Extreme Cloud Administration Toolkit

xCAT on z/VM and z/Linux

Version 0.11

Doc. Version	0.11
Date	04/20/10
Owner	Thang Pham
Author	Thang Pham (thang.pham@us.ibm.com)
	Wesley Yee (wesley.yee@us.ibm.com)
	Mike MacIsaac (mikemac@us.ibm.com)
File Name	xCat on zVM.odt

Document Abstract

The purpose of this document is to get a user started with xCAT on z/VM and z/Linux by providing the user with an overview, an installation, and a quick start guide on basic z/VM and z/Linux administration.

Table of Contents

Document Control Information

This chapter summarizes the change history for the document.

Revision History

Date	Version	Description	Author
08/05/09	0.1	Author	Thang Pham
08/07/09	0.2	Modify design	Thang Pham
09/02/09	0.3	Add installation, getting started, and limitations sections.	Thang Pham
09/08/09	0.4	Modify document based on review	Thang Pham
09/09/09	0.5	Modify document based on review	Thang Pham
09/14/09	0.6	Add *Installation of Ganglia* section	Wesley Yee
09/21/09	0.7	Add *Clone a Virtual Server* section	Thang Pham
10/05/09	0.8	Add *Using Ganglia with xCAT* section	Wesley Yee
12/08/09	0.9	Add *Installing Linux Using AutoYast or Kickstart* section	Thang Pham
01/05/10	0.10	Modify document based on review	Thang Pham
04/30/10	0.11	Update document based on new support	Thang Pham

Terminology

The purpose of this section is to outline the terminology used within this design document.

DirMaint	CMS application that helps manage an installation's VM directory.
Ganglia	*"Ganglia consists of two unique daemons (gmond and gmetad), a PHP-based web frontend and a few other small utility programs. Gmond is a mult-threaded daemon which runs on each cluster node you want to monitor. Gmetad is the daemon that monitors the other nodes by periodically polling them, parsing the collected XML, and saving all the numeric, volatile metrics to the round-robin databases."* - Ganglia Development Team
Image	A snapshot of an operating system running on a virtual server that is packaged and compressed into a file.
Life cycle	A collection of tasks that include: power on/off of a virtual server, and create/edit/delete of a virtual server.
OVF	*"The Open Virtualization Format (OVF) describes an open, secure, portable, efficient and extensible format for the packaging and distribution of virtual appliances. It enables portability and simplifies installation and deployment of virtual appliances across multiple virtualization platforms."* -Distributed Management Task Force
SMAPI	The Systems Management APIs simplify the task of managing many virtual images running under a single z/VM image.
Virtual server	A server composed virtualized resources. An operating system can be installed on a virtual server.
VMCP	Module that allows execution of CP commands.
CP	*"The Control Program (CP) is the operating system that underlies all of z/VM. It is responsible for virtualizing your z/Series machine's real hardware, and allowing many virtual machines to simultaneously share the hardware resource."* - IBM
xCAT	xCAT (Extreme Cloud Administration Tool) is a toolkit that provides support for the deployment and administration of large cluster environments.
zHCP	HCP (System z Hardware control point) is a Linux virtual server that interfaces with SMAPI and manages other virtual servers on z/VM.
AutoYaST	*"AutoYaST is a system for installing one or more SUSE Linux systems automatically and without user intervention. AutoYaST installations are performed using an autoyast profile with installation and configuration data."* -SUSE
Kickstart	*"Automated installation for Red Hat. It uses a file containing the answers to all the questions that would normally be asked during a typical Red Hat Linux installation."* -Red Hat

Support on z/VM and z/Linux

The purpose of this section is to provide a list of the current functionalities of xCAT on z/VM and z/Linux.

1. Life cycle support for xCAT on z/VM, which includes:
   a) Power on/off virtual server
   b) Create virtual server
   c) Edit virtual server configuration
   d) Delete virtual server
2. Software and hardware inventory of a virtual server
3. Cloning a virtual server
4. Installation of zLinux on a virtual server using kickstart/autoyast
5. Monitoring of zLinux systems using ganglia
6. Remote shell
7. Linux OS version update

Design Architecture

The purpose of this section is to provide information on the design architecture of xCAT on z/VM and z/Linux.
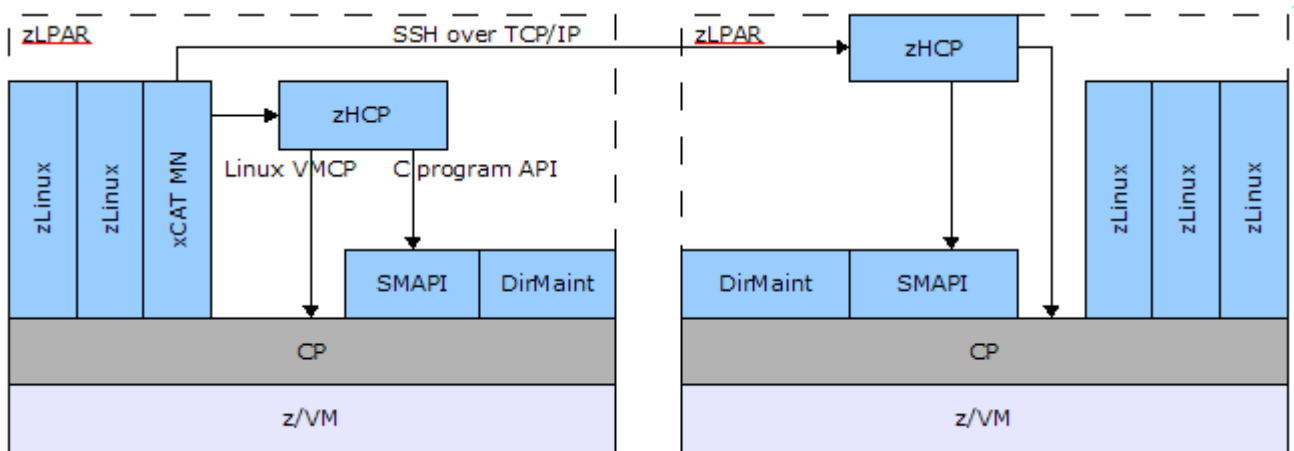

Figure 1. Shows the layout of xCAT on System z.

xCAT is used to manage virtual servers within multiple z/VM partitions. The xCAT management node (MN) can be installed on any zLinux. It communicates with a System z hardware control point (zHPC) over SSH, which is required on each z/VM partition in order to manage it. The zHCP is a zLinux, and its purpose is to interface with the SMAPI and CP layer. It utilizes a C socket interface to communicate with the SMAPI layer and VMCP Linux module to communicate with the CP layer. xCAT mainly relies on Linux VMCP module to perform its actions. However, more complicated actions, such as creating/deleting/editing virtual servers, require SMAPI.

