



let's move  
the **java** world

**Developing for multi-component  
environments**

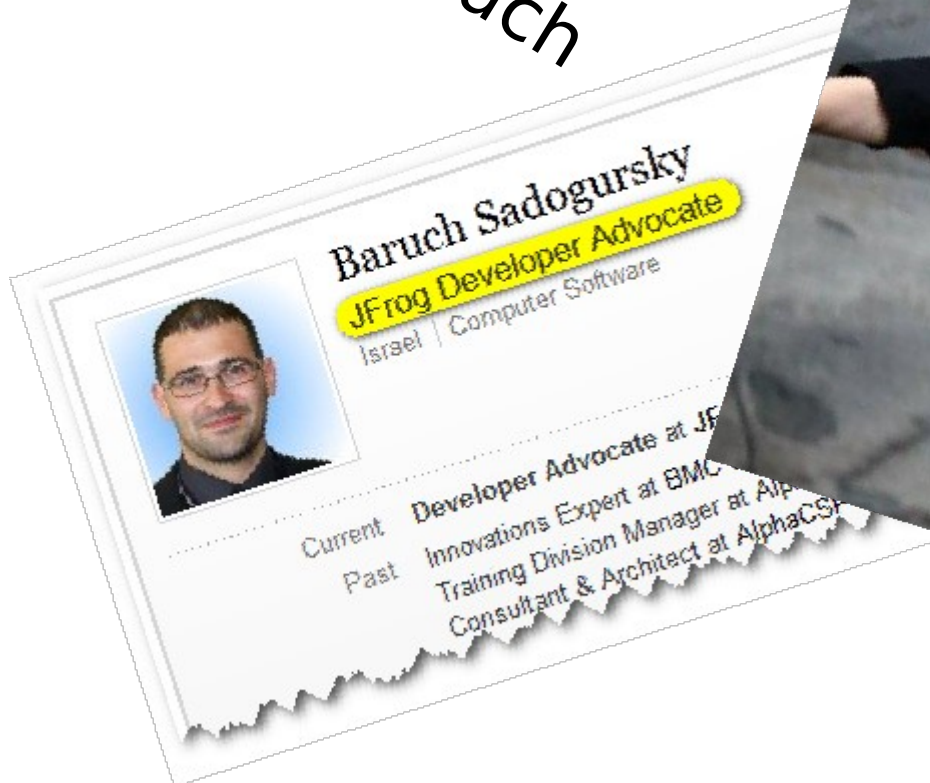
**while keeping your sanity**



# Who's talking?

github.com/jbaruch

@jbaruch



# Developer who?

Job Description:

1. Hang out  
with DevOps

2. Talk about it

# Developer who?



# Agenda

- Mostly, Boromir memes.


# Agenda

Mostly, Boromir memes.





7 movie references


# We Make Bintray



Search Bintray

 jbaruch

0  Get Help | Logout

 jfrog / artifactory / artifactory


<http://dl.bintray.com/content/jfrog/artifactory>


Owned by JFrog


★★★★★ (5)


Reviews 0


Watchers 3


 Un-watch

 Message

 Link

 Report






Artifactory open source standalone distributions

[CI](#) | [bamboo](#) | [maven](#) | [manager](#) | [repository](#) | [jenkins](#) | [Artifactory](#) | [gradle](#) | [ivy](#) | [TeamCity](#)

General

Read More

Statistics

 Set me up!

## About This Package

Website	<a href="http://www.jfrog.com">http://www.jfrog.com</a>
Issue Tracker	<a href="https://issues.jfrog.org/jira/browse/rtfact">https://issues.jfrog.org/jira/browse/rtfact</a>
VCS	<a href="https://subversion.jfrog.org/artifactory/public/trunk">https://subversion.jfrog.org/artifactory/public/trunk</a>
Licenses	LGPL-3.0 

## Downloads (v 2.6.6)

 [artifactory-2.6.6.zip](#)  
Latest release  
sha1: b7f449f752f337502f9fd8799caf38e548128b43

## Versions

[2.6.6](#)[2.6.5](#)[2.6.4](#)[2.6.3](#)[2.6.2](#)[View All](#)

# We Make Bintray

Firefox


https://bintray.com/version/show/general/aalmiray/Griffon


**SaaS Webapp**

Search Bintray

jbaruch


Get Help | Logout

 **Bintray** Beta

 **aalmiray / Griffon / griffon / 1.2.0**

<http://dl.bintray.com/content/aalmiray/Griffon>

Owned by **Andres Almira...** ★★★★★ (1) Reviews 0 Watchers 1 Watch





 This version has no description.

groovy | griffon | swing | java |

General Read More Statistics Files

Set me up!

