xCAT 2.0 on AIX

How-To: Deploy diskless AIX nodes

Version: 1.0

Date: 3/19/2008

1.0 Deploying AIX diskless nodes using xCAT

This is an example of how an AIX diskless node can be deployed using xCAT and AIX/NIM commands.

NIM enables a cluster administrator to centrally manage the installation and configuration of AIX and optional software on machines within a networked environment. Setting up NIM involves various tasks including, installing NIM software, configuring NIM and creating some basic NIM installation resources.

The specific tasks that you need to perform depend on which features of NIM that you plan to use. For more information about NIM, see the *IBM AIX Installation Guide and Reference*. (http://www-03.ibm.com/servers/aix/library/index.html)

This "How-To" describes only one basic set of steps that may be used to deploy diskless nodes and is not meant to be comprehensive.

1.1 Assumptions

- An AIX system has been installed to use as an xCAT management node.
- All relevant base AIX services are configured and running. This includes (but is not limited to) bootp, tftp, NFS, and hostname resolution.
- xCAT and prerequisite software has been installed on the management node.
- LPARs have already been created using the HMC interfaces.
- You have some experience using AIX Network Installation Management (NIM).

1.2 Configure NIM

To install the required NIM software and do the NIM configuration you can use the AIX **nim_master_setup** command. This command can also be used to create some basic NIM resources that can be used to do a standard ("diskfull") installation of the nodes. However, in our case we won't be needing those resources so we will run the command as follows:

nim_master_setup -a mk_resource=no -a filesystem=/install

Note: If you have already configured the NIM master on your xCAT management node you can skip this step.

1.3 Create a diskless image

In this case we need to create the basic diskless operating system (OS) image that can be used to boot the diskless nodes. Multiple nodes can be booted using the same diskless image.

To do this you can use the xCAT **mkdsklsimage** command. This command uses the AIX **mkcosi** command to create the diskless image and also makes some modifications to the image so that other xCAT features, such as **rnetboot**, will work correctly. See the **man** page for additional details.

When you run the command you must provide a source for the installable images. This is typically the AIX product media or the location of a NIM *lpp_source* resource. You must also provide a name for the image you wish to create.

For example, to create a diskless image called "*61cosi*" using the AIX product CDs you could issue the following command.

mkdsklsimage -s /dev/cd0 61cosi

(Note that this operation could take a while to complete!)

Creating an AIX diskless image is actually equivalent to creating a NIM *spot* (Shared Product Object Tree) resource. A *spot* provides a **/usr** file system for diskless nodes, as well as the network boot support. The *spot* resource is also referred to as a Common Operating System Image (COSI).

The **mkdsklsimage** command will create a NIM SPOT resource named *61cosi* that will be contained in a subdirectory of */install*. (This is the default location when using xCAT.)

To get details for the resource definition use the AIX **lsnim** command. For example, if the name of your SPOT resource is "*61cosi*" then you could get the details by running:

lsnim -l 61cosi

To see the actual contents of a resource use "*nim -o showres <resource name>*". For example, to get a list of the software installed in your SPOT you could run:

nim -o showres 61cosi

AIX provides commands that may be used to manage the SPOT (or COSI) resource. Refer to the AIX man pages for further details.

- **mkcosi** Create a COSI (SPOT) for a thin server (diskless or dataless client) to mount and use.
- **chcosi** Manages a Common Operating System Image (COSI).
- **cpcosi** Create a copy of a COSI (SPOT).
- **lscosi** List the properties of a COSI (SPOT).
- **rmcosi** Remove a COSI (SPOT) from the NIM environment.

1.4 Update the diskless image

The SPOT (or COSI) created in the previous step should be considered the basic minimal diskless image. It does not contain all the software that would normally be installed as part of AIX if you were installing a standalone ("diskfull") system. You may want to install additional software into the SPOT before it is used to boot the nodes.

Any other software that you want on your nodes must be installed into the SPOT. You can use the AIX **chcosi** command to install, update, commit, reject, or remove software in a SPOT resource.

For example, to install and commit the optional OpenSSH packages from the AIX Expansion Pack you could issue the following command.

chcosi –i –c –s /dev/cd0 –f openssh.base openssh.license openssh.man.en_US openssh.msg.en_US openssh.msg.EN_US 61cosi

Any additional software that is needed can be installed in a similar manner.

> **Note**: When installing software into a SPOT the pre and post install scripts for a particular software package will not run any code that will impact your running system, (like restarting daemons etc.). The script will check to see if it's installing into a SPOT and it will not run that code.

You can also manually update files in the SPOT. The root file system for the diskless node will be created by copying the "*inst_root"* directory contained in the SPOT. In the SPOT we created for this example the "*inst_root*" directory would be:

/install/61cosi/usr/lpp/bos/inst_root/

For example, if you need to update the */etc/inittab* file that will be used on the diskless nodes you could edit:

/install/61cosi/usr/lpp/bos/inst_root/etc/inittab

All the diskless nodes that are booted using this SPOT will get a copy of *inst_root* as the initial root directory.

