avg5 0.11 | avg15 0.20 | csw 5971 | intr 2336 |
MEM | tot 2.0G | free 245.9M | cache 226.0M | buff 511.4M | slab 103.2M |
SWP | tot 5.8G | free 3.1G | vmcom 2.6G | vmlim 6.8G |
DSK | sda busy 2% | read 14 | write 10 | avio 8 ms |
NET | transport | tcpi 69 | tcpo 60 | udpi 0 | udpo 0 |
NET | network | ipi 69 | ipo 60 | ipfrw 0 | deliv 69 |
NET | tun0 0% | pcki 3 | pcko 3 | si 0 Kbps | so 0 Kbps |
NET | eth0 0% | pcki 69 | pcko 55 | si 14 Kbps | so 8 Kbps |
NET | lo ---- | pcki 2 | pcko 2 | si 0 Kbps | so 0 Kbps |

  PID  SYSCPU  USRCPU  VGROW  RGROW  RDDSK  WRDSK  ST  EXC  S  CPU  CMD  1/1
 3328  0.00s  0.09s   OK    OK    OK    OK  --  -  S  1%  beam.smp
23760  0.00s  0.06s   OK    OK    OK    OK  --  -  S  1%  php
4606  0.00s  0.03s   OK    OK    OK    OK  --  -  S  0%  trac.fcgi
30271  0.01s  0.02s   OK    OK    OK    OK  --  -  S  0%  php
15078  0.01s  0.01s   OK    OK    OK    OK  --  -  R  0%  atop
12922  0.02s  0.00s   OK    OK    OK    OK  --  -  S  0%  snmpd
11720  0.02s  0.00s   OK    OK    OK    OK  --  -  S  0%  zabbix_agentd
7721  0.01s  0.00s   OK    OK    OK    OK  --  -  S  0%  ctrlproxy
14514  0.00s  0.01s   OK    OK    OK    4K  --  -  S  0%  apache2
31506  0.00s  0.00s   OK    4K    48K   60K  --  -  S  0%  mysqld
14621  0.00s  0.00s   OK   12K    OK    4K  --  -  S  0%  apache2
14623  0.00s  0.00s   OK  2244K   36K    OK  --  -  D  0%  apache2
14924  0.00s  0.00s  140K   32K    OK    OK  --  -  S  0%  sshd
15046  0.00s  0.00s   OK   36K    4K   12K  --  -  S  0%  sftp-server
741  0.00s  0.00s   OK    OK    OK    OK  --  -  S  0%  klogd
```


lsof

- Ever wondered which process has a file open?

```
sean@iconoclast:~$ ls -l sekret.log
```

```
-rw-rw-r-- 1 sean sean 32 2011-03-04 17:49 sekret.log
```

```
sean@iconoclast:~$ ls -l sekret.log
```

```
-rw-rw-r-- 1 sean sean 48 2011-03-04 17:49 sekret.log
```

```
sean@iconoclast:~$ lsof sekret.log
```

COMMAND	PID	USER	FD	TYPE	DEVICE	SIZE	NODE	NAME
---------	-----	------	----	------	--------	------	------	------

php	17142	sean	3w	REG	8,1	80	755383	sekret.log
-----	-------	------	----	-----	-----	----	--------	------------

```
sean@iconoclast:~$ ps auxwww | grep 17142
```

sean	17142	0.0	0.3	24008	6760	pts/10	S	17:49
------	-------	-----	-----	-------	------	--------	---	-------

0:00	php	./ninja.php
------	-----	-------------

strace

- Ever wondered which syscalls a process makes?

```
sean@iconoclast:~$ strace php -r 'echo "F00";' 2>&1 | grep F00
execve("/usr/local/bin/php", ["php", "-r", "echo \"F00\";"],
[/* 21 vars */) = 0
write(1, "F00"..., 3F00)          = 3
sean@iconoclast:~$ strace php -r 'error_log("F00");' 2>&1 | grep F00
execve("/usr/local/bin/php", ["php", "-r", "error_log(\"F00\")"],
[/* 21 vars */) = 0
write(2, "F00\n"..., 4F00)
```


ldd

- Ever wondered which libraries an executable references?

