

Infrastructure at your Service.

## Automated GI/RAC Staging



Invention & Infrastructure  
at your service.



# About me

**Daniel Westermann**

Consultant

Mobile +41 79 927 24 46

[daniel.westermann@dbi-services.com](mailto:daniel.westermann@dbi-services.com)

[www.dbi-services.com](http://www.dbi-services.com)



# Who we are dbi services

## Experts At Your Service

- > 40 specialists in IT infrastructure
- > Certified, experienced, passionate

## Based In Switzerland

- > 100% self-financed Swiss company
- > Over CHF 6 mio. turnover

## Leading In Infrastructure Services

- > More than 100 customers in CH, D, & F
- > 40 SLAs dbi FlexService contracted



# Our products

## Database Management Kit – for free

### Making IT easy for DBAs

- > Platform independent, standardized scripts for Unix/Linux & Windows – absolutely license free!
- > Numerous customers trust in DMK for a reliable & efficient database infrastructure
- > **Simple**, **safe**, and **speedy** way to carry out various DBA tasks:
  - > Setting up database environments
  - > Start/stop of database components
  - > Checking of relevant database settings
- > No time for maintenance & setup?  
Profit from **DMK Support!**

[www.dbi-services.com/DMK](http://www.dbi-services.com/DMK)



# Our products

## dbi InSite Workshops

### Expert insight from insiders

- > Your instructors are experienced and certified consultants
- > 2-4 day Workshop adjusted to your real needs
- > Organized on-site or externally
- > Few theory: practical exercises, live demos, and case studies

### Large portfolio

- > **Oracle Database:** NF 12c, DBA, B&R, Tuning, Data Guard, APEX
- > **MS SQLServer:** DBA, NF 2012, NF 2014, etc
- > **MS Sharepoint:** Administration
- > **Middleware:** WebLogic
- > **MySQL:** DBA
- > **Linux:** DBA

[www.dbi-services.com/InSite](http://www.dbi-services.com/InSite)



# Agenda

---

## 1. Last things first

- > System Overview
- > This is the result and this is what you'll get
- > Live Demo

## 2. Why?

## 3. Components

## 4. Putting everything together

- > On how the components interact
- > Command line examples

## 5. Core Messages

## 6. Q&A

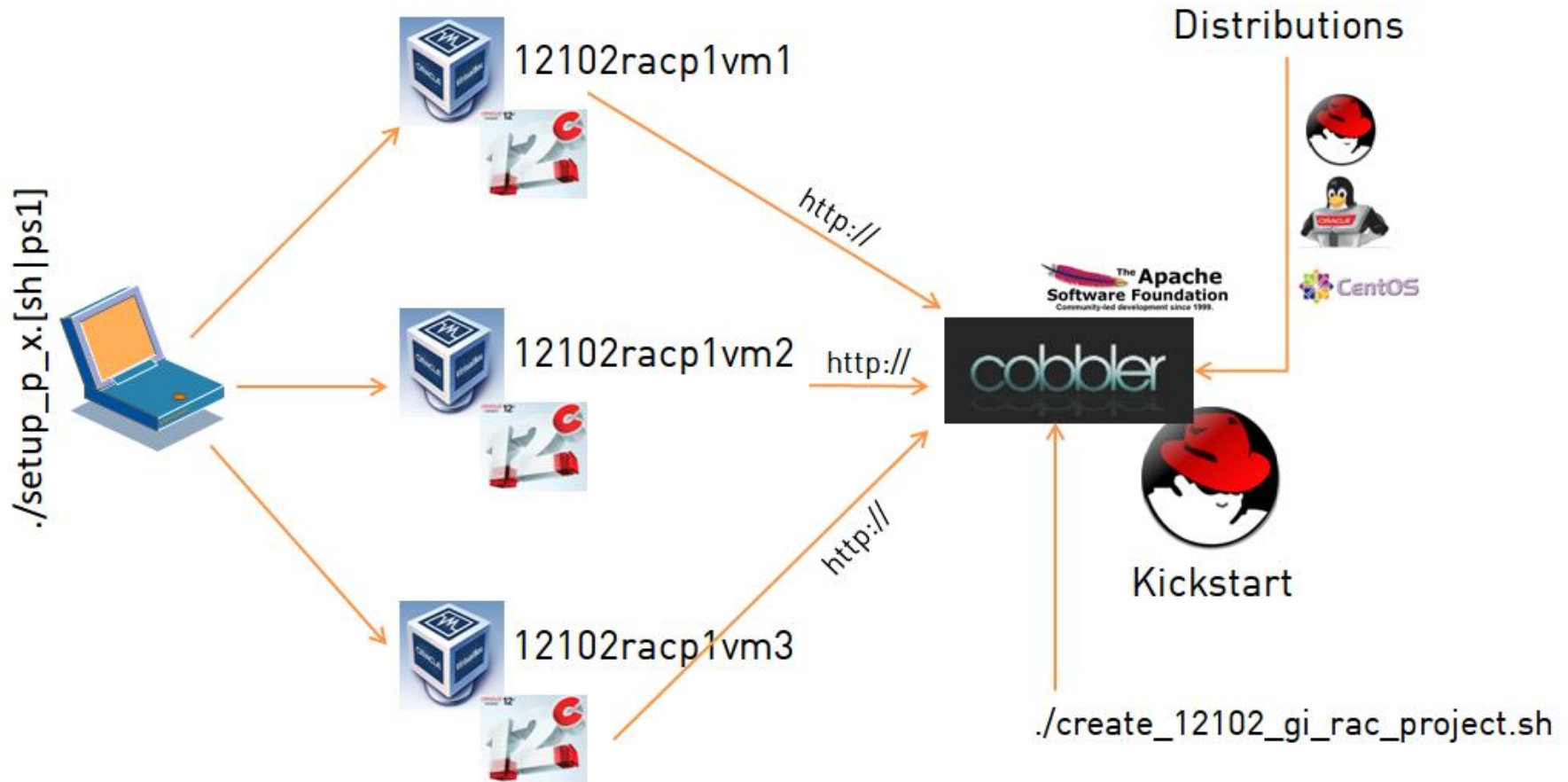
# Last things first

## You'll get this...



- > Last things first
- > Why
- > Components
- > Putting everything together
- > Core Messages
- > Q&A

