

IBM System z TCO: Man Bites Dog?

Quick Note

Wayne Kernochan
29 November 2006

One often sees the fake news headline “Man Bites Dog” used as an example of a startling reversal of the natural order of things. In the IT world, that natural order for over three decades has included the mainframe as high-cost alternative—very capable, but priced accordingly. But, recently, IBM conducted a series of TCO (total cost of ownership) studies suggesting that over the last three years, significant changes in the mainframe’s TCO costs, taken together with changes in the typical computer workload, have led to mainframe TCO advantages of between 5 and 60 percent over typical Unix, Linux, and Windows alternatives. It’s arresting, to be sure, but how credible are such claims? Is this just TCO foolishness that presents a view of the mainframe taken through a distorted and rose-colored lens? Is it truly a “Man Bites Dog” story about plucky mainframers aligning themselves with new ways that enterprises are starting to look at costs? Or is it something broader than that?

The New Mainframe; The New TCO

The mainframe has clearly seen substantial updates over the last decade, becoming more affordable, more attuned to modern APIs and middleware, and more network-savvy.¹ But in the grand scheme of things, what has changed isn’t so much the mainframe itself as the uses to which it (and all servers, for that matter) is being put. Increasing demands from both business applications and the Web, as well as the increasing centrality of the network and widespread server consolidation initiatives, have blurred the lines between central sites and the rest of the computing power of the organization. No servers or applications can act in isolation; they must integrate into an enterprise architecture, specialize in particular workloads, and flexibly adapt as the needs of the enterprise change.



The mainframe has changed to meet these needs and thus is no longer your father’s System z. This has a direct positive effect on the mainframe’s TCO. For example, today’s mainframe administration reflects a standardized enterprise-wide approach rather than something performed in an isolated glass room using skills

Copyright © 2006 Illuminata, Inc. Licensed to IBM Corporation for web posting. Do not reproduce. All opinions and conclusions herein represent the independent perspective of Illuminata and its analysts.

¹ See our [The Mainframe Reloaded](#).

both uniquely specialized and scarce. But the *major* change in mainframe TCO comes even more from new ways of measuring and thinking about TCO itself—ones that reflect new mandates.

Specifically, it no longer makes sense for the large enterprise to measure TCO strictly on a one-application-per-server basis. Because servers are so flexible, both scale-out and scale-up increasingly look like different sides of the same coin; adding a new application or scaling an old one is not a matter of buying a pre-bundled stand-alone product but of incrementally adding a blade, a rackmount server, or a virtual machine within a Unix box or mainframe. If we are to measure apples against apples, we must measure TCO for 10, 20, or 50 applications running on one mainframe, 10-20 blades, or a grid of 50 distributed systems.

And this is the basis for IBM's TCO studies.

Analysis of IBM's TCO Results

IBM correctly breaks down costs of ownership into hardware, software licenses, people (administration, implementation, and upgrade), and environmental (e.g., electricity and air conditioning) costs.² This is fairly standard stuff.

But, instead of considering single-application TCO scenarios, as is often done, IBM assesses cases where tens of applications are running, either on a single mainframe or on multiple distributed Linux/Unix or Wintel platforms. This, IBM notes, is a typical usage model in medium-sized to large enterprises.

IBM cites results that range from marginal TCO advantages (System z vs. System p) to estimated 30 to 60 percent improvements (vs. 30 Sun servers, vs. the IBM System x, and vs. 300 Linux servers of an unspecified brand). System z wins on the basis of administrative costs; its software costs are about the same, and its hardware costs much more. The

² Environmental costs are often highly significant for data centers in major cities where space is at a premium, and often viewed as insignificant in other cases. However, power and cooling costs are rapidly gaining mindshare even outside of New York, London, and Tokyo—if only because of the relentless “eco marketing” of Sun and others.

number of administrators required to run the operating systems, hardware, and networks of the distributed systems increases nearly linearly as the workload scales; while a single mainframe's administrative costs barely increase at all. Thus, for the typical IT shop with lots of different workloads, mainframe people costs are often a fraction of those costs required for distributed systems.

To be sure, there are exceptions. IBM understates the importance of the administrative costs of application and infrastructure management, and especially of database management³. In a world where just four Oracle instances frequently require a dedicated administrator, database administration costs can reach \$10 million in a three-year period for an application with 50 Oracle copies. However, these costs apply to *any* platform, and therefore do not affect the relative TCOs of the mainframe, Unix boxes, and PC servers. Therefore, in spite of glossing over this particular variable, it remains a valid point that mainframes are often cheaper—even *considerably* cheaper—to administer in a large, mixed-workload environment.

Meanwhile, mainframe software license costs—often cited as a concern in user surveys—are now actually competitive with other platforms. A key reason for this is IBM's focus on driving these costs down. For example, IBM's database-administration tools have had the effect of driving the prices of competitors such as CA and Compuware down significantly; IBM's license costs per unit of workload also go down as the workload increases. Specialty co-processors (zIIP for Java-related workloads, IFL to run Linux, and zAAP for WebSphere) offer even lower hardware and software pricing.

And, while still quite pricey by volume server standards, the price of mainframe memory and other components have been significantly chopped as well, with continuing aggressive price cuts planned. IBM's focus on cooling technology also means that the mainframe should typically require less electricity and air conditioning than many 1U servers running the same workload.

³ Studies suggest that application server administration can be surprisingly expensive as well.

