

Moderne Kommandozeilenwerkzeuge

(Alleine Zuhause Edition)



Factory Reset



Sven Guckes

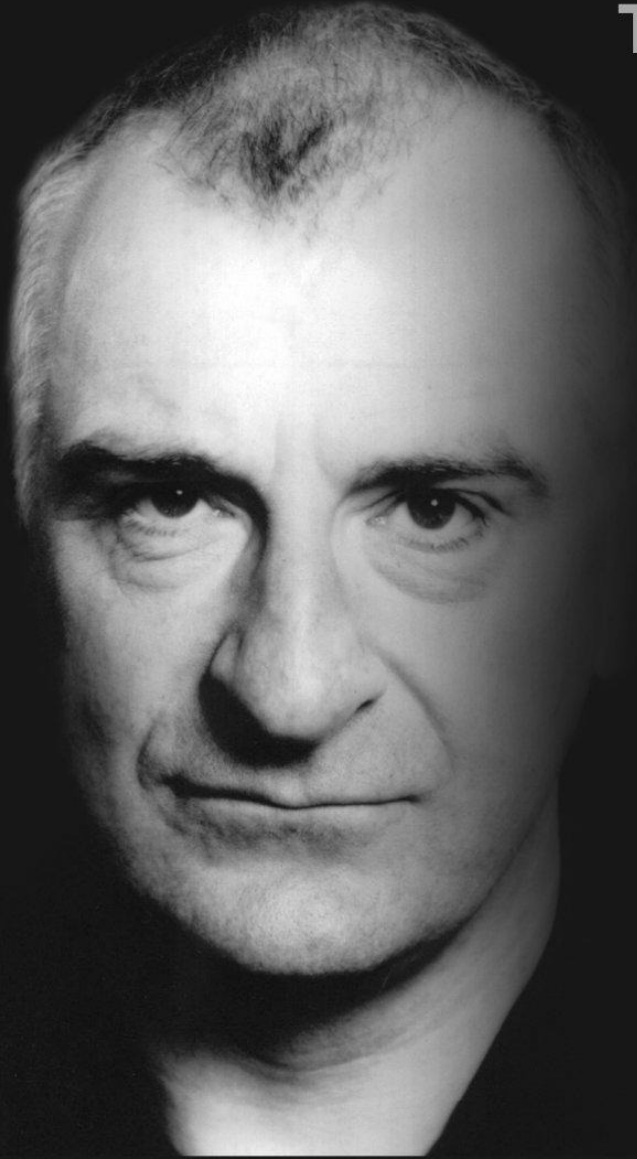
#GPN20

@LEYRER

<https://martin.leyrer.priv.at>

DANKE

- @amilajack
- @jpmens
- @mattiasgeniar
- @johncodezzz
- @ethanscoonover
- @karasuarts
- @stoeps
- @ibraheemdev
- @helpermethod
- "gallo"



Technology is....

1. *Anything that is in the world when you're born is normal and ordinary and is just a natural part of the way the world works.*
2. *Anything that's invented between when you're fifteen and thirty-five is new and exciting and revolution and you can probably get a career in it.*
3. *Anything invented after you're thirty-five is against the natural order of things.*

-Douglas Adams

Rear Admiral Grace Hopper, ph.D.

On the future of data processing, Hopper said the most dangerous phrase a DP manager can use is:

“We’ve always done it that way.”



<https://books.google.at/books?id=3u9H-xL4sZAC&lpg=PA9&vq=%22most%20dangerous%22&hl=de&pg=PA9#v=onepage&q&f=true>

The UNIX philosophy

Doug McIlroy, Bell System Technical Journal 1978

http://emulator.pdp-11.org.ru/misc/1978.07_-_Bell_System_Technical_Journal.pdf

Style

A number of maxims have gained currency among the builders and users of the UNIX system to explain and promote its characteristic style:

- (i) Make each program do one thing well. To do a new job, build afresh rather than complicate old programs by adding new "features."
- (ii) Expect the output of every program to become the input to another, as yet unknown, program. Don't clutter output with extraneous information. Avoid stringently columnar or binary input formats. Don't insist on interactive input.
- (iii) Design and build software, even operating systems, to be tried early, ideally within weeks. Don't hesitate to throw away the clumsy parts and rebuild them.
- (iv) Use tools in preference to unskilled help to lighten a

The UNIX philosophy

Erstelle Programme so, dass sie eine Aufgabe richtig erfüllen. Wenn es eine neue Aufgabe gibt, beginne ein neues Programm anstatt das andere zu verkomplizieren.

The UNIX philosophy

Gehe davon aus, dass die Ausgaben eines Programms die Eingabe eines anderen sein können, selbst wenn Du dieses noch nicht kennst. Programme sollten nicht schwafeln, halte die Ausgaben kurz und einfach verarbeitbar.

The UNIX philosophy

Lass Computer-Tools die Arbeit für Dich machen,
nicht die Hilfskräfte.

The UNIX philosophy

Peter H. Salus in “A Quarter-Century of Unix” (1994):

- Write programs that do one thing and do it well.
- Write programs to work together.
- Write programs to handle text streams, because that is a universal interface.

2018

dig

tracepath

mtr

oping

iproute2

tmux

zsh

ranger

HTTPIe

aria2c

jq

icdiff

colordiff

ripgrep

fd

z

exa

fzf

pv

progress

2019

hub

tig

lynis

top Familie

nethogs

socket statistics

dstat

saidar

nmon

kubetail

colorex

lolcat

gulasch

HTTP Prompt

XMLStarlet

Ultimate Plumber

reptyr

ttyd

WTF

no more secrets

2022

- xsv
- gron
- jc
- fx
- bat
- ccat
- delta
- diff-so-fancy
- *δyff*
- skim
- choose
- sd
- map
- xonsh
- crush
- xxh
- starship
- Oh My Posh
- q
- gping
- ioping
- broot
- ncdu
- duf
- dust
- dutree
- lsd
- goto
- mcfly
- hyperfine
- procs
- bottom
- btop++
- musikcube
- googler
- viu
- hexyl
- glow
- falsisign
- fuck
- oh-heck
- usbskill

Moderne Kommandozeilenwerkzeuge

(Alleine Zuhause Edition)



Factory Reset

XSV

- „jq“ for CSV files
- index, slice, analyze, split and join CVS files
- <https://github.com/BurntSushi/xsv>



XSV

```
xsv search -s BEZIRK '23' BAUMKATOGD.csv \  
| xsv select DATENFUEHRUNG,BAUMNUMMER,GATTUNG_ART \  
| xsv table
```

DATENFUEHRUNG	BAUMNUMMER	GATTUNG_ART
magistrat	N5004	Aesculus hippocastanum (Rosskastanie)
magistrat	N5020	Aesculus x carnea (Rotblühende Kastanie)
magistrat	16	Prunus avium (Vogelkirsche)
magistrat	1	Acer platanoides 'Crimson King' (Blutahorn)

gron

- makes JSON greppable
- transforms JSON into discrete assignments
- for APIs with terrible documentation
- <https://github.com/tomnomnom/gron>



gron

```
gron 'https://cfp.gulas.ch/gpn20/schedule/export/schedule.json' | grep -A 4 Leyrer | less
```

```
json.schedule.conference.days[0].rooms.Medientheater[1].persons[0].public_name = "Leyrer";  
json.schedule.conference.days[0].rooms.Medientheater[1].recording_license = "";  
json.schedule.conference.days[0].rooms.Medientheater[1].room = "Medientheater";  
json.schedule.conference.days[0].rooms.Medientheater[1].slug = "gpn20-12-die-unterhaltsamsten-  
sterreichischen-it-fails-der-letzten-jahre-und-was-wir-daraus-lernen-knnen";  
json.schedule.conference.days[0].rooms.Medientheater[1].start = "19:00";  
--  
json.schedule.conference.days[1].rooms.Medientheater[3].persons[0].public_name = "Leyrer";  
json.schedule.conference.days[1].rooms.Medientheater[3].recording_license = "";  
json.schedule.conference.days[1].rooms.Medientheater[3].room = "Medientheater";  
json.schedule.conference.days[1].rooms.Medientheater[3].slug = "gpn20-10--eu-digital-covid-certificate-  
keine-schwarze-magie";
```


Helft mir !!!

- Reicht Talks und Workshops ein
- Ich helfe gerne
 - Themenfindung
 - CfP Formulierung
 - Präsentation bauen
 - Präsentation Testlauf
- haecksen.org helfen

Your proposals

Title	State
Von Null auf root in 120 Minuten - Einführung ins Website Hacking	✓ confirmed
Besser leben mit SSH	✓ confirmed
Moderne Linux Kommandozeilenwerkzeuge - Edition "Allein zu Haus"	✓ confirmed
"EU Digital COVID Certificate" - Keine schwarze Magie	✓ confirmed
Die unterhaltsamsten österreichischen IT Fails der letzten Jahre und was wir daraus lernen können	✓ confirmed

txt | jc | {}

Had JSON been around when I was born in the 1970's Ken Thompson and Dennis Ritchie may very well have embraced it as a recommended output format [...]

