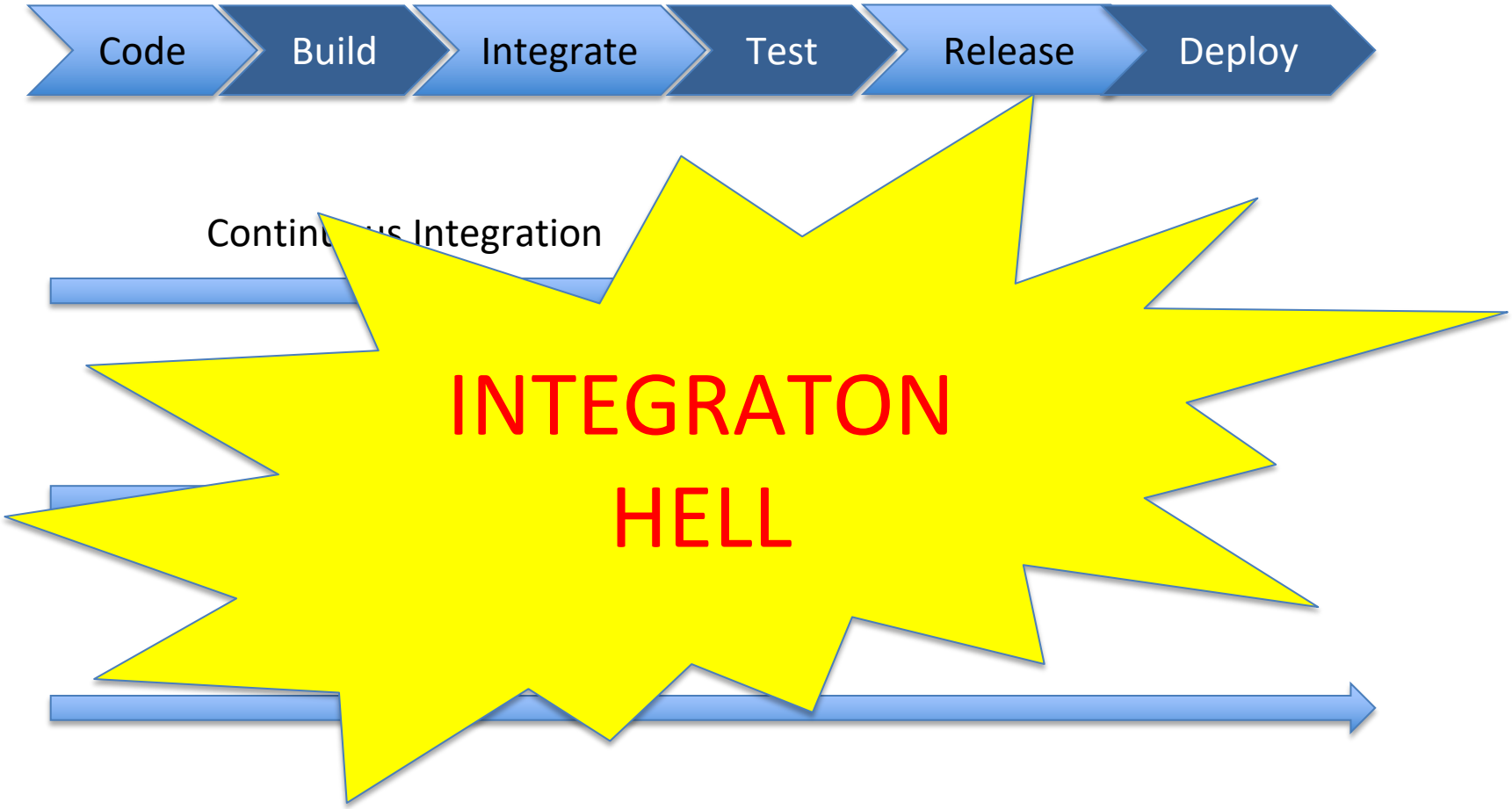
# Continuous Integration with Jenkins

James Strong
Cloud Person and other stuff

# Integrations



# What is Continuous Integration Continuous Deployment?

Continuous – All day, Every day.

