

P2P Bug Tracking with SD

<http://syncwith.us>

```
curl fsck.com/sd|perl;  
export PATH=~/.sd/bin:$PATH;  
sd
```

jesse@bestpractical.com

Jesse Vincent

Hi!

I'm Jesse (obra)

I own a small software company

(Best Practical)

I've been making issue
trackers since 1995

Our software
has some bugs

All software
has some bugs

(All software
is made of bugs)

I spend a lot of time
on airplanes...

...and at conferences
with bad wifi

I need to keep track of
our bugs and our work

I'm the boss

I have no excuse for
not doing my work

I need to keep track of
our bugs and our work
even when I don't have
Internet access

I've tried everything

Text files

Text files in version control

IMAP Servers

RSS Feeds

Running RT on my laptop

Keeping browsers open

Nothing was quite right

So we built SD

SD is a Bug Tracker

SD is a Distributed Bug Tracker

SD Features

Tracks bugs

CLI

Web UI

Scriptable

Works offline

Attachments

Comments

Customizable workflow

Custom properties

P2P Sync

Trac Sync

RT Sync

Hiveminder Sync

GitHub Sync

Google Code Sync

Conflict Resolution

Git integration

Darcs integration

Principles of distributed computing

The network is reliable

Latency is zero

Bandwidth is infinite

The network is secure

Transport cost is zero

The network is
homogeneous

Topology doesn't
change

There is one
administrator

Principles of distributed computing

Fallacies

Principles of distributed computing

Those are all **LIES**

The network is **not**
reliable

*SD isn't network-
dependent*

Latency hurts

SD runs at the edge

Bandwidth is **always**
a problem

*SD knows which
changesets you've already
seen*

The network is
insecure

*SD doesn't depend on a
network security layer*



Topology is
unpredictable
and fluid

SD is topology-agnostic

There is **no**
administrator

(But there are many
people who think
they're administrators)

