

# Owning Your Stuff (Escaping from Development Dependency Hell)

Tim Sutton, Montreal (YUL)  
Sauce Labs

21/01/2015

# MAIS POURQUOI YUL?

---

Qui dit YUL, dit Montréal. Le code de l'aéroport est aussi branché à Montréal que LAX est pour Los Angeles, ces trois lettres sont presque devenues une marque de commerce en soi. Mais d'où vient ce YUL? Pourquoi l'aéroport international situé à Dorval porte-t-il cette désignation?

## CATÉGORIES

Actualité

Histoire

toponymie

Urbanisme

## ÉTIQUETTES

aéroport

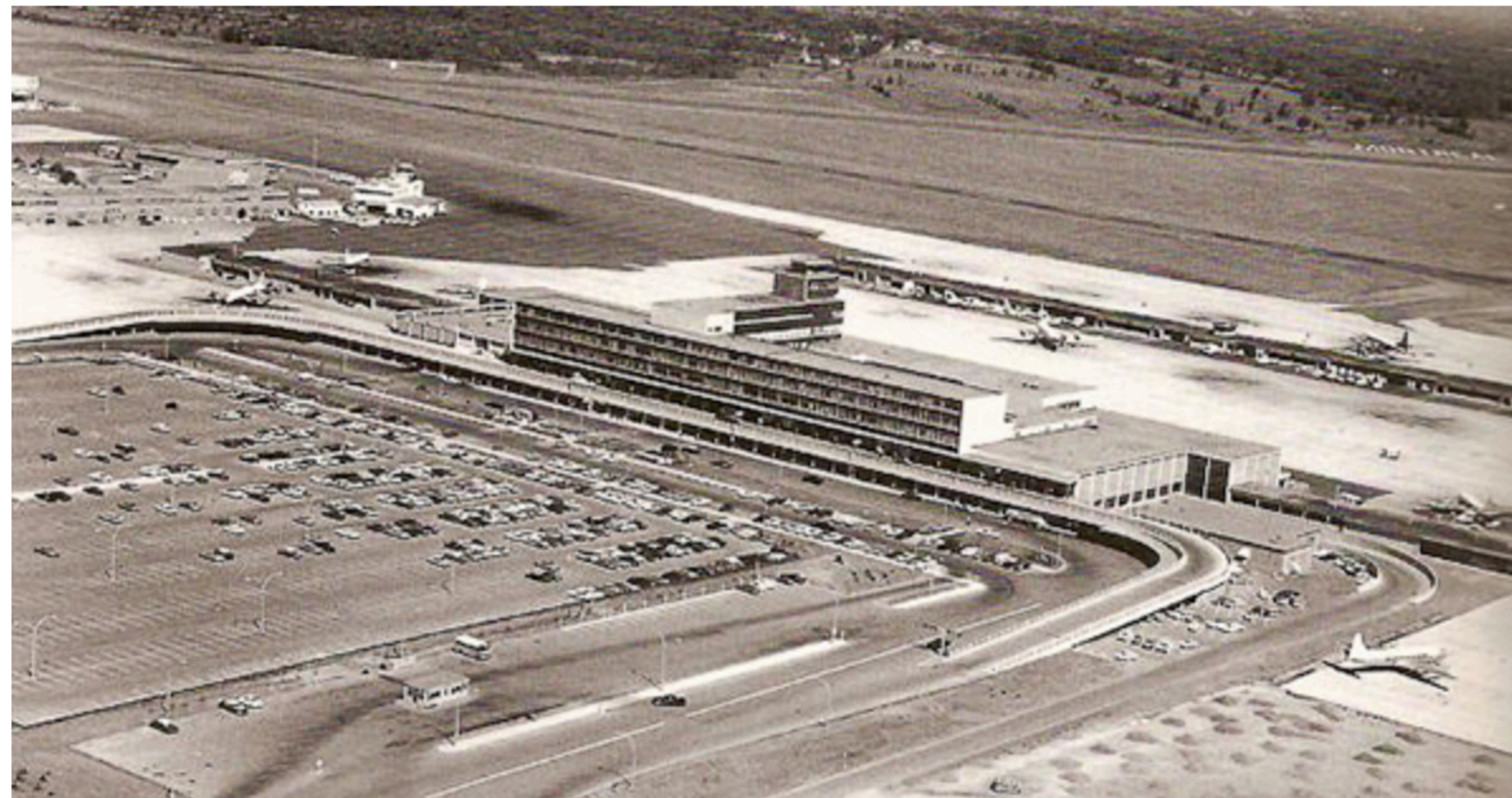
Antenne

Beacon

Dorval

Kirkland

Mirabel



<http://proposmontreal.com/index.php/mais-pourquoi-yul/>

st-Hubert



[View waymark gallery](#)



## Montreal NDB UL - Kirkland, QC, Canada

in [Ground Based Air Navigation Support Structures](#)

Posted by:  [denben](#)

**N 45° 27.606 W 073° 50.817**

18T E 590147 N 5034710

**Quick Description:** Montreal NDB UL is located in Kirkland, an on-island suburb on the Island of Montreal in southwestern Québec, Canada.

**Location:** Québec, Canada

**Date Posted:** 10/28/2017 9:22:56 AM

**Waymark Code:** WMWY12

**Published By:**  [fi67](#)

**Views:** 25

---

### Download this waymark:



[.GPX File](#)



[.LOC File](#)



[.KML File \(Google Earth\)](#)

---

### Long Description:

Radio aid to air navigation, this beacon continually emits the letters UL in morse (Dididah Didahdidit) on the LF frequency of 248 kHz.

It is owned and maintained by Nav Canada and is in a straight line with runway 10 at Montreal / Pierre Elliott Trudeau International Airport (YUL) (3.5 nautical miles from the runway threshold).

