

Virtualization Standards for Business Continuity: Part 10

This is the tenth of a series of articles defining the policies, guidelines, standards, and procedures that provide the foundation of a virtualized environment, thus enabling business continuity, disaster recovery, and high availability, with an emphasis toward Return On Investment (ROI).

This article will discuss standardized procedures for installing the AIX operating system on client LPAR's, configuring the network etherchannel adapters, and prioritizing the storage communication on the client LPAR between the VIO servers.

The first task is to perform an AIX installation, this can be from distribution media, mksysb media, or from the NIM server (the preferred method). It is suggested that a standard build be created and a mksysb image be saved from this standard build. The using the NIM server, use this standard build mksysb to perform all AIX installations. Once installed, then use a configuration script to customize the standard build according to the individual system needs and requirements.

Assuming the LPAR to be installed was built using the "mklpar" script as described in a previous article in this series, and assuming the LPAR to be installed utilizes virtual I/O resources from dual VIO servers (VIOS) built using the "mkviolpar" script as described in a previous article in this series, the following procedures describe a configuration option for utilizing the virtual I/O resources.

The LPAR to be installed will have multiple resources being provided by the dual VIOS including four (4) "vscsi" adapters and a virtual ethernet adapter from each VIOS. Additional physical devices can be manually configure in the LPAR configuration, however this procedure only describes the configuration associated with the virtual devices.

<i>Adapter Type</i>	<i>Example Slot</i>	<i>Example Device Name</i>	<i>Purpose</i>
Virtual SCSI adapter	10	vscsi0	Operating System Disks
Virtual SCSI adapter	11	vscsi1	Operating System Disks
Virtual SCSI adapter	12	vscsi2	Data/Application Disks
Virtual SCSI adapter	13	vscsi3	Data/Application Disks
Virtual SCSI adapter	16	vscsi4	Data/Application Disks
Virtual SCSI adapter	17	vscsi5	Data/Application Disks
Virtual SCSI adapter	18	vscsi6	Miscellaneous Disks
Virtual SCSI adapter	19	vscsi7	Miscellaneous Disks

Table 10.1: Virtual SCSI adapters on client LPAR

<i>Adapter Type</i>	<i>Example Slot</i>	<i>Example Device Name</i>	<i>Purpose</i>
Virtual Ethernet adapter	500	eth0	Data/Application Network
Virtual Ethernet adapter	501	eth1	Data/Application Network

Virtualization Standards for Business Continuity: Part 10

<i>Adapter Type</i>	<i>Example Slot</i>	<i>Example Device Name</i>	<i>Purpose</i>
Virtual Ethernet adapter	850	eth2	Management/Backup Network
Virtual Ethernet adapter	851	eth3	Management/Backup Network

Table 10.2: Virtual Ethernet adapters on client LPAR

The recommended configuration of the ethernet network adapters on the client LPAR is a standardized method that will work in stand-alone, high availability, or disaster recovery scenarios. This configuration consolidates the multiple ethernet adapters from dual VIOS into a single etherchannel adapter, then layers multiple IP addresses onto that single adapter. The IP addresses include boot, persistent, hostname, and resource group addresses. The boot addresses should be non-routed addresses and should be in separate subnets. The persistent, hostname, and resource group addresses should be routed addresses and may all be in the same subnet. The following procedure uses example addresses in a shell script oriented mechanism for configuration:

Begin by defining the application network default gateway address, and the management network default gateway address, the network masks to use, DNS name server address, and the DNS domain name:

```
APPGATE="10.1.1.254"  
MANGATE="10.1.2.254"  
NETMASK="255.255.255.0"  
NAMESRV="10.1.1.10"  
DOMNAME="mtxia.com"
```

Next define shell variables to contain the hostname of the client LPAR:

```
HNAME=$( hostname )  
HNAME="{HNAME%*.}"  
hostname ${HNAME}  
uname -S ${HNAME}  
chdev -l inet0 -a hostname=${HNAME}
```

Define shell variables to contain the boot IP addresses for both etherchannel adapters to be created. Notice they are in separate subnets, assuming a subnet mask of "255.255.255.0"

```
BT01IP="192.168.171.6"  
BT01IP="192.168.172.6"
```