To that end, I argue that linux and all of its supporting GNU and non-GNU utilities should offer JSON output options.

jc

- JSONifies the output of many CLI tools
- allows further processing with tools like jq
- <https://github.com/kellyjonbrazil/jc>



jc

```
dig cfp.gulas.ch | jc --dig
```

```
[{"answer":  
[{"name":"cfp.gulas.ch.", "class":"IN", "type":"A", "ttl":60, "data":"23.88.121.56"}]]}]
```

```
dig cfp.gulas.ch \  
  | jc --dig \  
  | jq -r '.[].answer[].data'
```

```
23.88.121.56
```

fx

- JSON viewer
- works as a reducer or an interactive viewer
- themable
- <https://github.com/antonmedv/fx>



fx



fx

```
{  
  "name": "Anton Medvedev",  
  "age": 30,  
  "married": true,  
  "car": null  
}
```

```
{  
  "name": "Anton Medvedev",  
  "age": 30,  
  "married": true,  
  "car": null  
}
```

```
{  
  "name": "Anton Medvedev",  
  "age": 30,  
  "married": true,  
  "car": null  
}
```

```
{  
  "name": "Anton Medvedev",  
  "age": 30,  
  "married": true,  
  "car": null  
}
```

```
{  
  "name": "Anton Medvedev",  
  "age": 30,  
  "married": true,  
  "car": null  
}
```

```
{  
  "name": "Anton Medvedev",  
  "age": 30,  
  "married": true,  
  "car": null  
}
```

```
{  
  "name": "Anton Medvedev",  
  "age": 30,  
  "married": true,  
  "car": null  
}
```

```
{  
  "name": "Anton Medvedev",  
  "age": 30,  
  "married": true,  
  "car": null  
}
```

```
{  
  "name": "Anton Medvedev",  
  "age": 30,  
  "married": true,  
  "car": null  
}
```

bat

- cat clone with syntax highlighting
- git integration
- shows non-printable characters
- <https://github.com/sharkdp/bat>



bat

► bat src/index.html

File **src/index.html**

```
1  <!DOCTYPE html>
2  <html lang="en">
3      <head>
4          <meta charset="utf-8">
5  ~   <title>a changed title</title>
6      <link rel="stylesheet" href="style.css">
7  </head>
8  <body>
9  +   New lines that have been
10 +   added since last commit.
11   </body>
12 </html>
```


ccat

- colorizing cat
- JavaScript, Java, Ruby, Python, Go, C, JSON
- tab-completion in zsh
- <https://github.com/owenthereal/ccat>



ccat

```
$ ccat --bg=dark FILE1 FILE2 ... # dark background
$ ccat -G String="_darkblue_" -G Plaintext="darkred" FILE # set color codes
$ ccat --palette # show palette
$ ccat # read from standard input
$ curl https://raw.githubusercontent.com/jingweno/ccat/master/main.go | ccat`,
    Run: ccatCmd.Run,
}

usageTempl := `{{ $cmd := . }}

Usage:
  {{.UseLine}}

Flags:
  {{.LocalFlags.FlagUsages}}
Using color is auto both by default and with --color=auto. With --color=auto,
ccat emits color codes only when standard output is connected to a terminal.
Color codes can be changed with -G KEY=VALUE. List of color codes can
be found with --palette.

Examples:
  {{ .Example }}`
rootCmd.SetUsageTemplate(usageTempl)

rootCmd.PersistentFlags().StringVarP(&ccatCmd.BG, "bg", "", "light", `set to "light" or "dark" depending on the terminal's background`)
rootCmd.PersistentFlags().StringVarP(&ccatCmd.Color, "color", "C", "auto", `colorize the output; value can be "never", "always" or "auto"`)
rootCmd.PersistentFlags().VarP(&ccatCmd.ColorCodes, "color-code", "G", `set color codes`)
rootCmd.PersistentFlags().BoolVarP(&ccatCmd.ShowPalette, "palette", "", false, `show color palettes`)
rootCmd.PersistentFlags().BoolVarP(&ccatCmd.ShowVersion, "version", "v", false, `show version`)

rootCmd.Execute()
}
```

📄 → ccat git:(master) x

delta

- language syntax highlighting
- word-level diff highlighting
- side-by-side view & line numbering
- <https://github.com/dandavison/delta>



delta

src/core/instance/render-helpers/resolve-scoped-slots.js

```
2
3 export function resolveScopedSlots (
4   fns: ScopedSlotsData, // see flow/vnode
5
6   hasDynamicKeys: boolean,
7   contentHashKey: number,
8   res?: Object
9 ): { [key: string]: Function, $stable: boolean } {
10   res = res || { $stable: !hasDynamicKeys }
11   for (let i = 0; i < fns.length; i++) {
12     const slot = fns[i]
13     if (Array.isArray(slot)) {
14       resolveScopedSlots(slot, hasDynamicKeys, null, res)
15     } else if (slot) {
16       // marker for reverse proxying v-slot without scope on this.$slots
17       if (slot.proxy) {
```

```
export function resolveScopedSlots (
```

```
20   }
21   }
22   if (contentHashKey) {
23     res.$key = contentHashKey
24   }
25   return res
26 }
```

```
2
3 export function resolveScopedSlots (
4   fns: ScopedSlotsData, // see flow/vnode
5   res?: Object,
6   // the following are added in 2.6
7   hasDynamicKeys?: boolean,
8   contentHashKey?: number
9 ): { [key: string]: Function, $stable: boolean } {
10   res = res || { $stable: !hasDynamicKeys }
11   for (let i = 0; i < fns.length; i++) {
12     const slot = fns[i]
13     if (Array.isArray(slot)) {
14       resolveScopedSlots(slot, res, hasDynamicKeys)
15     } else if (slot) {
16       // marker for reverse proxying v-slot without scope on this.$slots
17       if (slot.proxy) {
```

```
21   }
22   }
23   if (contentHashKey) {
24     (res: any).$key = contentHashKey
25   }
26   return res
27 }
```

diff-so-fancy

- human readable diffs
- spot defects faster
- simplifys git header chunks
- <https://github.com/so-fancy/diff-so-fancy>



diff-so-fancy

```
diff --git a/libs/header_clean/header_clean.pl b/libs/header_clean/header_clean.pl
index 7c544c9..27c841d 100755
--- a/libs/header_clean/header_clean.pl
+++ b/libs/header_clean/header_clean.pl
@@ -2,12 +2,11 @@
```

```
    use strict;
    use warnings;
-   use File::Basename;
+   use File::Core::Basename;

    my $remove_file_add_header    = 1;
-   my $remove_file_delete_header = 1;
    my $clean_permission_changes  = 1;
-   my $change_hunk_indicators    = 1;
+   my $change_chunk_indicators   = 1;
```

```
#####
```

```
@@ -35,9 +34,7 @@ for (my $i = 0; $i <= $#input; $i++) {
    # Find the second file on the next line: +++ b/README.md
    my $next = $input[++$i];
    $next =~ /^$ansi_sequence_regex\+\+\+ (\w\/)?(.*?) (\e|$)/;
-   if ($1) {
-       print $1; # Print out whatever color we're using
-   }
+   print $1; # Print out whatever color we're using
+   $file_2 = $5;
```

```
    # If they're the same it's a modify
```

```
diff --git a/package.json b/package.json
index 444c17a..f9db603 100644
--- a/package.json
```

```
modified: libs/header_clean/header_clean.pl
```

```
@ header_clean.pl:5 @
```

```
use strict;
use warnings;
use File::Basename;
use File::Core::Basename;
```

```
my $remove_file_add_header    = 1;
my $remove_file_delete_header = 1;
my $clean_permission_changes  = 1;
my $change_hunk_indicators    = 1;
my $change_chunk_indicators   = 1;
```

```
#####
```

```
@ header_clean.pl:37 @ for (my $i = 0; $i <= $#input; $i++) {
    # Find the second file on the next line: +++ b/README.md
    my $next = $input[++$i];
    $next =~ /^$ansi_sequence_regex\+\+\+ (\w\/)?(.*?) (\e|$)/;
    if ($1) {
        print $1; # Print out whatever color we're using
    }
    print $1; # Print out whatever color we're using
    $file_2 = $5;
```

```
    # If they're the same it's a modify
```

```
modified: package.json
```

```
@ package.json:3 @
```

dyff /'dyf/

- diff tool for YAML files
- ... and sometimes JSON
- Input files
 - local files (filesystem path)
 - remote files (URI)
 - standard input stream (using -).
- <https://github.com/homeport/dyff>



$$\delta y f f / ' d y f /$$
[illegible]

manifest_version

± value change

- v1.10.0

+ v1.20.0

```
instance_groups.database.jobs.mysql.properties.cf_mysql.mysql.port
```