Name: Montreal NDB

Identifier: UL (.- .-..)

Frequency: 248 KHz

Location: 45.459999, -73.848297

Intended use: high- and low-level enroute navigation (medium power)

Country: Canada

Associated airport: Montreal / Pierre Elliott Trudeau International Airport

[Gallery](#)



**Date Posted:** 10/28/2017 9:22:56 AM

**Waymark Code:** WMWY12

**Published By:**  [fi67](#)

**Views:** 25

---

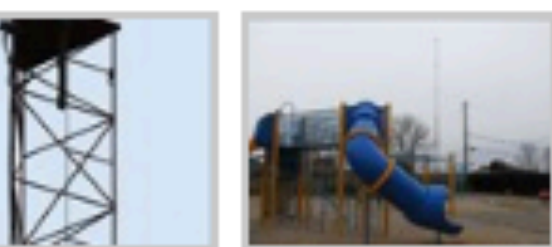
[rth\)](#)

---

on, this beacon continually emits the letters UL in morse (Dididah Didahdidit) on the LF

ned by Nav Canada and is in a straight line with runway 10 at Montreal / Pierre Elliott  
airport (YUL) (3.5 nautical miles from the runway threshold).

[Gallery](#)



**Date Posted:** 10/28/2017 9:22:56 AM

**Waymark Code:** WMWY12

**Published By:**  [fi67](#)

**Views:** 25


---


[rth\)](#)


---


on, this beacon continually emits the letters UL in morse (Dididah Didahdidit) on the LF


ned by Nav Canada and is in a straight line with runway 10 at Montreal / Pierre Elliott  
airport (YUL) (3.5 nautical miles from the runway threshold).




 Dashboard

 Live Testing

 Tunnels 1 ACTIVE

 Analytics ▾

 Archives

0 / 100

concurrent sessions


Access Real Devices Automation


Help ▾

Tim Sutton ▾

Web Browser

Mobile App

 Desktop

 Mobile Device

TEST URL

whatsmybrowser.org

Real Devices

Virtual Device

Please select your device. Can't find your device listed? [Let us know!](#)

MANUFACTURER

Apple ▾

DEVICE

iPhone X Simulator ▾

OS VERSION

iOS 11.3 ▾

SAUCE CONNECT PROXY (optional)

timsutton/33bb169fe59845c281057f7061ebfb59 ▾

Recent


Saved

whatsmybrowser.org

Google Chrome 67

macOS High Sierra

1024 x 768




whatsmybrowser.org

Internet Explorer 8

Windows 7

800 x 600




whatsmybrowser.org

Internet Explorer 8

Windows 7

800 x 600



whatsmybrowser.org

Internet Explorer 10

MEDIA SETTINGS

☒ Save Video, Screenshots and Logs

☐ Save this configuration

Start session

Dashboard

Live Testing

Tunnels 1 ACTIVE

Analytics

Archives

0 / 100

concurrent sessions

Test **Complete**

View Test History

Team

Report

Delete

Admin

MacDevOps 2018 Test

Install JIRA Plugin

Watch

Commands

Logs

Network

Metadata

FILTER: Command

Has Screenshot

Play

3 of 6

POST /session	00:00:00 + 2.60s	
POST url	00:04:28 + 4.70s	
POST element	00:09:16 + 0.03s	
POST element/0.431186644216881-1/...	00:09:31 + 4.86s	
POST back	00:14:46 + 1.38s	
DELETE /session/d25862dec34f4aa4...	00:16:09 + 0.00s	

00:00:0900:00:16

Schedule - MacDevOpsYVR

Secure | https://mdoyvr.com/schedule/

MAC DEV OPS YVR

[ Buy Tickets ]

Attendees

Speakers

Sponsors

Activities

Schedule

Conduct

Videos

Contact

SCHEDULE

Haven't attended MacDevOpsYVR before?

Here's the schedule for our 2018 conference to give you an idea of what you'll experience.

Schedule is subject to change. Stay tuned to changes via [Twitter](#), the [MacAdmins Slack](#) #macdevopsyvr channel and by joining our [e-mail list](#).

SAUCELABS

Dashboard

Live Testing

Tunnels1 ACTIVE

Analytics

Archives

0 / 100

concurrent sessions

WatchCommandsLogsNetworkMetadata

Hide Page Timeline | Show Statistics |

Page Load: 4.12s, 49 Requests2018-06-03, 4:41:42 PMhttps://www.eventbrite.com/e/mdoyvr2018-tickets-38821491125

https://mdoyvr.com/schedule/

GET schedule20094.8 KB224.90ms

HeadersHTML

Response Headers

status200

servernginx

dateSun, 03 Jun 2018 23:41:44 GMT

content-typetext/html; charset=UTF-8

strict-transport-securitymax-age=86400

varyAccept-Encoding

Cookie

last-modifiedSun, 03 Jun 2018 23:37:47 GMT

cache-controlmax-age=63, must-revalidate

x-nanananaBatcache

x-hackerIf you're reading this, you should visit automatic.com/jobs and apply to join the fun, mention this header.

link<https://wp.me/P6x36i-dq>; rel=shortlink

content-encodinggzip

x-ac1.sjc \_bur

Request Headers

Upgrade-Insecure-Requests1

User-AgentMozilla/5.0 (Macintosh; Intel Mac OS X 10\_13\_1) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/64.0.3282.140 Safari/537.36

