

[close window](#)

e-Newsletter Exclusive

[Print](#) 

# Systems Management from Scratch

December 2007 | by [Jaqui Lynch](#)

*Note: This is the full version of an article whose first half appeared in the December 2007 issue of IBM Systems Magazine, Open Systems edition.*

This article explains setting up an LPAR or server from scratch, covering steps from boot to being ready to add the applications. This is a fairly basic primer and will need to be tweaked to accommodate your company standards. The starting point for these instructions assumes that the hardware management console (HMC) is already set up if in use, or that the hardware is set up with a dedicated monitor if this is a standalone system. It also assumes we're installing AIX\* version 5.3, since the command syntax listed is for that version.

## Booting into SMS Mode

The first step is to boot into SMS mode to install the OS. There are two ways to do this:

- Standalone PCI server. Insert the CD or DVD into the drive and power on the system. At the first beep (when the keyboard and other prompts come up) hit F5 (or 5 if this is an ASCII terminal) several times, causing the system to boot in SMS mode.
- LPAR. At the HMC, create the partition and partition profile. Assign the disk, network, memory and processor resources to the partition profile as required. Then assign the DVD/CD drive as an optional resource. Make sure no one else has the DVD/CD. Insert the CD or DVD into the drive. At the HMC, right-click on the partition profile and select "Activate." You should then check the Open Terminal box (so you can use the vterm) and select "Advanced." Select "SMS" for the boot mode and select "OK" to close the advanced menu, then "OK" again. This ends the selection and opens the vterm, which should come up in SMS mode.

## Starting the SMS Mode Install

In this section you'll set up the boot devices and start the base-level installation. When the SMS mode menu comes up in the vterm or on the console, select "Remote IPL" (2) and then the boot device (2). The first boot device should be set to the CD/DVD drive and the second boot device should be set to the correct hard disk. Once these are set correctly, hit the x key to exit SMS, causing the system to boot from the install media. The vterm/console should then prompt you for the console; select "Enter" to use this as the console. Choose the language for the install and press the Enter key to continue, bringing you into the Welcome to BOS Install menu. Here you select Option 2 to change/show install settings. Option 1 is where you set the new/complete overwrite if you're setting up the system cleanly. Other options here include ensuring the correct disk is selected on which the OS is to be installed. You can also double-check the primary language here. Press "Enter" to confirm and begin the installation.

Once the installation is complete, the system reboots and should prompt you to accept the licenses. Once you do, press “F10.” When the install assistant comes up, set the date and time, the root password and then go on to configure the network. For this last step you’ll need to know the IP address, network host name, subnet mask, DNS server IP and the gateway IP. It also helps to know whether the switch allows you to set the port to GB full duplex (FDX) rather than auto-negotiate. “F10” then lets you exit the install assistant. You should select the option that stops install assistant from coming up again.

## Next Steps

At this point the system is basically installed but many of the more advanced filesets are missing. To install the rest of the filesets, expand some of the filesystems. Increasing the filesets’ size at this point can be done using smitty or the following commands:

```
chfs -a size='262144' /
chfs -a size='2110000' /usr
chfs -a size='262144' /var
chfs -a size='262144' /tmp
chfs -a size='262144' /home
chfs -a size='262144' /opt
```

These are minimum settings to ensure that you have sufficient space for the filesets to be installed. If they’re already larger, don’t run the chfs commands. It’s also here that I add two filesystems to use in the future: /usr/local (for third-party and other installs) and /usr/local/logs (for logging). I move logging to /usr/local/logs, as the system tends to crash if logging fills up /var, which is the default location.

First I create the JFS2 logs for the filesystems. I format them and then I create the filesystem logical volumes (LVs) and the filesystems themselves. Assuming rootvg is on hdisk0, start by doing an “lspv hdisk0”—this tells you the ppsize for the partitions on the disk. (For this exercise we’re assuming it’s 128 MB.) The following commands will create the JFS2 logs and the LVs for /usr/local and /usr/local/logs:

```
mklv -t jfs2log -y jlocallv rootvg 2
logform /dev/locallv
mklv -t jfs2log -y jlocloglv rootvg 2
logform /dev/locloglv
mklv -t jfs2 -y locallv rootvg 40 hdisk0
mklv -t jfs2 -y locloglv rootvg 3 hdisk0
mkdir /usr/local
crfs -d locallv-m /usr/local-a logname=jlocallv-A yes -p rw-V
jfs2
mount /usr/local
mkdir /usr/local/logs
crfs-d locloglv-m /usr/local/logs-a logname= jlocloglv-A yes-p
rw-V jfs2
mount /usr/local/logs
cd /usr/local
```

Use mkdir to create the following directories in /usr/local: bin src etc include lib sbin info share.

