10 Steps to Reducing Mainframe MLC Costs

By David G. Wilson
President and General Manager, SZS Consulting, Ltd
# TABLE OF CONTENTS

IMPORTANCE AND IMPACT OF SOFTWARE LICENSING CHARGES ............................................. 1
  » Chargeback Impacts Attitudes, Behaviour, and Cost Reduction ........................................ 2

MANAGING SOFTWARE CHARGES AND 10 STEPS TO REDUCE THEM ................................. 2
  1. Understand the Cost Base .................................................................................................... 3
  2. Analyze Peak Workloads .................................................................................................... 6
  3. Implement Capping ............................................................................................................ 6
  4. Review Workload Placement ............................................................................................... 7
  5. Leverage Technology – the Benefit of Technical Currency ................................................ 7
  6. Review the SCRT Process ................................................................................................... 8
  7. Manage the Entire Software Portfolio ............................................................................... 8
  8. Evaluate Possibilities to Qualify for Special-Situation Terms ........................................... 8
  9. Investigate Alternative Licensing Metrics ......................................................................... 8
 10. Plan Procurement Negotiation Strategy ........................................................................... 8

CONCLUSION ......................................................................................................................... 9
IMPORTANCE AND IMPACT OF SOFTWARE LICENSING CHARGES

Most software vendors and IT service providers conduct regular customer studies to evaluate market dynamics and customer priorities. This is either done as a general evaluation, such as IBM’s yearly CIO study, or for specific market segments, such as BMC’s Annual Mainframe Survey. These evaluations identify that CIO priorities are to:

- Develop and implement technology innovations that capture and leverage market trends;
- Meet service level agreements required by lines of business (LOB);
- Reduce the unit costs of providing IT services.

Over the last few years, almost everyone with mainframes in their environment cited overall cost of mainframe computing as one of the top challenges. Drilling down further reveals that software charges are likely to be the chief concern, and within that the MLC (IBM’s Monthly License Charges) are stated as the #1 issue.

There are two basic forms of software licensing for the mainframe:

1. Perpetual Licensing (sometimes referred to as OTC): a one-time fee for the right to use the software in perpetuity, plus some form of on-going maintenance and support.
2. Term Licensing: a rate for a specific unit of time or term, which may range from days to years.

In both cases, you have to pay for a quantity that might equate to the size of the machine on which it runs or, as is increasingly the case, some form of sub-capacity measure.

MLC can be considered a term license where the time period is a month. For most MLC software licensing, the quantity is defined as the peak rolling 4-hour average (R4HA) of all the logical partitions (LPARs) where a product runs. The compliance mechanism to track and report this is an IBM report called the Sub-Capacity Reporting Tool (SCRT), which has to be printed and submitted to IBM each month.

There are several reasons why MLC software receives so much attention:

- Rule of Big Numbers
  MLC tends to be the most prominent item on large corporations’ IT expense budgets, sometimes representing as much as 40% of the mainframe cost. It requires regular payment; thus, within the monthly reporting, there’s a constant reminder. Typically, there is insufficient detail on the price points or calculations, making it difficult to determine how the total price is calculated. It displays as one, large item that commands attention and is challenging to justify.

- Lack of Choice
  Although IBM has made other licensing formats available and even converted some of the traditional MLC software products to OTC products, MLC remains as the only licensing option for the majority of IBM software needed to support existing applications. Further, for most of the key products, there are no alternatives available from other vendors and there is little or no deviation from the list price (quantity discounts being inherent in the price structure).

- Lack of Transparency
  In addition to the lack of pricing details, it is unclear what value MLC truly imparts. It can be evaluated by business application or per transaction, for example, but few LOB departments can access the real cost of a business application or of a typical or average transaction. This is mainly the result of inaccurate charge-back practices used by IT departments to re-allocate IT costs to the various LOBs.

