



IBM Linux and Technology Center

Current & Future Linux on System z Technology

Holger Smolinski
IBM Lab Böblingen, Germany
Charts courtesy of Martin Schwidefsky



Trademarks & Disclaimer

The following are trademarks of the International Business Machines Corporation in the United States and/or other countries. For a complete list of IBM Trademarks, see www.ibm.com/legal/copytrade.shtml: AS/400, DB2, e-business logo, ESCON, eServer, FICON, IBM, IBM Logo, iSeries, MVS, OS/390, pSeries, RS/6000, S/390, System Storage, System z9, VM/ESA, VSE/ESA, WebSphere, xSeries, z/OS, zSeries, z/VM.

The following are trademarks or registered trademarks of other companies

Java and all Java-related trademarks and logos are trademarks of Sun Microsystems, Inc., in the United States and other countries. LINUX is a registered trademark of Linux Torvalds in the United States and other countries. UNIX is a registered trademark of The Open Group in the United States and other countries. Microsoft, Windows and Windows NT are registered trademarks of Microsoft Corporation. SET and Secure Electronic Transaction are trademarks owned by SET Secure Electronic Transaction LLC. Intel is a registered trademark of Intel Corporation. * 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. References in this document to IBM products or services do not imply that IBM intends to make them available in every country. Any proposed use of claims in this presentation outside of the United States must be reviewed by local IBM country counsel prior to such use. The 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.

How Linux on System z is developed

How does the “community” work.

IBM collaborates with the Linux community

- has been an active participant since 1999
- is one of the leading commercial contributors to Linux
- has over 600 full-time developers working with Linux and open source

Linux Kernel & Subsystem Development

Kernel Base
Security
Systems Mgmt
Virtualization
Filesystems,
and more...

Expanding the Open Source Ecosystem

Apache
Eclipse
Mozilla Firefox
OpenOffice.org,
and more...

Promoting Open Standards & Community Collaboration

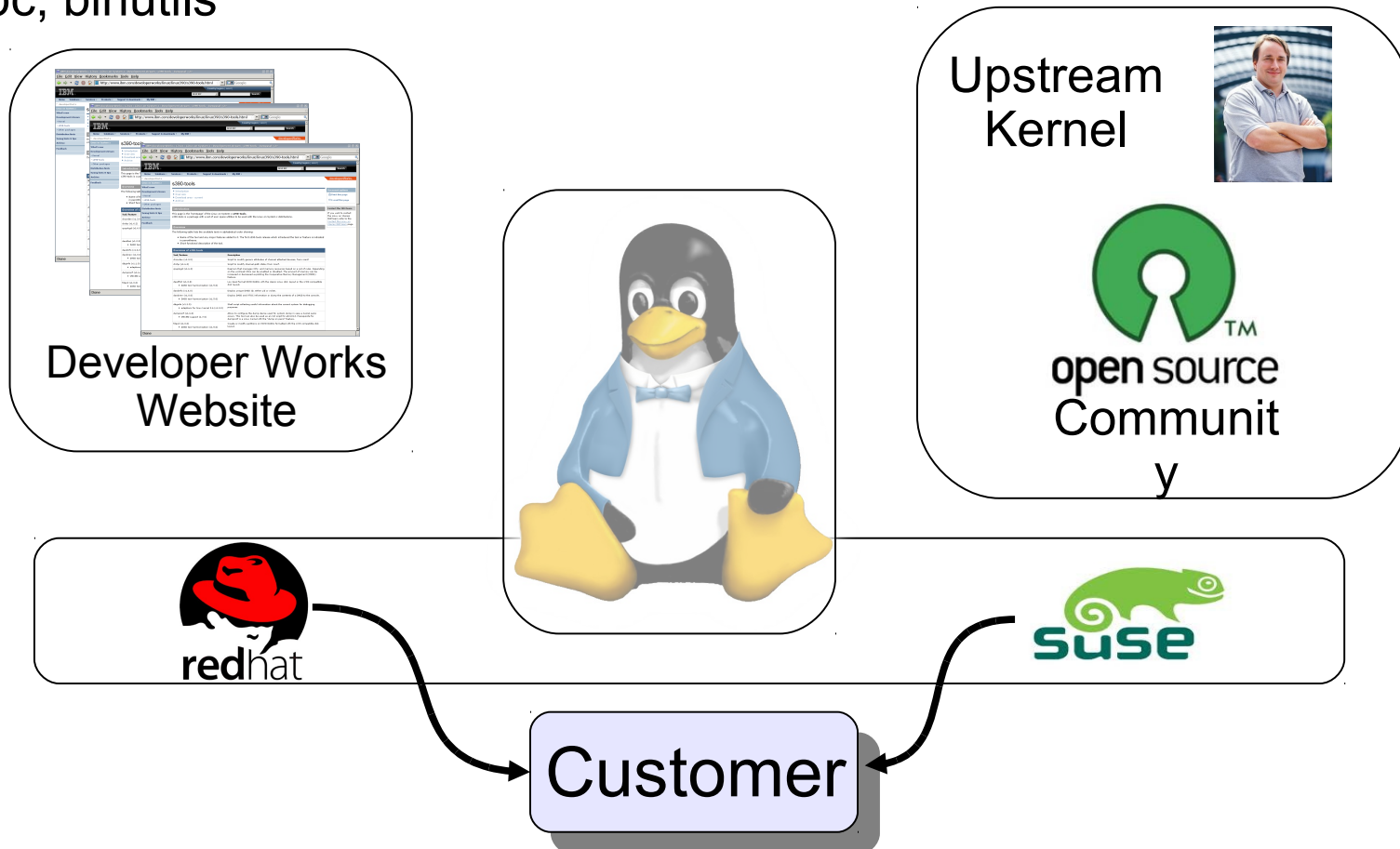
The Linux Foundation
Linux Standards Base
Common Criteria certification,
and more...

Foster and Protect the Ecosystem

Software Freedom Law Center
Free Software Foundation (FSF),
and more...

The IBM Linux development process

- IBM Linux on System z development contributes in the following areas: Kernel, s390-tools, open source tools (e.g. eclipse, ooprofile), gcc, glibc, binutils



