Continuous Integration in Data Centers Further 3 Years Later

Michael Prokop

OSDC.de

OPEN SOURCE DATA CENTER CONFERENCE

APRIL 26TH - 28TH, 2016 | BERLIN

% whoami

Mika || @mikagrml

Independent Consultant

Project leader of Grml.org

Debian Developer

Jenkins-debian-glue.org

Source: https://flic.kr/p/pzMqux

Recap from OSDC 2013

Why Continuous Delivery?

Jenkins

Vagrant/Veewee

Puppet

Jenkins-debian-glue.org

Admin Docs (Sphinx)

Custom Grml ISOs (grml-live/grml2iso)

Roadmap

What do we want?

Challenges

How did we get there?

Code Review

Testing Tools

Docker

Infrastructure Tooling

Antipatterns

Pain Points

What do we want?

Expectations

Independence

Reliability

Reproducibility

Traceability

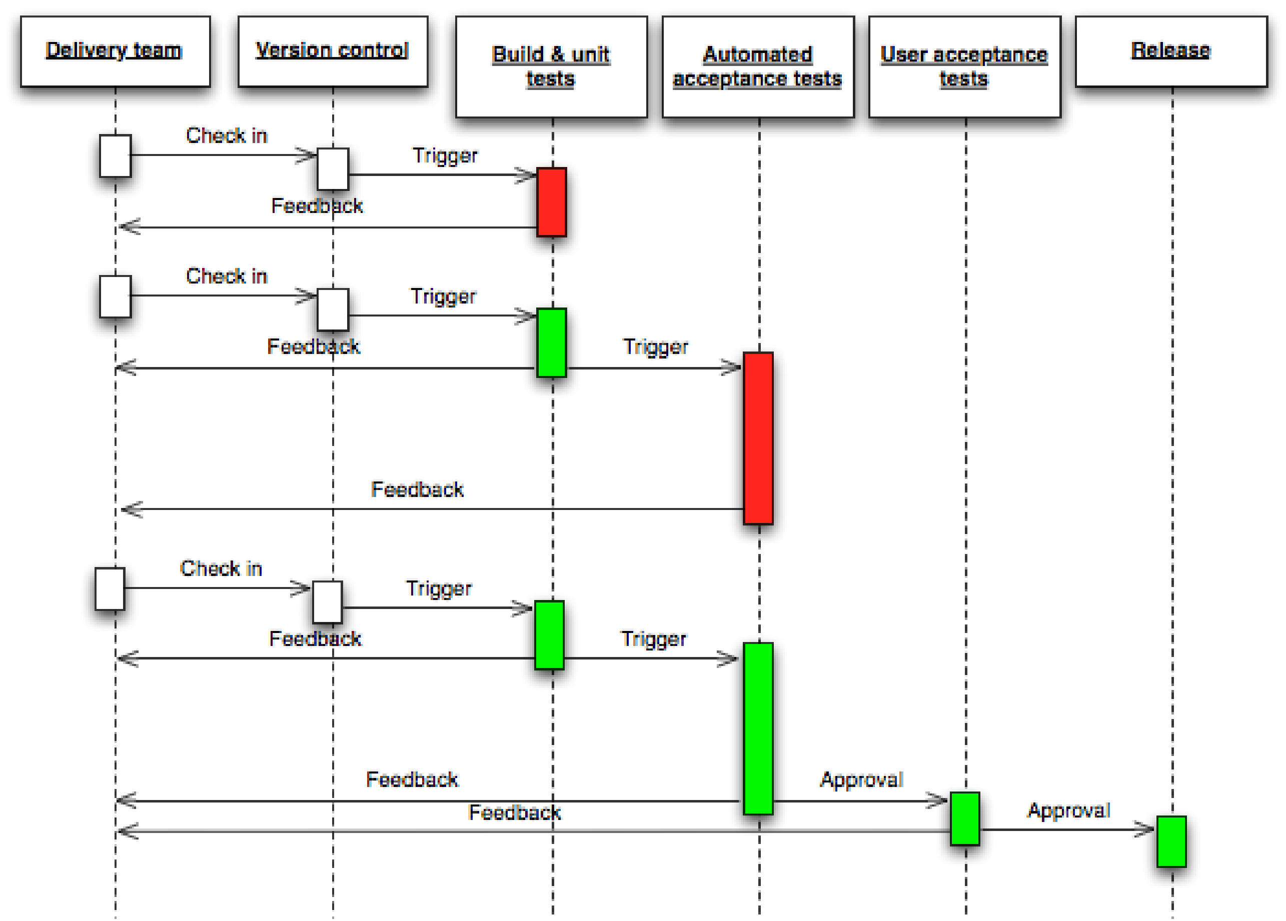
Scalability

Performance

Predictability

Innovation

Deployment Pipeline



Source: http://continuousdelivery.com/2010/02/continuous-delivery/

Challenges

Changes since 2013?

More employees

More remote workers

More customers

More projects

Source: https://twitter.com/gabrealness/status/723434940734431237

How did we get there?

Invest

“*If I had eight hours to chop down a tree, I'd spend six hours sharpening my ax.*” – Abraham Lincoln

“*We learn geology the morning after the earthquake.*” – Ralph Waldo Emerson

Improve

Better documentation

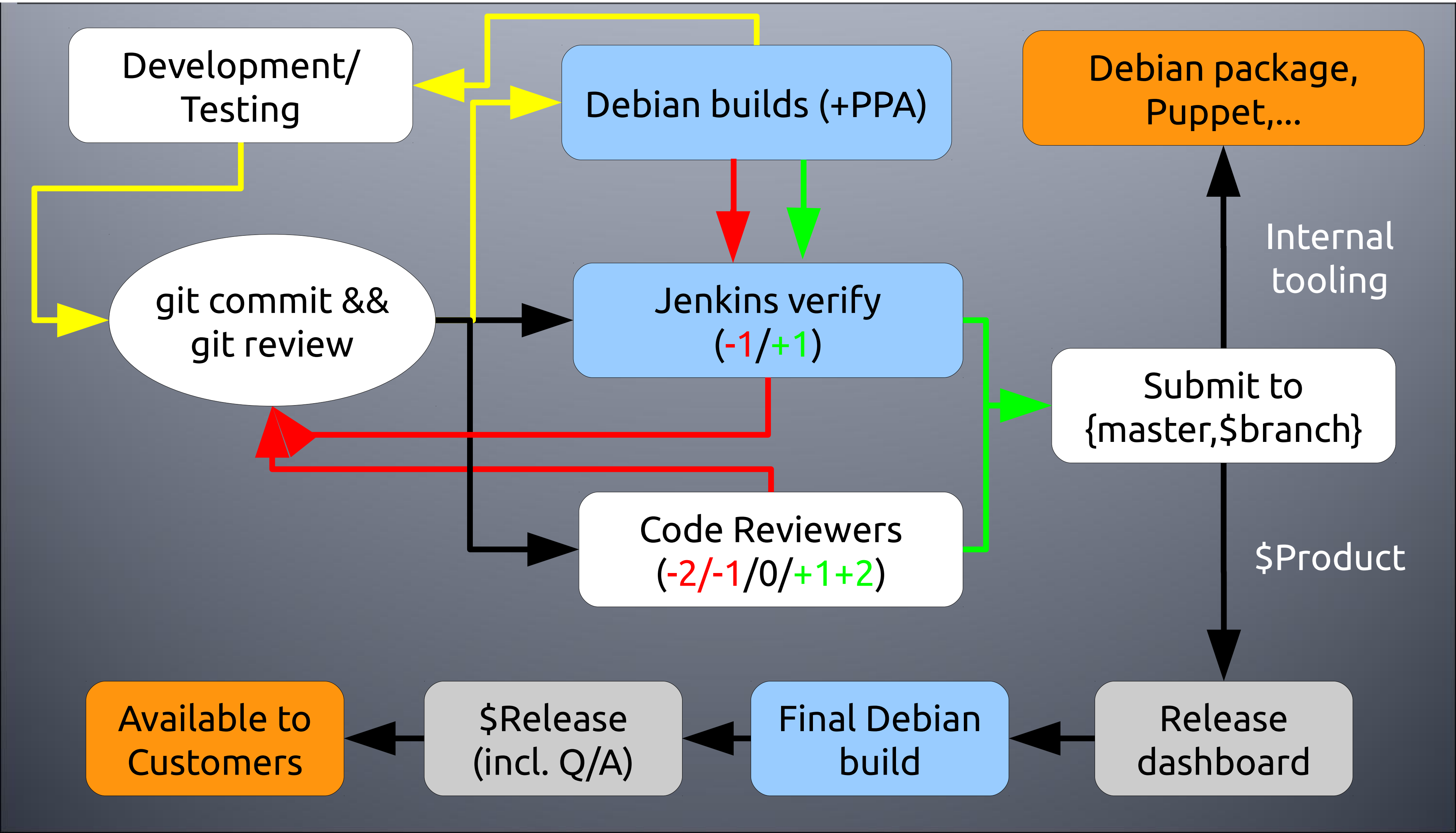
Better communication

Leverage new tools!

