Contents

Introduction

Current data indicates that database applications running on individual physical servers represent a large consolidation opportunity. Over 50 percent of these database applications run on two-way symmetric multiprocessor (SMP) machines, and in 90 percent of the cases transaction rates are under 20 transactions per second. CPU utilization averages less than 10 percent and approaches 20 percent at peak levels. As might be expected, I/O and data transfer rates are also low. Not surprisingly, many such database applications have been successfully migrated to virtual machines running on VMware ESX Server systems (for a link to further reading, see "Customer Success Stories Featuring ESX Server" in "References").

This paper describes transaction processing workload performance in virtual machines using Microsoft SQL Server 2005 and VMware Infrastructure 3. This performance study was conducted at the HP Strategic Alliances Engineering (SAE) lab in Cupertino. The primary goal is to prove that Microsoft SQL Server 2005 can successfully handle enterprise-level transaction-processing workloads when running inside VMware virtual machines. To facilitate planning for server consolidation, this study presents sizing data and data on system resource utilization at various load levels for uniprocessor (UP) virtual machines, two-way SMP virtual machines, and four-way SMP virtual machines. This study also compares the performance of UP, two-way, and four-way SMP virtual machines across 32-bit and 64-bit virtual environments.

Executive Summary

This performance study clearly demonstrates that VMware Infrastructure 3 provides an excellent production-ready virtualization platform for customers looking to deploy Microsoft SQL Server inside virtual machines. Furthermore, together with virtualization-based distributed infrastructure services such as VMotion, VMware High Availability, and VMware Distributed Resource Scheduler, VMware Infrastructure 3 can provide increased serviceability, efficiency, and reliability for your SQL Server deployments. This should offer transformative cost savings to your dynamic data center.

The main conclusions that can be drawn from these experiments are illustrated in Figure 1 and Figure 2.

32-bit versus 64-bit Virtual Machines

- Virtual machine throughput scales well for both 32-bit and 64-bit virtual machines.
- 64-bit virtual machines provide better scalability for SMP virtual machines as user load increases.

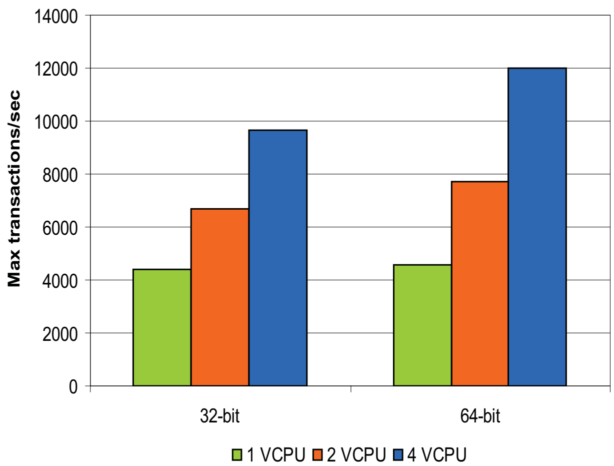

Figure 1: Throughput comparison for 32-bit and 64-bit virtual machines

UP versus SMP Virtual Machines

- SMP virtual machines provide better throughput scalability than UP virtual machines, but with a higher CPU cost.
- 64-bit SMP virtual machines scale better than the 32-bit SMP virtual machines as user load increases.

