IBM Systems Technical Symposium Series 2012

14-17 August, Sydney, Australia

AX45 - Migrating to AIX 7.1 with nimadm

Chris Gibson

Migrating to AIX 7.1 with nimadm

August 2012

2012 IBM Power Systems Symposium. 14th -17th August, Sydney, Australia

Chris Gibson (chris.gibson@iag.com.au) – AIX Specialist at IAG.

IBM Champion for Power Systems, IBM Redbooks co-author & technical editor/writer for IBM Systems Magazine and IBM developerWorks.

IAG
Insurance Australia Group

Introduction

Minimize AIX migration downtime with NIM Alternate Disk Migration

Purpose

- This presentation **demonstrates** how to migrate to AIX 7.1 using the NIM Alternate Disk Migration tool.

- You'll learn how to **minimize** the **downtime** required when migrating to the latest release of the AIX operating system.

NIM

- The **AIX OS** provides the Network Installation Manager (NIM)
- Assist in **administering** and **updating** large numbers of AIX systems.
- NIM Alternate Disk Migration (**nimadm**) **feature**.
- Perform AIX **migrations** without the need for lengthy outages.
- Demonstrate **procedures** we used to migrate our AIX systems.
- Assume you are already very **familiar** with AIX and **NIM**.
- Already have a **NIM master** in your environment.
- Assume you are **NOT** familiar with nimadm.
- Read this Redbook, **NIM from A to Z in AIX 5L**:

  http://www.redbooks.ibm.com/redpieces/abstracts/sg247296.html?Open

Migrating AIX
Overview

Without NIM…what would we do?

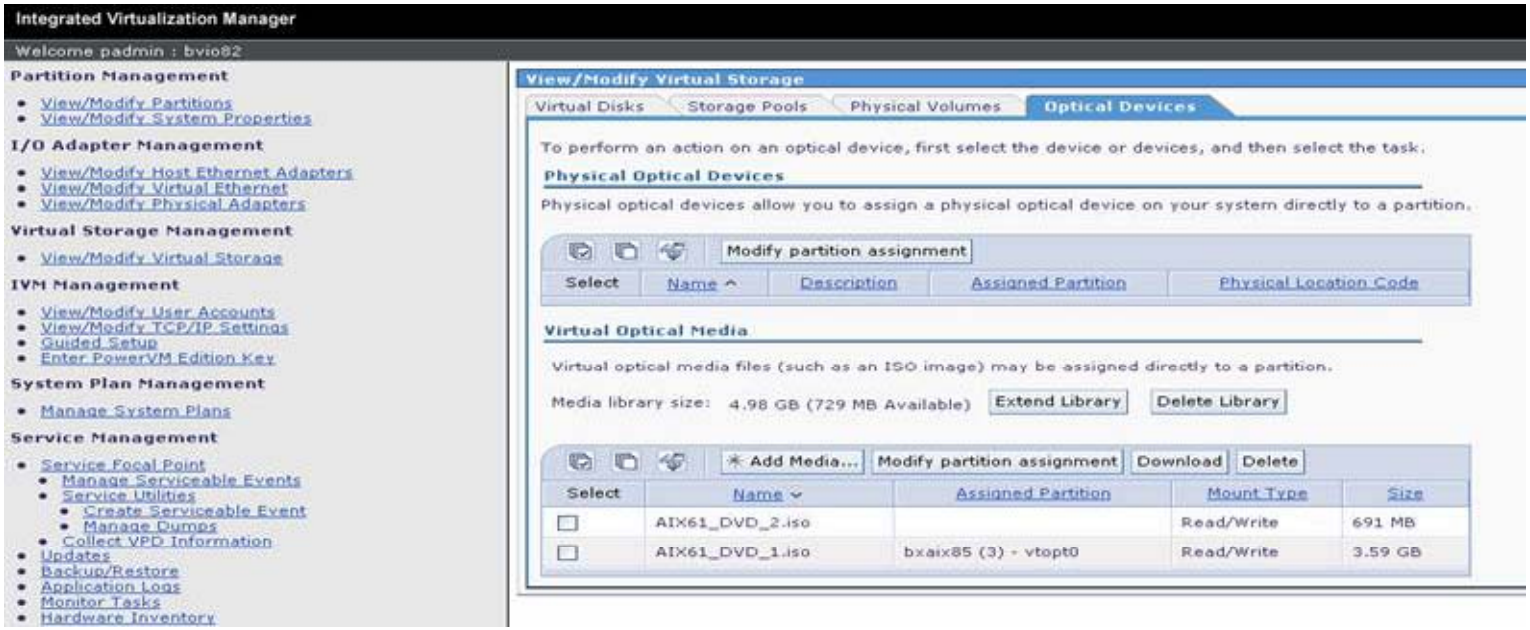
- Over the years, I've migrated to several new releases of the AIX OS.
- To do this I would have typically used one of the conventional methods.
- These methods consisted of either :
    A) Migration using the AIX installation DVD or
    B) Migration using NIM.
- **Method A** is still possible, even in virtualized environments via the use of File-Backed devices.
- **Method B** is also perfectly viable by **network booting** the client LPAR and performing the migration using a NIM master.

- http://gibsonnet.net/aix/AIX_Migration_with_File-Backed_VIOS_Devices.htm

- http://www.ibm.com/developerworks/aix/library/au-aix-system-migration-installation/index.html

Migrating AIX
Overview

- Virtual Media Library - File-backed devices - on VIOS. Poor man's NIM.



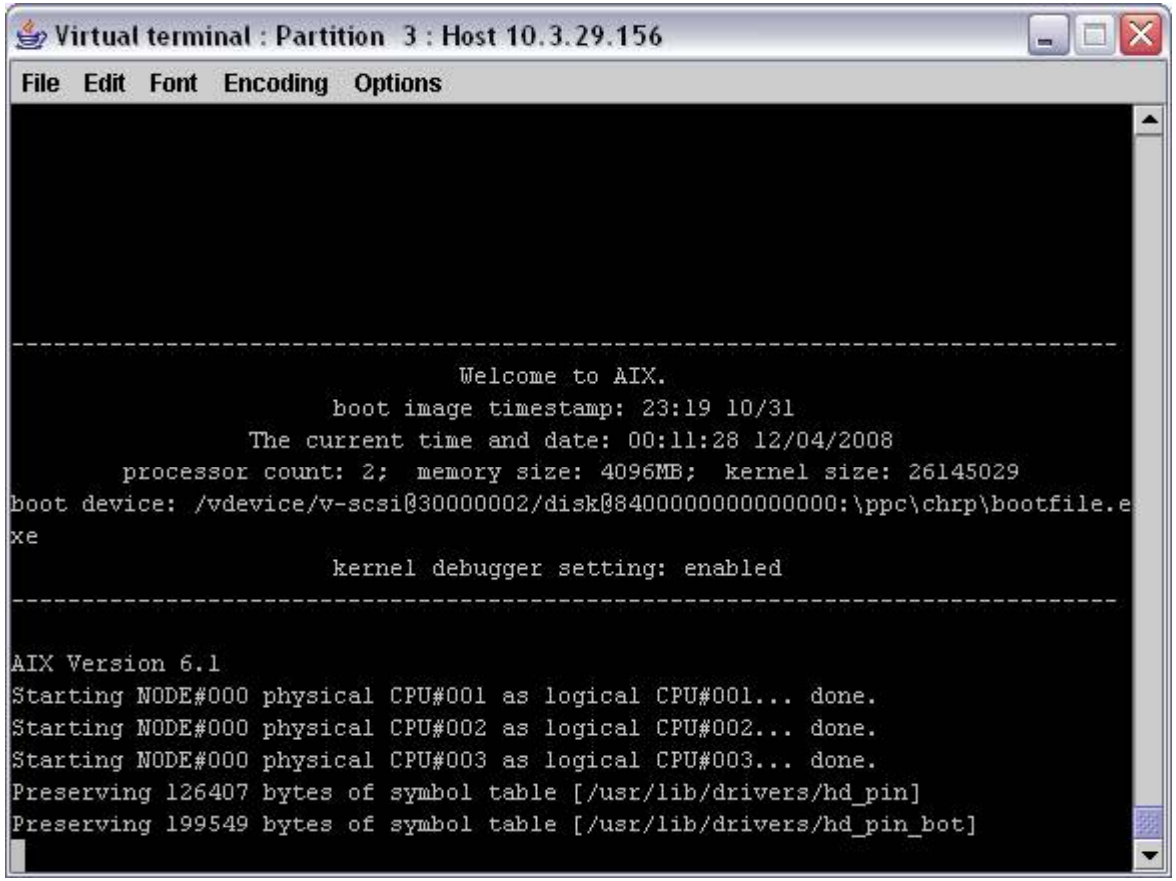

```
$ lsrep
Size(mb) Free(mb) Parent Pool         Parent Size      Parent Free
    5099      729 rootvg                  139776           110592

Name                                  File Size Optical         Access
AIX61_DVD_1.iso                            3679 vtopt0          rw
AIX61_DVD_2.iso                             691 None            rw
```