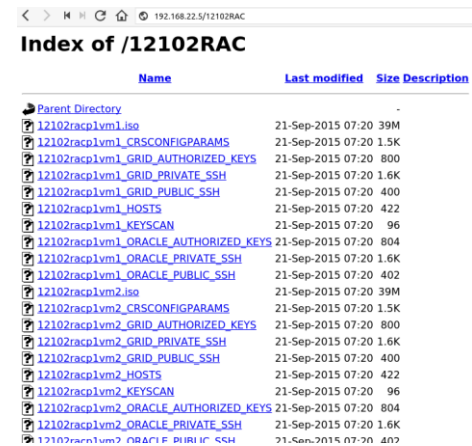
# System overview





# Last things first: This is the result

1. Generate the project
2. Everything is available over http
  - > Don't be confused by the various scripts and how they got there. I'll explain this in a few minutes
3. Download one of the setup\_\* scripts
4. Execute (and maybe adjust) the scripts
5. Start all VMs except for the last one



The screenshot shows a web browser window with the address bar displaying '192.168.22.5/12102RAC'. The page title is 'Index of /12102RAC'. Below the title is a table with columns for 'Name', 'Last modified', and 'Size Description'. The table lists various files and directories, including 'Parent Directory', '12102racp1vm1.iso', '12102racp1vm1\_CRSCONFIGPARAMS', '12102racp1vm1\_GRID\_AUTHORIZED\_KEYS', '12102racp1vm1\_GRID\_PRIVATE\_SSH', '12102racp1vm1\_GRID\_PUBLIC\_SSH', '12102racp1vm1\_HOSTS', '12102racp1vm1\_KEYSCAN', '12102racp1vm1\_ORACLE\_AUTHORIZED\_KEYS', '12102racp1vm1\_ORACLE\_PRIVATE\_SSH', '12102racp1vm1\_ORACLE\_PUBLIC\_SSH', '12102racp1vm2.iso', '12102racp1vm2\_CRSCONFIGPARAMS', '12102racp1vm2\_GRID\_AUTHORIZED\_KEYS', '12102racp1vm2\_GRID\_PRIVATE\_SSH', '12102racp1vm2\_GRID\_PUBLIC\_SSH', '12102racp1vm2\_HOSTS', '12102racp1vm2\_KEYSCAN', '12102racp1vm2\_ORACLE\_AUTHORIZED\_KEYS', '12102racp1vm2\_ORACLE\_PRIVATE\_SSH', and '12102racp1vm2\_ORACLE\_PUBLIC\_SSH'. The 'Last modified' column shows dates like '21-Sep-2015 07:20' and the 'Size Description' column shows sizes like '39M', '1.5K', '800', '1.6K', '400', '422', and '96'.

| Name                                 | Last modified     | Size Description |
|--------------------------------------|-------------------|------------------|
| Parent Directory                     | -                 | -                |
| 12102racp1vm1.iso                    | 21-Sep-2015 07:20 | 39M              |
| 12102racp1vm1_CRSCONFIGPARAMS        | 21-Sep-2015 07:20 | 1.5K             |
| 12102racp1vm1_GRID_AUTHORIZED_KEYS   | 21-Sep-2015 07:20 | 800              |
| 12102racp1vm1_GRID_PRIVATE_SSH       | 21-Sep-2015 07:20 | 1.6K             |
| 12102racp1vm1_GRID_PUBLIC_SSH        | 21-Sep-2015 07:20 | 400              |
| 12102racp1vm1_HOSTS                  | 21-Sep-2015 07:20 | 422              |
| 12102racp1vm1_KEYSCAN                | 21-Sep-2015 07:20 | 96               |
| 12102racp1vm1_ORACLE_AUTHORIZED_KEYS | 21-Sep-2015 07:20 | 804              |
| 12102racp1vm1_ORACLE_PRIVATE_SSH     | 21-Sep-2015 07:20 | 1.6K             |
| 12102racp1vm1_ORACLE_PUBLIC_SSH      | 21-Sep-2015 07:20 | 402              |
| 12102racp1vm2.iso                    | 21-Sep-2015 07:20 | 39M              |
| 12102racp1vm2_CRSCONFIGPARAMS        | 21-Sep-2015 07:20 | 1.5K             |
| 12102racp1vm2_GRID_AUTHORIZED_KEYS   | 21-Sep-2015 07:20 | 800              |
| 12102racp1vm2_GRID_PRIVATE_SSH       | 21-Sep-2015 07:20 | 1.6K             |
| 12102racp1vm2_GRID_PUBLIC_SSH        | 21-Sep-2015 07:20 | 400              |
| 12102racp1vm2_HOSTS                  | 21-Sep-2015 07:20 | 422              |
| 12102racp1vm2_KEYSCAN                | 21-Sep-2015 07:20 | 96               |
| 12102racp1vm2_ORACLE_AUTHORIZED_KEYS | 21-Sep-2015 07:20 | 804              |
| 12102racp1vm2_ORACLE_PRIVATE_SSH     | 21-Sep-2015 07:20 | 1.6K             |
| 12102racp1vm2_ORACLE_PUBLIC_SSH      | 21-Sep-2015 07:20 | 402              |

