8th European GSE/IBM TU for z/VSE, z/VM and Linux on System z

October 20th - 22nd, 2014, Dresden

VM01 – Aktuelles von Linux on System z

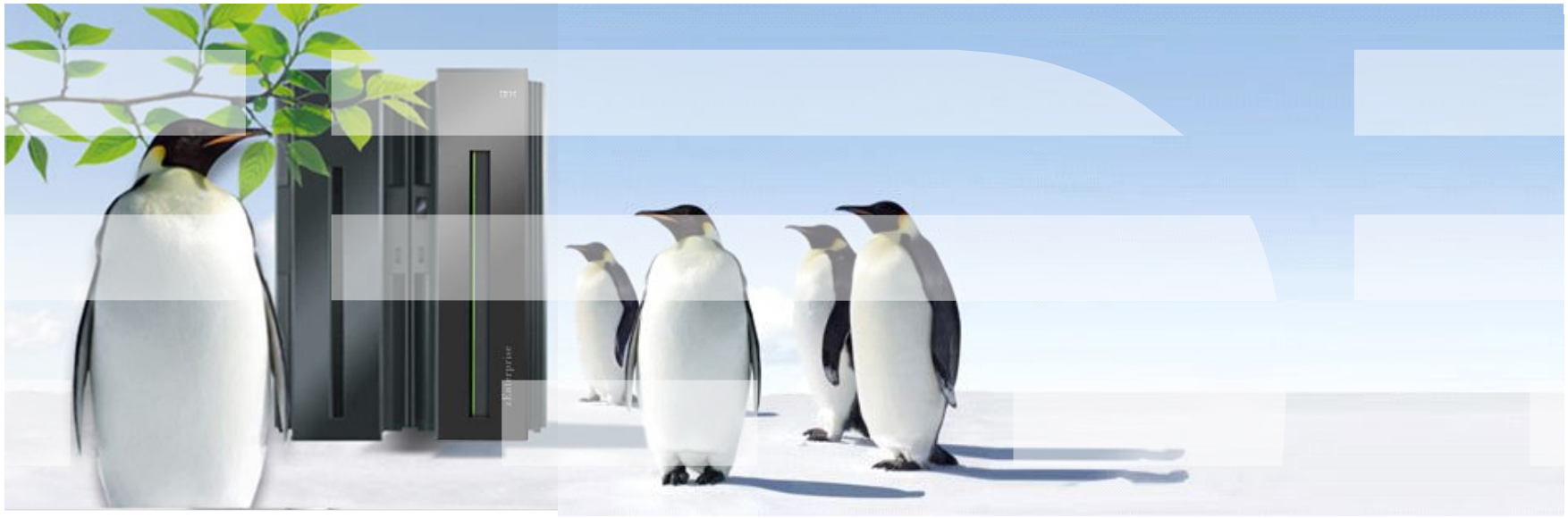
Arwed Tschoeke
Systems Architect
IBM Germany R&D
tschoeke@de.ibm.com

Hilton Hotel
Dresden

Dank an

Martin Schwidefsky

Dr. Eberhard Pasch



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:

IBM, the IBM logo, BladeCenter, Calibrated Vectored Cooling, ClusterProven, Cool Blue, POWER, PowerExecutive, Predictive Failure Analysis, ServerProven, System p, System Storage, System x , System z, WebSphere, DB2 and Tivoli are trademarks of IBM Corporation in the United States and/or other countries. For a list of additional IBM trademarks, please see http://ibm.com/legal/copytrade.shtml.

The following are trademarks or registered trademarks of other companies: Java and all Java based trademarks and logos are trademarks of Sun Microsystems, Inc., in the United States and other countries or both Microsoft, Windows,Windows NT and the Windows logo are registered 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 or both. Linux is a trademark of Linus Torvalds in the United States, other countries, or both. Cell Broadband Engine is a trademark of Sony Computer Entertainment Inc. InfiniBand is a trademark of the InfiniBand Trade Association.
Other company, product, or service names may be trademarks or service marks of others.

NOTES: Linux penguin image courtesy of Larry Ewing (lewing@isc.tamu.edu) and The GIMP

Any performance data contained in this document was determined in a controlled environment. Actual results may vary significantly and are dependent on many factors including system hardware configuration and software design and configuration. Some measurements quoted in this document may have been made on development-level systems. There is no guarantee these measurements will be the same on generally-available systems. Users of this document should verify the applicable data for their specific environment. IBM hardware products are manufactured from new parts, or new and serviceable used parts. Regardless, our warranty terms apply.
Information is provided “AS IS” without warranty of any kind. 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 are suggested US list prices and are subject to change without notice. Starting price may not include a hard drive, operating system or other features. Contact your IBM representative or Business Partner for the most current pricing in your geography. 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

Linux on System z introduction

Interesting facts and numbers

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 3.16 ?

