Mike Ebbers
Tanya S Buchanan
Andrea Greggo
Dinakaran Joseph
John Langer
Eugene Ong
Mark Wisniewski

Performance Test of Virtual Linux Desktop Cloud Services on System z

Cloud computing is an evolutionary deployment model for IT and business assets. For the user, cloud computing delivers flexible business models by providing consumable and affordable services that are invoked and accessed through self-service Web 2.0 portals. For the provider, cloud computing enables the delivery of IT and business assets that are operationally efficient and highly virtualized. Across industries, cloud computing is providing a fast, simple, and inexpensive means to provision computing resources for a variety of key applications, platforms, and infrastructure services.

Focusing on poorly performing business and technology processes enables companies to begin transforming to cloud computing while targeting near-term return on investment. Managing and provisioning end user desktops is an example of a costly and inefficient business process that can be greatly improved with cloud computing alternatives. Centrally controlling and managing end user desktop environments and making them available as a cloud service is a great way to reduce support costs and increase end user productivity.

This IBM® Redpaper™ document describes the results of testing a virtual Linux® desktop service that was created using *eyeos*, an open source Web desktop, running on an IBM System z® server. This configuration was then tested by creating thousands of virtual clients concurrently accessing the desktop platform on Linux on System z to determine how a shared server environment would respond to concurrent desktop demands.

The results of the desktop platform configuration test validate that a shared all-Linux platform running z/VM® virtualization is an excellent environment for efficiently using CPU, memory, and I/O to manage thousands of concurrent desktop users.

The business need

Customers are encountering significant problems with managing numerous client workstations. In particular, companies with large numbers of mobile employees are seeing significant costs in supporting and maintaining end user desktop environments. Due to the distributed nature of many of these businesses, end users' problematic desktops are often shipped to support facilities to be fixed and then shipped back to employees, leaving these employees unable to work for days at a time. These businesses need a low-cost, flexible solution to handle the business overhead of this activity. Cloud computing solutions fit this need.

Gartner's latest numbers predict over $300 million in virtual desktop hosting revenue worldwide this year. This reflects the trend of businesses to leverage virtual desktop cloud services as a way to become more efficient overall.

What a cloud is

Cloud computing is a general term for complete services delivered over the Internet using self-service end user portals and a flexible business model that incorporates hardware, software, and services into a single revenue stream. Users of cloud services have no visibility to the underlying technologies that enable these services. Cloud services are broadly divided into three categories: Infrastructure-as-a-Service (IaaS), Platform-as-a-Service (PaaS), and Software-as-a-Service (SaaS).

A cloud service has three distinct characteristics that differentiate it from traditional hosting:

► It is sold on demand, typically by the minute or the hour.
► It is elastic, meaning a user can have as much or as little of a service as they want at any given time.
► The service is fully managed by the provider. The consumer needs nothing but a personal computer and Internet access.

Significant innovations in virtualization and distributed computing, as well as improved access to high-speed Internet, and a weak economy, have accelerated interest in cloud computing.

A cloud can be private or public. A *public cloud* sells services to anyone on the Internet. (Currently, Amazon Web Services is the largest public cloud provider.) A *private cloud* is a proprietary network or a data center that supplies hosted services to users behind a firewall. Most large enterprise customers are not willing to assume the risk of public cloud services and are focused on enabling the efficiency of cloud computing in a private cloud. Whether private or public, the goal of cloud computing is to provide easily scalable access to computing resources and IT services.

Many large enterprise customers are considering cloud computing projects and initiatives. It is imperative to focus on poorly performing business processes to gain the largest return on investment; efficiency, and doing more with less, is critical, especially during an economic downturn. Because end user desktop environments are so inefficient and costly, businesses see value in transforming their delivery, but want to do so without significant investment in new hardware, software, and resources.

What eyeos is

eyeos is an IBM PartnerWorld® vendor that uses open source software to create virtual Linux desktop services that run natively on Linux in a LAMP stack. eyeos can be used to create a desktop platform that is suitable to run and manage diverse end user desktops in a central location from a server platform. The application itself is PHP based and runs on Linux platforms. When run on Linux on a System z server, many eyeos desktop images can be supported by one Linux guest. This is in contrast to most desktops, where each desktop client requires a Linux guest in the System z to support it. For many System z customers, eyeos provides a simple, cost-effective way to transform delivery of services to a cloud model, an approach that is both easy to get started with and that scales over time and volume.