± value change

+ 13306

```
instance_groups.diego-api.jobs.bbs.properties.diego.bbs
```

- two map entries removed:

etcd:

```
machines: []
```

```
ca cert: nil
```

```
client cert: nil
```

```
client key: nil
```

Locket:

```
api location: locket.service.cf.internal:8891
```

```
instance_groups.uaa.jobs.uaa.properties.login.saml
```

- two map entries removed:

```
serviceProviderKey: ((uaa_login_saml.private_key))
```

```
serviceProviderCertificate: ((uaa_login_saml.certificate))
```

+ two map entries added:

```
activeKeyId: key-1
```

```
keys:
```

```
key: ((uaa_login_saml.private_key))
```

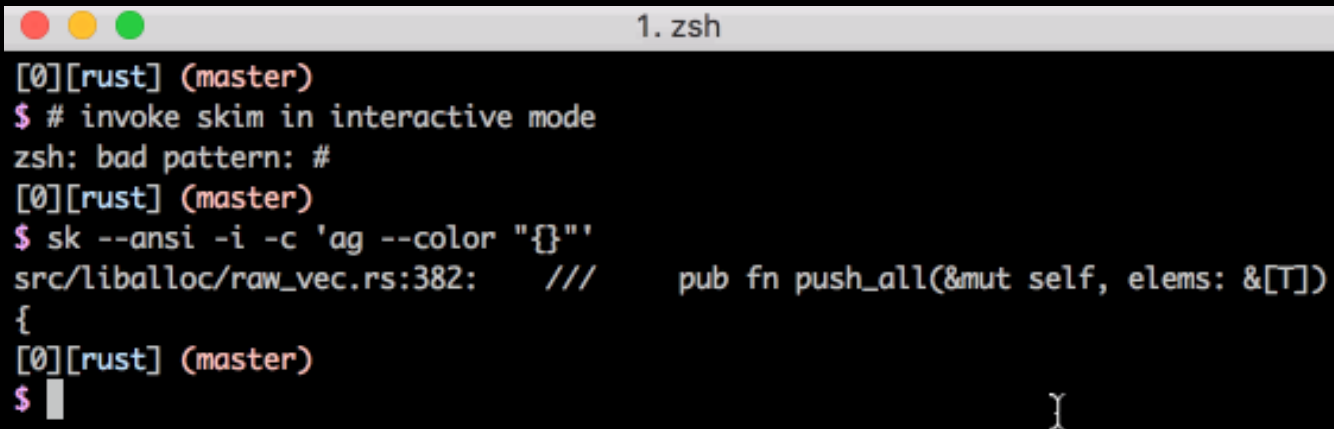
```
certificate: ((uaa_login_saml.certificate))
```


skim

- general fuzzy finder
- general filter
- interactive interface
- <https://github.com/lotabout/skim>



skim



```
1. zsh
[0][rust] (master)
$ # invoke skim in interactive mode
zsh: bad pattern: #
[0][rust] (master)
$ sk --ansi -i -c 'ag --color "{}"'
src/liballoc/raw_vec.rs:382:    /// pub fn push_all(&mut self, elems: &[T])
{
[0][rust] (master)
$ █
```

choose

- human-friendly and fast
- alternative to cut and (sometimes) awk
- regular expression field separators
- <https://github.com/theyrangeary/choose>



sd

- s[earch] & d[isplace]
- uses regex
- smart, common-sense defaults
- <https://github.com/chmln/sd>



sd Examples

Command	sd	sed
replace all occurrences	<code>sd before after</code>	<code>sed s/before/after/g</code>
replace newline/commas	<code>sd '\n' ','</code>	<code>sed ':a;N;\$!ba;s/\n/,/g'</code>
file in place replacement	<code>sd before after file.txt</code>	<code>sed -i -e 's/before/after/g' file.txt</code>

map

- process each line from stdin with a command of your choice
- designed as a filter
- <https://github.com/soveran/map>



map

```
ls | ./map f 'file "$f"'
```

BAUMKATOGD.csv: CSV text

baum.txt.csv: UTF-8 Unicode text

ccat: ELF 64-bit LSB executable, x86-64, version 1 (SYSV), statically linked, Go...

gpn20 - Moderne Kommandozeilentools.odp: OpenDocument Presentation

liste.txt: ASCII text, with very long lines

xonsh

- Python-powered
- shell language and command prompt
- Python 3.6+ with additional shell primitives
- <https://xon.sh/>



xonsh

```
$ len($(curl -L https://xon.sh))
```

```
$ for filename in `.*`:  
    print(filename)  
    du -sh @(filename)
```

crush

- a command line shell that is also a modern programming language
- type system, closures, lexical scoping
- <https://github.com/liljencrantz/crush>



crush Examples

```
crush# ll | sort ^size
```

user	size	modified			type	file
fox	31	2019-10-03 13:43:12	+0200		file	.gitignore
fox	75	2020-03-07 17:09:15	+0100		file	build.rs
fox	491	2020-03-07 23:50:08	+0100		file	Cargo.toml
fox	711	2019-10-03 14:19:46	+0200		file	crush.iml
...						

```
crush# ll | where {type == "directory"}
```

user	size	modified			type	file
fox	4_096	2019-11-22 21:56:30	+0100		directory	target
fox	4_096	2020-02-22 11:50:12	+0100		directory	tests
fox	4_096	2020-03-16 14:11:39	+0100		directory	.idea
fox	4_096	2020-02-15 00:12:18	+0100		directory	example_data
fox	4_096	2020-03-14 17:34:39	+0100		directory	src
fox	4_096	2020-03-14 19:44:54	+0100		directory	.git

xxh

- portable shells and plugins
- your favorite shell wherever you go
- without root access/installs
- <https://github.com/xxh/xxh>



xxh

```
start ~> ssh ubuntu
```

```
ubuntu# zsh
```

```
command not found
```

```
ubuntu# fish
```

```
command not found
```

```
ubuntu# xonsh
```

```
command not found
```

```
start ~> xxh ubuntu +s fish # or zsh, xonsh, bash, osquery
```

```
Welcome to fish
```

```
ubuntu ~> █
```

starship

- customizable prompt for any shell
- compatibility first
- Even for powershell
- <https://starship.rs/>





```
rocket on ↻ master via 🦀 v1.36.0  
> sudo -s
```

```
root in rocket on ↻ master via 🦀 v1.36.0  
> exit
```

```
rocket on ↻ master via 🦀 v1.36.0 took 4s  
> vim ~/.config/starship.toml
```

```
rocket on ↻ master via R v1.36.0 took 5s  
> █
```

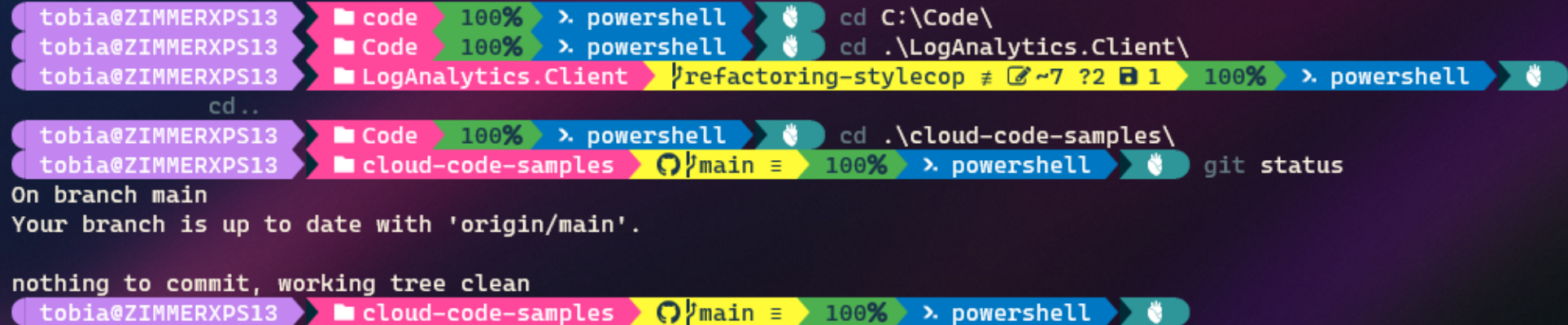
Oh My Posh

- prompt theme engine for any shell
- enables you to use the full color set
- customizable, portable
- <https://ohmyposh.dev/>



