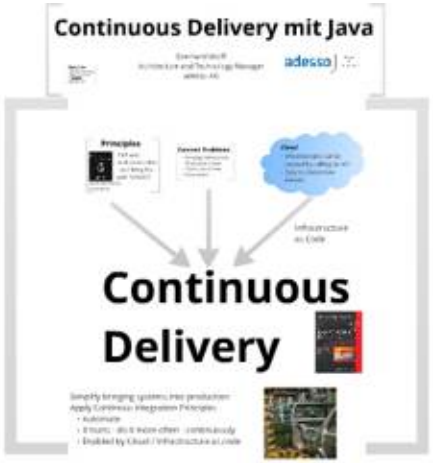
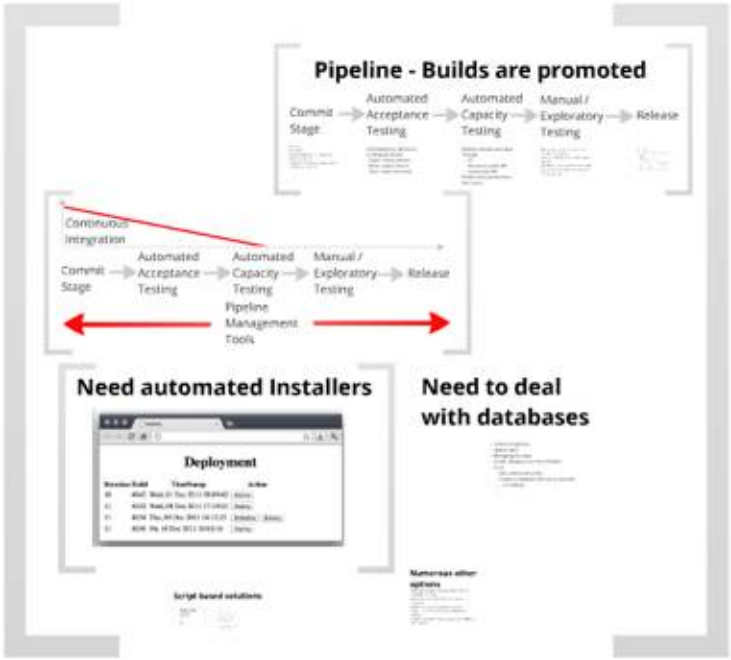




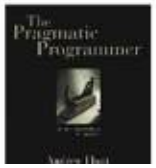
http://bit.ly/opscode-opentraining

Continuous Delivery mit Java

Eberhard Wolff
Architecture and Technology Manager
adesso AG

adesso business. people. technology.

Principles



If it hurts - do it more often - and bring the pain forward!

Current Problems

- Bringing Software into Production is hard
- Takes a lot of time
- Error prone

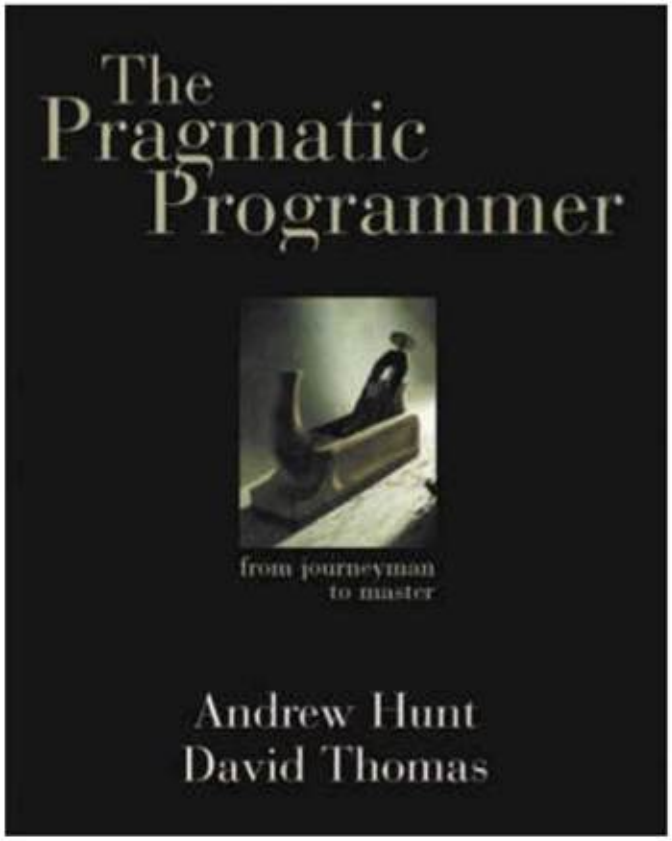
Cloud

- Infrastructure can be created by calling an API
- Easy to create new

About me

- Eberhard Wolff
- Architecture & Technology Manager at adesso
- adesso is a leading IT consultancy in Germany
- Speaker
- Author (e.g. first German Spring book)
- Blog: http://ewolff.com
- Twitter: @ewolff
- http://slideshare.com/ewolff
- eberhard.wolff@adesso.de

Principles

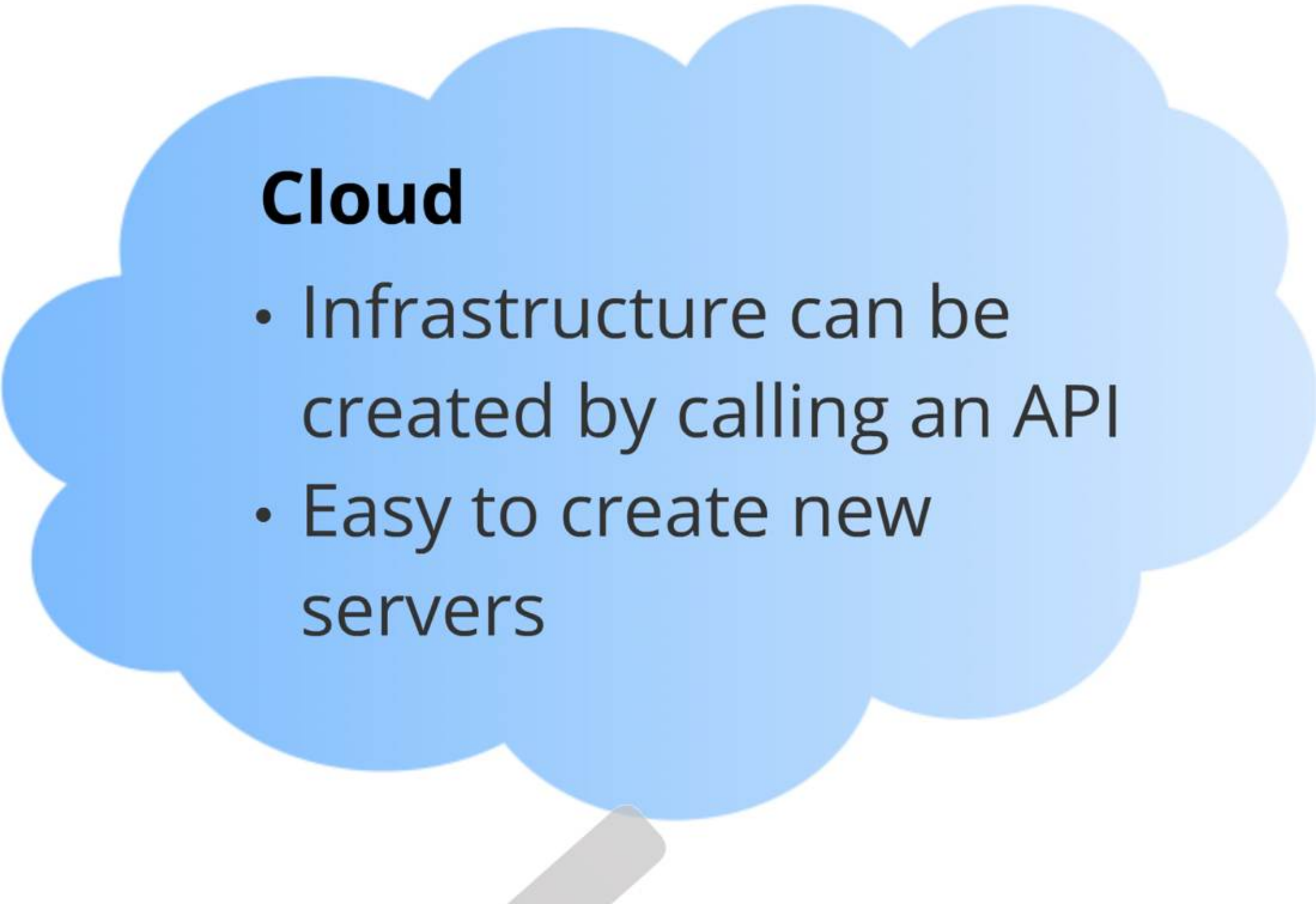


If it hurts -
do it more often
- and bring the pain forward!

Don’t Use Manual Procedures

Current Problems

- Bringing Software into Production is hard
- Takes a lot of time
- Error prone


Cloud
• Infrastructure can be created by calling an API
• Easy to create new servers

Infrastructure as Code

Principles

If it hurts - do it more often - and bring the pain forward!

Don't Use Manual Procedures

Current Problems

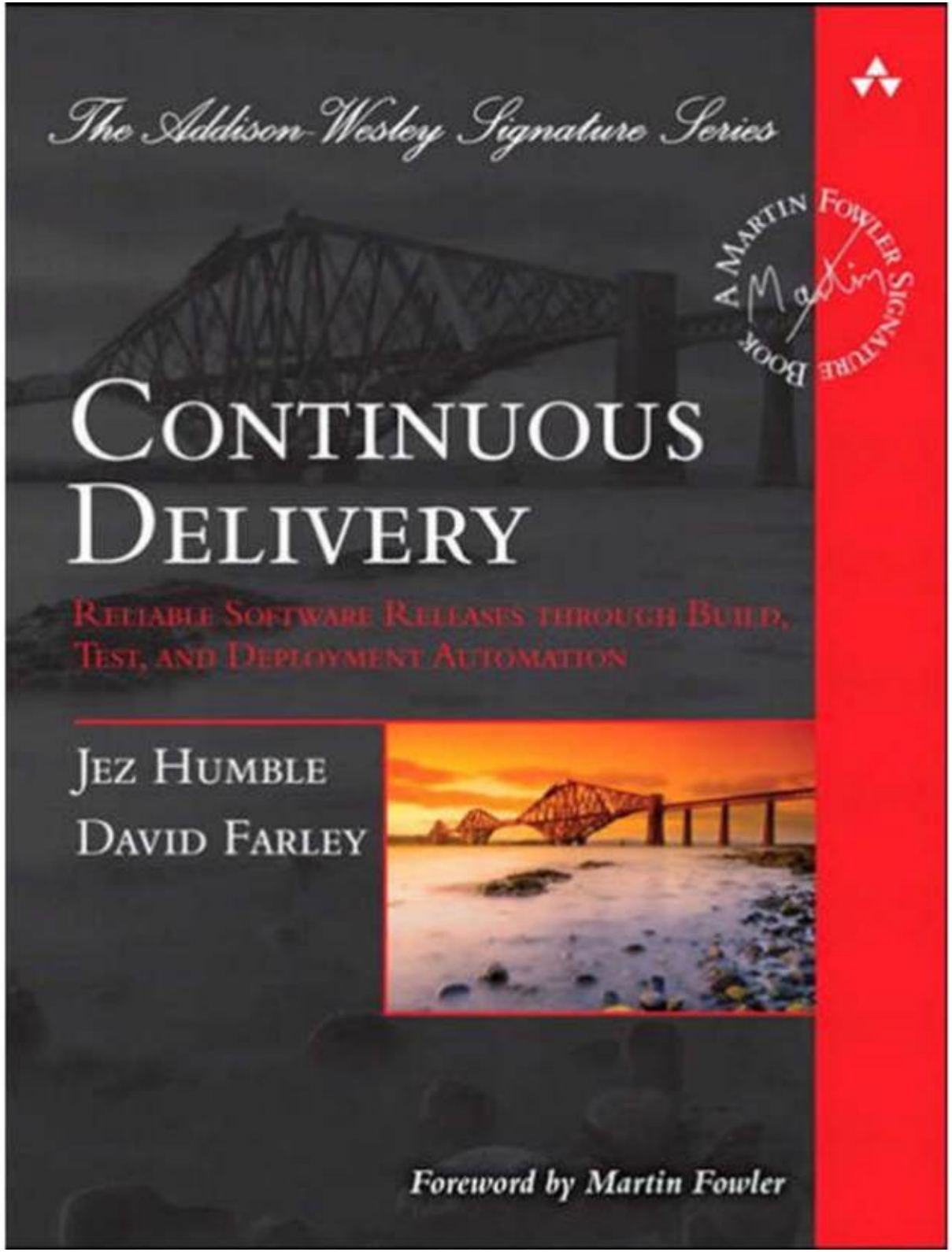
- Bringing Software into Production is hard
- Takes a lot of time
- Error prone

Cloud

- Infrastructure can be created by calling an API
- Easy to create new servers

Infrastructure as Code

Continuous Delivery


The Addison-Wesley Signature Series
A Martin Fowler Signature Book
Continuous Delivery
Reliable Software Releases through Build, Test, and Deployment Automation
Jez Humble
David Farley
Foreword by Martin Fowler

