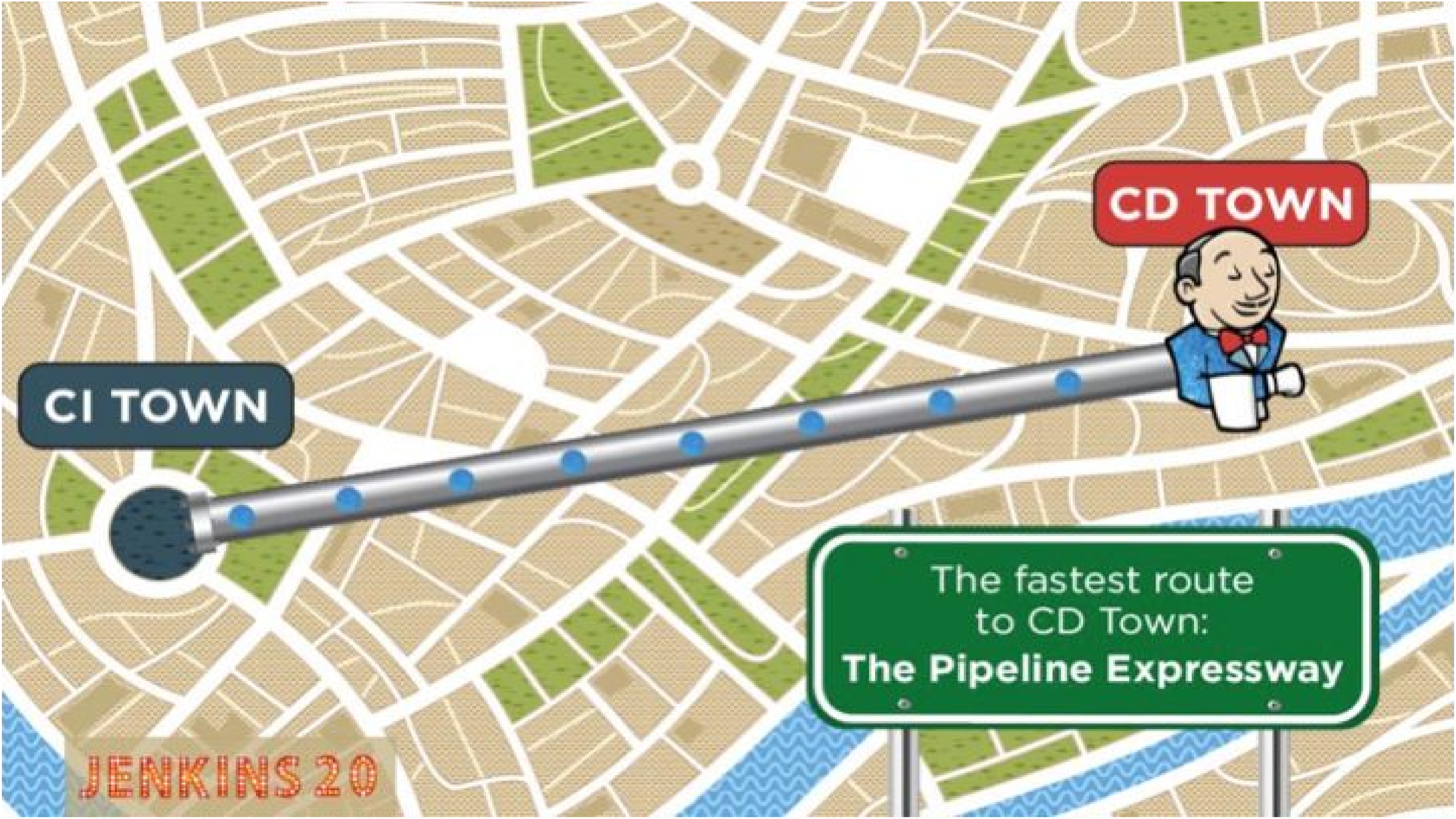
Pipeline as Code

Jenkins 2.0 and Pipeline


CD TOWN
CI TOWN
The fastest route to CD Town:
The Pipeline Expressway
JENKINS 20

Continuous Delivery

Continuous delivery (CD) is a software engineering approach in which teams produce software in short cycles, ensuring that the software can be reliably released at any time. It aims at building, testing, and releasing software faster and more frequently. The approach helps reduce the cost, time, and risk of delivering changes by allowing for more incremental updates to applications in production. ***A straightforward and repeatable deployment process is important for continuous delivery.***