18.879.129 lines of code

How many of the world's top 500 supercomputers run Linux (June 2014)

485 / 97%

What is the biggest (known) Linux installation (June 2014)?

Tianhe-2 with 3,120,000 cores rated at 33,862.7 TFLOPS/s

What percentage of web servers run Linux (June 2014)

67.6% run Unix, of those 56.7% run Linux (41.8% unknown) = 38.3%

What percentage of desktop clients run Linux (May 2014) ?

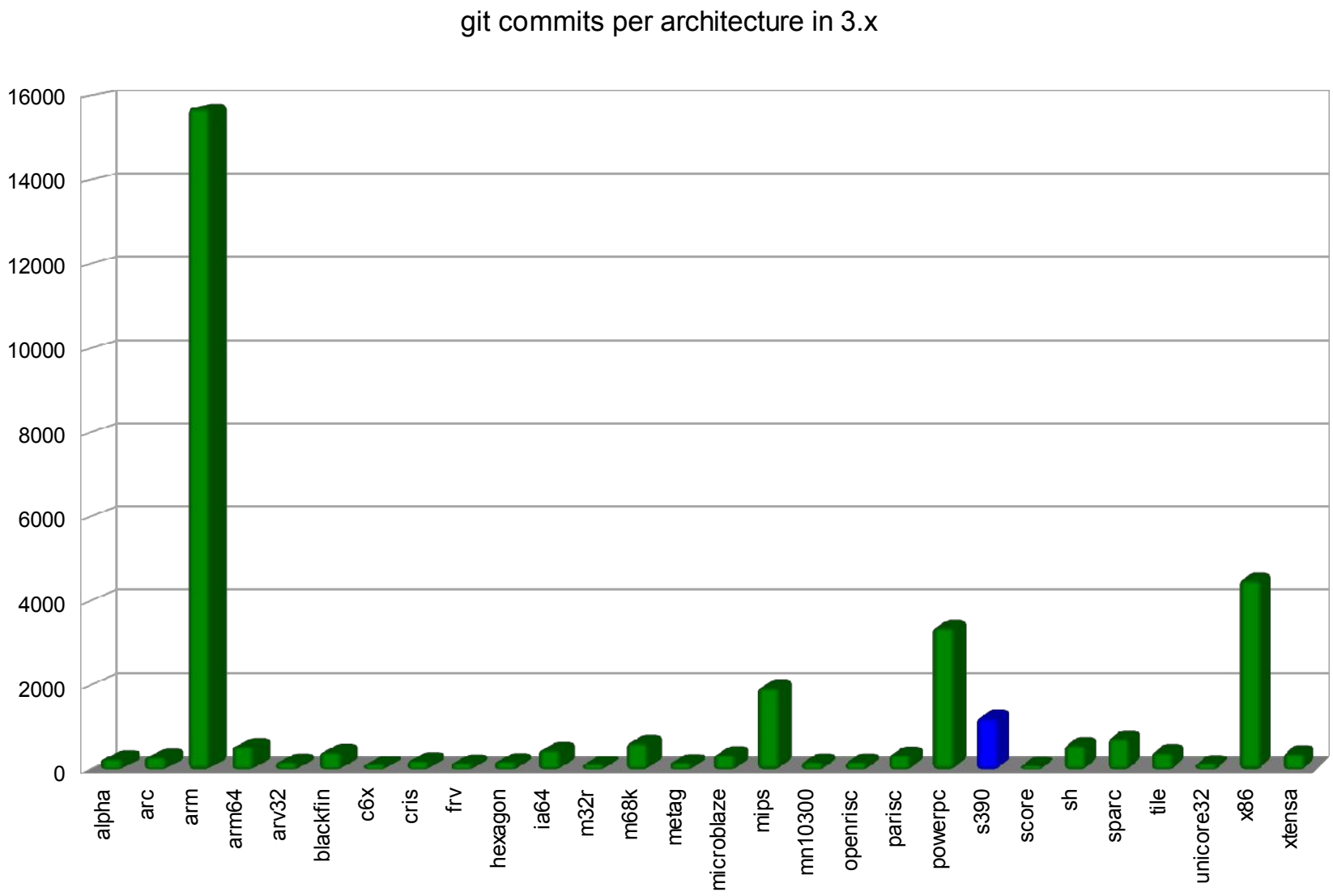
2.07% via Linux, 8.75% via Android

What is the architecture with the larges amount of core changes in v3.x

ARM ~112 KLOC/release, mips and powerpc ~25 KLOC/release,
x86 ~23 KLOC/release. System z (alias s390) ~7 KLOCS/release.

