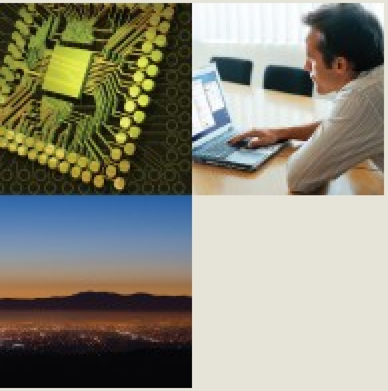


Monitoring Linux Guests and Processes with Linux Tools

Martin Schwidefsky
IBM Lab Boeblingen, Germany

August 14th, 2008
Session 9266

Agenda

- CPU Time Accounting
- z/VM Monitor Stream
- Hypervisor Data
- System Information

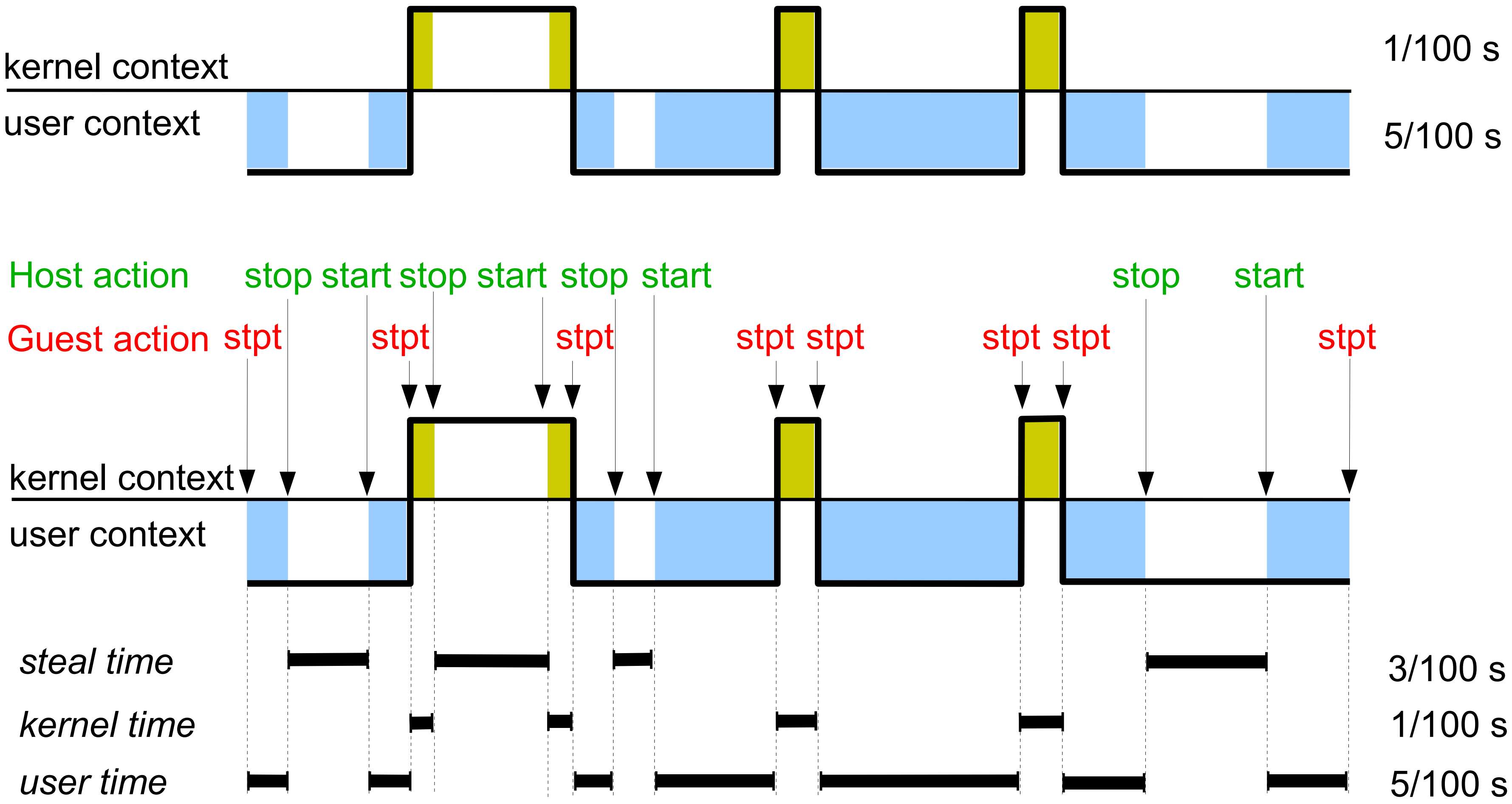
CPU time accounting

- **How much CPU time is spend on what kind of work?**
  - user processes
  - system
  - I/O wait
- **How much work is done per unit of time by a subsystem?**
  - I/O
  - memory
- **... CPU time is essential for monitoring**