Add it all together and mainframe TCO can look quite attractive. Administrative costs are often lower—even much lower—and neither hardware, software, nor physical plant costs are anything like the drag they once were.

Findings on Overall TCO Trends

IBM’s TCO findings also show clear and major trends in TCO over the last ten years that are driving these results. The three key trends that they identified are:

- An increase in the people-cost percentage of TCO, to the point where in many cases people costs are greater than any other cost.
- An increase in the software-license cost percentage of TCO—but at a much slower rate than people costs.
- A dramatic decrease in the hardware cost component.

Recent studies⁴ have also suggested that:

- Administration costs, including deployment, upgrade, and maintenance—and especially database administration costs—are the dominant component of TCO, and their percentage of overall costs continues to increase.
- Simplicity lowers administration costs. This simplicity includes both automating user administrative tasks and reducing complexity of the application and enterprise architecture.
- Programmer productivity is now a small but significant determinant of people costs. It is a large and increasing component of ROI, because the useful life of a given application is shrinking as competitors find easier ways to match an application’s features. For example, developing an

Table 1: Trends in TCO over the Last Ten Years

Component	1995 % of TCO	2004 % of TCO
People Costs	14 %	43 %
Software Costs	14 %	27 %
Hardware Costs	65 %	20 %
Other (environmental, average)	7 %	10 %

Table 1: Source: IBM Corp., May 2006

application one year faster means 33 percent better ROI when the useful life of the application is three years, whereas the same speedup only brings 20 percent better ROI to an application with a longer five year lifecycle.

Putting these two sets of findings together suggests four key trends that are likely to affect IT buyers most in the near future:

- Administrative costs will dominate almost all implementations, from the high end to the low end.
- Total cost of ownership is becoming comparable across all major platforms. Note that, although IBM’s mainframe advocates argue that today’s distributed systems scale almost linearly in administrative costs while System z does not, savvy users of distributed systems can use technologies such as clustering and virtual machines (such as from VMware or IBM’s Advanced POWER Virtualization) to bring some of the same benefits to the more scaled-out world.
- Architectural simplicity is becoming a key determinant of administrative costs, as is automation of administrative tools.
- Productivity—the ability to create, deploy, upgrade, and maintain software rapidly—is becoming a larger component of both TCO and ROI.

⁴ These conclusions are based on Infostructure Associates studies over the last seven years on TCO and ROI for SMBs and their ISVs, focused on the TCO/ROI of databases and infrastructure software “platforms.” Anecdotal evidence suggests that the same trends hold true in large enterprises’ data centers.

Beyond the Mainframe

TCO can serve as an IT “canary in the coal mine” to detect dangerously costly and outmoded assumptions. In this case, the results do indeed show that the mainframe is a cost-effective platform for datacenter workloads. But they also tell us that virtually any hardware platform can be appropriate for most applications, if strictly judged on costs. How so? With the ability to convert any application into a Web service provider in a platform-neutral SOA (service-oriented architecture), and with virtual machines and other forms of partitioning readily available, most of today’s platforms can flexibly run most of today’s applications. Therefore, the “ecosystem” of applications and software associated with a specific platform is neither the leverage point nor the lock-in it used to be. Even robustness and security differences between platforms are much less sharp than they once were.

That leaves, as differentiators between architectures, things like *simplicity* (ease of use and administration, as well as architectural simplicity) and *productivity* (support for rapid implementation, deployment, and upgrade of innovative software and products). Because these are more *people-oriented* criteria than are attributes like license costs, flexibility, or robustness, tomorrow’s IT should increasingly be focusing less on the immediate effects of an acquisition on the bottom line and more on its long-term effects on IT and business processes.

IT needs to place greater emphasis on architectural simplicity and administrative-tool automation in choosing a platform/solution. That is, does the new system require new skills, use tools requiring massive numbers of administrators, or require new connections to most existing systems? Or does the new solution simplify administration so that a non-specialist can handle it, avoid the need for additional application, database, storage, or other administrative personnel, and minimize specialized training?

IT needs to focus more attention on the ability to develop new sub-applications, and to upgrade the technology of the application, swiftly. For example, can ordinary programmers or knowledge workers—rather than specialized data miners—easily generate reports, mini-business-process software, or composite applications based on the application? Turning applications into Web services and choosing development tools that operate on data at a high level are both key to major improvements in programmer productivity.

Finally, in acquiring and implementing new systems, IT needs to focus more on how those systems affect IT processes and business processes. For example, how will the backup/recovery process expand to handle the new workload? Does an existing book-to-bill process need to be reengineered to take advantage of the new system? By focusing on these questions, IT is likely to have a greater positive impact on long-term TCO and ROI than focusing on license costs will.

Conclusion

IBM’s TCO results are indeed credible, and reflect a new lease on life for System z and the mainframe. The mainframe is no longer the high-priced spread. But this is not just a story about the mainframe. It’s also about a new way of planning new IT initiatives—a way in which, as one user recently put it, “when you are implementing an SOA, you don’t pick the box first; you decide what the services should be and then decide what mix of platforms makes the most sense for the services.”

While IT should reassess the value of System z, it should even more emphatically focus on simple architectures and automated administrative tools, on the effects of implementing business processes rather than just license costs, and on maximizing the productivity of consolidated Web services/solutions rather than the productivity of one application per box. In that case, the new headline will be not “IT to Mainframe: Drop Dead” but rather “It’s the People Costs, Stupid.”