GSE UK Conference 2012: Session OJ
z/OS Application Performance Tuning: A Cost Efficient & Low Overhead Solution
Introduction - Agenda

- Application Development Lifecycle: Simple & Detailed
- Application Performance Tuning: Software Costs/Business Value
- Application Performance Management (APM): ITSM & z/OS Tools
- CPU Measurement Facility (CPU MF): Introduction & Overview
- CPU MF: Activation, Considerations, Usage & Hidden Potential
- CPU MF Reprise: No Brainer Deployment Recommendations
- Application Performance: CPU MF & zHISR Benefits Summary
- CPU MF & zHISR: Useful Information Resources
Simple Application Development Lifecycle

Assess Needs

Design Specifications

Develop & Test Software Code

Implement Systems

Support & Maintain

Evaluate Performance

Classic Life Cycle Model

How often do we finish the project & evaluate performance?
Can we really identify an SLA before Production rollout?

14 November 2012
Application Performance Tuning: Software Costs

Sample IBM zSeries Server Pricing Configurations Using Industry Standard Pricing

<table>
<thead>
<tr>
<th>Program #</th>
<th>Software Description</th>
<th>MLC</th>
<th>Program #</th>
<th>Software Description</th>
<th>MLC</th>
</tr>
</thead>
<tbody>
<tr>
<td>5635DB2</td>
<td>DB2 for z/OS</td>
<td>~£15,000</td>
<td>5635DB2</td>
<td>DB2 for z/OS</td>
<td>~£22,500</td>
</tr>
<tr>
<td>5655M15</td>
<td>CICS TS for z/OS</td>
<td>~£16,000</td>
<td>5655M15</td>
<td>CICS TS for z/OS</td>
<td>~£23,500</td>
</tr>
<tr>
<td>5694A01</td>
<td>z/OS V1 Base</td>
<td>~£22,000</td>
<td>5694A01</td>
<td>z/OS V1 Base</td>
<td>~£40,000</td>
</tr>
<tr>
<td>IBM WLC Total Monthly Cost</td>
<td>~£53,000</td>
<td>IBM WLC Total Monthly Cost</td>
<td>~£86,000</td>
<td></td>
<td></td>
</tr>
<tr>
<td>IBM WLC Annual Saving @ 10%</td>
<td>~£63,000</td>
<td>IBM WLC Annual Saving @ 10%</td>
<td>~£103,000</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

But, Industry Averages State ~$2,000-$5,000 (~£1,500-£3,500) Cost per Year for Software MIPS

| Total z/OS Software Annual Cost | ~£750,000 | Total z/OS Software Annual Cost | ~£1,500,000 |
| Total z/OS Annual Saving @ 10%     | ~£75,000 | Total z/OS Annual Saving @ 10%     | ~£150,000 |

- Core z/OS products are a significant component of overall cost...
- Reducing MIPS/MSU consumption is of course the biggest saving!
- MIPS/MSU prices reduce year-on-year, but software stays the same

Even for a small user, 10% savings are conservative & worthwhile!

14 November 2012
Application Performance Tuning: Business Value

<table>
<thead>
<tr>
<th>Industry Sector</th>
<th># MIPS per $1M Revenue</th>
<th>MIPS Cost per $1M Revenue</th>
<th>Hour Downtime Cost ($ Millions)</th>
<th>Value: Pseudo Priceless Business Consideration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Banking</td>
<td>0.98</td>
<td>$8,441</td>
<td>1.0</td>
<td>Customer Satisfaction</td>
</tr>
<tr>
<td>Consumer/Retail</td>
<td>0.19</td>
<td>$2,520</td>
<td>1.1</td>
<td>End User Productivity</td>
</tr>
<tr>
<td>Financial Services</td>
<td>1.07</td>
<td>$9,574</td>
<td>1.4</td>
<td>Investor Confidence</td>
</tr>
<tr>
<td>Insurance</td>
<td>0.33</td>
<td>$3,143</td>
<td>1.2</td>
<td>Goodwill &amp; Brand Image</td>
</tr>
<tr>
<td>Manufacturing</td>
<td>0.21</td>
<td>$2,190</td>
<td>1.6</td>
<td>Lost Business (Revenue)</td>
</tr>
<tr>
<td>Telecommunications</td>
<td>0.85</td>
<td>$6,397</td>
<td>2.0</td>
<td>Regulatory Compliance</td>
</tr>
<tr>
<td>Transportation</td>
<td>0.23</td>
<td>$3,222</td>
<td>0.6</td>
<td>B2B Contracts (T’s &amp; C’s)</td>
</tr>
<tr>
<td>Utilities</td>
<td>0.16</td>
<td>$1,549</td>
<td>2.8</td>
<td>Penalties/Late Fees</td>
</tr>
</tbody>
</table>