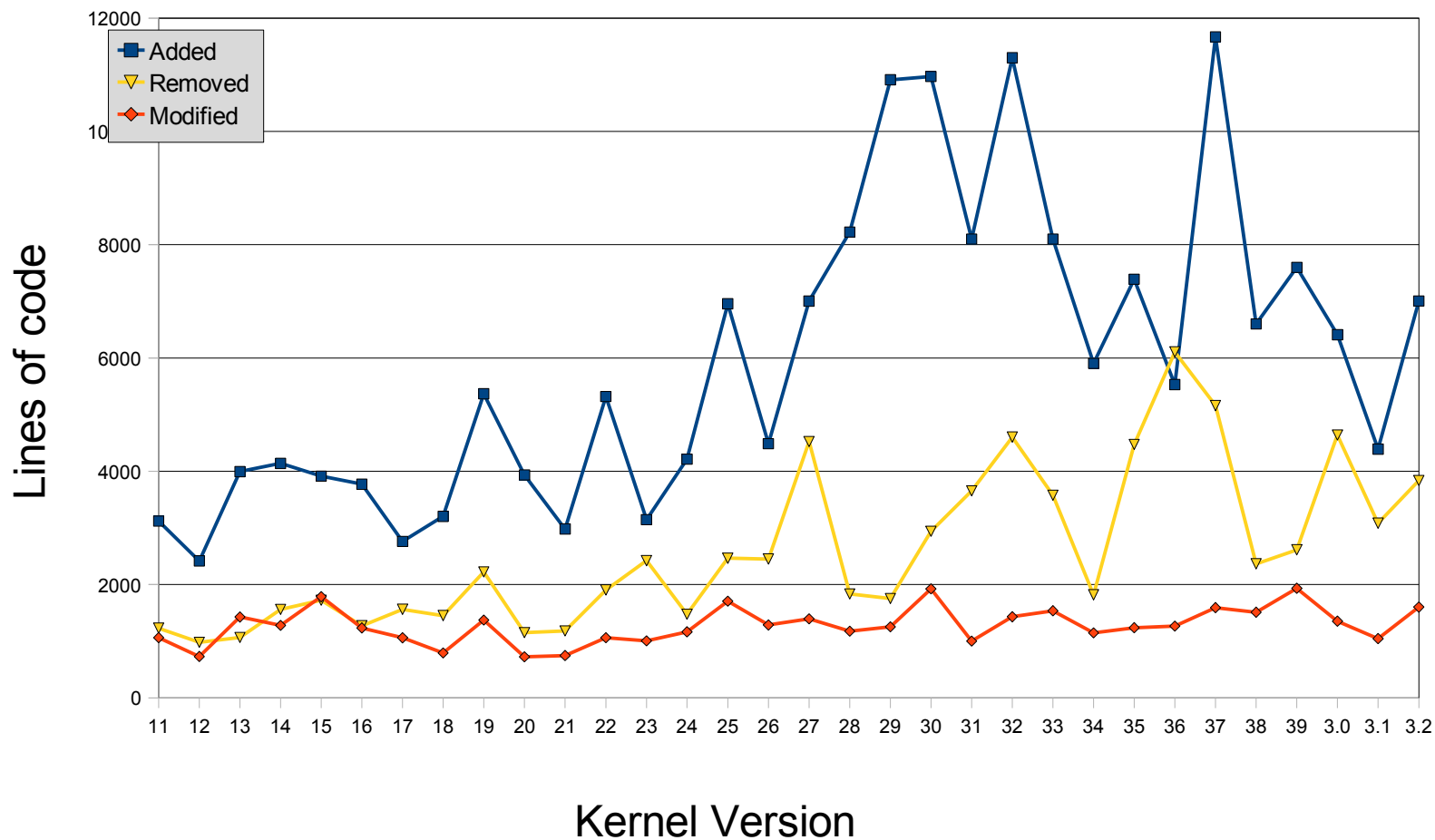
Facts on Linux

- Linux kernel 1.0.0 was released with 176,250 lines of code
How many lines of code has the kernel version 3.2 ?
14,998,737 lines of code
- How many of the world's top 500 supercomputers run Linux (Jan 2012)?
457 / 91.4%
- What percentage of web servers run Linux (Jan 2012) ?
63.6% run Unix, of those 51.6% run Linux (46.5% unknown) = 32.8%
- What percentage of desktop clients run Linux (Jan 2012) ?
1.6%
- What is the largest Linux architecture in number of devices ?
ARM, > 100 million activated android devices
- **Linux is Linux**, but ...features, properties and quality differ dependent on your platform and your use case

Source: <http://kernel.org>
<http://top500.org/stats>
<http://w3techs.com>
<http://www.w3counter.com>
<http://googleblog.blogspot.com/2011/05/android-momentum-mobile-and-more-at.html>

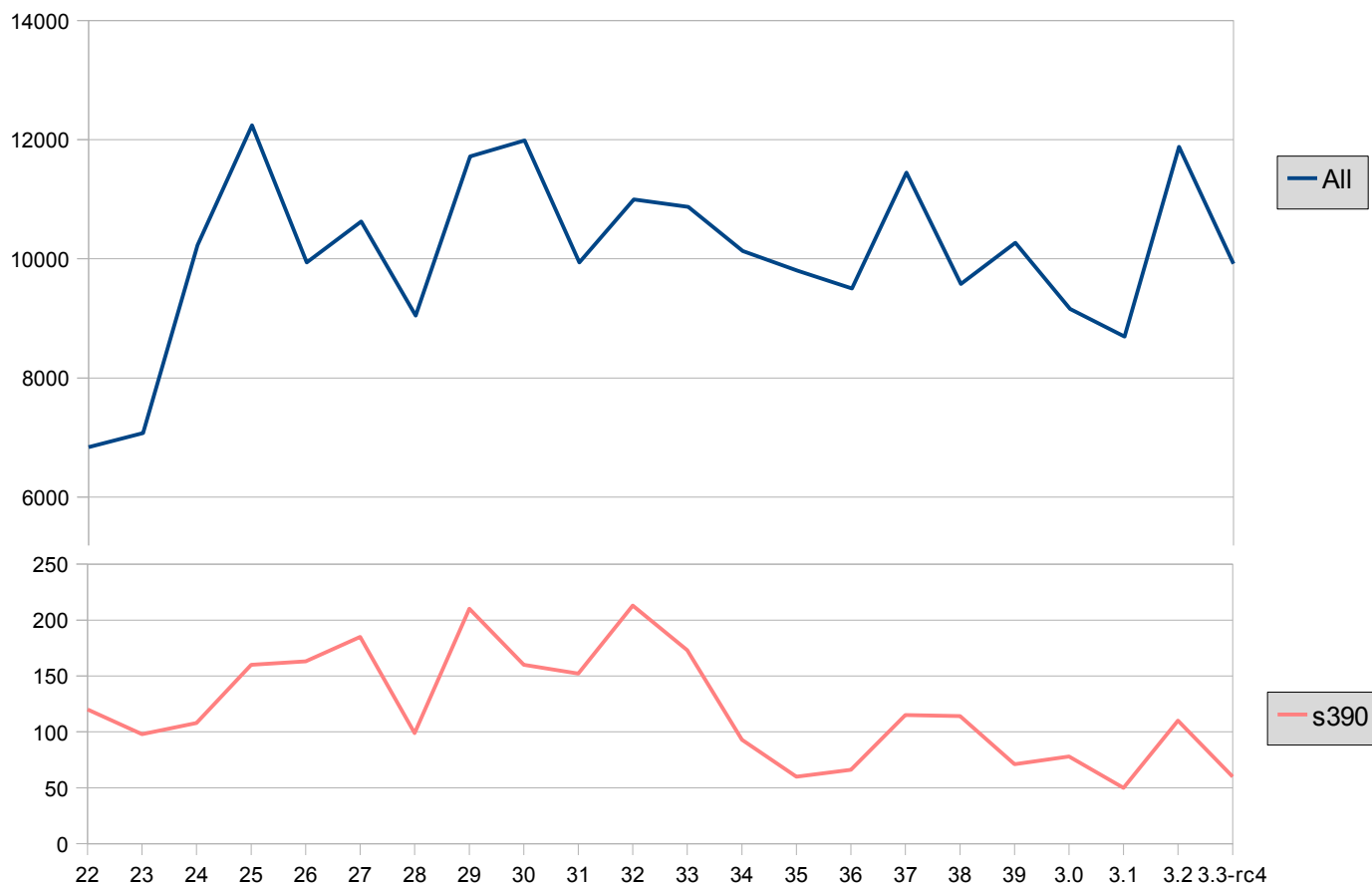
Linux kernel development: rate of change

