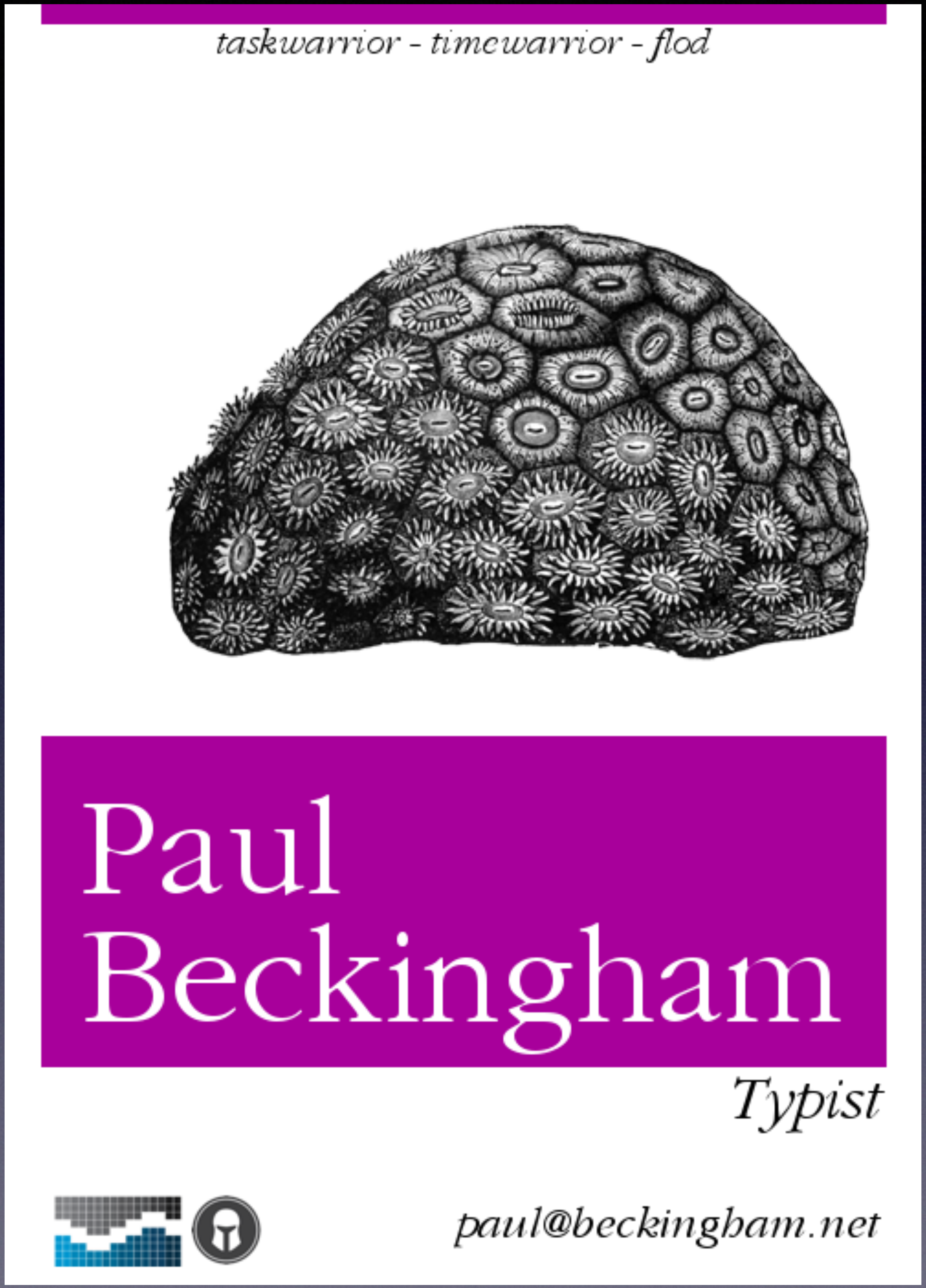
FLOD

agentless CI for the terminal

Poll

taskwarrior
timewarrior

Demo
30 second tutorial

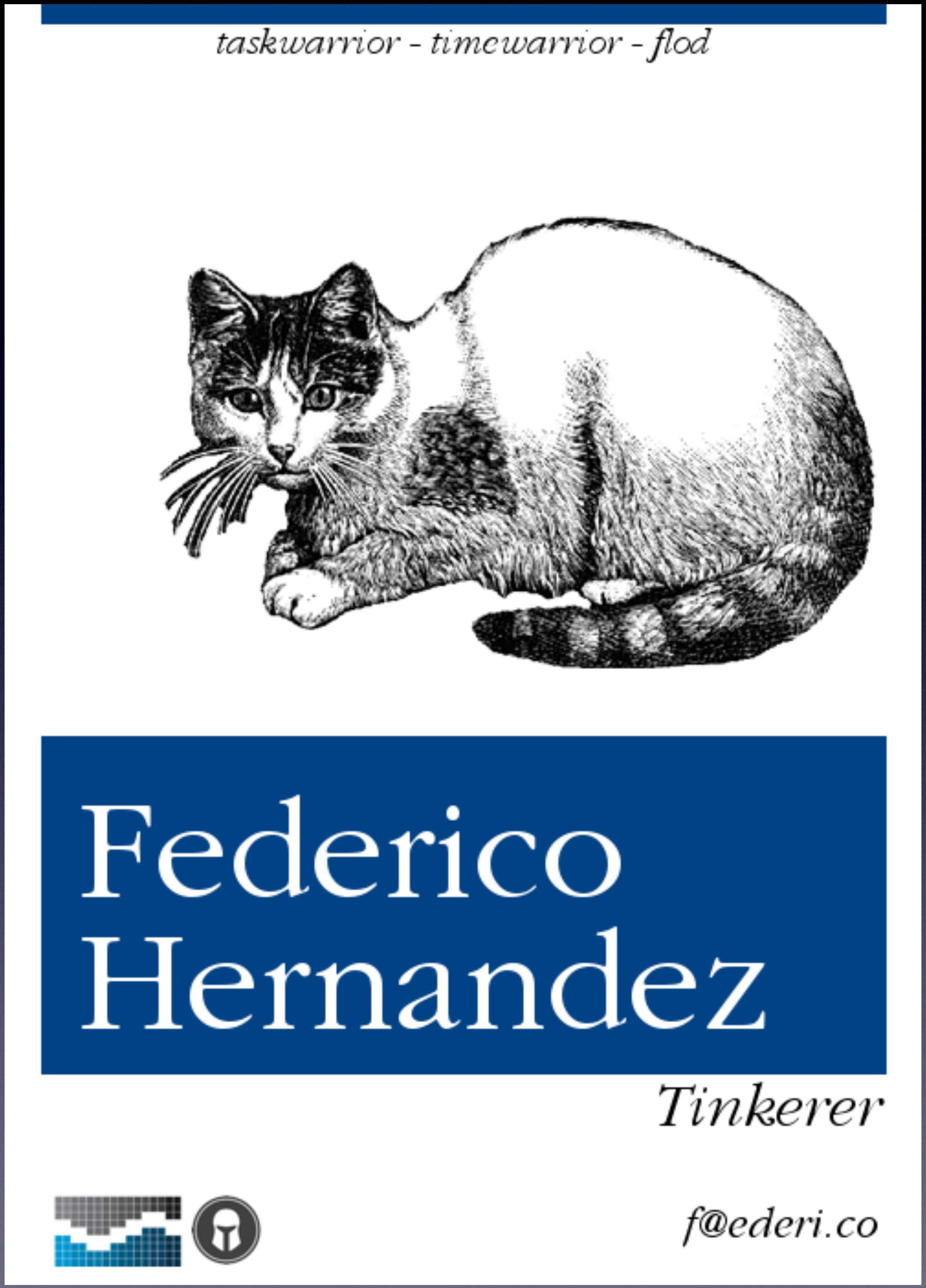


taskwarrior - timewarrior - flod

Paul Beckingham

Typist

paul@beckingham.net



taskwarrior - timewarrior - flod

Federico Hernandez

Tinkerer

f@ederi.co

omine 1492 a Christopho
bo nomine Regis Castel
detecta, et ab Ameri
nomen sortita 1497
SEPTEN
NALIS
NOVA FRANCIA
Oceanus
Ducale
domius
MARE
ATLAN
TICUM
MAR
DEL
NORT
DEL
AMERICA
OCEANUS
AETHIOPI
CUS
R
FICUM
MARE

```
commit 0ae76d3bf29faf2ae7a521f05f91ce2974e555b3
Author: Federico Hernandez <f@ederi.co>
Date:   Thu Jun 18 09:11:59 2009 +0200

    Fixed include statement for Linux
