The Mainstream
An article from the IBM @server zSeries software newsletter

Find the current version of The Mainstream here **ibm.com**/software/zseries/mainstream

"The" on demand engine

from The Mainstream, Issue 9 - 2004
The IBM @server zSeries and S/390 software newsletter

The z/OS® operating environment is a general-purpose operating system on which many businesses rely. Commonly referred to as "mainframes," zSeries® processors (and their predecessors) have been the backbone of commercial computing for decades, renowned for their reliability, scalability, availability and other industrial-strength attributes. But why the distinction between mainframes and other types of servers?

Historically, some platforms were designed for scientific and technical computing, with massive, extended "number crunching" on relatively small amounts of data and relatively low interactions between the processor and external storage. Other platforms were designed for a commercial environment, with exactly the opposite characteristics: constant movement of very large amounts of data between the processor and external storage, and processor resources consumed in short bursts for each transaction or read/write operation. The IBM zSeries server and its z/OS operating system were designed for just this kind of commercial environment.

zSeries is designed for the real world

z/OS is based on a unique architecture that shares enterprise data and dynamically allocates resources across multiple servers and heterogeneous workloads—which means zSeries systems can move the large amounts of data required in a typically data-intensive e-business application among processors, memory and I/O at a rate of up to 25 times that of other high-end servers. Its usable capacity allows z/OS to effortlessly handle workload spikes and enables customers to squeeze maximum utility out of their computing investment while keeping their end users satisfied. zSeries is designed for the real world, and therefore, a total cost of ownership study is the only way to make an equal comparison.

ibm.com/software/
zseries/mainstream

Platform decision-making criteria

The decision of whether to run new workload applications on z/Series or distributed servers is not necessarily an either/or proposition. You can do both, allocating some to each platform. In evaluating where to deploy new workloads, there are three decision criteria to consider:

Functionality. Does the platform allow you to do what you want with z/Series and z/OS?
Capability. Do you have to use different tooling? Do you have to have different skills?
Cost. What are the real costs of deploying on a mainframe using z/OS relative to those of distributed servers?

1. Functionality

From a server/mainframe comparison standpoint, functionality has really become a non-issue, because the underlying technology associated with applications like IBM WebSphere® Application Server is nearly identical on each platform. As a result, standard applications can be developed in the same manner for z/OS as they are for distributed environments, then deployed to the better platform for each application depending on the business need.

2. Capability

To decide where applications should be deployed, businesses are identifying the level of complexity in applications—including transactions, logic, workflow process, people, automation capabilities, monitoring and tracking requirements.

Skill set is not an issue in terms of building or deploying these applications. zSeries running z/OS has exactly the same programming model, development model, development tooling and administration as a distributed server platform. So, for example, your developers can use IBM WebSphere Application Studio Tooling to build new Java™ applications on a Linux® system, whether you intend to deploy those applications on a UNIX® platform or on z/OS.

3. Cost

The traditional thinking is that server-based computing is more cost-effective than mainframe computing. The problem is, that thinking is based on standard benchmarks, such as production-

only functions with security turned off. That doesn't even come close to a real-world server environment.

The cost picture changes dramatically when you look at a more realistic scenario: a total distributed environment, with available on-the-floor capacity encompassing storage, testing, production and disaster recovery. Accommodating this capacity requires scaling up with more servers—which is less efficient and requires additional software to run on them.

Consider the necessary inefficiencies of a server environment. Since companies need to over-configure their systems for spikes, disaster recovery and the like, servers usually run at about 20 to 30 % of capacity. Translated, it means significant unused, wasted computing capacity—that is, idle inefficiency.

Consider also the labor costs. Add servers, and you must add staff to manage them. New distributed technology requires new skills, too, because there is new programming involved. The fact is, these additional labor costs can far exceed the cost of hardware and software licensing combined.

Beware the benchmark trap

Total cost of ownership (TCO) is too often seen as simply the capital costs required to get the server running. These generally include costs for the hardware (CPU, memory, disk, network interface), the license for the operating system, and the license for the software and companion tools and utilities, as well as applications. These are easily quantifiable and accounted for, but they represent only a portion of the total cost of ownership. While there are several different elements of cost that are important, many cost sources are often overlooked.