Corporate Office

IAG Insurance Australia Group

Migrating AIX
Boot from media – then migrate….

- **Boot** the AIX system from either **NIM** or VML **virtual CD**. Perform **migration**.

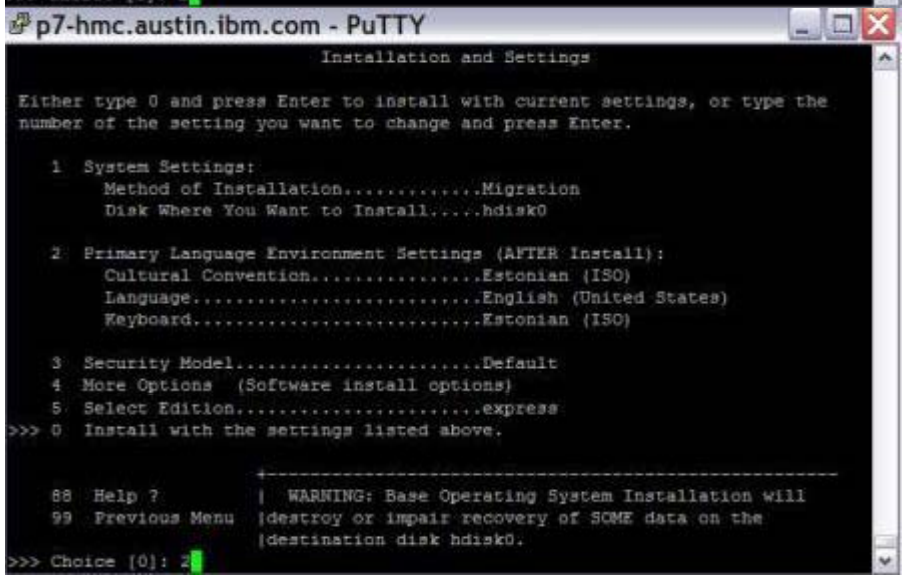


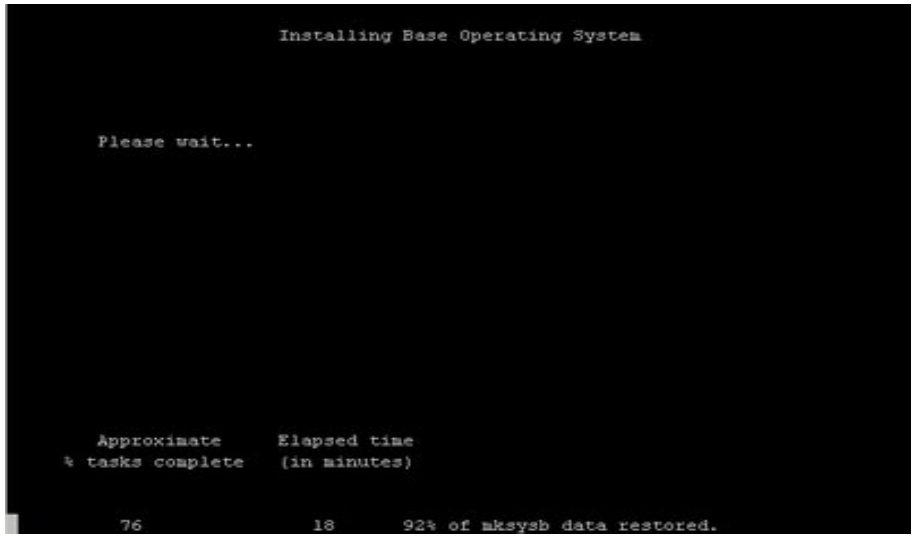
Corporate Office

IAG Insurance Australia Group

Migrating AIX
Boot from media – then migrate…

BOS menus – select Migration Install, etc…





Migrating AIX
Down time!

Down time!

- The downside with both of these methods is that they both require **significant downtime** on the system while the migration takes place.
- This **downtime** could be anywhere from **30-45 minutes** to an **hour or more**, depending on the system.
- This can be a **concern** in environments with **tight outage windows**. Systems not clustered for HA.

Advantages
nimadm

Why should I use nimadm?

- The nimadm utility offers several **advantages** over a **conventional** migration.
- Creates a copy of a NIM client's rootvg (on a spare disk on the client).
- Similar to **alternate disk install** (alt_disk_install).
- Migrate the alternate rootvg disk to a newer version of AIX.

- All of this can be done **without disruption** to the client .
- There is **no outage** required to perform the migration.

- After the migration is finished, the only downtime required will be a **scheduled reboot** of the system.
- Another advantage is that the actual **migration process** occurs on the **NIM master**.
- Taking the load off the client LPAR.
- This **reduces** the processing **overhead** on the client.

- **Minimizes** the performance **impact** to the running **applications**.

Advantages
nimadm

Benefits?

- For customers with a **large number** of AIX systems.
- nimadm tool supports migrating **several** clients **at once**.

- **Benefits** over other migration methods:

1. Reduced downtime for the client.
2. Migration is executed while the system is up and running.
3. No disruption to any of the applications or services running on the client.
4. Upgrade can be done at a time convenient to the administrator.
5. At a later stage, a reboot can be scheduled in order to restart the system at the later level of AIX.
6. Back out, to a previous version of AIX, is simple. Does not require recovery from mksysb.

Advantages
nimadm

Other reasons to use nimadm.

- Process is very flexible: Can be customized – optional customization resources i.e. **image_data**, **bosinst_data**, **pre/post_migration scripts**, **exclude_files**, and so on.

- **Quick recovery** from migration failures. All changes are performed on the rootvg copy (altinst_rootvg).
- Any serious problems with the migration, the original rootvg is still available and the system has not been impacted.

- If a migration **fails** or terminates at any stage, nimadm is able to quickly recover from the event and **clean up** afterwards.
- There is little for the administrator to do except determine why the migration failed, rectify the situation, and attempt the nimadm process again.

- If the migration completed but issues are discovered after the reboot, then the administrator can **back out easily** by booting from the **original rootvg** disk.

Preparing the NIM environment
nimadm

Preparation

- You must have a **NIM master** running **AIX 7.1** or higher with the latest Technology Level and Service Pack.

- The **lpp_source** and **SPOT** NIM resources that have been selected for the migration must **match** the **AIX level** to which you are **migrating**.

- The NIM master must have the **bos.alt_disk_install.rte** fileset installed in the AIX 7.1 **SPOT*** that will be used for the migration.

- The **NIM master** (as **always**) should be at the **same or higher AIX level** than the level you are migrating to on the **client**.

Preparing the NIM environment
nimadm

Preparation – continued.