```

```
$ git log | fgrep include | wc -l
     239
```

```
fredde@ack:~$ gcc --version
gcc (Ubuntu 5.4.0-6ubuntu1~16.04.4) 5.4.0 20160609

fredde@sputnik1:~$ gcc --version
gcc (Ubuntu 6.2.0-5ubuntu12) 6.2.0 20161005

[fredde@sputnik2 ~]$ gcc --version
gcc (GCC) 6.3.1 20161221 (Red Hat 6.3.1-1)

[fredde@sputnik3 ~]$ gcc --version
gcc (GCC) 6.2.1 20160830

[fredde@sputnik4 ~]$ gcc --version
gcc (GCC) 4.8.5 20150623 (Red Hat 4.8.5-11)

[fredde@havasu ~]$ gcc --version
Apple LLVM version 8.1.0 (clang-802.0.42)
```

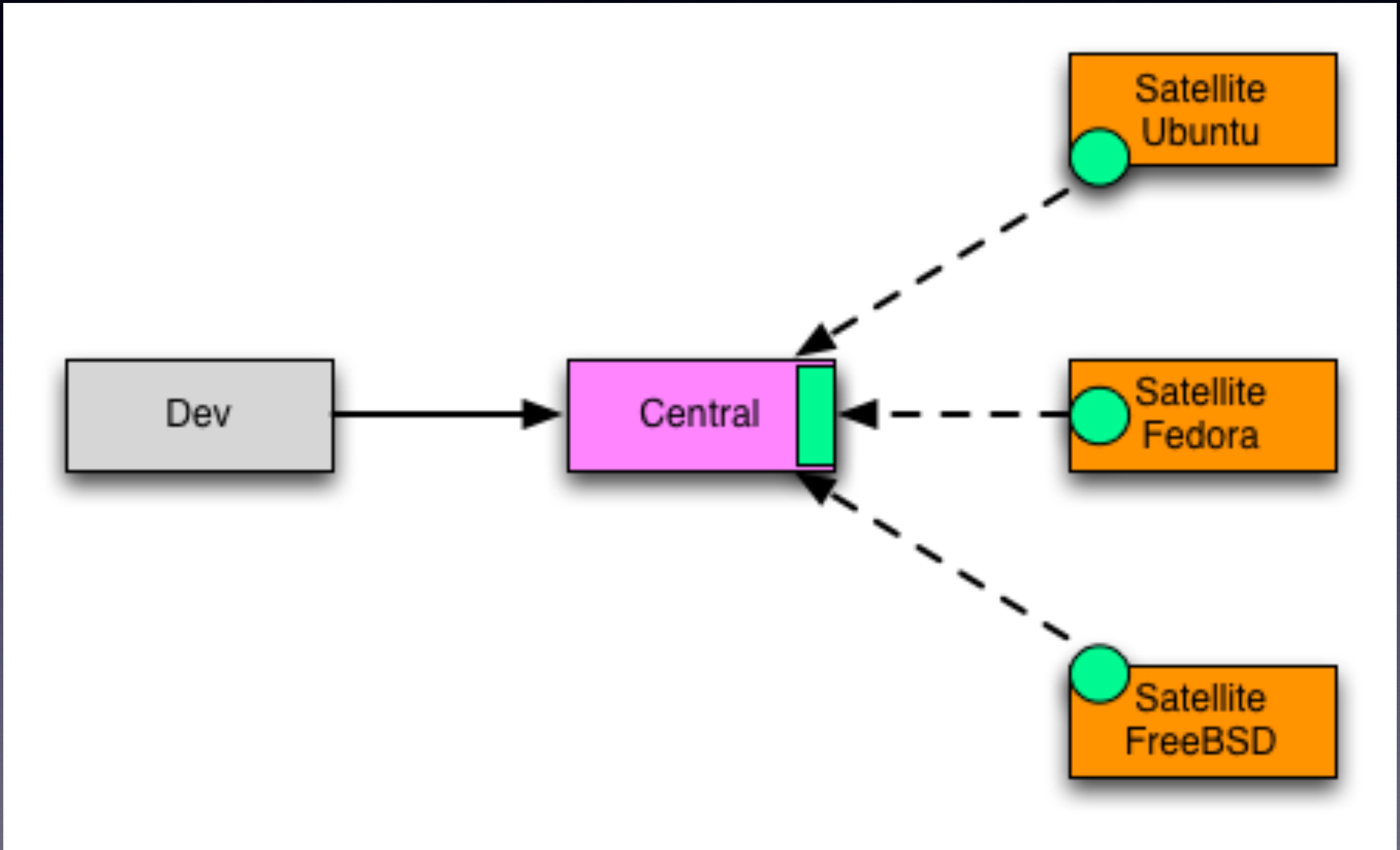
gnutls

3.3, 3.4.10, 3.5.8, 3.5.11

flod

linode

flod



Reuse donated “server capacity” like seti@home

flod

Still no access to shell for debugging.

