

Version Control | Revolution

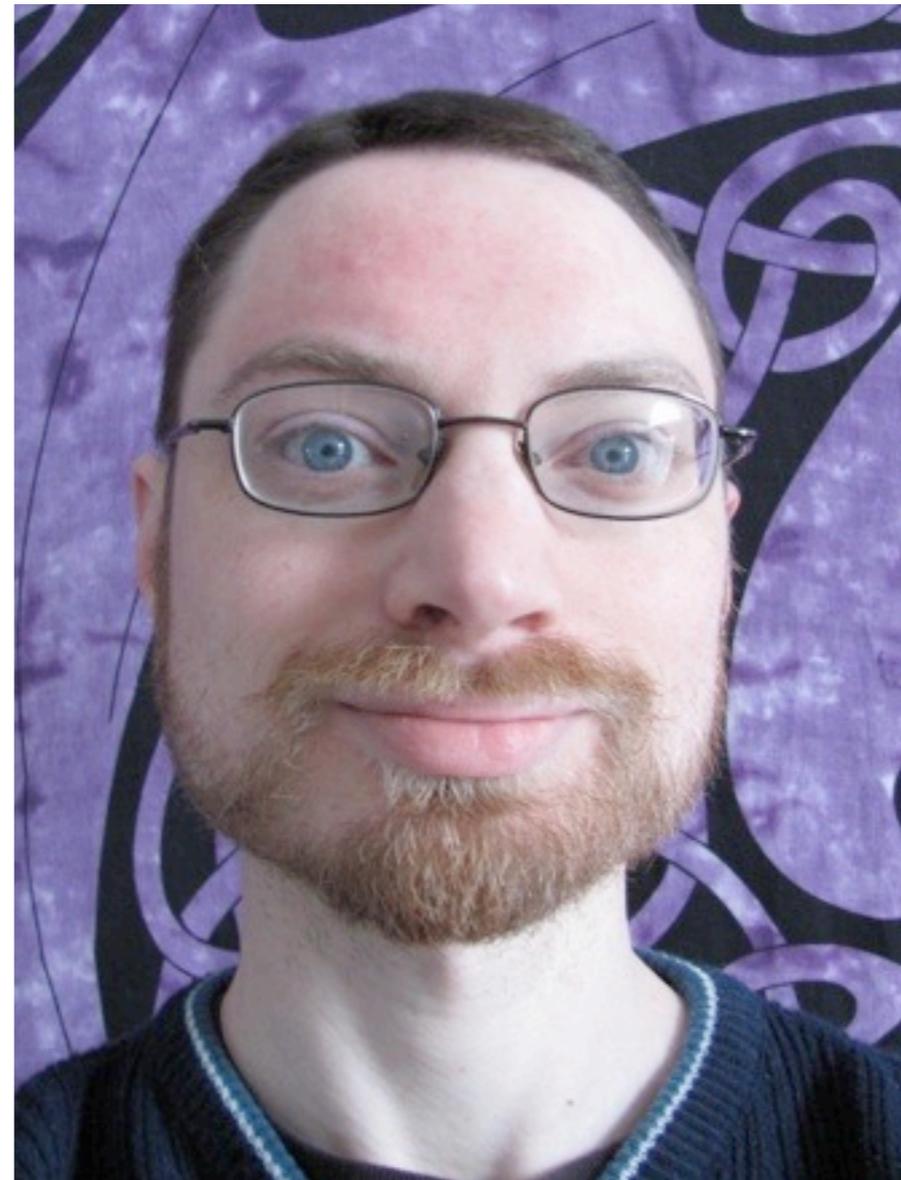
<http://www.delicious.com/rauros/amp>

Justin Love

slides@JustinLove.name

<http://JustinLove.name>

[@wondible](#)



**I am a
Mercurial
User**

THIS
should not
matter



Manifesto

We want a piece of software that:

- 1** Is Free of religious devotion to one VCS

Overview

- What is Amp?
- Why Amp?
- What will Amp do for me?
- How do I use Amp?
- How can I help?

My VCS Journey

College

1994-1998

~~VCS~~

vss

SVN

Interface

git

hg

arcane

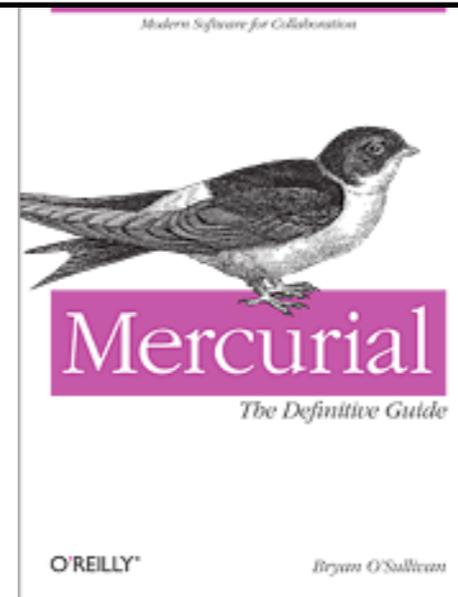
clean

2007

Documentation

git

hg



2007

Portability

git

hg

~~windows~~



2007

HG

github: 2008

The screenshot shows a web browser window titled "JustinLove's Profile - GitHub". The address bar contains "http://github.com/JustinLove". The browser's bookmark bar includes "rauros's Bookmarks ...", "follow cost", "LiveJournal.com - St...", "ING DIRECT USA - S...", "Online Banking : Wel...", "eBay - New & used e...", and "Weather Street: SOU...". The GitHub logo "github SOCIAL CODING" is in the top left. Navigation links include "Home", "Login", "Pricing and Signup", "Training", "Gist", and "Blog". The profile header for Justin Love shows a profile picture, the name "JustinLove (Justin Love)", and the text "You're not logged in!" with "Login" and "Pricing & Signup" buttons. The profile details list: Name: Justin Love; Email: git@JustinLove.name; Website/Blog: http://rauros.livejournal.com/; Company: Computer Generated Dreams; Location: South Elgin, IL; Member Since: Feb 13, 2009. On the right, it displays "12 public repos" and "5 followers", with a note "Following 1 coder and watching 19 repositories view all →" and a small profile picture of the followed coder. The "Public Repositories (12)" section features a filter box and a repository entry for "amp-front" (Ruby, 1 eye, 2 forks), which is a fork of "michaedgar/amp-front". The description for "amp-front" is "Generic front-end for Amp." and "Last updated October 22, 2010". The "Public Activity" section shows a recent activity: "JustinLove opened pull request 8 on michaedgar/amp-front October 22, 2010", with a description: "Collecting argument and option manipulation into ArgumentOptions" and "4 commits with 271 additions and 106 deletions".

JustinLove's Profile - GitHub

git http://github.com/JustinLove

rauros's Bookmarks ... follow cost LiveJournal.com - St... ING DIRECT USA - S... Online Banking : Wel... eBay - New & used e... Weather Street: SOU...

git JustinLove's Profile - GitHub

github
SOCIAL CODING

Home | Pricing and Signup | Training | Gist | Blog
Login

 **JustinLove** (Justin Love)

You're not logged in! [Login](#) [Pricing & Signup](#)

Name Justin Love
Email git@JustinLove.name
Website/Blog <http://rauros.livejournal.com/>
Company Computer Generated Dreams
Location South Elgin, IL
Member Since Feb 13, 2009

12 public repos **5** followers

Following 1 coder and watching 19 repositories [view all →](#)



Public Repositories (12)

 **amp-front** Ruby  1  2
Forked from [michaedgar/amp-front](#)

Generic front-end for Amp.
Last updated October 22, 2010

Public Activity 

 **JustinLove** opened pull request 8 on [michaedgar/amp-front](#) **October 22, 2010**
Collecting argument and option manipulation into ArgumentOptions
4 commits with 271 additions and 106 deletions

What Is



Interfaces

Repositories

git

.git

hg

.hg

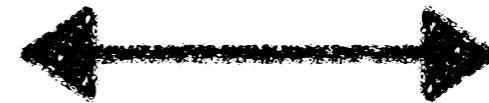
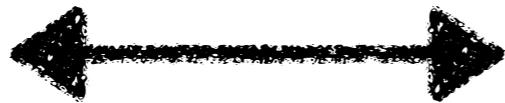
?

?



Full Mercurial Support

hg



.hg



**Flexible
Commands
Written in
Ruby**

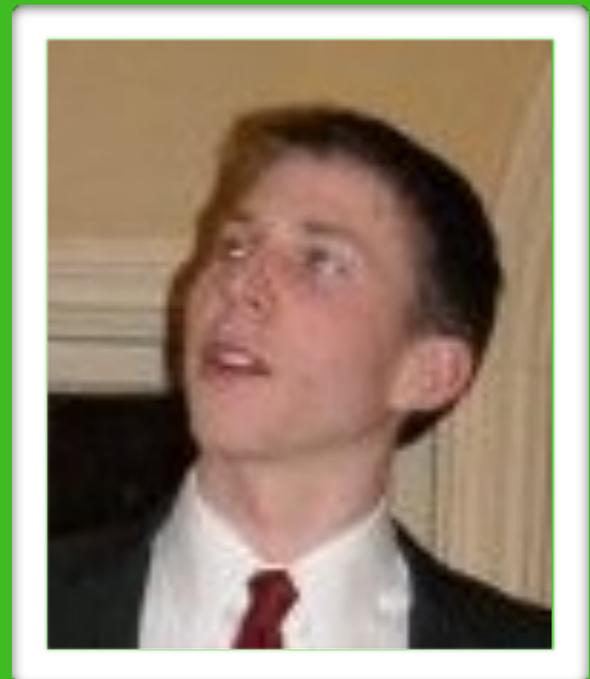




Michael 'adgar' Edgar

carbonara

Ari 'seydar' Brown



“Back in the day, someone pointed out that **Rails is slang for cocaine**. So, obviously, we needed to name it after a drug.”

