Using CSM in Large Scale Cluster Environments

Version 1.1

August, 2004

Vallard Benincosa
Jennifer Cranfill
Janet Ellsworth
Linda Mellor
Bruce Potter
Sean Safron

Introduction

IBM Cluster Systems Management (CSM) can be used to install, manage and monitor clusters of xSeries and pSeries servers from one single point of control. This paper chronicles the install and setup of a large cluster using CSM and some additional tools. Though some of the information pertains to this specific cluster and will need to be adjusted for different environments, the overall information will apply to clusters in general.

While CSM was beginning its development, the Extreme Cluster Administration Toolkit (xCAT), a script-based package, was developed by IBM's advanced technical sales support team and provided to customers purchasing Linux clusters based on IBM @server® xSeries®, IBM @server BladeCenter™ HS20 and IBM @server 325 (e325) servers and used by IBM Global Services, to address their need for tools to deploy and manage Linux clusters. While xCAT is not a fully supported product, it has been able to provide some valuable functionality for installing and configuring large Linux clusters.

In addition to install and configuration capabilities, CSM offers distributed command execution, configuration file management, monitoring and automated responses, hardware control, and diagnostic probes. CSM will be incorporating more of the additional install and hardware configuration capabilities that xCAT provides into its product functionality over its next several releases. This paper will review how CSM today can be used along with xCAT and some additional tools to setup a large cluster.

The job of installing over 1,000 nodes is not a simple task, however the tools and procedures outlined in this document will help to simplify the overall installation.

This document will go over how to take hardware shipped from manufacturing and install the operating system and CSM on all of the machines.

Hardware Configuration

Below is the hardware configuration for the cluster that this paper was written about:

- 1152 compute nodes (e325s) (3GB memory, 2 2GHz Opeteron CPU's)
- 8 user nodes (e325's) (9 GB memory, 2 2GHz Opteron CPU's)
- 12 storage nodes (x345's) (2GB memory, 2 2.7GHz Intel CPU's
- 2 cisco 3550's in each rack (one for the private interface, one for the public interface)
- 39 nodes per rack
- 1 40 port terminal server per rack (mrv itouch ir)
- 2 management servers: x345's (one as a back up)

Note that SLES 8 Service Pack 3 was the Linux distribution used on this cluster.

Note that the version of CSM used was CSM 1.3.3.

Management Node Setup

Install the management server

First you need to install the operating system on your management server. This is basic Linux installation and will not be covered in this paper.

Install CSM on the Management Server

You can obtain CSM as a Try and Buy from the following website:
http://www14.software.ibm.com/webapp/download/search.jsp?go=y&rs=csm
Directions on setting up a CSM management server can be found at in the CSM Install and Planning guide. This document and all the CSM online documentation can be found at http://publib.boulder.ibm.com/clresctr/windows/public/clusterbooks.html

Install xCAT tarballs

Get them here:
www.xcat.org

Tune the Management Server

Increase ARP tables

In large networks, the ARP table can be overloaded. This causes the appearance that CSM is slow. Here is an article on this:
http://www.uwsg.iu.edu/hypermail/linux/net/0307.3/0004.html
To fix it do this on the command line:

```
echo "512" >/proc/sys/net/ipv4/neigh/default/gc_thresh1
echo "2048" >/proc/sys/net/ipv4/neigh/default/gc_thresh2
echo "4096" >/proc/sys/net/ipv4/neigh/default/gc_thresh3
echo "240" >/proc/sys/net/ipv4/neigh/default/gc_stale_time
```

or, just run /opt/xcat/csm/sbin/helparp if you've installed the xcat csm tarball.

You will most likely want these changes to be permanent. Therefore, you should add these configurations into /etc/sysctrl.cfg as follows:

net.ipv4.conf.all.arp_filter = 1
net.ipv4.conf.all.rp_filter = 1
net.ipv4.neigh.default.gc_thresh1 = 512
net.ipv4.neigh.default.gc_thresh2 = 2048
net.ipv4.neigh.default.gc_thresh3 = 4096
net.ipv4.neigh.default.gc_stale_time = 240

Make nfs scale

Increase the number of threads by modifying /etc/sysconfig/nfs:
`USE_KERNEL_NFSD_NUMBER="80"`

Make /etc/hosts

This can be quite a chore. Look at the code in /opt/xcat/csm/bin/genhosts. It provides some examples that you can modify to make your own.