- The target **client** must be **registered** with the **NIM** master as a **standalone** NIM client.
- The NIM master must be able to **execute remote** commands on the client using **rsh**.
- Ensure the NIM **client** has a **spare disk** (not allocated to a volume group) large enough to contain a complete **copy** of its **rootvg**.
- If rootvg is mirrored, break the mirror and use one of the disks for the migration.
- Assign a new **SAN disk** for this purpose.
- Ensure the NIM master has a volume group (for example, **nimadmvg**) with enough **free space** to cater for a complete copy of the **client's rootvg**.
- If **more than one** AIX migration is occurring for **multiple** NIM clients, make sure there is **capacity** for a copy of **each clients** rootvg.

Local disk caching versus NFS

nimadm

Why not use NFS?

- By default, nimadm uses NFS for many of the tasks during the migration.
- This can be a problem on **slower networks** because NFS writes can be very expensive.

- To avoid using NFS, a **Local Disk Caching** option exists that can provide some **performance advantages**.
- Local disk caching allows the NIM master to avoid having to use NFS to write to the client.

- Local Disk Caching will create the client file systems in a volume group on the NIM master.

- It will then use **streams** (via **rshd**) to cache all of the data from the client to the file systems on the NIM **master**.

Local disk caching - Advantages

nimadm

Why Local Disk Caching?

- Local disk caching **provides**:
  - **Improved performance** for nimadm operations on relatively slow networks.
  - Improved performance for nimadm operations that are **bottlenecked** in **NFS** writes.
  - **Decreased CPU usage on the client.**
  - Client file systems not exported.
  - Allows TCB enabled systems to be migrated with nimadm.
  - For best performance, use a volume group on the NIM master that does not contain the NIM resources being used for the AIX migration.

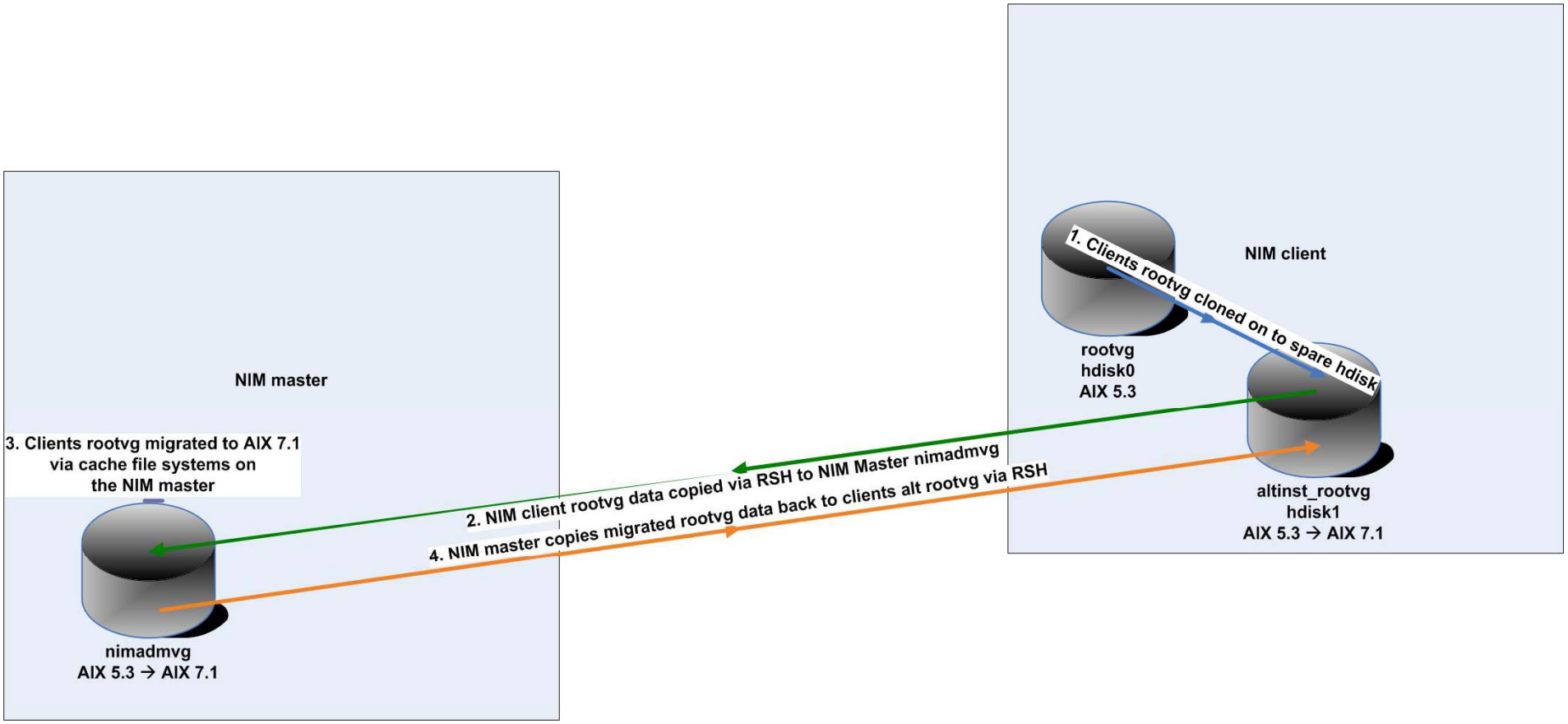
Some potential **disadvantages**:

- Cache file systems take up **space** on the **NIM master**.
- **Increased CPU** usage on the **NIM master**.
- Increased disk I/O activity on the master.

Local disk caching
nimadm

nimadm phases with disk caching

- There are 12 phases to nimadm, however…
- Here are the most important steps!



Corporate Office

IAG Insurance Australia Group

Phases
nimadm

12 Phases

- The **nimadm** command performs a migration in 12 phases:

1. Master issues **alt_disk_install** to the client. Makes a copy of the clients rootvg to target disk. In this phase, the alternate root volume group (**altinst_rootvg**) is created.

2. Master **creates** the **cache file systems** in the **nimadmvg** volume group. Some initial checks for the required migration disk space are performed.

3. The NIM **master copies** the NIM **client's data** to the **cache** file systems in nimadmvg. This data copy is done via **rsh**.

4. If a **pre-migration script** resource has been specified, it is executed in this phase.

5. System **configuration files** are **saved**. Initial migration space is calculated and appropriate file system expansions are made. The **bos image** is **restored** and the **device database** is **merged** (similar to a conventional migration).

6. All system **filesets** are **migrated** using **installp**. Any required RPM images are also installed during this phase.

Phases
nimadm

Phases

7. If a **post-migration script** resource has been specified, it is executed in this phase.

8. The **bosboot** command is run to create a **client boot image**, which is written to the client's alternate boot logical volume (**alt_hd5**).

9. All the **migrated data** is now **copied** from the NIM master's local cache file and synced to the client's **alternate rootvg** via **rsh**.

10. The NIM **master cleans up** and **removes** the local cache file systems.

11. The **alt_disk_install** command is called again to make the final adjustments and put altinst_rootvg to sleep. The **bootlist** is set to the target disk.

12. Cleanup is executed to end the migration.

Consider this…
nimadm

Before we start…

- A few points for you to consider.
- I recommended that you **do not to make any changes** to your system once the **migration** is **underway**, such as adding **users**, changing **passwords**, adding **print queues**, and the like.
- If possible, wait until the migration has finished and the system has been rebooted on the new version of AIX.
- If you must perform administration tasks prior to the reboot, you should take note of the changes and re-apply them to the system after it has been rebooted into AIX 7.1.
- We **developed**, **tested**, and **verified** our **migration procedures** several times **before** implementing them on our **production** systems.
- This allowed us time to verify that the steps were correct and that the AIX migrations would complete as expected. I **recommend** you **do the same**.
- If you have a **multibos** image in rootvg, remove it. AIX migrations are not supported with multibos enabled systems.
- Ensure all rootvg LVs are renamed to their legacy names. If necessary, create a new instance of rootvg and reboot the LPAR.

