Continuous Delivery of your infrastructure

Christophe Vanlancker
@Carroarmato0

Christophe Vanlancker

- Internal operations and consulting
- Mentor
- Kris couldn’t make it so I ‘s/Kris/Christophe/g’

Todays Goals

- A reproducable way to deploy and upgrade /etc/puppet
- With 10+ environments in /etc/puppet/environments
- Automatically
- Fast
- Consistent
- Continuously

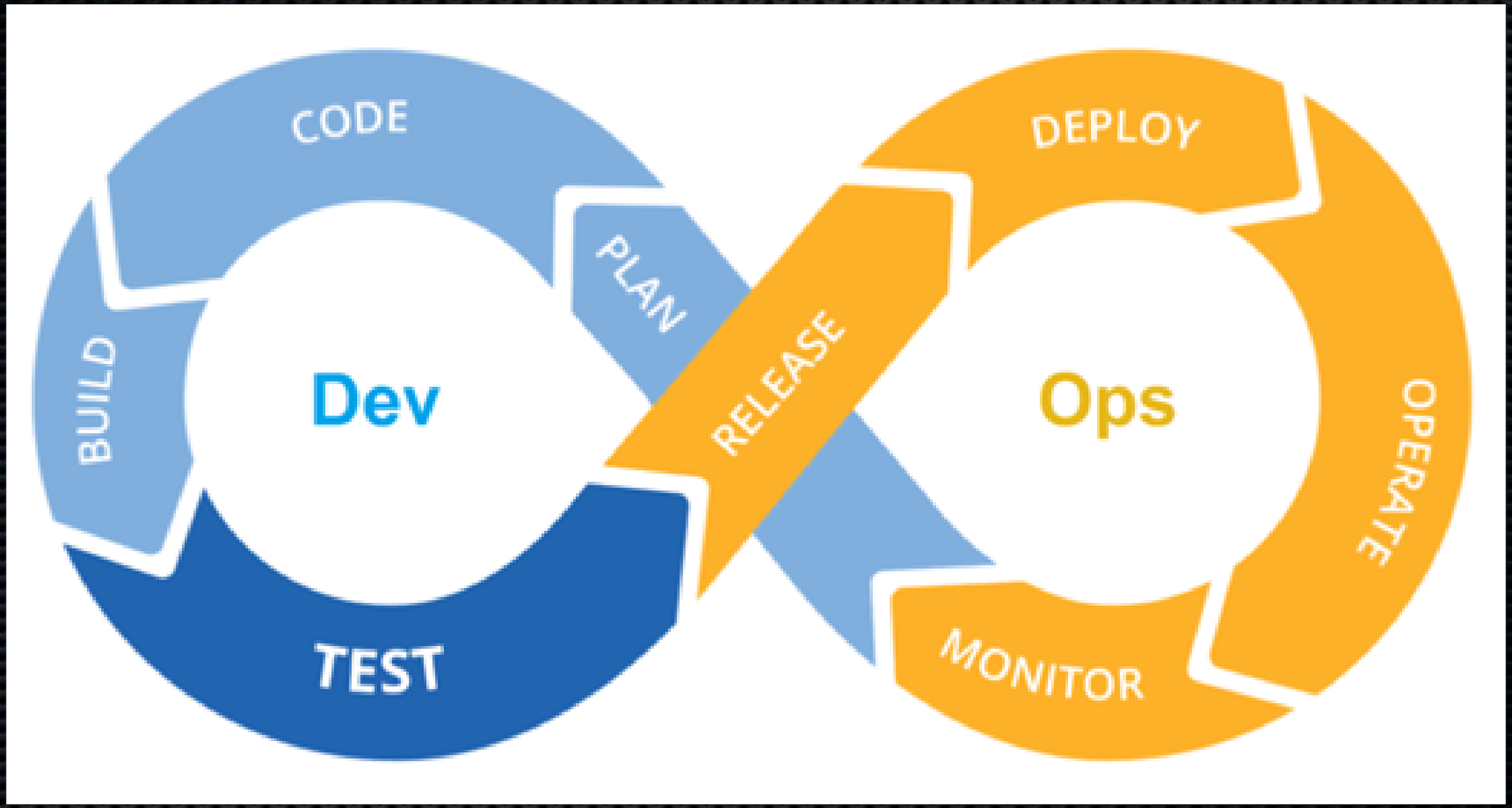
What's this devops thing anyhow ?