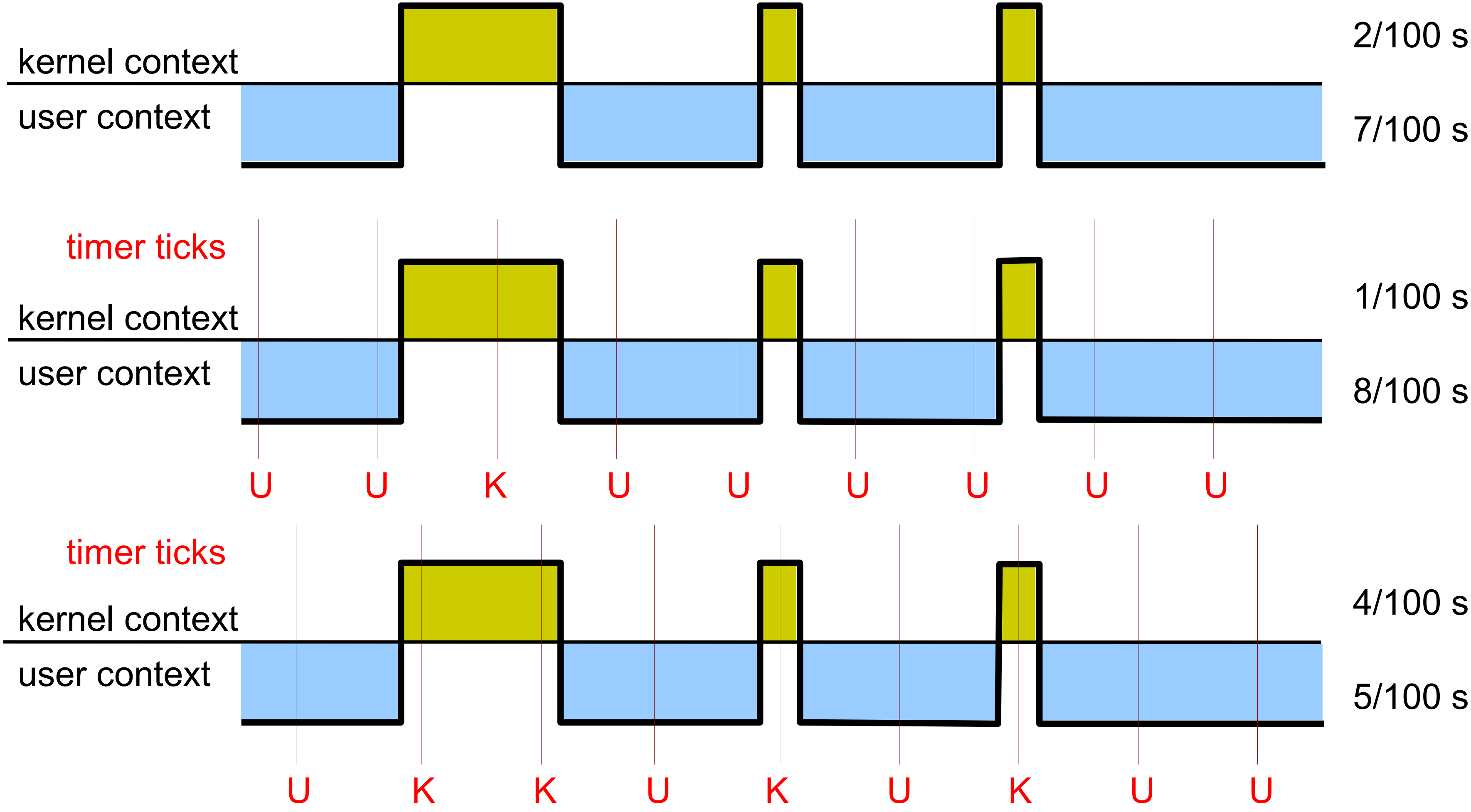
CPU time accounting

- **with Linux support for System z virtual CPU timer**
  - SLES 10 / RHEL 5 and up (upstream as of Linux 2.6.11)
  - time accounting based on virtual CPU timer
  - involuntary wait time exposed as “steal time” to user
  - recent Linux distributions get the numbers right
- **without Linux support for System z virtual CPU timer**
  - older Linux distributions
  - Linux has no notion of distinction between virtual CPU time and real time
  - Linux has no notion of involuntary wait time (steal time)
  - uses tick-based time accounting which is inherently inaccurate, particularly on virtual systems
  - use numbers carefully!

