Lightweight and reproducible environments with

Vagrant & Puppet

& Java

DEVOXX™
the java™ community conference

About me

- Hendrik Ebbers
- Lead of development at SIC GmbH in Dortmund, Germany
- Lead of JUG Dortmund

@hendrikEbbers

www.guigarage.com

hendrik.ebbers@web.de

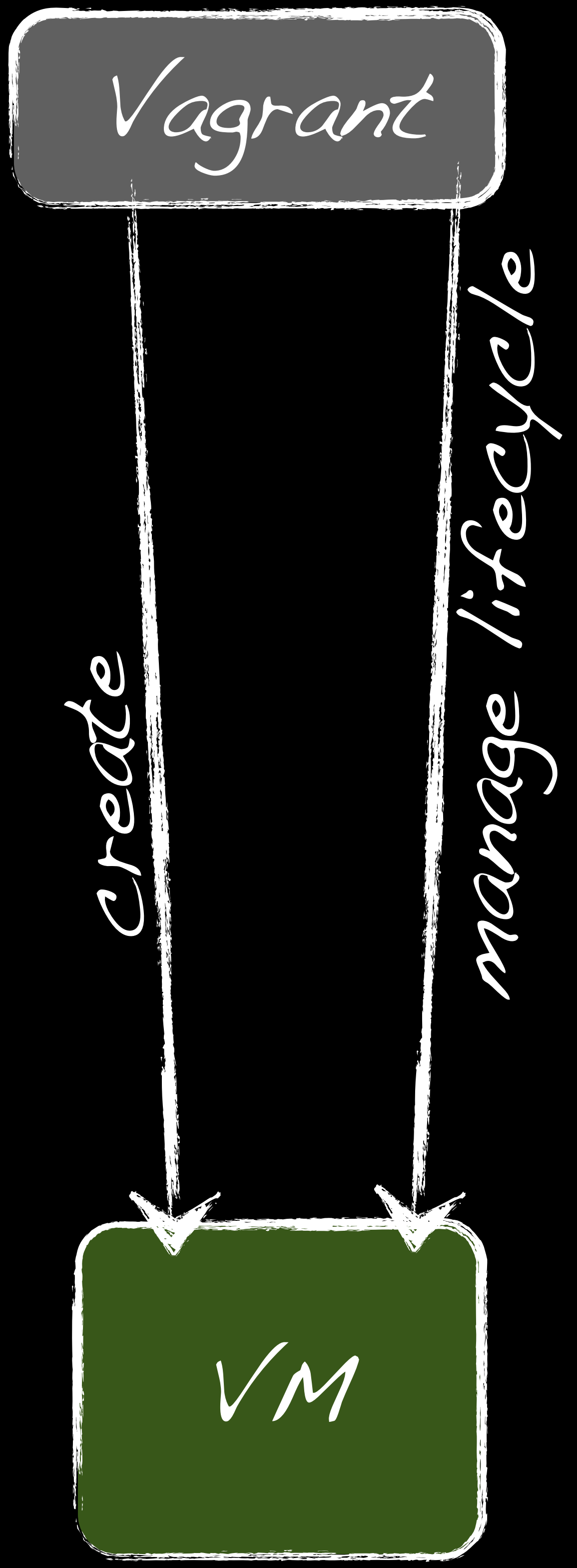
Content

- Vagrant
- Puppet
- Java Vagrant-Binding API

Vagrant

Vagrant

- configure virtual machines by script
- create new instances on the fly
- manage the VM lifecycle



Vagrant

add template VM to Vagrant

```
$ vagrant box add lucid32 http://
files.vagrantup.com/lucid32.box
```

```
$ vagrant init lucid32
```

creates VM configuration-script

```
$ vagrant up
```

start the virtual machine

Vagrant

- build on top of VirtualBox
- written in Ruby

access by shell & Ruby

Vagrant

Ubuntu Lucid
32- & 64-bit

- provides 2 template boxes by default

- simple config-files

it's just Ruby

```
Vagrant::Config.run do |config|
  config.vm.box = "lucid32"
end
```

see great Vagrant documentation

- easy ssh connection, shared folder, etc.

Puppet

Puppet

- configure your machines (nodes) by script
- install and configure software & services

Puppet

```
class apache {
  exec { 'apt-get update':
    command => '/usr/bin/apt-get update'
  }
  package { "apache2":
    ensure => present,
  }
  service { "apache2":
    ensure => running,
    require => Package["apache2"],
  }
}
include apache
```

Apache2 is installed
& started on node

Puppet

- package individual components in modules
- many online documentations & books out there

Vagrant
&
Puppet

Vagrant & Puppet

- define your VM with Vagrant & configure it with Puppet
- Puppet is pre-installed on Vagrant boxes

Vagrant & Puppet

Vagrantfile

```
Vagrant::Config.run do |config|
  config.vm.box = "lucid32"
  config.vm.provision :puppet do |puppet|
    puppet.manifests_path = "manifests"
    puppet.manifest_file = "my_manifest.pp"
  end
end
```

path to Puppet script

Vagrant-Binding

configure & manage VMs in Java

Vagrant-Binding

- Java Wrapper around Vagrant
- create & start VMs at runtime
- only VirtualBox is required

Vagrant-Binding

- Builder APIs
- JUnit support
- Puppet support

Builder API

```
VagrantVmConfig vmConfig = new VagrantVmConfigBuilder()
        .withLucid32Box()
        .withName("myLittleVm")
        .withHostOnlyIp("192.168.50.4")
        .build();

VagrantEnvironment environment = ...;

environment.up();

environment.getVm(0).createConnection().execute("touch /tmp1");

environment.destroy();
```

builder API for VM

also builder API available

manage VM lifecycle

ssh connection

JUnit support

```
@Test
public void testJdbc() {

   dbHandler = new MySql(ip, db, user, pwd);

   dbHandler.createMyTable();

   dbHandler.insertRow();

   assertEquals(1, dbHandler.getRowCount());

   dbHandler.close();
}
```

what if host not reachable?

what if table already exists?

parallel processes?

JUnit support

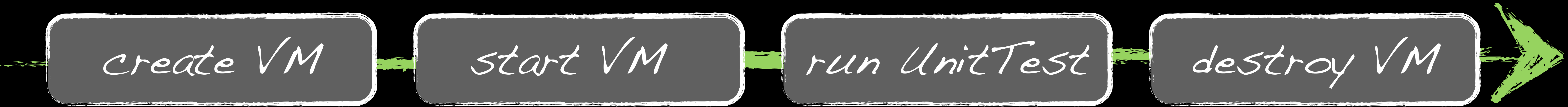
```
@Rule
public VagrantTestRule testRule =
       new VagrantTestRule(createConfig());


public static VagrantConfiguration createConfig() {
   //Configure VM with MySQL-Server & static ip
}
```

JUnit annotation

manage VM lifecycle

use builder API for VM specification



create VM

start VM

run UnitTest

destroy VM

Vagrant-Binding

fork me on github

https://github.com/guigarage/vagrant-binding

Thanks for watching

@hendrikEbbers

www.guigarage.com

hendrik.ebbers@web.de