*SD lets you implement
policy in the centralized
systems it syncs to*

Transport
costs money

*SD doesn't use much
bandwidth & can use
non-network substrates*

The network is
heterogeneous

*SD is designed to sync to
foreign systems*

I didn't make those up

Bill Joy, Tom Lyon and James Gosling did

They missed one

Data is **clean**

*SD doesn't expect to be
able to enforce data
integrity rules*

SD runs locally

SD plays well
with others

It syncs the way you do

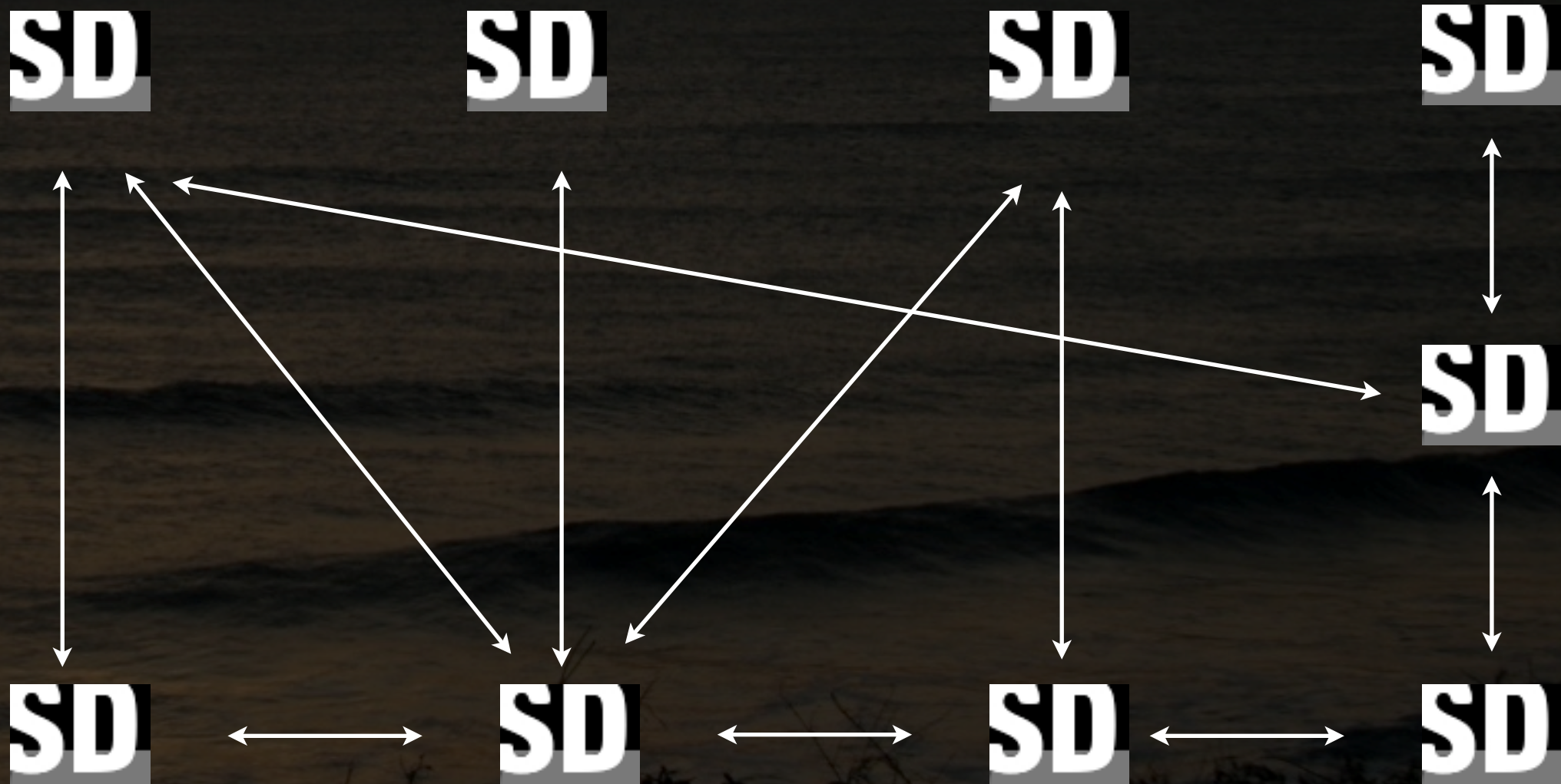
Clone a project's bug database (over HTTP)

Work offline

Pull *changesets* from
anyone you work with

Publish your *database replica* with `rsync`

Topology doesn't matter...



Don't worry

It won't break

SD learns how to
resolve each conflict...

...based on how
everyone else resolves it

Using SD (CLI)

Getting Started

SD Shell

```
$ sd
```



```
Terminal — perl — 80x24

SD for SD (0.01; Prophet 0.01)
Type "help", "about", or "copying" for more information.
SD> █
```


Getting help

```
$ sd help
```



```
Terminal — perl — 80x24
SD for SD (0.01; Prophet 0.01)
Type "help", "about", or "copying" for more information.
SD> help

sd 0.01 - Help Index
-----

help intro      - Getting started with SD
help search     - Searching for and displaying tickets
help tickets    - Working with tickets
help comments   - Working with ticket comments
help attachments - Working with ticket attachments
help sync       - Publishing and importing ticket databases
help history    - Viewing repository history
help environment - Environment variables which affect sd
help config     - Local configuration variables
help summary_format_ticket - Details of this config variable
help aliases    - Command aliases
help settings   - Database configuration variables

Running 'help' on a specific command should also redirect you
to the proper help file.

SD> █
```


Creating a new project

```
$ sd init
```



```
Terminal — less — 80x24

sd 0.01 - Getting started with SD
-----
SD is a peer to peer ticket tracking system built on the Prophet
distributed database. SD is designed to make it easy to work with tickets
and to share ticket databases with your collaborators.

To get started with SD, you need a ticket database. To get an ticket
database, you have two options: You can clone an existing database
or start a new one.

SD will store its local database replica in the ".sd" directory in your
home directory. You can override this behaviour by setting the SD_REPO
environment variable.

To clone a ticket database:

    clone http://example.com/path/to/sd

To start a new ticket database:

    init

:
```