Agent running, pulling jobs.

Unsecure communication (FW).

Unavailable build machines.

flod2

rackspace DigitalOcean

flod2

Access to *free* servers.

Reverse (push) the communication and use ***ssh*** *(Ansible).*

Model the steps in the CI process as queues.

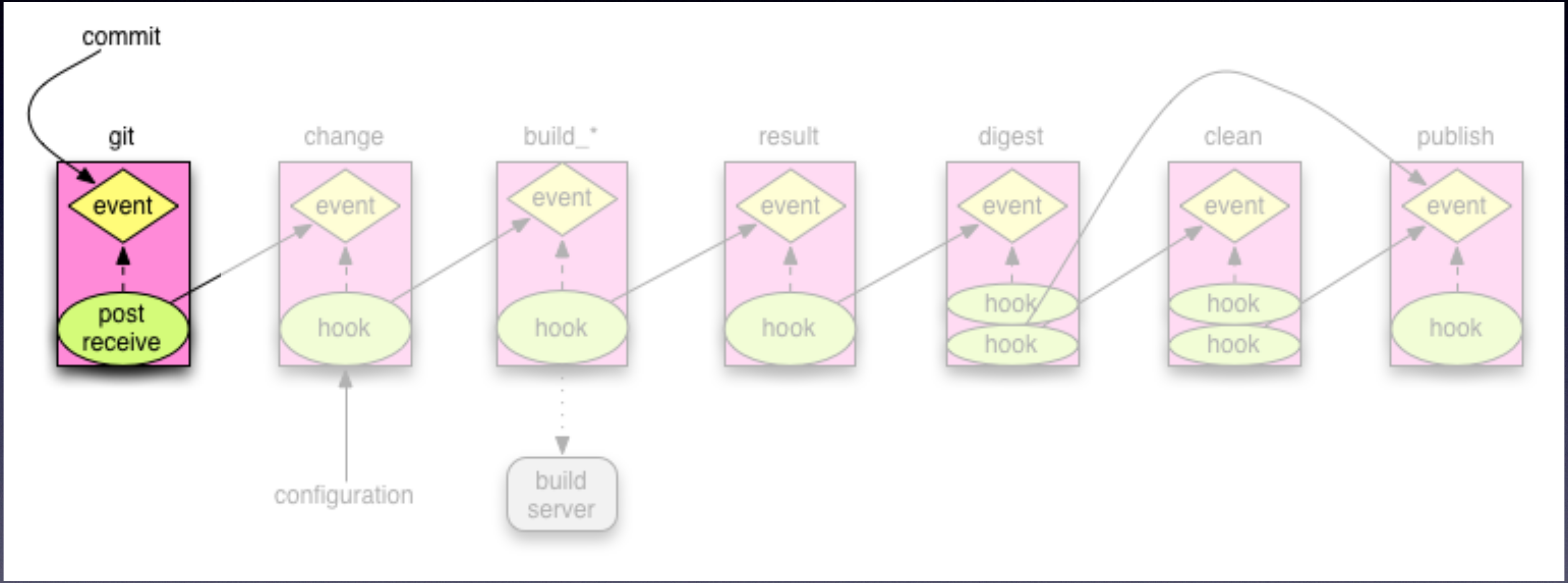
Use the filesystem/directories to store the state.

Attach hooks *(scripts)* that react to events in the directories *(git)*.

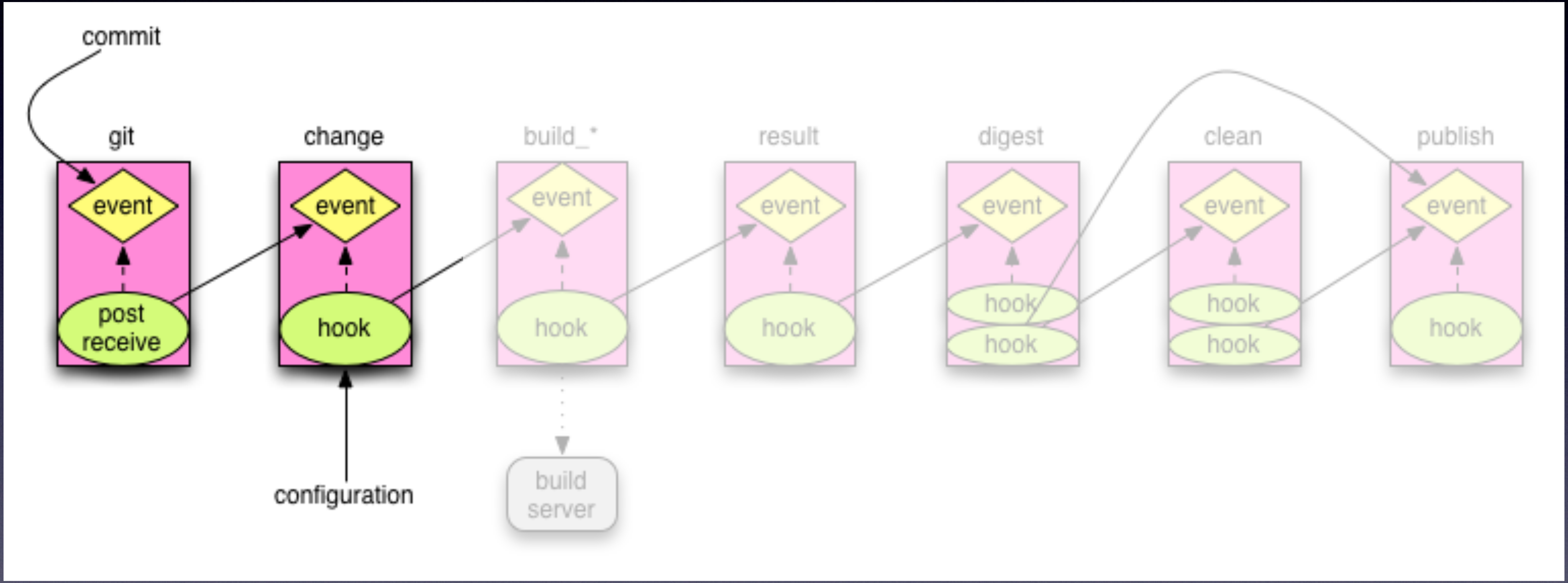
flod2

Text based configuration.

One small (1MB/5.5MB) binary to monitor the queues and trigger the registered hooks.