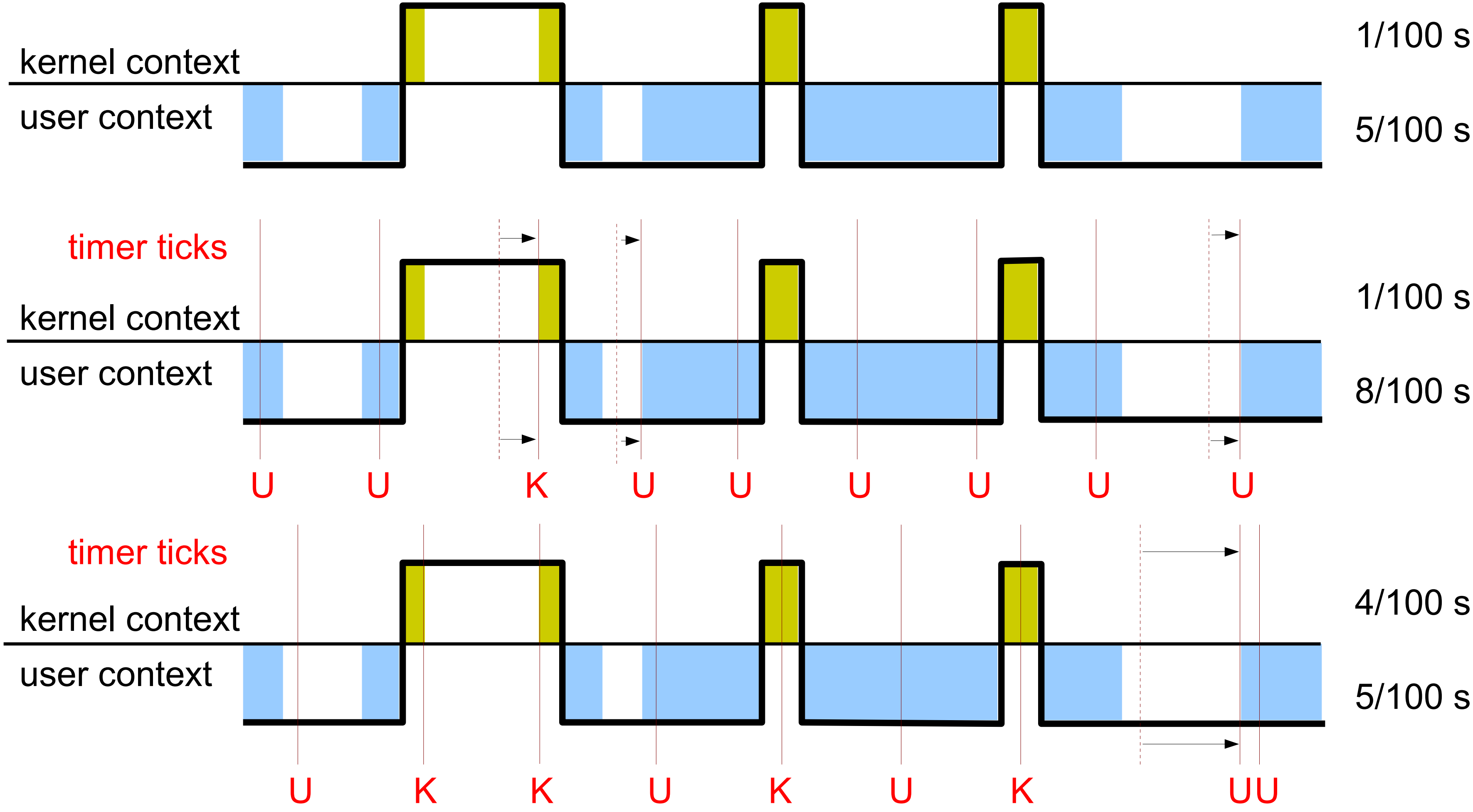
Timer-based CPU accounting



Tick-based (mis-)accounting



Tick-based (mis-accountig) on virtual CPUs



Tick-based accounting is wrong

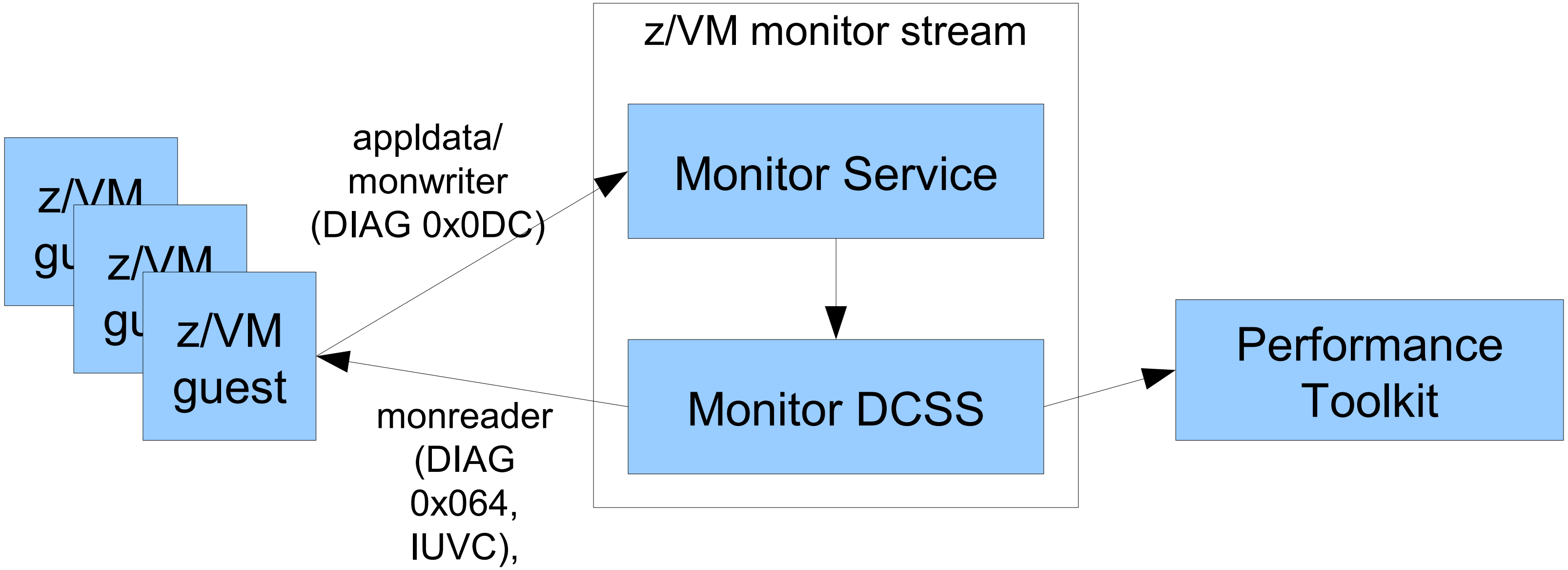
- **Tick-based accounting is inaccurate by design**
  - Sampling frequency, that is, tick rate is insufficient
  - System ticks in time with real clock, not virtual clock
- **On systems with virtual CPUs (z/VM, VMware, KVM, Xen, ...)**
  - Process time slices are based on real CPU time (usually 5-6 ticks)
  - The real CPU usually spends part of its time “elsewhere”
  - Processes can loose part or even all of their time slice
  - Processes get accounted time they did not use
- **On systems without virtual CPUs**
  - The approach is usually good enough, though

