Mainframe Performance Tuning: Fear and Loathing vs. Substantial Benefits — Part I

Application performance tuning, which serves to audit the mainframe environment, has become an increasingly important tool in the effort to produce stable systems and to ensure the best use of resources.

In today’s rapidly changing data processing environment, applications must be developed and modified in record time. It has become increasingly difficult to maintain stable and efficient mainframe systems. Although each new hardware generation offers reduced per-MIP costs, applications consume resources faster than we can add the requisite iron. To extend the challenge, most corporate data will soon be accessible online.

Worse, workloads vary more in resource use and transaction arrival rates, not only day-to-day but also hour-to-hour. New systems and new connections to older systems link corporate data to a larger population of users, from the traditional internal user to a broad Internet audience.

Application performance tuning, which audits mainframe efficiency, has become increasingly important to the effort to produce stable systems and ensure the best use of resources. Attention to application performance can produce substantial cost reductions by delaying hardware upgrades. Performance tuning also attacks less-quantifiable problems such as poor service and untimely startup of online networks due to late batch processing.

This article, the first in a two-part series, examines why performance auditing and tuning have become more critical as hardware prices have decreased. To start, let’s look at the following questions:

- Do companies really need mainframe performance auditing and tuning?
- If they do, why is performance management often a low priority?

The concluding article will then ask the following questions:

- How can performance contribute to good service levels, better planning, and increased savings?
- What techniques and tools can help companies stay on track?

WHY DO COMPANIES NEED MAINFRAME PERFORMANCE AUDITING AND TUNING?

Cheryl Watson’s Tuning Letter highlights both the importance and complexity of performance tuning. Watson’s summary, a SHARE Conference keynote speech, which appeared in the Tuning Letter in 1999 (“Focus: Why Tune?”), provides clear evidence of performance’s importance. In a number of examples, the savings were significant enough to capture the hearts of CEOs everywhere. Comments ranged from “Saved one shift and delayed a CPU upgrade for three to four years” to “... Returned enough capacity to defer a processor upgrade by eight months” and “Net Savings: four hours of batch cycle ...” to “Net savings: Delay of CPU upgrade for six months.”

That is the good news. There is unmined gold in system upgrades. Upgrading is expensive. A 25 to 50 percent upgrade in CPU power, including MIP-based software prices, can add from $500,000 to well over $1 million (in real and immediate dollars) to IT costs. Worse, the higher costs become permanent because of increased future maintenance charges.
Out-of-pocket upgrade costs are not the only concern. There are also the opportunity costs associated with the effect of sub-optimal performance on highly competitive e-business. As desktop processing speeds rise to 1.5GHz and high-speed networks become common, web response bottlenecks migrate to the mainframe site that owns the underlying data. In the past, companies could deal with poor data availability by notifying everyone that only day-old data were available. Unfortunately, this is not acceptable on the web.

As businesses become increasingly interlocked, another loss exposure emerges — the cost of legal action brought by a partner because of poor service. These losses may be much greater than upgrading a processor or implementing solid performance tuning.

Application tuning doesn’t seem to affect real expense, so businesses overlook the opportunity costs of poor performance. For example, businesses tend to ignore overnight batch performance because CPU is below 100 percent busy and run times appear acceptable. Planning is based on daytime peak requirements. Saving CPU cycles at night doesn’t save real money. Often, batch jobs run long because of I/O and not CPU at all.

This docile batch window can shatter and become costly for many reasons, including the following:

- Suddenly heavy overnight volumes or special processing such as large database maintenance jobs can result in missed deadlines.
- Production jobs run beyond the normal schedule, perhaps into scheduled database image copies. Backup jobs then run into daytime shifts and wreak havoc with online data availability.
- Production batch jobs run during daytime peaks, competing for CPU.

In a recent initial performance assessment (IPA) that CPT Software performed for a large insurance company, the estimated savings were 3,528 CPU hours a year, or $700,000 to $2.4 million (depending on the CPU charge-rate). There was also an elapsed time savings of 5,893 hours a year. This came from part of the low-hanging fruit. Greater savings are projected when the whole site is included.