Directory: /

Name	Updated	Size
.		
 <a href="#">griffon-1.2.0-1.noarch.rpm</a>	1 month ago	34.9 MB
 <a href="#">griffon-1.2.0-bin.dmg</a>	1 month ago	35.4 MB
 <a href="#">griffon-1.2.0-bin.tgz</a>	1 month ago	33.2 MB
 <a href="#">griffon-1.2.0-bin.zip</a>	1 month ago	36.5 MB



# We Make Bintray

The screenshot shows the Bintray website interface. At the top, there's a navigation bar with the Bintray logo, a search bar, and user information for 'jbaruch'. Below this, the package details for 'aalmiray / Griffon / griffon / 1.2.0' are displayed. The package is owned by 'Andres Almira...' and has 1 review. A callout box labeled 'SaaS Webapp' points to the top navigation area. Below the package details, there's a section for binaries. A callout box labeled 'Binaries storage' points to the list of binaries. The binaries list includes files like 'griffon-1.2.0-1.noarch.rpm', 'griffon-1.2.0-bin.dmg', 'griffon-1.2.0-bin.tgz', and 'griffon-1.2.0-bin.zip'.

**SaaS Webapp**

**Binaries storage**

Name	Updated	Size
.		
griffon-1.2.0-1.noarch.rpm	1 month ago	34.9 MB
griffon-1.2.0-bin.dmg	1 month ago	35.4 MB
griffon-1.2.0-bin.tgz	1 month ago	33.2 MB
griffon-1.2.0-bin.zip	1 month ago	36.5 MB

# We Make Bintray

The screenshot shows the Bintray website interface. At the top, there's a navigation bar with the Bintray logo, a search bar, and user information for 'jbaruch'. A yellow callout box labeled 'SaaS Webapp' points to the top navigation area.

The main content area displays the package 'aalmiray / Griffon / griffon / 1.2.0'. It includes the owner's name 'Andres Almira...', a star rating, and a 'Watch' button. A yellow callout box labeled 'Metadata' points to the package details section.

Below the package details, there's a 'Directory:' section with a table of binaries. A yellow callout box labeled 'Binaries storage' points to the table.

Name	Updated	Size
.		
<a href="#">griffon-1.2.0-1.noarch.rpm</a>	1 month ago	34.9 MB
<a href="#">griffon-1.2.0-bin.dmg</a>	1 month ago	35.4 MB
<a href="#">griffon-1.2.0-bin.tgz</a>	1 month ago	33.2 MB
<a href="#">griffon-1.2.0-bin.zip</a>	1 month ago	36.5 MB

# We Make Bintray

The screenshot shows the Bintray website interface. The browser address bar displays `https://bintray.com/version/show/general/aalmiray/Griffon`. The page features a blue header with the Bintray logo, a search bar, and user information for 'jbaruch'. The main content area shows the version details for 'aalmiray / Griffon', including ownership by 'Andres Almira...', a star rating, and a 'Watch' button. A red callout labeled 'SaaS Webapp' points to the search bar. Another red callout labeled 'Full text search' points to the version details. A third red callout labeled 'Metadata' points to the version description area. A fourth red callout labeled 'Binaries storage' points to the table of binaries. The table lists four binaries: 'griffon-1.2.0-1.noarch.rpm', 'griffon-1.2.0-bin.dmg', 'griffon-1.2.0-bin.tgz', and 'griffon-1.2.0-bin.zip', each with an 'Updated' date of '1 month ago' and a 'Size' in MB.