Ari Brown

Ruby **On** **Rails?**

“We looked online for nicknames for meth, and chose amp because it was short (3 letters) and wasn't blatantly drug related.”

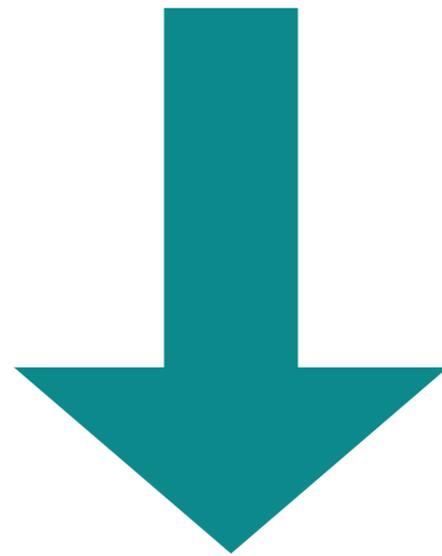
Ari Brown

\$ amp clone hg amp



Existing
Implementation
in a
Dynamic
Language

Mercurial (GPL)



Amp (GPL)



“Long story short: Amp kicks ass and you should use it. It's still growing and not bug-free.

If you're on Ruby 1.9, beware of Unicode”

Michael Edgar



John Locke

Amp 0.5.3



<http://en.wikipedia.org/wiki/File:JohnLocke.png>





Version Control Revolution

Home

About Amp

Get Amp

Learn Amp

Contribute t

Get Amp v0.5.2

1. Get **Ruby**. 2. `sudo gem install amp --no-wrappers`

What is Amp:

Amp aims to change the way we approach VCS.

Amp is:

- [Mercurial](#) in [Ruby](#). 100% compatible with hg.
- Uniquely [customizable](#)
- Superbly [documented](#) with [YARD](#)
- Free and Open-Source ([GPLv2](#))
- [Performance](#) Focused
- Dependency-free (you just need Ruby!)

Big Plans for Amp:

Amp has bigger dreams. Here's what we *want* to do:

- [git](#), [bazaar](#), [svn](#), [cvs](#), [darcs](#) In [Ruby](#). 100% Compatible.
- Common API to all repository formats.
- [Command system](#) independent of repository format
- Workflows matching each major VCS system
- Run on [Windows](#)
- Demonstrate Ruby [performance](#)
- A GUI application on top of all these features

Ambitious? Perhaps. That's why [we'd like your help](#).

Why Amp?

Amp is unlike any other VCS software to date. **Amp**'s novel command system gives you the flexibility to customize and personalize the way you develop software. Create in your own workspace with full [Mercurial](#) compatibility, and no external dependencies.



```
command "status" do |c|
  c.default : "no-color", true
end
```

Don't like how a command works out-of-the-box? Change the default settings. It's almost not even code

```
changeset: 661:7dbef9c17c2e
user:      seydar
date:      Sun Aug 08 2010
summary:   updated bug 24's...
```



```
$ amp pull git  
$ amp merge
```



```
changeset: 662:c6fb001fb520
user: Justin Love
date: Tue Aug 17 2010
summary: use ruby format...
```



RubyConf
CFP Close
2010-08-30

**A funny thing
happened on
the way to
RubyConf...**

```
changeset: 661:7dbef9c17c2e
user:      seydar
date:      Sun Aug 08 2010
summary:   updated bug 24's...
```



<http://amp.carboni.ca/>

404

“We're happy to let you know that your proposal, "AMP: Cross-repo version control in Ruby.", has been selected for inclusion in the program.”

David A. Black
2010-09-20

```
changeset: 661:7dbef9c17c2e
user:      seydar
date:      Sun Aug 08 2010
summary:   updated bug 24's...
```



😊 **amp_redux**

Ruby

A meta-VCS in Ruby, harnessing Ruby's strengths

amp-redux?

```
$ amp pull git  
requesting all changes  
adding changesets  
adding manifests  
adding file changes  
transaction abort!  
interrupted!
```



“FUN FACT!

Getting **git** to play right is **much more complicated** than whatever that law is that says everything is super complicated.”

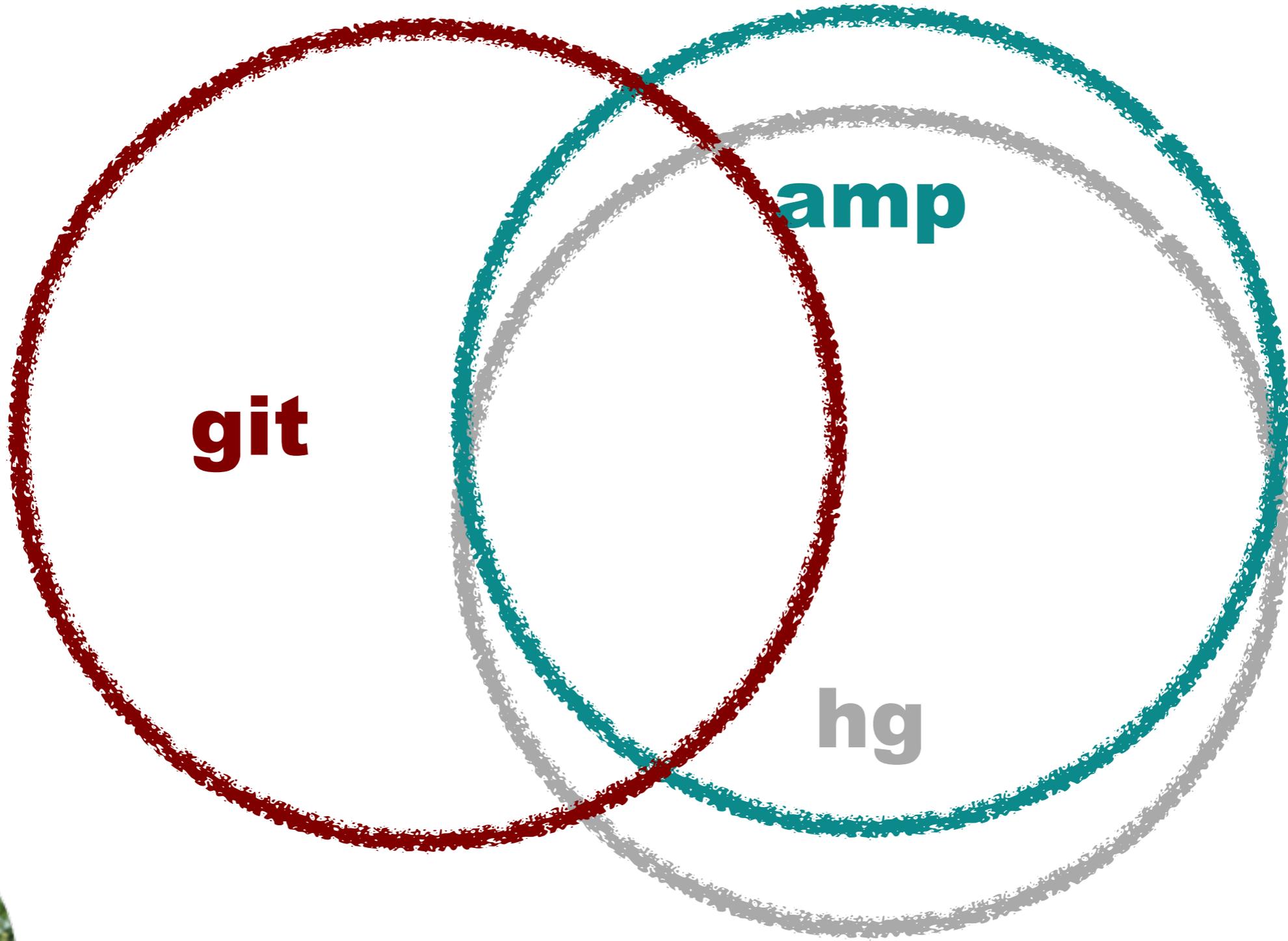
Ari Brown



amp

hg





“HG focuses on the implementation (filelogs, change-logs, indexes/indices), and we wanted to focus on changesets, repositories, and stuff like that”

Ari Brown



“It's like an old car that needs to get scrapped.”

“it got us hundreds of thousands of miles.”

Ari Brown





Manifesto

We want a piece of software that:

6

We want to discover where the limits are on Ruby for use in large applications. Where is the language slow? Where is it fast? What techniques are effective?

“plan to throw one away;
you will, anyhow.”