Average for the last 7 years (without renames): 102 days per release, 5897 lines added, 2586 lines removed and 1221 lines modified **per day**



Linux kernel development: System z contributions

- Changesets per 2.6.x/3.x kernel release



Linux on System z distributions (Kernel 2.6 based)

- SUSE Linux Enterprise Server 9 (GA 08/2004)
 - Kernel 2.6.5, GCC 3.3.3, Service Pack 4 (GA 12/2007), end of regular life cycle
- SUSE Linux Enterprise Server 10 (GA 07/2006)
 - Kernel 2.6.16, GCC 4.1.0, Service Pack 4 (GA 05/2011)
- SUSE Linux Enterprise Server 11 (GA 03/2009)
 - Kernel 2.6.27, GCC 4.3.3, Service Pack 1 (GA 06/2010), Kernel 2.6.32
 - Kernel 3.0.13, GCC 4.3.4, Service Pack 2 (GA 02/2012)
- Red Hat Enterprise Linux AS 4 (GA 02/2005)
 - Kernel 2.6.9, GCC 3.4.3, Update 9 (GA 02/2011), end of regular life cycle
- Red Hat Enterprise Linux AS 5 (GA 03/2007)
 - Kernel 2.6.18, GCC 4.1.0, Update 8 (GA 02/2012)
- Red Hat Enterprise Linux AS 6 (GA 11/2010)
 - Kernel 2.6.32, GCC 4.4.0, Update 2 (GA 12/2011)
- Others
 - Debian, Slackware,
 - Support may be available by some third party

Supported Linux Distributions

Distribution	zEnterprise – z114 and z196	System z10	System z9	zSeries
RHEL 6	✓	✓	✓	X
RHEL 5	✓	✓	✓	✓
RHEL 4 (*)	✓ ⁽¹⁾	✓	✓	✓
SLES 11	✓	✓	✓	X
SLES 10	✓	✓	✓	✓
SLES 9 (*)	✓ ⁽²⁾	✓	✓	✓

Two options for zSeries machines



Indicates that the distribution (version) has been tested by IBM on the hardware platform, will run on the system, and is an IBM supported environment. Updates or service packs applied to the distribution are also supported.

(1) RHEL 4.8 only. Some functions have changed or are not available with the z196, e.g. the Dual-port OSA cards support to name one of several. Please check with your service provider regarding the end of service.

(2) SLES 9 SP4 + latest maintenance updates only. Some functions have changed or are not available with the z196, e.g. the Dual-port OSA cards support to name one of several. Please check with your service provider regarding the end of service.



Indicates that the distribution is not supported by IBM.



Also available as 31-bit distribution.

Current Linux on System z Technology

Features & Functionality contained in the SuSE
& Red Hat Distributions

System z kernel features – Core

- Improved QDIO performance statistics (kernel 2.6.33)



- Converts global statistics to per-device statistics and adds new counter for the input queue full condition

- Breaking event address for user space programs (kernel 2.6.35)



- Remember the last break in the sequential flow of instructions
- Valuable aid in the analysis of wild branches

- z196 enhanced node affinity support (kernel 2.6.37)



- Allows the Linux scheduler to optimize its decisions based on the z196 topology

- Performance indicator bytes (kernel 2.6.37)



- Display capacity adjustment indicator introduced with z196 via `/proc/sysinfo`

System z kernel features – Core

- QDIO outbound scan algorithm (kernel 2.6.38)
 - Improve scheduling of QDIO tasklets, OSA / HiperSockets / zfcg need different thresholds

- Enabling spinning mutex (kernel 2.6.38)
 - Make use of the common code for adaptive mutexes.
 - Add a new architecture primitive `arch_mutex_cpu_relax` to exploit sigp sense running to avoid the mutex lock retries if the hypervisor has not scheduled the cpu holding the mutex.





CMSFS user space file system support

- Allows to mount a z/VM minidisk to a Linux mount point
- z/VM minidisk needs to be in the enhanced disk format (EDF)
- The cmsfs fuse file system transparently integrates the files on the minidisk into the Linux VFS, no special command required

```
# cmsfs-fuse /dev/dasde /mnt/cms
# ls -la /mnt/fuse/PROFILE.EXEC
-r--r----- 1 root root 3360 Jun 26 2009 /mnt/fuse/PROFILE.EXEC
```

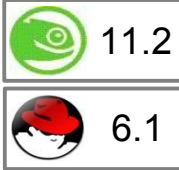
- By default no conversion is performed
 - Mount with '-t' to get automatic EBCDIC to ASCII conversion

```
# cmsfs-fuse -t /dev/dasde /mnt/cms
```

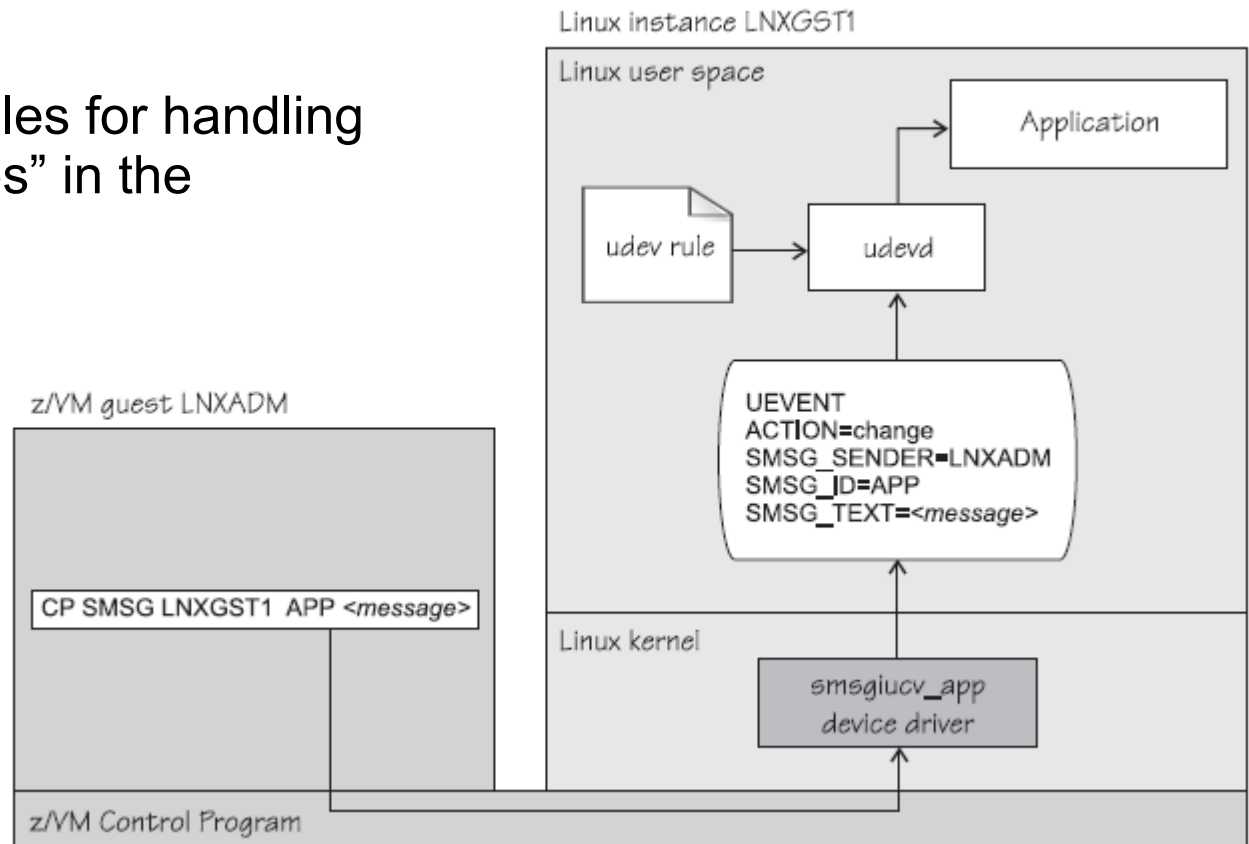
- Write support is work in progress, almost completed
 - use “vi” to edit PROFILE.EXEC anyone ?
- Use fusermount to unmount the file system again

```
# fusermount -u /mnt/cms
```