Source: IT Performance Engineering & Measurement Strategies - Quantifying Performance Loss, META Group
Source: Economics of Computing -The Internal Combustion Mainframe, Rubin

- No two businesses are the same, downtime cost is always subjective
- Costs can be calculated, but arguably business value is intangible
- Having happy customers and end users is the goal and priceless!

_The definition of a cynic is someone that knows the price of everything, and the value of nothing – Oscar Wilde_
Who owns Application Performance Management? Does it matter?

14 November 2012
Application Performance Management (APM) for z/OS

- Strobe Family
- Application Performance Analyzer
- ExpeTune
- ca technologies
- Application Tuner
- TriTune
- ASG Software Solutions
- TriTune
- BMC Software

- All the major ISV’s include an APM tool in their software portfolio
- Most if not all products were architected ~10-20 years ago, pre z10
- Activating the tool in Production generates CPU overhead concerns

A mature market of medium-high cost software that works on the fundamental premise of diagnosis by capturing CPU instructions...

14 November 2012
CPU Measurement Facility (CPU MF): Introduction

- Introduced with the z10 server, the CPU Measurement Facility provides optional hardware assisted collections of information about the logical CPUs work that is executed over a specified interval in selected logical partitions (LPAR).

- CPU MF does not replace existing functions (E.g. Capacity Planning, System Tuning, et al) or capabilities (E.g. SMF), but provides new function to assist with CPU optimization activities for Application Code and ISV Software tuning.

- CPU MF consists of two important, but independent, functions:
  1) The collection of **counters (workload)** that maintain counts of certain activities (The counters function is intended to be run on a constant basis to collect long-term performance data, in a similar manner to how you collect other performance data).
  2) The collection of **samples (application)** that provide information about precisely what the CPU is doing at the time of the sample (The sampling function is a short duration, precise function that identifies where CPU resources are being used, to help you improve application efficiency).

- CPU MF runs independently at the LPAR level and can run in multiple LPARs simultaneously and can be used with Central Processors (CPs), IBM System z Integrated Information Processor (zIIP), and IBM System z Application Assist Processor (zAAP).

**CPU MF is packaged in the CPU chip, easy-to-use and zero cost!**
CPU MF: Workload Performance Observations

Historically, LSPR workload capacity curves (primitives & mixes) had application names, being identified by a “software class” captured characteristic. However, CPU, capacity performance is more closely associated with how a workload is interacting with a processor “hardware” design. With the availability of CPU MF (z10+), the ability to gain an insight into the interaction of workload and hardware profiles has arrived...

Capacity (zPCR) performance has always been more closely associated with how a workload uses and interacts with a particular processor hardware design. The challenge has been that there was no ability to get insight into the interaction of workload and hardware design. CPU MF addresses this challenge by providing information about the interaction that was not available previously.

**CPU MF delivers a Capacity Planning ability based on customer data**
CPU MF: z/OS Operating System Interaction (Brief)

1) Ensure that the prerequisite hardware and software service levels are installed:
   Hardware (z10, z114/z196, zEC12), Software (z/OS 1.11+ or z/OS 1.8 plus APARs)

2) Authorize the collection of CPU MF data at the logical partition (LPAR) level using the HMC or SE:
   HMC Select Processor → Recovery → Single Object Operations → System Management → CPC Operational Customization → Change LPAR Security → Choose From Several Counters Settings

3) Define a User ID (RACF, ACF2, TopSecret) for the HIS started task:
   No special considerations for the HIS User ID other than STC and UNIX file system (OMVS) access

4) Ensure that SMF is set up to allow the collection of CPU MF SMF records:
   Safeguard Type 113 records are not suppressed, only the SYS parameter required (not SUBSYS)

5) Ensure that the HIS started task has an appropriate WLM service class:
   HIS requires access to CPU resources and the ability to offload data to UNIX HFS @ nn MB/Minute

6) Set up the UNIX file system that will contain the HIS files:
   Define multiple file systems, one for each system; Have a unique mount point for each one; Mount (Automount) each one on the system that writes to that file system.

