What's New in Linux on System z

Martin Schwidefsky
IBM Lab Böblingen, Germany
March 14, 2012, Session 10324

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-BM 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 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	Expanding the Open Source Ecosystem	Promoting Open Standards & Community Collaboration	Foster and Protect the Ecosystem
Kernel Base Security Systems Mgmt Virtualization Filesystems, and more...	Apache Eclipse Mozilla Firefox OpenOffice.org, and more...	The Linux Foundation Linux Standards Base Common Criteria certification, and more...	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

Developer Works Website

Upstream Kernel

**open** source ™
Community

redhat

suse

Customer
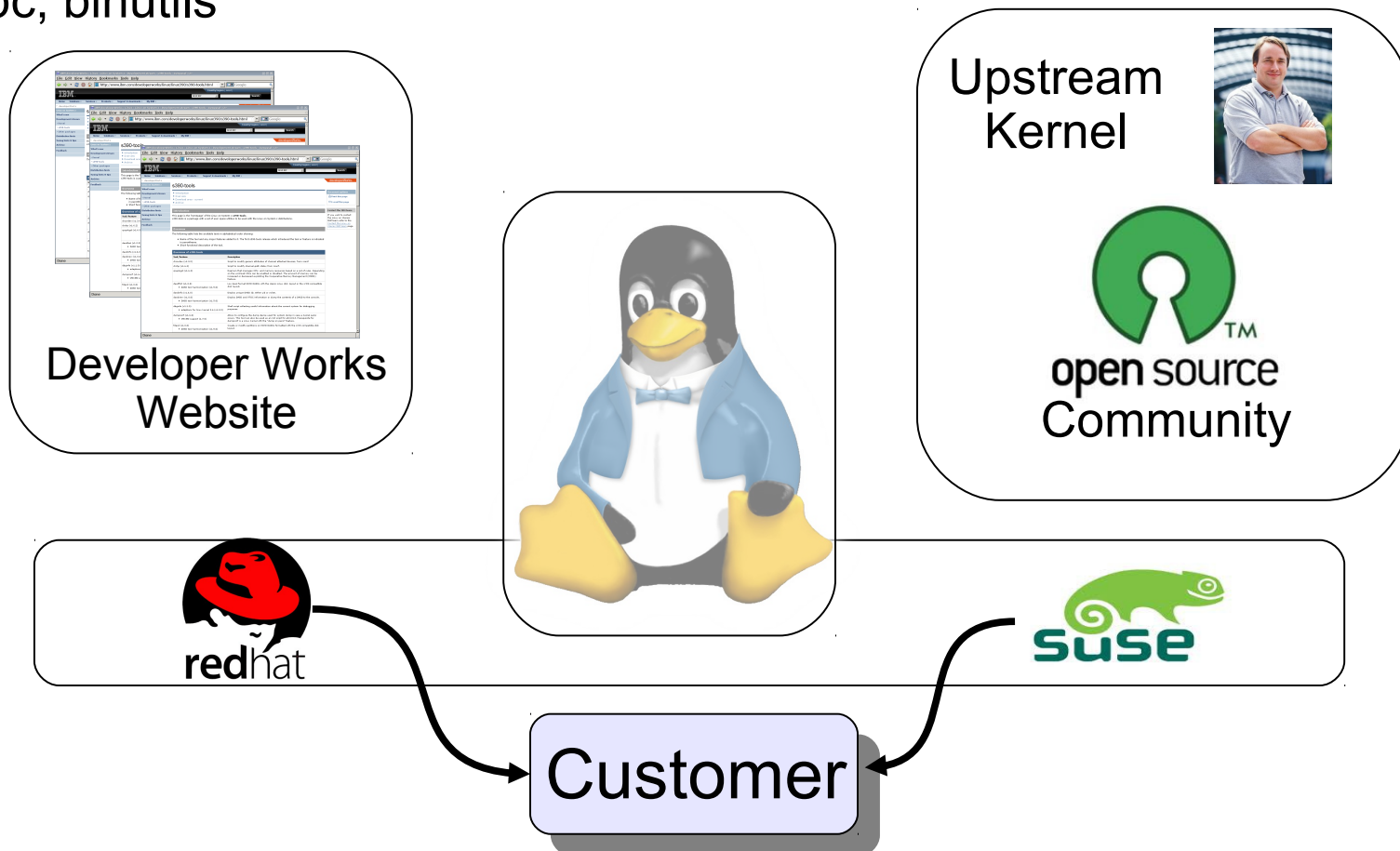
Facts on Linux

- 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
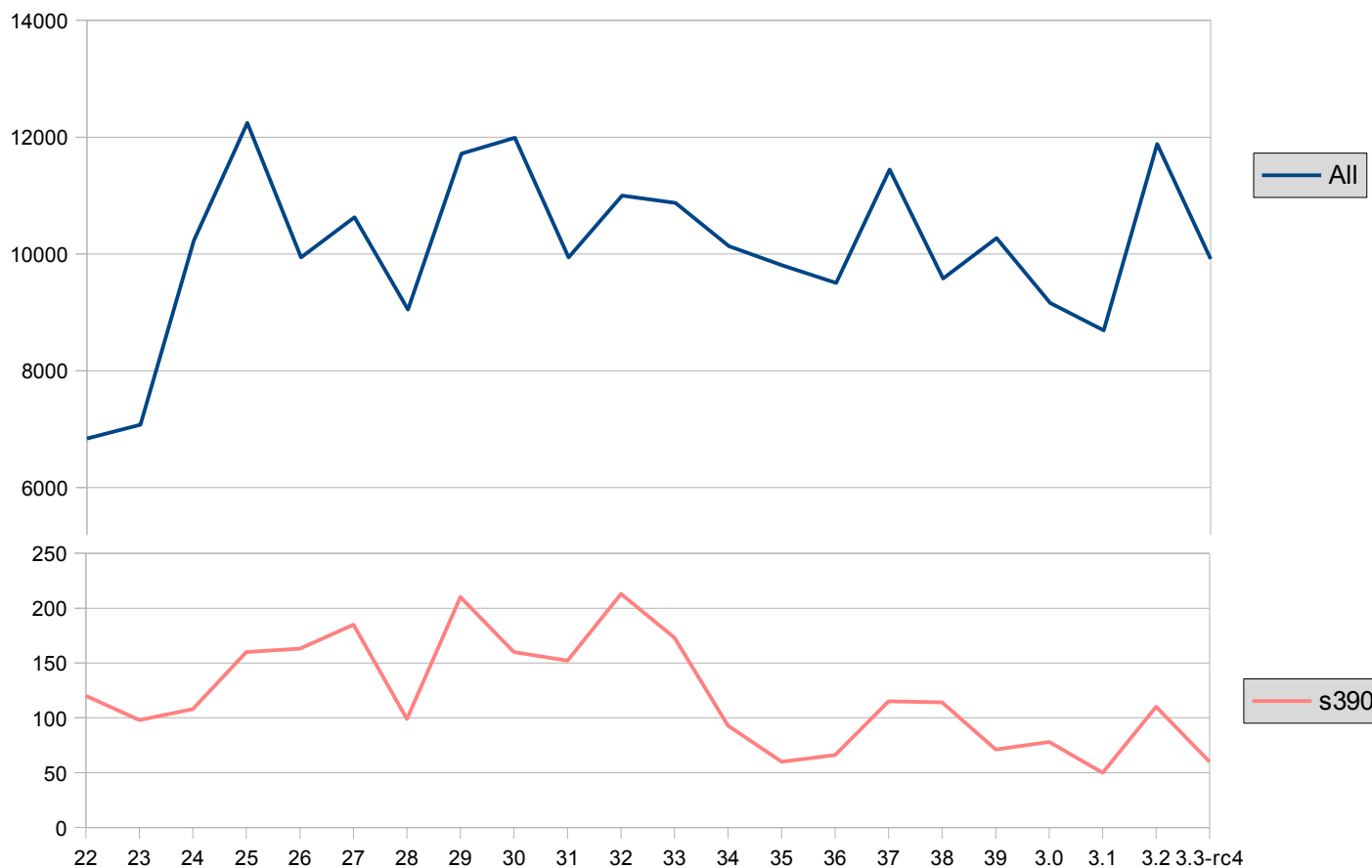
Linux kernel development: rate of change