- Lack of Predictability
  Finally, there is uncertainty over future costs. With the Mainframe Charter, IBM promised in 2003 to deliver significant price performance improvements (PPI) to their mainframe customers year over year. What that really meant was that if a certain workload was growing for a customer with an optimized mainframe environment
that maintained technology currency), the associated unit cost of that workload would decrease over time. This was ensured through the hardware technical dividend whereby software was cheaper on newer hardware, but customers mostly achieved savings through machine consolidation.

Recently, IBM has started to raise MLC prices. Since machines are now much larger than most users require, the opportunity for consolidation has been realized and real PPI has become harder to achieve unless a user is growing quickly. MLC cap setting in IBM’s enterprise software contracts tends to be an art, not a science, with overages on those caps requiring additional budget approvals. The result: more unpleasant justifications internally.

The overages are often caused by price increases for the operating system or the various middleware products (surprisingly, there is no MLC price increase protection even within running contracts). For example, with the price announcements since 2011, Customer Information Control System (CICS) prices increased by as much as 24 percent over an 18-month period for large users. Further, a series of price actions for DB2 and Information Management System (IMS)—the latest at the end of August 2013—have resulted in similar increases that now outweigh the five percent (or less) technical dividend earned with new hardware. Naturally, this impacts not only the IT departments but also the LOBs, and sometimes (depending on the charge-back practices applied by the IT department) increases costs for those whose applications and databases do not even use the products directly impacted by the price increases.

CHARGEBACK IMPACTS ATTITUDES, BEHAVIOUR, AND COST REDUCTION

Most organizations use an average cost per MIPS as the basis for cost recovery. While all of the actions suggested in this paper will help reduce mainframe costs at the enterprise level, there will be little appetite for change if the chargeback or internal accounting does not flow through to the LOBs. An issue surfaces when the chargeback mechanism is allowed to influence the business technology decision regarding where an application resides or when it runs.

The incremental software cost of capacity on a mainframe is always less than the average cost. Due to the complexities of the mainframe charging mechanisms combined with the virtualization characteristics, applications that do not run in peak time actually incur almost zero cost. Applications that do run in peak and have a rich software stack will cost more than the average cost per MIPS. Even if an application runs in the peak period, it can be argued that unless it is an application that is mandating the mainframe existence, the chargeback rate should be based on the incremental cost of supporting that work—not on the total average cost of the mainframe. For the rest of this paper, we will focus on the ten steps that we believe to be the best practices to reduce costs.

MANAGING SOFTWARE CHARGES AND 10 STEPS TO REDUCE THEM

It’s not all bad news. The benefit of a monthly charge system is that if you can reduce the quantity used, then there is a relatively short timeframe before this leads to a reduction in the cost. Aside from minor tuning, the most common approach to dramatically reduce mainframe MLC costs is to re-platform applications. Yet, this is also the most risky method and in fact rarely saves costs at the organizational level. In fact, this is a futile exercise, because much of the cost is already incurred and decremental costs (like incremental) are far lower than average costs. Ironically, re-platforming actions usually lead to higher average costs for the remaining mainframe applications.

Based on our experience in the last 15 years, there is a more effective way to reduce on-going software costs by as much as 30 percent. It requires a systematic, project-based approach, which crosses departmental boundaries, combined with a determination to really dig into the details.

With that as a backdrop, one would imagine that every mainframe installation would have a team working on cost reduction and optimization. But despite all the discussion about cost and the individual silos focused on it, not many organizations are actively managing it effectively. Why is this?
1. Very few people really understand how costs are derived in the mainframe environment. MLC is the only component with a realistic list price but with over 50 different licensing metrics, most users delegate calculating the price to IBM. With little understanding of the underlying structure (other than more MIPS = more cost), what hope is there to manage the cost down?

2. Due to differing organizational responsibilities, there is a divide between procurement and operations. The first priority for IT operations is service availability. Most IT operations departments understand that there is a direct relationship between capacity and cost, but they are still mostly focused on total systems (hardware) capacity. Procurement has to buy what IT tells them they need, and IBM doesn’t openly discount MLC. Thus, frustration and a sense of futility set in.