# Last things first: This is the result

---

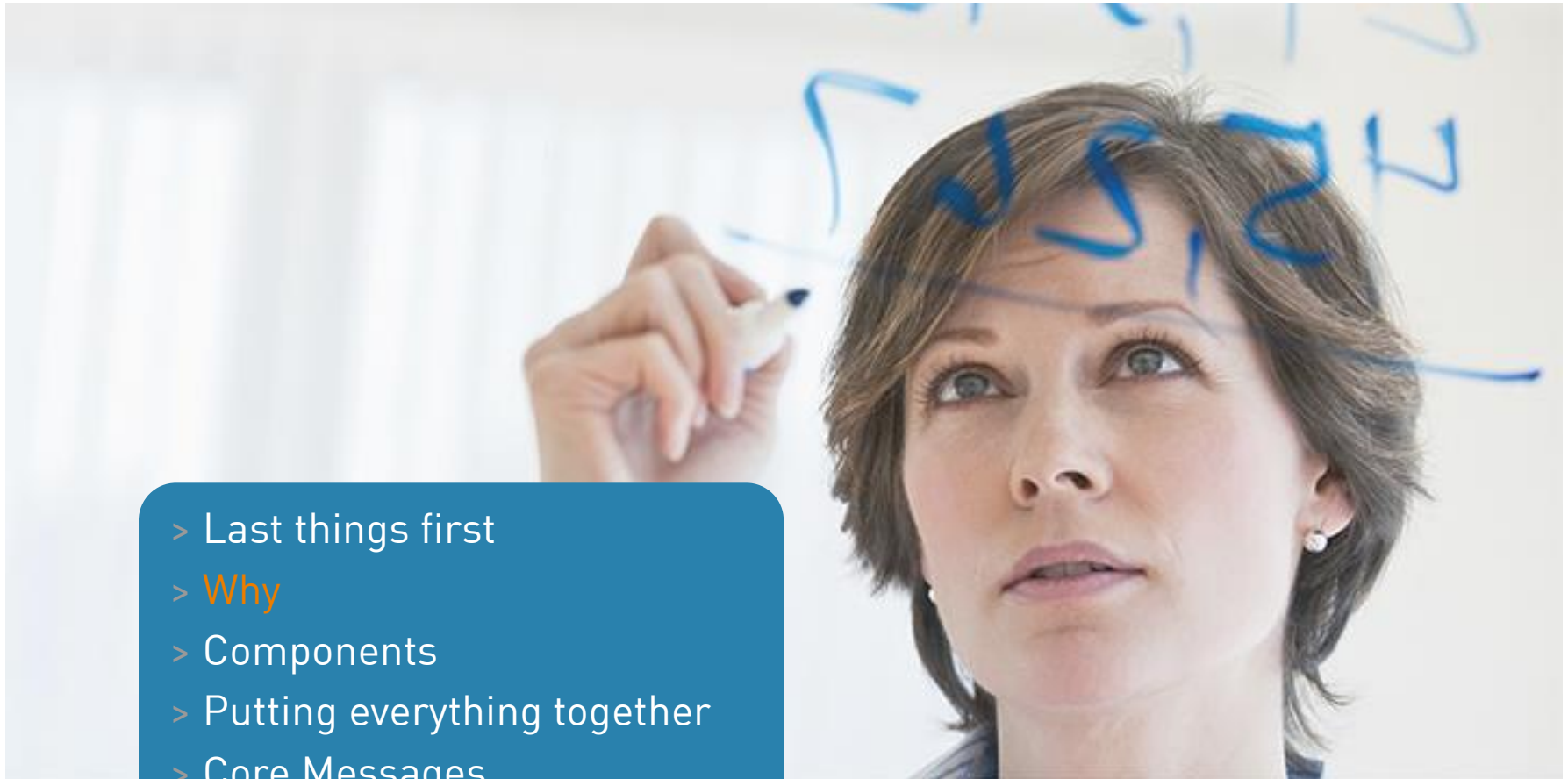
5. Wait a few minutes
6. Start the last VM
7. Sit back, relax, take your coffee and wait for the setup to complete (or just do some other work)
8. Your GI/RAC will be up and running in around 40 minutes (depending on your hardware) for 11gR2 and 1h10min for 12cR1

# Last things first: This is the result

```
[HOST N]# crsctl stat res -t
ora.CRS_DG.dg
      ONLINE  ONLINE          12102racp1vm1
      ONLINE  ONLINE          12102racp1vm2
ora.DATA.dg
      ONLINE  ONLINE          12102racp1vm1
      ONLINE  ONLINE          12102racp1vm2
ora.FRA.dg
      ONLINE  ONLINE          12102racp1vm1
      ONLINE  ONLINE          12102racp1vm2
ora.LISTENER.lsnr
      ONLINE  ONLINE          12102racp1vm1
      ONLINE  ONLINE          12102racp1vm2
ora.asm
      ONLINE  ONLINE          12102racp1vm1      Started
      ONLINE  ONLINE          12102racp1vm2      Started
...
ora.racdb.db
   1      ONLINE  ONLINE          12102racp1vm2      Open
   2      ONLINE  ONLINE          12102racp1vm1      Open
```

# Why?

Is it worth the initial effort?



- > Last things first
- > Why
- > Components
- > Putting everything together
- > Core Messages
- > Q&A

# Why?

It is all about Automation



# Why?

---

Does anybody in here has fun installing software?

Do you deliver Trainings? Why not use the same framework for providing customized environments to the trainees on their workstations by leveraging VirtualBox?

- > Give them the http link to the scripts and you're done
- > Did they destroy their environment? Just ask them to re-execute the setup script
- > You can control on how far the automated staging will go

# Why?

---

We can setup a GI/RAC environment consisting of 2-n Nodes without any modifications really fast (more on the details later)

Do you need to troubleshoot GI/RAC issues and do not have a test environment available?

Start the setup\_\* script and you are ready in around 40 minutes (1h10min for 12cR1)

# Why?

---

The whole setup is fully automated

This whole setup can be implemented in a real data center without much modifications

Highly customizable

All based on open source products

VirtualBox is available on all major platforms

- > Windows, Linux, OS X, Solaris amd64



# Components



- > Last things first
- > Why
- > **Components**
- > Putting everything together
- > Core Messages
- > Q&A

# Components

---

1. Cobbler
2. Kickstart
3. VirtualBox
4. One Script per Project:
  - > Single Node
  - > GI with RAC
  - > GI with RAConeNode
  - > Dataguard
  - > PostgreSQL
  - > ..... whatever you need

## What is Cobbler?

- > Cobbler is all about automated staging and provisioning
- > Cobbler is a Linux installation server
- > Cobbler uses distributions as its base
- > Cobbler uses profiles/templates for defining common attributes
- > Cobbler is open source

## How Does Cobbler Help?

“Automation is the key to speed, consistency and repeatability. These properties are critical to managing an infrastructure, whether it is comprised of a few servers or a few thousand servers. Cobbler helps by automating the process of provisioning servers from bare metal, or when deploying virtual machines onto various hypervisors.”



