IBM and Linux: Community Innovation for your Business

Using PAV and HyperPAV
with Linux on System z

Dortmund, 29.04.2009

IBM Deutschland
Hans-Joachim Picht
hans@linux.vnet.ibm.com

Trademarks

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

Not all common law marks used by IBM are listed on this page. Failure of a mark to appear does not mean that IBM does not use the mark nor does it mean that the product is not actively marketed or is not significant within its relevant market.

Those trademarks followed by ® are registered trademarks of IBM in the United States; all others are trademarks or common law marks of IBM in the United States.

For a complete list of IBM Trademarks, see www.ibm.com/legal/copytrade.shtml:

*, AS/400®, e business(logo)®, DBE, ESCO, eServer, FICON, IBM®, IBM (logo)®, iSeries®, MVS, OS/390®, pSeries®, RS/6000®, S/30, VM/ESA®, VSE/ESA, WebSphere®, xSeries®, z/OS®, zSeries®, z/VM®, System i, System i5, System p, System p5, System x, System z, System z9®, BladeCenter®

The following are trademarks or registered trademarks of other companies.

Adobe, the Adobe logo, PostScript, and the PostScript logo are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States, and/or other countries.
Cell Broadband Engine is a trademark of Sony Computer Entertainment, Inc. in the United States, other countries, or both and is used under license therefrom.
Java and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.
Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.
Intel, Intel logo, Intel Inside, Intel Inside logo, Intel Centrino, Intel Centrino logo, Celeron, Intel Xeon, Intel SpeedStep, Itanium, and Pentium are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.
UNIX is a registered trademark of The Open Group in the United States and other countries.
Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.
ITIL is a registered trademark, and a registered community trademark of the Office of Government Commerce, and is registered in the U.S. Patent and Trademark Office.
IT Infrastructure Library is a registered trademark of the Central Computer and Telecommunications Agency, which is now part of the Office of Government Commerce.

* All other products may be trademarks or registered trademarks of their respective companies.

Notes:
Performance is in Internal Throughput Rate (ITR) ratio based on measurements and projections using standard IBM benchmarks in a controlled environment. The actual throughput that any user will experience will vary depending upon considerations such as the amount of multiprogramming in the user's job stream, the I/O configuration, the storage configuration, and the workload processed. Therefore, no assurance can be given that an individual user will achieve throughput improvements equivalent to the performance ratios stated here.
IBM hardware products are manufactured from new parts, or new and serviceable used parts. Regardless, our warranty terms apply.
All customer examples cited or described in this presentation are presented as illustrations of the manner in which some customers have used IBM products and the results they may have achieved. Actual environmental costs and performance characteristics will vary depending on individual customer configurations and conditions.
This publication was produced in the United States. IBM may not offer the products, services or features discussed in this document in other countries, and the information may be subject to change without notice. Consult your local IBM business contact for information on the product or services available in your area.
All statements regarding IBM's future direction and intent are subject to change or withdrawal without notice, and represent goals and objectives only.
Information about non-IBM products is obtained from the manufacturers of those products or their published announcements. IBM has not tested those products and cannot confirm the performance, compatibility, or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.
Prices subject to change without notice. Contact your IBM representative or Business Partner for the most current pricing in your geography.

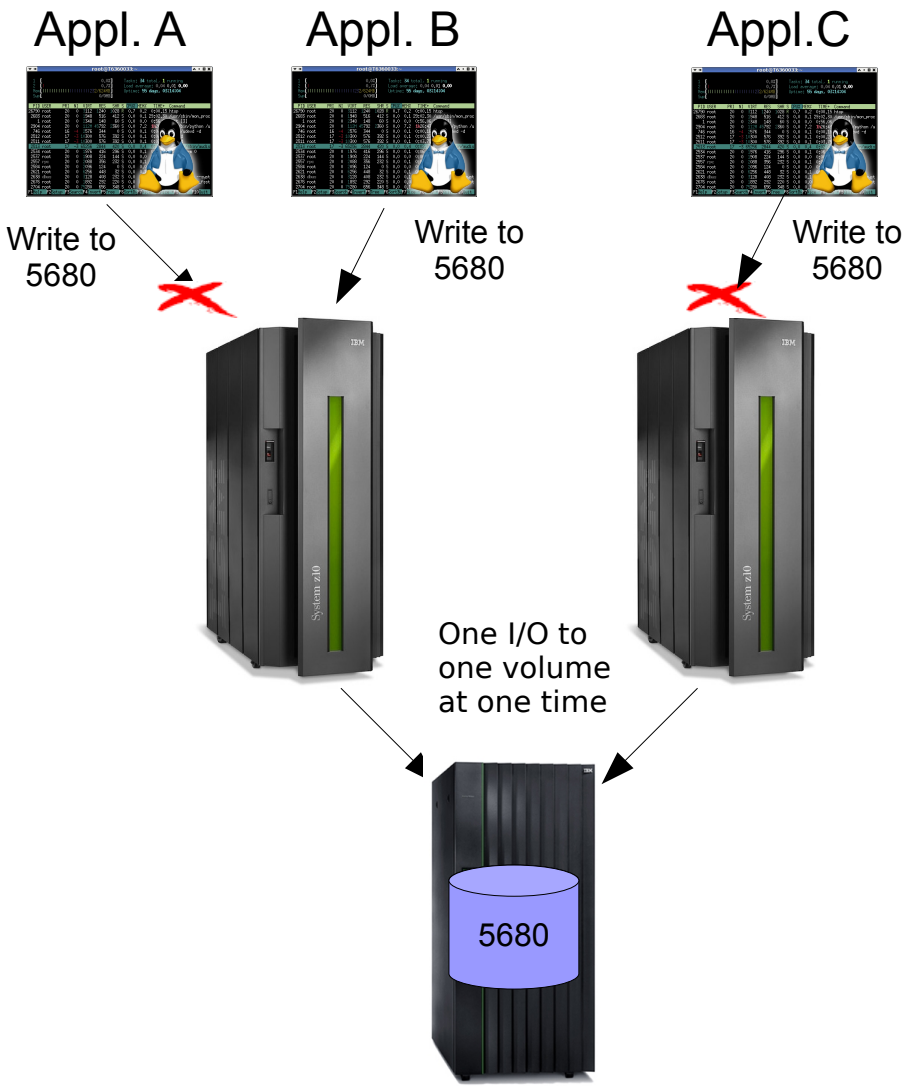
Agenda

* Traditional system behavior
* IBM System Storage DS8000 series
* Parallel Access Volume (PAV) for SLES11 & RHEL6
* Parallel Access Volume (PAV) for SLES10, RHEL4 & 5
* Configuring PAV volumes with multipath tools
* Linux PAV Setup
* HyperPAV
* Linux HyperPAV configuration
* PAV vs HyperPAV
* Performance results

Problem Statement:

Linux on System z customers need to address more data with good performance and high availability!

