



Looking out for the Application in Application Performance Management

Some notes and observations on the role and financial benefits of application performance tuning and the problems of integrating the discipline into a modern IT set-up

Contents

1	Introduction and overview	3
2	How to take advantage of VWLC to save money.....	4
3	How to perform application tuning.....	5
4	Challenges for application performance tuning.....	7
4.1	Where does application performance management fit into the IT organization?	7
4.2	Who are the application developers?.....	8
4.3	'Orphan' applications	8
5	Some ideas	9
	Macro 4's portfolio of application tuning products.....	10
	About the author	11

No part of this publication may be reproduced, stored in a retrieval system or transmitted, in any form or by any means, electronic, mechanical, recording or otherwise, without prior written permission from Macro 4 Limited. The information in this publication is furnished for informational use only, and should not be construed as a commitment by Macro 4 Limited. Macro 4 Limited reserves the right to update or modify the contents. Macro 4 Limited assumes no responsibility or liability for any errors or inaccuracies that may appear in this publication.

1 Introduction and overview

Application performance management, or APM is (according to Wikipedia):

the discipline within systems management that focuses on monitoring and managing the performance and availability of software applications. APM is the translation of IT metrics into business meaning (value). This discipline looks at workflow related IT tools deployed to detect, diagnose, and report on application performance issues to ensure that it meets or exceeds the expectations of end-users and the business.

The principle framework document relating to 'systems management' today is of course the Information Technology Infrastructure Library (ITIL). However, if you look into the ITIL documentation as it relates to APM, you will find that it talks more about service levels - setting them, measuring and reporting against them, and so on - rather than about measuring and improving application performance itself.

In fact the whole of ITIL is rather silent with regard to any issues concerning the applications themselves. Wherever and whenever applications are mentioned ITIL seeks to reduce them as quickly as possible to a list of business 'function points' to be measured and reported against. These ITIL can then manage without having to get into any detail about the real 'application side' of things.

The problem is that detailed application considerations don't fit in well with the business of running an IT operation on a day to day basis.

However, there has been a development in recent years in the business of running mainframe computer systems, which means that we should all pay far more attention to the way our business applications perform (and this goes beyond simply checking whether they meet service levels or not).

This development is Variable Workload License Charging (VWLC), or more simply the idea that you now pay for your mainframe processing power on a per use basis, like having a meter fitted to your mainframe to measure CPU and MIPS usage. If you use less capacity, by running your applications more efficiently, then you can save money, potentially lots of it. On the other hand, ignore application performance issues and you will pay more than you need to.

In summary that's what this document talks about. It shows that application performance tuning is not a difficult process to perform and that the financial benefits can be significant. It also looks at some of the managerial, organizational and other issues that may get in the way of making it happen for real in any particular company; issues that need to be dealt with, if the financial savings are to be made a reality.

2 How to take advantage of VWLC to save money

In the past you would have bought a mainframe processor and paid license fees based on the overall processing capabilities of the box. The cost for software usage wouldn't generally vary until the mainframe hardware was upgraded, at which point you would have to pay the software license fees for the new processor.

A number of years ago IBM introduced the concept of Variable Workload License Charging, along with the first scheme for paying for software on a mainframe system according to how much you use it. There have since then been a number of variations to the original scheme, but basically you will pay based on a peak four hour rolling average within each month that the system is in use.

Today, virtually everyone is paying for their mainframe system usage using some variation of the VWLC model. If you look at things from a cynical point of view, then it's a way for software vendors to charge you more as you make fuller use of the hardware that you have bought. If you take a more positive and optimistic outlook on things, then it is an opportunity to save money on your running costs, month to month, if you can make your applications run more efficiently and use less CPU processing resources and therefore fewer MIPS overall.

That's where the role of application performance tuning comes into the equation. While it's true that some savings on MIPS usage can be made by optimizing the operation of the system as a whole, it is only by looking at the performance of the applications running on the systems we all operate that we can achieve the substantial savings which are possible.

We should not forget that the running of applications is the reason our data processing infrastructure exists. It is the applications which are the major users of resources within our data centers and if we want to look at reducing our overall CPU or MIPS usage, then we can only do that through better understanding and tuning of our applications.

