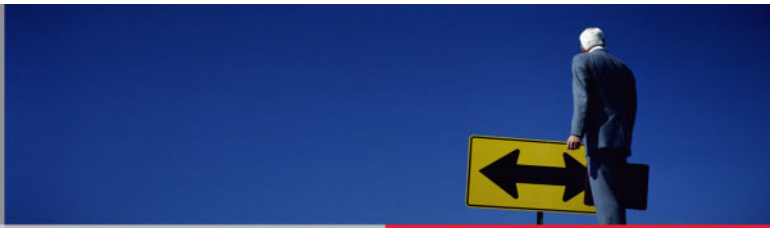
IBM

The IBM Mainframe Today: System z Strengths and Values



Redguides for Business Leaders

Mike Ebbers
Philippe Comte
Andrea Corona
James Guilianelli
Douglas Lin
Werner Meiner
Michel Plouin
Marita Prassolo
Kristine Seigworth
Eran Yona
Linfeng Yu

- Increase flexibility
- Respond to customer needs
- Improve cost effectiveness

Redbooks

Executive overview

IBM® mainframes process two thirds of the world's business information every day, and have been doing so for decades, because of the many strengths provided by the mainframe, such as high availability, good redundancy, high performance I/O capabilities, and much more.

This IBM Redguide™ publication describes the business drivers, their impact on IT, and how the System z® platform can be a major player for the business solution by providing flexibility, responsiveness to customers and market needs, as well as cost effectiveness. It also discusses the mainframe hardware, software, and environment, with an emphasis on the features and capabilities that make the System z mainframe the system of choice for a business enterprise.

A new way of looking at the mainframe

Although the mainframe is an effective processing solution, many people have the perception that the platform is also too expensive. When comparing a System z server with an Intel® system, the System z environment is in some ways more expensive. However, is the comparison a valid comparison of the two platform solutions?

IT administrators are tasked with reducing the IT budget while also facing an increased demand to give more results. Thus, the IT administrator is searching for alternatives. Studies have shown that the cost of staffing has increased dramatically over the past decade. In recent years, energy costs are also increasing. These trends, along with many others, indicate that the indirect cost of every IT environment is growing.

Experience shows that upgrading or installing a fix in a data center with hundreds or more blades or rack mount servers, most of them in a production environment, can take weeks and even longer. Alternatively, installing or even upgrading an entire System z operating system can take only seconds or a few minutes. Efficiency, achieved through automation and IT architecture, can lower administrative costs, which are the dominant component of total cost of ownership (TCO).

Calculating TCO or return on investment (ROI) of one application running on one or more Intel or UNIX® servers compared to running the same application on System z might show a result that is to the disadvantage of System z. For an accurate comparison, you need to compare all business-critical applications in the enterprise against the same number of applications running on a server farm. This kind of comparison can be difficult to make, however, because many companies do not have the correct numbers or cost information.

> *ROI has more credibility when it is stated in raw benefits, which are sometimes non-quantifiable, rather than translated into dollars. That translation is often fuzzy and tends to lose some audiences*.
>
> Sari Kalin, “Return on investment, Calculating ROI,” *CIO Magazine*, 15 August 2002

In this guide, we show why the System z environment can show better TCO and ROI than other systems. Although we do not quote pricing dollars in this guide, we do show directions and trends.

Business drivers

Businesses today seek to be more flexible and dynamic while also being more responsive to market needs. They want to continue doing business but be more cost effective. These challenges are major business drivers that influence business today.

Flexibility

The market today is evolving. Competition between companies is very aggressive, markets are global, and products must be innovated constantly to keep up with customer expectations. To compete, businesses need to produce products that respond to customer needs quickly.

Because of the rapid pace of market changes, companies need to be very flexible with their products and lines of business. In addition to market changes that put strains upon businesses, mergers and acquisitions are common business occurrences.

Businesses today more than ever need integrated processes. Processes for a single business unit are composed of several smaller processes, some which might be unique to a business unit, some which might be reused from other business units, and some which might be shared with other business units. The organization must be able to analyze specific parts of business processes and then change or outsource these parts without interfering with the overall processes.

Responsiveness

Along with flexibility to address change, every company needs to respond quickly to customers’ needs. Clients today know what they want, the way they want it, and how much they are willing to pay for it. In today’s very open and global market, it is easy for customers to switch suppliers when the product offering no longer satisfies them or is not available.

The shifts of customer demands require companies to be able to respond to increases (and, as a matter of fact, decreases) in demand very quickly. A business can serve more customers during a critical business period (such as a holiday season) without having to worry about reserving the necessary resources for those peak periods.

A customer often needs to deal with a new business direction as quickly as possible. Most business solutions are usually required in a few weeks instead of months or years. The business should know how to reuse its assets and how to implement solutions quickly to respond to market demands. To build a solution from scratch without reusing assets takes a long time—and that is something businesses cannot afford.

Cost effectiveness

The price of the products a company puts on the market must be in line with the competitive strategy. If the company is focused on cost leadership, this is an obvious requirement. However, even for companies with differentiation and focus strategies, there is constant pressure to be cost effective.

Business processes can be very complex. Businesses today are analyzing and redesigning their processes with the goal to identify and strengthen those processes that provide the company a competitive advantage. Optimizing processes in terms of reuse of resources can help business efficiency and can make businesses more flexible and responsive.

Impact on IT

These business drivers are connected to the information technology environment. To accomplish the business drivers, IT has to deal with infrastructure complexity and reliability, data center management, energy problems, and budgeting costs. This section describes these key points.

Simplifying the infrastructure

Modern IT infrastructure consists of multiple hardware architecture, a variety of software operating environments, complex internal and external networking, and applications that must interoperate in order to provide end-to-end service to both customers and internal users. The flexibility to meet new market demands and to seize opportunities before they are lost to competitors is a must for today's businesses.

To support these demands, the IT environment has to support changes on demand to the business needs. The IT systems must be able to communicate and integrate in a quick, flexible manner with other IT systems within the company, as well as with systems from partners, suppliers, and consumers in the external environment.

Figure 1 shows an architecture design of an IT infrastructure. The number of servers that are needed to manage the environment is very large with the following networks to manage:

- An internal and external IP network
- A storage network for the distributed servers (SAN)
- The System z connectivity network

Updating or maintaining the infrastructure can be a difficult job and at times seems impossible. Creating a disaster recovery solution for this environment can also be very complicated and demands continuous updates and maintenance. Backing up and monitoring processes requires careful coordination among all of the components in the infrastructure.

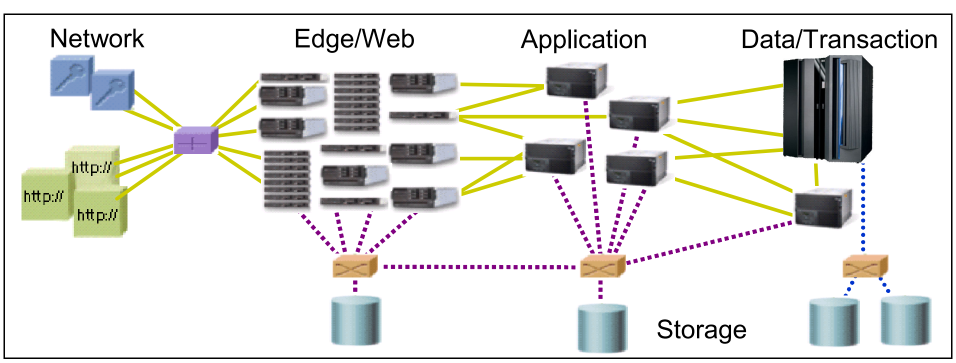


Figure 1 Infrastructure complexity

IT complexity impacts business process in that it requires more time to change and react to a new condition. Alternatively, simplicity of the infrastructure does not remove the need to change or adjust. It just allows it to happen faster.

For example, when you are evaluating, keep the following points in mind:

- The impact on the business from an unplanned outage or disaster
- Keeping up with planned maintenance
- Difficulty monitoring servers and business process flows
- Backing up

Reliability

The need to respond to the business needs pushes the IT environment to become more and more reliable. Reliability is an important part of the functionality of an IT architecture, and it is also crucial to the operating environment on which the architecture is deployed. When looking at reliability, the following considerations are important:

- Operating platform reliability and stability

  The IT must be available when the business needs them. Mission-critical application availability directly correlates to successful business operations. In today's on demand business environment, "downtime," whether planned or unplanned, is not only unwelcome—it is costly. Downtime statistics are staggering and range from tens of thousands of dollars to multiple millions of dollars per hour of downtime. In addition, although the direct financial impact is significant, the damage can extend well beyond the financial realm into key areas of customer loyalty, market competitiveness, and regulatory compliance.

- Predictable and consistent performance

  Even when strong fluctuations in demand occur, the supporting IT systems must remain responsive. Infrastructures must be adequately provisioned to handle peaks that can be predicted, such as seasonal increases in transaction volume, as well as those that might catch the IT department by surprise.

- Integrity

  When integrating different systems that are based on different technologies and possibly controlled by partners outside the control of the company, integrity becomes very important. Maintaining integrity of information exchanged between partners and systems is needed to ensure that information has not been tampered with during the exchange. Also the integrity of the business transaction that spans all of these systems must be controlled, and this requires support for atomic transaction capabilities.

- Security

  The IT systems and infrastructure must provide the level of security in accordance with the requirements of the business processes.