Delivery

Simplify bringing systems into production

Apply Continous Integration Principles

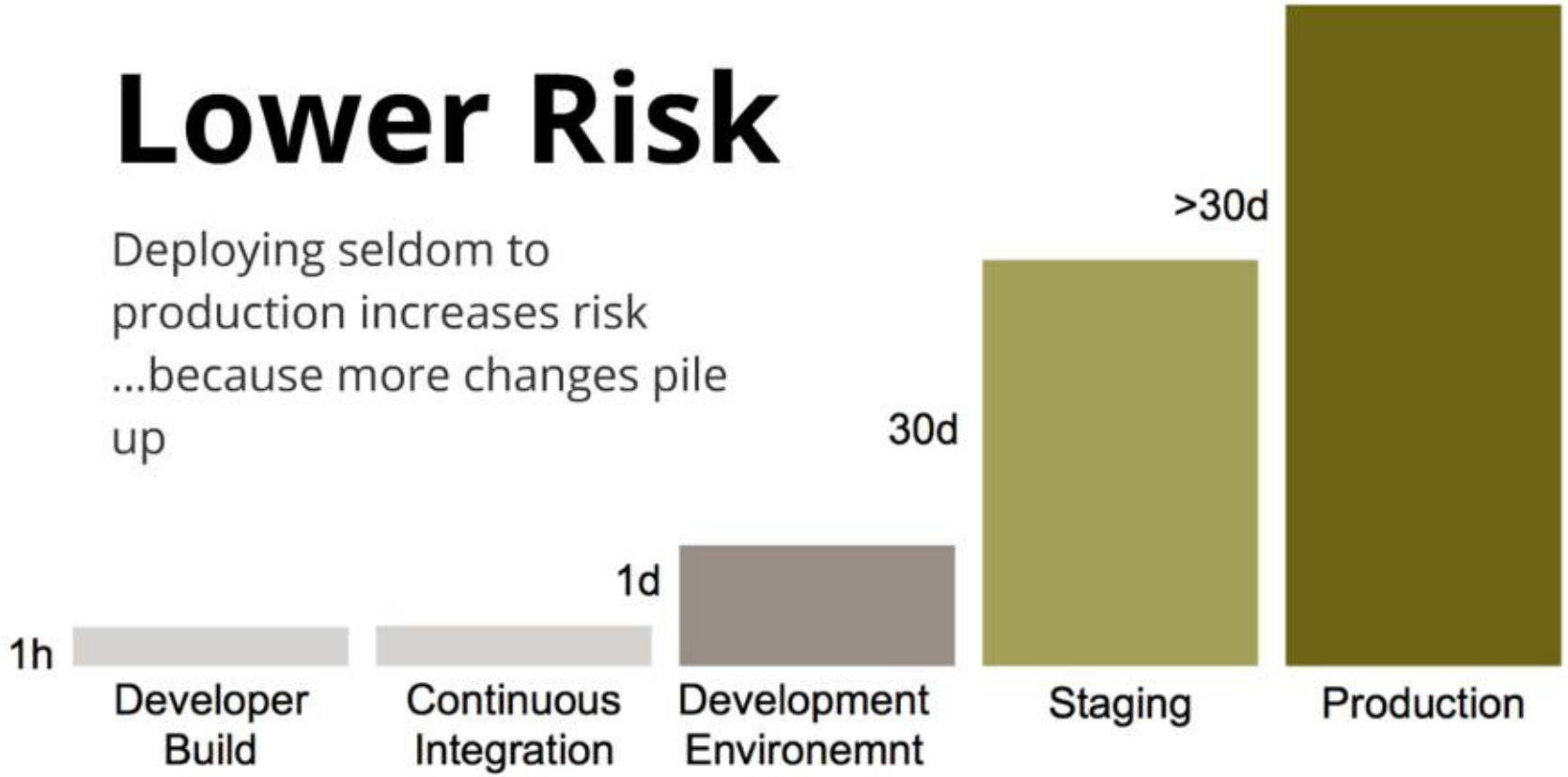
- Automate
- It hurts - do it more often - continuously
- Enabled by Cloud / Infrastructure as code

Faster Time to Market

Lower Risk

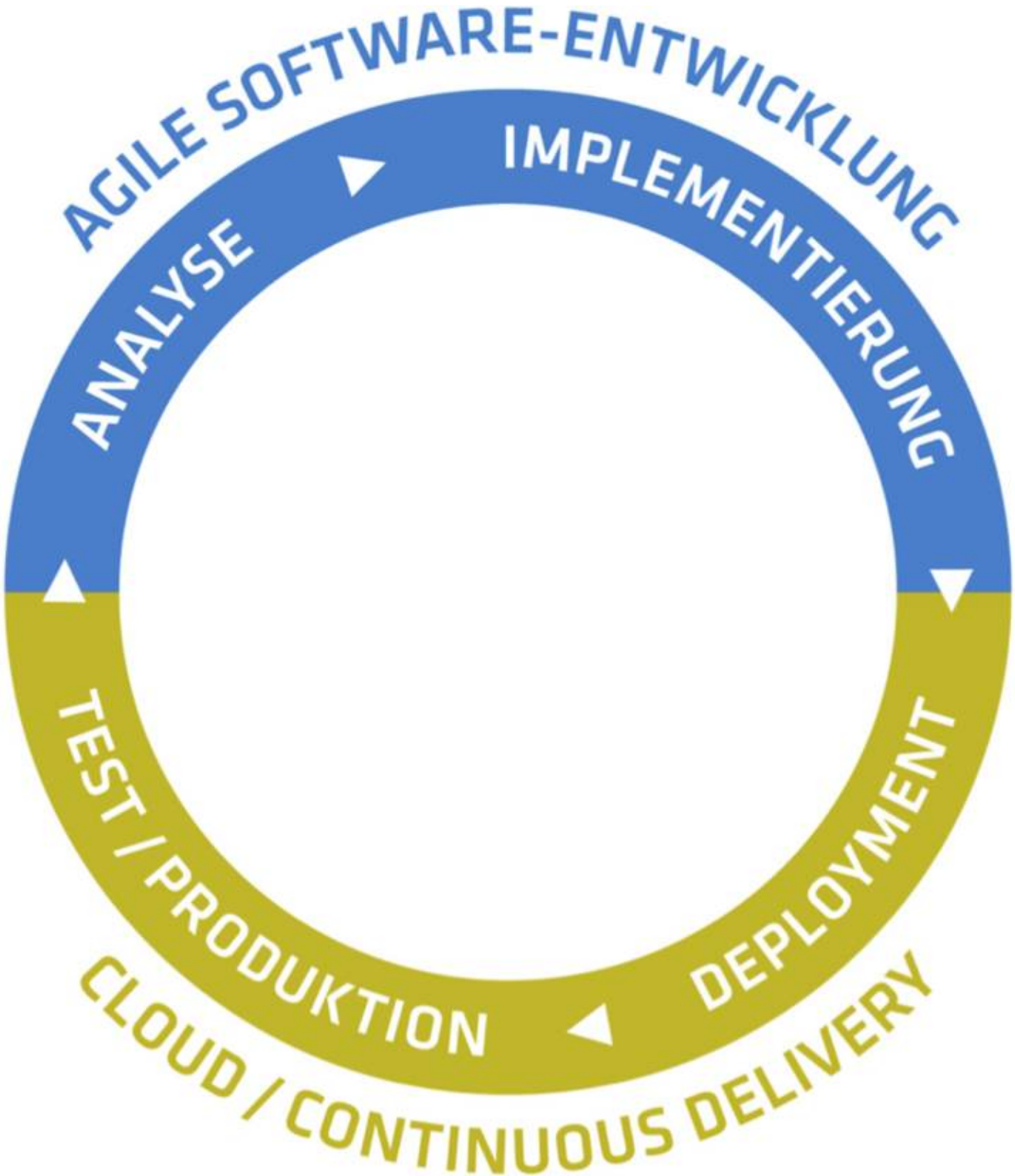
Deploying seldom to production increases risk

...because more changes pile up

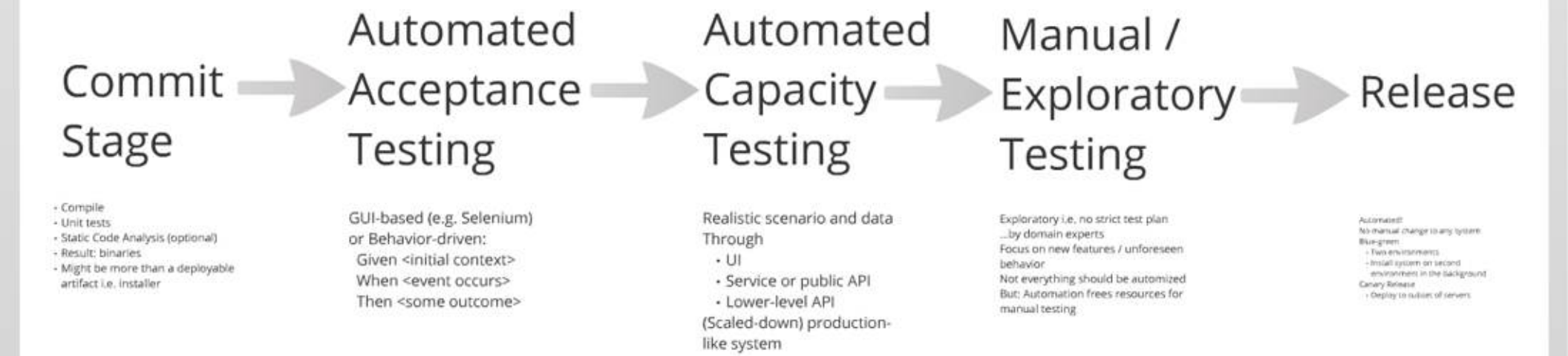


Higher Reliability

- Deployment automated
- ...and therefore reproducible


AGILE SOFTWARE-ENTWICKLUNG
ANALYSE
IMPLEMENTIERUNG
TEST / PRODUKTION
DEPLOYMENT
CLOUD / CONTINUOUS DELIVERY

Pipeline - Builds are promoted



Commit Stage

- Compile
- Unit tests
- Static Code Analysis (optional)
- Result: binaries
- Might be more than a deployable artifact i.e. installer

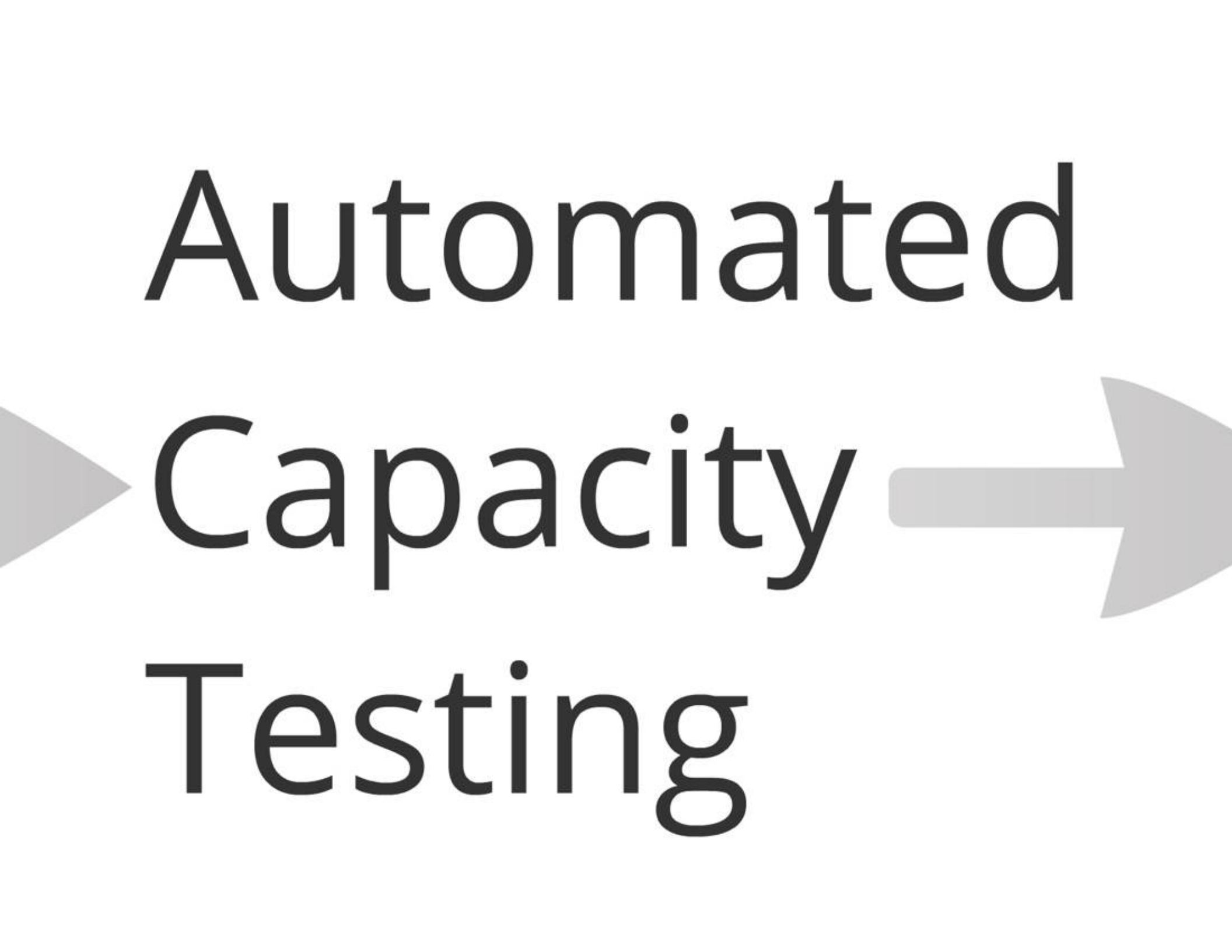
Automated Acceptance Testing

GUI-based (e.g. Selenium)
or Behavior-driven:

Given <initial context>

When <event occurs>

Then <some outcome>


Automated
Capacity
Testing

Realistic scenario and data

Through

- UI
- Service or public API
- Lower-level API

(Scaled-down) production-like system

Manual / Exploratory Testing

Exploratory i.e. no strict test plan

...by domain experts

Focus on new features / unforeseen behavior

Not everything should be automized

But: Automation frees resources for manual testing

Release

Automated!