More Tests/ More Testing

Faster Release Cycle

Workflow + Pipeline



Code Review

Gerrit
GitLab
Phabricator
Review Board

Source: https://memegenerator.net/instance/53083760

Advantages of CR

Share
knowledge

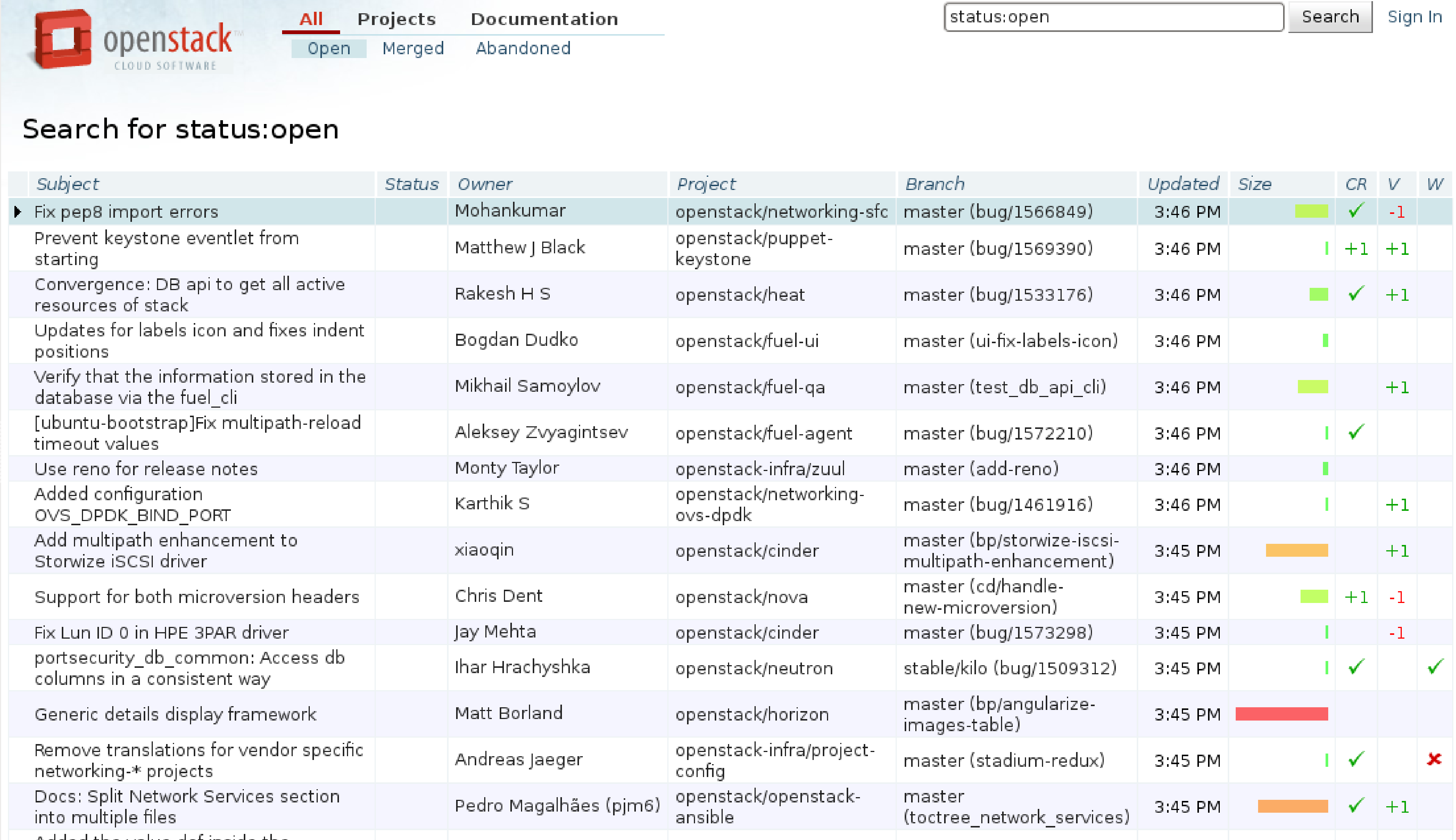
Improve
maintainability

Better
code

Broadcast
progress

Communal
ownership

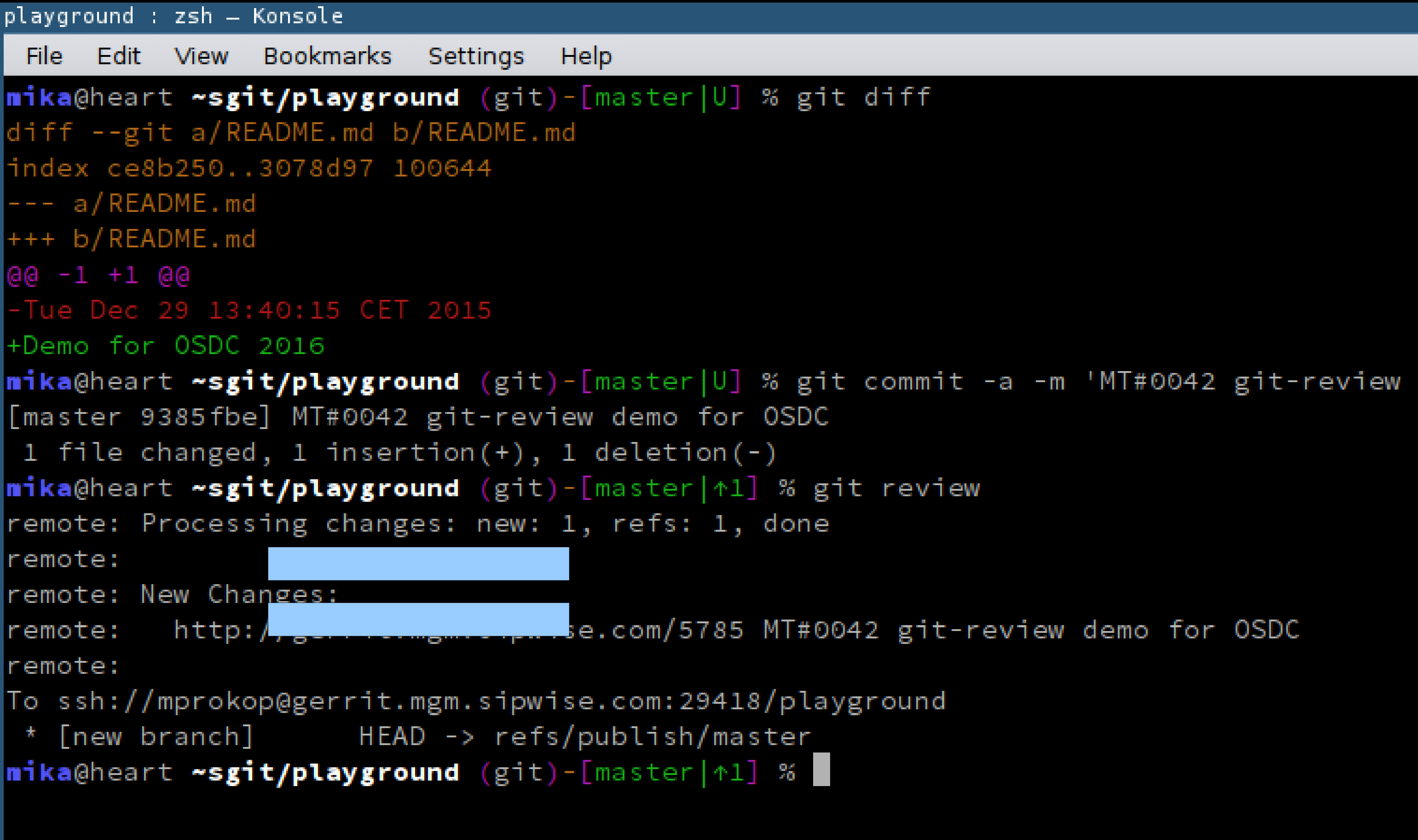
Gerrit



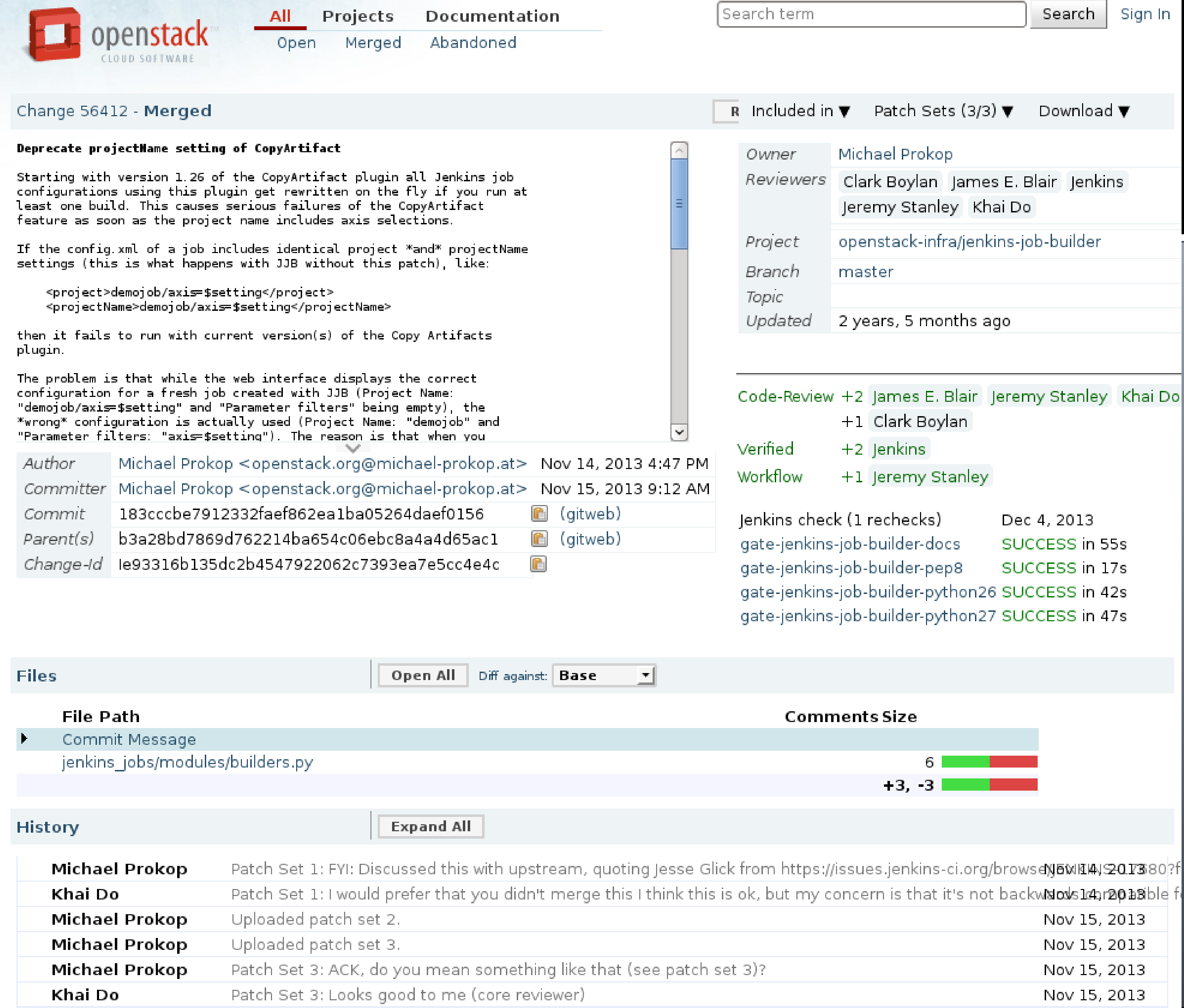
Search for status:open

	Subject	Status	Owner	Project	Branch	Updated	Size	CR	V	W
▶	Fix pep8 import errors		Mohankumar	openstack/networking-sfc	master (bug/1566849)	3:46 PM	🟩	✓	-1	
	Prevent keystone eventlet from starting		Matthew J Black	openstack/puppet-keystone	master (bug/1569390)	3:46 PM	🟩	+1	+1	
	Convergence: DB api to get all active resources of stack		Rakesh H S	openstack/heat	master (bug/1533176)	3:46 PM	🟩	✓	+1	
	Updates for labels icon and fixes indent positions		Bogdan Dudko	openstack/fuel-ui	master (ui-fix-labels-icon)	3:46 PM	🟩			
	Verify that the information stored in the database via the fuel_cli		Mikhail Samoylov	openstack/fuel-qa	master (test_db_api_cli)	3:46 PM	🟩		+1	
	[ubuntu-bootstrap]Fix multipath-reload timeout values		Aleksey Zvyagintsev	openstack/fuel-agent	master (bug/1572210)	3:46 PM	🟩	✓		
	Use reno for release notes		Monty Taylor	openstack-infra/zuul	master (add-reno)	3:46 PM	🟩			
	Added configuration OVS_DPDK_BIND_PORT		Karthik S	openstack/networking-ovs-dpdk	master (bug/1461916)	3:46 PM	🟩		+1	
	Add multipath enhancement to Storwize iSCSI driver		xiaoqin	openstack/cinder	master (bp/storwize-iscsi-multipath-enhancement)	3:45 PM	🟧		+1	
	Support for both microversion headers		Chris Dent	openstack/nova	master (cd/handle-new-microversion)	3:45 PM	🟩	+1	-1	
	Fix Lun ID 0 in HPE 3PAR driver		Jay Mehta	openstack/cinder	master (bug/1573298)	3:45 PM	🟩		-1	
	portsecurity_db_common: Access db columns in a consistent way		Ihar Hrachyshka	openstack/neutron	stable/kilo (bug/1509312)	3:45 PM	🟩	✓		✓
	Generic details display framework		Matt Borland	openstack/horizon	master (bp/angularize-images-table)	3:45 PM	🟥			
	Remove translations for vendor specific networking-* projects		Andreas Jaeger	openstack-infra/project-config	master (stadium-redux)	3:45 PM	🟩	✓		✗
	Docs: Split Network Services section into multiple files		Pedro Magalhães (pjm6)	openstack/openstack-ansible	master (toctree_network_services)	3:45 PM	🟧	✓	+1	
	Added the value def inside the									

Git review



```
mika@heart ~sgit/playground (git)-[master|U] % git diff
diff --git a/README.md b/README.md
index ce8b250..3078d97 100644
--- a/README.md
+++ b/README.md
@@ -1 +1 @@
-Tue Dec 29 13:40:15 CET 2015
+Demo for OSDC 2016
mika@heart ~sgit/playground (git)-[master|U] % git commit -a -m 'MT#0042 git-review
[master 9385fbe] MT#0042 git-review demo for OSDC
 1 file changed, 1 insertion(+), 1 deletion(-)
mika@heart ~sgit/playground (git)-[master|↑1] % git review
remote: Processing changes: new: 1, refs: 1, done
remote: [redacted]
remote: New Changes:
remote:   http:/[redacted]se.com/5785 MT#0042 git-review demo for OSDC
remote:
To ssh://mprokop@gerrit.mgm.sipwise.com:29418/playground
 * [new branch]      HEAD -> refs/publish/master
mika@heart ~sgit/playground (git)-[master|↑1] %
```



Change 56412 - **Merged**

```
Deprecate projectName setting of CopyArtifact

Starting with version 1.26 of the CopyArtifact plugin all Jenkins job
configurations using this plugin get rewritten on the fly if you run at
least one build. This causes serious failures of the CopyArtifact
feature as soon as the project name includes axis selections.

If the config.xml of a job includes identical project *and* projectName
settings (this is what happens with JJB without this patch), like:

    <project>demojob/axis=$setting</project>
    <projectName>demojob/axis=$setting</projectName>

then it fails to run with current version(s) of the Copy Artifacts
plugin.

The problem is that while the web interface displays the correct
configuration for a fresh job created with JJB (Project Name:
"demojob/axis=$setting" and "Parameter filters" being empty), the
*wrong* configuration is actually used (Project Name: "demojob" and
"Parameter filters: "axis=$setting"). The reason is that when you
```