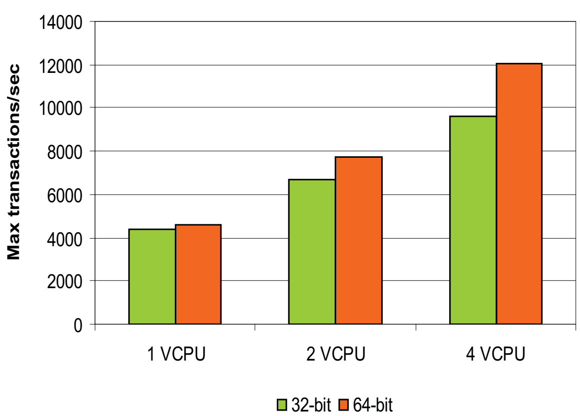

Figure 2: Throughput comparison for UP and SMP virtual machines

Experimental Setup

The DBHammer test tool, contained in the Microsoft SQL Server 2000 Resource Kit, was used to generate load for the experiments. It consists of a client application, written in Visual Basic, which generates transactions against a database server. Transactions either read or update a single record in the database. The default code generates an equal number of read and update transactions. This is a good representation of a typical transaction processing workload. To simulate decision support or business intelligence workloads, the DBHammer test should be run with appropriate read/write ratios. The kit contains source code and binaries for database creation and benchmark execution. Detailed instructions for setup, execution, and troubleshooting are also included.

The client component of the test was executed on a separate physical machine. The database server was hosted on a four-way HP ProLiant DL585 server. When we performed the virtual machine tests, the database server virtual machine was the only virtual machine running on the server. The data in this paper demonstrates the performance characteristics under circumstances in which the virtual machine has sufficient resources.

The standard test application setup was used. Table 1 provides the parameter values for this configuration.

Application Parameter	Value
Read/write ratio	50/50
Database row count	10 million rows
Interval (think time)	100 ms

Table 1: Benchmark parameter values

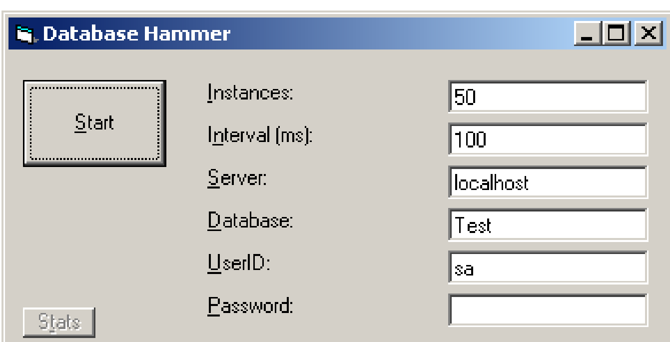

Figure 3: DBHammer client application

Varying the Instances configuration option on the client side, as shown in Figure 3, controls the load on the system. This option determines the number of clients generating transactions that are serviced by the database server.

Hardware Setup

The test machine, the SAN storage array, and the client machine used to drive the workload were set up as shown in Figure 4. A four-way HP ProLiant DL585 acted as the test server, and multiple two-way HP Proliant DL360 servers were used as client machines, while an HP EVA 6000 SAN fulfilled all storage needs. The test machines were connected via a gigabit Ethernet link. The link between the test server and the storage array was a Fibre Channel switch. Detailed specifications of all components are listed in "Appendix 1: Test Environment."

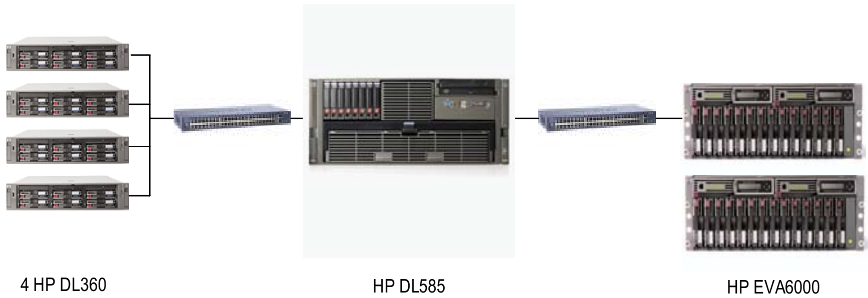

Figure 4: Schematic representation of test configuration

In order to ensure data accuracy, all components and links were dedicated to these experiments and were not shared with other machines.

Virtual Machines

We set the database server to run in UP and SMP virtual machines in which the guest operating system was Microsoft Windows Server 2003, Enterprise Edition with SP2. We configured the virtual machines with 3.6GB of main memory. These virtual machines were freshly created to ensure HAL correctness. We placed the guest operating system software (on the virtual C: drive) on the VMware VMFS partition created on the internal storage for the DL585 server.