**SaaS Webapp**

**Full text search**

**Metadata**

**Binaries storage**

Name	Updated	Size
.		
<a href="#">griffon-1.2.0-1.noarch.rpm</a>	1 month ago	34.9 MB
<a href="#">griffon-1.2.0-bin.dmg</a>	1 month ago	35.4 MB
<a href="#">griffon-1.2.0-bin.tgz</a>	1 month ago	33.2 MB
<a href="#">griffon-1.2.0-bin.zip</a>	1 month ago	36.5 MB

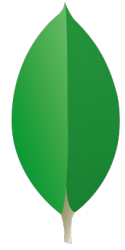
We love our tools  
(and frameworks)







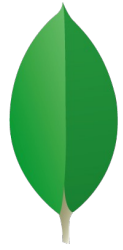




mongoDB





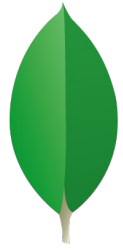


mongoDB



EHCACHE





mongoDB

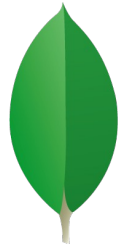


EHCache



Sphinx



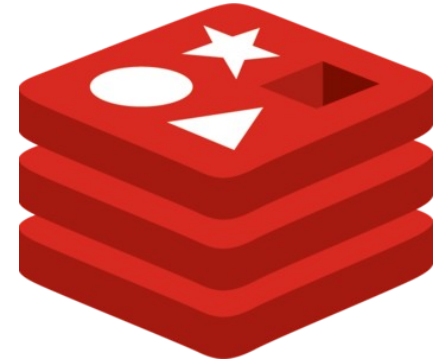


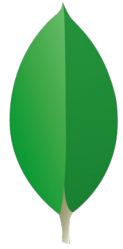
mongoDB



EHCACHE



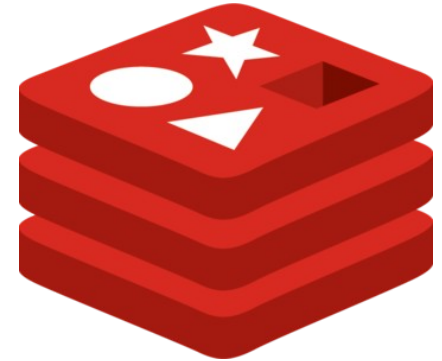




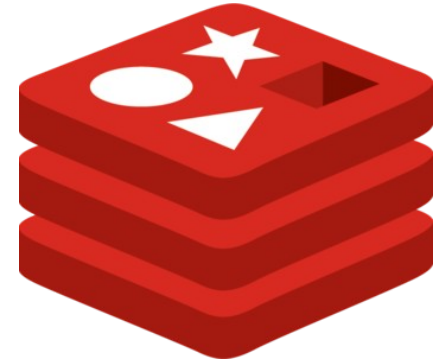
mongoDB



EHCache

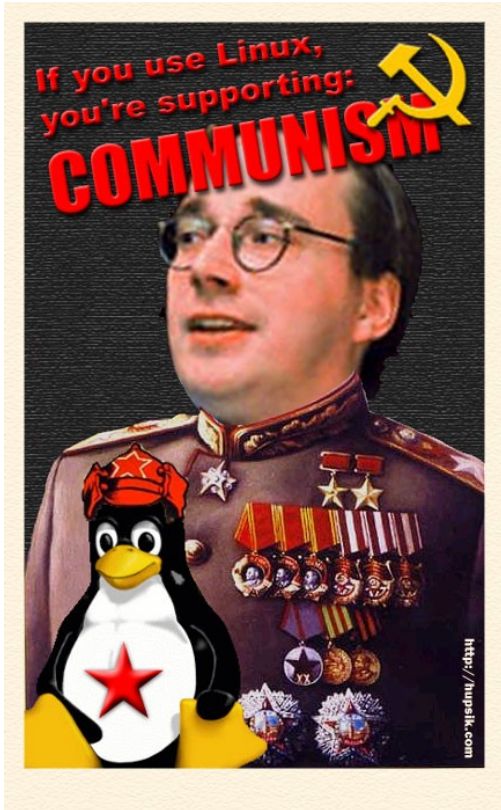


**Apache**



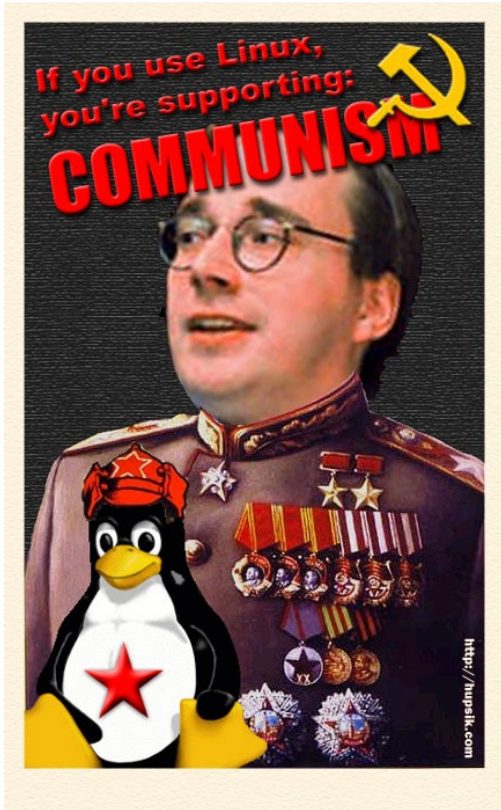
We are liberal

# We are liberal

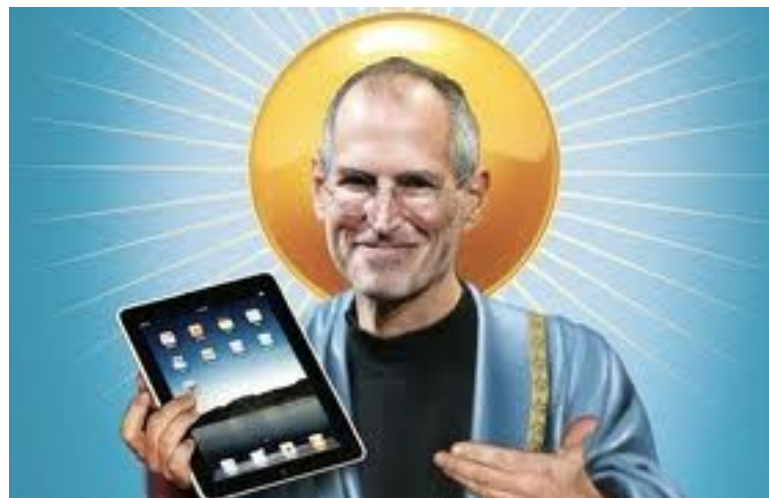
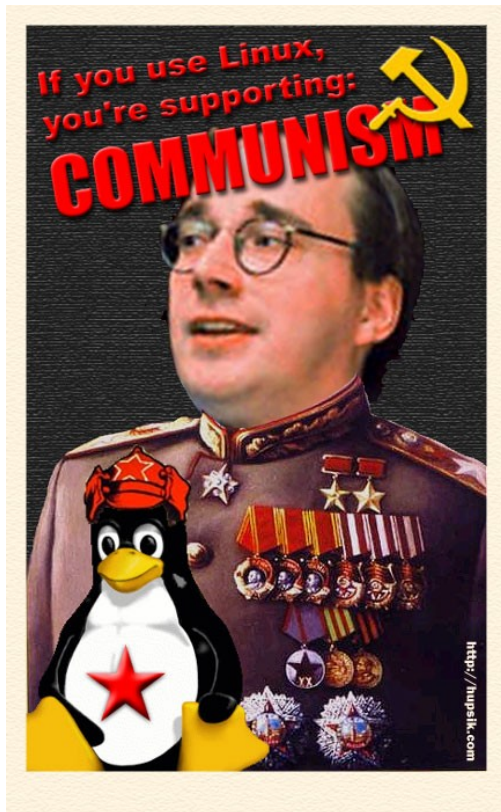