No manual change to any system

Blue-green

- Two environments
- Install system on second environment in the background

Canary Release

- Deploy to subset of servers

Commit Stage

- Compile
- Unit tests
- Static Code Analysis (optional)
- Result: binaries
- Might be more than a deployable artifact i.e. installer

Acceptance Testing

GUI-based (e.g. Selenium)
or Behavior-driven:
Given <initial context>
When <event occurs>
Then <some outcome>

Continuous Integration

Commit Stage → Automated Acceptance Testing → Automated Capacity Testing → Manual / Exploratory Testing → Release

Pipeline Management Tools

Need automated Installers

Need with

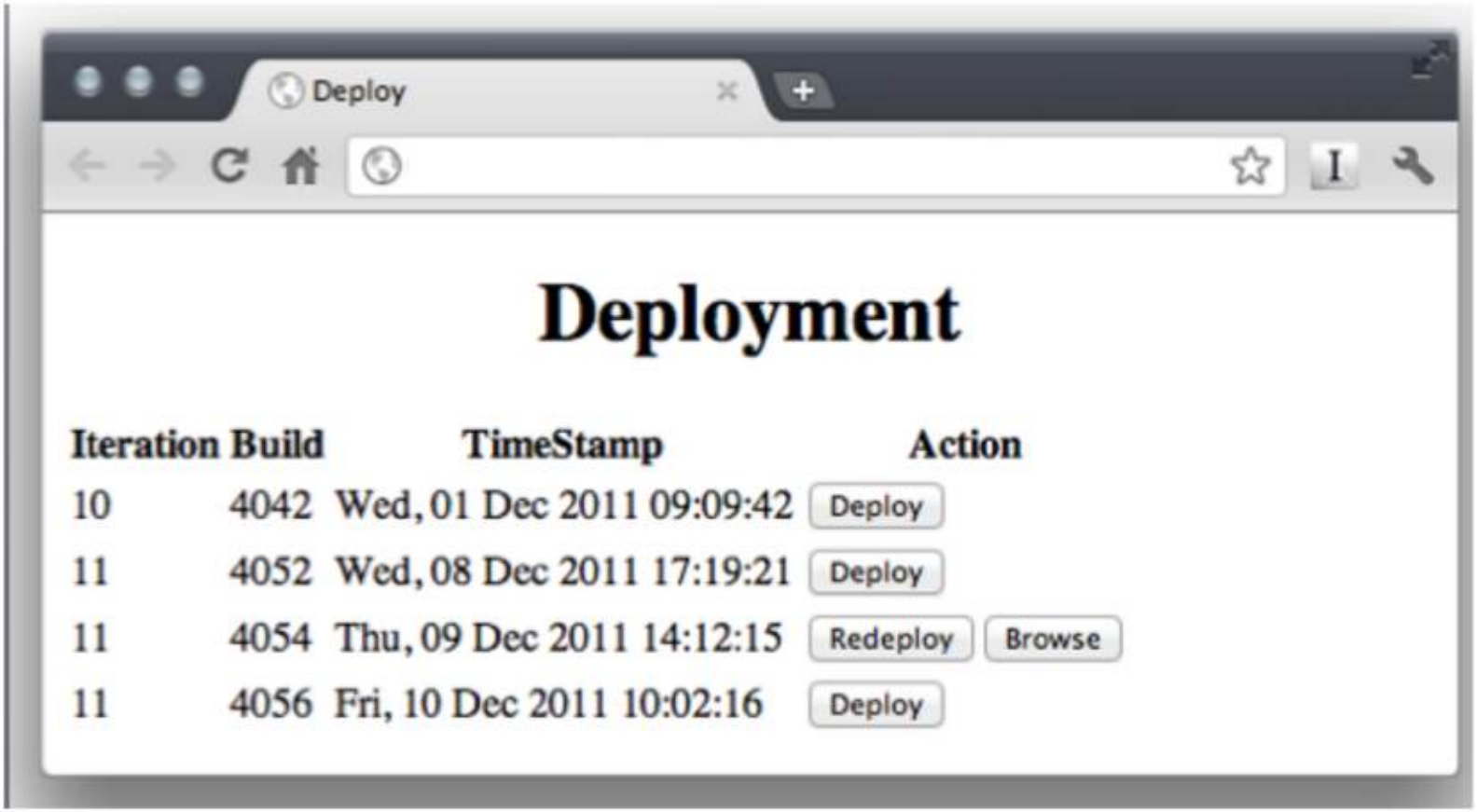
Thoughtworks Go

Etsy Deployinator

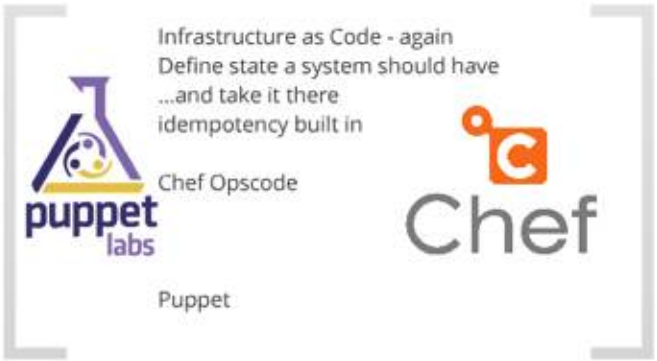
Dreadnot

coordination of stages
i.e. skripts, Ruby commands etc

Need automated Installers



Script based solutions



+

- Portable
- Collections of predefined recipes
- Configurations can be easily added

-

- Recipes usually need to be customized

Infrastructure as Code - again

Define state a system should have

...and take it there

idempotency built in

Chef Opscode

Puppet

+

- Portable
- Collections of predefined recipes
- Configurations can be easily added

-

- Recipes usually need to be customized

Chef

- Chef cooks VMs for you
- Ruby based DSL
- i.e. flexible and powerful

Chef server

- One server to configure and manage a set of clients
- Clients run chef-client
- Minimal setup on client needed
- Ideal for larger setups

Opscode hosted Chef

- Like Chef server
- + specific tools for Amazon IaaS

Chef solo

- Command line tool
- Needs all config available
- Focus for the rest of the presentation

- One server to configure and manage a set of clients
- Clients run chef-client
- Minimal setup on client needed
- Ideal for larger setups

Chef server

- One server to configure and manage a set of clients
- Clients run chef-client
- Minimal setup on client needed
- Ideal for larger setups

Opscode hosted Chef

- Like Chef server
- + specific tools for Amazon IaaS

Chef solo

- Command line tool
- Needs all config available
- Focus for the rest of the presentation

- Like Chef server
- + specific tools for Amazon IaaS

Chef server

- One server to configure and manage a set of clients
- Clients run chef-client
- Minimal setup on client needed
- Ideal for larger setups

Opscode hosted Chef

- Like Chef server
- + specific tools for Amazon IaaS

Chef solo

- Command line tool
- Needs all config available
- Focus for the rest of the presentation

- Command line tool
- Needs all config available
- Focus for the rest of the presentation

Ruby DSL

- Fundamental configuration object
- i.e files, Linux packages etc
- Resources are made to comply with a policy

[illegible]

Resource

- Directories, files, templates, remote files
- Packages, services, users, groups
- Scripts, commands, ruby code blocks
- Subversion and git code repositories
- Application deployment, HTTP requests
- Network interfaces, filesystem mounts

- Can create your own resources and providers

- Package "tomcat" should be installed
- User "adesso" should exist

Policy

Provider

- Execute idempotent actions through providers
- Determine current state of resources
- Can change state it to comply to a policy

- Easily create a VM or set of VMs
- ...using a basic VM image
- Virtualization with VirtualBox

Ruby DSL

- Fundamental configuration object
- i.e files, Linux packages etc
- Resources are made to comply with a policy

```
template "/etc/tomcat6/server.xml" do
  source "server.xml.erb" Policy
  owner "root"                - Configuration should match template
  group "root"                - Owner / group / mode should be set
  mode "0644"                 - Changes -> Tomcat restarted
  notifies :restart, resources(:service => "tomcat")
end
```

Resource

- Directories, files, templates, remote files
- Packages, services, users, groups
- Scripts, commands, ruby code blocks
- Subversion and git code repositories
- Application deployment, HTTP requests
- Network interfaces, filesystem mounts

- Can create your own resources and providers

- Fundamental configuration object
- i.e files, Linux packages etc
- Resources are made to comply with a policy

- Package "tomcat" should be installed
- User "adesso" should exist

Policy

- Package "tomcat" should be installed
- User "adesso" should exist

- Directories, files, templates, remote files
- Packages, services, users, groups
- Scripts, commands, ruby code blocks
- Subversion and git code repositories
- Application deployment, HTTP requests
- Network interfaces, filesystem mounts

- Can create your own resources and providers

Provider

- Execute idempotent actions through providers
- Determine current state of resources
- Can change state it to comply to a policy

- Execute idempotent actions through providers
- Determine current state of resources
- Can change state it to comply to a policy

- Directories, files, templates, remote files
- Packages, services, users, groups
- Scripts, commands, ruby code blocks
- Subversion and git code repositories
- Application deployment, HTTP requests
- Network interfaces, filesystem mounts

- Can create your own resources and providers

Ruby DSL

```
template "/etc/tomcat6/server.xml" do
  source "server.xml.erb"
  owner "root"
  group "root"
  mode "0644"
  notifies :restart, resources(:service => "tomcat")
end
```

Policy

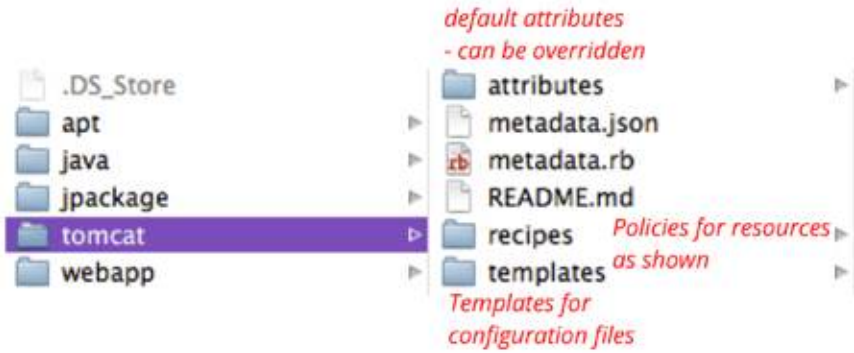
- *Configuration should match template*
- *Owner / group / mode should be set*
- *Changes -> Tomcat restarted*

Recipes

Recipes contain a combination of resources

For Tomcat

- Install Tomcat packages
- Make sure Java is installed
- Define the service
- Create configuration from templates



Cookbooks organize recipes

- http://community.opscode.com/cookbooks
- https://github.com/opscode/cookbook

(used here)

- https://github.com/37signals/37s_cookbooks
- https://github.com/engineyard/ey-cloud-recipes

