

Continuous Delivery Pipeline

Aaron Blythe

<https://www.youtube.com/watch?v=SlaVsG7m8n4>

@ablythe

@ablythe



<http://www.slideshare.net/aaronblythe>

https://github.com/aaronblythe/jenkins_scripts

@ablythe

research



academic programs

welcome



people

department of
Mechanical & Industrial
engineering

University of Illinois at Urbana-Champaign

calendar



alumni



welcome

academic programs

website overview

research

people

calendar

alumni

opportunities

@ablythe



DevOps Kansas City



@DevOpsKC

<http://www.meetup.com/DevOps-Kansas-City/>

@ablythe

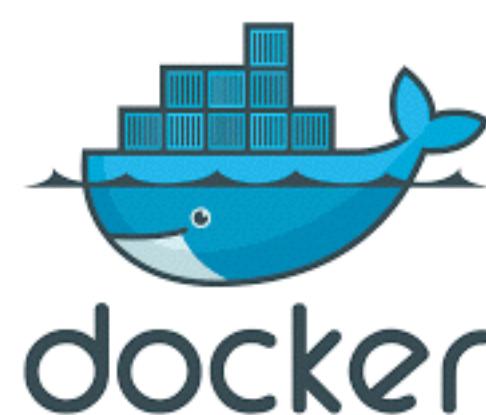
Tools



Source Control



Cloud Platform



Containerization



Continuous Integration/Delivery



Configuration Management

@ablythe

hp laserjet firmware team

2008

10% - code integration

20% - detailed planning

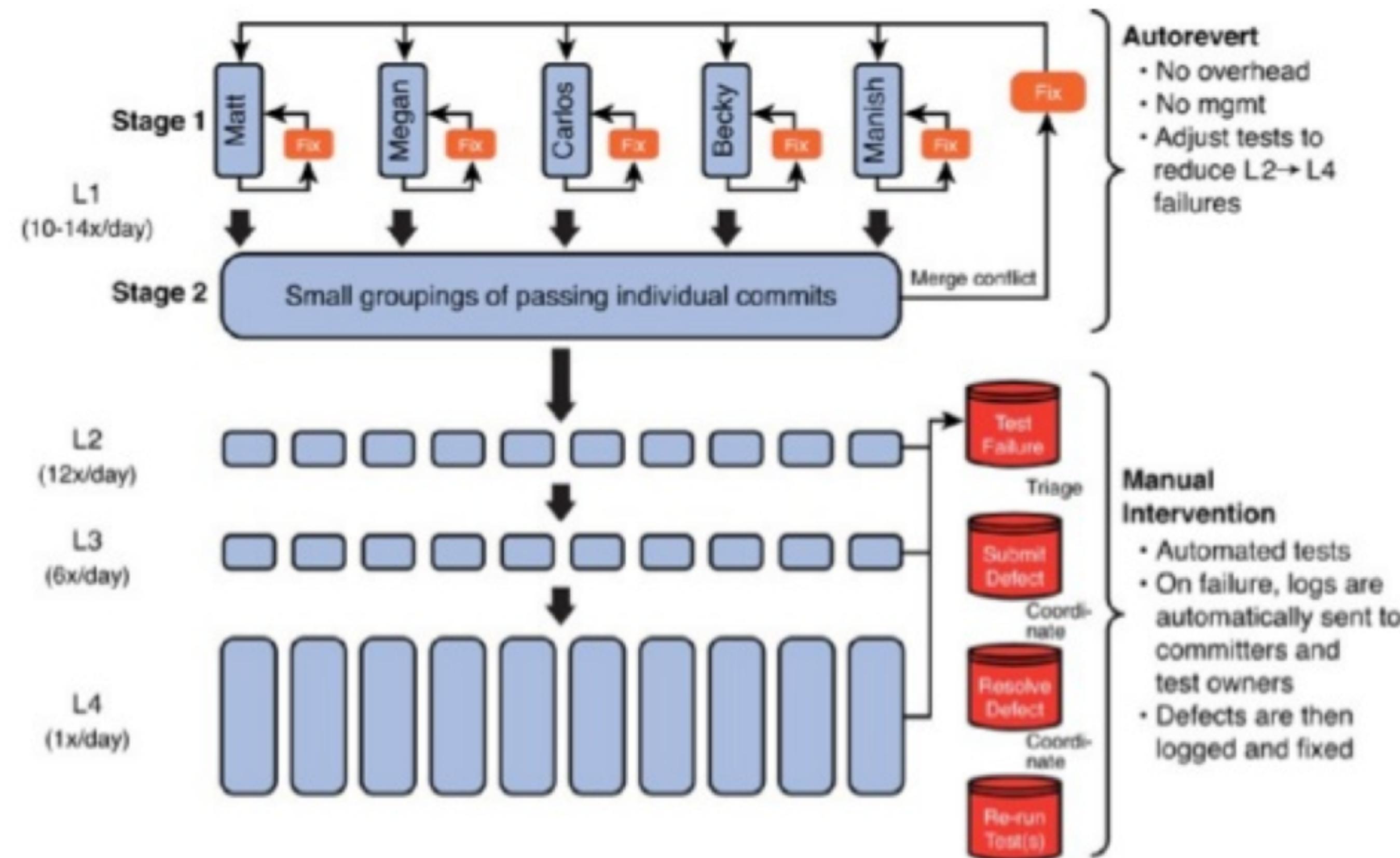
25% - porting code

25% - current product support

15% - manual testing

~5% - innovation

deployment pipeline



Friday, July 5, 13

<https://twitter.com/ablythe/status/643866472377401344>
@ablythe

hp laserjet firmware team

2008

10% - code integration

20% - detailed planning

25% - porting code

25% - current product support

15% - manual testing

~5% - innovation

2011

2% - continuous integration

5% - agile planning

15% - one main branch

10% - one branch cpe

5% - most testing automated

~40% - innovation



The remaining 23% on RHS is spent on managing automated tests.

So when can you say you're doing continuous delivery? I'd say it's when you could flip a switch to go to continuous deployment if you decided that was the best way to deliver value to your customers.

<http://continuousdelivery.com/2010/08/continuous-delivery-vs-continuous-deployment/>

CONTINUOUS DELIVERY



CONTINUOUS DEPLOYMENT



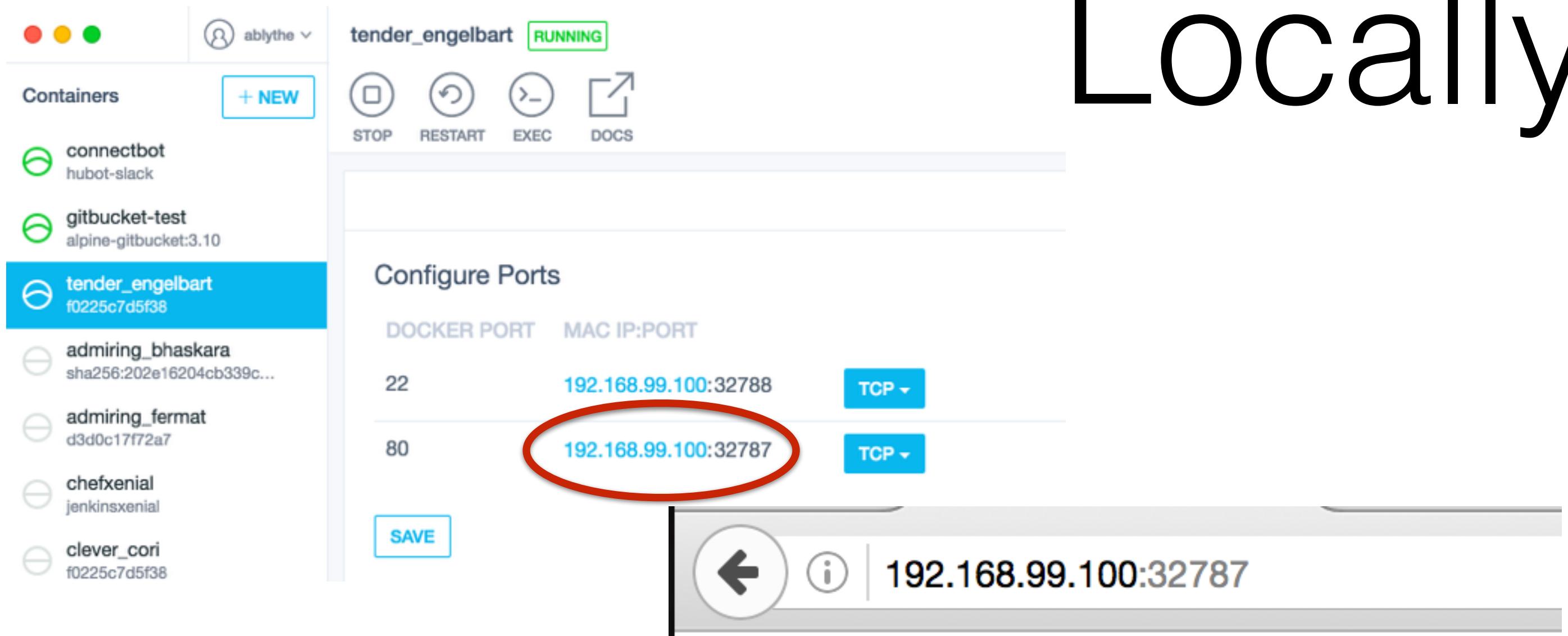
Live Demo



<http://www.slideshare.net/aaronblythe>

https://github.com/aaronblythe/jenkins_scripts
@ablythe

Locally



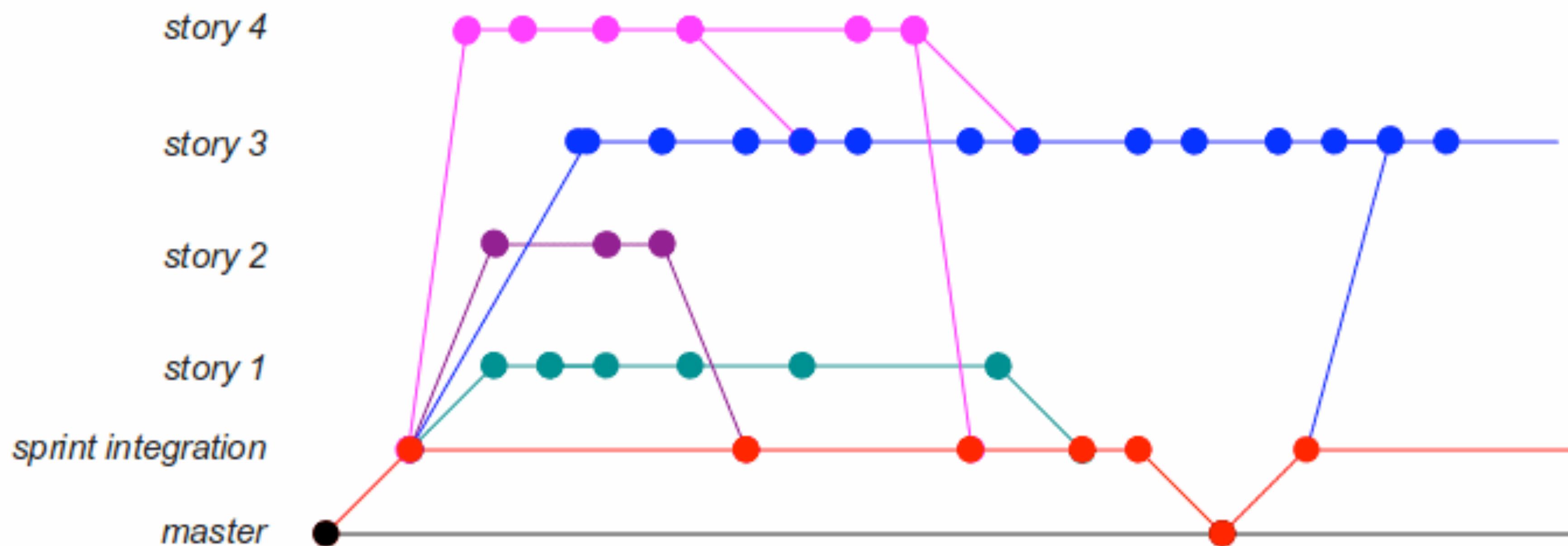
Hello Kansas City!!!