Linux is Linux, but …features, properties and quality differ dependent on your platform and your use case

Source: http://kernel.org
http://top500.org/statistics
http://w3techs.com
http://www.w3counter.com



Linux on IBM System z in 1Q2014

*Installed Linux MIPS at 49% CAGR**

26.4% of Total installed MIPS run Linux as of 4Q13

Installed IFL MIPS increased 24% from 1Q13 to 1Q14

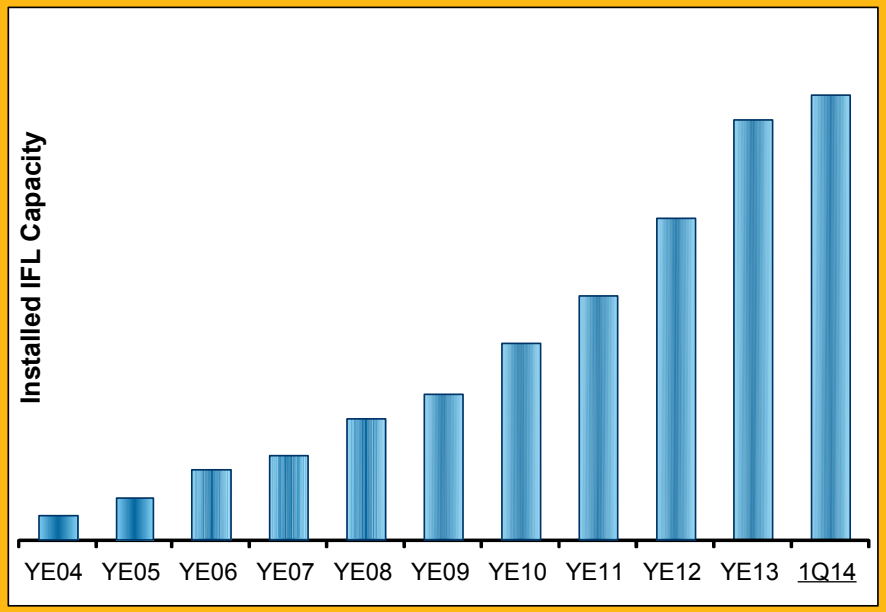
39% of System z Customers have IFL's installed as of 1Q14

80 of the top 100 System z Customers are running Linux on the mainframe as of 1Q14 **

34% of all System z servers have IFLs

55% of new FIE/FIC System z Accounts run Linux (FY10-FY13)

Installed Capacity Over Time



* Based on YE 2003 to YE 2013
**Top 100 is based on total installed MIPS

Linux on System z distributions

What is available today

Linux on System z distributions in service

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 04/2011)

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, GCC 4.3.4, Service Pack 3 (GA 07/2013)

SUSE Linux Enterprise Server 12 (GA Q4/2014?)

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 10 (GA 10/2013)

Red Hat Enterprise Linux AS 6 (GA 11/2010)

Kernel 2.6.32, GCC 4.4.0 Update 5 (GA 11/2013)

Red Hat Enterprise Linux AS 7 (GA 06/2014)

Kernel 3.10, GCC 4.8

Others

Debian, Slackware,

Support may be available by some third party

Supported Linux Distributions

Distribution	zEnterprise - BC12 and EC12	zEnterprise - z114 and z196	System z10	System z9	zSeries
RHEL 6	✔ (1)	✔	✔	✔	✘
RHEL 5	✔ (2)	✔	✔	✔	✔
RHEL 4 (*)	✘	✔ (5)	✔	✔	✔
SLES 11	✔ (3)	✔	✔	✔	✘
SLES 10	✔ (4)	✔	✔	✔	✔
SLES 9 (*)	✘	✔ (6)	✔	✔	✔

✔ 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) Recommended level: RHEL 6.3

(2) Recommended level: RHEL 5.8

(3) Recommended level: SLES 11 SP3

(4) Recommended level: SLES 10 SP4 with latest maintenance updates

(5) 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.

(6) SLES 9 SP4 with 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 on this server.