Benchmarks do not adequately reflect operational resource requirements—such as the fact that distributed servers run at 10% or less on average in production. But benchmarks don't consider availability, security and development/testing. The actual operational footprint is usually much

ibm.com/software/zseries/mainstream

larger than benchmarks indicate. Industry-standard benchmarks measure price/performance under well-controlled conditions, which typically include:

- Static loads, unconstrained resources
- Single application per OS, single OS per box
- Tuned and driven to 100% utilization

Through a number of consulting-led engagements, IBM has developed a methodology that can assess the cost and efficiency of the current IT infrastructure. It typically starts by producing a series of three server "scorecards" for mainframe, UNIX and Microsoft® Windows NT® servers. Each scorecard includes four financial/cost metrics: IT people efficiency, server utilization, application availability and total cost. These metrics can give an accurate cost and service-level snapshot of very complex server infrastructures in any enterprise, and can be used to model and build alternative future server investment cases. Most importantly, these engagements have shown that there are major misconceptions of the relative cost of deploying new applications on mainframes, UNIX and Windows NT servers.

> *z/OS is a platform that provides the lowest total cost of ownership for larger heterogeneous corporate ecosystems, when all terms of the cost equation are accounted for.*

The conclusion of many of these cost studies is that, for a "typical" online application, the following cost ratios hold true,when all the people, hardware and software costs are included in a three-year business case.

For a representative zSeries solution, incremental hardware, software and people costs are $1 million annually, dominated by ISV software costs.

For a comparable UNIX solution, incremental costs are typically $1.6 million to $2.4 million per year, dominated by people costs. In addition, unscheduled and scheduled outage costs can typically add $1 million to $2 million to UNIX and NT three-year cases.

Consider this

Here are a few things to consider when comparing platforms based on total cost of ownership:

Total costs must be predictable

- Service delivery people cost dominates total systems cost
- People efficiency is a key selection criteria
- Look for automatic tuning and allocation of resources; self-healing systems

e-business workloads are volatile

- 10:1 spikes common, 5:1 typical peak to average
- Systems efficiency is a key selection criteria
- Look for ability to prioritize and run mixed workloads

Outages and response time are highly visible to customers

- Operator error continues to be the leading cause of downtime
- Cost of downtime must be limited
- Look for avoidance of errors; automatic recovery from errors that do occur; non-disruptive change

e-business requires growth without limits

- Non-disruptive growth in the face of fast growing capacity demand
- Service delivery must be flexible, incremental and cost-effective
- Look for just-in-time, granular upgrades

Peaks and spikes: a 10-to-1 ratio on average

E-business applications involve dramatic swings in user activity, with orders of magnitude changes occurring in seconds. In order to survive these peaks, many platforms are typically over-configured to run at a peak CPU utilization of 50% to 60%, with average CPU utilization of 20% to 30%. In general, even the most highly loaded database server platforms usually run at 60% to 70%. Why? Because each distributed server is dedicated to a particular workload, and a

distributed system must be over-configured to allow for spikes. Distributed technology does not offer the virtualization and resource pooling capabilities that zSeries offers today.

It's not uncommon for a mainframe to support server utilization rates of greater than 90%—meaning that that asset is busy, on average, over a 24-hour period from 70%, 80% and even 90%. Contrast that to a typical Intel® or UNIX environment, which tends to run a single application per server where these environments tend to be utilized at less than 20%; in effect, 80% of your assets are not being utilized.

When customer data is analyzed, it becomes apparent that many organizations have a ten-to-one ratio of spikes to normal deployment, with an average of five-to-one. Assuming the more conservative, five-to-one peak-to-average ratio, most servers in a distributed environment must be configured with a substantial amount of unused and spare capacity—which means your servers are not operating efficiently. The result: More servers are needed, and more people are required to manage those additional servers.

zSeries systems, on the other hand, can automatically and continuously reallocate system resources (processors, memory and I/O) in response to changes in demand. This allows z/OS to run at a peak of 100%, and an average of 65% to 75%—although it is fully capable of running at 100% CPU utilization 24 hours a day.

This capacity capability is overlooked when platform comparisons are made in benchmark studies. Traditionally, what's measured in a distributed benchmark is only a fraction of the actual servers required on the floor—because of the need to allow for spikes, rollover servers, the manpower to manage the additional servers, and the resources required to meet additional software licensing costs.

Cost of ownership—looking at the complete picture

Let's compare a Sun Solaris™ and a zSeries solution deployment. The Solaris configuration is comprised of 24 images across 23 servers running production. However, what's required to deploy it is actually 73 images running across 62 servers. The reality is you need three times as

much capacity on the floor than what the original benchmark includes. At first glance, you would look primarily at the cost of the production servers, estimating that a Sun environment would be about one-fifth the cost of the zSeries solution.

Using this type of calculation, the original cost-estimate would include only the production servers in a typical environment, which accounts for about one-third of the total number of servers required. This cost-equation doesn't include development servers, test servers or the disaster and recovery servers required.