Define the nodes

This can also be a big chore.

Here are some example scripts to make things easier:
define all the compute nodes
definenode –n n0001-n1152
# assign console, and power stuff to a frame (note, F34 is a node group, see the section below for defining the node groups)
P=1; for i in $(lsnode -N F34); do chnode -n $i ConsolePortNum=$P ConsoleServerName=ts34 ConsoleMethod=mrv PowerMethod=bmc HWControlNodeId=$i HWControlPoint=172.29.123.$P; P=$(echo "$P + 1" | bc -l); done

You can also define console information automatically for each node by running:
definenode –n n0001-n1152 –C ts34::1:39,ts35::1:39,ts36::1:39 ConsoleMethod=mrv
This allows you to define all the nodes and their console server attributes when you inidially define the node. This assumes there are 39 nodes per console server

Note: if you are not using BMC, and hardware control points connect to more than one node (such as RSA's or management modules), you can also use the –H flag to define all the hardware control attributes at the same time

Make sure you back up the node definitions using lsnode –F >/store/nodedef. You can also use the csmbackup command to back all the management server information.

Define nodegroups

We recommend making the following node groups as it will save time on the command line
R1,R2,…R*n*
This is based on the rack numbers. So if your rack has 39 nodes, then n0001-n0039 would be members of the R1 node group.
Install Server Nodes
This is the bottom node in each rack and will be the install servers.

NoIsvrF1,NoIsvrF2,…NoIsvrF*n*
These are all the nodes in a frame except for the Install Servers

If you want to make a node group with all the nodes in the cluster that are not install servers
nodegrp -a 'nodegrp -d "," -S AllNodes InstallServers' NoIsvrgrp

run setupxcsm

Run: /opt/xcat/csm/sbin/setupxcsm to configure tables. Some of this may not be needed but some of it will. This command sets up the environment so that CSM can use the xcat commands. We recommend you log off and back in so that the environment is completely set after running.

Install Server configuration on the Management Server

The idea of an install server is to provide scalability and efficiency for installs across a large cluster. An install server is a node that can be used to install other nodes in the cluster. In this particular cluster (and in many clusters in the field) there will be an install server per rack of nodes. The install server will act as a dhcp server, tftp server and NFS server and help scale the management server.

Note that the current CSM concept of install server only has the NFS server on the install server – the dhcp server and tftp server currently are on the CSM management server only. In an upcoming release of CSM we will provide the capability officially to have the dhcp server and tftp server on the install servers. This paper will review how to set this up manually for now.

The flow of events is to first install all of the nodes that will be your install servers, configure them as install servers, and then have them help you install your compute nodes.

Note: Using the CSM_SKIP_ISVR environment variable to skip the install server sync
When you use install servers in a large CSM cluster, the CSM commands installnode and updatenode always update the install server with the latest files from /csminstall on the management server. This is done to ensure the node configuation files are current on the install servers and that the install server has all the needed files to do an install. However, in some cases if you have recently updated the install servers (through updateisvr, installnode or updatenode), you know the install servers have the latest versions of the files, and therefore there is no reason to update them. In this instance, you can set the environment variable CSM_SKIP_ISVR=1 before running installnode or updatenode and CSM will skip the update of the install servers. Please note: updateisvr also respects this environment variable, and is thus rendered useless when it is set. Please only set this

environment variable when you are sure your install servers are up-to-date otherwise you may encounter problems during the install.

post install scripts

There are a few post install scripts that will set up the Install Server right away, so that no manual work is needed to install them as long as cfmroot is setup correctly (in the next section). Together, the post install scripts and the cfmroot updates make the install server setup very easy.

We have created two post reboot scripts: Mount._InstallServers and setupInstallServers._InstallServers. Both of these executable scripts are placed in /csminstall/csm/scripts/installprereboot.