Project settings

\$ sd settings


```
Terminal — vim — 80x24
Knlc3r8_1E
# uuid: 1AF5CF74-A6D4-417E-A738-CCE64A0A7F71
milestones: ["cavil", "diana"]

# uuid: 0AEC922F-57B1-44BE-9588-816E5841BB18
default_component: ["core"]

# uuid: 6CBD84A1-4568-48E7-B90C-F1A5B7BD8ECD
components: ["core", "cli", "webui", "sync", "rt-sync", "hm-sync", "gcode-sync", "trac-
sync", "docs", "tests"]

# uuid: 3f0a074f-af13-406f-bf7b-d69bbf360720
common_ticket_props: ["id", "summary", "status", "milestone", "component", "owner", "c
reated", "due", "creator", "reporter", "original_replica"]

# uuid: 2F9E6509-4468-438A-A733-246B3061003E
default_status: ["new"]

# uuid: C879A68F-8CFE-44B5-9EDD-14E53933669E
active_statuses: ["new", "open"]

# uuid: BAB613BD-9E25-4612-8DE3-21E4572859EA
Knlc3r8_1E [conf] (confComment) #1 1/30,1
"/var/folders/32/322-aCE4FAajlp...+++TI/-Tmp-/Knlc3r8_1E" 30L, 1209C
```


Config file

```
$ sd config
```



```
Terminal — bash — 80x24
17h:~ jesse$ cat .sd-for-sd.sqlite/config
email_address = jesse@bestpractical.com
default_group_ticket_list = milestone
summary_format_ticket = %5.5s,$luid | %8.8s,status | %7.7s,component |%12.12s,owner| %-44.44s,summary

source http://sartak.org/misc/sd = http://sartak.org/misc/sd | 98DAF902-D464-11DD-8353-082EA2FCADC8
source http://fsck.com/~clkao/sd-bugs = http://fsck.com/~clkao/sd-bugs | 8E081976-2985-11DE-AC47-59B9DBC7CFEF

alias cavil = ticket update --id $1 --milestone cavil
alias take = ticket update $1 --owner=jesse@bestpractical.com
alias pull -a = pull --all
alias close = ticket close
alias upload = publish --replica --html --to jesse@fsck.com:public_html/sd-bugs
alias give = ticket update $1 -- owner=$2
alias mine = ticket list -- owner=~^(jesse@bestpractical.com|)$ milestone=cavil
status!~closed|rejected
17h:~ jesse$
```


Create a bug

```
$ sd ticket create
```



```
Terminal — vim — 80x24
8nEKwqk91I
=== required ticket metadata (changes here will not be saved) ===

creator: jesse@bestpractical.com
original_replica: 09E374FA-EBD3-11DD-9A7A-BF638F31E82D

=== edit ticket details below ===

summary:
status: new
milestone: cavil
component: core
owner:
due:
reporter: jesse@bestpractical.com

=== add new ticket comment below ===

~
~
~
~

8nEKwqk91I ( ) #1 1/17,1
"/var/folders/32/322-aCE4FAajlp...E+++TI/-Tmp- /8nEKwqk91I" 17L, 333C
```


Listing bugs

```
$ sd ticket list
```



```
Terminal — perl — 80x24

SD for SD (0.01; Prophet 0.01)
Type "help", "about", or "copying" for more information.
SD> ticket search -- owner=~jesse
```



```
Terminal — less — 80x24

cavil
=====

  51  closed      - jesse@bestpr a failed pull still adds a replica to the so
  68  closed      - jesse@bestpr reported_by and creator are inconsistent. ('
 182  closed      webui jesse@bestpr No ticket update web ui
 192  closed      webui jesse@bestpr after updating a ticket without comments, co
 218  closed      webui jesse@bestpr setting only the 'owner' field when updatin
 220  rejected    webui jesse@bestpr owner field should default to "me"
 221  closed      webui jesse@bestpr create/edit ticket forms need styling
 222  closed      webui jesse@bestpr summary field shouldn't try to autocomplete
 253  closed      webui jesse@bestpr "add a comment" and "initial comments" boxes
 257  closed      core jesse@bestpr due keeps getting set from blank to blank an
 259  closed      core jesse@bestpr "ticket search -- status!=closed status!=res
 284  closed      cli jesse@bestpr the prompt in the cli has the cwd rather tha
 289  closed      core jesse@bestpr Running SD without any arguments and without
 290  closed      core jesse@bestpr SD_REPO/PROPHET_REPO env variable doesn't ex
 291  closed      core jesse@bestpr publish --html doesn't work
 296  closed      webui jesse@bestpr Comments appear to be getting recorded twice
 300  closed      core jesse@bestpr webui updates with no changes record an empt
 302  closed      webui jesse@bestpr Button labels in the web ui don't appear to
 321  closed      webui jesse@bestpr history view is ugly

:
```