Storage Layout

In the context of these experiments, storage layout refers to the placement of the database and log files. We placed these files on separate VMware VMFS partitions located on independent disk arrays. We placed the database files on a 10GB RAID 5 LUN for the 32-bit virtual machine and a 40GB RAID 5 LUN for the 64-bit virtual machine. We placed the log files on a10GB RAID 1 LUN for the 32-bit virtual machine and a 40GB RAID 1 LUN for the 64-bit virtual machine. This is a common configuration for general-purpose database installations. RAID 5 arrays provide reasonable fault tolerance and performance with acceptable cost and overhead unless the workload is dominated by a large number of write operations. RAID 1 provides the higher level of fault tolerance required for database log files.

We left parameter settings for the SAN, HBA, and Fibre Channel switch at their default values.

The SQL Server TempDB was located on the operating system partition (C: drive). It may be possible to achieve higher throughput rates by moving TempDB to a SAN partition, especially for decision support or business intelligence workloads.

Software Configuration

SQL Server 2005 Enterprise Edition with SP2 was used for these tests. While it may be possible to obtain better system performance by manually altering some SQL Server configuration parameters based on knowledge of application characteristics, it is recommended that most SQL Server configuration options be left at their default values. This recommendation is based on the fact that SQL Server monitors system load and fine tunes many of these parameters based on this dynamic feedback. We used default values for all configuration parameters in this study.

We configured VMware ESX Server to operate with default options and parameter settings.

Test Cases

The number of clients submitting transactions to the database determines the load on the system. We conducted tests with a large range of values for this parameter. The objective was to characterize performance for a range of load levels. We conducted these experiments for UP, two-way SMP, and four-way SMP virtual machines using both 32-bit and 64-bit versions of the guest operating system and SQL Server software.

Performance Results for 32-bit Virtual Machines

This section presents an overview of transaction processing workload performance as measured using the DBHammer test utility.

The primary measure of transaction-processing workloads is the throughput in units of transactions per second. As with any system-level workload, resource usage is a critical aspect of database workload and is particularly important for sizing and capacity planning. CPU usage data collected within virtual machines is not useful for two reasons. First, this data does not always accurately reflect the overhead of virtualization that is incurred by the ESX Server host. Second, because of the way time is kept within virtual machines, the usage data itself may be inaccurate (for details, see "Timekeeping in VMware Virtual Machines" in "References"). For these reasons, we use CPU usage data collected on the ESX Server host in this study. We used the `esxtop` tool to collect resource utilization statistics for VMware ESX Server. For further details, see "Appendix 2: Data Collection."

UP and SMP virtual machine performance can be compared in many different ways and along many different dimensions. In the server consolidation process, one important choice is the processor count of the virtual machines (UP, two-way SMP, or four-way SMP). SMP virtual machines usually have higher virtualization overhead than UP virtual machines, therefore the benefit to the end user must be carefully considered (for details see "Performance Tuning Best Practices for ESX Server 3" in "References").

In the following sections, we compare the performance of UP and SMP virtual machines in tests that feature constant load on the virtual machine. The results of these tests highlight the advantages of using SMP virtual machines.

Figure 5 shows the throughput scalability we achieved for 32-bit UP virtual machine by successively increasing the number of clients. The throughput demonstrates near-linear scalability until the CPU resources for the virtual machine are saturated.