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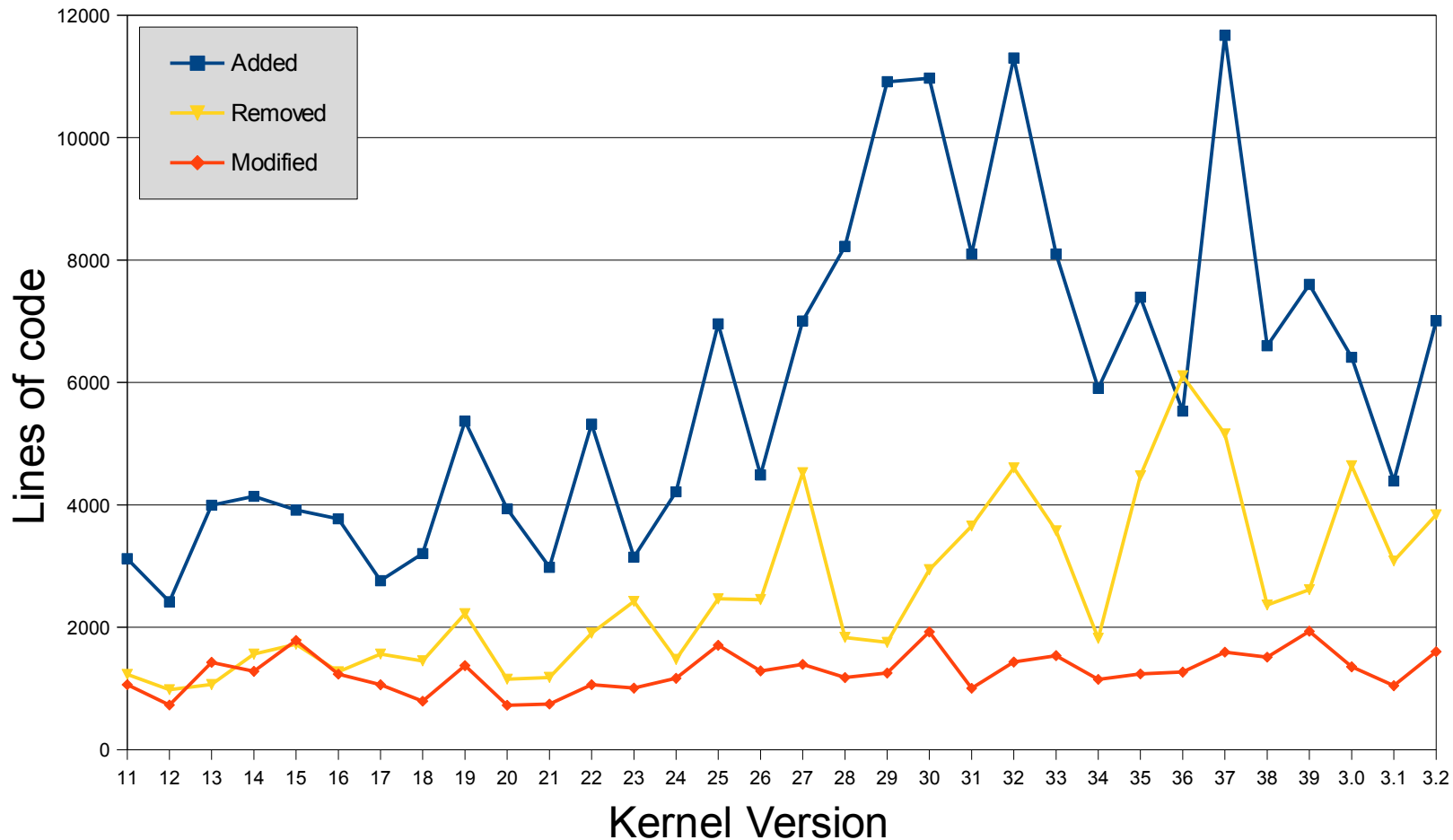
Linux kernel development: System z contributions

- 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
- 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.
X Indicates that the distribution is not supported by IBM.
(*) Also available as 31-bit distribution.

Kernel news – Common code

Linux version 2.6.38 (2011-03-14)

– Automatic process grouping (SCHED_AUTOGROUP)

– RCU-based path name lookup (dcache scalability)

– Transparent huge pages

– Transmit packet steering (XPS) for multiqueue devices

Linux version 2.6.39 (2011-05-18)

– Ext4 SMP scalability

– IPset network resource groups

– Transcendent memory

– Unicore32 architecture

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

Kernel news – Common code

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

- QDIO outbound scan algorithm (kernel 2.6.38)
  - Improved scheduling of QDIO tasklets, OSA / HiperSockets / zfcp 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.

