The most popular innovation of IBM® AIX® Version 6.1 is clearly workload partitioning (WPARs). Once you get past the marketing hype, you'll need to determine the value that WPARs can provide in your environment. What can WPARs do that Logical Partitions (LPARs) could not? How and when should you use WPARs? Equally as important, when should you not use Workload Partitioning. Finally, how do you create, configure, and administer workload partitions? These topics will all be discussed in this article.

Introduction

WPARs are a bold new innovation, implemented within AIX 6.1. It allows administrators to virtualize their operating system, which allows for fewer operating system images on your IBM System p™ partitioned server. Prior to WPARs, you would need to create a new Logical Partition (LPAR) for each new "isolated" environment. This is no longer necessary (with AIX 6.1 only), as there are many circumstances when one can get along fine with multiple WPARs within one LPAR. Why is this important? Every LPAR requires its own operating system image and a certain number of physical resources. While you can virtualize many of these resources, there are still some physical resources that must be allocated to the system. Furthermore, you need to install patches and technology upgrades to each LPAR. Each LPAR requires its own archiving strategy and DR strategy. It also takes some time to create an LPAR; you also need to do this outside of AIX, through a Hardware Management Console (HMC) or the Integrated Virtualization Manager.
WPARs are much simpler to manage and can actually be created from the AIX command line or through SMIT. LPARs cannot. By far the biggest disadvantage of LPARs is maintaining multiple images, which goes along with possibly over-committing expensive hardware resources, such as CPU and RAM. In other words, while partitioning helps you consolidate and virtualize hardware within a single box, operating system virtualization through WPAR technology goes one step further and allows for an even more granular approach of resource management. It does this by sharing OS images and is clearly the most efficient use of CPU, RAM, and I/O resources.

Rather than a replacement for LPARs, WPARs are a complement to them and allow one to further virtualize application workloads through operating system virtualization. WPARs allow for new applications to be deployed much more quickly, which is an important side-benefit. On the other side of the coin, it's important to understand the limitations of WPARs. For example, each LPAR is a single point of failure for all WPARs that are created within the LPAR. In the event of an LPAR problem (or a scheduled system outage, for that matter), all underlying WPARs will also be affected.

**WPARs: How and when to use them**

This section further defines the different types of workload partitions and discusses scenarios where WPARs should be used.

As discussed earlier, Workload Partitions (WPARs) are virtualized operating system environments that are created within a single AIX (only supported on AIX 6.1) image. While they may be self-contained in the sense that each WPAR has its own private execution environment with its own filesystems and network addresses, they still run inside the global environment. The global environment -- the actual LPAR -- owns all the physical resources of the logical partition. It is important to also note that the global environment can see all the processes running inside the specific WPARs.

There are two types of WPARs: system workload partitions and application workload partitions. The system WPAR is much closer to a complete version of AIX. The system WPAR has its own dedicated, completely writable filesystems along with its own inetd and cron. Application WPARs are real, lightweight versions of virtualized OS environments. They are extremely limited and can only run application processes, not system daemons such as inetd or cron. One cannot even define remote access to this environment. These are only temporarily objects; they actually disintegrate when the final process of the application partition ends, and as such, are more geared to execute processes than entire applications. Overall, WPARs have no real dependency on hardware and can even be used on POWER4 systems that do not support IBM's PowerVM (formerly known as APV). For AIX administrators,
the huge advantage of WPARs is the flexibility of creating new environments without having to create and manage new AIX partitions. Let's look at some scenarios that call for the use of WPARs.

**Application/workload isolation**

WPARs are tailor-made for working with test and/or QA and development environments. Most larger organizations have at least three environments for their applications. These include development, test, and production. Some environments have as many as five, including demo/training and stress/integration environments. Let's use an example of a common three-tier application environment: Web, application server, and database server. In the land of the LPARs, in an environment where one has five isolated environments, you would need to create 15 LPARs. This is where the WPAR has the most value. In this environment, we would need to create just five LPARs. How is that?

In Table 1, we have five different environments, consisting of a Web server, an application server, and a database server. If we wanted to isolate our environments, the only way to do this would be through logical partitioning. That would involve architecting 15 logical partitions. Of course, we could run some of our Web, application, and database on one LPAR, but if we did that, how would we be able to really mimic our production environments (which would run on separate partitions)? In today's world of 99.9% availability, it is extremely common to give each application environment its own home. With WPARs, we can now do that, without having separate AIX images.