Deliver z/VM CP special messages as uevent



- Allows to forward SMSG messages to user space programs
 - Message needs to start with “APP”
- The special messages cause uevents to be generated
- See “Writing udev rules for handling CP special messages” in the Device Drivers Book



System z kernel features – Usability / RAS

- Dump on panic – prevent reipl loop (s390-tools 1.8.4)
 - Delay arming of automatic reipl after dump.
 - Avoids dumps loops where the restarted system crashes immediately.

- Add support for makedumpfile tool (kernel 2.6.34, s390-tools 1.9.0)
 - Convert Linux dumps to the ELF file format
 - Use the makedumpfile tool to remove user data from the dump.
 - Multi-volume dump will be removed.

- Address space randomization (kernel 2.6.38)
 - Enable flexible mmap layout for 64 bit to randomize start address for the runtime stack and the mmap area

- Get CPC name (kernel 2.6.39)
 - Useful to identify a particular hardware system in a cluster
 - The CPC name and the HMC network name are provided



6.1



11.2



6.1



11.2







11.2






11.2

System z kernel features – FICON

- Unit check handling (kernel 2.6.35)  6.1  11.2
 - Improve handling of unit checks for internal I/O started by the common-I/O layer
 - After a unit check certain setup steps need to be repeated, e.g. for PAV

- Dynamic PAV toleration (kernel 2.6.35)  6.1  11.2
 - Tolerate dynamic Parallel Access Volume changes for base PAV
 - System management tools can reassign PAV alias device to different base devices

- Tunable default grace period for missing interrupts in DASD (kernel 2.6.36)  6.1  11.2
 - Provide a user interface to specify the timeout for missing interrupts for standard I/O operations on DASD

- Query DASD reservation status (kernel 2.6.37)  11.2
 - New DASD ioctl to read the 'Sense Path Group ID' data
 - Allows to determine the reservation status of a DASD in relation to the current system

System z kernel features – FICON

- Multi-track extension for HPF (kernel 2.6.38)
 - Allows to read from and write to multiple tracks with a single CCW

- Access to raw ECKD data from Linux (kernel 2.6.38)
 - This item allows to access ECKD disks in raw mode
 - Use the 'dd' command to copy the disk level content of an ECKD disk to a Linux file, and vice versa.
 - Storage array needs to support read-track and write-full-track command.

- Automatic menu support in zipl (s390-tools 1.11.0)
 - Zipl option to create a boot menu for all eligible non-menu sections in zipl.conf

- reIPL from device-mapper devices (s390-tools 1.12.0)
 - The automatic re-IPL function only works with a physical device
 - Enhance the zipl support for device-mapper devices to provide the name of the physical device if the zipl target is located on a logical device



System z kernel features – FCP

- Store I/O and initiate logging (SIOSL) (kernel 2.6.36)



6.1



11.2

- Enhance debug capability for FCP attached devices
- Enables operating system to detect unusual conditions on a FCP channel

- Add NPIV information to symbolic port name (kernel 2.6.39)



11.2

- Add the device bus-ID and the network node to the symbolic port name if the NPIV mode is active.

- SAN utilities (kernel 2.6.36, lib-zfcp-hbaapi 2.1)



6.1



11.2

- Two new utilities have been added: zfcp_ping and zfcp_show
- They are useful to discover a storage area network

SAN Utilities: zfcplib



- Query Fiber Channel name server about ports available for my system:

```
# zfcplib -n
Local Port List:
    0x500507630313c562 / 0x656000 [N_Port] proto = SCSI-FCP  FICON
    0x50050764012241e4 / 0x656100 [N_Port] proto = SCSI-FCP
    0x5005076401221b97 / 0x656400 [N_Port] proto = SCSI-FCP
```

- Query SAN topology, requires FC management server access:

```
# zfcplib
Interconnect Element Name      0x100000051e4f7c00
Interconnect Element Domain ID 005
Interconnect Element Type      Switch
Interconnect Element Ports     256
  ICE Port 000  Online
    Attached Port [WWPN/ID] 0x50050763030b0562 / 0x650000 [N_Port]
  ICE Port 001  Online
    Attached Port [WWPN/ID] 0x50050764012241e5 / 0x650100 [N_Port]
  ICE Port 002  Online
    Attached Port [WWPN/ID] 0x5005076303008562 / 0x650200 [N_Port]
  ICE Port 003  Offline
  ...
```

SAN Utilities: zfcplib



- Check if remote port responds (requires FC management service access):

```
# zfcplib_ping 0x5005076303104562
Sending PNG from BUS_ID=0.0.3c00 speed=8 GBit/s
    echo received from WWPN (0x5005076303104562) tok=0 time=1.905 ms
    echo received from WWPN (0x5005076303104562) tok=1 time=2.447 ms
    echo received from WWPN (0x5005076303104562) tok=2 time=2.394 ms

----- ping statistics -----
min/avg/max = 1.905/2.249/2.447 ms
-----
```

- zfcplib_show and zfcplib_ping are part of the zfcplib-hbaapi 2.1 package:

<http://www.ibm.com/developerworks/linux/linux390/zfcplib-hbaapi-2.1.html>

System z kernel features – Networking

■ Offload outbound checksumming (kernel 2.6.35)



- Move calculation of checksum for non-TSO packets from the driver to the OSA network card

■ OSX/OSM CHPIDs for hybrid data network (kernel 2.6.35)



- The OSA cards for the zBX Blade Center Extension will have a new CHPID type
- Allows communication between zBX and Linux on System z

■ Toleration of optimized latency mode (kernel 2.6.35)




- OSA devices in optimized latency mode can only serve a small number of stacks / users. Print a helpful error message if the user limit is reached.
- Linux does not exploit the optimized latency mode



■ NAPI support for QDIO and QETH (kernel 2.6.36)






- Convert QETH to the NAPI interface, the “new” Linux networking API
- NAPI allows for transparent GRO (generic receive offload)

System z kernel features – Networking

- QETH debugging per single card (kernel 2.6.36)  11.2
 - Split some of the global QETH debug areas into separate per-device areas
 - Simplifies debugging for complex multi-homed configurations

- Support for assisted VLAN null tagging (kernel 2.6.37)  6.1  11.2
 - Close a gap between OSA and Linux to process null tagged frames correctly
 - z/OS may sent null-tagged frames to Linux

- New default qeth configuration values (kernel 2.6.39)  11.2
 - Receive checksum offload, generic receive offload & number of inbound buffers

- IPv6 support for the qetharp tool (kernel 2.6.38)  6.2  11.2
 - Extend the qetharp tool to provide IPv6 information in case of a layer 3 setup.
 - This is required for communication with z/OS via HiperSockets using IPv6.