Cost effectiveness

As the need to provide higher levels of service rises, budgets tend to move in the opposite direction. IT executives are constantly challenged to provide better and additional service with smaller budgets.

In the early days of the IT industry, the hardware was expensive and the need for processing power was often at a premium. Applications were written by specialists in low-level coding languages such as Assembler, COBOL, and others in order to use a minimal amount of processing power. The expenses for those IT developers were costly. Today, IT can use high-level programming languages such as Java™, Visual Designer, and others that might use more processing power but that can be written by programmers who cost less to the businesses that hire them.

More hardware means using more energy, performing more hardware management, and in most cases deploying more servers with less than 50% resource utilization. Having this kind of unused or underused performance power and resources is not cost effective. Dealing with large numbers of servers with low utilization on each of one is a situation that most IT executives want to avoid.

New technologies based on sharing resources and data, plus the use of high-level programming languages, can help the IT architecture become more cost effective.

One of the problems that data centers face is the cost of energy. More servers require more and more processing power. Businesses continue to add more customers and applications to their IT environment, and that results in greater demands for processing power. The demand to scale up and scale out has an effect on energy consumption. Many data centers that are trying to minimize energy costs for computing power are being stretched to the limit. These data centers often need to spend more resources on cooling and on power electrical demands.

System z functions

When determining the cost of a platform, you need to take into consideration the specific strengths of the System z platform that contribute to the creation of infrastructures, making it easy to manage, control, and monitor. A comparison makes sense only if it is made between platforms with similar functions, from both an application and a management point of view. These functions include:

- System management
- Security
- Data integrity
- Disaster recovery

Recent studies suggest that operational costs are one of the major costs of an installation. One way to contain or reduce this cost is to simplify the enterprise, a goal that can be achieved through automation of daily tasks or through the linearity of the architecture. Many functionalities regarding system management are built into z/OS® and the hardware platform. Examples of these standard features are:

- Accumulation of accounting records to classify the users and what they consume
- Detection and recording of the system and application errors that might and might not impact the system in order to identify and diagnose potential problems, and reporting and dump generation for errors to analyze application malfunction
- Self-diagnostic and self-healing system components
- Dynamic adjustment of resources and priorities to guarantee the achievement of the desired service level
- Activation of spare hardware components (memory and processors) to guarantee continuous operations to the installation

Some of these functions are typical of and exclusive to the System z platform. Other platforms might be able to acquire some similar functionalities through the purchase of additional software or hardware products that can result in additional licensing cost and additional personnel needed to install, customize, and manage the additional elements.

Compared to other IT costs, people cost is on the rise. At the same time, the System z platform has proven in recent years that less personnel is required to manage and operate a growing environment from both the application and the workload points of view.

You need to take into account similar considerations when you need to determine the benefits derived from how easy it is to create a disaster recovery environment.

Quantifying all of these elements is somehow installation-dependent, but there are some general rules based on the industry sector that you can use to make an initial quantification.

All of these elements are important when you value the functionalities that are available with each platform and, when they are not present, the cost of adding such elements to the base.

The same concept applies to the need for a disaster recovery plan with a restart window that is becoming more and more narrow and with the requirements for data integrity and a minimum data loss.

In defining the optimal target platform for an application, you need to consider the cost of maintaining application development and test environments. The System z environment can optimize the required resources sharing the hardware and software across multiple application environments. It is not uncommon in the System z environment to have shared servers across both the production and the development environments.

Moreover, the System z platform, through the On/Off COD function, can adjust the resources to support specific stress tests or benchmark activities, such as verifying the behavior before introducing a new and complex application. The same would be more expensive on other platforms in terms of dedicated resources and the technical personnel required to set up and manage the environment.

Multiple and different workloads can coexist and share resources within the z/OS system; the Workload Manager component can manage them all according to their work priorities. In addition, with the IFL processor, it is possible to use the System z processors to consolidate distributed workloads and environments using the Linux® on System z operating system. With the software virtualization provided by z/VM® and hardware virtualization with PR/SM™ it is possible to create complex environments, often at a lower cost than when using distributed servers.

For all of these reasons, in the cost analysis, consider the capability of sharing common infrastructures and understand the potential synergy with the existing workloads (for example, peaks at different times).

The System z platform

Infrastructure simplification is key to solving many IT problems. Simplification can be achieved by resource sharing among servers. It is all about sharing data, sharing applications, and simplified operational controls. The System z platform, along with its highly advanced operating systems, provides standard format, protocols, and programming interfaces that enable resource sharing among applications running on the mainframe or a set of clustered mainframes.

Resource sharing is intended to help reduce redundancy that often comes from maintaining multiple copies of duplicate data on multiple servers. Sharing can also improve privacy management by enabling better control and enforcing privacy regulations for data sources. Sharing data can help simplify disaster recovery scenarios because fewer servers are being deployed; therefore, sharing data means that less data needs to be protected during periodic back-up operations (for example, daily or weekly maintenance) compared to having multiple copies. But most of all, infrastructure simplification helps a business assess its entire computing capabilities to determine the best directions and strategy for overall, integrated workflow—and in doing so, helps to take advantage of existing assets and drive higher returns on IT investments.

Using System z technology to reduce complexity

System z servers offer capabilities that can help reduce the size and the complexity of a modern IT infrastructure. The ability to “scale up,” or add processor power for additional workloads, is a traditional mainframe strength. Today’s System z servers are available with up to 54 processors in a single footprint. Businesses can order a System z server configured with less than the maximum amount of processor power, and upgrade it on demand, which means using a customer-initiated procedure to add processing power when it is needed to support new applications or increased activity for existing applications, without waiting for a service representative to call.

Processing power can also be turned on (or activated) when needed and turned off when it is no longer needed. This is particularly useful in cases of seasonal peaks or disaster recovery situations.

Adding processing power and centralizing applications represents one strategy to help control the cost and complexity of an infrastructure. This approach can also provide a highly effective way to maximize control while minimizing server sprawl—in essence, reducing the number of single-application servers operating in uncontrolled environments. A number of single-application servers can typically be deployed to support business processes in both production and supporting test environments. Hot stand-by failover servers, quality assurance servers, backup servers, and training, development, and test servers are some of the types of resources that are required to support a given application. A System z server can help reduce the numbers of those servers by its ability to *scale out*.

The term *scale out* describes how the virtualization technology of the System z server lets users define and provision virtual servers that have all of the characteristics of distributed servers, except they do not require dedicated hardware. They coexist, in total isolation, sharing the resources of the System z server.

Virtual servers on System z can communicate between each other, using inter-server communication called HiperSockets™ (think of it as an “in-memory” TCP/IP network). This technology uses memory as its transport media without the need to go out of the server into a real network, thereby simplifying the need to use cables, routers, or switches to communicate between the virtual servers.

Business integration and resiliency

We have seen how the need to be flexible and responsive drives businesses. If the site is not up or responsive to its clients or employees when they need it, the more likely it will lose the customers, or it will take the employees more time to do their jobs. A resilient infrastructure and integrated applications are also critical to the success of any business.

Availability

One of the basic requirements for today’s IT infrastructure is to provide continuous business operations in the event of planned or unplanned disruptions. The availability of the installation’s mission-critical applications, based on a highly available platform, directly correlates to successful business operations.

System z hardware, operating systems, and middleware elements have been designed to work together closely, providing an application environment with a high level of availability. The System z environment approaches application availability with an integrated and cohesive strategy that encompasses single-server, multi-server, and multi-site environments.

The System z hardware itself is a highly available server. From its inception all of the hardware elements have always had an internal redundancy. Starting with the energy components and ending with the central processors, all of these redundant elements can be switched automatically in the event of an error. As a result of this redundancy, it is possible to make fixes or changes to any element that is down without stopping the machine from working and providing support for the customers.

The System z operating system that sits on top of the hardware has traditionally provided the best protection and recovery from failure. For example, z/OS, the flagship operating system of the System z platform, was built to mask a failure from the application. In severe cases, z/OS can recover through a graceful degradation rather than end in a complete failure. Operating system maintenance and release change can be done in most cases without stopping the environment.

Middleware running on z/OS is built to take advantage of both the hardware and operating system availability capabilities. IBM middleware such as IBM DB2® for z/OS, IBM CICS® products, IBM WebSphere® Application Server, and IBM IMS can provide an excellent solution for an available business application.

System z Parallel Sysplex® architecture allows clustered System z servers to provide resource sharing, workload balancing, and data sharing capabilities for the IT, delivering ultimate flexibility when supporting different middleware applications. Although System z hardware, operating systems, and middleware have long supported multiple applications on a single server, Parallel Sysplex clustering enables multiple applications to communicate across servers, and even supports the concept of a large, single application spanning multiple servers, resulting in optimal availability characteristics for that application.

Parallel Sysplex is a cluster solution that is implemented from the IBM hardware to the middleware layer and, as a consequence, does not have to be designed and developed in the application layer.

With Parallel Sysplex and its ability to support data sharing across servers, IT architects can design and develop applications that have a single, integrated view of a shared data store. System z shared databases also provide high-quality services to protect data integrity.

This single-view database simplicity helps remove management complexity in the IT infrastructure. And simpler IT infrastructures help reduce the likelihood of errors while allowing planned outages to have a smaller impact across the overall application space.

Figure 2 shows the System z high availability family solution, from single system to the IBM Geographically Dispersed Parallel Sysplex™ (GDPS®).