Consider this…
nimadm

Before we start…

- Ensure all rootvg LVs are renamed to their legacy names. If necessary, create a new multibos instance and reboot the LPAR. For example:

```
# multibos –sXp
# multibos –sX
# shutdown –Fr
```

- Remove the old multibos instance. `# multibos -R`
- Confirm the legacy LV names are now in use that is, not bos_.

```
rootvg:
LV NAME             TYPE       LPs     PPs     PVs  LV STATE      MOUNT POINT
hd5                 boot       1       1       1    closed/syncd  N/A
hd6                 paging     32      32      1    open/syncd    N/A
hd8                 jfslog     1       1       1    open/syncd    N/A
hd4                 jfs        4       4       1    open/syncd    /
hd2                 jfs        31      31      1    open/syncd    /usr
hd9var              jfs        20      20      1    open/syncd    /var
hd3                 jfs        4       4       1    open/syncd    /tmp
hd1                 jfs        4       4       1    open/syncd    /home
hd10opt             jfs        4       4       1    open/syncd    /opt
```

Migrating to AIX 7.1 using nimadm

nimadm

Let's start…

- Let's use **nimadm** now to migrate an AIX system.
- We will migrate a system from AIX 5.3 to AIX 7.1. The NIM master in this environment is running AIX 7.1 TL1 SP4.
- Our NIM client is running AIX 5.3 TL12 SP4 (migrating to AIX 7.1 TL1 SP4).

- Ensure that you **document** the system and perform a mksysb before performing any maintenance activity. **You know this already, right? But I have to say it!**

- Ensure that you read the **AIX 7.1 release notes** and review the documented requirements such as the amount of free disk space required.
- Check your firmware levels, HMC code, VIOS code etc. Use the **FLRT** to check compatibility.

- Prior to a migration, it is always a good idea to run the **pre_migration script** on the system to catch any issues that may prevent the migration from completing successfully. You can find this script on the AIX 7.1 installation media.

Migrating to AIX 7.1 using nimadm

nimadm

Pre-Migration checks…

- Run this script, review the output (in **/home/pre_migration**), and correct any issues that it reports before migrating.

```
# ./pre_migration

All saved information can be found in: /home/pre_migration.120621143915
Checking size of boot logical volume (hd5).
Listing software that will be removed from the system.
Listing configuration files that will not be merged.
Listing configuration files that will be merged.
Saving configuration files that will be merged.
Running lppchk commands. This may take awhile.
Please check /home/pre_migration.120621143915/software_file_existence_check for possible
    errors.
Please check /home/pre_migration.120621143915/software_checksum_verification for possible
    errors.
Please check /home/pre_migration.120621143915/tcbck.output for possible errors.
All saved information can be found in: /home/pre_migration.120621143915
It is recommended that you create a bootable system backup of your system
before migrating.
```

- **Commit** any applied filesets.

Migrating to AIX 7.1 using nimadm
nimadm

Spare disk, rsh…

- If rootvg is **mirrored**, break the mirror and reduce it to a single disk.
- Assign another **SAN** disk.
- This gives you a **spare disk** that can be used for the migration.
- To allow nimadm to do its job, we must **temporarily enable rshd** on the client LPAR.
- We will **disable** it again **after** the migration.

```
# chsubserver -a -v shell -p tcp6 -r inetd
# refresh –s inetd
# cd /
# vi .rhosts
nim01 root
# chmod 600 .rhosts
```

- On the NIM master, I can now '**rsh**' to the client and run a command as root.

```
# rsh aix1 whoami
root
```

Corporate Office

IAG Insurance Australia Group

Migrating to AIX 7.1 using nimadm

nimadm

Ready to migrate…

- At this point I'm ready to migrate.
- The process will take around **30-45 minutes**; all the while the **applications** on the LPAR will continue to **function** as **normal**.
- On the NIM master, I have created a new volume group (VG) named **nimadmvg**.
- This VG has enough **capacity** to cater for a full copy of the NIM clients rootvg.
- This VG will be empty until the migration is started.
- NIM client: spare disk with enough capacity for a full copy of its rootvg.
- On the master (**nim1**):

```
# lsvg -l nimadmvg
nimadmvg:
LV NAME TYPE LPs PPs PVs LV STATE MOUNT POINT
```

- On the client (**aix1**):

```
# lspv
hdisk0 0000273ac30fdcfc rootvg active
hdisk1 000273ac30fdd6e  None
```

Migrating to AIX 7.1 using nimadm

nimadm

bos.alt_disk_install.rte

- The fileset bos.alt_disk_install.rte fileset is installed on the NIM master:

```
# lslpp -l bos.alt_disk_install.rte
Fileset                      Level  State      Description
  ----------------------------------------------------------------------------
Path: /usr/lib/objrepos
  bos.alt_disk_install.rte  7.1.1.15  APPLIED    Alternate Disk Installation
                                                 Runtime

Path: /etc/objrepos
  bos.alt_disk_install.rte  7.1.1.15  APPLIED    Alternate Disk Installation
                                                 Runtime
```

- And it is also installed in the AIX 7.1 TL1 SP4 SPOT:

```
# nim -o showres spotaix710104 | grep bos.alt
  bos.alt_disk_install.rte  7.1.1.15    C     F    Alternate Disk Installation
```

nimadm command line options

nimadm

Flags and options

- The **nimadm** command is executed from the NIM master.

```
# nimadm -j nimadmvg -c aix1 -s spotaix710104 -l lpp_sourceaix710104 -d hdisk1 –Y
```

Where:

- –j flag specifies the **VG** on the master which will be used for the migration
- -c is the **client** name
- -s is the **SPOT** name
- -l is the l**pp_source** name
- -d is the **hdisk** name for the alternate root volume group (altinst_rootvg)
- -Y agrees to the software **license** agreements for software that will be installed during the migration.

nimadm command line options

nimadm

Executing nimadm…

- Now I can sit back and **watch** the migration take place.
- All migration activity is logged on the NIM master in the **/var/adm/ras/alt_mig** directory.
- For this migration, the log file name is **aix1_alt_mig.log**.
- Here's a sample of some of the output you can expect to see for each phase:

```
MASTER DATE: Thu Jun  7 12:22:14 EETDT 2012
CLIENT DATE: Thu Jun  7 12:22:14 EETDT 2012
NIMADM PARAMETERS: -j nimadmvg -c aix1-s spotaix710104 -l lpp_sourceaix710104 -d hdisk1 -Y
Starting Alternate Disk Migration.

+-----------------------------------------------------------------------------+
Executing nimadm phase 1.
+-----------------------------------------------------------------------------+
Cloning altinst_rootvg on client, Phase 1.
Client alt_disk_install command: alt_disk_copy -j -M 7.1 -P1 -d "hdisk1"
Calling mkszfile to create new /image.data file.
Checking disk sizes.
Creating cloned rootvg volume group and associated logical volumes.
Creating logical volume alt_hd5
Creating logical volume alt_hd6
Creating logical volume alt_hd8
Creating logical volume alt_hd4
…etc..
Generating a list of files
for backup and restore into the alternate file system...
Phase 1 complete.
```