## What is Cobbler?

- › Was initially for rpm based distributions only and was hosted by the fedora project
- › Has been enhanced for Ubuntu
  - › Canonical Ltd uses Cobbler for test automation of OpenStack
- › RedHat Satellite uses Cobbler for provisioning



## What are Kickstart installations ?

- > Automated RedHat based installations
- > Based on a Kickstart file
- > The Kickstart file answers all the questions that are needed for an installation, e.g:

```
...
logvol / --vgname=vgroot --size=4096 --name=lvroot \
        --fstype=ext4 --fsoptions="barrier=0"
logvol /var --vgname=vgroot --size=8192 --name=lvvar \
        --fstype=ext4 --fsoptions="barrier=0"
...
%packages --excludedocs --nobase
kernel
yum
...
```

# Components (3) - VirtualBox



Anybody in here who does not know what VirtualBox is?

- > For this demo all is based on the host only network but this can easily be changed to any other network

Portable: VirtualBox hosts can be any of:

- > Windows
- > Linux
- > OS X
- > Solaris amd64

(almost) Platform independent

# Components (4) – One Script per Project

The project script is located on the cobbler server

The existing script for staging the GI/RAC project can be used as a template for other projects

- > Probably other projects will require a less complex script and can be implemented in at most half a day

**Templates already available:**

- > GI/RAC consisting of 2-n Nodes ( for 1-n participants, if needed )
- > Single Node Oracle 12c Database with Oracle Restart on OL7
- > PostgreSQL 9.4 on OL7

**Should be source code controlled by Subversion/Git**

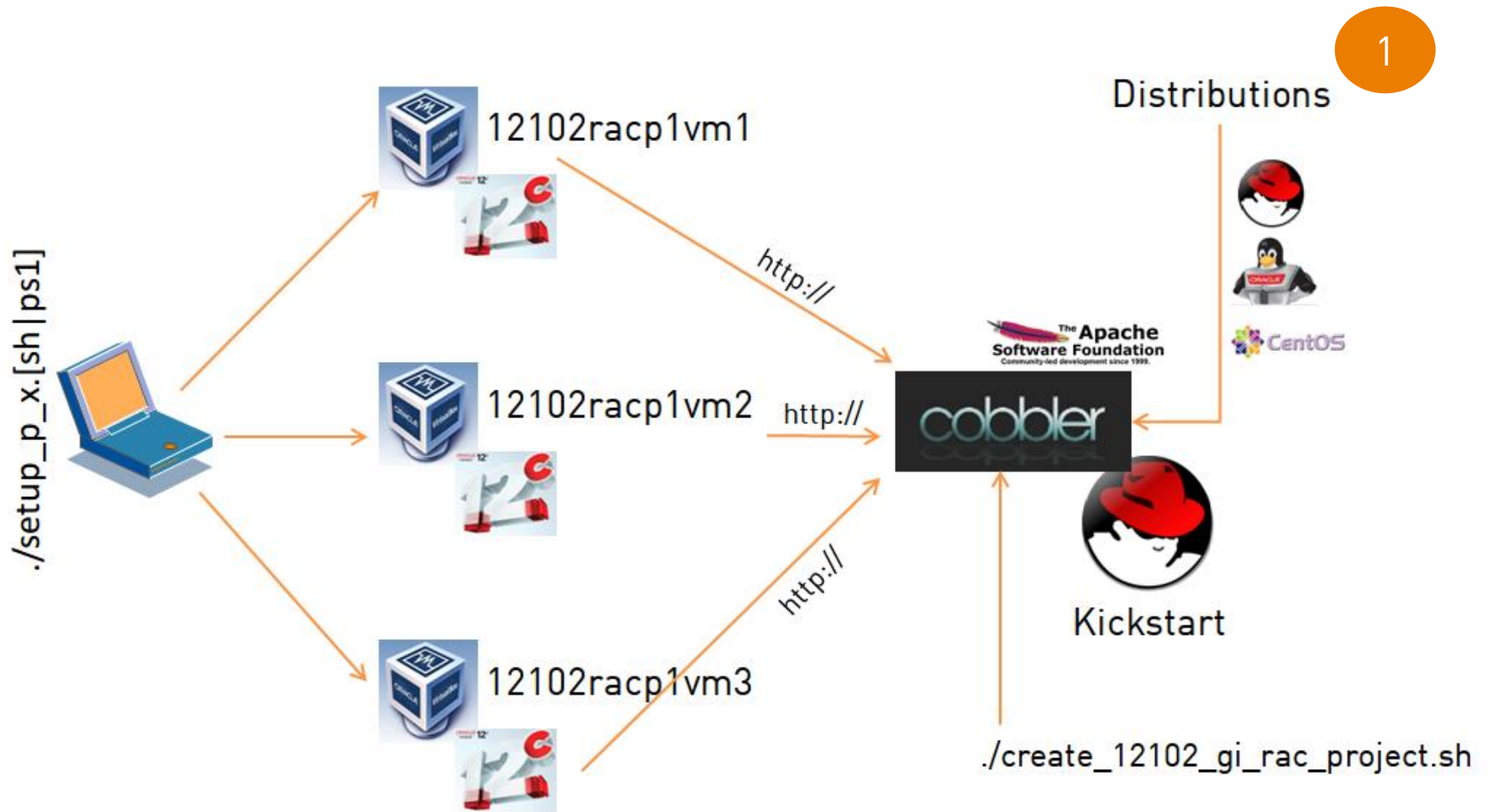
# Putting everything together



- > Last things first
- > Why
- > Components
- > Putting everything together
- > Core Messages
- > Q&A