Prerequisite

The purpose of this section is to detail what is required before you setup xCAT on z/VM and z/Linux.
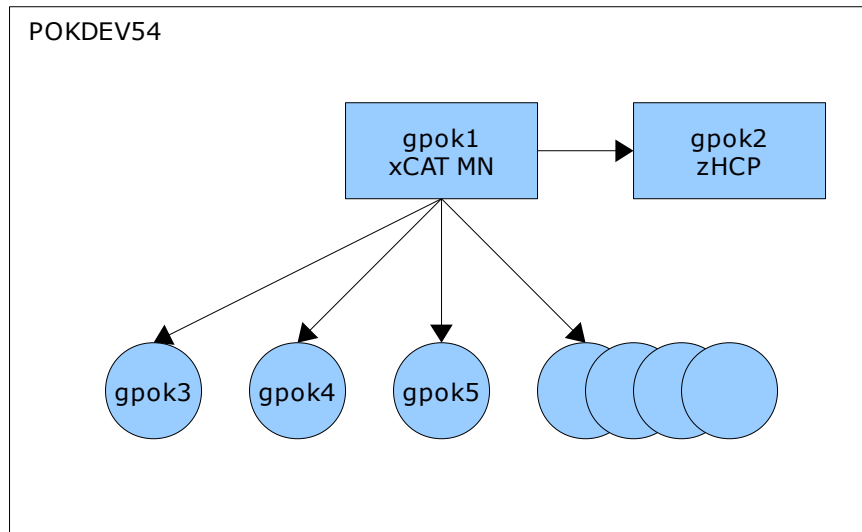

Figure 1. Sample environment setup.

Before you can install xCAT, there are a couple of prerequisites. You need to have <u>two</u> virtual servers (one server for the xCAT MN and the other for the zHCP) running zLinux.

The <u>xCAT MN</u> (gpok1) should have the following user entry:

```
USER LINUX253 PWD 512M 1G G
CPU 00 BASE
CPU 01
IPL CMS
MACHINE ESA 7
CONSOLE 0009 3215 T
NICDEF 0600 TYPE QDIO LAN SYSTEM GLAN3
SPOOL 000C 2540 READER *
SPOOL 000D 2540 PUNCH A
SPOOL 000E 1403 A
LINK MAINT 0190 0190 RR
LINK MAINT 019D 019D RR
LINK MAINT 019E 019E RR
LINK LNXMAINT 0191 0191 RR
LINK TCPMAINT 0592 0592 RR
MDISK 0100 3390 0001 10016 DM61A7 MR
MDISK 0101 3390 0001 10016 DM61A6 MR
```

MDISK 0100 is mounted with the root partition (/), and MDISK 0101 is mounted with /install (this is the directory that will contain the Linux ISOs). In our environment, two 3390 MOD-3 were used, but you can give the xCAT MN whatever disks you desire. It is recommended that you use LVM for /install, so the disk space can grow as the number of Linux ISOs grows. NICDEF 0600 is a Layer 2 GuestLAN. Virtual servers on POKDEV54 must use this GuestLAN. It is through this GuestLAN that the xCAT MN communicates with each virtual server. xCAT supports Layer 2 GuestLAN and Layer 2 VSWITCH.

The xCAT MN can run on any Linux distribution, SLES or RHEL. In the sample environment above, the xCAT MN was setup on SLES 10 SP2 with Server Base, Gnome, and X Windows packages installed.

The zHCP (gpok2) should have the following user entry:

```
USER LINUX2 PWD 512M 1G BCDG
CPU 00 BASE
CPU 01
OPTION LNKNOPAS
IPL CMS
MACHINE ESA 7
CONSOLE 0009 3215 T
NICDEF 0600 TYPE QDIO LAN SYSTEM GLAN3
SPOOL 000C 2540 READER *
SPOOL 000D 2540 PUNCH A
SPOOL 000E 1403 A
LINK MAINT 0190 0190 RR
LINK MAINT 019D 019D RR
LINK MAINT 019E 019E RR
LINK LNXMAINT 0191 0191 RR
LINK TCPMAINT 0592 0592 RR
MDISK 0100 3390 0001 3338 DM615F MR
```

MDISK 0100 is mounted with the root partition (/). In our environment, one 3390 MOD-3 was used, but you can give the zHCP whatever disks you desire. NICDEF 0600 is a Layer 2 GuestLAN. The zHCP has B, C, D, and G privileges. The zHCP needs class B privilege so it could use the FLASHCOPY command (if permitted), class C privilege so it could use the SEND command, and class D privilege so it could use the PURGE command. In order for the zHCP to have these privileges, you must log onto MAINT after the user has been created and issue:

```
DIRM FORUSER LINUX2 CLASS BCDG
```

The zHCP can run on any Linux distribution, SLES or RHEL. In the sample environment above, the zHCP was setup on SLES 10 SP2 with Server Base package installed.

Important

- The zHCP must use the same network managed by the DHCP server running on the xCAT MN. For example, if the DHCP server manages VSwitch VSW1, then the zHCP must use VSwitch VSW1. Also, the zHCP should only be on one network, the network managed by DHCP.

- Both the xCAT MN and zHCP are linked to LNXMAINT, which contains files used by both CMS and Linux.

  LNXMAINT has the following user entry.

```
USER LNXMAINT LNXMAINT 64M 128M BEG 1
INCLUDE TCPCMSU 2
LINK TCPMAINT 592 592 RR 3
MDISK 0191 3390 0001 0020 MMD857 MR READ WRITE MULTIPLE
MDISK 0192 3390 0021 0300 MMD857 MR ALL WRITE MULTIPLE
```

  It is recommended that you have the following PROFILE EXEC on LNXMAINT 191 disk (which is linked to all virtual servers on POKDEV54).

```
/* PROFILE EXEC for Linux virtual servers */
'CP SET RUN ON'
'CP SET PF11 RETRIEVE FORWARD'
'CP SET PF12 RETRIEVE'
'ACC 592 C'
'SWAPGEN 300 524288' /* create a 256M VDISK disk swap space */
'SWAPGEN 301 1048576' /* create a 512M VDISK disk swap space */
'PIPE CP QUERY' userid() '| var user'
parse value user with id . dsc .
```

```
if (dsc = 'DSC') then /* User is disconnected */
    'CP IPL 100'
else /* User is interactive -> prompt */
do
    say 'Do you want to IPL Linux from minidisk 100? y/n'
    parse upper pull answer .
    if (answer = 'Y') then 'CP IPL 100'
end /* else */
```

This statement in the PROFILE EXEC enables each virtual server to IPL 100 upon startup.

For more information on how to setup the LNXMAINT user, please refer to the *z/VM and Linux on IBM System z The Virtualization Cookbook for SLES 10 SP2* (SG24-7493).

Installation of xCAT

The purpose of this section is to provide details on the installation of the xCAT management node.

