

Naming standards for business continuity

Best practices and definitions

Skill Level: Intermediate

[Mr. Dana L. French \(dfrench@mtxia.com\)](mailto:dfrench@mtxia.com)

President
Mt Xia Inc.

25 Nov 2008

In today's partitioned and virtualized computing environments, the requirement for naming standards is more apparent than ever before. Unfortunately, there is a widespread lack of understanding between the different categories of names used to identify managed systems, LPAR's, profiles, node names, host names, and aliases. This article defines naming categories and provide standards for creating enterprise-wide unique names associated with each category. This article also explains why it is important to distinguish between the different naming categories and how they are an integral part of data center automation projects and an overall business continuity plan.

Introduction

Computer naming standards for systems, nodes, hosts, and aliases are an integral part of implementing a successful business continuity environment. These standards must provide a mechanism for defining "enterprise-wide unique" identifiers for all computer naming requirements within an organization.

Considerations for computer naming standards

Naming standards should be:

- Consistent and result in repeatable procedures.
- Compatible with standalone, high availability, disaster recovery, business

continuity, partitioned, and virtualized environments.

- Used with disaster-recovery procedures to eliminate resource conflicts.
- Used with high-availability procedures to eliminate resource conflicts.
- Used with storage path prioritization to balance SAN traffic loads.
- Used with etherchannel adapter configuration to balance network traffic loads.
- Used with host ethernet adapter (HEA) configuration to balance network traffic loads.
- Used to coordinate and balance I/O through VIO servers.
- Used to distinguish and define partition, profile, node, and host names.

In order to explain the significance of computer naming standards in business continuity, it is necessary to define the term "business continuity" as it is used in this article. A definition for disaster recovery, as it is used in this article, is also provided to distinguish it from business continuity. Both definitions below were taken verbatim from the [Mt Xia: Technical Consulting Group](#) definitions page.

Business Continuity (BC)

Consists of the activities performed on a daily basis to ensure the business operates normally today, tomorrow, and beyond. Business continuity is sometimes confused with disaster recovery, but they are separate entities. Disaster recovery is a small subset of business continuity. The business continuity plan may be thought of as a methodology, or as an enterprise wide mentality of conducting day-to-day business.

Disaster Recovery (DR)

The implementation of a project plan which describes the tasks necessary to recover critical business functions after a disaster. The recovery occurs between geographically separated data centers using one or more methods of storage replication between the data centers.

The current trend in system design is to provide a separate system for each application or instance of an application, such as a database, web application, or financial package. In the past a computer system was contained within a single piece of physical hardware, and consisted of the box, CPU, memory, boards, adapters, disks, etc. The current IBM hardware platforms permit the system administrator to segment a physical managed system into multiple systems called Logical Partitions (LPAR's). These LPAR's can assign all or part of the hardware resources contained in the managed system, such as CPU, memory, and I/O adapters. Each LPAR may be used to host any application that normally runs on a standalone machine, high availability cluster, or in a disaster recovery environment.

Naming standards for business continuity

Implementing these types of environments requires a naming structure that accounts for more than just a single host name on a system. A standard must be adopted that is extensible into any environment deemed necessary. This article describes some of the requirements that must be considered when designing a naming standard for a modern partitioned and virtualized business continuity environment.

Managed system names

The term "Managed System" refers to a physical piece of hardware containing components such as CPU's, memory, I/O adapters, storage disks, and more. This is also sometimes called a "Frame" or a "System," and may be logically divided into numerous partitions. In the IBM® System p® world, the managed system or frame does not have a user-accessible network address, although the logical partitions (LPAR) almost always have one or more network addresses. The managed system name is used by the Hardware Management Console (HMC) to identify a physical container of hardware and the name should be an enterprise-wide unique identifier. By default, when the HMC discovers a new piece of hardware, it automatically generates and assigns a managed system name that is composed of the hardware model number, model type, and serial number. This managed system name is guaranteed to be enterprise-wide unique since it contains the serial number of the physical machine. Some example automatically generated managed system names are shown in Table 1, below.

Table 1. Example managed system names

Model Number	Model Type	Serial Number	Frame or Managed System Name
9119	590	12A345B	Server-9119-590-SN12A345B
9117	MMA	12C678D	Server-9117-MMA-SN12C678D
9133	55A	066ABCD	Server-9133-55A-SN066ABCD

These automatically generated managed system names should be adopted as the standard, as they are compatible with available data center automation software; however, if your organization decides to rename them, here are a few guidelines to follow:

- Use alpha-numeric characters only -- no spaces, underscores, periods, commas, or other punctuation marks. Since the automatically generated naming structure includes dashes, these could be included in the new system name if desired.
- Use enterprise-wide unique names, meaning there are no other entities in

the organization with the same name. This means that not just other managed systems, but no other partitions, profiles, nodes, hosts, or aliases have the same name. There should be no confusion about what the name refers to, and it should be instantly recognizable as a managed system name.