Windows PowerShell

Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell <https://aka.ms/pscore6>The image shows a Windows PowerShell terminal window with a dark background. The title bar at the top says "Windows PowerShell" and has standard window controls. The terminal output shows a series of commands and their results, each preceded by a prompt "tobia@ZIMMERXPS13". The commands are: 1. "cd C:\Code\" with a progress bar for "code" at 100%. 2. "cd .\LogAnalytics.Client\" with a progress bar for "Code" at 100%. 3. "refactoring-stylecop --7 2 1" with a progress bar for "LogAnalytics.Client" at 100%. 4. "cd .." with no progress bar. 5. "cd .\cloud-code-samples\" with a progress bar for "Code" at 100%. 6. "git status" with a progress bar for "cloud-code-samples" at 100%. The output for "git status" is "On branch main" and "Your branch is up to date with 'origin/main'.". 7. "git status" again, with output "nothing to commit, working tree clean". Each command line is followed by a blue arrow pointing right, indicating the command is complete. The progress bars are colored: pink for folders, green for percentages, and blue for the command name. The output for "git status" is shown on two lines.

```
tobia@ZIMMERXPS13 > cd C:\Code\
```

```
tobia@ZIMMERXPS13 > cd .\LogAnalytics.Client\
```

```
tobia@ZIMMERXPS13 > refactoring-stylecop --7 2 1
```

cd ..

```
tobia@ZIMMERXPS13 > cd .\cloud-code-samples\
```

```
tobia@ZIMMERXPS13 > git status
```

On branch main

Your branch is up to date with 'origin/main'.

nothing to commit, working tree clean

```
tobia@ZIMMERXPS13 >
```

q

- DNS client
- supports UDP, TCP, DoT, DoH, DoQ, ODoH
- colourful
- <https://github.com/natesales/q>



q



~ ▲ q A AAAA MX natesales.net

natesales.net. 24h0m0s A 23.141.112.33

natesales.net. 24h0m0s AAAA 2602:809:3009::33

natesales.net. 24h0m0s MX 10 mail.natesales.net.

q Examples

```
q --dnssec A AAAA example.com  
@quic://dns.emeraldonion.org:8853  
example.com. 23h57m50s A 93.184.216.34 example.com.  
23h57m50s RRSIG A 8 2 86400 20210416051959  
20210325232537 45150 example.com. N40WYg7CBxSc...
```

```
q --dnssec A example.com @tls://dns.emeraldonion.org  
example.com. 23h57m40s A 93.184.216.34 example.com.  
23h57m40s RRSIG A 8 2 86400 20210416051959  
20210325232537 45150 example.com. N40WYg7CBxSc...
```

gping

- ping, but with a graph
 - graph the execution time for commands
 - custom colours
-
- <https://github.com/orf/gping>



~/PycharmProjects/github/orf/gping multihost*

> gping google.com facebook.com

ioping

- monitor I/O latency in real time
- shows disk latency in the same way as ping shows network latency
- <https://github.com/koct9i/ioping>



disk I/O latency

```
$ ioping .  
4096 bytes from . (ext4 /dev/sda3): request=1 time=0.2 ms  
4096 bytes from . (ext4 /dev/sda3): request=2 time=0.2 ms  
4096 bytes from . (ext4 /dev/sda3): request=3 time=0.3 ms  
4096 bytes from . (ext4 /dev/sda3): request=4 time=12.7 ms  
4096 bytes from . (ext4 /dev/sda3): request=5 time=0.3 ms  
^C  
--- . (ext4 /dev/sda3) ioping statistics ---  
5 requests completed in 4794.0 ms, 364 iops, 1.4 MiB/s  
min/avg/max/mdev = 0.2/2.8/12.7/5.0 ms
```


disk seek rate (iops, avg)

```
$ ioping -R /dev/sda
```

```
--- /dev/sda (device 465.8 GiB) ioping statistics ---
```

```
186 requests completed in 3004.6 ms, 62 iops, 0.2  
MiB/s
```

```
min/avg/max/mdev = 6.4/16.0/26.8/4.7 ms
```

disk sequential speed (MiB/s)

```
$ ioping -RL /dev/sda
```

```
--- /dev/sda (device 465.8 GiB) ioping  
statistics ---
```

```
837 requests completed in 3004.1 ms, 292 iops,  
72.9 MiB/s
```

```
min/avg/max/mdev = 2.0/3.4/28.9/2.0 ms
```

broot

- navigate directories
- one screen hierachy view
- auto-cutoff
- <https://github.com/Canop/broot>



/home/dys/dev/miaou/src

```
405K  └─ main-js
      6K   └─ groumf.js
      5K   └─ hu.js
          └─ 51 unlisted

327K  └─ main-scss
      192  └─ autocomplete.scss
      3149 └─ bubble.scss
      24K  └─ chat.scss
          └─ 25 unlisted

65K   └─ page-js
      1233 └─ auths.js
      1444 └─ chat.js
      1155 └─ chat.mob.js
          └─ 10 unlisted

176K  └─ page-scss
      6K   └─ intro.scss
      4092 └─ pad.mob.scss
      341  └─ room.scss
      197  └─ rooms.mob.scss
      590  └─ readme.md
      6M   └─ rsc
          16K └─ adM.png
          16K └─ adMw.png
          10K └─ ajax-loader-bar.gif
              └─ 26 unlisted

8K    └─ sw-js
      3840 └─ sw-notifs.js
```

Hit **esc** to go back, **enter** to go up, **?** for help, or a few letters to search

h:n gi:y

ncdu

- disk usage analyzer with an ncurses interface
- find space hogs
- needs minimal POSIX-like env. with ncurses
- <https://dev.yorhel.nl/ncdu>



ncdu 1.13 ~ Use the arrow keys to navigate, press ? for help

--- /usr/portage/distfiles -----

../

239.4 MiB [#####] firefox-57.0.1.source.tar.xz

239.1 MiB [#####] **firefox-57.0.source.tar.xz**

238.6 MiB [#####] firefox-57.0.4.source.tar.xz

153.0 MiB [#####] rust-1.22.0-x86_64-unknown-linux-gnu.tar.gz

96.1 MiB [####] linux-4.14.tar.xz

83.3 MiB [Confirm delete]

78.3 MiB [Are you sure you want to delete "firefox...urce.tar.xz"?

78.2 MiB [

78.1 MiB [

72.6 MiB [yes ☒ no don't ask me again

57.2 MiB [

55.6 MiB [##] wxPython-src-3.0.2.0.tar.bz2

52.8 MiB [##] go-linux-amd64-bootstrap-1.8.tbz

33.8 MiB [#] cairo-1.14.8.tar.xz

29.4 MiB [#] qemu-2.10.1.tar.bz2

27.8 MiB [#] binutils-2.29.1.tar.bz2

27.5 MiB [#] wireshark-2.4.3.tar.xz

24.5 MiB [#] graphviz-2.40.1.tar.gz

22.3 MiB [] llvm-5.0.1.src.tar.xz

Total disk usage: 2.7 GiB Apparent size: 2.7 GiB Items: 503

duf

- disk Usage/Free Utility
- user-friendly, colorful output
- can conveniently output JSON
- <https://github.com/muesli/duf>



4 local devices							
MOUNTED ON	SIZE	USED	AVAIL	USE%		TYPE	FILESYSTEM
/	80.0G	70.3G	8.3G	[#####...]	87.9%	btrfs	/dev/bender/root
/boot	511.0M	74.5M	436.4M	[##.....]	14.6%	vfat	/dev/sda1
/home	128.0G	118.7G	8.1G	[#####..]	92.7%	btrfs	/dev/bender/home
/media/hole	7.2T	5.5T	1.4T	[#####.....]	75.7%	ext4	/dev/mapper/hole