1.5 Define xCAT networks

Create a network definition for each network that contains cluster nodes. You will need a name for the network and values for the following attributes.

net The network address.
mask The network mask.
gateway The network gateway.

In our example we will assume that all the cluster node management interfaces and the xCAT management node interface are on the same network. You can use the xCAT **mkdef** command to define the network.

For example:

mkdef -t network -o net1 net=9.114.113.224 mask=255.255.255.224 gateway=9.114.113.254

Note: NIM also requires network definitions. When NIM was configured in an earlier step the default NIM master network definition was created. The NIM definition should match the one you create for xCAT. If multiple cluster subnets are needed then you will need an xCAT and NIM network definition for each one. A future xCAT enhancement will simplify this by automatically creating NIM network definitions based on the xCAT definitions.

1.6 Define the HMC as an xCAT node

The xCAT hardware control support requires that the hardware control point for the nodes also be defined as a cluster node.

The following command will create an xCAT node definition for an HMC with a host name of "*hmc01*". The *groups, nodetype, mgt, username*, and *password* attributes must be set.

mkdef –t node –o hmc01 groups="all" nodetype=hmc mgt=hmc username=hscroot password=abc123

1.7 Discover the LPARs managed by the HMC

This step assumes that the LPARs were already created using the standard HMC interfaces.

Use the xCAT **rscan** command to gather the LPAR information. In this example we will use the "-z" option to create a stanza file that contains the information gathered by **rscan** as well as some default values that could be used for the node definitions.

To write the stanza format output of **rscan** to a file called "*mystanzafile"* run the following command.

rscan –z hmc01 > mystanzafile

This file can then be checked and modified as needed. For example you may need to add a different name for the node definition or add additional attributes and values etc.

Note: The stanza file will contain stanzas for objects other than the LPARs. This information must also be defined in the xCAT database. It is not necessary to modify the non-LPAR stanzas in any way.

1.8 Define xCAT cluster nodes

The information gathered by the **rscan** command can be used to create xCAT node definitions.

Since we have put all the node information in a stanza file we can now pass the contents of the file to the **mkdef** command to add the definitions to the database.

cat mystanzafile | mkdef –z

You can use the xCAT **lsdef** command to check the definitions (ex. "*lsdef –l node01*"). After the node has been defined you can use the **chdef** command to make any additional updates to the definitions, if needed.

1.9 Define xCAT groups (optional)

You can see in the stanza file sample above that we have set the node "groups" attribute to "all". This means that you have also created a group definition named "all" that now contains the nodes we just defined.

You may also want to create additional groups. There are several ways to do this. One option would be to modify the node definitions to add another group name to the "groups" attribute value. You could do this using the **chdef** command. For example:

chdef –t node –o node01,node02,node03 groups=all,aixnodes

Another option would be to create a new group definition directly using the **mkdef** command as follows.

mkdef –t group –o aixnodes members="node01,node02,node03"

1.10 Create NIM diskless node definitions

You can set up NIM to support a diskless boot of nodes by using the xCAT **mkdsklsnode** command. This command uses information from the xCAT database and default values to run the appropriate NIM commands.

For example, to set up all the nodes in the group "*aixnodes*" to boot using the COSI "*61cosi*" you could issue the following command.

mkdsklsnode -c 61cosi aixnodes

To verify that you have allocated all the NIM resources that you need you can run the "**lsnim –l**" or the "**lsts**" command. For example, to check node "*node01*" you could run the following command.

lsnim -l node01

or

lsts node01

AIX provides several commands that can be used to manage diskless (also called thin server) nodes. See the AIX man pages for further details.

mkts Create a thin server and all necessary resources.
lsts List the status and software content of a thin server.
swts Switch a thin server to a different COSI.
dbts Perform a debug boot on a thin server.
rmts Remove a thin server from the NIM environment.

In preparation for the network boot, NIM configures bootp. You can verify that the /etc/bootptab file has an entry for each node. Also, it is recommended that you stop and restart the inetd service to ensure the new bootp configuration is loaded:

stopsrc -s inetd

startsrc -s inetd

1.11 Initiate a network boot

Initiate a remote network boot request using the xCAT **rnetboot** command. For example, to initiate a network boot of all nodes in the group "*aixnodes*" you could issue the following command.

rnetboot aixnodes

If you receive timeout errors from the **rnetboot** command, you may need to increase the default 60-second timeout to a larger value by setting ppctimeout in the site table:

chdef -t site -o clustersite ppctimeout=180

1.12 Verify the deployment

- As soon as the **rnetboot** command returns you can open a remote console to monitor the boot progress using the xCAT **rcons** command. This command

requires that you have conserver installed and configured. To configure conserver, run:

makeconservercf

Kill the conserver daemon if it is running, and restart it:

conserver &

(You may need to add /opt/freeware/bin and /opt/freeware/sbin to your PATH first).