C(L)AMS

- Culture
- (Lean)
- Automation
- Measurement
- Sharing

Damon Edwards and John Willis

Gene Kim


CODE
PLAN
BUILD
Dev
TEST
DEPLOY
RELEASE
Ops
OPERATE
MONITOR

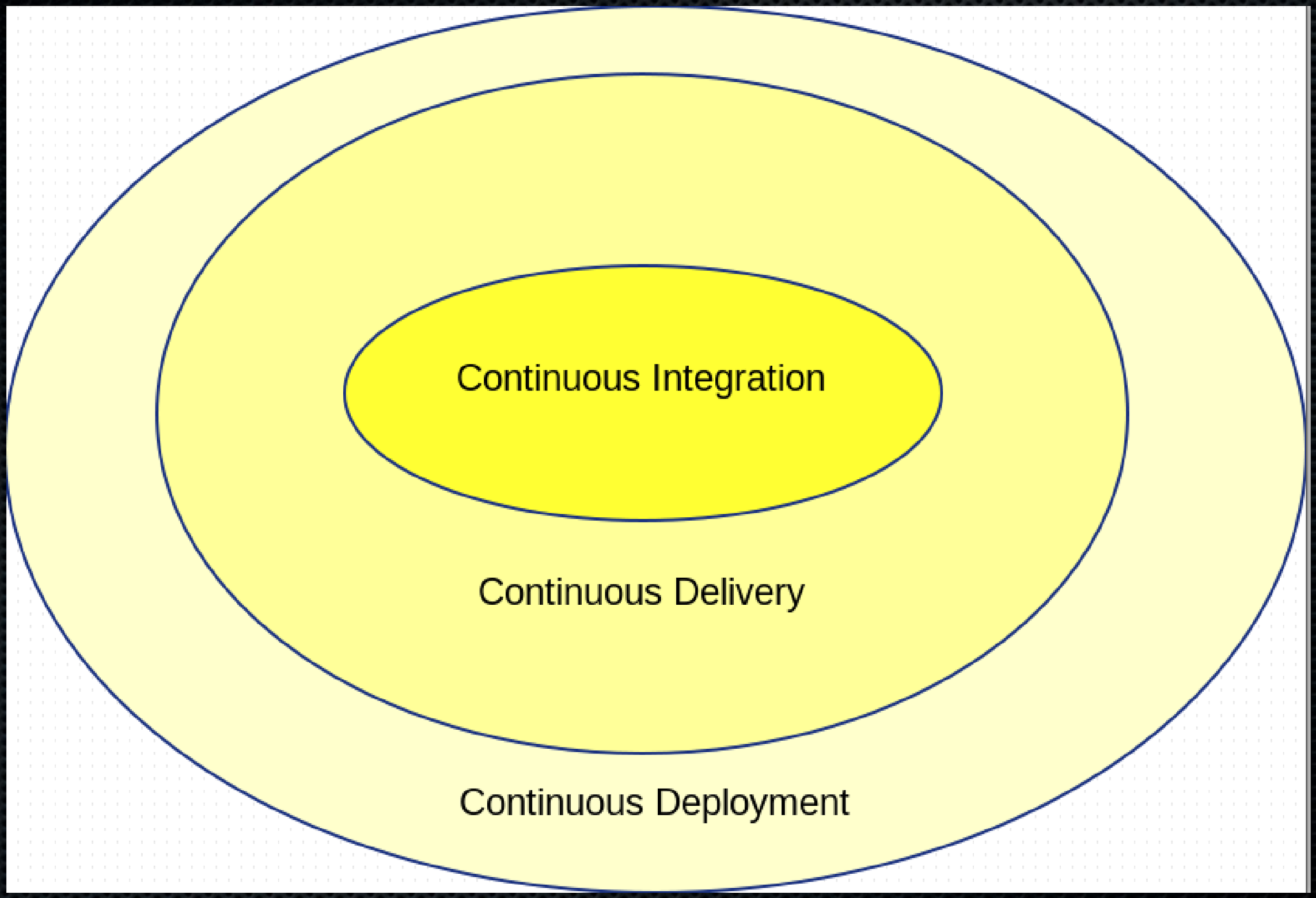
devops (<)> continuous delivery

Nirvana

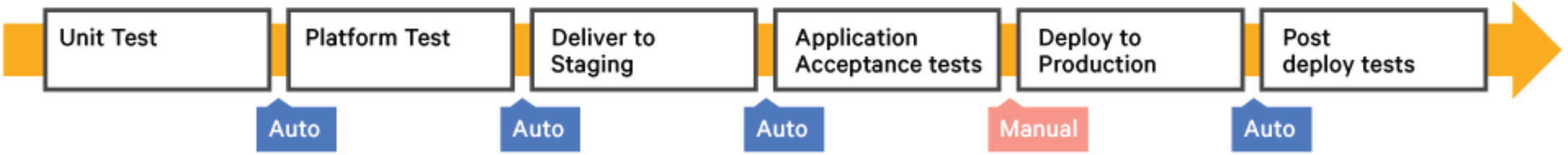
An “ecosystem” that supports continuous delivery, from infrastructure, data and configuration management to business.

Through automation of the build, deployment, and testing process, and improved collaboration between developers, testers, and operations, delivery teams can get changes released in a matter of hours — sometimes even minutes–no matter what the size of a project or the complexity of its code base.

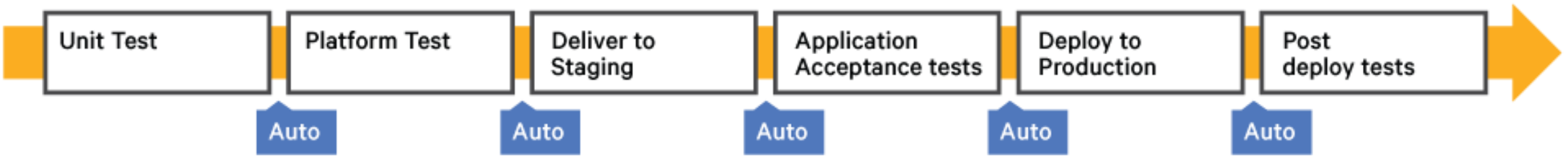
Continuous Delivery , Jez Humble


Continuous Integration
Continuous Delivery
Continuous Deployment

Continuous Delivery



Continuous Deployment



How many times a day ?

- 10 @ Flickr
- Deployments used to be pain
- Nobody dared to deploy a site
- Practice makes perfect
- Knowing you can vs constantly doing it

" Our job as engineers (and ops, dev-ops, QA, support, everyone in the company actually) is to enable the business goals. We strongly feeI that in order to do that you must have the ability to deploy code quickly and safely. Even if the business goals are to deploy strongly QA'd code once a month at 3am (it's not for us, we push all the time), having a reliable and easy deployment should be non-negotiable."

Etsy Blog upon releasing Deployinator

http://codeascraft.etsy.com/2010/05/20/quantum-of-deployment/

For years we've tolerated humans to make structural manual changes to the infrastructure our critical applications are running on.

Whilst at the same time demanding those critical applications to go through rigid test scenarios.

Who let this happen ?