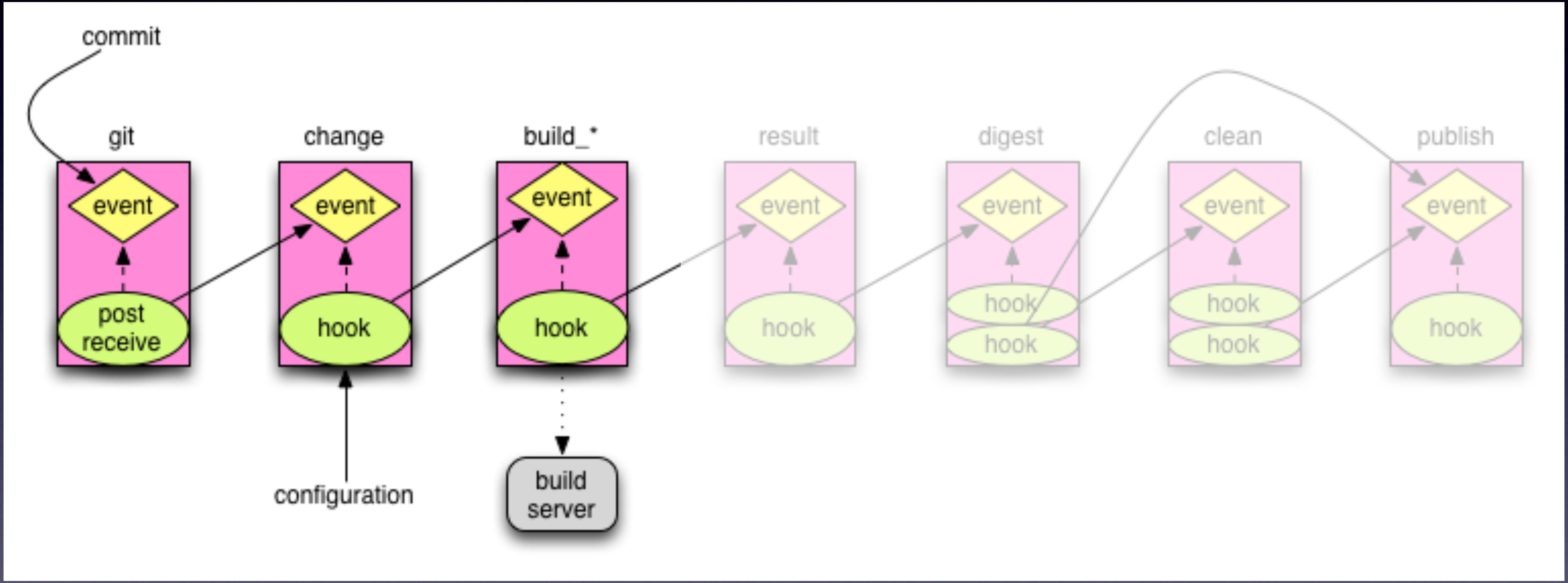

commit
git
event
post receive
change
event
hook
configuration
build_*
event
hook
build server
result
event
hook
digest
event
hook
hook
clean
event
hook
hook
publish
event
hook

```
git rev-list ${FROM}..${TO} | while read COMMIT
do
  AUTHOR=`git log -1 --format="%an" $COMMIT`
  TIMESTAMP=`git log -1 --format="%at" $COMMIT`

  EVENT=`mktemp`
  cat <<EOF >$EVENT
    change.repository: $REPOSITORY
    change.project: $PROJECT
    change.branch: $BRANCH
    change.commit: $COMMIT
    change.author: $AUTHOR
    change.timestamp: $TIMESTAMP
  EOF

  cd $FLD
  /usr/local/bin/central post change $EVENT
  cd $GWD
  rm $EVENT
done
```


commit
git
event
post receive
change
event
hook
configuration
build_*
event
hook
build server
result
event
hook
digest
event
hook
hook
clean
event
hook
hook
publish
event
hook

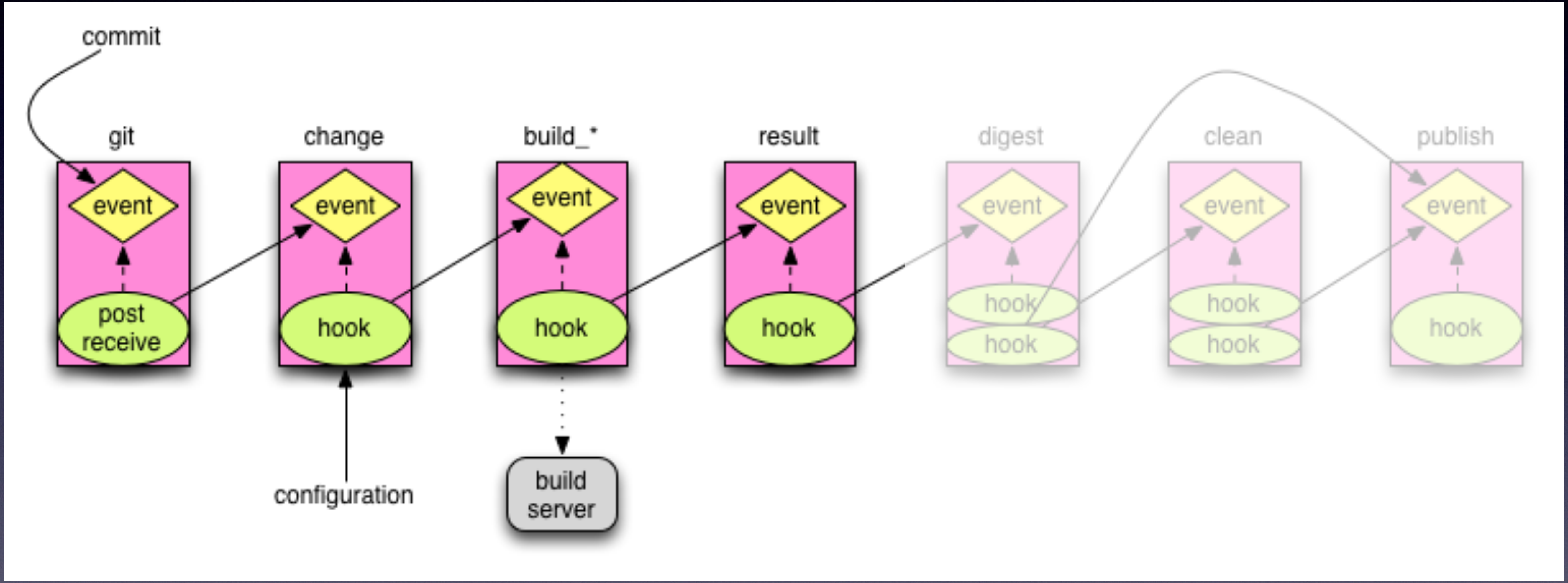
```
project.task.description=Taskwarrior

project.task.ignore=README*,*.md

project.task.platforms=ubuntu1610,fedora25,arch2016,
centos73,macos10_12
project.task.dependencies=cmake,git,c++,
gnutls,uuid,python

project.task.repository=https://git.tasktools.org/
TM/task.git

project.task.job={bash}{nl}{nl}set -x{nl}git clone
--recursive -b {change.branch} {project.repository}
{change.project}.git{nl}cd {change.project}.git{nl}
git reset --hard {change.commit}{nl}cmake .{nl}
make{nl}make test{nl}

project.task.snapshot={bash}{nl}{nl}make package{nl}
```