Integration – This is my code, that's your code, Let's get together.

Deployment – Deploy to ALL the servers.

# Solution

Jenkins

# Some Running Rules

1. Maintain a code repository
2. Automate the build
3. Keep the build fast
4. Make the build self-testing
5. Commit early, commit often
6. Every commit to the mainline gets built
7. Everyone can see the results of the build
8. Automate the deployment

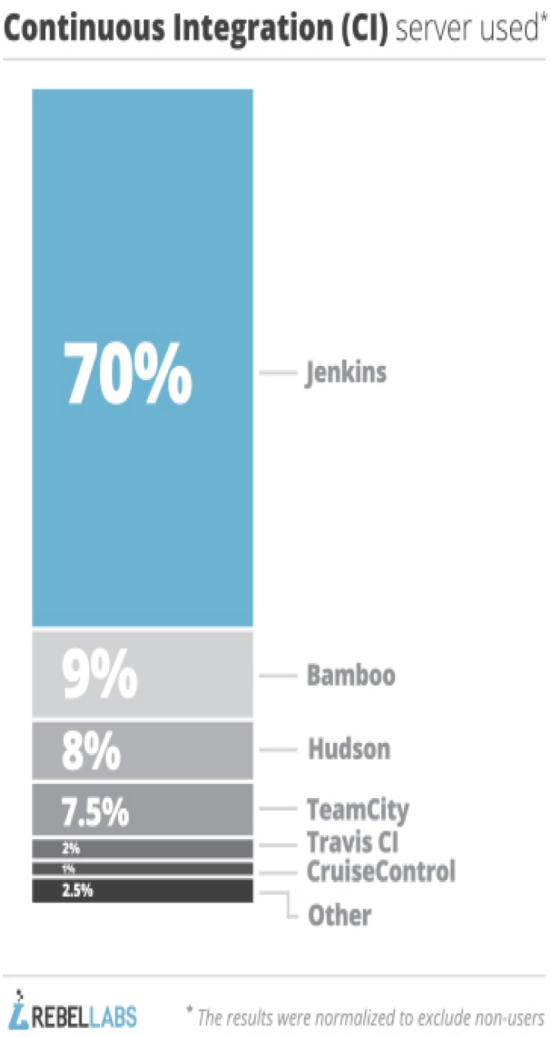
# What is Jenkins?

Some Facts:

- Written in Java.
- Jenkins was originally developed as the Hudson project. Hudson's creation started in summer of 2004 at Sun.
- Jenkins - , first released in Feb 2005
- CloudBees - Jenkins as a Service