1. Install the DHCP and DNS server through Yast (if not already)
   ```
   yast -i dhcp-server
   yast -i yast2-dns-server
   ```
2. Use Yast to install the following packages (accept the dependencies)
   ```
   perl-IO-Socket-SSL
   perl-IO-Tty
   ```
3. Make an xcat directory under /root
   ```
   mkdir /root/xcat
   ```
4. Download the latest xCAT tarballs, xcat-core-2.3.tar.bz2 and xcat-dep-2.3.tar.bz2, from http://xcat.sourceforge.net/#download
5. Transfer the xCAT tarballs to /root/xcat
6. Extract the contents of each tarball
   ```
   tar jxf xcat-core-2.3.tar.bz2
   tar jxf xcat-dep-2.4.tar.bz2
   ```
7. Install additional perl packages
   ```
   rpm -i /root/xcat/xcat-dep/sles10/perl-XML-Simple-2.18-1.noarch.rpm
   rpm -i /root/xcat/xcat-dep/perl-version-0.76-1.noarch.rpm
   rpm -i /root/xcat/xcat-dep/perl-SOAP-Lite-0.710.08-1.noarch.rpm
   ```
8. Add the xCAT service to zypper
   a) If you are on SLES 10:
   ```
   zypper sa file:///root/xcat/xcat-dep/sles10/s390x xCAT-dep
   zypper sa file:///root/xcat/xcat-core xcat-core
   ```
   If you are on SLES 11:
   ```
   zypper ar file:///root/xcat/xcat-dep/sles10/s390x xCAT-dep
   zypper ar file:///root/xcat/xcat-core xcat-core
   ```
   Ignore the warning message about the keys and accept them.
   b) Use `zypper` to install xCAT
   ```
   zypper install xCAT
   ```
9. Add the xCAT commands to path
   ```
   source /etc/profile.d/xcat.sh
   ```
10. Check if the database is initialize
    ```
    tabdump site
    ```
    The output should look similar to the following:
    ```
    gpok253:~/xcat # tabdump site
    #key,value,comments,disable
    "blademaxp","64",,
    "domain","endicott.ibm.com",,
    "fsptimeout","0",,
    "installdir","/install",,
    "ipmimaxp","64",,
    "ipmiretries","3",,
    "ipmitimeout","2",,
    "consoleondemand","no",,
    "master","9.60.18.253",,
    ```

```
"maxssh","8",,
"ppcmaxp","64",,
"ppcretry","3",,
"ppctimeout","0",,
"rsh","/usr/bin/ssh",,
"rcp","/usr/bin/scp",,
"sharedtftp","0",,
"SNsyncfiledir","/var/xcat/syncfiles",,
"tftpdir","/tftpboot",,
"xcatdport","3001",,
"xcatiport","3002",,
"xcatconfdir","/etc/xcat",,
"timezone","US/Eastern",,
"nameservers","10.0.0.1",,
```

11. Setup an FTP server on the xCAT MN to contain Linux images
    a) Download the desire Linux distribution ISO to /root
    b) Go into /root directory
        ```
        cd /root
        ```
    c) Extract the ISO into the xCAT install tree /install
       For SLES:
        ```
        copycds SLES-10-SP2-s390x-GMC-DVD1.iso
        ```
       For RHEL:
        ```
        copycds -n rhel5.3 -a s390x RHEL5.3-Server-20090106.0-s390x-DVD.iso
        ```
    d) Remove the ISO from /root since we do not need the ISO any longer
        ```
        rm SLES-10-SP2-s390x-GMC-DVD1.iso
        ```
    e) Add the new OS image into the database
       For Red Hat:
        ```
        chtab imagename=rhel5.3 osimage.profile=compute.rhel5.s390x.tmpl
        osimage.imagetype=rhel5.3 osimage.osname=linux
        ```
       For SUSE:
        ```
        chtab imagename=sles10.2 osimage.profile=compute.sles10.s390x.tmpl
        osimage.imagetype=sles10.2 osimage.osname=linux
        ```

Installation of xCAT UI

The purpose of this section is to provide details on the installation of the xCAT UI.

1. Install php5-openssl, apache2 and apache2-mod_php5 through Yast. Once you select these packages, additional packages will be selected for you automatically. Accept these additional ones.
2. Upgrade to the latest xcat-dep and xcat-core (if not already)
   a) Download the latest xCAT tarballs, xcat-core.tar.bz2 and xcat-dep.tar.bz2, into /root/xcat. These tarballs can be obtained from http://xcat.sourceforge.net/#download
   b) Extract the contents of each tarball
      ```
      tar jxf xcat-core-2.3.tar.bz2
      tar jxf xcat-dep-2.4.tar.bz2
      ```
3. Install perl-version and perl-SOAP-Lite, which are located in /root/xcat/xcat-dep directory (if not already)
   ```
   rpm -i perl-version-0.76-1.noarch.rpm
   rpm -i perl-SOAP-Lite-0.710.08-1.noarch.rpm
   ```
4. Update xCAT (if necessary)
   ```
   zypper refresh
   zypper update -t package '*xCAT*'
   ```
5. Install the xCAT-UI RPM located in /root/xcat/xcat-core directory
   ```
   rpm -i xcat-ui-2.4.noarch.rpm
   ```

SSL Configuration

The purpose of this section is to provide details on configuring SSL on the xCAT server. The following instructions were taken from http://en.opensuse.org/Apache_Howto_SSL.

1. Make sure that apache starts with mod_ssl loaded.
   ```
   a2enmod ssl
   ```
2. Make sure that the SSL configuration is active.
   ```
   a2enflag SSL
   ```
3. The following steps will create self signed keys in a very simple way.
   ```
   /usr/bin/gensslcert
   ```
4. Copy /etc/apache2/vhosts.d/vhost-ssl.template to /etc/apache2/vhosts.d/vhost-ssl.conf
5. For the enabled modules, server flags, generated keys and vhosts to take effect, start or restart the apache service
   ```
   /etc/init.d/apache2 restart
   ```

Installation of zHCP

The purpose of this section is to provide details on the installation of the zHCP.

Before the zHCP can be installed on a virtual server running Linux, the following must be done (if not already).