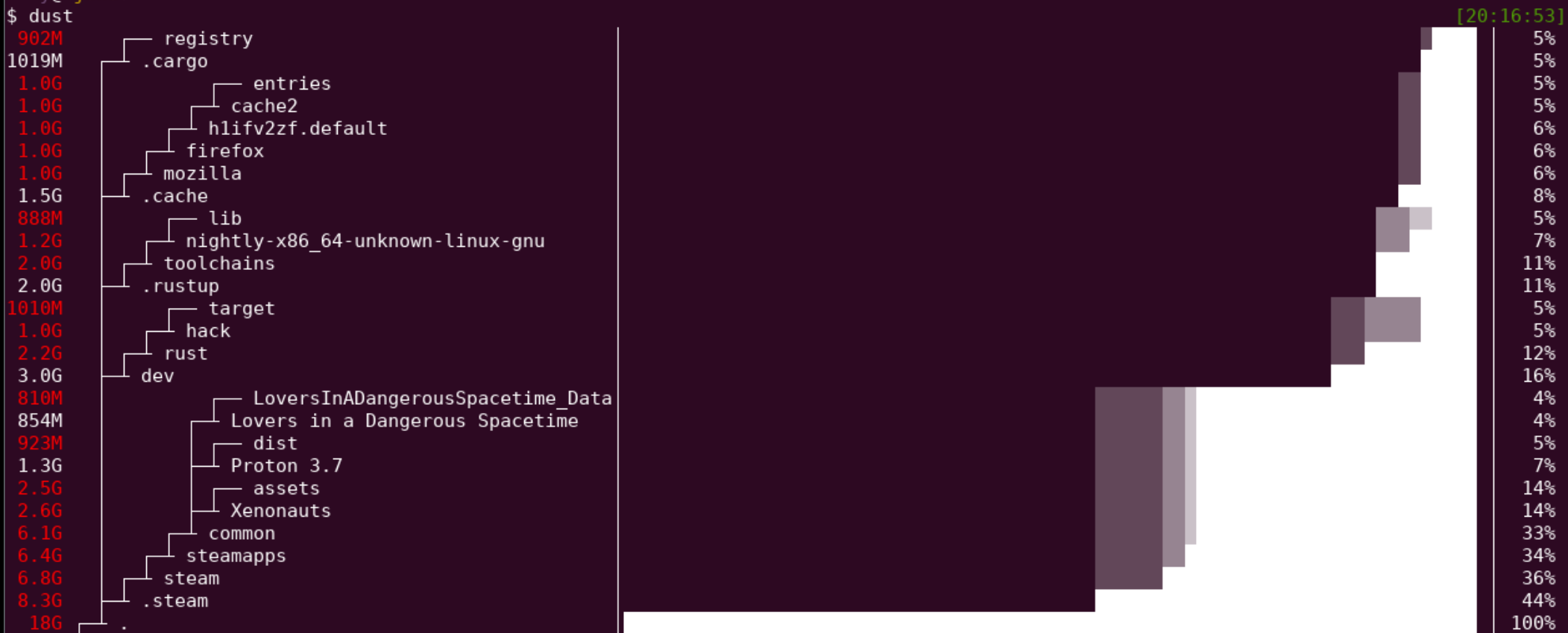
9 network devices							
MOUNTED ON	SIZE	USED	AVAIL	USE%		TYPE	FILESYSTEM
/media/nibbler/books	21.8T	14.2T	7.6T	[#####.....]	65.0%	cifs	//nibbler/books
/media/nibbler/development	21.8T	14.2T	7.6T	[#####.....]	65.0%	cifs	//nibbler/development
/media/nibbler/documents	21.8T	14.2T	7.6T	[#####.....]	65.0%	cifs	//nibbler/documents
/media/nibbler/downloads	21.8T	14.2T	7.6T	[#####.....]	65.0%	cifs	//nibbler/downloads
/media/nibbler/music	21.8T	14.2T	7.6T	[#####.....]	65.0%	cifs	//nibbler/music
/media/nibbler/photo	21.8T	14.2T	7.6T	[#####.....]	65.0%	cifs	//nibbler/photo
/media/nibbler/recording	21.8T	14.2T	7.6T	[#####.....]	65.0%	cifs	//nibbler/recording
/media/nibbler/transfer	21.8T	14.2T	7.6T	[#####.....]	65.0%	cifs	//nibbler/transfer
/media/nibbler/video	21.8T	14.2T	7.6T	[#####.....]	65.0%	cifs	//nibbler/video

6 special devices						
MOUNTED ON	SIZE	USED	AVAIL	USE%		FILESYSTEM
/dev	15.6G	0B	15.6G			devtmpfs
/dev/shm	15.6G	1.2G	14.4G	[#.....]	7.9%	tmpfs
/run	6.2G	10.0M	6.2G	[.....]	0.2%	tmpfs
/run/user/1000	3.1G	148.0K	3.1G	[.....]	0.0%	tmpfs
/sys/fs/cgroup	4.0M	0B	4.0M			tmpfs
/tmp	15.6G	2.2M	15.6G	[.....]	0.0%	tmpfs

dust

- like du but more intuitive
- will print max. 1x 'Did not have permissions'
- largest subdirectories will be colored
- <https://github.com/bootandy/dust>





dutoe

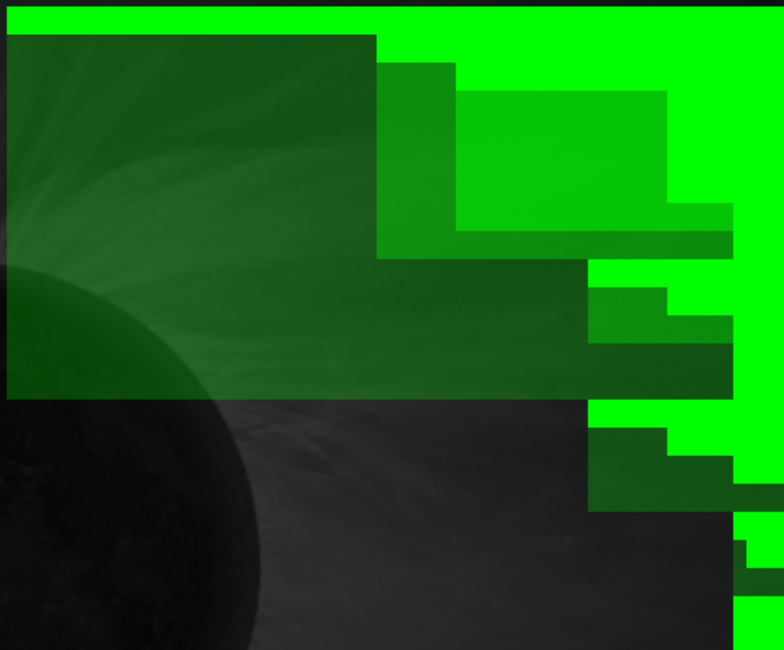
- coloured file system tree
- allows for comparison of different directories
- aggregates small files
- <https://github.com/nachoparker/dutoe>



```

└─ dir1
   └─ test3
      └─ test2
         └─ test2
            ├── file1
            └── file2
         └─ file1
            ├── file1
            └── file2
      └─ file1
   └─ test2
      ├── file1
      └── file2
   └─ file1
   └─ file2
└─ dir2
   ├── file1
   ├── file2
   └─ dir
└─ dir3
   ├── file1
   └─ link
file1
file_noperms
link_orphan
link
file_sparse
dir_other_writable
.file_hidden
file.avi
file.exe
file.mp3
file.tar
dir_noperms
file_special_á
file_special_消
picture.png
fifo
block_dev
char_dev

```

[illegible]

lsd

- ls deluxe
- colors, icons, tree-view
- more formatting options
- <https://github.com/Peltoche/lsd>



```

~ Code > Perso > lsd > master 1↑ % *prodild prod lsd -l target/debug
drwxr-xr-x peltoche users - Thu Dec 13 12:27:23 2018 build
drwxr-xr-x peltoche users - Thu Dec 13 15:28:44 2018 deps
drwxr-xr-x peltoche users - Thu Dec 13 10:51:57 2018 examples
drwxr-xr-x peltoche users - Thu Dec 13 14:58:34 2018 incremental
drwxr-xr-x peltoche users - Thu Dec 13 10:51:57 2018 native
.rw-r--r-- peltoche users 879 B Thu Dec 13 14:29:26 2018 liblsd.d
.rw-r--r-- peltoche users 0 B Thu Dec 13 15:28:44 2018 liblsd.rmeta
.rwxr-xr-x peltoche users 11.7 MB Thu Dec 13 14:50:47 2018 lsd
.rwxr-xr-x peltoche users 9.2 MB Thu Dec 13 15:28:41 2018 lsd-e1fd8883f3461759
.rw-r--r-- peltoche users 887 B Thu Dec 13 15:27:01 2018 lsd-e1fd8883f3461759.d
.rw-r--r-- peltoche users 870 B Thu Dec 13 10:52:06 2018 lsd.d

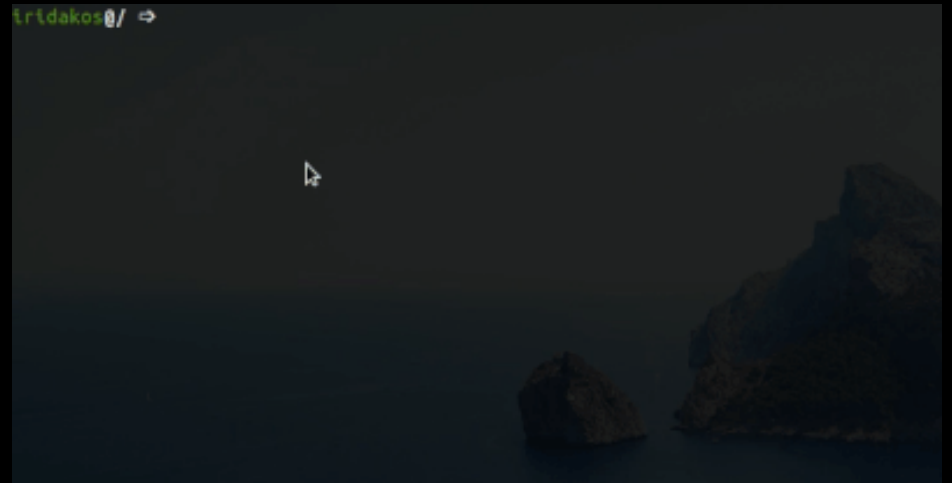
```