Traditional system behavior



* Traditional storage disk subsystems have allowed for only one channel program to be active to a volume at a time, in order to ensure that data being accessed by one channel program cannot be altered by the activities of some other channel program.
* Subsequent simultaneous I/Os to volume 5680 are queued while volume 5680 is still busy with a preceding I/O.
* From a performance standpoint, it did not make sense to send more than one I/O at a time to the storage disk subsystem, because the hardware could process only one I/O at a time.
* The DS8000 has the capability to do more than one I/O to a volume.

IBM System Storage DS8000 series

Enterprise Disk for the World's Most Demanding Clients

Over 10,000 systems sold worldwide!!!

Performance, resiliency, and flexibility to satisfy the world's most demanding clients

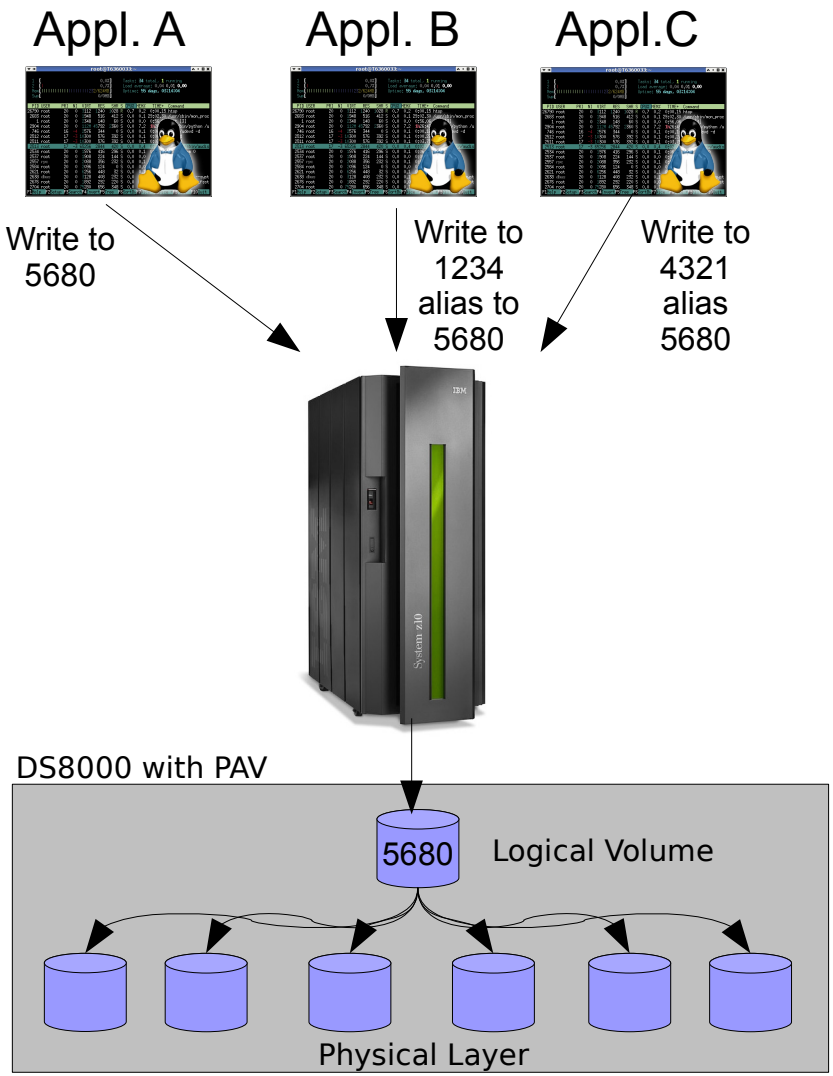
- ***Performance*** – *Architected for highest total throughput*
- ***Availability*** – Designed for 24X7 environments
- ***Resiliency*** – Outstanding Copy and Mirroring Capability
- ***Virtualization*** –Simplification with Variable LPARs
- ***Flexibility*** – High performance, online & high capacity, nearline disk options to satisfy tiered storage objectives
- ***Scalability*** – High scalability for growing transaction workloads
- *Heterogeneous server support* - IBM z/OS, z/VM, OS/400, i5/OS, AIX operating systems, as well as, Linux, HP-UX, Sun SOLARIS, Novell NetWare, VMware and Microsoft Windows environments, among others
- ***Long-Term Cost Advantage*** – Enterprise Choice Warranty and Model-to-Model Upgradeability

Built on 50+ years of enterprise class innovation

- Server/Storage Integration – POWER5™ Technology
- Market share leader for System z environments and outstanding support for distributed platforms
- Exploitation of IBM Virtualization Engine™ Technology
- IBM technology leadership and innovation

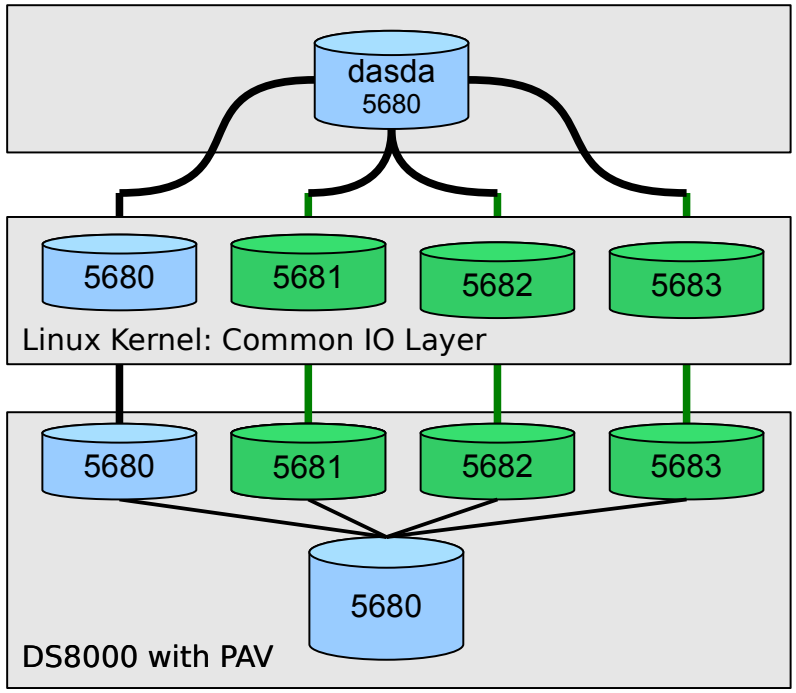
Parallel Access Volume (PAV)