System z kernel features – Networking

- Add OSA concurrent hardware trap (kernel 3.0)
 - To ease problem determination the qeth driver requests a hardware trace when the device driver or the hardware detect an error
 - Allows to correlate between OSA and Linux traces.
- Configuration tool for System z network devices (s390-tools 1.8.4)
 - Provide a shell script to ease configuration of System z network devices





znetconf network device configuration tool

- Allows to list, add, remove & configure System z network devices
- For example: list all potential network devices:

```
# znetconf -u
Device Ids                Type      Card Type  CHPID  Drv.
-----
0.0.f500,0.0.f501,0.0.f502 1731/01  OSA (QDIO)  00     qeth
0.0.f503,0.0.f504,0.0.f505 1731/01  OSA (QDIO)  01     qeth
```

- Configure device 0.0.f503

```
znetconf -a 0.0.f503
```

- Configure device 0.0.f503 in layer2 mode and portname "myport"

```
znetconf -a 0.0.f503 -o layer2=1 -o portname=myport
```

- Remove network device 0.0.f503

```
znetconf -r 0.0.f503
```

System z toolchain

zEnterprise 196 exploitation (gcc 4.6)

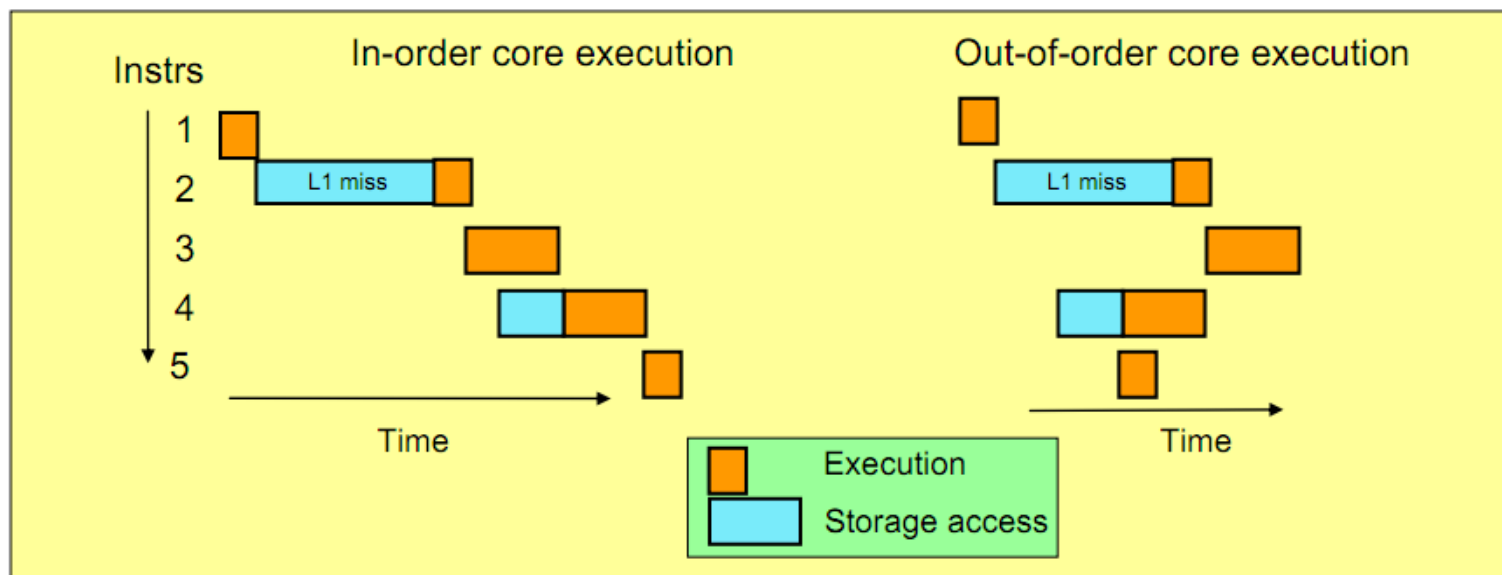


6.1



11.2

- Use option `-march=z196` to utilize the new instructions added with z196
- Use `-mtune=z196` to schedule the instruction appropriate for the new out-of-order pipeline of z196
- Re-compiled code/apps get further performance gains through 110+ new instructions



System z kernel features – Crypto

- 4096 bit RSA fast path (kernel 2.6.38)



- Make use of 4096 bit RSA acceleration available with Crypto Express 3 GA2 cards.

- CP ACF exploitation of System z196 (kernel 3.0)



- Add support for new HW crypto modes:
cipher feedback mode (CFB), output feedback mode (OFB),
counter mode (CTR), Galois counter mode (GCM),
XEX based Tweaked Code Book with Cipher Text Stealing (XTS),
cipher based message authentication mode (CMAC),
and counter with cipher block chaining message authentication (CCM)

- New libica APIs for supported crypto modes (libica 2.1.1)



- Provide a programmatic way to query for supported crypto ciphers, modes and key sizes.
- Deliver information whether the cryptographic features are implemented in hardware or in software

LNxHC – Linux Health Checker

- The Linux Health Checker is a command line tool for Linux.
- Its purpose is to identify potential problems before they impact your system's availability or cause outages.
- It collects and compares the active Linux settings and system status for a system with the values provided by health-check authors or defined by you. It produces output in the form of detailed messages, which provide information about potential problems and the suggested actions to take.
- The Linux Health Checker will run on any Linux platform which meets the software requirements. It can be easily extended by writing new health check plug-ins.
- The Linux Health Checker is an open source project sponsored by IBM. It is released under the Eclipse Public License v1.0
- <http://lnxhc.sourceforge.net/>

Future Linux on System z Technology

Software which has already been developed and integrated into the upstream Linux Kernel
- but is **not** yet available in any Enterprise Linux Distribution

Kernel news – Common code

Linux version 3.0 (2011-07-21)

- New kernel version numbering scheme
- Cleancache (was transcendent memory) support for ext4, btrfs and XFS
- Preemptible mmu_gather for reduced latency
- Enhancements for the memory cgroup controller

Linux version 3.1 (2011-10-24)