@ablythe

Points

1. Test locally matching production as much as possible

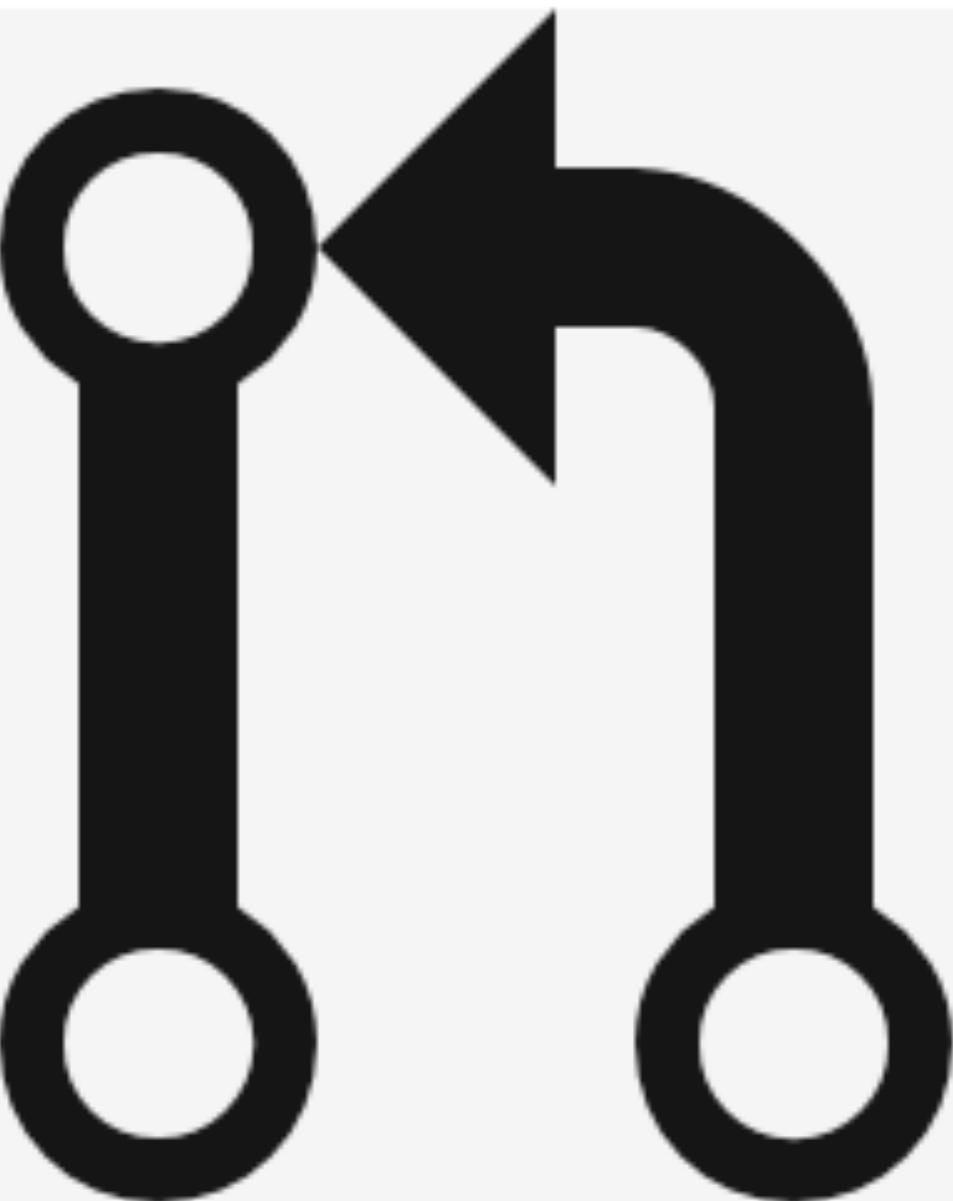
Check In on a Branch



Points

1. Test locally matching production as much as possible
2. Everything in Source Control

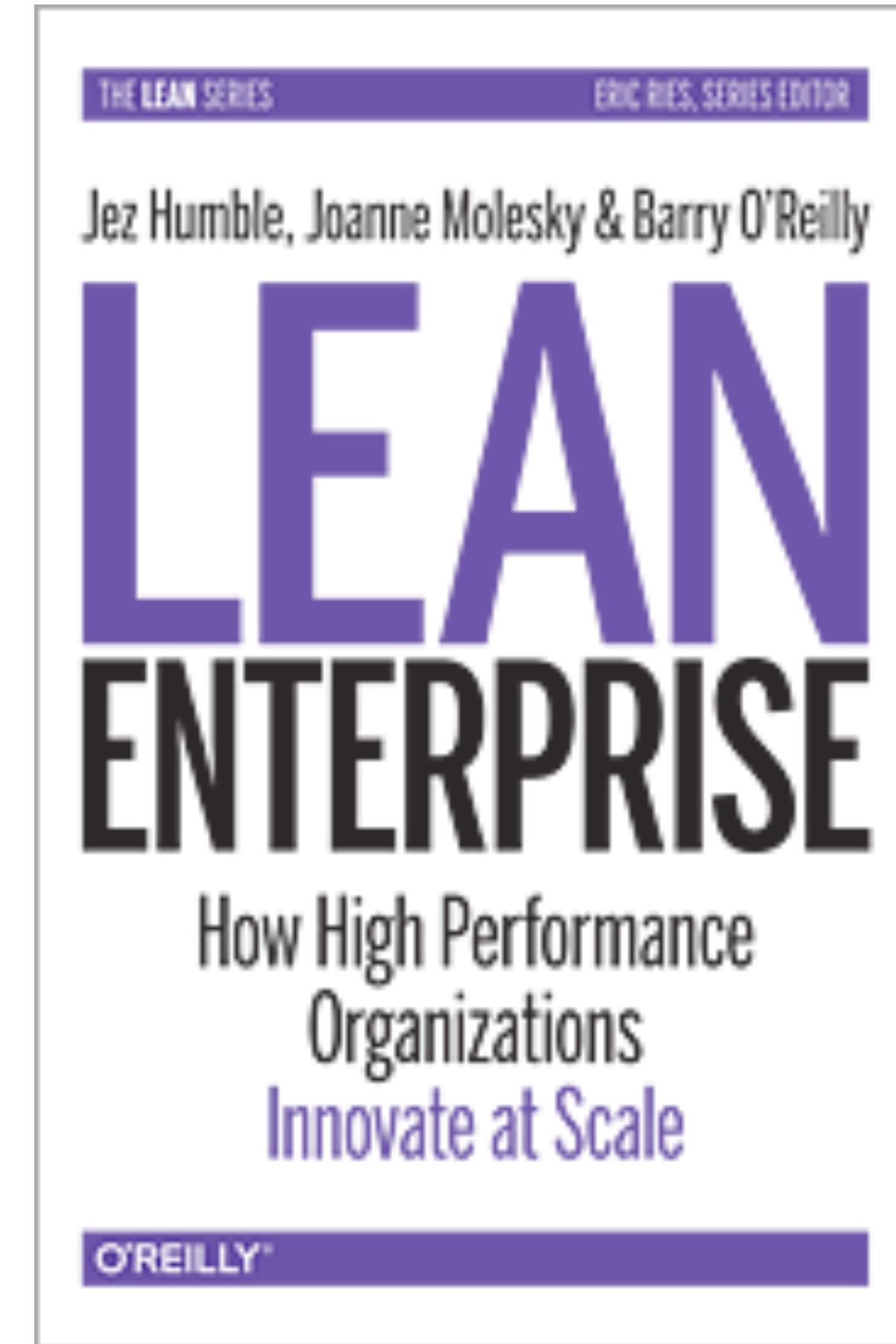
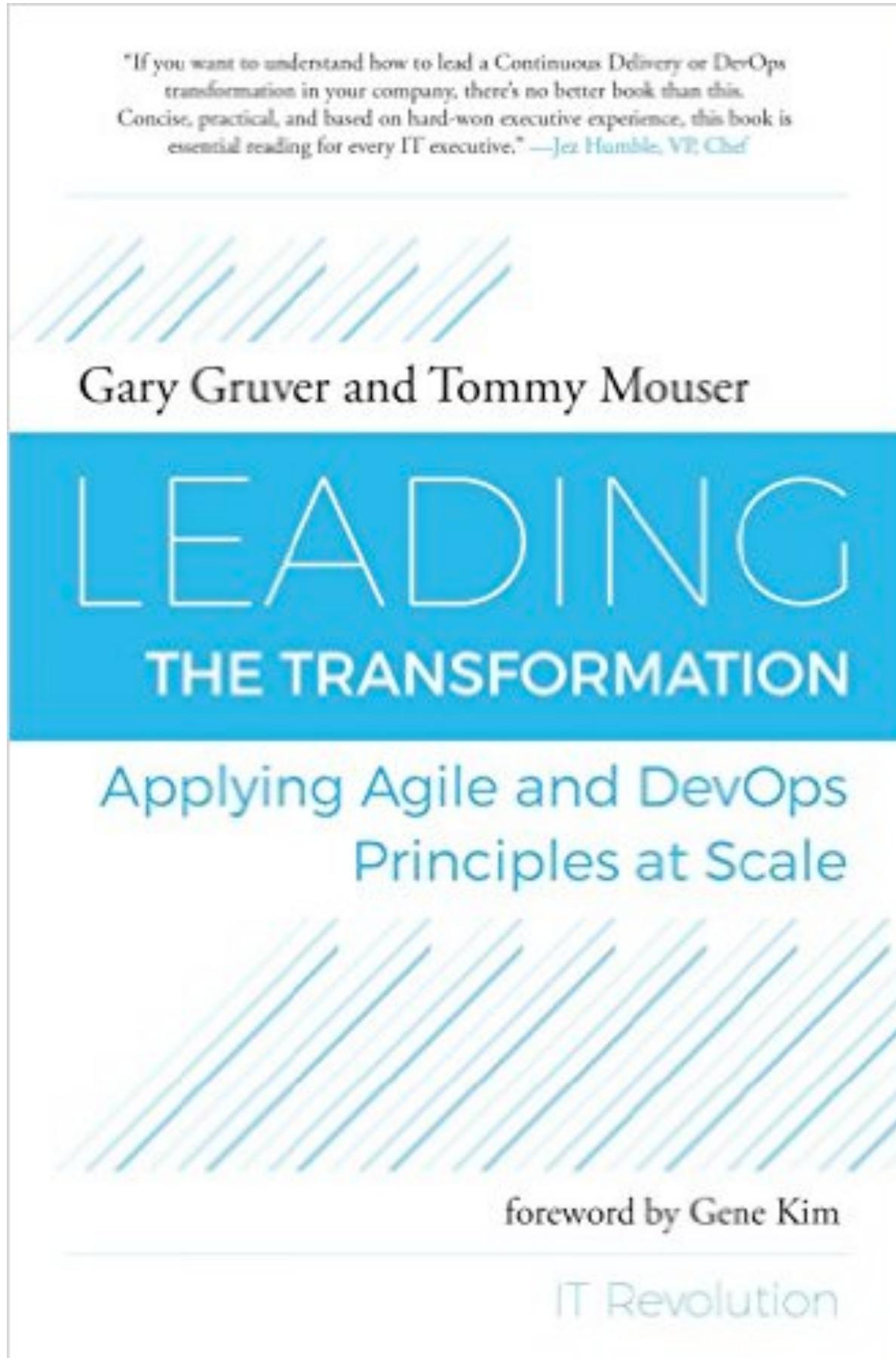
Create Pull Request