Author	Michael Prokop <openstack.org@michael-prokop.at>	Nov 14, 2013 4:47 PM
Committer	Michael Prokop <openstack.org@michael-prokop.at>	Nov 15, 2013 9:12 AM
Commit	183cccbe7912332faef862ea1ba05264daef0156	(gitweb)
Parent(s)	b3a28bd7869d762214ba654c06ebc8a4a4d65ac1	(gitweb)
Change-Id	Ie93316b135dc2b4547922062c7393ea7e5cc4e4c	

Owner	Michael Prokop
Reviewers	Clark Boylan James E. Blair Jenkins Jeremy Stanley Khai Do
Project	openstack-infra/jenkins-job-builder
Branch	master
Topic	
Updated	2 years, 5 months ago

Code-Review	+2	James E. Blair Jeremy Stanley Khai Do
	+1	Clark Boylan
Verified	+2	Jenkins
Workflow	+1	Jeremy Stanley

Jenkins check (1 rechecks)	Dec 4, 2013
gate-jenkins-job-builder-docs	SUCCESS in 55s
gate-jenkins-job-builder-pep8	SUCCESS in 17s
gate-jenkins-job-builder-python26	SUCCESS in 42s
gate-jenkins-job-builder-python27	SUCCESS in 47s

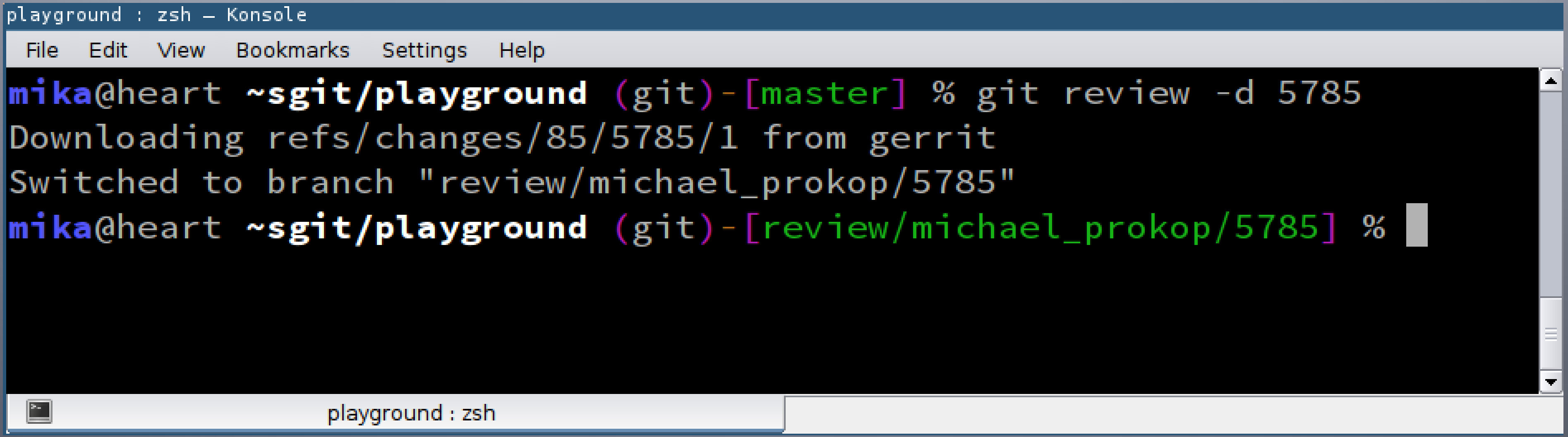
Files Open All Diff against: Base

File Path	Comments	Size
Commit Message		
jenkins_jobs/modules/builders.py		6
		+3, -3

History Expand All

Michael Prokop	Patch Set 1: FYI: Discussed this with upstream, quoting Jesse Glick from https://issues.jenkins-ci.org/brows	Nov 14, 2013
Khai Do	Patch Set 1: I would prefer that you didn't merge this I think this is ok, but my concern is that it's not backw	Nov 14, 2013
Michael Prokop	Uploaded patch set 2.	Nov 15, 2013
Michael Prokop	Uploaded patch set 3.	Nov 15, 2013
Michael Prokop	Patch Set 3: ACK, do you mean something like that (see patch set 3)?	Nov 15, 2013
Khai Do	Patch Set 3: Looks good to me (core reviewer)	Nov 15, 2013

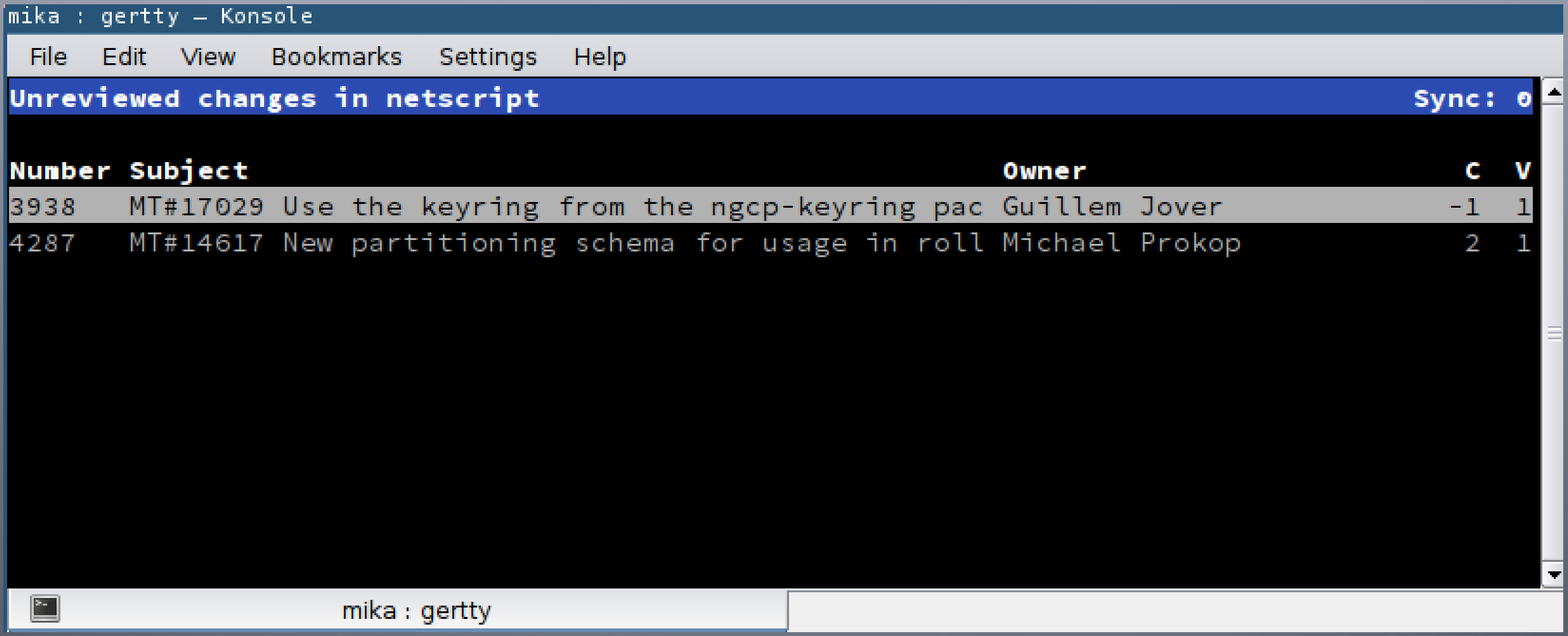
Git review - Download



playground : zsh — Konsole
File Edit View Bookmarks Settings Help
playground : zsh

```
mika@heart ~sgit/playground (git)-[master] % git review -d 5785
Downloading refs/changes/85/5785/1 from gerrit
Switched to branch "review/michael_prokop/5785"
mika@heart ~sgit/playground (git)-[review/michael_prokop/5785] %
```

~~Webinterface~~ Gertty



```
Unreviewed changes in netscript                                                      Sync: 0

Number Subject                                             Owner                          C  V
3938   MT#17029 Use the keyring from the ngcp-keyring pac Guillem Jover                 -1  1
4287   MT#14617 New partitioning schema for usage in roll Michael Prokop                 2  1
```

Gertty: use gerrit without the web

Good review culture

meaningful response cycles

be **friendly**

include + **integrate** newbies/new employees

THE #1 PROGRAMMER EXCUSE FOR LEGITIMATELY SLACKING OFF:
"~~MY CODE'S COMPILING.~~"
HEY! GET BACK TO WORK!
~~COMPILING!~~
Waiting for review!
OH. CARRY ON.

Source: https://xkcd.com/303/

Best Practices 1

keep reviews as **small** as feasible
(hint: also makes testing easier)

Best Practices 2

use Code Review also
for **Infrastructure**
changes! (IAC)

Best Practices 3

no direct pushes to
production branch

Traceability via Branches

Branch/Tag

↕

Environment/Release

↕

“vagrant up $release”

Traceability via Commits

“*ID#XXXXX*“ in commit message
to point to according
issue/bug number

Challenges

Different **timezones**

New **workflow**

Refactorings

Added **delays**

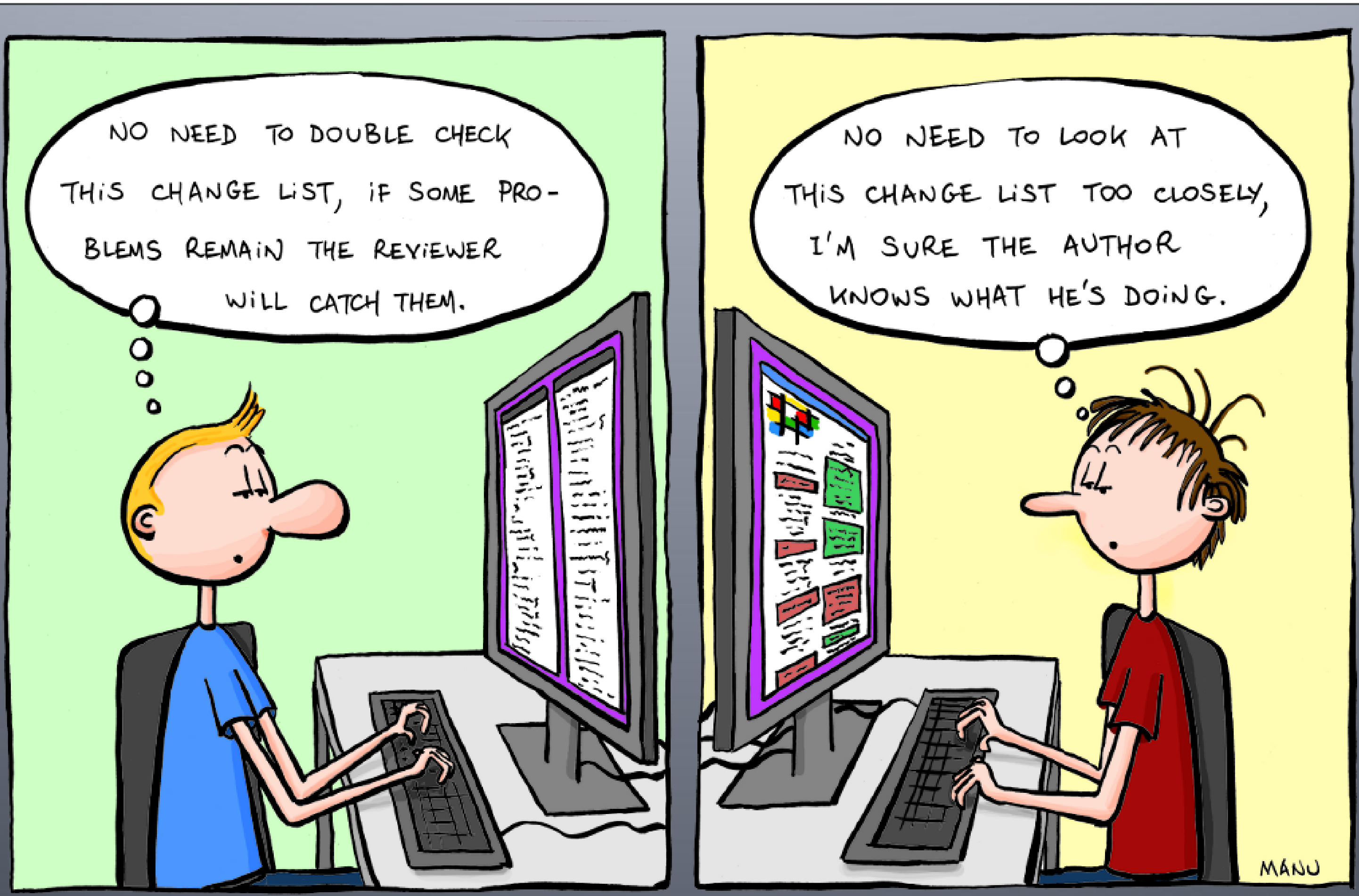
Culture **change**

Warning

Laaaaaarge change? → “Uff, +2”