In essence, then, what first appears to be a viable, less-expensive solution, begins to increase in cost considerably when you take into account:

- The additional servers required (three times more)
- Cost of hardware and the people to run it
- Licensing charges for the software running on 62 servers

The cost of ownership adds up quickly. Using a fixed ratio of one-person-to-five-servers, you'll encounter erosion of both people efficiency and systems efficiency. On a three-year total cost of ownership basis, the true cost of ownership of the distributed solution ends up at about 10% more than the zSeries solution.

Mainframe capabilities differentiate and deliver TCO benefits

More efficient utilization of server capacity, higher people productivity and lower environmental costs add up to total cost of ownership advantages and reduced IT complexity. It is these advantages that have many CIOs concluding that every infrastructure can benefit from the proven capabilities of a mainframe, and that the role of the mainframe in e-business continues to evolve.

The zSeries platform offers:

- A basic capability to divide a single server into multiple partitions or virtual servers, each one running multiple applications simultaneously and securely.

ibm.com/software/zseries/mainstream

- The ability to self-manage the allocation of system resources based on business priorities among multiple workloads and virtual servers, allowing the mainframe to sustain very high utilization rates (90+%) while meeting IT service level objectives. This compares to average utilization rates of less than 30% for UNIX and Microsoft Windows NT environments
- Industry leadership availability and scalability characteristics that allow for consolidation of large applications.

When 10% just doesn't cut it

Many technologies across the IBM server brand today originated on the mainframe, including logical partitioning, capacity of demand, "self-correcting" hardware and workload management, to name a few. Here are some key advancements that have been introduced that will help ensure continued TCO advantage in the future:

- Intelligent Resource Director—the ability to automatically allocate system resources to the applications and virtual servers defined as the highest priority to a business.
- HiperSockets—a TCP/IP LAN in memory capability that reduces network infrastructure requirements and improves communications between virtual servers.
- Extensions to industry leading capabilities like Parallel Sysplex clustering technologies and system automation.

While 10% is a solid savings, you may not feel that it warrants reevaluating your platform deployment decision. Perhaps you feel there is less risk with the distributed solution. Well, now there is a real game-changer on the horizon. In April, IBM announced the zSeries Application Assist Processor (zAAP).

The first processor totally devoted to Java

The zSeries Application Assist Processor (zAAP) is the first processor totally devoted to Java. It is a full z/890 or z/990 processor that can be purchased for a fixed price of $125,000 US per engine. zAAP is a significantly less expensive general-purpose engine that provides an economical Java execution environment for customers who want the traditional qualities of service and the integration advantages of the zSeries platform.

ibm.com/software/
zseries/mainstream

So how does zAAP work?

When configured with general-purpose processors within logical partitions running z/OS, zAAPs may help increase general purpose processor productivity and may contribute to lowering the overall cost of computing for z/OS Java technology-based applications. zAAPs are designed to operate asynchronously with the general-purpose processors to execute Java programming under control of the IBM Java Virtual Machine (JVM). This can help reduce the demands and capacity requirements on general-purpose processors, which may then be available for relocation to other zSeries workloads.

The IBM JVM processing cycles can be executed on configured zAAPS with no anticipated modifications to Java applications. Execution of the JVM processing cycles on a zAAP is a function of the Software Developer's Kit (SDK) 1.4.1 for zSeries, z/OS 1.6, and the Processor Resource/Systems Manager™ (PR/SM™). The zAAPP engine executes Java cycles with no application software charges. So basically, all Java that executes in this environment essentially runs at no cost after you've purchased the engine. Your general-purpose processor savings will vary based on the amount of Java application code executed by zAAPs. This is dependent upon the amount of Java cycles used by the relevant applications and on the zAAP execution mode selected by the customer.

Execution of the Java applications on zAAPs, within the same z/OS SMP LPAR as their associated database subsystems, can also help simplify the server infrastructures and improve operational efficiencies. For example, use of zAAPs could reduce the number of TCP/IP programming stacks, firewalls and physical interconnections (and their associated processing) that might otherwise be required when the application servers and their database servers are deployed on separate physical server platforms.

zAAPs offer new economics for zSeries

zAAPs provide investment flexibility and new economics for zSeries. They can increase system productivity by reducing the demands and capacity requirements on general-purpose processors, which may then be available for reallocation to other zSeries workloads. Before, IBM WebSphere may have been isolated in its own partitions in order to drive down software licensing charges.

ibm.com/software/zseries/mainstream

With zAAP, WebSphere workloads can now be consolidated. The result: higher efficiencies, gained from combining traditional transaction applications and WebSphere workloads.

zAAPs provide a more cost effective Java execution environment via reduced total cost of ownership (TCO)—since there are no no additional IBM software charges—and reduced acquisition costs for running J2EE workloads (it offers hardware, software and maintenance savings). The zAAP engine reduces TCO considerably, because, when you consider that you can now offload Java workloads can now be offloaded.