Options for systems without Linux support for the virtual CPU timer

- **Either do not use Linux accounting numbers, but use per-image accounting numbers from hypervisor instead**
  - limited granularity of per-image measurement data
- **Or normalize Linux accounting numbers:**
  - Retrieve average CPU usage numbers from hypervisor
  - Multiply Linux CPU accounting numbers by average CPU usage numbers
- **Anyway, it's not as good as using a virtual CPU counter**

z/VM Monitor Service Infrastructure

- **Provides monitor data through the monitor stream**
  - z/VM monitor service collects data in a shared memory segment (DCSS)
  - Producer: a range of facilities, e.g. Linux through appldata / monwriter
  - Consumer: Performance Toolkit, or Linux application through monreader



z/VM Monitor Service Infrastructure

- **There are different record domains:**
  - system, storage, user, appldata, ...
- **There are different record types:**
  - event records, sample records
- **MONITOR – the CP command that controls monitoring**
  - sampling interval
  - record domains
  - records types
- **Performance Toolkit – the consumer of monitoring data**
  - accessible through 3270 terminal or http

appldata – Linux monitoring modules

- **Linux Kernel modules which gather information**
- **appldata_os**
  - CPU utilization, processes
- **appldata_mem**
  - memory, paging, cache
- **appldata_net_sum**
  - packets, bytes, errors
- **appldata modules are controlled through sysfs attributes**

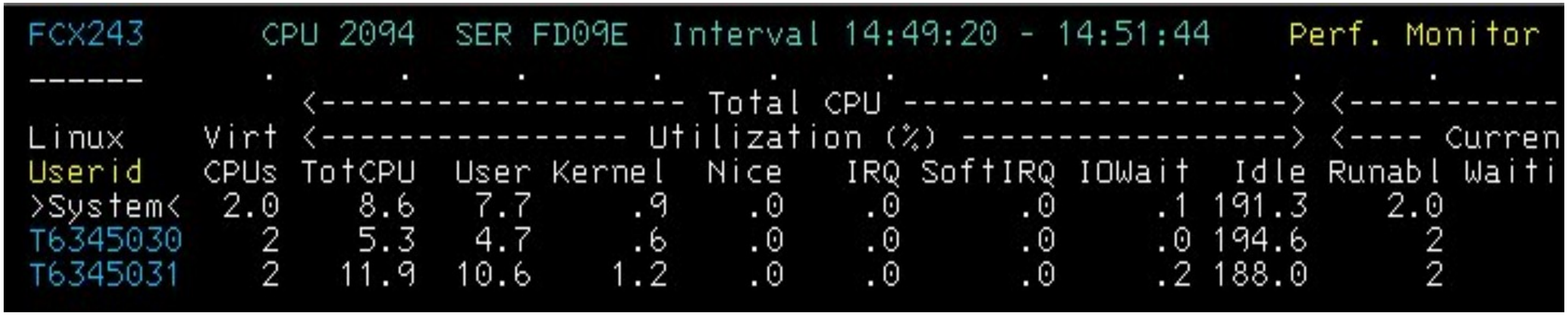
```
# modprobe appldata_os
# echo 20000 > /proc/sys/appldata/interval
# echo 1 > /proc/sys/appldata/timer
# echo 1 > /proc/sys/appldata/os
```