X-DevTools-Emulate-Network-Conditions-Client-IdACFB05E23723EDFBD1028E2C7E1B7FF9

GET remote-login.php20001.05s

GET \_static??-eJx9j9E2005 KB873.80ms

GET css?family=Libre2006.9 KB1.4s

GET style.css?m=15220063 KB874.50ms

# Dependencies

# **AutoPkg and GitHub TLSv1.0/1.1 removal (Feb 22, 2018)**

```
#!/usr/bin/python
```

```
>>> platform.mac_ver()[0]  
'10.11.6'  
>>> ssl.OPENSSL_VERSION  
'OpenSSL 0.9.8zh 14 Jan 2016'
```

```
>>> platform.mac_ver()[0]  
'10.12.6'  
>>> ssl.OPENSSL_VERSION  
'OpenSSL 0.9.8zh 14 Jan 2016'
```

```
>>> platform.mac_ver()[0]  
'10.13.4'  
>>> ssl.OPENSSL_VERSION  
'LibreSSL 2.2.7'
```



# Weak cryptographic standards removal notice

Feb 02, 2018 ptoomey3 Engineering

Last year we announced the deprecation of several weak cryptographic standards. Then we provided a status update toward the end of last year outlining some changes we'd made to make the transition easier for clients. We quickly approached the February 1, 2018 cutoff date we mentioned in previous posts and, as a result, pushed back our schedule by one week. On February 8, 2018 we'll start disabling the following:

- TLSv1 / TLSv1.1 : This applies to all HTTPS connections, including web, API, and git connections to https://github.com and https://api.github.com.
- diffie-hellman-group1-sha1 : This applies to all SSH connections to github.com
- diffie-hellman-group14-sha1 : This applies to all SSH connections to github.com

We'll disable the algorithms in two stages:

- February 8, 2018 19:00 UTC (11:00 am PST): Disable deprecated algorithms for one hour
- February 22, 2018 19:00 UTC (11:00 am PST): Permanently disable deprecated algorithms

For more details, head to the Engineering Blog.

Featured

All Posts

Announcements

Data

Editor Tools

Education

Engineering

Events

GitHub for Business

Policy

Changelog

Subscribe



# github.com/clburlison/vendored



# vendored

✓ PASSED

The goal of this repo is to make it easy to "vendor" your own frameworks and programming languages in an automated fashion.

Once this project is complete you will be able to have own version of python with custom pip modules included, ruby, OpenSSL, and more all in one nice big package or multiple smaller packages for easy deployment.


# Clay Caviness - Securing Management Tools - MacADUK 2017

Python!

<https://github.com/google/macops/tree/master/packages/python>

```
configure-python: ${PYTHON_BUILD} /Library/MegaCorpSupport/openssl
cd "${PYTHON_BUILD}" && \
echo "_socket socketmodule.c timemodule.c" >> Modules/Setup.dist && \
echo "_ssl _ssl.c -DUSE_SSL "\
    "-I/Library/MegaCorpSupport/openssl/include " \
    "-I/Library/MegaCorpSupport/openssl/include/openssl " \
    "-L/Library/MegaCorpSupport/openssl/lib -lssl -lcrypto" >> \
    Modules/Setup.dist && \
./configure \
    --prefix="${WORK_D}/${PYTHON_PATH}" \
    --enable-ipv6 \
    --with-system-expat \
    --with-threads \
    --datarootdir="${WORK_D}/${PYTHON_PATH}/share" \
    --datadir="${WORK_D}/${PYTHON_PATH}/share"
patch ${PYTHON_BUILD}/Lib/ssl.py < log_cert.patch
```

Google



## Clay Caviness - Securing Management Tools

98 views

 3  0  SHARE  



MacADUK

Published on Mar 29, 2017

SUBSCRIBE 377

**What is so hard? It's just ``pip install``?**

# **NPM (part 1)**

## **(Feb 21, 2018)**

# Library Package Managers (LPMs)

# Critical Linux filesystem permissions are being changed by latest version #19883

Closed

Crunkle opened this issue on Feb 21 · 121 comments



Crunkle commented on Feb 21 • edited ▼

I'm opening this issue because:

- ☐ npm is crashing.
- ☐ npm is producing an incorrect install.
- ☒ npm is doing something I don't understand.
- ☒ Other (see below for feature requests):

What's going wrong?

This issue has been happening ever since 5.7.0 was released a few hours ago. It seems to have completely broken my filesystem permissions and caused me to have to manually fix the permissions of critical files and folders. I believe that it is related to the commit `94227e1` which is traversing and running `chown` on the wrong, often critical, filesystem files and folders.

By running `sudo npm` under a non-root user (root users do not have the same effect), filesystem permissions are being heavily modified. For example, if I run `sudo npm --help` or `sudo npm update -g`, both commands cause my filesystem to change ownership of directories such as `/etc`, `/usr`, `/boot`, and other directories needed for running the system. It appears that the ownership is recursively changed to the user currently running npm.

I found that a selection of directories in `/` were owned by a non-root user after running `sudo npm` and many binaries in `/usr/bin` stopped working as their permissions were changed. People experiencing this bug will likely have to fully reinstall their system due to this update.

`npm update -g` as `root` :

No output, all packages up to date. Likely still causes a `chown` to be run silently to `root:root` .  
`drwxr-xr-x 10 root root 129 Feb 22 03:39 /usr`

Assignees

No one assigned

Labels

None yet

Projects

None yet

Milestone

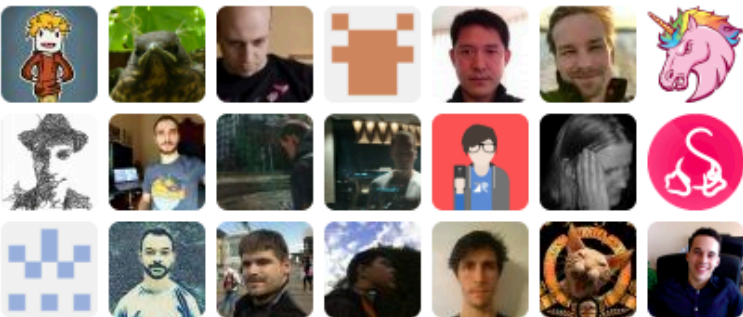
No milestone

Notifications

**Subscribe**

You're not receiving notifications from this thread.