Tiny change? → “Please rename $this, quote the variable, the logic could be simplified like…”

Cough


NO NEED TO DOUBLE CHECK THIS CHANGE LIST, IF SOME PROBLEMS REMAIN THE REVIEWER WILL CATCH THEM.
NO NEED TO LOOK AT THIS CHANGE LIST TOO CLOSELY, I'M SURE THE AUTHOR KNOWS WHAT HE'S DOING.
MANU

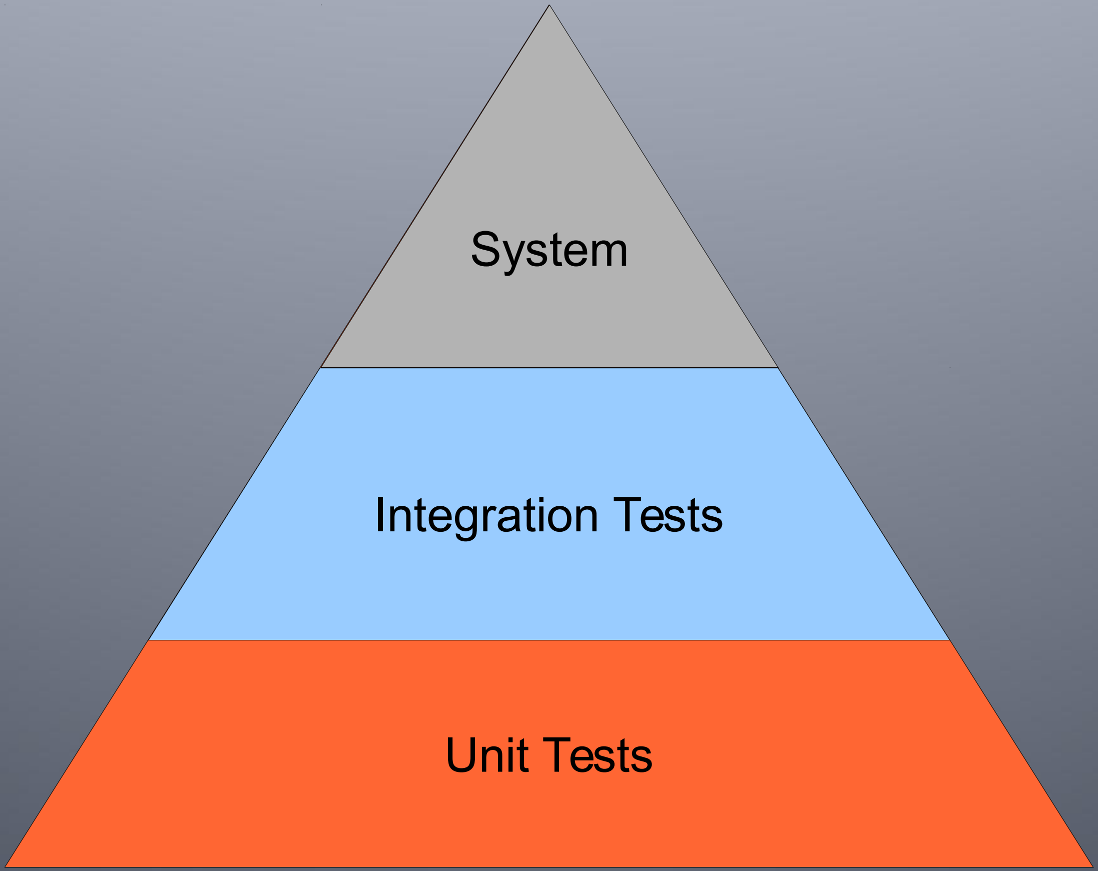
Source: http://www.bonkersworld.net/code-reviews/

Resources

- “Expectations, Outcomes, and Challenges of Modern Code Review” [URL]
- “Characteristics of Useful Code Reviews: An Empirical Study at Microsoft” [URL]
- “Code Reviews Do Not Find Bugs. How the Current Code Review Best Practice Slows Us Down” [URL]
- “On Rapid Releases and Software Testing” [URL]
- “Modern Code Reviews in Open-Source Projects: Which Problems Do They Fix?” [URL] and also the other “TestRoots Publications” [URL]

Testing Tools

Testing Pyramid



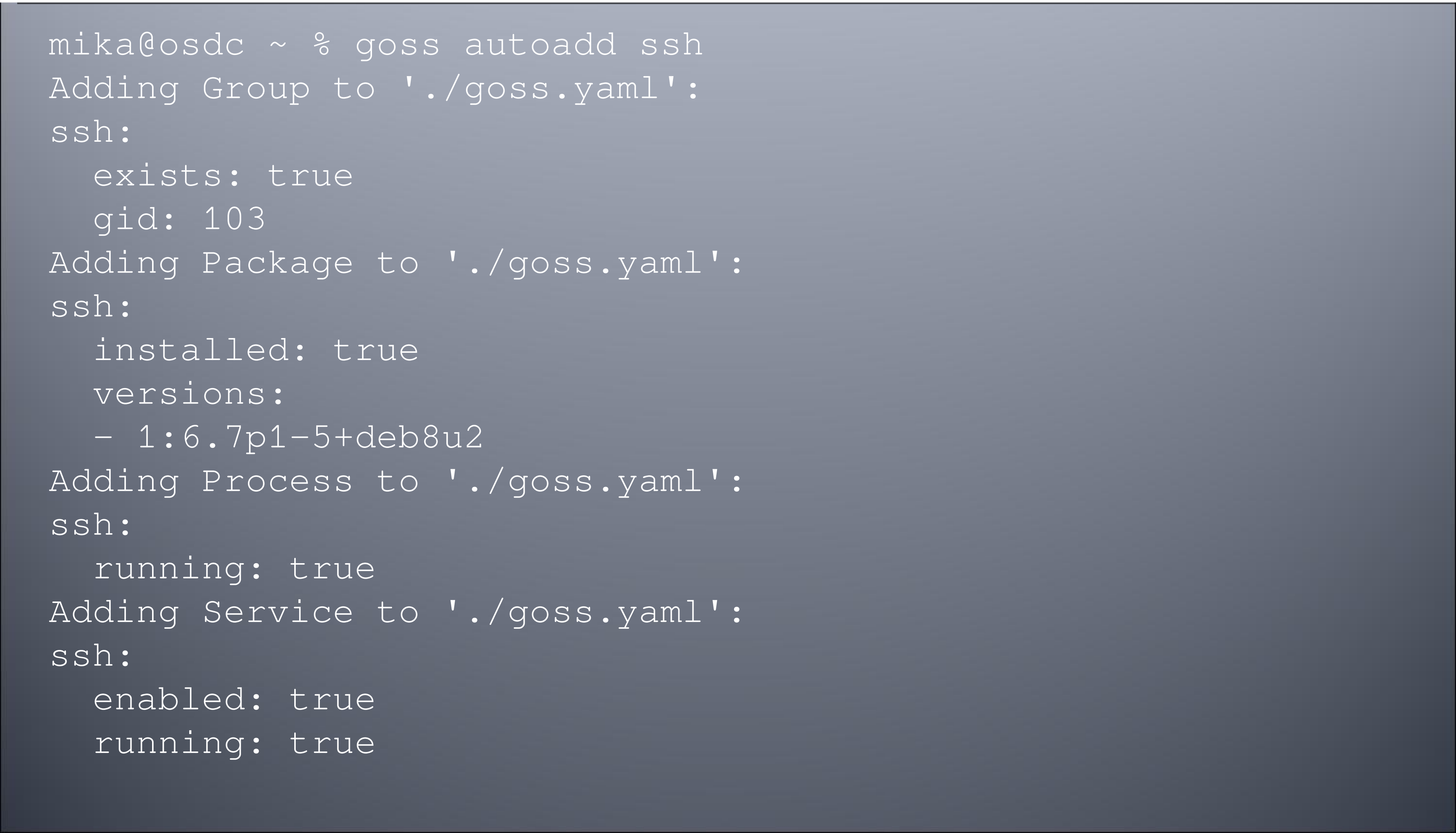
Goss

- https://github.com/aelsabbahy/goss/
- Serverspec-like tool for validation
- Golang → one single static binary
- RSpecish, nagios, json, JUnit + TAP output available
- Limited to Linux (so far)

Goss Resources

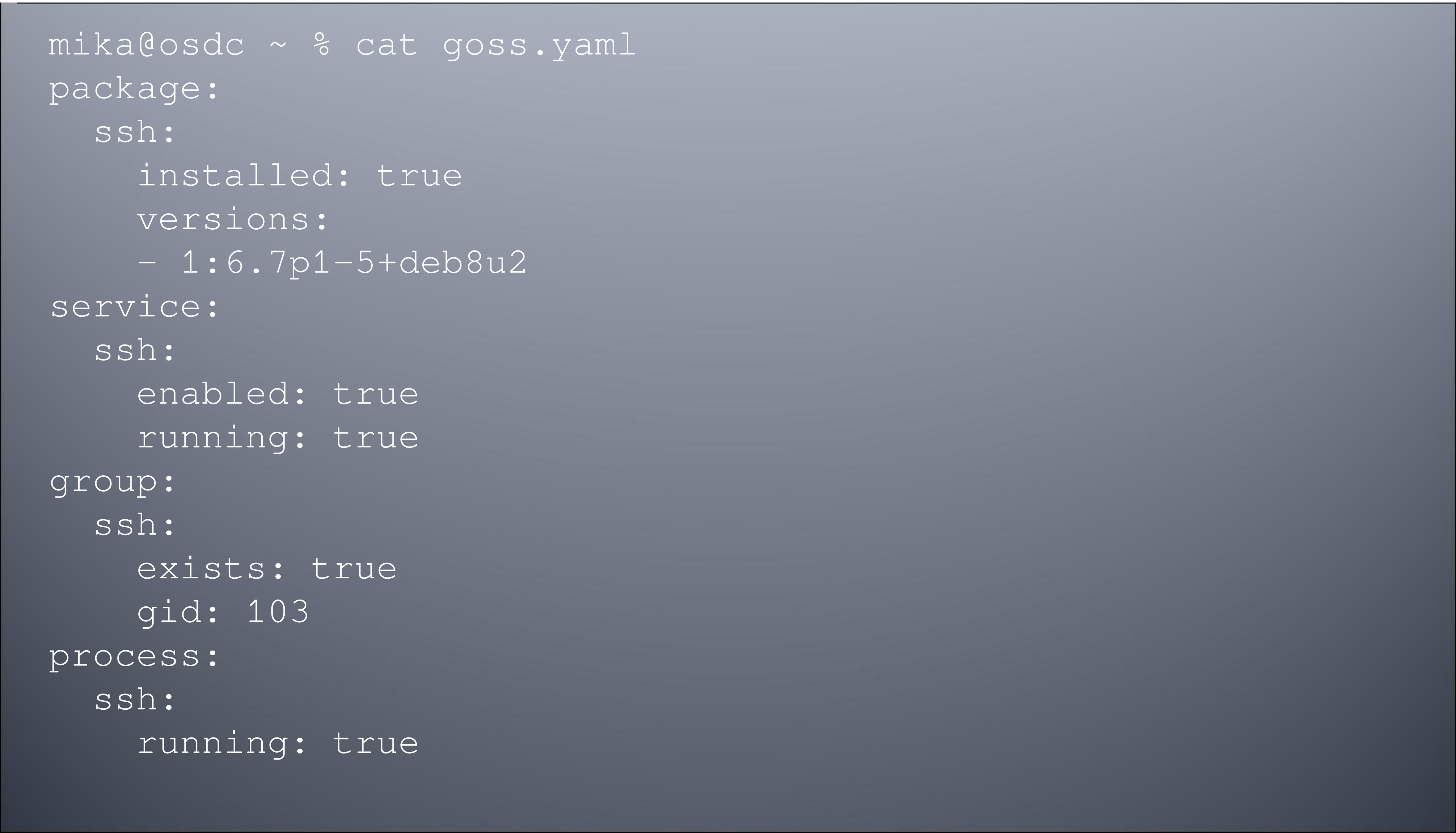
addr	is reachable
command	exit status and outputs
dns	is resolvable
file	file exists, owner/perm, content
group	uid
package	is listening, listening ip
port	is running
process	Is running
user	uid, home, etc..