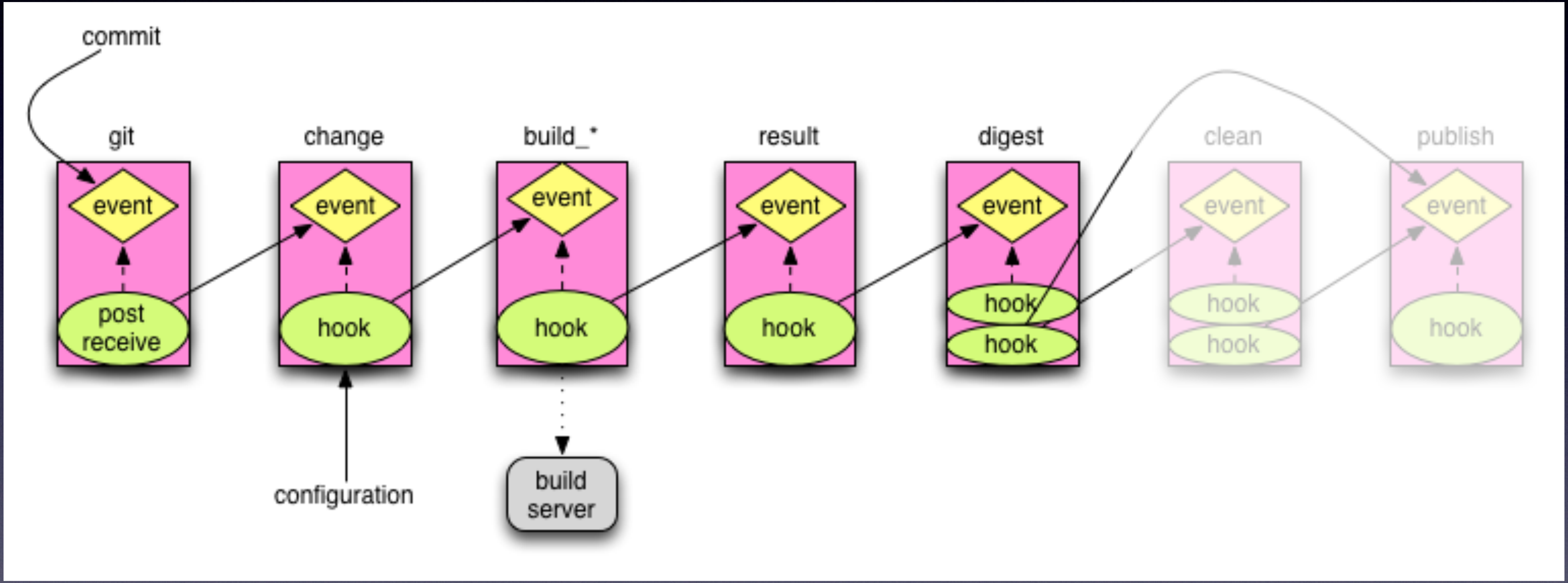
```
platform.ubuntu1610.description=Ubuntu 16.10
platform.ubuntu1610.details=64-bit;GCC 6.2.0;CMake 3.5.2
platform.ubuntu1610.pkg.cmake=cmake
platform.ubuntu1610.pkg.git=git
platform.ubuntu1610.pkg.c++=build-essential
platform.ubuntu1610.pkg.gnutls=gnutls-dev
platform.ubuntu1610.pkg.uuid=uuid-dev
platform.ubuntu1610.pkg.python=python
platform.ubuntu1610.host=sputnik1.tasktools.org
platform.ubuntu1610.scratch=/tmp
```

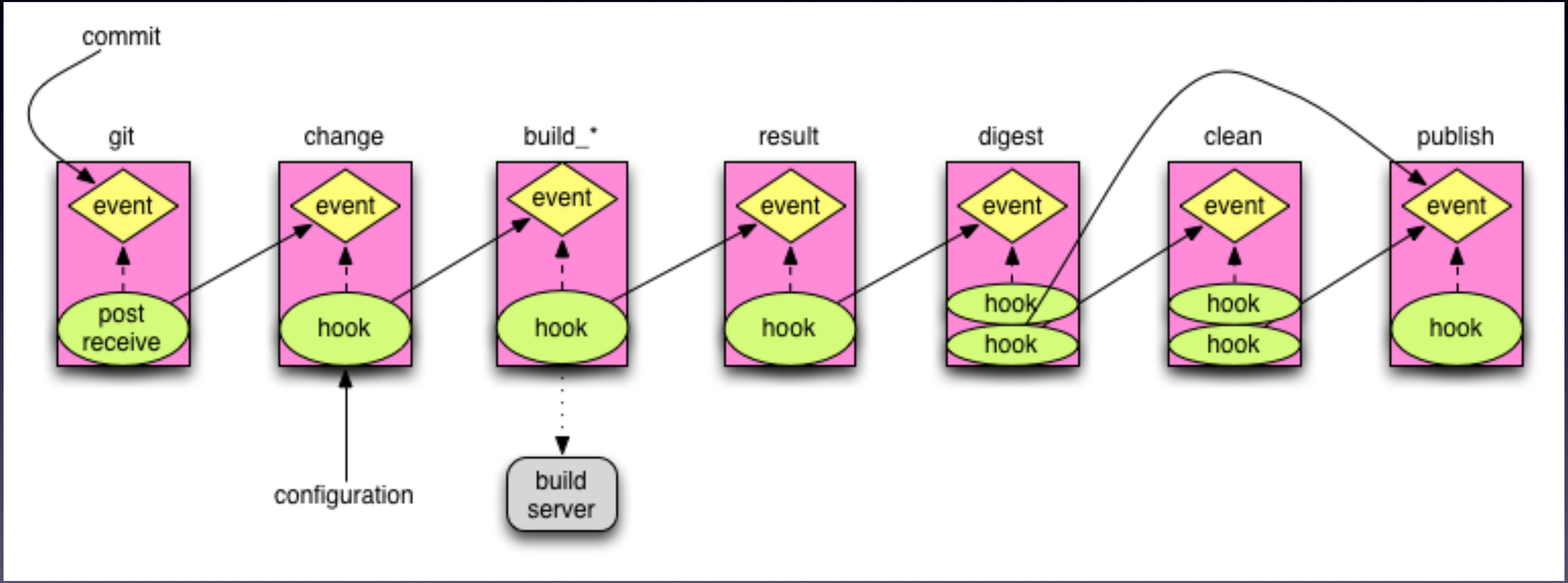

commit
git
event
post receive
change
event
hook
configuration
build_*
event
hook
build server
result
event
hook
digest
event
hook
hook
clean
event
hook
hook
publish
event
hook