```
sean@iconoclast:~$ ldd `which php` | grep mysql
```

```
libmysqlclient.so.15 => /usr/lib/libmysqlclient.so.15 (0xb77ed000)
```

```
sean@iconoclast:~$ ldd `which php`
```

```
linux-gate.so.1 => (0xb7ee2000)
```

```
libcrypt.so.1 => /lib/tls/i686/cmov/libcrypt.so.1 (0xb7ea2000)
```

```
libc-client.so.2007b => /usr/lib/libc-client.so.2007b (0xb7d8e000)
```

```
libreadline.so.5 => /lib/libreadline.so.5 (0xb7d5b000)
```

```
libncurses.so.5 => /lib/libncurses.so.5 (0xb7d29000)
```

```
librt.so.1 => /lib/tls/i686/cmov/librt.so.1 (0xb7d20000)
```

```
libpam.so.0 => /lib/libpam.so.0 (0xb7d14000)
```

...

nice

- Ever wanted to run a job that needs a higher or lower priority than other jobs?

```
sean@gimmebarvm-sean:~$ sudo nice -n -19 php -r  
    'for ($a=0; $a<=100000000; $a++) {}; echo "DONE\n";' &  
[1] 7025  
sean@gimmebarvm-sean:~$ time ls  
gimmebar-access_log  mime.php  prod_dump  src  
real 0m0.129s
```

nice

- The priority can range from -19 (highest) to +20 (lowest), and the default is 10.

```
sean@gimmebarvm-sean:~$ sudo nice -n 19 php -r  
    'for ($a=0; $a<=100000000; $a++) {}; echo "DONE\n";' &  
[1] 7044  
sean@gimmebarvm-sean:~$ time ls  
gimmebar-access_log  mime.php  prod_dump  src  
real 0m0.003s
```

cron

- Users have individual cron spools on most Unix systems
- Need a system-wide cron? Try `/etc/crontab`
 - Note: it has a 6th field (not just the time fields): user
- Much easier to maintain in version control and to provision with `sysdmin` tools
- Try also `/etc/cron.daily`, `/etc/cron.hourly`, etc.

GNU Screen

- Terminal multiplexer
- Keep shell apps running even after logout
- Handles scrollback, window splits, pane management
- See also: tmux

visudo, vipw, vigr

- Imagine what kind of horrible things could happen if you break a file like `/etc/sudoers`? or `/etc/passwd`?
 - visudo: safely edit `/etc/sudoers`
 - vipw: safely edit `/etc/passwd`
 - vigr: safely edit `/etc/group`
- Files are parsed for errors before they're committed



Shell

reverse-i-search (ctrl-r)

reverse-i-search (**ctrl-r**)

gimmebar\$ █



reuse last argument (**esc-.**)

reuse last argument (**esc-.**)

gimmebar\$ █



edit+run last command (**fc**)


edit+run last command (fc)

gimmebar\$ █



bash completion ((tab))

bash completion ((tab))

gimmebar\$ 



pipes

- One of the great beauties of Unix: text streams
- Chain these together, easily

```
$ cat /www/logs/seancoates.com-access_log \  
  | grep 'gimme' \  
  | wc -l  
12
```

- Small, purposeful utilities, not monoliths

grep, zgrep, ack

- Filter text streams with grep (previous slide)
- Uncompress **and** filter with zgrep
- Use ack (non-standard) grouping, and to do the right thing:

```
sarcasm:~/src/gimmebar/api (master)$ ack 'Config::'
```

```
inc/content/Flickr.php
```

```
22:          'api_key' => Config::FLICKR_API_KEY,
```

```
inc/storage/S3Storage.php
```

```
24:          Config::S3_BUCKET,
```

```
34:          return 'http://' . Config::S3_BUCKET .
```


download without the file

- Pipe curl directly into tar:

```
$ curl http://files.seancoates.com/test.tar.gz | tar xzf
```

- Alternative to wget + tar, but doesn't leave a mess
- (not always what you want, but still convenient)

bash: always there

- Bash is on almost every system
- Scriptable and powerful, even when alternatives aren't available

```
$ A='0'; \  
  for f in *.php; do A="$A + `cat $f | wc -c`"; done; \  
  echo $A | bc
```

- Shows the sum of the file sizes of all PHP files in the current directory (thanks to `bc`)

perl pie (see also: sed)

- Perl pie runs an arbitrary expression against the passed files and modifies them in place
- My favourite (but least intelligent) refactoring tool

```
$ perl -p -i -e 's/Config::/Configuration::/g' *.php
```

- Replaces “Config” with “Configuration” static calls:

```
-          'database' => Config::DATABASE_URL  
+          'database' => Configuration::DATABASE_URL
```

sticky shell start

- Put this in ~/.bashrc or ~/.profile

```
cd () { builtin cd "$@" ; pwd > ~/.pwd; }  
cd "`cat ~/.pwd`"
```

- New shells launch where the last shell operated

```
sarcasm:~/src/gimmebar/api/www-api (master)$ login  
login: sean  
Password:  
Last login: Sun Mar  6 18:09:15 on ttys000  
sarcasm:~/src/gimmebar/api/www-api (master)$
```

misc. shell stuffs

- The ``find`` command is your friend when working in a directory tree
- Use ``tail -f filename`` to continually watch a file
- When scripting the shell, the ``xargs`` command is often useful
- Use ``2>&1`` to merge stderr into stdout, as in:

```
a_command_that_uses_stderr >/dev/null 2>&1
```



SSH

keypairs

- Skip the password, use pre-shared keys to log in to your shells
- Use ``ssh-keygen`` to generate a pair
- Keep the private key protected
- Put the public key in `~/.ssh/authorized_keys` on the server
- Log in without a password
- (If it doesn't work, one common error is permissions; ``ssh -v ...`` will often tell you; if not: use more ``v``)

ssh-copy-id

- Easily copy your new key to a remote server
- Sets permissions properly, and uses the right file if it's not
~/.ssh/authorized_keys

ssh config

- Local configuration allows host-specific settings for:
 - username
 - private key
 - forwarded ports
 - many other settings (compression, X forwarding, etc.)
- ~/.ssh/config:

```
Host *.php.net  
    User scoates
```

ssh tunnels (port forwarding)

- Make the remote machine's MongoDB port available as localhost:27017

```
ssh -L27017:localhost:27017 local.gimmebar.com
```

- Forward your local port 80 to a remote machine as port 8000

```
ssh -R80:localhost:8000 iconoclast.caedmon.net
```

- (can now access with ``curl localhost:8000/`` on iconoclast)

kill ssh

- ssh sessions can hang
- `killall ssh` sucks if you have more than one session
- [enter] ~ .
- (SSH over VPN == extremely robust)

A large, stylized letter 'V' in a dark teal color, centered on the slide. It has a slight shadow or reflection effect at the bottom.

Development

const == var (not var == const)

- All else being equal, I prefer this:

```
if (3 == $tries) { ... }
```

- Over this:

```
if ($tries == 3) { ... }
```

- Because this causes a compiler error:

```
if (3 = $tries) { ... }
```

git: publish on post-receive

- put this in your post-receive hook:

```
echo "Staging dev.seancoates.com ..." && \  
cd /www/webroots/seancoates.com/seancoates-dev && \  
env -i git pull && \  
echo "Staged."
```

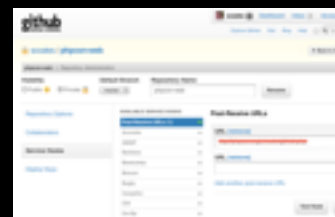
- When you push, this path is pulled

github: publish on post-receive


- Put this script somewhere in your web root (beware of security and stuff):