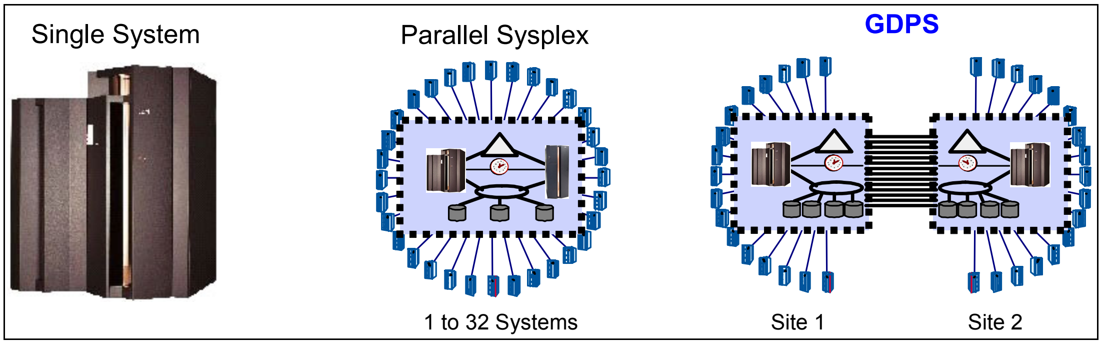


Figure 2 System z availability

GDPS technology provides a total business continuity solution for the z/OS environment. GDPS is a sysplex that spans multiple sites, with disaster recovery capability, based on advanced automation techniques. The GDPS solution allows the installation to manage remote copy configuration and storage subsystems, automate Parallel Sysplex operation tasks, and perform failure recovery from a single point of control.

GDPS extends the resource sharing, workload balancing, and continuous availability benefits of a Parallel Sysplex environment. It also significantly enhances the capability of an enterprise to recover from disasters and other failures, and to manage planned exception conditions, enabling businesses to achieve their own continuous availability and disaster recovery goals.

Hardware and software synergy

System z operating systems are designed to use central processors (CP). The vital connection between the hardware and the software resulted in the development of instructions for the central processor that over time are able to respond to new application demands. The System z platform database product DB2 for z/OS also makes use of the specialized instructions to speed up some basic database calculations.

The IBM System z Application Assist Processor (zAAP) is used by the z/OS Java Virtual Machine. z/OS can shift Java workloads to this new zAAP, thereby letting the CP focus on other non-Java workloads.

The IBM Integrated Facility for Linux (IFL) is another processor that enables the Linux for System z operating system to run on System z hardware.

The IBM System z9® Integrated Information Processor (zIIP) is yet another specialized processor. The zIIP is designed to help improve resource optimization for running database workloads in z/OS.

Processors such as zAAP and zIIP lower the cost of the platform, making it more cost effective and reducing software costs.

Managing the System z platform to meet business goals

When new workloads are added to a System z server, they are not simply added randomly. Usually a workload is distinguished by its importance to the business. Some workloads, such as those associated with customer ordering and fulfillment, tend to have a higher degree of importance than applications used internally. Making resources available to mission-critical applications when they need them is a priority for System z hardware and software designers.

System z servers running a single z/OS image or z/OS images in Parallel Sysplex can take advantage of the Workload Manager (WLM) function. The overall mission of these advanced workload management technologies is to use established policy and business priorities to direct resources to key applications when needed. These policies are set by the user based on the needs of the individual business. These time-tested workload management features provide the System z environment with the capability to effectively operate at average utilization levels exceeding 70% and sustained peak utilization levels of 100% without degradation to high-priority workloads.

Figure 3 shows the effect of CPU sharing on a System z server with multiple and different workloads running concurrently. In an environment not constrained for CPU, the response time for each application is not affected by the other applications running at the same time.

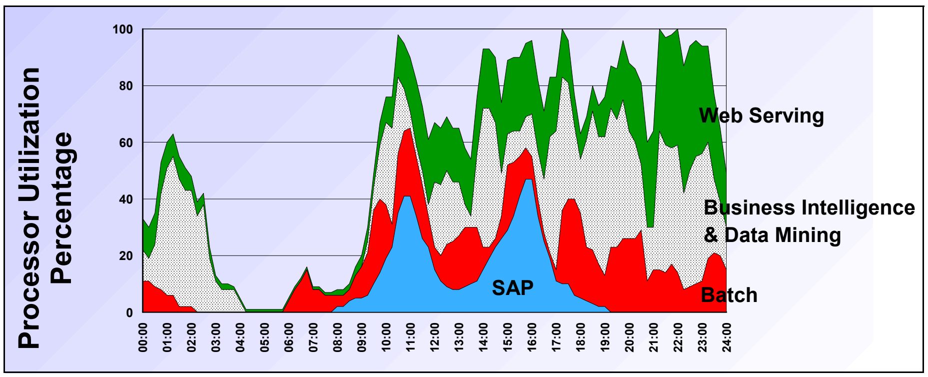


Figure 3 Mixed workloads on the System z platform

The higher degree of workload management represents a key System z advantage. Workload management can start at the virtual server level and drill down to the transaction level, enabling the business decide which transaction belonging to which customer has a higher priority over others.

The Intelligent Resource Director (IRD) is a technology that extends the WLM concept to virtual servers residing on a System z server. IRD, a combination of System z hardware and z/OS technology that is tightly integrated with WLM, is designed to dynamically move server resources to the systems that are processing the highest priority work.

Important: All of these capabilities can be observed by running the Mettle Test. The Mettle test is a simulation of a real z/OS production environment running WebSphere Application Server and traditional workloads with high transaction volume and high-availability configuration. Contact your IBM representative for details.

Security

For a business to remain flexible and responsive, it must be able to give access to its systems to existing customers and suppliers as well as to new customers, while still requiring the proper authorization to access e-commerce systems and data. The business must provide access to the data that is required for the business transaction, but also be able to secure other data from unauthorized access. The business needs to prevent rogue data from being replicated throughout the system and to protect the data of the trusted partners. In summary, the business must be open and secure at the same time.

The System z environment, as with its previous mainframe models, has the security concept deeply designed in the operating system. The ability to run multiple applications concurrently on the same server demands isolating and protecting each application environment. The system has to be able to control access, allowing users to get to only the applications and data that they need, not to those that they are not authorized to use.

Hardware components, such as those for the cryptographic function implemented on each central processor, deliver support to the System z platform for encryption and decryption of data, and for scaling up the security throughput of the system.

In addition, other security components such as Resource Access Control Facility (RACF®) provide centralized security functions such as user identification and authentication, access control to specific resources, and the auditing functions that can help provide protection and meet the business security objectives.

Virtualization

The ability to share everything is based on one of the major concepts of the System z mainframe: virtualization. As it is commonly used in computing systems, virtualization refers to the technique of hiding the physical characteristics of the computing resources from users of those resources. For example, each operating system in a logical partition of the System z platform thinks it has exclusive access to a real hardware image. The actual resources are shared with other logical partitions. The virtualization concept is not an add-on to the platform. z/Architecture® contains many facilities in both the hardware and software that facilitate resource virtualization. All elements of System z enable the virtualization concept.

Virtualizing the System z environment involves creating virtual systems (logical partitions and virtual machines), and assigning virtual resources, such as memory and I/O channels, to them. Resources can be dynamically added or removed from these logical partitions though operator commands. The customer can also use a facility called Capacity on Demand, which allows new hardware resources to be added to the z9 non-disruptively.

Dynamic logical partitioning and virtual machines increase flexibility, enabling selected system resources such as processors, memory, and I/O components to be added and deleted from partitions while they are actively in use. The ability to reconfigure logical partitions dynamically enables system administrators to dynamically move resources from one partition to another.

Processor Resource/Systems Manager™ (PR/SM) is a hypervisor integrated with all System z elements that maps physical resources into virtual resources so that many logical partitions can share the physical resources.

PR/SM and logical partitions

Processor Resource/Systems Manager is a feature of System z9 and previous IBM mainframes that enables logical partitioning of the central processor complex (CEC). This feature can be used to aggregate physical system resources into shared pools from which virtual systems receive virtual resources. It enables the user to consolidate workloads and operating environments currently running on separate processors into one physical system while using resource sharing to improve resource utilization and maintain performance.

PR/SM provides the logical partitioning function of the central processor complex (CPC). It provides isolation between partitions, which enables installations to separate users into distinct processing images, or to restrict access to certain workloads where different security clearances are required.

Each logical partition operates as an independent server running its own operating environment. On the latest System z models, you can define up to 60 logical partitions running z/VM, z/OS, Linux on IBM System z, z/TPF, and more operating systems. PR/SM enables each logical partition to have dedicated or shared processors and I/O channels, and dedicated memory (which you can dynamically reconfigure as needed).

In other words, PR/SM transforms physical resources into virtual resources so that several logical partitions can share the same physical resources.

Memory

Another concept that is based on multiprogramming is the *multithreading* or, as it is called in the mainframe world, *multitasking*. Multitasking is done explicitly by the computer programmer. The program is structured and written to be divided into different tasks that can run in parallel on different processors.

Many users running many separate programs means that, along with large amounts of complex hardware, z/OS needs large amounts of memory to ensure suitable system performance. Large companies run sophisticated business applications that access large databases and industry-strength middleware products. Such applications require the operating system to protect privacy among users, as well as enable the sharing of databases and software services.