We have identified a set of ten actions that typically result in a reduction of 10–30 percent in MLC and other mainframe costs. Broadly, these actions fit into three main categories:

1. Analysis - Understanding where you are from a cost standpoint (steps 1-2)
2. Actions - Reducing the billable workload by identifying and executing appropriate actions and processes or selecting more attractive licensing metrics (steps 3-7)
3. Negotiations - Negotiating special terms with IBM (or other vendors) (steps 8-10)

The key to all of the above is having the information and knowledge on which to base business, technology, and negotiation strategies. We have concluded that the lack of tooling and standardized practices in this area prevents most organizations from making significant progress.

1. UNDERSTAND THE COST BASE

If you want to manage something, then you need to understand what influences it. However, most organizations regard MLC charges as an art, not a science, and they lack the ability to dynamically calculate MLC cost themselves.

<table>
<thead>
<tr>
<th>Product Number</th>
<th>Product Name</th>
<th>Price</th>
</tr>
</thead>
<tbody>
<tr>
<td>S605DB2</td>
<td>DB2 10 for z/OS</td>
<td>45716.81</td>
</tr>
<tr>
<td>S605DB2</td>
<td>QMF Classic Edition</td>
<td>14215.18</td>
</tr>
<tr>
<td>S635A02</td>
<td>IMS Database Manager</td>
<td>29155.20</td>
</tr>
<tr>
<td>S635A02</td>
<td>IMS Extended Term Opt</td>
<td>1901.41</td>
</tr>
<tr>
<td>S635A02</td>
<td>IMS Transaction Manager</td>
<td>34224.34</td>
</tr>
<tr>
<td>S665G53</td>
<td>COBOL Full Function</td>
<td>3094.20</td>
</tr>
<tr>
<td>S665Y04</td>
<td>CICS TS for z/OS</td>
<td>39341.23</td>
</tr>
<tr>
<td>S694A01</td>
<td>z/OS V1 Base</td>
<td>86400.76</td>
</tr>
<tr>
<td>S694A01</td>
<td>z/OS V1 C/C++ with Debug</td>
<td>5452.10</td>
</tr>
<tr>
<td>S694A01</td>
<td>z/OS V1 DFSMS dss</td>
<td>1591.99</td>
</tr>
<tr>
<td>S694A01</td>
<td>z/OS V1 DFSMS dshsm</td>
<td>6771.91</td>
</tr>
<tr>
<td>S694A01</td>
<td>z/OS V1 DFSMS rmm</td>
<td>3493.39</td>
</tr>
<tr>
<td>S694A01</td>
<td>z/OS V1 HCM</td>
<td>542.00</td>
</tr>
<tr>
<td>S694A01</td>
<td>z/OS V1 RMF</td>
<td>3701.58</td>
</tr>
<tr>
<td>S694A01</td>
<td>z/OS V1 SDSF</td>
<td>1977.81</td>
</tr>
<tr>
<td>S697NV6</td>
<td>NetView for z/OS - TEMA</td>
<td>0.00</td>
</tr>
<tr>
<td><strong>Total Monthly:</strong></td>
<td></td>
<td><strong>277,579.86</strong></td>
</tr>
</tbody>
</table>

The table to the left is a typical example of the pricing information provided by IBM. Aside from stating what each product costs for a given configuration and usage, the reports provide very little in terms of valuable information for controlling IT costs. For example, the unit price is not stated and there is no way to visualize how a change to SCRT reported capacity would be reflected in the invoice that follows six weeks later.

Some organizations have created spreadsheets to calculate the cost, often over-simplifying things and thus not allowing for scenario planning or simulations. Very few have built sophisticated models that allow for multiple price versions to predict the impact of version upgrades or IBM price actions. Even fewer have price models that identify incremental costs at the LPAR level, etc. Finally, almost none of the organizations we have seen calculate the real impact of incremental or decremental workloads.