Let's rewind for a minute, and think back to the earlier example where the zSeries deployment model's TCO was 10% less than the Sun Solaris solution. Using the same data configurations used in the earlier comparison, if you recalculate those same projections—taking into account the zAAP engine's capability to offload 50% Java workload—that initial 10% savings increases to nearly a 40% reduction on zSeries and z/OS.

Of course, the amount of Java in an application varies considerably, from zero in a traditional IMS application to upwards of 98% in a typical XML parse. So the amount of Java offloaded will vary considerably. Typically, today's applications will run about 50% Java, which means that out of a given application, up to 50% of those MIPS can be offloaded to Java engines, and in effect not be priced from a software perspective.

zAAP in practice

Here's an example of how a zAAP might work. A typical transactional IBM WebSphere application takes 1,000 MSUs today on zSeries, consuming 80% of the machine or LPAR capacity. Assuming that half the application is Java-based, zAAP can reduce the standard CP capacity requirement to 500 MSUs—a 50% reduction—by taking on the processing of those cycles.

This can effectively reduce the capacity requirements on standard CPs, allowing them to process other workloads. The total system can now get more work done overall than it could without zAAPs. Best of all, Java processing cycles can be executed on the zAAPs with no modifications to the Java applications.

ibm.com/software/zseries/mainstream

zSeries: the platform for on demand business

With a zAAP installed, z/OS on zSeries is a natural fit for on demand business, for good reason: It was specifically engineered to be. As a sophisticated operating system, z/OS has built-in capabilities that can't be found anywhere else in the industry. Advanced clustering and data sharing is built in, thanks to Parallel Sysplex™. Sharing resources across multiple images is built in as well, with IRD. And a sophisticated transaction manager with rollback capability? It's built in, too.

And IBM WebSphere for z/OS, one of today's central tools for e-commerce and on demand business, is specifically designed to take advantage of z/OS capabilities. That's a very different approach from distributed environments.

In developing and deploying Web-based Java applications—especially high-volume, low-quality-of-service applications—you have a choice: Run them on your servers and manage the associated issues of scalability, security and total cost of ownership. Or run them on your existing, zSeries mainframe that was built for e-business—zAAP-equipped for up to a 40% savings in total cost of ownership.

A simple choice

The underlying value of zSeries continues to be its ability to run business critical workloads efficiently, and to do it in a way that is reliable, secure and highly available. You need this environment to allow you to achieve your business and continuance objectives. You need your infrastructure to be not only resilient but also to be responsive, flexible and able to handle changes in demand.

So, technology today needs to be able to quickly adapt to changes through intelligent provisioning technologies—through automation, workload management and technologies like virtualization—so that you can handle these fluctuations in demand that occur efficiently, effectively and without impacting service level agreements.

ibm.com/software/zseries/mainstream

Back to basics - security and availability

There is a renewed focus around security and availability, driven by organizations that are exposing more and more of their assets and customer information to the Web. The mainframe provides a robust environment for security in business and availability, as well as the basis for a strong business recovery environment. When you do business on the Web, you're open 24 hours a day. When you reach out to your customers and suppliers via the Internet, you're exposing customer information or supplier information that needs to be protected.

The mainframe is known for the quality of the service (QOS) that it has provided over the years—security, levels of security, reliability, availability and manageability. QOS is accomplished, not only through investments in the hardware, but also investments in the middleware and systems management software. The combination of systems management, middleware and hardware offer you an industry leading platform for running business critical applications.

This is the IBM sweet spot. The combination of z/OS and its exploitation of 64-bit zSeries hardware architecture and middleware allows IBM to provide you with an environment that supports an availability-level of less than five minutes of downtime per year. The environment can also scale to the point where IBM can support literally millions of transactions per day across a number of different industries, ranging from the financial sector to government to manufacturing to distribution.

Virtualization

An additional differentiator for zSeries is its capability to intelligently manage work across multiple sets of applications. The ability to run and support lots of applications in a single server environment, through virtualization support, workload management support and some of the systems management products and capabilities that are available enables you to drive the zSeries server at an 80% to 90% utilization rate.