@ablythe

Points

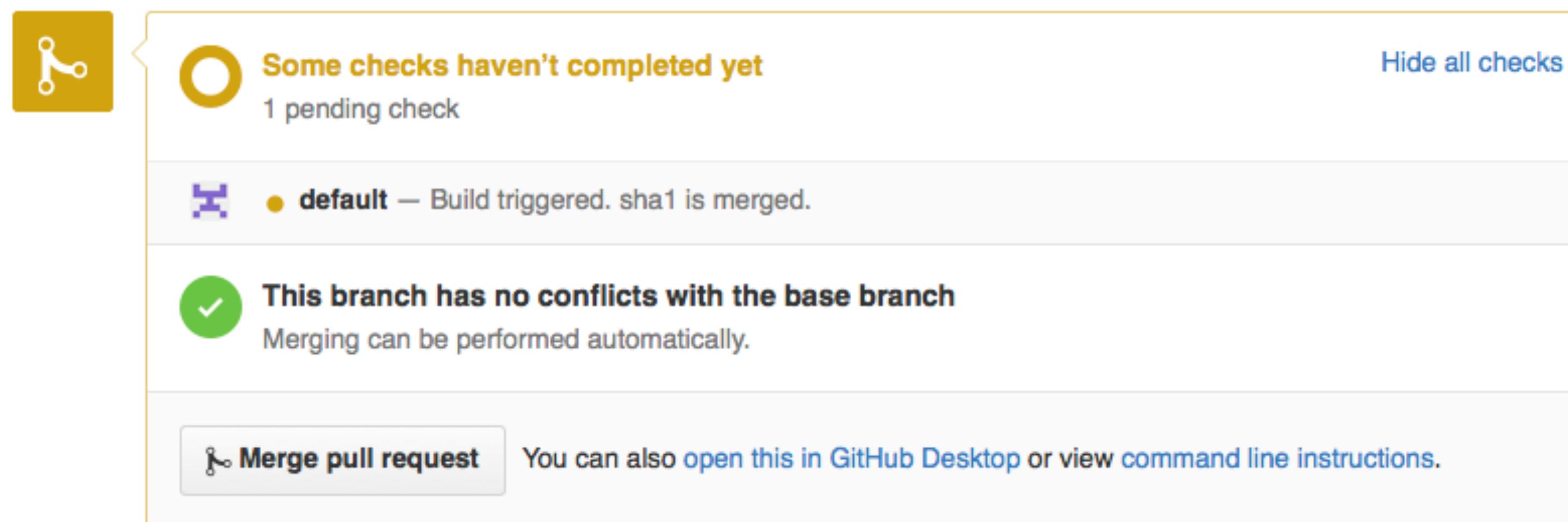
1. Test locally matching production as much as possible
2. Everything in Source Control
3. Agree as a team on Branching standards



<https://medium.com/@aaronblythe/gary-gruver-s-story-at-hp-and-macy-s-5ee89d4bcd9#.yc1ecomxp>

@ablythe

Jenkins Integration to Github



@ablythe

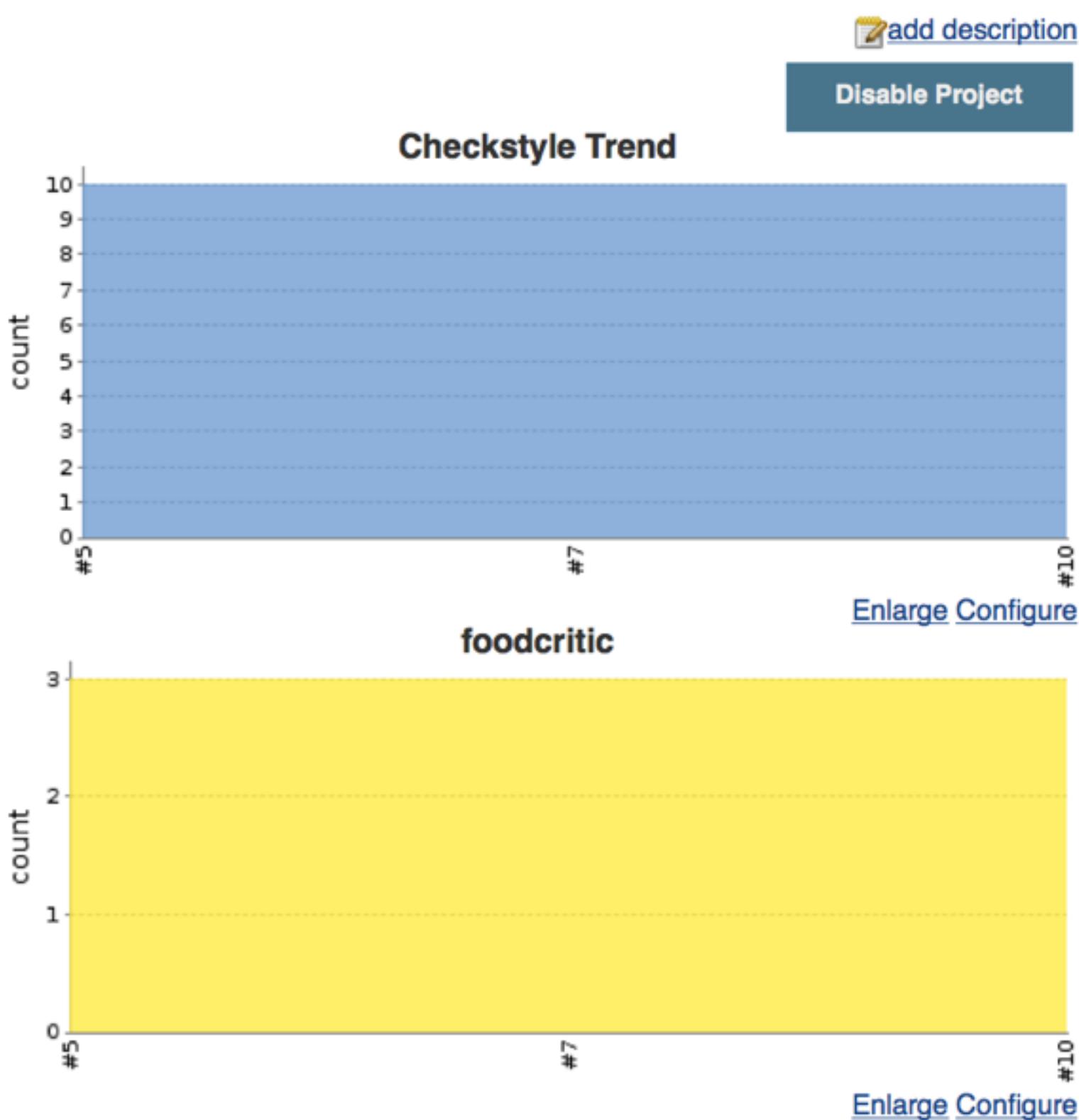
Starting our pipeline from the Pull Request



Build History		
	#6	Apr 24, 2016 9:27 PM
	#5	Apr 24, 2016 4:41 PM
	#4	Apr 24, 2016 4:41 PM

@ablythe

Static Analysis (Linting)



@ablythe

DooD (not to be confused with DinD)

(Trick: mount docker socket)

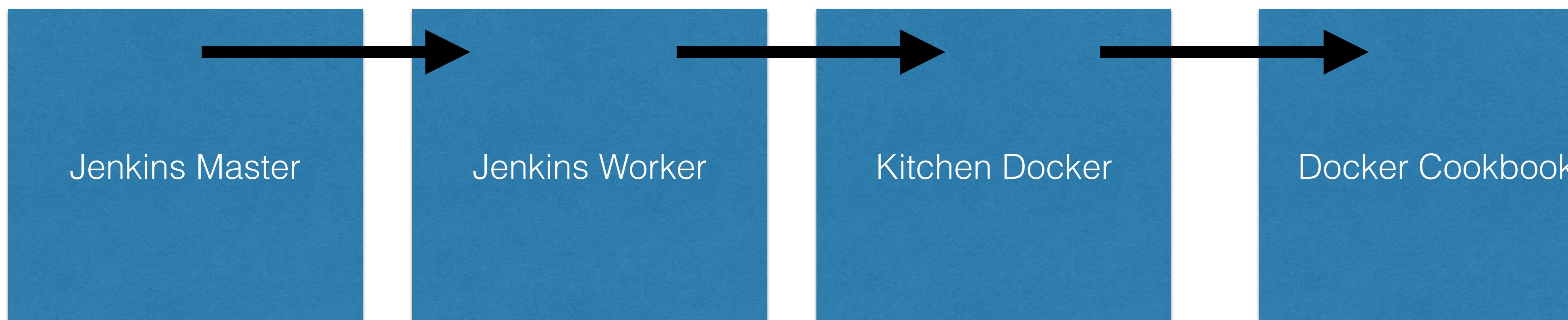
Kitchen.yml

Docker Cloud
Configuring IP

Volumes

```
/var/run/docker.sock:/var/run/docker.sock  
/home/ubuntu/.ssh:/home/jenkins/.ssh/
```

```
---  
driver:  
  name: docker  
  volume: ['/var/run/docker.sock:/var/run/docker.sock']  
  privileged: true  
  forward: 80
```