**Table 1. Web portal -- LPARs only**

<table>
<thead>
<tr>
<th>Environment</th>
<th>LPARs</th>
<th>WPARs</th>
</tr>
</thead>
<tbody>
<tr>
<td>Development</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>Demo/Train</td>
<td>4</td>
<td>3</td>
</tr>
<tr>
<td>Test</td>
<td>7</td>
<td>3</td>
</tr>
<tr>
<td>Pre-Prod</td>
<td>10</td>
<td>3</td>
</tr>
<tr>
<td>Production</td>
<td>13</td>
<td>3</td>
</tr>
</tbody>
</table>

Table 2 illustrates how that is done. Each environment would have its own LPAR, with three WPARs created within each LPAR. Now let's imagine if we had four Web servers, two application servers, and two database servers supporting this environment. Yikes! AIX administrators supporting Fortune 500 companies know what I'm talking about. It can be a nightmare maintaining all these environments. WPARs dramatically simplify the overall work-effort involved in administrating this environment, while at the same time minimizing the expense of having to assign physical resources to logical partitions.

**Table 2. Web portal -- WPARs inside of LPARs**

<table>
<thead>
<tr>
<th>Environment</th>
<th>LPARs</th>
<th>WPARs</th>
</tr>
</thead>
<tbody>
<tr>
<td>Development</td>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td>Demo/Train</td>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td>Test</td>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td>Pre-Prod</td>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td>Production</td>
<td>1</td>
<td>3</td>
</tr>
</tbody>
</table>
Playing nicely in the sandbox

In virtually every environment I've managed, my staff has begged to have sandbox environments in which to work. These environments would be used only by the systems administrators. It is here that administrators have the opportunity to install new software, test out new patches, install new technology levels, and generally be free to break the system without any effect to the business. Unfortunately, it is always the sandbox that is the first environment that must be given up when a new application needs to be deployed. With WPARs, you can quickly create an isolated environment in which to play. While my preference is to have several WPAR sandboxes within an overall LPAR sandbox, each of these owned by a different administrator, this now becomes less of a luxury than it used to be. Looking at this from another perspective, these WPARs are the training ground for new administrators to learn and practice their craft on. With WPARs, they can now be managed much more efficiently and created without having to assign dedicated devices to them.

Quickly testing an application

The application WPAR can be created in just a few seconds. What better way is there to quickly troubleshoot an application or wayward process? As these are temporary resources, they are destroyed as soon as they end, simplifying the manageability of these partitions.

WPARGS: When not to use them

This section discusses situations and scenarios where you may not want to use WPARs.

Security

As stated previously, WPAR processes can be seen by the global environment from the central LPAR. If you are running a highly secure type of system, this may be a problem for you from a security standpoint. Further, the root administrator of your LPAR will now have access to your workload partition, possibly compromising the security that the application may require.
**Performance**

Each WPAR within LPAR is now using the same system resources of the LPAR. You need to be that much more careful when architecting your system and also when stress testing the system. For example, if you're running a performance benchmark on your pre-production system after a new build has been deployed and there are some developers working on the application server while you are testing your database, this will all be done within one LPAR sharing the same resources. Your teams will all need to understand that there will be competing resources now for the same product.

**Availability**

If you are in an environment where it is very difficult to bring a system down, it's important to note that when performing maintenance on an LPAR that every WPAR defined will be affected. At the same time, if there is a system panic and AIX crashes, every WPAR has now been brought down. From this standpoint, LPARs without WPARs can provide increased availability across your environment, albeit at a cost that may be prohibitive.

**Production**

I'm extremely conservative when it comes to production. I like to run each tier in production within its own logical partition. I do this because I like the granularity and complete OS isolation that LPARs provide, without having multiple environments (Web, application, and database) to worry about.

**Physical devices**

Physical devices are not supported within a WPAR. While there is a way to export devices, this can be a big problem for applications that require non-exportable devices. In this case, they would be restricted to only running in the global environment. For example, Oracle RAC is not supported using Solaris zones because of this limitation, and should not work in a WPAR environment for the very same reason.

**Creating, configuring, and administering WPARs**

This section creates, configures, and administers WPARs, both system and application.