Mount._InstallServers

This script tells the install servers that they should mount /tftpboot from the management server. For this to work, /tftpboot should be exported from the management server:

```
echo "/tftpboot/      *(rw,no_root_squash,sync)" >>/etc/exports
rcnfsserver restart
```

Note: for Red Hat, you would run service nfs restart

The contents of Mount._InstallServers looks like this:

```
#!/bin/sh
# mount files on boot:
LOG=/var/log/csm/install.log

mkdir -p /tftpboot
echo "$MGMTSVR_HOSTNAME:/tftpboot       /tftpboot       nfs
rsize=8192,wsize=8192,timeo=14,intr 1 2" >>/etc/fstab

if [ "$DISTRO_NAME" = "SLES" ]
then
        chkconfig nfs 5 >>$LOG
        chkconfig -a nfs >>$LOG
else  # if RedHat
        chkconfig --add nfs >>$LOG
        chkconfig --level 345 nfs on >>$LOG
fi
```

It places /tftpboot into the fstab so that when the node boots it will attempt to mount /tftpboot and it also adds nfs to the startup so that nfs will be active.

setupInstallServers._InstallServers

This will install the tftp daemon. First, on the management server, copy the tftp-hpa rpm into /csminstall/csm/scripts/data directory so that it can be installed on the install servers. The script setupInstallServers._InstallServers is placed in the /csminstall/csm/scripts/installpostreboot directory. Its contents are as follows:

```
#!/bin/ksh
```

```
rpm –ivh --force ../data/tftp-hpa*
chkconfig –a inetd
chkconfig –a nfsserver
```

syslog
In any cluster, it is recommended that all system logs be redirected to the management server. We do this by creating a post install script as well as configuring the management server.

Management server setup:
edit /etc/sysconfig/syslog by adding the –r flag to accept remote logging:
SYSLOGD_OPTIONS=”-r “
now restart the syslog

Post install script:
create a file:
/csminstall/csm/scripts/installprereboot/0001CSM_syslog
The contents should look like this:

```
#!/bin/ksh

mv -f /etc/syslog.conf /etc/syslog.conf.ORIG
echo "*.*       $MGMTSVR_IP" >/etc/syslog.conf

case $DISTRO_NAME in
      SLES*)
            if grep 'SYSLOGD_PARAMS="-m0' /etc/sysconfig/syslog >/dev/null 2>&1
            then
                  :
            else
                  perl -pi -e 's/SYSLOGD_PARAMS="/SYSLOGD_PARAMS="-m0 /'
/etc/sysconfig/syslog
            fi
            /etc/init.d/syslog restart
            ;;
      RedHat*)
            /etc/rc.d/init.d/syslog start
            ;;
esac
exit 0
```

cfmroot settings

ssh
To allow all nodes to be able to ssh to eachother, link the .ssh file in /root to cfmroot/root. Now, ssh to yourself so that the management server keys are in the file as well.. You should be able to do this with out supplying a password.

```
ln –sf /root/.ssh /cfmroot/root/.ssh
```

dhcp
The install servers should have a dhcp server running that is exactly like the one on the management server. This helps with dhcp scaling. Otherwise, dhcp can only handle about 200 requests before it starts denying everybody.

```
ln –sf /etc/dhcpd.conf /cfmroot/etc/dhcpd.conf._InstallServers
```
In addition you should make a
/cfmroot/etc/dhcpd.conf.post file to run after the new file is copied. This is a script that has one line it in:
```
rcdhcpd restart   # for SuSE
or
service dhcp restart # for RedHat
```
make sure you `chmod 755 dhcpd.conf.post` so that it is an executable.

The limitation of this approach is that anytime hereafter the dhcpd.conf file is updated on the management server, cfmupdatenode will need to be rerun on the install servers as well.

tftp
SLES 8
You should link the inetd.conf from the management server to the /cfmroot/etc directory:
```
ln –sf /etc/inetd.conf /cfmroot/etc/inetd.conf._InstallServers
```
RedHat/SLES 9
link /etc/xinetd.d/tftp from the management server to the /cfmroot directory.
```
ln –sf /etc/xinetd.d/tftp /cfmroot/etc/xinetd.d/tftp._InstallServers
```

In addition you should do a post script to restart the inetd server so that when inetd.conf is copied to the install server it is restarted.
This script *inted.conf.post* is a one line script that looks like this:
```
rcinetd restart  # for SLES
service xinetd restart # for RedHat
```

additional cfmroot configuration
You should set up ntp as recommended by CSM as well as link the /etc files: hosts, passwd, shadow, and group in the /cfmroot/etc/ directory for all the nodes. The dhcp and tftp configuration only goes to the install servers.