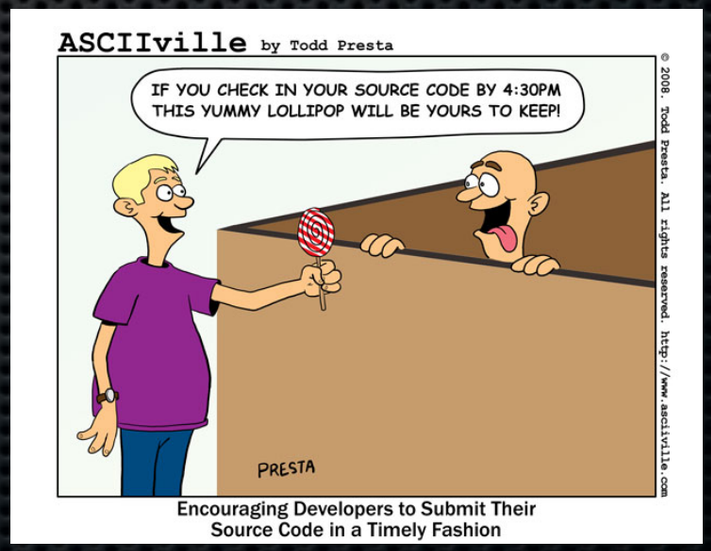
Infrastructure as Code

- Treat configuration automation as code
- Development best practices
  - Model your infrastructure
  - Version your cookbooks / manifests
  - Test your cookbooks/ manifests
  - Dev/ test /uat / prod for your infra
- Model your infrastructure
- A working service = automated (Application Code + Infrastructure Code + Security + Monitoring)

Version all the things

- Source code Application
- Source code Infrastructure
- Builds
- Tests
- Pipelines
- Scripts
- Documentation
- Monitoring scripts

No more excuses !



A random project

```
[sdog@mine vagrant-graphite]$ ls
manifests  modules  README  TODO  Vagrantfile
[sdog@mine vagrant-graphite]$ tree -dL 2
.
├── manifests
│   └── hosts
└── modules
    ├── apache
    ├── collectd
    ├── graphite
    ├── jmxtrans
    ├── logster
    ├── statsd
    └── tattle

10 directories
```

Unless you understand Git Submodules

- Basic git,
- No extra tools required

Integrates with other projects too.

(No need for *-librarian etc ..)

Librarian Puppet / r10k

- Insert ugly shell script
-
- Even with this in place .. people can still hack on the PuppetMaster

sdog@mine:/home/inuits/puppetmaster 67x42

```
 710caf0538914cf1b296d49d142b7e41b8b738d3 modules/activemq (remotes
/origin/HEAD)
 8d1f8ba010277f3c3605bdc06352e5c5137b2cbb modules/apache (0.12.6-11
4-g8d1f8ba)
 9cf1882b79a40792ecac7cd1aa66851045fc5888 modules/apc (0.3-5-g9cf18
82)
 258f2fab689a9698880a3bcf646d8a8f7e1fb889 modules/apt (heads/master
)
 54062f3dc53fc8571dc973ea109d8e60d1c0104a modules/augeas (heads/mas
ter)
 441e329126373dd32366b5be89ca4b313e35e58c modules/bacula (remotes/o
rigin/refactor)
 ca33540fc792a1cc8e233013789012226cd9f401 modules/bash (heads/maste
r)
 5a1df5dc66968a3b630aabb4bb34e68bd043f712 modules/bind (heads/maste
r)
 0b6fe2764caf7d34d2bf5bd8f0b3b7f3aa804499 modules/collectd (remotes
/origin/HEAD)
 e8e1282b6d367a202d2b63f967e4d2d484c16470 modules/concat (20100507-
64-ge8e1282)
 ba2d97b488a1646b0a570a45c1d730040391af0b modules/crond (heads/mast
er)
 ce88c1fb74d47d9d65a43ef5c326a474519bd80b modules/dell (heads/maste
r)
 ac82189a0ca4b045e0b476e06476da4c651e721a modules/deploy_and_graph
(remotes/origin/HEAD)
 ed842ece457d727d3655f004b556a1e1cbc6eb38 modules/drupal7 (remotes/
origin/HEAD)
 c46c4a4f9aca5f77d649020c7d5cbbe16e183af0 modules/ejabberd (heads/m
aster)
 62570ae5d3a575e14b50b223dca4da82df0ceaba modules/elasticsearch (re
motes/origin/HEAD)
 3ffc8df6cd79a9357bdae3b40b29c26a6caa5bb9 modules/fail2ban (3ffc8df
)
 fc67332c735e6f446a865289f827fc20ab79398b modules/ffmpeg (heads/mas
ter)
 441eeefb4c313033479765d1261152ee9d39f52b modules/filemapper (heads
/master)
 60cee13265ea1e1f8fb097bf136a22e40a7edd47 modules/firewall (remotes
/origin/inuits)
 b8454e0cae39626ff353d9a1c785adbebeb9b385 modules/foreman (remotes/
:
```

sdog@mine:/home/intelliplan/provision/puppet 69x42

```
mod 'haf/nginx',
  :git => 'https://github.com/haf/puppet-nginx.git',
  :ref => 'e6554069bbc97acaaac874c5964c648dd81a3b8d'

mod 'puppetlabs/ntp',
  :git => 'https://github.com/puppetlabs/puppetlabs-ntp.git',
  :ref => 'feb5d6e936c5301c1181e932419b15b51fff0e48'

mod 'puppetlabs/concat',
  :git => 'https://github.com/puppetlabs/puppetlabs-concat.git',
  :ref => '8695e7ff638d201dbcc64f3b7c45514ba224c27c'

mod 'puppetlabs/stdlib',
  :git => 'https://github.com/puppetlabs/puppetlabs-stdlib.git',
  :ref => '44c181ec0e230768b8dce10de57f9b32638e66e1'

mod 'stahnma/epel',
  :git => 'https://github.com/stahnma/puppet-module-epel.git',
  :ref => '449a63e089d051a0216649a7a53d042a0a39ea89'

mod 'puppetlabs/firewall',
  :git => 'https://github.com/puppetlabs/puppetlabs-firewall.git',
  :ref => '5e2b054c3bedfbfb5236c8d351e1b79a7494fca5'

mod 'haf/eventstore',
  :git => 'https://github.com/haf/puppet-eventstore.git',
  :ref => 'cd8b311fad7072825ee1a0a002004d1d7ffa16f3'

mod 'BashtonLtd/timezone',
  :git => 'https://github.com/BashtonLtd/puppet-timezone.git',
  :ref => '2cc4f54e6d23aa8071bf62fcd1af1b1852f399cd'

mod 'treydock/yum_cron',
  :git => 'https://github.com/treydock/puppet-yum_cron.git',
  :ref => 'e826aa5e35345e0230f1d013052d546b640a3eed'

# yum_cron depends on
mod 'hercules-team/augeasproviders',
  :git => 'https://github.com/hercules-team/augeasproviders.git',
  :ref => 'be18e2ee3803bfe39dced1fa9113b4e87139e857'

--More--(26%)
```

Software Release management is not a solved problem

Librarian Puppet

- Hides complexity of submodules
- Easy if you use Forge Modules
  - Does anyone ?
  - Do you trust the internet to be around

- Librarian = Old English for “can't use submodules”
- And hmm... which customer uses which patched version again ?

