Linux on System z9 and zSeries

How to Improve Performance with PAV December 14, 2005

Linux Kernel 2.6 (October 2005 stream)

SC33-8292-00

Linux on System z9 and zSeries

How to Improve Performance with PAV December 14, 2005

Linux Kernel 2.6 (October 2005 stream)

SC33-8292-00

Note
Before using this information and the product it supports, read the information in "Notices" on page 7.

First Edition (December 2005)

This edition applies to Linux kernel 2.6 (October 2005 stream) and to all subsequent releases and modifications until otherwise indicated in new editions.

This edition is the October 2005 stream equivalent to LNUX-H6PA, that applies to the Linux on zSeries kernel 2.6, April 2004 stream. This edition applies to both the October 2005 stream and the April 2004 stream.

Contents

Summary of changes

This book is the kernel 2.6 October 2005 stream equivalent to LNUX-HTPA-00 which applies to kernel 2.6, April 2004 stream. This book applies to both the October 2005 stream and the April 2004 stream. Changes compared to LNUX-HTPA-00 are indicated by a vertical line to the left of the change. The changes are:

- PAV enabled volumes can now be accessible to more than one Linux® instance at a time
- PAV support is now also available on IBM® TotalStorage® enterprise disk storage systems (for example IBM TotalStorage DS8000), in addition to IBM TotalStorage Enterprise Storage Server® (ESS).

This revision also includes maintenance and editorial changes.

About this publication

This document describes how to set up volumes for PAV using the Enterprise Volume Management System (EVMS).

In this book, System z9™ and zSeries® is taken to include System z9, zSeries in 64- and 31-bit mode, as well as S/390® in 31-bit mode.

You can find the latest version of this document on developerWorks® at:
`ibm.com/developerworks/linux/linux390/october2005_documentation.html`

Where to get more information

The descriptions are based on the EVMS Ncurses interfaces. Visit `http://evms.sourceforge.net` for details on EVMS.

How to improve performance with PAV

Note: The procedure described in this HowTo assumes that the volumes to be set up for PAV do not already hold data.

The concurrent operations capabilities of the IBM TotalStorage enterprise disk storage systems and IBM TotalStorage Enterprise Storage Server (ESS), support concurrent data transfer operations to or from the same volume from the same system or system image. A volume that can be accessed in this way is called a Parallel Access Volume (PAV).

The operating system does not attempt to start more than one I/O operation at a time to a device, but today's storage subsystem design, with large caches and RAID 5 arrays, makes it possible for the storage control unit to do I/Os in parallel.

When software is using PAV, it can issue multiple channel programs to a volume, allowing simultaneous access to the logical volume by multiple users or processes. Reads can be satisfied simultaneously, as well as writes to different domains. The domain of an I/O consists of the specified extents to which the I/O operation applies. Writes to the same domain still have to be serialized to maintain data integrity.

Prerequisites: Linux on an IBM System z9 or @server zSeries mainframe can use PAV if all of the following apply:

- Linux runs as a z/VM® guest.
- The volume resides on either of:
  - An IBM TotalStorage Enterprise Storage Server (ESS)
  - An IBM TotalStorage enterprise disk storage system

Restrictions:

- To ensure data integrity Linux must use EVMS. By default, Linux interprets each path as leading to a separate volume. EVMS allows Linux to recognize where multiple paths lead to the same volume.

Setting up a disk volume for PAV includes these tasks:

1. Configuring the volume on the storage system. The volume must be configured as a base device with at least one alias device. There is no separate real disk space associated with alias devices.

   Refer to your storage system documentation for details. For example, *IBM TotalStorage Enterprise Storage Server Web Interface User's Guide*, SC26-7448.
2. Defining the volume to the System z9 or zSeries hardware. See "Enabling volumes for PAV."
3. Configuring paths to the volume on Linux. See "Configuring PAV base and alias volumes with EVMS" on page 2.

Enabling volumes for PAV

This section describes how you must define disk volumes to your hardware so that Linux can use them for PAV.

Prerequisites:

- You need to know the device numbers of the base devices and their aliases as defined on the storage system.
- You need privilege class B authorization on z/VM.

Perform the following steps to define the base devices and their aliases to the hardware. In the examples, we assume that device 0x5680 is a base device and 0x56BF an alias device for the same physical disk space on the storage system.