1. Install and configure SMAPI for each z/VM partition. Refer to the <u>z/VM V5R4.0 Systems Management Application Programming</u>.
2. Install and configure DirMaint for each z/VM partition. Refer to the <u>z/VM V5R2.0 Directory Maintenance Facility Tailoring and Administration Guide</u>.
3. Create a disk pool on z/VM
   ```
   TBD
   ```
4. Grant the zHCP access to DirMaint. Logon MAINT and issue the following commands
   ```
   DIRM FOR ALL AUTHFOR LINUX2 CMDL 140A CMDS ADGHOPS
   DIRM FOR ALL AUTHFOR LINUX2 CMDL 150A CMDS ADGHOPS
   ```
   Change VSMWORK1 AUTHLIST
   ```
   SET FILEPOOL VMSYS
   QUERY FILEPOOL CONNECT
   ACCESS VMSYS:VSMWORK1.DATA A (FORCERW
   ACCESS VMSYS:VSMWORK1. B (FORCERW
   X VSMWORK1 AUTHLIST B
   ```
   Append the z/VM userID you wish to have DIRMAINT access to the end of VSMWORK1 AUTHLIST
   ```
   LINUX2          ALL             ALL
   ```
5. Restart SMAPI
   ```
   FORCE VSMWORK1
   XAUTOLOG VSMWORK1
   ```
6. Give the virtual server where you will install the zHCP B, C, D, and G privileges. The zHCP needs class B privilege so it could use the FLASHCOPY command (if permitted), class C privilege so it could use the SEND command, and class D privilege so it could use the PURGE command. In order for the zHCP to have these privileges, you must log onto <u>MAINT</u> after the user has been created and issue:
   ```
   DIRM FORUSER LINUX2 CLASS BCDG
   ```

If the conditions above are met, do the following on the zHCP.

1. Install `gcc` and `gcc-c++` (if not already)
2. Make a zhcp directory under /root
   ```
   mkdir /root/zhcp
   ```
3. Download the zHCP RPM (zhcp-1-1.s390x.rpm)
4. Install the RPM
   ```
   rpm -i zhcp-1-1-s390x.rpm
   ```
5. Append the zHCP library path to /etc/ld.so.conf
   ```
   echo /opt/zhcp/lib >> /etc/ld.so.conf
   ```
6. Configure dynamic linker run time bindings
   ```
   ldconfig
   ```

The zHCP was tested on SLES 10.2/10.3. Other releases of SLES and RHEL should be supported.

Getting Started

The purpose of this section is to provide a starting guide on how to manage z/VM and z/Linux with xCAT.

1. Make sure the Linux VMCP module is loaded on the xCAT MN
    ```
    modprobe vmcp
    ```
2. Configure the DHCP server
    a. Add networks into the DHCP configuration
        ```
        makedhcp -n
        ```
    b. Restart DHCP
        ```
        service dhcpd restart
        ```
    c. Add the relevant networks to DHCP
        ```
        makedhcp -a
        ```
3. Configure the DNS server
    a. Setup DNS from entries in /etc/hosts
        ```
        makedns
        ```
    b. Setup /etc/resolve.conf
        ```
        domain endicott.ibm.com
        nameserver 10.0.0.1
        ```
    c. Start DNS
        ```
        service named start
        ```
    d. Start DNS on boot
        ```
        chkconfig --level 345 named on
        ```
4. Set up the <u>passwd</u> table. This table will contain the default password for new nodes installed through autoyast/kickstart.
    ```
    chtab key=system passwd.username=root passwd.password=rootpw
    ```
    You can change this password on the new node once it has been installed using
    ```
    passwd
    ```
5. Set up the <u>hosts</u> table (it will be used to setup /etc/hosts). You need to determine the regular expression that represents the nodes that xCAT will manage.
    ```
    chtab node=all hosts.ip="|gpok(\d+)|10.0.0.(\$1+0)|" hosts.hostnames="|(.*)|(\
    $1).endicott.ibm.com|"
    ```

    *   Here, nodes belonging to <u>group=all</u> will have hostnames of gpok1, node2, etc. and IP addresses of 10.0.0.1, 10.0.0.2, etc. in /etc/hosts.
    *   Each time a new node is added to xCAT, you will need to run `makehosts`.
    *   You will need to setup the hosts table for each group you create.
6. Setup the <u>networks</u> table. You need to determine the dynamic range that the DHCP server will manage. You also need to point the DHCP, DNS, and FTP server to the IP address of your xCAT MN.
    ```
    chtab net=10.0.0.0 networks.mask=255.255.255.0 networks.mgtifname=eth1
    networks.gateway=10.0.0.1 networks.dhcpserver=10.0.0.1
    networks.tftpserver=10.0.0.1 networks.nameservers=10.0.0.1
    networks.dynamicrange=10.0.0.3-10.0.0.100
    ```
7. Define the DHCP interfaces in the <u>site</u> table to limit which NIC DHCP will listen on.
    ```
    chtab key=dhcpinterfaces site.value='all|eth1'
    ```
8. Start by adding the zHCP node (gpok2). Use the DNS <u>hostname</u> of that node when adding.
    ```
    mkdef -t node -o gpok2 userid=LINUX2 hcp=gpok2.endicott.ibm.com nodetype=osi
    mgt=zvm groups=all
    ```

- `userid=LINUX2` is the z/VM userID of zLinux running the zHCP
- `hcp=gpok2.endicott.ibm.com` is the zHCP that will be managing those nodes
- `mgt=zvm` is the plugin to use when actions are performed against those nodes
- `groups=all` is the group name that the nodes belong to

9. Add nodes you want to manage. Use the DNS hostname of that node when adding.

```
mkdef -t node -o gpok3 userid=LINUX3 hcp=gpok2.endicott.ibm.com
nodetype=osi mgt=zvm groups=all
```

10. Update /etc/hosts

```
makehosts
```

11. Update DNS

```
makedns
```

12. Push the public SSH key of the xCAT management node onto a <u>group</u> of nodes that you want to manage

```
xdsh all -K
```

- This pushes the SSH key onto nodes within the group=all
- <u>Warning</u>: `xdsh` will hang if one or more nodes are powered down

You can push the public SSH key onto a <u>single</u> node

```
xdsh gpok3 -K
```

- This will push the public SSH key onto gpok3
- `xdsh` will hang if gpok3 is powered down

The `xdsh` command will prompt you for a root password. It is the root password for the node or group of nodes you are trying to push the public SSH key to. It is <u>recommended</u> that you put nodes with the same root password into the same group. More importantly, the `xdsh` command will only work for nodes that are online.

13. Modify the table definitions of existing node.

```
rscan gpok254 >/tmp/stanza.txt
cat /tmp/stanza.txt | mkdef -z
```

- This collects basic information about each on managed by the zHCP (gpok254) and outputs it to a file. The file is then used to modify the table definitions.
- The most important part of this command is that it gets the z/VM userID of each node. Actions performed by the zHCP are performed against the z/VM userID.

14. Start using supported xCAT commands

xCAT Commands

The purpose of this section is to list the current xCAT on z/VM commands.

