Installation Tools for Clusters

Rajesh K.,

Computer Division,

BARC

Outline of the presentation

- ❑ Cluster Intro
- ❑ Steps involved in a cluster installation
- ❑ Different approaches to installation
- ❑ Issues in cluster installation
- ❑ Auto installation mechanics
- ❑ Types of installation tools
- ❑ Various available tools for cluster installation
  - – Anaconda (Kickstart)
  - – System Imager
  - – SIS, OSCAR

A Typical cluster
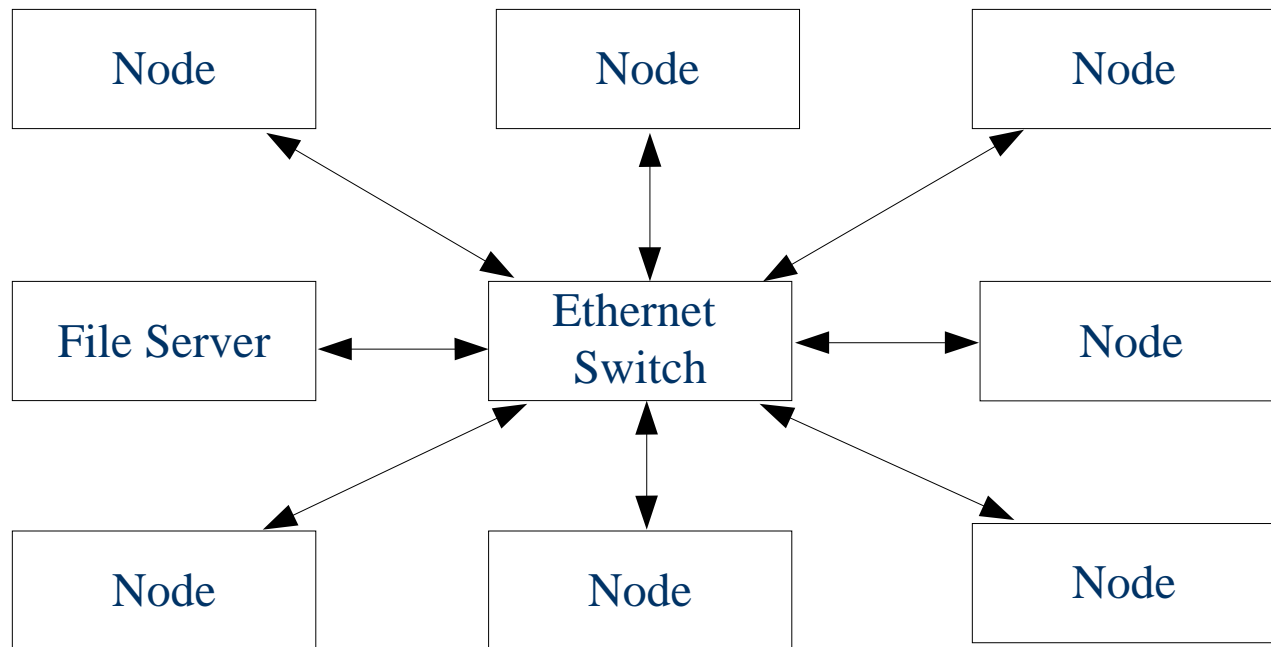
A typical Linux cluster

- ❑ Clusters are the most popular kind of parallel machines being built today
- ❑ A cluster consists of a set of machines performing different roles
  - – Nodes (for computation)
  - – File servers (for storing users files)
  - – Front end (optional, also sometimes called master node)
  - – Management and monitoring nodes (optional)
- ❑ All machines run Linux OS
- ❑ Interconnected using standard networks and protocols
- ❑ Configuration of each machine depends on the role

Steps involved in installation of a typical cluster

- Installation of Linux OS on each machine
  - Creation of partitions
  - Selection of packages
  - IP Address, root password, timezone etc.
- Configuration of Linux on each machine
  - Configure NIS, NFS, Automount, NTP
- Installation of additional software components
  - Compilers, MPI, PBS etc.