So application tuning is the way to exploit VWLC to your advantage; or more specifically to develop a better understanding of what happens as your applications run, how and where they use resources and how you can target your tuning efforts so that they use fewer resources.

3 How to perform application tuning

The process required to perform application tuning is not complicated, yet few sites get around to doing it on a regular basis, it tends to be 'reserved' for dealing with immediate and critical performance problems.

A proactive application tuning exercise should start with some high level analysis of where the processing power is being used within the computer systems; by which applications or application groups. This can be done by analysis of the SMF and RMF records that the system logs as it runs. There are a number of ways of performing this analysis, but Macro 4's ExpeTune product is ideal for this purpose.

Having identified the applications that are currently using the most resources, then these are obvious first targets for further investigation in your attempts to find resource usage savings.

Next you have to turn your attention to the applications themselves and their pattern of CPU usage as they operate. We'll see later that this can be difficult to accept as an operational role, but for now we will continue with the description of how to do it.

To 'profile' applications, to see where and in which code they spend their time and use resources as they execute, requires the right sort of performance tool. Most sites have several monitors that look at system performance and the internal performance of some major pieces of software infrastructure (z/OS, CICS, DB2, etc.); what these products do not do, though, is present an application centric view.

There are far fewer performance tools that look at things from the point of view of the application as it runs, giving details of the resources it requires to run and how well these are delivered by the system. This group of performance analysis tools, of which Macro 4's FreezeFrame is an excellent example, provides all the data you need to target tuning efforts to both improve the performance and reduce the running costs of your applications at the same time. A rare win, win situation.

It sounds very simple, but that is basically all there is to it. With the right tools and a top down, focused approach, it is not difficult to find major savings, especially if you have not undertaken this kind of exercise before.

One final point about application tuning that relates to DB2: when working with mainframe applications it is not going to be long before you come across applications that work using DB2 - either batch jobs or online processing.

It is a well known fact, or more specifically it becomes apparent through experience of application tuning, that where an application makes use of DB2, the application processing is entirely defined by the SQL calls that it makes to DB2 and the CPU resources that those calls use.

When DB2 based applications are profiled using an APM tool such as FreezeFrame, then it is not unusual to see negligible amounts of time and resources used in performing the application logic compared to that spent in servicing the DB2 calls.

As a result, the tuning of applications that use DB2 often becomes no more than a matter of tuning the SQL calls that they issue. The performance of SQL is determined by the access paths that the DB2 Optimiser chooses to resolve each SQL call; there can be vast differences in the resources used to satisfy a SQL call between having the optimum paths set up and other options.

Profiling applications using a product like FreezeFrame will point to the SQL that needs your attention. You then have to look at the access path that has been chosen, why it has been chosen and what other access paths may be possible to improve performance and reduce CPU usage.

To support staff in performing this process, there is a third product in the Macro 4 portfolio of application tuning products, ExpeTune/DB, that brings all the information and functions required together in a single tool which specifically targets this area.

Identifying and optimizing badly performing SQL is the single most profitable area to work in within the area of application performance tuning. Savings in excess of 90% on individual processes are not uncommon, especially when working on application systems that have not been subject to any previous application performance review.

4 Challenges for application performance tuning

Everything looks so promising, and relatively easy with the right tools for the job; so you might ask 'What is the problem? Why isn't everyone doing this and saving lots of money?'

The problem basically comes down to this question:

'Whose responsibility is performance management... and where does it fit into the IT organization?'

4.1 Where does application performance management fit into the IT organization?

The modern IT production operation runs like a well-oiled machine; everyone knows what they are doing, what they are responsible for and (for better or for worse) what they are not responsible for. Teams in IT, driven by the ITIL framework, seem to know intuitively when something falls outside their immediate area of responsibility, and because of their production commitments are reluctant to get involved with anything like that; hence the difficulty of finding a home for application performance tuning.

ITIL has disciplines concerned with cost, capacity planning, service levels etc. along with an underlying principle of 'continuous improvement'; but as none of these areas directly reference application performance tuning, it tends not to get done despite the enormous potential for savings and service level improvements.