appldata – Linux monitoring modules

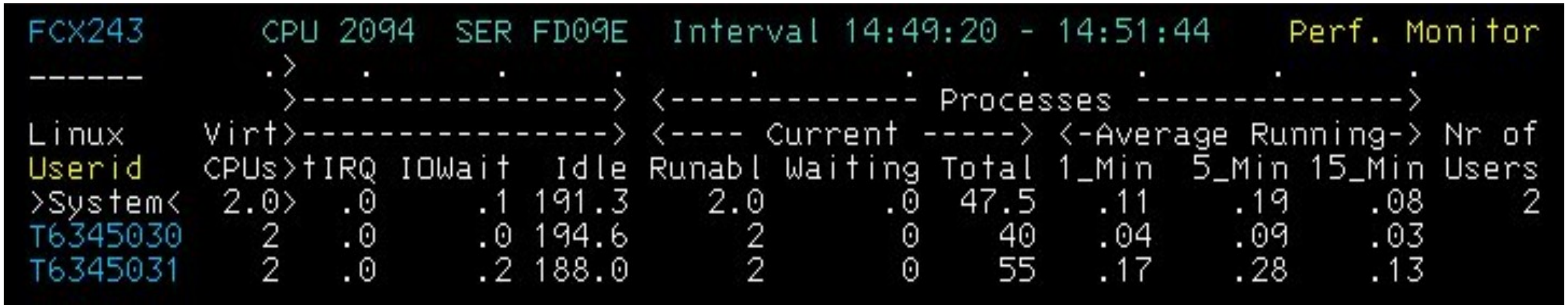
- **sampling interval**
  - in milliseconds
  - based on virtual CPU time
  - reduced sampling rate on idle systems
  - independent from z/VM sampling interval
- **Support for steal time has been added recently**
  - Linux kernel 2.6.18, RHEL5, SLES 10 SP1, z/VM Perf. Toolkit V5R3
- **Setting up monitoring in z/VM:**
  - Permit write access to monitor stream (option in z/VM user directory)
    - `OPTION APPLMON`
  - Enable selected sample records and events:
    - `MONITOR SAMPLE ENABLE APPLDATA ALL`
    - `MONITOR EVENT ENABLE APPLDATA ALL`

appldata – Linux monitoring modules

- **Linux monitoring data collected by appldata_os as processed and displayed by z/VM Performance Toolkit:**



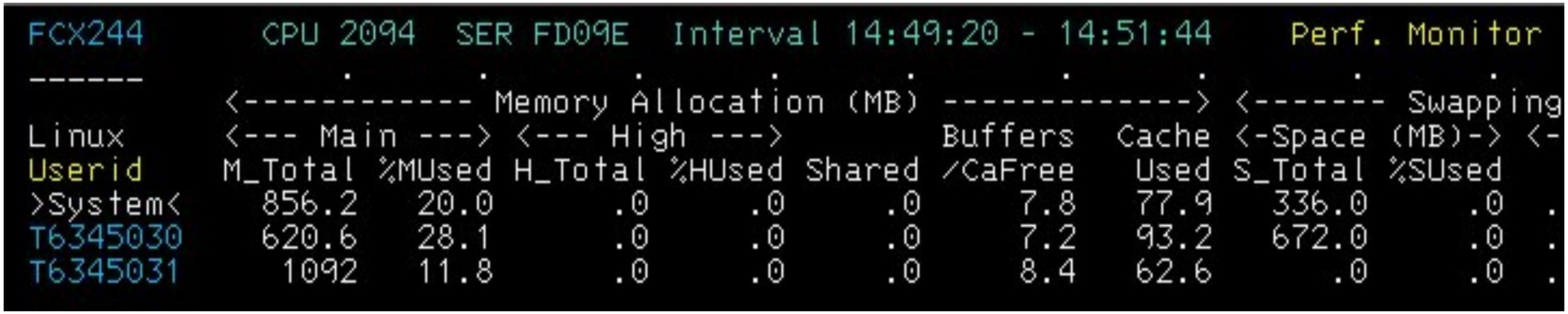
```
FCX243      CPU 2094  SER FD09E  Interval 14:49:20 - 14:51:44    Perf. Monitor
------       .      .     .      .      .      .      .      .      .      .
             <------------------- Total CPU -------------------> <----------
Linux    Virt <---------------- Utilization (%) ----------------> <---- Curren
Userid   CPUs TotCPU  User Kernel  Nice   IRQ SoftIRQ IOWait  Idle Runabl Waiti
>System<  2.0    8.6   7.7     .9    .0    .0      .0     .1 191.3    2.0
T6345030    2    5.3   4.7     .6    .0    .0      .0     .0 194.6      2
T6345031    2   11.9  10.6    1.2    .0    .0      .0     .2 188.0      2
```