7) Modify your SMF archiving processes to save the SMF type 113 records:
   Safeguard that any IFASMFDP (DUMP-CLEAR) activities include the offloading of the HIS 113 record. For categorization, although RMF/CMF does not directly support the 113 record, grouping this record with other CPU related records (E.g. Type 7n) seems logical.

Hardware Instrumentation Services (HIS) - z/OS CPUMF Interaction

14 November 2012
**CPU MF: z/OS HIS Address Space Operation Control**

BEGIN the HIS data collection (collect hardware event data) process

**TITLE** (identify) the collection (E.g. Production CICS Widgets Application)

z/OS UNIX System Services **PATH** for .MAP, .CNT and .SMP files

Command File **(DDNAME)** for controlling (z/OS Modify) HIS parameters

**COUNTERS** indicates that this event collection is for system-wide analysis

**CTRONLY** B (Basic)+P(Problem) default, C(Crypto) & E(Extended) options

**CTRSET** collect only event counter set data (.CNT)

Specify the **DURATION** in Minutes (1-1440, default 10)

**SAMPLINGS** indicates detailed event collection (Job or Address Space)

**BUFCNT** is system calculated if not specified, to minimize event DATALOSS

DATALOSS should not occur if WLM service is OK or BUFCNT is sufficient

**SAMPFREQ** defaults to 800000 (10 Minutes), reduce for smaller CPC

**SAMPTYPE** defaults to B(Basic), D(Diagnostic) also includes B(Basic)

Specify the **DURATION** in Minutes (1-1440, default 10)

**MAPS** indicates the granularity of load module mapping data (.MAP)

**MAPONLY** dictates load module CSECT type analysis for an ASID or JOB

**MAPASID** address space ID(s) for private load module data (1-32 or ALL)

**MAPJOB** job name(s) for private load module data (1-32 (wildcards) or *)

**MAPVERBOSE** defaults (no), collect load module map error diagnostics

Miscellaneous and optional keywords

**SMFINVAL** default is 15 (1-60) or **SYNC** hronise with global SMF interval

**STAECHANGE** control action if CPU status changes (default SAVE)

END the HIS data collection process

**HIS control is simple with delineation for system & load module data**

14 November 2012
### CPU MF: SAMPLING Rates & Related CPU Overhead

<table>
<thead>
<tr>
<th>CPU Model ID</th>
<th>Sampling Rate/Minute</th>
<th>Max %CPU Overhead</th>
<th>CPU Model ID</th>
<th>Sampling Rate/Minute</th>
<th>Max %CPU Overhead</th>
</tr>
</thead>
<tbody>
<tr>
<td>2097-701</td>
<td>800,000</td>
<td>~1.0-2.0</td>
<td>2098-O01</td>
<td>165,000</td>
<td>~0.2-0.4</td>
</tr>
<tr>
<td>2097-601</td>
<td>550,000</td>
<td>~0.5-1.0</td>
<td>2098-N01</td>
<td>145,000</td>
<td>~0.2-0.4</td>
</tr>
<tr>
<td>2097-501</td>
<td>405,000</td>
<td>~0.5-1.0</td>
<td>2098-M01</td>
<td>130,000</td>
<td>~0.1-0.2</td>
</tr>
<tr>
<td>2097-401</td>
<td>190,000</td>
<td>~0.25-0.5</td>
<td>2098-L01</td>
<td>110,000</td>
<td>~0.1-0.2</td>
</tr>
<tr>
<td>2098-Z01</td>
<td>580,000</td>
<td>~0.75-1.5</td>
<td>2098-K01</td>
<td>100,000</td>
<td>~0.1-0.2</td>
</tr>
<tr>
<td>2098-Y01</td>
<td>530,000</td>
<td>~0.75-1.5</td>
<td>2098-J01</td>
<td>85,000</td>
<td>~0.1-0.2</td>
</tr>
<tr>
<td>2098-X01</td>
<td>470,000</td>
<td>~0.66-1.3</td>
<td>2098-I01</td>
<td>80,000</td>
<td>~0.1-0.2</td>
</tr>
<tr>
<td>2098-W01</td>
<td>420,000</td>
<td>~0.5-1.0</td>
<td>2098-H01</td>
<td>70,000</td>
<td>&lt;0.1</td>
</tr>
<tr>
<td>2098-V01</td>
<td>370,000</td>
<td>~0.5-1.0</td>
<td>2098-G01</td>
<td>65,000</td>
<td>&lt;0.1</td>
</tr>
<tr>
<td>2098-U01</td>
<td>330,000</td>
<td>~0.5-1.0</td>
<td>2098-F01</td>
<td>50,000</td>
<td>&lt;0.1</td>
</tr>
<tr>
<td>2098-T01</td>
<td>290,000</td>
<td>~0.3-0.6</td>
<td>2098-E01</td>
<td>50,000</td>
<td>&lt;0.1</td>
</tr>
<tr>
<td>2098-S01</td>
<td>265,000</td>
<td>~0.3-0.6</td>
<td>2098-D01</td>
<td>40,000</td>
<td>&lt;0.1</td>
</tr>
<tr>
<td>2098-R01</td>
<td>230,000</td>
<td>~0.3-0.6</td>
<td>2098-C01</td>
<td>35,000</td>
<td>&lt;0.1</td>
</tr>
<tr>
<td>2098-Q01</td>
<td>210,000</td>
<td>~0.25-0.5</td>
<td>2098-B01</td>
<td>30,000</td>
<td>&lt;0.1</td>
</tr>
<tr>
<td>2098-P01</td>
<td>190,000</td>
<td>~0.25-0.5</td>
<td>2098-A01</td>
<td>20,000</td>
<td>&lt;0.1</td>
</tr>
</tbody>
</table>