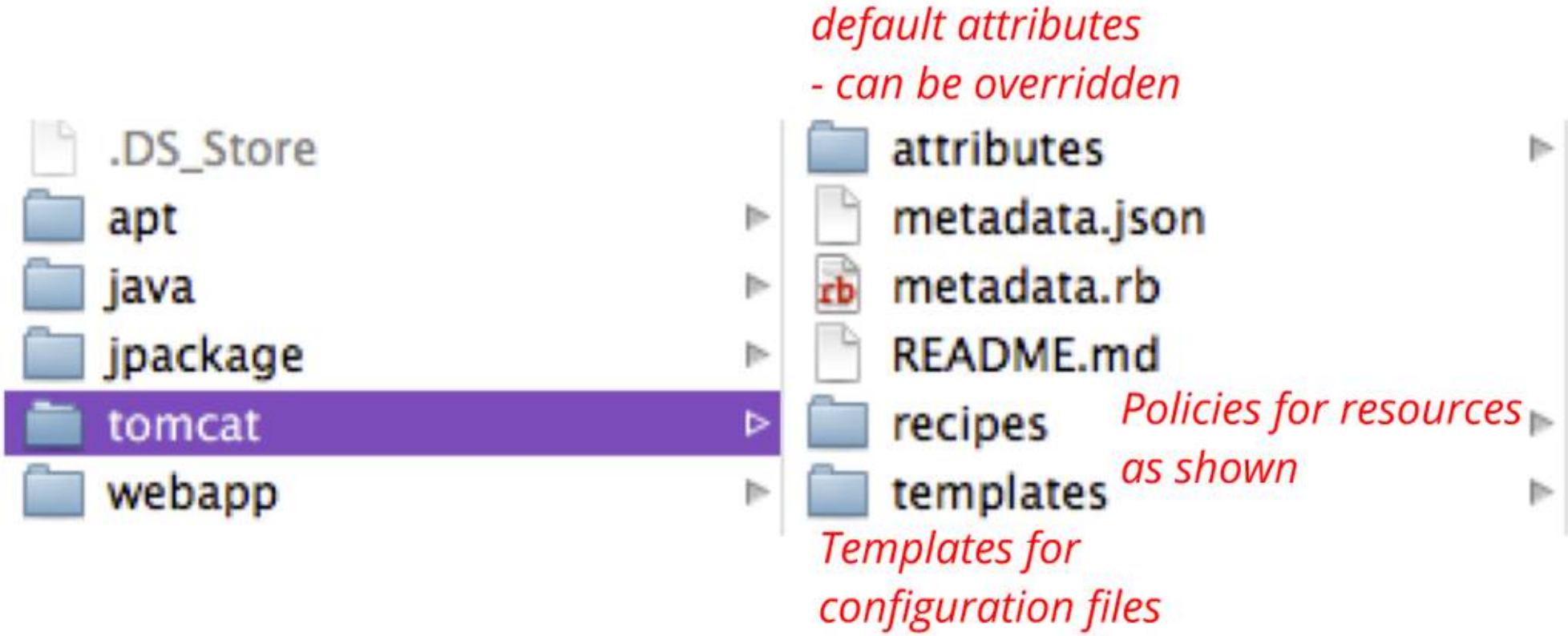
Recipes contain a combination of resources

For Tomcat

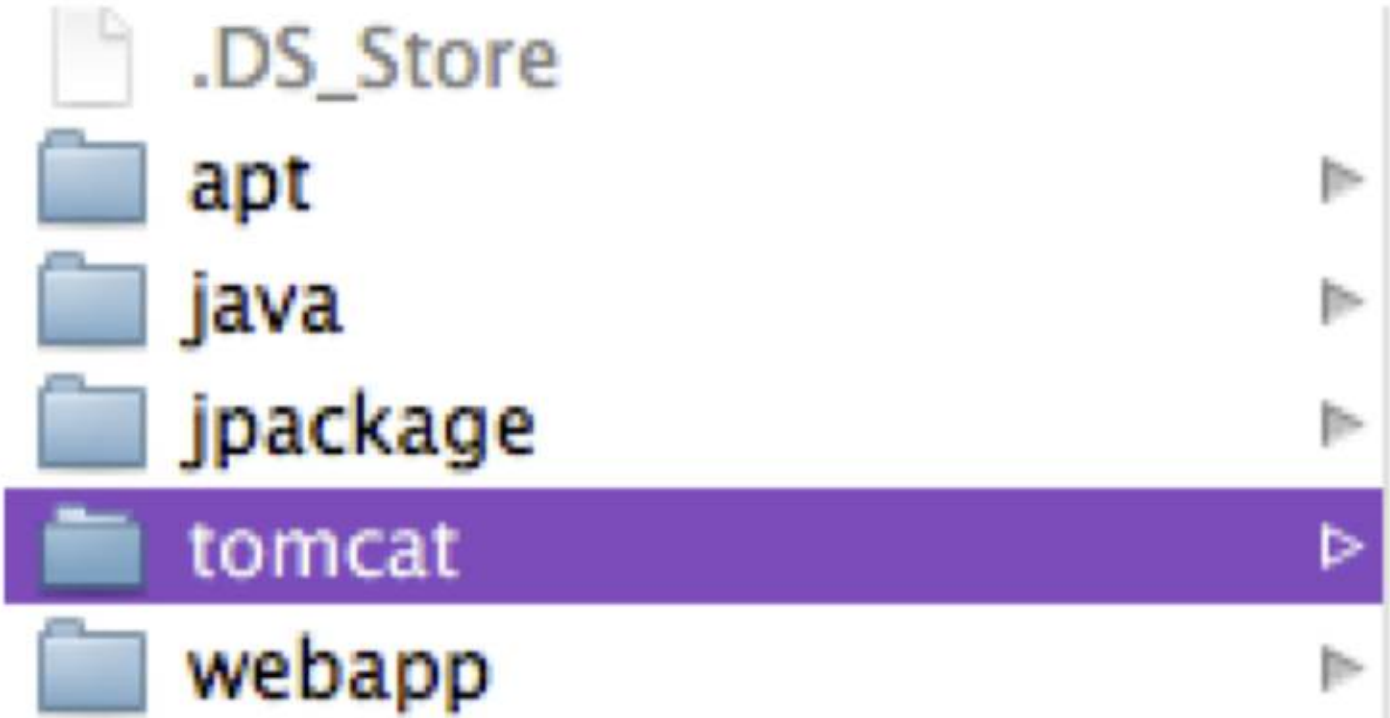
- Install Tomcat packages
- Make sure Java is installed
- Define the service
- Create configuration from templates

Cook

- h



ooks organize recipes
://community.opscode.com/cookbooks
s://github.com/opscode/cookbook
ere)
s://github.com/37signals/37s_cookbooks



Templ
configu

Cookbooks organize recipes

- http://community.opscode.com/cookbooks
- https://github.com/opscode/cookbook

(used here)

- https://github.com/37signals/37s_cookbooks

y

http://bit.ly/opscode-opentraining

providers
- Determine current state of resources
- Can change state it to comply to a policy

Vagrant

https://github.com/ewolff/JavaChefVagrantEC2

veewee

- Build your own basic VM for Vagrant
- Includes basic Chef installation
- ssh accounts etc
- Several popular Operating Systems
- Even Windows

https://wiki.jenkins-ci.org/display/JENKINS/Vagrant+Plugin

- Easily create a VM or set of VMs
- ...using a basic VM image
- Virtualization with VirtualBox
- Provision with e.g. Chef
- Vagrant ssh into the VM

```
Vagrant::Config.run do |config|
  config.vm.box = "lucid32"              basic VM image
  config.vm.box_url="http://files.vagrantup.com/lucid32.box"

  config.vm.forward_port 8080, 18080     port forwarding VM
  config.vm.forward_port 8081, 18081     to localhost

  config.vm.provision :chef_solo do |chef|
    chef.cookbooks_path = ["cookbooks"]
    chef.add_recipe("apt")
    chef.add_recipe("tomcat")
    chef.add_recipe("webapp")
    chef.json.merge!({
      :tomcat => {
        :port => 8081
      },        Chef recipes and coniguration
      :webapp => {
        :webapp => "demo.war"
      }
    })
  end

end
```

- Easily create a VM or set of VMs
- ...using a basic VM image
- Virtualization with VirtualBox
- Provision with e.g. Chef
- Vagrant ssh into the VM

```
Vagrant::Config.run do |config|
  config.vm.box = "lucid32"
  config.vm.box_url="http://files.vagrantup.com/lucid32.box"

  config.vm.forward_port 8080, 18080
  config.vm.forward_port 8081, 18081

  config.vm.provision :chef_solo do |chef|
    chef.cookbooks_path = ["cookbooks"]
    chef.add_recipe("apt")
    chef.add_recipe("tomcat")
    chef.add_recipe("webapp")
    chef.json.merge!({
      :tomcat => {
        :port => 8081
      },
      :webapp => {
        :webapp => "demo.war"
      }
    })
  end

end
```

basic VM image

port forwarding VM to localhost

Chef recipes and coniguration

with e.g. Chef
n into the VM

```
Vagrant::Config.run do |config|
  config.vm.box = "lucid32"
  config.vm.box_url="http://files.vagrantup.com/lucid32.box"

  config.vm.forward_port 8080, 18080
  config.vm.forward_port 8081, 18081

  config.vm.provision :chef_solo do |chef|
    chef.cookbooks_path = ["cookbooks"]
    chef.add_recipe("ant")
```

basic VM image

port forwarding VM to localhost

```
Vagrant::Config.run do |config|
  config.vm.box = "lucid32"
  config.vm.box_url="http://files.vagrantup.com/lucid32.box"

  config.vm.forward_port 8080, 18080
  config.vm.forward_port 8081, 18081

  config.vm.provision :chef_solo do |chef|
    chef.cookbooks_path = ["cookbooks"]
    chef.add_recipe("apt")
    chef.add_recipe("tomcat")
    chef.add_recipe("webapp")
    chef.json.merge!({
```

basic VM image

port forwarding VM to localhost

```
config.vm.provision :chef_solo do |chef|
  chef.cookbooks_path = ["cookbooks"]
  chef.add_recipe("apt")
  chef.add_recipe("tomcat")
  chef.add_recipe("webapp")
  chef.json.merge!({
    :tomcat => {
      :port => 8081
    },
    :webapp => {
      :webapp => "demo.war"
    }
  })
end
```