**System WPARs**

The mkwpar command creates the WPAR, installs the filesystems, and prepares the system (see Listing 1). It also synchronizes the root section of the installed software.
Listing 1. The mkwpar command

```bash
lpar5ml162f_pub[] > mkwpar -n devpayrollWPAR01
mkwpar: Creating file systems...

/home
/opt
/proc
/tmp
/usr
/var
```

<< End of Success Section >>

FILESET STATISTICS
------------------

241 Selected to be installed, of which:

---
241 Passed pre-installation verification

241 Total to be installed

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

Filesets processed: 6 of 241 (Total time: 2 secs).
installp: APPLYING software for:
X11.base.smt 6.1.0.1
Filesets processed: 7 of 241 (Total time: 3 secs).
installp: APPLYING software for:
X11.help.EN_US.Dt.helpinfo 6.1.0.0
Filesets processed: 8 of 241 (Total time: 3 secs).
installp: APPLYING software for:
bos.acct 6.1.0.1
Filesets processed: 9 of 241 (Total time: 3 secs).
installp: APPLYING software for:
bos.acct 6.1.0.2
Filesets processed: 10 of 241 (Total time: 4 secs).
installp: APPLYING software for:
bos.adt.base 6.1.0.0
bos.adt.insttools 6.1.0.0
Filesets processed: 12 of 241 (Total time: 4 secs).
installp: APPLYING software for:
bos.compat.links 6.1.0.0
bos.compat.net 6.1.0.0
bos.compat.termcap 6.1.0.0

Workload partition devpayrollWPAR01 created successfully.
mkwpar: 0960-390 To start the workload partition, execute the following as root: startwpar [-v] devpayrollWPAR01

Depending on the type of system you are using, this generally takes between two and four minutes. It took me two minutes and 40 seconds, installing 241 filesets on a one-CPU POWER5 processor running at 1654 MHz. To check the status of the WPAR, use the lswpar command (see Listing 2).

Listing 2. Use the lswpar command to check the status of the WPAR

```bash
lpar5ml162f_pub[] > lswpar
Name State Type Hostname Directory
```

Workload Partitioning (WPAR) in AIX 6.1
Page 6 of 18 © Copyright IBM Corporation 1994, 2007. All rights reserved.
In this case, it is still in what is called the "defined state." We'll need to use the startwpar command to make it active (see Listing 3).

**Listing 3. Using the startwpar command**

```
1par5ml162f_pub[/] > startwpar -v devpayrollWPAR01
Starting workload partition devpayrollWPAR01.
Mounting all workload partition file systems.
Mounting /wpars/devpayrollWPAR01
Mounting /wpars/devpayrollWPAR01/home
Mounting /wpars/devpayrollWPAR01/opt
Mounting /wpars/devpayrollWPAR01/proc
Mounting /wpars/devpayrollWPAR01/tmp
Mounting /wpars/devpayrollWPAR01/usr
Mounting /wpars/devpayrollWPAR01/var
Loading workload partition.
$corral_t = {
    'name' => 'devpayrollWPAR01',
    'wlm_cpu' => [ undef,
    undef,
    undef,
    undef
    ],
    'path' => '/wpars/devpayrollWPAR01',
    'hostname' => 'devpayrollWPAR01',
    'wlm_procVirtMem' => [ -1,
    undef
    ],
    'wlm_mem' => [ undef,
    undef,
    undef,
    undef
    ],
    'key' => 3,
    'vips' => [],
    'wlm_rset' => undef,
    'opts' => 4,
    'id' => 0
};
Exporting workload partition devices.
Starting workload partition subsystem cor_devpayrollWPAR01.
0513-059 The cor_devpayrollWPAR01 Subsystem has been started. Subsystem PID is 753708.
Verifying workload partition startup.
Return Status = SUCCESS.
1par5ml162f_pub[/] >
```

You can now see that it is in an active state (see Listing 4)

**Listing 4. The WPAR is in an active state**

```
1par5ml162f_pub[/] > lswpar
Name  State  Type  Hostname  Directory
```
Let's run some standard AIX commands (see Listing 5).