(*) The distribution is out of service, extended support is required.

source: http://www.ibm.com/systems/z/os/linux/resources/testedplatforms.html

Current Linux on System z Technology

Key features & functionality already
contained in the SUSE & Red Hat Distributions

IBM zEnterprise EC12 and BC12 support

Transactional execution (kernel 3.7)

6.4 11.3

- Also known as hardware transactional memory
- CPU features that allows to execute a group of instructions atomically
- Optimistic execution, if a transaction conflicts a rollback to a saved state is done

Storage class memory – Flash Express (kernel 3.7)

6.4 11.3

- Internal Flash Solid State Disk (SSD)
- Accessed via Extended Asynchronous Data Mover (EADM) sub-channels
- Support for concurrent MCL updates with kernel version 3.8

Support for Crypto Express 4S cards (kernel 3.7)

6.4 11.3

- New generation of crypto adapters plug-able into the I/O drawer
- New type 10 which uses a bit field to indicate capabilities of the crypto card

Native PCI feature cards (base in kernel 3.8, ongoing)

- Support for native PCIe adapters visible to the operating system

System zEC12 features – Transactional Execution

11.3

6.4

Transactional execution is a concurrency mechanism of the CPU comparable to database transactions

- Several reads and stores from/to memory logically occur at the same time
- Improves performance for fine-grained serialization
- Useful for lock-less data structures and speculative compiler optimizations

Two types of transactions: constraint and non-constraint

Conflicting memory accesses will cause the transaction to abort

- Transaction abort is rather expensive
- Constraint transaction will automatically restart
- Ratio of successful vs. aborted transaction is important for performance

Kernel support is required to enable user programs to use transactional execution

- Control registers setup
- Debugging support for additional PER controls via ptrace

System zEC12 features – Transactional Execution

11.3

6.4

Example of a list_add operation

```
struct spinlock_t list_lock;
  struct list_head list_head;
  void list_add(struct list_head *new)
  {
      spin_lock(&list_lock, 0, 1);
      list_add(new, &list_head);
      spin_unlock(&list_lock, 1, 0);
  }
```

Typical pattern:
1) lock, 2) a short operation, 3) unlock

Traditional code:

```
      # spin_lock
        larl    %r3,list_lock
        lhi     %r1,1
lock:   lhi     %r0,0
        cs      %r0,%r1,0(%r3)
        ltr     %r0,%r0
        jne     lock
        # list_add
        larl    %r4,list_head
        lg      %r5,0(%r4)
        stg     %r4,0(%r2)
        stg     %r5,8(%r2)
        stg     %r2,0(%r5)
        stg     %r2,8(%r4)
        # spin_unlock
        cs      %r1,%r0,0(%r3)
        br      %r14
```

Transactional code

```
# begin transaction
tbeginc 0,0

# list_add
larl    %r4,list_head
lg      %r5,0(%r4)
stg     %r4,0(%r2)
stg     %r5,8(%r2)
stg     %r2,0(%r5)
stg     %r2,8(%r4)
# end transaction
tend
br      %r14
```

zEC12/zBC12 features – Flash Express

11.3

6.4

PCIe I/O adapter with NAND Flash SSDs

- Flash Express cards are plugged as pairs to build a RAID10
  - Pair is connected with interconnect cables
  - Card replacement is concurrent if one card fails
- Up to 4 pairs of cards are supported (4 * 1.4TB = 5.6TB)

New tier of memory: Storage Class Memory

- Accessed via Extended Asynchronous Data Mover (EADM) subchannels via the new Storage Class Memory (SCM) block driver

Flash Express is split into memory increments

- Memory increments are assigned to LPARs via the SE or HMC
- Memory increment size is 16 GB

Flash Express is not persistent over IML

Linux on System z features – Compiler toolchain

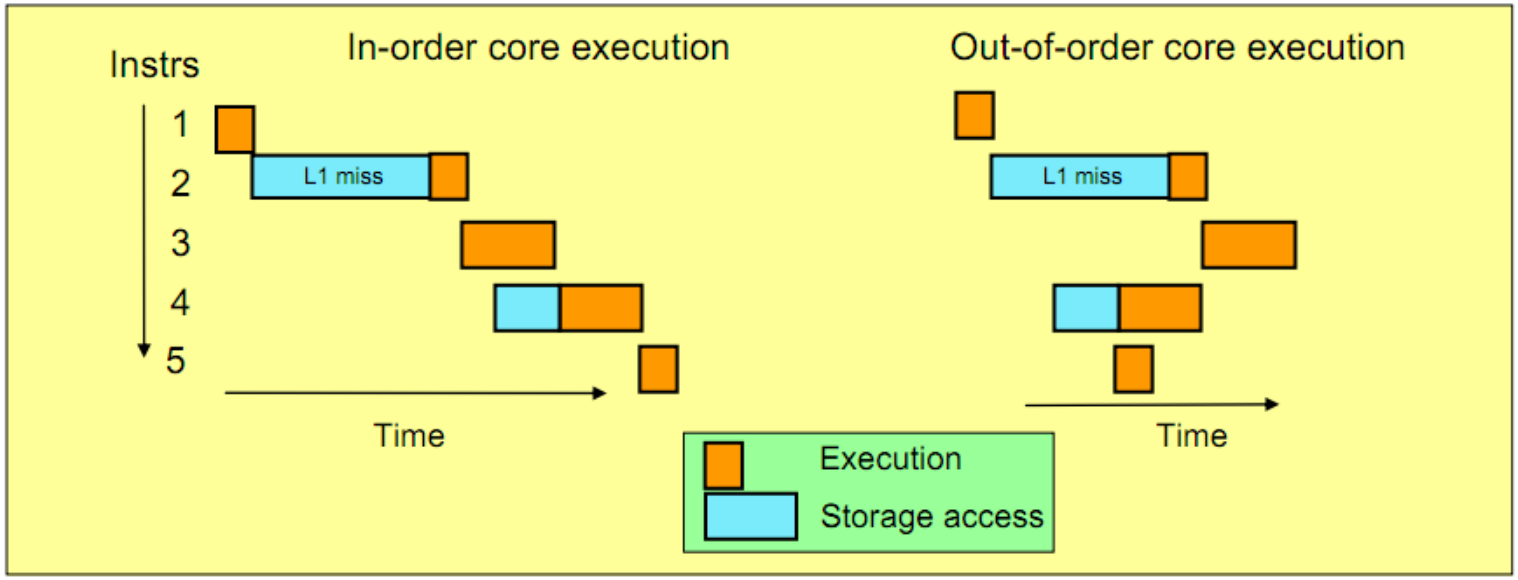
zEnterprise 196 exploitation (gcc 4.6)