- New architecture: OpenRISC
- Dynamic writeback throttling
- Slab allocator speedups
- VFS scalability improvements
- New iSCSI implementation
- Software RAID: Bad block management

Linux version 3.2 (2012-01-04)

- New architecture: Hexagon
- btrfs improvements:
 - faster scrubbing
 - automatic backup of tree roots
- ext4: support for bigger block sizes up to 1MB
- Process bandwidth controller
- I/O-less dirty throttling
 - reduce file system write-back from page reclaim
- TCP Proportional Rate Reduction

System z kernel features – Core

- Add support for physical memory > 4TB (kernel 3.3)
 - Increase the maximum support memory size from 4TB to 64TB.
- Two stage dumper / kdump support (kernel 3.2, s390-tools-1.17.0)
 - Use a Linux kernel to create a system dump
 - Use a preloaded crashkernel to run in case of a system failure
 - Can be triggered either as panic action or by the stand-alone dumper, integrated into the shutdown actions framework
 - Pro
 - Enhanced dump support that is able to reduce dump size, shared disk space, dump to network, dump to a file-system etc.
 - The makedumpfile tool can be used to filter the memory of the crashed system
 - Con
 - kdump is not as reliable as the stand-alone dump tools
 - kdump cannot dump a z/VM named saved system (NSS)
 - For systems running in LPAR kdump consumes memory

Two stage dumper / kdump support

- Add a `crashkernel=` parameter to the kernel parameter

```
crashkernel=<size>@<offset>
```

- Boot your system and check the reservation

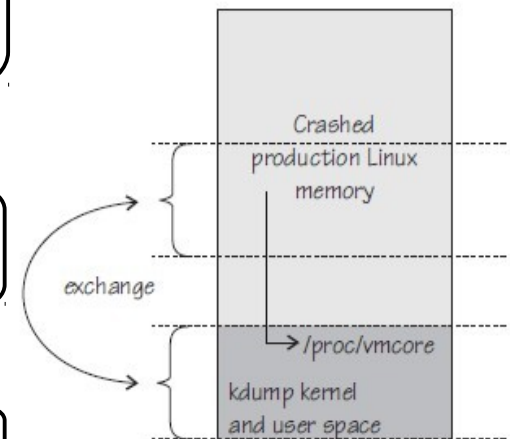
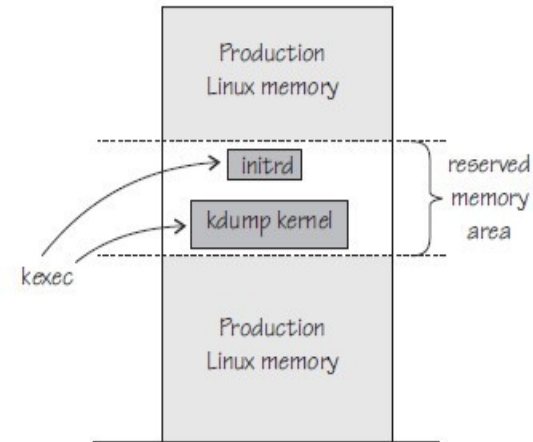
```
# cat /proc/iomem
00000000-3fffffff : System RAM
  00000000-005f1143 : Kernel code
  005f1144-00966497 : Kernel data
  00b66000-014c4e9f : Kernel bss
40000000-47fffffff : Crash kernel
48000000-7fffffff : System RAM
```

- Load the kdump kernel with `kexec`

```
# kexec -p kdump.image --initrd kdump.initrd
--command-line="dasd=1234 root=/dev/ram0"
```

- Manually trigger for kdump under z/VM

```
#cp system restart
```



System z kernel features – Storage FICON

- DASD sanity check to detect path connection errors (kernel 3.3)
 - An incorrect physical connection between host and storage server which is not detected by hardware or microcode can lead to data corruption
 - Add a check in the DASD driver to make sure that each available channel path leads to the same storage server

- Extended DASD statistics (kernel 3.1)
 - Add detailed per-device debugging of DASD I/Os via debugfs
 - Useful to analyze problems in particular for PAV and HPF

Extended DASD statistics

- Start data collection

```
# dasdstat -e dasda 0.0.1234
```

- Reset statistics counters

```
# dasdstat -r dasda
```

- Read summary statistics

```
# dasdstat
statistics data for statistic: 0.0.6527
start time of data collection: Fri Feb 24 16:00:19 CET 2012

1472 dasd I/O requests
with 14896 sectors(512B each)
0 requests used a PAV alias device
0 requests used HPF
  __<4  __8  __16  __32  __64  _128  _256  _512  __1k  __2k  __4k  __8k  _16k  _32k  _64k  128k
  _256  _512  __1M  __2M  __4M  __8M  _16M  _32M  _64M  128M  256M  512M  __1G  __2G  __4G  _>4G
Histogram of sizes (512B secs)
  0    0 1441    8   13    5    2    2    0    1    0    0    0    0    0    0
  0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0
Histogram of I/O times (microseconds)
  0    0    0    0    0    0    1 1160   49   52   61  142    7    0    0    0
  0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0
```

System z kernel features – Storage FCP

- FICON Express8S hardware data router support for FCP (kernel 3.2)
 - FICON Express8S supports hardware data router, which requires an adapted qdio request format.
 - Improves performance by reducing the path length for data.

- FCP support for DIF/DIX (kernel 3.2)
 - End to end data checking (aka data integrity extension) is no longer experimental.
 - Can be used with either direct I/O or with a file system that fully supports end-to-end data consistency checking. Currently XFS only.

- SCSI device management tool (> s390-tools 1.14.0)
 - Implement a tool analog chccwdev which allows to enable/disable a SCSI LUN addressed by HBA/target port/LUN.

System z kernel features – Networking

- Add support for AF_IUCV HiperSockets transport (kernel 3.2)
 - Use HiperSockets with completion queues as transport channel for AF_IUCV sockets

- Allow multiple paths with netiucv between z/VM guests (kernel 3.3)
 - Speed up netiucv by using parallel IUCV paths.

System z toolchain

- 64 bit register in 31 bit compat mode (gcc 4.6)
 - Make use of 64 bit registers in 31 bit application running in z/Architecture mode.
 - Allows to use instruction operating on 64 bits, e.g. 64 bit multiplication
 - Needs kernel support for asynchronous signals

- ATLAS support (libatlas 3.9.52)
 - Add support for System z to the “Automatically Tuned Linear Algebra Software”
 - Improve performance of the library functions for System z

System z application development tools

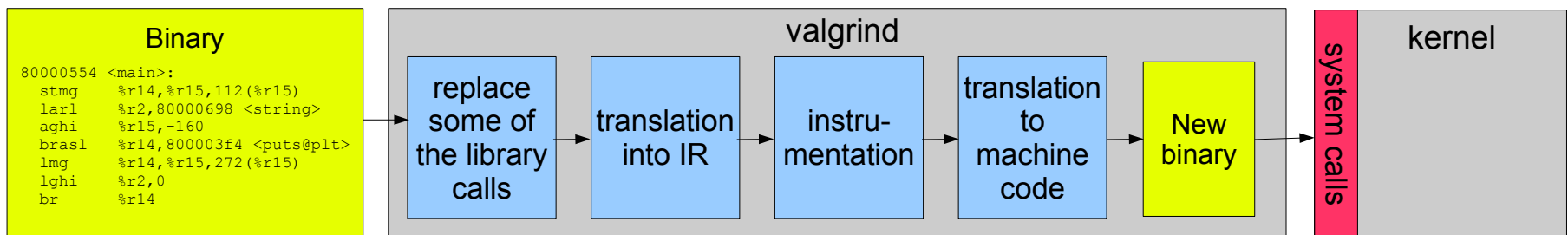
- Oprofile support for hardware sampling introduced with z10 (2.6.39)
 - Provide CPU measurement data to applications for performance tuning
 - Based on hardware counters and samples built into the CPU
 - Use oprofile to communicate the information to user space programs

- Oprofile z196 hardware customer mode sampling (kernel 3.3)
 - Extend the hardware sampling to support z196.