### Listing 5. Some standard AIX commands

```bash
# hostname
devpayrollWPAR01
# w
10:59AM up 13 mins, 1 user, load average: 0.00, 0.00, 0.00
User tty login@ idle JCPU PCPU what
root Global 10:59AM 1 0 0 -
rw  
# whoamioot
# ps -ef
  UID PID PPID  C   STIME   TTY TIME CMD
root 258064  573578 0 10:47:42 - 0:00 /usr/sbin/sshd
root 340006  573578 0 10:47:55 - 0:00 /usr/sbin/rsct/bin/IBM.Servic
root 356468  573578 0 10:47:56 - 0:00 /usr/sbin/rsct/bin/IBM.AuditR
root 421948  573578 0 10:47:41 - 0:00 /usr/sbin/rpc.lockd -d 0
root 471122  1   0  10:47:23 - 0:00 /usr/lib/errdemon
root 508124  643204 28 11:00:15 ? 0:00 ps -ef
root 508124  643204 28 11:00:15 ? 0:00 ps -ef
root 504032  573578 0 10:47:56 - 0:00 /usr/sbin/portmap
root 512114  573578 0 10:47:39 - 0:00 /usr/sbin/portmap
root 512114  573578 0 10:47:39 - 0:00 /usr/sbin/portmap
root 561344  573578 0 10:47:56 - 0:00 /usr/sbin/rsct/bin/IBM.CSMAge
root 573578   1  0 10:47:33 - 0:02 /usr/sbin/srmstr
root 602226   1  0 10:47:41 - 0:00 /usr/sbin/cron
root 606358  573578 0 10:47:41 - 0:00 /usr/sbin/gdaemon
root 630928   1  0 10:59:02 ? 0:00 clogin devpayrollWPAR01
root 635076  573578 0 10:47:39 - 0:00 sendmail: accepting connectio
root 643204  630928 0 10:59:02 ? 0:00 -ksh
root 651276  573578 0 10:47:39 - 0:00 /usr/sbin/biod 6
root 655560  573578 0 10:47:41 - 0:00 /usr/sbin/writesrv
root 737494  573578 0 10:47:54 - 0:00 /usr/sbin/rsct/bin/rmcd -a IB
root 741406  573578 0 10:47:39 - 0:00 /usr/sbin/inetd
root 749714  573578 0 10:47:38 - 0:00 /usr/sbin/syslogd
root 753788   1  0 10:47:21 - 0:00 /etc/init
```

Your systems administrator can start and stop processes from the WPAR using the SRC or from the command line, just as they would from the global environment. As the Global (LPAR) system administrator, you will note that a WPAR has lots of filesystems. The WPAR environment is created under /wpars (see Listing 6).
### Listing 6. Creating the WPAR environment under /wpars

```bash
lpar5ml162f_pub[/wpars/devpayrollWPAR01/wpars] > hostname
lpar5ml162f_pub
# df -k
Filesystem  1024-blocks  Free  %Used  Iused  %Iused Mounted on
/dev/hd4     131072     19472   86%    8278    62%   /
/dev/hd2     3538944    150480   96%    91842    70%   /usr
/dev/hd9var  262144     246796    6%     522     1%   /var
/dev/hd3     262144     259540   1%     56      1%   /tmp
/dev/hd1     131072     130688   1%      8      1%   /home
/dev/hd1admin 131072     130708   1%      5      1%   /admin
/dev/hd10opt  262144     119804   55%    3048    11%   /opt
/dev/fs1v12   131072     103476   22%    2244     9% /wpars/devpayrollWPAR01/ora01
/dev/fs1v13   131072     128660   2%      5      1%   /wpars/devpayrollWPAR01/home
/opt         262144     119804   55%    3048     11% /wpars/devpayrollWPAR01/opt
/proc        -         -        -      -        -   /proc
/dev/fs1v14   131072     128424   3%      9      1%   /wpars/devpayrollWPAR01/tmp
/usr         3538944    150480   96%    91842    70%   /wpars/devpayrollWPAR01/usr
/dev/fs1v15   131072     116448  12%     370     2%   /wpars/devpayrollWPAR01/var
```

Here is the view from the WPAR

```bash
# hostname
devpayrollWPAR01
# df -k
Filesystem  1024-blocks  Free  %Used  Iused  %Iused Mounted on
/dev/fs1v12   131072     103476   22%    2244     9%   /
/dev/fs1v13   131072     128660   2%      5      1%   /home
/opt         262144     119804   55%    3048     11% /opt
/proc        -         -        -      -        -   /proc
/dev/fs1v14   131072     128424   3%      9      1%   /tmp
/usr         3538944    150480   96%    91842    70%   /usr
/dev/fs1v15   131072     116448  12%     370     2%   /var
```

### Creating filesystems

Let's turn our attention back to the global environment. Let's create a filesystem through SMIT. You cannot create a f/s or volume group from the WPAR, only from the global environment (LPAR).

