

# Using cobbler in a not-so-small environment

Christian Horn / [chorn@fluxcoil.net](mailto:chorn@fluxcoil.net)



OPEN SOURCE DATA  
CENTER CONFERENCE  
23. & 24. Juni | Nürnberg

## Outline

- 1 **cobbler-basics**
- 2 **details of our setup**
- 3 **infrastructure**
- 4 **community**
- 5 **cobblers/our future**

## ...Christian Horn?

- OpenSource enthusiast, Linux Engineer, Sysadmin
- 日本語を習う, cycling, reading
- first computer was a KC85/3 build in my hometown Muehlhausen
- RHCE in 2002, RHCA in 2009
- playing with all bsd/linux, mostly redhat/suse at work, debian on private boxes



# Basic steps of networkbased linux installation

Installserver



new linuxbox



- 1) acquire network-info via dhcp
- 2) fetching of kernel&initrd via tftp
- 3) fetching of instructionsfile,  
i.e. kickstartfile
- 4) get actual installdata and  
perform installation



## Whats sysadmins usual workflow for this?

- find out new servers mac, edit /etc/dhcpd.conf
- create new file in pxeboot-dir for new box
- copy some template to new ks.cfg
- customize ks.cfg, config hostname/packages etc.
- restart dhcpd, restart new server
- (find out the errors you made above)
- watch new box installing linux



# Whats cobbler anyway?

## Cobbler is..

A tool to simplify all aspects around deployment of operating systems.

## Our new workflow using cobbler:

- find out new servers mac
- tell cobbler about new server, ip/hostname/whatever
- (have cobbler taking care of syntaxchecking your input, pxeboot file, customizing ks.cfg, restarting dhcpd etc. )
- watch new box installing linux



## Most important cobbler commands..

### Introducing the system to cobbler \*

```
# telling cobbler about the system
cobbler system add \
  --name=sakura \
  --profile=rhel5u5-x86_64 \
  --mac=00:0c:29:fd:b2:e8 \
  --ip=10.0.22.13 \
  --hostname=sakura.dom.ain
# restarting dhcpd etc.
cobbler sync
```

\* webgui also exists



# cobbler glossary

## the cobbler distro

Describes a linux-distro, where the kernel is located and http-url for deployment.

## the cobbler profile

Contains basic descriptions how to deploy a system, which kickstart-template to use etc.

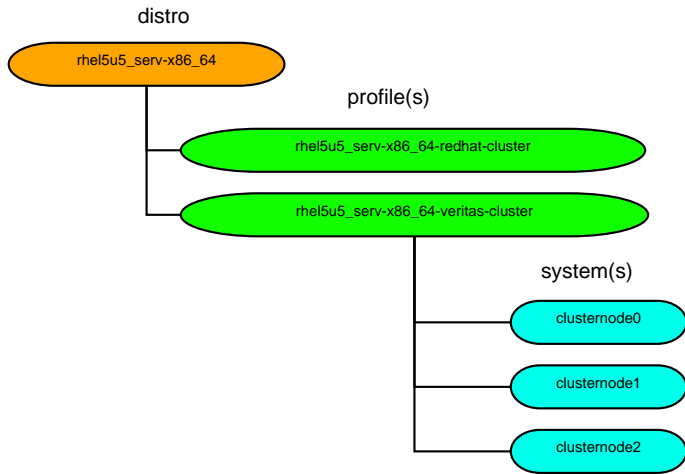
## the cobbler system

Describes everything specific to the system, ip/mac etc.





# How these terms play together..



## What cobbler does under the hood..

- 1 take a template ks.cfg
- 2 '#include' other codeparts, snippets in cobbler speak
- 3 have cheetah parse it: exchange the variables there with values from the distro/profile/system objects
- 4 result is handed over to to-be-deployed-box
- 5 watch deployment of the new box



## Why we started looking for something new..

- increasing number of linux-deployments
- several adminteam doing their own deployments, making own experiences => lets implement reasonable defaults
- skills of admins span a wide range
- increasing number of rules for deployment, this asks for standardization
- engineering-team providing support wants to have defined environments when asked for advise, i.e. debugtools installed



## Possible solutions

- colds - internal scripts originating from autoyast
- rh satellite - focused on updating/maintaining servers
- cobbler - open project targeting linux deployments
- fai - next good thing after cobbler, implementing for debian what kickstart/autoyast already provide
- writing something new - we do not suffer from NIH (mostly)



# comparison

	colds	satellite	cobbler	fai
initial costs		--		
commercial support	-	++		+
community support	--	+	++	++
sles deployments	+			
rhel deployments	+	++	++	