E-BUSINESS AND THE MAINFRAME

In the last decade, new application architectures have appeared. Migrating to these platforms seemed the direction of the future. Reports of the mainframe’s imminent demise abounded. Like Samuel L. Clemens, the mainframe has continued undeterred. Indeed, many applications and their data seem to be moving back to their original mainframe home. Companies are now implementing large web applications on mainframe hardware.

Two factors determine whether a business can accommodate shifts in workloads: a measure of extra capacity and a rigorous, continuous performance tuning effort.

Regardless of the type of business, the web-enabled population can choose between many competing companies. The GVU Center at Georgia Institute of Technology publishes Internet user surveys twice a year (www.cc.gatech.edu/surveys). In the October 1998 survey, 18 percent of those surveyed cited slow download speeds as a reason for dissatisfaction with the web.

As web design gets better, other reasons (confusing site, poor page design, couldn’t find what was wanted, etc.) will become less significant. Poor site performance (read response) may well emerge as the major reason for selecting one company over another. Whether the web application is on the mainframe or only its data, the performance implications are staggering.

CAPACITY PLANNING ON THE MAINFRAME

It is impossible to produce a meaningful capacity plan for an untuned system. The usefulness of the capacity plan is inversely proportional to the randomness of the system for which the plan is written. Often, when the plan’s predictions are compared with the eventual actual workloads, great variations appear. Among the contributing factors are the following:

- New and ever-changing technologies are in play. Technological changes used to come more slowly. A two- to five-year planning horizon made excellent sense. No longer. The need for better and faster direct online access to data makes it imperative that companies use new technology or risk being left in the dust by their competitors.
- Business is much better than expected. “If I am doing so well, why do I feel so bad?” As many startup dot-coms are discovering, if your infrastructure is unstructured, there is no simple way to handle any realized volume of business.
- Data flows and work environments change. The well-established peaks from 10 a.m. to noon and from 2 p.m. to 4 p.m. have suddenly shifted. This shift can be intermittent (i.e., when online applications occasionally come up late) or permanent.
- Audience participation shifts. As operations (particularly online operations) run 24x7, influenced by e-business and the web, support and performance issues take on a new dimension.

Two factors determine whether a business can accommodate shifts in workloads: a measure of extra capacity and a rigorous, continuous performance tuning effort. The capacity margin is a result of the capacity plan. One of the few ways to improve the system stability on which capacity planning depends is a well-organized program of performance auditing and tuning.

Performance tuning that reclaims CPU power and lowers I/O times produces a system better able to sustain those sudden higher activity rates. This is partly because additional work is less likely to produce unintended side effects. It is also because the available spare capacity is more accurately sized because the capacity predictions are more accurate.

ESTABLISHING PRIORITIES

The return on performance tuning is substantial, so why are performance and tuning such a low priority, and why is
there reluctance to take action? If performance tuning is so necessary, and we would argue that it is, why doesn’t senior corporate management implement a rigorous program to manage application performance? An answer is beginning to emerge. Part of this answer is a fundamental misunderstanding about the difference between tuning savings and tuning cost. Measuring the “cost” of unstable online response times or excessive batch elapsed time is difficult. In addition, there are the following classic IT management principles:

- “Late” is not an acceptable option.
- If a problem appears during development and it is not critical (and what performance issue outweighs the need to be “first to market?”), we will fix it later (see Rule 1).
- We will document it later (see Rule 1).
- We will worry about the implications of success when we are successful, i.e. later (see Rule 1).

These rules make perfect business sense. Management is under intense pressure to produce short-term results. We cannot afford what does not help next quarter’s earnings. As an executive once told me about business continuity planning, “We are here to enhance shareholder value. If a disaster occurs, we can always improvise a solution after the fact and at no greater cost than if we planned. Therefore, to spend the shareholders’ resources on planning is to waste them and is a violation of our charter.”