Set up Terminal Servers

At this point we need to make sure that the terminal servers are functioning. We need the terminal servers so that we can do MAC address collecting. The terminal servers [today] come with a default setting from manufacturing. In the cluster for this document the default IP address was: 172.30.20.[frame #]
If you want to change the IP address, you can not do this through telnet, you have to do it with a serial connection. But first you must enable the serial connection.
If the default IP address is fine with you then you can move to the next section.
Otherwise, follow these steps to change the terminal server IP address: [This is for the MRV IR]
1. in /opt/xcat/csm/sbin/ there is a command called setup.itouch.ir. This command will enable a serial connection to be made from port 40 on the terminal server:

```
./setup.itouch.ir 172.30.20.1
```

2. Now take a terminal cable and connect one end to another terminal server in port 20 [lets say we do it in 172.30.20.2] and the other end into port 40 on the node we just enabled.

3. Now run the command:

```
./setup.itouch.ir 172.30.20.2 10.10.1.21
```

Where: 172.30.20.2 is the terminal server 2 that you are connecting from. And 10.10.1.21 is the IP address that you want to assign to 172.30.20.1.

At this point you can go on down the line and do this to all the machines. The alternative is to remove the plates on the front of the racks and set them to default. This however, requires some time, and this other way was faster.

Set up/Install the Install Servers

At this point the management server should be set up to install the install servers. We will get the install servers fully set up so that we can install the other nodes from them. The install servers will help scale the whole installation process.
There are three steps:

stage2: Mac address collection
stage1: CMOS settings, BIOS updates, Firmware updates
stage3: Service Processor naming and configuration

Notice that stage2 is before stage1. That is because historically, we would do stage1 with a floppy disk. That is not practical in large clusters, so we use stage2 first so we can do stage1 over the network.

Install Servers: Stage2

At this point the management server is set up and ready to run. We now need to collect the MAC addresses of the install servers. We assume that no hardware set up has been performed yet and that we have a terminal server connection for each node and networking is set up correctly.

First, try running the CSM command:

```
/opt/csm/bin/getadapters -w -N StageNodes
```

This may or may not work, but at least it will set up the files that are needed for the manual way, which tends to go a lot faster.

If getadapters takes too long for you, then there is a manual way that acts as a short cut.

First, you need to put a dynamic range of IP addresses that is not in the /etc/hosts file into the /etc/dhcpd.conf file. This range of IP addresses will correspond to nodes that are not installed and they will get the getmacs ramdisk and automatically start displaying their mac addresses for collection.

In my cluster, I have set the range 10.1.99.1 – 10.1.99.254.
This only gives 254 IP addresses, so more may be needed. You may have to make several ranges:
10.1.98.1 – 10.1.98.254, 10.1.97.1 – 10.1.97.254, etc
Running getadapters should have created an /etc/dhcpd.conf file. Put these ranges inside that newly created /etc/dhcpd.conf file. Make sure you put it in the correct subnet:
example:

```
…
subnet 10.1.0.0 netmask 255.255.0.0 {
                option routers 10.1.19.51;
          range 10.1.96.1 10.1.96.254;
          range 10.1.97.1 10.1.97.254;
          range 10.1.98.1 10.1.98.254;
          range 10.1.99.1 10.1.99.254;
                # CSM RANGE 10.1.0.0 (Please do not remove this line)
                default-lease-time -1;
                filename "/pxelinux.0";
                next-server 10.1.19.51;
}
…
```

restart dhcp

Now you need to create a pxe file that these nodes will get. Suppose that n0001 was one of your stage nodes that you ran get adapters on.
Go to /tftpboot/pxelinux.cfg/ and look. There should be a file called n0001.getmacs.
Assuming all nodes to be the same, figure out a partial hex file for your nodes so that all the nodes will tftp this file.
For example:
run the command:

```
m1:/opt/xcat/csm/sbin # gethostip -x 10.1.96.1
0A016001
```

now I want every node in the range of 10.1.96.1 – 10.1.96.254 to get the same file. So, I do this:

```
cp –p n0001.getmacs 0A0160
```

This means that every node that gets one of the 10.1.96.XX IP addresses will get this file. Additionally, you could just copy n0001.getmacs to 0A01, which would get all of the dynamic IP addresses in the example above.

Make sure that the file permissions of this newly created hex file is owned by tftpd:

```
chown tftpd:tftpd 0A01
```

and that permissions are read only:

```
chmod 440 0A01
```

Now test that you can get this file:

```
# cd
# tftp localhost
tftp> get 0A01
Received 11502 bytes in 0.0 seconds
tftp>
```

You should now be ready to go. Turn on all the install servers, or reboot them if they are already on.