Thus, multitasking and the need for a large amount of memory mean that z/OS must provide functions beyond simple, single-user applications. The sections that follow explain, in a general way, the attributes that enable z/OS to manage complex computer configurations.

In System z terminology, the concept of memory is called *storage*. The System z platform uses two types of memory (storage):

- *Real storage* (also called *central storage*), located on the System z hardware itself. Instructions accessing real storage will appear to have accessed the memory synchronously with the processor. That is, certain instructions must be delayed while data is retrieved from central memory. Both data and programs must be loaded into central storage (from input devices) before they can be processed by the CP.
- *Auxiliary storage*, (also called *external storage*) located external to the System z hardware. Auxiliary storage is accessed asynchronously, through an I/O request, where one or more pages of memory are written or read. During an I/O request, the processor is free to execute other, unrelated work.

Inside the System z hardware we can find the physical memory. This is not the real memory, but has a major role in creating it. Current System z models can have up to 512 GB of physical memory. The amount of physical memory can be increased concurrently. Real memory is built over the physical memory. The System z9 provides physical memory in four identically sized storage cards. The real memory assigned to a logical partition is assigned from all four cards, so that the memory access can be “interleaved” or striped over the four cards. This decreases the average time to fetch memory from the real storage, because four fetch operations occur simultaneously. Figure 4 describes this concept.

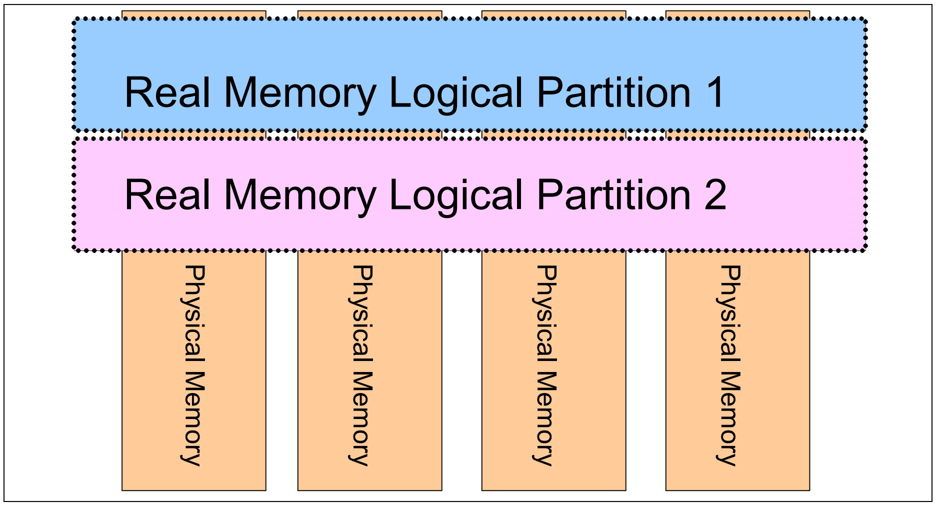


Figure 4 Real and physical memory assignment

Note: In the past, when memory was expensive, another type of memory was built on top of physical memory, called expanded storage. This memory was built by using physical memory that was slower and cheaper than the physical memory used for the central memory. Expanded storage was addressable only at page level, so instructions could not be executed from expanded storage. z/OS no longer uses expanded storage. The z/VM operating system uses expanded memory as a high-speed paging device. Expanded storage is now implemented using the same physical memory as the central memory.

Virtual memory

The System z environment uses both types of memory (real and auxiliary) to enable another kind of memory called *virtual storage.* In z/OS, each user has access to virtual memory storage, rather than real memory.

The virtual memory is a combination of a portion of the real memory and a portion of the external memory. The data sets on the external memory that are part of the virtual storage are called page data sets. Figure 5 shows the concept of System z virtual memory.

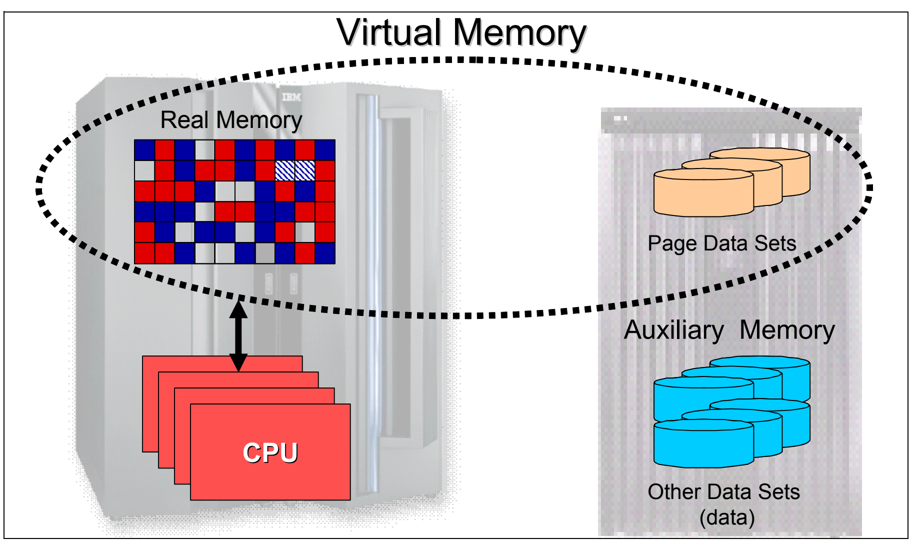


Figure 5 Virtual memory in System z

Currently, the architecture supports 64-bit long addresses, which enables a program to address up to 16 exabytes. In reality, the mainframe will have much less central memory installed. How much less depends on the model of the computer and the system configuration. To enable each user to act as though this much memory really exists in the computer system, z/OS keeps only the active block of data of each program in real memory. *Working set* is the technical term to indicate the pages of virtual memory that stay active during a time interval and are kept in real memory/central storage.

The pieces of a program executing in virtual memory must be moved between real and auxiliary memory. To allow this, z/OS manages memory in units, or page, of 4 KB. The following terms are defined:

- A block of central storage is a *frame*.
- A block of virtual memory is a *page*.
- A block of auxiliary storage is a *slot*.

An active virtual storage page resides in a central storage frame. A virtual memory page that becomes inactive is written out to an auxiliary memory slot (in a paging data set).

Memory addressing

For purposes of addressing the System z memory, three basic types of addresses are used: *absolute*, *real*, and *virtual*. The addresses are distinguished on the basis of the transformations that are applied to the address during a memory access. Address translation converts virtual to real, and prefixing converts real to absolute.

Dynamic address translation is the process of translating a virtual address during a memory reference into the corresponding real address.

As a consequence, the System z operating systems use a series of tables and indexes to relate locations on auxiliary memory to locations in central memory, and special settings (bit settings) to keep track of the identity and authority of each user or program. These tables are defined in the *z/Architecture Principles of Operation,* A22-7832-04.

z/VM

z/VM is key to the software side of virtualization on the mainframe. The z/VM hypervisor is designed to help clients extend the business value of mainframe technology across the enterprise by integrating applications and data while providing exceptional levels of availability, security, and operational ease. z/VM virtualization technology is designed to allow the capability for clients to run hundreds to thousands of Linux servers on a single mainframe running with other System z operating systems, such as z/OS, or as a large-scale Linux-only enterprise server solution.

z/VM offers an ideal platform for consolidating select UNIX and Linux workloads on a single System z server. Running Linux as a guest of z/VM is designed to provide the capability of running hundreds to thousands of Linux images while benefiting from the reliability, availability, scalability, security and serviceability characteristics of System z servers. At the same time, it allows customers to exploit the exceptional capabilities of z/VM virtualization.

Consolidation

Consolidation is the concept of combining many systems onto one system. The cost benefits can be great. However, consolidation is much more than simply replacing a lot of small servers with fewer, bigger, more powerful servers. Consolidation is about finding ways to align and manage your existing IT infrastructure to better support the business model, while establishing a flexible foundation designed to handle future requirements.

The goal is to optimize and simplify your IT infrastructure from end to end, not just the servers, but the storage, data, applications, networks, resources, and system management tools that help bring the entire infrastructure together. In addition to offering potential cost savings and improvements in efficiency, availability, and productivity, consolidation can provide a stable foundation for the rapid deployment of new initiatives as business needs continue to change.

Typically, in a UNIX, Linux, or Microsoft® Windows® environment, workloads are physically partitioned across different physical servers. The distributed servers normally are sized with big white spaces to handle workload spikes. The servers generally run under very low utilization (Windows 5%-10%, UNIX 25%-30%). These workloads are the candidates for consolidation.

Consolidation benefits

Consolidation provides the following major benefits:

- Lower TCO

  Consolidation can help reduce complexity, establish better system management practices, optimize resource utilization, and control spiraling hardware and software licensing fees.

- Improved service levels of applications

  Consolidation can help you enable the applications that drive the integrated enterprise to deliver increased data access, higher levels of availability, and faster response times to users.

- Increased security and operational resiliency

  Consolidation can help manage the security of key information. Leveraging the industry-leading System z capabilities, you can provide a security-rich, agile, near-24x7 environment that is highly resistant to unplanned interruptions and can mitigate risk, transparently adapt to change, and scale to upswings and downswings in the market, as well as being capable of rapid recovery.