Cluster nodes installation – Different approaches

- ❑ Local boot + Local installation (using local devices such as CD)
  - Time-consuming, error prone
  - Lots of human intervention
- ❑ Local boot + Network installation
  - Needs a boot diskette/boot CD
  - Needs NFS server setup where the distribution is maintained
  - Faster than the previous since installations can be overlapped
- ❑ Network boot + Network installation
  - Eliminate boot diskette/CD
  - Needs PXE compliant network card or port
  - Setup of TFTP server

Cluster nodes installation – Different modes

❑ Interactive: Human involvement

❑ Non-interactive or Automatic:
- Less human interaction and less error prone
- Programs such as Anaconda (Kickstart) and Yast
- Simulate interactive answering through configuration files (which may contain state info)
- Postinstall scripts for customization
- DHCP can be setup for delivering state info (IP address and hostname) so that configuration file can be unchanged across machines
- Specialized tools such System Imager, SIS also available

Automatic installation tools for clusters - Need

- Installation of large numbers of identical machines – traditional methods time consuming and error prone
- Configuration changes or software updates have to be replicated on all machines – easy to miss out some machines

Objectives of an installation tool

- Fully automatic installation
- Fast
- Distribution independent
- Easy upgradation
- Examples of auto installation tools
  - Kickstart, FAI (Fully Automatic Installation), AutoYAST
  - System Installation Suite
  - OSCAR (actually a complete cluster suite)

Issues in cluster installation

- Nodes in a cluster are not really identical
- Each node contains unique state
  - IP Address
  - MAC Address
  - Hostname
  - Specific applications
- Automatic installation tool should handle the state information of each node

Dynamics of a network installation

1. Node requests for an IP address and hostname using DHCP

3. Node configures itself using this IP address and requests the server for the boot loader image using TFTP

5. Node boots using this bootstrap loader which in-turn asks the server for the linux kernel, initial root filesystem and configuration of auto installer

7. Autoinstaller starts the installation. After installation, post install scripts are run and the node is ready

Server

2. Server assigns IP address and hostname according to rules in dhcp configuration file.

4. Server provides the boot loader image using TFTP

6. Server supplies the linux kernel image, initial root file system and auto install configuration files to node
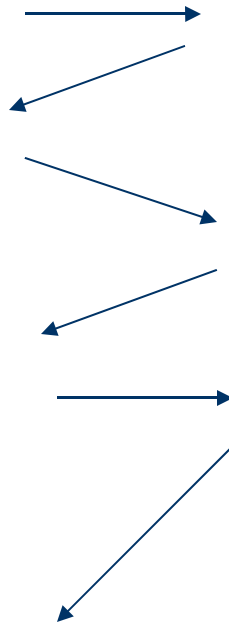
Delivering IP address and hostnames

- ❑ Each node in the cluster has to have a unique IP address and hostname

- ❑ During automatic installations this info has to be delivered to each node

- ❑ Dynamic Host Configuration Protocol (DHCP) is used to assign IP addresses to requesting nodes
  - Assign dynamically from an available pool of IPs
  - Assign from a MAC address – IP address mapping which is present in /etc/dhcpd.conf

- ❑ DHCP also assigns hostnames from /etc/hosts which contains hostname to IP mapping

DHCP: Assigning IP addresses to nodes

- ❑ Define a pool of available addresses and let DHCP assign from it as and when machines are booted. The addresses are allotted in the order of machine booting

- ❑ Define a MAC address – IP address mapping. In this case DHCP allots the particular addresses to particular nodes
  - We have to know the MAC addresses of nodes beforehand

- ❑ Allocate a temporary IP address from a predefined range. A program running in the node can sense this temporary address and request the user for a permanent IP address which is updated into the DHCP configuration file to make it permanent
  - Approach used in BARC's ANUPAM clusters with kickstart

- ❑ A program on the server automatically collects MAC address of each node when it makes the first DHCP request, asks the user for the IP address to be assigned and enter it into the above mapping to make it permanent
  - Approach used by OSCAR

Delivering bootstrap image, boot kernel and initial root filesystem

- In manual installation using CD, the CD contains the boot loader and kernel images for installation
- In networked installation, these have to be brought from a central server
- Apart from IP address and hostname, DHCP also supplies the IP address of the central server and the filename of the boot image
- Node downloads the boot image from the central server using TFTP (Trivial File Transfer Protocol)
- Bootstrap loader in-turn downloads its configuration file, the kernel image and initial root file system
- Initial root file system (initrd) contains the program which starts the auto installation.

Bootstrap loaders for linux

- ❑ A working Linux machine generally uses LILO or GRUB as the bootstrap loader
  - – LILO or GRUB load linux kernels from filesystems in the hard disk
  - – Configuration files are /etc/lilo.conf and /etc/grub.conf respectively
- ❑ During installation using floppy media or CDROM, SYSLINUX boot loader is used
  - – SYSLINUX loads linux kernels from floppy or CD media
  - – Config file syslinux.cfg should also be present in the same floppy or CD
- ❑ During installation from network, PXELINUX boot loader is used (downloaded using TFTP)
  - – PXELINUX loads linux kernels from network using TFTP
  - – Config file is also brought from the server using TFTP

Mechanics of auto installation

❑ BIOS in the node loads the bootstrap loader from the floppy, CD (syslinux) or network (pxelinux)

❑ The bootstrap loader looks for its configuration file in the floppy/cd or in the server

❑ This config file contains information about the linux kernel to be loaded, the initial root file system image and some arguments to the kernel

❑ Bootstrap loader loads the linux kernel and the initial root file system (initrd) and gives control to the kernel

❑ Kernel runs the auto install program present in the initrd

❑ This auto install program looks for its config file which may be present in the floppy/CD or the central server and loads it.