Goss - Demo



```
mika@osdc ~ % goss autoadd ssh
Adding Group to './goss.yaml':
ssh:
  exists: true
  gid: 103
Adding Package to './goss.yaml':
ssh:
  installed: true
  versions:
  - 1:6.7p1-5+deb8u2
Adding Process to './goss.yaml':
ssh:
  running: true
Adding Service to './goss.yaml':
ssh:
  enabled: true
  running: true
```

Goss - Demo



```
mika@osdc ~ % cat goss.yaml
package:
  ssh:
    installed: true
    versions:
    - 1:6.7p1-5+deb8u2
service:
  ssh:
    enabled: true
    running: true
group:
  ssh:
    exists: true
    gid: 103
process:
  ssh:
    running: true
```

Goss - Demo

```
mika@osdc ~ % goss validate goss.yaml
.......

Total Duration: 0.066s
Count: 7, Failed: 0
mika@osdc ~ % goss add port 22
Adding Port to './goss.yaml':

tcp:22:
  listening: true
  ip:
  - 0.0.0.0
mika@osdc ~ % goss validate goss.yaml
.........

Total Duration: 0.070s
Count: 9, Failed: 0
```

Goss - Demo

```
mika@osdc ~ % sed -i 's/tcp:22/tcp:23/' goss.yaml
mika@osdc ~ % goss validate --format tap goss.yaml
1..9
ok 1 - Group: ssh: exists: matches expectation: [true]
ok 2 - Group: ssh: gid: matches expectation: [103]
ok 3 - Service: ssh: enabled: matches expectation: [true]
ok 4 - Service: ssh: running: matches expectation: [true]
ok 5 - Process: ssh: running: matches expectation: [true]
not ok 6 - Port: tcp:23: listening: doesn't match, expect:
[true] found: [false]
not ok 7 - Port: tcp:23: ip: doesn't match, expect:
[["0.0.0.0"]] found: [null]
ok 8 - Package: ssh: installed: matches expectation: [true]
ok 9 - Package: ssh: version: matches expectation: [["1:6.7p1-
5+deb8u2"]]
```

py.test

- Pytest.org
- Fixtures, Scopes, Monkeypatching
- JUnit + TAP output (hello Jenkins!)
- Plugins
- Example + starting point:
  - https://github.com/vincentbernat/lldpd/tree/master/tests/integration
  - http://www.slideshare.net/VincentBernat/pytest-all-the-things

Package dependencies

- Dose-distcheck
- Are package dependencies/conflicts satisfiable?
  - dose-debcheck for Debian packages
  - dose-rpmcheck for rpm packages
  - dose-eclipsecheck for OSGi plugins

Puppet Testing

Puppet-lint

RSpec-puppet

Beaker

Puppet testing: see David Schmitt’s talk “*Introduction to Testing Puppet Modules*” at 2:15pm on Thursday at OSDC

Tests + Infrastructure 1

Tests for **everything** ASAP!

avoid problems during commit/push time +
don't rely on historic/explicit knowledge →
new employees will have a hard time otherwise

Tests + Infrastructure 2

Test systems to
work on infrastructure
without breaking production

Tests + Infrastructure 3

Test infrastructure
really for **everything**

testing a change for the **test** builds/scripts?
use **test** environment!

Related Resources

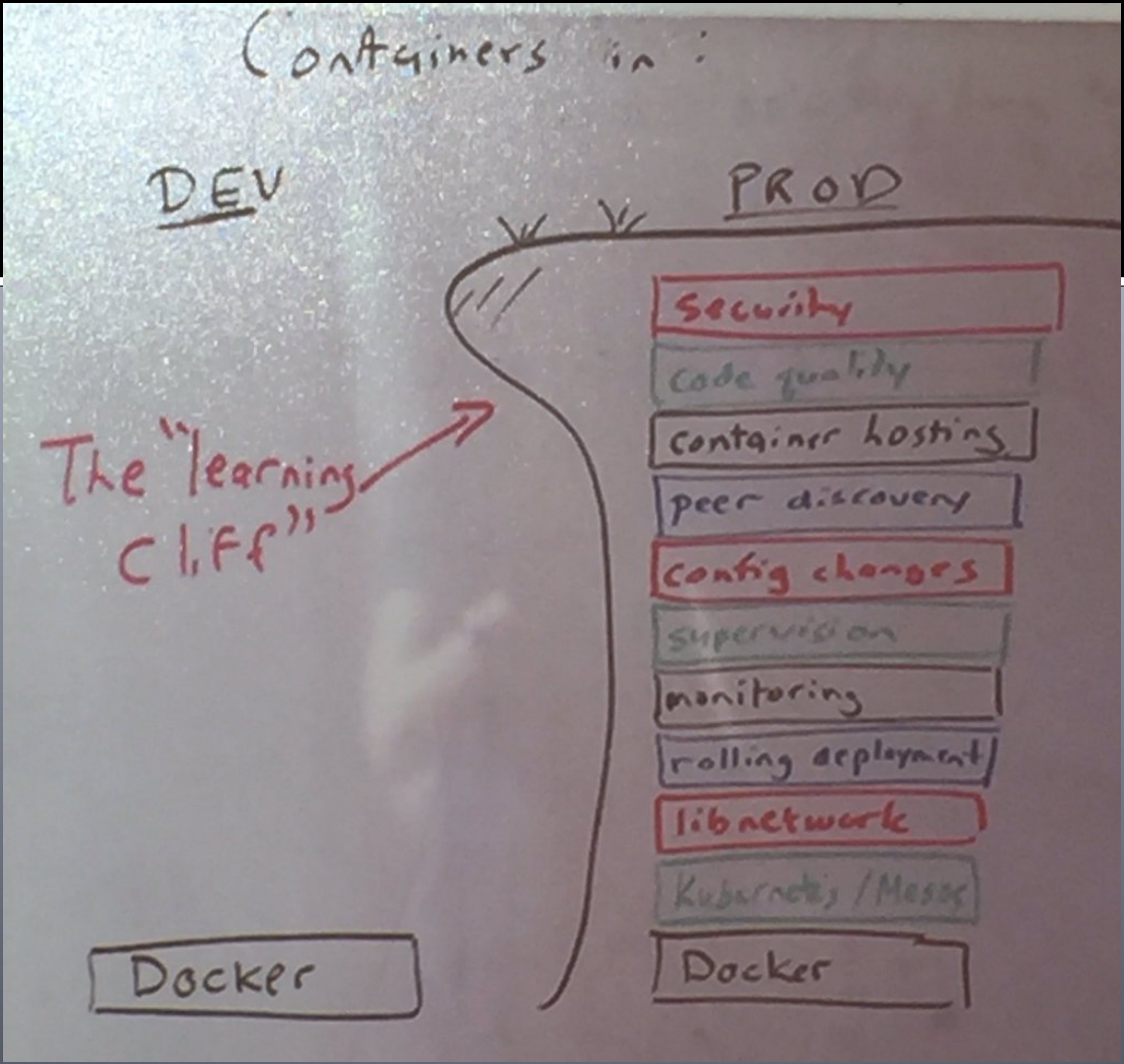
- „Advanced Testing with Go“ [URL]
- „System Testing with pytest and docker-py“ [URL]
- Book “How Google Tests Software”

Source: https://twitter.com/francesc/status/718604718294097920

Docker

Obligatory Docker Slide

Source: https://twitter.com/sadserver/status/718455853540487168



Source: https://twitter.com/mfdii/status/697532387240996864

Our use cases

Fast test cycle to avoid long feedback loops in CI/CD pipeline

Share environment between developers and testing infrastructure

Developers should be able to control testing infrastructure

Internal docker registry

~~Docker-registry~~
Docker-distribution (implementation of Docker Registry HTTP API V2 for docker 1.6+)

Fast moving

Workflow

Every project ships its own Dockerfile(s), Jenkins identifies changes and **rebuilds** the docker images

Image lifecycle is **tricky** though (when to build/destroy, naming + tagging conventions,…)

Infrastructure Tooling

Puppet related

- Puppet environments + r10k
- Hiera
- Puppetdb + ansible
- Mcollective
- Puppet codebase from 2.7 to 4.3
- Puppetdashboard → puppetboard
- Own puppet modules → Puppet Forge

Monitoring?

Long running tests failing with ENODISKSPACE

automatic cleanups
(get rid of old artifacts, caches,…) +
use appropriate monitoring

Monitoring

Metrics + Logging

what is taking long + is worth improving/investigating/....?
(ELK, Graylog, InfluxDB, Grafana,…)

Monitoring Software

Check-mk

Icinga 2 with puppet-icinga2

Vagrant/Packer + grml-debootstrap

Vagrant + Packer by HashiCorp
(Packer = replacement for Veewee,
hit Debian/unstable *today* [URL])

Grml-debootstrap for building Debian
(based) base boxes [URL]

Disaster recovery

- Daily fresh installation of nodes
- PIN-protected USB pen drive with sensitive data on it (e.g. hiera)
- iPXE + Grml ISO with netscript=… boot option to deploy
- netboot.xyz

Source: https://www.istorage-uk.com/product/datashur/

Antipatterns

Like a boss

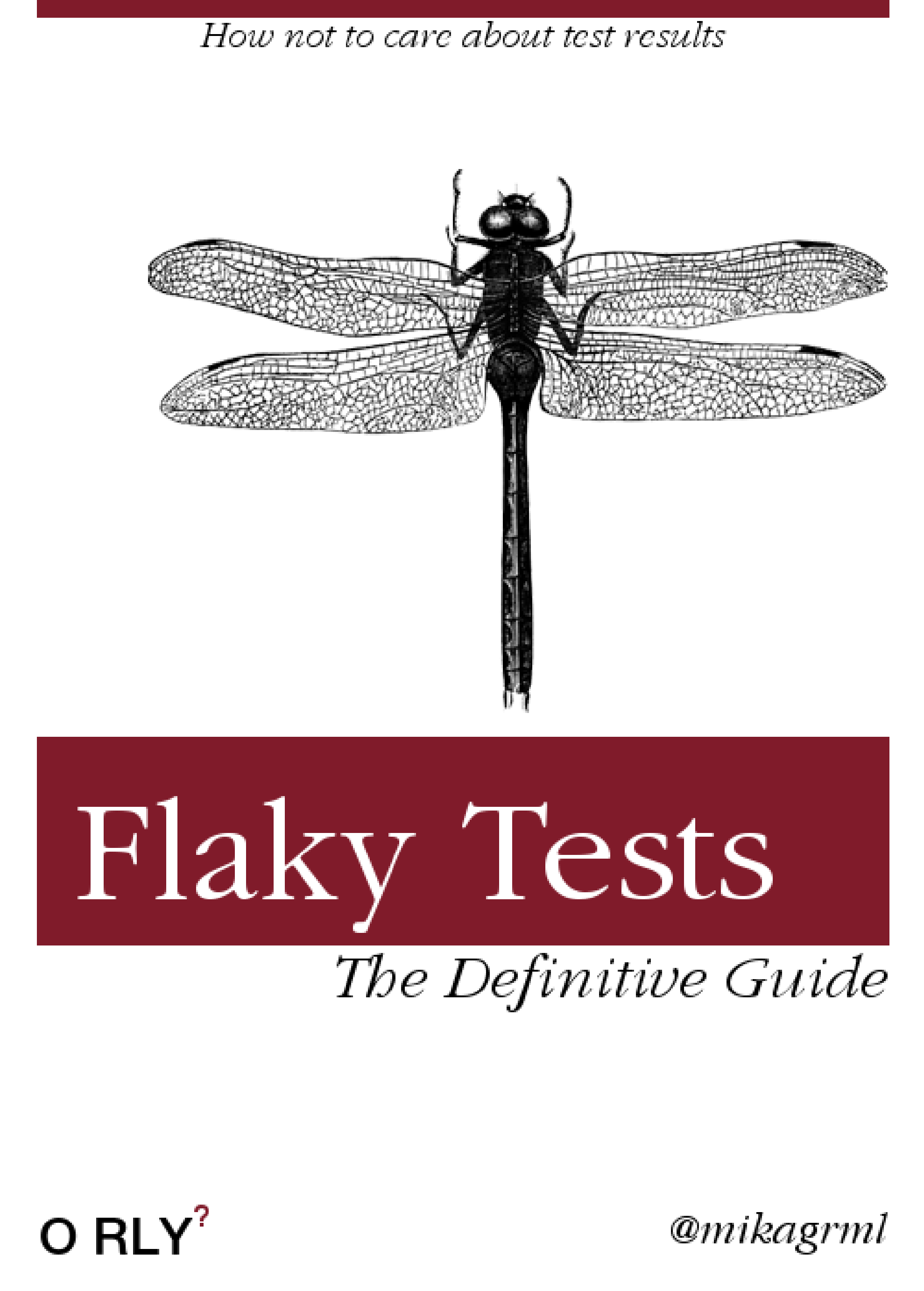
Let's just fix this via SSH

The Definitive Guide

O RLY?