78 participants



and others

✓ Other (see *below for feature requests*):

## What's going wrong?

This issue has been happening ever since 5.7.0 was released a few hours ago. It seems to have completely broken my filesystem permissions and caused me to have to manually fix the permissions of critical files and folders. I believe that it is related to the commit [94227e1](#) which is traversing and running `chown` on the wrong, often critical, filesystem files and folders.

By running `sudo npm` under a non-root user (root users do not have the same effect), filesystem permissions are being heavily modified. For example, if I run `sudo npm --help` or `sudo npm update -g`, both commands cause my filesystem to change ownership of directories such as `/etc`, `/usr`, `/boot`, and other directories needed for running the system. It appears that the ownership is recursively changed to the user currently running npm.

I found that a selection of directories in `/` were owned by a non-root user after running `sudo npm` and many binaries in `/usr/bin` stopped working as their permissions were changed. People experiencing this bug will likely have to fully reinstall their system due to this update.

```
npm update -g as root:
```

No output, all packages up to date. Likely still causes a `chown` to be run silently to `root:root`.

```
drwxr-xr-x 10 root root 129 Feb 22 03:39 /usr
```

## Projects

None yet

## Milestone

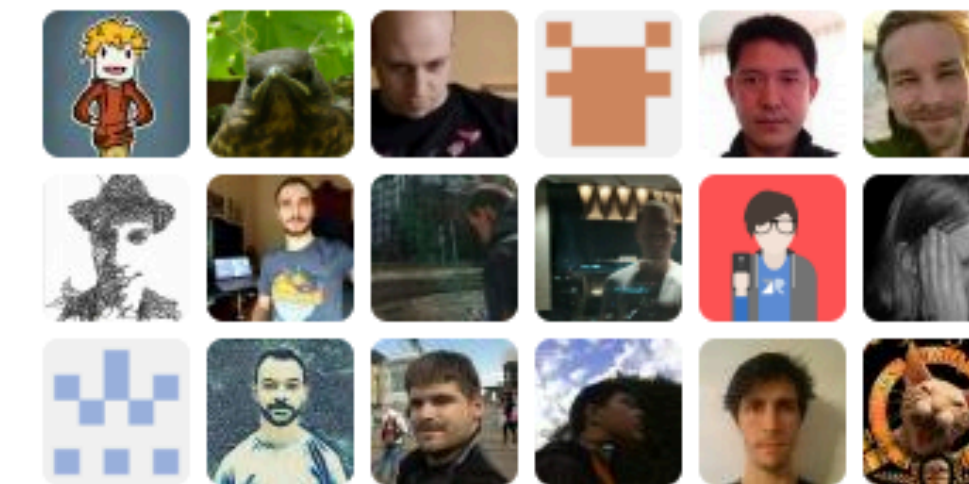
No milestone

## Notifications

🔊 **Subscribe**

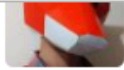
You're not receiving notifications from this thread.