There are also a number of server solutions available. System z has strong virtualization and management capabilities. The z/VM hypervisor provides proven technology for serving virtual clients. Our results validate how well System z virtualization scales up and out in this virtual desktop environment. For businesses that plan on implementing cloud solutions in the near future, System z should be considered as an integral part of the solution architecture.

Figure 1 on page 4 shows the components of a sample eyeos environment.

Desktop solutions

There are a number of desktop solutions available. They can be powered by Windows®, Linux, or other products. The desktops in our project were powered by eyeos, interfacing to a System z server.

The objective of this project was to demonstrate that System z can host a large number of workstations, provision these systems quickly, and manage these virtual clients efficiently.
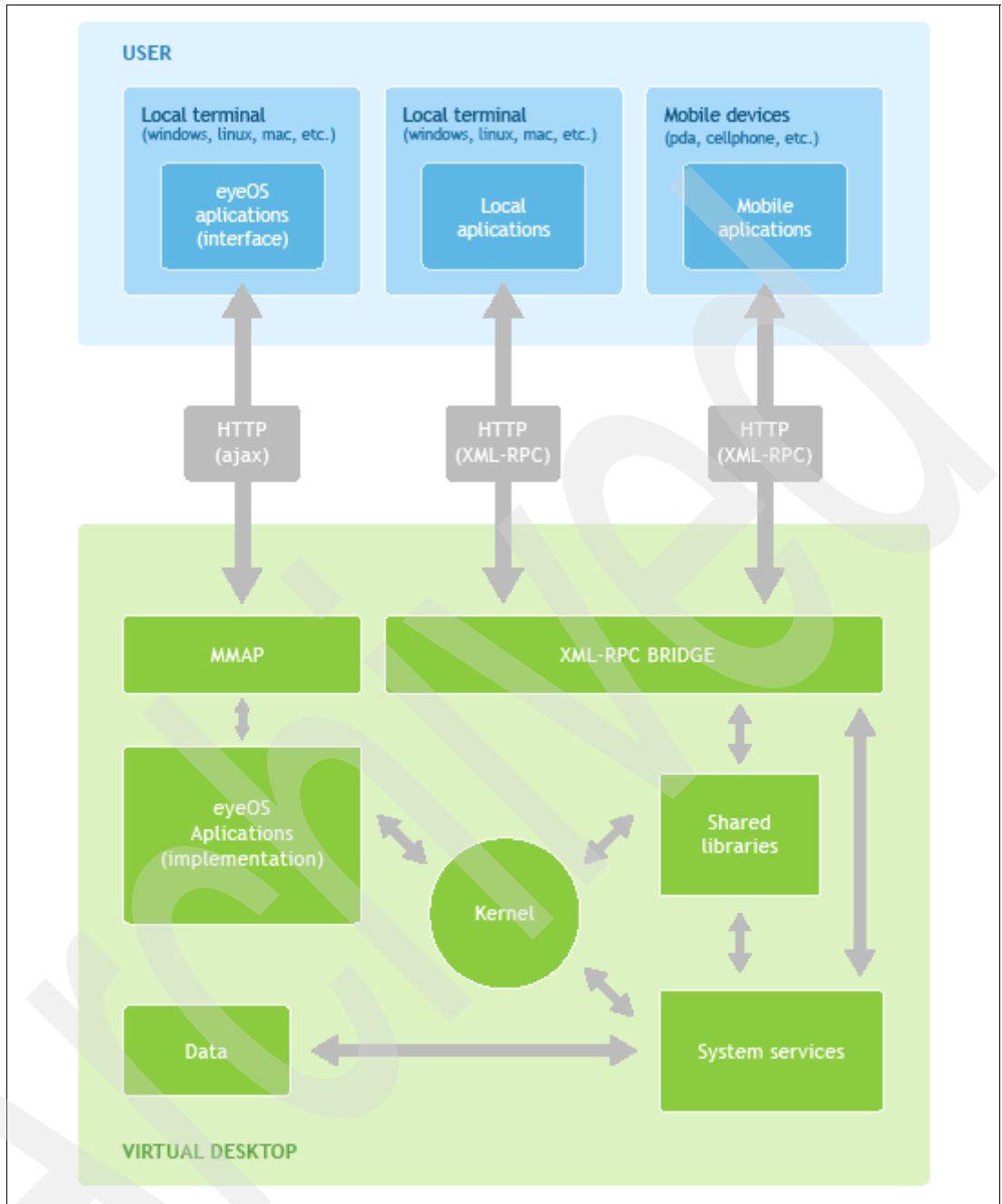
Figure 1   A sample eyeos environment