# Putting everything together



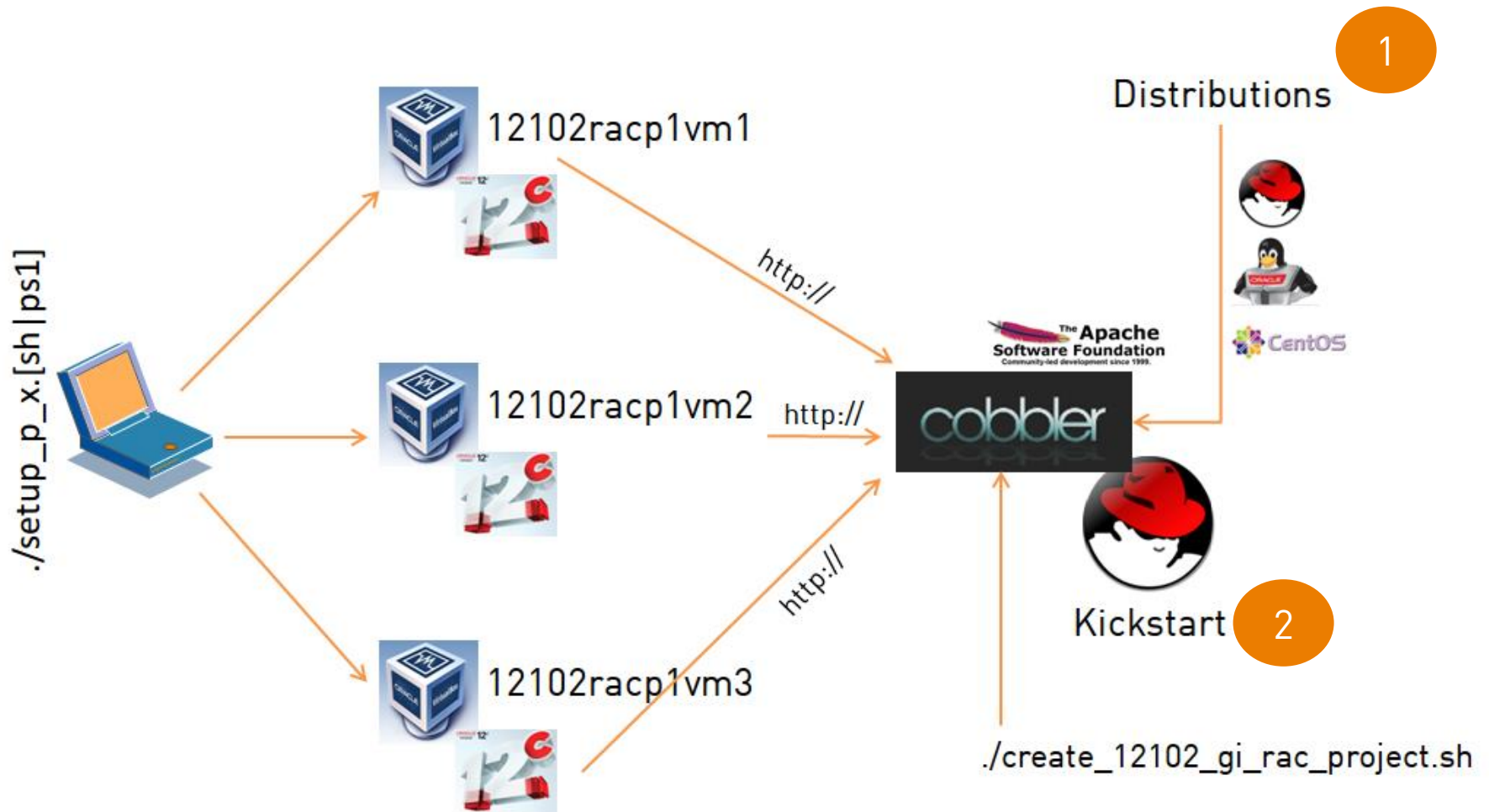
# Putting everything together

## Import a distribution into cobbler:

```
[cobbler ~]# mount -o ro,loop /var/tmp/oel71.iso /mnt
[cobbler ~]# cobbler import --name=oel71 --arch=x86_64 --path=/mnt
[cobbler ~]# cobbler distro list
CentOS7-x86_64
oel66-x86_64
oel71-x86_64
```

A loopback mount on the ISO is the common way to import a distribution into cobbler

# Putting everything together



# Putting everything together

## Create a Kickstart file:

```
[cobbler ~]# head -50 /var/lib/cobbler/kickstarts/ora12102_gi_rac.ks
auth --useshadow --enablemd5
bootloader --location=mbr --driveorder=sda --append="audit=1"
clearpart --all --initlabel
text
firewall --disabled
firstboot --disable
keyboard sg-latin1
lang en_US.UTF-8
...
```