`rpower` - It boots, resets powers on and off, and queries node hardware and devices. The syntax is: rpower <node> [on|off|stat|reset]

        rpower gpok3 stat

`mkvm` – Creates a new virtual machine with the same profile/resources as the specified node (Clone). It can also create a new virtual machine based on a user directory entry. The syntax is: mkvm <new node> /tmp/ <directory entry text file>.

        mkvm gpok3 /tmp/dirEntry.txt

For clonning, the syntax is: `mkvm <target Linux> <source Linux> pool=<disk pool> pw=<multi password>`

        mkvm gpok4,gpok5 gpok3 pool=POOL1 pw=PWD

`rmvm` – Removes a virtual machine. The syntax is: `rmvm <node>`.

        rmvm gpok3

`lsvm` – List virtual machine's configuration. Options supported are:
   • List virtual machine directory entry
       The syntax is: `lsvm <node>`
             lsvm gpok3
   • List virtual machine's defined network names
       The syntax is: `lsvm <node> --getnetworknames`
             lsvm gpok3 --getnetworknames
   • List virtual machine's network configuration
       The syntax is: `lsvm <node> --getnetwork [networkname]`
             lsvm gpok3 --getnetwork GLAN1
   • List virtual machine's disk pool names
       The syntax is: `lsvm <node> --diskpoolnames`
             lsvm gpok3 --diskpoolnames
   • List virtual machine's disk pool
       The syntax is: `lsvm <node> --diskpool [pool name] [space (free or used)]`
             lsvm gpok3 --diskpool POOL1 free

`chvm` – Changes the virtual machine profile. Options supported are:
   • Adds a 3390 (ECKD) disk to a virtual server's directory entry
       The syntax is: `chvm <node> --add3390 [disk pool] [device address] [cylinders] [mode]` `[read password] [write password] [multi password]`
             chvm gpok3 --add3390 POOL1 0101 3338 MR
   • Adds a 3390 (ECKD) disk that is defined in a virtual server's directory entry to that virtual server's active configuration
       The syntax is: `chvm <node> --add3390active [device address] [mode]`
             chvm gpok3 --add3390active 0101 MR
   • Adds a 9336 (FBA) disk to a virtual server's directory entry
       The syntax is: `chvm <node> --add9336 [disk pool] [virtual device] [block size] [mode]`

```
[blocks] [read password] [write password] [multi password]
      chvm gpok3 --add9336 POOL3 0101 4096 3338 MR
```

- Adds a network adapter to a virtual server's directory entry

  The syntax is: `chvm <node> --addnic [address] [type] [device count]`
  ```
        chvm gpok3 --addnic 0600 QDIO 3
  ```
- Adds a virtual processor to a virtual server's directory entry

  The syntax is: `chvm <node> --addprocessor [address]`
  ```
        chvm gpok3 --addprocessor 01
  ```
- Adds a virtual processor to a virtual server's active configuration

  The syntax is: `chvm <node> --addprocessoractive [address] [type]`
  ```
        chvm gpok3 --addprocessor 01 ZAAP
  ```
- Adds a v-disk to a virtual server's directory entry

  The syntax is: `chvm <node> --addvdisk [userID] [device address] [size]`
  ```
        chvm gpok3 --addvdisk 0300 2097120
  ```
- Connects a network adapter to a GuestLAN

  The syntax is: `chvm <node> --connectnic2guestlan [address] [lan] [owner]`
  ```
        chvm gpok3 --connectnic2guestlan 0600 GLAN1 LN1OWNR
  ```
- Connects a network adapter to a VSwitch

  The syntax is: `chvm <node> --connectnic2vswitch [address] [vswitch]`
  ```
        chvm gpok3 --connectnic2vswitch 0600 VSW1
  ```
- Copy a disk attached to a given virtual server

  The syntax is: `chvm <node> --copydisk [target address] [source node] [source address]`
  ```
        chvm gpok3 --copydisk 0100 gpok2 0101
  ```
- Adds a dedicated device to a virtual server's directory entry

  The syntax is: `chvm <node> --dedicatedevice [virtual device] [real device] [mode]`
  ```
        chvm gpok3 --dedicatedevice 0101 637F RW
  ```
- Deletes the IPL statement for the z/VM guest with the specified name

  The syntax is: `chvm <node> --deleteipl`
  ```
        chvm gpok3 --deleteipl
  ```
- Formats a disk attatch to a specified node (only ECKD disks supported)

  The syntax is: `chvm <node> --formatdisk [disk address] [multi password]`
  ```
        chvm gpok3 --formatdisk 0100 PWD
  ```
- Disconnects a network adapter

  The syntax is: `chvm <node> --disconnectnic [address]`
  ```
        chvm gpok3 --disconnectnic 0600
  ```
- Grant VSwitch access for specified node

  The syntax is: `chvm <node> --grantvswitch [VSwitch]`
  ```
        chvm gpok3 --grantvswitch VSW1
  ```
- Removes a minidisk from a virtual server's directory entry

  The syntax is: `chvm <node> --removedisk [virtual device]`
  ```
        chvm gpok3 --removedisk 0101
  ```
- Removes a network adapter from a virtual server's directory entry

  The syntax is: `chvm <node> --removenic [address]`
  ```
        chvm gpok3 --removenic 0700
  ```
- Removes a processor from an active virtual server's configuration

  The syntax is: `chvm <node> --removeprocessor [address]`
  ```
        chvm gpok3 --removeprocessor 01
  ```

- Replaces a user directory entry

  The syntax is: `chvm <node> --replacevs [user directory entry]`

  ```
  chvm gpok3 --replacevs /tmp/dirEntry.txt
  ```
- Sets the IPL statement for the specified z/VM userID

  The syntax is: `chvm <node> --setipl [ipl target] [load parms] [parms]`

  ```
  chvm gpok3 --setipl CMS
  ```
- Sets the password for the specified z/VM userID

  The syntax is: `chvm <node> --setpassword [password]`

  ```
  chvm gpok3 --setpassword PW4U
  ```

`updatenode` – Updates nodes. The supported options are:

- Updates node operating system. Supported OS are RHEL and SLES. The OS images must already be copied into the FTP directory using copycds. The syntax is `updatenode <node> --release <updated version>`.

  ```
  updatenode gpok3 --release sles10.3
  ```

`rscan` – Collects node information from one or more hardware control points. The syntax is `rscan <hcp node>`.

```
rscan gpok2
```

`rinv` - Remote hardware inventory. The syntax is: `rinv <node>`.

```
rinv gpok3 all
```

Installing Linux Using AutoYast or Kickstart

The purpose of this section is to detail the installation of Linux using AutoYast or Kickstart.

Before you begin, make sure the following is done.
- The FTP server must be setup during the xCAT MN installation. The FTP root directory /install must contain the desired Linux distribution
- Nodes should be on the same network as the zHCP and xCAT MN

There are two ways to install Linux onto a z/VM virtual machine, depending on which Linux distribution you want. One is through <u>AutoYast</u>, which is used to install SUSE Linux Enterprise Server (SLES) releases. The other is through <u>Kickstart</u>, which is used to install Red Hat Enterprise Linux (RHEL) releases.