Then in /usr/local/man, create the following directories: man1 man3 man5 man7 man8. This is also a good time to increase the initial page space, hd6, to at least 4 GB, using smitty or the

chps command.

Since the CD is still loaded, you can use smitty install to install the filesets you may need on top of the base install. The best way to do this is to go to “smitty install” and choose “install and update software” and then “install from all available.” Select F4 to choose the CD device. On the next screen, set it to preview, extend filesystems, verify install, detailed output and accept new licenses. On the “software to install” line hit F4, then go through using F7 to choose the filesets you need. Run it in preview mode and then change it to “preview no” to run the upgrade. Additionally, if there’s an update CD or you’ve downloaded service packs or maintenance, you should install those the same way.

Now you have a basic system installed, and it’s time to double-check the microcode levels using the “ismcode -rd sys0” command. Make any necessary microcode changes and set the bootlist, etc and reboot. The bootlist is set by typing:

```
bootlist-m normal hdisk0 cd0
bosboot-a -d hdisk0
```

This sets the system to boot from hdisk0 and the bosboot makes sure the boot record is correctly set. Then you can issue “shutdown -r now” and the system will reboot.

## Environment Variables

This is a good time to set up some of the environment variables as follows in /etc/environment:

1. Add /usr/local/bin to the end of the PATH statement
2. Check the TZ is set correctly
3. Add the “EDITOR=vi” statement if it’s not there
4. Update the MANPATH to read:  
`MANPATH=$MANPATH:/usr/share/man:/usr/local/man`

Now we want to stop the X11 console and common desktop environment (CDE) console coming up and wasting cycles. This doesn’t stop you starting an xsession—it just removes the GUI login prompt.

To stop Xservers to stop x coming up at console:

```
cd /usr/dt/config
vi Xservers and comment out line starting :0
cp Xservers /etc/dt/config
cd /usr/lpp/X11/lib/X11/xdm
vi Xservers and comment out line starting :0
logout and in again
```

The next step is to set up logging as the default logging is very basic. Before you can have the system start logging to files those files must exist or you’ll get an error when the syslogd starts. So the first step is to create the empty log files:

```
cd /usr/local/logs
touch maillog syslog messages infolog sudo.log
```

Then we vi /etc/syslog.conf and code the following:

```
#
mail.debug /usr/local/logs/maillog
auth.notice /usr/local/logs/syslog
*.info /usr/local/logs/messages
daemon.info /usr/local/logs/infolog
# sudo log is below
local2.debug /usr/local/logs/sudo.log
```

The next step is to start and stop the syslogd:

```
stopsrc -s syslogd
startsrc -s syslogd
```

The system should now be logging correctly and the logs will all go to /usr/local/logs.

## Final Steps

The remaining steps include setting tunables, cleaning out files and general cleanup activities as well as installing any additional software.

The first step is to create the tunables file:

```
vi /etc/tunables/rc.tune
```

Contents are:

```
/usr/sbin/no -p -o rfc1323=1
/usr/sbin/no -p -o sb_max=1310720
/usr/sbin/no -p -o tcp_sendspace=262144
/usr/sbin/no -p -o tcp_recvspace=262144
/usr/sbin/no -p -o udp_sendspace=65536
/usr/sbin/no -p -o udp_recvspace=655360
/usr/sbin/nfso -p -o nfs_rfc1323=1
/usr/sbin/nfso -p -o nfs_socketsize=60000
/usr/sbin/nfso -p -o nfs_tcp_socketsize=60000
vmo -p -o minperm%=5
vmo -p -o maxperm%=90
vmo -p -o maxclient%=90
vmo -p -o lru_file_repage=0
ioo -p -o j2_nBufferPerPagerDevice=1024
ioo -p -o pv_min_pbuf=1024
ioo -p -o numfsbufs=2048
```

```
chmod u+x /etc/tunables/rc.tune
```

```
Run it - /etc/tunables/rc.tune
```

Check /etc/tunables/nextboot to make sure they were added.

Next, clean out files such as /etc/inittab, /etc/inetd.conf/ and /etc/rc.tcpip, ensuring that a copy is taken before making any changes. Particular attention should be paid to inetd.conf. I normally delete everything except services I

specifically want active—this includes removing all the services that are commented out. I remove those because occasionally a maintenance run will uncomment one and it's difficult to see that when the file has so many entries.