https://en.wikipedia.org/wiki/Continuous_delivery

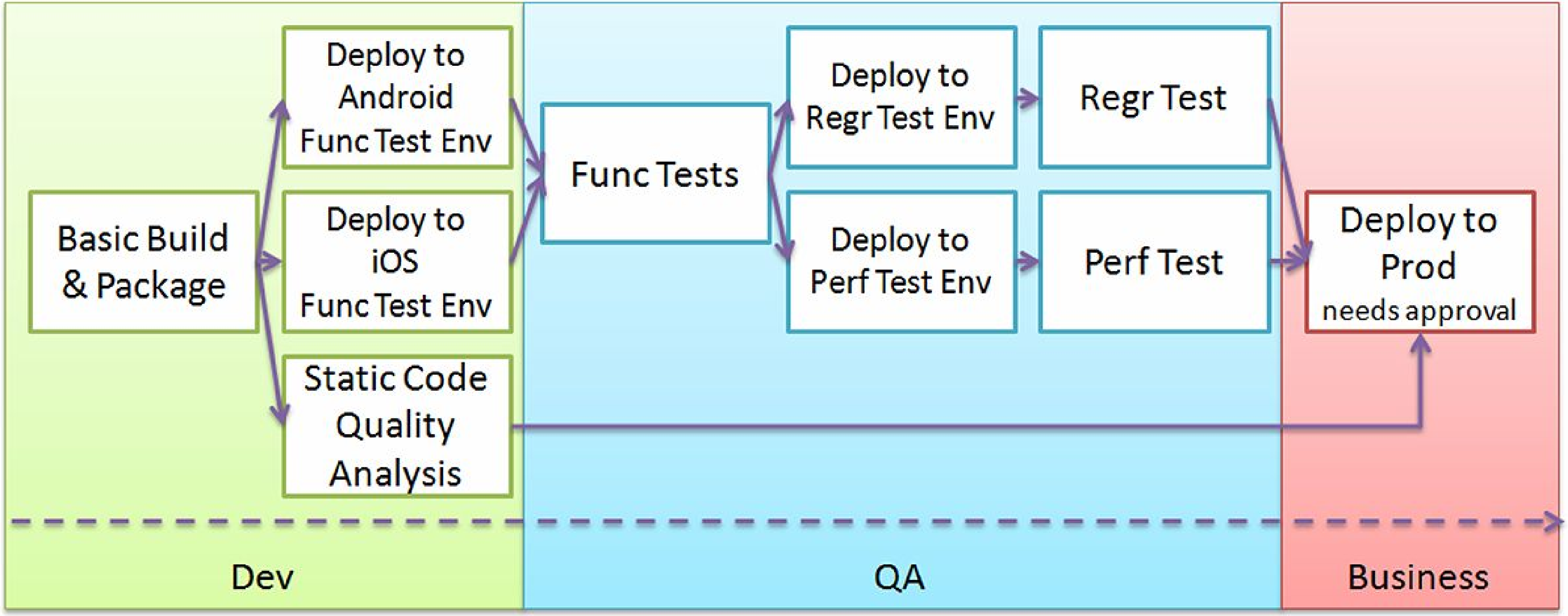
CD with Jenkins 1

Steps to CD in Jenkins

- Ensuring reproducible builds
- Sharing build artifacts throughout the pipeline
- Choosing the right granularity for each job
- Parallelizing and joining jobs
- Gates and approvals
- Visualizing the pipeline
- Organizing and securing jobs
- Good practice: versioning your Jenkins configuration

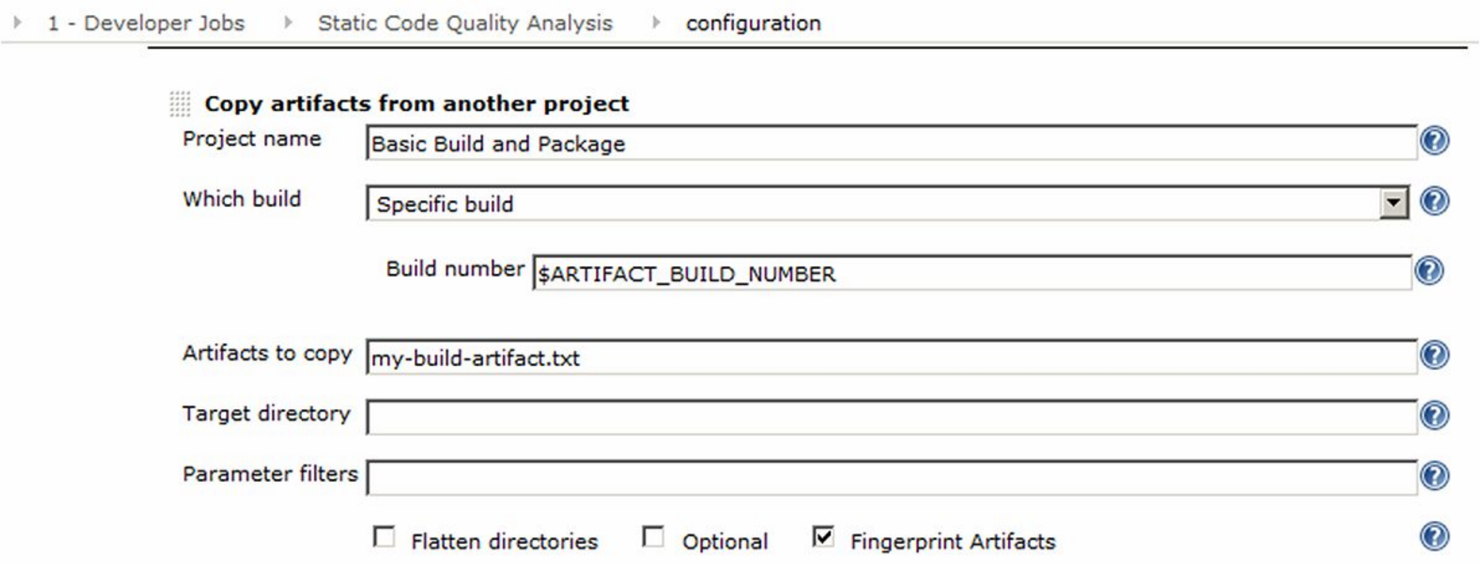
http://www.infoq.com/articles/orch-pipelines-jenkins

CD with Jenkins circa 2014


Basic Build & Package
Deploy to Android Func Test Env
Deploy to iOS Func Test Env
Static Code Quality Analysis
Func Tests
Deploy to Regr Test Env
Regr Test
Deploy to Perf Test Env
Perf Test
Deploy to Prod needs approval
Dev
QA
Business