## 78 participants



and others





dwardae commented on Feb 22



34

10

10

5



cjdelisle commented on Feb 22

This thread



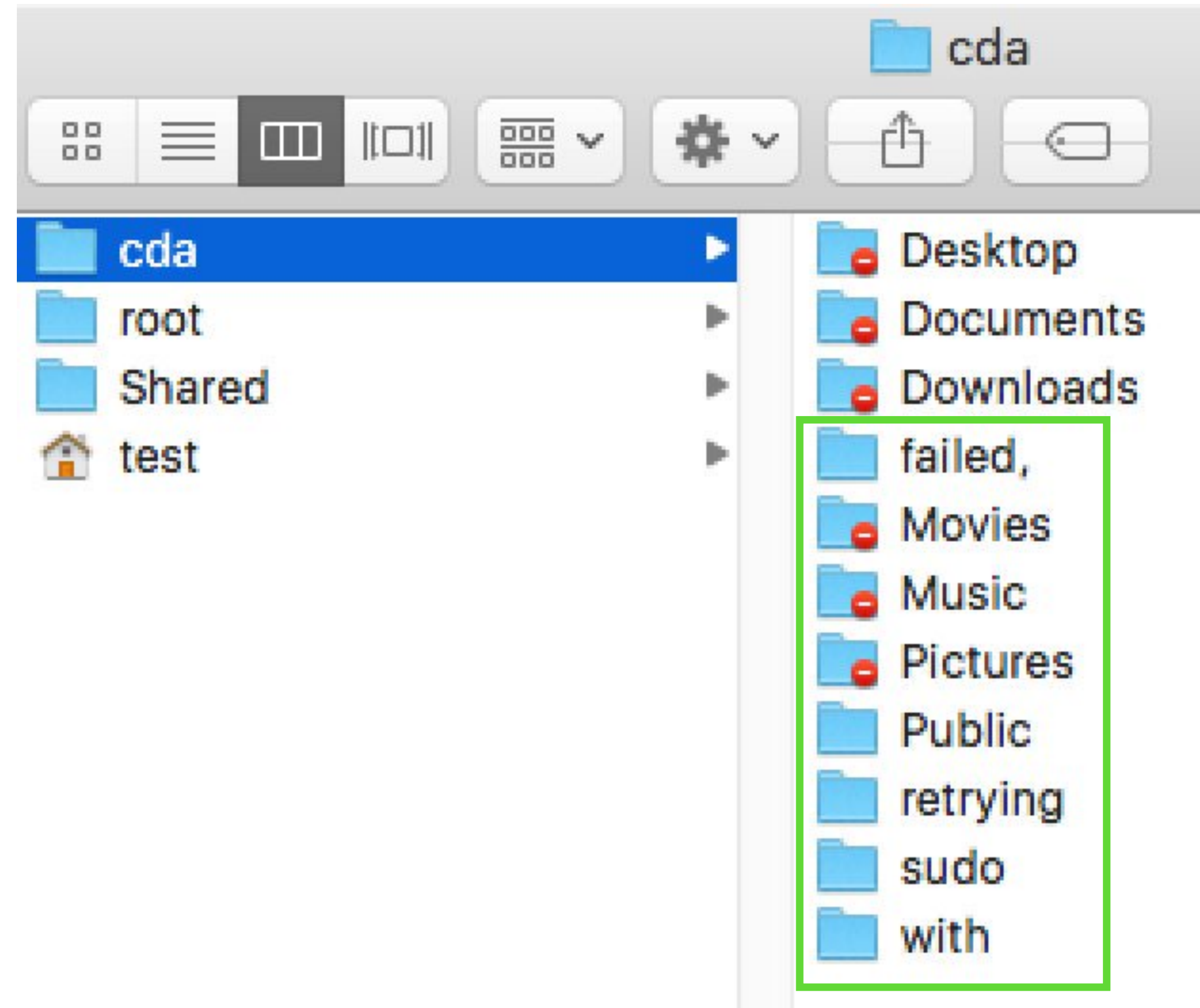
25

3

16

10

8





**Repo  
mirrors**

**Version pinning**

**Isolated runtime**

**Isolated packages**



**Repo  
mirrors**

**Version pinning**

**Isolated runtime**

**Isolated packages**

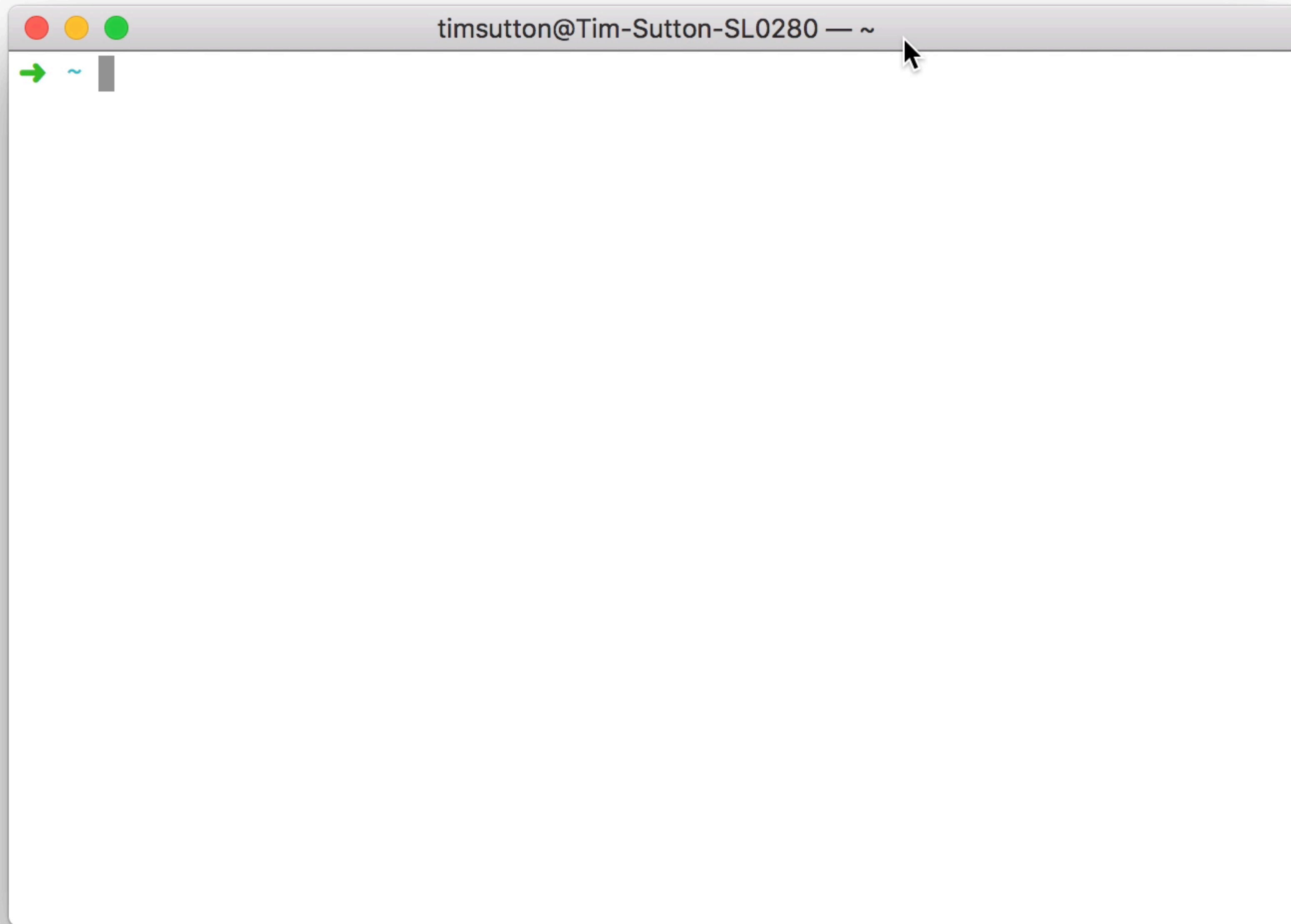
# Isolate packages

- Don't sudo `<pip|gem|npm> install` anything
- Eliminates contamination of OS managed/supplied packages
- Your project's environment lives independently of any other projects and packages
- (Mostly) portable to any other OS / environment