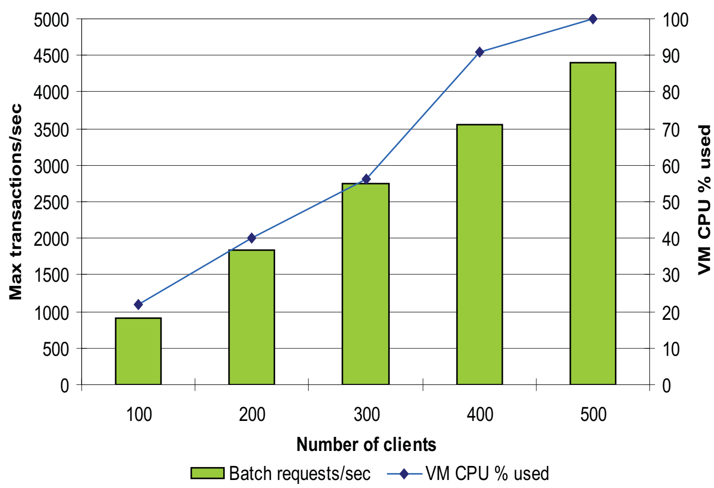

Figure 5: Throughput for 1-VCPU 32-bit virtual machine

Figure 6 and Figure 7 show similar throughput scalability for 32-bit SMP virtual machines. As with the UP virtual machine, the throughput demonstrates near-linear scalability until the CPU resources of the virtual machine are saturated. Furthermore, the SMP virtual machines can support higher transaction rates because of the additional CPU resources available.

Note: The measurement of virtual machine CPU used is cumulative and reflects the fact that there are two virtual CPUs in the virtual machine. If both virtual CPUs were fully saturated, the measurement would show 200 percent CPU used. In a virtual machine with four VCPUs, the maximum would be 400 percent CPU used.

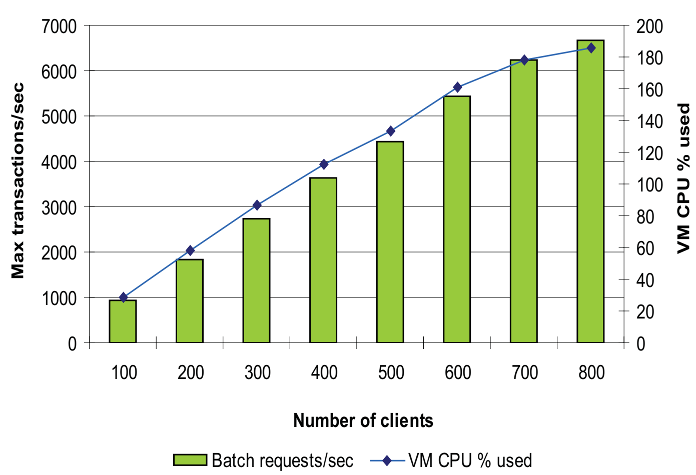

Figure 6: Throughput for 2-VCPU 32-bit virtual machine

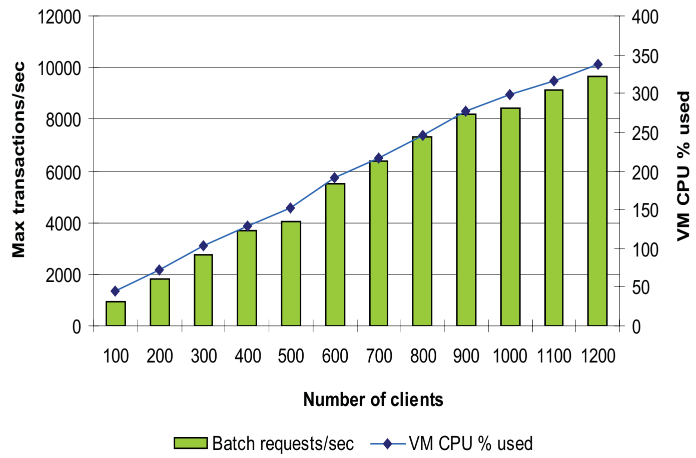

Figure 7: Throughput for 4-VCPU 32-bit virtual machine

Performance Results for 64-bit Virtual Machines

This section presents performance results for 64-bit virtual machines. As in the 32-bit performance results sections, we compare the performance characteristics of UP and SMP virtual machines to analyze behavior under constant load (that is, with the same number of clients). The results of these tests highlight the advantages of using SMP virtual machines.