Show a bug

\$ sd ticket show


```
Terminal — less — 80x24

= METADATA

id:          349 (76E449C0-298C-11DE-A698-3D3C27CF6674)
summary:     pulling from someone who cloned from you should do less work a
nd be faster
status:      new
milestone:   cavil
component:   core
created:     2009-04-15 07:10:13
creator:     jesse@bestpractical.com
reporter:    jesse@bestpractical.com
original_replica: 09E374FA-EBD3-11DD-9A7A-BF638F31E82D

= HISTORY

jesse@bestpractical.com at 2009-04-15 07:10:13 (1032@09E374FA-EBD3-11DD-9A7A-BF
638F31E82D)
+ "created" set to "2009-04-15 07:10:13"
+ "original_replica" set to "09E374FA-EBD3-11DD-9A7A-BF638F31E82D"
+ "creator" set to "jesse@bestpractical.com"
+ "component" set to "core"
+ "summary" set to "pulling from someone who cloned from you should do less wo
:

```


Update a bug

```
$ sd ticket update
```



```
Terminal — bash — 80x24

SD for SD (0.01; Prophet 0.01)
Type "help", "about", or "copying" for more information.
SD> ticket update 34 -- status=closed owner=jesse@bestpractical.com
Ticket 34 (F2DEB79E-6859-11DD-A3A9-61873178437E) updated.
SD>
```



```
Terminal — perl — 80x24

SD for SD (0.01; Prophet 0.01)
Type "help", "about", or "copying" for more information.
SD> ticket update 349
```



```
Terminal — vim — 80x24
RQJWJ9PpQ
=== required ticket metadata (changes here will not be saved) ===

id: 349 (76E449C0-298C-11DE-A698-3D3C27CF6674)
created: 2009-04-15 07:10:13
creator: jesse@bestpractical.com
original_replica: 09E374FA-EBD3-11DD-9A7A-BF638F31E82D

=== edit ticket details below ===

summary: pulling from someone who cloned from you should do less work and be faster
status: new
milestone: cavil
component: core
owner:
due:
reporter: jesse@bestpractical.com

=== add new ticket comment below ===

~
RQJWJ9PpQ ( ) #1 1/19,1
"/var/folders/32/322-aCE4FAajlp...E+++TI/-Tmp-~/RQJWJ9PpQ" 19L, 483C
```


Log

```
$ sd log
```



```
Terminal — perl — 80x24

SD for SD (0.01; Prophet 0.01)
Type "help", "about", or "copying" for more information.
SD> log
```



```
Terminal — less — 80x24

2009-04-15 08:26:59 - 1034 : clkao@bestpractical.com @ 8E081976-2985-11DE-AC47-5
9B9DBC7CFEF
# Ticket 350 (PROPHET_REPLICA_TYPE=profit should give better error)
+ "creator" set to "clkao@bestpractical.com"
+ "original_replica" set to "8E081976-2985-11DE-AC47-59B9DBC7CFEF"
+ "status" set to "new"
+ "description" set to "unknown prophet replica type gives the following error
:"

Use of uninitialized value in concatenation (.) or string at /Library/Perl/5.8.8
/Prophet/Replica.pm line 81.
Can't use string ("Prophet::Replica") as a HASH ref while "strict refs" in use a
t (eval 138) line 3.
"
+ "reporter" set to "clkao@bestpractical.com"
+ "created" set to "2009-04-15 08:26:59"
+ "component" set to "core"
+ "summary" set to "PROPHET_REPLICA_TYPE=profit should give better error"
+ "milestone" set to "cavil"

2009-04-15 07:30:56 - 1033 : clkao@bestpractical.com @ 8E081976-2985-11DE-AC47-5
9B9DBC7CFEF
:
```


Git Integration

```
$ git sd
```



```
Terminal — perl — 80x24

17h:sd.git jesse$ git sd
SD for SD (0.01; Prophet 0.01)
Type "help", "about", or "copying" for more information.
SD> █
```


Using SD (Web)

Web? Isn't SD an
offline tool?