1. Create the node definition
   ```
   mkdef -t node -o gpok3 userid=LINUX3 nodetype=osi hcp=gpok2.endicott.ibm.com
   mgt=zvm groups=all
   ```
2. Update /etc/hosts
   ```
   makehosts
   ```
3. Update DNS
   ```
   makedns
   ```
4. Create the new virtual machine using the desired user entry. For example,
   ```
   USER LINUX3 PWD 512M 1G G
   CPU 00 BASE
   CPU 01
   IPL CMS
   MACHINE ESA 7
   CONSOLE 0009 3215 T
   NICDEF 0600 TYPE QDIO LAN SYSTEM GLAN3
   SPOOL 000C 2540 READER *
   SPOOL 000D 2540 PUNCH A
   SPOOL 000E 1403 A
   LINK MAINT 0190 0190 RR
   LINK MAINT 019D 019D RR
   LINK MAINT 019E 019E RR
   LINK LNXMAINT 0194 0191 RR
   LINK TCPMAINT 0592 0592 RR
   ```
   This is accomplished by copying the user entry above into a text file (userEntry.txt), and issuing
   ```
   mkvm gpok3 /tmp/userEntry.txt
   ```
   The user entry text file should <u>not</u> contain any extra lines. A MAC address will be assigned to the userID upon creation. The first NICDEF within the user entry will be used to boot the Linux kernel. Make sure that this virtual server is on the same network as the zHCP.
5. Add disks to the new node (the default autoyast/kickstart template available in xCAT requires 2 3390-MOD3 disks, you can edit this template under /opt/xcat/share/xcat/install/<distribution>). For more information on how to customize the template, see Appendix B.
   ```
   chvm gpok3 --add3390 POOL1 0100 3338 MR
   chvm gpok3 --add3390 POOL1 0101 3338 MR
   ```
6. Format each disk (not necessary, but recommended)
   ```
   chvm gpok3 --formatdisk 0100
   chvm gpok3 --formatdisk 0101
   ```

7. Set up the <u>noderes</u> table. You need to determine the OS image (Linux distribution) for the node. When xCAT was setup, the OS image was added into the database. You can find available OS images by issuing

```
tabdump osimage
```

For SUSE:
```
chtab node=gpok3 noderes.netboot=zvm noderes.current_osimage=sles10.2
```
For Red Hat:
```
chtab node=gpok3 noderes.netboot=zvm noderes.current_osimage=rhel5.3
```

8. Verify the definition
```
lsdef gpok3

Object name: gpok3
    groups=all
    hcp=gpok2.endicott.ibm.com
    hostnames=gpok3.endicott.ibm.com
    ip=10.0.0.3
    mac=02:00:01:FF:FF:FF
    mgt=zvm
    netboot=zvm
    nodetype=osi
    postscripts=syslog,remoteshell,otherpkgs,syncfiles
    userid=LINUX3
```

9. Add the new node to DHCP
```
makedhcp -a
```

10. Prepare the new node for installation
```
nodeset gpok3 install
```

11. Boot the new node from reader
```
rnetboot gpok3 ipl=000C
```

12. In Gnome or KDE, open the VNC viewer to see the installation progress. It might take a couple of minutes before you can connect.
```
vncviewer gpok3.endicott.ibm.com:1
```
The default VNC password is `123456`.

13. (Only for SLES 10 SP2 or older) Once the <u>first</u> phase of installation is complete, restart the virtual server to complete the <u>final</u> phase of installation
```
rpower gpok3 reset
```

14. The default password for the node can be found in the <u>passwd</u> table. See *Getting Started* section step 10.

The default template available on xCAT was tested on SLES 10.2/10.3 and RHEL 5.3/5.4. Other releases of SLES and RHEL should be supported.

Cloning Virtual Servers

The purpose of this section is to show how to clone a virtual server running Linux.

1. The source node must be online. If it is not online, bring it online.
   ```
   rpower gpok3 on
   ```
2. Exchange SSH keys between the xCAT server and the source node to be cloned, if not already.
   ```
   xdsh gpok3 -K
   ```
3. Create the table definitions for new nodes (gpok4 and gpok5)
   ```
   mkdef -t node -o gpok4 userid=LINUX4 nodetype=osi hcp=gpok2.endicott.ibm.com mgt=zvm
   groups=all


   mkdef -t node -o gpok5 userid=LINUX5 nodetype=osi hcp=gpok2.endicott.ibm.com mgt=zvm
   groups=all
   ```

   The `nodetype=osi` option enables this node to be monitored through Ganglia. You can leave this option out.
4. Update /etc/hosts
   ```
   makehosts
   ```
5. Update DNS
   ```
   makedns
   ```
6. Add the new node to DHCP
   ```
   makedhcp -a
   ```
7. In order to clone a virtual server running Linux, the partition must be mounted by path.

   In SUSE, the root directory under /etc/fstab, which contains information on the system partitions and disks, is similar to this:
   ```
   /dev/disk/by-path/ccw-0.0.0100-part1  /  ext3  acl,user_xattr  1 1

   /dev/disk/by-path/ccw-0.0.0101-part1  /opt  ext3  acl,user_xattr  1 2
   ```
   The parameters under `/etc/zipl.conf`, which specifies which disks to bring online when the system is IPLed, is similar to this:
   ```
   parameters = "root=/dev/disk/by-path/ccw-0.0.0100-part1 TERM=dumb"
   ```
   If you happen to edit `zipl.conf`, you <u>must</u> run `zipl` after you made the changes so that changes are written to the boot record.

   Moreover, to clone a virtual server, the source virtual server running Linux must be <u>online</u> at the time of cloning. This allows xCAT to gather information about the source virtual server.
8. Clone virtual server(s) running Linux:
   ```
   mkvm gpok4,gpok5 gpok3 pool=POOL1
   ```
   This will create two virtual servers (gpok4 and gpok5) identical to gpok3. It will use disks in disk pool `POOL1`.

   This command takes the following steps, which were taken from the Virtualization Cookbook (it can be downloaded from http://linuxvm.org/present/misc/virt-cookbook-S10.pdf):
   a) Link the target disk read-write.
   b) Copy the source to the target disk via FLASHCOPY or the Linux dd command based on which is supported.
   c) Mount the newly copied root file system.
   d) Modify the networking information on the target system.
   e) Detach the target disk.

f) IPL the target system.

Setting Up Ganglia on xCAT

The purpose of this section is to provide details on the set up of Ganglia on xCAT.