Chef recipes and coniguration

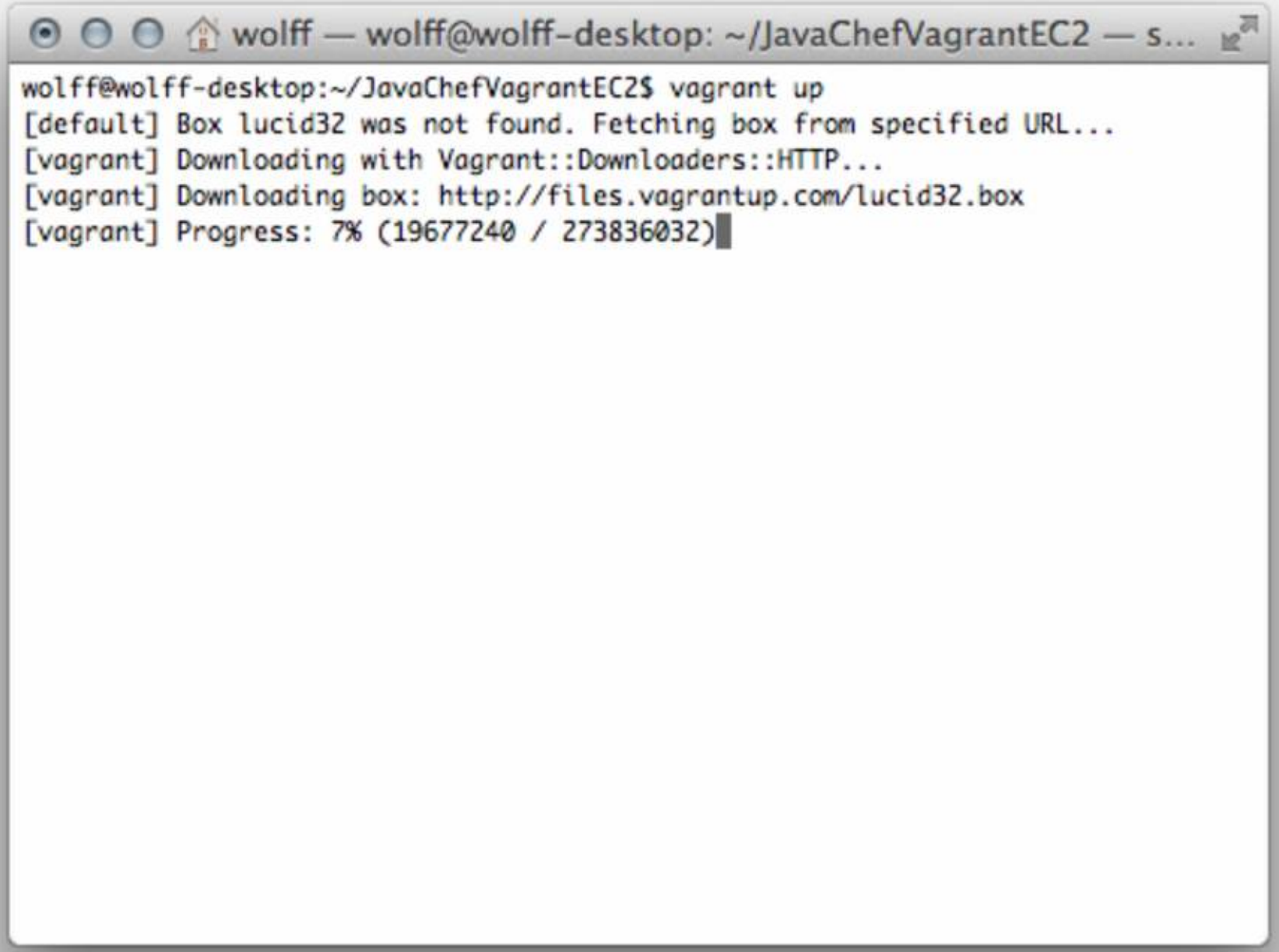
agrant

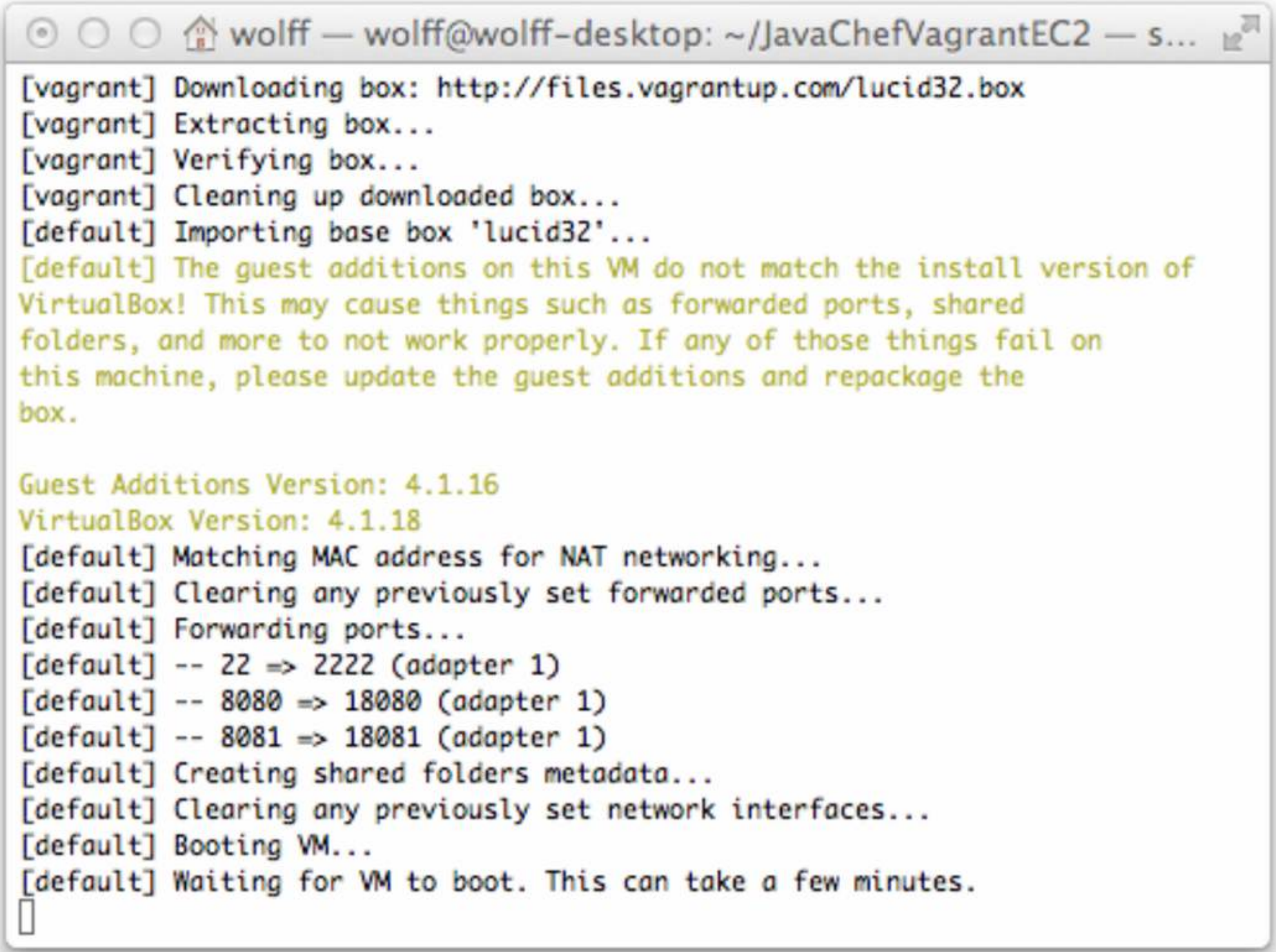
https://github.com/ewolff/JavaChefVagrantEC2

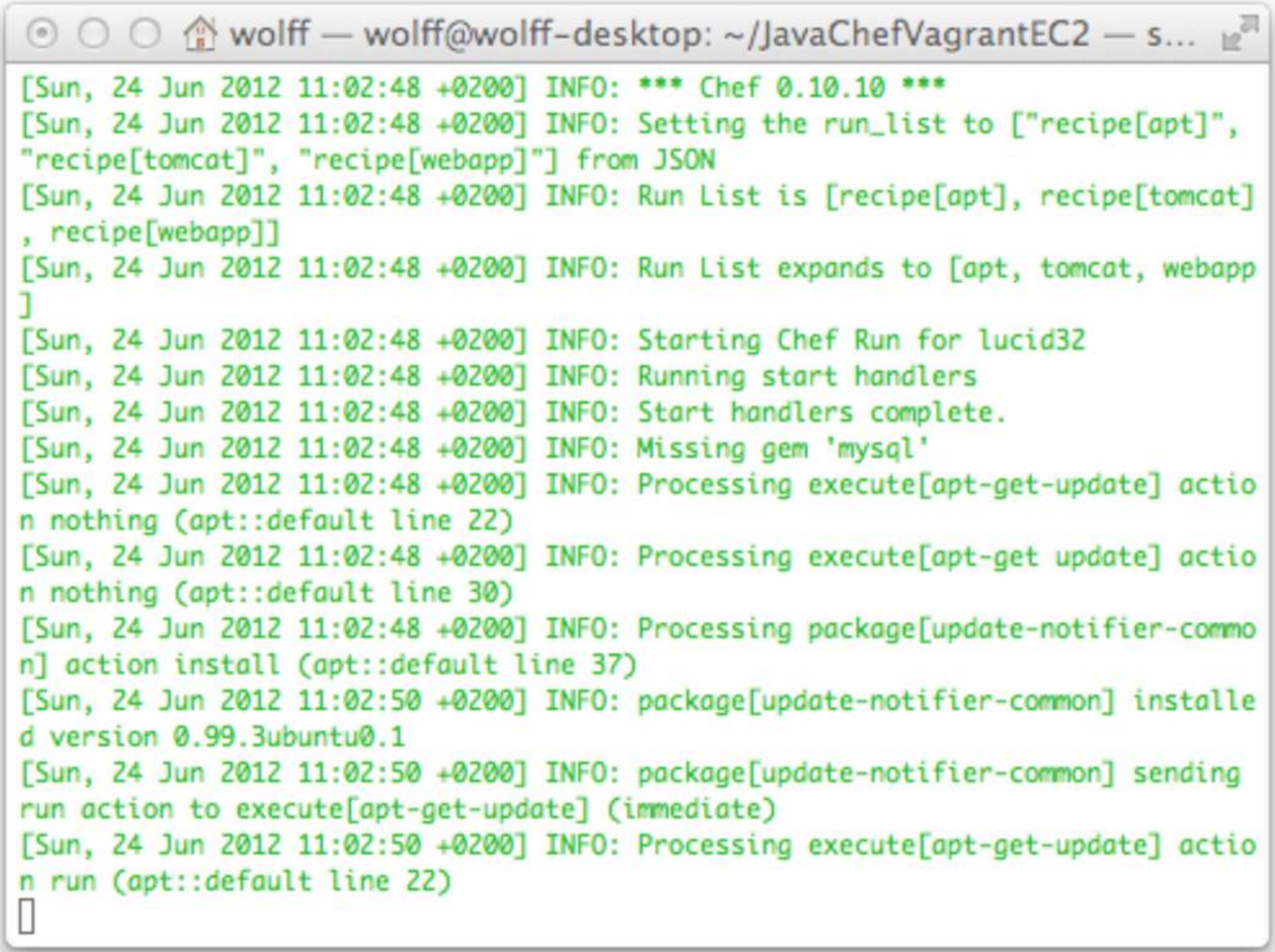
ewee

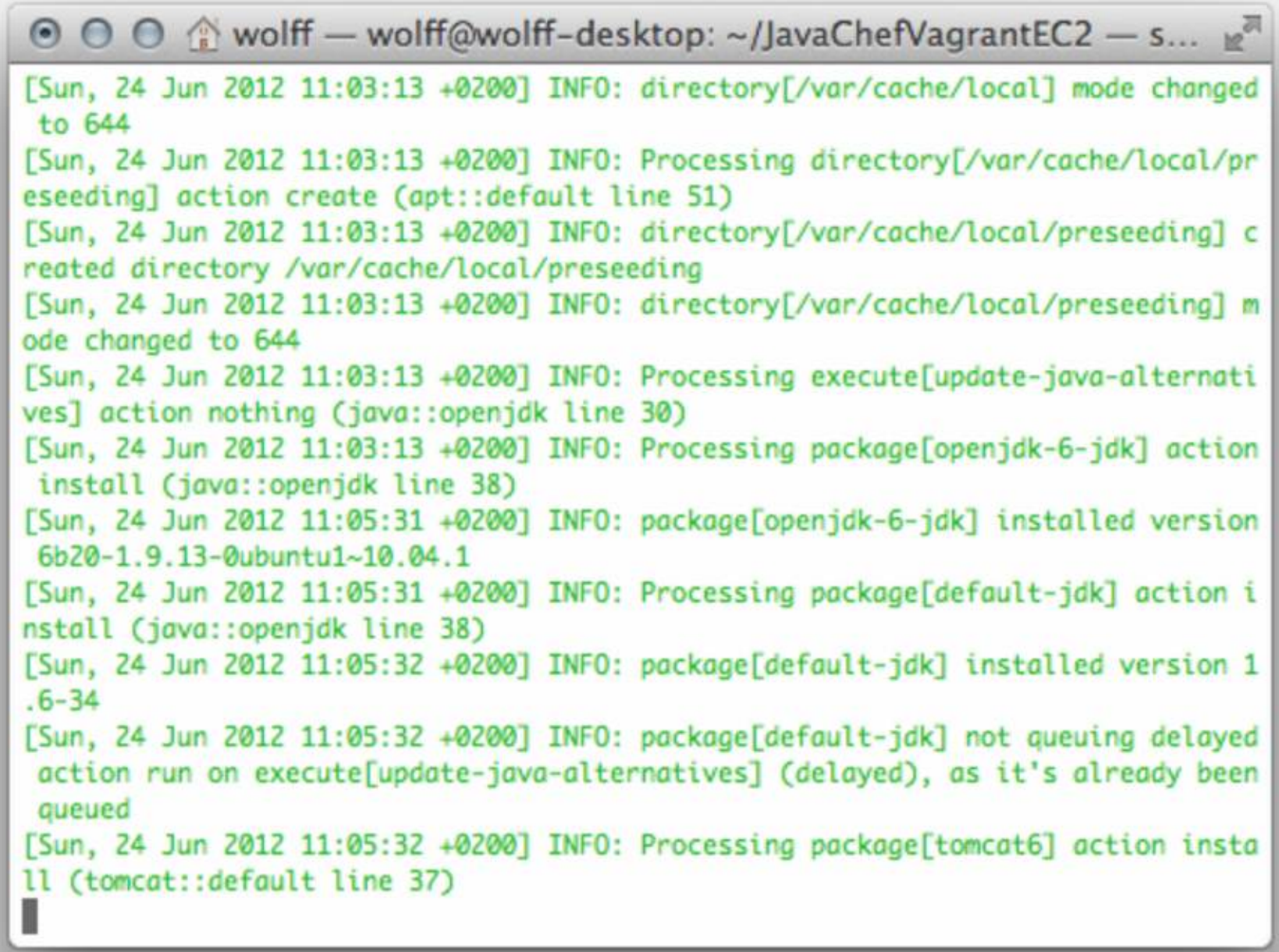
- Build your own basic VM for Vagrant
- Includes basic Chef installation
- ssh accounts etc
- Several popular Operating Systems
- Even Windows