We need to make sure that the full path of the filesystem (including the WPAR path) is specified (see Figure 1).

**Figure 1. The full path of the filesystem is specific in SMIT**
Figure 2 shows the file system has been created successfully.

**Figure 2. The file system has been created successfully**

After it’s successfully created, you’ll need to make one minor change to the filesystem: the mount group needs to be explicitly defined (see Figure 3). Note that this step is not necessary when using the command line to create the filesystem: `# smit chjfs2`.

**Figure 3. Explicitly defining the mount group**
Now let's turn back to the WPAR, where you'll create the mountpoint and mount the newly created filesystem (see Listing 7).

Listing 7. Creating the mountpoint and mounting the filesystem

```
# mkdir ora
# pwd
/
# mount ora /ora01
# df -k

Filesystem 1024-blocks Free %Used Iused %Iused Mounted on
/dev/fslv12 131072 103444 22% 2246 9% /
/dev/fslv13 131072 128660 2% 5 1% /home
/opt 262144 119804 55% 3048 11% /opt
/proc - - - - - /proc
/dev/fslv14 131072 128424 3% 9 1% /tmp
/usr 3538944 150480 96% 91842 70% /usr
/dev/fslv15 131072 116448 12% 370 2% /var
/ora 131072 103444 22% 2246 9% /ora01
```

Note that you also cannot increase the size of a filesystem from the WPAR, only from the global environment. You also cannot serve NFS filesystems from within the WPAR; only NFS clients are supported.

**Backups**

Remember, there are no physical devices in a WPAR. When backing up the WPAR environment, you need to use the savewpar command, again from the global environment.

Listing 8. Using the savewpar command

```
$ lpar5ml62f_pub[/wpars/devpayrollWPAR01/wpars] > savewpar
-f /admin/payroll.backup devpayrollWPAR01
```
Creating information file for workload partition devpayrollWPAR01.

Creating list of files to back up.
Back up ing 2829 files
2829 of 2829 files (100%)
0512-038 savewpar: Backup Completed Successfully.
lpar5ml162f_pub[/wpars/devpayrollWPAR01/wpars] >

You can restore using the restwpar command.

Users and groups

You can maintain users and groups within the WPAR, either from the command line or through SMIT. You should understand that the root user for this environment does not have access to the global environment, only to the WPAR (see Listing 9).

Listing 9. Maintaining users and groups within the WPAR

```
# mkuser test
# mkgroup testing
# hostname
devpayrollWPAR01
# luser
Usage: luser [-R load_module] [ -c ] [ -f ] [ -a attr attr ... ]
{ "ALL" | user1,user2 ... }
# luser test
test id=204 pgrp=staff groups=staff home=/home/test shell=/usr/bin/ksh
login=true su=true rlogin=true daemon=true admin=false sugroups=ALL admgroups=
tpath=nosak ttyps=ALL expires=0 auth1=SYSTEM auth2=NONE umask=22 registry=files
SYSTEM=compat logintimes= loginretries=0 pwarntime=0 account_locked=false
minage=0 maxage=0 maxexpired=-1 minalpha=0 minother=0 mindiff=0 maxrepeats=8 minlen=0
histexpire=0 histsize=0 pwdchecks= dictionlist= default_roles= filesystem=files
SYSTEM=compat logintimes= loginretries=0
# lsgroup testing
testing id=203 admin=false users= registry=files
```

Now let's turn our attention back to the global environment. You can clearly see in Listing 10 that the user was not created in the global environment, only within that specific WPAR.