Watch through the remote console to see them display the MAC addresses.

Once you see that they are all displaying the MAC addresses, go to /opt/xcat/csm/bin and run getmacs on these nodes:

```
./getmacs –N StageNodes
```

It will start writing the MAC addresses in the csm database for you.

When it is done, check that you have all the MAC addresses:

```
# lsmac –N StageNodes
m1:/tftpboot/pxelinux.cfg # lsmac –N StageNodes
n0001:  00:0D:60:14:54:3C
n0040:  00:0D:60:14:49:CA
n0079:  00:0D:60:14:74:C8
n0118:  00:0D:60:14:4D:6C
n0157:  00:0D:60:14:50:C4
n0196:  00:0D:60:14:4D:74
n0235:  00:0D:60:14:4A:20
n0274:  00:0D:60:14:4C:0E
n0313:  00:0D:60:14:78:B8
n0352:  00:0D:60:14:72:2A
n0391:  00:0D:60:14:78:00
n0430:  00:0D:60:14:62:3C
n0469:  00:0D:60:14:4C:4A
n0508:  00:0D:60:14:4E:56
n0547:  00:0D:60:14:46:D4
n0586:  00:0D:60:14:4B:C4
…
```

Troubleshooting:

1. If rconsole just shows “couldn’t connect to terminal server” then check your connections, etc.

2. Try running:
/opt/xcat/lib/setup.itouch.ir.oneport [terminal server ip] [speed (9600)] [port]

This information for the node can be gleaned from the lscons command in /opt/xcat/csm/bin.
3. Check that tftp works as well as dhcp. Make sure they are on and ready.
4. Send a note to the CSM mailing list if that fails.
5. Call your MAC expert friend.

At this point, we have the MAC's for the nodes. Run /opt/csm/csmbin/updatedhcp to put these values in.

Install Servers: Stage1

At this point stage1 is not supported. There are tools in xCAT that will help you, but you're mostly on your own to build the image. You need to download the flash images from www.pc.ibm.com/support
take these and roll them into a ram disk image. Using memdisk and pxelinux you can update everything through the network.
LCIT will be providing this in the future. If you have specific questions, please send a note to the CSM mailing list.

Install Servers: Stage 3

Stage 3 will do the service processor naming and configuration.

See the xCAT stage3 How to that comes in xCAT in the /opt/xcat/doc directory.
Let me go over briefly how to get this to work with the e325's:

You need to set up some of the xCAT tables for this to work.
ipmi.tab:
format: <node> <ipmi interface name>
example:
n0006 n0006-man0
n0007 n0007-man0
n0008 n0008-man0
n0009 n0009-man0

nodelist.tab
format: <node> <group,group,>
[csm2xcat should fill some of these in for you]
example:
n0001 all,csmnode
…

site.tab
In addition to the defaults that are set by csm2xcat, make sure you have:

```
tftpxcatroot          csm
ipmimaxp               100
ipmitimeout            3
ipmiretries           10
ipmisdrcache           yes
```

Now, make sure that the littlecat daemon is running:
m1:/opt/xcat/etc # ps -ef | grep littlecat
root   32559   1 0 May17 ?   00:00:05 /usr/bin/perl /opt/xcat/csm/sbin/littlecatd.pl

[This should've been started when you ran setupxcatcsm]
If it isn't running, start it up: /opt/xcat/csm/sbin/littlecatd.pl

Now, run:
/opt/xcat/csm/bin/nodeset –N StageNodes stage3

This will set it up for you. tail –f /var/log/messages to see the littlecatd respond. You may also need to configure this.

Install the Install Servers

At this point, the install servers are ready to install. Install them in the normal CSM fashion. The post install scripts that you setup on the management server above should be done. Make sure they are run when the node installs.

Post Install Server Setup

Make sure that cfmupdatenode ran on this node. Run it if it wasn't.
mount /tftpboot
rcinetd restart
rcdhcpd restart

Note – the above is for SLES. Run "service inetd restart" and "service dhcpd restart" for RedHat

Now update the install servers so they can become NFS servers:
updateisvr –vN StageNodes
(using the v option allows you to make sure things are going smoothly. This command can take about 20 minutes to run, since it is copying the entire distribution on to the stage nodes.

Congratulations, your install servers are now set up. Try tftp'ing to them to make sure they work.