I also empty out /etc/hosts.equiv and /etc/hosts.lpd as many audit tools will flag them if they have any entries including comments. Obviously, if this system is a print server then printers may end up in hosts.lpd but initially it should be empty. In rc.tcpip you'll be commenting out things like sendmail, atm and snmp unless you plan to use them. If you want to use NTP then rc.tcpip is the place where you'll uncomment NTP after you configure it.

Next, edit /etc/motd and tailor it to your company's policy. Additionally you should change the login banner to something less friendly. This is done by:

```
vi /etc/security/login.cfg -set herald to change login prompt—  
this is coded at the end of the default section.
```

```
herald = "\n\n\n\n Unauthorized Use Prohibited \n\n\r Login :  
"
```

Since we may be starting some additional tasks at boot we have two options as to how to do that. One is to put each thing in inittab/, but I don't use this option. Instead I create a file called /etc/rc.local and I kick it off from inittab. Anything I want to start at boot gets kicked off from within rc.local/:

```
touch /etc/rc.local  
chmod u+x /etc/rc.local
```

Ensure that any local startup stuff goes in here, not in inittab. Add the following line to /etc/inittab:

```
rclocl:2:once:/etc/rc.local >/dev/null 2>&1
```

Then set up ntp to run, uncomment in /etc/rc.tcpip and configure /etc/ntp.conf. Then, add:

```
#broadcastclient  
server 0.north-america.pool.ntp.org  
server 1.north-america.pool.ntp.org  
server 2.north-america.pool.ntp.org
```

Save the file using:

```
startsrc -s xntpd.
```

Now you'll have consistent time on this server.

Other files that I work on here include /etc/security/user

where I set up login defaults. I also update /etc/security/.profile as the PATH export should say:

```
PATH=$PATH:/.....
```

By default it doesn't have the \$PATH at the front so each new user gets the path without the changes made in /etc/environment unless you change this. Additionally you may want to go into smitty, system environments and set things like maxuproc.

Next, add third-party software such as GCC, NMON, SUDO, LSOF, TCP Wrappers, OpenSSL and OpenSSH. These all go into /usr/local. There are binaries available for these but I normally compile them from scratch.

The last step is to mirror rootvg, setup dump devices, perform any other customizations and then reboot. Mirroring rootvg is a fairly simple process. Assuming the second disk to be used is hdisk0 the following steps will take care of it:

```
extendvg rootvg hdisk1 Adds hdisk1 to rootvg
mirrorvg rootvg hdisk1 Takes care of mirroring everything
bosboot -a -d hdisk1
bosboot -a -d hdisk0
bootlist -m normal hdisk0 hdisk1 cd0 rmt0
bootlist -m service cd0 rmt0 hdisk0 hdisk1
```

The above four commands write out the boot records and set the boot lists.

The next step is to set up the dump device:

```
sysdumpdev -e
```

The results of this command will be something like:

```
Estimated dump size in bytes: 141348044
```

This number needs to be divided by the PP size in bytes. A quick "lspv hdisk0" shows:

```
PP SIZE: 128 megabyte(s)
```

So to calculate PPs for the dump:

```
128 * 1024 * 1024 = 134217728
```

Now take the dump size and divide it by the PP bytes:

```
141348044 / 134217728 = 1.05
```

So the dump needs 2 PPs:

```
mklv -y dumplv01 rootvg 2 hdisk0
mklv -y dumplv02 rootvg 2 hdisk1
sysdumpdev -P -p /dev/dumplv01
sysdumpdev -P -s /dev/dumplv02
```

You're now at the point where you should reboot again. Once the system comes up you should backup rootvg using a mksysb before you go any further. This is your golden backup and should be saved. Then you can go on and add any other disks to new volume groups (keep rootvg as OS stuff only), new users, other products and so on. Don't forget to have a standard for user names and unique user IDs, so that users have the same user ID no matter which system they logon to. It also helps to have standards for LV, filesystem and log names.

### **Lay the Groundwork**

There are many steps involved in performing an OS installation, regardless of whether it's a standalone system or an LPAR. The best way to do it is to have a standardized base install, similar to the one above. You can then use that install as the basis for all others. This level of consistency allows you to easily move between systems that you have set up. The other advantage is using a standardized approach is that you're now setup nicely to use this OS image as the basis for a NIM install of other systems.

IBM Systems Magazine is a trademark of International Business Machines Corporation. The editorial content of IBM Systems Magazine is placed on this website by MSP TechMedia under license from International Business Machines Corporation.

©2007 MSP Communications, Inc. All rights reserved.

---