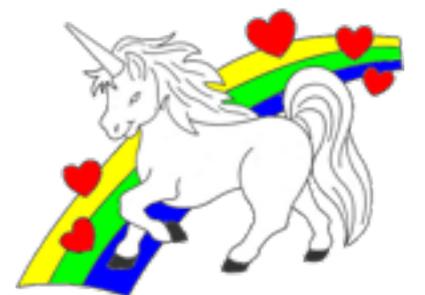
Fred Brooks

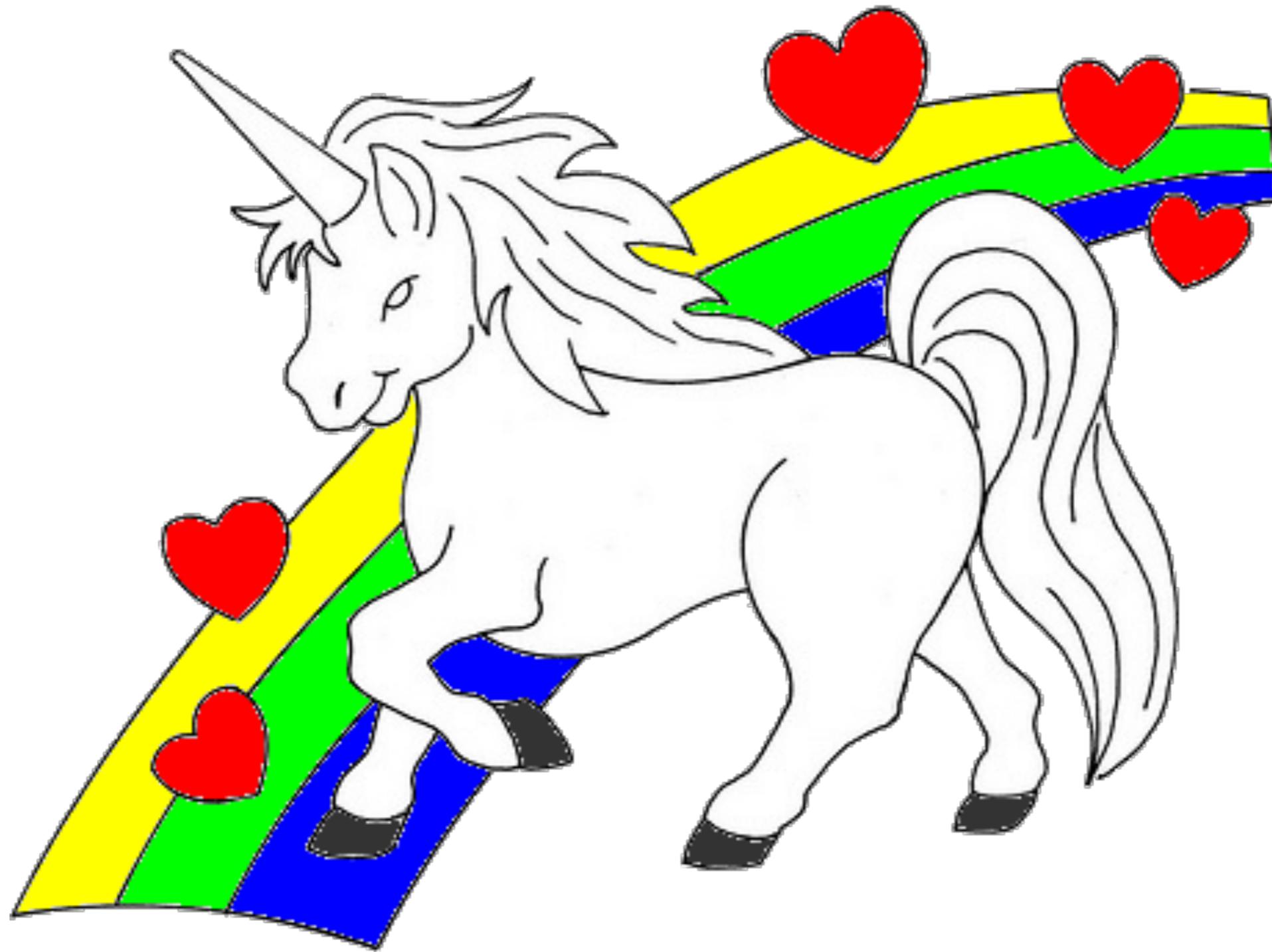
The Mythical Man-Month



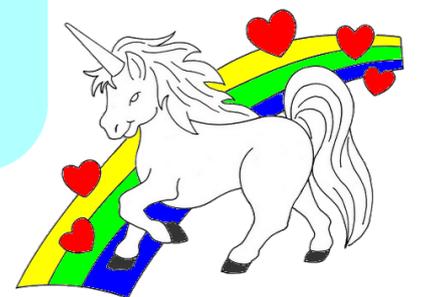


Redux





Not backed by code



“I'm learning a shitton about software design and engineering which will pay back Amp in spades.”

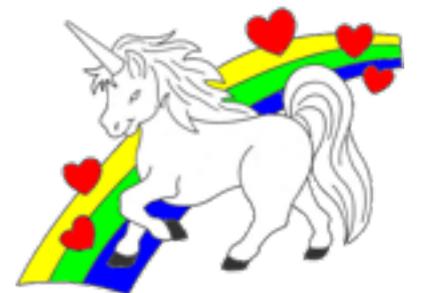
Michael Edgar



Interface



Implementation



Modular



Amp

Front



Amp

Core



Amp-hg

Amp-git



why



git

C

Mercurial

Python

Bazaar

Python

Darcs

Haskell

Monotone

C+++

SVN

C

SVK

Peri

C/++

3

Python

2

Perl

1

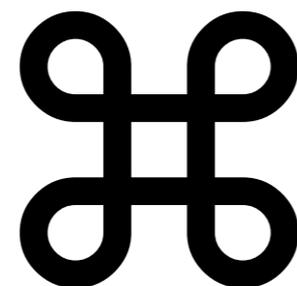
Haskell

1

Ruby?

<http://en.wikipedia.org/wiki/>

Comparison of revision control software

-F
(Find)

Ruby



Manifesto

We want a piece of software that:

5

We want to prove ruby is viable for large-scale applications, not just web frameworks. Can you think of any non-trivial actual applications written in ruby?

**Do we
really need
another
VCS?**



Ruby

צמח סמך





Rubinius



Sinatra

merlo



3

vss

svn

Mercurial

CVS

SVN

Git

**There
Will Be a
Next Thing**

How will



Mercurial

Repos





\$ amp



require 'amp'



git ↔ .git

hg ↔ .hg

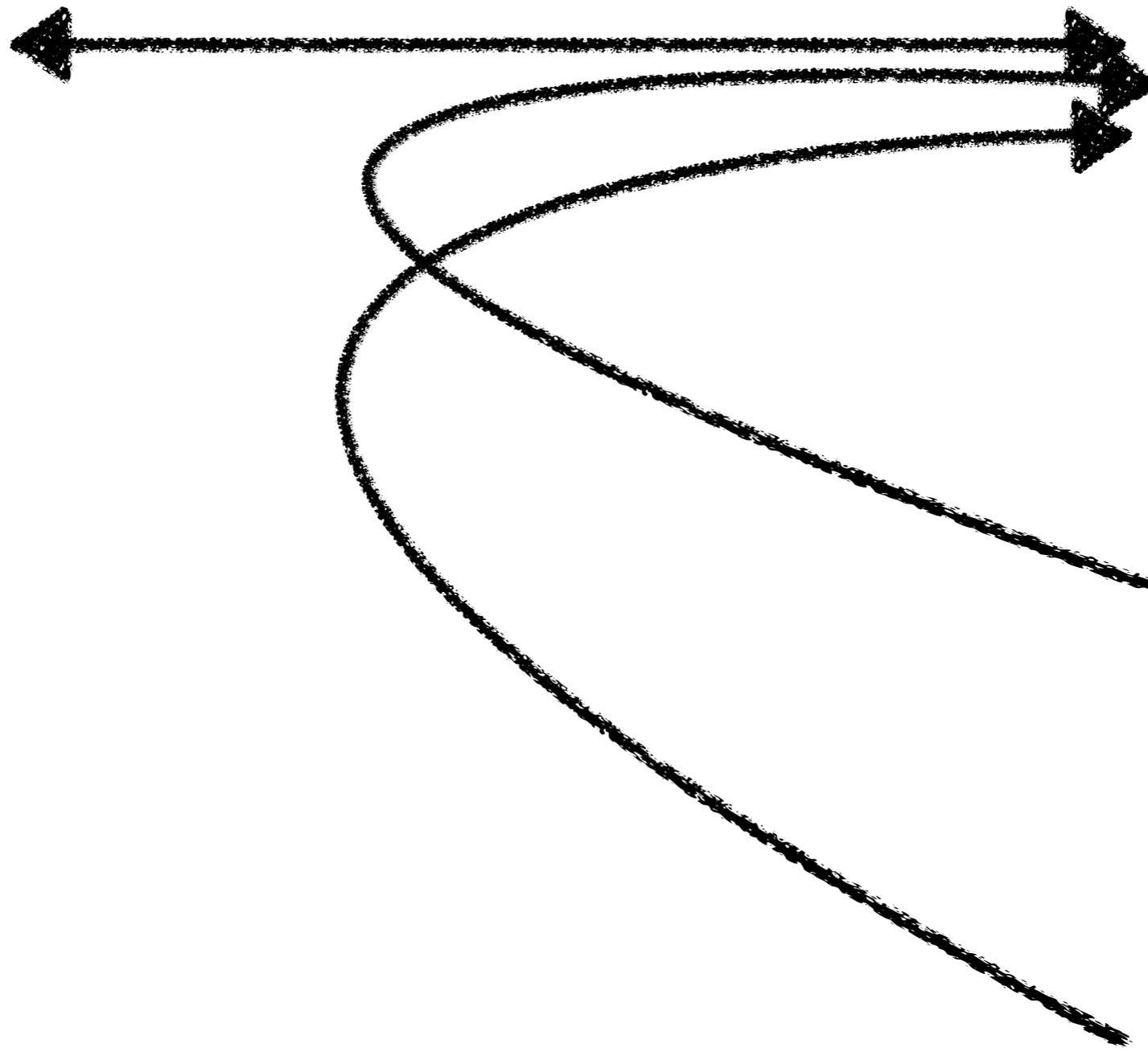
? ↔ ?