goto

- navigate to aliased directories
- supports auto-completion
- satisfy fax fans around the world
- <https://github.com/iridakos/goto>



```
$ goto <tab>  
bc /etc/bash_completion.d  
dev /home/iridakos/development  
rubies /home/iridakos/.rvm/rubies
```



mcfly

- replaces the ctrl-r shell history search
- current directory matters
- in the context of recently executed commands
- <https://github.com/cantino/mcfly>



```
$ git
```

```
git st
```

```
git add -p
```

```
git push origin head --tags
```

```
git add -A
```

```
git ci --amend
```

```
git push
```

```
git pull
```

```
git push origin head -u -f
```

```
git add README.md
```

```
git add Cargo.toml
```

hyperfine

- command-line benchmarking tool
- statistical analysis across multiple runs
- support for arbitrary shell commands.
- <https://github.com/sharkdp/hyperfine>



```
► hyperfine --warmup 3 'fd -e jpg -uu' 'find -iname "*.jpg"'
```

```
Benchmark #1: fd -e jpg -uu
```

```
Time (mean ± σ):      329.5 ms ±   1.9 ms    [User: 1.019 s, System: 1.433 s]
```

```
Range (min ... max):   326.6 ms ... 333.6 ms    10 runs
```

```
Benchmark #2: find -iname "*.jpg"
```

```
Time (mean ± σ):      1.253 s ±  0.016 s    [User: 461.2 ms, System: 777.0 ms]
```

```
Range (min ... max):   1.233 s ... 1.278 s    10 runs
```

```
Summary
```

```
'fd -e jpg -uu' ran
```

```
3.80 ± 0.05 times faster than 'find -iname "*.jpg"'
```

```
► █
```

procs

- ps replacement written in rust
- colored and human-readable output
- multi-column keyword search
- <https://github.com/dalance/procs>



PID	User	State	Nice	TTY	CPU	MEM	VmPeak	VmSize	VmRSS	TCP	UDP	Read	Write	CPU Time	Start	Docker	Command
					[%]	[%]	[bytes]	[bytes]	[bytes]			[B/s]	[B/s]				
1		S	0		0.0	0.0	252.957M	188.957M	3.82M	[111]	[111]	0	0	03:04:21	2018/12/26 19:23		/usr/lib
2143		S	0		0.0	0.0	2.094M	2.094M	116K	[60001]	[]	0	0	00:00:00	2019/03/10 12:54		/home/ha
353209		S	0		0.0	0.0	188.813M	188.809M	3.277M	[60000]	[]	0	0	00:00:26	2019/03/29 03:02		/usr/bin
30366		S	0		0.0	0.0	303.582M	138.605M	76.781M	[]	[]	0	0	00:25:12	2018/12/26 19:23		/usr/lib
30401		S	0		0.0	0.0	409.914M	401.91M	668K	[]	[]	0	0	00:00:00	2018/12/26 19:23		/usr/sbi
30418		S	0		0.0	0.0	45.305M	44.57M	1.441M	[]	[]	0	0	00:03:13	2018/12/26 19:23		/usr/lib
32261		S	0		0.0	0.0	251.59M	251.57M	1M	[]	[]	0	0	00:00:05	2019/04/01 19:52		/usr/sbi
32262		S	0		0.0	0.0	302.234M	302.023M	6.02M	[]	[]	0	0	00:00:03	2019/04/01 19:52		/usr/lib
32265		S	0		0.0	0.0	275.234M	275.23M	2.52M	[]	[]	0	0	00:00:02	2019/04/01 19:52		/usr/lib
32266		S	0		0.0	0.0	239.207M	239.203M	2.258M	[]	[]	0	0	00:00:01	2019/04/01 19:52		/usr/lib
32267		S	0		0.0	0.0	230.723M	230.723M	908K	[]	[]	0	0	00:00:00	2019/04/01 19:52		/usr/lib
60228		S	0		0.0	0.0	44.879M	44.879M	136K	[]	[]	0	0	00:00:00	2018/12/26 19:24		/usr/sbi
60229		S	-4		0.0	0.0	60.59M	60.59M	596K	[]	[]	0	0	00:01:10	2018/12/26 19:24		/sbin/au
60231		S	-8		0.0	0.0	146.57M	82.57M	332K	[]	[]	0	0	00:01:12	2018/12/26 19:24		/sbin/au
60233		S	-4		0.0	0.0	54.324M	54.324M	404K	[]	[]	0	0	00:00:45	2018/12/26 19:24		/usr/sbi
60256		S	0		0.0	0.0	21.301M	21.297M	732K	[]	[]	0	0	00:49:44	2018/12/26 19:24		/usr/sbi
60259		S	0		0.0	0.0	223.25M	223.125M	1.043M	[]	[]	0	0	00:00:03	2018/12/26 19:24		/usr/sbi
60260		S	0		0.0	0.0	72.066M	72.004M	816K	[111]	[111, 647]	0	0	00:00:09	2018/12/26 19:24		/sbin/rp
60264		S	0		0.0	0.0	158M	115.023M	832K	[]	[323]	0	0	00:00:05	2018/12/26 19:24		/usr/sbi
60265		S	1		0.0	0.0	258.133M	194.133M	244K	[]	[]	0	0	00:00:46	2018/12/26 19:24		/usr/lib
60288		S	0		0.0	0.0	423.016M	420.508M	1.039M	[]	[]	0	0	00:01:07	2018/12/26 19:24		/usr/sbi
60290		S	0		0.0	0.0	8.48M	8.375M	148K	[]	[]	0	0	00:00:11	2018/12/26 19:24		/usr/bin
60291		S	0		0.0	0.0	224.508M	220.512M	1.496M	[]	[]	0	0	00:00:35	2018/12/26 19:24		/usr/bin
60293		S	0		0.0	0.0	220.551M	220.512M	1.238M	[]	[]	0	0	00:00:00	2018/12/26 19:24		/usr/bin
60296		S	0		0.0	0.0	629.453M	629.449M	13.063M	[]	[]	0	0	00:07:59	2018/12/26 19:24		/usr/lib
60301		S	0		0.0	0.0	88.277M	88.273M	136K	[]	[]	0	0	00:45:13	2018/12/26 19:24		/sbin/rn
60302		S	0		0.0	0.0	78.375M	60.23M	2.176M	[]	[]	0	0	00:31:35	2018/12/26 19:24		/usr/bin
60325		S	0		0.0	0.0	614.391M	551.496M	2.926M	[]	[]	0	0	00:47:26	2018/12/26 19:24		/usr/sbi
60326		S	0		0.0	0.0	24.441M	23.918M	536K	[]	[]	0	0	00:00:00	2018/12/26 19:24		/usr/sbi
60329		S	0		0.0	0.0	443.07M	443.066M	1.711M	[]	[]	0	0	00:08:30	2018/12/26 19:24		/usr/lib
60331		S	0		0.0	0.0	71.113M	70.98M	1.102M	[]	[5353, 34945]	0	0	00:08:32	2018/12/26 19:24		avahi-da
60332		S	0		0.0	0.0	70.613M	70.613M	84K	[]	[]	0	0	00:00:00	2018/12/26 19:24		avahi-da
60352		S	0		0.0	0.0	115.398M	115.398M	3.508M	[]	[]	0	0	00:10:55	2018/12/26 19:24		/bin/bas
310925		S	0		0.0	0.0	105.422M	105.422M	616K	[]	[]	0	0	00:00:00	2019/04/03 10:27		sleep 60
60366		S	0		0.0	0.0	36.512M	36.254M	1.477M	[]	[]	0	0	00:04:54	2018/12/26 19:24		/usr/lib
60368		S	0		0.0	0.0	398.648M	398.371M	3.133M	[]	[]	0	0	00:01:54	2018/12/26 19:24		/usr/lib
60658		S	0		0.0	0.0	77.992M	77.91M	55.879M	[]	[]	0	0	00:18:52	2019/02/22 17:47		tmux
37171		S	0	pts/91	0.0	0.0	124.383M	124.383M	1.809M	[]	[]	0	0	00:00:00	2019/03/01 11:26		-bash

bottom

- graphical process/system monitor
- support for zooming in/out
- customizable behaviour
- <https://github.com/ClementTsang/bottom>