In our experience, this is fundamental to changing behaviour. As soon as employees realize that a peak million service units (MSU) in an LPAR costs $500 USD, for example, they can make fact-based business decisions on resource actions. Reducing 50 MSUs for a month will lead directly to a saving of $25,000 USD in the bill for that month’s usage. Running a 500 MSU stress test either side of a billing period boundary rather than scheduling them both to occur
in the same billing period, can waste $250,000 USD. In any engagement, the first action we take is to ensure a cost model exists to provide the kind of detailed information required to make cost based resource decisions.

2. ANALYZE PEAK WORKLOADS

Armed with an understanding of what drives unit cost, the next step is to understand what constitutes the peak workload and how to reduce it. Many organizations start with the SCRT report, but the most confusing aspect of the metrics for IBM’s workload license charges (WLC) is that the peak time on which WLC pricing is based is usually different for each product. Although the SCRT report lists the peak hour for each product / LPAR, it does not show what is happening during the other hourly periods. As a result, it is easy (even for the experienced) to draw the wrong conclusions when reviewing the report. The required information is stored in the SMF records but is not printed on the SCRT report. Thus, it has to be extracted using other means and imported to some form of analysis tool.

To quickly see where there is opportunity to reduce peaks, users need a monthly cost-oriented view that can be sliced and diced. This requires extensive analysis of SMF records and knowledge of the underlying cost structure. The end point is a chart that shows the R4HA for a period of at least one month and allows the user to select different LPARs, different product combinations, and workload priority or ’importance’. The chart to the left is an example.

This chart which covers the period of a whole month suggests that the peak on the first reporting day of the month (highlighted with red ellipse) is out of place. It also shows that the next highest peak is much less. It is worth finding out what happened at that time—either to take action to prevent it from happening again or to evaluate whether there is a case to use the SCRT “EXCLUDE” statement. This chart is the R4HA by LPAR; the same format should be used to view by workload “importance” and by MLC product.

We have found that whenever such charts are created, the user invariably starts to drill into the details and review different aspects – it’s rare for an action not to be quickly identified that will reduce either the current or future month’s costs. There are many ways this data can be used; following are three case examples.

Case 1 – Importance

Being able to see the low-priority work that is running at peak times enables us to move some work out of the peak period. In the example to the right, most of the work being executed at the peak time is low priority (Imp5) and we can therefore take action to move it from the peak period. Ways that we can do this include reducing the Defined Capacity (effectively forcing WLM to defer processing) or changing the scheduler.
Case 2 - Low utilization of middleware product

MLC products are charged based on the peak R4HA of the LPARs in which they run, not on the consumption of the licensed product itself. In the example below, we identified that CICS is actually contributing very little to the total machine peak, although it is present in four LPARs. Nevertheless, CICS will be charged at the rate of the LPAR peak.

By isolating the CICS applications to 2 LPARs, the MLC cost for CICS was reduced by 12% and total MLC costs were reduced by 2%.

Case 3– Unusual Peaks

The R4HA by LPAR chart below shows an unusual spike on the fourth Friday of the month, mostly driven by a peak in the ‘Lichfield’ LPAR. The peak for the CPC on this day is about 125 MSUs above the next highest peak day (which occurred the following Tuesday). It is instructive to note that the SCRT report listed the second highest peak as occurring one hour after the peak on the fourth Friday, which is not much help.
By drilling down to review the Lichfield (below), we can see that this LPAR peaked 90 MSUs higher than it did in the following week. Investigation revealed a DB2 issue caused the peak, which justified using the EXCLUDE statement at this time on the SCRT report. The result: a reduction in the monthly bill of 90 MSUs, amounting to $43,000 USD.

3. IMPLEMENT CAPPING

The ability to cap resources is a very powerful feature of Workload Manager (WLM), but if used incorrectly can compromise service levels. Hence, most users may implement capping to some degree, for example setting ‘loose’ caps to stop a loop or runaway job, rather than using it to tightly control the software bill. This is especially true for on-line workloads.