Corporate Office

IAG Insurance Australia Group

Phase 1
nimadm

```
+-----------------------------------------------------------------------------+
Executing nimadm phase 1.
+-----------------------------------------------------------------------------+
Cloning altinst_rootvg on client, Phase 1.
Client alt_disk_install command: alt_disk_copy -j -M 7.1 -P1 -d "hdisk1"
Calling mkszfile to create new /image.data file.
Checking disk sizes.
Creating cloned rootvg volume group and associated logical volumes.
Creating logical volume alt_hd5
Creating logical volume alt_hd6
Creating logical volume alt_hd8
Creating logical volume alt_hd4
Creating logical volume alt_hd2
Creating logical volume alt_hd9var
Creating logical volume alt_hd3
Creating logical volume alt_hd1
Creating logical volume alt_hd10opt
Creating logical volume alt_local
Creating logical volume alt_tsmlog
Creating logical volume alt_loglv
Creating logical volume alt_hd7
Creating logical volume alt_hd71
Creating logical volume alt_chksyslv
Creating logical volume alt_rvgj2loglv
Creating logical volume alt_ncplv
Creating logical volume alt_auditloglv
Creating logical volume alt_symanteclv
Creating logical volume alt_hd11admin
Creating /alt_inst/ file system.
Creating /alt_inst/admin file system.
Creating /alt_inst/home file system.
Creating /alt_inst/home/ncp file system.
Creating /alt_inst/opt file system.
Creating /alt_inst/opt/Symantec file system.
Creating /alt_inst/tmp file system.
Creating /alt_inst/usr file system.
Creating /alt_inst/usr/local file system.
Creating /alt_inst/usr/local/chksys file system.
Creating /alt_inst/var file system.
Creating /alt_inst/var/log file system.
Creating /alt_inst/var/log/audit file system.
Creating /alt_inst/var/tsm/log file system.
Generating a list of files
for backup and restore into the alternate file system...
Phase 1 complete.
```

Corporate Office

IAG Insurance Australia Group

Phase 2, 3 & 4

nimadm

```
+-----------------------------------------------------------------------------+
Executing nimadm phase 2.
+-----------------------------------------------------------------------------+
Creating nimadm cache file systems on volume group nimadmvg.
Checking for initial required migration space.
Creating cache file system /aix1_alt/alt_inst
Creating cache file system /aix1_alt/alt_inst/admin
Creating cache file system /aix1_alt/alt_inst/home
Creating cache file system /aix1_alt/alt_inst/home/ncp
Creating cache file system /aix1_alt/alt_inst/opt
Creating cache file system /aix1_alt/alt_inst/opt/Symantec
Creating cache file system /aix1_alt/alt_inst/tmp
Creating cache file system /aix1_alt/alt_inst/usr
Creating cache file system /aix1_alt/alt_inst/usr/local
Creating cache file system /aix1_alt/alt_inst/usr/local/chksys
Creating cache file system /aix1_alt/alt_inst/var
Creating cache file system /aix1_alt/alt_inst/var/log
Creating cache file system /aix1_alt/alt_inst/var/log/audit
Creating cache file system /aix1_alt/alt_inst/var/tsm/log

+-----------------------------------------------------------------------------+
Executing nimadm phase 3.
+-----------------------------------------------------------------------------+
Syncing client data to cache ...

+-----------------------------------------------------------------------------+
Executing nimadm phase 4.
+-----------------------------------------------------------------------------+
nimadm: There is no user customization script specified for this phase.
```

Corporate Office

IAG Insurance Australia Group

Phase 5
nimadm

```
+-------------------------------------------------------------------------------+
Executing nimadm phase 5.
+-------------------------------------------------------------------------------+
Saving system configuration files.
Checking for initial required migration space.
Expanding /aix1_alt/alt_inst/ local filesystem.
Filesystem size changed to 262144
Setting up for base operating system restore.
/aix1_alt/alt_inst
Restoring base operating system.
Merging system configuration files.
Running migration merge method: ODM_merge Config_Rules.
Running migration merge method: ODM_merge SRCextmeth.
Running migration merge method: ODM_merge SRCsubsys.
Running migration merge method: ODM_merge SWservAt.
Running migration merge method: ODM merge pse.conf.
Running migration merge method: ODM_merge vfs.
Running migration merge method: ODM_merge xtiso.conf.
Running migration merge method: ODM_merge PdAtXtd.
Running migration merge method: ODM_merge PdDv.
Running migration merge method: convert_errnotify.
Running migration merge method: passwd_mig.
Running migration merge method: login_mig.
Running migration merge method: user_mrg.
Running migration merge method: secur_mig.
Running migration merge method: RoleMerge.
Running migration merge method: methods_mig.
Running migration merge method: mkusr_mig.
Running migration merge method: methods_mig.
Running migration merge method: mkusr_mig.
Running migration merge method: group_mig.
Running migration merge method: ldapcfg_mig.
Running migration merge method: ldapmap_mig.
Running migration merge method: convert_errlog.
Running migration merge method: ODM_merge GAI.
Running migration merge method: ODM_merge PdAt.
Running migration merge method: merge_smit_db.
Running migration merge method: ODM_merge fix.
Running migration merge method: merge_swvpds.
Running migration merge method: SysckMerge.
```

Phase 6
nimadm

```
+-----------------------------------------------------------------------------+
Executing nimadm phase 6.
+-----------------------------------------------------------------------------+
Installing and migrating software.
Updating install utilities.
+-----------------------------------------------------------------------------+
                    Pre-installation Verification...
+-----------------------------------------------------------------------------+
Verifying selections...done
Verifying requisites...done
Results...

SUCCESSES
---------
  Filesets listed in this section passed pre-installation verification
  and will be installed.

  Mandatory Fileset Updates
  -------------------------
  (being installed automatically due to their importance)
  bos.rte.install 7.1.1.16                  # LPP Install Commands

  << End of Success Section >>

+-----------------------------------------------------------------------------+
                   BUILDDATE Verification ...
+-----------------------------------------------------------------------------+
Verifying build dates...done
FILESET STATISTICS
------------------
    1  Selected to be installed, of which:
        1  Passed pre-installation verification
  ----
    1  Total to be installed

+-----------------------------------------------------------------------------+
                         Installing Software...
+-----------------------------------------------------------------------------+

installp:  APPLYING software for:
        bos.rte.install 7.1.1.16


. . . . . << Copyright notice for bos >> . . . . . .
 Licensed Materials - Property of IBM
Copyright Interactive Systems Corporation 1985, 1991.
```

Phase 7, 8 & 9

nimadm

```
+-------------------------------------------------------------------------------+
Executing nimadm phase 7.
+-------------------------------------------------------------------------------+
nimadm: There is no user customization script specified for this phase.

+-------------------------------------------------------------------------------+
Executing nimadm phase 8.
+-------------------------------------------------------------------------------+
Creating client boot image.
bosboot: Boot image is 53248 512 byte blocks.
Writing boot image to client's alternate boot disk hdisk1.

+-------------------------------------------------------------------------------+
Executing nimadm phase 9.
+-------------------------------------------------------------------------------+
Adjusting client file system sizes ...
Adjusting size for /
Adjusting size for /admin
Adjusting size for /home
Adjusting size for /opt
Adjusting size for /tmp
Adjusting size for /usr
Adjusting size for /usr/local
Adjusting size for /var
Adjusting size for /var/log
Syncing cache data to client ...
```

Phase 10
nimadm

```
+-----------------------------------------------------------------------------+
Executing nimadm phase 10.
+-----------------------------------------------------------------------------+
Unmounting client mounts on the NIM master.
forced unmount of /aix1_alt/alt_inst/var/tsm/log
forced unmount of /aix1_alt/alt_inst/var/log/audit
forced unmount of /aix1_alt/alt_inst/var/log
forced unmount of /aix1_alt/alt_inst/var
forced unmount of /aix1_alt/alt_inst/usr/local/chksys
forced unmount of /aix1_alt/alt_inst/usr/local
forced unmount of /aix1_alt/alt_inst/usr
forced unmount of /aix1_alt/alt_inst/tmp
forced unmount of /aix1_alt/alt_inst/opt/Symantec
forced unmount of /aix1_alt/alt_inst/opt
forced unmount of /aix1_alt/alt_inst/home/ncp
forced unmount of /aix1_alt/alt_inst/home
forced unmount of /aix1_alt/alt_inst/admin
forced unmount of /aix1_alt/alt_inst
Removing nimadm cache file systems.
Removing cache file system /aix1_alt/alt_inst
Removing cache file system /aix1_alt/alt_inst/admin
Removing cache file system /aix1_alt/alt_inst/home
Removing cache file system /aix1_alt/alt_inst/home/ncp
Removing cache file system /aix1_alt/alt_inst/opt
Removing cache file system /aix1_alt/alt_inst/opt/Symantec
Removing cache file system /aix1_alt/alt_inst/tmp
Removing cache file system /aix1_alt/alt_inst/usr
Removing cache file system /aix1_alt/alt_inst/usr/local
Removing cache file system /aix1_alt/alt_inst/usr/local/chksys
Removing cache file system /aix1_alt/alt_inst/var
Removing cache file system /aix1_alt/alt_inst/var/log
Removing cache file system /aix1_alt/alt_inst/var/log/audit
Removing cache file system /aix1_alt/alt_inst/var/tsm/log
```