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



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

PCI support

Native PCIe feature cards introduced on zEC12 and zBC12

- 10GbE RoCE Express, network card for SMC-R
- zEDC Express, data compression/decompression card

Native PCIe adapter concept

- Plugged into an PCIe I/O drawer
- Managed by an internal firmware processor (IFP)
- Device driver for the PCIe function is located in the operating system

Uses standard Linux PCI support and drivers with some constraints

- Only MSIX, no port I/O, memory mapped I/O by use of PCI load/store instructions
- Provides ability to assign individual functions of an adapter to an LPAR
- Converted System z architecture code to use generic hardirqs
- Only selected PCIe adapters are known to the IFP and surfaced to the OS

10GbE RoCE Express

Native PCIe networking card

- 10 Gigabit remote direct memory access (RDMA) capable network card
- Uses Infiniband RDMA over Converged Ethernet (RoCE) specification
- Up to 16 10GbE RoCE Express adapters per machine
- Reduced latency and lower CPU overhead
- Supports point-to-point connections and switch connection with an enterprise-class 10 GbE switch

Software support

- z/OS V2R1 with PTFs supports SMC-R with RoCE
- z/VM support planned
- Linux support in available upstream but not included in any distribution yet

zEDC Express

Native PCIe data compression / decompression card

- Up to 8 adapters can be installed into a single machine
- With large blocks, it can compress data at more than 1 GB per second
- Implements compression as defined by RFC1951 (DEFLATE)
- Comparable to “gzip -1”

Software support

- z/OS V2R1, V1R13 and V1R12 with PTFs
- Linux device driver to gain access to zEDC has been posted on LKML and has been accepted into the upstream kernel
- The zlib open source library is a C implementation commonly used to provide compression and decompression services.

System z kernel features – memory management

Add support for physical memory > 4TB (kernel 3.3)

7 12

- Increase the maximum supported memory size from 4TB to 64TB.
- Memory sizes large than 4TB require a 4-level page table
- Makes memory accesses by the kernel slightly slower, the kernel will automatically use a 3-level page table for memory sizes <= 4TB
- Requires next HW generation

Transparent huge page support (kernel 3.7)

- Make the common code transparent huge page support available for Linux on System z.
- With THP 1MB pages will be used to back normal anonymous memory mappings.
- Any application will benefit from using huge pages.

Add page table dumper (kernel 3.7)

- Add a sysfs interface to read the current layout of the kernel address space.
- Useful information for the kernel developer.

System z kernel features – memory management

Implement write protection based dirty page detection (kernel 3.8)

- Convert dirty page detection from the change-bit in the storage key to a fault based method. An unmodified page is now always mapped read-only.
- Due to dirty page accounting for memory mappings no additional faults are necessary
- Removes the storage key operations to detect page dirty state

Implement fault based referenced page detection (kernel 3.12)

- Convert referenced page detection from the reference-bit in the storage key to a fault based method. An old page is now always mapped with the invalid bit set (no read, no write access).
- New mappings are always created with the software referenced bit set
- Removes the storage key operations to detect page referenced state.