* Linux on System z can use the PAV feature of enterprise storage systems to perform multiple concurrent data transfer operations to or from the same DASD volume.
* The storage control units of present-day IBM enterprise storage systems can use large caches and RAID 5 arrays to perform I/O operations in parallel.
* The Linux kernel can concurrently issue multiple channel programs to volumes that are configured as PAV.
* This results in enhanced performance for I/O intensive applications.



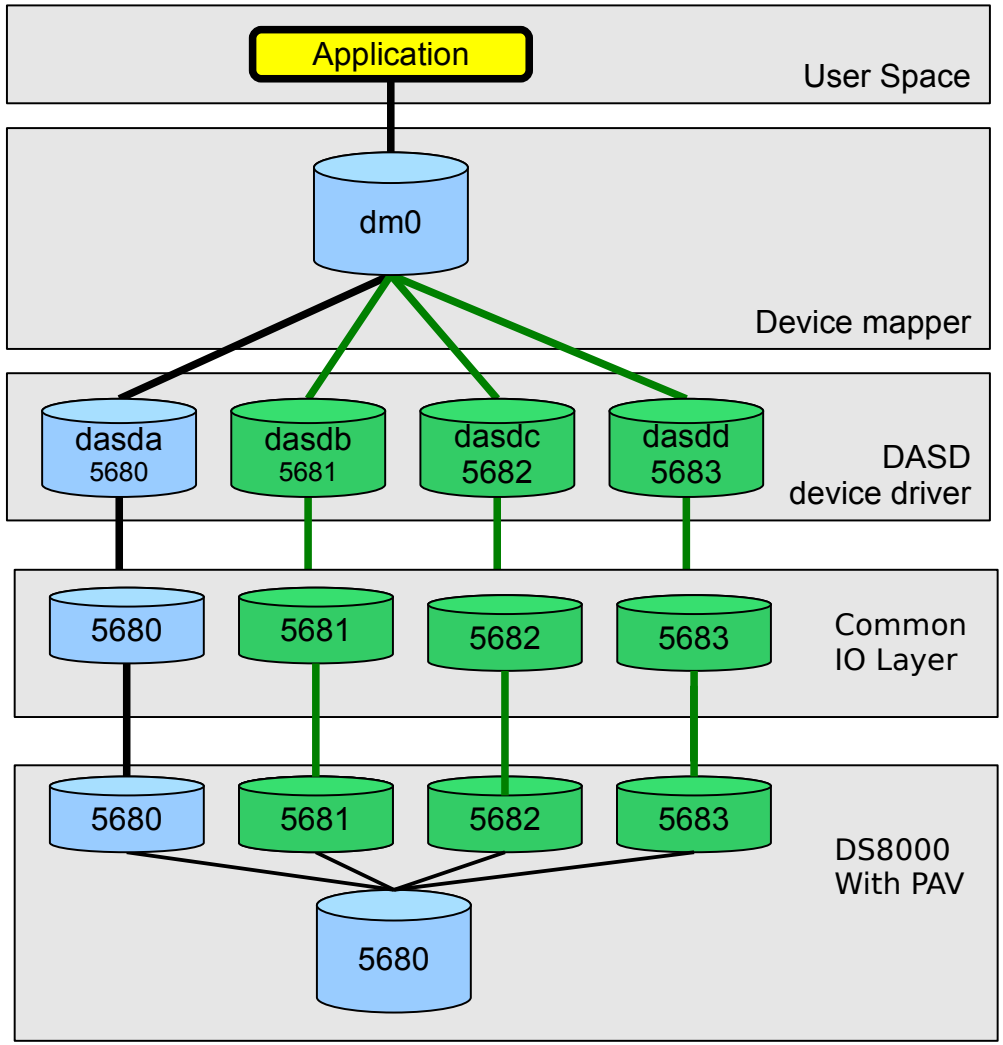
Parallel Access Volume (PAV) for SLES11 & RHEL6

* Through the PAV feature, storage systems can present the same physical disk space as a **base device** and one or more **alias devices**.
* The DASD device driver initially senses the base device.
* The DASD device driver creates device nodes for the base devices but not for the aliases.
* The base device is set online.
* The aliases can lead to gaps in the naming scheme for device nodes.
* If multiple user space processes concurrently access a base device, the device driver uses the aliases to issue multiple channel programs.
* *Apart from assuring that the corresponding aliases for a base device are online, user space processes need no special handling for accessing a PAV.*
* Example: If device 5680 is a base device and devices 5681, 5682 and 5683 are alias devices, all devices are available at the common IO layer. There will be a device node `/dev/dasda`, for device 5680 but no device nodes for 5681-5683 (`/dev/dasdb,dasdc,dasdd`).



Parallel Access Volume (PAV) for SLES10, RHEL4 & 5

* Through the PAV feature, storage systems can present the same physical disk space as a **base device** and one or more **alias devices**.
* The DASD device driver initially senses the base device
* Each DASD base device and each alias device has a separate device node assigned
* Example: If device 5680 is a base device and devices 5681, 5682 and 5683 are alias devices, all device are available at the common IO layer and device nodes `dasda` to `dasdd` will be created.
* Finally device-mapper is required to combine the device dasda to dasdd to a single multipath device, `dm0`



Configuring PAV volumes with multipath tools

- Issue `lsdasd` to ensure that device nodes exist for the PAV base volume and its aliases and that the devices are online.
- Use `chccwdev` to set the DASD online, if needed.
- 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`
- Ensure that the device is partitioned. If it is not already partitioned, use `fdasd` to create one or more partitions. The following command creates both a partition `/dev/dasdc1` for the base device and also a partition `/dev/dasdd1` for the alias.
  - Example: `fdasd -a /dev/dasdc`
- Set the base device and all its aliases offline and back online to assure that the device driver detects the partitions for each device name.
  - Example: `chccwdev -d 0.0.5680,0.0.56bf && chccwdev -e 0.0.5680,0.0.56bf`