- Information as a strategic business tool

  The distributed computing model often creates islands of applications and data. Consolidating IT resources can help ensure that critical business data, and processes are accessible and shared across the enterprise.

Different types of consolidation

There are several different types of consolidation, which can be distinguished from different perspectives:

- Site consolidation

  Site consolidation means moving servers from different locations to one common location. Because centralization simplifies access for IT staff, it helps to reduce operations support costs, improve security, and ensure uniform systems management.

- Application consolidation (n-to-1)

  Application consolidation means merging applications that are running in several servers into one server with one operating system image. Reducing the number of systems and application installations from many to one can significantly lower the expense and effort of administration and maintenance cost and improve disaster recovery. This consolidation is possible if you have an operating system that can scale up and is one of the reasons why z/OS provides value. In addition, depending on the software licensing model, application consolidation can also save software license fees. However, application consolidation can be time consuming and require application architecture and code changes and migrations. At times application consolidation might be very difficult or not even feasible for various reasons. The complexity of application integration could offset the cost savings mentioned before.

- Server consolidation (*n*-to-n)

  Server consolidation consolidates the distributed servers into the same number of operating system images on one box by using virtualization technologies. The virtualization technologies allow sharing of hardware between multiple operating systems, leading to better utilization of hardware resource. Even when the number of operating system image installed is still the same, the administration and maintenance cost can still be reduced. Deployment of new servers will be faster because the new server is just a logical definition and can share the available hardware resources.

- Data consolidation

  Data consolidation is the process of merging data from different sources into a single repository and a common format and, at the same time, consolidating the data storage. When all corporate data resides on the same robust system, the efficiencies can deliver immediate payback to end users. Data sharing throughout the enterprise is vastly simplified. Consolidation enables high levels of security and data integrity.

  Data consolidation is not an easy task. It requires an enterprise data architecture for the actual data consolidation. The application architects have to adjust the application architectures to accommodate the new data access layer.

This consolidation overview does not include a Quality of Service (QoS) discussion for consolidation. However, an IT architect must fully understand the QoS implication of a consolidation solution before making any consolidation-related architecture decisions.

Why consolidate on System z

System z has been the main platform for enterprise data processing operations for many years. The unique strengths of the System z platform (its ability to run multiple workload types with exceptionally high reliability, throughput, security and manageability) have been imitated but never matched by other server platforms.

Figure 6 shows the basic vision of System z consolidation, that is a data center in a box, not a server farm.

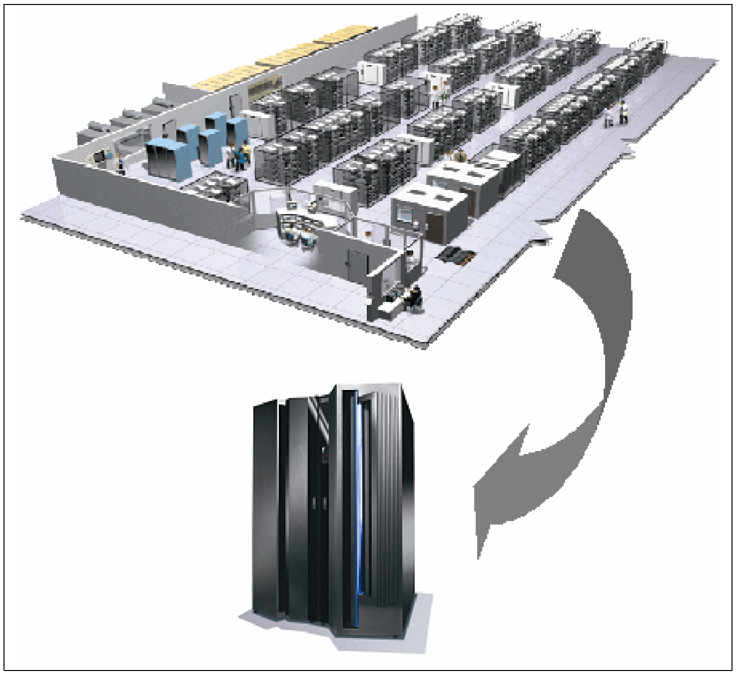


Figure 6 Data center in a box

The following strengths make the System z platform ideal for consolidation:

- z/Architecture is designed for handling heterogeneous workloads. The Workload Manager (WLM) and the IRD manage the mixed workloads and computing resources to help applications that share everything.
- The System z platform has advanced virtualization technologies.
- The System z platform has a strong history in data serving. Its architecture is designed for massive data access, whether across the Internet, to storage devices, or to remote backup sites. The specialty engine zIIP makes data integration more attractive.
- Building on top of WebSphere Application Server for z/OS, the System z family includes a large stack of middleware products to support new workloads for on demand business. The zAAP specialty engine makes running Java workload on z/OS cost effective. These new System z workload capabilities make application consolidations feasible.
- The System z environment is designed to deliver the system integrity and security-rich solutions needed to help meet today's on demand operating environment security requirements.
- The System z platform provides unmatchable QoS. Consolidating to System z allows the applications to leverage its exceptional reliability, high availability, scalability, serviceability, and resiliency (see "Resiliency" on page 18).
- Sophisticated system monitoring and management mechanisms have been available on System z for years.

However, this does not mean that you should consolidate everything to the System z platform, because this platform might not be the right platform for some workloads. The architecture decision is always a trade-off between the benefits and constraints.

Choosing the right consolidation approach

For the most part, consolidation enables administrators to gain a single point of control while maintaining flexibility and the ability to prioritize service levels for end users. However, one of the most important things to remember about consolidation is that there is no off-the-shelf solution or one-size-fits-all answer for IT optimization.

In general, z/OS is more suitable for application consolidation, especially for consolidating a smaller number of applications with high transaction volume or having high QoS requirements. Linux on System z is capable of consolidating a large number of underutilized distributed servers. Each approach leverages classic System z strengths in a different way.

Resiliency

Resiliency is considered one of the most important value propositions of the IT platform, and business strategists usually consider System z as the most resilient platform. This section discusses the cost of downtime and the need for resiliency, as well as the core strengths that make the System z platform resilient.

Built for business

The cost of downtime is high, forcing businesses to strive to be resilient to outages. A business must recover quickly from all interruptions, whether planned or unplanned. Almost nothing is worse than being hit by a system outage, even if the duration of the outage is only minutes long. Consider how an unexpected outage, caused perhaps by a natural disaster or a minor software glitch, can put customer relationships, productivity, and possibly billions of dollars at stake if vital data and applications go unprotected.

Business resilience, which can be considered the ability of an enterprise to continue to function effectively in the face of natural and man-made problems and disasters affecting its IT, has been one of three most important value propositions of IT itself, along with competitive advantage and cost savings. Business strategists usually consider the System z environment to be the most resilient platform, whose resiliency can be counted on but must be integrated with other systems to achieve a business' resiliency objectives. The System z platform, with its strengths in availability and reliability, is an ideal platform from which to build a resilient infrastructure and from which to test new resiliency technologies and strategies.

Some computer platforms were designed originally for academic or other purposes, but the System z environment was created as a solution for business. Over the years, resiliency technologies have evolved and become deeply embedded into System z design. Today, the System z platform continues to innovate and enhance its resilient technologies. Resiliency is imperative to a business' bottom line, and System z is the premier platform in ensuring business systems stay up through disaster recovery, repair and upgrade, and also software and application changes.

A solution for continuous business operation

Downtime is costly. The cost of downtime is so great that many of today's enterprises can no longer afford planned or unplanned outages. The statistics are staggering as businesses can potentially face losses from tens of thousands of dollars to multiple millions of dollars per hour of downtime. Even beyond the financial aspects, downtime can also affect key areas of customer loyalty, market competitiveness, and regulatory compliance. For example, if a Web site is not responsive, Internet-savvy customers will go elsewhere. Continuous operation is a business imperative, and it requires having state-of-the-art resilient technology.

Some businesses must shut down their systems to make scheduled updates, perform maintenance, or upgrade their servers. Built-in features on the System z platform, however, enable businesses to avoid those scheduled outages. In addition to being an excellent platform solution for eliminating planned outages, System z is also an excellent platform for avoiding unscheduled outages. With technologies such as Parallel Sysplex and GDPS, System z can position a data center for world-class disaster recovery.

World-class availability and reliability

Every minute, the pace of the business world increases and the volume of data grows. Businesses need to keep operations running 24x7 and overcome issues such as higher network traffic, unpredictable workload spikes, and unwieldy databases. System z hardware, firmware, middleware, and operating systems are designed and tightly integrated to provide an application environment with world-class levels of high availability. System z provides many of the best-of-breed technologies to ensure availability and reliability.

System availability can be measured in different contexts. One common measurement is hardware availability against unplanned outages. A more stringent measurement of availability examines application availability at the user level, including planned outages. Measuring availability from the user perspective requires consideration of the hardware, OS, middleware, and applications. System z holds itself to the strictest measurements and can achieve world-class application availability at the user level.

Core hardware strengths

System z, built for recovery and continuous operation, offers highly available servers. The result of the focus on resiliency is a design point called Mean Time Between Failure (MTBF). The System z product line is designed to offer layer upon layer of fault tolerance and error-checking features. If a failure occurs, the redundancy that is built into the platform shifts the work from failing components to ones that work to prevent the application and user service from being interrupted. In addition, failed components can be removed and replaced while applications are active and continue to run, eliminating downtime.