Avoiding storage key operations improves performance

- The savings in storage key operations outweigh the slightly increase number of faults
- After IPL a system without KVM will not access the storage keys at all
- KVM still makes use of storage keys for provide correct guest virtualization

System z kernel features – core improvements

BPF JIT compiler for System z (kernel 3.7)

- The Berkeley Packet Filter is an interface and a language definition that allows to pass a filter to the kernel to select network packets to send on a socket
- The BPF JIT compiler in the kernel translates the interpreted BPF code to System z code.
- A secondary use of the BPF language is system call filtering.

Expose CPU cache topology in sysfs (kernel 3.7)

- Add an interface to expose the CPU cache topology to user space.
- System z only provides information about CPU caches which are private to a CPU, information about shared caches is not exposed.

Add interface for partition-resource management (kernel 3.14)

- The diagnose 0x304 interface is used to inspect and change the different LPAR partition-resource parameters
- The LPAR needs to be authorized to participate in CPU management
- The binary kernel interface allows a system management software to control the partition weight and partition-capping flags

System z kernel features – core improvements

CPU-Measurement Sampling Facility (kernel 3.14)

- Uses the hardware CPU sampling facility to take snapshots of a set of sample data at a specified sampling interval, e.g. the cycle counter
- Integrated into the Linux 'perf' tool
- The basic-sampling mode and the diagnostic-sampling mode are supported
- The diagnostic-sampling mode is intended for use by IBM support only

Example how to record sampling data for an application

```
# perf record -e rB0000 – /bin/df
 Filesystem 1K-blocks Used Available Use% Mounted on
/dev/dasda1 6967656 3360508 3230160 51% /
none 942956 88 942868 1% /dev/shm
/dev/dasdb1 6967656 4132924 2474128 63% /root
[ perf record: Woken up 1 times to write data ]
[ perf record: Captured and wrote 0.001 MB perf.data (~29 samples) ]
```

Display the collected sample data

```
# perf report
```

System z kernel features – I/O improvements

No automatic port rescan on events (kernel 3.7)

The rescan of a zfcp port following a fabric change event can cause high fabric traffic, especially when many Linux images share an FCP channel over multiple subchannels with NPIV enabled. This can lead to errors due to timeouts.

Ports are still scanned when the adapter is set online and on manual user triggered writes to the port_rescan sysfs attribute.

Safe offline interface for DASD devices (kernel 3.8, s390-tools 1.21)

Gracefully complete all outstanding I/O requests before a DASD is set offline.

Add robustness against missing interrupts to non-path-grouped internal IO requests (kernel 3.8, s390-tools 1.22)

Improve the Linux behavior in case of a missing interrupt during path grouping

Improve speed of dasdfmt (kernel 3.10)

Reorganize format I/O requests and enable usage of PAV.

Add channel ID sysfs attribute (kernel 3.10)

Add an attribute to each channel-path description with the channel-ID of the path

System z kernel features – networking & security

HiperSockets layer 2 bridge port functionality (kernel 3.14)

- With Linux acting as a software network bridge the network port acting as the bridge needs to be able to receive frames addressed to unknown MAC addresses
- HiperSocket devices can be configured as primary and secondary bridge ports

Add support for EP11 coprocessor cards (kernel 3.14)

- Extend the zcrypt driver with a new capability to service EP11 requests for the Crypto Express4S card in EP11 (Enterprise PKCS#11 mode) coprocessor mode
- For more information about EP11, see "*Exploiting Enterprise PKCS #11 using OpenCryptoki*", SC34-2713