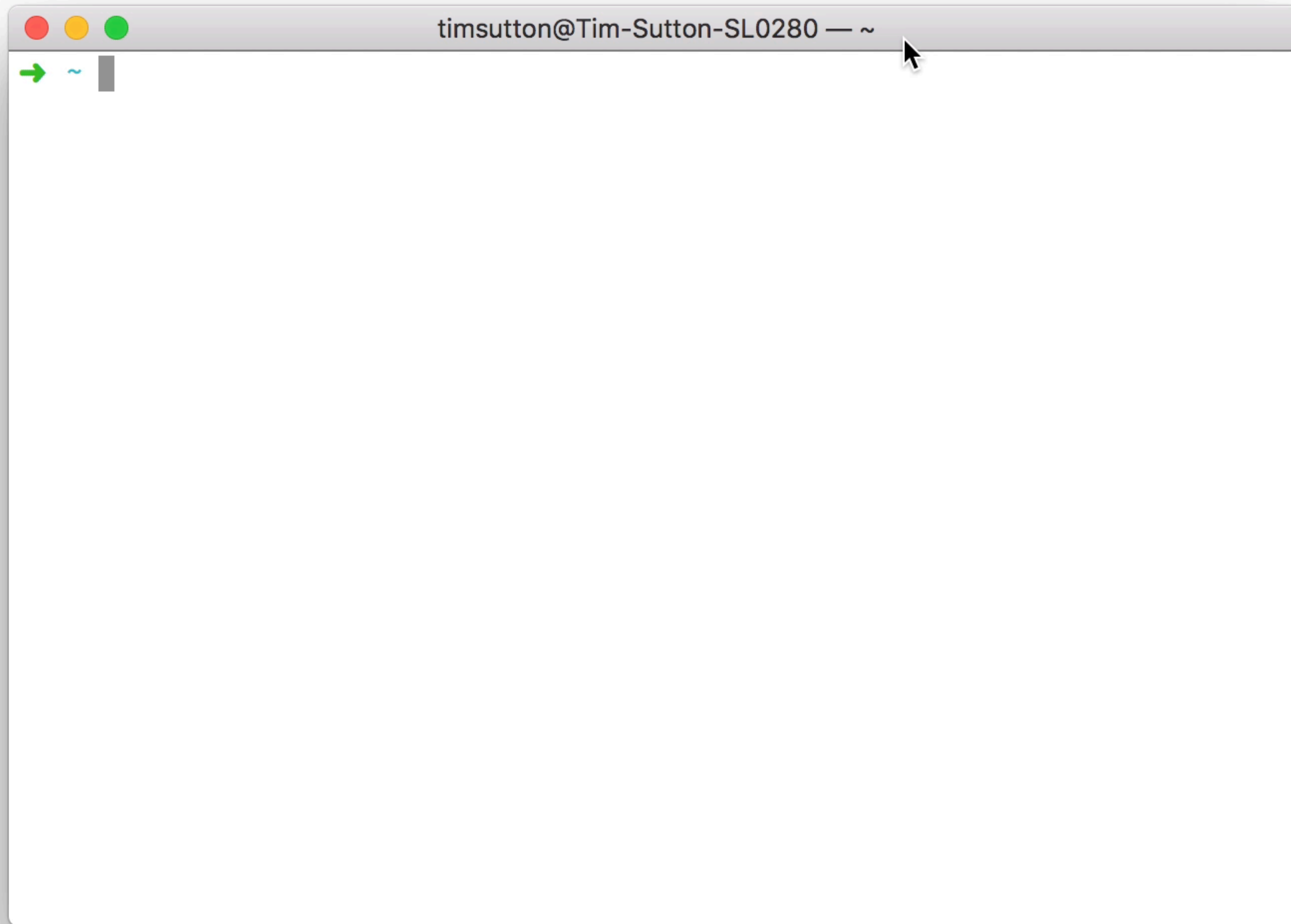
# Isolate packages (Python)

- Install virtualenv once (and ideally virtualenvwrapper or similar)
- `virtualenv / mkvirtualenv -p` to specify which Python you want to use when creating
- Use `workon` to easily switch between environments
- `pip install`, then `pip freeze` to save a `requirements.txt`

# Isolate packages (Python)



# Isolate packages (Python)



# Isolate packages (Ruby)

- Install Bundler
- Define project gems in `Gemfile`, constrain versions as you wish
- `bundle update` to resolve and "freeze" the constraints to `Gemfile.lock`
- Now any `bundle install` will only install the exact versions from `Gemfile.lock` - check both files into version control
- Run `bundle update` to update it later
- Run things via `bundle exec <command>`

# Isolate packages (Ruby)

## Gemfile

```
gem "sinatra", "~> 1"
```

# Isolate packages (Ruby)

## Gemfile

```
gem "sinatra", "~> 1"
```

## Gemfile.lock

GEM

specs:

rack (1.6.10)

rack-protection (1.5.5)

rack

sinatra (1.4.8)

rack (~> 1.5)

rack-protection (~> 1.4)

tilt (>= 1.3, < 3)

tilt (2.0.8)

PLATFORMS

ruby

DEPENDENCIES

sinatra (~> 1)

# Isolate packages (Ruby)

## Gemfile

```
gem "sinatra", "~> 2"
```

# Isolate packages (Ruby)

## Gemfile

```
gem "sinatra", "~> 2"
```

→ bundle update

Fetching gem metadata from <https://rubygems.org/>.....

Resolving dependencies...

Using bundler 1.16.1

Fetching mustermann 1.0.2

Installing mustermann 1.0.2

Fetching rack 2.0.5 (was 1.6.10)

Installing rack 2.0.5 (was 1.6.10)

Fetching rack-protection 2.0.2 (was 1.5.5)

Installing rack-protection 2.0.2 (was 1.5.5)

Using tilt 2.0.8

Fetching sinatra 2.0.2 (was 1.4.8)

Installing sinatra 2.0.2 (was 1.4.8)

Bundle updated!