Figure 8 shows the throughput scalability we achieved for a 64-bit UP virtual machine by successively increasing the number of clients. The throughput demonstrates near-linear scalability until the CPU resources for the virtual machine are saturated.

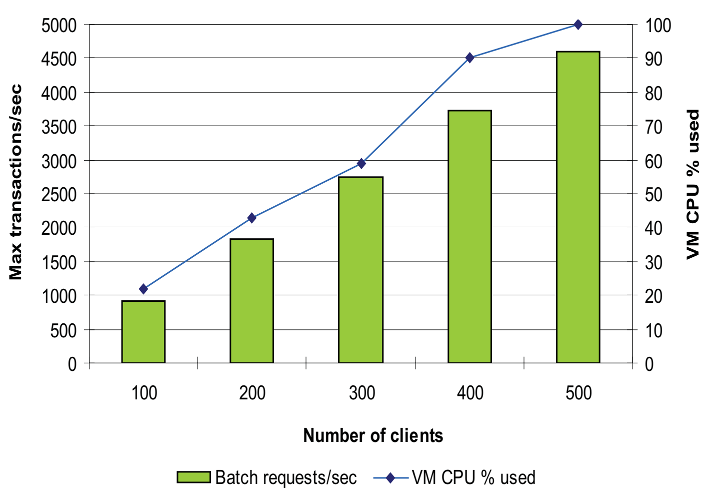

Figure 8: Throughput for 1-VCPU 64-bit virtual machine

Figure 9 and Figure 10 show similar throughput scalability for 64-bit SMP virtual machines. As with the UP virtual machine, the throughput demonstrates near-linear scalability until the CPU resources of the virtual machine are saturated. Furthermore, the SMP virtual machines can support higher transaction rates because of the additional CPU resources available.

Note: The measurement of virtual machine CPU used is cumulative and reflects the fact that there are multiple virtual CPUs in the virtual machine. If both virtual CPUs in a two-VCPU virtual machine were fully saturated, the measurement would show 200 percent CPU used. In a virtual machine with four VCPUs, the maximum would be 400 percent CPU used.

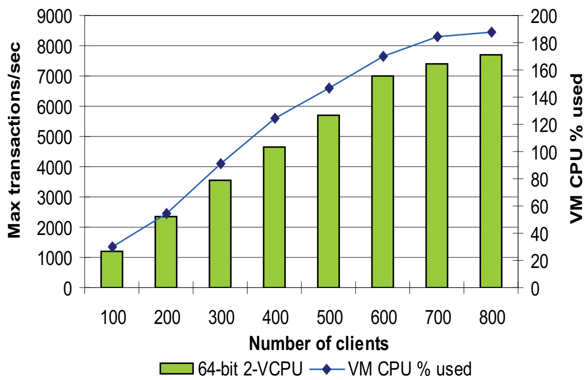

Figure 9: Throughput for 2-VCPU 64-bit virtual machine

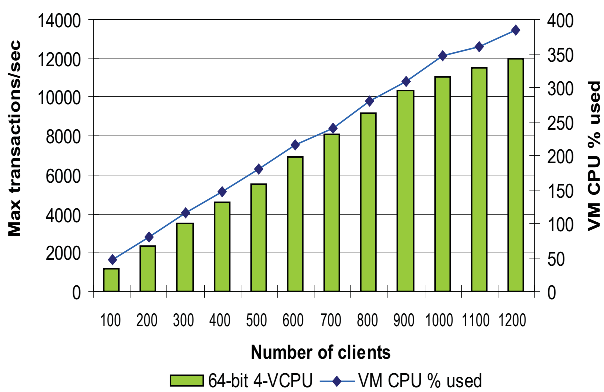

Figure 10: Throughput for 4-VCPU 64-bit virtual machine

Comparing Performance Results for 32-bit and 64-bit Virtual Machines

In this section, we compare the performance characteristics of 32-bit and 64-bit virtual machines to analyze behavior under constant load (that is, with the same number of clients). This set of comparisons highlights the benefits, if any, that a 64-bit environment may provide over a 32-bit environment. In these comparisons, the transaction processing rate is similar when using UP virtual machine in both 32-bit and 64-bit environments, as shown in Figure 11.