# Putting everything together

## Create a profile and link it to the Kickstart file:

```
[cobbler ~]# cobbler profile add --name=ora12102_gi_rac --distro=oe171-x86_64
[cobbler ~]# cobbler profile list

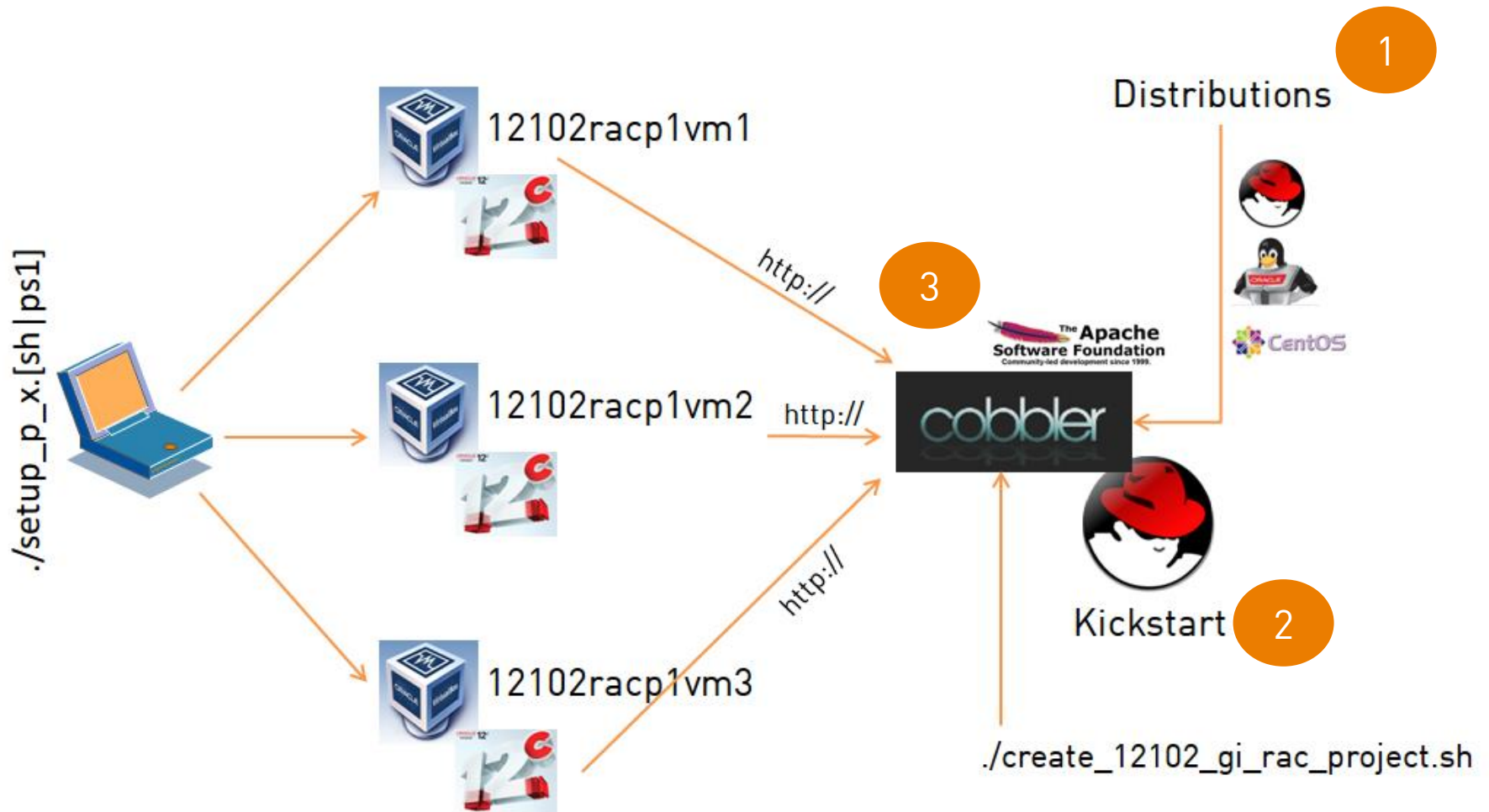
CentOS7-x86_64
oe166-x86_64
oe171-x86_64
ora12102_gi_rac
ora1202single
pg940

[cobbler ~]# cobbler profile edit --name=ora12102_gi_rac \
                                --kickstart=/var/lib/cobbler/kickstarts/ora12102_gi_rac.ks

[cobbler ~]# cobbler profile report --name=ora12102_gi_rac

Name                : ora12102_gi_rac
TFTP Boot Files     : {}
...
Kickstart            : /var/lib/cobbler/kickstarts/ora12102_gi_rac.ks
...
```

# Putting everything together



# Putting everything together

## Create a system:

```
[cobbler ~]# cobbler system add --name=host1 -profile=ora12102_gi_rac
[cobbler ~]# cobbler system edit --name=host1 --interface=eth0 \  
                --ip-address=192.168.56.110 \  
                --netmask=255.255.255.0 \  
                --hostname=host1 \  
                --static=1 --dns-name=host1.lab.ch
[cobbler ~]# cobbler system edit --name=host1 --interface=eth1 \  
                --ip-address=10.1.1.1 \  
                --netmask=255.255.255.0 \  
                --static=1
```