Errors have many possible causes. There are transient errors, those due to environmental conditions such as noise or cosmic particles, and more permanent errors such as hardware failures. System z products are designed to handle these various errors. Error detection and correction is built into the logic to detect errors in memory and correct them. Similar technologies help find problems at the earliest possible moment and enable the System z environment to take the necessary actions to correct them, thereby helping to minimize the impact they might have on applications. The System z strategy also focuses on a recovery design to mask errors and make them transparent to customer operations. Extensive recovery is built into the hardware.

Over the years, more and more features have been added that improve the resiliency of the System z platform. The most recent version, System z9 hardware, has a number of unique new features that highlight the newest innovations in resiliency technology.

Redundancy

Redundancy is perhaps the most prevalent mechanism to ensure resiliency, and is a key theme throughout the System z platform. System z designers have worked through the years to eliminate any single point of failure. The platform has many redundant components. For example, there are two power feeds that enable the system to survive the loss of one. There are two Modular Refrigeration Units (MRUs). The service subsystem, internal battery, oscillator, and all of the processors are redundant. Consider a fully loaded System z9 mainframe that has 54 central processors, eight SAPs, two spare CPs, and 336 PowerPC®

engines for the FICON® channels, which would total 400 processors. After taking into account redundancy, that same system would have 800 processors.

Eliminate planned outages with concurrent hardware changes

In the past, a business often had to take a system down on a Saturday night to do hardware change and maintenance. Today, the System z platform eliminates such planned outages with its capability to concurrently change hardware without affecting the application. The concurrent hardware maintenance capability helps System z to provide service to its users at any time, 24x7, without scheduled outages for maintenance of system and data.

In today's business world, backups and system maintenance, such as upgrades or replacements, must be done without interrupting operations. Many System z components can be maintained and upgraded concurrently, including processor books, memory, cryptographic cards, and coupling links. Even non-redundant components such as channel adapter cards are hot-pluggable and can be replaced without any application impact.

Capacity on Demand

The resilient technologies of System z make it well suited for various Capacity on Demand solutions, offering companies the flexibility to rapidly increase or decrease computing capability as requirements change. To achieve near-continuous availability, System z Capacity on Demand offerings tap into the additional capacity. The flexibility of upgrading processing power concurrently enables a business computing infrastructure to better manage risk of volatile, high-growth, and high-volume applications.

System z Capacity on Demand offerings include:

- Capacity Upgrade on Demand (CUoD): CUoD enables a customer upgrade capacity for nondisruptive growth. With CUoD, a customer can upgrade general CPs, specialty engines, memory, and I/O. The extra capacity can be brought online without an outage.
- Customer Initiated Upgrade (CIU): CIU provides the customer with the ability to upgrade the number of general CPs, specialty engines, and memory.
- On/Off Capacity on Demand (On/Off CoD): On/Off COD provides the customer with the ability to temporarily upgrade capacity for fluctuating workloads. One potential use for this offering would be to handle peak loads. The customer may temporarily upgrade the number of general CPs and specialty engines.
- Capacity BackUp (CBU): CBU is a temporary capacity upgrade for customers who require a robust disaster-recovery solution. With this solution the customer can maintain a hot backup site with limited capacity. In an emergency the extra CBU capacity can be brought online without an outage, bringing the backup site up to a capacity that is capable of taking over the whole workload from the failed production site.

Beyond hardware

Resiliency in the System z platform goes far beyond the hardware. The System z resiliency design point focuses on the applications, which results in an integrated environment where hardware, firmware, operating systems, and middleware work together to provide application and data availability.

z/OS, the platform's primary operating system, is built around a philosophy of reliability that recognizes that there will be software errors. Every system service routine has a recovery routine associated with it. In the case of an abnormal end to a program, the recovery routine gets control to repair the data structures used by the failing component and ensure that subsequent calls to this service will not fail again. Based on lines of code, roughly one-third of the z/OS code is associated with the goal of providing reliability, availability, and scalability.

In addition to the primary z/OS system, the z/VM operating system also has many built-in resiliency features. Those familiar with Linux on System z probably have heard about the user who started, as a proof-of-concept exercise, thousands of instances of Linux in thousands of virtual machines on a predecessor of System z9. Although the number of Linux instances is impressive, what is often overlooked is the reliability demonstrated by VM. When VM ran out of resources, it did not crash.

For even more stringent requirements, System z can provide even higher levels of resiliency with its Parallel Sysplex and GDPS architectures. These architectures extend the platform's resiliency technologies to multiple servers and multiple sites. A properly designed Parallel Sysplex with GDPS should be able to provide 100% application availability.

z/OS

The z/OS operating system includes a comprehensive set of capabilities that provides its unique qualities of service in combination with System z hardware. With z/OS, most outage events are completely masked from applications, and in severe cases result in graceful degradation rather than complete failure. z/OS has unmatched availability, reliability, scalability, flexibility, and integrity. Therefore, it has been long trusted for database and transaction workloads, and is also a top choice for Web applications that require the highest quality of service.

The z/OS design philosophy dictates a comprehensive approach of error isolation, identification, and recovery. All z/OS components and subsystems provide recovery routines. Furthermore, all z/OS code is written to a set of reliability, availability, and serviceability guidelines. By design, if an application fails, workload can be picked up by a backup of the same application that is running on the same physical hardware. In addition, z/OS provides a feature that allows for automatic recovery and restart of services and transactions based on predefined policies.

z/OS system software also has a total commitment to system integrity. Using techniques such as storage keys and multiple address spaces, data and system functions are protected from unauthorized access. With a system-wide two-phase commit, client application data is recoverable even when hardware and software failures occur. Some features are also designed to directly support key middleware such as CICS, which enable higher levels of availability for transaction execution.

Elimination of planned outages

In addition to the many other methods that are in place to avoid unplanned downtime, System z provides methods to address planned firmware driver updates and reconfigurations.

Customers who run mission-critical applications, usually those without a sysplex environment, have found it difficult to find time to add new functions. On System z, a feature referred to as Enhanced Driver Maintenance (or Concurrent Driver Upgrade) enables a customer to upgrade firmware to the next driver level without having an impact on workload. Enhanced Driver Maintenance allows upgrades of the Licensed Internal Code for CPs, IFLs, ICFs, zAAPs, memory, and I/O adapters that are transparent to the application.

It is also possible to make I/O configuration changes without having a scheduled outage. This reconfiguration capability is called *Dynamic I/O Reconfiguration*, and allows adding, deleting, or modifying the definitions of channel paths, control units, and I/O devices to the software and hardware configurations.

Parallel Sysplex

Parallel Sysplex builds on the strengths of System z to provide even greater availability and even more flexibility to address planned and unplanned outages. Parallel Sysplex is not a single product, but a collection of cooperating z/OS systems. Parallel Sysplex's unique clustering approach makes the multi-server scale-out topology look and behave like a massively parallel SMP (Symmetric Multiprocessing) configuration.

The Parallel Sysplex architecture allows clustered System z servers to provide resource sharing, workload balancing, and data-sharing capabilities for on demand data centers, delivering ultimate flexibility when supporting different application topologies.

You could think of a Parallel Sysplex as a symphony orchestra, with each kind of instrument representing a different product in the sysplex. There are several of each instrument, just as there would be several images of the same product in a sysplex. All violins, for example, sound basically the same and play the same part. There are enough violinists so that if one is out sick, it will likely not be noticed. And if that violinist quits for good, he can be replaced. Similarly in a sysplex, all systems, or a subset of them, can be made to look alike and do the same work. A sysplex exhibits analogous availability characteristics: A failure or planned removal of one system from a sysplex would not result in the loss of application or data availability, only temporary loss of capacity.

Another advantage of Parallel Sysplex technology is the ability to nondisruptively install or maintain hardware and software. Servers can be removed or added while applications continue to run on the other systems. Furthermore, different systems can be running at different software levels, thereby making software upgrades easier. Software can be upgraded on one system while the rest of the systems remain available. Major software changes can be made using the "rolling IPL" technique to stage the software change throughout the sysplex without ever interrupting the application availability.

With Parallel Sysplex clustering and its ability to support data sharing across servers, IT architects can design and develop applications that have one integrated view of a shared data store, effectively eliminating the need to partition databases. Data sharing with Parallel Sysplex has the unique advantage of allowing nondisruptive database growth with automatic load re-balancing. Without a partitioned database environment, application and database growth would likely require lengthy and disruptive database re-partitioning and mean downtime for the application. Parallel Sysplex data sharing capabilities help avoid the availability obstacles encountered with partitioned database architectures.

Parallel Sysplex technology is exploited by System z middleware to provide continuous operation for the applications. Current exploiters from IBM include: CICS Transaction Sever, DB2 for z/OS, WebSphere MQ, WebSphere Application Sever, IMS Transaction Manager, TCP/IP, VTAM®, APPC/MVS, and RACF.

System-Managed Coupling Facility Structure Duplexing

One self-healing attribute of a Parallel Sysplex is System-Managed Coupling Facility (CF) Structure Duplexing. System-Managed CF Structure Duplexing is designed to provide a hardware-assisted mechanism for duplexing CF structure data, without requiring any involvement with the application or database subsystem. The redundancy of duplexing results in faster recovery because data is already in a second CF when a failure occurs. This feature can provide a robust recovery mechanism for failures such as loss of a single structure or CF, or loss of connectivity to a single CF, through rapid failover to the other structure instance of the duplex pair.