Our test scenarios

The eyeos benchmark developed in this project simulates clients logging on to a virtual desktop, accessing the Web through the Apache Web server, browsing various Web pages, and doing some page manipulation and editing.

Workload description

The workload simulates users accessing files and browsing a Web page from their virtual workstations running on a Linux guest. The goal was to simulate as many virtual workstations as possible on the Linux LPAR, while still maintaining a reasonable response time.

Rational® Performance Tester was used to drive the simulated users. A 90 second delay was added to the Rational script. This delay was meant to simulate the time the user would spend browsing the Web page just retrieved as part of the workload.

The main measurement scenarios consisted of the following objectives:

► Determine the number of simulated Web-based desktop clients that will drive an 8CP System z10™ to 90+% CPU usage. See "Our project environment" for more details.

► Based on this data, scale up from a 1CP z10 to a 16CP z10 at optimum utilization.

► Based on the 8CP numbers of clients, scale out with multiple z/VM Linux guests.

Scenario overview

We ran scenario tests to see how to maximize the number of Web-based desktops that we could support. We varied the numbers of users (290-4500), CPs (1-16), memory (3-6 GB), and guest machines (1-4). Our scenarios are listed in Tables 1 and 2 and the results are discussed there also. They validate that a shared all-Linux platform running z/VM virtualization is an excellent environment for efficiently using CPU, memory, and I/O.

Our project environment

This section provides details on the hardware and software used in our testing, specifically:

► Server and client hardware used

► Storage server setup

► Server and client software used

► Configuration of the test environment

► Performance monitoring tools

Hardware for server and client workstations

Server
One LPAR on a 64-way IBM System z10 EC, 4.4 GHz. Model 2097-E64, equipped with:

► 16 physical CPUs, dedicated

► 26 GB central storage

► 1 GB expanded storage

► One OSA Express 2 Ethernet card

► 8 FICON® Express channels

Client workstations
► IBM System x® with 2-way 3.6 GHz Xeon Processors, 4 GB RAM, 73GB hard drive(s)

► 12 workstations provided

Storage server setup

- ► IBM DS8000®, Disk Drive Modules, each 146 GB / 15000 RPMS
- ► For the operating system and the eyeos application, a total of 20 ECKD™ mod 9 DASD
- ► DASD was spread over 5 ranks, connected with 8 FICON channels

Software for host and client workstations

This section identifies the software products and version levels we used in out test environment.

Host

z/VM	5.4.0 SLU 0801
SUSE Enterprise Linux Server	SLES-10 SP2, 64 bit
eyeos	Version 1.7

Client workstations

Microsoft® Windows XP	Server Edition SP2
Rational Performance Tester	8.0
Apache	Version 2.0

Configuration

Twelve client workstations were used as the Rational Performance Tester (RPT) workload generator. Client Workstation #1 is known as the Workbench. The Workbench was configured to evenly distribute the load across ten other workstations, that is, client 2 through client 11.

Client 12 was installed with the Apache Web server and was used to serve up a dummy eyeos Web page during test runs.

When deployed, the Workbench would FTP a copy of the test script to each of the client workstations. It also controlled workload balancing as well as performance data gathering.