http://www.infoq.com/articles/orch-pipelines-jenkins

Static Code Quality Analysis Job

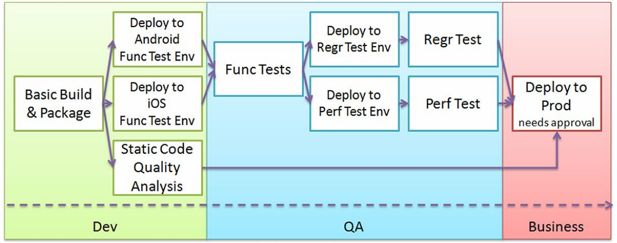


1 - Developer Jobs › Static Code Quality Analysis › configuration

Copy artifacts from another project

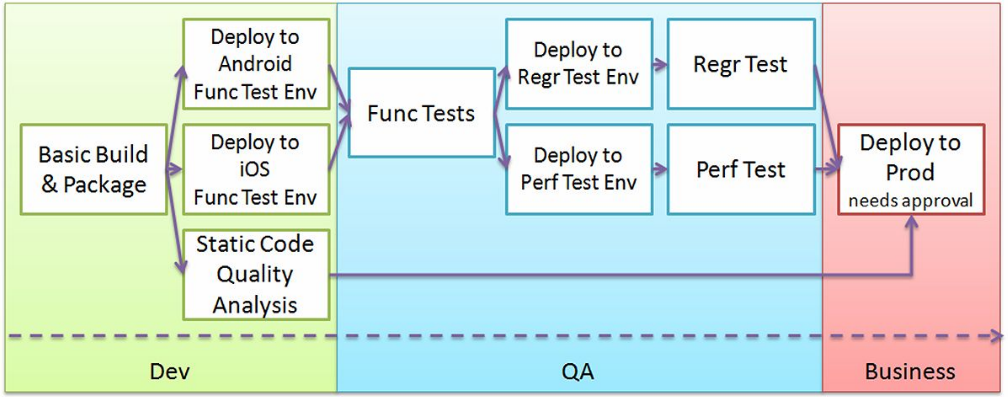
Project name	Basic Build and Package
Which build	Specific build
Build number	$ARTIFACT_BUILD_NUMBER
Artifacts to copy	my-build-artifact.txt
Target directory	
Parameter filters	

□ Flatten directories □ Optional ☑ Fingerprint Artifacts

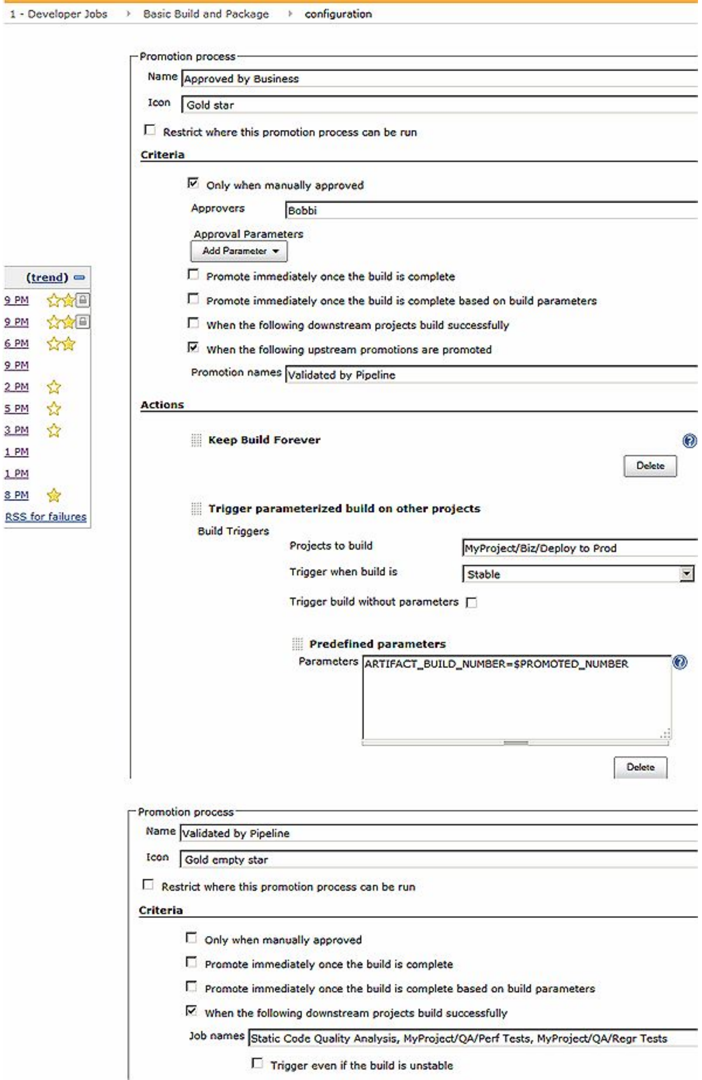


Basic Build & Package
Deploy to Android Func Test Env
Deploy to iOS Func Test Env
Static Code Quality Analysis
Func Tests
Deploy to Regr Test Env
Regr Test
Deploy to Perf Test Env
Perf Test
Deploy to Prod needs approval
Dev
QA
Business

Approval



Basic Build & Package
Deploy to Android Func Test Env
Deploy to iOS Func Test Env
Static Code Quality Analysis
Func Tests
Deploy to Regr Test Env
Regr Test
Deploy to Perf Test Env
Perf Test
Deploy to Prod needs approval
Dev
QA
Business



1 - Developer Jobs › Basic Build and Package › configuration

(trend)

9 PM ☆☆
9 PM ☆☆
6 PM ☆☆
9 PM
2 PM ☆
5 PM ☆
3 PM ☆
1 PM
1 PM
8 PM ☆

RSS for failures

Promotion process

Name: Approved by Business

Icon: Gold star

□ Restrict where this promotion process can be run

Criteria

☑ Only when manually approved

Approvers: Bobbi

Approval Parameters

□ Promote immediately once the build is complete

□ Promote immediately once the build is complete based on build parameters

□ When the following downstream projects build successfully

☑ When the following upstream promotions are promoted

Promotion names: Validated by Pipeline

Actions

Keep Build Forever

Trigger parameterized build on other projects

Build Triggers

Projects to build: MyProject/Biz/Deploy to Prod

Trigger when build is: Stable

Trigger build without parameters □

Predefined parameters

Parameters: ARTIFACT_BUILD_NUMBER=$PROMOTED_NUMBER

Promotion process

Name: Validated by Pipeline

Icon: Gold empty star

□ Restrict where this promotion process can be run

Criteria

□ Only when manually approved

□ Promote immediately once the build is complete

□ Promote immediately once the build is complete based on build parameters

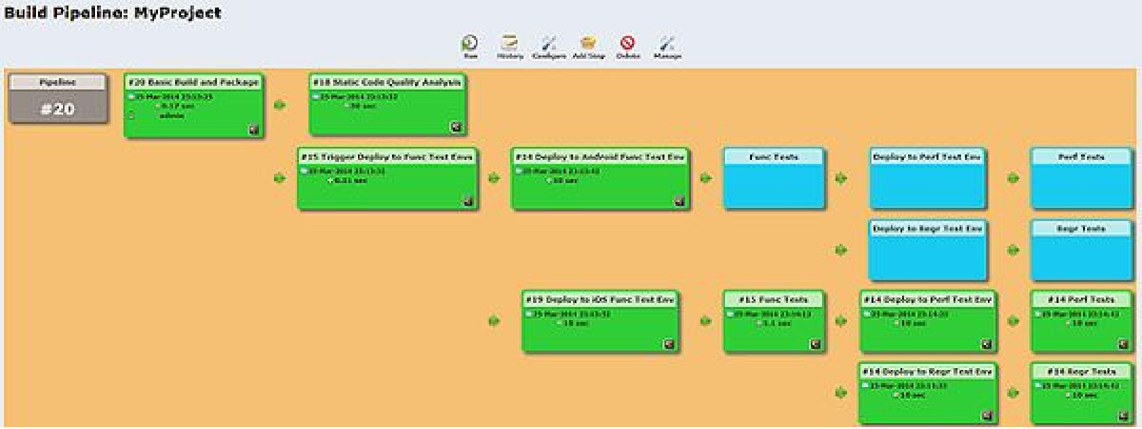
☑ When the following downstream projects build successfully