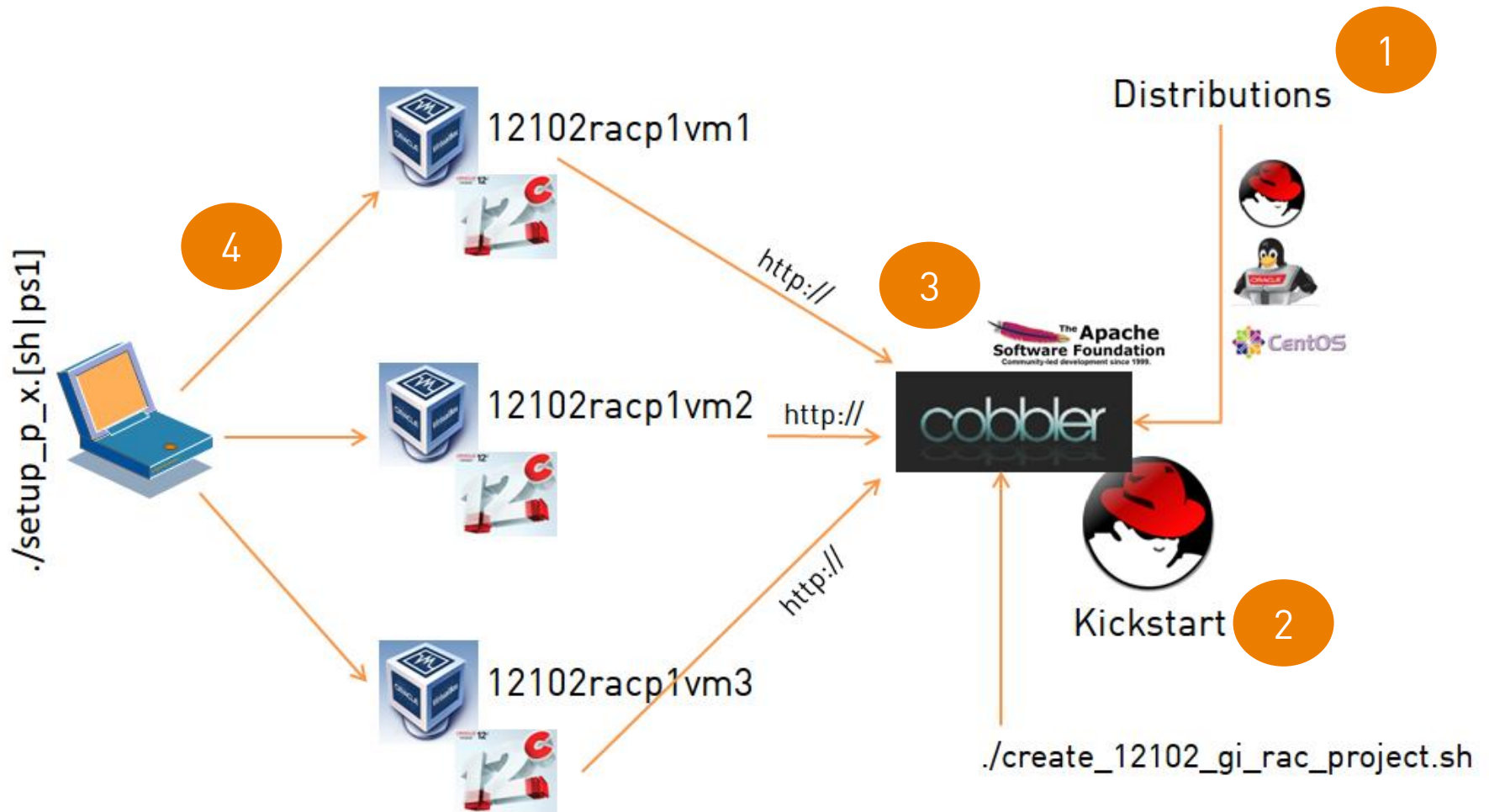
# Putting everything together

Prepare the sources you'll need for your setup:

```
[cobbler ~]# ls -la /var/www/html/sources/
-rw-r--r--. 1 root root      152 Jan 27 14:00 .bashrc_postgres
-rw-r--r--. 1 root root     6114 Jan 27 14:00 init_postgresql-9.4.0
-rw-r--r--. 1 root root 1673544724 Jan 28 08:43 linuxamd64_12102_database_1of2.zip
-rw-r--r--. 1 root root 1014530602 Jan 28 08:44 linuxamd64_12102_database_2of2.zip
-rw-r--r--. 1 root root 1747043545 Jan 28 09:20 linuxamd64_12102_grid_1of2.zip
-rw-r--r--. 1 root root  646972897 Jan 28 09:20 linuxamd64_12102_grid_2of2.zip
-rw-r--r--. 1 root root 1395582860 Jan  5 11:22 p13390677_112040_Linux-x86-64_1of7.zip
-rw-r--r--. 1 root root 1151304589 Jan  5 11:22 p13390677_112040_Linux-x86-64_2of7.zip
-rw-r--r--. 1 root root 1205251894 Jan  5 11:22 p13390677_112040_Linux-x86-64_3of7.zip
-rw-r--r--. 1 root root  22755540 Jan 27 12:04 postgresql-9.4.0.tar.gz
```



# Putting everything together



# Putting everything together

Generate the ISOs and boot your VM or your server from it:

```
[cobbler ~]# cobbler buildiso --systems="host1" \  
--iso="/var/tmp/host1.iso"
```

The screenshot shows a virtual machine management interface. On the left, there is a list of VMs: 12102racp1vm1 and 12102racp1vm2, both with a 64-bit icon and a power button icon, and both are 'Powered Off'. The main area displays the configuration for 12102racp1vm1. The configuration is organized into sections: General, Display, and Storage. The General section shows Base Memory: 4096 MB, Processors: 2, Boot Order: Hard Disk, Optical, Hard Disk, and Acceleration: VT-x/AMD-V, Nested Paging, PAE/NX, KVM Paravirtualization. The Display section shows Video Memory: 8 MB, Remote Desktop Server: Disabled, and Video Capture: Disabled. The Storage section shows Controller: ctl1, SCSI Port 0: hd\_root.vdi (Normal, 100.00 GB), SCSI Port 1: crs.vdi (Shareable, 4.88 GB), SCSI Port 2: asmarch.vdi (Shareable, 4.88 GB), and SCSI Port 3: asmdata.vdi (Shareable, 4.88 GB). A red box highlights the IDE Secondary controller, which is set to [Optical Drive] 12102racp1vm1.iso (39.35 MB). The Slave controller is currently empty.

|    |               |             |
|----|---------------|-------------|
| 64 | 12102racp1vm1 | Powered Off |
| 64 | 12102racp1vm2 | Powered Off |

**General**

- Base Memory: 4096 MB
- Processors: 2
- Boot Order: Hard Disk, Optical, Hard Disk
- Acceleration: VT-x/AMD-V, Nested Paging, PAE/NX, KVM Paravirtualization