Continuous Integration

Continuous integration (CI) is the practice, in software engineering, of merging all developer working copies with a shared mainline several times a day. It was first named and proposed as part of extreme programming (XP). Its main aim is to prevent integration problems, referred to as "integration hell"

(WikiPedia)

Does the app you are deploying still work ?

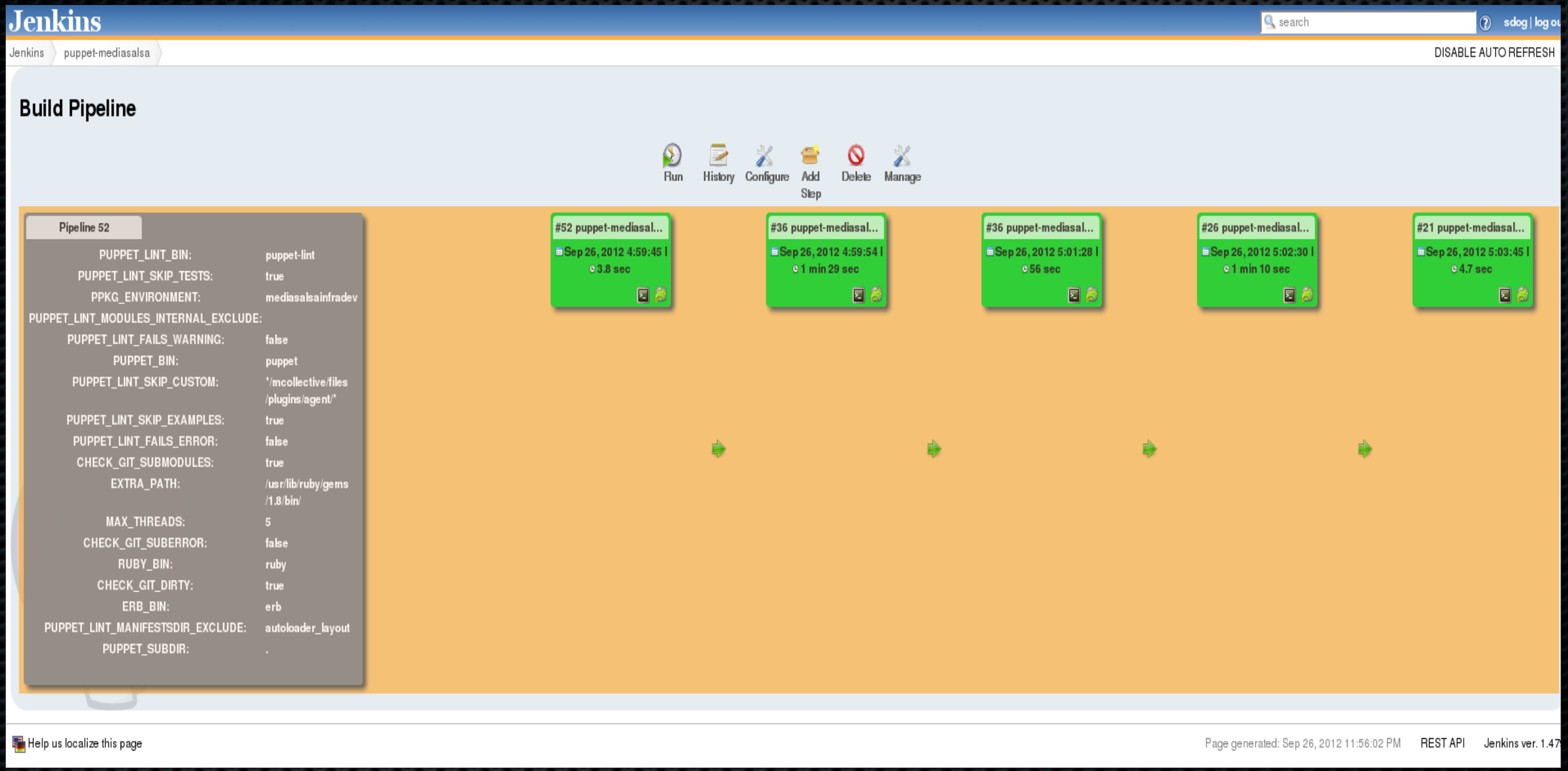
Did you break your puppet / chef code ?

Jenkins

- Open Source Continuous Integration Server
- A zillion plugins (400)

- Have developers build stable and deployable code
- Test Infra code

Jenkins Pipeline



What's in your Pipeline ?

A pipeline

- Checkout code
- Syntax
- Style
- Code Coverage
- Tests
- Build
- More Tests
- Package

Syntax and Style

- Initially ,

  all code, all the time
- Now,

  only the changed code
- Why not in post Commit Hooks ?

Package all the things

Artifacts:

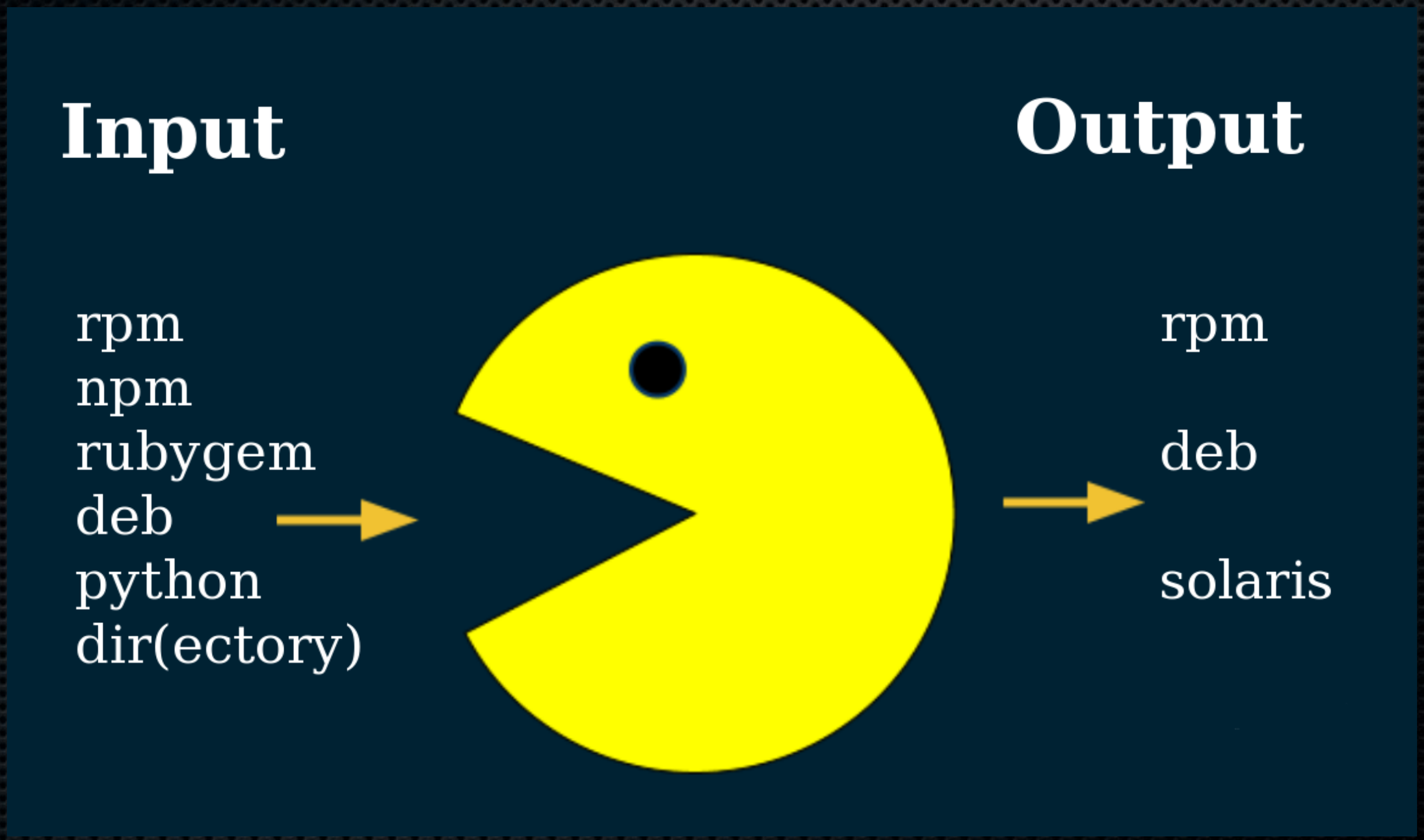
- Tested artifacts that go through a pipeline
  - application code,
  - Infra code
  - metadata
  - tests

Why ops like to package

- Packages give you features
- Consistency, security, dependencies
- Uniquely identify where files come from
- Package or cfg-mgmt
- Source repo not always available
- Firewall / Cloud etc ..
- Weird deployment locations , no easy access
- Little overhead when you automate

Jordan Sissel is a Hero !