Configuring PAV volumes with multipath tools (cont)

- If it is not already loaded, load the `dm_multipath` module: (e.g: `modprobe dm_multipath`)
- Make sure that your multipath.conf configuration file does **not** contain blacklist-entries for dasd devices

```
# cat /etc/multipath.conf
…
blacklist {
        devnode "^(ram|raw|loop|fd|md|dm-|sr|scd|st)[0-9]*"
        devnode "^hd[a-z][[0-9]*]"
        devnode "^cciss!c[0-9]d[0-9]*[p[0-9]*]"
        devnode "^dasd[a-z]+[0-9]*"
}
…
```

- Use the `multipath` command to detect multiple paths to devices for failover or performance reasons and coalesce them
- Make sure multipathd is started (e.g: `/etc/init.d/multipathd start`)
- Enter `multipath -ll` to display the resulting multipath configuration.

Configuring PAV volumes with multipath tools (cont)

```
# multipath -ll
IBM.75000000092461.2a00.1bdm-1 IBM,S/
390 DASD ECKD [size=2.3G][features=0]
[hwhandler=0]
  \_ round-robin 0 [prio=4][enabled]
     \_ 0:0:10779:0 dasde 94:16
[active][ready]
          \_ 0:0:10924:0 dasdf 94:20
[active][ready]
           \_ 0:0:10925:0 dasdg 94:24
[active][ready]
          \_ 0:0:10926:0 dasdh 94:28
[active][ready]
IBM.75000000092461.2a00.1adm-0 IBM,S/
390 DASD ECKD [size=2.3G][features=0]
[hwhandler=0]                    \_
round-robin 0 [prio=4][enabled]
\_ 0:0:10778:0 dasdc 94:12 [active]
[ready] \_ 0:0:10927:0 dasdd 94:32
[active][ready]
```

* The DASDs can now be accessed as multipath devices IBM.75000000092461.2a00.1a and IBM.75000000092461.2a00.1b.
* You can find the corresponding device nodes in /dev/mapper.

```
/dev/mapper/IBM.75000000092461.2a00.1a
/dev/mapper/IBM.75000000092461.2a00.1ap1
/dev/mapper/IBM.75000000092461.2a00.1b
/dev/mapper/IBM.75000000092461.2a00.1bp1
/dev/mapper/control
```

* There is a device node for each multipath device and for each partition on these multipath devices.

You can now use LVM2 or an equivalent logical volume manager to configure the multipath device into a volume group, for example, for striping. If you use LVM2 to work with multipath devices, set a filter to ensure only the multipath devices are used and not the underlying base and aliases.

PAV Prerequisites

* Before you can use PAV on your Linux instance, the PAV feature must be enabled on your storage system.
* The PAV feature is available, for example, for the following systems:
  - IBM System Storage DS8000 series systems
  - IBM System Storage DS6000 series systems
  - IBM TotalStorage Enterprise Storage Server (ESS)
* The HyperPAV feature is available, for example, for IBM System Storage DS8000 series systems.
* PAV base and alias volumes require special IOCDS specifications
* You need to know the device numbers of the base devices and their aliases as defined on the storage system.
* If your Linux system runs as a z/VM guest operating system, you need privilege class B authorization.

IOCDS Configuration

* Configuring base and alias volumes for PAV or HyperPAV on the storage system is beyond the scope of this presentation.
  - See your storage system documentation for details.
  - For information about IOCDS specifications for multiple subchannel sets see the Input/Output Configuration Program User's Guide for your mainframe system.
* The IOCDS examples in this presentation apply to mainframe systems with a single subchannel set.
* Perform the following steps to define the base devices and their aliases to the hardware:
  - Define the base devices to the storage hardware
  - Define the alias devices to the storage hardware
* Example: The following statement defines device number 0x5680 as a base device.

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

* Example: The following statement defines device 0x5681 as an alias device. The mapping to the associated base device 0x5680 is given by the storage system configuration.

```
IODEVICE
ADDRESS=(5681),UNITADD=18,
CUNUMBR=(5680), *
STADET=Y,UNIT=3390A
```

z/VM PAV setup

* If your Linux system runs as a z/VM guest operating system, you can confirm the mapping of base and alias devices.
* After the hardware configuration with the base and alias device statements has become active, enter the z/VM QUERY PAV command

```
# CP QUERY PAV
00: Device 5680 is a base Parallel Access Volume with the following aliases: 56BF
00: Device 5681 is an alias Parallel Access Volume device whose base device is 5680
# CP Q PAV 4DE1
00: Device 4DE1 is not a Parallel Access Volume
```

* To make a base device 0x5680 and its alias 0x5681 available to the current z/VM guest, enter the following commands:

```
# CP ATTACH 5680 *
# CP ATTACH 5681 *
```

Linux PAV Setup cont.

- If your Linux instance runs natively in an LPAR, the `nopav` keyword must not have been set for the `dasd=` kernel or module parameter.
- You need to know the device numbers of the base devices and their aliases as defined on the storage system.
- Perform the following steps to start and confirm your PAV environment. (we assume that device 0x5680 is a base device and 0x5681 a corresponding alias device)
  - Set your base devices online.
  - Set the alias devices online.
  - Confirm the mapping of base and alias devices by reading the `uid` attributes from sysfs.

```
# chccwdev -e 0.0.5680
```

```
# chccwdev -e 0.0.5681
```

```
# cat /sys/bus/ccw/drivers/dasd-eckd/0.0.5680/uid
IBM.75000000092461.2a00.1a
# cat /sys/bus/ccw/drivers/dasd-eckd/0.0.5681/uid
IBM.75000000092461.2a00.1a
```

The logical subsystem of the storage system.

A particular volume within the logical subsystem.

The `uid` attributes are both the same which confirms that both devices map to the same physical disk space.