1. Download the Ganglia RPMs: libconfuse-2.6-1.s390x.rpm, libganglia-3.1.1-1.s390x.rpm, ganglia-gmetad-3.1.1-1.s390x.rpm, ganglia-gmond-3.1.1-1.s390x.rpm, and ganglia-web-3.1.1-1.s390x.rpm.
2. Set up ganglia on the xCAT MN (gpok1)
   a) Install libapr1, pkgconfig, php5-pear, php5-gd, apache2, apache2-mod_php5 through Yast (if not already)
   b) Install the Ganglia library RPMs
   ```
   rpm -i libconfuse-2.6-1.s390x.rpm
   rpm -i libganglia-3.1.1-1.s390x.rpm
   ```
   c) Install gmetad (This monitors other nodes by periodically polling them)
   ```
   yast -i rrdtool
   rpm -i ganglia-gmetad-3.1.1-1.s390x.rpm
   ```
   d) Install gmond
   ```
   rpm -i ganglia-gmond-3.1.1-1.s390x.rpm
   ```
   e) Install the ganglia web RPM
   ```
   rpm -i ganglia-web-3.1.1-1.s390x.rpm
   ```
   f) Configure the apache server
   - Specify the NameVirtualHost in /etc/apache2/listen.conf
     ```
     NameVirtualHost 9.60.18.253:80
     ```
   - Copy /etc/apache2/vhosts.d/vhost.template
     ```
     cd /etc/apache2/vhosts.d
     cp vhost.template ganglia.conf
     ```
   - Edit ganglia.conf
     ```
     <VirtualHost 9.60.18.253:80>
             ServerName gpok253.endicott.ibm.com
             DocumentRoot /srv/www/htdocs/ganglia
             <Directory "/srv/www/htdocs/ganglia">
                     Order allow,deny
                             Allow from all
             </Directory>
     </VirtualHost>
     ```
   g) Restart the apache server
   ```
   /etc/init.d/apache2 restart
   ```
   h) Restart gmond and gmetad
   ```
   service gmond restart
   service gmetad restart
   ```
3. Create the directory /opt/xcat/install on the xCAT MN
   ```
   mkdir /opt/xcat/install
   ```
4. Download the following packages and place them in /opt/xcat/install
   - `libganglia-3.1.1-1.s390x.rpm`
   - `libconfuse-devel-2.6-1.s390x.rpm`
   - `ganglia-gmond-3.1.1-1.s390x.rpm`

Ganglia Monitoring on xCAT

The purpose of this section is to provide details on using Ganglia on xCAT.

1. Transfer ganglia RPMs required to run gmond over to nodes you want to monitor

   ```
   xdcp <node> /opt/xcat/install/ganglia-gmond-3.1.1-1.s390x.rpm
   xdcp <node> /opt/xcat/install/libconfuse-2.6-1.s390x.rpm
   xdcp <node> /opt/xcat/install/libganglia-3.1.1-1.s390x.rpm
   ```

   • This command transfers the files into /root directory on the target node

2. Install the RPMs

   ```
   xdsh <node> rpm -i libconfuse-2.6-1.s390x.rpm
   xdsh <node> rpm -i libganglia-3.1.1-1.s390x.rpm
   xdsh <node> rpm -i ganglia-gmond-3.1.1-1.s390x.rpm
   ```

   • Make sure the target node has libapr1 package

3. Ensure the nodetype of all nodes you wish to monitor have the type of '*osi*'. This can be done by editing the nodetype table.

   ```
   tabedit nodetype
   ```

4. Add gangliamon to the monitoring table

   ```
   monadd gangliamon
   ```

5. Configure the node

   ```
   moncfg gangliamon -r
   ```

   • This runs the ganglia configuration script on all the nodes

6. Start gangliamon:

   ```
   monstart gangliamon -r
   ```

   • This will start the gmond daemon on all the nodes
   • The -r flag is required to ensure the gmond daemon is started on each node

   You may also specify a particular node to start:

   ```
   monstart gangliamon gpok246 -r
   ```

7. Stopping gangliamon:

   ```
   monstop gangliamon -r
   ```

Update Linux

The purpose of this section is to detail how to update a Linux operating system.

1. Extract the Linux distribution ISO into the xCAT install directory (if not already)
   For <u>SuSE Linux Enterprise Server</u> (SLES):
   ```
   copycds SLES-10-SP3-DVD-s390x-DVD1.iso
   ```
   For <u>Red Hat Enterprise Server</u> (RHEL):
   ```
   copycds -n rhel5.4 -a s390x RHEL5.4-Server-20090819.0-s390x-DVD.iso
   ```
2. Update the desired node (TBD)
   ```
   updatenode gpok4 --release sles10.3
   ```

   The example above updates the operating system on gpok4 from SLES 10.2 to SLES 10.3. The command will take a while to complete (around 10 minutes).

Limitations

The purpose of this section is to highlight the limitations of xCAT on z/VM and z/Linux.

1. The zHCP only support RHEL 5.3/SLES 10 SP2 or greater.
2. Nodes that the zHCP manages must have the VMCP Linux module.
3. SCSI/FCP disks are not yet supported in cloning. Only ECKD disks are supported.
4. Profiles within a user entry are not supported for cloning and Linux installation.
5. The zHCP must be on a Layer 2 network, either Guest LAN or VSwitch. Each node can have more than one NIC, but the nodes that are managed by this zHCP must also be on the same network.

Appendix A: Setting Up a Private Network

The purpose of this section is to detail how to setup a private network.

SSH to the desire Linux where you want to setup the private network, in this case gpok253. Two configuration files must be added under /etc/sysconfig/ to let the system know about the new interface, one for hardware and one for network settings.

Copy the hardware settings from the existing network /etc/sysconfig/hardware/hwcfg-qeth-bus-ccw-0.0.0600. Both interfaces will use the qdio/qeth drivers, therefore, the configuration files can be identical except for the virtual addresses. The existing file is copied to specify the new NIC. The only difference needed is to change the 060X values to 080X.

```
cd /etc/sysconfig/hardware/
sed *600 -e 's/060/080/g' > hwcfg-qeth-bus-ccw-0.0.0800
```

It should look like the following.

```
cat hwcfg-qeth-bus-ccw-0.0.0800
```

```
STARTMODE="auto"
MODULE="qeth"
MODULE_OPTIONS=""
MODULE_UNLOAD="yes"
SCRIPTUP="hwup-ccw"
SCRIPTUP_ccw="hwup-ccw"
SCRIPTUP_ccwgroup="hwup-qeth"
SCRIPTDOWN="hwdown-ccw"
CCW_CHAN_IDS="0.0.0800 0.0.0801 0.0.0802"
CCW_CHAN_NUM="3"
CCW_CHAN_MODE="OSAPORT"
QETH_LAYER2_SUPPORT="1"
```

Copy the network settings from the existing network /etc/sysconfig/network/ifcfg-qeth-bus-ccw-0.0.0600.

```
cd /etc/sysconfig/network
cp ifcfg-qeth-bus-ccw-0.0.0600 ifcfg-qeth-bus-ccw-0.0.0800
```

Edit the network settings.

```
vi ifcfg-qeth-bus-ccw-0.0.0800
```

```
BOOTPROTO="static"
UNIQUE=""
STARTMODE="onboot"
IPADDR="10.0.0.1"
NETMASK="255.255.255.0"
NETWORK="10.0.0.0"
BROADCAST="10.0.0.255"
_nm_name='qeth-bus-ccw-0.0.0800'
```