❑ Auto installation continues according to the rules given in the config file

❑ Installation can be customized by supplying different configuration files to different classes of machines in the cluster

Types of installation tools

❑ Package based installation
  – For example: Kickstart, AutoYast, FAI
  – The config file specifies the packages to be installed in the node. Different config files for different kinds of nodes
  – Distribution specific (Redhat, SUSE, Debian)
  – Works with dissimilar nodes

❑ Image based installation
  – For exampe: System Imager
  – Entire contents of one node is replicated on all other nodes
  – A central "image server" may store directory contents of various kinds of nodes as images.
  – Independent of distribution, but more space requirements
  – Requires all nodes to be identical; else involves additional work

Kickstart

- Native installation tool from Redhat
- Driven by a kickstart configuration file which contains "answers" to question posed by the installer
    - Partitions and sizes
    - Root password and timezone
    - Packages to be installed
    - IP address, hostname etc.
    - Post install actions in the form of shell scripts
- 'ks.cfg' file can be present in the boot floppy or a central server (delivered by NFS)
- State info (IP address and hostname) can be 'hardcoded' in the config file or delivered by DHCP.

Sample Kickstart config file

```
lang en_US

langsupport --default en_US en_US

keyboard us

mouse generic3ps/2

text

timezone --utc Asia/Calcutta

rootpw xyz123

nfs --server 90.2.5.225 --dir /home/usr1/Redhat-
8.0

network --bootproto dhcp

install

auth --enablenis --nisdomain compudom

skipx

firewall --disabled

zerombr yes

clearpart --all

part /  --size 1000

part /usr  --size 5000

part /tmp  --size 1000

part /var  --size 1000

part swap --size 4000
```

```
part /usr/spool --size 1000

part /usr1  --size 100 --grow

lilo --location mbr

reboot

%packages

@Everything

%post --nochroot

cp /mnt/source/postfiles/hosts /mnt/sysimage/tmp/hosts

cp /mnt/source/postfiles/rhosts   \
                      /mnt/sysimage/root/.rhosts

rm -f /mnt/sysimage/etc/ntp.conf

cp /mnt/source/postfiles/ntp.conf /
mnt/sysimage/etc/ntp.conf

cp /mnt/source/postfiles/cscript /mnt/sysimage/tmp/

mkdir -p /mnt/sysimage/usr/local/updatedhcp
```
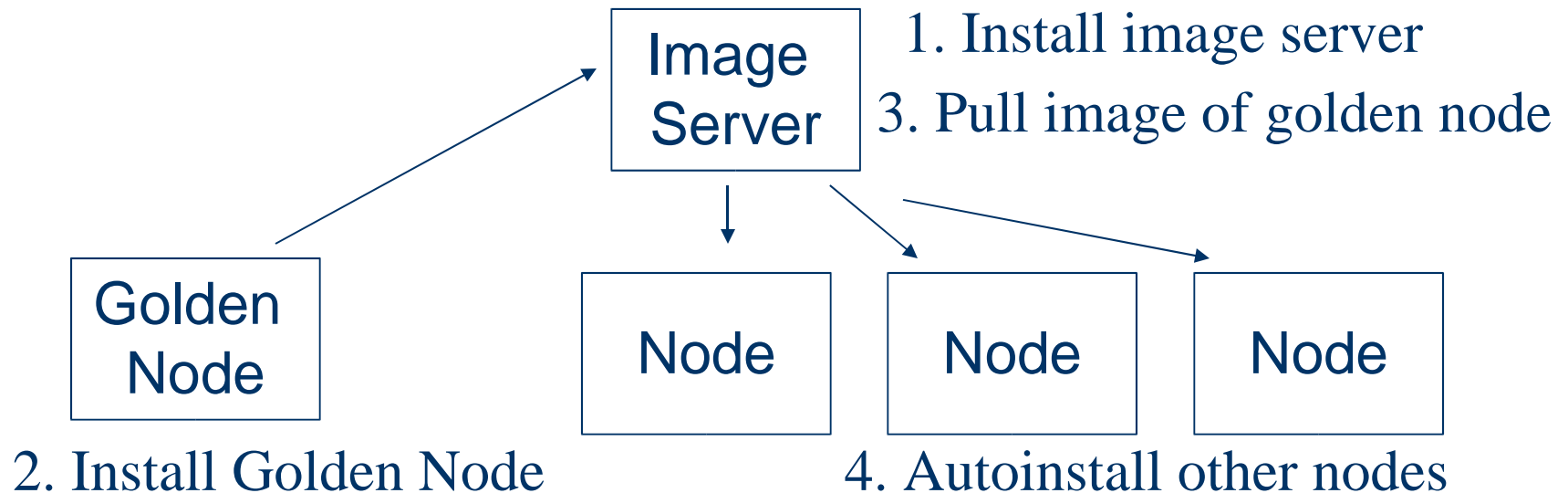
Kickstart : Steps