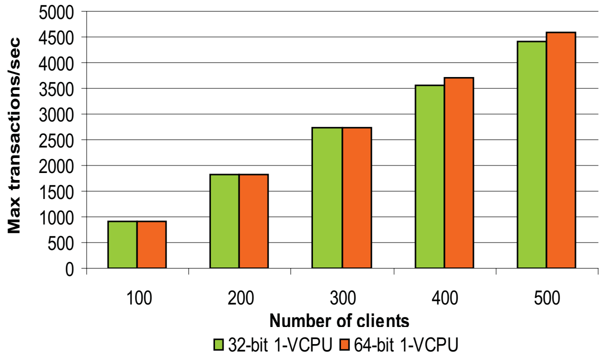

Figure 11: Throughput for 4-VCPU 64-bit virtual machine

For SMP virtual machines, the throughput rate is consistently higher in a 64-bit environment compared to a 32-bit environment, under constant load. Because of time constraints, we could not investigate this difference further. Figure 11 and Figure 12 show these results for 2-VCPU and 4-VCPU virtual machines.

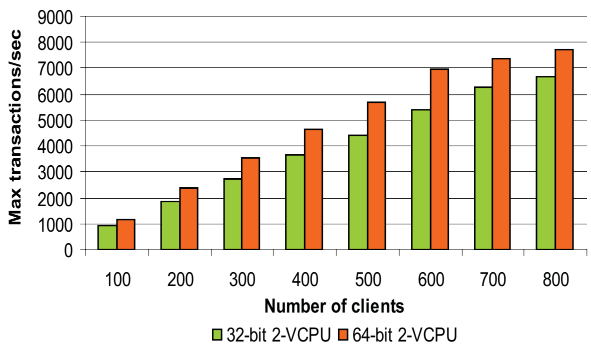

Figure 12: Throughput for 2-VCPU 64-bit virtual machine

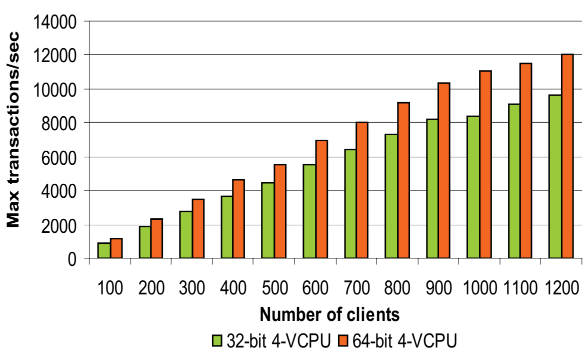

Figure 13: Throughput for 4-VCPU 64-bit virtual machine

Additional Observations

During these tests, we found that the DBHammer client seemed to saturate when running more than 250 instances. Consequently, we used a separate benchmark client for every 300 clients simulated by DBHammer. This DBHammer saturation behavior is particularly evident in the 4-VCPU SMP graphs above for the 500- and 1000-client data points.

The transaction throughput shown in these results represents the highest steady-state throughput achievable for the virtual machine until its CPU resources are saturated. However, note that while the CPU resources inside the virtual machine were exhausted, the physical CPU resources used on the ESX Server host were only used corresponding to the number of virtual CPUs because of the excellent isolation capability offered by VMware ESX Server. In other words, for UP virtual machines, only 13 percent of total physical CPU resources were utilized. Likewise, for 2-VCPU virtual machine, total physical CPU utilization was about 29 percent, and for the 4-VCPU virtual machine, the maximum total physical CPU utilization was about 52 percent. This implies that configurations running multiple instances of SQL Server virtual machines can achieve higher transaction throughput rates than those running only a single instance. It is possible that the transaction throughput in a configuration with multiple instances may be limited by the IOPS that the storage can handle. In such a situation, you may want to configure storage adapter queue depths and ESX Server outstanding disk requests parameters to minimize any queued disk I/O requests.