Next define a shell variable to contain the persistent IP address for the client LPAR. This is a routable address that will always be available on the node regardless of the "state" of the operating system, "state" referring to standalone, primary or fail over node of an HACMP cluster, disaster recovery node, etc. This IP address is referred to as the "persistent" address.

```
PERSIP="10.1.1.101"
```

Now define a shell variable that will contain the IP address that will be associated with the node name

Virtualization Standards for Business Continuity: Part 10

or default host name of the client LPAR. This address may or may not be available at boot time, depending upon the state of the operating system, and it may be in the same subnet as the persistent and resource group addresses:

```
NODEIP="10.1.1.102"
```

The next IP address shell variable will contain the resource group address or service address. This address through which all application, data, and users access to the system will occur, as such, it will be routed. It may be in the same subnet as the persistent and node name addresses:

```
RGRPIP="10.1.1.103"
```

The last IP address needed for this standardized configuration is the management address. This address will be used for system administration and management of the system. It should be a routed address in a separate subnet from the resource group (service) address.

```
MGMTIP="10.1.2.101"
```

When implementing this network configuration, it is usually easier to remove all ethernet devices and start from the beginning, rather than attempting to reconfigure the existing networks. The following Korn shell code brings down the existing ethernet interfaces and detaches them from the adapters. Since all network access will be removed during this procedure, the system administrator must be connected to the system via a console or Hardware Management Console (HMC) terminal:

```
# Unconfigure all ethernet adapters
for i in 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
do
  ifconfig en${i} down
  ifconfig en${i} detach
  ifconfig et${i} down
  ifconfig et${i} detach
  rmdev -Rdl ent${i}
  rmdev -Rdl en${i}
  rmdev -Rdl et${i}
done
ifconfig -a
```

Now rediscover the ethernet devices via “cfgmgr” and turn off the “chksum_offload” parameter:

```
cfgmgr
for i in 10 9 8 7 6 5 4 3 2 1 0
do
  chdev -l ent${i} -a chksum_offload=no
  chdev -l ent${i} -a chksum_offload=no -P
done
```

In preparation for rebuilding the etherchannel devices, the traffic should be distributed evenly between the dual VIO servers providing the virtual ethernet adapters. This task can be automated, assuming the host naming standards in the first article of this series were followed. This standard dictates that all host names end with an even or odd number, which will be used to determine which ethernet adapter is

Virtualization Standards for Business Continuity: Part 10

used as the primary communication channel, and which is used as the backup. Even numbered hosts will use the even numbered virtual ethernet adapter as it's primary network path, odd numbered hosts will use the odd numbered virtual ethernet adapter as it's primary network path.

For Even numbered hosts, set the following shell variables, for odd numbered hosts, reverse the adapter number order:

```
NETADDR="${APPGATE}"
MC2ADDR="${MANGATE}"
ZERO=0
ONE=1
TWO=2
THREE=3
if [[ "$(hostname)" == *[13579] ]]
then
    ZERO=1
    ONE=0
    TWO=3
    THREE=2
fi
```

Create the etherchannel devices for both the resource group network and the management network, based on the shell variables previously configured:

```
if mkdev -c adapter -s pseudo -t ibm_ech \
    -a adapter_names="ent${ZERO}" \
    -a backup_adapter="ent${ONE}" \
    -a netaddr="${RGPADDR}"
then
    /usr/lib/methods/defif
fi

if mkdev -c adapter -s pseudo -t ibm_ech \
    -a adapter_names="ent${TWO}" \
    -a backup_adapter="ent${THREE}" \
    -a netaddr="${NETADDR}"
then
    /usr/lib/methods/defif
fi
```

Once the etherchannel devices are created, IP addresses can be assigned. Assuming the VSCSI slot numbering standards were followed, the first etherchannel adapter will exist as “ent4” and will be used for access to the application/user network. The second etherchannel adapter will exist as “ent5” and will be used for management network access. The first IP address to be assigned to each etherchannel adapter will be the boot address:

```
/usr/sbin/mktcpip -h ${HNAME} \
-a ${BT01IP} \
-m ${NETMASK} \
-i en4 \
-n ${NAMESRV} \
-d ${DOMNAME} \
```

Virtualization Standards for Business Continuity: Part 10

```
-g ${APPGATE} \  
-A no \  
-t 'N/A'  
  
/usr/sbin/mktcpip -h ${HNAME} \  
-a ${BT02IP} \  
-m ${NETMASK} \  
-i en5 \  
-n ${NAMESRV} \  
-d ${DOMNAME} \  
-g ${APPGATE} \  
-A no \  
-t 'N/A'
```