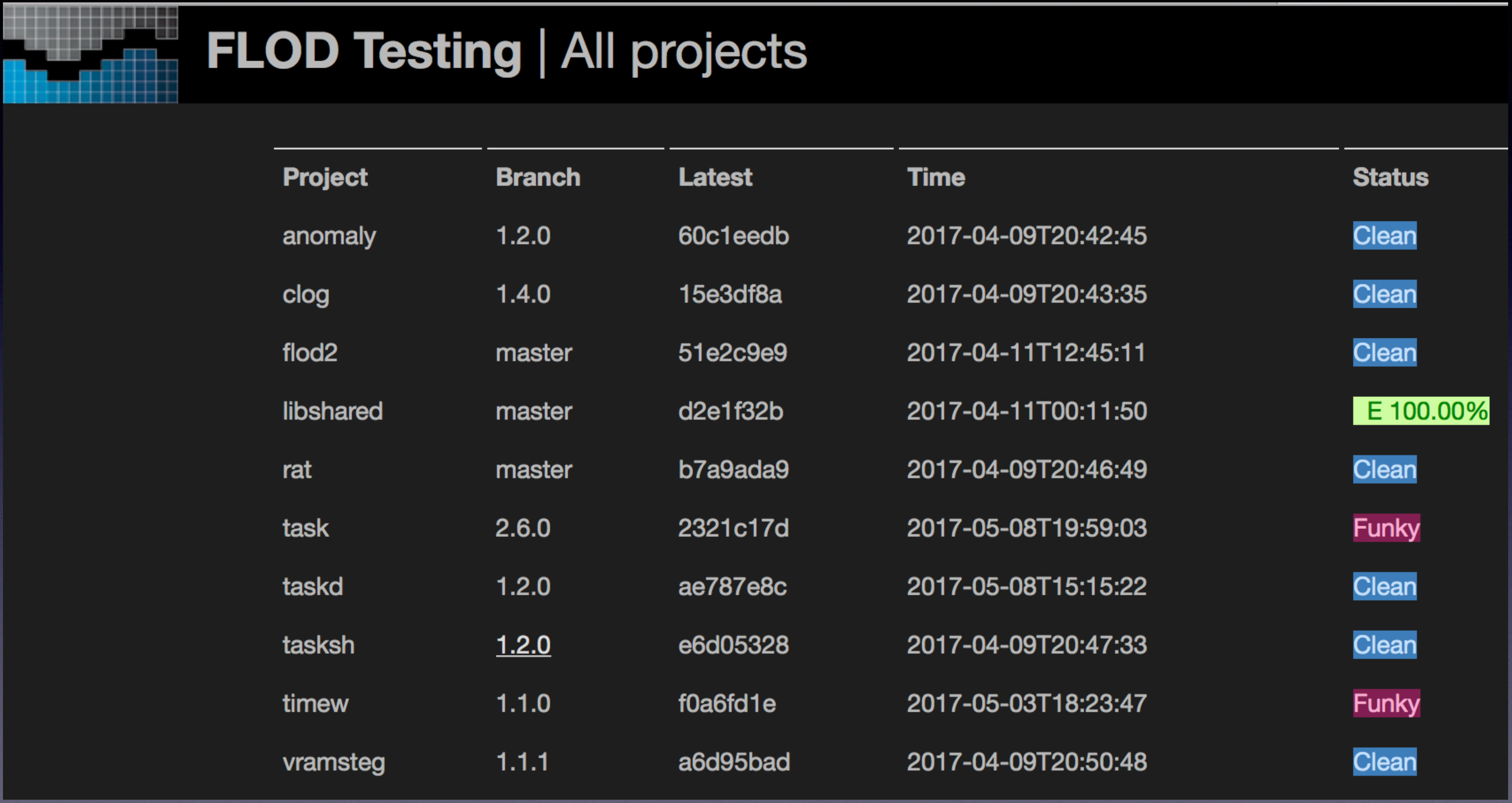
```
def createJobScript(tempdir, contents):
  expanded = contents.replace('{sh}', '#!/bin/sh')
  expanded = expanded.replace('{bash}', '#!/bin/
bash')
  expanded = expanded.replace('{nl}', '\n')

  script = '{}/job'.format(tempdir)
  with open(script, 'w') as fh:
    fh.write(expanded)
  os.chmod(script, 0o700)
  return script
```

```
host = event.header('platform.host')
scratch = event.header('platform.scratch')
script = createJobScript(tempdir, event.header('project.job'))
id = ‘{0}.{1}'.format(event.header('change.project'),
  event.header('change.commit'))

log, err = subprocess.Popen(['ssh', host, 'mkdir {0}/
  {1}'.format(scratch, id)],
             stdout=subprocess.PIPE,
  stderr=subprocess.STDOUT).communicate()

out, err = subprocess.Popen(['scp', script, ‘{0}:{1}/{2}/
  {2}.job'.format(host, scratch, id)],
             stdout=subprocess.PIPE,
  stderr=subprocess.STDOUT).communicate()
log += out

out, err = subprocess.Popen(['ssh', host, 'cd {0}/{1} && ./
  {1}.job'.format(scratch, id)],
                  stdout=subprocess.PIPE,
  stderr=subprocess.STDOUT).communicate()
log += out
```


commit
git
change
build_*
result
digest
clean
publish
event
event
event
event
event
event
event
post receive
hook
hook
hook
hook
hook
hook
hook
hook
configuration
build server