**Repo  
mirrors**

**Version pinning**

**Isolated runtime**

**Isolated packages**



Text	Explanation
\$PATH	\$PATH refers to the <a href="#">PATH</a> environment variable, which determines where to search for executable files. In this case, it indicates that the pip, Homebrew Python (2.7), and OSX's pre-installed Python are accessible on path, with ~/newenv/ and a mysterious ???? as part of PATH.
pip	<a href="#">pip</a> is the Python <a href="#">package management system</a> , and is used to install and manage python packages. As it is written in Python, it requires Python to run. It leads to easy_install, Homebrew Python (2.7), "(misc folders owned by root)", and ????.
Homebrew Python (2.7)	<a href="#">Homebrew</a> is the de facto standard third-party OSX package manager. Homebrew Python (2.7) is the Python 2 version installed through Homebrew. This leads to Python.org binary (2.6) and /usr/local/Cellar.
OS Python	Apple bundles an (out of date) version of Python with OSX. This only leads to ????.
????	
easy_install	easy_install, much like pip, is a cpan-like tool to download and install Python packages. As of the creation of the comic, many people discourage its use. (e.g., <a href="#">this question on stack exchange</a> . 📖)
Anaconda Python	<a href="#">Anaconda</a> is a Python distribution for data science and machine learning related applications.
Homebrew Python (3.6)	As of the creation of the comic, Python 3.6 is the current stable version of Python.
Python.org binary (2.6)	
(Misc folders owned by root)	
/usr/local/Cellar	
/usr/local/opt	Both <code>/usr/local</code> and <code>/opt</code> are directories that store files that do not belong to a Unix-like operating system. Usually, files in <code>/usr/local</code> were created with a <a href="#">make command</a> , and files in <code>/opt</code> are unbundled packages. The joke is that <code>/usr/local/opt</code> should really, really not exist.
/(A bunch of items with "Frameworks" in them somewhere)/	The system-included Python distribution in macOS resides in /System/Library/Frameworks
\$PYTHONPATH	
Another pip??	There should only be one PIP (package management system) in place. More that one would lead to them contradicting each other. Randall is confused as to how the second one got there.
~/python/	

[https://www.explainkcd.com/wiki/index.php/1987:\\_Python\\_Environment](https://www.explainkcd.com/wiki/index.php/1987:_Python_Environment)

virtualenv's Python instances, while more common packages are still referenced via the system Python paths.

# Isolate runtime / interpreter

# Isolate runtime / interpreter

- Remove variables in the OS which limits supported OSes for your project
  - Apple: Python/Ruby linked to OpenSSL until 10.13 (switched to LibreSSL)
    - Ruby versions included on macOS make large jumps in versions
- Linux: runtime provided by the OS may be too old
  - Needing to ever change switch base OS flavours could change your runtime version
- Windows: Never came with Python 🙄

# Isolate runtime / interpreter

- Remove variables in the OS which limits supported OSes for your project
  - Apple: Python/Ruby linked to OpenSSL until 10.13 (switched to LibreSSL)
    - Ruby versions included on macOS make large jumps in versions
  - Linux: runtime provided by the OS may be too old
    - Needing to ever change switch base OS flavours could change your runtime version
  - Windows: Never came with Python 🙄
- Isolating the runtime also makes it easier to isolate packages

# Isolate runtime / interpreter

# Isolate runtime / interpreter

- Python
  - pyenv

# Isolate runtime / interpreter

- Python
  - pyenv
- Ruby
  - RVM
  - rbenv
  - ruby-install + chruby

# Isolate runtime / interpreter

- Python
  - pyenv
- Ruby
  - RVM
  - rbenv
  - ruby-install + chruby
- Node
  - NVM

# Isolate runtime / interpreter

- Python
  - pyenv
- Ruby
  - RVM
  - rbenv
  - ruby-install + chruby
- Node
  - NVM
- Go
  - GVM

# Isolate runtime / interpreter

- Python
  - pyenv
- Ruby
  - RVM
  - rbenv
  - ruby-install + chruby
- Node
  - NVM
- Go
  - GVM
- Or... Docker



**Repo  
mirrors**

**Version pinning**

**Isolated runtime**

**Isolated packages**

# Version pinning

- No `>=1.2`, only `==1.2.3`
- Remove all surprises about versions
- Every version update becomes intentional
  - Easier to correlate via Git history any bugs or regressions, and understand context of why a package was updated
- Can also pin to specific forks, internal or patched versions of libraries

# Version pinning

- With Bundler (Ruby) you'll get this for free
- pip (Python) also lets you specify exact package versions (==), although pip doesn't have a true dependency resolver
- Resist the urge to define Python dependencies loosely, even if it is more tedious to update requirements files

# Version pinning

**pip install --upgrade pip ?**

# Version pinning

**pip install --upgrade pip ?**

```
>>> pip.__version__  
'9.0.1'  
>>> from pip import req  
>>>
```

# Version pinning

**pip install --upgrade pip ?**

```
>>> pip.__version__  
'9.0.1'  
>>> from pip import req  
>>>
```

```
>>> pip.__version__  
'10.0.1'  
>>> from pip import req  
Traceback (most recent call last):  
  File "<stdin>", line 1, in <module>  
ImportError: cannot import name req  
>>>
```



**Repo  
mirrors**

**Version pinning**

**Isolated runtime**

**Isolated packages**


# NPM (part 2)

# left-pad

## How one developer just broke Node, Babel and thousands of projects in 11 lines of JavaScript

Code pulled from NPM – which everyone was using

By [Chris Williams, Editor in Chief](#) 23 Mar 2016 at 01:24

167  SHARE ▼



Careful, careful ... Don't fumble this like the JS world (Credit: Claus Rebler)

**Updated** Programmers were left staring at broken builds and failed installations on Tuesday after someone toppled the Jenga tower of JavaScript.