IBM zEnterprise EC12 and BC12 compiler support

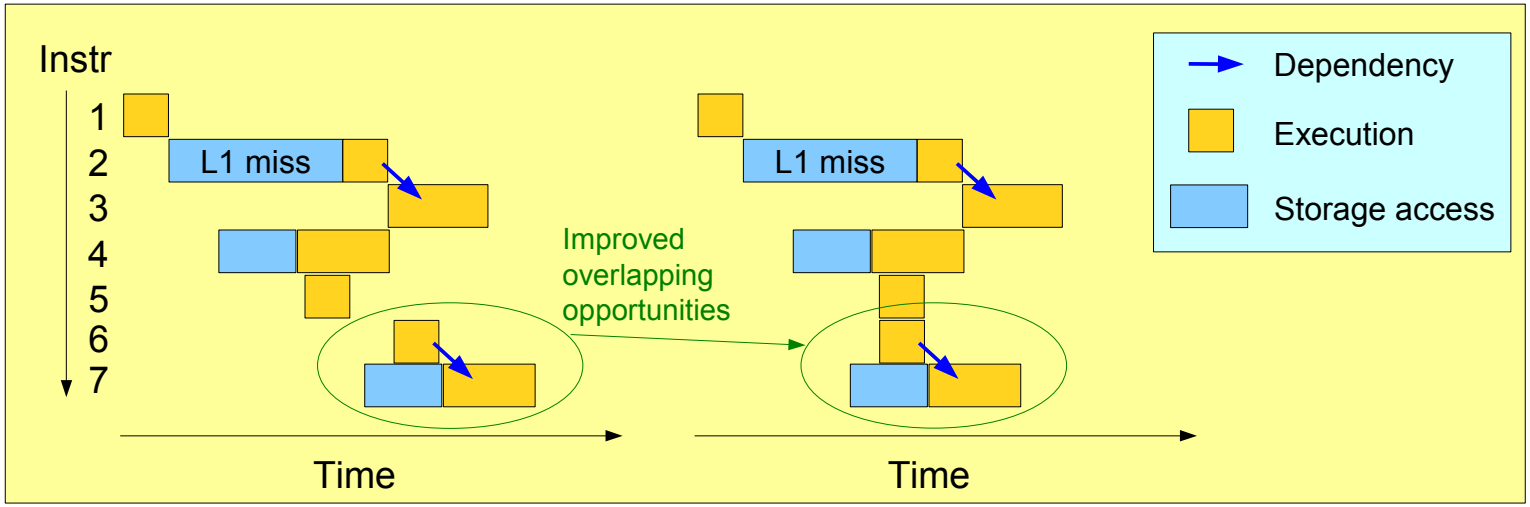
7.0

New compiler options in support of the zEC12/zBC12 CPU (gcc 4.8)

- Option -march=zEC12 to utilize the instructions added with zEC12
- Option -mtune=zEC12 to schedule the instructions appropriate for the pipeline of zEC12

zEC12/zBC12 comes with new instructions

- Transactional Memory support
- Improved branch instructions



XL C/C++ for Linux on System z Managed Beta Program

XL C/C++ for Linux on System z

Will be part of a family of advanced C/C++ compiler products already available on z/OS, AIX, and Linux on Power.

Expected to ease application migration to Linux on System z through:

Conformance to the latest C and C++ programming standards

Compatibility with GNU C/C++

Will maximize application performance through IBM's industry-leading optimization technology

Idea: a compiler to exploit the new HW functionalities without the need to change the distribution

Benefits of participating in this Beta include:

Opportunity to influence the product and future product direction

Ability to test code and documentation, and help ensure compatibility in their environment

Free education, code, and documentation during the beta

Free support by development during the beta for questions and problems

For more information and how to submit a nomination to participate see http://bit.ly/xlbeta

New - ibm.com/support/knowledgecenter/

Firefox

IBM Knowledge Center - Linux on Syste...

IBM Knowledge Center | Sign In | English

Save Search

Products: *Linux on IBM Systems* × Clear Add Products...

Close | Share | Save to Collection

Table of Contents

- **Linux on System z**
  - Library overview
  - Distributions
  - Performance
  - Security
  - Service, support, and troubleshooting

My Collections

Search Results

Linux on IBM Systems > Linux information for IBM systems > Linux on System z

Linux on System z

Find information for your Linux distribution on IBM® System z® mainframes.

Library overview
IBM provides technical documentation with a consolidated view of System z-specific concepts, tool descriptions, and reference information for Linux on System z.

Distributions
Find out which information is relevant to your distribution.

Performance
Improve performance with PAV, learn tips and tricks from the whitepapers, and set up Execute-in-Place technology.

Linux on System z info at:

ibm.com/support/knowledgecenter/linuxonibm/liaaf/lnz_r_main.html

Documentation news – Updates available

Linux Distributions

SUSE Linux Enterprise Server 11 SP 3

Red Hat Enterprise Linux 7

Upstream Linux 3.16

ibm.com/developerworks/linux/linux390/documentation_dev.html

Security

Secure Key Common Cryptographic Architecture 4.2.10 Application Programmer's Guide