@mikagrml

Thx @ThePracticalDev + http://dev.to/rly



How not to care about test results

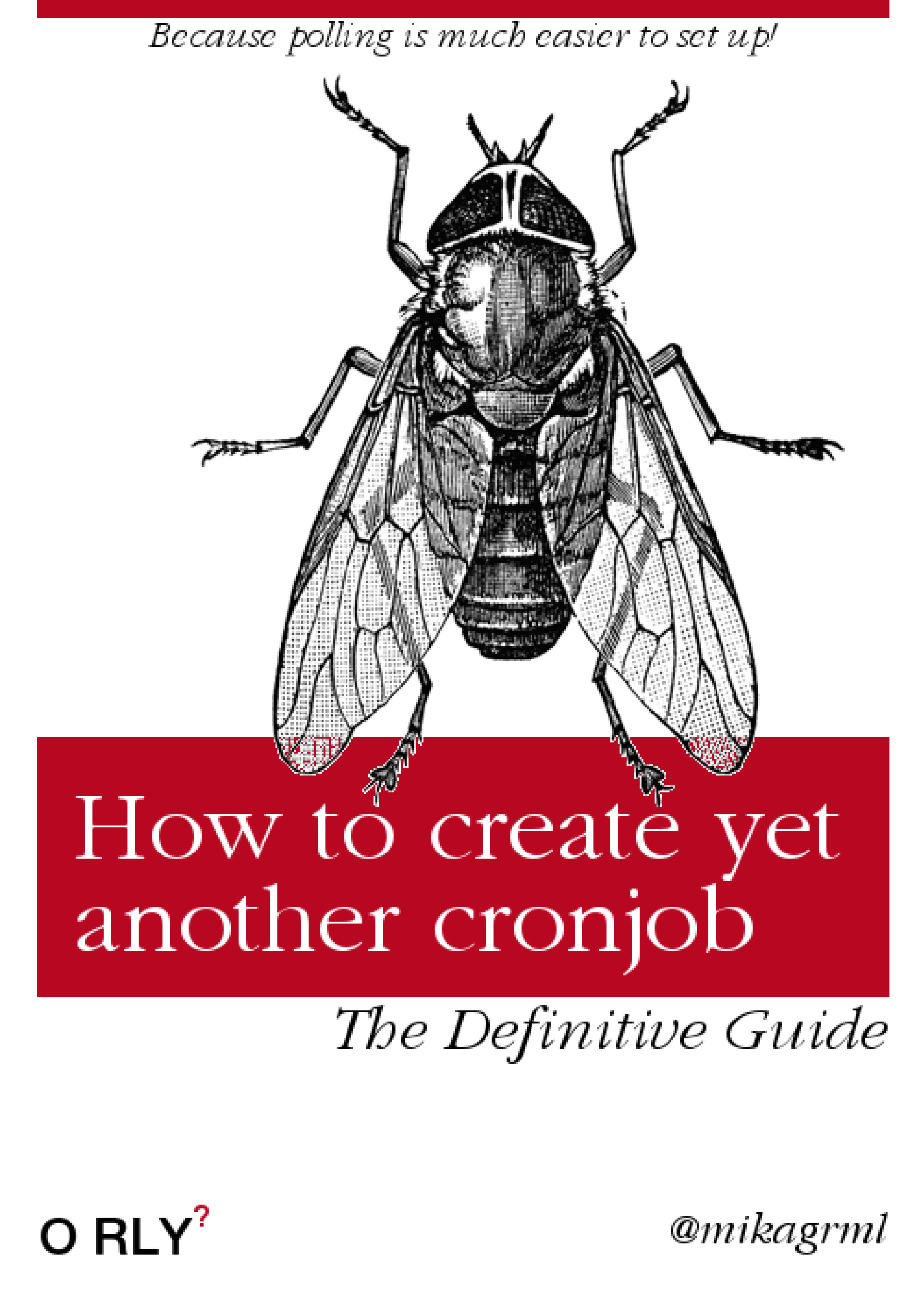
Flaky Tests

The Definitive Guide

O RLY?

@mikagrml

Thx @ThePracticalDev + http://dev.to/rly



Because polling is much easier to set up!

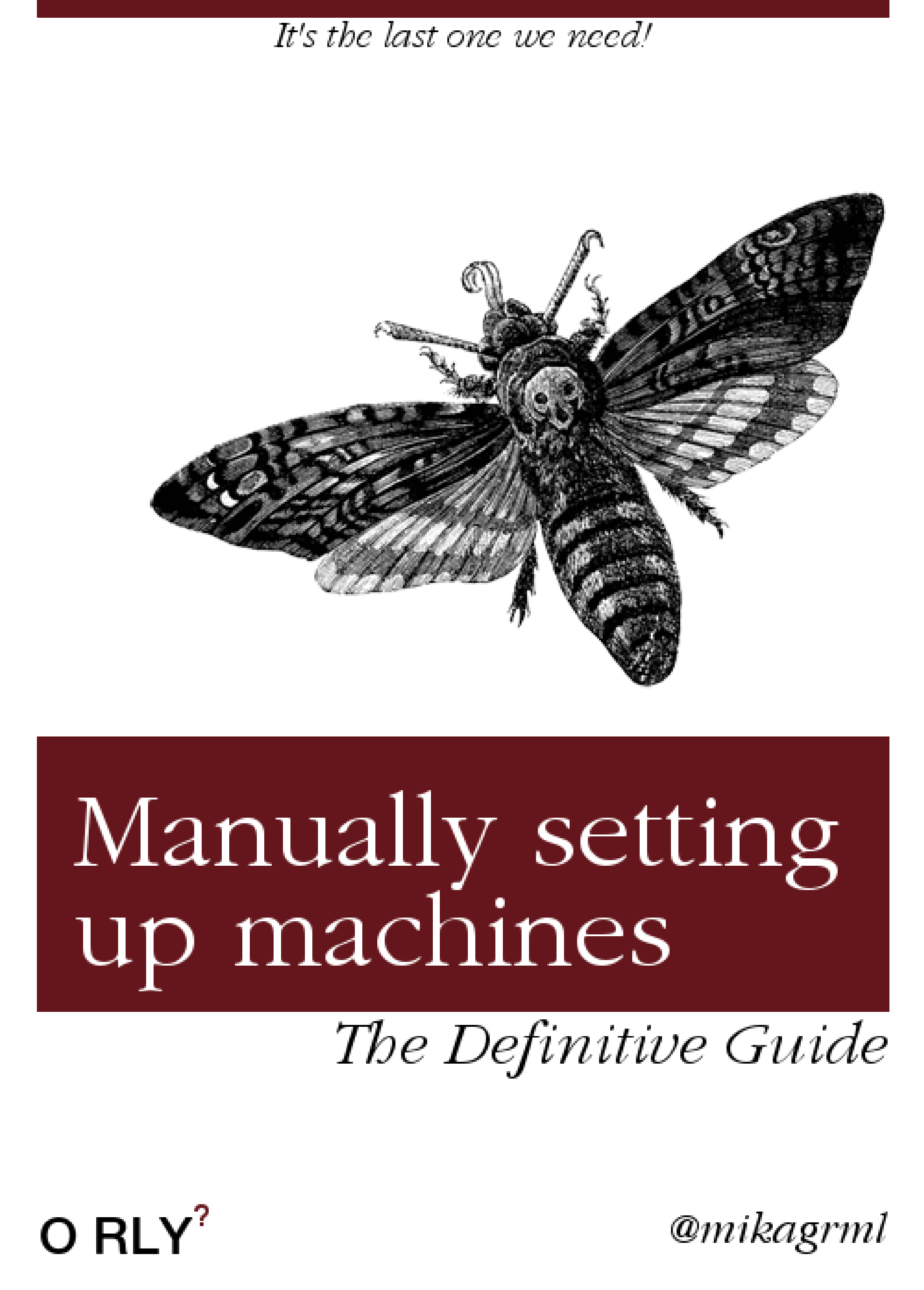
How to create yet another cronjob

The Definitive Guide

O RLY?

@mikagrml

Thx @ThePracticalDev + http://dev.to/rly



It's the last one we need!

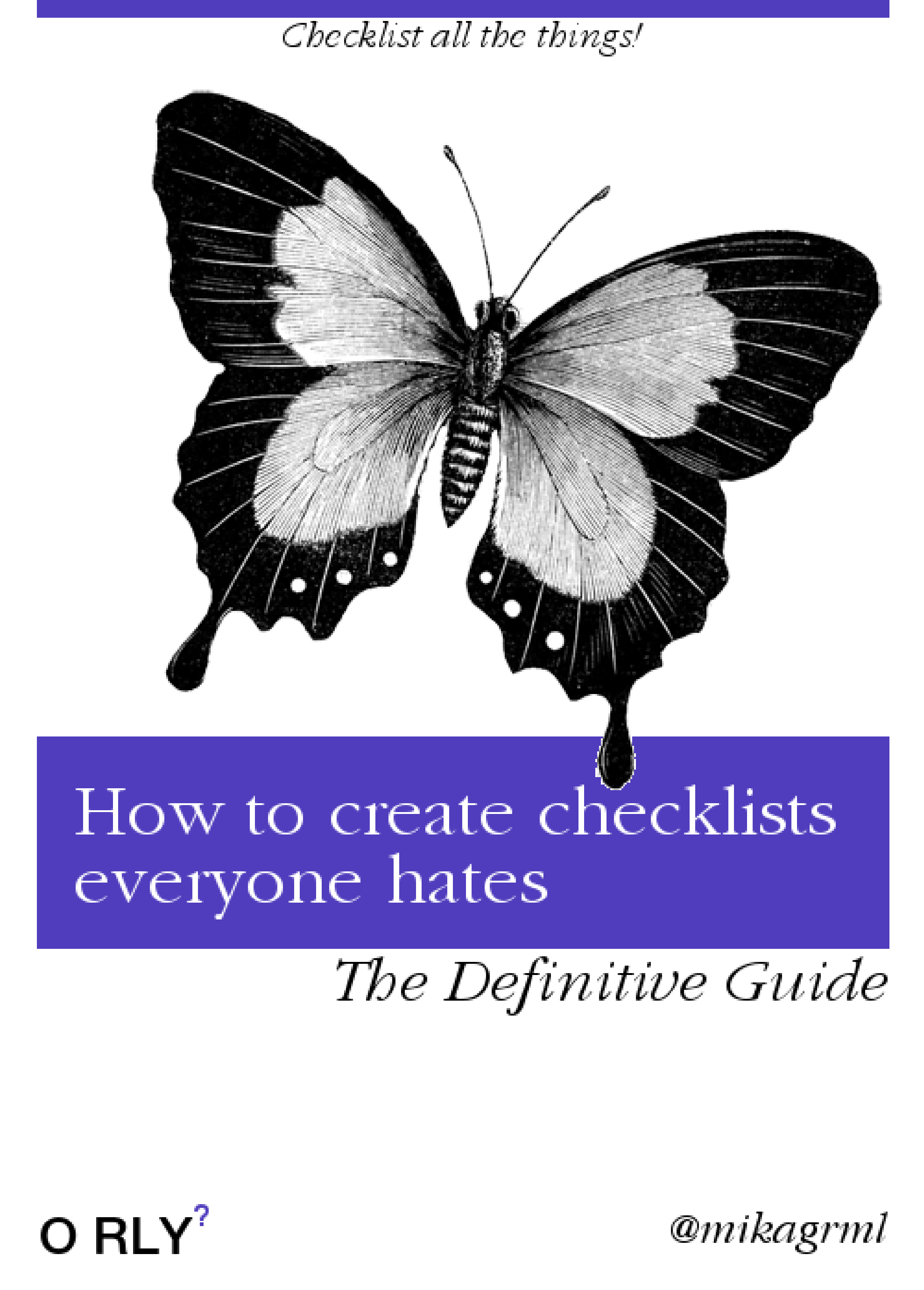
Manually setting up machines

The Definitive Guide

O RLY?

@mikagrml

Thx @ThePracticalDev + http://dev.to/rly



Checklist all the things!