It is not unusual for 'performance' only to become a real concern when service levels are clearly not being met and the business impacted. Then questions are asked, such as, 'What is going on?' and 'When is it going to be fixed?'

A crisis meeting takes place and often every manager is able to show that there is nothing wrong with the individual part of the system or infrastructure that they are responsible for. Of course nobody is looking at things from the application point of view, and the application is still 'not working'. You may recall an advertisement on television sometime ago from one of the world's biggest IT companies showing this 'approach' to problem solving!

So when it is concluded that it is not a system problem and there is nothing more that 'production' can do, an attempt will be made to pass the buck over to 'development', which leads neatly onto the next section of this document.

4.2 Who are the application developers?

So now the question becomes: who are the application people as far as your production applications are concerned? The obvious answer is application developers, but the truth is that application developers are often only available to look after systems that have been recently developed or are being actively maintained. For other systems you are not likely to find 'developers' in the normal sense of the word.

Developers and development departments work on a different basis to the operational side of things. Operations is an example of an ongoing 'production' process (hence why it is often called production) and the staff and responsibilities are organized to support that process.

Development however works on a project basis, where teams are built for a development project, then dismantled and rebuilt as the next development project comes along. This is a rather different way of operating and means that over any length of time the developers that worked on any particular application system will have been redeployed, moved on, left the company and so on. What is more, if the application was brought in as a package, or belongs to a company acquired as part of a merger or takeover, then you may not have access to the original developers at all.

So do application developers have a role to play in application performance tuning? Ideally yes, but for all sorts of practical reasons it's likely that for existing production applications their responsibilities are going to begin and end with changing code to fix performance problems identified by other teams.

They are just not going to be available to do the job of looking into performance problems, and if any problems are handed to them by production on the basis of just 'there is something wrong with your application', then not much progress should be expected on that front either.

4.3 'Orphan' applications

It's a strange idea to consider, when computer applications are at the core of everything that a modern day company does, but where an application has existed for many years (often 20 or more for mainframe applications) and is not actively being worked on, but just keeps on doing what it does day after day, it may be considered to have reached a type of 'orphaned' status within the company.

No one really cares about these orphaned applications within a company, at least until something goes wrong and performance levels drop. The production department runs applications, measures service levels, etc. but operations staff may not feel qualified to take their application involvement further than that; and the developers who maintained all of the application knowledge have probably moved on.

5 Some ideas

There's no easy answer to many of these organizational and managerial problems, so this is an ideas section, rather than a solutions section.

If you are convinced that application performance tuning is something that you should be doing, not just because this document says so, but because you can see the potential savings that can be made, as well as the improvements to service that are possible, then what can you do to make it a reality on your site?

The first thing that you need to do is make sure that you have the right tools to do the job. Macro 4 has a suite of such tools, consisting of ExpeTune, FreezeFrame and ExpeTune/DB.

Then you need to find some resource and people time to perform the first stage of the process which is analyzing and profiling your application workload. It's easy to think that this is not an operational responsibility because it relates to 'the applications', but in practical terms it's only operational staff that can do this.

There are a number of areas in operations, such as capacity planning and service level management that work in areas that are closely related to performance, so one option would be to extend their responsibilities to take on this role. It's important to remember at this stage that the aim is only to identify tuning opportunities, not to make any changes to the applications as of yet.

Often a catalyst is required to get things going. Some organizations have set up one off projects that cut across ordinary department responsibilities to get application performance tuning going and benefit from the savings that can be realized. Others choose to use external consultants, who as well as providing the catalyst to move things forward have the expertise to do the job. Again this is an area where Macro 4 can help and advise you.

If there is one message that is important to remember, it is that application performance tuning - and its benefits - will only be achieved through interpersonal and interdepartmental cooperation, and top level commitment and sponsorship of it within your company.

Macro 4's portfolio of application tuning products

Macro 4 has a portfolio of products that cover all the areas and stages in application performance tuning. The three mentioned in this white paper are:

ExpeTune

ExpeTune is a tool that performs a high level analysis of where CPU computing resources are being used throughout your mainframe systems, so that you can then focus more detailed analysis on the highest use application workload areas.