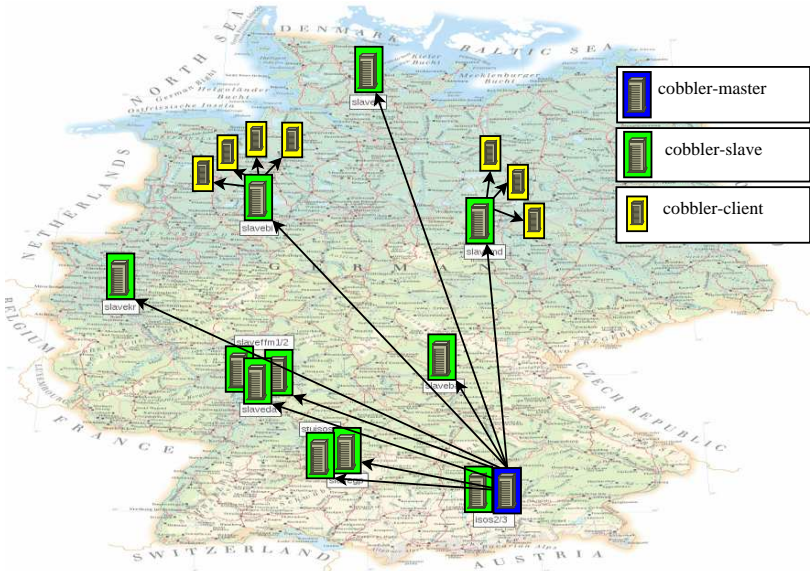


## requirement: wide area focus

- perform linux-deployments in locations all over the company - keeping wan-traffic and time for a deployment down
- PXE-boot usually restricted to single datacenters, work around this
- deployments should be done in easy and generic way; servers on datacenter as well as virtual machines on the laptop for testging are desired



# solution: multiple servers



## protocols used for master<->slave datatransfer

- cobbler-sync not used: its all-or-nothing
- transfer of distros (rhel5, sles11, fedora13 etc.):  
cobbler-slaves pull via rsync from master what they need
- transfer of snippets: low volume, high changerate,  
distinguish stable vs. head => subversion in use for this
- possible room for improvement: git. Could enable simple  
pulling of sysadmins additions from slaves to the master;  
could also enable clients to directly pull data from slaves





## requirement: Still deploy if PXE not possible

- 'cobbler buildiso' creates a cd.iso the cobbler-client can boot from. Kernel boots, starts kickstart, continues as with pure PXE-install
- alternative: iso booted, real kernel fetched via http and started via kexec
- iso-boot via remote-control-board or optical drive of server
- upcoming: NICs with gpXE appearing



## Whats running on the cobbler-slaves?

- RedHat rhel5/Centos
- basic services: httpd, dhcp, tftpd via xinetd
- cobblerd
- named delivering the local zone managed by cobbler
- virtualization for easy testing of deployments in VMs (KVM on some workplaces, vmware-server on the other cobbler-slaves)



## Whats cobbler doing for us?

- dhcp/pxe management, preparing instructions for deployments of rhel/suse sles/fedora
- maintain dns-zones
- configure network-interfaces of cobbler-clients
- deployment & configuration of software: backup, monitoring clients etc.
- hookup of the cobbler-client to ldap for authentication/authorization
- strip services down, do hardening



## Challenging areas: SuSE & cobbler

- cobbler/cheetah handle autoyast xml-files fine
- detailed network-configuration via autoyast doesnt work as of sles11
- code should be shared among rhel & sles deployments but simple including of snippets in autoyast.xml doesnt work because xmlish header/footer required

### our solution

```
#set global $wrappedscript = 'post_config_ntpd'  
$SNIPPET('sles/post_sles_scriptwrapper')
```

This and our other sles-snippets are on the net.



## more realistic host-adding..

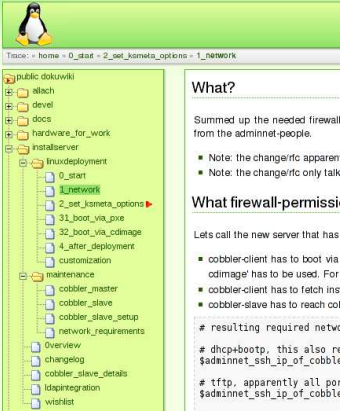
```
cobbler system add --name=chhorn_rhel5 \  
  --profile=rhel5u5_serv-x86_64 \  
  --mac=00:0c:29:fd:b2:e8 \  
  --ip=10.0.22.13 --subnet=255.255.255.0 --static=1 \  
  --hostname=rhel5.dom.ain \  
  --dns-name=rhel5.isos3.cobbler \  
  --ksmeta='cProxyPW=asdfgh IDNumber=ID1234 noprelink \  
  defaultServerList=12.32.34.4#10.2.3.11 \  
  defaultSearchBase=dc=as,dc=lx,dc=stu,dc=com \  
  BootGateway=10.0.22.1 MonitorSystemID=S0123 \  
  backup=hibacknetbackup avmon=nagios packagesel=tiny'
```