Corporate Office

IAG Insurance Australia Group

Phase 11 & 12

nimadm

```
+-----------------------------------------------------------------------------+
Executing nimadm phase 11.
+-----------------------------------------------------------------------------+
Cloning altinst_rootvg on client, Phase 3.
Client alt_disk_install command: alt_disk_copy -j -M 7.1 -P3 -d "hdisk1"
## Phase 3 ###################
Verifying altinst_rootvg...
Modifying ODM on cloned disk.
forced unmount of /alt_inst/var/tsm/log
forced unmount of /alt_inst/var/log/audit
forced unmount of /alt_inst/var/log
forced unmount of /alt_inst/var
forced unmount of /alt_inst/usr/local/chksys
forced unmount of /alt_inst/usr/local
forced unmount of /alt_inst/usr
forced unmount of /alt_inst/tmp
forced unmount of /alt_inst/opt/Symantec
forced unmount of /alt_inst/opt
forced unmount of /alt_inst/home/ncp
forced unmount of /alt_inst/home
forced unmount of /alt_inst/admin
forced unmount of /alt_inst
Changing logical volume names in volume group descriptor area.
Fixing LV control blocks...
Fixing file system superblocks...
Bootlist is set to the boot disk: hdisk1 blv=hd5

+-----------------------------------------------------------------------------+
Executing nimadm phase 12.
+-----------------------------------------------------------------------------+
Cleaning up alt_disk_migration on the NIM master.
Cleaning up alt_disk_migration on client aix1.
```

Corporate Office

IAG Insurance Australia Group

After the migration
nimadm

Migration complete!

- After the migration is complete, confirm that the **bootlist** is set to the **altnst_rootvg** disk.

```
# lspv | grep rootvg
hdisk0 0000273ac30fdcfc rootvg active            << AIX 5.3
hdisk1 000273ac30fdd6e altinst_rootvg active     << AIX 7.1
```

- At an agreed time, reboot the LPAR and confirm that the system is now running AIX 7.1.

```
# oslevel –s
5300-12-04-1119

# shutdown –Fr

; system reboots here…takes a few minutes.

# oslevel –s
7100-01-04-1216
```

After the migration
nimadm

Health checks and clean up

- At this point, perform some general AIX system health checks to ensure that the system is configured and running as expected.
- http://www.ibm.com/developerworks/aix/library/au-upgrade_power6/index.html
- There is also a **post_migration** script that you can run to verify the migration. You can find this script in **/usr/lpp/bos**, after the migration.
- You may want to consider upgrading other software such as **openssl**, **openssh**, **lsof**, etc at this stage.
- The **rsh** daemon can now be **disabled** after the migration.

```
# chsubserver -d -v shell -p tcp6 -r inetd
# refresh –s inetd
# cd /
# rm .rhosts
# ln -s /dev/null .rhosts
```

After the migration
nimadm

Post migration

- With the migration finished, the **applications** are started and the application support team verify that everything is functioning as expected.
- Take a **mksysb** and **document** the system configuration after the migration.
- Once we are all **satisfied** that the migration has completed successfully, we then return rootvg to a mirrored disk configuration (only if required).

```
# lspv | grep old_rootvg
hdisk0 000071da26fe3bd0 old_rootvg
# alt_rootvg_op -X old_rootvg
; If mirrored rootvg
# extendvg –f rootvg hdisk0
# mirrorvg rootvg hdisk0
# bosboot -a -d /dev/hdisk0
# bosboot -a -d /dev/hdisk1
# bootlist -m normal hdisk0 hdisk1
# bootlist -m normal -o hdisk0 blv=hd5 hdisk1 blv=hd5
```

After the migration
nimadm

Uh oh! Back out!

- If there was an issue with the migration, easily back out to the previous release of AIX.
- Instead of re-mirroring rootvg (above), we would **change** the **boot list** to point at the previous rootvg disk (**old_rootvg**) and reboot the LPAR.

```
# lspv | grep old_rootvg
hdisk0 000071da26fe3bd0 old_rootvg

# bootlist -m normal hdisk0
# bootlist -m normal –o hdisk0 blv=hd5
# shutdown –Fr
```

- This is much **simpler** and **faster** than **restoring** a **mksysb** image (via NIM, tape, or DVD), as you would with a **conventional** migration method.

Reduced downtime
nimadm

No more after hours upgrades!

- By using **nimadm**, we were able to **reduce** the **overall downtime** required when migrating our systems to **AIX 7.1**.
- We were also given a **convenient** way to **back out** a migration, had it been necessary to do so.
- I hope this provides you with **some ideas** on how to best migrate your systems to AIX 7.1, when the **time comes**.
- http://www.ibm.com/developerworks/aix/library/au-migrate_nimadm/?S_TACT=105AGY20&S_CMP=HP

nimadm – migrate mksysb image

nimadm

Alternative migration technique

- Migrate mksysb image from one version of AIX to another.
- Migrate AIX 6.1 mksysb image to AIX 7.1.
- Similar to nimadm – uses cache file systems (nimadmvg).
- Use the migrated mksysb image to install a new LPAR via NIM.
- Source LPAR running AIX 6.1 TL6 SP3. Took a mksysb of the source system to NIM master.

```
root@aix61lpar / # oslevel -s
6100-06-03-1048
```

- NIM mksysb resource named cg-aix61.

```
root@nim1 / # lsnim -t mksysb | grep cg
cg-aix61        resources       mksysb
```

Corporate Office

IAG Insurance Australia Group

nimadm – migrate mksysb image
continued

Alternative to standard nimadm

- AIX 7.1 lpp_source and SPOT ready and waiting on my NIM master.

```
root@nim1 / # lsnim -t lpp_source | grep aix7
aix7tl0sp2      resources       lpp_source

root@nim1 / # lsnim -t spot | grep aix7
spotaix7tl0sp2      resources       spot
```

nimadm – migrate mksysb image
continued

Alternative to standard nimadm

- To migrate the mksysb image from AIX 6.1 to AIX 7.1, I ran the following **nimadm** command:

```
root@nim1 / # nimadm -T cg-aix61 -O /export/mksysb/cg-aix71 -s spotaix7tl0sp2 -l
  aix7tl0sp2 -j nimadmvg -Y -N cg-aix71
```

- The flags indicate the following:
- -T specifies the existing AIX 6.1 NIM mksysb resource.
- -O specifies the output location for the migrated mksysb resource.
- -s indicates the AIX 7.1 NIM SPOT resource for the migration.
- -l indicates the AIX 7.1 NIM lpp_source resource for the migration.
- -j identifies the volume group that will be used on the NIM master to create file systems.
- -Y Agress to required software license agreements.
- -N specifies the name of the new AIX 7.1 NIM mksysb resource to be created after migration.