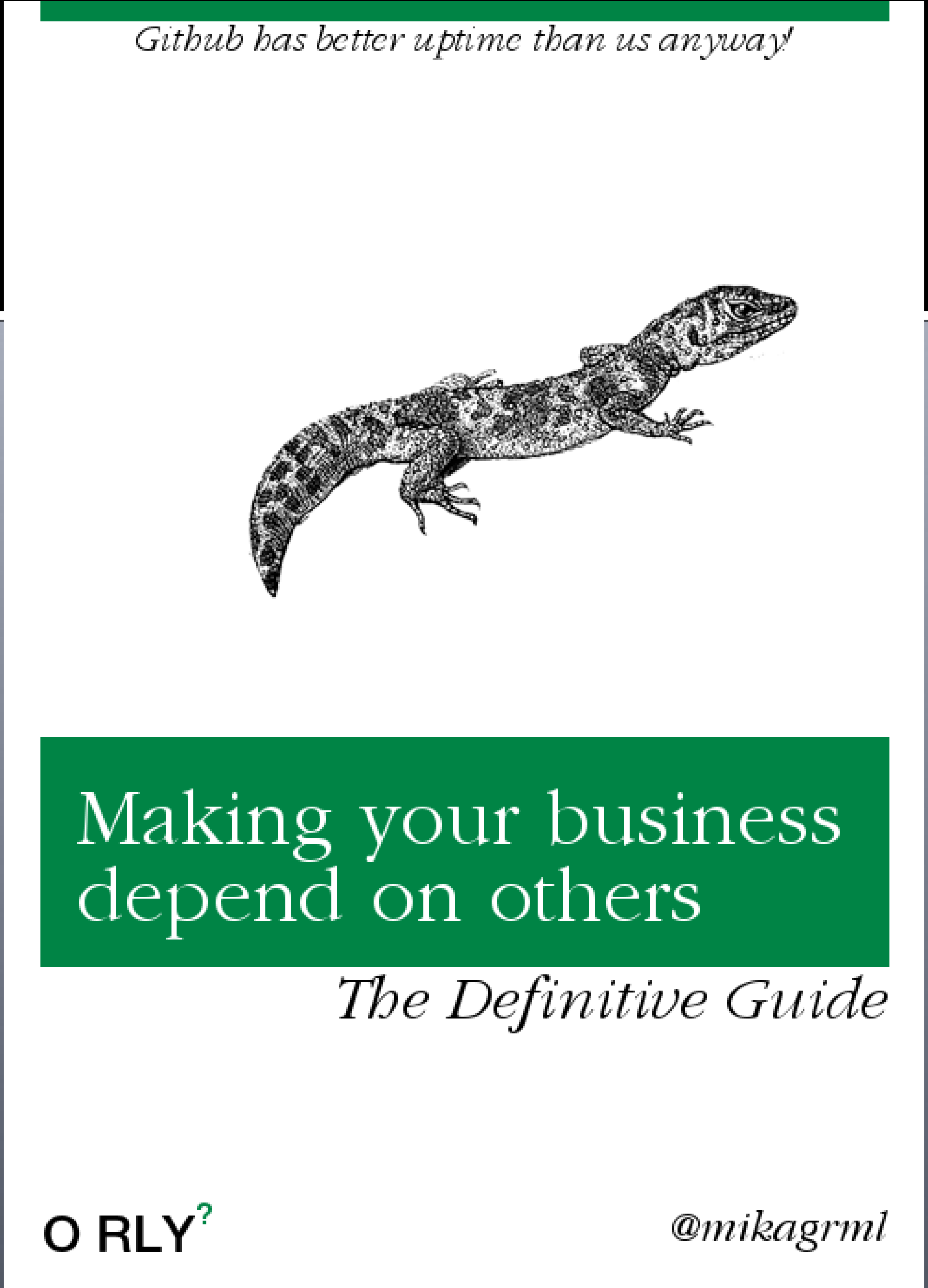
How to create checklists everyone hates

The Definitive Guide

O RLY?

@mikagrml

Thx @ThePracticalDev + http://dev.to/rly



Github has better uptime than us anyway!

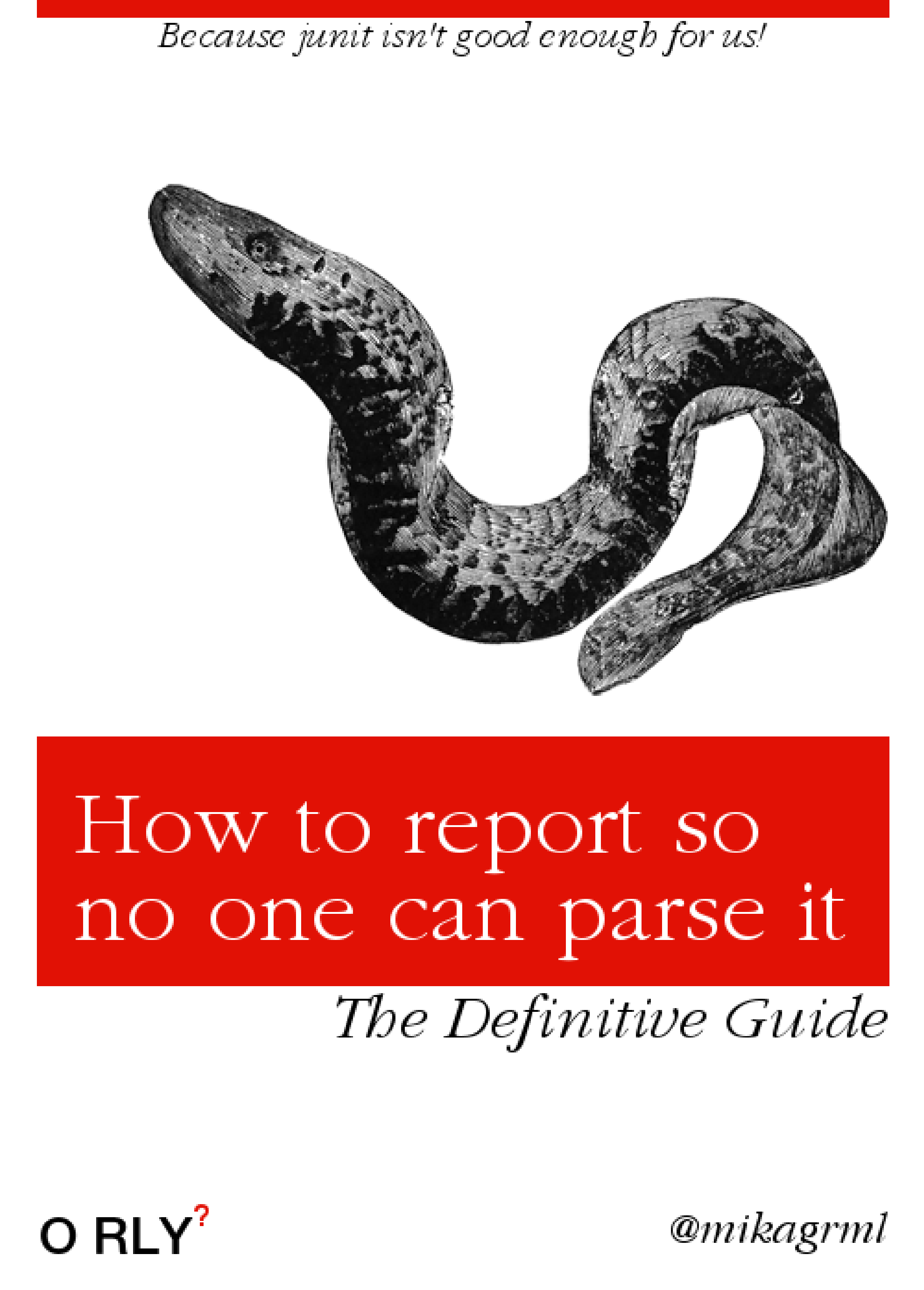
Making your business depend on others

The Definitive Guide

O RLY?

@mikagrml

Thx @ThePracticalDev + http://dev.to/rly



Because junit isn't good enough for us!

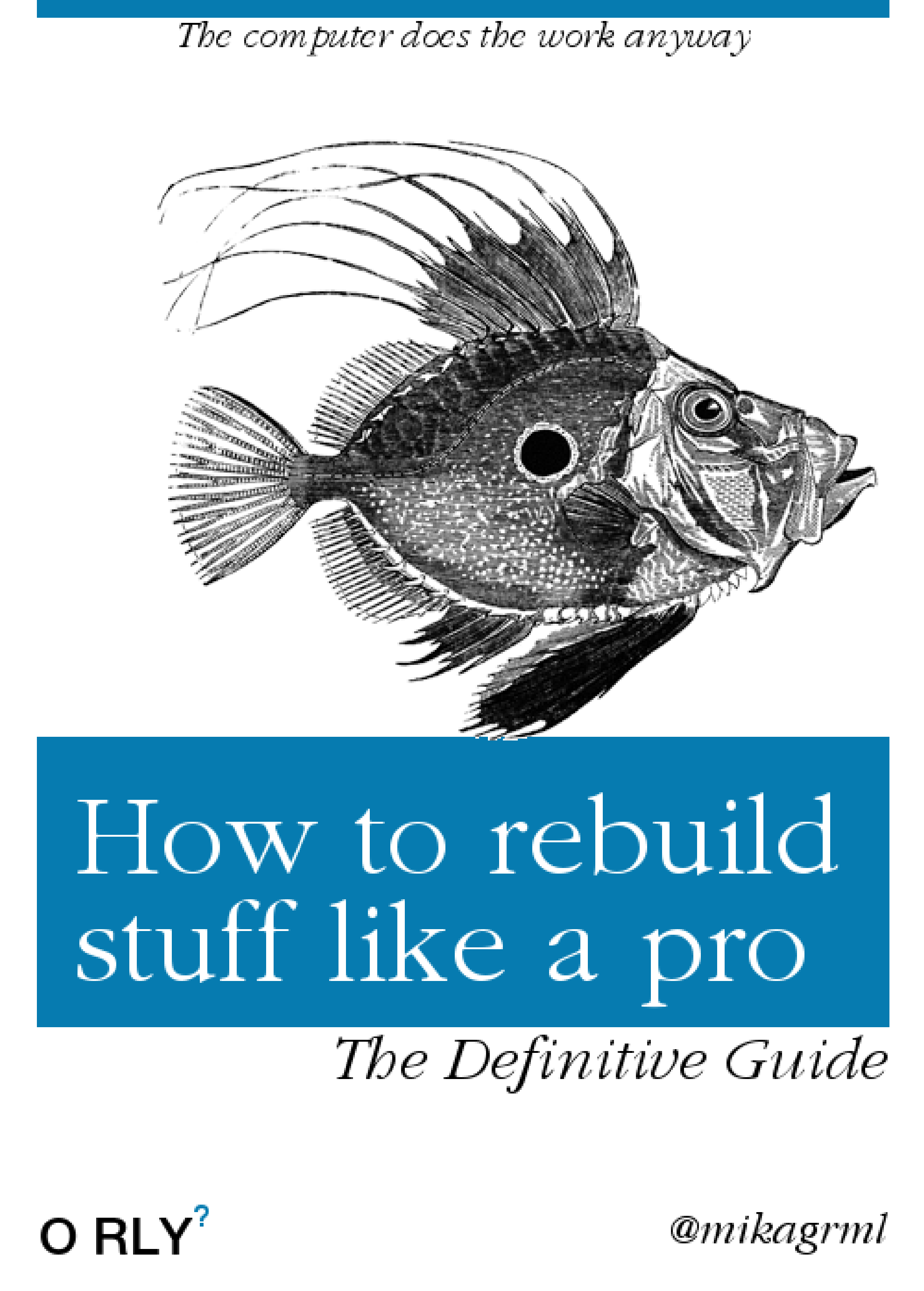
How to report so no one can parse it

The Definitive Guide

O RLY?