Jenkins is just war file that schedules all things

- java -jar jenkins.war



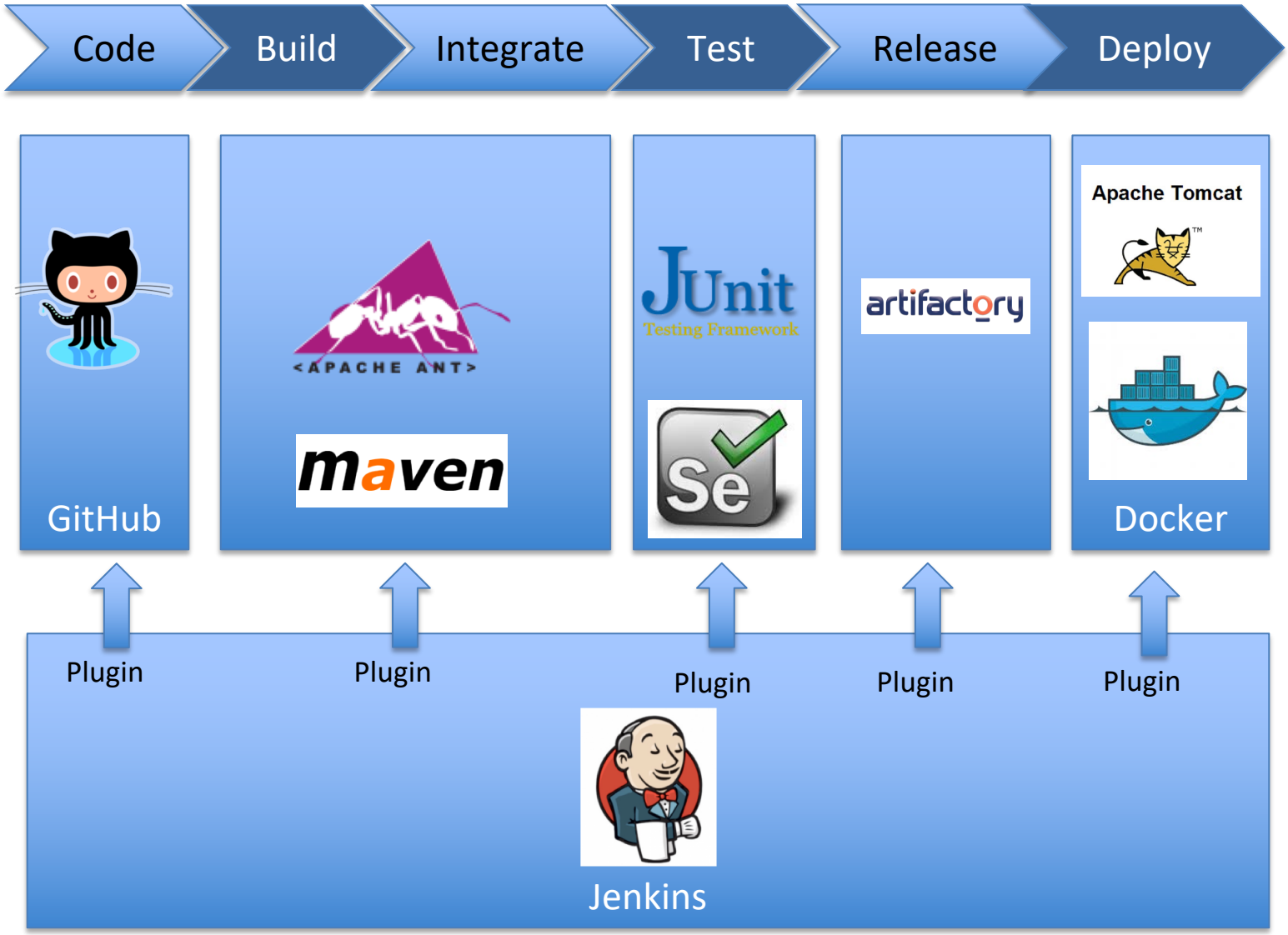

http://pages.zeroturnaround.com/Java-Tools-Technologies.html

# Why is Jenkins?

Why Jenkins?