1. Define the base devices to the hardware. In an IOCDS IODEVICE statement, use UNIT=3390B.

   Example: The following statement defines device number 0x5680 as a base device.

   ```
   IODEVICE ADDRESS=(5680),UNITADD=00,CUNUMBR=(5680),        *
                 STADET=Y,UNIT=3390B
   ```

2. Define the alias devices to the hardware. In an IOCDS IODEVICE statement, use UNIT=3390A.

   Example: The following statement defines device 0x56BF as an alias device. The mapping to the associated base device 0x5680 is in the storage system configuration.

   ```
   IODEVICE ADDRESS=(56BF),UNITADD=18,CUNUMBR=(5680),        *
                 STADET=Y,UNIT=3390A
   ```

3. After the hardware configuration with the base and alias device statements has become active, use z/VM to check the mapping of base and alias devices.

   Example:

   ```
   # CP QUERY PAV
   00: Device 5680 is a base Parallel Access Volume with the following aliases: 56BF
   00: Device 56BF is an alias Parallel Access Volume device whose base device is 5680
   ```

4. From z/VM, use `CP ATTACH` commands to make base devices and their aliases accessible to the Linux guest.

   Example: To make a base device 0x5680 and its alias 0x56BF available to a guest with ID “LNX1” issue:

   ```
   # ATTACH 5680 LNX1
   # ATTACH 56BF LNX1
   ```

You can now configure the devices in Linux.

Configuring PAV base and alias volumes with EVMS

This section describes how to define a PAV base device and its aliases as a single logical volume.

Prerequisites:

- You must know the device numbers of the PAV base device and its aliases.
- You need root authorization on the Linux system

From the IPLed Linux guest, perform the following steps:

1. Assure that EVMS controls the PAV base device and all its aliases.

   By default, EVMS controls all available disk devices. You can provide a configuration file, `/etc/evms.conf`, to define which devices are to be excluded from EVMS control.

a. Make a copy of the configuration file template `/etc/evms.conf.template` and name it `/etc/evms.conf`.

b. Open `/etc/evms.conf` with your preferred text editor and look for the following lines:

```
sysfs_devices {
     ...
     include = [ * ]
     ...
     exclude = [ ]
}
```

c. In the exclude statement, explicitly specify devices you do not want to be controlled by EVMS. You can use the asterisk '*' as a wildcard. The default specification is empty and does not exclude any devices.

   Example: Assuming that there are two DASD, dasda and dasdb, that are not to be controlled by EVMS, change the exclude statement to:

```
exclude = [ dasda* dasdb* ]
```

2. Ensure that the devices are ready for use.

a. Issue **lsdasd** to ensure that device nodes exist for the PAV base volume and its aliases and that the devices are online.

   Device nodes are usually created automatically (for example, by udev). If there are no device nodes, create them yourself.

   For information on how to create DASD device nodes see the DASD chapter of *Linux on System z9 and zSeries Device Drivers, Features, and Commands*. You can find the latest version on the developerWorks Web site at:

   `ibm.com/developerworks/linux/linux390/october2005_documentation.html`

   Use **chccwdev** to set the DASD online, if needed. For details refer to the **chccwdev** man page.

   Example: This **lsdasd** output shows that both the base device, 0x5680, and the alias, 0x56bf, of our example are online and that the standard device names dasdc and dasdd have been assigned to them.

```
# lsdasd
0.0.5601(ECKD) at ( 94:  0) is dasda  : active at blocksize: 4096, 1803060 blocks, 7043 MB
0.0.5602(ECKD) at ( 94:  4) is dasdb  : active at blocksize: 4096, 1803060 blocks, 7043 MB
0.0.5680(ECKD) at ( 94:  8) is dasdc  : active at blocksize: 4096, 1803060 blocks, 7043 MB
0.0.56bf(ECKD) at ( 94: 12) is dasdd  : active at blocksize: 4096, 1803060 blocks, 7043 MB
```

b. Ensure that the device is formatted. If it is not already formatted, use **dasdfmt** to format it. Because a base device and its aliases all correspond to the same physical disk space, formatting either the base device or one of its aliases formats the base device and all alias devices.