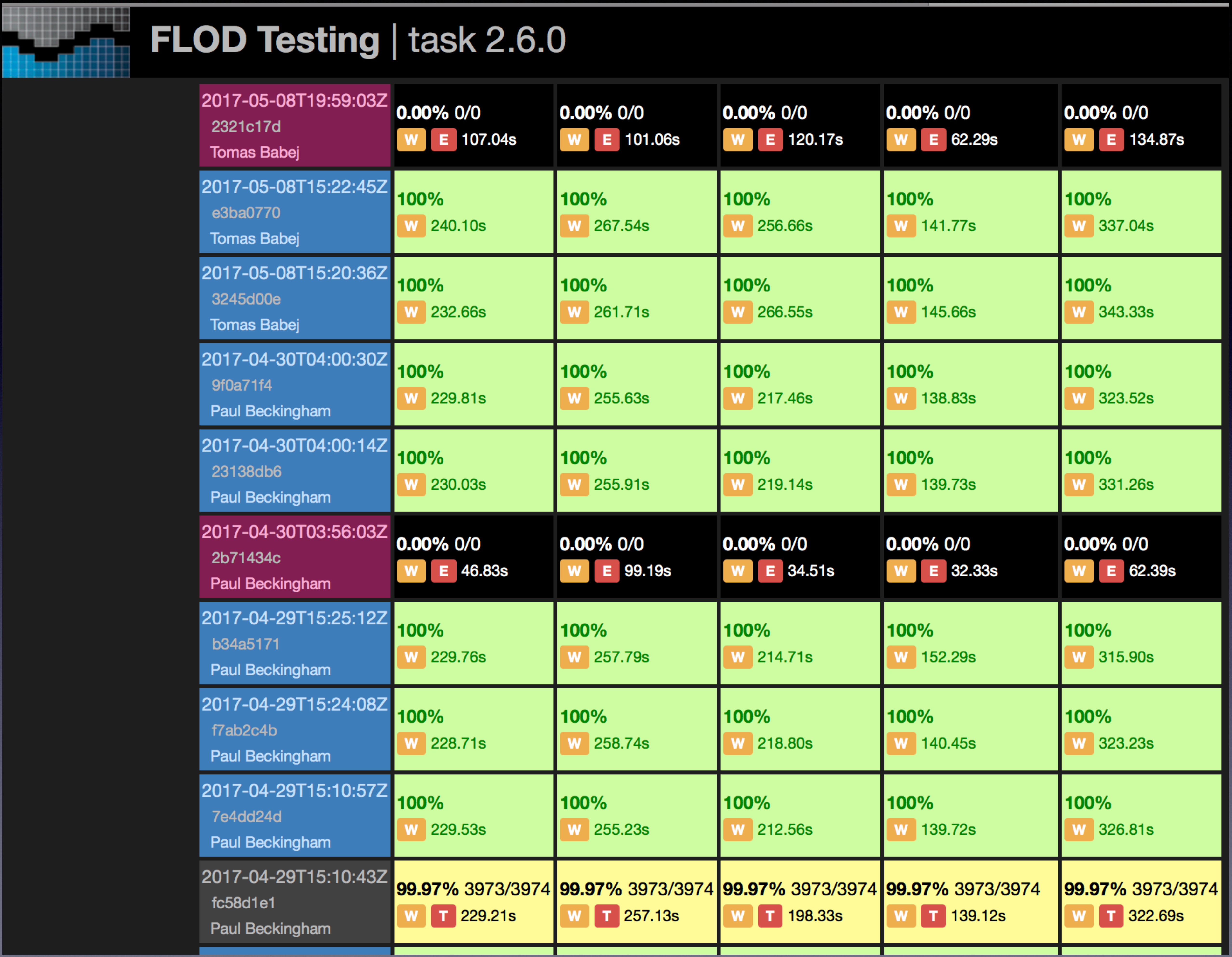

commit
git
event
post receive
change
event
hook
configuration
build_*
event
hook
build server
result
event
hook
digest
event
hook
hook
clean
event
hook
hook
publish
event
hook


commit
git
change
build_*
result
digest
clean
publish
event
event
event
event
event
event
event
post receive
hook
hook
hook
hook
hook
hook
hook
hook
configuration
build server