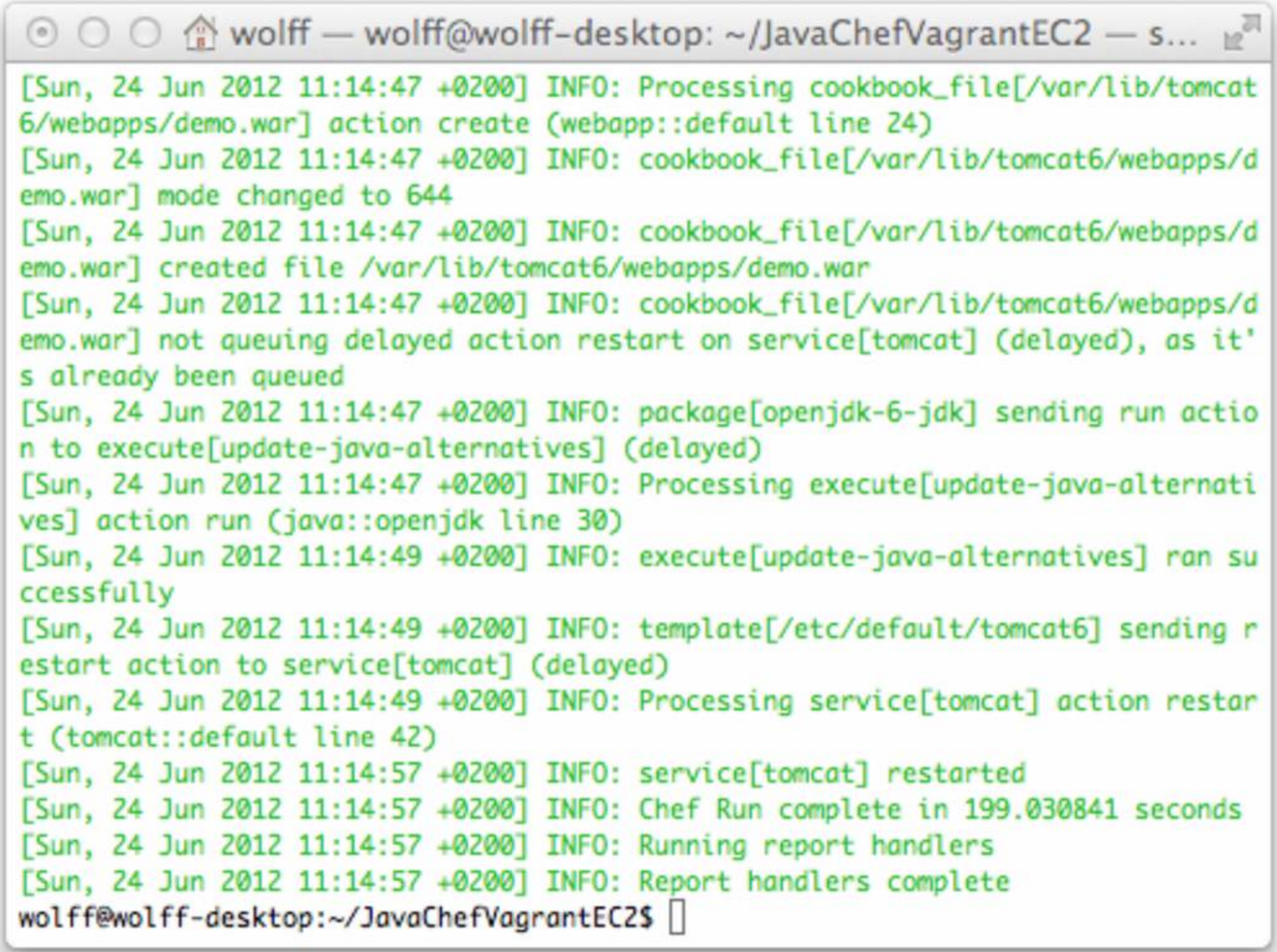
g/display/JENKINS/Vagrant+Plugin

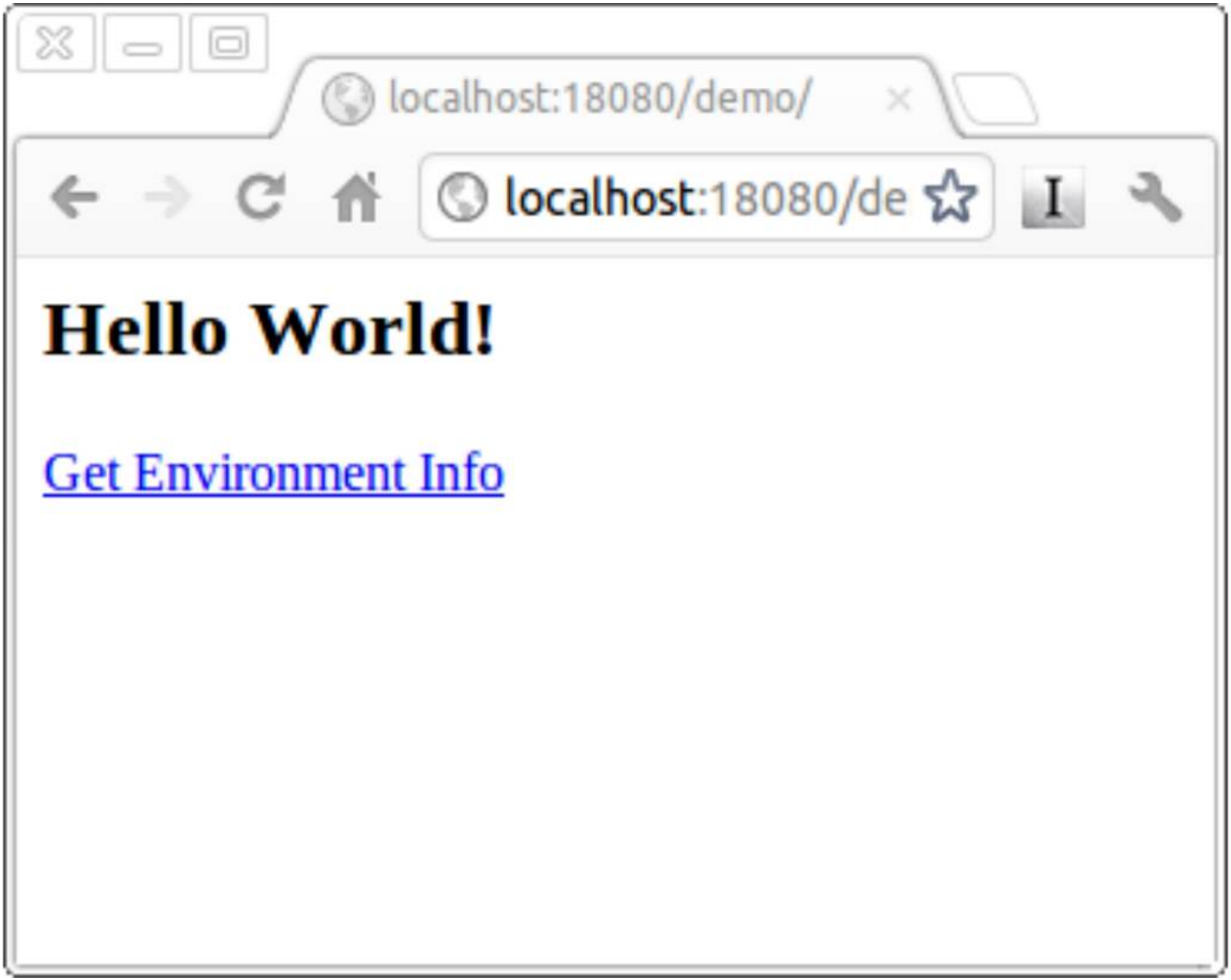

wolff — wolff@wolff-desktop: ~/JavaChefVagrantEC2 — s...
wolff@wolff-desktop:~/JavaChefVagrantEC2$ vagrant up
[default] Box lucid32 was not found. Fetching box from specified URL...
[vagrant] Downloading with Vagrant::Downloaders::HTTP...
[vagrant] Downloading box: http://files.vagrantup.com/lucid32.box
[vagrant] Progress: 7% (19677240 / 273836032)

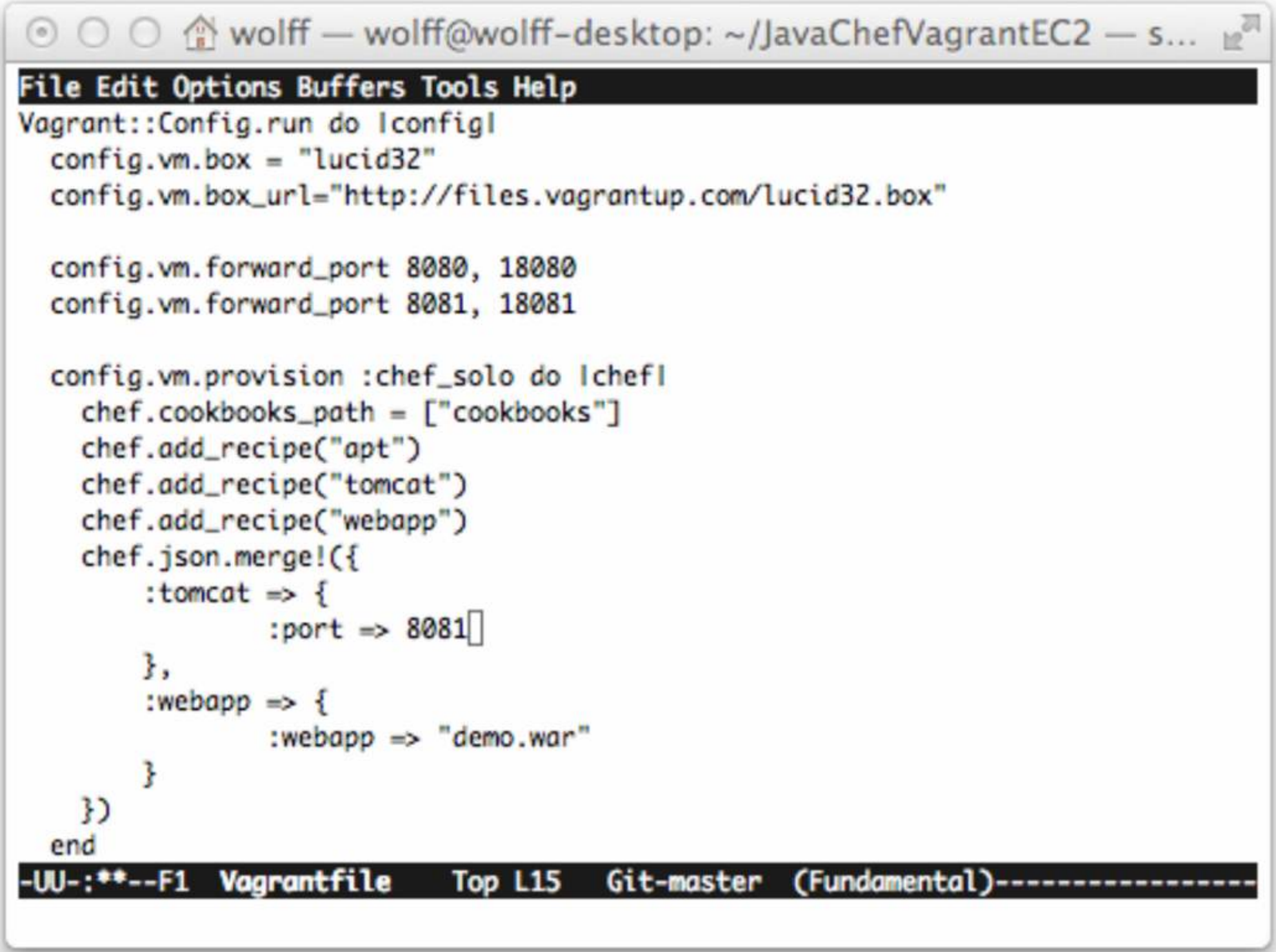

wolff — wolff@wolff-desktop: ~/JavaChefVagrantEC2 — s...
[vagrant] Downloading box: http://files.vagrantup.com/lucid32.box
[vagrant] Extracting box...
[vagrant] Verifying box...
[vagrant] Cleaning up downloaded box...
[default] Importing base box 'lucid32'...
[default] The guest additions on this VM do not match the install version of VirtualBox! This may cause things such as forwarded ports, shared folders, and more to not work properly. If any of those things fail on this machine, please update the guest additions and repackage the box.
Guest Additions Version: 4.1.16
VirtualBox Version: 4.1.18
[default] Matching MAC address for NAT networking...
[default] Clearing any previously set forwarded ports...
[default] Forwarding ports...
[default] -- 22 => 2222 (adapter 1)
[default] -- 8080 => 18080 (adapter 1)
[default] -- 8081 => 18081 (adapter 1)
[default] Creating shared folders metadata...
[default] Clearing any previously set network interfaces...
[default] Booting VM...
[default] Waiting for VM to boot. This can take a few minutes.

