NETAPP TECHNICAL REPORT

Microsoft SQL Server 2008: New Compression Technology Enhances Decision Support Workload Performance

Wei Liu, Ian Breitner, NetApp

Mostafa Mokhtar, Shirley Wang, Ashit Gosalia, Microsoft

Vinay Kukarni, IBM

October 2008 | TR-3719

EXECUTIVE SUMMARY

This report details a series of tests performed by NetApp, Microsoft, and IBM designed to demonstrate the benefits of Microsoft® SQL Server® 2008 compression technology by using a large-scale Decision Support System (DSS) workload on IBM x3950 M2 and NetApp® FAS3070 storage systems.

TABLE OF CONTENTS

1 INTRODUCTION 3
1.1 NETAPP FAS3070 3
1.2 IBM SYSTEM x3950 M2 4
1.3 MICROSOFT SQL SERVER 2008 4
2 TEST ENVIRONMENT 5
2.1 TEST TOPOLOGY 5
2.2 SERVER AND STORAGE CONFIGURATION 6
2.3 SQL SERVER 2008 DATABASE 9
3 DSS TESTING QUERIES 9
3.1 PROFIT ENHANCEMENT (PE) 9
3.2 ADVERTISEMENT PROFIT (AP) 10
3.3 UNSHIPPED ORDER SUMMARY (UO) 10
3.4 COUNTRY TRADE SUMMARY (CT) 10
3.5 PRICING SUMMARY (PS) 10
3.6 LARGEST BUYERS (LB) 10
3.7 MARKET SHARE MOVEMENT (MS) 11
3.8 SHIPPING MODES (SM) 11
3.9 LATE SHIPPING SUPPLIERS (LS) 11
4 RESULTS AND ANALYSIS 11
4.1 QUERY EXECUTION TIME 11
4.2 DISK READ PERFORMANCE 12
4.3 QUERY CPU CONSUMPTION 12
4.4 STORAGE AND SPACE REDUCTION 12
5 CONCLUSION 13
6 ACKNOWLEDGEMENTS 13

1 INTRODUCTION

Microsoft SQL Server 2008 makes a major advance in data warehouse scalability. The database engine contains numerous enhancements designed to improve both absolute performance and multi-CPU scaling on decision support workloads. One important strategic improvement is compression. Database compression enables data compression on disk and in memory, reducing the amount of I/O from disk and the in-memory footprint. As a result, compression can not only save space but also improve the performance of I/O-intensive workloads.

This technical report documents a series of tests, run jointly by NetApp, Microsoft, and IBM, that demonstrate the benefits of Microsoft SQL Server 2008 compression technology by using a large-scale Decision Support System (DSS) workload on IBM x3950 M2 server and NetApp FAS3070 storage systems.

This is the second in a series of papers that covering various aspects of the performance and deployment possibilities of SQL Server 2008. We intend to highlight some of the advances made in SQL Server 2008 and the impact that SQL Server 2008 has on the attached storage subsystem.

For this series of tests, a SQL Server 2008 1TB data warehouse was used running on an IBM x3950 M2 x64 32-core server connected via 4Gb Fibre Channel (FC) to NetApp FAS3070 storage controllers.

The selection of the FAS3070 storage platform and the configurations used for the SQL Server 2008 tests were determined in collaboration with the Microsoft SQL Server Performance team. The requirements were to provide a flexible yet scalable and high-performance storage subsystem. It was important that no bottlenecks occur in the storage system or the storage interconnects. Not only was it important to understand the storage loads generated by SQL Server 2008 under various application requirements, but more importantly, the opportunity was provided for the SQL Server Performance team to examine and address any performance bottlenecks in the SQL Server 2008 engine. Both FC and iSCSI storage interconnects were also a requirement—4Gb for the FC interconnect and both 1Gb and 10Gb Ethernet for the iSCSI interconnect. These requirements for flexible configuration, large capacity, high throughput, and capability to sustain a high IOPS rate resulted in the selection of the NetApp FAS3070 as the appropriate storage platform. The IBM x3950 M2 server was selected to meet the performance and scalability requirements of the workload. IBM x3950 M2 scales from a 4-socket, 16-core configuration to a 16-socket, 64-core configuration.

1.1 NETAPP FAS3070

NetApp fabric-attached storage (FAS) systems simplify data management, enabling enterprise customers to reduce costs and complexities, minimize risks, and control change. NetApp FAS systems are the most versatile storage systems in the industry for storage consolidation.