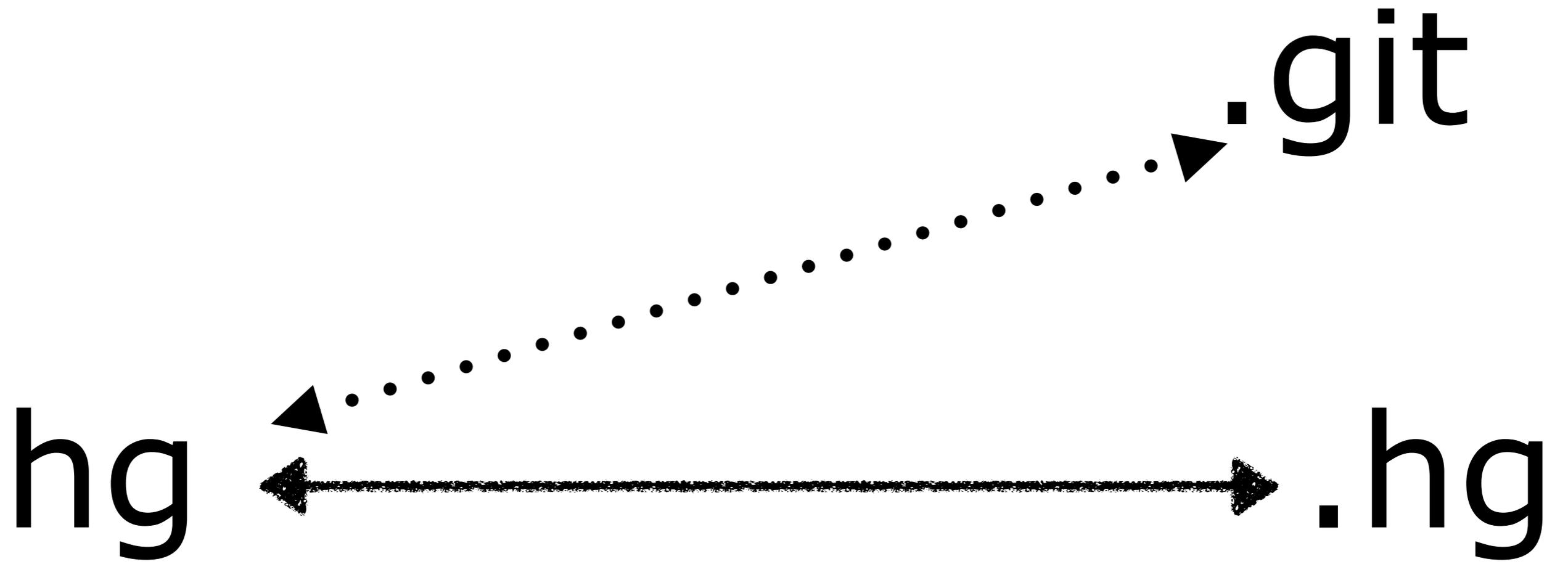
git

.git

.hg

?