Many users still implement LPAR level defined-caps, albeit some alter the caps according to predefined profiles depending on the shift. LPAR caps (soft or hard), sometimes compromise the ability of WLM and IRD (Intelligent Resource Director) to prioritize work according to importance and maximize processor utilization across the CPC, therefore we encourage Group Soft Capping wherever possible. When group capping is used, we see much higher average processor utilization than when LPAR capping has been implemented because the system allocates resources in a balanced way across multiple LPARs according to the priorities set in WLM. Mathematically, group capping will deliver lower costs than LPAR soft capping, provided the WLM profiles are set correctly.

One of the challenges with capping is in knowing the level at which to set the cap, and ensuring there is a clear mechanism and mandate to increase it if SLAs look like they will be compromised. In addition to the technical considerations, operational and management processes need to be changed to enable caps to be raised very quickly if the need arises. Once a cap has been raised in the month, there is little value in reducing it below the peak R4HA already achieved. It will not alter the current invoice; however it is important to reset it on the first calendar day of the next month to align with the forecast for the next month.

Additionally, the Capacity Provisioning Manager (new with IBM z/OS v2) dynamically increases the caps if certain criteria are met. While it can de-risk many of the issues with setting the cap too low, we still believe that it is better to understand and manage workload demand in advance—rather than managing the consequences of capping after the event. Similar to driving a car for energy efficiency, it is better to anticipate and ease off the accelerator early rather than constantly braking and accelerating. In this context, capping is like applying the brakes.
4. REVIEW WORKLOAD PLACEMENT

WLC charges are based on maximum CPU utilization by LPAR. A product that uses three MSUs in an LPAR that has a peak at 100 MSUs is charged at 100 MSUs. That is why optimizing workload placement is so important. One can distinguish between two different forms of workload placement in a traditional mainframe environment that affect overall MLC charges and the R4HA. Both approaches attempt to balance out the peaks and valleys in MSU usage.

1. Time Placement: This approach is focused on controlling the time when jobs are run, so that only high-priority work is executed during the peak times. It requires moving workload around within an LPAR based on scheduling, thus bringing it forward or delaying its execution to after the peak times. Customers who apply this technique could evaluate whether all month-end jobs have to be run immediately after month end. Sometimes if the last day of the month falls on a Sunday, transactions will not be processed after the end of business on Friday, so month-end processing can start on Saturday. The load may be lighter then, eliminating the need to run all month-end jobs on the Monday during the day when the online demand is highest for most customers. Similarly, some month-end processing could most likely wait a few days for processing. It could be run outside of the peak period when resources become available again, without incurring additional charges. As a result, in the absence of proper planning tools that enable what-if scenario analysis, an organization may depend on home-grown labor-intensive cost tools that make changing workload schedules rather cumbersome.

2. LPARs: This approach centers on deciding which LPARs should host the work that requires specific middleware products, such as CICS, IMS, MQ, DB2, etc. We often find that clients run work in several LPARs because historically they needed to balance demand across several processors due to cycle time. Over the last three generations of mainframes, the cycle time and input/output (I/O) have improved dramatically—enough that consideration can be given to running all jobs that require CICS in fewer LPARs or IMS in other LPARs, for example.

5. LEVERAGE TECHNOLOGY – THE BENEFIT OF TECHNICAL CURRENCY

Since 2003, one way of delivering mainframe PPI and to encourage hardware technology upgrades has been the introduction of a changed and improved basis for MLC charges with every new hardware generation. Arguably IBM subsequently recovers this through price increases, but the fact remains that MLC software reduces in price as newer machines are used. The exact mechanism used to implement these improvements since 2003 varied from hardware generation to generation and is often referred to as a ‘Technology Dividend’. Initially it was by increasing the MIPS to MSU ratio, which affected all IBM software and some independent software vendor (ISV) software. Since this also reduced the quantity of OTC licenses required, IBM changed the mechanism and introduced new ‘advanced’ price tariffs with the z196/z114 ranges. For the z12 generation, there is a graduated (according to machine size) discount off z196/z114 prices. Initially these reductions were in the range five to six percent off all software; however, recently this has been reduced to four to five percent just for the variable WLC products. Nevertheless, it is still worthwhile to assess the impact of the changed pricing metrics for cost reductions that could be achieved by moving to the newer machines.