Processes									Esc to go back
PID(p)	Name(n)	CPU%(c)▼	Mem%(m)	R/s	W/s	T.Read	T.Write	User	State
431477	ffmpeg	17.8%	1.5%	0B/s	0B/s	0B	6MB	clement	Runnable
1583	Web Content	0.7%	1.4%	0B/s	0B/s	34MB	0B	clement	Sleeping
1441	firefox	0.4%	2.0%	0B/s	0B/s	488MB	2.4GB	clement	Sleeping
398956	btm	0.4%	0.0%	0B/s	0B/s	0B	0B	clement	Sleeping
1017	Xorg	0.3%	0.4%	0B/s	0B/s	92MB	418KB	clement	Sleeping
2220	codium	0.1%	0.7%	0B/s	0B/s	143MB	50MB	clement	Sleeping
2167	codium	0.1%	1.0%	0B/s	0B/s	65MB	1MB	clement	Sleeping
2117	codium	0.1%	0.4%	0B/s	0B/s	11MB	0B	clement	Sleeping
144186	Discord	0.1%	0.4%	0B/s	0B/s	5MB	0B	clement	Sleeping
144265	Discord	0.1%	1.2%	0B/s	0B/s	31MB	0B	clement	Sleeping
397118	kitty	0.1%	0.4%	0B/s	0B/s	0B	0B	clement	Sleeping
1660	Web Content	0.1%	1.7%	0B/s	0B/s	2MB	0B	clement	Sleeping
1678	Web Content	0.1%	1.8%	0B/s	0B/s	28MB	0B	clement	Sleeping
988	(sd-pam)	0.0%	0.0%	0B/s	0B/s	0B	0B	clement	Sleeping
183	acpi_thermal_pm	0.0%	0.0%	0B/s	0B/s	0B	0B	root	Unknown
184	amd_iommu_v2	0.0%	0.0%	0B/s	0B/s	0B	0B	root	Unknown
1164	at-spi-bus-launcher	0.0%	0.0%	0B/s	0B/s	1MB	0B	clement	Sleeping
1196	at-spi2-registryd	0.0%	0.0%	0B/s	0B/s	106KB	0B	clement	Sleeping
160	ata_sff	0.0%	0.0%	0B/s	0B/s	0B	0B	root	Unknown
519	avahi-daemon	0.0%	0.0%	0B/s	0B/s	0B	0B	avahi	Sleeping
532	avahi-daemon	0.0%	0.0%	0B/s	0B/s	0B	0B	avahi	Sleeping
140486	bash	0.0%	0.0%	0B/s	0B/s	0B	0B	clement	Sleeping
1151	bash	0.0%	0.0%	0B/s	0B/s	7MB	0B	clement	Sleeping
370795	bash	0.0%	0.0%	0B/s	0B/s	0B	0B	clement	Sleeping
159	blkcg_punt_bio	0.0%	0.0%	0B/s	0B/s	0B	0B	root	Unknown
740	card0-crtc0	0.0%	0.0%	0B/s	0B/s	0B	0B	root	Sleeping
741	card0-crtc1	0.0%	0.0%	0B/s	0B/s	0B	0B	root	Sleeping
742	card0-crtc2	0.0%	0.0%	0B/s	0B/s	0B	0B	root	Sleeping
743	card0-crtc3	0.0%	0.0%	0B/s	0B/s	0B	0B	root	Sleeping
744	card0-crtc4	0.0%	0.0%	0B/s	0B/s	0B	0B	root	Sleeping
745	card0-crtc5	0.0%	0.0%	0B/s	0B/s	0B	0B	root	Sleeping
469	ccp-1-q2	0.0%	0.0%	0B/s	0B/s	0B	0B	root	Sleeping
470	ccp-1-q3	0.0%	0.0%	0B/s	0B/s	0B	0B	root	Sleeping
471	ccp-1-q4	0.0%	0.0%	0B/s	0B/s	0B	0B	root	Sleeping
439	cfg80211	0.0%	0.0%	0B/s	0B/s	0B	0B	root	Unknown
236	charger_manager	0.0%	0.0%	0B/s	0B/s	0B	0B	root	Unknown
370872	clipit	0.0%	0.1%	0B/s	0B/s	0B	0B	clement	Sleeping
2146	codium	0.0%	0.2%	0B/s	0B/s	4MB	147KB	clement	Sleeping
2248	codium	0.0%	0.6%	0B/s	0B/s	15MB	0B	clement	Sleeping
2087	codium	0.0%	0.5%	0B/s	0B/s	43MB	9MB	clement	Sleeping
393252	codium	0.0%	0.2%	0B/s	0B/s	0B	0B	clement	Sleeping
2090	codium	0.0%	0.1%	0B/s	0B/s	2MB	0B	clement	Sleeping
2091	codium	0.0%	0.1%	0B/s	0B/s	0B	0B	clement	Sleeping
2233	codium	0.0%	0.4%	0B/s	0B/s	10MB	12KB	clement	Sleeping
393214	codium	0.0%	0.2%	0B/s	0B/s	31MB	0B	clement	Sleeping
2777	codium	0.0%	0.6%	0B/s	0B/s	6MB	0B	clement	Sleeping
2266	codium	0.0%	0.2%	0B/s	0B/s	504KB	0B	clement	Sleeping
157343	codium	0.0%	0.2%	0B/s	0B/s	221KB	0B	clement	Sleeping

btop++

- resource monitor
- stats for CPU, RAM, I/O, processes
- C++ continuation of bashtop and bpytop
- <https://github.com/aristocratos/btop>



1cpu

menu

21:02

i7-5775C

- 2000ms +

3.3 GHz

CPU ██████████ 8% 46°C

C0 6% 45°C

C1 5% 46°C

C2 5% 45°C

C3 13% 46°C

C4 6% 45°C

C5 9% 46°C

C6 9% 45°C

C7 8% 46°C

Load AVG: 0.45 0.45 0.51

up 52d 05:25

2mem

disks

io

Total: 15.5 GiB

Used: 8.90 GiB

57%

Available: 6.66 GiB

43%

Cached: 4.03 GiB

26%

Free: 1.54 GiB

10%

root ▾44K 226 GiB

IO%

Used: 39% ██████████ 88.8 GiB

Free: 61% ██████████ 137 GiB

swap 4.95 GiB

Used: 48% ██████████ 2.39 GiB

Free: 52% ██████████ 2.56 GiB

Media2 3.58 TiB

IO%

Used: 25% ██████████ 930 GiB

Free: 75% ██████████ 2.67 TiB

Media 7.21 TiB

IO%

Used: 53% ██████████ 3.83 TiB

1210131 btop

0.0%

terminate kill signals

hide

Status: Elapsed: IO/R: IO/W: Parent:

Sleeping 00:01:23 20.0 KiB 4.00 KiB zsh

Memory: 0.0% 4.68 MiB

C

M

D

./bin/btop

3net 192.168.1.11

sync auto zero <b br0 n>

2M

download

▼ 1.30 KiB/s (10.4 Kibps)

▼ Top: (153 Mibps)

▼ Total: 898 GiB

▲ 346 Byte/s (2.70 Kibps)

▲ Top: (1.59 Mibps)

▲ Total: 486 GiB

upload

21K

4proc filter

per-core reverse tree < cpu lazy >

Pid: Program: Command: Threads: User: MemB Cpu% ↑

1186798 mpv mpv --geometry=1462 15 gnm 547M 19.0

999834 node /home/gnm/.vscode-s 12 gnm 308M 18.5

474394 python3 python3 /home/gnm/b 3 gnm 16M 5.0

3144881 firefox /usr/lib/firefox/fi 140 gnm 757M 1.0

1639 Xorg /usr/lib/xorg/Xorg 2 root 79M 2.5

1211129 tracker-extract /usr/libexec/tracke 16 gnm 41M 0.0

1186800 java java -Dswing.defaul 64 gnm 621M 0.5

3145432 Web Content /usr/lib/firefox/fi 27 gnm 206M 1.0

1186781 python3 /usr/bin/python3 /h 4 gnm 76M 0.0

3145189 Web Content /usr/lib/firefox/fi 30 gnm 314M 0.0

1210131 btop ./bin/btop 3 gnm 5M 0.0

7378 compiz compiz 4 gnm 114M 5.5

1000249 node /home/gnm/.vscode-s 15 gnm 585M 0.0

7286 terminator /usr/bin/python3 /u 4 gnm 56M 1.0

3145058 Web Content /usr/lib/firefox/fi 27 gnm 234M 0.5

1000068 node /home/gnm/.vscode-s 11 gnm 145M 0.0 ↓

↑ select ↓ info terminate kill signals

0/455



musikcube

- music player
- streaming audio server
- musikdroid android app
- <https://github.com/clangen/musikcube>



mad

track filter results

Above

1	Wake Up	7:38	Mad Season
2	X-Ray Mind	5:13	Mad Season
3	River Of Deceit	5:03	Mad Season
4	I'm Above	5:45	Mad Season
5	Artifical Red	6:15	Mad Season
6	Lifeless Dead	4:27	Mad Season
7	I Don't Know Anything	5:02	Mad Season
8	Long Gone Day	4:51	Mad Season
9	November Hotel	7:07	Mad Season
10	All Alone	4:15	Mad Season

Axis: Bold As Love [+video]

9	Castles Made Of Sand	2:48	Jimi Hendrix
---	----------------------	------	--------------

BBC Sessions

8	I Was Made To Love Her	3:05	Jimi Hendrix
---	------------------------	------	--------------

Electric Ladyland [+Video]

1	...And The Gods Made Love	1:22	Jimi Hendrix
---	---------------------------	------	--------------

Electriclarryland

playing **I Don't Know Anything** by **Mad Season** from **Above**

shuffle

vol ———■—— 65% 4:12 —————■———— 5:02 repeat list

googler

- google from the command-line
- shows the title, URL and abstract
- directly open links in a browser
- <https://github.com/jarun/googler>



22:41:32 [~] >>>

googler --count 3 --exact googler

1 Urban Dictionary: Googler

<http://www.urbandictionary.com/define.php?term=Googler>

Googler. An employee working at Google. Employee benefits include free massages, go set working hours, constant talks from presidential ...