PAV: LPAR and z/VM

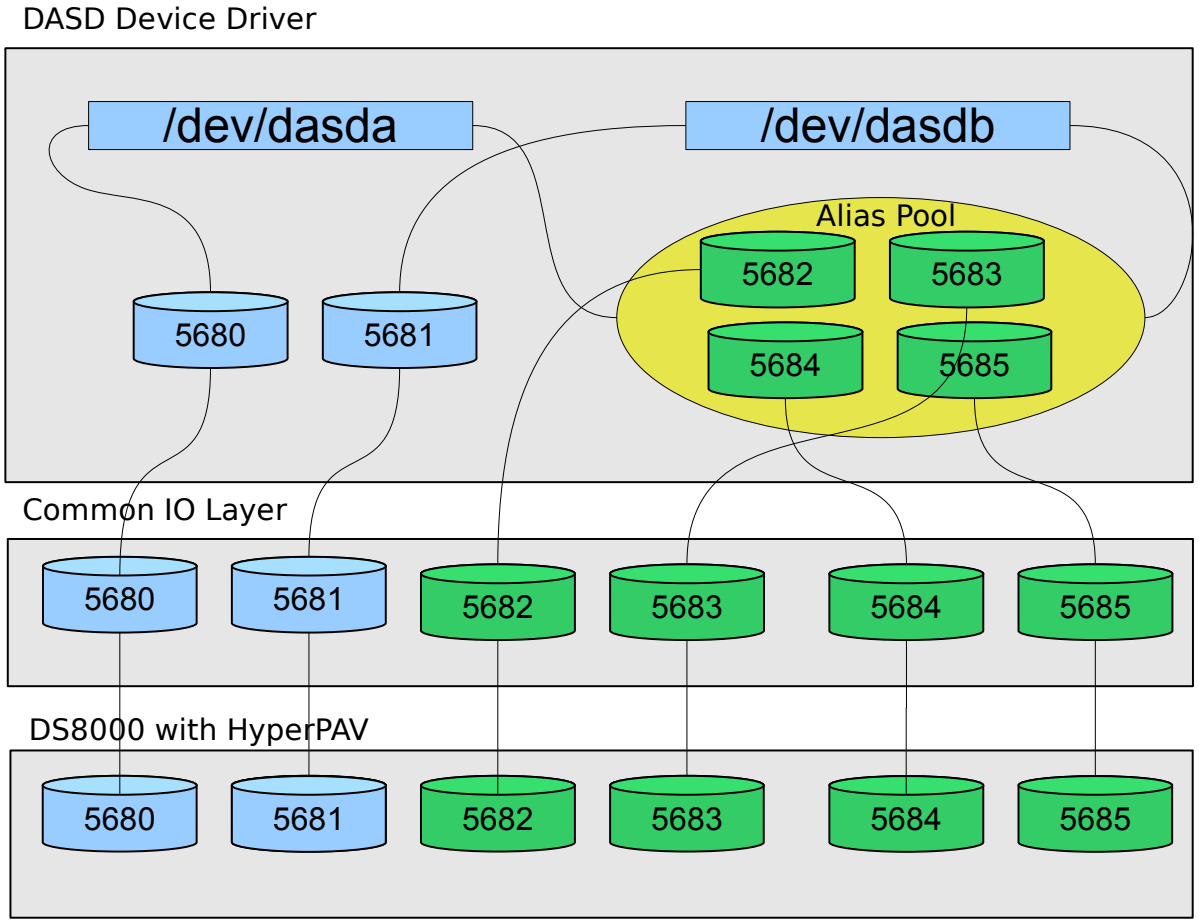
* Alias devices can not be detected until at least one base device is set online

* PAVs were traditionally supported by VM for guests as dedicated DASD
* Base and Alias devices could be dedicated to a single guest or distributed across multiple guests
* Configured to guest(s) with the CP ATTACH command or DEDICATE user directory statement
* Only for guests that exploited the PAV architecture, like z/OS and Linux
* With APAR VM63952 on z/VM 5.2.0, a Base and its Aliases may only be dedicated to one guest
* With APAR VM63952 on z/VM 5.2.0, VM now supports unique UIDs for PAV minidisks

PAV Performance

- Generally speaking, disk performance depends on the time required to do a single I/O operation on a single disk volume in z/VM.
- This time is generally understood as response time, which is composed of queue time and service time. Queue time is the time a guest's I/O spends waiting to access the real volume.
- Service time is the time required to finish the real I/O operation.
- PAV can help to increase the access
- *But if a volume is not experiencing queuing, adding aliases does not help to enhance the volume's performance, for the time required to perform an I/O operation is not appreciably changed by the queue time.*
- In some cases, increasing the number of aliases for a volume might result in the increased service time for that volume.
- Most of the time, the decrease in wait time outweighs the increase in service time, so the response time improves.
- The z/VM Performance Toolkit provides a report on the response time and related performance statistics for the real devices. Use the command DEVICE in the performance toolkit to access the report.

HyperPAV



If the prerequisites for HyperPAV are not there, base-PAV is used if the PAV feature is enabled on the storage server. Otherwise the DASD driver works without using PAV.

* With the old base-PAV support the mapping between base and alias devices was static.
* Now HyperPAV removes the requirement to dedicate alias devices to specific base devices.
* Instead each alias device can be combined with multiple base devices on a per request basis.
* HyperPAV is activated automatically when the necessary prerequisites are there (DS8000 with HyperPAV LIC, z/VM 5.3)
* SLES11, RHEL6

HyperPAV Setup

HyperPAV simplifies systems management and improves performance using an on demand I/O model

- HyperPAV Base and Alias subchannels are defined on control unit's Hardware Management Console and in IOCP no differently than traditional PAVs
- HyperPAV hardware, priced feature enables floating Alias function associated with the HyperPAV architecture for each LSS (logical control unit)
- Operating system host determines which LSS (logical control unit) is in HyperPAV vs. traditional PAV mode

(1) Setup PAV configuration on Storage Server

(2) zSeries storage configuration (IOCP)

(3) Basic DASD configuration

(4) That's it – nothing else to do.*No multipath configuration needed.No formatting / partitioning related pitfalls!*

11

z/VM HyperPAV configuration

* VM dedicated DASD support via CP ATTACH command or DEDICATE user directory statement
* VM Minidisk Support:
  - workload balancing for guest's that don't exploit HyperPAV
  - linkable full-pack minidisks for guests that do exploit HyperPAV
  - New CP DEFINE HYPERPAVALIAS command creates HyperPAV Alias minidisks for exploiting guests
  - z/VM, z/OS and Linux on System z are current exploiters of HyperPAV
  - Restricted to fullpack minidisks for exploiting guests; architecture change in the works.
* Use the Class B, CP QUERY PAV command to view the current HyperPAV Base and Alias subchannels along with their associated Pools.