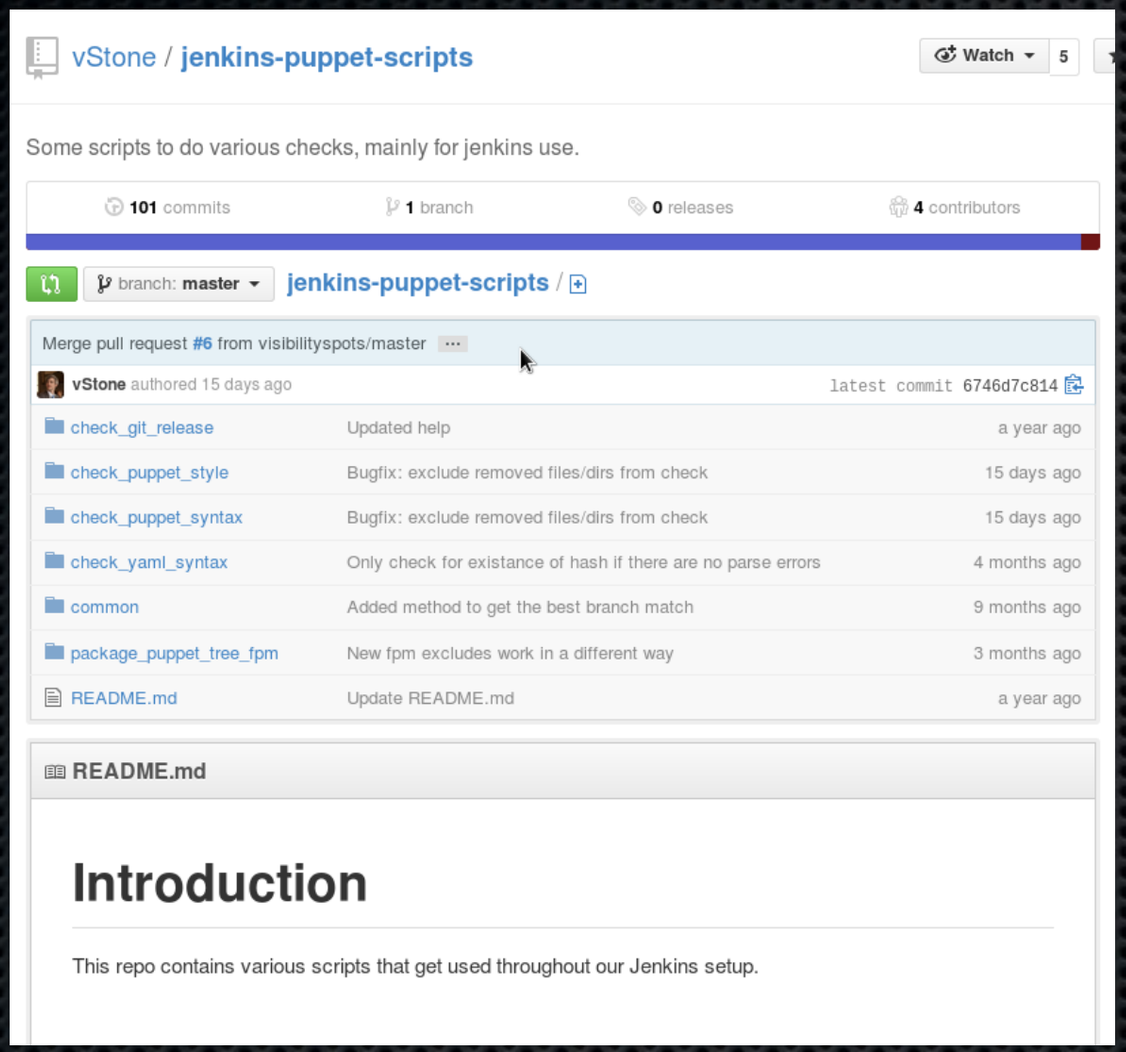
#packaginlove



It's not really packaging

- It's an immutable branch
- It's a tracable release artefact

https://github.com/vStone/jenkins-puppet-scripts



- Tests
- Packages full tree in / etc/puppet/environments/$environment/

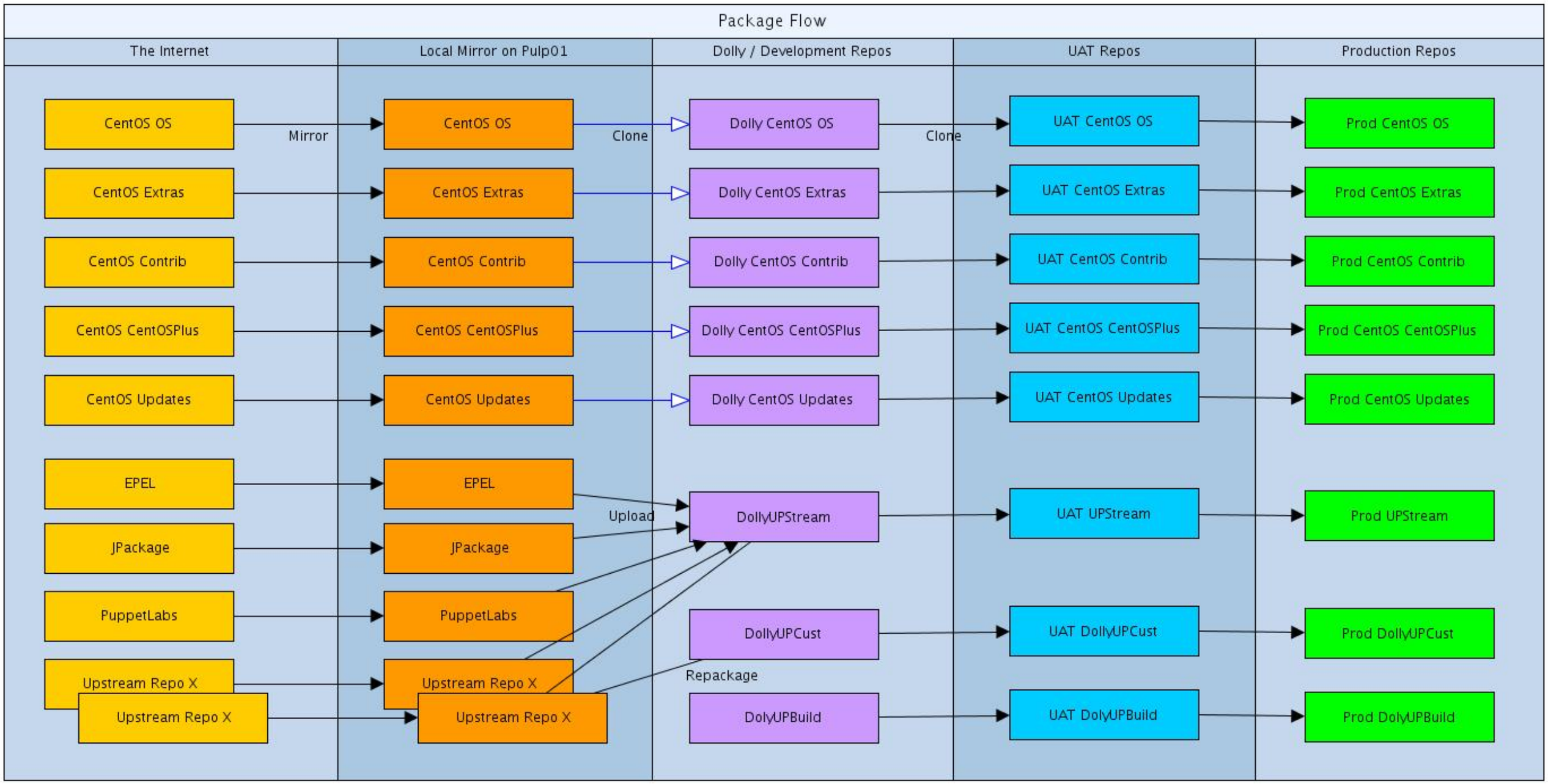
A pipeline

- Checkout code
- Syntax
- Style
- Code Coverage
- Tests
- Build
- More Tests
- Package
- Upload to Repo

Repository Management

- Pulp
  - Pro : MirroringLove
  - Con : Mongo, Stability, .deb
- Aptly (deb only)
- Prm (missing snapshot features for .rpm)

Repository Management



A pipeline

- Checkout code
- Syntax
- Style
- Code Coverage
- Tests
- Build
- More Tests
- Package
- Upload to Repo
- Deploy on Test

Repos are SLOW

- Createrepo is slow.
- Pulp is slow

- Bypass repos , upload straight to appropriate PuppetMaster
- Upload to repo for rebootstrapping

A pipeline

- Checkout code
- Syntax
- Style
- Code Coverage
- Tests
- Build
- More Tests
- Package
- Upload to Repo
- Deploy on Test
- Check Puppetruns
- Check Monitoring

Testing = Monitoring

- Deploy a host,
- Add it to the monitoring framework
- Add collection tools
- Add check definitions
- Update the monitoring tool config

FULLY AUTOMATED

A pipeline

- Checkout code
- Syntax
- Style
- Code Coverage
- Tests
- Build
- More Tests
- Package
- Upload to Repo
- Deploy on Test
- Check Puppetruns
- Check Monitoring
- Promote to UAT

Jenkins Promotion

From: jenkins@inuits.eu
To: roidelapluie@inuits.eu, mediasalsa@lists.inuits.eu, carroarmato0@inuits.eu
Subject: [Mediasalsa] puppet-mediasalsa-infradev-promote-me - Build # 461 - Successful!
Date: Thu, 23 Jan 2014 15:08:48 +0100 (CET)

The last changes in the mediasalsa puppet tree have been deployed to infradev.