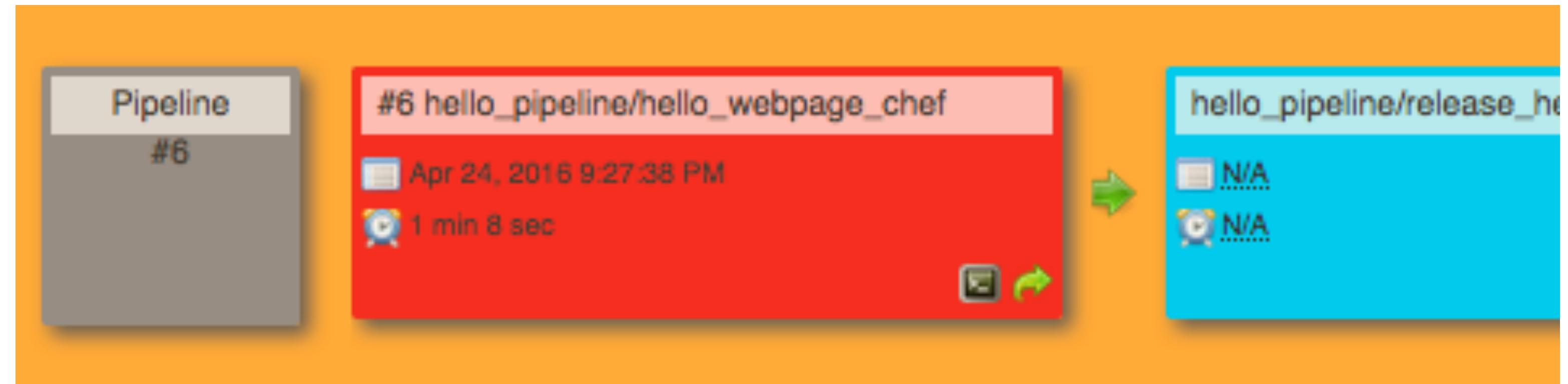
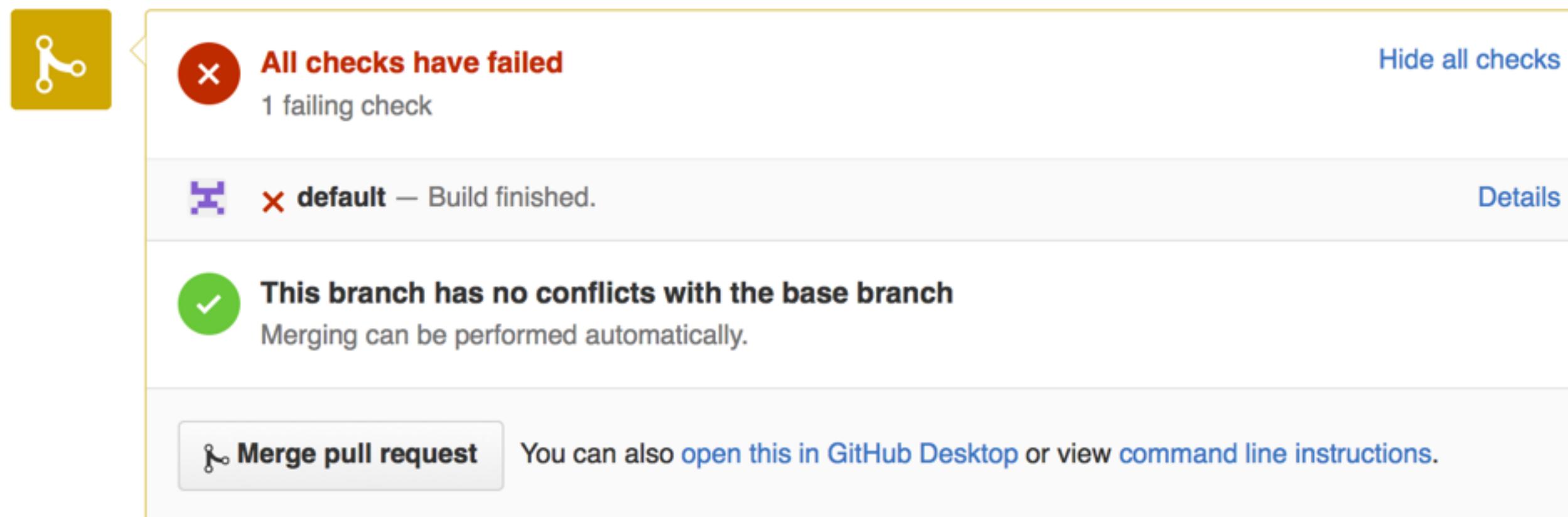


Docker Host

@ablythe

Failed Build!!

Add more commits by pushing to the `test1` branch on [aaronblythe/hello_webpage_chef](#).



@ablythe

Check the logs of the test

```
Search `/var/lib/jenkins/workspace/hello_pipeline/hello_webpage_chef/test/integration/default/inspec` for tests
.....F

Failures:

1) File `/var/www/html/index.html` content should match `/Champaign/
Failure/Error: DEFAULT_FAILURE_NOTIFIER = lambda { |failure, _opts| raise failure }

expected "<!DOCTYPE html>\n<html>\n<body>\n\n<h1>Hello Kansas City!!!</h1>\n</body>\n</html>\n" to match `/Champaign/
Diff:
@@ -1,2 +1,9 @@
-/Champaign/
+<!DOCTYPE html>
```

The build should automatically restart

Add more commits by pushing to the **test1** branch on **aaronblythe/hello_webpage_chef**.

 Some checks haven't completed yet Hide all checks

 ● **default** — Build started sha1 is merged. Details

 **This branch has no conflicts with the base branch**
Merging can be performed automatically.

 **Merge pull request** You can also [open this in GitHub Desktop](#) or [view command line instructions](#).

@ablythe

Points

1. Test locally matching production as much as possible
2. Everything in Source Control
3. Agree as a team on Branching standards
4. Have good tests and coverage

Restart From Next Push to branch



Points

1. Test locally matching production as much as possible
2. Everything in Source Control
3. Agree as a team on Branching standards
4. Have good tests and coverage
5. Fix the build right away

All possible interested users should know

[HearstAI/cloudfformation-chet-haj] Issue closed: #3 Add missing param export IAM_ROLE by TheFynx

jenkins BOT 11:14 PM

Hello Pipeline » hello_webpage_chef - #11 GitHub pull request #8 of commit e24735955cf39aa3bf1123e08050b8b78cdde239, no merge conflicts. ([Open](#))

Hello Pipeline » hello_webpage_chef - #11 Failure after 1 min 10 sec ([Open](#))

Hello Pipeline » hello_webpage_chef - #12 GitHub pull request #8 of commit d52920968b9b014ea2d7595ee4e6126f3eb62d0d, no merge conflicts. ([Open](#))

Hello Pipeline » hello_webpage_chef - #12 Back to normal after 1 hr 10 min ([Open](#))

Hello Pipeline » release_hello_webpage_chef - #25 Started by user Aaron Blythe ([Open](#))

Hello Pipeline » release_hello_webpage_chef - #25 Success after 17 sec ([Open](#))

Hello Pipeline » deploy_staging_hello_webpage_chef - #5 Started by upstream project "hello_pipeline/release_hello_webpage_chef" build number 25 ([Open](#))

Hello Pipeline » deploy_staging_hello_webpage_chef - #5 Success after 12 sec ([Open](#))

Hello Pipeline » deploy_production_hello_webpage_chef - #4 Started by user Aaron Blythe ([Open](#))

Hello Pipeline » deploy_production_hello_webpage_chef - #4 Success after 12 sec ([Open](#))

@ablythe

Points

1. Test locally matching production as much as possible
2. Everything in Source Control
3. Agree as a team on Branching standards
4. Have good tests and coverage
5. Fix the build right away
6. Make things visible

Merge Pull Request

The image is a composite of several screenshots from the GitHub interface, illustrating the process of merging a pull request:

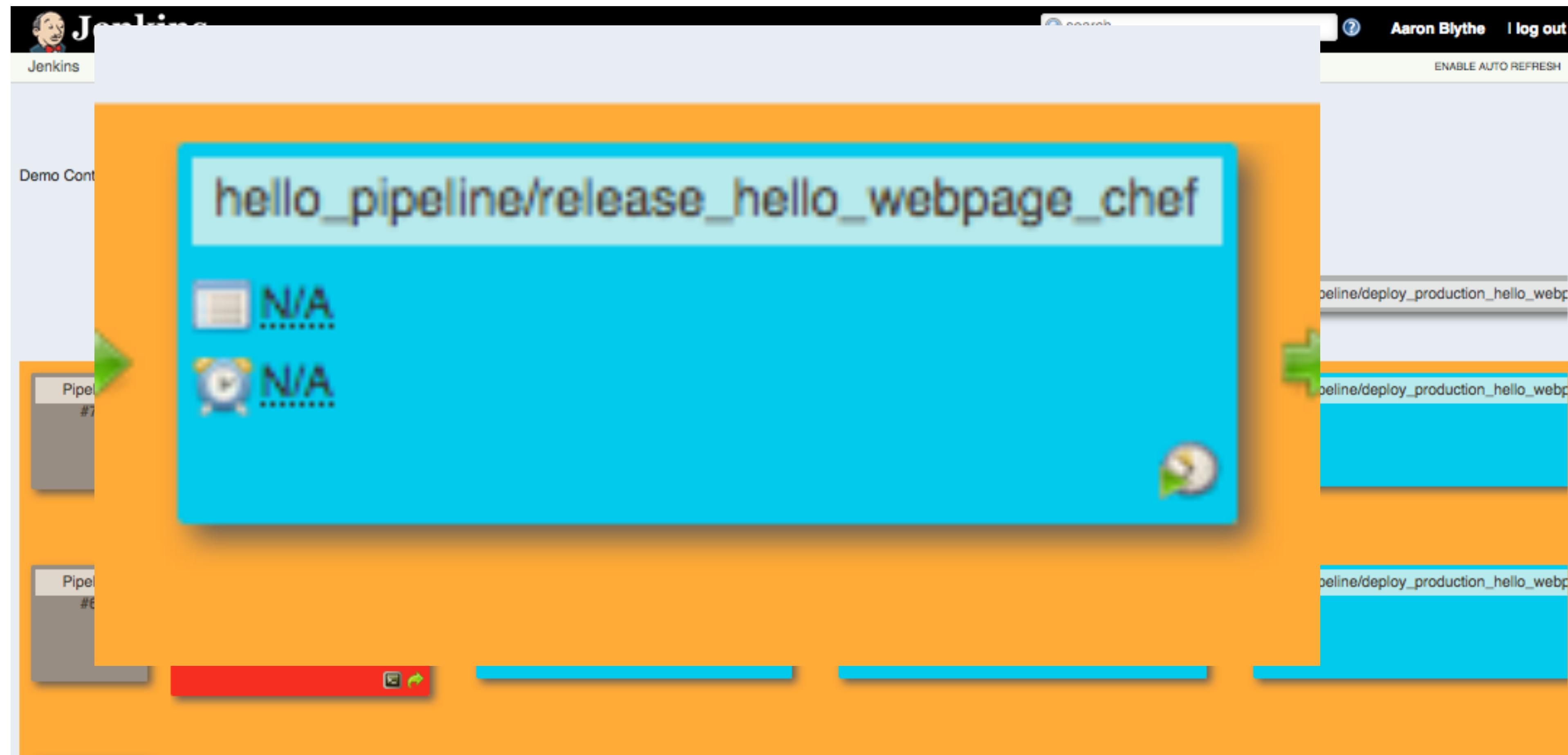
- Top Left:** Shows the 'Write' tab of a pull request. The user's profile picture is at the top left. Below it are two buttons: 'Write' (selected) and 'Preview'. Underneath the buttons are two emojis: a thumbs-up and a squirrel wearing a hat.
- Top Right:** A preview of the merge commit. It shows a profile picture of the user, the title 'Merge pull request #7 from aaronblythe/...', and the commit message 'Updating city to Kansas City'.
- Bottom Left:** A summary of the merge conditions. It includes:
 - A green circular icon with a checkmark and a gear symbol, followed by the text 'All checks have passed' and '1 successful check'.
 - A green circular icon with a checkmark and a gear symbol, followed by the text 'This branch has no conflicts with the base branch' and 'Merging can be performed automatically.'
- Bottom Right:** The final merge history. It shows a purple circular icon with a gear symbol, followed by the text 'aaronblythe merged commit 84251c3 into master' and '1 check passed'.
- Bottom Center:** A summary of the merge result. It shows a purple circular icon with a gear symbol, followed by the text 'Pull request successfully merged and closed' and 'You're all set—the test1 branch can be safely deleted'.