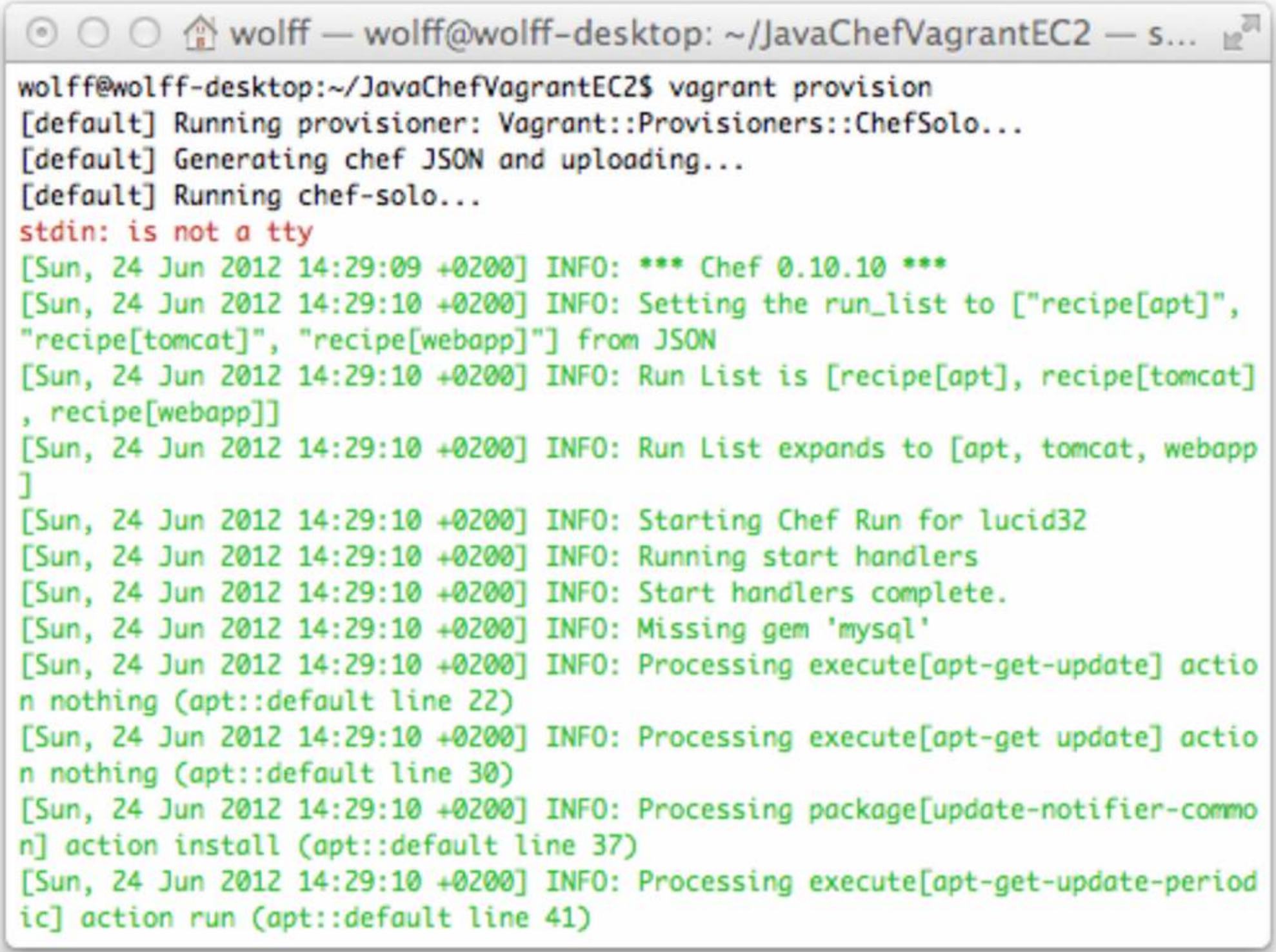

wolff — wolff@wolff-desktop: ~/JavaChefVagrantEC2 — s...
[Sun, 24 Jun 2012 11:02:48 +0200] INFO: *** Chef 0.10.10 ***
[Sun, 24 Jun 2012 11:02:48 +0200] INFO: Setting the run_list to ["recipe[apt]", "recipe[tomcat]", "recipe[webapp]"] from JSON
[Sun, 24 Jun 2012 11:02:48 +0200] INFO: Run List is [recipe[apt], recipe[tomcat], recipe[webapp]]
[Sun, 24 Jun 2012 11:02:48 +0200] INFO: Run List expands to [apt, tomcat, webapp]
[Sun, 24 Jun 2012 11:02:48 +0200] INFO: Starting Chef Run for lucid32
[Sun, 24 Jun 2012 11:02:48 +0200] INFO: Running start handlers
[Sun, 24 Jun 2012 11:02:48 +0200] INFO: Start handlers complete.
[Sun, 24 Jun 2012 11:02:48 +0200] INFO: Missing gem 'mysql'
[Sun, 24 Jun 2012 11:02:48 +0200] INFO: Processing execute[apt-get-update] action nothing (apt::default line 22)
[Sun, 24 Jun 2012 11:02:48 +0200] INFO: Processing execute[apt-get update] action nothing (apt::default line 30)
[Sun, 24 Jun 2012 11:02:48 +0200] INFO: Processing package[update-notifier-common] action install (apt::default line 37)
[Sun, 24 Jun 2012 11:02:50 +0200] INFO: package[update-notifier-common] installed version 0.99.3ubuntu0.1
[Sun, 24 Jun 2012 11:02:50 +0200] INFO: package[update-notifier-common] sending run action to execute[apt-get-update] (immediate)
[Sun, 24 Jun 2012 11:02:50 +0200] INFO: Processing execute[apt-get-update] action run (apt::default line 22)

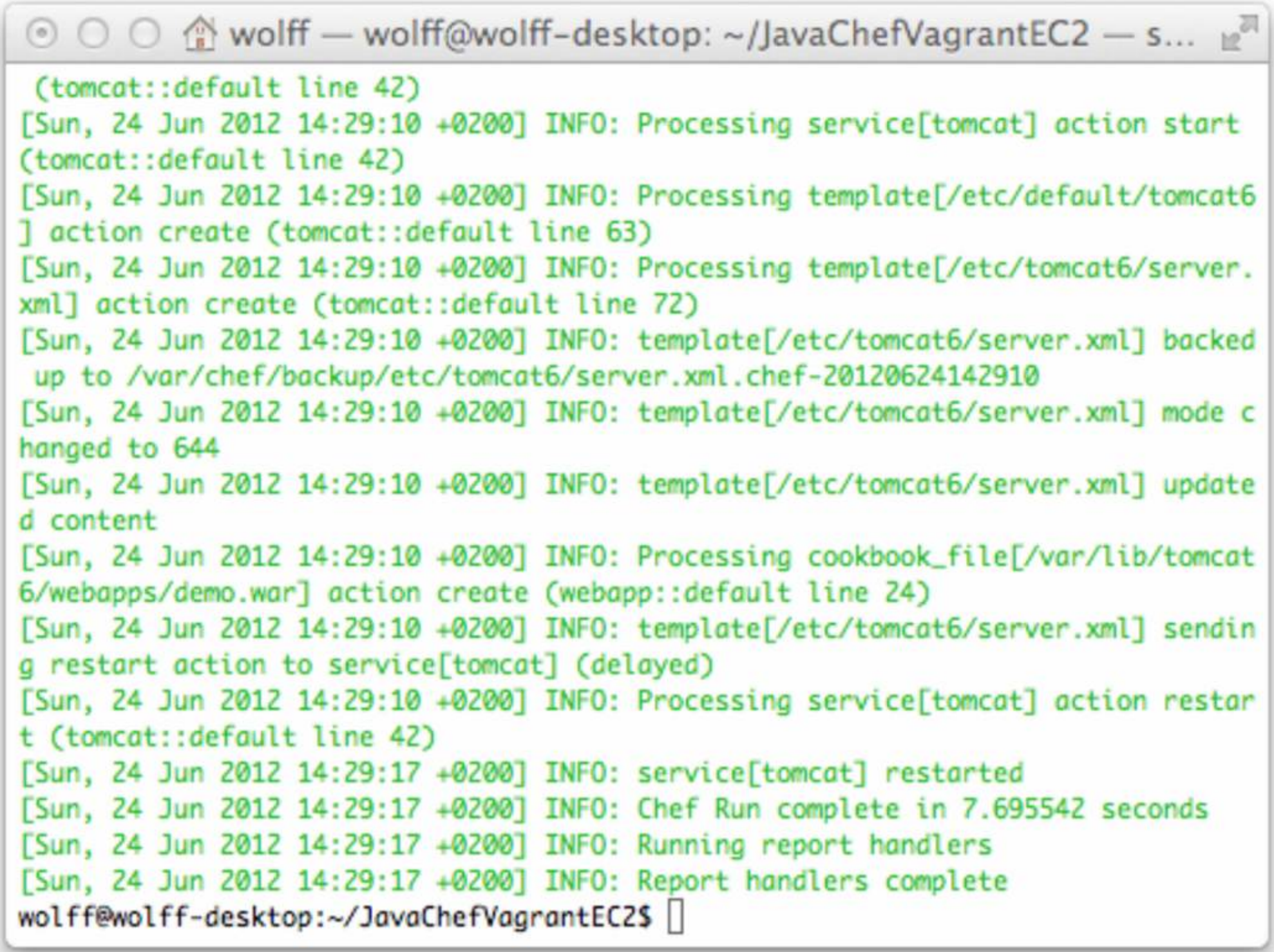

wolff — wolff@wolff-desktop: ~/JavaChefVagrantEC2 — s...
[Sun, 24 Jun 2012 11:03:13 +0200] INFO: directory[/var/cache/local] mode changed to 644
[Sun, 24 Jun 2012 11:03:13 +0200] INFO: Processing directory[/var/cache/local/preseeding] action create (apt::default line 51)
[Sun, 24 Jun 2012 11:03:13 +0200] INFO: directory[/var/cache/local/preseeding] created directory /var/cache/local/preseeding
[Sun, 24 Jun 2012 11:03:13 +0200] INFO: directory[/var/cache/local/preseeding] mode changed to 644
[Sun, 24 Jun 2012 11:03:13 +0200] INFO: Processing execute[update-java-alternatives] action nothing (java::openjdk line 30)
[Sun, 24 Jun 2012 11:03:13 +0200] INFO: Processing package[openjdk-6-jdk] action install (java::openjdk line 38)
[Sun, 24 Jun 2012 11:05:31 +0200] INFO: package[openjdk-6-jdk] installed version 6b20-1.9.13-0ubuntu1~10.04.1
[Sun, 24 Jun 2012 11:05:31 +0200] INFO: Processing package[default-jdk] action install (java::openjdk line 38)
[Sun, 24 Jun 2012 11:05:32 +0200] INFO: package[default-jdk] installed version 1.6-34
[Sun, 24 Jun 2012 11:05:32 +0200] INFO: package[default-jdk] not queuing delayed action run on execute[update-java-alternatives] (delayed), as it's already been queued
[Sun, 24 Jun 2012 11:05:32 +0200] INFO: Processing package[tomcat6] action install (tomcat::default line 37)

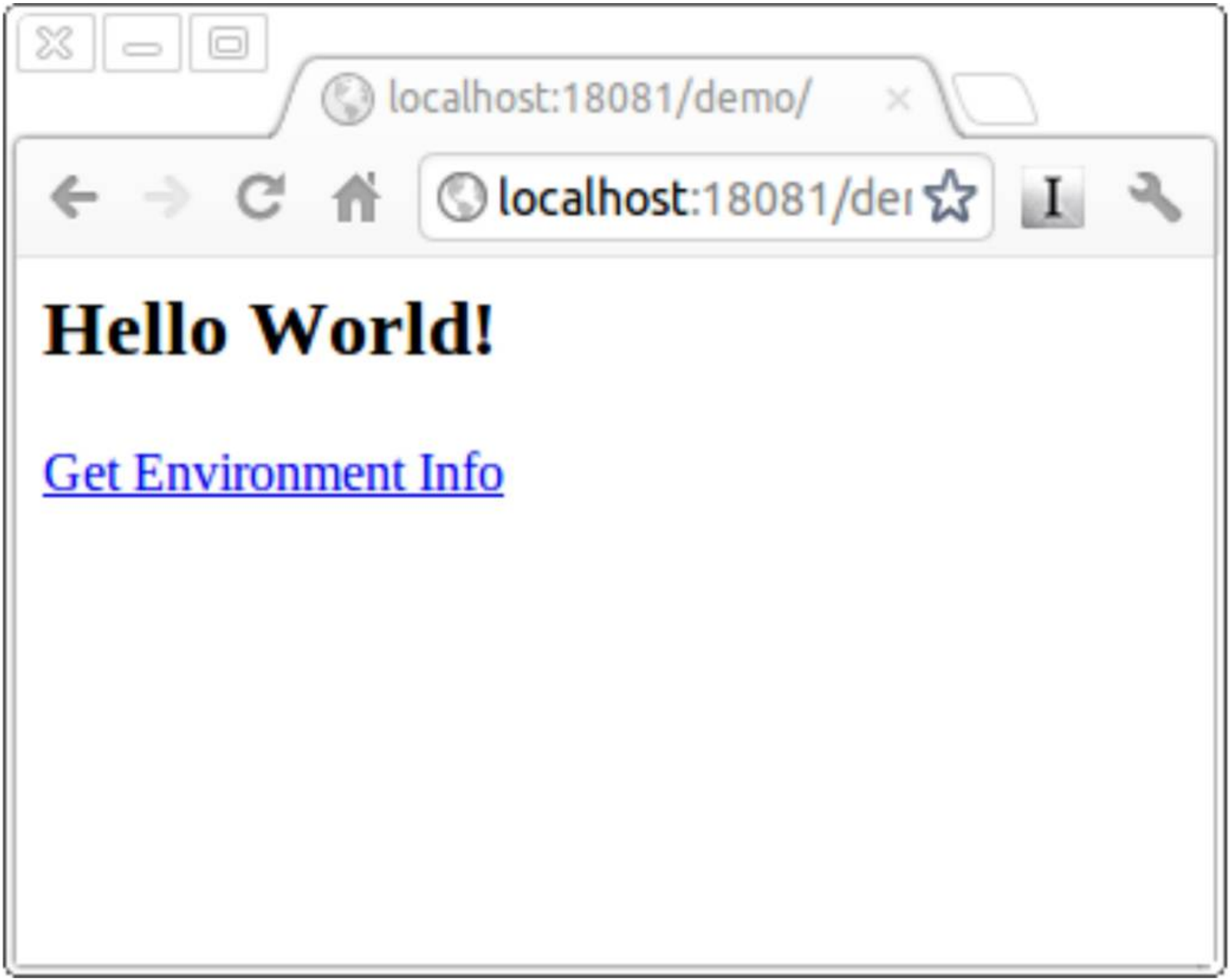

wolff — wolff@wolff-desktop: ~/JavaChefVagrantEC2 — s...
[Sun, 24 Jun 2012 11:14:47 +0200] INFO: Processing cookbook_file[/var/lib/tomcat6/webapps/demo.war] action create (webapp::default line 24)
[Sun, 24 Jun 2012 11:14:47 +0200] INFO: cookbook_file[/var/lib/tomcat6/webapps/demo.war] mode changed to 644
[Sun, 24 Jun 2012 11:14:47 +0200] INFO: cookbook_file[/var/lib/tomcat6/webapps/demo.war] created file /var/lib/tomcat6/webapps/demo.war
[Sun, 24 Jun 2012 11:14:47 +0200] INFO: cookbook_file[/var/lib/tomcat6/webapps/demo.war] not queuing delayed action restart on service[tomcat] (delayed), as it's already been queued
[Sun, 24 Jun 2012 11:14:47 +0200] INFO: package[openjdk-6-jdk] sending run action to execute[update-java-alternatives] (delayed)
[Sun, 24 Jun 2012 11:14:47 +0200] INFO: Processing execute[update-java-alternatives] action run (java::openjdk line 30)
[Sun, 24 Jun 2012 11:14:49 +0200] INFO: execute[update-java-alternatives] ran successfully
[Sun, 24 Jun 2012 11:14:49 +0200] INFO: template[/etc/default/tomcat6] sending restart action to service[tomcat] (delayed)
[Sun, 24 Jun 2012 11:14:49 +0200] INFO: Processing service[tomcat] action restart (tomcat::default line 42)
[Sun, 24 Jun 2012 11:14:57 +0200] INFO: service[tomcat] restarted
[Sun, 24 Jun 2012 11:14:57 +0200] INFO: Chef Run complete in 199.030841 seconds
[Sun, 24 Jun 2012 11:14:57 +0200] INFO: Running report handlers
[Sun, 24 Jun 2012 11:14:57 +0200] INFO: Report handlers complete
wolff@wolff-desktop:~/JavaChefVagrantEC2$