Job names: Static Code Quality Analysis, MyProject/QA/Perf Tests, MyProject/QA/Regr Tests

□ Trigger even if the build is unstable

Visualizing the Pipeline

Build Pipeline Plugin



Build Pipeline: MyProject

Pipeline
#20

#20 Basic Build and Package
25 Mar 2014 23:13:25
0.17 sec
admin

#18 Static Code Quality Analysis
25 Mar 2014 23:13:32
30 sec

#15 Trigger Deploy to Func Test Envs
25 Mar 2014 [illegible]
0.21 sec

#14 Deploy to Android Func Test Env
25 Mar 2014 [illegible]
10 sec

Func Tests

Deploy to Perf Test Env

Perf Tests

Deploy to Regr Test Env

Regr Tests

#19 Deploy to iOS Func Test Env
25 Mar 2014 [illegible]
10 sec

#15 Func Tests
25 Mar 2014 [illegible]
1.1 sec

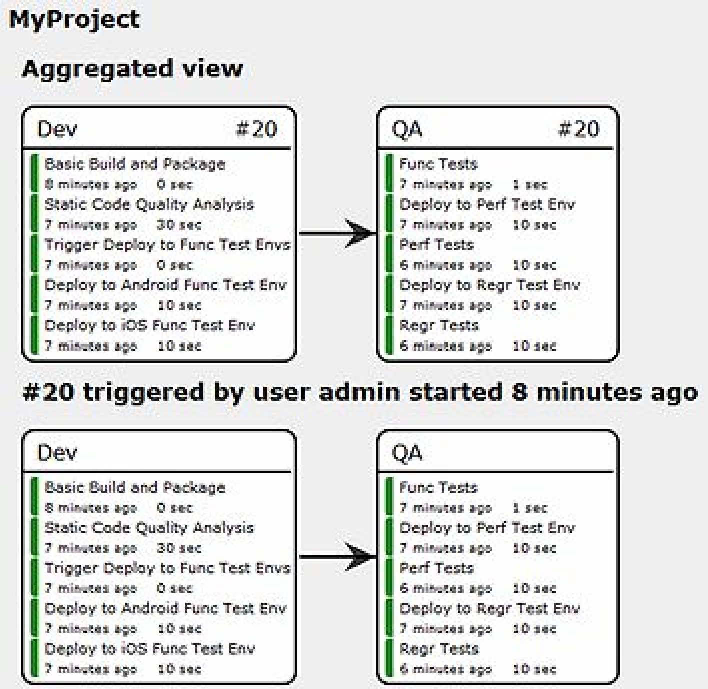
#14 Deploy to Perf Test Env
25 Mar 2014 [illegible]
10 sec

#14 Perf Tests
25 Mar 2014 [illegible]
10 sec

#14 Deploy to Regr Test Env
25 Mar 2014 [illegible]
10 sec

#14 Regr Tests
25 Mar 2014 [illegible]
10 sec

Delivery Pipeline Plugin



MyProject

Aggregated view

Dev #20
Basic Build and Package
8 minutes ago 0 sec
Static Code Quality Analysis
7 minutes ago 30 sec
Trigger Deploy to Func Test Envs
7 minutes ago 0 sec
Deploy to Android Func Test Env
7 minutes ago 10 sec
Deploy to iOS Func Test Env
7 minutes ago 10 sec

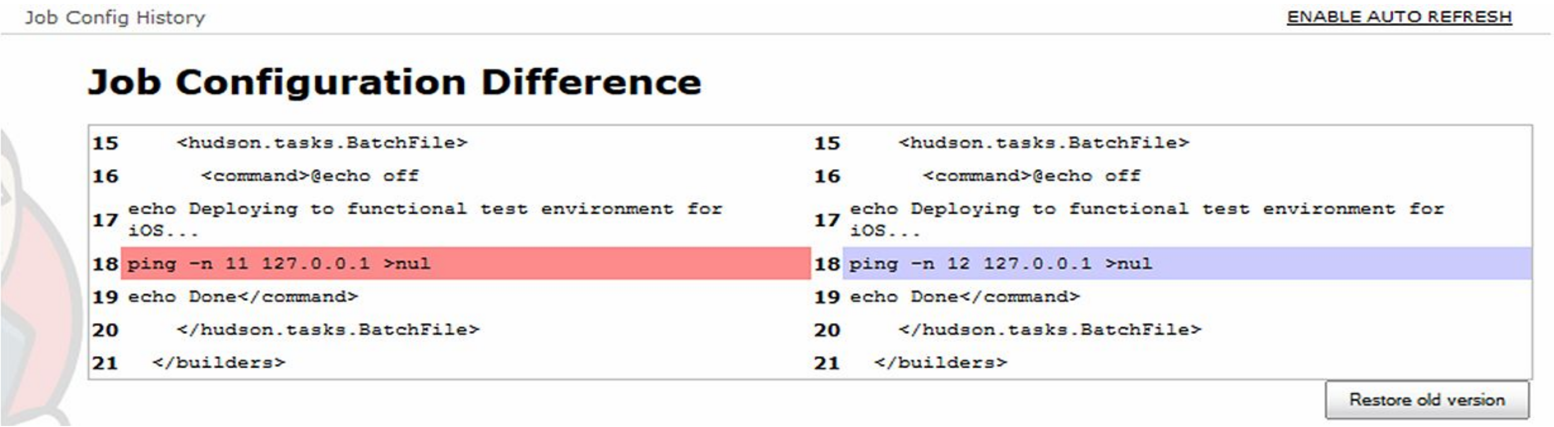
QA #20
Func Tests
7 minutes ago 1 sec
Deploy to Perf Test Env
7 minutes ago 10 sec
Perf Tests
6 minutes ago 10 sec
Deploy to Regr Test Env
7 minutes ago 10 sec
Regr Tests
6 minutes ago 10 sec