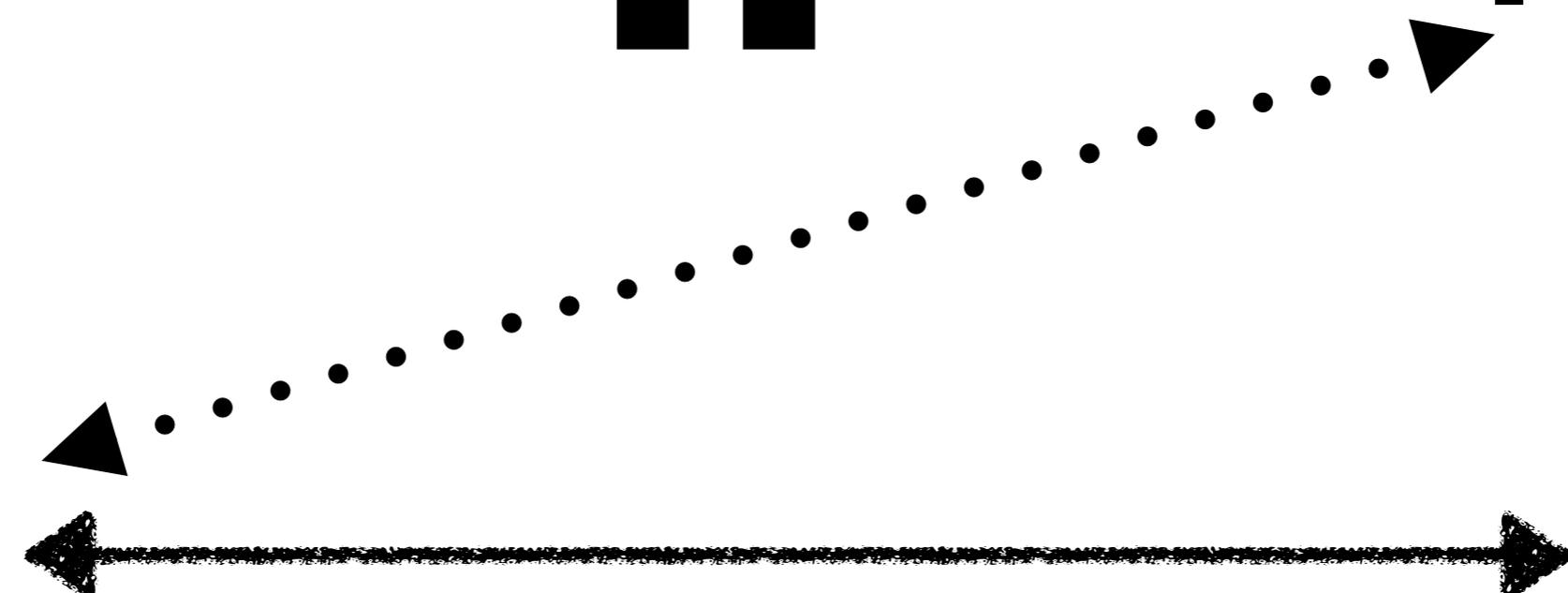


n²

.git

hg

.hg

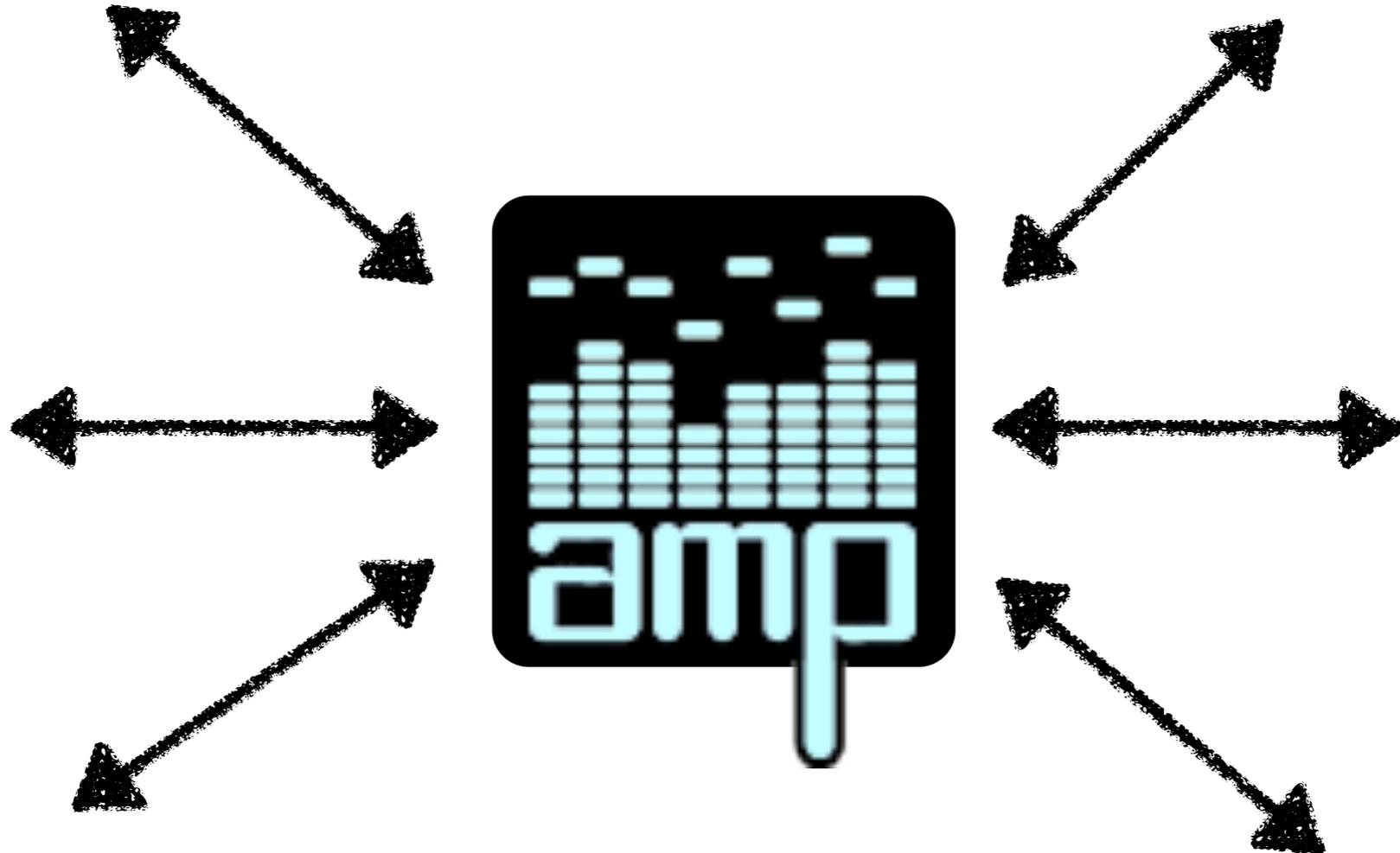


git

n+m

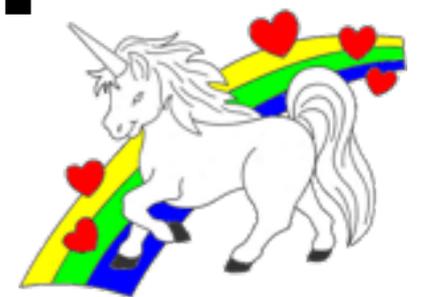
.git

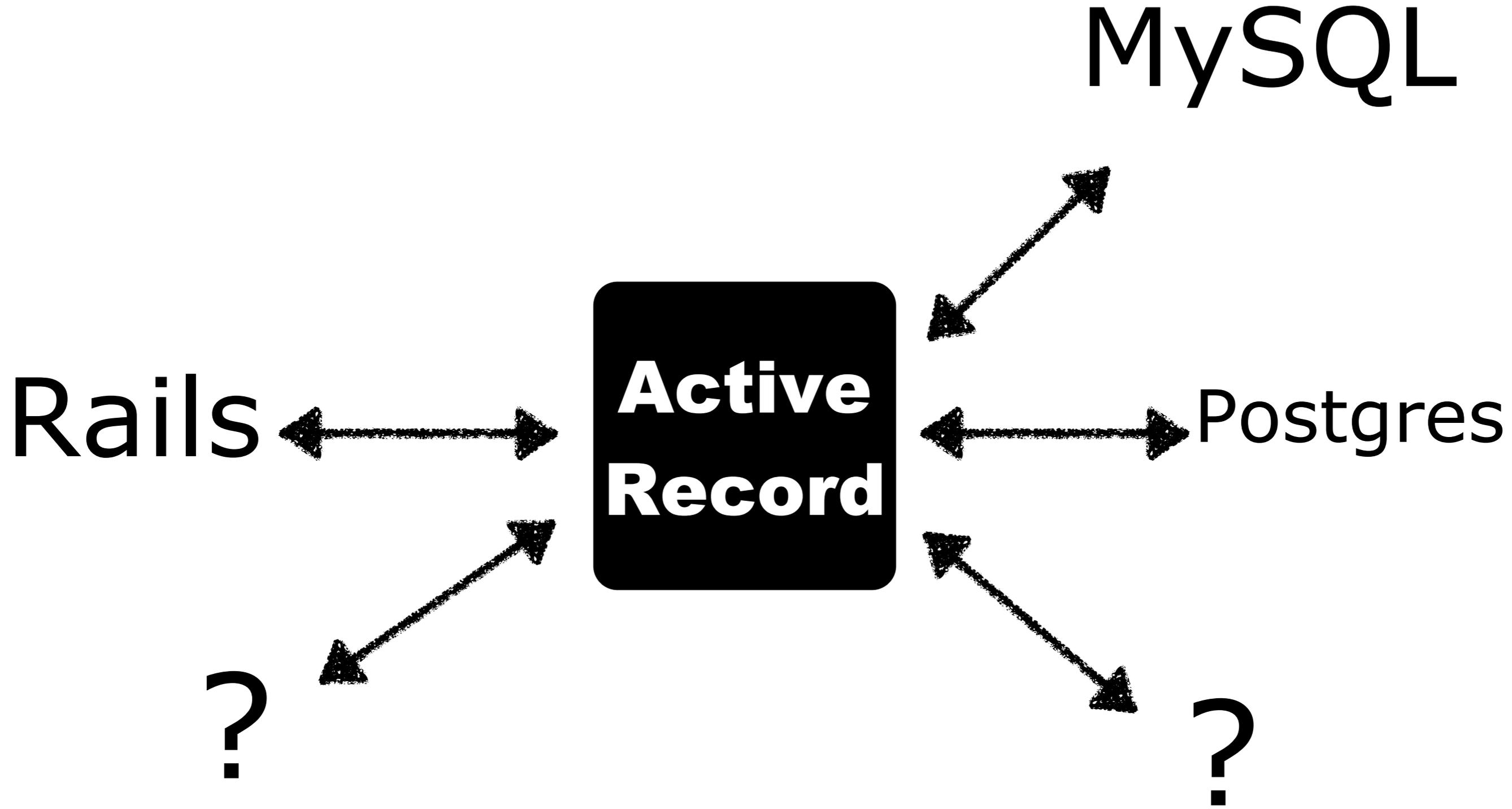
hg



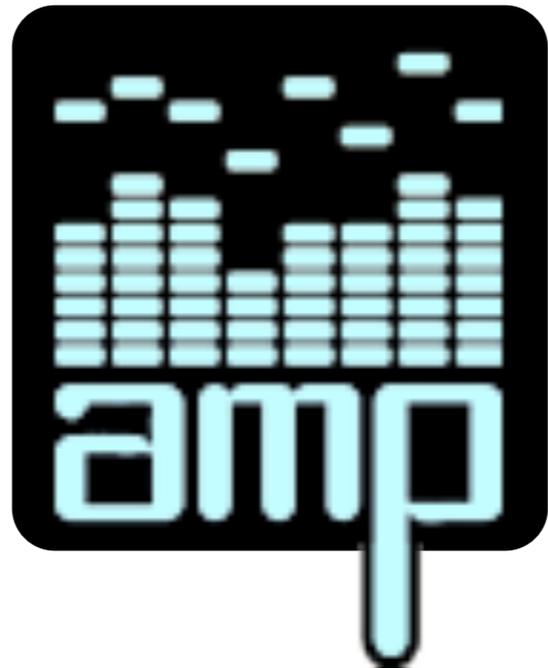
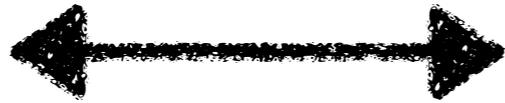
?

?

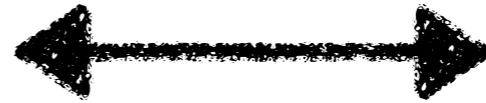




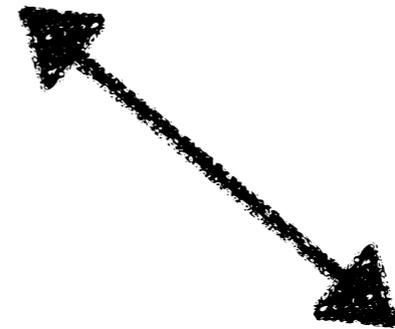
git



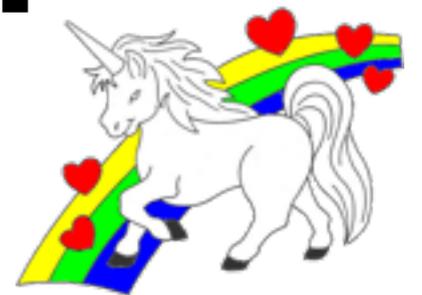
.git



.hg



?





Manifesto

We want a piece of software that:

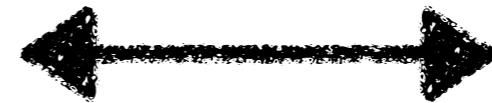
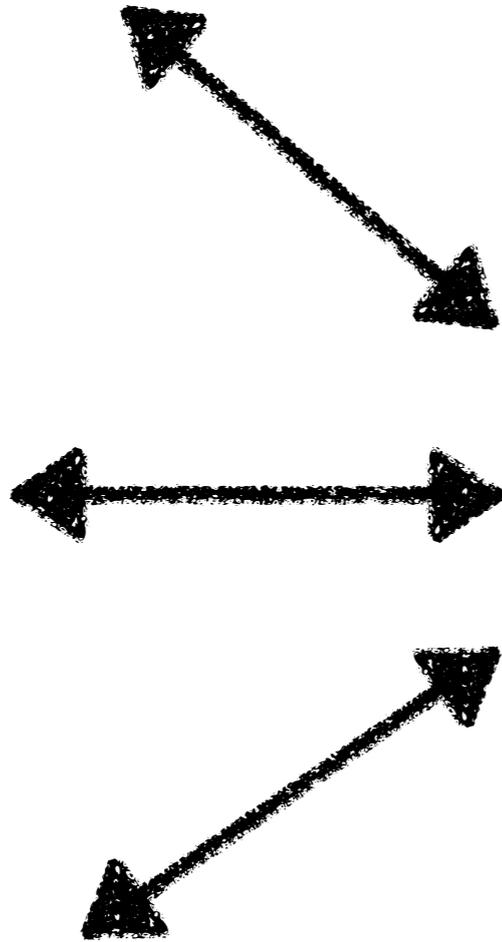
2

Lets us customize how we interact with it. If we want git's commmands because I used it the longest, let us use them with a mercurial repo.