To start a console:

rcons node01

-
- You can use the AIX **lsnim** command to see the state of the NIM installation for a particular node, by running the following command on the NIM master:

  lsnim -l <clientname>
- Retry and troubleshooting tips:
  - For p6 lpars, it may be helpful to bring up the HMC web interface in a browser and watch the lpar status and reference codes as the node boots.
  - Verify network connections
  - If the **rnetboot** returns "unsuccessful" for a node, verify that bootp and tftp is configured and running properly.
    - View /etc/bootptab to make sure an entry exists for the node.
    - Verify that the information in /tftpboot/<node>.info is correct.
    - Stop and restart inetd:

      stopsrc -s inetd
      startsrc -s inetd
    - Stop and restart tftp:
      stopsrc -s tftp
      startsrc -s tftp
    -
  - Verify NFS is running properly and mounts can be performed with this NFS server:
    - View /etc/exports for correct mount information.
    - Run the showmount and exportfs commands.
    - Stop and restart the NFS and related daemons:

      stopsrc -g nfs
      startsrc -g nfs

- Attempt to mount a filesystem from another system on the network.

- If the rnetboot operation is successful, but lsnim shows that the node is stuck at one of the netboot phases, you may need to redo your NIM definitions. Try the "short" approach first:

  nim -F -o reset node01

  nim -o dkls_init node01

  rnetboot -f node01

- If that doesn't work, you may need to delete the entire client definition from NIM and recreate it:

  nim -F -o reset node01

  nim -o deallocate -a root=root -a paging=paging -a dump=dump -a spot=61cosi node01

  nim -o remove node01

  mkdsklsnode -c 61cosi node01

  rnetboot -f node01

2.0 Notes

2.1 NIM diskless resources

The following list describes the required and optional resources that are managed by NIM to support diskless and dataless clients.

boot

Defined as a network boot image for NIM clients. The **boot** resource is managed automatically by NIM and is never explicitly allocated or deallocated by users.

SPOT

Defined as a directory structure that contains the AIX® run-time files common to all machines. These files are referred to as the **usr** parts of the fileset. The **SPOT** resource is mounted as the **/usr** file system on diskless and dataless clients.

Contains the **root** parts of filesets. The **root** part of a fileset is the set of files that may be used to configure the software for a particular machine. These **root** files are stored in special directories in the **SPOT**, and they are used to populate the root directories of diskless and dataless clients during initialization.

The network boot images used to boot clients are constructed from software installed in the **SPOT**.

A **SPOT** resource is required for both diskless and dataless clients.

root

Defined as a parent directory for client "**/**" (**root**) directories. The client root directory in the **root** resource is mounted as the "**/**" (**root**) file system on the client.

When the resources for a client are initialized, the client **root** directory is populated with configuration files. These configuration files are copied from the **SPOT** resource that has been allocated to the same machine.

A **root** resource is required for both diskless and dataless clients.

This resource is managed automatically by NIM.

dump

Defined as a parent directory for client dump files. The client dump file in the **dump** resource is mounted as the dump device for the client.

A **dump** resource is required for both diskless and dataless clients.

This resource is managed automatically by NIM.

paging

Defined as a parent directory for client paging files. The client paging file in the **paging** resource is mounted as the paging device for the client.

A **paging** resource is required for diskless clients and optional for dataless clients.

This resource is managed automatically by NIM.

home

Defined as a parent directory for client **/home** directories. The client directory in the **home** resource is mounted as the **/home** file system on the client.

A **home** resource is optional for both diskless and dataless clients.

When using the **csmsetupinstall** command you can specify that a **home** resource be created for the nodes by using the *attr=value* format. (Ex. "*home=yes*")

shared_home

Defined as a **/home** directory shared by clients. All clients that use a **shared_home** resource will mount the same directory as the **/home** file system.

A **shared_home** resource is optional for both diskless and dataless clients.

When using the **csmsetupinstall** command you can specify that a **shared_home** resource be created for the nodes by using the *attr=value* format. (Ex. "*shared_home=yes*")

tmp

Defined as a parent directory for client **/tmp** directories. The client directory in the **tmp** resource is mounted as the **/tmp** file system on the client.

A **tmp** resource is optional for both diskless and dataless clients.

When using the **csmsetupinstall** command you can specify that a **tmp** resource be created for the nodes by using the *attr=value* format. (Ex. "*tmp=yes*")

resolv_conf

This resource is a file that contains nameserver IP addresses and a network domain name.

It is copied to the **/etc/resolv.conf** file in the client's root directory.

A **resolv_conf** resource is optional for both diskless and dataless clients.

When using the **csmsetupinstall** command the name of the resolv_conf resource may be specified with the *attr=value* format. (Ex. *resolv_conf= master_net_conf*")

The AIX/NIM resources will remain allocated and the node will remain initialized until they are specifically unallocated and uninitialized (by running NIM commands). After the initial bring up, the node may be rebooted (using **rnetboot**) without having to redo any of the initial setup.

For diskless nodes the default is to allocate the SPOT, root, dump, and paging resources. All these resources are created under the same directory location that was specified with the **mkcosi** command when creating the SPOT that will be used for the node.