#20 triggered by user admin started 8 minutes ago

Dev
Basic Build and Package
8 minutes ago 0 sec
Static Code Quality Analysis
7 minutes ago 30 sec
Trigger Deploy to Func Test Envs
7 minutes ago 0 sec
Deploy to Android Func Test Env
7 minutes ago 10 sec
Deploy to iOS Func Test Env
7 minutes ago 10 sec

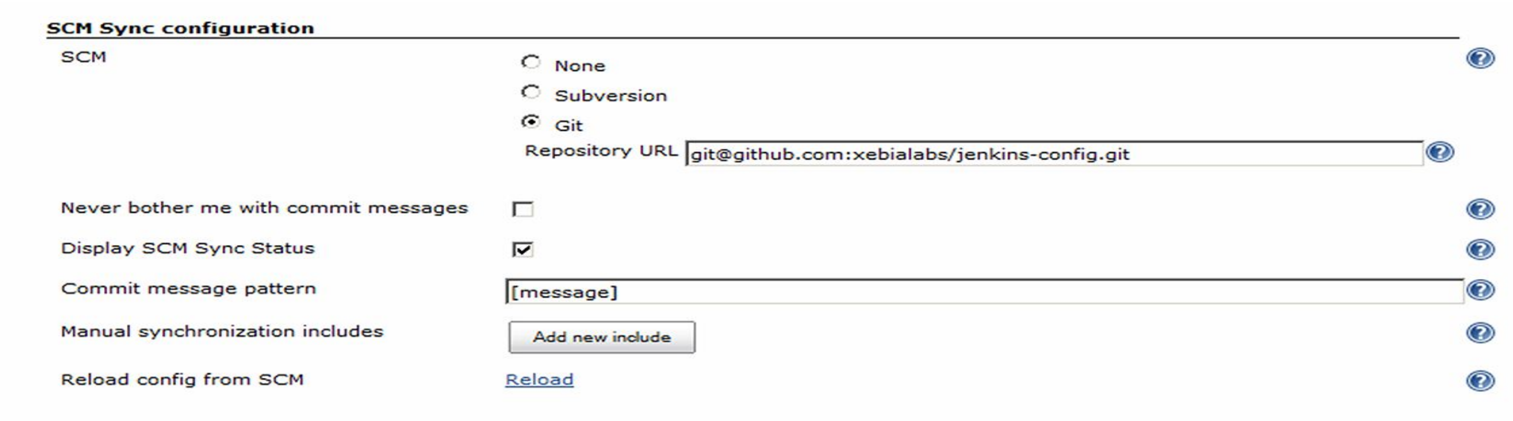
QA
Func Tests
7 minutes ago 1 sec
Deploy to Perf Test Env
7 minutes ago 10 sec
Perf Tests
6 minutes ago 10 sec
Deploy to Regr Test Env
7 minutes ago 10 sec
Regr Tests
6 minutes ago 10 sec

SCM Sync Configuration Plugin



Job Configuration Difference

```
    <hudson.tasks.BatchFile>                          <hudson.tasks.BatchFile>
      <command>@echo off                                <command>@echo off
echo Deploying to functional test environment for  echo Deploying to functional test environment for
iOS...                                             iOS...
ping -n 11 127.0.0.1 >nul                          ping -n 12 127.0.0.1 >nul
echo Done</command>                                echo Done</command>
    </hudson.tasks.BatchFile>                          </hudson.tasks.BatchFile>
  </builders>                                        </builders>
```



SCM Sync configuration

SCM: ○ None ○ Subversion ● Git

Repository URL: git@github.com:xebialabs/jenkins-config.git

Never bother me with commit messages □

Display SCM Sync Status ☑

Commit message pattern: [message]

Manual synchronization includes: Add new include

Reload config from SCM: Reload

Jenkins Pipeline

Pipeline as Code

- ➢ Introduce “pipeline” as a new type in Jenkins
- ➢ Codify an implicit series of stages into an explicit `Jenkinsfile` in your source repository
- ➢ Resumability/durability of the pipeline state
- ➢ Extend the DSL with your own steps
- ➢ DRY - Reusable components and flows