The eyeos servers and the client workstations were connected to both the 1 Gb Ethernet Admin LAN for administrative purposes and the 1 Gb Performance LAN for test runs. Figure 2 shows the eyeos servers, client workstations and network topology.
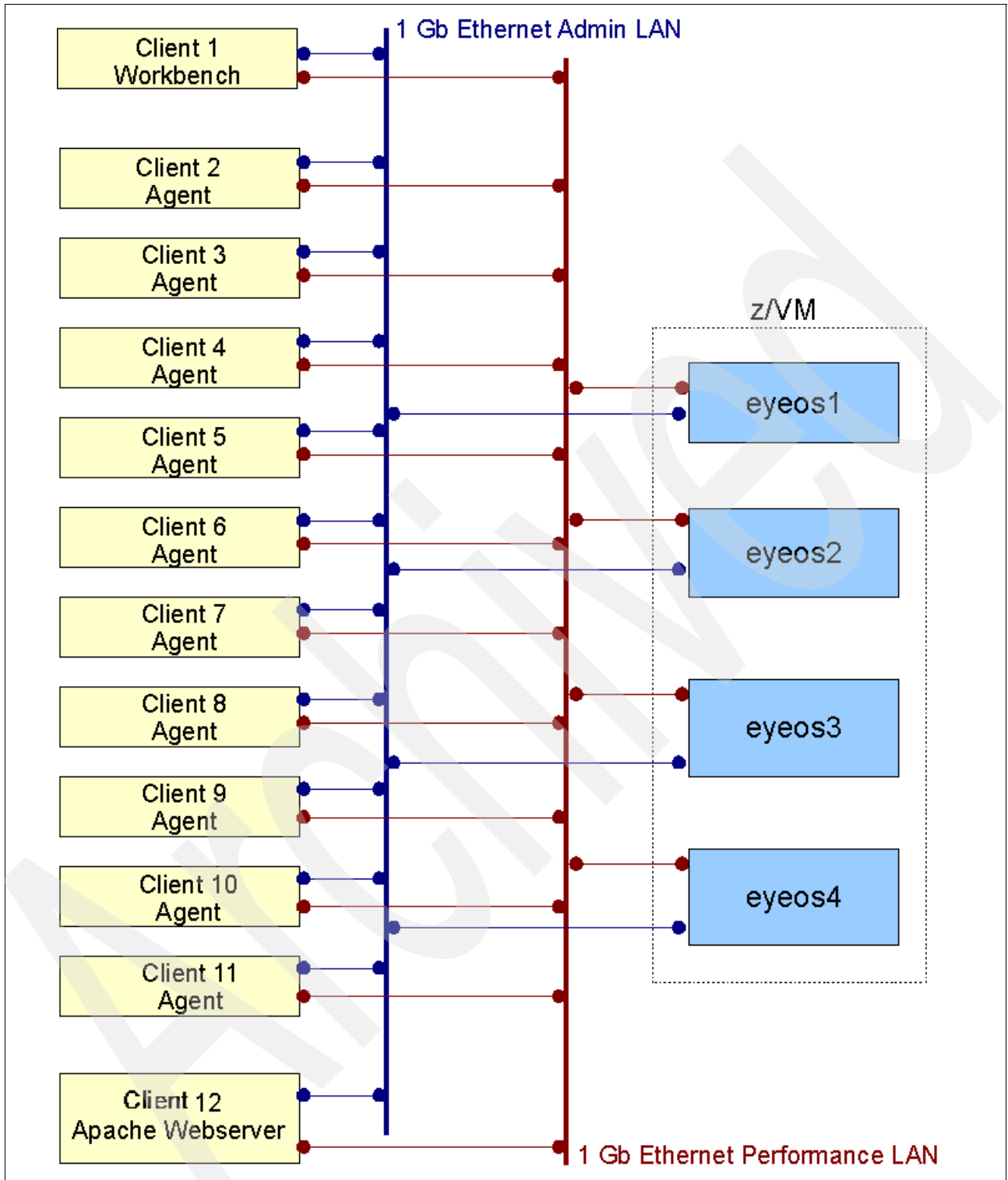
Figure 2   Topology of the system

Tools

The tools we used to conduct our tests an take measurements are described in this section.

sar

The `sar` command in Linux collects reports and saves system activity information. The contents of selected cumulative activity counters in the operating system are written to standard output. The accounting system then uses supplied sar parameters and writes information based on the given parameters.

iostat

The `iostat` command is used for monitoring system input/output device loading by observing the time the devices are active in relation to their average transfer rates. The iostat command generates reports that can be used to change system configuration to better balance the input/output load between physical disks.