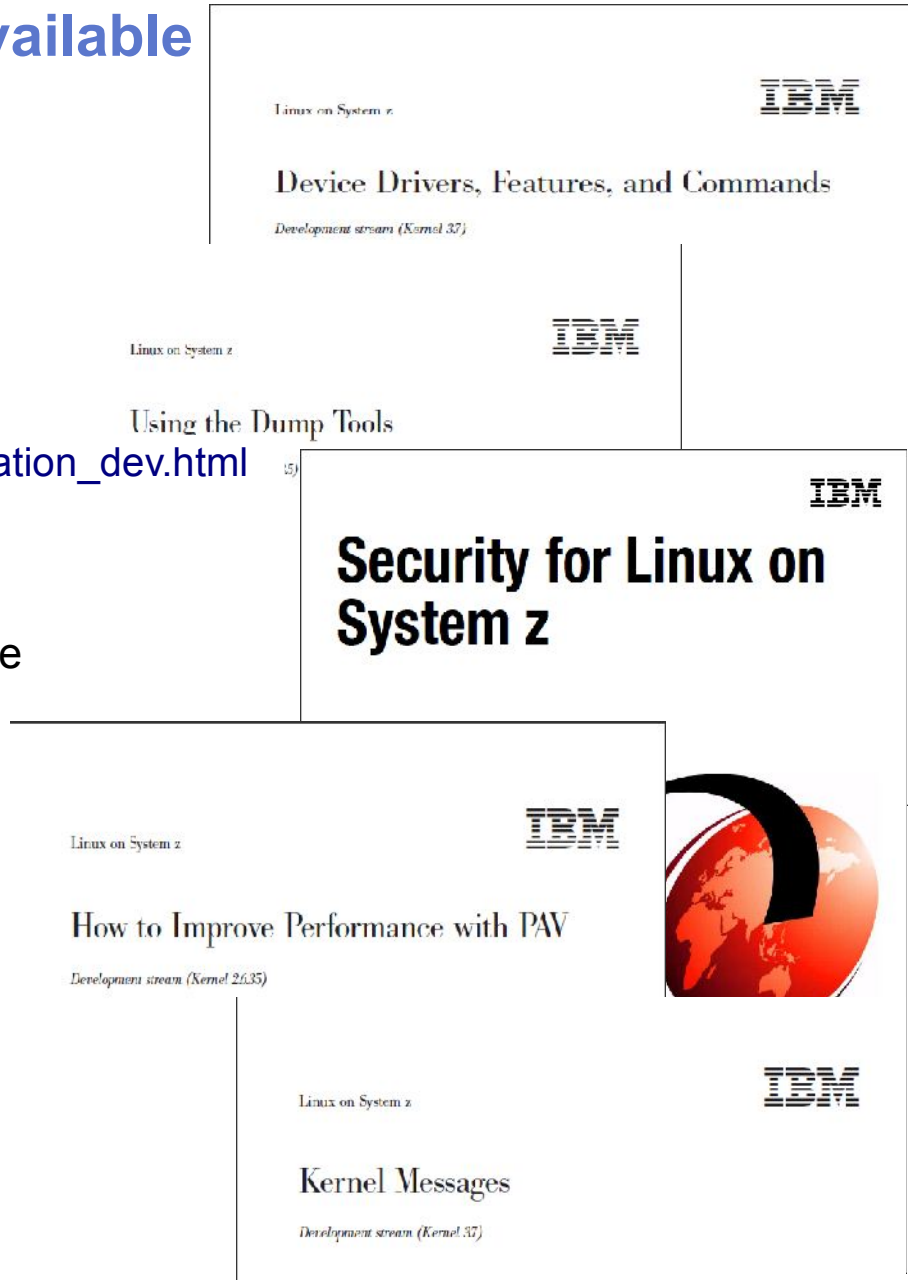
Libica 2.2.0. Programmer's Guide

Redbooks

IBM Wave, Virtualization, Oracle, Security

Whitepapers

FileNet P8 5.1, Live Guest Relocation, iSCSI



Kernel news – Common code

Linux version 3.12 (2013-11-03)

- RAID5 multithreading
- VFS locking improvements (lockref)
- Better Out-Of-Memory handling
- Improved tty layer locking
- IPC locking improvements

Linux version 3.13 (2014-01-19)

- A scalable block layer for high performance SSD storage
- nftables, the successor of iptables
- Improved page table access scalability in hugepage workloads
- TCP Fast Open enabled by default

Kernel news – Common code

Linux version 3.14 (2014-03-30)

- Deadline scheduling class for better real-time scheduling
- zram memory compression mechanism considered stable
- Btrfs inode properties
- Userspace locking validator
- TCP automatic corking

Linux version 3.15 (2014-05-08)

- Improved working set size detection
- New file locking scheme: open file description locks
- Faster erasing and zeroing of parts of a file
- File cross-renaming support
- FUSE improved write performance

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 version 4, 5, and 6
- SuSE Linux Enterprise Server version 9, 10, and 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

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

DASD
- dasdfmt
- dasdinfo
- dasdstat
- dasdview
- fdasd
- tunedasd

MONITOR
- mon_fsstatd
- mon_procd
- ziomon
- hyptop

NETWORK
- ip_watcher
- osasnmpd
- qetharp
- qethconf
- qethqoat

TAPE
- tape390_display
- tape390_crypt

DUMP & DEBUG
- dbginfo
- dumpconf
- zfcpdump
- zfcpdbf
- zgetdump
- scsi_logging_level

z/VM
- vmconvert
- vmcp
- vmur
- cms-fuse

MISC
- cpuplugd
- iucvconn
- iucvtty
- ts-shell
- ttyrun

BOOT
- zipl

Questions?