Statistics Summary

The following tables summarize the performance data for the UP and SMP virtual machines in both 32-bit and 63-bit environments presented in this study.

Number of Clients	Transaction Rate	Host % CPU Busy
100	919	22
200	1835	40
300	2748	56
400	3562	91
500	4406	100

Table 2: 32-bit 1-VCPU virtual machine performance summary

Number of Clients	Transaction Rate	Host % CPU Busy
100	918	29
200	1830	58
300	2745	87
400	3649	112
500	4439	133
600	5422	161
700	6250	178
800	6660	186

Table 3: 32-bit 2-VCPU SMP virtual machine performance summary

Number of Clients	Transaction Rate	Host % CPU Busy
100	920	44
200	1838	72
300	2753	103
400	3662	128
500	4439	152
600	5493	191
700	6406	217
800	7320	245
900	8178	277
1000	8426	298
1100	9126	316
1200	9637	337

Table 4: 32-bit 4-VCPU SMP virtual machine performance summary

Number of Clients	Transaction Rate	Host % CPU Busy
100	920	22
200	1834	43
300	2748	59
400	3717	90
500	4594	100

Table 5: 64-bit 1-VCPU virtual machine performance summary

Number of Clients	Transaction Rate	Host % CPU Busy
100	1184	30
200	2367	55
300	3533	91
400	4650	125
500	5697	147
600	6978	170
700	7403	184
800	7712	188

Table 6: 64-bit 2-VCPU virtual machine performance summary

Number of Clients	Transaction Rate	Host % CPU Busy
100	1156	46
200	2307	79
300	3462	115
400	4607	147
500	5537	181
600	6916	216
700	8063	241
800	9203	279
900	10311	310
1000	11079	346
1100	11492	361
1200	12016	384

Table 7: 64-bit 4-VCPU SMP virtual machine performance summary

Conclusions

The performance results we describe in this paper support the conclusion that running Microsoft SQL Server 2005 inside VMware virtual machines can provide an effective production-ready platform for running transaction-processing workloads.

Your IT organization can leverage the sizing data we present in this white paper to deploy Microsoft SQL Server successfully using VMware Infrastructure 3. By comparing effective workload throughput and resource utilization for your datacenter with the results presented in this paper, you can design and size your virtualized environment to meet your SLA targets.

The number of actual users and transactions you can support in a production environment will, of course, depend on the specific applications used and the intensity of user activity. The results of our tests clearly demonstrate that in a VMware Infrastructure 3 environment, the transaction throughput scales almost linearly even as the number of clients increases, for every configuration we tested. Our results show that the throughput rate of UP virtual machine is comparable for 32-bit and 64-bit environments. However, for SMP virtual machines, the throughput rate is slightly higher in a 64-bit environment compared to a 32-bit environment. Further, SMP virtual machines can service a higher number of users. One disadvantage of SMP virtual machines is that they have higher CPU costs per transaction than UP virtual machines. From a performance perspective, we recommend using SMP virtual machines rather than UP virtual machines for Microsoft SQL Server, and if possible, a 64-bit environment if your application stack supports it.

References

VMware

- VMware Virtual Performance Web site
  http://www.vmware.com/overview/performance/
- Customer Success Stories Featuring ESX Server
  http://www.vmware.com/customers/stories/
- Timekeeping in VMware Virtual Machines
  http://www.vmware.com/pdf/vmware_timekeeping.pdf
- Performance Tuning Best Practices for ESX Server 3
  http://www.vmware.com/pdf/vi_performance_tuning.pdf
- *Server Configuration Guide*
  http://www.vmware.com/pdf/vi3_server_config.pdf
- *SAN Configuration Guide*
  http://www.vmware.com/pdf/vi3_esx_san_cfg.pdf