A lower sampling rate reduces CPU overhead proportionally

As the number of LPAR logical CPUs increases, the overhead (% total capacity) decreases

For counters (SMF) data, CPU overhead cost is so small, it's immeasurable. Therefore it's recommended to turn on the collection of counters data in all LPs on a permanent basis.

Bottom Line: ~1/100 of a CPU second for HIS address space in a 15 Minute SMF interval!

Arguably Application diagnostic CPU usage concerns are eradicated!

14 November 2012
The files that HIS uses for COUNTERS data have the SYSHISyyymmmdd.hhmmss.CNT naming convention. Each of the separate HIS file types has a unique low level qualifier. The CNT file type is always used for counters files. This file contains the delta values for the entire collection period. It also contains the counts for all online PUs. For sampling data, HIS creates one file for each active logical CPU in the system as per SYSHISyyymmmdd.hhmmss.SMP.cpu#.

**Basic information at first glance, but there is precious gold, if you dig...**

14 November 2012
If you request the collection of map information, HIS creates a human-readable file containing the start and end address of every program in the MVS common area. This file contains the start and end address of every program in the private area of every swapped-in address space, depending on which options you specify. The file also contains information about the system that HIS was run on and a storage map showing the various parts of virtual storage.

The virtual storage can be in the Common Service Area (CSA), Extended Common Service Area (ECSA), Nucleus, and so on. HIS uses the following naming convention for these files:

SYSHISyyymmmdd.hhmmss.MAP

The first part of the file name follows the same convention as the counters and sampling files.

The low level qualifier is MAP. HIS only creates one file, regardless of the number of logical CPUs in the LPAR, or the number of address spaces that were requested on the MAPASID keyword.
CPU MF: SMF 113 Impact & Potential In One Slide!

An SMF 113 record is 452 bytes in size for each LP per SMF interval. The example below is for 3 z196 CPC’s, 10 LPAR’s and SMF recording for 5 * 24 hour days. SMF 113 data size is ~1% when compared with Type 70/72!

![SMF 113 record size example](image)

Compare zSeries Server evolution, incorporate new CPU MF data into zPCR/LSPR Capacity Planning disciplines

![CPU MF data comparison](image)

**CPI**: Cycles per Instruction  
**Prb State**: % Problem State  
**Est Instr Cmplx CPI**: Estimated Instruction Complexity CPI (infinite L1)  
**Est Finite CPI**: Estimated CPI from Finite cache/memory  
**Est SCPL1M**: Estimated Sourcing Cycles per Level 1 Miss  
**L1MP**: Level 1 Miss Per 100 instructions  
**L15P**: % sourced from L1.5 cache  
**L2P**: sourced from Level 2 cache  
**L2LP**: sourced from Level 2 Local cache (on same book)  
**L2RP**: sourced from Level 2 Remote cache (on different book)  
**L3P**: sourced from Level 3 on same Chip cache (New resource for z196/z114)  
**L4LP**: sourced from Level 4 Local cache (on same book)  
**L4RP**: sourced from Level 4 Remote cache (on different book)  
**MEMP**: % sourced from Memory  
**Rel Nest Intensity**: distribution and latency from shared caches and memory  
**LPARCPU - APPL%**: (GCPs, zAAPs, zIIPs) (un)captured

Activate CPU MF Counters, collect SMF 113 records, model workloads!