The FAS3070 addresses the core requirements of the midrange enterprise storage market, delivering a superb blend of price, performance, and scalability for SQL Server databases and business applications. The compact, modular design provides integrated FC SAN and IP SAN (iSCSI) storage with scalability to over 500 disk drives. The FAS3070 storage controller supports both FC and SATA disk drives for tiered storage. FAS3070 systems support as many as 32 FC ports or 32 Ethernet ports, including support for 2Gb and 4Gb FC and for 10 Gigabit Ethernet.

The FAS3070 runs the NetApp Data ONTAP® operating system, which is optimized for fast, efficient, and reliable data access and retention. Data ONTAP 7G dramatically simplifies common storage provisioning and management operations. LUNs and volumes created and configured using FlexVol® technology can be dynamically expanded or contracted with a single command. FlexVol volumes also enable thin provisioning, which avoids the cost of overprovisioning and the time-consuming reconfiguration typical with other storage solutions. Host-based NetApp SnapDrive® extends this flexible storage provisioning capability to databases and applications. Another Data ONTAP 7G feature, FlexClone®, instantaneously creates cloned LUNs or volumes without requiring additional storage. FlexClone technology can dramatically improve the effectiveness and productivity of application and database development and predeployment testing.

FAS hardware design and the Data ONTAP operating system are tightly integrated to provide resilient system operation and high data availability. FAS systems incorporate redundant and hot-swappable components and patented double-parity RAID-DP®. NetApp RAID-DP, a high-performance implementation of RAID 6, provides superior data protection with negligible impact on performance. NetApp Snapshot™

technology provides up to 255 data-in-place, point-in-time images per LUN or file system, available for near-instantaneous file-level or full data set recovery. The minimal performance overhead of NetApp Snapshot technology makes it well suited for protecting production data. Host-based SnapManager® software integrates Snapshot management with applications, providing consistent backup images and application-level recovery in minutes. SnapMirror® uses Snapshot copies to provide incremental block-level synchronous and asynchronous replication; SnapVault® uses it for block-level incremental backups to another system. Together, these SnapSuite™ products help deliver the high application-level availability that enterprises require for 24×7 operation.

1.2 IBM SYSTEM x3950 M2

Engineered with the needs of enterprise organizations in mind, the IBM x3950 M2 server provides a scalable, efficient, and highly reliable solution. This fourth-generation X-Architecture enterprise server combines 64-bit performance with a balanced design. The x3950 M2 can help organizations meet business demands with confidence. Many organizations require servers that expand as business grows, and the x3950 M2 provides the flexibility to run more applications on the same piece of hardware. These features deliver an optimized solution for large database enterprise applications.

IBM X-Architecture pioneered XpandOnDemand ("pay as you grow") scalability, which allows chassis to be simply cabled together to form larger scale-up systems. This unique capability allows IBM to offer a large symmetric multiprocessing (SMP) system at entry price points. With XpandOnDemand, you can start small, with a 4- or 8-socket configuration, and later expand as your needs change, without needing to buy more than you need up front or throw away parts later as you expand. IBM's X4 technology-based systems are the ideal solution for scale-up database-serving applications on Microsoft Windows® with Microsoft SQL Server. Database hosting demands ultimate server reliability features, and once installed, they grow and grow.

The 4-socket X4 servers are designed to protect your data with high performance, high reliability, and high availability. They support the latest quad- and dual-core Intel® Xeon™ MP processors, designed with high-performance quad 1066MHz front-side buses (FSB), 64-bit extensions (EM64T), and either 8MB (dual-core) or 4MB, 6MB, or 8MB (quad-core) of L2 cache, to help provide you the computing power you need to match your business needs and growth. In addition, x3950 M2 uses industry-standard DDR II memory with Chipkill ECC (Error Checking and Correcting) protection for high performance and reliability and lower energy consumption than fully buffered memory.

For even higher levels of availability, the eX4 servers also offer Memory ProteXion, memory scrubbing, and optional memory mirroring. A dual-port integrated high-speed Gigabit Ethernet controller with TOE (TCP Offload Engine) is standard, as are seven high-performance PCI-E x8 adapter slots. The x3950 M2 offers industry-leading scalability, including quad-processor support (upgradeable to 16 processors/64 cores); up to 256GB of memory (upgradeable to 1TB); and high performance—up to four 2.5-inch internal Serial Attached SCSI (SAS) hot-swap hard disk drives with an internal storage capacity of 587.2GB. Hardware-based RAID-0/1 support is standard. The 4U size of the chassis helps you maximize your rack investments. Up to 10 of these servers can be installed in a single 42U rack, for a total of up to 40 processors, 70 PCI-E slots, and 40 HDDs, offering an ideal balance of performance, storage, and I/O slots per rack. Optional Advanced Connectivity Technology (ACT) interconnect cabling helps reduce cable clutter and cost and minimizes installation time when interconnecting many rack-mounted servers.