- Valgrind System z support
 - Valgrind is a generic framework for creating dynamic analysis tools and can be used for memory debugging, memory leak detection and profiling (e.g. cachegrind)
 - Valgrind is in essence a virtual machine using just-in-time (JIT) compilation techniques
 - Memory debugging is available with Valgrind version 3.7.0

Valgrind System z support

- `valgrind --tool=memcheck [--leak-check=full] [--track-origins] <program>`
 - Detects if your program accesses memory it shouldn't
 - Detects dangerous uses of uninitialized values on a per-bit basis
 - Detects leaked memory, double frees and mismatched frees
- `valgrind --tool=cachegrind`
 - Profile cache usage, simulates instruction and data cache of the cpu
 - Identifies the number of cache misses
- `valgrind --tool=massif`
 - Profile heap usage, takes regular snapshots of program's heap
 - Produces a graph showing heap usage over time



s390-tools package: what is it?

- s390-tools is a package with a set of user space utilities to be used with the Linux on System z distributions.
 - It is **the** essential tool chain for Linux on System z
 - It contains everything from the boot loader to dump related tools for a system crash analysis .
- This software package is contained in all major (and IBM supported) enterprise Linux distributions which support s390
 - RedHat Enterprise Linux 4
 - RedHat Enterprise Linux 5
 - RedHat Enterprise Linux 6
 - SuSE Linux Enterprise Server 9
 - SuSE Linux Enterprise Server 10
 - SuSE Linux Enterprise Server 11
- Website:
<http://www.ibm.com/developerworks/linux/linux390/s390-tools.html>
- Feedback: linux390@de.ibm.com

s390-tools package: the content

chccwdev
 chchp
 chreipl
 chshut
 chcrypt
 chmem

CHANGE

dasdfmt
 dasdinfo
dasdstat
 dasdview
 fdasd
 tunedasd

DASD

dbginfo
 dumpconf
 zfcpdump
 zfcpdbf
 zgetdump
 scsi_logging_level

DUMP
 &
 DEBUG

lscss
 lschp
 lsdasd
 lsluns
 lsqeth
 lsreipl
 lsshut
 lstape
 lszcrypt
 lszfcp
 lsmem

DISPLAY

mon_fsstatd
 mon_procd
 ziomon
 hyptop

MONITOR

vmconvert
 vmcp
 vmur
 cms-fuse

z/VM

ip_watcher
 osasmpd
 qetharp
 qethconf

NETWORK

cpuplugd
 iucvconn
 iucvty
 ts-shell
 ttyrun

MISC

tape390_display
 tape390_crypt

TAPE

zipl

BOOT

s390-tools package

- **Version 1.13.0 (2011-01-27)**
 - hyptop: Provides real-time view of System z hypervisor environment
 - cio_ignore: Add query option
 - cmsfs-fuse: Configurable code page conversion
 - tunedasd: Add option to query reservation status of a device
 - zgetdump: Add kdump support for –info option
 - zfcpdump/zipl: Disable automatic activations of LUNs
- **Version 1.13.0 (2011-05-19)**
 - qetharp: Support IPv6 for query ARP cache for HiperSockets
 - zfcpdbf: Adjust to 2.6.38 zfc driver changes
- **Version 1.14.0 (2011-06-30)**
 - fdasd: Implement new partition types “Linux raid” and “Linux LVM”
- **Version 1.15.0 (2011-08-31)**
 - cpuplugd: improved controls for the cmm memory balloon
- **Version 1.16.0 (2011-11-30)**
 - dasdstat: new tool to configure and format the debugfs based DASD statistics

hyptop: Display hypervisor utilization data

- The hyptop command is a top-like tool that displays a dynamic real-time view of the hypervisor environment
 - It works with both the z/VM and the LPAR hypervisor
 - Depending on the available data it can display information about CPU and memory
 - running LPARs or z/VM guest operating systems

- The following is required to run hyptop:
 - The debugfs file system must be mounted
 - The hyptop user must have read permission for the required debugfs files:
 - z/VM: <debugfs mount point>/s390_hypfs/diag_2fc
 - LPAR: <debugfs mount point>/s390_hypfs/diag_204
 - To monitor all LPARs or z/VM guests your instance requires additional privileges
 - For z/VM: The user ID requires privilege class B
 - For LPAR: The global performance data control box in the LPAR activation profile needs to be selected

hyptop: Display hypervisor utilization data

- Example of z/VM utilization data

```
10:11:56 CPU-T: UN(16) ?=help
```

system (str)	#cpu (#)	cpu (%)	Cpu+ (hm)	online (dhm)	memuse (GiB)	memmax (GiB)	wcur (#)
T6360003	6	<u>506.92</u>	3404:17	44:20:53	7.99	8.00	100
T6360017	2	<u>199.58</u>	8:37	29:23:50	0.75	0.75	100
T6360004	6	<u>99.84</u>	989:37	62:00:00	1.33	2.00	100
T6360005	2	<u>0.77</u>	0:16	5:23:06	0.55	2.00	100
T6360015	4	<u>0.15</u>	9:42	18:23:04	0.34	0.75	100
T6360035	2	<u>0.11</u>	0:26	7:18:15	0.77	1.00	100
T6360027	2	<u>0.07</u>	2:53	62:21:46	0.75	0.75	100
T6360049	2	<u>0.06</u>	1:27	61:17:35	0.65	1.00	100
T6360010	6	<u>0.06</u>	5:55	61:20:56	0.83	1.00	100
T6360021	2	<u>0.06</u>	1:04	48:19:08	0.34	4.00	100
T6360048	2	<u>0.04</u>	0:27	49:00:51	0.29	1.00	100
T6360016	2	<u>0.04</u>	6:09	34:19:37	0.30	0.75	100
T6360008	2	<u>0.04</u>	3:49	47:23:10	0.35	0.75	100
T6360006	2	<u>0.03</u>	0:57	25:20:37	0.54	1.00	100
NSLCF1	1	<u>0.01</u>	0:02	62:21:46	0.03	0.25	100
VTAM	1	<u>0.00</u>	0:01	62:21:46	0.01	0.03	100
T6360023	2	<u>0.00</u>	0:04	6:21:20	0.46	0.75	100
PERFSVM	1	<u>0.00</u>	2:12	7:18:04	0.05	0.06	0
AUTOVM	1	<u>0.00</u>	0:03	62:21:46	0.00	0.03	100
FTPSERVE	1	<u>0.00</u>	0:00	62:21:47	0.01	0.03	100
TCPIP	1	<u>0.00</u>	0:01	62:21:47	0.01	0.12	3000
DATAMOVE	1	<u>0.00</u>	0:06	62:21:47	0.00	0.03	100
VMSEVRU	1	<u>0.00</u>	0:00	62:21:47	0.00	0.03	1500
OPERSVMP	1	<u>0.00</u>	0:00	62:21:47	0.00	0.03	100