A better way to implement performance improvements is to address them during testing. I have even heard the radical suggestion that performance should be a system design consideration. If tuning is managed correctly, it should not delay most development projects. In fact, it might even speed them up. To achieve thorough testing, part of the process involves stress testing with massive volumes of data. As the development/test cycle proceeds, regression testing ensures results are not skewed by data changes. If one looked for and fixed performance issues at this stage, each regression pass could be faster.

Attending to performance tuning during development provides a two-fold return: once during development and again in the ultimate production performance.

**MANAGEMENT ISSUES**

Failure to perform adequate performance management adequately is partly due to technical issues. Some issues, however, are managerial. Among these are:

- an inability to attach a real cost to poor response times or late batch
- no formal process to recognize and address performance issues
- a lack of coordination between affected departments
- a shortage of trained staff

**Y2K ISSUES**

Even though January 1, 2000 and January 1, 2001 are now history, the Y2K fallout lingers, affecting budgets as well as personnel workloads. Additionally, this fallout has had other influences on IT performance efforts:

- Applications that required Y2K modifications have raised and left behind a backlog of performance issues.
- Attempts to reduce the large expense of Y2K have resulted in fiscal stringency.
- The contract programmers who did the work are no longer present to make modifications.

**ANALYSIS COMPLEXITIES**

Data used to live in a file, worked by an application. Now they live in databases with interconnected applications, driving a web site, and supporting a data warehouse. Companies use data warehouses to perform both online analytical processing (OLAP) and online transaction processing (OLTP). With so much complexity and so little time, systems and data flow are harder to understand. It is difficult to find someone who knows how it all fits together. Other complicating factors are:

- DB2, ADABAS, DATACOM, Oracle, enterprise resource planning (ERP), and all purchased and inherited systems require different analytical approaches and produce variable (and often incompatible) meta-data and system definitions.
- The “strategic direction” changes (mainframe to midsize to server and back).
- Workloads are too variable to allow much automation.
- The sheer volume of data is overwhelming.

**STAFF ISSUES**

Companies often claim they pay attention to performance and have well-qualified staff working on these issues. It may be true that personnel have been assigned. However, whether staff actually works on performance can be an issue. Performance professionals have other simultaneous assignments such as help desk duties, application support, continual fire fighting, capacity planning, and maintenance and upgrade of their own tools. These responsibilities, while undeniably contributing to IT’s mission, rarely leave time for serious (or any) performance auditing and tuning. Lack of an attempt may actually account for the mistaken belief that there is no significant gain to be had from an attempt. If something else is always a higher priority for a scarce resource (how many really good systems people have you seen on the market lately?), it should come as no surprise when projects reach production.
surprise that performance is sub-optimal. One gets what one pays for.

“WINDOW” PROCESSING

Whatever comes in must be processed — it is IT’s mission. How does one plan for a future requirement while containing costs if one must process whatever comes in? The plan too often amounts to ensuring there is so much extra capacity that nothing can overwhelm it. This solution has a long and honorable history (the technical term is “throwing iron at the problem”). However, this is hardly cost control. Performance professionals must convince management that performance management is a better (and less expensive) way to do business. Someone must take responsibility for the workloads thrown into the processing “window.” More to the point, someone must clean them up.

How can a CEO/CFO/CIO ensure the company’s data processing costs are really under control while still providing for present and future needs? How can companies effectively focus on performance auditing and tuning efforts? What’s the best way to reap the rewards of such efforts? Can we do this more efficiently? Part II will address these issues.

This article series, which is an excerpt from a White Paper, is available in its entirety by faxing (617-254-8979) or emailing your request to info@cptsoftware.com.

Howard Beale has spent nearly 20 years with mainframe computers as a programmer, analyst, and systems programmer. This includes five years in a support role with vendors specializing in performance analysis tools. He was also an EDP Auditor for six years where he divided his time between the efficient and effective use of computing resources and the problems of continuity.