Listing 10. The user was not created in the global environment

```
lpar5ml162f_pub[/wpars/devpayrollWPAR01/wpars] > lsuser test
3004-687 User "test" does not exist.
lpar5ml162f_pub[/wpars/devpayrollWPAR01/wpars] >
```

WPAR manager

It's worth noting that there is a graphical tool called WPAR manager, which is Java™ based and allows for the centralized management of WPARs (see Figure 4).
While a thorough review of this utility is outside the scope of this article, it's definitely worth looking at because using it will increase your ability to manage the overall environment. It will also help you harness innovations such as Workload Partition Manager and WPAR Mobility. Workload Partition Manager allows for resource optimization, allowing you to distribute workloads more efficiently throughout your managed system. WPAR mobility allows you to move running partitions from one frame to another, which increase availability of workloads during scheduled outages.

**Application WPARs**

To reiterate, an application WPAR is defined as a WPAR that allows an application and/or a process to run inside of it, similar to a wrapper. It is only temporary, not a permanent object, and it will end when the application and/or process ends. To create one, use the wparexec command.

**Listing 11. Using the wparexec command to create an application WPAR**

```
 1par5ml162f_pub[/wpars/devpayrollWPAR01/wpars] > wparexec -n templsl /usr/bin/ls
Starting workload partition templsl.
Mounting all workload partition file systems.
```
To see how the process works while it is working, you will see the creation of the WPAR (see Listing 12).

**Listing 12. Seeing the creation of the WPAR**

```bash
lpar5ml162f_pub[/wpars/devpayrollWPAR01/wpars] > lswpar
```

<table>
<thead>
<tr>
<th>Name</th>
<th>State</th>
<th>Type</th>
<th>Hostname</th>
<th>Directory</th>
</tr>
</thead>
<tbody>
<tr>
<td>MyTestWpar1</td>
<td>A</td>
<td>S</td>
<td>MyTestWpar1</td>
<td>/wpars/MyTestWpar1</td>
</tr>
<tr>
<td>MyTestWpar2</td>
<td>A</td>
<td>S</td>
<td>MyTestWpar2</td>
<td>/wpars/MyTestWpar2</td>
</tr>
<tr>
<td>devpayrollWPAR01</td>
<td>A</td>
<td>S</td>
<td>devpayrollWPAR01</td>
<td>/wpars/devpayrollWPAR01</td>
</tr>
<tr>
<td>evpayrollWPAR01</td>
<td>D</td>
<td>S</td>
<td>evpayrollWPAR01</td>
<td>/wpars/evpayrollWPAR01</td>
</tr>
<tr>
<td>tempsl1</td>
<td>T</td>
<td>A</td>
<td>tempsl1</td>
<td></td>
</tr>
</tbody>
</table>

When the process completes, it is gone, just as fast as it was created.

**Listing 13. The process is gone**

```bash
lpar5ml162f_pub[/] > lswpar
```

<table>
<thead>
<tr>
<th>Name</th>
<th>State</th>
<th>Type</th>
<th>Hostname</th>
<th>Directory</th>
</tr>
</thead>
<tbody>
<tr>
<td>MyTestWpar1</td>
<td>A</td>
<td>S</td>
<td>MyTestWpar1</td>
<td>/wpars/MyTestWpar1</td>
</tr>
<tr>
<td>MyTestWpar2</td>
<td>A</td>
<td>S</td>
<td>MyTestWpar2</td>
<td>/wpars/MyTestWpar2</td>
</tr>
<tr>
<td>devpayrollWPAR01</td>
<td>A</td>
<td>S</td>
<td>devpayrollWPAR01</td>
<td>/wpars/devpayrollWPAR01</td>
</tr>
<tr>
<td>evpayrollWPAR01</td>
<td>D</td>
<td>S</td>
<td>evpayrollWPAR01</td>
<td>/wpars/evpayrollWPAR01</td>
</tr>
</tbody>
</table>

Truthfully, although it's impressive that you can create application WPARs in a matter of seconds, and it's a feature that Solaris does not have, I think it is most useful for providing additional flexibility for testing purposes.

**Summary**

This article introduced WPARs and discussed the context in which to use them. The article looked at various scenarios in which WPARs should be used. It also discussed the installation, configuration, and administration of WPARs and how they relate to the global (LPAR) environment. You added users, created filesystems, and backed up WPARs. You also introduced utilities such as WPAR manager, which could be used to help manage the WPAR environment. You looked at the different types of WPARs that are available and the limitations of application WPARs compared to system WPARs. You also looked at scenarios in which WPARs may not be considered. The bottom line is that WPARs are an important innovation of AIX 6.1, and used judiciously, can increase your ability to effectively manage your system and reduce cost to the business.
Resources

Learn

• Read the Redbook *Introduction to Workload Partition Management in IBM AIX Version 6.1* for detailed information on WPAR.