@mikagrml

Thx @ThePracticalDev + http://dev.to/rly



The computer does the work anyway

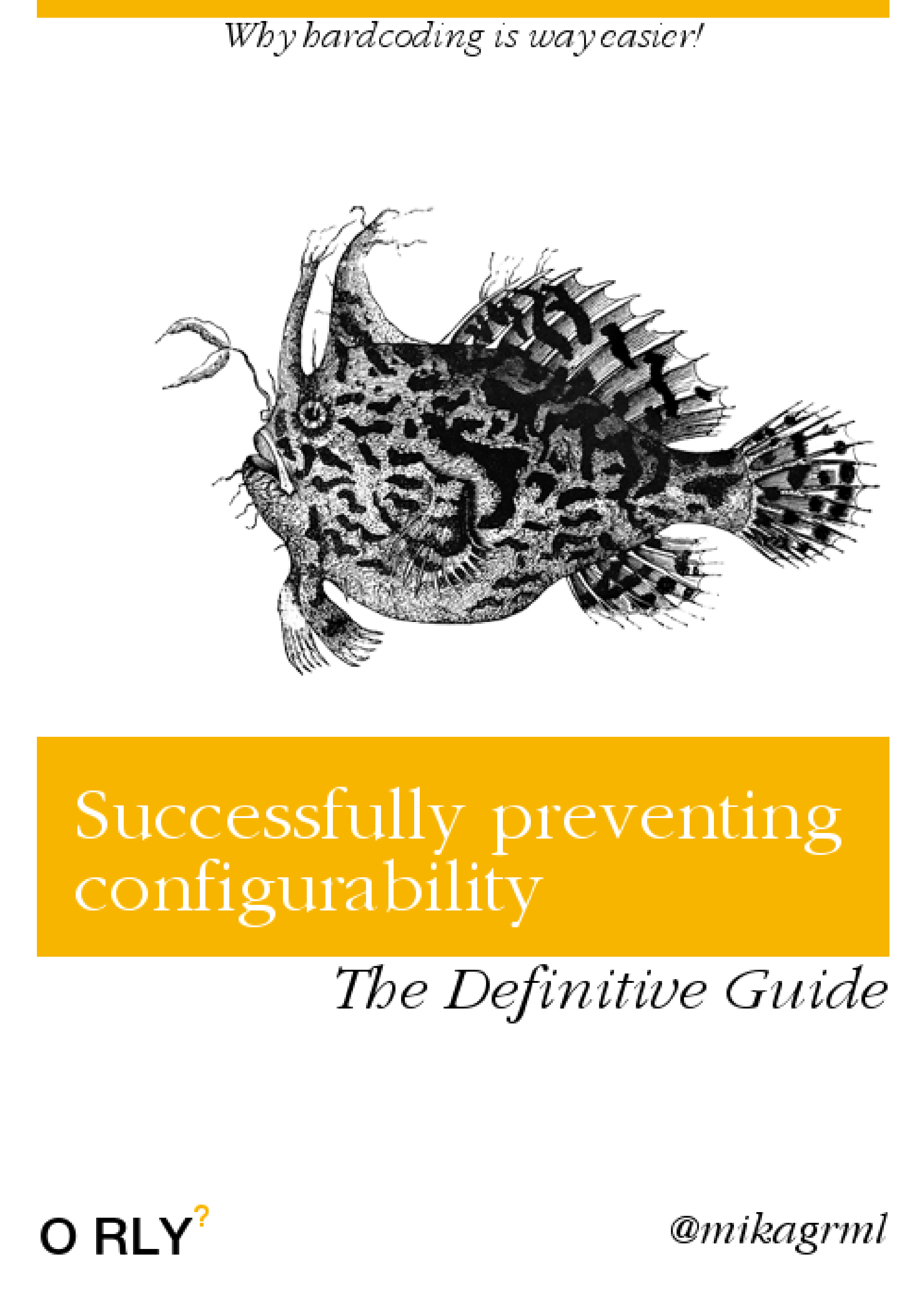
How to rebuild stuff like a pro

The Definitive Guide

O RLY?

@mikagrml

Thx @ThePracticalDev + http://dev.to/rly



Why hardcoding is way easier!

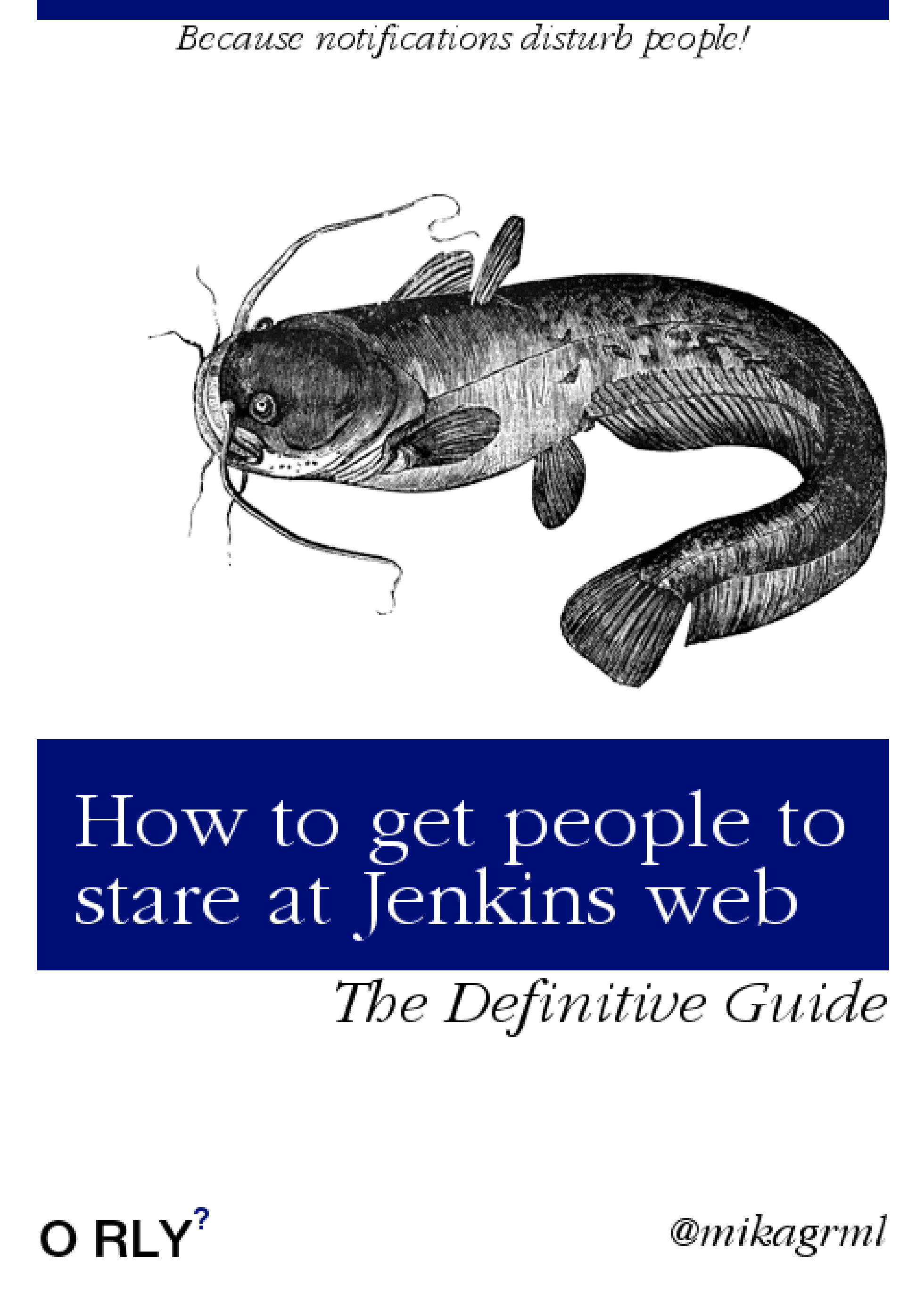
Successfully preventing configurability

The Definitive Guide

O RLY?

@mikagrml

Thx @ThePracticalDev + http://dev.to/rly



Because notifications disturb people!

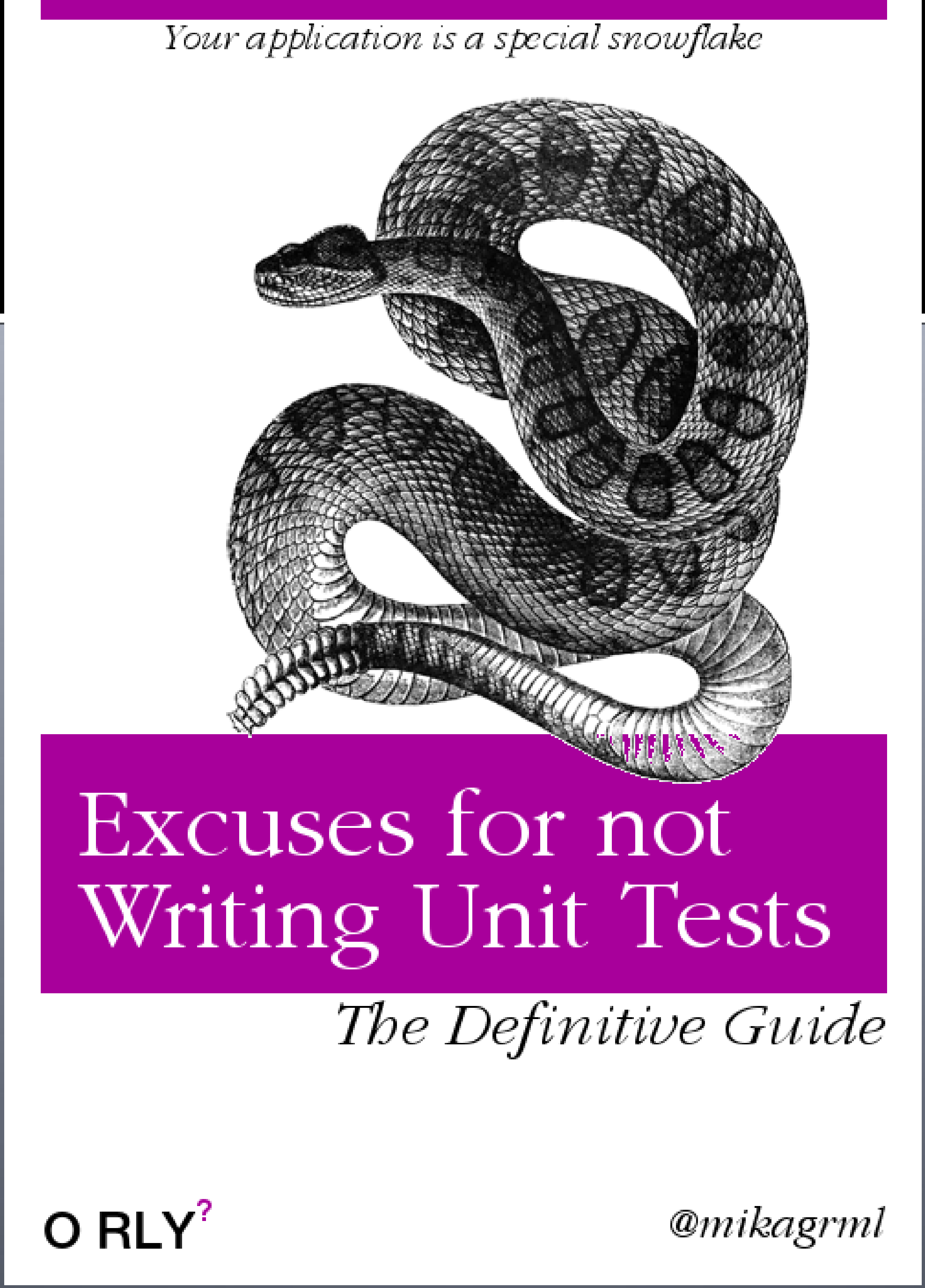
How to get people to stare at Jenkins web

The Definitive Guide

O RLY?

@mikagrml

Thx @ThePracticalDev + http://dev.to/rly



Your application is a special snowflake

Excuses for not Writing Unit Tests

The Definitive Guide

O RLY?

@mikagrml

Thx @ThePracticalDev + http://dev.to/rly

Pain Points

Misunderstandings 1

Different people/teams have **different use cases, understandings,…** of certain things (e.g. meaning + usage of -1/-2/+1/+2 in Gerrit)

Fix via **documentation**!

Misunderstandings 2

Cultural **differences**

Fix via „Team Handbook“,
like the one from Gitlab [URL]

Debian related 1

Race-free package info updates
AKA „Hashsum Mismatch Error“

Fix see [URL]

Debian related 2

Avoid **Pre-Depends**, they impose harsh constraints on the package manager, consequently make upgrades harder + also cause more problems with piuparts

Debian related 3

The more sophisticated the systems become, the closer the **toolchain** needs stuff Debian has + uses (britney/dak/piuparts/nose/…)

Debian related 4

Mass changes?

Repository **locking** problem
with e.g. reprepro :(

Jenkins related 1

Structuring of hundreds/thousands of Jenkins jobs in **jenkins-job-builder** isn't easy, esp. the more exceptions you have

Jenkins related 2

Unreliable tests?

Run them outside of the production pipeline!
Use whitelists/blacklists to reach 100% coverage over time!

Jenkins related 3

Matrix jobs in Jenkins are not always fun, use simple plain **freestyle Jenkins jobs** when possible

Availability

Downtime of Jenkins :(

Lenient Shutdown
Zuul + Gearman
Jenkins 2.0 with Pipeline

System Design

Some architecture **decisions** are visible only after surviving a new $release (Debian: wheezy→jessie)

Separation

Full-featured PPAs: bound together with Gerrit topics to share packages from different projects (new features, introducing new components/build-depends,…)

Avoid backwards compatibility nightmare

Release-specific settings into **release-specific** configurations/scripts/…

Take-aways

Take it home

- Automation is essential
- Configuration Management
- Traceability
- Code Review
- Tests + Testing
- Communication is important
- Bring devs and ops together (“devops“)

Wishes || Questions?

@mikagrml
mika @ github
michael-prokop.at/blog/
prokop (at) grml-solutions.com