The "z" in zSeries stands for zero down time

z/OS is the direct descendant of OS/390® and MVS, dating back to the early 1960s. Nearly every operating system that exists today builds upon the concepts and designs that were first invented

ibm.com/software/zseries/mainstream

for MVS and its ancestors. That lineage continues today as "new" features on other platforms, such as the ability to logically partition hardware or autonomously configure and self-tune in real time. These are features taken directly from zSeries and z/OS. This leadership in innovation is matched with a robust implementation that gives z/OS a well-deserved reputation as a rock solid platform.

As businesses have come to rely more and more on the continuous availability of their largest systems, the verification techniques used by IBM in developing those systems have had to evolve. Methodologies, techniques, and tools need continuous enhancements to develop the necessary verification processes that support development for a "zero down time" system.

Many z/OS customers have business requirements for continuous system availability. System down time or unplanned outages, even of short duration, can cost millions of dollars in lost revenue or other significant negative business impact. Thus, z/OS customers don't think about availability in terms of minimizing the time of an outage; they think in terms of minimizing outages, period.

From the beginning, for z/OS as well as its predecessors, a basic assumption has been that hundreds or even thousands of users would depend on it. That is why its error detection and correction systems are so deeply ingrained. In fact, the anticipated mean time between failures of IBM z900® systems approaches 30 years.

On a distributed platform, unscheduled outages occur for several reasons, including power failures, computer viruses, natural disaster, or product/platform failures. Steps can be taken to avoid such outages, usually by configuring redundant portions of the server environment. This significantly reduces the capacity available on the floor; servers are often servers are configured at well below capacity. Inadequate preparations can lead to downtime costs that are more severe than the costs to avoid them.

ibm.com/software/
zseries/mainstream

The bottom line

The bottom line is that zSeries provides the hardware and software components of an "on demand" stack that deliver unmatched stability and reliability. This diminishes the potential for outages, and provides unmatched quality of service.

The zSeries environment encourages planners and system architects to think about costs explicitly and comprehensively. It allows tighter control of the environment, and brings greater stability to the organizations key on demand processes, not to mention the mission-critical, "bet-your-business" applications.

Designed to keep costs down through higher resource utilization, and greater economies of scale, it often exploits the zSeries infrastructure that you probably already have in place. And perhaps most importantly, it enables you to free up your skilled people from common administrative chores so that they can focus on more important tasks and be more productive. Doing more with less. Isn't that what you're under constant pressure to do?

Find more information on the Web

WebSphere Business Integration Server Foundation

This white paper covers building and deploying service-oriented applications that extend and integrate existing IT assets.

ftp://ftp.software.ibm.com/software/integration/library/whitepapers/wbisf_dhbrown0414.pdf

IBM Redbook: XML on z/OS and OS/390: Introduction to a Service-Oriented Architecture

This IBM Redbook describes the use of XML on IBM servers running z/OS or OS/390, and how it can be extended to modernize legacy applications. It provides both a high-level discussion of service-oriented architecture along with practical, detailed information about XML.

http://publibb.boulder.ibm.com/Redbooks.nsf/9445fa5b416f6e32852569ae006bb65f/ecba76920e89360585256b87006dd7d8?OpenDocument&Highlight=0,waswebservices

ibm.com/software/zseries/mainstream

IBM Redbook: z/OS WebSphere Application Server V5 and J2EE 1.3 Security Handbook
What do you think of when someone mentions z/OS security? Probably of something that is trustworthy, or even impenetrable. Perhaps you also think of something that is a little complex and challenging to administer. This IBM Redbook will help application programmers, WebSphere and security administrators, and application and network architects to understand and use these products.

http://publibb.boulder.ibm.com/Redbooks.nsf/9445fa5b416f6e32852569ae006bb65f/73ebcdfb3ccdf3fd85256d4400680c13?OpenDocument

WebSphere Business Integration Family

http://www.ibm.com/software/integration/integrate/

The WebSphere Business Integration zone of Developer Domain

http://www.ibm.com/developerworks/websphere/zones/businessintegration

The Washington Systems Center Techdocs site

http://www.ibm.com/support/techdocs/atsmastr.nsf/Web/Techdocs

z/OS and OS/390 Internet libraries

http://www.ibm.com/servers/eserver/zseries/zos/bkserv/
http://www.ibm.com/s390/os390/bkserv/

z/Series and S/390® e-business Internet
This site presents a wide range of information including the latest announcements, customer success stories, and technical information.

http://www.ibm.com/servers/eserver/zseries/ebusiness/

ibm.com/software/
zseries/mainstream

Intelligent Resource Director

http://www-1.ibm.com/servers/eserver/zseries/library/techbriefs/pdf/irdtechbrief.pdf