github.com/jenkinsci/pipeline-examples

Pipeline DSL

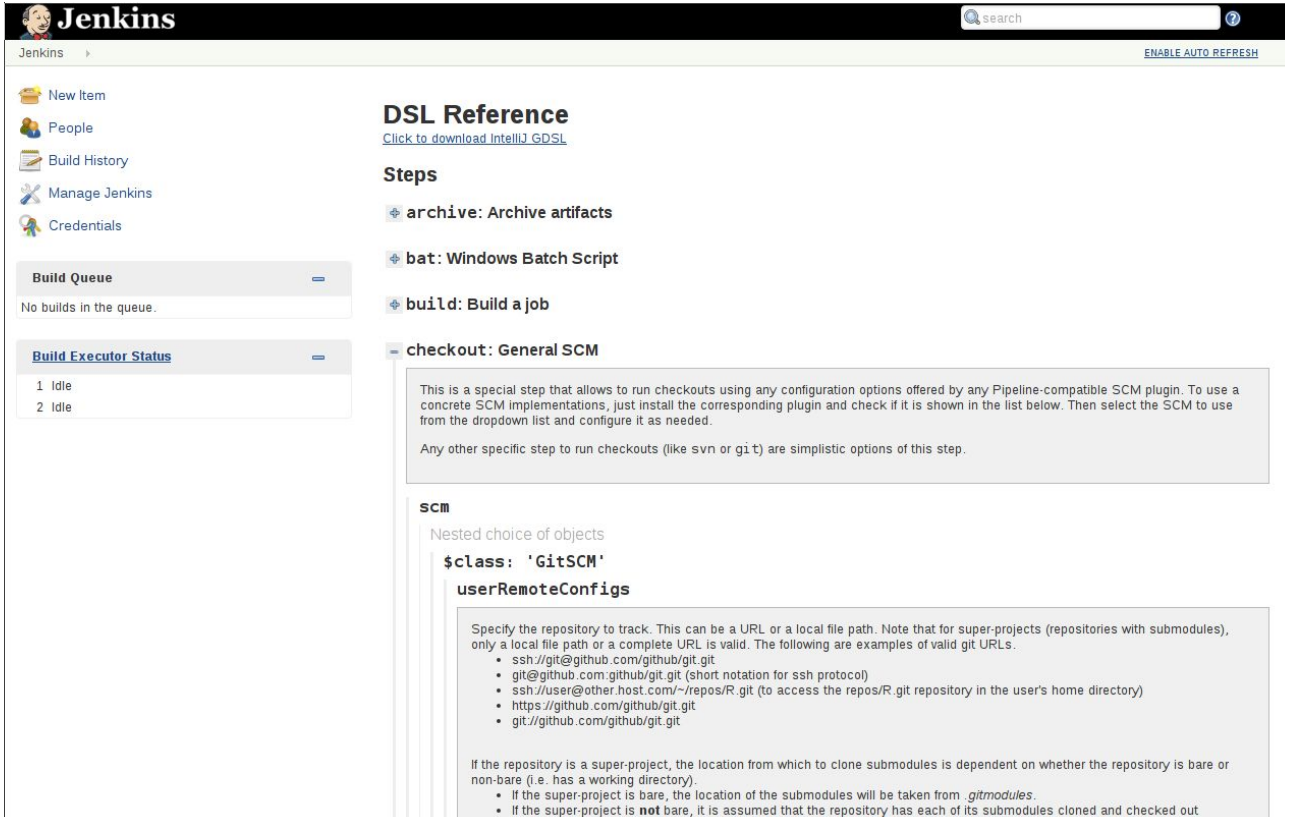
```
node('docker') {
    checkout scm

    /* Grab the abbreviated SHA1 of our pipeline’s commit.*/
    sh 'git rev-parse HEAD > GIT_COMMIT'
    def shortCommit = readFile('GIT_COMMIT').take(6)

    stage 'Build'
    def image = docker.build("jenkinsciinfra/bind:build-${shortCommit}")

    stage 'Deploy'
    image.push()
}
```

Pipeline DSL



Build Queue

No builds in the queue.

Build Executor Status

1 Idle
2 Idle

DSL Reference

Click to download IntelliJ GDSL

Steps

- archive: Archive artifacts
- bat: Windows Batch Script
- build: Build a job
- checkout: General SCM

This is a special step that allows to run checkouts using any configuration options offered by any Pipeline-compatible SCM plugin. To use a concrete SCM implementations, just install the corresponding plugin and check if it is shown in the list below. Then select the SCM to use from the dropdown list and configure it as needed.

Any other specific step to run checkouts (like svn or git) are simplistic options of this step.

scm

Nested choice of objects

$class: 'GitSCM'

userRemoteConfigs

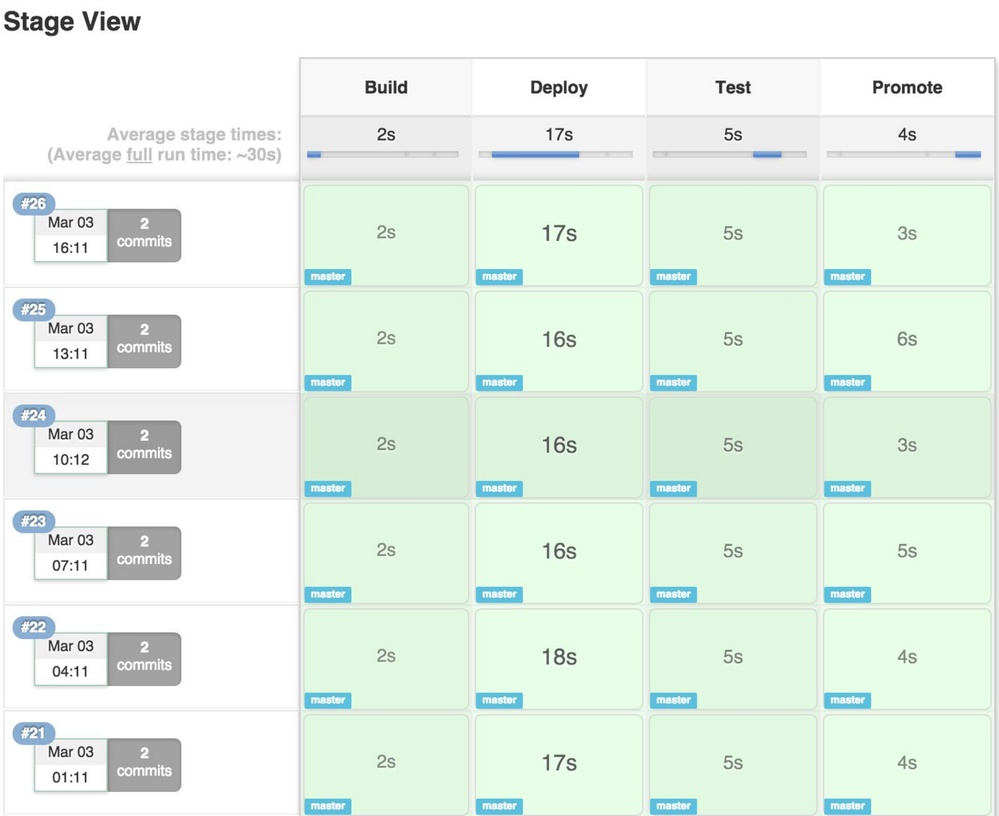
Specify the repository to track. This can be a URL or a local file path. Note that for super-projects (repositories with submodules), only a local file path or a complete URL is valid. The following are examples of valid git URLs.

- ssh://git@github.com/github/git.git
- git@github.com:github/git.git (short notation for ssh protocol)
- ssh://user@other.host.com/~/repos/R.git (to access the repos/R.git repository in the user's home directory)
- https://github.com/github/git.git
- git://github.com/github/git.git

If the repository is a super-project, the location from which to clone submodules is dependent on whether the repository is bare or non-bare (i.e. has a working directory).