@ablythe

Go here:

- <http://stage-deploy-demo.hearst.at/>

Move On With the Pipeline



Tag in Github and update necessary version identifiers

Commits on Apr 24, 2016

Bumping version to 0.7.0
jenkins committed an hour ago

Merge pull request #6 from aaronblythe/test1 ...
aaronblythe committed an hour ago

Bumping version to 0.7.0
master 0.7.0
jenkins committed an hour ago

Showing 1 changed file with 1 addition and 1 deletion.

2 metadata.rb

		@@ -4,7 +4,7 @@
4	4	license 'all_rights'
5	5	description 'Installs/Configures hello_w'
6	6	long_description 'Installs/Configures he'
7	7	-version '0.6.0'
		+version '0.7.0'
8	8	
9	9	depends 'apache2'
10	10	depends 'apt'

19 commits

Branch: master New pull request N

Switch branches/tags

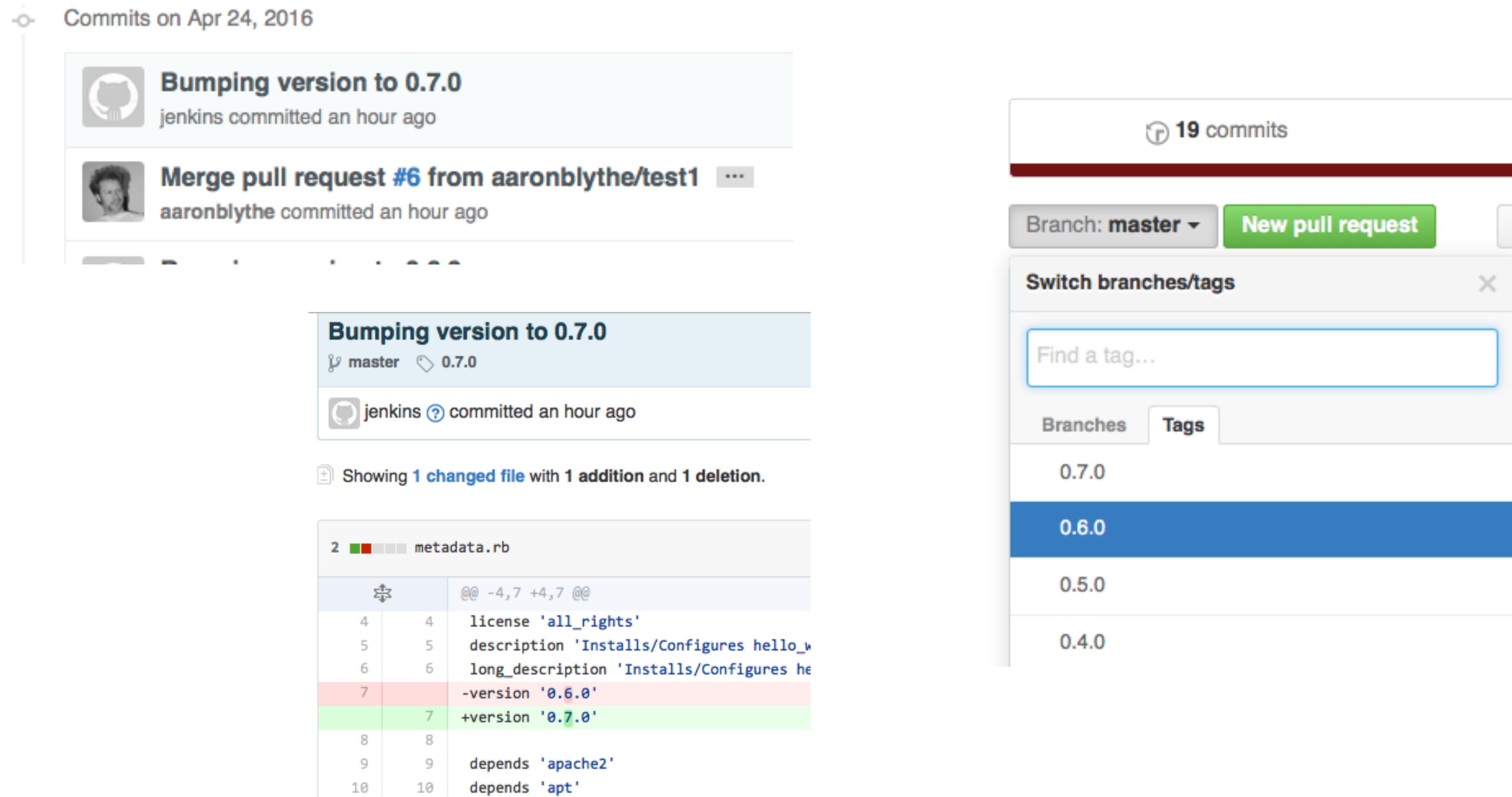
Find a tag... Tags

0.7.0

0.6.0

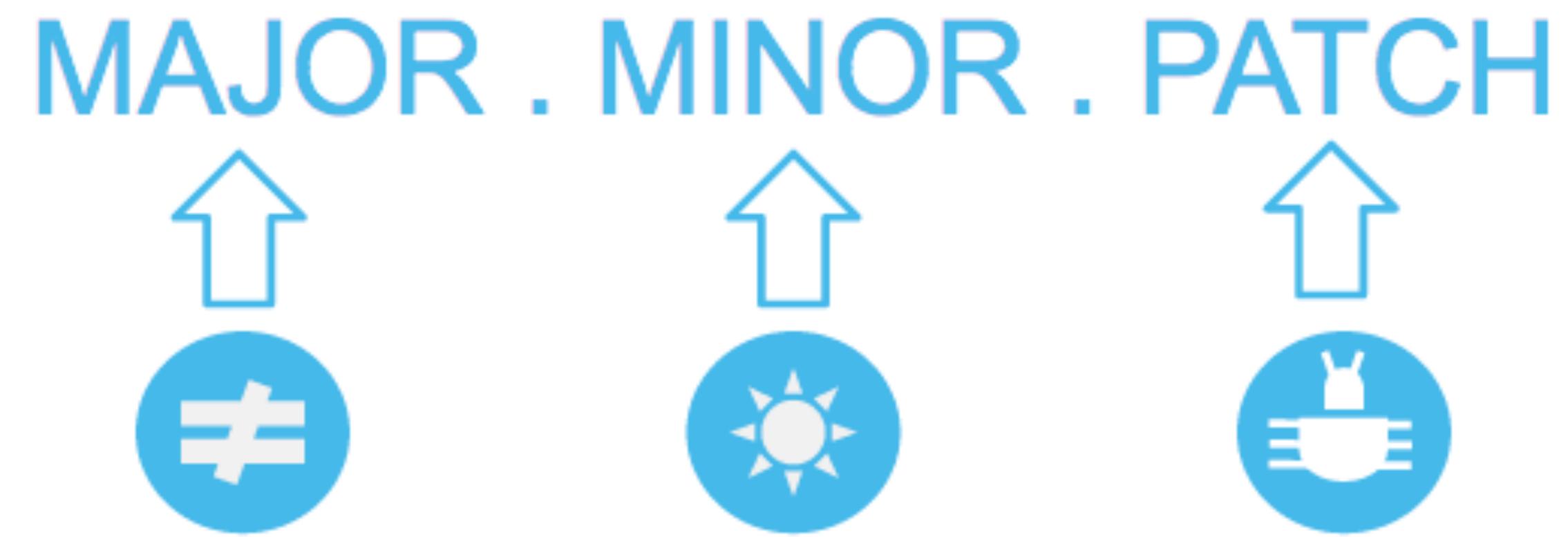
0.5.0

0.4.0



@ablythe

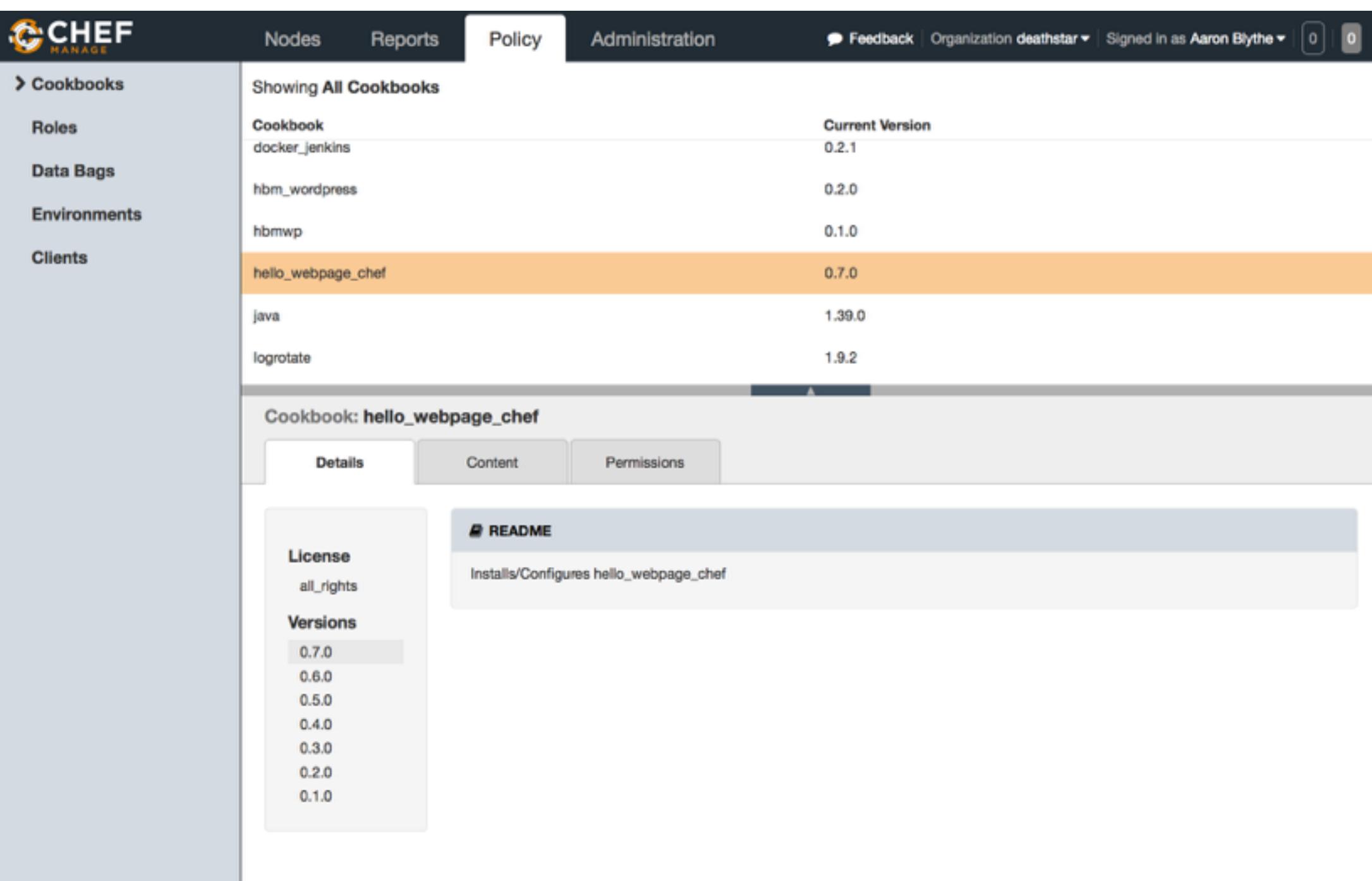
Semantic Versioning



<http://semver.org/>

@ablythe

Release to Artifact Repository



@ablythe

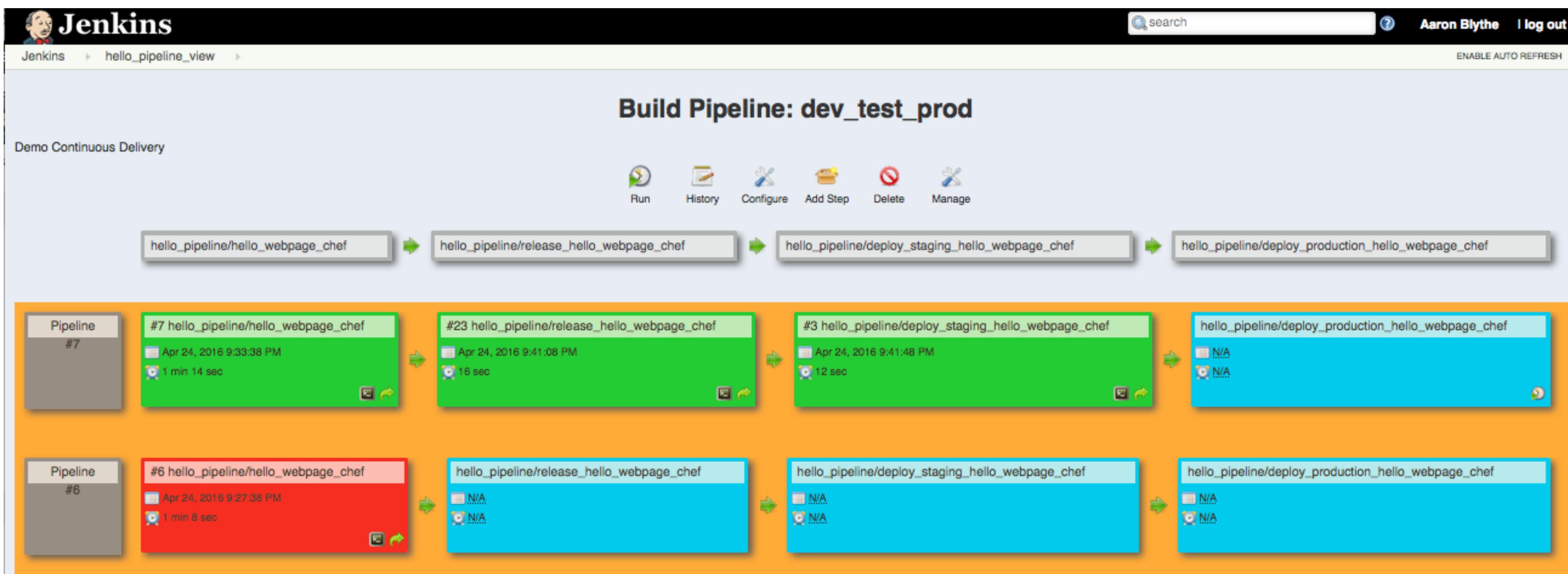
Points

1. Test locally matching production as much as possible
2. Everything in Source Control
3. Agree as a team on Branching standards
4. Have good tests and coverage
5. Fix the build right away
6. Make things visible
7. Automate your release process

Promise Theory

- Voluntary cooperation between individual, **autonomous** actors or agents who publish their intentions to one another in the form of **Promises**.
- Quite different than **Obligations** that we often deal with.

Deploy to Staging



@ablythe

Points

1. Test locally matching production as much as possible
2. Everything in Source Control
3. Agree as a team on Branching standards
4. Have good tests and coverage
5. Fix the build right away
6. Make things visible
7. Automate your release process
8. Stage exactly like production

Configuration Management

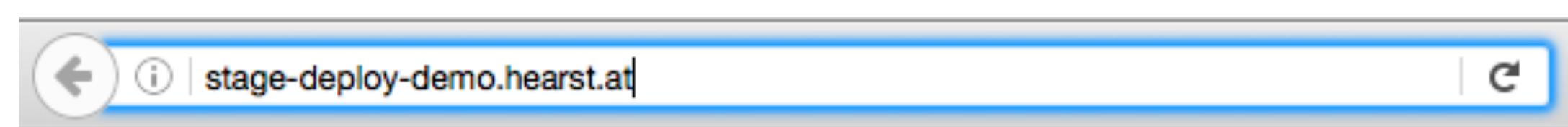
- Idempotent (e.g. can run repeatedly and achieve the same result, PUT in HTTP is Idempotent)
- No-op when already done before

Chef

```
ec2-52-18-190-186.eu-west-1.compute.amazonaws.com Recipe: hello_webpage_chef::default[0m