14 November 2012
IBM provides an insight into what potential is available with CPU MF (HIS) sampling data (load module) for a “CPU usage hot spot summary” @ www-03.ibm.com/systems/z/os/zos/features/unix/tools/hisreport.html ...

CPU MF, lots of data, little usable information! Is there a better way?

14 November 2012
**CPU MF Deployment: No Brainer Recommendations**

**Recommendation:** Turn on HIS COUNTERS facility to generate CPU MF SMF 113 records for all LPARs on all z10, z196/z114 & zEC12 servers. Incorporate this data into zPCR activities, and learn how LSPR categories are evolving based on meaningful workload categories (Relative Nest Intensity – RNI). Therefore CPU MF (SMF 113) data can assist you in identifying why workload performance might have changed. Consider producing a CPU hot spot analysis using HIS SAMPLES data, collaborating with Application Support.

**Recommendation:** Collaborate with Technical Support and consider introducing or enhancing Application Performance Tuning within the Development lifecycle (I.E. Prepare/Execute Performance/STress/Failover Test), before Production implementation, based upon meaningful and real-life data from your installation, namely CPU MF SAMPLING data. Identify a methodology for highlighting the module/csect that is producing the high CPU usage, from the CPU MF SAMPLING data, either writing your own reports, or using a software tool. Therefore tuning your code before Production roll-out.

**Recommendation:** Regardless of who “owns” the Application Performance Tuning process, Operations staff are not robots, but they are at the sharp end of customer support, and they experience the System and related Applications, day in and day out. Both the Technical and Application Support teams should consider a “back to the floor” activity, spending a day and night shift with Operations, for a normal and peak day, learning accordingly. Worst case, maybe a new team member might be identified!

*If CPU MF is available, it’s zero cost, deploy the function & find benefit!*

14 November 2012
zHISR Introduction: Application CPU Hot Spot Analysis

zHISR uses data created by CPU MF & z/OS Hardware Instrumentation Services (HIS) on IBM System z10 & zEnterprise mainframe processors to produce CPU hotspot analysis reports, which can be used to help tune applications by locating the specific sections of code that are the biggest CPU consumers.

zHISR includes a z/OS UNIX file system interface for navigating, browsing & deleting files, including HIS collection files. Programmers can start or stop a HIS data collection event or view the status of a current or prior run. If zIIP specialty engines are available, nearly all zHISR CPU processing is redirected to a zIIP.

For ease of use, zHISR includes a familiar fully CUA compliant ISPF interface, allowing System Programmers, Application Developers & others to manage an Application CPU Hot Spot analysis from start to end, with an intuitive and logical flow, for managing the raw HIS data & subsequent formatted screens & reports.

From a reporting viewpoint, zHISR performs an intelligent data reduction analysis for the raw HIS data files, producing intelligent and easy-to-follow information reports, in both detailed and summary format, allowing “Office” type reporting (E.g. PC) for Managers, Technicians & Business Users alike.

zHISR, an easy-to-use, low resource & cost optimized productivity tool!

14 November 2012
The zHISR system allows you to control the HIS collection, manage event collection data files, perform data reduction & analysis activities, identify high CPU users & associated source code.
The zHISR Primary Menu has a Directory field for entering a UNIX file system path (/his is typically the installation default) as well as presenting list of files you are currently working with.
### zHISR Working Overview: UNIX File System Dialog

The zHISR UNIX File System Directly Display lists the HIS data collection event (MAP, CNT, SMP) files. Use the B or BL command to browse a HIS event file. BL prompts for logical record length.