PAV/HyperPAV Toolbox

We introduced new sysfs attributes:

```
[root@t6345057 sys]# ls -la  /sys/devices/css0/0.0.0003/0.0.4de0/
total 0
drwxr-xr-x 2 root root    0 2009-04-14 14:07 .
drwxr-xr-x 3 root root    0 2009-04-14 14:07 ..
-r--r--r-- 1 root root 4096 2009-04-15 12:36 alias
-r--r--r-- 1 root root 4096 2009-04-15 12:36 availability
-rw-r--r-- 1 root root 4096 2009-04-15 12:36 cmb_enable
-r--r--r-- 1 root root 4096 2009-04-14 14:07 cutype
-r--r--r-- 1 root root 4096 2009-04-15 12:36 devtype
-r--r--r-- 1 root root 4096 2009-04-15 12:36 discipline
lrwxrwxrwx 1 root root    0 2009-04-15 12:36 driver -> ../../../../bus/ccw/drive
rs/dasd-eckd
-rw-r--r-- 1 root root 4096 2009-04-15 12:36 eer_enabled
-rw-r--r-- 1 root root 4096 2009-04-15 12:36 erplog
-rw-r--r-- 1 root root 4096 2009-04-15 12:36 failfast
-r--r--r-- 1 root root 4096 2009-04-15 12:36 modalias
-rw-r--r-- 1 root root 4096 2009-04-15 12:36 online
-rw-r--r-- 1 root root 4096 2009-04-15 12:36 readonly
-r--r--r-- 1 root root 4096 2009-04-15 12:36 status
lrwxrwxrwx 1 root root    0 2009-04-15 12:36 subsystem -> ../../../../bus/ccw
-rw-r--r-- 1 root root 4096 2009-04-14 14:07 uevent
-r--r--r-- 1 root root 4096 2009-04-15 12:36 uid
-rw-r--r-- 1 root root 4096 2009-04-15 12:36 use_diag
-r--r--r-- 1 root root 4096 2009-04-15 12:36 vendor
```

'alias': 0 for base device, 1 for alias device

'uid': unique-id (vendor.serial.SSID.UA) of the physical (base) device

'vendor': vendor/manufacturer

dasdinfo is a new tool to support device-mapper setup

The DASD kernel module has a new parameter 'nopav', to disable PAV support.

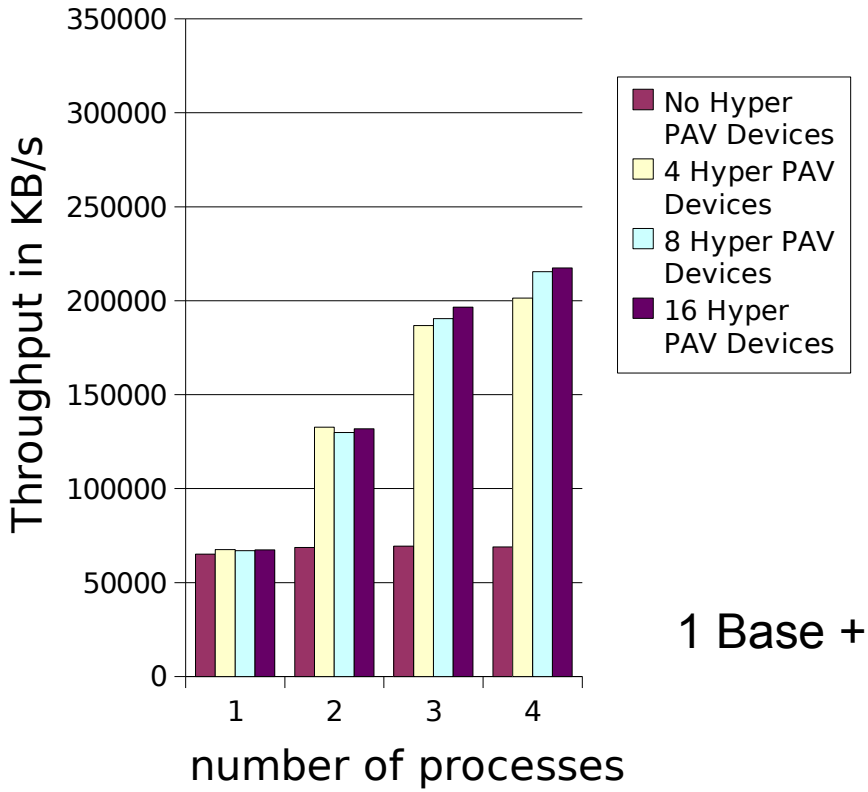
HyperPAV: LPAR and z/VM

* Alias devices can not be detected until at least one base device is set online
* z/VM V5.3 supports the Hyper Parallel Access Volume (HyperPAV) function optionally provided by the IBM System Storage DS8000™ disk storage systems.
* HyperPAV support complements the existing basic PAV support in z/VM V5.2, for applicable supporting disk storage systems.
* z/VM provides support of HyperPAV volumes as linkable minidisks.
* This support is also designed to transparently provide the potential benefits of HyperPAV volumes for minidisks owned or shared by guests that do not specifically exploit HyperPAV volumes, such as Linux and CMS.