```
<?php
$gitpath = '/usr/local/bin/git';
header("Content-type: text/plain");
system("/usr/bin/env -i " .
    "HOME=/www/webroots/phpcon.org " .
    "{$gitpath} pull 2>&1");
echo "\nDone.\n";
```

- ...and set this up on GitHub:



github: publish on post-receive



 SOCIAL CODING

scoates 1 | [Dashboard](#) | [Inbox](#) 0 | [Account Settings](#)

[Explore GitHub](#) | [Gist](#) | [Blog](#) | [Help](#) |

scoates / [phpcon-web](#) [← Back to Source](#)

phpcon-web → Repository Administration











Visibility
☐ Public  ☒ Private 

Default Branch
master

Repository Name
phpcon-web [Rename](#)

[Repository Options](#)
[Collaborators](#)
Service Hooks
[Deploy Keys](#)

AVAILABLE SERVICE HOOKS

Post-Receive URLs (1)	
Acunote	
AMQP	
Bamboo	
Basecamp	
Boxcar	
Bugly	
Campfire	
CIA	
Co-Op	

Post-Receive URLs

[URL \(remove\)](#)
~~http://phpcon.org/_hooks/github.php~~

[URL \(remove\)](#)

[Add another post-receive URL](#)

[Test Hook](#) [Update](#)

Email override

- If your code uses sendmail to... send... mail... and if you can override the path, change the path to point to his on dev:

```
cat >> /tmp/logmail.log
```

- PHP makes this easy with the sendmail_path INI setting

Crockford on JS

- <http://developer.yahoo.com/yui/theater>

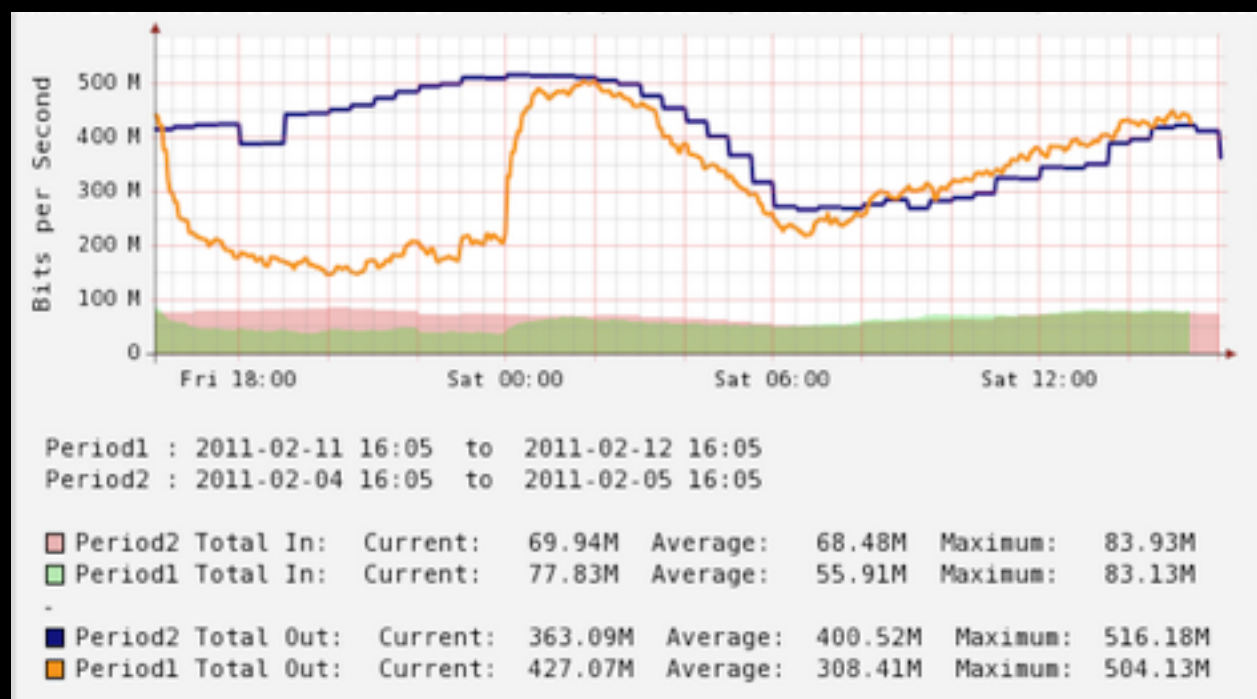


A large, dark teal, stylized letter 'W' serves as a background for the title. It is composed of three vertical strokes and two diagonal strokes, creating a modern, geometric look. The word 'Utilities' is centered over the middle of the 'W'.

Utilities

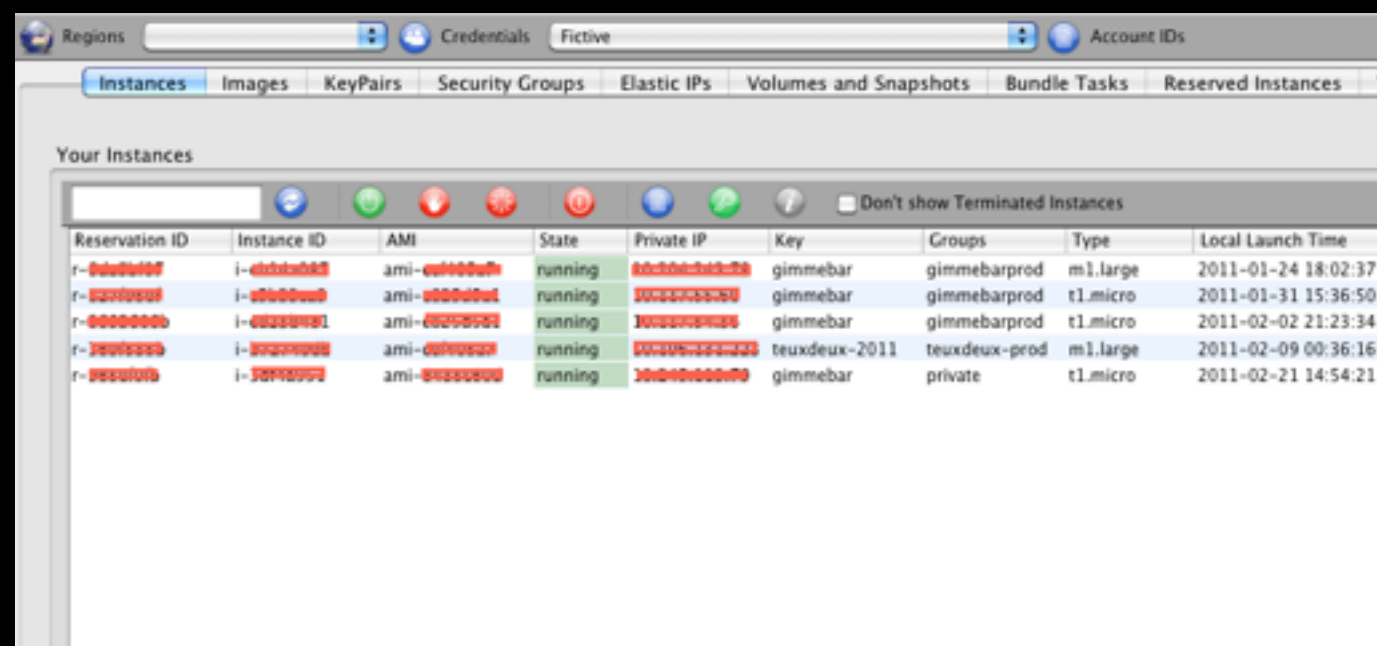
rrdtool

- round-robin