HP

- HP ProLiants
  http://h18004.www1.hp.com/products/servers/platforms/
- HP Resource Web site on VMware
  http://www.hp.com/go/vmware
  http://www.hp.com/go/storage/vmware
- HP whitepapers on Microsoft SQL
  http://h71019.www7.hp.com/ActiveAnswers/cache/70729-0-0-0-121.html

Microsoft

- DBHammer test utility, in Microsoft SQL Server 2000 Resource Kit
- SQL Server 2005 Books Online
  http://technet.microsoft.com/en-us/library/ms130214.aspx

Appendix 1: Test Environment

This section provides details about the hardware and software environment in which the tests described in this study were run.

Server

Server Hardware

- HP ProLiant DL585 G2
- Processor (four-way): Dual-Core AMD Opteron Processor 8220 SE
  x86 Family 15 Model 65 Stepping 2, AuthenticAMD 2.8GHz
  L2 Cache 512KB
- Memory: 40GB
- Local storage: HP SmartArray P400
  4 73.4GB 10Krpm SAS hard disk drives
- Ethernet adapters (2): HP NC371i Multifunction Gigabit Server Adapter
- HBA: Emulex LightPulse LP1050

Server Software

- VMware Software: ESX Server 3.0.1 Build 32039

Virtual Machine Configurations

- CPU: UP, two-way and four-way SMP
- Memory: 3.6GB
- Connectivity: vmxnet
- Guest Operating System: Windows Server 2003 Enterprise Edition with SP2 (32-bit or 64-bit)
- SQL Server: SQL Server 2005 Enterprise Edition with SP2 (32-bit or 64-bit)

Storage

SAN Storage

- HP EVA 6000 SAN; 2C4D Configuration
- SPA: HSV200-6100
- SPB: HSV200-6100
- Total 56 72.8GB FC SCSI 15Krpm disks

Fibre Channel Switch

- HP 4/64 Switch
- Link speed: 4Gb/sec

Client

Benchmark Driver Machines

- Windows Server 2003 Enterprise Edition (32-bit) with SP1
- HP Proliant 360 G4
- Two-way Intel Xeon 2.8GHz CPU HT enabled
- Memory: 3.5GB
- Local storage: 33GB
- Ethernet adapter: HP NC371i Multifunction Gigabit Server Adapter

Appendix 2: Data Collection

During these tests, performance data was collected at two levels. The esxtop tool was used to collect resource utilization statistics for VMware ESX Server. Furthermore, esxtop data helps us understand how the ESX Server host handles a virtual machine. This information may be useful in diagnosing performance issues.

We also used the Windows MMC snap-in Performance Monitor (Perfmon) as a data collection tool. Perfmon categorizes data by objects (for example, processor or physical disk) and provides counters relevant to each object (for example, disk writes/sec for physical disk objects). The default instance for such counters sums the data for all instances of the object and may not be useful. It may therefore be necessary in some cases to select the proper instance of the counter. For example for the Physical Disk Object it is necessary to specify the disk for which the statistics are to be collected.

The Perfmon objects and counters relevant to this report are listed below.

Physical Disk Object

- Disk Bytes/sec: Shows the data transfer rate to or from the disk during write or read operations.
- Disk Read Bytes/sec: Shows the data transfer rate from the disk during read operations.
- Disk Reads/sec: Shows the rate of read operations on the disk.
- Disk Transfers/sec: Shows the rate of read and write operations on the disk.
- Disk Write Bytes/sec: Shows the data transfer rate to the disk during write operations.
- Disk Writes/sec: Shows the rate of write operations on the disk.

Processor Object

- % Processor time: This counter is the primary indicator of processor activity and displays the average percentage of busy time observed during the sample interval.

SQL Server Databases Object

- Transactions/sec: Shows the number of transactions started for the database.

SQL Server SQL Statistics Object

- Batch requests/sec: Shows the number of SQL Batch requests received by server.

Because both `esxtop` and Perfmon report data averaged over the sampling interval, it is important to use the same sampling interval for both.