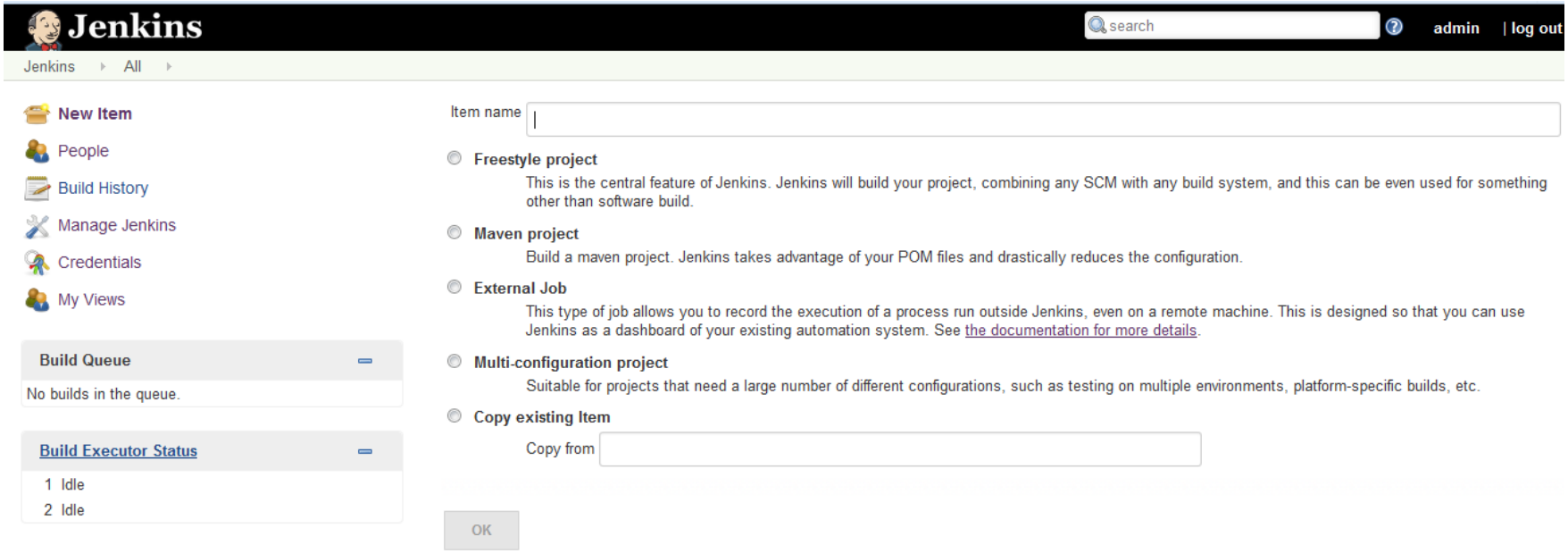
- **Easy installation**
- **Easy configuration**
- **Rich plugin ecosystem**
- **Extensibility**
- **Distributed builds**

# How Jenkins enables CI/CD



# What is a Job/Item ?

- Basic currency of the Continuous Integration server
- Usually contains steps to compile, test, package, deploy your app and report
- Leverages plugins
- Establishes an Upstream-Downstream Relationship with other Jobs
- Can be scheduled or run manually
- Can run freestyle Bash commands. Anything you run in Linux you can run in Jenkins

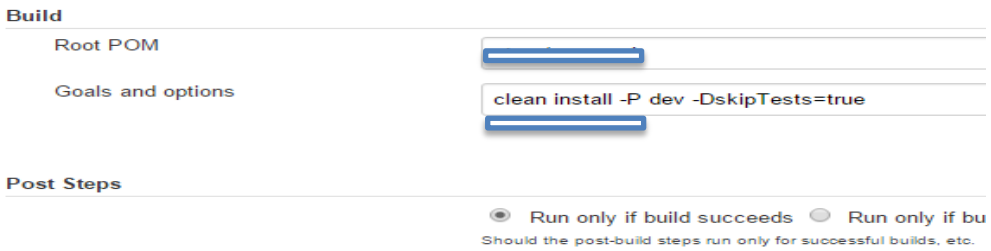


# Anatomy of a Job

Project Name/Description


Plugin


Build and Post Build Steps




Job chaining


Notification

# Plugins

**Customize Jenkins**

Choose from **1103** plugins to customize Jenkins exactly to your needs.



- Help Jenkins to interface with other tools to make CI possible
- Can be installed/updated using Jenkins web UI or by hand
- Developers can create their own plugins if needed (rarely if ever)

https://wiki.jenkins-ci.org/display/JENKINS/Plugins

# Github Webhooks

- Webhooks provide a way for notifications to be delivered to an external web server whenever certain actions occur on a repository or organization.
  - A repository is pushed to
  - A pull request is opened
  - A GitHub Pages site is built
  - A new member is added to a team

- Trigger a build when a change is pushed to GitHub

- This feature enables builds after post-receive hooks in your GitHub repositories
  - Manual
  - Automatic

https://wiki.jenkins-ci.org/display/JENKINS/GitHub+Plugin
https://help.github.com/articles/about-webhooks/