ec2-52-18-190-186.eu-west-1.compute.amazonaws.com   * template[/etc/apache2/sites-available/my_app.conf] action create (up to date)
ec2-52-18-190-186.eu-west-1.compute.amazonaws.com   * execute[a2ensite my_app.conf] action run (skipped due to not_if)
ec2-52-18-190-186.eu-west-1.compute.amazonaws.com   * template[/var/www/html/index.html] action create
ec2-52-18-190-186.eu-west-1.compute.amazonaws.com     [32m- update content in file /var/www/html/index.html from 43fc0b to 6f5737[0m
ec2-52-18-190-186.eu-west-1.compute.amazonaws.com     [37m--- /var/www/html/index.html 2016-04-24 16:45:47.917260392 +0000[0m
ec2-52-18-190-186.eu-west-1.compute.amazonaws.com     [37m+++ /var/www/html/.chef-index.html20160424-4957-u7kpf 2016-04-24 21:40:51.861260392 +0000[0m
ec2-52-18-190-186.eu-west-1.compute.amazonaws.com     [37m@@ -2,7 +2,7 @@[0m
ec2-52-18-190-186.eu-west-1.compute.amazonaws.com     [37m <html>[0m
ec2-52-18-190-186.eu-west-1.compute.amazonaws.com     [37m <body>[0m
ec2-52-18-190-186.eu-west-1.compute.amazonaws.com     [37m [0m
ec2-52-18-190-186.eu-west-1.compute.amazonaws.com     [37m-<h1>Hello Kansas City!!!</h1>[0m
ec2-52-18-190-186.eu-west-1.compute.amazonaws.com     [37m+<h1>Hello Champaign!!!</h1>[0m
ec2-52-18-190-186.eu-west-1.compute.amazonaws.com     [37m [0m
ec2-52-18-190-186.eu-west-1.compute.amazonaws.com     [37m </body>[0m
ec2-52-18-190-186.eu-west-1.compute.amazonaws.com     [37m </html>[0m
ec2-52-18-190-186.eu-west-1.compute.amazonaws.com   [0m]0m
```

```
ec2-52-18-190-186.eu-west-1.compute.amazonaws.com Running handlers:[0m
ec2-52-18-190-186.eu-west-1.compute.amazonaws.com Running handlers complete
ec2-52-18-190-186.eu-west-1.compute.amazonaws.com [0mChef Client finished, 1/111 resources updated in 06 seconds[0m
Started calculate disk usage of build
Finished Calculation of disk usage of build in 0 seconds
```

Deploy to Staging



Hello Champaign!!!

@ablythe

Perceptual Diffs

 **CD Example**

Release: Hello City #3 **Status:** Reviewing

Test: My homepage Approved now by superuser@example.com

✓ Diff found and approved

Reject difference

Before	Diff	Log	After
Hello Kansas City!!!	Champaign!!!		Hello Champaign!!!