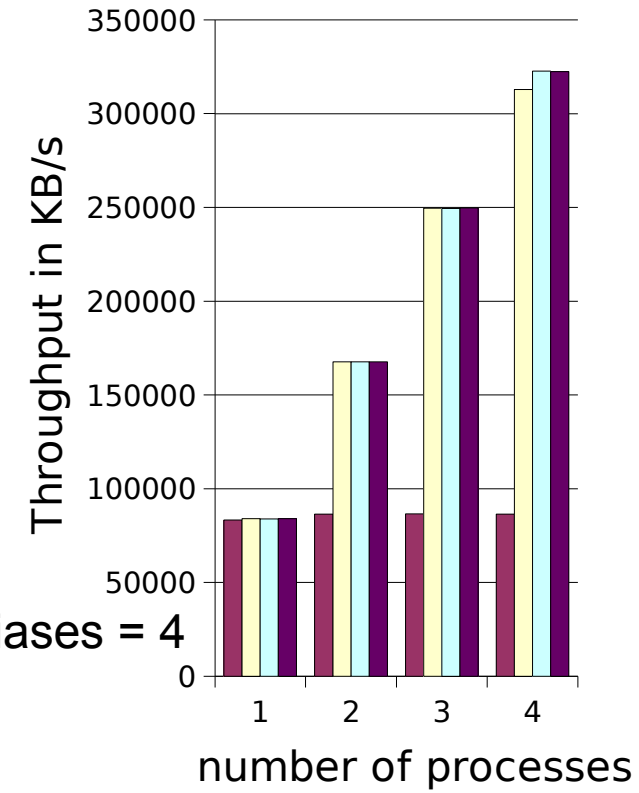
HyperPAV Performance

Single Disk Test – Sequential DIO - 700MB file size - 256MB Memory

Throughput for initial writers



Throughput for readers



1 Base + 3 aliases = 4

Effektive Parallelität, 4x mehr Reader = 4x mehr Throughput

PAV vs. HyperPAV

* PAV
  - Well defined mapping between base and alias devices.
  - This results in a predictable resource utilization
* HyperPAV
  - Dynamic allocation and utilization of resources via pooled alias devices
  - Advantage: In average this leads to a higher utilization, resulting in I/O transfer rates

If you run current Linux on System z distributions, prefer HyperPAV over PAV

Impact of DS8000 Storage Pool Striping

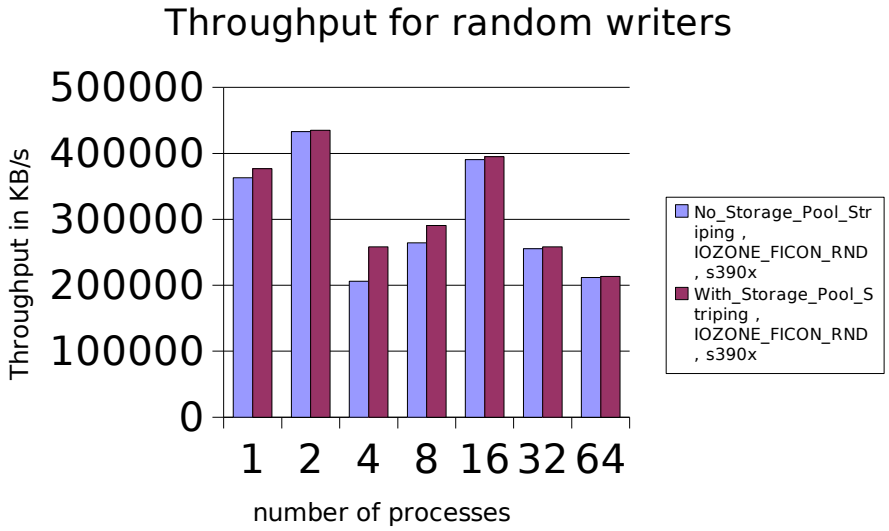
Starting with DS8000 License Machine Code 5.30xx.xx, it is possible to stripe the extents of a DS8000 Logical Volume across multiple RAID arrays.

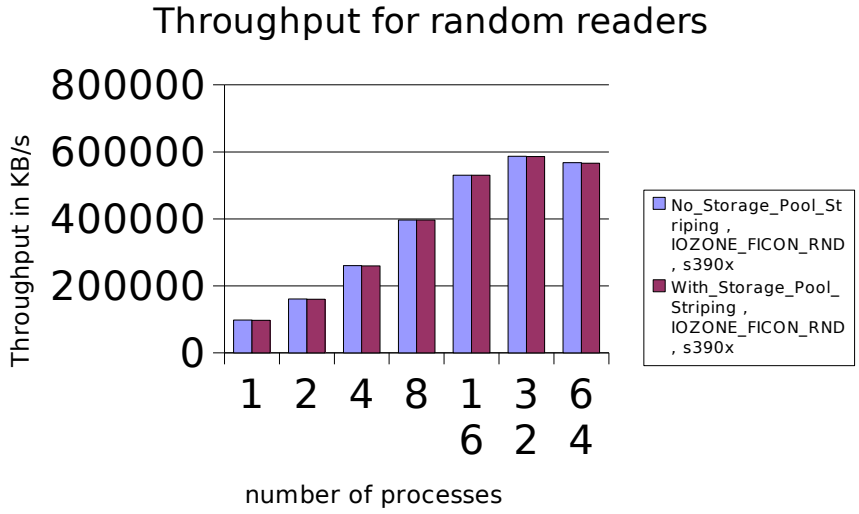
DS8000 Storage Pool Striping will improve throughput for some workloads.

It is performed on a 1 GB granularity, so it will generally benefit random workloads more than sequential ones.

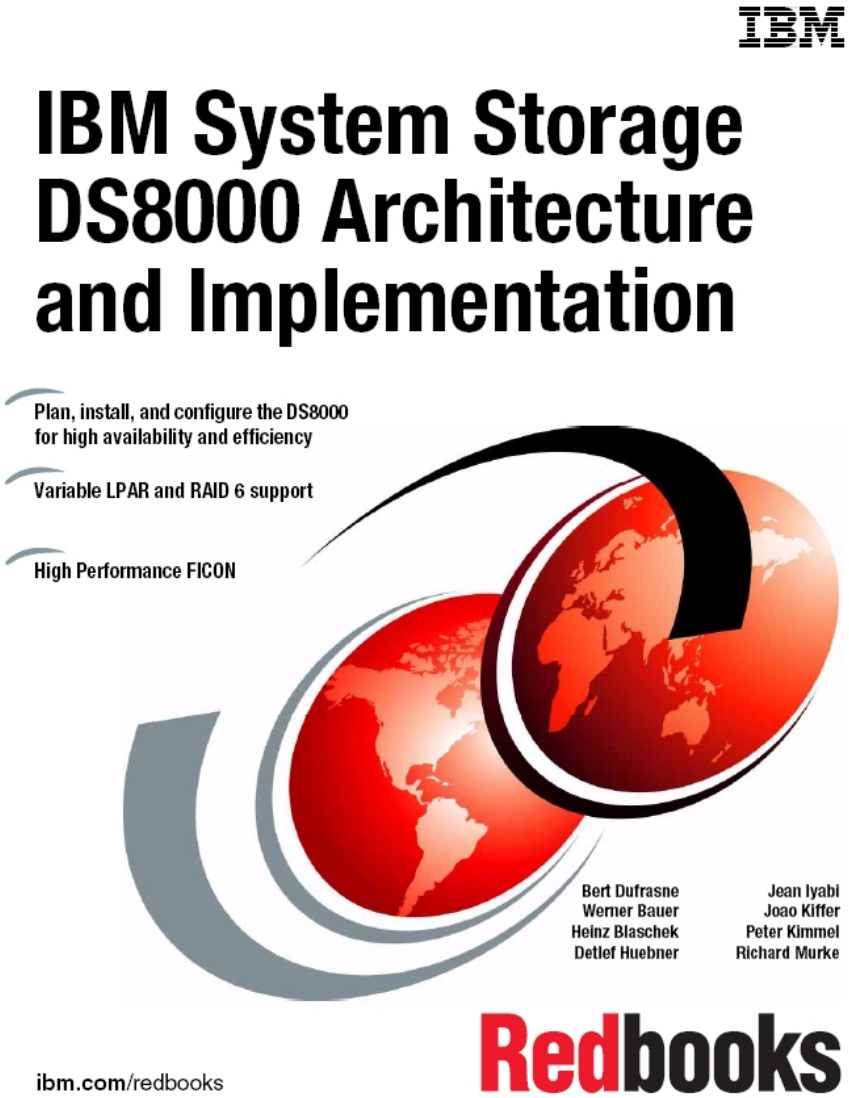
If you are already using a host-based striping method (for example, LVM Striping or DB2 database container striping), there is no need to use Storage Pool Striping.