Local microserver

\$ sd browser

Home

Project overview

Components

- 2 - cli
- 10 - core
- 0 - hm-sync
- 0 - trac-sync
- 0 - sync
- 3 - webui
- 0 - rt-sync
- 0 - tests
- 0 - gcode-sync
- 0 - docs
- 9 - None

Statuses

- 142 - closed
- 0 - stalled

Your active tickets for cavil

id	Status	Milestone	Component	Owner	Reporter	Due	Created
344	new	cavil	webui		jesse@bestpractical.com	3 days ago	3 months ago
351	new	cavil	core		clkao@bestpractical.com		3 months ago
308	new	cavil	core		jesse@bestpractical.com		5 months ago
366	new	cavil	core		jesse@bestpractical.com		6 weeks ago
325	new	cavil	core		jesse@bestpractical.com		5 months ago

Create a ticket

Create a new ticket

summary

status

new

component

core

milestone

cavil

owner

reporter

jesse@bestpractical.com

due

tags

description

Save

Initial comments on this ticket

Search

My open tickets for the current milestone

Show ticket #

Milestones ▶

cavil

Components ▶

diana

All

None

3 days ago

id	Status	Milestone	Component	Owner	Reporter	Created
344	new	cavil	webui		Jesse@bestpractical.com	
Can't unset a ticket due date from the web ui						
351	new	cavil	core		clkao@bestpractical.com	21 hours ago
some database settings records of sd bug are in lower case uuid						
350	new	cavil	core		clkao@bestpractical.com	Yesterday
PROPHET_REPLICA_TYPE=profit should give better error						
348	new	cavil	core		Jesse@bestpractical.com	2 days ago
publish/export should be incremental						
334	new	cavil	core		Jesse@bestpractical.com	7 days ago
sd aliases -add takes one argument, but SD prepares it						
349	new	cavil	core		Jesse@bestpractical.com	2 days ago
pulling from someone who cloned from you should do less work and be faster						
308	new	cavil	core		Jesse@bestpractical.com	3 months ago
sd server readonly mode appears to have vanished						
325	new	cavil	core		Jesse@bestpractical.com	3 months ago
stop users from 'pushing' to a foreign replica they haven't pulled from without --force						

Show a bug

344: Can't unset a ticket due date from the web ui

[Show](#)[Update](#)[History](#)

UUID 96339B24-28A0-11DE-81D0-F841DEF9AFC5

status

new

component

webui

milestone

cavil

reporter jesse@bestpractical.com

created 3 days ago

creator jesse@bestpractical.com

original_replica 09E374FA-EBD3-11DD-9A7A-BF638F31E82D

Update a bug

344: Can't unset a ticket due date from the web ui

Show Update History

summary

status

component

- webui**
- webui
- core
- cli
- sync
- rt-sync
- hm-sync
- gcode-sync

milestone

owner

reporter

due

tags

description

Save

Add a comment

Comments

317: push should have a 'dry run' option

[Show](#) [Update](#) [History](#)

UUID 4B230438-F729-11DD-B769-08D095F21BD5

status	component	milestone
closed	sync	diana

owner jesse@bestpractical.com

reporter jesse@bestpractical.com

created 5 months ago

creator jesse

original_replica 09E374FA-EBD3-11DD-9A7A-BF638F31E82D

Comments

10 days ago jesse@bestpractical.com wrote:

now it does, because I was too scared to push to gcode without it.

History

317: push should have a 'dry run' option

[Show](#) [Update](#) [History](#)

5 months ago jesse

967 @ 09E374FA-EBD3-11DD-9A7A-BF638F31E82D

created set to 2009-02-10 04:14:21**original_replica** set to 09E374FA-EBD3-11DD-9A7A-BF638F31E82D**creator** set to jesse**component** set to sync**summary** set to push should have a 'dry run' option**status** set to new**milestone** set to diana**reporter** set to jesse@bestpractical.com

10 days ago jesse@bestpractical.com

1077 @ 09E374FA-EBD3-11DD-9A7A-BF638F31E82D

owner set to jesse@bestpractical.com**status** changed from new to closed

Working with others

Working with others (Using SD)

Any topology

It doesn't matter who
you sync with

You get all the updates
eventually

Cloning

```
$ sd clone
```


clone makes a replica of
someone else's
database

A terminal window titled "Terminal — bash — 80x24" is shown against a dark background of a sunset over water. The terminal has a black background with white text. At the bottom, the command "sd clone --from http://fsck.com/~jesse/sd-bugs" is being typed, with a cursor at the end of the line. The window has standard macOS window controls (red, yellow, green buttons) in the top-left corner and a scrollbar on the right side.

```
17h:~ jesse$ sd clone --from http://fsck.com/~jesse/sd-bugs
```



```
Terminal — perl — 80x24
17h:~ jesse$ sd clone --from http://17h.local:8080/replica

###..... 14.1% 0:30
```