# Job Notifications

- Slack
- E-mail
- PagerDuty


pagerduty

slack

# Bringing them together

Dev pushes code to Github

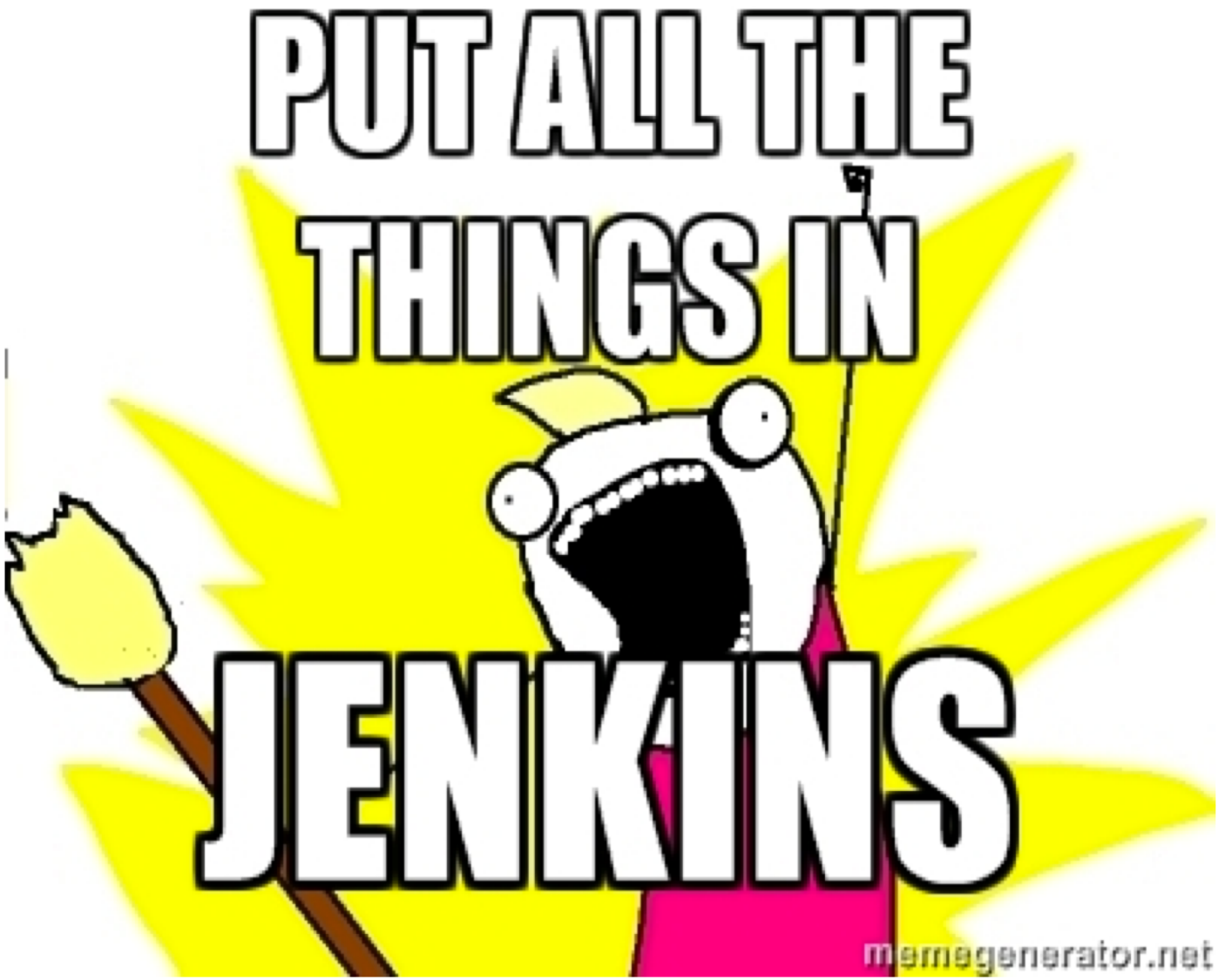
Github Triggers Jenkins to build

Build Passes!

Jenkins deploys code to Servers

Jenkins lets the team know

# THE END



# More Information

https://jenkins-ci.org/

https://dzone.com/refcardz/jenkins-paas

https://www.cloudbees.com/jenkins/about

https://yow.eventer.com/events/1004/talks/1062