- If the super-project is bare, the location of the submodules will be taken from *.gitmodules*.
- If the super-project is **not** bare, it is assumed that the repository has each of its submodules cloned and checked out

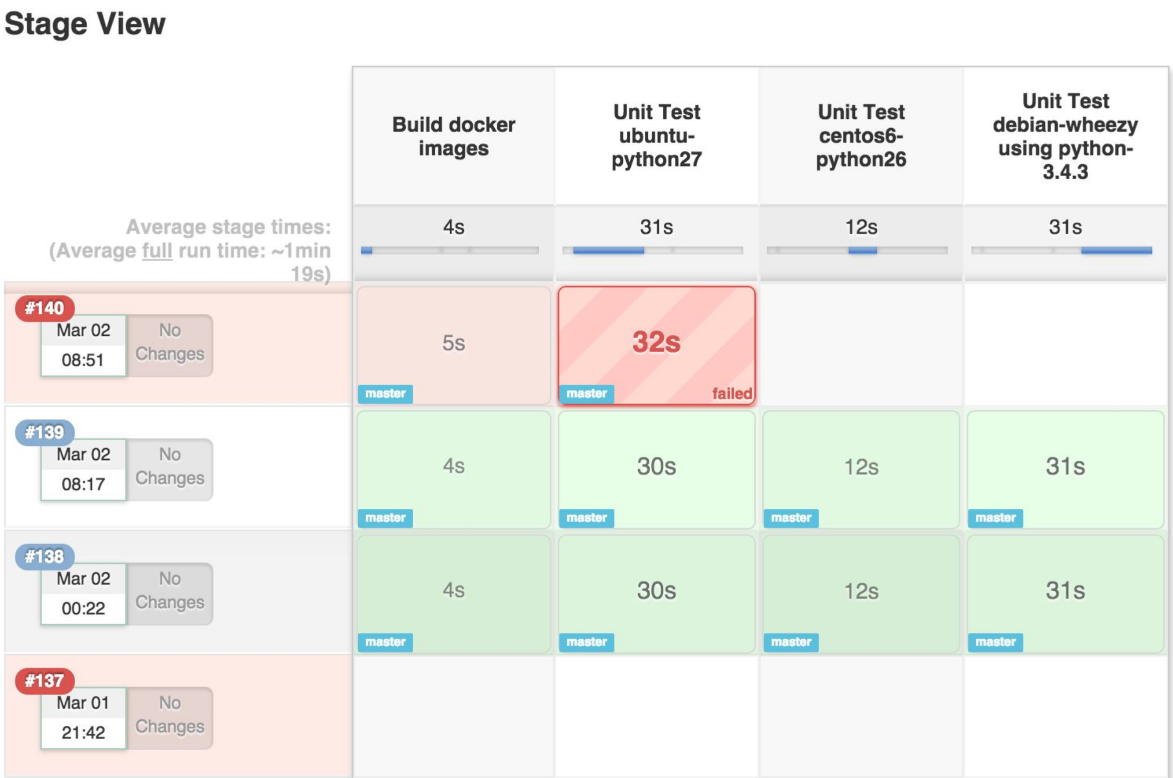
Pipeline Stage View



Stage View

	Build	Deploy	Test	Promote
Average stage times: (Average full run time: ~30s)	2s	17s	5s	4s
#26 Mar 03 16:11 2 commits	2s master	17s master	5s master	3s master
#25 Mar 03 13:11 2 commits	2s master	16s master	5s master	6s master
#24 Mar 03 10:12 2 commits	2s master	16s master	5s master	3s master
#23 Mar 03 07:11 2 commits	2s master	16s master	5s master	5s master
#22 Mar 03 04:11 2 commits	2s master	18s master	5s master	4s master
#21 Mar 03 01:11 2 commits	2s master	17s master	5s master	4s master