# surrounding infrastructure: docs

## documentation:

- how to setup new cobbler-slaves, how to do deployments
- done with dokuwiki/php/apache
- and more static docs in a repo with html/pdf generator



The screenshot shows a DokuWiki interface with a file tree on the left and a text area on the right. The file tree is expanded to show the 'linuxdeployment' directory, with '1\_network' selected. The right side of the page contains the text of the '1\_network' page, which discusses firewall requirements for cobbler-slaves.

```
Trace: => home => 0_start => 2_set_kameta_options => 1_network
```

public dokuwiki

- attach
- devel
- docs
- hardware\_for\_work
- installserver
- linuxdeployment
  - 0\_start
  - 1\_network
  - 2\_set\_kameta\_options
  - 31\_boot\_via\_pxe
  - 32\_boot\_via\_cdimage
  - 4\_after\_deployment
  - customization
- maintenance
  - cobbler\_master
  - cobbler\_slave
  - cobbler\_slave\_setup
  - network\_requirements
- Overview
- changelog
- cobbler\_slave\_details
- ldapintegration
- wishlist

### What?

Summed up the needed firewall from the adminnet-people.

- Note: the change/rfc appear
- Note: the change/rfc only talk

### What firewall-permissi

Lets call the new server that has

- cobbler-client has to boot via cdimage' has to be used. For
- cobbler-client has to fetch ins
- cobbler-slave has to reach col

```
# resulting required netw
```

```
# dhcp+bootp, this also re
```

```
$adminnet_ssh_ip_of_cobble
```

```
# tftp, apparently all por
```

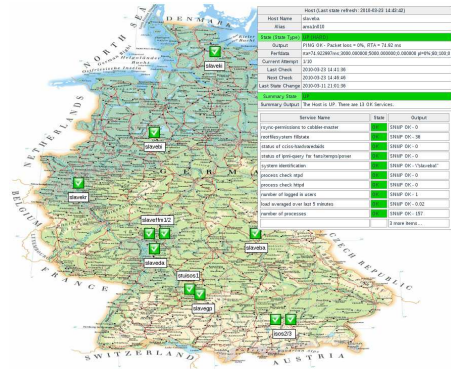
```
$adminnet_ssh_ip_of_cobl
```



# infrastructure: availability monitoring

## monitoring:

- we want to detect full disk, dying raid, killed httpd etc.
- ldap containing data of the cobbler-slaves
- snmpd running on cobbler-slaves
- icinga/nagvis polling the slaves, icinga-configfiles generated from the ldap-db



# infrastructure: performance monitoring

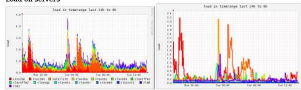
performance:

- to see stressed slaves
- collected with some perl/net::snmp/rrdtool
- delivered as cgi by apache

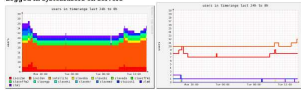
issue tracker:

- handle requests for enhancements and bugs
- a trac does this currently

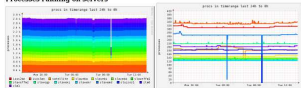
Load on servers



Logged in systemusers on servers



Processes running on servers





## community interaction

- great community around cobbler, RedHat paying coders (got bit quiet on ML thou since Michael DeHaan left)
- since we were among first to deploy SLES related snippets were published
- cobbler is part of satellite as of 5.3
- part of spacewalk => community grows



## upcoming cobbler topics

- improving deployment of virtual systems
- improvement of !rh-distros like debian and suse
- remote-management of cobbler-servers via an api
- cobbler doing all of the server-touching like resetting via RSB etc.
- deployments of Mac (!PXE but different booting)
- already doing image-based deployments, improving also scripted win-deployments



# todolist

## constantly going on

- training sysadmins
- fixing bugs, scripting new features
- preparing new distros

## our todolist

- getting each system first into ldap, filling cobbler from there
- this as base to use puppet
- maybe: profile 'cobbler-slave'
- maybe: script openvpn tunnelendpoint – allows install via layer2-transport over layer3



# Questions?

<http://fluxcoil.net>



**Next OSDC: 06. & 07.04.2011**

**OPEN SOURCE DATA  
CENTER CONFERENCE**  
23. & 24. Juni | Nürnberg