Pulling

```
$ sd pull
```


pull imports unseen
changes from another
database replica


```
Terminal — bash — 80x24

17h:~ jesse$ sd pull --all
Pulling from http://fsck.com/~clkao/sd-bugs
Done.

Pulling from http://sartak.org/misc/sd
Done.

17h:~ jesse$ █
```


Publishing

\$ sd publish

publish writes out a
copy of your database
replica for sharing

(As SD changesets
and static HTML)


```
Terminal — bash — 80x24

17h:~ jesse$ sd publish --html --replica --to jesse@17h.local:public_html/sd
Exporting a static HTML version of this replia
Exporting a clone of this replia
  Exporting records...
  Exporting changesets...
Exporting resolution database
  Exporting records...
  Exporting changesets...
Publishing the exported clone of the replica to jesse@17h.local:public_html/sd w
ith rsync
The authenticity of host '17h.local (fe80::21f:f3ff:feba:9693%en0)' can't be est
ablished.
RSA key fingerprint is e5:e7:c0:3c:0f:ce:f1:b4:bd:f6:d0:69:c0:ce:9d:99.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '17h.local' (RSA) to the list of known hosts.
Warning: the RSA host key for '17h.local' differs from the key for the IP addres
s 'fe80::21f:f3ff:feba:9693%en0'
Offending key for IP in /Users/jesse/.ssh/known_hosts:101
Are you sure you want to continue connecting (yes/no)? yes
Publication complete.
17h:~ jesse$
```


Hackathon mode (using Bonjour)

Publish your replica

```
$ sd server
```


Pull updates

```
$ sd pull --local
```


Working with others (Using other systems)

But you already have a
bug tracker?

No Problem!

I use at least
two others.

I wrote at least
two others.

We designed SD talk to
foreign bug trackers

RT

(RT::Client::REST)

Hiveminder

(Net::Jifty)

Trac

(Net::Trac)

Google Code

(Net::Google::Code)

GitHub

(Net::GitHub)

Redmine

(Read-only for now)

(Net::Redmine)

Want to help with
Bugzilla, Jira, FogBugz or
something else?

SD

SD



SD

SD

Work with
foreign
replicas in
a Star
Topology



An SD node
can act as a
gateway

Clone

```
$ sd clone
```


Clone from Google Code

```
$ sd clone --from gcode:k9mail
```


Clone from RT

```
$ sd clone --from "rt:https://rt.cpan.org|DBI|"
```


Clone from Trac

```
$ sd clone --from trac:https://trac.parrot.org/parrot
```


Clone from GitHub

```
$ git sd clone --from github:miyagawa/remedie
```


SD reverse engineers a database history

Pull

```
$ sd pull
```


SD reverse engineers a partial database history

Push

```
$ sd push
```


SD figures out local
updates and sends
them upstream

(Then it does a bunch
of book-keeping)

So what's that all look
like in the real world?