# left-pad proof our CI #1976

New issue

**Closed** carolynvs opened this issue 25 days ago · 2 comments



carolynvs commented 25 days ago • edited

Contributor + 🧑

Our CI builds were taken out by golint's go metadata server being down last night. We can vendor our go tools with dep so that this doesn't happen again.

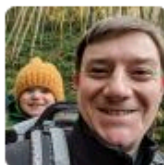
- ✓ Remove workaround from #1975
- ✓ Vendor go tools installed in our build image with dep, and just compile them in the Dockerfile



carolynvs added the build label 25 days ago



carolynvs self-assigned this 25 days ago



n3wscott commented 22 days ago

Member + 🧑

what does "left-pad" refer to?



logicfox commented 20 days ago

+ 🧑

@n3wscott look up NPM leftpad fiasco

Assignees

carolynvs

Labels

build

Projects

None yet

Milestone

No milestone

Notifications

🔔 Subscribe

You're not receiving notifications from this thread.

3 participants



On March 22nd 2016, a terrible tragedy befell the Node.JS community. A popular microframework for robust string expansion, ``left-pad``, [was removed from npmjs.com](#).

This resulted in broken deploys worldwide, a sudden and complete inability to appropriately zero-pad the fractional real cent value of many stock options, and untold billions of dollars in economic damage to the still nascent startup ecosystem of String Manipulation as a Service.

(From <http://left-pad.io>)

➔ ~ curl 'https://api.left-pad.io/?len=20&str=moar padding'

{"str": "moarpadding"}%

# Mirroring

- 1. One less point of failure**

# Mirroring

**2. It's (probably) faster**

# Mirroring

## 3. Can host internal-only packages

### requirements.txt

```
--index-url https://org.my.pypi/simple/

my-internal-package==18.06.07
pathlib2==2.3.2
pexpect==4.5.0
selenium==3.12.0
```

### Gemfile

```
BASE_URL = 'http://org.my.gems'

source BASE_URL
gem 'rails', '~>5'
gem 'xcode-install', '2.4.0'

source "#{BASE_URL}/private" do
  gem 'private-gem', '~> 1.1'
end
```

# Mirroring

**4. You have backups / archives to roll back to**

# Mirroring

# Mirroring

- Python
  - devpi (caches on demand, private packages, mirrors packages from anywhere, not only pypi)
  - Bandersnatch (caches all of pypi up front - ~500GB)

# Mirroring

- Python
  - devpi (caches on demand, private packages, mirrors packages from anywhere, not only pypi)
  - Bandersnatch (caches all of pypi up front - ~500GB)
- Ruby
  - gemstash (part of the Bundler org, caches on demand, private packages)

# Mirroring

- Python
  - devpi (caches on demand, private packages, mirrors packages from anywhere, not only pypi)
  - Bandersnatch (caches all of pypi up front - ~500GB)
- Ruby
  - gemstash (part of the Bundler org, caches on demand, private packages)
- Artifactory Pro does 20+ package manager mirrors/registry types, and is \$\$\$

# Mirroring

# Mirroring

- <https://github.com/gaoyifan/homebrew-bottle-mirror> (Yifan Gao)

# Mirroring

- <https://github.com/gaoyifan/homebrew-bottle-mirror> (Yifan Gao)

```
docker run -itd \  
  --name=homebrew-bottles \  
  -v /bottle-mirror/bottles:/srv/data \  
  gaoyifan/homebrew-bottle-mirror
```

# Mirroring

- <https://github.com/gaoyifan/homebrew-bottle-mirror> (Yifan Gao)

```
docker run -itd \  
  --name=homebrew-bottles \  
  -v /bottle-mirror/bottles:/srv/data \  
  gaoyifan/homebrew-bottle-mirror
```

```
> update package info...
```

```
From git://github.com/homebrew/homebrew-core
```

```
* branch          master      -> FETCH_HEAD
```

```
* [new branch]     master      -> origin/master
```

```
HEAD is now at 60e56c1 prometheus: update 2.3.0 bottle.
```

```
> RUN brew bottle-mirror...
```

```
==> ascii-3.18.high_sierra.bottle.tar.gz downloaded
```

```
==> libdvdnav-6.0.0.high_sierra.bottle.tar.gz downloaded
```

```
==> ascii-3.18.sierra.bottle.tar.gz downloaded
```

```
==> libdvdnav-6.0.0.sierra.bottle.tar.gz downloaded
```

```
==> makepp-2.0.high_sierra.bottle.tar.gz downloaded
```

```
==> ld64-97.17.snow_leopard.bottle.tar.gz downloaded
```

# Mirroring

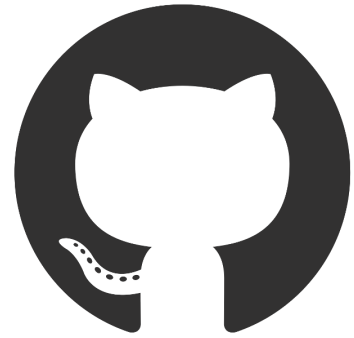
- <https://github.com/gaoyifan/homebrew-bottle-mirror> (Yifan Gao)

# Mirroring

- <https://github.com/gaoyifan/homebrew-bottle-mirror> (Yifan Gao)

```
HOMEbrew_BOTTLE_DOMAIN=https://my.bottle.mirror \  
brew install cocoapods jq libimobiledevice
```

# Thank you!



@timsutton



@tvsutton

<https://macops.ca/dependencies>