# We are liberal

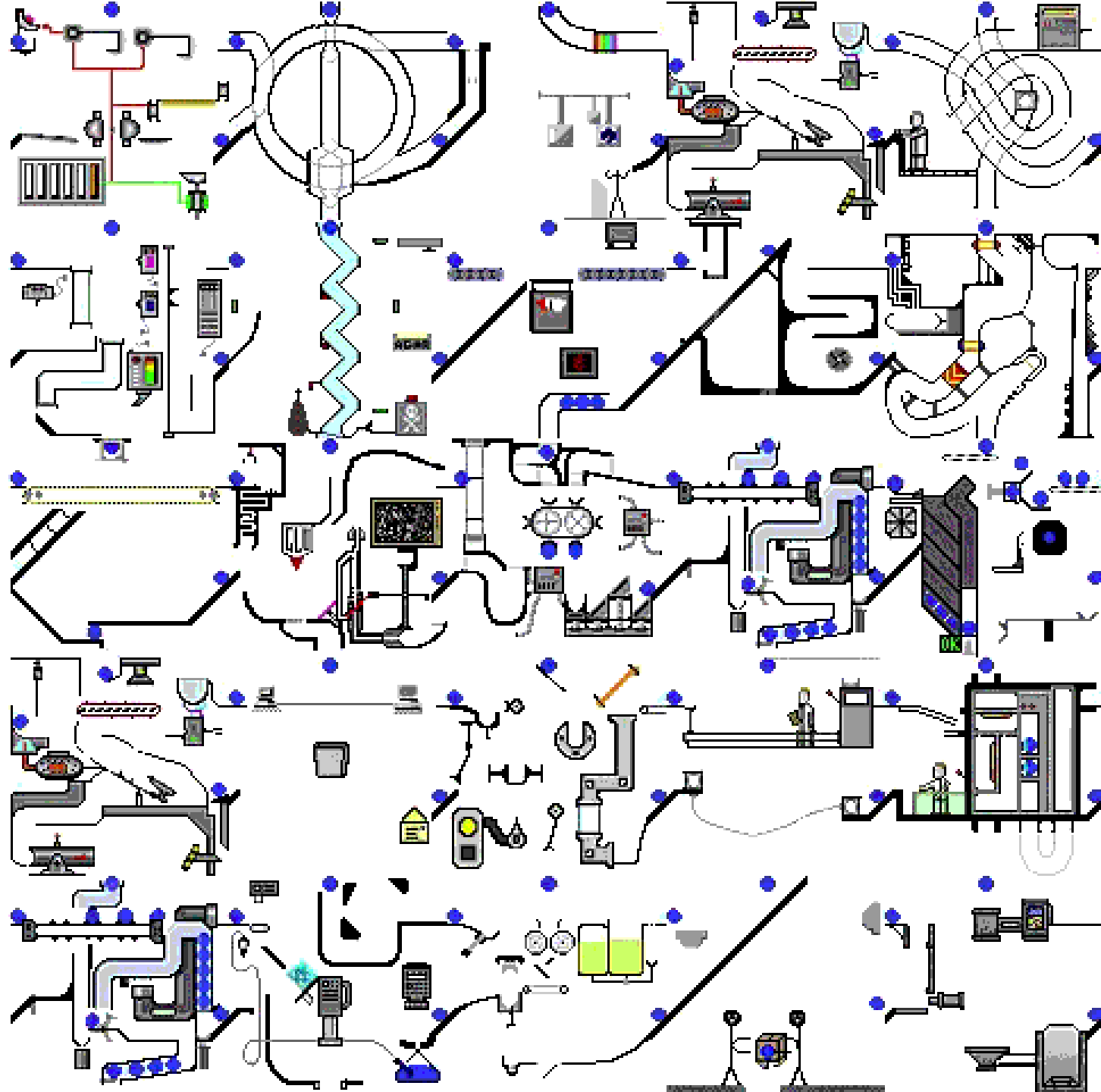


# We are liberal





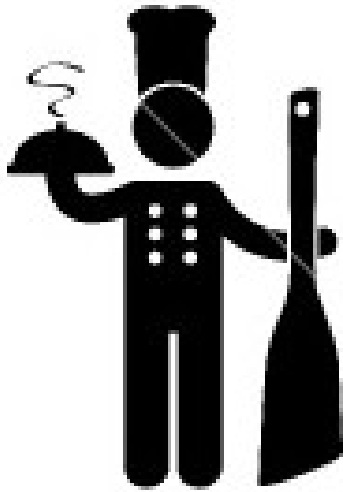
# The Solution



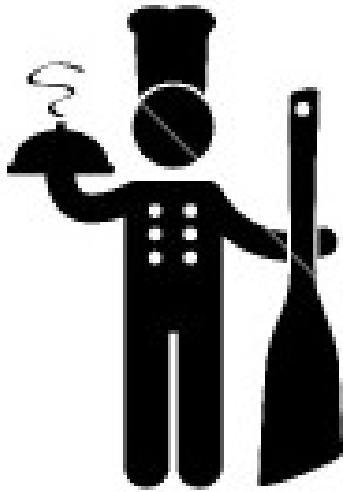
# The Solution



# The Solution



# The Solution





# The Solution



# Chef What?



# Opscode Chef



**Chef is a systems integration framework, built to bring the benefits of configuration management to your entire infrastructure.**

# Opscode Chef

## About Cookbooks

---

A cookbook is the fundamental **unit of configuration** and policy distribution in Chef. Each cookbook defines **a scenario**, such as everything needed to install and configure MySQL, and then it contains all of the components that are required to support that scenario, including:

# Opscode Chef

## About Cookbooks

---

A cookbook is the fundamental **unit of configuration** and policy distribution in Chef. Each cookbook defines **a scenario**, such as everything needed to install and configure **MySQL**, and then it contains all of the components that are required to support that scenario, including:

# Opscode Chef

```
115
116   if platform_family? 'windows'
117     require 'win32/service'
118
119     windows_path node['mysql']['bin_dir'] do
120       action :add
121     end
122
123     windows_batch "install mysql service" do
124       command "\"#{node['mysql']['bin_dir']}\\mysql.exe\" --install #{node['mysql']['service_name']}"
125       not_if { Win32::Service.exists?(node['mysql']['service_name']) }
126     end
127   end
128
```

# The Solution



# The Solution





# Vagrant Who?



# Vagrant

## Getting Started with Vagrant

Vagrant uses [Oracle's VirtualBox](#) to build configurable, lightweight, and portable virtual machines dynamically.

# Vagrant

## Configuring Chef and the Vagrant

Since a tutorial on how to use Chef is out of scope for this getting started guide, I've prepackaged the cookbooks for you for provisioning. You just have to configure your Vagrantfile to point to them:

```
Vagrant::Config.run do |config|  
  config.vm.box = "lucid32"  
  
  # Enable and configure the chef solo provisioner  
  config.vm.provision :chef_solo do |chef|  
    # We're going to download our cookbooks from the web  
    chef.recipe_url = "http://files.vagrantup.com/getting_started/cookbooks.tar.gz"  
  
    # Tell chef what recipe to run. In this case, the `vagrant_main` recipe  
    # does all the magic.  
    chef.add_recipe("vagrant_main")  
  end  
end
```

# Vagrant

## Configuring Chef and the Vagrant

Since a tutorial on how to use Chef is out of scope for this getting started guide, I've prepackaged the cookbooks for you for provisioning. You just have to configure your Vagrantfile to point to them:

```
Vagrant::Config.run do |config|  
  config.vm.box = "lucid32"  
  
  # Enable and configure the chef solo provisioner  
  config.vm.provision :chef_solo do |chef|  
    # We're going to download our cookbooks from the web  
    chef.recipe_url = "http://files.vagrantup.com/getting_started/cookbooks.tar.gz"  
  
    # Tell chef what recipe to run. In this case, the `vagrant_main` recipe  
    # does all the magic.  
    chef.add_recipe("vagrant_main")  
  end  
end
```

# Vagrant

## Configuring Chef and the Vagrant

Since a tutorial on how to use Chef is out of scope for this getting started guide, I've prepackaged the cookbooks for you for provisioning. You just have to configure your Vagrantfile to point to them:

```
Vagrant::Config.run do |config|  
  config.vm.box = "lucid32"  
  
  # Enable and configure the chef solo provisioner  
  config.vm.provision :chef_solo do |chef|  
    # We're going to download our cookbooks from the web  
    chef.recipe_url = "http://files.vagrantup.com/getting_started/cookbooks.tar.gz"  
  
    # Tell chef what recipe to run. In this case, the `vagrant_main` recipe  
    # does all the magic.  
    chef.add_recipe("vagrant_main")  
  end  
end
```

# Vagrant

## Configuring Chef and the Vagrant

Since a tutorial on how to use Chef is out of scope for this getting started guide, I've prepackaged the cookbooks for you for provisioning. You just have to configure your Vagrantfile to point to them:

```
Vagrant::Config.run do |config|  
  config.vm.box = "lucid32"  
  
  # Enable and configure the chef solo provisioner  
  config.vm.provision :chef_solo do |chef|  
    # We're going to download our cookbooks from the web  
    chef.recipe_url = "http://files.vagrantup.com/getting_started/cookbooks.tar.gz"  
  
    # Tell chef what recipe to run. In this case, the `vagrant_main` recipe  
    # does all the magic.  
    chef.add_recipe("vagrant_main")  
  end  
end
```

# The Solution



# The Solution





# Development



# Development



Chef

Chef Solo allows you to run Chef Cookbooks in the absence of a Chef Server.

To provide the raw execution of cookbook recipes, the complete cookbook needs to be present on the disk.