- compare week-over-week, day-over-day
- easy to feed
- not terribly easy to set up
- takes care of the hard parts



Elasticfox

- add-on for FireFox
- a little quirky
- handles most (but not all) of EC2's maintenance functions
- WAY easier than the Java CLI tools



Firebug

- add-on for FireFox
- JavaScript debugger
- DOM inspector
- Network monitor
- Other browsers have similar



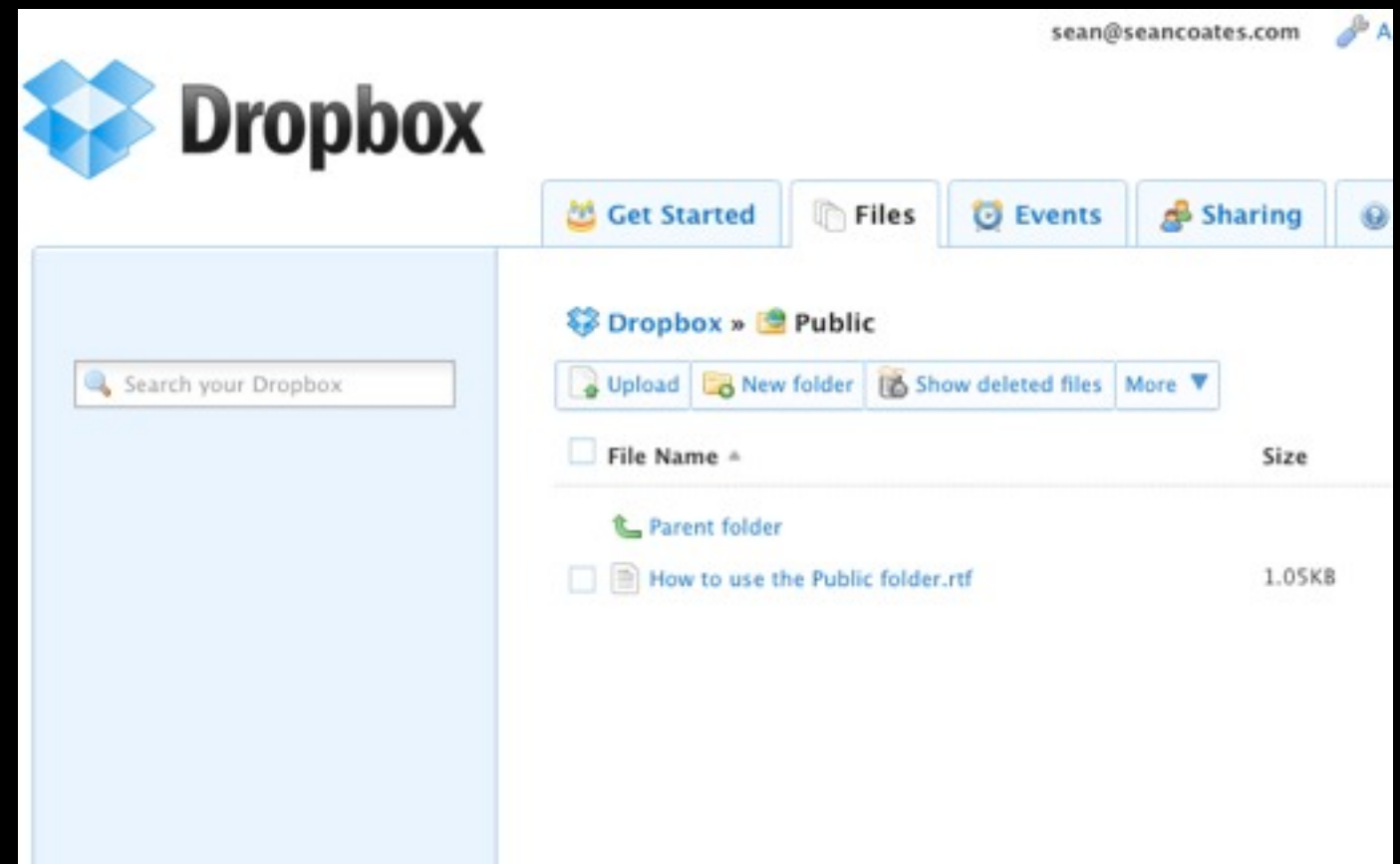
Webshell

- <plug type=self/>
- scripted HTTP using JS
- ... lots of other stuff, but that was another talk (-:

```
http://local.gimmebar.com:8000 > $_.json
{ _apirate: '2011011400'
  , _id: 'ade815b1ff1ebffdb95b5c84e5e7ac38'
  , avatar_asset_id: '6468a1168a3c834e93b69461eb004d3d'
  , dismissed_goodies: true
  , email: 'sean@seancoates.com'
  , featured: 'team'
  , followers:
    [ 'd78290a27b7a7e06a5391e28090000b2'
      , 'fc397c28bb843a34b72a3857af006e7d'
      , '0529f5c0ab9fbddc90dd7d1f1e0c7b65'
      , '0529f5c0ab9fbddc90dd7d1f1e0fa8e2'
      , '14c8e4339e4c8c3873a7cee2d305749d'
      , '0529f5c0ab9fbddc90dd7d1f1e101d74'
      , '32b37ff7c88002d8fd44a11d4b010279'
      , 'd35fa15e0c6e27cfccb2977e0b0074a0'
      , 'd35fa15e0c6e27cfccb2977e0b01f5ec'
      , 'd35fa15e0c6e27cfccb2977e0b01f5ec'
    ]
}
```

Dropbox

- Great for sharing among teams
- Easy sync
- Free (sort of)
- Supports growl



VirtualBox

- Free Virtual Machine
- Ideal for developers, VPN
- Runs our dev hosts very well
- Easy to distribute images



VirtualBox

Download VirtualBox

Here, you will find links to VirtualBox binaries and its source code.

VirtualBox binaries

By downloading, you agree to the terms and conditions of the respective licenses.

- **VirtualBox platform packages.** The binaries are released under the [VirtualBox License](#).
 - **VirtualBox 4.0.4 for Windows hosts** [↗ x86/amd64](#)
 - **VirtualBox 4.0.4 for OS X hosts** [↗ Intel Macs](#)