localhost:18080/demo/
localhost:18080/de
Hello World!
Get Environment Info


wolff — wolff@wolff-desktop: ~/JavaChefVagrantEC2 — s...
File Edit Options Buffers Tools Help
Vagrant::Config.run do |config|
config.vm.box = "lucid32"
config.vm.box_url="http://files.vagrantup.com/lucid32.box"
config.vm.forward_port 8080, 18080
config.vm.forward_port 8081, 18081
config.vm.provision :chef_solo do |chef|
chef.cookbooks_path = ["cookbooks"]
chef.add_recipe("apt")
chef.add_recipe("tomcat")
chef.add_recipe("webapp")
chef.json.merge!({
:tomcat => {
:port => 8081
},
:webapp => {
:webapp => "demo.war"
}
})
end
-UU-:**--F1 Vagrantfile Top L15 Git-master (Fundamental)


wolff — wolff@wolff-desktop: ~/JavaChefVagrantEC2 — s...
wolff@wolff-desktop:~/JavaChefVagrantEC2$ vagrant provision
[default] Running provisioner: Vagrant::Provisioners::ChefSolo...
[default] Generating chef JSON and uploading...
[default] Running chef-solo...
stdin: is not a tty
[Sun, 24 Jun 2012 14:29:09 +0200] INFO: *** Chef 0.10.10 ***
[Sun, 24 Jun 2012 14:29:10 +0200] INFO: Setting the run_list to ["recipe[apt]", "recipe[tomcat]", "recipe[webapp]"] from JSON
[Sun, 24 Jun 2012 14:29:10 +0200] INFO: Run List is [recipe[apt], recipe[tomcat], recipe[webapp]]
[Sun, 24 Jun 2012 14:29:10 +0200] INFO: Run List expands to [apt, tomcat, webapp]
[Sun, 24 Jun 2012 14:29:10 +0200] INFO: Starting Chef Run for lucid32
[Sun, 24 Jun 2012 14:29:10 +0200] INFO: Running start handlers
[Sun, 24 Jun 2012 14:29:10 +0200] INFO: Start handlers complete.
[Sun, 24 Jun 2012 14:29:10 +0200] INFO: Missing gem 'mysql'
[Sun, 24 Jun 2012 14:29:10 +0200] INFO: Processing execute[apt-get-update] action nothing (apt::default line 22)
[Sun, 24 Jun 2012 14:29:10 +0200] INFO: Processing execute[apt-get update] action nothing (apt::default line 30)
[Sun, 24 Jun 2012 14:29:10 +0200] INFO: Processing package[update-notifier-common] action install (apt::default line 37)
[Sun, 24 Jun 2012 14:29:10 +0200] INFO: Processing execute[apt-get-update-periodic] action run (apt::default line 41)


wolff — wolff@wolff-desktop: ~/JavaChefVagrantEC2 — s...
(tomcat::default line 42)
[Sun, 24 Jun 2012 14:29:10 +0200] INFO: Processing service[tomcat] action start (tomcat::default line 42)
[Sun, 24 Jun 2012 14:29:10 +0200] INFO: Processing template[/etc/default/tomcat6] action create (tomcat::default line 63)
[Sun, 24 Jun 2012 14:29:10 +0200] INFO: Processing template[/etc/tomcat6/server.xml] action create (tomcat::default line 72)
[Sun, 24 Jun 2012 14:29:10 +0200] INFO: template[/etc/tomcat6/server.xml] backed up to /var/chef/backup/etc/tomcat6/server.xml.chef-20120624142910
[Sun, 24 Jun 2012 14:29:10 +0200] INFO: template[/etc/tomcat6/server.xml] mode changed to 644
[Sun, 24 Jun 2012 14:29:10 +0200] INFO: template[/etc/tomcat6/server.xml] updated content
[Sun, 24 Jun 2012 14:29:10 +0200] INFO: Processing cookbook_file[/var/lib/tomcat6/webapps/demo.war] action create (webapp::default line 24)
[Sun, 24 Jun 2012 14:29:10 +0200] INFO: template[/etc/tomcat6/server.xml] sending restart action to service[tomcat] (delayed)
[Sun, 24 Jun 2012 14:29:10 +0200] INFO: Processing service[tomcat] action restart (tomcat::default line 42)
[Sun, 24 Jun 2012 14:29:17 +0200] INFO: service[tomcat] restarted
[Sun, 24 Jun 2012 14:29:17 +0200] INFO: Chef Run complete in 7.695542 seconds
[Sun, 24 Jun 2012 14:29:17 +0200] INFO: Running report handlers
[Sun, 24 Jun 2012 14:29:17 +0200] INFO: Report handlers complete
wolff@wolff-desktop:~/JavaChefVagrantEC2$


localhost:18081/demo/
localhost:18081/dei
Hello World!
Get Environment Info

veewee

- Build your own basic
- Includes basic Chef i
- ssh accounts etc

eewee

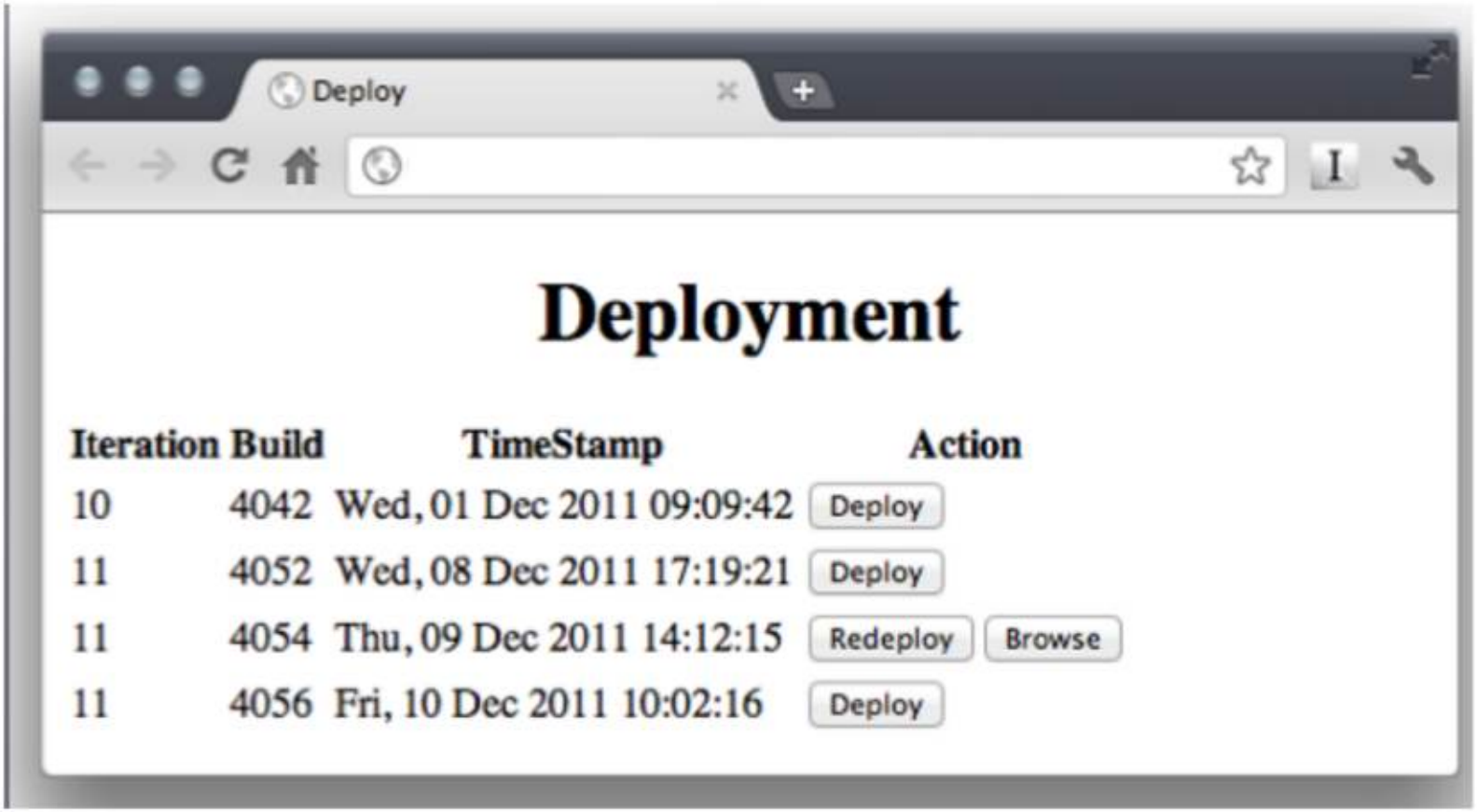
- Build your own basic VM for Vagrant
- Includes basic Chef installation
- ssh accounts etc
- Several popular Operating Systems
- Even Windows

-ci.org/display/JENKINS/Vagrant+Plugin

- Build your own basic VM for Vagrant
- Includes basic Chef installation
- ssh accounts etc
- Several popular Operating Systems
- Even Windows

https://wiki.jenkins-ci.org/display/JENKINS/Vagrant+Plugin

Need automated Installers



Numerous other options

- Package Managers & automated installers (apt-get, yum etc)
- Deploy to running server (only partial solution)
- Extensions to home grown solutions
- Much like initial Continuous Integration

options

- Package Managers & automated installers (apt-get, yum etc)
- Deploy to running server (only partial solution)
- Extensions to home grown solutions
- Much like initial Continuous Integration affords
- Quite a lot established solutions for software distribution

Need to deal with databases

- Schema migration
- Update data
- Managing test data
- NoSQL database are more flexible
- Tools
  - DB schema versiones
  - Scripts to update a version to another

- Schema migration
- Update data
- Managing test data
- NoSQL database are more flexible
- Tools
  - DB schema versiones
  - Scripts to update a version to another
  - ...or rollback

- simple and handy
- SQL

- Powerful and complex
- SQL abstraction

Principles

If it hurts - do it more often - and bring the pain forward!

Don't Use Manual Procedures

Current Problems

- Bringing Software into Production is hard
- Takes a lot of time
- Error prone

Cloud

- Infrastructure can be created by calling an API
- Easy to create new servers

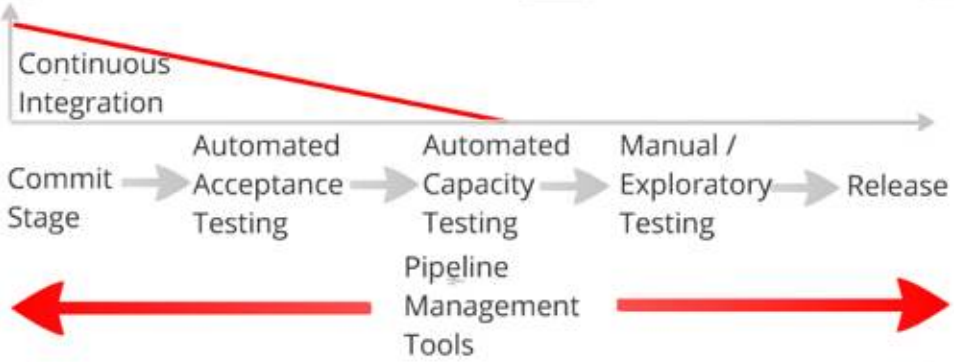
Infrastructure as Code

Continuous Delivery

Simplify bringing systems into production
Apply Continous Integration Principles

- Automate
- It hurts - do it more often - continuously
- Enabled by Cloud / Infrastructure as code

Pipeline - Builds are promoted



Need automated Installers

Script based solutions

Need to deal with databases

Numerous other options

Chef