Rational Performance Tester (RPT)

Rational Performance Tester 8.0 was used for conducting measurements. Our project involved the use of the following components.

Script

The script logged in to the eyeos application, then created a file and saved it to the desktop. It logged in multiple users who performed a number of specified iterations. It also logged on to a simulated Web site. A 90 second delay was incorporated for examination of this Web site.

Data pool

The data pools consisted of up to 8000 usernames and corresponding passwords for use with the script.

Test schedule

The test schedule changed the following parameters as needed:

► Users

► Think time

► Script

► Number of times script should be executed

► Remote locations for running users

► Web site delay

The benefits of virtualization

We ran Linux servers on our System z in both native mode (one Linux in a dedicated LPAR) and virtual mode (multiple Linux guests sharing system resources). Virtualization has the potential to lower the cost of managing the environment, as well as reducing its complexity and improving its flexibility to quickly respond to business needs. The benefits of virtualization include:

► Higher resource utilization

Running as guests on z/VM, several Linux servers can share the physical resources of the underlying box, resulting in higher levels of sustained resource utilization, comfortable even when approaching 100% for CPU.

► More flexibility

Virtualization can provide Linux guests with great flexibility. Linux servers can dynamically reconfigure resources without having to stop all of the running applications.

► Improved security and guest isolation

Each virtual machine on z/VM can be completely isolated from the control program and the other virtual machines.

► Higher availability and scalability

Linux servers running on z/VM can achieve excellent availability due to the reliability of the underlying mainframes, which have refined hardware and software architectures that have evolved over four decades. Physical resources can be removed, upgraded, or changed without affecting their users. z/VM enables its Linux guests to be easily scaled up or scaled out to accommodate changing workload demands.

► Lower management costs and improved provisioning

Since virtual resources are abstracted from hardware and operating system issues, they are often capable of recovering much more rapidly after a crash situation and running in this virtualized mode greatly facilitates high-speed cloning.

Results

The results show that eyeos has excellent scalability on System z running as a Linux guest under z/VM up through a 16 processor image at optimum utilization. This is partially due to the fact that a single Linux guest container can support thousands of clients, rather than requiring one Linux guest per client.

Tables of metrics for each scenario

Table 1 on page 10 summarizes the results of our tests. The goal was to demonstrate the scalability of the eyeos workload. The column headings in the table have the following meanings:

Number of users The number of users is a count of the simulated Web-based desktop clients running on the Linux LPAR. Each user accesses files and browses a Web page from their virtual workstation.

ETR ETR is the External Transaction Rate. In this case, we used the number of users as the ETR.

ITR					ITR is the Internal Throughput Rate. Internal Throughput Rate shows the transactions per CPU second and is ETR divided by CPU percentage, that is, the ETR normalized to 100% CP Utilization. Another way to think of ITR is the number of units of work accomplished per unit of processor busy time in an unconstrained environment.			

ITR
ITR is the Internal Throughput Rate. Internal Throughput Rate shows the transactions per CPU second and is ETR divided by CPU percentage, that is, the ETR normalized to 100% CP Utilization. Another way to think of ITR is the number of units of work accomplished per unit of processor busy time in an unconstrained environment.

Response time
Response time (ms) is the average time in milliseconds it takes for each user to complete a single transaction. In this case, a transaction is the access of a file and the invocation of the Web browser.

CPU%
CPU% is percentage of time the Central Processing Unit is busy executing program instructions.

Memory config
Memory config - GB is the amount of physical memory, in gigabytes, that is configure for use by the system.

Memory used
Memory used - GB is the amount of physical memory, in gigabytes, that is actually in use by the system.

tps
This metric is transfers per second, as reported by the Linux command `iostat`. tps indicates the number of transfers per second that were issued to the physical disk. A transfer is an I/O request through the device driver level to the physical disk. Multiple logical requests can be combined into a single I/O request to the disk. A transfer is of indeterminate size.