1.3 MICROSOFT SQL SERVER 2008

Microsoft SQL Server 2008 has been greatly enhanced to improve the performance of DSS workloads, capable of meeting the mission-critical needs of large enterprise customers running in terabytes of data warehouses. Here are a few highlights of new performance enhancements for DSS workloads:

- **Star join.** With dimensionally modeled data warehouses, a big part of the workload typically consists of what are known as star join queries. These queries follow a common pattern that joins the fact table with one or several dimension tables. In addition, star join queries usually express filter conditions against the non-key columns of the dimension tables and perform an aggregation on a column of the fact table. With SQL Server 2008, customers will experience significant performance improvements for many star join queries that process a significant fraction of fact table rows. The new star join optimization uses a series of hash joins, building a hash table for each dimension table that participates. As a byproduct of building this hash table, additional information, called a *bitmap filter*, is built. These filters are pushed down into the scan on the fact table, and they effectively eliminate almost all the rows that would be eliminated later by the joins. This eliminates the need to spend CPU time later copying the eliminated rows and probing the hash tables for them.

- **Partition table parallelism.** Data warehouse applications typically collect large amounts of historical data in fact tables, which are often partitioned by date. In SQL Server 2005, queries that touch more than one partition use one thread and therefore one processor core per partition. This sometimes limits the performance of queries that involve partitioned tables, especially when running on parallel shared memory multiprocessor (SMP) computers with many processor cores. Partitioned table parallelism improves the performance of parallel query plans against partitioned tables by better using the processing power of the existing hardware, regardless of how many partitions a query touches.
- **Few outer row.** In some DSS queries, the outer side of a nested loop is a parallel scan with a filter, and if the qualifying data is only a few rows and clustered, they are picked by a single thread. This thread then has to do all the work for the nested loop join, even though there are idle threads. SQL Server 2008 introduces an exchange above the outer side of a nested loop join that produces few rows, and as a result more evenly redistributes the rows among threads and greatly improves the scalability.
- **Compression.** Database compression is a space-saving feature that helps to compress data on disk and in memory. SQL Server 2008 offers two types of compression; row and page. Row compression compresses data within a row, and page compression looks for additional opportunities across the rows in a page. Compression helps to reduce the amount of I/O from disk and the in-memory footprint and therefore helps I/O-intensive workloads. Compression and decompression have a CPU overhead.
- **Backup compression.** Backup compression, added in SQL Server 2008, saves both time and space in backups. With this feature, the backup stream is compressed before it is written out to the destination. Compression results are highly dependent on the data being compressed, but testing on typical customer databases has shown significant savings in space. Because SQL backup is typically I/O intensive, the reduction in I/O actually results in time savings as well. Creation of compressed backups is a feature of the Enterprise Edition SKU, although any SKU can restore a compressed backup.

This section describes just a small subset of the many data warehousing performance and scalability enhancements introduced by SQL Server 2008. For a complete list and more in-depth description of data warehousing improvements in SQL Server Relational Database Management System (DBMS), as well as in Integration Services (SSIS), Analysis Services (SSAS), and Reporting Services (SSRS), read this technical white paper by Microsoft:http://www.microsoft.com/sql/techinfo/whitepapers/SQL2008IntroDW.mspx

2 TEST ENVIRONMENT

The NetApp equipment on site in the Microsoft SQL lab serves as a high-performance, highly flexible, scalable, and reliable storage platform for SQL Server 2008 development and testing. For instance, the NetApp storage platform is used for SQL Server 2008 development tracking, validation, performance testing, and regression testing. The equipment is also intended for a variety of different SQL Server workload performance baselines and comparisons. Therefore the storage configurations used in this testing are overconfigured, so that the storage or the storage interconnect does not become a bottleneck during the test runs.

The following sections detail the test topology, and the server and storage configurations used, as well as database layout in this series of DSS workload tests.

2.1 TEST TOPOLOGY

Decision support workloads are frequently very I/O intensive. The high I/O requirements of these tests were met by using NetApp FAS3070 storage systems connected to the IBM x3950 M2 server using multiple4Gb/sec Fibre Channel interconnects.

Figure 1 illustrates the topology of the SQL Server 2008 1TB data warehouse workload testing. The IBM x3950 M2 server was connected to 12 FAS3070 controllers using twenty-four 4Gb/sec FC interconnects, 2 FC links per controller.