- 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)
  - Enhanced dump support that is able to reduced dump size, share disk space, dump to network, etc.
  - Integrated into the System z stand-alone dump tools and shutdown actions framework

Two stage dumper / kdump support

- **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

- **Pro**
  - The makedumpfile tool can be used to filter the memory of the crashed system
  - The existing kernel be used to write dumps directly to a filesystem or to the network

- **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 you system an check the reservation

```
# cat /proc/iomem
00000000-3fffffff : System RAM
  00000000-005f1143 : Kernel code
  005f1144-00966497 : Kernel data
  00b66000-014c4e9f : Kernel bss
40000000-47ffffff : 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
```
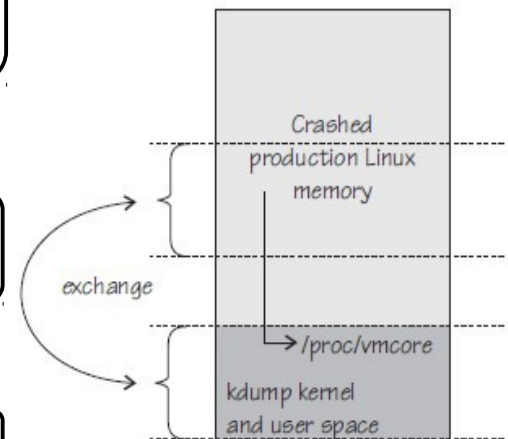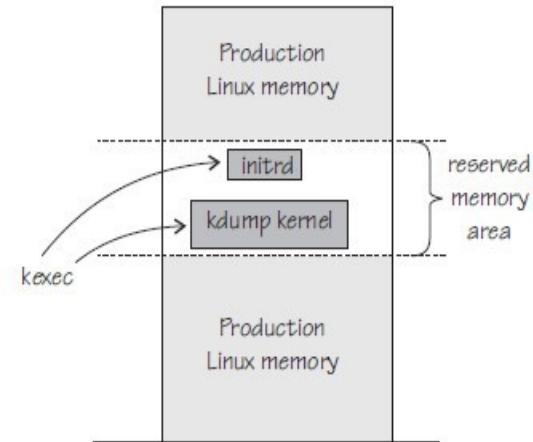
System z kernel features – Storage FICON

- **Multi-track extension for HPF (kernel 2.6.38)**
  - Allows to read from and write to multiple tracks with a single CCW
  - For z10 zHPF multi-track support is limited to 64k byte data transfers
  - For z196 zHPF multi-track supports > 64b byte data transfers

- **Improve handling of stolen DASD reservation (kernel 2.6.38)**
  - Provide alternatives to handle unit checks that indicate stolen reservations
    - Fail any request to a device until it is set offline
    - Queue I/O until reservation is released again

- **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
  - Storage array needs to support the read-track and write-full-track commands.

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

- **Add NPIV information to symbolic port name (kernel 2.6.39)**
  - Add the device bus-ID and the network node to the symbolic port name if the NPIV mode is active.

- **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

- IPv6 support for the qetharp tool (kernel 2.6.38)
  - 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.

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

- 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.

- 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 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

System z kernel features – Usability / RAS

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

- **Get CPC name (kernel 3.0)**
  - Useful to identify a particular hardware system in a cluster
  - The CPC name and HMC network name are provided

- **Removal of data execution protection (kernel 3.0)**
  - "no execute" support relies on the secondary space mode for data separation
  - With System z10 the new instructions LRL, LGRL and LGFRL for pc-relative data access have been added
  - These new instructions access the memory operand in the same address space from where the instructions has been fetched.

System z toolchain

- **zEnterprise 196 exploitation (gcc 4.6)**
  - 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

- **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

Binary	valgrind					system calls	kernel

`80000554 <main>:`
`  stmg    %r14,%r15,112(%r15)`
`  larl    %r2,80000698 <string>`
`  aghi    %r15,-160`
`  brasl   %r14,800003f4 <puts@plt>`
`  lmg     %r14,%r15,272(%r15)`
`  lghi    %r2,0`
`  br      %r14`
	replace some of the library calls	translation into IR	instru-mentation	translation to machine code	New binary	system calls	kernel
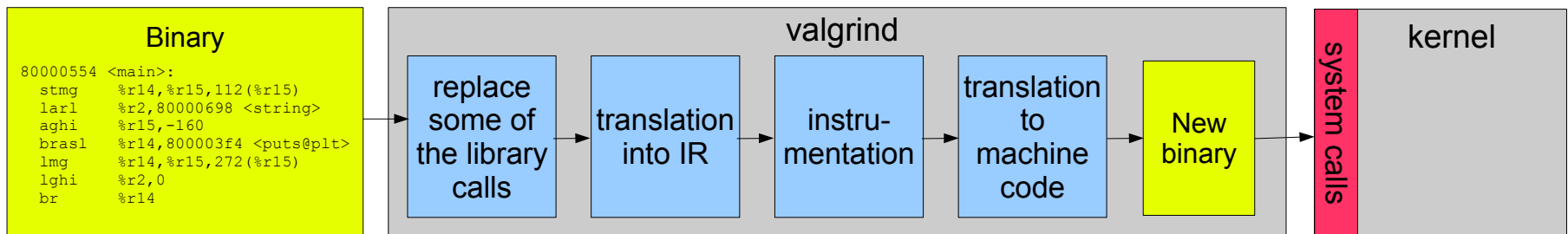
s390-tools package: what is it?