Specialty processors, such as zIIP and zAAP, are cheaper to buy than standard CP processors and carry no software charges. On average, in the mature markets of Europe and in the USA, roughly one-sixth of processors installed are specialty processors. Some vendors have re-written much of their software to run on zIIPs using the approved IBM development APIs and continue to offload more with each release. It’s worth reviewing regularly, but instead of asking if a vendor offloads tasks, ask how much of the workload is offloaded as it would apply in your environment during the peak period. Some vendors offload over 90 percent, while others only offload around 10 percent of workload. zIIPs and zAAPs can reduce the peak R4HA, but you should check that the work being offloaded is priority work that has to run during the peak period (steps 2 and 3). The same applies to accelerators, such as the IBM DB2 Analytics Accelerator (IDAA). Workload offloaded to a zIIP will bring related additional hardware and maintenance savings.

Installing the newer versions of software, particularly key middleware like CICS, IMS, and DB2, can reduce CPU consumption. That is because in addition to normal code optimization, IBM incorporates code that exploits...
improvements in the base machine architecture. This used to be a trade-off because newer versions cost more and usually the performance improvements were negated by price uplifts for newer versions. Some customers have experienced that recent versions of DB2, for example, have tended to deliver marginally more price performance than the associated version price uplift, making it worthwhile for customers to attempt to maintain high middleware currency. Additionally, since IBM has recently started increasing the prices of older versions, one might as well upgrade to the latest version sooner rather than later.

6. REVIEW THE SCRT PROCESS
Every user with any kind of IBM WLC pricing has to submit an SCRT report each month. We have found that many customers simply print it and ship it off to IBM, waiting several weeks to see what the license fee will be. We recommend that users print the report as soon as the expected peak has occurred. This allows time to review the results and investigate any deviation from expected MSU values and budget before the report has to be submitted. While users are not allowed to change SCRT reports, they may address abnormalities by using the EXCLUDE statement. This removes the SMF data related to erroneous jobs caused by loops, bugs, or other unplanned and abnormal events prior to SCRT processing. Exceptions requested after a report has been submitted or the actual invoice is received are rarely approved. An example of this situation is in Case 3 of Step 2 above.

After reviewing the peaks, using the developed costing model previously discussed, customers can calculate the precise cost themselves and subsequently check the IBM invoice.

7. MANAGE THE ENTIRE SOFTWARE PORTFOLIO
It is still common for many organizations to have enterprise portfolio licenses from several vendors with overlapping functionality. They may have even deployed products from two or more vendors to do basically the same thing. While there may appear to be little immediate benefit from standardizing on one particular product, it makes long-term sense to rationalize the vendor portfolio in preparation for the next contract negotiation. From an MLC standpoint, there are several products or priced features of products that can be easily replaced with perpetual license equivalents from other vendors. The key ones are QMF (priced feature of DB2) and several of the priced features of z/OS, such as RMF.

8. EVALUATE POSSIBILITIES TO QUALIFY FOR SPECIAL-SITUATION TERMS
Standard software licensing terms are intended to cover the majority of situations, but there are times when a particular industry or segment is required to operate in a specific way due to external regulatory or governance rulings. This might introduce a conflict with the standard licensing terms and consequently result in significantly higher software licensing charges. However, if the requirement is truly externally imposed and unique then most vendors will amend their standard terms (if they want to do this and can convince their own external regulators and accountants that this is a valid exception).