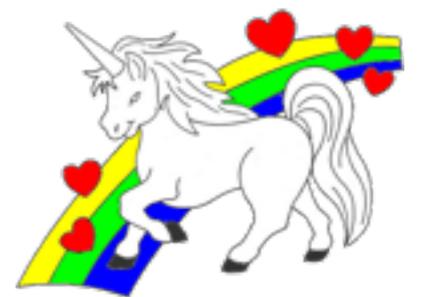
git

hg

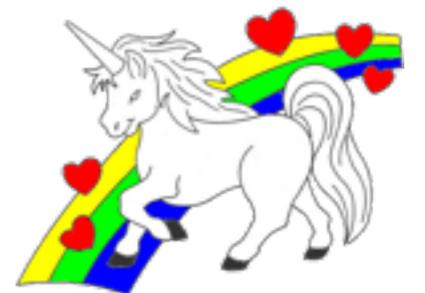
?



.git



Mix and Match



Rails 3

- raities
- actionmailer
- actionpack
- activemodel
- activerecord
- activeresource
- activesupport



AMP Redux

- amp-front
- amp-core
- amp-hg
- amp-git



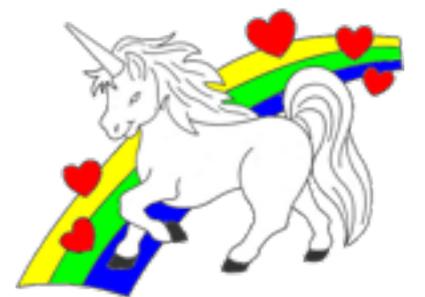
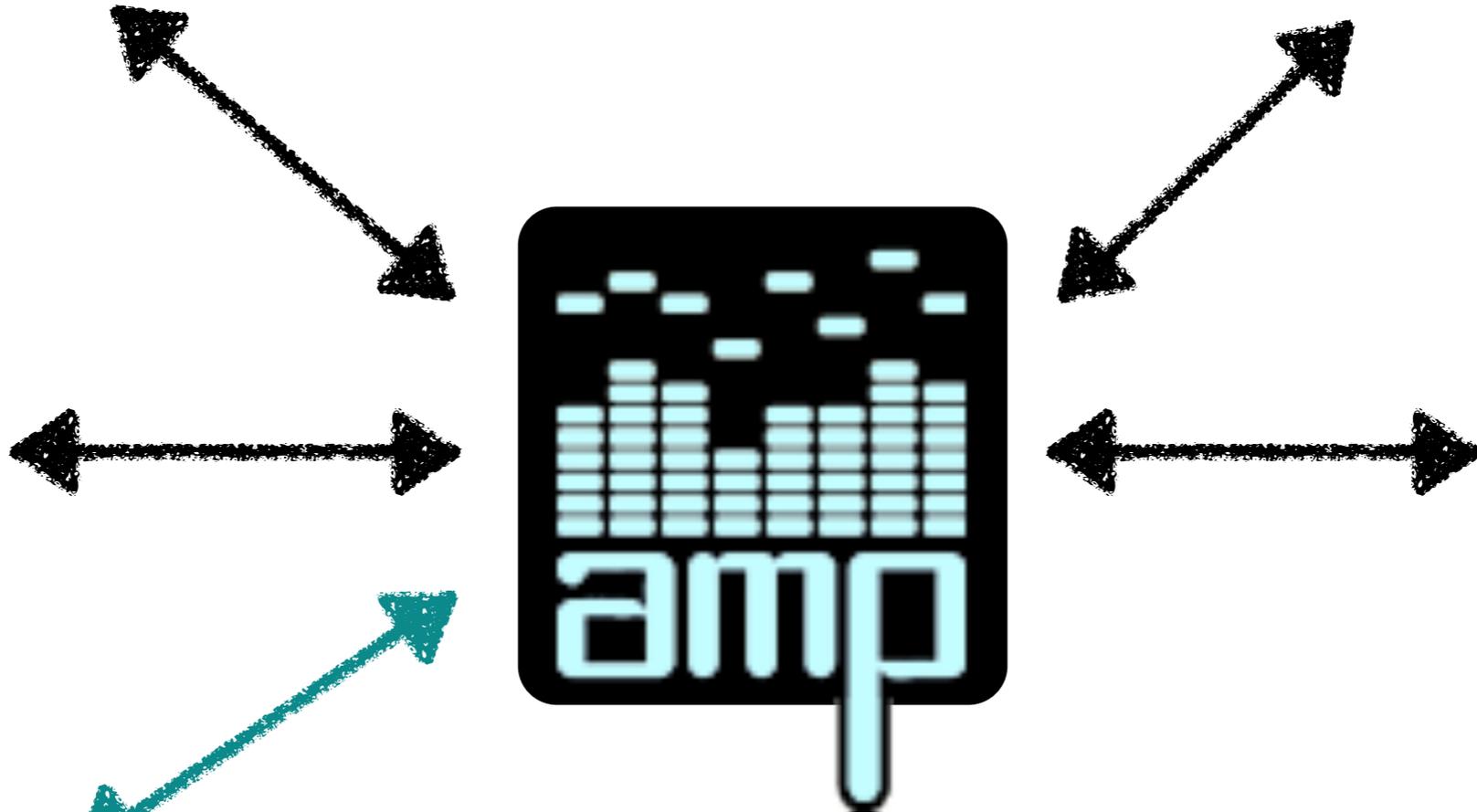
git

.git

hg

.hg

?

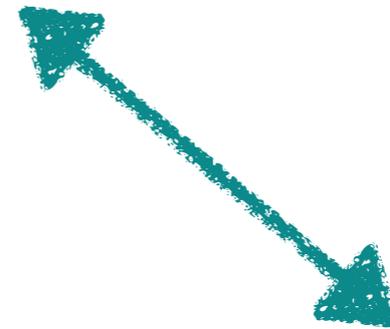
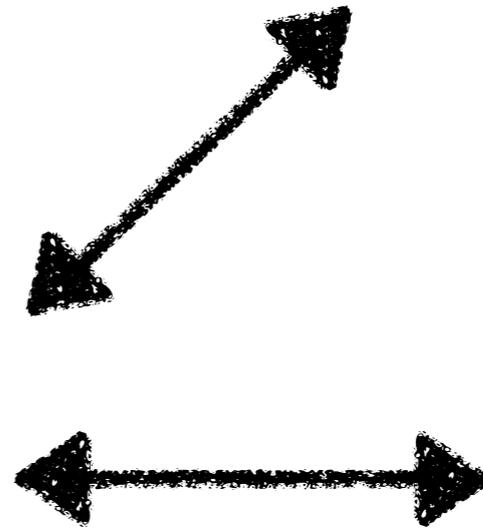
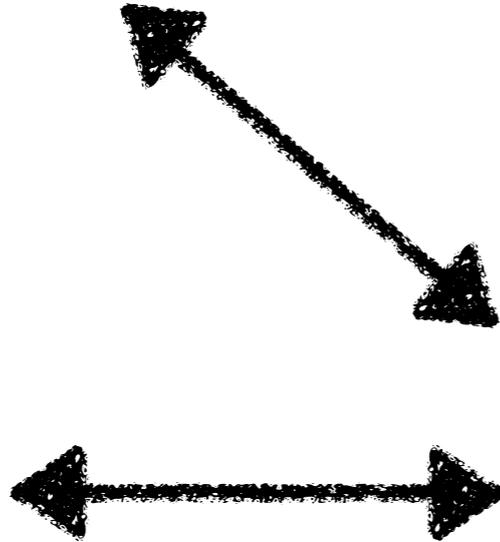


git

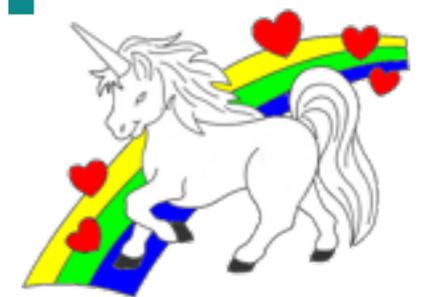
.git

hg

.hg



?