hyptop: Display hypervisor utilization data

- Example of single LPAR utilization data

```

10:16:59 H05LP30 CPU-T: IFL(18) CP(3) UN(2)                                     ?=help
cpuid  type      cpu  mgm  visual
( # )   (str)    (%)  (%)  (vis)
0_____ IFL   29.34 0.72 |#####
1_____ IFL   28.17 0.70 |#####
2_____ IFL   32.86 0.74 |#####
3_____ IFL   31.29 0.75 |#####
4_____ IFL   32.86 0.72 |#####
5_____ IFL   30.94 0.68 |#####
6_____ IFL    0.00 0.00 |
7_____ IFL    0.00 0.00 |
8_____ IFL    0.00 0.00 |
9_____ IFL    0.00 0.00 |
=:V:N          185.46 4.30

```

More information

The screenshot shows the IBM Linux on System z documentation website. The main heading is "Documentation for Development stream". The page is organized into sections: "Development stream" (with links to Novell SUSE and Red Hat), "Introduction", "Base documentation", "How to documents", and "Reference documentation".

Development stream | Novell SUSE | Red Hat

- Introduction
- Linux on System z documentation for 'Development stream'
- General Linux on System z documentation
- Documentation for IBM System z

Introduction

This page contains links to IBM documentation applicable to the Linux on System z 'Development stream'. The 'Documentation'-tab of the 'Development stream' has the same information as this page.

Linux on System z documentation for 'Development stream'

Base documentation

Device Drivers, Features, and Commands (kernel 2.6.33) - SC33-8411-05 (PDF, 4.4MB)	March 2010
Using the Dump Tools (kernel 2.6.33) - SC33-8412-04 (PDF, 0.6MB)	March 2010

How to documents

How to Improve Performance with PAV - SC33-8414-00 (PDF, 0.1MB)	May 2008
How to use FC-attached SCSI devices with Linux on System z (kernel 2.6.33) - SC33-8413-04 (PDF, 1.0MB)	March 2010
How to use Execute-in-Place Technology with Linux on z/VM - SC34-2594-01 (PDF, 0.5MB)	March 2010
Download a tarball with sample scripts.	
How to Set up a Terminal Server Environment - SC34-2596-00 (PDF, 0.3MB)	June 2009

Reference documentation

Kernel Messages (Kernel 2.6.33) (PDF, 0.4MB)	March 2010
libica Programmer's Reference - SC34-2602-00 (PDF, 0.3MB)	June 2009

Linux on System z

How to use Execute-in-Place Technology with Linux on z/VM

March, 2010

Linux on System z

How to use FC-attached SCSI devices with Linux on System z

Linux on System z

How to Set up a Terminal Server Environment on z/VM

June 2009

Linux Kernel 2.6 - Development stream

Linux on System z

Using the Dump Tools

Development stream (Kernel 26.33)

Linux on System z

Kernel Messages

Development stream (Kernel 26.33)

Linux on System z

Device Drivers, Features, and Commands

Development stream (Kernel 26.33)

New Redbooks



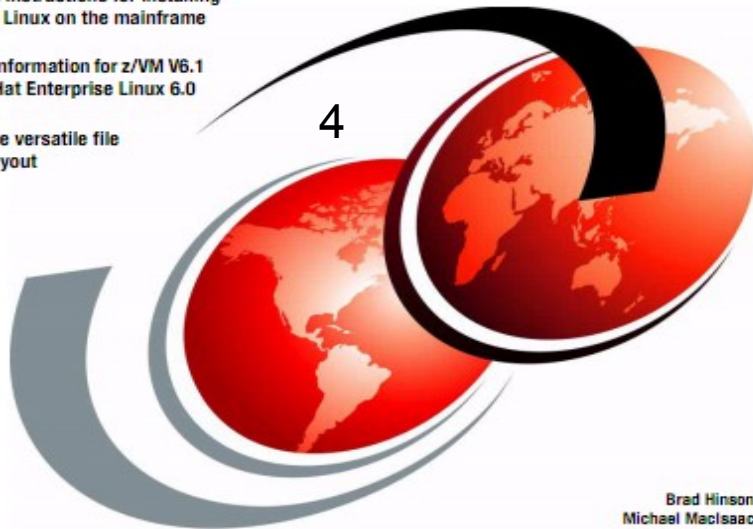
z/VM and Linux on IBM System z The Virtualization Cookbook for Red Hat Enterprise Linux 6.0

Hands-on instructions for installing z/VM and Linux on the mainframe

Updated information for z/VM V6.1 and Red Hat Enterprise Linux 6.0

New, more versatile file system layout

4



Brad Hinson
Michael MacIsaac

Redbooks

ibm.com/redbooks

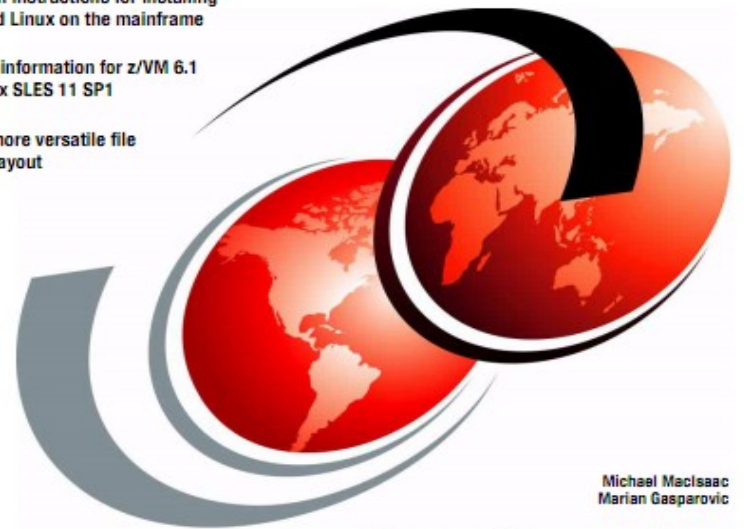


z/VM and Linux on IBM System z The Virtualization Cookbook for SLES 11 SP1

Hands-on instructions for installing z/VM and Linux on the mainframe

Updated information for z/VM 6.1 and Linux SLES 11 SP1

A new, more versatile file system layout



Michael MacIsaac
Marian Gasparovic

Redbooks

ibm.com/redbooks

Visit <http://www.redbooks.ibm.com>

Questions?



Holger Smolinski

Certified IT Specialist

Linux on System z

*Schönaicher Strasse 220
71032 Böblingen, Germany*

*Phone +49 (0)7031-16-4652
Holger.Smolinski@de.ibm.com*