   Example:

```
# dasdfmt -f /dev/dasdc
```

c. Ensure that the device is partitioned. If it is not already partitioned, use **fdasd** to create one or more partitions. Because a base device and its aliases all correspond to the same physical disk space, partitioning either the base device or one of its aliases creates partitions for the base device and all alias devices.

   Example: The following command creates both a partition `/dev/dasdc1` for the base device and also a partition `/dev/dasdd1` for the alias.

```
# fdasd -a /dev/dasdc
```

You now have PAV enabled devices for which multiple subchannels are configured. You can display the subchannels for a particular PAV enabled device by issuing a command like this:

```
# lscss | egrep "<devno base device>|<devno alias1>|<devno alias2>| ..."
```

Example: For a base device 0x5680 and alias 0x56BF the command and its output might look like this:

```
# lscss | egrep "5680|56BF"
0.0.5680 0.0.0030  3390/0C 3990/E9  yes FF  FF  FF  C6C7C8CA CBC90000
0.0.56BF 0.0.0031  3390/0C 3990/E9  yes FF  FF  FF  C6C7C8CA CBC90000
```

In the example:

- The base device 0x5680 can be accessed through subchannel 0x0030.
- The alias device 0x56BF can be accessed through subchannel 0x0031.

3. Issue **evmsn** to start the EVMS Ncurses interface. Refer to the *EVMS User Guide* at `http://evms.sourceforge.net` for general information on how to work with EVMS Ncurses.

   EVMS Ncurses comes up with the Logical Volumes view. If you are in a different view, press the **0** key or use the **Tab** key to navigate among the available views.

   The Logical Volumes view shows the base volume and each alias as a separate logical volume. In the following steps we will create a single logical volume for the base volume and all aliases to make Linux treat it as one physical disk.

   Example: The Logical Volumes view for the base volume and alias of our example initially looks like this:

```
Actions   Settings
 0=Logical Volumes

 Name                                    Size  Dirty    Active    R/O  Plug-in        Mountpoint
 ------------------------------------------------------------------------------------------------
 /dev/evms/dasdc1                       6.9 GB
 /dev/evms/dasdd1                       6.9 GB
```

4. Delete the logical volumes for the base and alias device.
   a. Select **Actions —> Delete —> Volume...** to display the Delete logical volumes panel.
   b. Select the logical volumes that correspond to the base volume and to its aliases.

      To select a volume highlight it using the **up** and **down** keys and then press the **spacebar**. An "X" indicates that the volume has been selected.

      Example: This example shows the base device and alias of our example selected for deletion:

```
------------------------------------------ Delete Logical Volume -------------

      Logical Volume                                                     Size
 ---------------------------------------------------------------------------
  X    /dev/evms/dasdc1                                                6.9 GB
  X    /dev/evms/dasdd1                                                6.9 GB
```

   c. Activate **Delete**.
   d. Assuming that you are using a PAV that does not already contain data activate **Write zeros** when prompted.
   e. When prompted whether you want to continue, activate **Continue**.

5. Create a multipath region that contains the base device and its aliases.
   a. Go to the Available objects view. This view shows the partitions on the base volume and on the aliases.

      Press the **1** key or use the **Tab** key to navigate among the available views.

      Example: The Available objects view for the base volume and alias of our example looks like this:

```
Actions   Settings
 1=Available objects

Name                                  Size  Dirty    Active    R/O  Plug-in
 ------------------------------------------------------------------------------
 dasdc1                              6.9 GB                          S390SegMgr
 dasdd1                              6.9 GB                          S390SegMgr
```

   b. Select **Actions —> Create —> Region...**.
   c. From the listed EVMS components, select "MD Multipath Manager".
   d. Activate **Next**.
   e. From the list of available objects, select the base device and its aliases.
   f. Activate **Create**.
   g. Activate **OK**.

      Result: EVMS creates a region in memory. The "X" in the Dirty column indicates that the region is not persistent.

      Example: If the region for our base and alias device is the first region, the name md/md0 is assigned to it.

```
Actions   Settings
 1=Available Objects

 Name                                   Size     Dirty          Active        R/O    Plug-in
 ------------------------------------------------------------------------------------------------
 md/md0                                6.9 GB       X                                MD Multipath
```

   h. Confirm the creation of the region to make it persistent.
      1) Select **Actions —> Save...**.
      2) Activate **Save**.