- Use a naming structure format that is unique to managed system names, thereby further delineating the name from other entities in the organization.

Logical Partition or LPAR names

Within a managed system is the capability of dividing the physical resources into groups called Logical Partitions or LPAR's. The physical resources consist of things such as CPU's, memory, I/O adapters, storage disks, and more. Each managed system may be divided into one or more LPAR's, each one requiring a name. Experience with data center automation techniques has shown these LPAR names should correspond with the node name used with each instance of an operating system and should be an enterprise-wide unique LPAR name. See the [Node names](#) section for further details.

Partition profile names

The partition profile is a definition of physical and virtual resources associated with an LPAR, and an LPAR can have one or more profiles. Each profile must have a name and this name should be an enterprise-wide unique profile name. As with the LPAR name, experience with data center automation techniques has shown the default profile name should correspond with the node name and be an enterprise-wide unique profile name. More than one profile can be defined for each LPAR, and the name of each additional profile should be a variation on the default profile name. See the [Node names](#) section for further details.

Node names

The node name is associated with an instance of an operating system and should not be confused with a TCP/IP network host name. A host name is associated with a network adapter. An operating system (OS) instance may have many different host names, but only one node name. Also, the node name always remains with an instance of an operating system, whereas a host name may float between adapters within a OS instance, or between nodes, or across data centers. All partition, host, and node names should be enterprise-wide unique values in order to eliminate conflicts during fail-overs, whether planned, unplanned, manual, automated, or part of a disaster-recovery effort. In order to accommodate data center automation applications, software, and techniques, node naming standards are provided and

suggested in Table 2, although the coordination of naming standards between these software entities is beyond the scope of this article.

Table 2. Node name structure

Location Code	+	OS Type	+	Environment	+	Application Code	+	Sequence ID
3 char	+	1 char	+	1 char	+	3 char	+	2 char

The node name should contain alpha-numeric characters only and be consistent across all platforms in an organization. Example information for details of each component of the node name standard is shown in Table 3, below.

Table 3. Node name components

Node Name Component	Number of Characters	Example Values
Location Code	3	bos = Boston dal = Dallas phx = Phoenix
OS Type	1	a = AIX l = Linux o = OS/400 v = VIO Server w = Microsoft® Windows®
Environment	1	a = Acceptance Testing d = Development p = Production t = Testing x = Disaster Recovery
Application Code	3	vio = VIO Server nim = NIM Server sap = SAP mqs = MQ Series ora = Oracle db2 = DB2 ifx = Informix
Sequence ID	2	A two-character identifier to distinguish multiple instances of a node type. This two-character identifier may contain the following characters: 0-9, A-Z, a-z

As an example, assume an AIX® node exists in the Boston data center location, which is a production system running Oracle, and it is the first node in the sequence. The example node name would consist of the components shown in Table 4.

Table 4. Example node name components

Location Code	+	OS Type	+	Environment	+	Application Code	+	Sequence ID
bos	+	a	+	p	+	ora	+	01

Host names

A host name is a reference to an IP address, and an IP address is associated with one or more network adapters. It is important to recognize that an IP address is not necessarily permanently tied to a network adapter, but may float across adapters, nodes, and data centers. The same is true with the host names. A host name should be viewed as being independent from any node, managed system, or data center. The host name must be an enterprise-wide unique identifier in order to eliminate conflicts during manual, automated, or disaster recovery fail-overs.

For any single node, one or more host names may be created to identify all of the various network interfaces. Normally, each node has a host name that is identical to the node name. Using the example of "bosapora01" as depicted in [Table 4](#), this node name would also be used as a host name with an IP address assigned to it. The point being illustrated here is the node name and the host name of a system are separate entities and should be thought of in that way when designing standalone, virtualized, and clustered systems.

Additional host names

The best solution for avoiding networking conflicts during a disaster recovery implementation is to always ensure that each network (TCP/IP) address or name is an enterprise-wide unique value. In organizations with multiple active data centers, network (TCP/IP) addresses from the production data center should not be failed-over to the DR site. To do so requires reconfiguration of routers and switches and could endanger the existing production systems running in the data center accepting the disaster recovery workload. Therefore, the production applications should never be tied to or dependent upon a specific TCP/IP network address, because in an actual disaster those addresses will change and the applications will not work. Additionally, applications and users (non-administrators) should never use or specify a network service by its TCP/IP address; they should only use a symbolic name. Furthermore, the symbolic name used by applications and regular users should only be an "alias" pointing to a host name.