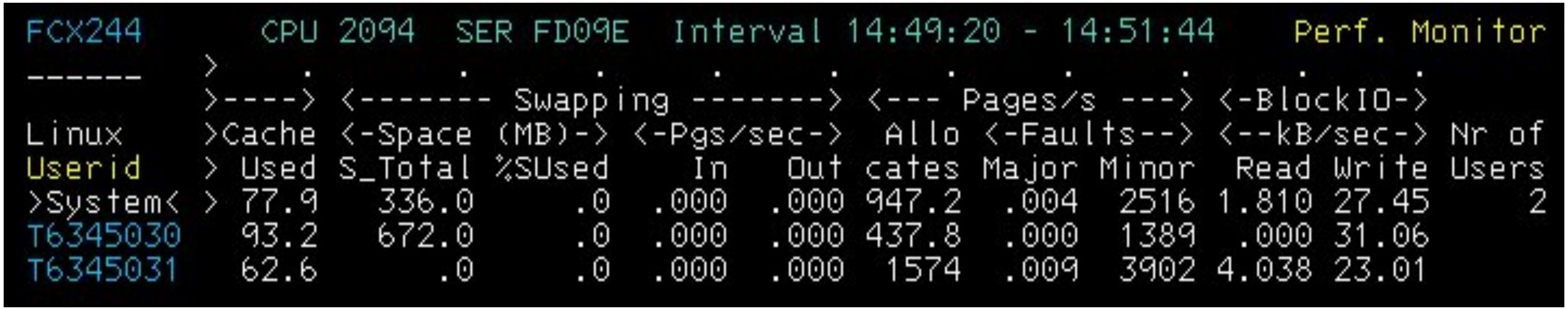
```
FCX243      CPU 2094  SER FD09E  Interval 14:49:20 - 14:51:44    Perf. Monitor
------       .>    .      .     .      .      .      .      .      .      .
              >----------------> <------------- Processes ------------->
Linux    Virt>----------------> <---- Current -----> <-Average Running-> Nr of
Userid   CPUs>tIRQ IOWait  Idle Runabl Waiting Total 1_Min  5_Min 15_Min Users
>System<  2.0>   .0     .1 191.3    2.0      .0  47.5   .11    .19    .08     2
T6345030    2    .0     .0 194.6      2       0    40   .04    .09    .03
T6345031    2    .0     .2 188.0      2       0    55   .17    .28    .13
```

appldata – Linux monitoring modules

- **Linux monitoring data collected by appldata_mem as processed and displayed by z/VM Performance Toolkit:**



```
FCX244      CPU 2094   SER FD09E  Interval 14:49:20 - 14:51:44    Perf. Monitor
------          .      .       .      .      .       .      .       .      .
          <------------ Memory Allocation (MB) -------------> <------- Swapping
Linux     <--- Main ---> <--- High --->        Buffers  Cache <-Space (MB)-> <-
Userid    M_Total %MUsed H_Total %HUsed Shared /CaFree   Used S_Total %SUsed
>System<    856.2   20.0      .0     .0     .0     7.8   77.9   336.0     .0  .
T6345030    620.6   28.1      .0     .0     .0     7.2   93.2   672.0     .0  .
T6345031     1092   11.8      .0     .0     .0     8.4   62.6      .0     .0  .
```



```
FCX244      CPU 2094   SER FD09E  Interval 14:49:20 - 14:51:44    Perf. Monitor
------   >    .       .      .     .     .     .     .     .     .     .
         >----> <------- Swapping -------> <--- Pages/s ---> <-BlockIO->
Linux    >Cache <-Space (MB)-> <-Pgs/sec->  Allo <-Faults--> <--kB/sec-> Nr of
Userid   > Used S_Total %SUsed    In   Out cates Major Minor  Read Write Users
>System< > 77.9   336.0     .0  .000  .000 947.2  .004  2516 1.810 27.45     2
T6345030   93.2   672.0     .0  .000  .000 437.8  .000  1389  .000 31.06
T6345031   62.6      .0     .0  .000  .000  1574  .009  3902 4.038 23.01
```

monwriter – Linux monitor record writer

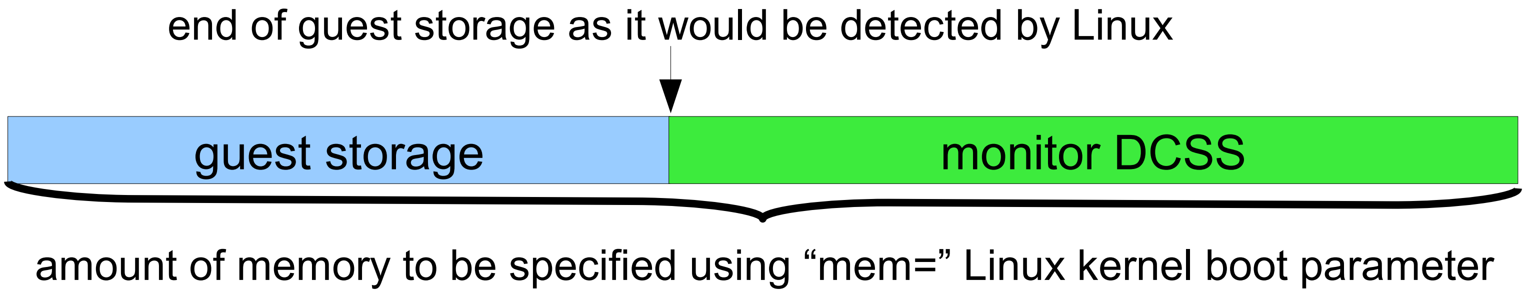
- **Linux kernel module which allows Linux applications to feed monitor records into z/VM monitor stream**
- **monwriter enables user space daemons**
  - mon_fsstatd: filesystem related data (SLES10 SP1)
  - mon_procd: process related data (SLES10 SP2)
  - record format description see “Device drivers, Features and Commands”.
- **monwriter in comparison to appldata:**
  - similar to appldata with regard to use of z/VM monitor service
  - similar to appldata with regard to z/VM setup procedure
  - monwriter: data gathered in user space; appldata: data gathered in kernel
- **/dev/monwriter**
  - write-only character device

monreader – Linux monitor record reader

- **Linux kernel module for reading z/VM monitor stream**
  - Linux kernel 2.6.10, SLES9 SP2, SLES10, RHEL5
- **/dev/monreader exposes monitor records**
  - read-only character device
  - attention: reader should discard data and retry if reading is not terminated by zero byte read
- **Raw format as retrieved from monitor stream**
  - similar to data retrieved with the MONWRITE CMS command