The application and the database manager are not aware that the structure is being duplexed or whether one of the duplex copies has failed or recovered.

GDPS

Picture the average computer room immediately following a basic system failure. Phones are ringing, managers are moving to determine when everything will be recovered, operators are scrambling for procedures, and systems programmers are vying with operators for control of the consoles. Now imagine instead a scenario where the only manual intervention is to confirm whether to proceed with a well-tested recovery procedure. The smooth recovery scenario requires a business continuity solution such as GDPS.

GPDS, the premier IBM solution for high availability, provides business continuity solutions both for system outages and disaster recovery situations. Based on geographical separation and advanced automation techniques, GDPS is a multi-site application availability solution that provides the capability to manage remote copy configuration and storage subsystems, automate Parallel Sysplex operation tasks, and perform failure recovery from a single point of control. It extends the resource sharing, workload balancing, and continuous availability benefits of a Parallel Sysplex environment. It also significantly enhances the capability of an enterprise to recover from disasters and other failures and to manage planned exception conditions.

GDPS is a solution for environments that need to protect data and minimize operational risks. Two key business requirements that should be analyzed when considering GDPS are:

- *Recovery Time Objective (RTO)*, the time a business can afford to wait for IT services to be resumed after a disaster occurs.
- *Recovery Point Objective (RPO)* represents the amount of data that a business is willing to restore in the event of a disaster. For example, for an RPO of six hours, the objective would be to restore systems back to the state they were in six hours ago.

Both RTO and RPO are important decision factors when designing a GDPS solution. Any resource that cannot be replaced within the RTO should be available in multiple locations, and this incorporates not just buildings and hardware but also employees and data. Customers who desire an RPO of zero, which means zero data loss, will require a synchronous remote copy solution, such as that provided by GPDS/PPRC, one of three different GDPS solutions. The GDPS solutions, which differ based on their mirroring technologies, are:

- GDPS/PPRC

  Based on Peer-to-Peer Remote (PPRC) copy, GDPS/PPRC synchronously mirrors data residing on a set of disk volumes to a secondary site up to 100 km away.

  GDPS/PPRC is capable of the following attributes:

  – Near-continuous availability
  – Near-transparent disaster recovery
  – RTO less than an hour
  – RPO of zero
  – Protects against localized area disasters (limited to 100 km fiber)

- GDPS/XRC

  GDPS/XRC is a combined hardware and z/OS software asynchronous remove copy solution, and is capable of the following attributes:

  – Disaster recovery solution
  – RTO between one to two hours
  – RPO less than one minute
  – Protects against localized and regional disasters
  – Minimal remote copy performance impact

- GDPS/Global Mirror

  GDPS/Global Mirror uses asynchronous PPRC replication technology and is capable of the following attributes:

  - Disaster recovery solution
  - RTO between one and two hours
  - RPO less than one minute
  - Protect against regional disasters
  - Minimal remote copy performance impact
  - Support for z/OS and open data

Exclusive to GDPS in the PPRC environment is HyperSwap™, a function designed to broaden the near-continuous availability attributes of GDPS/PPRC by extending the Parallel Sysplex redundancy to disk subsystems. HyperSwap is designed to swap a large number of devices and do so with minimal impact on application availability.

Additionally, GDPS/PPRC technology can manage a heterogeneous environment of z/OS and Open Systems data. This is important for the common situation when a multi-tier application has dependencies on multiple operating system architectures.

Systems management

The IT environment is more complex than ever and the resources that are required to manage these systems are increasingly hard to find. The increased complexity of hardware and software in modern systems have reached a point where humans cannot reasonably manage them with conventional methods.

Systems management is the enterprise-wide management of computer systems including hardware, operating systems, middleware, applications, and storage subsystems. Systems management consists of the processes, techniques, and tools that are required to manage those systems. System z hardware and software provide a robust set of system-management tools that help IT professionals monitor, manage, and automate their systems.

Systems management, one of the strengths of z/OS, provides a large portion of the value of z/OS. In the mixed workload environment supported by System z and z/OS, system resources are utilized at a high capacity, eliminating wasted resources such as idle CPUs.

The goal of z/OS system management is to make z/OS easier to use, to increase business application availability, and to reduce the skill requirement and cost of managing the system by providing an end-to-end solution.

IBM autonomic computing initiative

Computer systems do not become autonomic overnight. This is a process that evolves over time. Best practices are determined through trial and error. For a computer to be able to be autonomic, the logic of these best practices must be embedded into the software logic. Autonomic computing systems evolve over time as system designers become better at predicting and fixing these problems. Businesses slowly incorporate autonomic principles into their IT infrastructure. *Autonomic maturity* is the continuum of the self-managing properties of a system.

There are five levels of autonomic maturity:

- *Basic*: IT professionals manage each infrastructure element independently.
- *Managed*: System management technologies are used to collect information onto fewer consoles.
- *Predictive*: Analytical processes are introduced into the system to monitor situations that arise and analyze the situations to provide possible courses of action, but the IT professional decides what action to take.
- *Adaptive*: The IT environment can take actions based on available information and a knowledge of what is happening in the environment.
- *Autonomic*: Business policies and objectives govern the IT infrastructure operation. IT professionals monitor business processes, not IT processes, and alter the objectives of the system based on business policy.

There are examples of predictive and adaptive systems on the market today, but most systems remain at the basic or managed level. z/OS is continually building on the autonomic capabilities of the system with the goal of having a fully autonomic system.

A system at the highest level of autonomic maturity will have the ability to manage itself and dynamically adapt to change in accordance with business polices and objectives. IT personnel set these policies and objectives, which the system acts to fulfill.

Today's System z products incorporate a variety of autonomic capabilities based on the four characteristics of self-managing systems:

- *Self-Configuring*: A self-configuring system has the ability to dynamically configure itself in response to changing technological or business conditions. A self-configuring system will adapt to changing IT environments based on policies set by the IT personnel.
- *Self-Healing*: A self-healing system can detect hardware or firmware faults proactively or as they happen and contain the effects of the faults within defined boundaries. This enables the system to recover with minimal or no impact to the execution of the operating system or user-level workloads. A self-healing system can also recover automatically if an abend or other system error occurs.
- *Self-Optimizing*: A self-optimizing system can detect performance problems by monitoring and correct them through tuning the hardware or software resources.
- *Self-Protecting*: A self-protecting system can detect hostile behavior as it happens and take corrective action to prevent harm to the system. A self-protecting system can protect against internal and external threats to the integrity and privacy of applications and data.

z/OS provides system management that either eliminates the need to take management actions, automates as many management activities as possible, or makes those actions that cannot be automated as intuitive and easy as possible.

Self-configuring technologies

z/OS is not fully self-configuring yet, but it does have *dynamic configuration*. meaning that settings can be changed while the system is running. Most of the middleware products across the System z family are self-configuring.

With the System z platform, a data center does not have to upgrade the operating system whenever it adds a new machine. System z hardware will always run the current and three previous releases of z/OS. (Supported middleware is the current and one previous version.) This enables data centers to reduce the complexity in upgrading by choosing to upgrade just the hardware, and upgrade the software and middleware later.

z/OS requires no outages for configuration changes even if there is more than one image of the software running on the system. Self-configuring technologies are not meant to simplify the complexity of z/OS, but to simplify its configuration. z/OS has tools to help with the dynamic configuration of systems.

IBM intends to provide a new user interface for z/OS management that is planned to help the new generation of IT workers by automating, eliminating, and simplifying many z/OS management tasks. A new management console powered by Tivoli® technology is in the works, as well as other enhancements to z/OS to make it fundamentally easier to set up and manage.

The following tools are available to help with configuring z/OS:

- z/OS wizards

  IBM provides Web-based wizards that assist with system customization. These wizards ask business-related questions to help customers decide which products can help them achieve their business goals.

- Hardware configuration

  Hardware Configuration Definition (HCD) is an integrated part of the operating system. It provides an interactive interface for configuring your channel subsystem and operating system. HCD creates the I/O Definition File (IODF). Changes can be made to the current I/O configuration without having to IPL the software or power-on-reset (POR) the hardware.

  The Hardware Configuration Manager (HCM) is a graphical user interface to HCD. HCM runs on a PC and interacts with HCD in a client/server relationship. The host system is required to maintain an internal model of the IODF, but it is easier to maintain the model in a visual form. HCM enables you to navigate easily through configuration diagrams, make changes to your configuration dynamically, and manage the physical aspects of your configuration.

Self-healing technologies

Since its inception, z/OS has incorporated self-healing philosophies and reliability, availability, and serviceability features. Developments in z/OS are consistently trying to improve these features. The following tools help the system heal itself:

- Parallel sysplex automation

  IBM Tivoli System Automation for z/OS is the leading Parallel Sysplex automation product. It provides policy-based self-healing of applications, system, and sysplex resources. For a full description of Tivoli System Automation for z/OS, see “End-to-end enterprise systems management” on page 28.

- IBM Health Checker for z/OS

  IBM provides a framework for a predictive system with IBM Health Checker for z/OS. IBM Health Checker for z/OS helps simplify and automate the identification of potential configuration problems before they affect system availability or cause outages. It compares settings and values on the active system to values and settings suggested by IBM or defined by your installation. IBM Health Checker for z/OS, a continuously running preventive tool, alerts you when critical system settings and values change. On the autonomic maturity continuum, IBM Health Checker provides predictive capabilities for z/OS.