nimadm – migrate mksysb image
continued

Migrate mksysb…

- Output.

```
root@nim1 / # nimadm -T cg-aix61 -O /export/mksysb/cg-aix71 -s spotaix7tl0sp2 -l aix7tl0sp2 -j nimadmvg -Y -N cg-aix71
Initializing the NIM master.
Verifying alt_disk_migration eligibility.
Initializing log: /var/adm/ras/alt_mig/cg-aix61_alt_mig.log
Starting Alternate Disk Migration.

+-------------------------------------------------------------------------------+
Executing nimadm phase 1.
+-------------------------------------------------------------------------------+
Processing target mksysb resource cg-aix61.
Restoring /image.data from mksysb image.

+-------------------------------------------------------------------------------+
Executing nimadm phase 2.
+-------------------------------------------------------------------------------+
Creating nimadm cache file systems on volume group nimadmvg.
Checking for initial required migration space.
Creating cache file system /cg-aix61_mm_alt/alt_inst
Creating cache file system /cg-aix61_mm_alt/alt_inst/admin
Creating cache file system /cg-aix61_mm_alt/alt_inst/home
Creating cache file system /cg-aix61_mm_alt/alt_inst/opt
Creating cache file system /cg-aix61_mm_alt/alt_inst/tmp
Creating cache file system /cg-aix61_mm_alt/alt_inst/usr
Creating cache file system /cg-aix61_mm_alt/alt_inst/var
Creating cache file system /cg-aix61_mm_alt/alt_inst/var/adm/ras/livedump
```

nimadm – migrate mksysb image
continued

Migrate mksysb…NIM mksysb resource

- New NIIM msysb resource after successful migration.

```
+-----------------------------------------------------------------------------+
Executing nimadm phase 11.
+-----------------------------------------------------------------------------+
Defining NIM mksysb resource ...
New NIMmksysb resource name is cg-aix71
```

- Once the migration had completed, I found a new mksysb file in /export/mksysb (as expected). The file was named cg-aix71. And a new NIM mksysb resource was now available.

```
root@nim1 / # ls -ltr /export/mksysb/ | grep aix7
-rw-r--r--    1 root     system   1967411200 Jan 06 11:00 cg-aix71

root@nim1 / # lsnim -t mksysb | grep aix7
cg-aix71         resources       mksysb
```

Corporate Office

IAG Insurance Australia Group

nimadm – migrate mksysb image
continued

Migrate mksysb…NIM mksysb resource

- I was able to use the new, migrated, AIX 7.1 mksysb image to install a new LPAR.

```
AIX Version 7
Copyright IBM Corporation, 1982, 2010.
login: root
root's Password:
*******************************************************************************
*                                                                             *
*  Welcome to AIX Version 7.1!                                                *
*                                                                             *
*                                                                             *
*  Please see the README file in /usr/lpp/bos for information pertinent to    *
*  this release of the AIX Operating System.                                  *
*                                                                             *
*******************************************************************************
Last login: Thu Jan  6 13:21:04 EETDT 2011 on /dev/vty0

# oslevel -s
7100-00-02-1041
```

https://www.ibm.com/developerworks/mydeveloperworks/blogs/cgaix/entry/migrating_to_aix_7_1_with_nimadm?lang=en

Other tips and tricks
Migrating to AIX 7.1

AIX Tunables Post Migration

- After an AIX migration, I usually like to run the **tuncheck** command to verify the current tunable parameters are valid. One area that may indicate a tuning problem is the **AIX error report**. If you see the following messages in the **errpt** output, you may want to **verify** the current **settings** are valid:

```
IDENTIFIER TIMESTAMP  T C RESOURCE_NAME  DESCRIPTION
D221BD55   0523115112 I O perftune       RESTRICTED TUNABLES MODIFIED AT REBOOT
---------------------------------------------------------------------------
LABEL:          TUNE_RESTRICTED
IDENTIFIER:     D221BD55

Date/Time:       Wed May 23 11:51:16 EET 2012
Sequence Number: 676
Machine Id:      00C342C64C00
Node Id:         lparaix53
Class:           O
Type:            INFO
WPAR:            Global
Resource Name:   perftune

Description
RESTRICTED TUNABLES MODIFIED AT REBOOT

Probable Causes
SYSTEM TUNING

User Causes
TUNABLE PARAMETER OF TYPE RESTRICTED HAS BEEN MODIFIED

        Recommended Actions
        REVIEW TUNABLE LISTS IN DETAILED DATA

Detail Data
LIST OF TUNABLE COMMANDS CONTROLLING MODIFIED RESTRICTED TUNABLES AT REBOOT, SEE FILE /etc/tunables/lastboot.log
vmo
```

Other tips and tricks
Migrating to AIX 7.1

AIX Tunables Post Migration - continued

- In the output above you'll notice that we are advised to check the **/etc/tunables/lastboot.log** for a modified restricted **vmo** tuning parameter.
- At this point I usually like to run the **tuncheck** command against the current **/etc/tunables/nextboot** file and review its output.

- As you can see, in the example below, we are warned that several restricted tunables are not set to their default values.
- These values **may not be appropriate** for your newly migrated **AIX 7.1** (or **6.1**) system.

- Settings that worked well with **5.3** are most likely **no longer appropriate** with 7.1.

```
# tuncheck -p -f /etc/tunables/nextboot
Warning: restricted tunable lrubucket is not at default value
Warning: restricted tunable strict_maxperm is not at default value
Warning: unknown parameter lru_file_repage in stanza vmo
Warning: restricted tunable maxperm% is not at default value
Warning: restricted tunable maxclient% is not at default value
Checking successful
```

Other tips and tricks
Migrating to AIX 7.1

AIX Tunables Post Migration - continued

- Based on the output above, the tuning for this newly migrated 7.1 system appears to be **inappropriate**. Unless we have a **valid reason** (which has been verified by **IBM AIX support**) we should **set** these **tunables** to their **default** AIX 7.1 settings.

- You can reset individual tunables to their defaults using the **–d flag** and the corresponding **tuning command**. For example to set the maxperm% tunable to its default you would run the following **vmo** command:

```
# vmo -p -d maxperm%
Modification to restricted tunable maxperm%, confirmation required yes/no yes
Setting maxperm% to 90 in nextboot file
Setting maxperm% to 90
Warning: a restricted tunable has been modified
```

- If you want to set **all** the **vmo tunables** back to their defaults you would run the following vmo command with the –D option (requires **bosboot** & **reboot**):

```
#  vmo -r -D
```

Other tips and tricks
Migrating to AIX 7.1

AIX Tunables Post Migration - continued

- The tundefault command can also reset tuning to default parameters. The command launches all the tuning commands (ioo, vmo, schedo, no, nfso, and raso) with the -D flag. This resets all the AIX tunable parameters to their default values.

- The –r flag defers the reset to default value to the next reboot. This clears the stanza(s) in the /etc/tunables/nextboot file and if necessary, runs bosboot and warns that a reboot is needed.

```
# tundefault -r
```

- Once the tunables have been reset, re-run the **tuncheck** command and ensure it runs without errors:

```
# tuncheck -p -f /etc/tunables/nextboot
Checking successful
```

Other tips and tricks
Migrating to AIX 7.1

AIX Tunables Post Migration - continued

- **Unless** you've **permanently set restricted tunables** in your **/etc/tunables/nexboot** file, the migration will change the systems default tuning to match the newer version of AIX.
- For example, we **observed** the **following** tuning **changes** on our AIX 5.3 system after migrating to 7.1.

- The maxperm default value changed from 80 to 90:

```
maxperm%        80      80      80      1       100     % memory        D
maxperm%        90      90      90      1       100     % memory        D
```

- The minperm default value changed from 20 to 3:

```
minperm%        20      20      20      1       100     % memory        D
minperm%        3       3       3       1       100     % memory        D
```

- Note that with AIX 7.1 **lru_file_repage** is **hardcoded** to 0 and removed from the list of vmo tunables. Please refer to the following document, Oracle Architecture and Tuning on AIX v2.20, for more information. http://www-03.ibm.com/support/techdocs/atsmastr.nsf/WebIndex/WP100883

Corporate Office

IAG Insurance Australia Group

Other tips and tricks
Migrating to AIX 7.1

Multibos

- As mentioned **earlier**, **multibos** is **not supported** in **nimadm** environments. Before you start a **nimadm** migration make sure you have **removed any old standby BOS** instance and that your rootvg logical volumes are **not** using any **bos**_ LV names.