2 Googler - Wiktionary

<https://en.wiktionary.org/wiki/Googler>

Noun[edit]. Googler (plural Googlers). A full-time Google corporation employee. A habitual user of the Google search engine.

3 GitHub - jarun/googler: Google Search, Google Site Search, Google ...

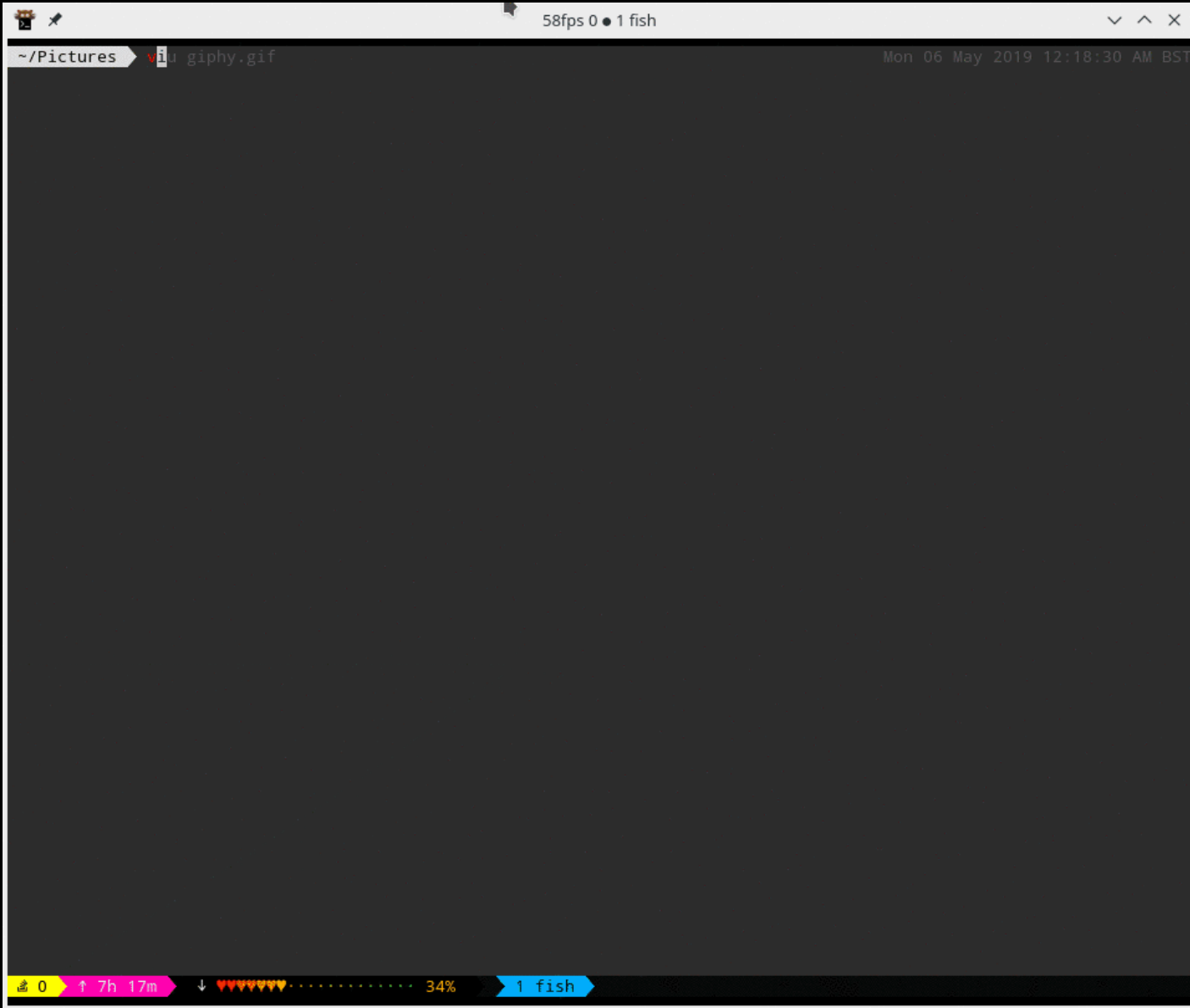
<https://github.com/jarun/googler>

googler is a power tool to Google (Web & News) and Google Site Search from the comm shows the title, URL and abstract for each result, which can ...

viu

- view images from the terminal
- uses iTerm or Kitty graphics protocol
- lower half blocks (▀) as fallback
- <https://github.com/atanunq/viu>





hexyl

- hex viewer
- colored output to distinguish different categories of bytes
- <https://github.com/sharkdp/hexyl>



► hexyl small.png

```
89 50 4e 47 0d 0a 1a 0a
00 00 00 80 00 00 00 44
3e 00 00 00 c2 49 44 41
30 14 40 d1 b7 34 dd ff
6c 28 73 e2 aa 34 49 03
8d 3b 87 fe 01 00 80 00
00 08 00 00 01 00 20 00
00 02 00 40 00 00 08 00
5e 6a 64 4b 94 f5 98 7c
3f 73 71 58 5f af 8b 79
d1 ce b6 e3 75 3b e6 b9
c6 33 93 7b 66 37 cf ab
10 00 00 02 00 40 00 00
04 00 80 00 00 10 00 00
01 00 20 00 00 8c 37 db
00 00 00 49 45 4e 44 ae
```

```
00 00 00 0d 49 48 44 52
08 02 00 00 00 c6 25 aa
54 78 5e ed d4 81 06 c3
ff 6f b3 74 56 ea 89 12
87 d6 fe d8 7b 89 bb 52
00 10 00 00 02 00 40 00
00 04 00 80 00 00 10 00
00 01 00 20 00 00 00 d4
d1 f4 92 5c 5c 3e cf 9c
5b ee 96 b6 47 eb f1 ea
95 8d c7 ce 03 39 c9 af
bf f9 c9 2f 08 80 00 00
08 00 00 01 00 20 00 00
02 00 40 00 00 08 00 00
68 03 20 fb ed 96 65 00
42 60 82
```

```
xPNG . 000 IHDR
000x000D . .000x%x
>000xIDA Tx^xxx.x
0•@xx4xx xoxTVxx•
l(sxx4I• xxxx{xxR
x;xx•0x0 0•00•0@0
0•00•0 0 0•0x00•0
0•0@00•0 0•0 000x
^jdKxxx| xxx\>xx
?sqX_xxy [xxxGxxx
xxxxu;xx xxxx•9xx
x3x{f7xx xxx/•x00
•00•0@00 •00•0 00
•0x00•00 •0@00•00
•0 00x7x h• xxxe0
000IENDx B`x
```



glow

- render markdown on the CLI
- browse local and stashed markdown
- less comaptible keystrokes
- <https://github.com/charmbracelet/glow>



> █

falsisign

- make a PDF look hand signed
- add print/scan cycle artefacts
- satisfy fax fans around the world
- <https://gitlab.com/edouardklein/falsisign>



Communications From Elsewhere

Home of R/L20

The Dialectic of Consensus: Social realism and postconceptualist objectivism

W. Henry Reicher

*Department of Literature, University of Massachusetts,
Amherst*

1. The deconstructivist paradigm of discourse and postcultural dialectic theory

"Society is used in the service of class divisions," says Lacan. The main

Communications From Elsewhere

Home of R/L20

The Dialectic of Consensus: Social realism and postconceptualist objectivism

W. Henry Reicher

*Department of Literature, University of Massachusetts,
Amherst*

1. The deconstructivist paradigm of discourse and postcultural dialectic theory

"Society is used in the service of class divisions," says Lacan. The main

fuck

- corrects errors in previous console commands
- blindly running corrected commands is possible
- <https://github.com/nvbn/thefuck>





nvbn@nvbn-desktop: ~/exp/thefuck 76x18

→ thefuck git:(master) X

usbkill

- anti-forensic kill-switch
- waits for a change on your USB ports
- immediately shuts down your computer
- <https://github.com/hephaest0s/usbkill>



It's now safe to turn off
your computer.

Danke schön!

Fragen ?