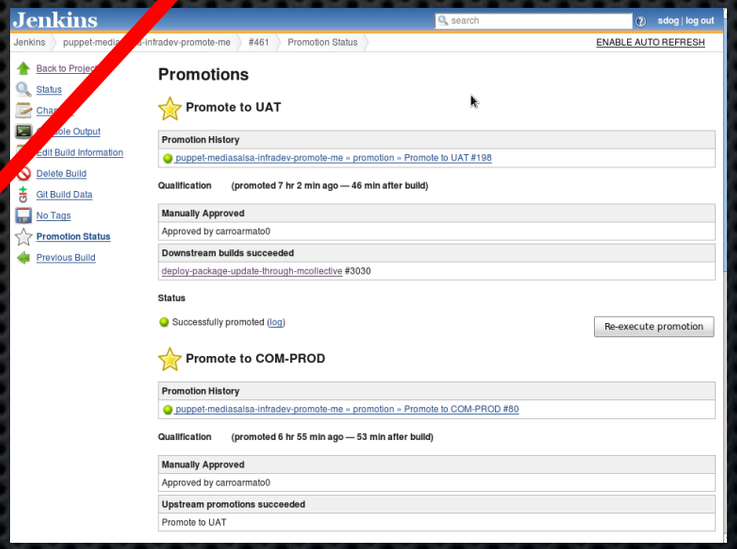
Check the lastest changes in in https://jenkins.dev.inuits.eu/view/puppet-mediasalsa/

To promote me to UAT or PROD or POL-PROD, click here: https://jenkins.dev.inuits.eu/job/puppet-mediasalsa-infradev-promote-me/461/promotion/

Changes:
[carroarmato0] Refactor stor01 and stor02 to be in sync and use use_drbd flag for enabling replication

regards,
Mr Jenkins

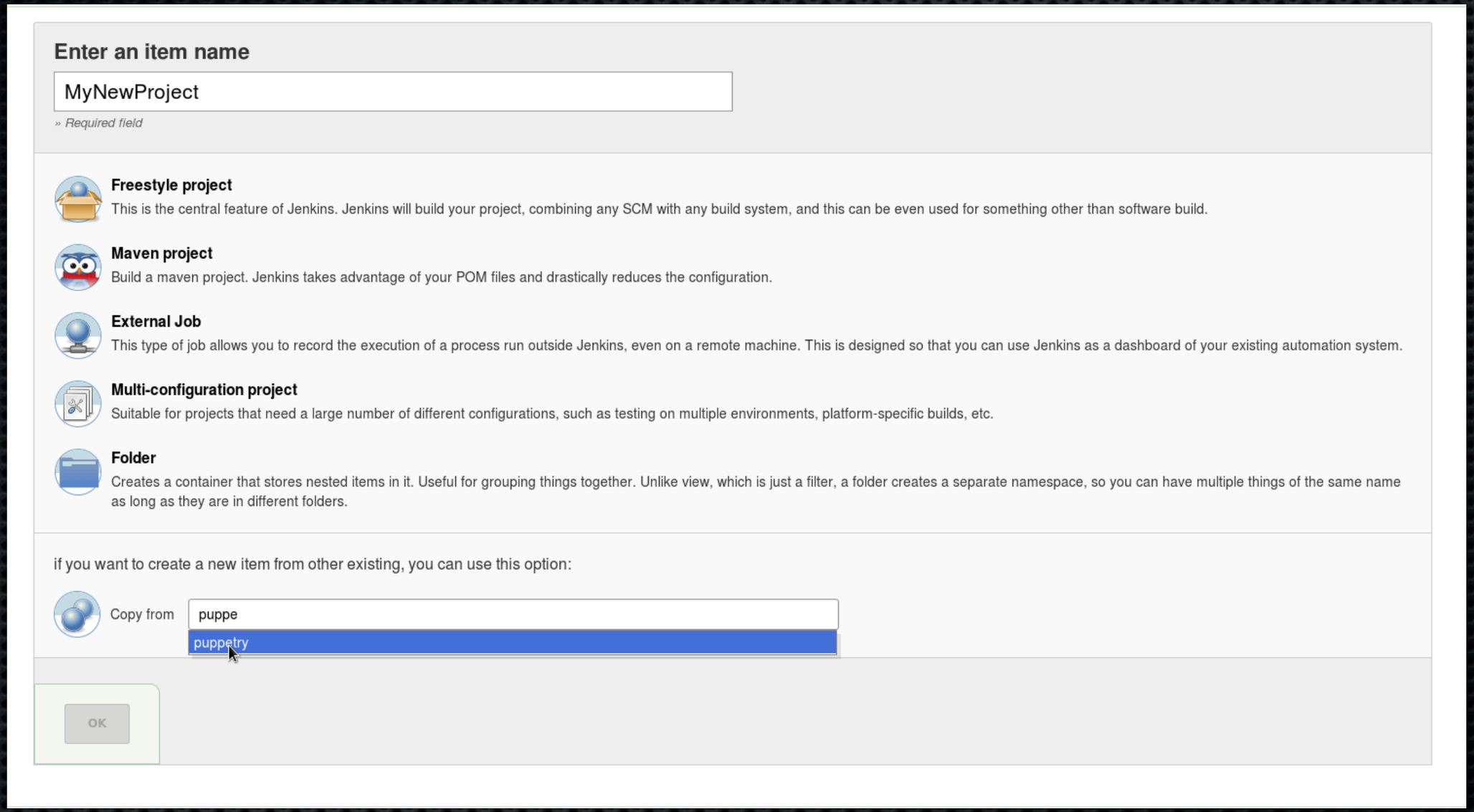
Mediasalsa mailing list
Mediasalsa@lists.inuits.eu
https://lists.inuits.eu/mailman/listinfo/mediasalsa