Of the remaining IP addresses to be configured, the “persistent”, “node”, and “resource group” addresses will be aliased onto the application network etherchannel. The “management” IP address will be aliased onto the management network etherchannel:

```
/etc/ifconfig en4 ${PERSIP} netmask ${NETMASK} alias  
/etc/ifconfig en4 ${NODEIP} netmask ${NETMASK} alias  
/etc/ifconfig en4 ${RGRPIP} netmask ${NETMASK} alias  
/etc/ifconfig en5 ${MGMTIP} netmask ${NETMASK} alias
```

If the client LPAR is to be used as a node in an HACMP cluster, the “persistent”, “node”, “resource group”, and “management” addresses will be implemented as part of the HACMP configuration. If the client LPAR is a standalone node, the commands to alias these IP addresses onto each etherchannel adapter must be added to the “/etc/rc.net” file to be activated at system boot time.

The default route should be configured to use the application/user network gateway:

```
route add default ${APPGATE}  
chdev -l inet0 -P -a route="net,-hopcount,0,,0,${APPGATE}"  
mkdev -l inet0
```

Finally, add the IP addresses and host names to the “/etc/hosts” file. This completes the ethernet configuration using the standardized procedures provided by the business continuity methodology described by this series of articles. The virtualized storage presented to the client LPAR by the VIO servers also requires configuration to fulfill the methodologies discussed.

The disk configuration described by this series of articles assumes that each virtualized disk has multiple paths to the storage SAN, one path from each VIO server. The default configuration of the Multi-Path I/O (MPIO) is to give all paths the same priority, this has the effect of directing all SAN traffic across the first VIO server. In order to distribute the storage communication traffic evenly across both VIO servers, each path in an MPIO configuration must be assigned a path priority. The goal is to distribute the load evenly across the VIO servers, however it is undesirable for all “hdisk0” disks to have their highest priority path always point to the even numbered VIO server. Therefore, the following logic should be implemented when assigning path priorities:

- Even numbered disk + even numbered client LPAR host name = highest priority path is the even numbered VIO server

Virtualization Standards for Business Continuity: Part 10

- Odd numbered disk + odd numbered client LPAR host name = highest priority path is the even numbered VIO server
- Even numbered disk + odd numbered client LPAR host name = highest priority path is the odd numbered VIO server
- Odd numbered disk + even numbered client LPAR host name = highest priority path is the odd numbered VIO server

This path priority logic is captured in a previously presented shell script called “vscsiPriority.ksh” and was published in the February 2006 issue of “AIX Update Journal”. This shell script automatically prioritizes all disks on a client LPAR.

The next task is to configure the virtual storage assigned to the client LPAR, illustrating the VSCSI slot numbering standards and why they are useful. The VSCSI slot numbers associated with each client LPAR is a known entity on the VIO server, however on the client LPAR it is sufficient to recognize the only important part of the slot number is the last digit. Slots that end with a zero (0) or one (1) are used for operating system disks. Slots that end with a two (2) or three (3) are used for application and data disks. The slot ending in a two (2) is served by the VIO server whose host name is even numbered, the slot ending in a three (3) is served by the VIO server whose host name is odd numbered. Therefore, on the client LPAR all disks whose parent VSCSI adapter has a slot number ending in a two (2) or three (3), can be automatically detected and assembled into a volume group. The disks within a volume group can then be divided into logical volumes and file systems. The following Korn shell statement utilizes the output from the “lscfg” command to obtain a list of disks associated with the VSCSI adapter at slot numbers ending in two(2) or three(3):

```
VGDISKS=$( lscfg -l 'hdisk*' | egrep -- '-C[0-9]*[23]-' | awk '{ print $1 }' | sort -n )
```