- During our tests we found that even though we removed the standby instance (**multibos –R**), the **nimadm** process failed with the following error:

```
+-----------------------------------------------------------------------------+
    Executing nimadm phase 11.
+-----------------------------------------------------------------------------+
    Cloning altinst_rootvg on client, Phase 3.
    Client alt_disk_install command: alt_disk_copy -j -M 7.1 -P3 -d "hdisk1"
    ## Phase 3 ###################
    Verifying altinst_rootvg...
    alt_disk_copy: 0505-218 ATTENTION: init_multibos() returned an unexpected
    result.
    Cleaning up.
    …etc..
    0505-187 nimadm: Error cloning altinst_rootvg on client.
    Cleaning up alt_disk_migration on the NIM master.
    Cleaning up alt_disk_migration on client lpar1.
    Client alt_disk_install command: alt_disk_install -M 7.1 -X
    Bootlist is set to the boot disk: hdisk0 blv=hd5
```

Other tips and tricks
Migrating to AIX 7.1

Multibos - continued

- Given that the error appeared to be related to **init_multibos**, we assumed the failure was due to some **multibos checks** being performed by **alt_disk_copy** on the client.
  - Unfortunately, it appears that '**multibos –R**' **may not clean up** the **/bos_inst** directory. If this directory exists the nimadm operation will most likely fail.
  - The **simple fix** (in our case) was to remove the **/bos_inst** directory before attempting the AIX migration.

```
# rm –r /bos_inst
```

Other tips and tricks
Migrating to AIX 7.1

AIX 7.1 NIM master - migrating clients to AIX 6.1

- If you plan on using your AIX 7.1 NIM master to migrate your **AIX 5.3** clients to **AIX 6.1**, then make sure you install the **AIX 7.1 bos.alt_disk_install.rte** fileset into the **AIX 6.1 SPOT** resource first. Failure to do so will result in your **nimadm** operation reporting the following error message:

```
# nimadm -j nimadmvg -c lparaix53 -s spotaix610605 -l lpp_sourceaix610605 -d hdisk2 -Y
Initializing the NIM master.
Initializing NIM client lparaix53.
0042-001 nim: processing error encountered on "master":
   /usr/bin/lslpp: Fileset bos.alt_disk_install.rte not installed.

0505-204 nimadm: SPOT spotaix610605 does not have bos.alt_disk_install.rte installed.
0505-205 nimadm: The level of bos.alt_disk_install.rte installed in SPOT
spotaix610605 (0.0.0.0) does not match the NIM master's level (7.1.1.0).
Cleaning up alt_disk_migration on the NIM master.
```

- You can **verify** the correct fileset is installed in your **6.1 SPOT** using the following nim command:

```
# nim -o showres spotaix610605 | grep bos.alt_disk_install.rte
  bos.alt_disk_install.rte   7.1.1.15    A     F    Alternate Disk Installation
```

Corporate Office

IAG Insurance Australia Group

Other tips and tricks
Migrating to AIX 7.1

AIX 7.1 NIM master migrating clients to AIX 6.1

- You must install the AIX 7.1 **bos.alt_disk_install.rte** fileset into your AIX 6.1 SPOT resource.

```
# smit nim_res_op

....etc...

> spotaix610605

Customize a SPOT

Type or select values in entry fields.
Press Enter AFTER making all desired changes.

                                                [Entry Fields]
* Resource Name                                  spotaix610605
* Source of Install Images                      [lpp_sourceaix710104] +
  Fileset Names                                 [bos.alt_disk_install.rte]
```

Other tips and tricks
Migrating to AIX 7.1

Other considerations….

- Some ifixes left behind after migration from 5.3 to 7.1 (or 6.1):
http://www-01.ibm.com/support/docview.wss?uid=isg1IV23311&myns=apar&mynp=DOCTYPEcomponent&mync=E

- Certain device filesets left behind after the migration from 5.3 to 6.1. lppchk errors:
http://unix.ittoolbox.com/groups/technical-functional/ibm-aix-l/aix-53-to-61-migration-lppchk-v-shows-the-error-4363856

- maxuproc sys0 attribute set to default. 5.3 to 6.1 migration:
https://www.ibm.com/developerworks/mydeveloperworks/blogs/cgaix/entry/aix_6_1_migration_iostat_and_maxuproc_change_to_their_defaults?lang=en

- Latest AIX 7.1 installation tips:
http://www14.software.ibm.com/webapp/set2/subscriptions/onvdq?mode=18&ID=2496&myns=pwraix71&mync=E

- AIX 7.1 TL1 SP3 – netstat crash. IV09942

Other tips and tricks
Migrating to AIX 7.1

Other considerations…. - continued

- DB2 – intermittent crash – AIX 7.1 TL1 SP4. ifix IV22132.

http://www-304.ibm.com/support/docview.wss?uid=swg21165448

- Check /etc/tunables/nextboot for any “legacy” tuning.
- Not at recommended defaults e.g. Applying AIX 7.1 TL1 **SP4** with legacy values – system hang at “`Setting tunable parameters`”. SP3 booted OK!?

```
# mv /etc/tunables/nextboot /etc/tunables/nextboot.old
```

Review nextboot file after migration.

Other tips and tricks
Migrating to AIX 7.1

Other considerations…. - continued

- nimadm does NOT convert rootvg file systems from JFS to JFS2.
- I have requested this feature. You should too!
- Starting with 6.1 TL4**, alt_disk_copy** has new **–T** flag to convert to JFS2.
- One option is to use nimadm to migrate, then run **alt_disk_copy** with **–T**.
- Really only a problem for AIX 5.3 systems – **alt_disk_copy –T** not available.
- If you are on AIX 6.1 TL4 or higher, run **alt_disk_copy –T** first, then migrate to AIX 7.1.

https://www.ibm.com/developerworks/mydeveloperworks/blogs/cgaix/entry/convert_rootvg_file_systems_to_jfs2_using_alt_disk_copy?lang=en

.

Other tips and tricks
Migrating to AIX 7.1

Other considerations…. - continued

- Pre & Post migration scripts.

```
-a PreMigrationScript Specifies the pre-migration NIM script resource.
-z PostMigrationScript Specifies the post-migration NIM script resource.
```

- Use post migration script to update MPIO device filesets
- e.g. SDDPCM, HDLM, HDS ODM, etc.

```
# nimadm -j nimadmvg -c lparaix01 -s spotaix610605 -l
   lpp_sourceaix610605 -d hdisk2 -z XYZPOST –Y
```

https://www.ibm.com/developerworks/mydeveloperworks/blogs/cgaix/entry/using_a_post_migration_script_with_nimadm1?lang=en

.

AIX blogs and other online resources

- AIXpert.
https://www.ibm.com/developerworks/mydeveloperworks/blogs/aixpert/?lang=en

- AIXchange.
http://ibmsystemsmag.blogs.com/aixchange/

- Chris's AIX blog.
https://www.ibm.com/developerworks/mydeveloperworks/blogs/cgaix/?lang=en

- AIX Down Under.
https://www.ibm.com/developerworks/mydeveloperworks/blogs/AIXDownUnder/?lang=en

- AIX on IBM developerWorks.
http://www.ibm.com/developerworks/views/aix/libraryview.jsp?sort_by=&show_abstract=true&show_all=&search_flag=&contentarea_by=AIX+and+UNIX&search_by=chris+gibson&topic_by=-1&industry_by=-1&type_by=All+Types&ibm-search=Search

- AIX resources on Twitter:@cgibbo, @mr_nmon, @aixdownunder, @chromeaix

Before We Conclude

Migrating to AIX 7.1 with nimadm

ANY QUESTIONS

Corporate Office

IAG Insurance Australia Group