In this context, a "node" refers to an instance of an operating system, whether or not it is part of a cluster or a standalone system, and the node name is a separate entity from the host name. The node name structure should consist of alpha-numeric characters only. One or more host names can be derived based on the node name of the system. Table 5 illustrates a node named "bosapora00" with 5 host names.

Table 5. Node and associated host names

Node Name	Host Name	Description
bosapora00	bosapora00-bt01	refers to IP address assigned to a network adapter at boot time
bosapora00	bosapora00-pr01	refers to a persistent IP address

		assigned to a network adapter that will always be present.
bosapora00	bosapora00-rg01	refers to a service IP address assigned to a network adapter that is available when the application services are running.
bosapora00	bosapora00-mt01	refers to a system management IP address assigned to a network adapter that is always present.
bosapora00	bosapora00	refers to a persistent IP address assigned to a network adapter that is always available on a node.

[Table 5](#) also shows a host name called "bosapora00," which happens to correspond with the node name of the system. Recognize that even though these names are the same, their purposes are different. This is an instance where the node name will not be an enterprise-wide unique value; each node name will correspond to a host name with the same value.

Aliases

The network name referenced by applications and regular users should not be any of the host names referenced in [Table 5](#); they should only use an alias name to these host names as shown in [Table 6](#).

Table 6. Host names and aliases

Host Name	Alias
bosapora00-rg01	myappl5
bosapdb201-rg22	db2sys
bosapmq503-rg05	mqseries5

Notice in [Table 6](#) that all host names have an extension of "-rg##." These are references to an entity called a "resource group address name" or "service address name." Even though this is a cluster concept, it should be implemented on all nodes: standalone, virtualized, or clustered. The service address name is used to refer to a network application or service. Any applications or regular users requiring access to a network application service should only refer to the alias name, which redirects them to the service address host name associated with the service IP address.

In the event of a disaster, the production applications are restarted at the DR site on systems with different TCP/IP addresses, different node names, and different host

names. To provide access to the applications now restarted at the DR site, the only change that is necessary is to re-point the alias names in the DNS to the new RG host name in the DR site. The applications and regular users do not need to make any changes and are automatically rerouted to the correct location and application server.

The rules for defining alias names are significantly less rigid than for host names, but should still be an enterprise-wide unique identifier. The alias can be any name as long as it is unique within the domain. This allows the application to be accessed through a name that makes logical sense to the user. For example, the production Oracle Application Server at the Boston Data Center may have a host name of "bosapora01"; however, the alias referenced by all users may be "myappl5." The use of aliases preserves the structure needed for host names and the ease of use desired by users.

Distribution of communication traffic

One technique for evenly distributing communication traffic across dual networks, switches, and VIO servers is to automatically select a primary and secondary path based on the sequence ID number associated with the node name. For example, if the sequence ID number of the node name is an even number, the even-numbered storage communication adapter may be selected as the primary path, and the odd-numbered selected as the secondary path. However, this technique assumes multiple communication adapters on each node, and equal quantities of even- and odd-numbered nodes are configured on each managed system. This means that all even-numbered nodes are not configured on one managed system, and all odd-numbered nodes are not configured on another managed system.

As an example of this concept, [Table 7](#) shows the primary communication path for the even-numbered hdisks on the even-numbered client LPAR's is through an even-numbered VIO server (actually, it is through the even-numbered virtual SCSI adapter; for now assume the even-numbered virtual SCSI adapters are associated with the even-numbered VIO server). The secondary path for the even-numbered hdisks on the even-numbered client LPAR's is through the odd-numbered VIO server. The logic is, of course, reversed for odd-numbered hdisks.

Using path prioritization scripts and extrapolating this node naming standard to dozens of LPAR's on a p590-managed system, it becomes apparent that LPAR's with even-numbered node names, "hdisk0" always have a primary communication path through the even-numbered VIO server. Since "hdisk0" usually contains the "rootvg" volume group, it is NOT desirable to have the primary path of "hdisk0" for all LPAR's on a managed system going through the same VIO server. Therefore, it is recommended when configuring LPAR's between two managed systems (such as with HACMP), do not use all even-numbered node names for the LPAR's on one managed system, and all odd-numbered node names on the other managed system.

Otherwise, the result of using path prioritization scripts would be that all LPAR's with even-numbered node names would use the VIO server with the even-numbered node name as the primary path for "hdisk0." Additionally, all LPAR's with odd-numbered node names would use the VIO server with the odd-numbered node name as the primary path for "hdisk0," which is undesirable, as shown in Table 7.