The details of the file system display are as follows:

- **Dir**: Directory
- **File**: Files
- **Type**: Type of file
- **Permission**: Permissions and ownership
- **Group**: File group name
- **FileName**: File name

**Example Command Output**

```
04/06/2011  zHISR: UNIX File System Directory Display 12:03:11

Directory /u/his

<table>
<thead>
<tr>
<th>Type</th>
<th>Permission</th>
<th>Owner</th>
<th>Group</th>
<th>FileName</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dir</td>
<td>drwxr-xr-x</td>
<td>HIS</td>
<td>DEV</td>
<td>.</td>
</tr>
<tr>
<td>Dir</td>
<td>dr-xr-xr-x</td>
<td>SYSOPER</td>
<td>TTY</td>
<td>..</td>
</tr>
<tr>
<td>File</td>
<td>-rwxr-xr-x</td>
<td>HIS</td>
<td>SYS1</td>
<td>SYSHIS20081210.113303.CNT</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>3303.MAP</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>3303.SMP.00</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>3303.SMP.01</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>3303.SMP.02</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>4324.CNT</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>4324.SMP.00</td>
</tr>
<tr>
<td>File</td>
<td>-rwxr-xr-x</td>
<td>HIS</td>
<td>SYS1</td>
<td>SYSHIS20090205.194324.SMP.01</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>3303.MAP</td>
</tr>
<tr>
<td>File</td>
<td>-rwxr-xr-x</td>
<td>HIS</td>
<td>SYS1</td>
<td>SYSHIS20090205.194324.SMP.02</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>3303.MAP</td>
</tr>
<tr>
<td>File</td>
<td>-rwxr-xr-x</td>
<td>HIS</td>
<td>SYS1</td>
<td>SYSHIS20090205.194324.SMP.03</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>3303.MAP</td>
</tr>
<tr>
<td>File</td>
<td>-rwxr-xr-x</td>
<td>HIS</td>
<td>SYS1</td>
<td>SYSHIS20090214.100114.CNT</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>3303.MAP</td>
</tr>
<tr>
<td>File</td>
<td>-rwxr-xr-x</td>
<td>HIS</td>
<td>SYS1</td>
<td>SYSHIS20090214.100114.MAP</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>3303.MAP</td>
</tr>
<tr>
<td>File</td>
<td>-rwxr-xr-x</td>
<td>HIS</td>
<td>SYS1</td>
<td>SYSHIS20090214.100114.SMP.00</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>3303.MAP</td>
</tr>
<tr>
<td>File</td>
<td>-rwxr-xr-x</td>
<td>HIS</td>
<td>SYS1</td>
<td>SYSHIS20090214.100114.SMP.01</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>3303.MAP</td>
</tr>
</tbody>
</table>
```

**Note:**

- **BL** command prompts for LRECL (2-255) to be used or press PF4 to accept the shown default value: **072**
- **3=End 4=Accept**
zHISR Working Overview: HIS Event File Browse Mode

The Browse Mode dialog opens the HIS Event File in human readable format, displaying the header information, allowing the user an opportunity to confirm their HIS file creation/selection.

14 November 2012
The Event Data Collection dialog allows the user to control (start, stop, specify parameters) their HIS data collection activity. Pressing PF2 from the Primary Menu invokes the HIS Collection Panel.
The user can Analyze a data collection run by typing an A in the command column. zHISR then invokes an intuitive dialog via a series of ISPF panels to limit & refine the scope of your analysis.
zHISR Working Overview: Event Data Analysis - CPU

The CPU panel lets you select (X) 1 or more CPUs for analysis. The last 2 digits of the displayed file name represent the CPU number in hexadecimal. Press PF5 to continue to the next screen.
Use the HASN Panel to select the Home Address Space ID of the address space that is the target for analysis. From here, you can select (PF5) the ASID for Analysis, & perform the Analysis (PF2).
The zHISR Module Panel, allows you to choose the modules for Analysis. A generic masking ability allows you to include or exclude multiple modules.
The zHISR Csect Panel, allows you to choose the module Csect(s) for Analysis. Once again, a generic masking ability allows you to include or exclude multiple module Csects.
The zHISR Boundary Panel, allows you to choose the choose which working storage areas should be included. A generic masking ability allows you to include or exclude multiple selections.
The Full Analysis Panel displays the selected data, sorted by percentage of total (CPU) entries (sort by any column). Place your cursor on an entry; press Enter & invoke the Spot Analysis Panel.
The Spot Analysis Panel is a detailed display of the Load Module activity, sorted by PSW Address (sort by any column). The Tot_Pct column will highlight the highest CPU usage resource (offset).
zHISR Working Overview: Event Data Analysis - ADATA

For Assembly enter the SYSADATA dataset name, for Cobol enter the SYSPRINT dataset name, correct the member name if necessary and then press the PF2 key to view the associated data.