Manifesto

We want a piece of software that:

3

We want the extension we wrote for HG to work on my github project too. Because they're both DVCS, damnit.

git

.git

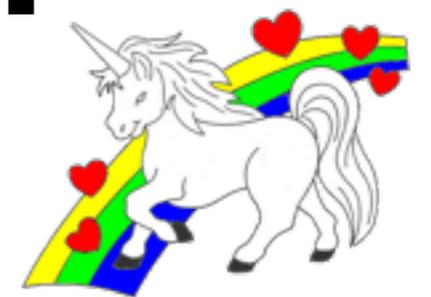
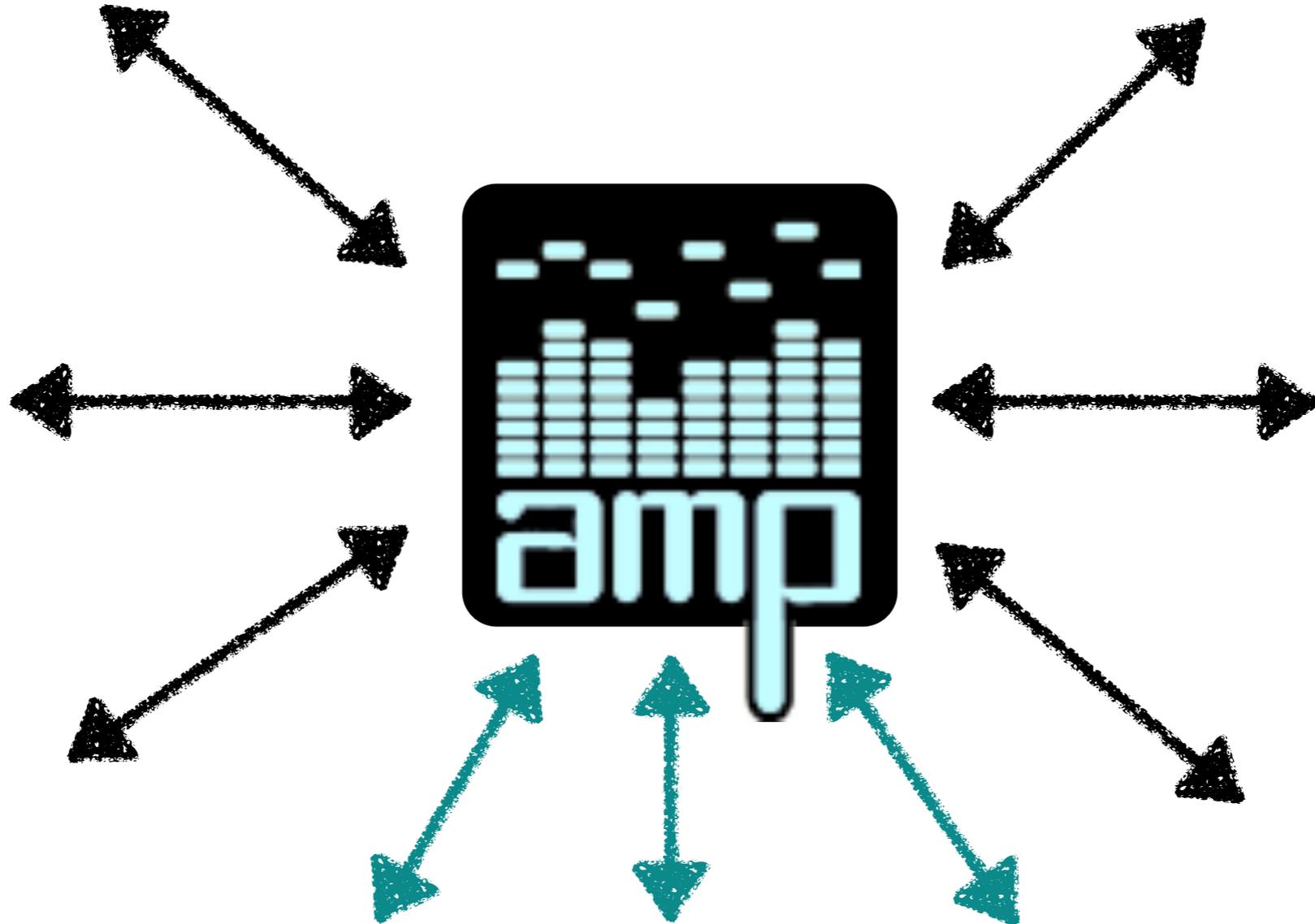
hg

.hg

?

?

extensions



**How do
I use**



```
$ gem install amp
```



```
$ amp help
```

```
$ amp add
```

```
$ amp commit
```

```
$ amp push
```





Manifesto

We want a piece of software that:

4

We want to be able to directly modify the commands they give us. Git's remove command is `git rm`. It should be dead simple to make `git remove` map directly to that, and we mean less than 1 minute of thought/work.

ampfile.rb



“One huge feature is **open commands**: the same syntax for **creating a command** will **re-open an existing one**, just like **classes in Ruby**.”

Michael Edgar



```

command :log do |c|
  c.workflow :hg
  c.desc "Prints the commit history."
  c.opt :verbose, "Verbose output", {:short => "-v"}
  c.opt :limit, "Limit how many revisions to show",
    {:short => "-l", :type => :integer, :default => -1}
  c.opt :template, "Which template to use while printing",
    {:short => "-t", :type => :string, :default => "default"}
  c.opt :no_output, "Doesn't print output (useful for benchmarking)"
  c.on_run do |options, args|
    repo = options[:repository]
    limit = options[:limit]
    limit = repo.size if limit == -1

    start = repo.size - 1
    stop = start - limit + 1

    options.merge! :template_type => :log
    start.downto stop do |x|
      puts repo[x].to_templated_s(options) unless options[:no_output]
    end
  end
end
end

```



```
command : log do |c|
```



c.workflow :hg



```
c.desc "Prints the  
commit history."
```



```
c.opt :verbose, "Verbose output", {:short => "-v"}
c.opt :limit, "Limit how many revisions to show",
  {:short => "-l", :type => :integer, :default => -1}
c.opt :template,
  "Which template to use while printing",
  {:short => "-t", :type => :string,
   :default => "default"}
c.opt :no_output, "Doesn't print output
  (useful for benchmarking)"
```



Trollop



```
c.on_run do |options, args|
  repo = options[:repository]
  limit = options[:limit]
  limit = repo.size if limit == -1

  start = repo.size - 1
  stop = start - limit + 1

  options.merge! :template_type => :log
  start.downto stop do |x|
    puts repo[x].to_templated_s(options)
    unless options[:no_output]
    end
  end
end
```



```
c.on_run do |options, args|
  repo = options[:repository]
  limit = options[:limit]
  limit = repo.size if limit == -1

  start = repo.size - 1
  stop = start - limit + 1

  options.merge! :template_type => :log
  start.downto stop do |x|
    puts repo[x].to_templated_s(options)
    unless options[:no_output]
    end
  end
end
```



```
c.on_run do |options, args|
  repo = options[:repository]
  limit = options[:limit]
  limit = repo.size if limit == -1

  start = repo.size - 1
  stop = start - limit + 1

  options.merge! :template_type => :log
  start.downto stop do |x|
    puts repo[x].to_templated_s(options)
    unless options[:no_output]
    end
  end
end
```



User Command: Billing Time



```
$ amp billing --limit 30
2010-09-21 0.25
2010-09-02 2.22
2010-09-01 0.25
2010-08-31 4.73
7.46
```



```
$ amp help billing
Estimate time spent
```

```
--limit, -l <i>:      Limit how many revisions to show
--rate, -r <i>:       Billing Rate
--help, -h:          Show this message
```



...

archive

billing

bisect

...

Create...

Estimate...

subdivision...



```

command :billing do |c|
  c.desc "Estimate time spent"
  c.opt :limit, "Limit how many revisions to show",
    {:short => "-l", :type => :integer}
  c.opt :rate, "Billing Rate", {:type => :integer}

  c.on_run do |options, args|
    # boilerplate
    repo = options[:repository]
    limit = options[:limit] || repo.size
    rate = options[:rate] || 1

    last = repo.size - 1
    first = last - limit + 1

    # calculate time
    hours = repo.to_a[first,last].each_cons(2).map(&BillingTime.method(:new))

    # daily summary
    days = Hash.new(0)
    hours.each {|b| days[b.date] += b.billable}
    puts days.keys.sort.reverse.map {|k| "#{k} #{days[k]}"}

    # total
    bills = hours.map {|b| b.billable * rate}
    puts bills.inject {|a,b| a + b}
  end
end
end

```



```
command :billing do |c|  
  c.desc  
  "Estimate time spent"
```



```
c.opt :limit,  
"Limit how many revisions to show",  
  {:short => "-l",  
   :type => :integer}  
c.opt :rate, "Billing Rate",  
  {:type => :integer}
```



```
c.on_run do |options, args|  
  # boilerplate  
  repo = options[:repository]  
  limit = options[:limit] ||  
    repo.size  
  rate = options[:rate] || 1  
  
  last = repo.size - 1  
  first = last - limit + 1
```



```
# calculate time
hours = repo[first..last].
  each_cons(2).
  map(&BillingTime.method(:new))
```



```
# daily summary
days = Hash.new(0)
hours.each {|b|
  days[b.date] += b.billable}
puts days.keys.sort.reverse.
  map {|k| "#{k} #{days[k]}"}
```



```
# total
bills = hours.map { |b|
  b.billable * rate}
puts bills.inject { |a,b| a + b }
```



How can I help





Manifesto

We want a piece of software that:

- 7** We want to be an example of proper documentation.



Get Amp v0.5.2
1. Get **Ruby**. 2. `sudo gem install amp --no-wrappers`

What is Amp:

Amp aims to change the way we approach VCS.

Amp is:

- [Mercurial](#) in [Ruby](#). 100% compatible with hg.
- Uniquely [customizable](#)
- Superbly [documented](#) with [YARD](#)
- Free and Open-Source ([GPLv2](#))
- [Performance](#) Focused
- Dependency-free (you just need Ruby!)

Big Plans for Amp:

Amp has bigger dreams. Here's what we *want* to do:

- [git](#), [bazaar](#), [svn](#), [cvs](#), [darcs](#) In [Ruby](#). 100% Compatible.
- Common API to all repository formats.
- [Command system](#) independent of repository format
- Workflows matching each major VCS system
- Run on [Windows](#)
- Demonstrate Ruby [performance](#)
- A GUI application on top of all these features

Ambitious? Perhaps. That's why [we'd like your help](#).

Why Amp?