❏ Install a central file server using usual methods

❏ Copy whole distribution from CDs to some directory in the file server

❏ Export the directory over NFS

❏ Create Kickstart configuration file – put this file in the boot floppy or on the central server

❏ If network booting is required, download and configure PXELINUX

❏ Boot the node – installation through kickstart automatically begins

System Imager

- An image based installation tool for clusters independent of distribution
- Replicates the entire contents of a "golden node" (called an image) onto a large set of machines
- Faster installation as compared to the package based installers such as kickstart
- All machines need to be identical in hardware – same type of hard disks, same ethernet cards, same display adapters etc.
- Multiple different types of images can be stored in an image server for different classes of machines in a cluster
- Needs more disk space on the image server

System Imager mechanics

Image Server

1. Install image server

3. Pull image of golden node

Golden Node

Node

Node

Node

2. Install Golden Node

4. Autoinstall other nodes

Installation procedure using System Imager (Part 1)

- ❏ Download System Imager software from http://www.systemimager.org/
  - – Presently it is available as an entire SIS package
- ❏ Choose an image server (generally the file server of the cluster)
  - – This should have adequate disk space; DHCP and TFTP software should be available
  - – System Imager boot package and server package should be installed
- ❏ Choose a golden client machine
  - – Should be identical in hardware to all other nodes on which its contents is to be replicated
- ❏ Install Linux on the golden client using normal methods
  - – Install and configure all required software
  - – Install System Imager client package on the golden client

Installation procedure using System Imager (Part 2)

- ❑ Run 'prepareclient' command on the golden client
- ❑ Run 'getimage' on the image server to pull the image of the golden client to a directory in the server. This image will have a specific name. This command also creates 'autoinstall scripts' which will be used by the nodes when installing
- ❑ Choose and configure method of assigning IP addresses to nodes
  - – By DHCP: use mkdhcpserver command
  - – Statically: by creating a local.cfg file containing the IP address and hostname
- ❑ Run 'addclients' command on the image server
  - – Adds entries into /etc/hosts (IP address to name mapping)
  - – Creates mapping between hostname and image

Installation procedure using System Imager (Part 3)

- Auto install the golden client's image on other nodes
  - Create boot floppy or CD using mkautoinstalldiskette or mkautoinstallcd commands
  - Boot nodes with this floppy or CD
  - If network booting is needed, configure PXE and TFTP using mkbootserver command

Updating nodes of a cluster

- ❑ Required when some software needs to be upgraded or installed on all nodes
- ❑ Update the image stored in the image server and then run updateclient command on all nodes
- ❑ Images in the image server can be updated by
  - – Directly modifying files in the image directory
  - – OR modify the golden client and then pull its image on to the image server

System Installation Suite (SIS)

- Combination of System Installer, System Imager and System Configurator Programs

- System Installer: Tool to create images on an image server without having a separate golden client

- System Configurator: Standalone program which runs after a system install and provides hooks to configure the system parameters such as network interfaces, date and time, boot loader and also the hardware such as device drivers etc.

OSCAR

- ❑ A complete linux cluster suite
- ❑ Currently supports Redhat based distributions
- ❑ Uses SIS to actually perform the installation
- ❑ Value additions such as Ganglia, OpenPBS, MPI, PVM etc.
- ❑ C3 cluster wide command set
- ❑ Graphical user interface for all System Imager steps
- ❑ Information about cluster nodes and their configuration are stored in a database

Summary

- Automatic installation of clusters is necessary to save time and effort
- Issues in cluster installation
    - Delivery of IP address, hostname etc.
    - Different modes of booting – local media, network
    - Handling different classes of machines
    - Propogating updates
- All major linux distributions have their own auto installation tools
- Tools such as SIS achieve distribution independence
- All tools have learning curves, but once they are mastered they prove to be very versatile

Acknowledgements and References

- Vaibhav Kumar, Computer Division, BARC
- References:
  - http://sisuite.sourceforge.net
  - http://www.systemimager.org
  - http://oscar.openclustergroup.org
  - http://www.redhat.com
  - Google