Input dataset name: BOB1.ADATA
Target member name: ODE
Report page limit: 1000
Instruction address: 00000000_0DD10B00
Instruction offset: 00001B00

1=Help 2=Adata 3=End 6=Clear

By cross-referencing the high CPU user from the spot analysis (Instruction address/offset) with the load module Associated Data (ADATA), the user will be then able to display the source code.
Pressing **PF2** from the ADATA panel invokes the Associated Data Panel. This panel consists of multiple pages of program text & offset (highlighted) information for the selected Csect/Module.
To print the current report provide a valid SYSOUT class and press the PF4 key. To save the current report supply the name of a PDS or PDSE dataset that you are authorized to update and the member name that is to contain the report and press the PF5 key. To export the current report as a CSV file specify the output path name and press the PF7 key.

Sysout Class

Output Dataset Name

Output Member Name

Output Path Name

Character encoding ASCII

1=Help  3=End  4=Print  5=Save  6=Clear  7=Export

The user can print or save a Full Analysis, Spot Analysis or Associated Data report by pressing PF12 from any of those panels. The user can also export Full & Spot Analysis reports to a CSV file.
If the user exports their analysis results as a CSV file, they can import that file into a spreadsheet (e.g., Excel) or any program supporting CSV formats to graphically analyze and visualize their data.
As per any CUA compliant software product, zHISR has an inbuilt extensive online Help Tutorial with step-by-step, detailed instructions for using zHISR, allowing easy use without RTM activities.
Application Performance: CPU MF-zHISR Benefits Summary

**Benefit:** CPU MF (SMF 113) provides zSeries Server sizing information not previously available & should be used for Capacity Planning & System Sizing activities. The major ethos change is LSPR/zPCR sizing based on customer workload data, not “simulation” profiles. z10 GA was October 2008, and z10 is now an n-2 (i.e. z196/z114, zEC12) server, so in all likelihood, most committed Mainframe customers will deploy a CPU MF eligible server. The cost of activating CPU MF for SMF 113 records is arguably, **nothing**.

**Benefit:** Historically Application Performance Tuning has been an activity that might have been overlooked, not necessarily owned by the Application Support team. Furthermore, the cost of deploying a specialized tool, in terms of software license & CPU overhead might have been restrictive. With the CPU MF SAMPLING function, the ability to analyze CPU usage during Application testing activities now exists, with minimal (~1%) CPU overhead. If HIS data analysis code is written internally, cost is arguably, **nothing**.

**Consideration:** Simplifying the CPU MF data collection & analysis process is possible with the zHISR software, which has been designed for this specific task, using the latest programming techniques. Software licensing & CPU overhead costs are low when compared with traditional Application Performance Tuning techniques, as CPU MF SAMPLING data is used, while software licensing metrics are flexible (E.g. User or MSU based). This solution expedites the analysis of application code that could be optimized.

**CPU MF delivers benefit for no cost, zHISR offers process optimization...**
CPU MF & zHISR: Useful Information Resources

How to Benefit From Hardware Instrumentation Services Data
http://enterprisesystemsmedia.com/article/how-to-benefit-from-hardware-instrumentation-services-data

Setting Up and Using the IBM System z CPU Measurement Facility with z/OS (REDP-4727-00)

CPU MF Overview and WSC (z10) Experiences: SHARE 2010
www-03.ibm.com/support/techdocs/atsmastr.nsf/WebIndex/TC000041

CPU MF Overview and WSC (z196) Experiences: SHARE 2012
www-03.ibm.com/support/techdocs/atsmastr.nsf/WebIndex/TC000066

z/OS CPU MF Enablement Education
www-03.ibm.com/support/techdocs/atsmastr.nsf/WebIndex/PRS4922

Report Generator for Hardware Instrumentation Sample Data
www-03.ibm.com/systems/z/os/zos/features/unix/tools/hisreport.html

zHISR Product Information Introduction
www.value-4it.com/products/zHISR.html

Getting Started with zPCR (IBM's Processor Capacity Reference)
www-03.ibm.com/support/techdocs/atsmastr.nsf/WebIndex/PRS1381?OpenDocument&TableRow=4.1.1

Disclaimer: These web links, valid at the time of this document's creation, are provided for the convenience of the reader. They do not constitute endorsement for the information, products or services contained therein.