9. INVESTIGATE ALTERNATIVE LICENSING METRICS
Most of the software licensing from IBM and ISVs is based on high-water-mark usage or total machine capacity. However, if your company profile does not align well with the average, an exception may be negotiated or there might be an alternative metric available. Additionally, if you’re finding that it’s cost prohibitive to bring incremental workload into the mainframe environment with a new business application, most vendors will consider terms that restrict the license to the new environment and therefore attract a lower cost. IBM offers zNALC and Solutions Edition, which discount the software and associated hardware. There are also other announced and unannounced special metrics, which can be negotiated. Having information on the exact product usage is a key starting point for vendor negotiation. Customers who have knowledge of the terms that software vendors have offered to other customers in similar situations usually gain some price advantage—especially when they know how to build a case for it.

10. PLAN PROCUREMENT NEGOTIATION STRATEGY
For all vendors, we have a principle that one should buy only what is really needed and use all of what was purchased. There is no benefit in having shelfware. Given that most vendors look for long-term commitments, it is important to
plan ahead carefully before making a contractual commitment to any vendor. Understanding which products could be replaced and having a replacement alternative for all products will ensure a stronger negotiation position when the time comes to renew a relationship agreement.

For example, the composite software agreements from IBM work off an MLC base run rate. If customers expect to reduce the MLC over time, they need a clear understanding of how this is going to be achieved before discussions with IBM take place. They also need to implement entitled reductions before composite contract negotiations start. Many customers delay starting projects to reduce MLC spend, because the short-term benefit is not sufficient if there is already an MLC-committed spend rate in place and if agreed caps are not being fully utilized. This approach can result in perpetual paralysis, stifling progress on cost reduction.

Research has shown that in any buying transaction it pays to be the first to make an offer—if that offer is realistic. Unfortunately, with complex agreements, it can be very hard to judge what the realistic price point should be. As a result, many customers allow the vendor to lead and then subsequently try to wear them down. Few of the organizations that choose to lead by communicating the desired price points and required terms and conditions manage to get a realistic initial proposal, unless they enlist expert advice. Without consultants or extensive sophisticated internal cost management and optimization practices in place, many mainframe customers end up either paying more than they should or fail to reach agreement. This is because the target they set is outside of the vendors’ legal or financial possibilities.

CONCLUSION
Organizations that follow a set of clearly defined actions and processes can optimize software licensing charges and significantly reduce mainframe spend. There is no silver bullet, instead this is an iterative process that requires continuous focus over an extended period. Just as performance tuning for service level improvement requires ongoing effort, MLC optimization should not be viewed as a one-time effort. Very often these activities are postponed or assigned a low priority, because the payback is not immediate. Put simply though, the sooner you start taking action, the sooner you will reduce costs.

The key to this, as well as to fair and sustainable charge back practices, is for the operational units of mainframe customers to have comprehensive and timely access to the information and knowledge on which to base business, technology, and negotiation decisions. Availability of tooling to extract the large amounts of data from SMF and then model that in a way that simulates variable licensing in a virtualized environment is long overdue. For many years we have witnessed users struggling with million-row spread-sheets, but considering the potential for financial savings it is time to implement a more structured and better engineered approach.

Having comprehensive cost analysis tooling not only helps reduce the cost of running existing mainframe workloads but also helps build business cases to bring additional workload into the mainframe environment at no additional, or a very marginal incremental rate—thus optimizing IT in the broader sense.

These activities will become increasingly important over time if mainframes are to be optimized and used for cloud hosting, and as mainframe-style licensing metrics become more commonplace in the virtualized blade environments being created to support the Cloud and Software-as-a-Service (SaaS) offerings.

To get started:

» Establish a process and strategy for MLC cost reduction
» Support the process with appropriate tooling and required organizational support
» Engage LOBs (where applicable) in costing analyses and decisions
» Document and share cost reduction success with stakeholders
» Maintain constant, continuous execution of the process.
BMC SOFTWARE. IT’S AMAZING WHAT I.T. WAS MEANT TO BE.

BMC Software helps leading companies around the world put technology at the forefront of business transformation, improving the delivery and consumption of digital services. From mainframe to cloud to mobile, BMC delivers innovative IT management solutions that have enabled more than 15,000 customers to leverage complex technology into extraordinary business performance—increasing their agility and exceeding anything they previously thought possible.