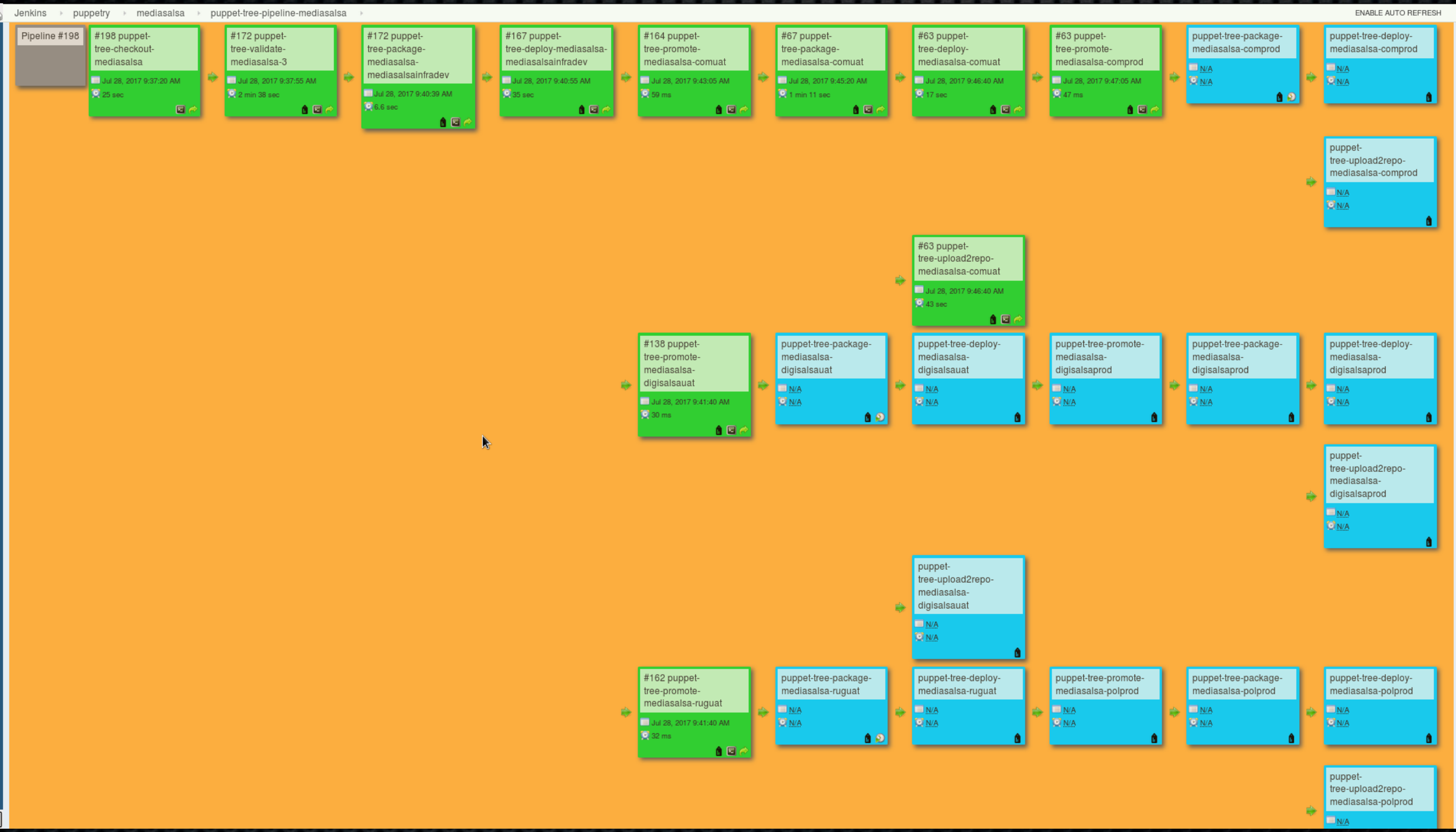
Pipelines

- Lots of them
- Similar ones, but not identical ones
- One project = different deployment targets
- People move teams expect same patterns
- Mostly unmaintaned

Dirty Clickers



Complex Pipelines



Scaling Pipelines

- Create a Pipeline,
- For job in Pipeline
  - Create new Job Based on OldJob
- Update One Job
- Never refactor the rest

Generating Jenkins Pipelines

- Template the XML
- Put it in Puppet
  - Worked for stable pipelines
  - Kinda
- XML gets rewritten on the fly

JenkinsJobBuilder

- First Usable Attempt
- Python Based
- Openstack Community
- Limited Functionality
- Little Adoption / Openstack Abandonned it

PipelineDSL

- Jenkinsfile
- Inside a repo
- Groovy
- Limited functionality
- Initially buggy
- Popular for Easy tasks
  - (straight, no splits etc)

Jenkins Job DSL

- Groovy
- Flexible
- Well Supported
- Suitable for more complex Pipelines

Seedjobs

- Groovy
- Git
- Rebuild jobs on commit
- Projects in folders

Larger CI Stacks

- Generate Pipelines / Jobs based on config files ,
- Build libraries
  - CheckoutJob
  - DeployJob
  - PackageJob
- Use Groovy / JobDSL to generate PipelineDSL

Testing Multiple Versions

- Initial stage tests code on multiple versions
  - e.g current puppet version
  - Next puppet version
  - Bleeding Edge version
- Only current version breaks build
- Goal = get all versions green

Problems solved

- One job per task, no reuse of jobs with different parameters
- All Jobs are in sync, update one = update all
- Centrally managed jobs (git)

Stop Clicking , Write Code

Contact

Kris Buytaert
Kris.Buytaert@inuits.be

Further Reading
@krisbuytaert
http://www.krisbuytaert.be/blog/
http://www.inuits.be/

Inuits

Duboistraat 50
2060 Antwerpen
Belgium
891.514.231

+32 475 961221