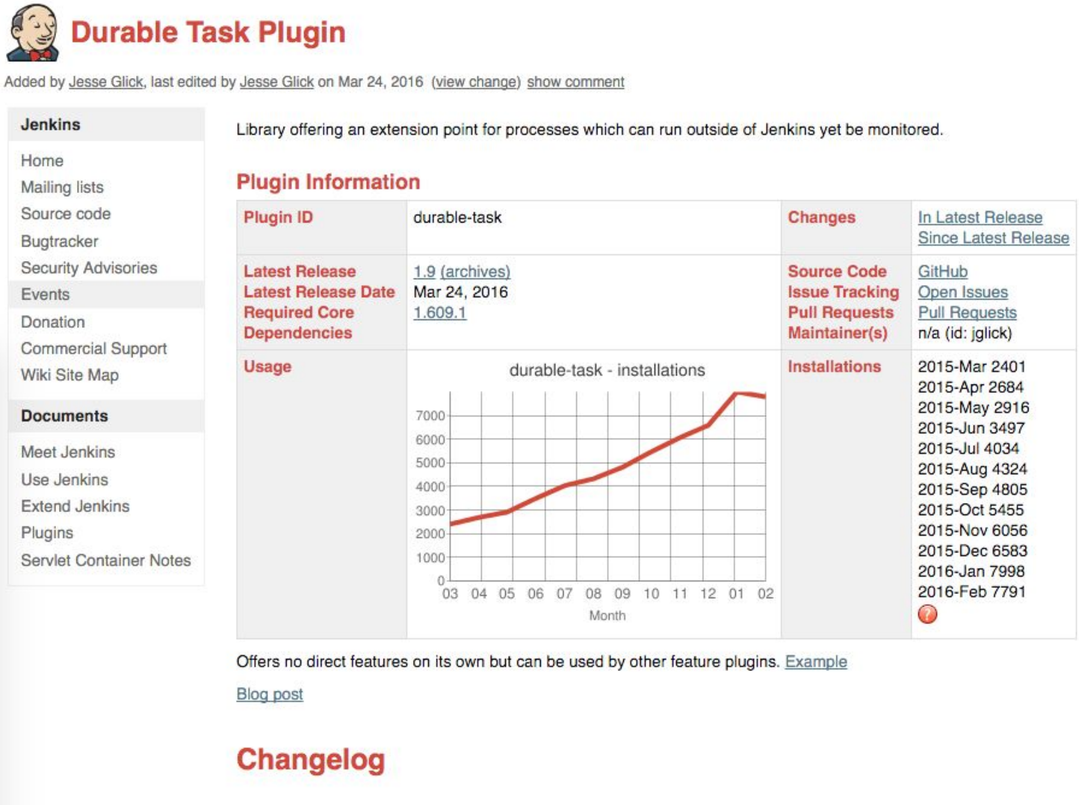
Pipeline Stage View



Stage View

	Build docker images	Unit Test ubuntu-python27	Unit Test centos6-python26	Unit Test debian-wheezy using python-3.4.3
Average stage times: (Average full run time: ~1min 19s)	4s	31s	12s	31s
#140 Mar 02 08:51 No Changes	5s master	32s master failed		
#139 Mar 02 08:17 No Changes	4s master	30s master	12s master	31s master
#138 Mar 02 00:22 No Changes	4s master	30s master	12s master	31s master
#137 Mar 01 21:42 No Changes				

Durable Task Plugin



Durable Task Plugin

Added by Jesse Glick, last edited by Jesse Glick on Mar 24, 2016 (view change) show comment

Jenkins

- Home
- Mailing lists
- Source code
- Bugtracker
- Security Advisories
- Events
- Donation
- Commercial Support
- Wiki Site Map

Documents

- Meet Jenkins
- Use Jenkins
- Extend Jenkins
- Plugins
- Servlet Container Notes

Library offering an extension point for processes which can run outside of Jenkins yet be monitored.

Plugin Information

Plugin ID	durable-task	Changes	In Latest Release Since Latest Release
Latest Release Latest Release Date Required Core Dependencies	1.9 (archives) Mar 24, 2016 1.609.1	Source Code Issue Tracking Pull Requests Maintainer(s)	GitHub Open Issues Pull Requests n/a (id: jglick)
Usage	durable-task - installations	Installations	2015-Mar 2401 2015-Apr 2684 2015-May 2916 2015-Jun 3497 2015-Jul 4034 2015-Aug 4324 2015-Sep 4805 2015-Oct 5455 2015-Nov 6056 2015-Dec 6583 2016-Jan 7998 2016-Feb 7791

Offers no direct features on its own but can be used by other feature plugins. Example

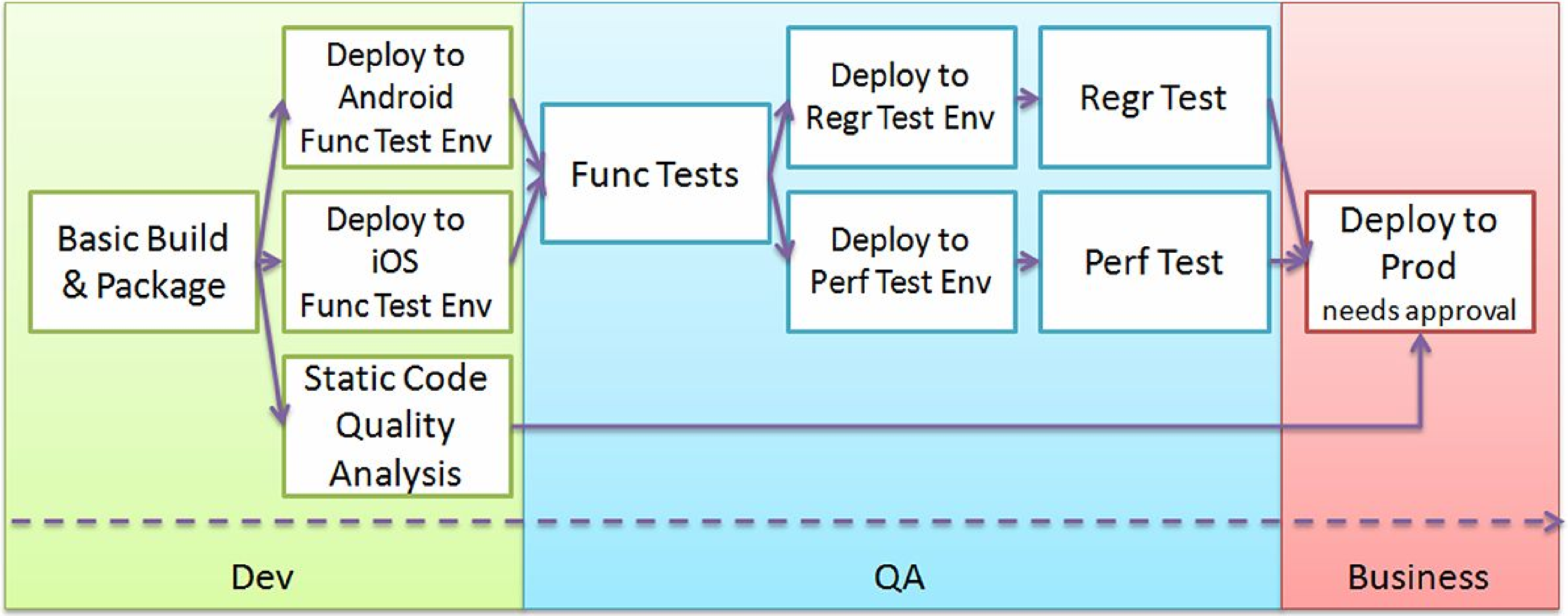
Blog post

Changelog

Extensibility

- CPS Global Library
- Remote Loader Plugin
- ‘load’ step
- Plugins

CD with Jenkins circa 2014



Basic Build & Package
Deploy to Android Func Test Env
Deploy to iOS Func Test Env
Static Code Quality Analysis
Func Tests
Deploy to Regr Test Env
Regr Test
Deploy to Perf Test Env
Perf Test
Deploy to Prod needs approval
Dev
QA
Business

http://www.infoq.com/articles/orch-pipelines-jenkins

Pipeline

- Ensuring reproducible builds
- Sharing build artifacts throughout the pipeline
- Choosing the right granularity for each job
- Parallelizing and joining jobs
- Gates and approvals
- Visualizing the pipeline
- Organizing and securing jobs
- Good practice: versioning your Jenkins configuration