Table 7. Undesirable storage communication paths

Managed System Name	Client LPAR Node Name	Even Numbered hdisks Primary VIOS	Odd Numbered hdisks Primary VIOS
Server-9119-590-SN12A345B	bosapora00	bosapvio00	bosapvio01
	bosapora02	bosapvio00	bosapvio01
	bosapora04	bosapvio00	bosapvio01
	bosapora06	bosapvio00	bosapvio01
Server-9119-590-SN67D809E	bosapora01	bosapvio03	bosapvio02
	bosapora03	bosapvio03	bosapvio02
	bosapora05	bosapvio03	bosapvio02
	bosapora07	bosapvio03	bosapvio02

A more desirable configuration is to evenly distribute the "hdisk0" traffic across the dual VIO servers, which can be easily automated if both even- and odd-numbered node names are equally used on each managed system. Table 8 shows a desirable node name configuration for a group of eight Oracle servers across two p590-managed systems and the distribution of "hdisk0" traffic across dual VIO servers. The primary and secondary paths of all subsequently numbered hdisks are also distributed evenly, as previously discussed.

Table 8. Desirable storage communication paths

Managed System Name	Client LPAR Node Name	Even Numbered hdisks Primary VIOS	Odd Numbered hdisks Primary VIOS
Server-9119-590-SN12A345B	bosapora00	bosapvio00	bosapvio01
	bosapora01	bosapvio01	bosapvio00
	bosapora02	bosapvio00	bosapvio01
	bosapora03	bosapvio01	bosapvio00
Server-9119-590-SN67D809E	bosapora04	bosapvio02	bosapvio03
	bosapora05	bosapvio03	bosapvio02
	bosapora06	bosapvio02	bosapvio03

bosapora07

bosapvio03

bosapvio02

Of course, the distribution of storage and network traffic can be manually configured across the dual VIO servers; however, this takes a lot of time and effort. Also, the path-prioritization scripts designed to perform this task can be configured to reverse the logic of how the traffic is distributed, so the administrator can specify the primary and secondary paths, but this means the administrator must keep track of how each LPAR is configured to make a determination of how to configure new LPAR's. It is much easier and more efficient to implement a node naming structure that can be used to automate at least a portion of this configuration process, and relieve the administrator from having to monitor and track this information.

Conclusions

Users and applications should only use alias names to refer to network application services and should never refer directly to host names or TCP/IP network addresses. Aliases used for network application access should only point to host names specifically implemented as service names for applications. Each network adapter should be configured with a boot address and associated host name. Additional TCP/IP addresses are configured on each network adapter as necessary and assigned a corresponding host name. Each operating system instance is assigned a node name. This name is used as the partition and default profile name, as well. By using the node name as the LPAR and default profile name, it enables automation scripts to easily identify the LPAR and profile for manipulation and modification. Obtaining a list of node names on a managed system becomes as easy as generating a list of LPAR's from the HMC. And since each node should be configured with a host name that corresponds to the node name, network access is simplified, as well.

Resources

Learn

- See the [Mt Xia: Technical Consulting Group](#) definitions page for definitions of terms used in this article.
- The [Path Prioritization Script](#) assigns a priority to vscsi paths based on even/odd numbers associated with each disk and each path to disk.
- [Enterprise Wide Unique identifier generator](#) generates an Enterprise Wide Unique UID number for any given user name.
- [AIX Disaster Recovery: Resolving Resource Conflicts](#) (developerWorks, September 2007) identifies resource conflicts that typically occur during a disaster recovery implementation and provides suggestions for resolving these conflicts.
- [The AIX and UNIX developerWorks zone](#) provides a wealth of information relating to all aspects of IBM® AIX® systems administration and expanding your UNIX skills.
- [New to AIX and UNIX?](#) Visit the New to AIX and UNIX page to learn more.
- [developerWorks technical events and webcasts](#): Stay current with developerWorks technical events and webcasts.
- [Podcasts](#): Tune in and catch up with IBM technical experts.

Get products and technologies

- [IBM trial software](#): Build your next development project with software for download directly from developerWorks.

Discuss

- Participate in the AIX and UNIX forums:
 - [AIX Forum](#)
 - [AIX Forum for developers](#)
 - [Cluster Systems Management](#)
 - [IBM Support Assistant Forum](#)
 - [Performance Tools Forum](#)
 - [Virtualization Forum](#)
 - [More AIX and UNIX Forums](#)
 - [AIX Networking](#)

About the author

Mr. Dana L. French

Mr. French's career in the IT industry has spanned three decades and numerous industries. His work focuses primarily on the fields of business continuity, disaster recovery, and high availability, and he has designed and written numerous software packages to automate the processes of business continuity, disaster recovery and high availability. He is most noted for his approach to system administration as an automated, business oriented process, rather than as a system oriented, interactive process. He is also a noted authority on the subject of Korn Shell programming.