• For a whitepaper that introduces WPAR concepts, see AIX 6 Workload Partition and Live Application Mobility white paper introduces WPAR concepts (Mr. Thierry Fauck, III, developerWorks, September 2007).

• For a Redbook that focuses on the differences introduced in IBM AIX Version 6.1 when compared to AIX 5L Version 5.3, see *IBM AIX Version 6.1 Differences Guide*.


• For a three-part series that focuses on the challenges of optimizing disk I/O performance, see Optimizing AIX 5L performance: Tuning disk performance, Part 1 (Ken Milberg, developerWorks, July 2007).

• For a three-part series on memory tuning on AIX, see Optimizing AIX 5L performance: Tuning your memory settings, Part 1 (Ken Milberg, developerWorks, June 2007).

• For an article that discusses the benefits of Direct I/O and tells how to implement it, see Use Direct I/O to improve performance of your AIX applications (Shiv Dutta, developerWorks, November 2002).

• Read the IBM whitepaper Improving Database Performance with AIX concurrent I/O.

• Learn about AIX memory affinity support from the IBM System p and AIX InfoCenter.

• The Redbook, Database Performance Tuning on AIX, is designed to help system designers, system administrators, and database administrators design, size, implement, maintain, monitor, and tune a Relational Database Management System (RDBMS) for optimal performance on AIX.

• Get ready to take Test 234: AIX 5L Performance and Systems Tuning as part of IBM's certification program.

• Learn about IBM's Power Architecture: High-Performance Architecture with a History.

• Read Power to the People; A history of chip making at IBM (developerWorks,
December 2005) for coverage of IBM's power architecture.

- Using process affinity settings to bind or unbind threads can help you find the root cause of troublesome hang or deadlock problems. Read Processor Affinity on AIX (developerWorks, November 2006) to learn how to use processor affinity to restrict a process and run it only on a specified central processing unit (CPU).

- Read CPU Monitoring and Tuning (Wayne Huang et al. developerWorks, March, 2002) to learn how standard AIX tools can help you determine CPU bottlenecks.

- For a comprehensive guide about the performance monitoring and tuning tools that are provided with AIX 5L Version 5.3, see the IBM Redbook AIX 5L Practical Performance Tools and Tuning Guide.


- Operating System and Device Management from IBM provides users and system administrators with complete information that can affect your selection of options when performing such tasks as backing up and restoring the system, managing physical and logical storage, and sizing appropriate paging space.


- The AIX and UNIX developerWorks zone provides a wealth of information relating to all aspects of AIX systems administration.

- New to AIX and UNIX: Visit the New to AIX and UNIX page to learn more about AIX and UNIX.

- Open source: Visit the developerWorks Open source zone for extensive how-to information, tools, and project updates to help you develop with open source technologies, and use them with IBM products.

- developerWorks technical events and webcasts: Stay current with developerWorks technical events and webcasts.

- AIX Wiki: A collaborative environment for technical information related to AIX.

- Podcasts: Tune in and catch up with IBM technical experts.

- For a comprehensive guide about the performance monitoring and tuning tools that are provided with AIX 5L Version 5.3, see the IBM Redbook AIX 5L Practical Performance Tools and Tuning Guide.

Discuss

- Participate in the AIX and UNIX forums:
About the author

Ken Milberg
Ken Milberg, PMP, is a technology writer/site expert for techtarget.com and provides Linux technical information and support at searchopensource.com. Ken is also a writer and technical editor for IBM Systems Magazine, Open Edition. Ken holds a Bachelor of Science in Computer and Information Science, as well as a Master of Science in Technology Management from the University of Maryland University College. He is the founder and group leader of the Long Island POWER-AIX users group. Through the years, he has worked for both large and small organizations and has held diverse positions from CIO to Senior AIX Engineer. Today, he works for Future Tech, a Long Island-based IBM business partner. Ken is a PMI certified Project Management Professional (PMP) and an IBM Certified Advanced Technical Expert (CATE, IBM System p5 2006).