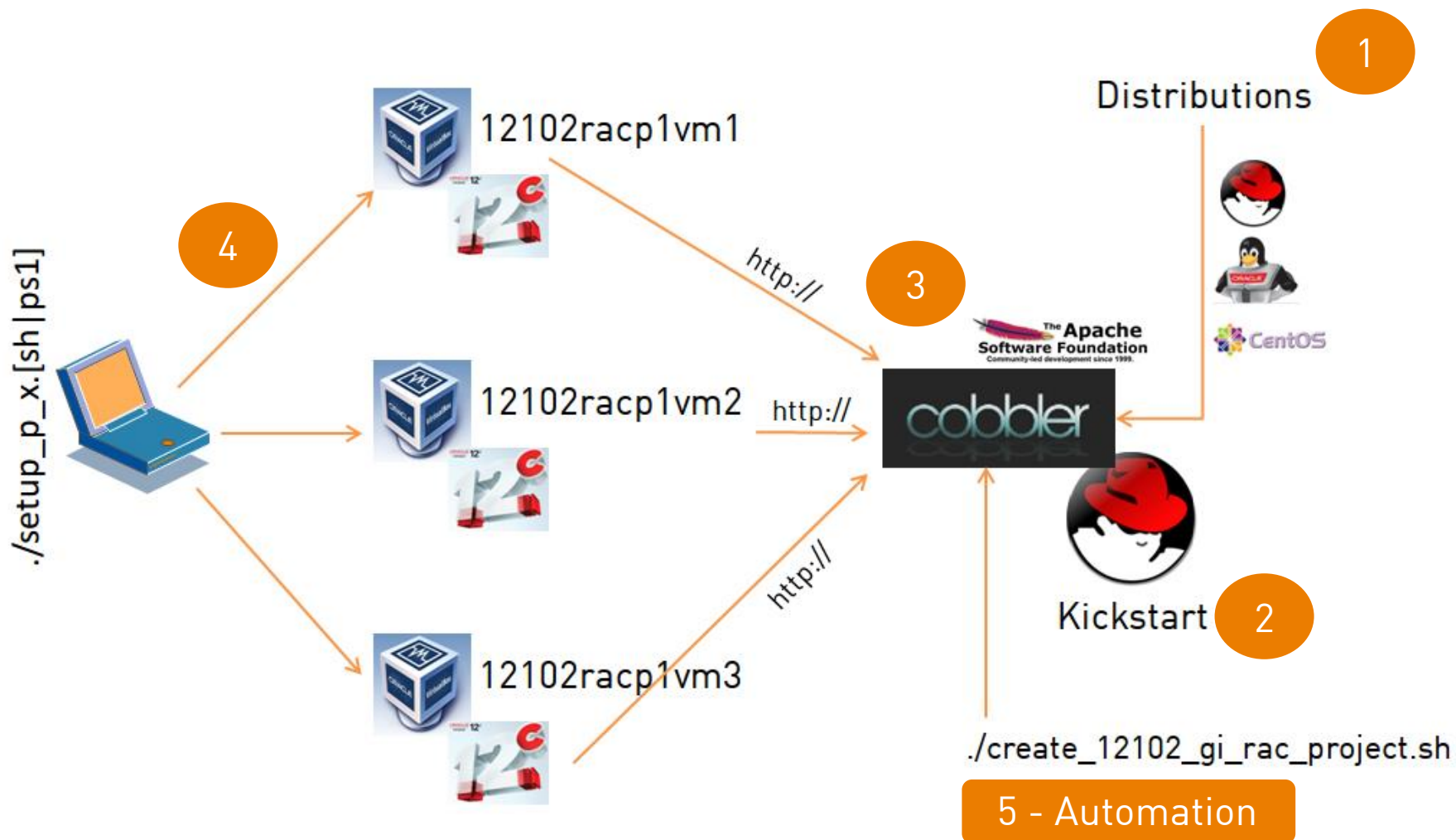
**Display**

- Video Memory: 8 MB
- Remote Desktop Server: Disabled
- Video Capture: Disabled

**Storage**

- Controller: ctl1
- SCSI Port 0: hd\_root.vdi (Normal, 100.00 GB)
- SCSI Port 1: crs.vdi (Shareable, 4.88 GB)
- SCSI Port 2: asmarch.vdi (Shareable, 4.88 GB)
- SCSI Port 3: asmdata.vdi (Shareable, 4.88 GB)
- Controller: ctl2
- IDE Secondary: [Optical Drive] 12102racp1vm1.iso (39.35 MB)
- Slave:

# Putting everything together



# Putting everything together

All these steps are automated by the project script(s):

```
[cobbler html]# grep -i cobbler create_12102_gi_rac_project.sh
COBBLERSERVER="192.168.56.101"
# the name the node will get in cobbler as well as the OS itself
# add the system to cobbler
cobbler system add --name=${SYSTEM_NAME} --profile=${PROFILE}
cobbler system edit --name=${SYSTEM_NAME} --interface=eth0 \
cobbler system edit --name=${SYSTEM_NAME} --interface=eth1 \
cobbler buildiso --systems="${SYSTEM_NAME}" \
echo
"BOOTISOURL=\"http://${COBBLERSERVER}/${LABNAME}/${BOOTISONAME}\"" >>
${VBSETUPSCRIPT}
cobbler sync 2>&1 > /dev/null
cobbler system list
```

# Putting everything together



- > Last things first
- > Why
- > Components
- > Putting everything together
- > Core Messages
- > Q&A

# Core messages

---

**Fully automated staging environment suitable for:**

- > Trainings
- > Personal test environments
- > (Almost) data center ready

**Can be highly customized**

**All based on open source products**

**With VirtualBox it runs on:**

- > Windows
- > OS X
- > Linux
- > Solaris amd64

# Core messages

All scripts are available at the dbi blog

<http://blog.dbi-services.com/>



The screenshot shows the top portion of the dbi services blog website. At the top, a grey navigation bar contains the text "Infrastructure at your Service" followed by a red Swiss flag icon. Below this is the dbi services logo, which consists of the letters "dbi" in blue and "services" in a smaller, grey font. To the right of the logo, there are two dropdown menus labeled "OFFERING" and "EXP". The main content area features a blue banner with the text "dbi services BLOG" in white. Below the banner, a blue box contains the following text: "Welcome to the dbi services Blog! This IT blog focuses on database, middleware, and OS technologies such as Oracle, Microsoft SQL Server & SharePoint, EMC Documentum, MySQL, PostgreSQL, Sybase, Unix/Linux, etc. The dbi services blog represents the view of our consultants, not necessarily that of dbi services. Feel free to comment on our blog postings."

# Putting everything together



- > Last things first
- > Why
- > Components
- > Putting everything together
- > Core Messages
- > Q&A



Any questions? Please do ask.

**Daniel Westermann**  
Consultant

Mobile +41 79 927 24 46

[daniel.westermann@dbi-services.com](mailto:daniel.westermann@dbi-services.com)

[www.dbi-services.com](http://www.dbi-services.com)



We look forward to working with you!

Infrastructure at your Service.



Invention & Infrastructure  
at your service.