# Development



Chef

**Chef Solo** allows you to run Chef Cookbooks in the absence of a Chef Server.

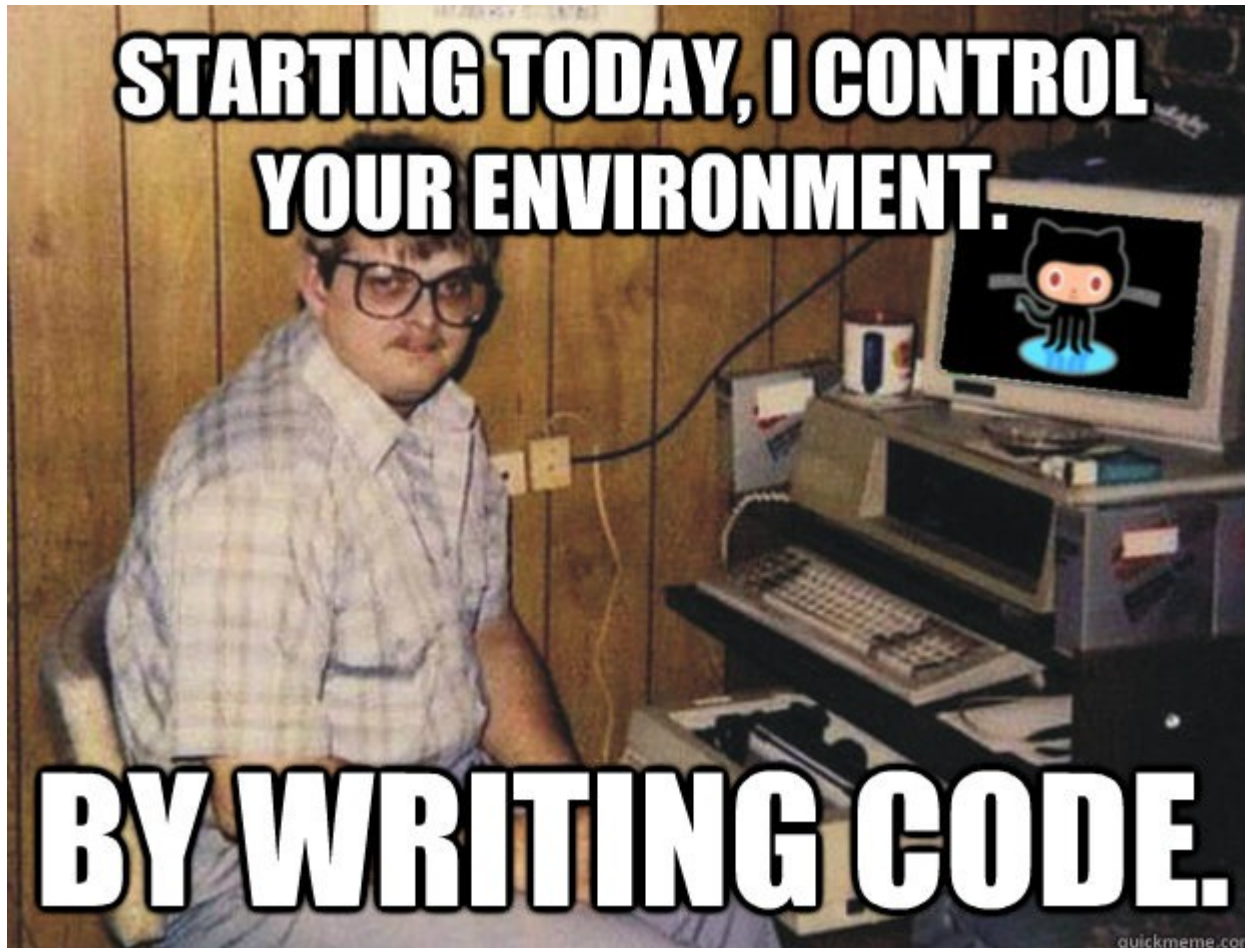
To provide the raw execution of cookbook recipes, the complete **cookbook needs to be present on the disk.**

# Development



```
cookbooks>ls
apache2      erlang      mysql       searchconf
apt          haproxy    nagios      selinux
ark          incapsula  network-host serverdensity
artifactory  install_from network-keys simple_iptables
aws         java       nginx       sshid
basic-pack  jfrog     nrpe       system_services
bluepill    libs      ohai       timezone
bt_install  loggly    openssl    tomcat7
build-essential logicmonitor postgresql  update_databag
chef_handler logrotate  redis      user
couchdb     memcached redis-0.0.3 vagrant_main
deploy      metachef  redisconf  windows
devnullsmtp mongodb   route53    yum
elasticsearch mount     runit      zookeeper
```