Amp is unlike any other VCS software to date. **Amp**'s novel command system gives you the flexibility to customize and personalize the way you develop software. Create in your own workspace with full [Mercurial](#) compatibility, and no external dependencies.



```
command "status" do |c|
  c.default : "no-color", true
end
```

Don't like how a command works out-of-the-box? Change the default settings. It's almost not even code

MANIFESTO



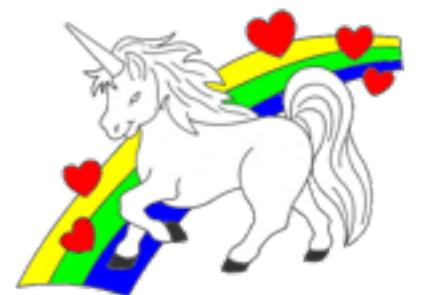
STYLE



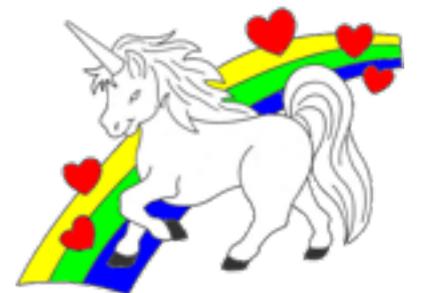
TODO



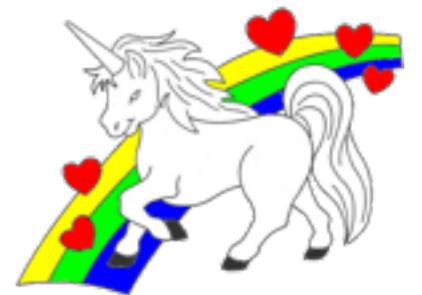
Amp-front



Option Parsing



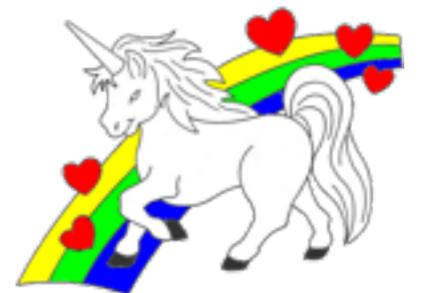
Help



commmmand

==

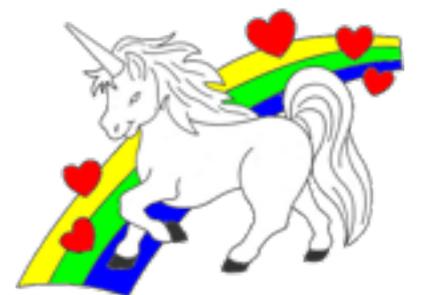
class



Amp-core



Plugins



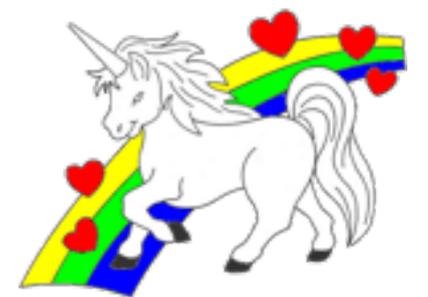
Generic Repository



Amp-hg



GPL



Amp-git



Multiple Backends

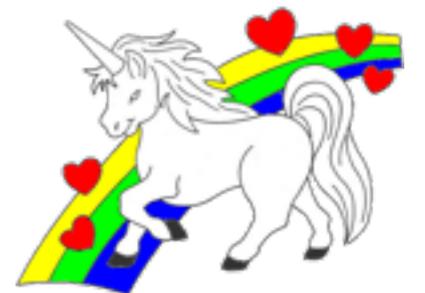
?



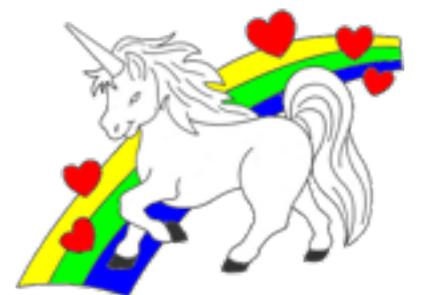
Commmmand

line

wrapperr



Amp-grit?

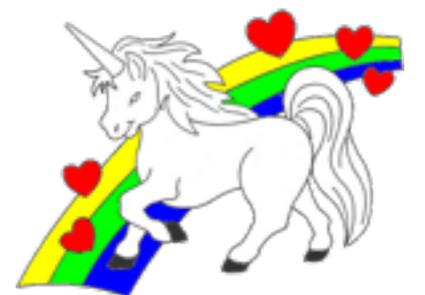


Pure Ruby



Amp-git-C

?



Review



```
$ gem install amp
```





Manifesto

We want a piece of software that:

- 1** Is Free of religious devotion to one VCS



Manifesto

We want a piece of software that:

2

Lets us customize how we interact with it. If we want git's commmands because I used it the longest, let us use them with a mercurial repo.



Manifesto

We want a piece of software that:

3

We want the extension we wrote for HG to work on my github project too. Because they're both DVCS, damnit.



Manifesto

We want a piece of software that:

4

We want to be able to directly modify the commands they give us. Git's remove command is ``git rm``. It should be dead simple to make ``git remove`` map directly to that, and we mean less than 1 minute of thought/work.



Manifesto

We want a piece of software that:

5

We want to prove ruby is viable for large-scale applications, not just web frameworks. Can you think of any non-trivial actual applications written in ruby?



Manifesto

We want a piece of software that:

6

We want to discover where the limits are on Ruby for use in large applications. Where is the language slow? Where is it fast? What techniques are effective?



Manifesto

We want a piece of software that:

- 7** We want to be an example of proper documentation.

**These are
our goals**

These
are our
tools



Ruby

RSpec

Cucumber 

YARD

Yay! A Ruby Documentation Tool

Now
is the
time.



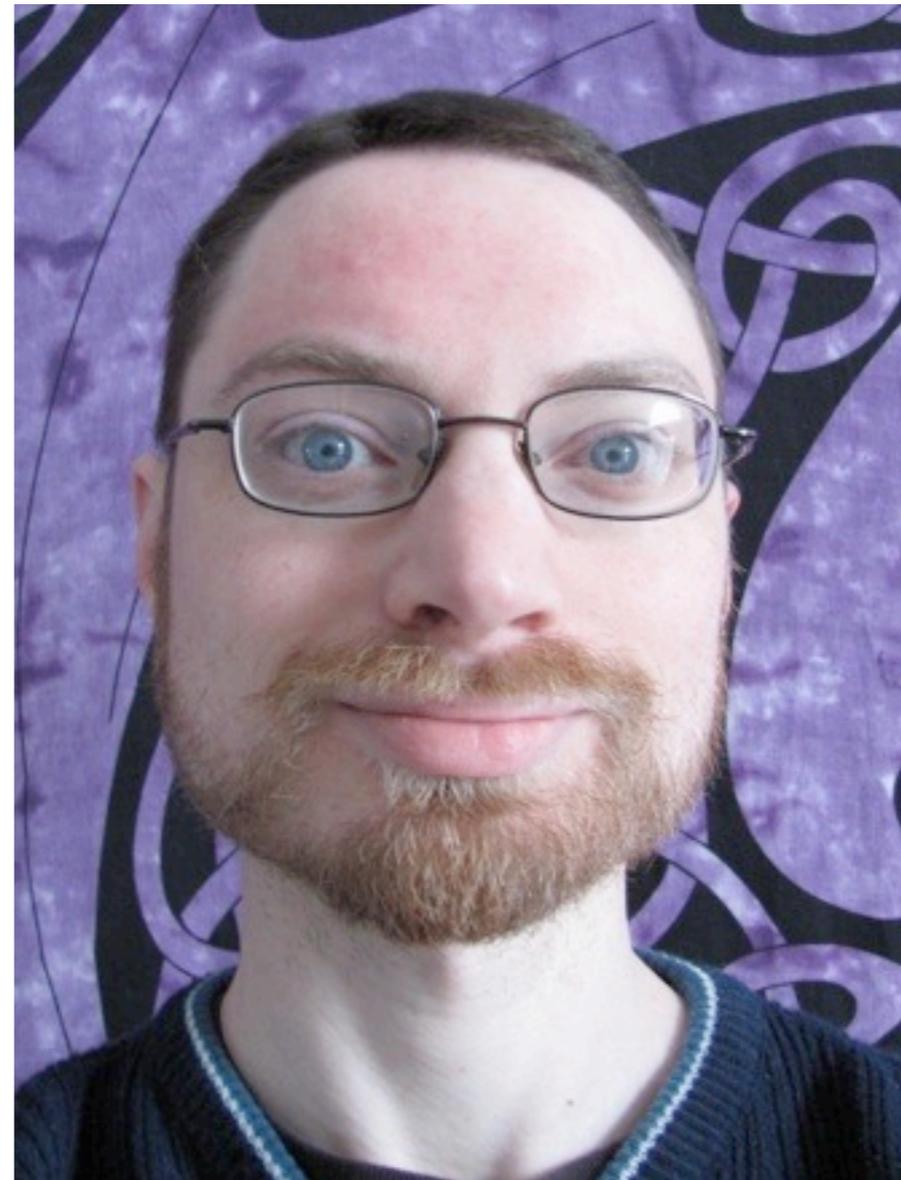


Justin Love

slides@JustinLove.name

<http://JustinLove.name>

[@wondible](#)



<http://www.delicious.com/rauros/amp>