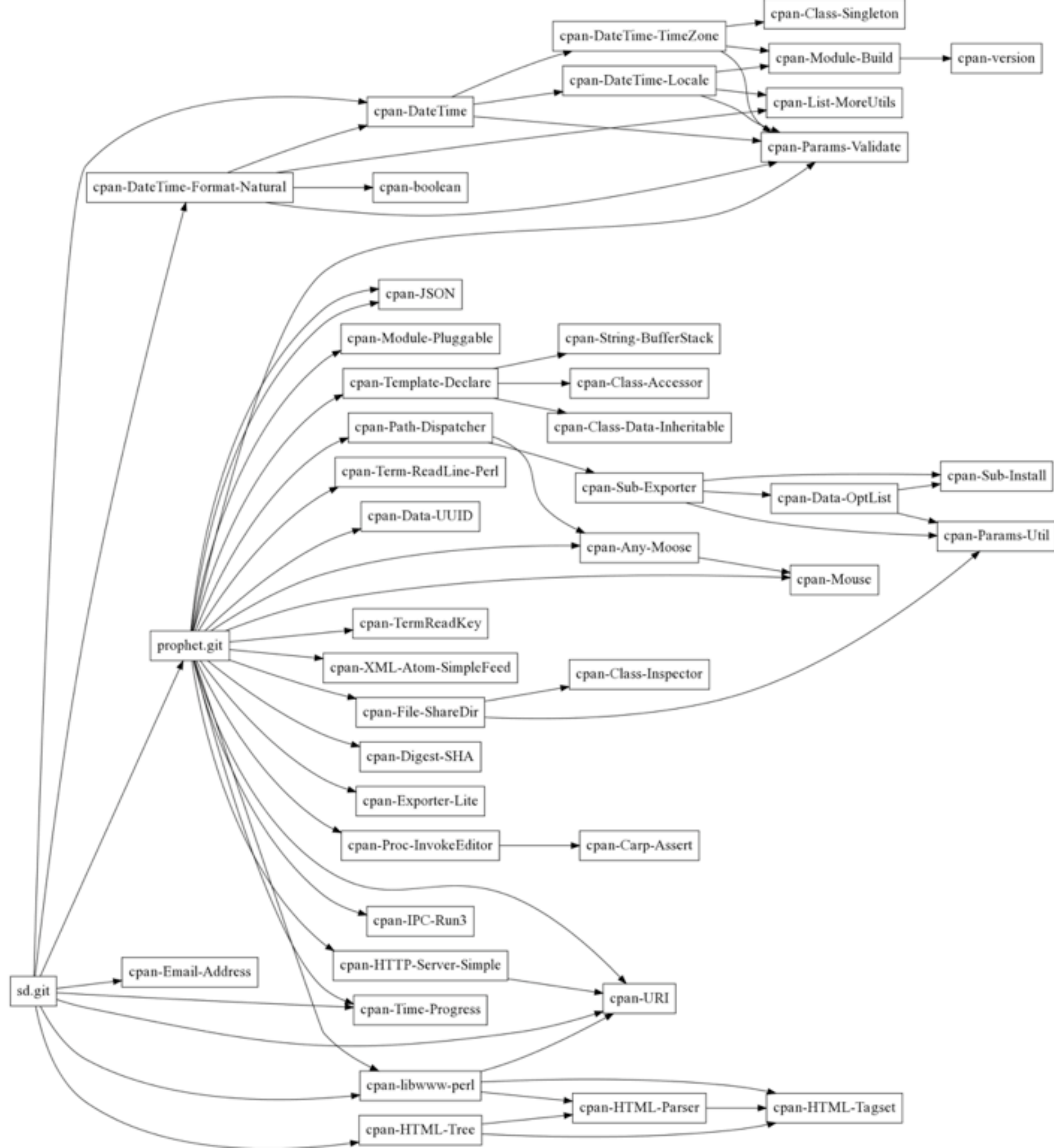
Google Code	Alice's SD	Bob's SD	
Create some bugs			
	clone --from gcode:k9mail		Alice's SD makes a full copy of the k9mail from google code
Someone updates ticket 12			
	ticket update 13		
	ticket create		
	pull --from k9mail		SD integrates upstream changes from Google Code since Alice ran the 'clone' command.
	push --to k9mail		SD figures out what's changed locally since the original 'clone' command. SD then sends any changes that Google Code hasn't seen upstream as ticket updates. Alice's new ticket and her updates to ticket 13 are now part of the upstream bug database in Google Code
	publish --to alice.com:public_html/k9		SD exports a full copy of Alice's bug database and rsyncs it up to alice.com/k9. Anyone browsing alice.com/k9 can clone her SD replica. (Or see it in static HTML)
		clone --from http://alice.com/k9	Bob makes a full copy of Alice's SD replica of the k9mail database. He's now free to make changes to his local database replica.
		ticket update 45	
		ticket create	
	ticket create		
	ticket update 45		
		publish --to bob.com:public_html/k9	Bob publishes his full K9 bug database. Anyone browsing bob.com/k9 can clone Bob's K9 replica. Anyone who already has a copy of Alice or Bob's database can incorporate Bob's new ticket and his changes to ticket 45.
	pull --from http://bob.com/k9		Alice's SD pulls all previously unseen updates from Bob's published database replica and incorporates them. If Bob's edits to ticket 45 didn't conflict, Alice's SD incorporate them. If they did conflict, Alice will be prompted to resolve the conflict.
	push --to k9mail		Alice publishes all updates her bug database has seen since she last sent updates to Google Code. Google Code now has all of Alice and Bob's changes.