Visit – Config – Log Visit – Config – Log

superuser@example.com – [Sign out](#)

dpxdt

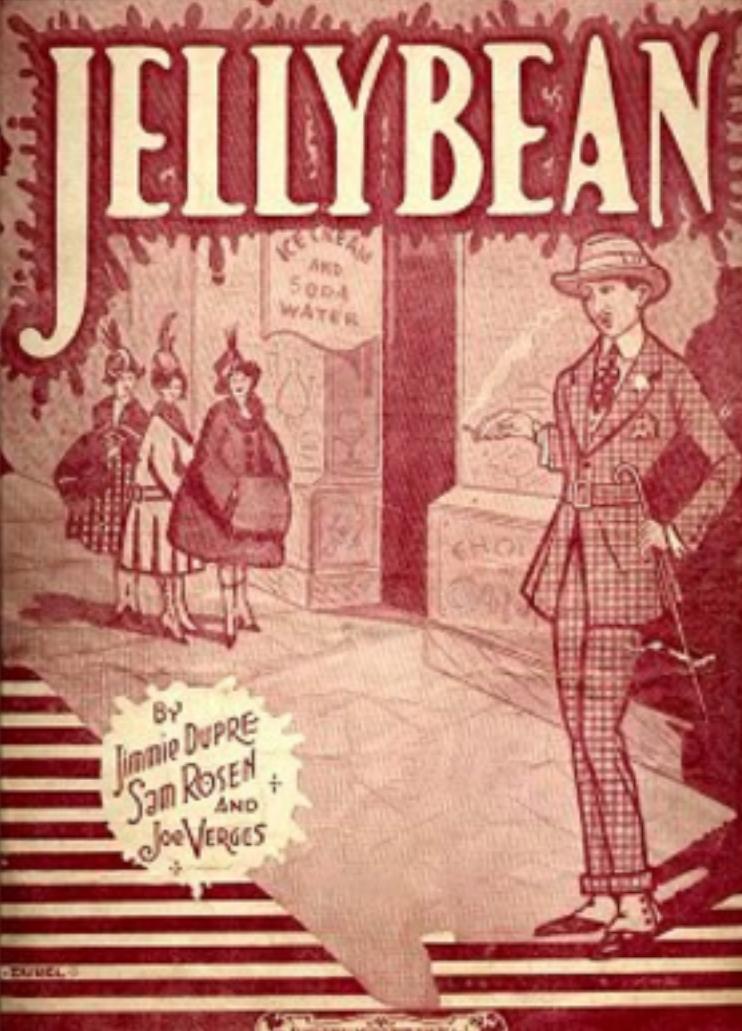
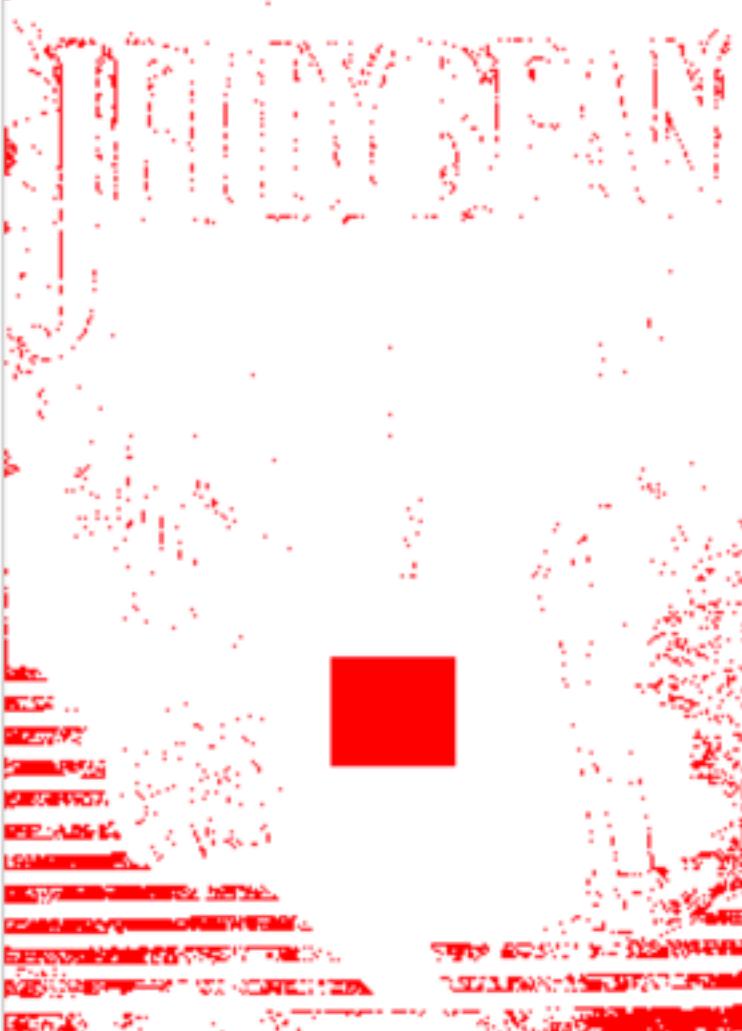
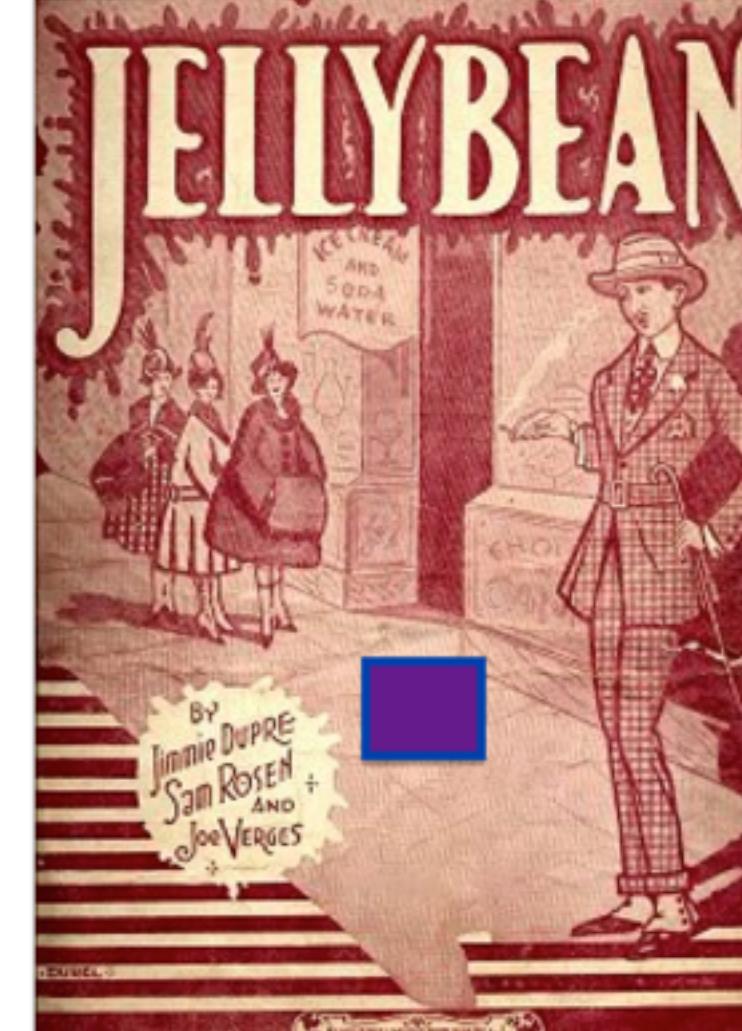
 **CD Example**

Release: Awesome #2 **Status:** Reviewing

Test: My homepage Updated 55 minutes ago

X Diff found: 5.9%

[Approve difference](#) [Next test →](#)

Before	Diff	After
		

[Visit Log](#) [Log](#) [Visit Log](#)

@ablythe

<https://github.com/bslatkin/dpxdt>

 **CD Example**

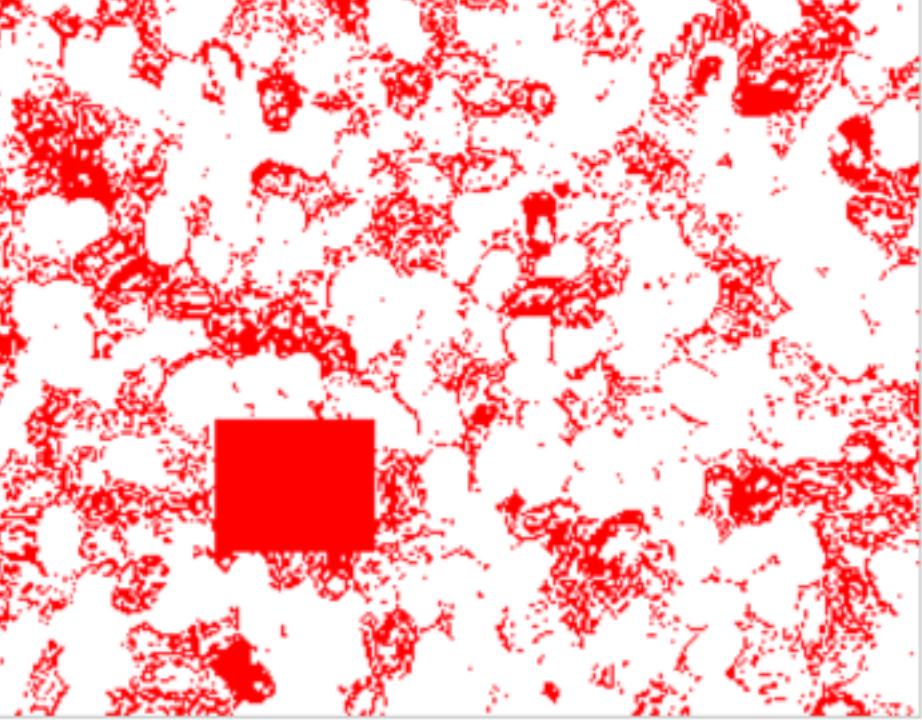
Release: Awesome #2 **Status:** Reviewing

Test: My other page **Updated** 55 minutes ago

X Diff found: 7.3%

[Approve difference](#) [← Prev test](#)

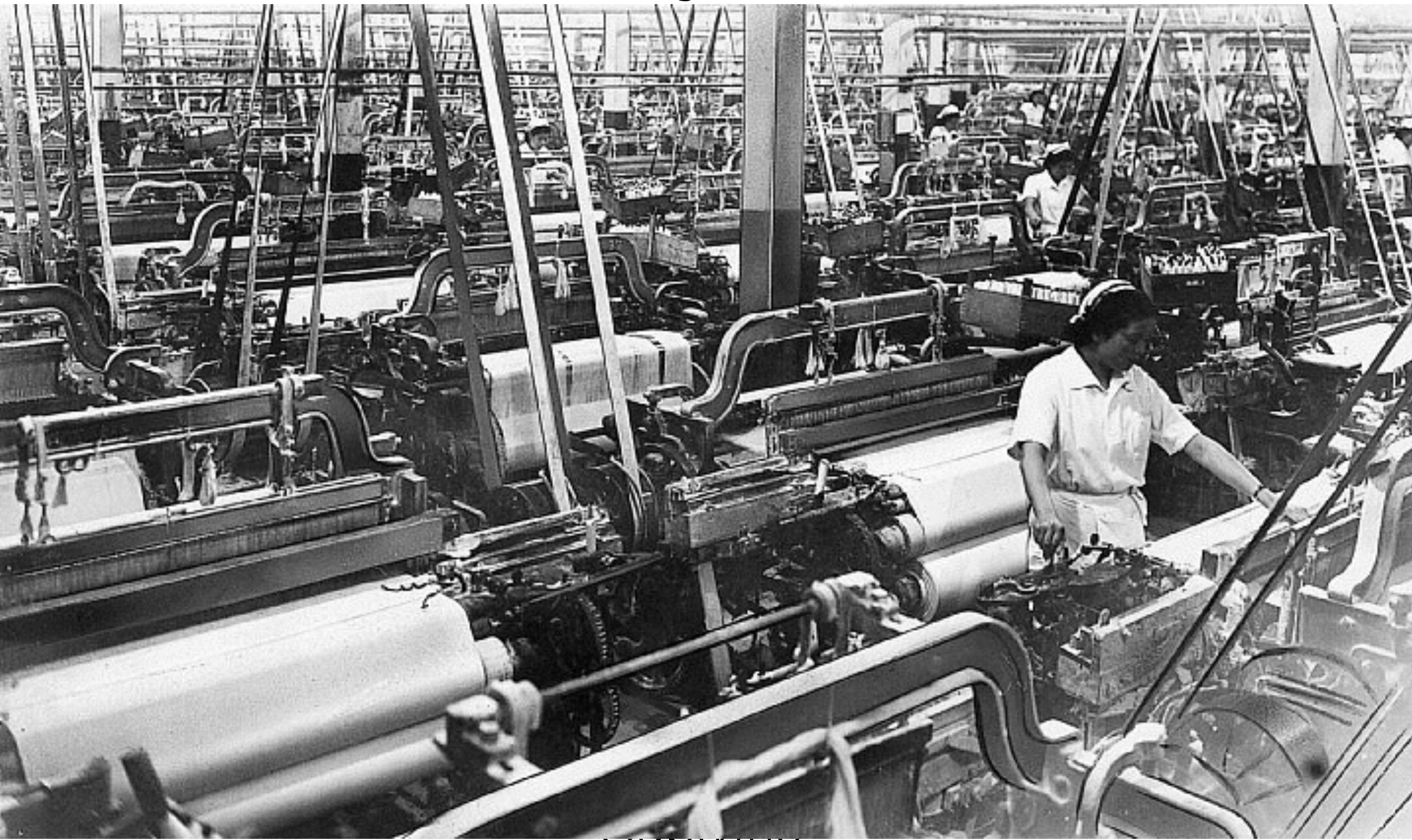
Before [Visit](#) **Diff** [Log](#) [After](#) [Visit](#)