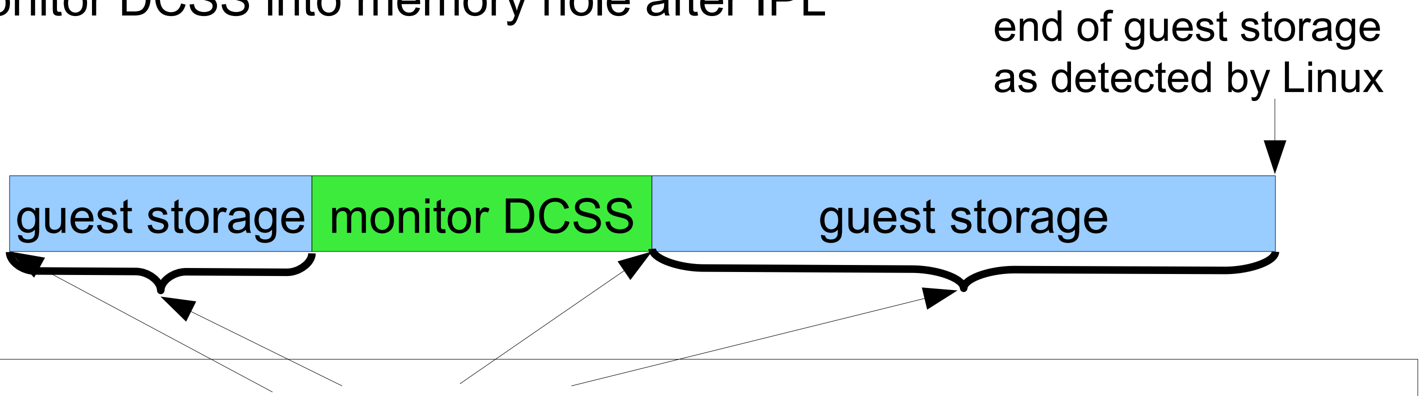
monreader – Linux monitor record reader

- **z/VM user directory entry required**
  - `IUCV *MONITOR`
  - `NAMESAVE <name of monitor DCSS>`
- **setting up access to monitor DCSS – with guest storage limited by position of monitoring DCSS:**
  - specify “mem=” boot parameter to make Linux memory management leave room beyond detectable guest storage for monitor DCSS
  - map monitor DCSS on top of detected guest storage after IPL


end of guest storage as it would be detected by Linux
guest storage
monitor DCSS
amount of memory to be specified using “mem=” Linux kernel boot parameter

monreader – Linux monitor record reader

- **Setting up access to monitor DCSS – with memory hole:**
  - memory hole detected by Linux at IPL
  - map monitor DCSS into memory hole after IPL



```
CP DEF STOR CONFIG 0.144M 160M.8642M
STORAGE = 1008M
Storage Configuration:
0.144M 160M.864M
Extent Specification                 Address Range
----------------------------------- -----------------------------------
                 0.140M             0000000000000000 - 0000000008BFFFFF
              160M.864M             000000000A000000 - 000000003FFFFFFF
Storage cleared - system reset.
```

hypfs – hypervisor data

- Filesystem exposing LPAR and z/VM hypervisor data
  - guest systems hosted by hypervisor
  - resources controlled by hypervisor, i.e. physical CPUs
  - resources provided to guest systems, i.e virtual CPUs
- Utilises DIAG calls
  - DIAG 0x204 – LPAR hypervisor data
    - DIAG 0x224 – CPU type name table
  - DIAG 0x2FC – CPU and memory accounting data (z/VM 5.3)
- Differences between hypfs on LPAR and z/VM
  - hypfs exposes z/VM specific data if running in z/VM
  - hypfs is unavailable if z/VM doesn't support DIAG 0x2FC:

```
# mount none -t hypfs /sys/hypervisor/s390
mount: unknown filesystem type 'hypfs'
```

hypfs – hypervisor data

- hypfs needs to be mounted

```
sample entry for /etc/fstab:
none /sys/hypervisor/s390 s390_hypfs defaults 0 0
```