Compute Node Installation

Now we need to repeat the same steps that we did for the install servers to the rest of the nodes. However, this time, we are aware that the compute nodes will each rely heavy on their respective install servers, so it is essential that they are set up right.

Now, set the compute node's install server.
i.e:
chnode –N NoIsvrF1 InstallServer=n0001

Do this for each frame. Or do this:
# P=1; for i in $(lsnode StageNode); do chnode –N NoIsvrF$P InstallServer=$i;
P=$(echo "$P + 1" | bc –l); done

Compute Nodes: Stage 2

The dhcp settings should already be set up as well as tftp. With each install server mirroring a DHCP server, DHCP should now be able to handle all the requests that it receives. Turn all of the nodes on. Make sure your terminal settings are correct. You may just want to try the 2nd node in each rack to make sure it works. Once you see it does, turn all the rest on. They should all start displaying their MAC addresses.

Go to /opt/xcat/csm/bin and run getmacs –n +NoIsvr1-NoIsvrX, where X is the last frame. This will get all the mac addresses.

You should now update the dhcp file so that is has all the correct MAC addresses. Run: /opt/csm/csmbin/updatedhcp on the management server

Now that it is done, run /opt/xcat/csm/bin/healdhcp –a . This will add the following specifics to each node stanza:
option host
option dhcp-server
option next-server

This will add the install server as the dhcp server and next server for each host. The script determines this by looking at the InstallServer attribute in the CSM data base.

Restart dhcp, and run cfmupdatenode –N InstallServers.

Alternative method for gathering xSeries MAC addresses

The CSM getadapters command is used to gather MAC addresses for nodes in order to prepare for node installation. For xLinux and pLinux installs, getadapters is called automatically from csmsetupks or csmsetupyast if the InstallAdapterMacAddress ManagedNode attribute for a node is not set. The getadapters command uses an appropriate MAC address collection method (MAC method) for each node based on the node's hardware. For non-Blade xSeries systems, getadapters uses the pxeboot MAC method by default.

The basic flow of the default pxeboot MAC method is:

1. CSM creates a special inital RAM disk image (getmacs initrd) that, when loaded and run on the node, will display the node's MAC address to the console.
2. CSM configures DHCP to respond to any network boot request with a dynamic IP address and sets up PXE to respond with the special getmacs initrd.
3. The rpower command is run to power on the node, which will cause the node to issued a DHCP broadcast request followed by a PXE broadcast request.
4. The CSM management server will respond with the initrd, and the node will begin displaying its MAC address to the console.
5. The rconsole command is run to open the node's console, and an expect script is run to read the MAC address being displayed to the console.
6. The pxeboot MAC method returns this MAC address back to the base getadapters command to be stored in the ManagedNode class.

However, there are some customers who have felt they would like more control over this methodology or who would like to gather MAC addresses before CSM hardware control has been set up. Therefore, an alternative MAC method called pxenoboot has been made available on the ECT web site. This MAC method will only run the rconsole command and the EXPECT script to read the MAC address from the console. Using this MAC method, you can use the following manual procedure to gather MAC addresses:

1. The customer runs the internal CSM command to set up for PXE boot MAC address collection:
   csmsetupboot -b getmacs -n *nodelist*
   This command was written and shipped as an internal-only command in CSM 1.3.1, a future release will externalize it.
2. The customer powers on all the nodes in *nodelist*. If hardware control is set up, the rpower command can be used. If not, the user can apply power manually to the nodes in whatever way is supported. When the node is powered, it will do a network boot and run the special getmacs initrd created by the csmsetupboot command and will display the MAC address to teh console.
3. The customer can then bring up some rconsole sessions to ensure the MAC addresses are being displayed.
4. The customer runs getadapters with the new pxenoboot MAC method to collect the MAC addresses.

This MAC method will start an rconsole session for each node and run the EXPECT scripts to gather the MAC addresses and return them to the main getadapters script. Once the MAC address collection has completed, the pxenoboot MAC method will reset the DHCP and PXE files back to their default state for the node. If the pxenoboot MAC method needs to be run for a node again, the user must start the entire process over beginning with the csmsetupboot command.

Compute Nodes: Stage 1

Same applies here as to the compute nodes above.

Compute Nodes: Stage 3

See the install server instructions. It's the same thing. Make sure you have dhcp, tftp, and nfs set correctly.

Compute Node Installation

Now you can install your nodes. At this point, we usually take the files in /opt/csm/csmbin, updatedhcp and createdhcp and make it so they do not do anything. Go to the main part and just type "exit 0;" so that they are not run by csmsetupyast.

