



Editorial staff: Dino Gigli, Danilo Gipponi, Fabio Massimo Ottaviani (EPV Technologies) – Jon Olley (Inspired Solutions)

In this number:

- 1) **Tech Papers** - Measuring IMS Transactions - Part 3
- 2) **Tech News** - GSE/UKCMG zCapacity Management and zPerformance Analysis Working Group

Past numbers of this newsletter are available on the web at
<http://www.epvtech.com>



1) **Tech Papers** - Measuring IMS Transactions - Part 3

After almost half a century, IMS (Information Management System) is still alive and in good health. According to IBM, more than 75% of the world's Top 25 companies listed in the Fortune 100 entrust IMS to run their business.

So being able to measure and tune IMS applications is still an essential skill for every z/OS performance analyst.

From the measurement point of view, IMS is a kind of exception in the z/OS world; unlike all the other subsystems, IMS doesn't write any SMF records. All the relevant events are mapped to a specific log record number and written to the IMS log.

For many years IMS didn't provide a specific IMS log record, collecting all the performance related information. This information was spread across many different log records (e.g. x'07' for program termination and x'08' for program schedule) which had to be combined, by using a fairly complex algorithm, to analyse application throughput, response time and resource utilisation.

This has been one of the reasons for the success of the BMC Mainview for IMS (a.k.a. IMF) product. From the beginning Mainview for IMS has written its own IMS log records which are designed to analyse application

performance. The most important of them is the "Transaction log record", identified by the log record number x'FA' (FA in the following), which provides all the details you could wish, about each IMS transaction execution, in a simple and straightforward way.

From IMS V10, IBM finally decided to provide an IMS log record specifically designed to collect performance information: the x'56FA' log record (56FA in the following).

In this presentation we will discuss what you have to do in order to produce, manage and analyse both 56FA and FA records.

We will also present a case study where we will compare the values of the most important performance metrics collected in these records.

Finally we will briefly discuss the new metrics available to analyse zAAP/zIIP usage of IMS applications.

If you want to receive the paper you can reply to this e-mail writing "**Measuring IMS Transactions - Part 3**" in the subject



2) Tech News - GSE/UKCMG zCapacity Management and zPerformance Analysis Working Group

First group meeting will be held on Wednesday 9th April at IBM, Bedfont Lakes.

Provisional Agenda includes the following presentations:

- **zIIP Capacity Planning** - *Martin Packer, Principal Systems Investigator, IBM*

zIIP Capacity Planning tends to be neglected - in favour of General-Purpose Engines (GCPs). With recent enhancements to DB2 allowing you to offload critical CPU to zIIPs, and to get the most out of zAAP-on-zIIP, it's time to take zIIP Capacity Planning seriously.

This presentation describes how to do zIIP Capacity Planning properly - with instrumentation and guidelines.

- **Parallel Batch Performance Considerations** - *Martin Packer, Principal Systems Investigator, IBM*

With the laws of physics providing a nice brick wall that chip builders are heading towards for processor clock speed, we are heading into the territory where simply buying a new machine won't necessarily make your batch go faster. So if you can't go short, go wide!

This session looks at some of the performance issues and techniques of splitting your batch jobs into parallel streams to do more at once.

- **CPU Overhead Necessary Evil or Mistake?** - *Danilo Gipponi - EPV Technologies & Fabio Massimo Ottaviani - EPV Technologies*

If you are a z/OS Performance Analyst you know that the most important overhead to worry about is the CPU overhead. The reason is very simple: z/OS hardware and software costs are largely influenced by the CPU usage. CPU overhead is a necessary evil: it can't be eliminated completely. However you have to fight, every day, to keep it as low as possible.

In this presentation, illustrated using the EPV tools on-site, we will discuss the most frequent culprits for CPU overheads that you can find in a z/OS environment, providing suggestions about understanding when they are excessive and how to reduce them.

• **New Metrics to Tune z/OS Applications** - Danilo Gipponi – EPV Technologies & Fabio Massimo Ottaviani – EPV Technologies

Performance analysts know that having complete and comprehensive statistical measurements is an essential pre-requisite to being able to tune systems, subsystems and applications. This is why every z/OS site regularly collects millions of records for statistics (e.g. SMF, IMS).

Unfortunately this information is not always complete and is sometimes even misleading.

The three major impediments are:

- the need to keep data collection overheads as low as possible;
- delays in IBM subsystems in updating the available metrics to cater for new technologies;
- the requirement for customers to make system changes to implement the new metrics.

During this presentation we will discuss the most crucial new performance metrics which can be used to tune z/OS applications.

The presentation will be split in different sections dedicated to the following workloads: CICS; IMS; DB2; Batch jobs; WebSphere.

More details at: www.zcmpa.gse.org.uk

Copyright © 2014 EPV Technologies s.r.l., All rights reserved.

If you've received this mail by mistake, or you don't want to receive any more such messages, please send a mail to epv.info@epvtech.com with a subject "REMOVE". You'll be promptly removed from the list. If you want to subscribe to this list you can do that simply send a mail to epv.info@epvtech.com with a subject "SUBSCRIBE".

Our mailing address is:
EPV Technologies s.r.l.
Viale Angelico, 54
Roma, RM 00195
Italy

Add us to your address book

Follow us on



MailChimp

[unsubscribe from this list](#) | [update subscription preferences](#)