Scenarios
Scenarios is a description of the physical configuration of the LPAR used during the measurement.

Table 1   Results table

Number of users	ETR	ITR	Response time (ms)	CPU%	Memory config - GB	Memory used - GB	tps	Scenarios
2250	2250	2380.9	2129	94.5	4.5	2	181.94	1) 8-way guest Linux
2500	2500	2543.3	5016	98.4	4.5	2.5	228.5	2) 8-way guest Linux
1250	1250	2390	220	52.52	4.5	1.1	84.5	3) 8-way guest Linux
625	625	2446	210	25.55	4.5	.630	37.66	4) 8-way guest Linux
2250	2250	2410.94	940	93.32	4.5	2.3	202.12	5) 8-way native Linux
4500	4500	4885	4206	92.1	10	5.6	323.63	6) 16-way guest Linux
4500	4500	4772	3716	94.3	12	7.2	292.66	7) 16-way guest Linux
562 per guest or 2248 total	2248	2306	3834	97.5	6	1.4 (average of 4 guests)	198.78	8) 4 guests, 2-way each
1125 per guest or 2250 total	2250	2344	2006	96.0	6	2.0 (average of 2 guests)	244.38	9) 2 guests, 4-way each
290	290	345.89	2315	83.84	3	.480	23.43	10) 1-way guest Linux

The information in Table 1 demonstrates the scalability of the workload on the System z platform from several perspectives. Consider the first two scenarios. Our objective here was to determine the maximum number of simulated users that could simultaneously run the workload on a single Linux guest running under an 8-way z/VM. Scenario 2 shows 2500 users accessing files from the cloud, such as Project Management reports, spreadsheets or other work products, and browsing the Web. In this case, the average response time is slightly higher than 5 seconds.

We wanted to test for a lower response time, so scenario 1 shows the results of reducing the number of users by 10%, from 2500 to 2250. This resulted in a 57% improvement in response time (from 5016 milliseconds to 2129 milliseconds). Scenarios 3 and 4 are variations of the same concept. In those scenarios, we reduced the number of users by 50% (1250 users) and 75% (625 users). The result shows a 95% drop in response time, when compared to the 2500 user scenario. This information can be used as input to capacity planning exercises, showing the results of trade-offs made between system load and response time.

Scenario 5 was done running Linux on System z as a native LPAR, as opposed to a virtual Linux guest running under z/VM. We chose the 2250 user load as the comparison point. See "The benefits of virtualization" on page 9.

Scenarios 6 and 7 illustrate the results of vertical scaling for this workload. In this case, we again used the 2250 user scenario as the base. In scenario 6, we doubled the size of the z/VM system to 16 CPs, and doubled the number of users on the Linux guest, from 2250 to 4500. The ITR ratio going from the 8-way to the 16-way is above 2 to 1, which is an excellent result. However, this was achieved with a response time penalty.

In scenario 7, we experimented with the effect of memory on the system. Increasing the memory by 20% in scenario 7 resulted in a response time improvement, with a small decrease in ITR. This is another example of trade-offs that can be made when configuring the system.

Scenarios 8 and 9 illustrate the results of horizontal scaling for this workload. In this case, we again used the 2250 user scenario as the base, and we kept the size of the z/VM LPAR the same, 8 CPs. Instead of running a single guest with 2250 users, we defined smaller Linux guests, 4 smaller guests in scenario 8 and 2 smaller guests in scenario 9, and distributed the users across the guests, 562 users per guest in scenario 8 and 1125 users in scenario 9. These results highlight the flexibility in capacity provisioning this solution provides, as well as the reliability and failover characteristics.

Scenario 10 demonstrates the capacity of a Linux guest running under a z/VM system defined with a single CP. In this case, 290 users were able to run simultaneously on the smaller guest.

Table 2 on page 12 shows the memory tests we ran on an 8-way single Linux guest z/VM LPAR. 2250 users were used because it was the optimal number of users found for an 8-way system that also satisfied our success criteria. All tests utilized the processor optimally.