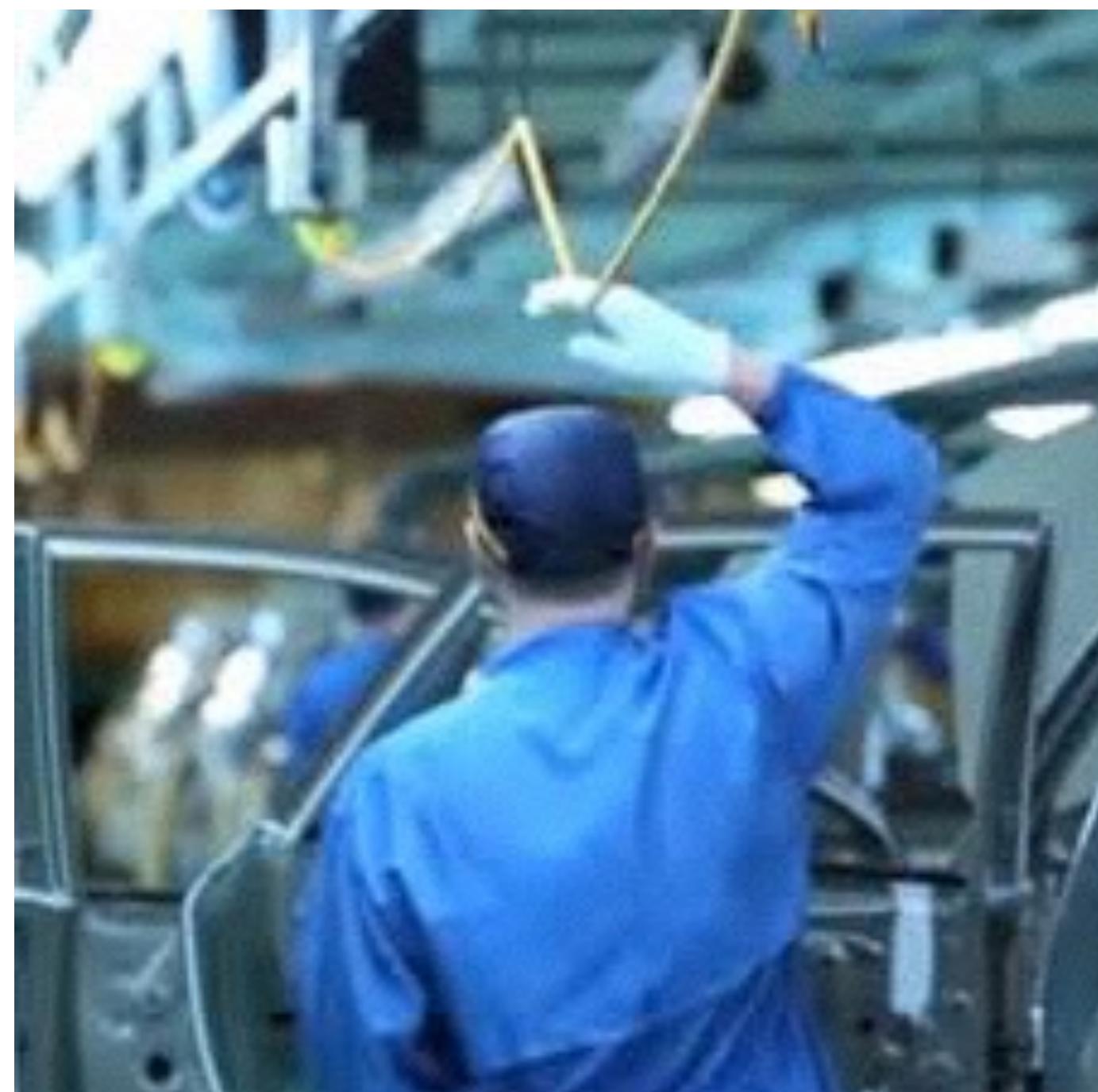
Points

1. Test locally matching production as much as possible
2. Everything in Source Control
3. Agree as a team on Branching standards
4. Have good tests and coverage
5. Fix the build right away
6. Make things visible
7. Automate your release process
8. Stage exactly like production
9. Use perceptual diffs

Toyoda



Andon Cord



@ablythe

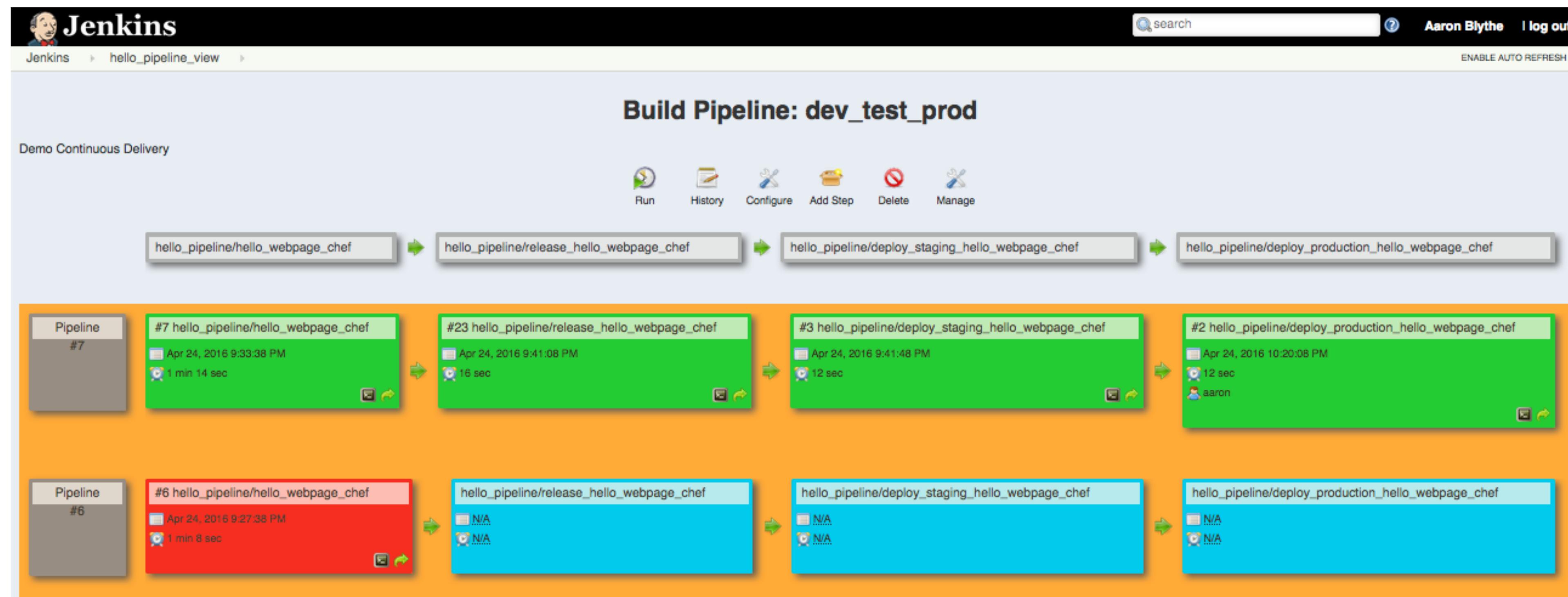
Points

1. Test locally matching production as much as possible
2. Everything in Source Control
3. Agree as a team on Branching standards
4. Have good tests and coverage
5. Fix the build right away
6. Make things visible
7. Automate your release process
8. Stage exactly like production
9. Use perceptual diffs
10. Trust and value your people

Deploy to Production

- <http://deploy-demo.hearst.at/>

Deploy to Production



@ablythe

Deploy to Production



Hello Kansas City!!!



Hello Champaign!!!

Recap

1. Test locally matching production as much as possible
2. Everything in Source Control
3. Agree as a team on Branching standards
4. Have good tests and coverage
5. Fix the build right away
6. Make things visible
7. Automate your release process
8. Stage exactly like production
9. Use perceptual diffs
10. Trust and value your people

Scoped Out of This Talk

- It should always be cheaper to spin up a new environment than fix an existing one
- Blue-Green Deployments
- Orchestration of Multiple Nodes
- 12 Factor App
- Container Scheduling
- Heroku

Points

1. Test locally matching production as much as possible
2. Everything in Source Control
3. Agree as a team on Branching standards
4. Have good tests and coverage
5. Fix the build right away
6. Make things visible
7. Automate your release process
8. Stage exactly like production
9. Use perceptual diffs
10. Trust and value your people

Fremont Assembly Plant



@ablythe

NUMMI plant



<http://www.lean.org/shook/DisplayObject.cfm?o=1166>

@ablythe

Tesla Factory



@ablythe

high trust culture

Pathological (<i>power oriented</i>)	Bureaucratic (<i>rule oriented</i>)	Generative (<i>performance oriented</i>)
Low cooperation	Modest cooperation	High cooperation
Messengers shot	Messengers neglected	Messengers trained
Responsibilities shirked	Narrow responsibilities	Risks are shared
Bridging discouraged	Bridging tolerated	Bridging encouraged
Failure leads to scapegoating	Failure leads to justice	Failure leads to enquiry
Novelty crushed	Novelty leads to problems	Novelty implemented

Points

1. Test locally matching production as much as possible
2. Everything in Source Control
3. Agree as a team on Branching standards
4. Have good tests and coverage
5. Fix the build right away
6. Make things visible
7. Automate your release process
8. Stage exactly like production
9. Use perceptual diffs
10. Trust and value your people

@ablythe