- 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

CHANGE
chccwdev
chchp
chreipl
chshut
chcrypt
chmem

DASD
dasdfmt
dasdinfo
dasdstat
dasdview
fdasd
tunedasd

DUMP & DEBUG
dbginfo
dumpconf
zfcpdump
zfcpdbf
zgetdump
scsi_logging_level

DISPLAY
lscss
lschp
lsdasd
lsluns
lsqeth
lsreipl
lsshut
lstape
lszcrypt
lszfcp
lsmem

MONITOR
mon_fsstatd
mon_procd
ziomon
hyptop

NETWORK
ip_watcher
osasnmpd
qetharp
qethconf

z/VM
vmconvert
vmcp
vmur
cms-fuse

MISC
cpuplugd
iucvconn
iucvtty
ts-shell
ttyrun

TAPE
tape390_display
tape390_crypt

BOOT
zipl

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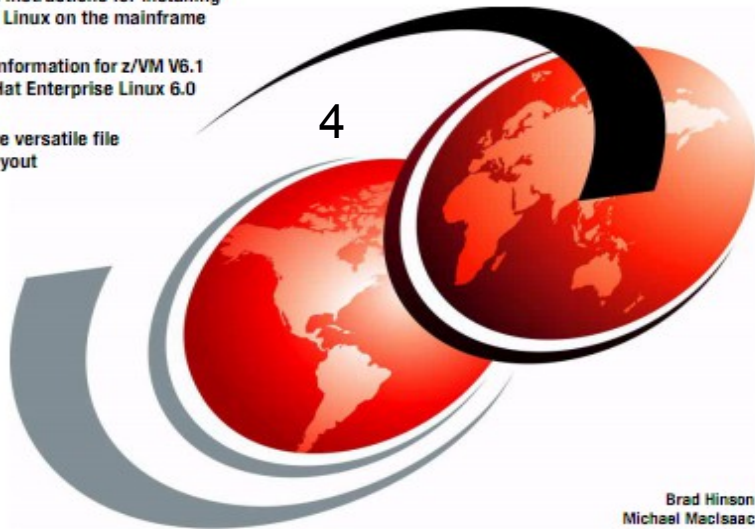
More information

New Redbooks

IBM

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

4

New, more versatile file system layout

Brad Hinson
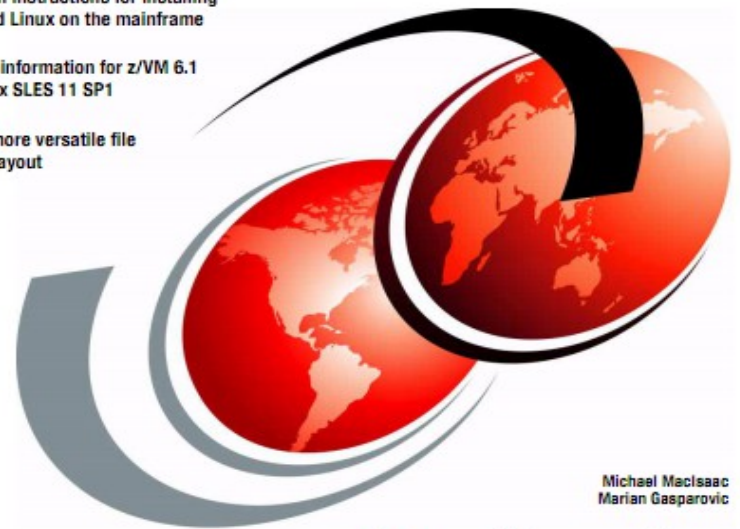Michael MacIsaac

Redbooks

ibm.com/redbooks

IBM

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
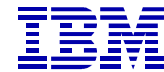
A new, more versatile file system layout

Michael MacIsaac
Marian Gasparovic

Redbooks

ibm.com/redbooks

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

Questions?

Martin Schwidefsky

*Linux on System z
Development*

*Schönaicher Strasse 220
71032 Böblingen, Germany*

*Phone +49 (0)7031-16-2247
schwidefsky@de.ibm.com*

Session 10324