Alternatively, you can export CSM_NO_SETUP_DHCP=1 (an unpublished environment variable).

This will prevent the commands csmsetupyast (or csmsetupks) and getadapters, from changing the /etc/dhcpd.conf

Now run:
csmsetupyast or csmsetupks on the nodes.

When finished, run:
CSM_FANOUT_DELAY=0 installnode –n +NoIsvrF1-NoIsvrFX.
All the nodes will now install. Good job. Go get yourself a drink. You deserve it.

Additonal Information on using CSM in a Large Scale Environment

Using FANOUT settings

CSM Fanout Settings
There are a number of ways to set the fanouts for CSM commands. Many of them can be set in environment variables and should be set in the root users shell profile. The default fanouts for the parallel CSM commands are listed below, along with recommendations for large clusters.

Installnode
Installnode uses two environment variables to control how many nodes are rebooted (and thus installed) at once, CSM_FANOUT and CSM_FANOUT_DELAY:

- CSM_FANOUT: sets the maximum number of concurrent reboots. If this variable is not set, 16 nodes are rebooted concurrently. If it is set to 0, all nodes are rebooted concurrently.
- CSM_FANOUT_DELAY: sets the delay in seconds between rebooting groups of nodes. If this variable is not set, the delay is 1200 seconds (20 minutes).

If you are using install servers you can increase the CSM_FANOUT variable by about 16 nodes per each install server. So if you have 10 install servers, you can install 160 nodes at once. If your nodes take less than 20 minutes to install, you can also decrease the CSM_FANOUT_DELAY to the amount of time it takes a node to install.

Updatenode
The number of machines updated in parallel with updatenode is controlled by the CSM_FANOUT environment variable (the same variable that is used for installnode). If this variable is not set, updatenode will run to 32 nodes in parallel. If you are using install servers, the same value for CSM_FANOUT can be used by both updatenode and installnode (so for 10 install servers, a CSM_FANOUT value of 160 is appropriate). Since cfmupdatenode is called by updatenode and should not be used with a high fanout, you should specify a lower CFM fanout to updatenode by calling it with --cfmoptions "-M 32", where 32 is an sample fanout for CFM.

Dsh and Dcp
Both dsh and dcp use the same fanout settings. You can specify the fanout via the DSH_FANOUT environment variable or on the command line with the -f flag. If neither is set the default is 64. The dsh and dcp fanout is only restrained by the number of remote shell commands that can be executed in parallel. Experiment with the dsh fanout on your system to see if higher values are appropriate.

Cfmupdatenode
The cfmupdatenode command uses the CSM_FANOUTenvironment variable or -M flag to control fanout. If neither of these is set, the fanout is 32. CFM does not utilize install servers, therefore you should not increase its fanout because of the existence of install servers. If you are setting the CSM_FANOUT environment variable in shell profile, you may want to always call cfmupdatenode with the -M flag to make sure it does not inherit high settings that were designed for updatenode or installnode.

Smsupdatenode
The number of machines updated in parallel with smsupdatenode is controlled by the CSM_FANOUT environment variable (the same variable that is used for installnode). If this variable is not set, smsupdatenode will run to 32 nodes in parallel. The SMS files reside on install servers. So, if you are using install servers, the same value for CSM_FANOUT can be used by smsupdatenode, updatenode and installnode (so for 10 install servers, a CSM_FANOUT value of 160 is appropriate).

Heartbeat Tunables

The Status attribute on the managed nodes in the cluster has its information provided by the underlying cluster infrastructure (RSCT). In a large cluster environment, the heartbeat mechanism can be tuned for efficiency based on size of the cluster, network configurations taking into account bandwidth/performance/traffic, and administrator preference…..

If you go to the management server and run "csmconfig", you will see that the HeartbeatFrequency is set to 12 and HeartbeatSensitivity is set to 8
Heartbeat Frequency is the number of seconds between heartbeat messages sent to the nodes. Heartbeat Sensitivity is the number of missed heartbeat messages sent to a node to declare that node is unreachable (its Status would then go to 0).

In a large cluster, you will often want to have less heartbeats than the default to reduce network traffic, and you would do this by increasing the frequency interval.

You can use the csmconfig command in the CSM Command and Technical Reference manual (see pointer to online documentation above) to change these values.