The following Korn shell code defines several values that will be used during the automated virtual storage configuration.

```
# Determine the length of the host name
LEN="#{#HNAME}"
# extract the last two characters of the host name, assumed to be a two digit number
SEQNBR=${HNAME:LEN-2:2}
# assign a resource group name which will be used to define the VG, LV's, and file systems.
RG="mtxdev${SEQNBR}"
# Assign the volume group major number
VGMJ="8${SEQNBR}"
# Assign a unique identifier to use during the creation of the volume group
VGID="00vg"
# Assign a unique identifier to use during the creation of the Log Logical Volume
LGID="jfs1"
# Assign a directory mount point for the file system using the resource group name
MTPT="/${RG}"

# Create the volume group using the previously defined values
mkvg -f -y ${RG}${VGID} -V ${VGMJ} ${VGDISKS}
# Create the log logical volume using the previously defined values
```

Virtualization Standards for Business Continuity: Part 10

```
/usr/sbin/mklv -y ${RG}${LGID}lv -t jfs2log -a e "${RG}${VGID}" 1

# Determine the number of free physical partitions associated with the volume group
FREEPPS=$( print "a=0; $( lsvg -p ${RG}${VGID} | sed -e '1,2 d' | awk '{ print $4 }' | sed -e 's/^/a+/' ); a" | bc )

# Assign the number of physical partitions to use for the application/data logical volume.
LVSIZE=$(( FREEPPS - 5 ))

# Create the application/data logical volume
/usr/sbin/mklv -y ${RG}${LVID}lv \
-t jfs2 \
-x 5000 \
-a e \
"${RG}${VGID}" \
${LVSIZE}

# Create the application/data file system
/usr/sbin/crfs -v jfs2 \
-d "${RG}${LVID}lv" \
-m "${MTPT}" \
-A y \
-p rw \
-a agblksize=4096 \
-a logname="${RG}${LGID}lv"

# Mount the newly created file system
mount /${RG}
```

The result of the above commands will be a fully configured file system mounted on a directory identified by the resource group name. The point to all of the shell script commands shown in this article is to re-enforce the business continuity mentality associated with standardized procedures for all aspects of system administration, including the build-out of networking and storage. Standardized procedures such as these lead quickly into process automation, and data center automation.

With the automated build-out of networking and storage, comes the ability to consider other components for process automation such as application deployment, database deployment, workload manager configuration, high availability (HACMP) implementation, disaster recovery implementation, automated documentation, audit compliance, and audit response.

The Business Continuity policies, guidelines, standards, and procedures to be learned from this article are as follows:

Policies: *Those tasks that must be implemented*

- Each LPAR will be accessible through two ethernet networks, application and management.
- Volume group names shall be enterprise wide unique values based on the resource group name.
- Each logical volume shall have an enterprise wide unique name for the purpose of eliminating naming conflicts during manual, HACMP, or Disaster Recovery fail overs.
- Each volume group that contains file systems will require at least one JFS log. The name of the

Virtualization Standards for Business Continuity: Part 10

JFS Log shall be an enterprise wide unique value.

- Every application file system shall have an enterprise wide unique mount point directory. The purpose is to eliminate conflicts during manual, HACMP, disaster recovery, or consolidation fail overs.

Guidelines: *Those tasks that should be implemented*

- Physical I/O Adapters in production environments may be dedicated to an LPAR based on bandwidth requirements.
- Where possible, create volume groups with a unique major number.
- More than one JFS log may be configured for a volume group if necessary, dependent upon file system load.
- In order to ensure successful installation of an application into unique mount points, the application team must work closely with the system engineer.

Standards: *Technical specifications derived from the policies and guidelines*

- The resource group naming standard was presented in part 7 of this series of articles, November 2007, AIX Update Journal.
- The volume group, logical volume, log logical volume, and directory mount point naming standards were presented in the November 2005 issue of AIX Update Journal.
- A standard for prioritizing disk communication paths between dual VIO servers is presented in this article.
- A standard for distributing the ethernet network traffic between dual VIO servers is presented in this article.

Procedures: *Step-by-step implementation instructions of the standards*

- Procedures for automatic identification of application and user storage disks is presented in this article.
- Procedures for automatic creation and configuration of etherchannels is presented in this article.
- Procedures for automatic creation and configuration of volume groups, logical volumes, log logical volumes, and directory mount points is presented in this article.
- A shell script procedure for prioritizing the communication paths to storage disks was discussed, and was published in the February 2006 issue of AIX Update Journal.

The next article in this series will discuss several methods for tracking, monitoring, and supporting the physical and virtual resources across dual VIO servers and client LPARs.

Dana French
President, Mt Xia, Inc.
<http://www.mtxia.com>
615.556.0456