 - **VirtualBox 4.0.4 for Linux hosts**
 - **VirtualBox 4.0.4 for Solaris hosts** [↗ x86/amd64](#)
- **VirtualBox 4.0.4 Oracle VM VirtualBox Extension Pack** [↗](#)
Support for USB 2.0 devices, VirtualBox RDP and PXE boot

[About](#)
[Screenshots](#)
[Downloads](#)
[Documentation](#)
 [End-user docs](#)
 [Technical docs](#)
[Contribute](#)
[Community](#)

OpenVPN

- Arbitrary virtual networking
- Ties multiple locations together as one network
- Private network for production servers
- Good Mac client: Viscosity



supervisord

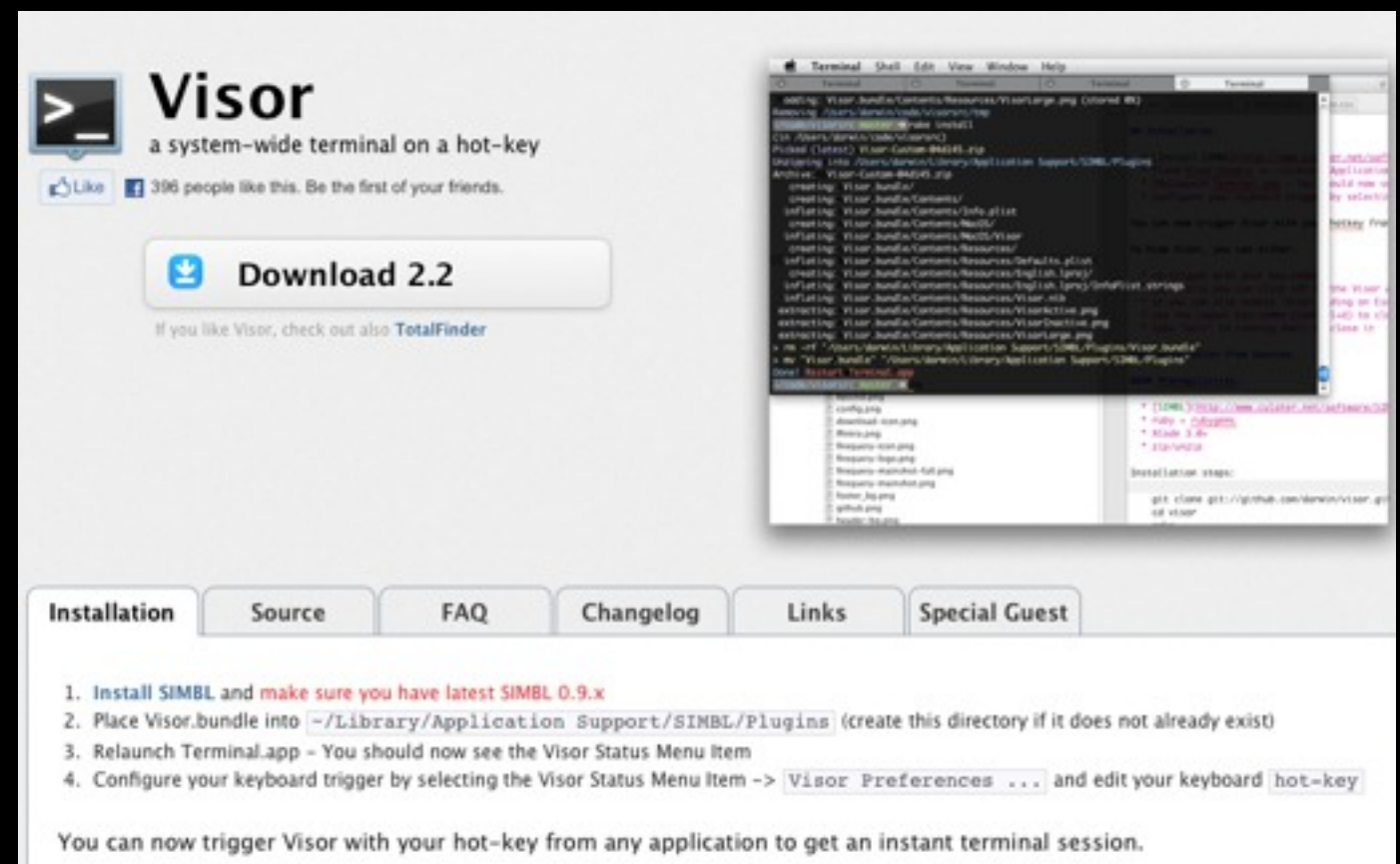
- Dæmonize apps, easily
- (avoid writing dæmon code yourself)
- Handles launch, respawn, concurrency
- Handles logging
- See: <http://phpadvent.org/200915>

netcat

- Turn anything into a server (sort of)
- Useful for testing (like telnet, but without the telnet stuff)
- pbcopy trick: <http://seancoates.com/blogs/remote-pbcopy>
- almost as much of a Swiss-Army knife as SSH

Visor

- System-wide terminal
- Accessible by hotkey
- Great for resident CLI apps (Webshell, REPL, DB client)
- Mac only
- <http://visor.binaryage.com/>



Contact

- I've probably lost my voice by now, so here's my contact info:
 - sean@seancoates.com
 - seancoates.com
 - Work: fictivekin.com, gimmebar.com
 - The Twitter: [@coates](https://twitter.com/coates)