   Result: The region is now marked *active* and can be used by Linux.

   Example:

```
Actions   Settings
 1=Available Objects

 Name                                   Size     Dirty          Active        R/O    Plug-in
 ------------------------------------------------------------------------------------------------
 md/md0                               6.9 GB                       X                MD Multipath
```

6. Create a single logical volume for the new region.
   a. Ensure that the new region is highlighted.
   b. Select **Actions —> Create —> EVMS Volume...**.
   c. Type the name you want to use for the new logical volume into the volume name entry field.
   d. Activate **Create**.

      Result: The newly created logical volume is shown in the Logical Volumes view. It is not persistent and marked as *dirty*.

      Example: In our example, the logical volume has been named my_pav:

```
Actions   Settings
 0=Logical Volumes

 Name                        Size  Dirty    Active    R/O  Plug-in       Mountpoint
 ------------------------------------------------------------------------------
 /dev/evms/my_pav          6.9 GB    X
```

e. Confirm the creation of the logical volume.

   1) Select **Actions —> Save...**

   2) Activate **Save**.

   Result: The logical volume is now marked *active* and can be used by Linux.

   Example:

```
Actions   Settings
 0=Logical Volumes

 Name                        Size  Dirty    Active    R/O  Plug-in       Mountpoint
 ------------------------------------------------------------------------------
 /dev/evms/my_pav          6.9 GB              X
```

Result: Now EVMS is ready to use multiple paths to the PAV. A file system can be created and mounted as usual.

Be sure to access the PAV through the EVMS device node only. Refrain from accessing the PAV through the DASD device nodes for the base device and the alias. In our example, this would mean using `/dev/evms/my_pav` but not `/dev/dasdc1` or `/dev/dasdd1`.

Note: To reuse an existing EVMS configuration after rebooting Linux you need to issue **evms_activate**.

Notices

This information was developed for products and services offered in the U.S.A. IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

IBM World Trade Asia Corporation
Licensing
2-31 Roppongi 3-chome, Minato-ku
Tokyo 106-0032, Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law:
INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

The licensed program described in this information and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement, or any equivalent agreement between us.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

This information is for planning purposes only. The information herein is subject to change before the products described become available.

Trademarks

The following terms are trademarks of International Business Machines Corporation in the United States, other countries, or both:

developerWorks
Enterprise Storage Server
@server
IBM
S/390
System z9
TotalStorage
z9
z/VM
zSeries

Linux is a trademark of Linus Torvalds in the United States, other countries, or both.

Other company, product, and service names may be trademarks or service marks of others.

Readers' Comments — We'd Like to Hear from You

Linux on System z9 and zSeries
How to Improve Performance with PAV
December 14, 2005
Linux Kernel 2.6 (October 2005 stream)

Publication No. SC33-8292-00

Overall, how satisfied are you with the information in this book?

	Very Satisfied	Satisfied	Neutral	Dissatisfied	Very Dissatisfied
Overall satisfaction	☐	☐	☐	☐	☐

How satisfied are you that the information in this book is:

	Very Satisfied	Satisfied	Neutral	Dissatisfied	Very Dissatisfied
Accurate	☐	☐	☐	☐	☐
Complete	☐	☐	☐	☐	☐
Easy to find	☐	☐	☐	☐	☐
Easy to understand	☐	☐	☐	☐	☐
Well organized	☐	☐	☐	☐	☐
Applicable to your tasks	☐	☐	☐	☐	☐

Please tell us how we can improve this book:

Thank you for your responses. May we contact you? ☐ Yes ☐ No

When you send comments to IBM, you grant IBM a nonexclusive right to use or distribute your comments in any way it believes appropriate without incurring any obligation to you.

Name _____ Address _____

Company or Organization _____

Phone No. _____

Readers' Comments — We'd Like to Hear from You
SC33-8292-00

Cut or Fold
Along Line

Fold and Tape | **Please do not staple** | Fold and Tape

PLACE
POSTAGE
STAMP
HERE

IBM Deutschland Entwicklung GmbH
Information Development
Department 3248
Schoenaicher Strasse 220
71032 Boeblingen
Germany

Fold and Tape | **Please do not staple** | Fold and Tape

SC33-8292-00

Cut or Fold
Along Line

SC33-8292-00