Installing SD

It's time to get SD
up and running

SD is in Perl

SD uses 45-90
CPAN modules



Are you afraid?

A 4?

Don't be

I have a novel idea for
a Perl application

One-tweet install.

I'm installing #SD (<http://syncwith.us>)
after seeing @obra's talk at #OSCON!

80 bytes! I still have 60
to work with!

I'm installing #SD (<http://syncwith.us>)
after seeing @obra's talk at #OSCON!

```
curl fsck.com/sd|perl;  
export PATH=~/.sd/bin:$PATH; sd
```


You'll need:

curl, perl 5.8, C compiler

You won't need:

CPAN

You won't need:
to answer prompts

You won't need:
to fix dependencies

SD is in Perl

SD uses CPAN
modules.

This is a blessing.

This is a curse.

CPAN

=

Dependency Hell

When we first built
SD, we used anything
we thought was useful.

(The first version of SD
used a SVN backend)

The first ~useful
version of SD:

123 Dependencies

...a couple hours later

54 Dependencies

Only one *needs* a
compiler.

Shipwright gives us
one-command install


```
curl fsck.com/sd|perl  
export PATH=$PATH:~/sd/bin
```


What's that do?



curl

perl


```
$ head /Library/WebServer/Documents/sd

open (my $tar, '|tar xz 2>/dev/null');
while (<DATA>) {
    print $tar $_;
}
close $tar;
exec("cd sd-build; bin/shipwright-builder \
    --install-base=$ENV{HOME}/sd");
__DATA__
?I?I?sd-build.tar?<is?F??_1?盡yH?"#☞$U??H?eH
...
```


Shipwright installs a few
Perl modules...

...in order

Scalar-List-Utills
String-BufferStack
Class-Accessor
Class-Data-Inheritable
Tree-DAG_Node
Test-Simple
Sub-Uplevel
Test-Exception
Array-Compare
Test-Warn
Template-Declare
URI
HTTP-Server-Simple
Params-Util
Class-Inspector
File-ShareDir
DBI
DBD-SQLite
HTML-Tagset
HTML-Parser
HTML-Tree
Crypt-SSLey
JSON
YAML-Syck
JSON-XS
JSON-DWIW
JSON-Any
Mouse
Any-Moose
Compress-Raw-Zlib
Compress-Raw-Bzip2
IO-Compress
libwww-perl
HTTP-Response-Encoding
WWW-Mechanize

WWW-Mechanize-GZip
File-Slurp
Test-MockModule
Net-GitHub
MIME-Types
Class-Singleton
Params-Validate
version
ExtUtils-CBuilder
ExtUtils-ParseXS
Test-Harness
File-Temp
Module-Build
DateTime-TimeZone
List-MoreUtils
DateTime-Locale
DateTime
File-MMagic
Net-Google-Code
Term-ReadLine-Perl
Digest-SHA1
Digest-HMAC
Net-IP
Net-DNS
Net-Bonjour
TermReadKey
Data-UUID
XML-Atom-SimpleFeed
Digest-SHA
Exporter-Lite
IPC-Run3
MIME-Base64-URLSafe
Data-UUID-Base64URLSafe
Sub-Install
Data-OptList

Sub-Exporter
Path-Dispatcher
Module-Pluggable
Time-Progress
Carp-Assert
Proc-InvokeEditor
Test-HTTP-Server-Simple
Module-Refresh
Carp-Assert-More
Test-LongString
Test-WWW-Mechanize
Test-Script-Run
prophet.git
Devel-StackTrace
Exception-Class
Error
RT-Client-REST
Email-Address
YAML
Path-Class
Clone
Hash-Merge
UNIVERSAL-isa
UNIVERSAL-can
Test-MockObject
Net-Jifty
Lingua-EN-Inflect
Text-CSV
Net-Trac
boolean
Time-Piece
Test-MockTime
DateTime-Format-Natural
sd.git

(104 dists with our
sync plugins)

What's wrong with SD?

\$ sd clone --from <http://fsck.com/~jesse/sd-bugs>

Don't want to
install SD?

Point your browser at
<http://fsck.com/~jesse/sd-bugs/html>

What's next?

Indexing

GPG-signed changesets

Actually releasing 1.0

Thanks!

Questions?

`curl fsck.com/sd | perl`

<http://syncwith.us>

jesse@bestpractical.com