The Ops are part of the DevOps







# 1. Vagrant boots centos on virtualbox



1. Vagrant boots centos on virtualbox
2. Chef installs all db and service rpms from private YUM repo



1. Vagrant boots centos on virtualbox
2. Chef installs all db and service rpms from private YUM repo
3. Profit!

# Integration



```
cookbooks>ls
apache2      erlang      mysql       search
apt          haproxy     nagios      seli
ark          incapsula   network-host serv
artifactory  install_from network-keys simp
aws          java        nrpe        sshi
basic-pack   jfrog       ohai        syst
bluepill     libs        openssl     time
bt_install   loggly      postgresql  tomca
build-essential logicmonitor postgresql  upda
chef_handler logrotate   redis       user
```

# As Easy as...

## Jenkins

Jenkins bintray-releases #2184



[Back to Project](#)



[Status](#)



[Changes](#)



[Console Output \[raw\]](#)



[View Build Information](#)



[Polling Log](#)



[Injected Environment Variables](#)



[Git Build Data:bintray](#)



## Console Output

Skipping 23,946 KB.. [Full Log](#)

```
+ chmod +x bintray/jenkins-scripts/vagrant_starter.sh
+ ./bintray/jenkins-scripts/vagrant_starter.sh /media/ramdisk
Checkout directory is '/media/ramdisk/jenkins-workspace' -
Current cookbook revision is a28a1f2b0c5ca51d4b64934fd156d20
Last documented cookbook revision is a28a1f2b0c5ca51d4b64934
Documenting current cookbooks revision
Checking to see if the last 'vagrant up' log contains any errors
Raising Vagrant machine and documenting process
[default] VM already created. Booting if it's not already running
```



# Going to Production...







# Production – cloud all the way



Chef

Chef Solo allows you to run Chef Cookbooks in the absence of a Chef Server.

To provide the raw execution of cookbook recipes, the complete **cookbook needs to be present on the disk.**

# Production – cloud all the way



Chef

Chef Solo allows you to run Chef [Cookbooks](#) in the absence of a [Chef Server](#).

To provide the raw execution of cookbook recipes, the complete [cookbook needs to be present on the disk.](#)



Production – cloud all the way



# Production – cloud all the way

jclouds™



# Production – cloud all the way

jclouds™

SOFTLAYER

CentOS



mongoHQ



Chef



Cloudfant

The Do More Data Layer

The good



# The good

1. Easy setup

# The good

1. Easy setup

2. Same setup ( inc. Ci )



# The good

1. Easy setup

2. Same setup ( inc. Ci )

3. changes provisioning

# The good

1. Easy setup
2. Same setup ( inc. Ci )
3. changes provisioning
4. Kill instead of fix



# The Bad

# The Bad

1. Chef solo != chef server



# The Bad

1. Chef solo != chef server

2. Black box magic for devs

---

# The Ugly



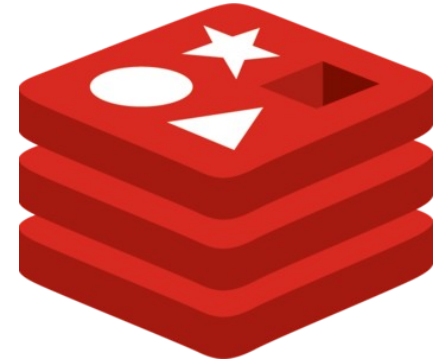
# The Ugly





# Demo Time!





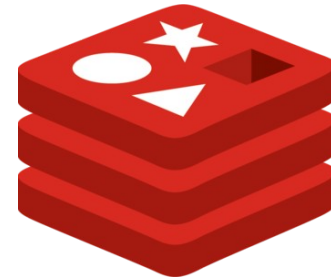




mongoDB



EHCache



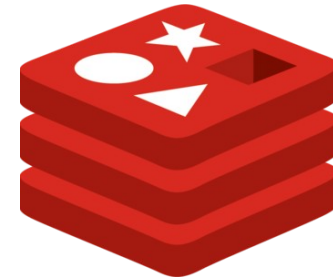
NGINX



mongoDB



EHCache



NGINX

No, thank you!