  This framework manages functions such as check registration, messaging, scheduling, command processing, logging, and reporting. This framework is provided as an open architecture in support of check writing. Installations can write their own checks to determine whether their system is configured to meet their business goals.

Self-optimizing technologies

One of the core values of z/OS is its ability to optimize the workloads running on the system to fully utilize the system resources. z/OS has a long history of self-optimization in the area of workload management and virtualization. The virtualization of resources on z/OS means that they can be allocated wherever they are needed, not wherever they are physically connected. Self-optimizing technologies include:

- Workload Manager and Intelligent Resource Director

  Workload Manager provides the ability to handle mixed workloads. z/OS has a very high degree of utilization and can use system resources that would otherwise be wasted in a mixed-workload environment. Workload management on z/OS uses policies set by the installation to determine the allocation of resources. This management is highly automated and includes the scheduling of workloads, resource allocations, and responding to potential and major error.

  The unique workload and self-management capabilities provided by z/OS Workload Manager (WLM) and the Intelligent Resource Director (IRD) enable z/OS to handle unpredictable workloads and meet response goals through effective use of CPU and I/O resources with minimal human intervention for setup and operation, making it the most advanced self-managing system.

  For more information about WLM, see *z/OS V1R8.0 MVS Planning Workload Management,* SA22-7602-12. For more information about IRD, see *z/OS Intelligent Resource Director,* SG24-5952.

- System-managed storage

  Requirements for data and storage continue to grow. Although the cost of physical storage is decreasing, the cost and complexity of managing that storage continues to grow. The Data Facility Storage Management Subsystem (DFSMS) component of z/OS provides a range of automated data and space management functions to help improve disk usage and manage disk and tape storage growth with a centralized point of control with a storage administration policy. DFSMS can help reduce out-of-space failures, fragmentation, poor capacity utilization, and device-type constraints. DFSMS allows the most critical data to be on the fastest devices where and when the application needs it and it moves other data to slower, less-expensive devices. DFSMS also provides system management of tape devices. It can keep track of where the tapes are kept in a manual or robotic tape system.

  DFSMS can automate and simplify data maintenance processes with policies. The processes that can be automated include backup, recovery, archiving, and disaster backup and recovery.

Self-protecting technologies

The isolation provided to applications by z/OS is a key differentiating factor. This isolation helps keep applications running on the same system from taking each other down. z/OS is not vulnerable to buffer overrun attacks due to this isolation.

Intrusion Detection Services is an example of the new technologies that improve on the self-protecting features of z/OS. The security features of z/OS are unparalleled and one of the key features of z/OS self-protection is the Intrusion Detection Services (IDS). IDS enables the detection of attacks and the application of defense mechanisms on the z/OS server. It utilizes policies to determine what to do when different types of attacks occur. Types of events IDS can detect include scans, single packet attacks, and flooding. Actions include packet discard, connection limiting, and reporting.

z/OS includes key security features such as PKI, LDAP, Kerberos, SSL, digital certificates, and tape encryption.

End-to-end enterprise systems management

IBM Tivoli System Automation for z/OS plays an important role in building the end-to-end automation of the IBM autonomic computing initiative: It brings together the four facets of autonomic computing to provide a self-managing infrastructure for z/OS. Tivoli System Automation for z/OS can help customers manage single-process z/OS systems or Parallel Sysplex clusters, reducing the frequency and duration of incidents that have an impact on IT availability.

Tivoli System Automation for z/OS helps customers ease management, minimize costs, and maximize application availability. Tivoli System Automation for z/OS is a high-availability solution for critical business applications, providing self-healing processes and automation for I/O, processor, and system operations. System administrators define the “desired” state for the system, and the software will monitor and launch the appropriate response if the system deviates from the desired state.

Tivoli System Automation for z/OS comes with more than 40 plug-and-play automation policy modules including IMS, CICS, Tivoli Workload Scheduler, DB2, mySAP™, GDPS, and WebSphere. These policy modules are based on best practices, customer experiences and a deep knowledge of each application. Many commercial applications and z/OS components provide out-of-the-box automated message management that requires minimal effort in setup and helps ensure high-quality installation. This powerful message policy supports different automation for different start types or message codes.

Tivoli System Automation for z/OS can be used as a base to take advantage of your autonomic end-to-end solution for systems management.

The value of IBM Tivoli System Automation for z/OS

Tivoli System Automation for z/OS provides the following value:

- Parallel Sysplex application automation

  One advantage of Tivoli System Automation for z/OS is that it makes Parallel Sysplex automation a reality. It manages any number of stand-alone systems and clusters from a single point of control, a Java-based GUI, by allowing for sysplex-wide grouping of resources, dependencies, and goals.

- Policy-based self-healing

  Another advantage of Tivoli System Automation for z/OS is policy-based self-healing. It reduces complexity, implementation time, coding, and support effort through an automation policy that can be shared or cloned across the enterprise. The benefits of this are that policies are built more easily using more cloning variables and more predefined and self-configured message rules. With Tivoli System Automation for z/OS you can monitor the system for signs of an outage condition and relieve those conditions before an outage actually occurs. Tivoli System Automation for z/OS includes performance-related information from IBM Tivoli OMEGAMON® monitors in your automation processing framework. This framework enables you to proactively inform system administrators of performance issues in the system and terminate jobs or move workloads based on the appropriate action defined the policy, before the outage occurs.

- Integration

  Tivoli System Automation for z/OS integrates with Tivoli products including IBM Tivoli Enterprise Console® and IBM Tivoli Business Systems Manager to provide seamless automation of both system and network resources.

- mySAP high-availability automation

  Tivoli System Automation for z/OS provides a high availability management solution for mySAP and its related components. It combines the concepts of automation, high availability, and transparent failover.

Enterprise-wide systems management

IBM Tivoli System Automation for Multiplatforms uses an adapter infrastructure to integrate with Tivoli System Automation for z/OS, enabling you to effectively automate composite applications that span systems. This feature provides a single point of high availability across z/OS, Linux, and IBM AIX® and enables you to display aggregated and detailed status of application components, manage application components on all platforms with a single interface, and increase availability by helping you resolve cross-platform dependencies. Tivoli System Automation for z/OS, in conjunction with Tivoli System Automation for Multiplatforms, enables you to establish a single team for operations and automation of z/OS, Linux, and AIX applications, greatly simplifying problem determination and resolution.

For more information about IBM Tivoli System Automation, see:

http://www.ibm.com/servers/eserver/zseries/software/sa

Summary

The System z platform with its rock-solid reliability, highly advanced scalability, and integration capabilities, as well as its attractive cost efficiencies, can be a major player in today’s IT infrastructure. Business requirements such as flexibility, responsiveness to customer and market needs, and the ability to be cost effective are provided by the System z hardware and software.

The team that wrote this guide

This guide was produced by a team of specialists from around the world working at the International Technical Support Organization (ITSO), Poughkeepsie Center.

Mike Ebbers is an IT Specialist who has worked for IBM for 35 years on mainframe systems.

Thanks to the authors of the IBM Redbooks® publication on which this guide was based (*IBM System z Strengths and Values*, SG24-7333):

Philippe Comte, IBM France
Andrea Corona, IBM Italy
James Guilianelli, IBM Poughkeepsie
Douglas Lin, IBM New York
Werner Meiner, IBM Germany
Michel Plouin, IBM France
Marita Prassolo, IBM Italy
Kristine Seigworth, IBM Poughkeepsie
Eran Yona, Israeli Ministry of Defense
Linfeng Yu, Insurance Services Office, Inc.

Thanks to the following people for their contributions to this project:

Paola Bari, Paolo Bruni, Franck Injey
International Technical Support Organization, Poughkeepsie Center

Paul diMarzio, Rich Guski, Carl Parris
IBM Poughkeepsie

Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:
IBM Director of Licensing, IBM Corporation, North Castle Drive, Armonk, NY 10504-1785 U.S.A.

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs.

This document REDP-4521-00 was created or updated on March 15, 2009.

Trademarks

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both. These and other IBM trademarked terms are marked on their first occurrence in this information with the appropriate symbol (® or ™), indicating US registered or common law trademarks owned by IBM at the time this information was published. Such trademarks may also be registered or common law trademarks in other countries. A current list of IBM trademarks is available on the Web at http://www.ibm.com/legal/copytrade.shtml

The following terms are trademarks of the International Business Machines Corporation in the United States, other countries, or both:

AIX®
CICS®
DB2®
FICON®
GDPS®
Geographically Dispersed Parallel Sysplex™
HiperSockets™
HyperSwap™
IBM®
OMEGAMON®
Parallel Sysplex®
PowerPC®
PR/SM™
Processor Resource/Systems Manager™
RACF®
Redbooks®
Redbooks (logo) ®
Redguide™
System z9®
System z®
Tivoli Enterprise Console®
Tivoli®
VTAM®
WebSphere®
z/Architecture®
z/OS®
z/VM®
z9®

The following terms are trademarks of other companies:

mySAP, and SAP logos are trademarks or registered trademarks of SAP AG in Germany and in several other countries.

Java, and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

Microsoft, Windows, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

Intel, Intel logo, Intel Inside logo, and Intel Centrino logo are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Linux is a trademark of Linus Torvalds in the United States, other countries, or both.

Other company, product, or service names may be trademarks or service marks of others.