ExpeTune works by analysis of SMF and RMF record data recorded by systems as they run. It has various PC based facilities for displaying the results of its analysis, can produce 'Watch Lists' of high resource utilizing jobs, and will interface automatically with FreezeFrame to drive the next step in the application performance analysis process.

FreezeFrame

The key product in Macro 4's application performance management portfolio, FreezeFrame provides facilities to profile application processes as they run with your system. It shows clearly and precisely where the application's execution resources are being used, providing clear targets for your application tuning efforts.

FreezeFrame can be used in both batch and online environments: before (to target) and after (to confirm) tuning efforts; to improve service levels and realize those all important financial savings.

ExpeTune/DB

Most modern computer systems on the mainframe use the DB2 relational database management system. When DB2 is used, it is the SQL calls that the application programs make that are usually the major CPU resource users; and represent the targets for tuning efforts and cost savings.

Profiling, with FreezeFrame, will identify the SQL calls that use the most resources; but special analysis of the access paths chosen by DB2 to resolve each call are necessary to understand 'what is going on' and what scope there is for performance improvements.

Macro 4's ExpeTune/DB product brings all the tools and data required to perform this analysis together in one place for those working on SQL performance tuning to use.

About the author

Phil Mann is a veteran in the IBM mainframe arena, with over 30 years' experience, including ten years with Macro 4. He has worked as both a technician and consultant, mostly on the production and operational side of running things in a mainframe environment. This has included periods in capacity planning and performance tuning. Currently he works as a consultant across all Macro 4's mainframe product portfolios, but with a special interest in the area of application performance optimization and Macro 4's products that support this area. Phil is convinced that the way to get the best out of your mainframe, and reduce mainframe operating costs, is to continue the performance tuning process beyond the computer infrastructure itself and focus more on the performance of the applications that run there. In Phil's experience, focusing on the applications that are the biggest users of computing resources not only helps organizations to deliver the high standard of service that users expect, but also delivers real savings in running costs. A rare 'win, win' situation, not to be ignored in the current commercial climate.

**Macro 4 Limited**

The Orangery
Turners Hill Road
Worth, Crawley
West Sussex
RH10 4SS
United Kingdom

Tel: +44 (0) 1293 872000
Fax: +44 (0) 1293 872001
Email: market@macro4.com

www.macro4.com

Belgium

Tel: +32 (0) 15 74 74 80
Fax: +32 (0) 15 74 74 89
Email: market.be@macro4.com

France

Tel: +33 (0) 1 41 03 32 00
Fax: +33 (0) 1 41 03 32 01
Email: market.fr@macro4.com

Germany

Tel: +49 (0) 89 610 0970
Fax: +49 (0) 89 610 09777
Email: market.de@macro4.com

Italy

Tel: +39 (0) 2 213 1941
Fax: +39 (0) 2 213 9611
Email: market.it@macro4.com

Spain

Tel: +34 91 443 0220
Fax: +34 91 457 8457
Email: market.es@macro4.com

Switzerland

Tel: +41 (0) 44 723 40 00
Fax: +41 (0) 44 723 40 02
Email: market.ch@macro4.com

USA

Tel: +1 973 526 3900
Fax: +1 973 526 3899
Email: market.usa@macro4.com

About Macro 4

Macro 4 is a global software and services company that helps organizations to improve the performance of their business-critical applications and processes. The company's solutions for document and data management, fault analysis and performance management are easy to use, fast to implement and deliver value quickly in even the most complex IT environments. Macro 4 is a division of UNICOM Global, which operates in every major market worldwide via US and European subsidiaries and a network of international business partners.

For more information on Macro 4 products and services visit www.macro4.com.

© Copyright 1990-2012 All Rights Reserved. Macro 4 Limited – a division of UNICOM Global.

UNICOM® Systems, Inc.

UNICOM Plaza Suite 310, 15535 San Fernando Mission Blvd., Mission Hills, CA. 91345 USA

Tel: +1 818 838 0606 Fax: +1 818 838 0776 www.unicomglobal.com