However, it is possible. You should then combine the wide stripes on DS8000 with small granularity stripes on the host. The recommended size for these is usually between 8 and 64 MB. If large stripes on both DS8000 and attached host interfere with each other, I/O performance may be affected.

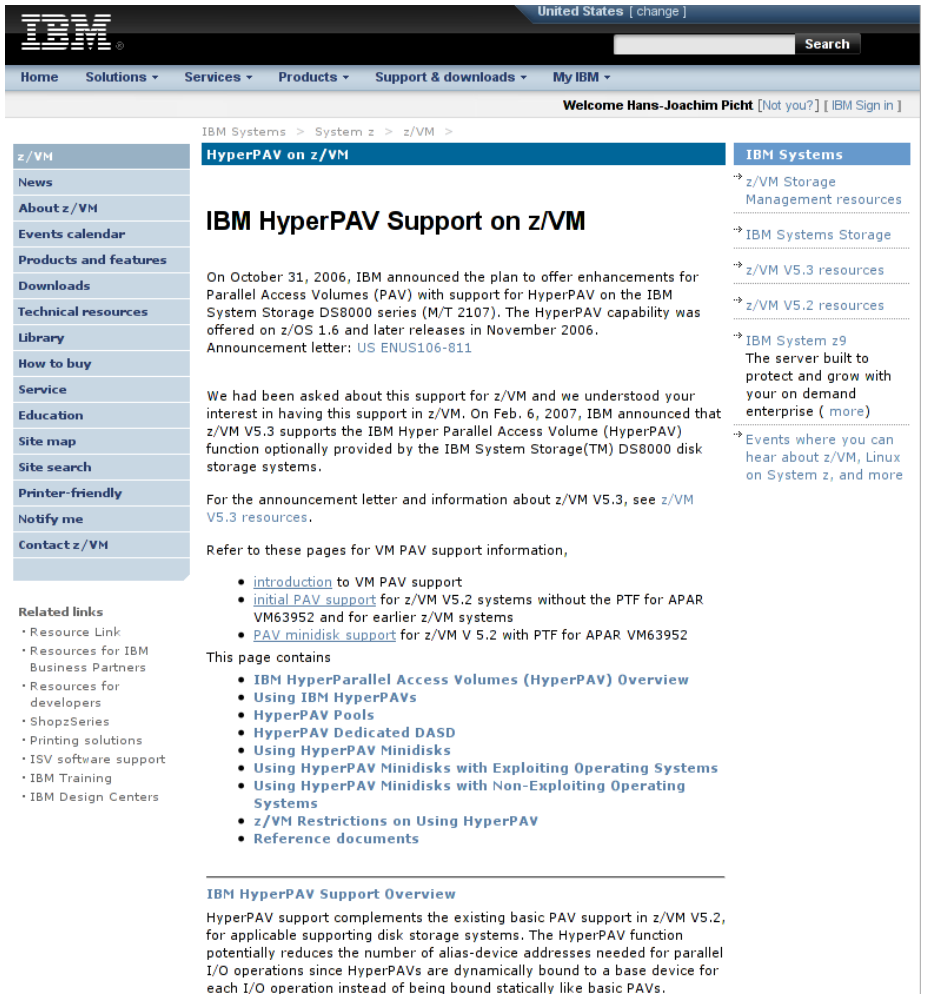




More information



http://www.vm.ibm.com/storman/pav/pav3.html

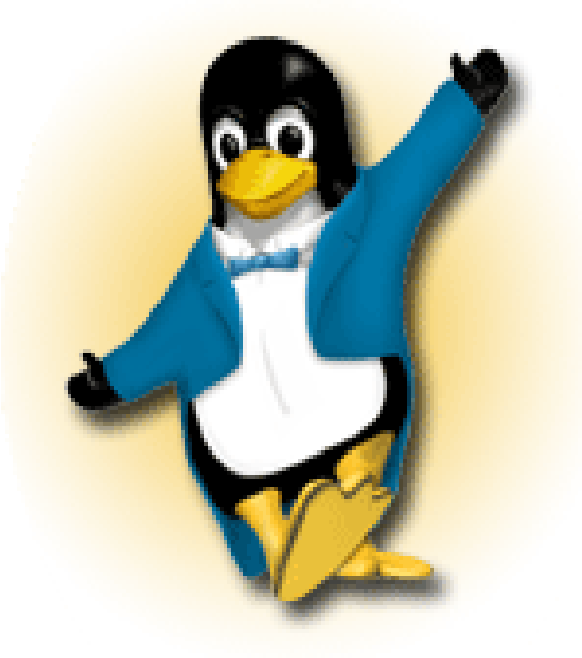


More information

ibm.com/systems/z/linux

www.vm.ibm.com

Questions?



Hans-Joachim Picht

Linux Technology Center

Linux on System z Kernel Development & Red Hat Liaision

IBM Deutschland Research & Development GmbH
Schönaicher Strasse 220
71032 Böblingen, Germany

Phone +49 (0)7031-16-1810
Mobile +49 (0)175 - 1629201
hans@linux.vnet.ibm.com