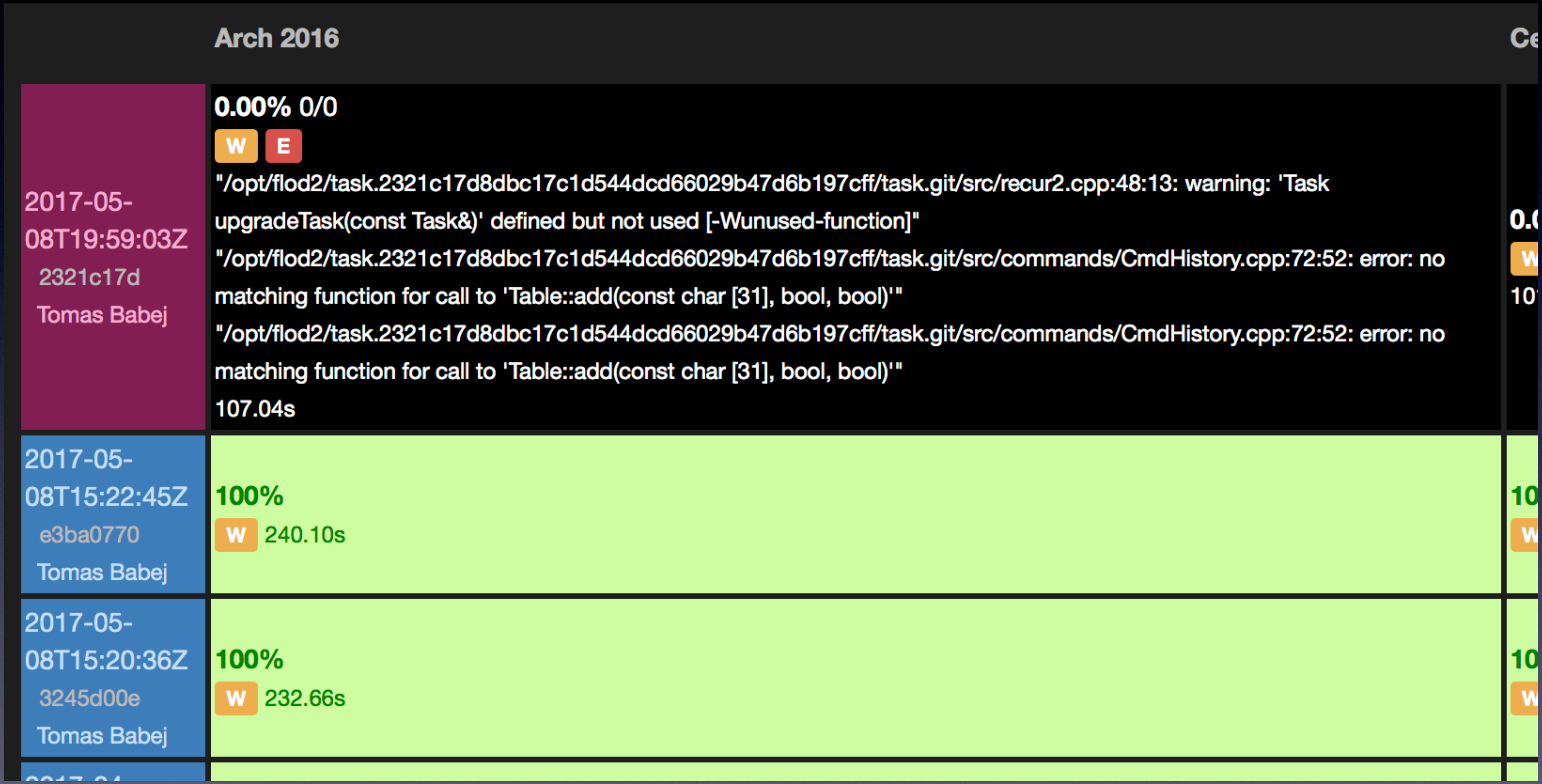
FLOD Testing | All projects

Project	Branch	Latest	Time	Status
anomaly	1.2.0	60c1eedb	2017-04-09T20:42:45	Clean
clog	1.4.0	15e3df8a	2017-04-09T20:43:35	Clean
flod2	master	51e2c9e9	2017-04-11T12:45:11	Clean
libshared	master	d2e1f32b	2017-04-11T00:11:50	E 100.00%
rat	master	b7a9ada9	2017-04-09T20:46:49	Clean
task	2.6.0	2321c17d	2017-05-08T19:59:03	Funky
taskd	1.2.0	ae787e8c	2017-05-08T15:15:22	Clean
tasksh	1.2.0	e6d05328	2017-04-09T20:47:33	Clean
timew	1.1.0	f0a6fd1e	2017-05-03T18:23:47	Funky
vramsteg	1.1.1	a6d95bad	2017-04-09T20:50:48	Clean



FLOD Testing | task 2.6.0

2017-05-08T19:59:03Z 2321c17d Tomas Babej	0.00% 0/0 W E 107.04s	0.00% 0/0 W E 101.06s	0.00% 0/0 W E 120.17s	0.00% 0/0 W E 62.29s	0.00% 0/0 W E 134.87s
2017-05-08T15:22:45Z e3ba0770 Tomas Babej	100% W 240.10s	100% W 267.54s	100% W 256.66s	100% W 141.77s	100% W 337.04s
2017-05-08T15:20:36Z 3245d00e Tomas Babej	100% W 232.66s	100% W 261.71s	100% W 266.55s	100% W 145.66s	100% W 343.33s
2017-04-30T04:00:30Z 9f0a71f4 Paul Beckingham	100% W 229.81s	100% W 255.63s	100% W 217.46s	100% W 138.83s	100% W 323.52s
2017-04-30T04:00:14Z 23138db6 Paul Beckingham	100% W 230.03s	100% W 255.91s	100% W 219.14s	100% W 139.73s	100% W 331.26s
2017-04-30T03:56:03Z 2b71434c Paul Beckingham	0.00% 0/0 W E 46.83s	0.00% 0/0 W E 99.19s	0.00% 0/0 W E 34.51s	0.00% 0/0 W E 32.33s	0.00% 0/0 W E 62.39s
2017-04-29T15:25:12Z b34a5171 Paul Beckingham	100% W 229.76s	100% W 257.79s	100% W 214.71s	100% W 152.29s	100% W 315.90s
2017-04-29T15:24:08Z f7ab2c4b Paul Beckingham	100% W 228.71s	100% W 258.74s	100% W 218.80s	100% W 140.45s	100% W 323.23s
2017-04-29T15:10:57Z 7e4dd24d Paul Beckingham	100% W 229.53s	100% W 255.23s	100% W 212.56s	100% W 139.72s	100% W 326.81s
2017-04-29T15:10:43Z fc58d1e1 Paul Beckingham	99.97% 3973/3974 W T 229.21s	99.97% 3973/3974 W T 257.13s	99.97% 3973/3974 W T 198.33s	99.97% 3973/3974 W T 139.12s	99.97% 3973/3974 W T 322.69s



	Arch 2016	Ce
2017-05-08T19:59:03Z 2321c17d Tomas Babej	**0.00%** 0/0 W E "/opt/flod2/task.2321c17d8dbc17c1d544dcd66029b47d6b197cff/task.git/src/recur2.cpp:48:13: warning: 'Task upgradeTask(const Task&)' defined but not used [-Wunused-function]" "/opt/flod2/task.2321c17d8dbc17c1d544dcd66029b47d6b197cff/task.git/src/commands/CmdHistory.cpp:72:52: error: no matching function for call to 'Table::add(const char [31], bool, bool)'" "/opt/flod2/task.2321c17d8dbc17c1d544dcd66029b47d6b197cff/task.git/src/commands/CmdHistory.cpp:72:52: error: no matching function for call to 'Table::add(const char [31], bool, bool)'" 107.04s	**0.0** W 10
2017-05-08T15:22:45Z e3ba0770 Tomas Babej	**100%** W 240.10s	**10** W
2017-05-08T15:20:36Z 3245d00e Tomas Babej	**100%** W 232.66s	**10** W

Demo

Likely Enhancements

- Bugfixes and improvements.
- Auto cleanup of main result page.
- auto tagging, auto run performance scripts, auto snapshot tar balls for stable commits.
- smarter configuration, e.g. * for platform requests.
- discovery (self registration) for build satellites.
- build and test-run metrics.
- better notification of results.
- start VMs for build on demand, docker (cost vs performance).

What we have learned

- A lot can be learned from when a build only fails on macOS, FreeBSD and Solaris. [BSD Potability]
- If a build/test cycle is too long, then by the time the CI results are in, you've moved on to the next thing.
- No matter how much detail you can cram into the platform-specific build/test log, you still need ssh access to properly investigate.
- A CI report is a great way to answer a whole class of "it won't build on xxxx" complaints.
- Having build/test machines in different time zones is a great way to identify brittle tests that are dependent on time of day.
- Having the report identify "funky" builds is a great way to highlight problems on specific platforms.
- More than just running tests, knowing that a project builds "clean" (no warnings) is an important extra piece of data.
- CI is so useful, we won't release without it now. It almost eliminates the need for x.y.1 releases.

Graph of dependencies

Graph/event/hook system is capable of implementing basically anything.

Join us at
FOSDEM 2018
in Brussels

https://git.tasktools.org/TS/flod2
https://taskwarrior.org
support@taskwarrior.org
@taskwarrior
IRC: #taskwarrior

Disclaimer

Pictures and logos from corresponding searches at images.google.com