- hypfs is populated with initial data when being mounted
- hypfs data is only updated on request

```
echo 1 > /sys/hypervisor/s390/update
```

hypfs – LPAR hypervisor data

```
/sys/hypervisor/s390
|-- update
|-- cpus
|   |-- <cpu-id>
|   |   |-- mgmtime
|   |   `-- type
|   `-- [...]
|-- hyp
|   `-- type
`-- systems
    |-- <lpar-name>
    |   `-- cpus
    |       `-- <cpu-id>
    |       |   |-- cputime
    |       |   |-- mgmtime
    |       |   |-- onlinetime
    |       |   `-- type
    |       `-- [...]
    |
    `-- [...]
```

- hyp/type: “LPAR hypervisor”
- cpus: physical CPU data
  - type: “CP” or “IFL”
  - mgmtime: LPAR overhead *
- systems: logical CPU data for all LPARs
  - type: “CP” or “IFL”
  - mgmtime: LPAR overhead *
  - cputime: actual use time *
  - onlinetime: time since activation *

* all times in microseconds

hypfs – z/VM hypervisor data

```
/sys/hypervisor/s390
|-- update
|-- cpus
|   `-- count
|-- hyp
|   `-- type
`-- systems
    |-- <guest-name>
    |   |-- onlinetime_us
    |   |-- cpus
    |   |   |-- capped
    |   |   |-- count
    |   |   |-- cputime_us
    |   |   |-- dedicated
    |   |   |-- weight_cur
    |   |   |-- weight_max
    |   |   `-- weight_min
    |   |
    |
```

- hyp/type: “z/VM hypervisor”
- cpus/count: number logical CPUs controlled by z/VM
- systems/onlinetime_us: time since guest activation
- systems/cpus:
  - capped: 0=off, 1=soft, 2=hard
  - count: number of virtual CPUs
  - cputime_us: actual use time
  - dedicated: 0=no, 1=yes
  - weight_cur, weight_min, weight_max: current, minimum and maximum share of guest (1-10000; 0=ABSOLUTE SHARE)

hypfs – z/VM hypervisor data

```
/sys/hypervisor/s390
 [...]
`-- systems
    |-- <guest-name>
    |    [...]
    |   |-- mem
    |   |   |-- max_KiB
    |   |   |-- min_KiB
    |   |   |-- share_KiB
    |   |   `-- used_KiB
    |   `-- samples
    |       |-- cpu_delay
    |       |-- cpu_using
    |       |-- idle
    |       |-- mem_delay
    |       |-- other
    |       `-- total
    `-- [...]
```

- systems/mem:
  - max_KiB: memory limit granted to guest
  - min_KiB: minimum memory requirement of guest
  - share_KiB: suggested guest memory size estimated by z/VM
  - used_KiB: current memory footprint of guest
- systems/samples:
  - cpu_delay: guest waiting for CPU
  - cpu_using: guest doing work
  - idle: guest being idle
  - mem_delay: guest waiting for memory to be paged in
  - other: other samples
  - total: total samples

Resources

- Linux documentation (october 2005 stream)
  - “Linux on System z - Device Drivers, Features, and Commands”
    - Monitoring of z/VM guests (appldata, monwriter, monreader)
    - Hypervisor data (hypfs)
  - “How to use Execute-in-Place Technology with Linux on z/VM”

    www.ibm.com/developerworks/linux/linux390/

- z/VM documentation (version 5 release 3)
  - z/VM data areas, control blocks, and monitor records

    www.vm.ibm.com/pubs/ctlblk.html
  - z/VM CP Commands and Utilities Reference
    - `MONITOR, QUERY MONITOR, NAMESAVE`
  - z/VM Performance Toolkit
    - screens: FCX227, FCX228, FCX229, FCX230
  - z/VM Performance
    - `IUCV *MONITOR`
  - ***www.ibm.com/servers/eserver/zseries/zos/bkserv/zvmpdf/zvm53.html***

/proc/sysinfo – System information

```
# cat /proc/sysinfo
Manufacturer:          IBM
Type:                  2094
Model:                 715  S18
Sequence Code:  00000000000D6AAD
Plant:                 02
Model Capacity:        715

CPUs Total:            20
CPUs Configured:       15
CPUs Standby:          0
CPUs Reserved:         5
Capability:            1456 1920
Adjustment 02-way:     245 249
...
Adjustment 20-way:     174 178
Secondary Capability: 1456
...
```

```
...
LPAR Number:                 31
LPAR Characteristics:  Shared
LPAR Name:                   T29LP30
LPAR Adjustment:             800
LPAR CPUs Total:             15
LPAR CPUs Configured:  12
LPAR CPUs Standby:           3
LPAR CPUs Reserved:          0
LPAR CPUs Dedicated:   0
LPAR CPUs Shared:            12

VM00 Name:                   T2930041
VM00 Control Program:  z/VM 5.2.0
VM00 Adjustment:             333
VM00 CPUs Total:             4
VM00 CPUs Configured:  4
VM00 CPUs Standby:           0
VM00 CPUs Reserved:          0
```

Acknowledgements

- I would like to thank for providing material and fielding questions
  - Christian Borntraeger
  - Michael Holzheu
  - Carsten Otte
  - Gerald Schaefer
  - Martin Peschke

Trademarks

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, DBE, e-business logo, 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

The following are trademarks or registered trademarks of other companies

Lotus, Notes, and Domino are trademarks or registered trademarks of Lotus Development Corporation
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
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.