Add the IP address of the virtual server you want to access into the /etc/hosts file.

```
vi /etc/hosts
```

```
#
```

```
# hosts          This file describes a number of hostname-to-address
#                mappings for the TCP/IP subsystem.  It is mostly
#                used at boot time, when no name servers are running.
#                On small systems, this file can be used instead of a
#                "named" name server.
# Syntax:
#
# IP-Address  Full-Qualified-Hostname  Short-Hostname
#

127.0.0.1       localhost

# special IPv6 addresses
::1             localhost ipv6-localhost ipv6-loopback

fe00::0         ipv6-localnet

ff00::0         ipv6-mcastprefix
ff02::1         ipv6-allnodes
ff02::2         ipv6-allrouters
ff02::3         ipv6-allhosts
9.60.18.253     gpok253.endicott.ibm.com gpok253
10.0.0.1        gpok1.endicott.ibm.com gpok1
```

Reboot the virtual server to have the changes take effect.

```
        Reboot
```

Appendix B: Customizing Autoyast and Kickstart

The purpose of this section is to detail how to customize the autoyast and kickstart templates.

Autoyast and kickstart allows you to customize a Linux system based on a template. While you would typically go through various panels to manually customize your Linux system during boot, you no longer have to with autoyast and kickstart. This allows you to configure a vanilla Linux system faster and more effectively.

The following applies to SuSE Linux Enterprise Server (SLES):

1. Base you customization on the default template in /opt/xcat/share/xcat/install/sles. This template is pre-configured to setup the network for you using DHCP.
2. Determine the number of disks your vanilla system will have and the mount points for each disk.
3. Copy the default template: /opt/xcat/share/xcat/install/sles/compute.sles10.s390x.tmpl
4. Put the new template in the same directory: /opt/xcat/share/xcat/install/sles
   Remember to add this template to the osimage table (if you want to use it):
   ```
   chtab imagename=sles10.2 osimage.profile=custom.sles10.s390x.tmpl
   osimage.imagetype=sles10.2 osimage.osname=linux
   ```
5. Add the disk to the template using the following format:
   ```
   <!-- Dasd attached at 0101 -->
   <listentry>
     <bus>None</bus>
     <bus_hwcfg>none</bus_hwcfg>
     <channel>0.0.0101</channel>
     <dev_name>/dev/dasdb</dev_name>
     <dev_names config:type="list">
       <listentry>/dev/dasdb</listentry>
       <listentry>/dev/disk/by-path/ccw-0.0.0101</listentry>
     </dev_names>
     <device>DASD</device>
     <driver>io_subchannel</driver>
     <drivers config:type="list">
       <listentry>
         <active config:type="boolean">true</active>
         <modprobe config:type="boolean">true</modprobe>
         <modules config:type="list">
           <module_entry config:type="list">
             <listentry>dasd_eckd_mod</listentry>
             <listentry></listentry>
           </module_entry>
         </modules>
       </listentry>
     </drivers>
     <formatted config:type="boolean">true</formatted>
     <partition_info>/dev/dasdb1 (Linux native)</partition_info>
     <resource>
       <disk_log_geo config:type="list">
         <listentry>
           <cylinders config:type="integer">3338</cylinders>
   ```

```
        <heads config:type="integer">15</heads>
        <sectors config:type="integer">12</sectors>
      </listentry>
    </disk_log_geo>
    <io config:type="list">
      <listentry>
        <active config:type="boolean">true</active>
        <length config:type="integer">1</length>
        <mode>rw</mode>
        <start config:type="integer">256</start>
      </listentry>
    </io>
  </resource>
  <sysfs_bus_id>0.0.0101</sysfs_bus_id>
  <udi>/org/freedesktop/Hal/devices/ccw_0_0_0101_storage</udi>
</listentry>
```

The variables highlighted in red need to be modified based on how you customize your Linux. In the example above, a disk attached at 0101 is added. The size of the disk is 3338 cylinders. The device name for this disk is dasdb. The drive name is determined by how many devices are attached to the Linux system. For example, the first device gets a name of dasda, the second device gets a name of dasdb, and so on.

Once complete, add it to the `<dasd>` section of the template.

6. Add a module entry for the new disk

```
<module_entry>
  <device>dasd-bus-ccw-0.0.0101</device>
  <module>dasd_eckd_mod</module>
  <options></options>
</module_entry>
```

The variables highlighted in red need to be modified based on how you customize your Linux.   In the example above, a disk attached at 0101 is added.

Once complete, add it to the `<modules>` section of the template.

7. Add the mount point for the new disk

```
<!-- /usr partition -->
<drive>
  <device>/dev/dasdb</device>
  <partitions config:type="list">
    <partition>
      <create config:type="boolean">true</create>
      <filesystem config:type="symbol">ext3</filesystem>
      <format config:type="boolean">true</format>
      <mount>/usr</mount>
      <mountby config:type="symbol">path</mountby>
      <partition_id config:type="integer">131</partition_id>
      <partition_nr config:type="integer">1</partition_nr>
      <partition_type>primary</partition_type>
      <size>max</size>
```

```
      </partition>
    </partitions>
    <use>all</use>
  </drive>
```

The variables highlighted in red need to be modifed based on how you customize your Linux.   In the example above, a disk attached at 0101 is added. The device name given to disk is dasdb. For the example, this disk will be mounted at /usr and will have a ext3 file system.

Once complete, add it to the `<partitioning>` section of the template.

8. Add the software you need to the `<software>` section of the template. You need to determine the package name you want installed on your Linux system, e.g. `<package>`perl-URI`</package>`.

The following applies to <u>Red Hat Enterprise Server</u> (RHEL):

1. Base you customization on the default template in /opt/xcat/share/xcat/install/rh. This template is preconfigured to setup the network for you using DHCP.
2. Determine the number of disks your vanilla system will have and the mount points for each disk.
3. Copy the default template: /opt/xcat/share/xcat/install/rh/compute.rhel5.s390x.tmpl
4. Put the new template in the same directory: /opt/xcat/share/xcat/install/rh

   Remember to add this template to the osimage table (if you want to use it):
   ```
   chtab imagename=rhel5.3 osimage.profile=custom.rhel5.s390x.tmpl
   osimage.imagetype=rhel5.3 osimage.osname=linux
   ```
5. Add the disk and mount point to the template using the following format:
   ```
   clearpart --initlabel –drives=dasda,dasdb,dasdc
   part / --fstype ext3 --size=100 --grow –ondisk=dasda
   part /usr --fstype ext3 --size=100 --grow –ondisk=dasdb
   part /home --fstype ext3 --size=100 --grow –ondisk=dasdc
   ```

   The variables highlighted in red need to be modified based on how you customize your Linux. In the example above, a disk is added with a device name of dasdc. The disk will be mounted at /home and will have a ext3 file system.
6. Add the software you need to the `%packages` section.