Table 2   Results from the memory tests

Number of users	ETR	ITR	Response time (ms)	CPU%	Memory configured - GB	Memory used - GB	tps	Scenarios
2250	2250	2360.9	1956	95.3	6	3.2	231.22	8-way memory test
2250	2250	2356	3870	95.5	5	3.4	194.61	8-way memory test
2250	2250	2358	3854	95.4	4	3.6	185.5	8-way memory test
2250	2250	2344	3771	96.0	3.5	3.2	180.66	8-way memory test
2250	2250	2337	5208	96.7	3	3.1	n/a	8-way memory test

This set of tests was designed to illustrate the impact of memory on the workload. We kept the user load constant at 2250 and gradually reduced the amount of memory configured on the system.

Although the CP utilization remained relatively constant, we observed an increase in response time as the amount of configured memory was reduced. This is another example of information that could be used as input to capacity planning exercises, showing the results of trade-offs made between memory and response time.

Observations

- ► The transfers per second (tps) value is taken from the output of the `iostat` command. All the I/O was done against the "/" filesystem on /dev/dasda.
- ► Memory test CPU utilization grew and response times lengthened as the amount of defined memory of the LPAR was reduced from 6 GB to 3 GB.
- ► With a memory definition of 3 GB defined to the LPAR in the memory tests, the Linux system began to swap storage.

Conclusions

System z virtualization technology through the Linux and z/VM products enables this virtual desktop environment to perform and scale to higher numbers of CPs very well at high cpu utilization. System z is well positioned to be a key player in the emerging cloud computing growth markets (including data cloud, developer cloud, and public sector cloud) because it is able to support thousands of concurrent virtual clients.

From the perspective of numbers of supported users, our memory tests show that a z/VM LPAR that is defined with 4 GB of central storage will still yield results within our success criteria. With 4 GB defined, the working set size of the Linux guest was approximately 3.6 GB. This illustrates how efficiently z/VM manages resources between all virtual machines within the LPAR.

The team who wrote this IBM Redpaper

This paper was produced by a team of specialists working at the IBM System z Benchmark Center, Poughkeepsie.

Mike Ebbers is a project leader and Consulting IT Specialist at the ITSO, Poughkeepsie Center. He has worked for IBM since 1974 on mainframe projects, and has been in the ITSO since 1994, writing redbooks and redpapers on various topics.

Tanya S Buchanan is a Performance Tester for System z Performance. She has been with IBM since 2001. She holds a masters degree from Rensselaer Polytechnic Institute. Her areas of expertise include WebSphere® and DB2® with Linux on System z.

Andrea Greggo is the Mainframe Cloud Initiative Leader for IBM System z, Systems & Technology Group, Poughkeepsie, NY.

Dinakaran Joseph is an application and integration middleware (AIM) software Lead Architect in IBM Raleigh.

John Langer is a Project leader for System z Performance Test in IBM Poughkeepsie, NY. He has worked for IBM since 1978. His areas of expertise include Linux and WebSphere on the System z platform.

Eugene Ong is a Certified Consulting IT Specialist working for IBM since 1987. He spent 11 years in large system software design and development of products such as TSO/E, APPC/MVS, APPC/RRS, and System Logger. He joined IBM's System z Benchmark Center (zBMC) in 1998. His areas of expertise include benchmarking and proofs of concept in areas such as Linux for System z, Webserver, DB2, Oracle, and Cloud Computing.

Mark Wisniewski is a Senior Technical Staff Member at IBM Poughkeepsie, NY. He has worked on System z performance issues for the last 28 years.

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

Christopher Williams
IBM Client Benchmark Center, 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-4593-00 was created or updated on February 5, 2010.

IBM®

Send us your comments in one of the following ways:
► Use the online **Contact us** review Redbooks form found at:
  ibm.com/redbooks
► Send your comments in an e-mail to:
  redbooks@us.ibm.com
► Mail your comments to:
  IBM Corporation, International Technical Support Organization
  Dept. HYTD Mail Station P099
  2455 South Road
  Poughkeepsie, NY 12601-5400 U.S.A.

Redpaper™

Trademarks