Note: Because of the choice of workload size, this topology uses only a subset of both the IBM x3950 M2 server capability and the NetApp storage capability deployed in the Microsoft SQL Server lab.

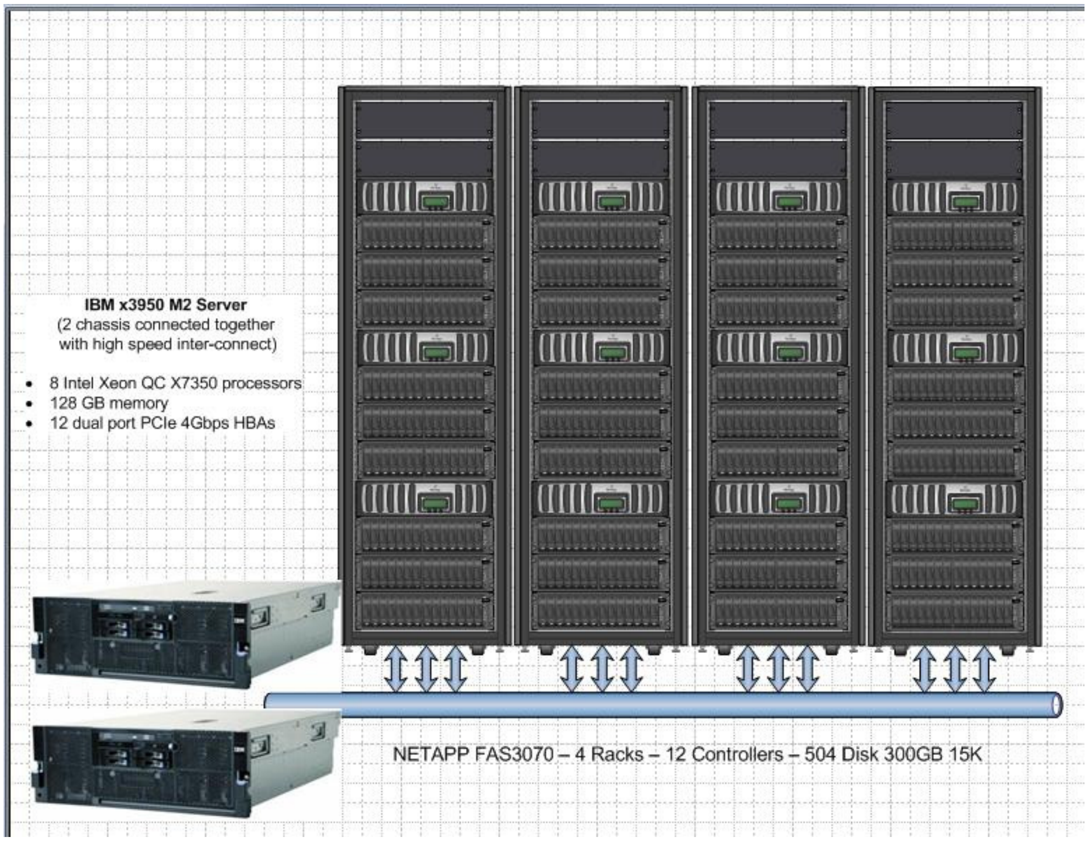

Figure 1) Server, storage, and Fibre Channel interconnects for the 1TB data warehouse.

2.2 SERVER AND STORAGE CONFIGURATION

The IBM x3950 M2 configuration consists of two physically separate chassis connected with two scalability port cables. There is an IBM ScaleXpander key in each chassis, which permits them to logically operate as a single SMP server. The total system has eight Intel quad core Xeon X7350 processors and 128GB memory. Six of the seven PCI-E slots in each chassis were populated with Emulex LP111002 and QLogic QLE2462 HBAs.

Table 1) Summary of IBM x3950 M2 configuration.

Server	IBM x3950 M2
Processors	8 Intel Xeon X7350 2.93Hz (Quad core)
Cores	32
Front-side bus frequency	1,066Mhz
Memory	128GB (64 2GB DIMMs)
PCI	6 Emulex LP111002-M4 dual-port 4Gb/sec HBAs 6 QLogic QLE2462 dual-port 4Gb/sec HBAs

NetApp storage controllers were used for SQL Server 2008 DSS databases and transaction logs. Figure 2 shows 12 NetApp storage controllers (FAS3070) directly connected to the IBM x3950 M2 server via 6 Emulex LP111002-M4 Dual-port 4Gb/sec HBAs and 6 Qlogic QLA2642 Dual-port 4Gb/sec HBAs. Therefore two 4Gb/sec FC links connected each FAS3070 controller to the IBM x3950 M2 server.

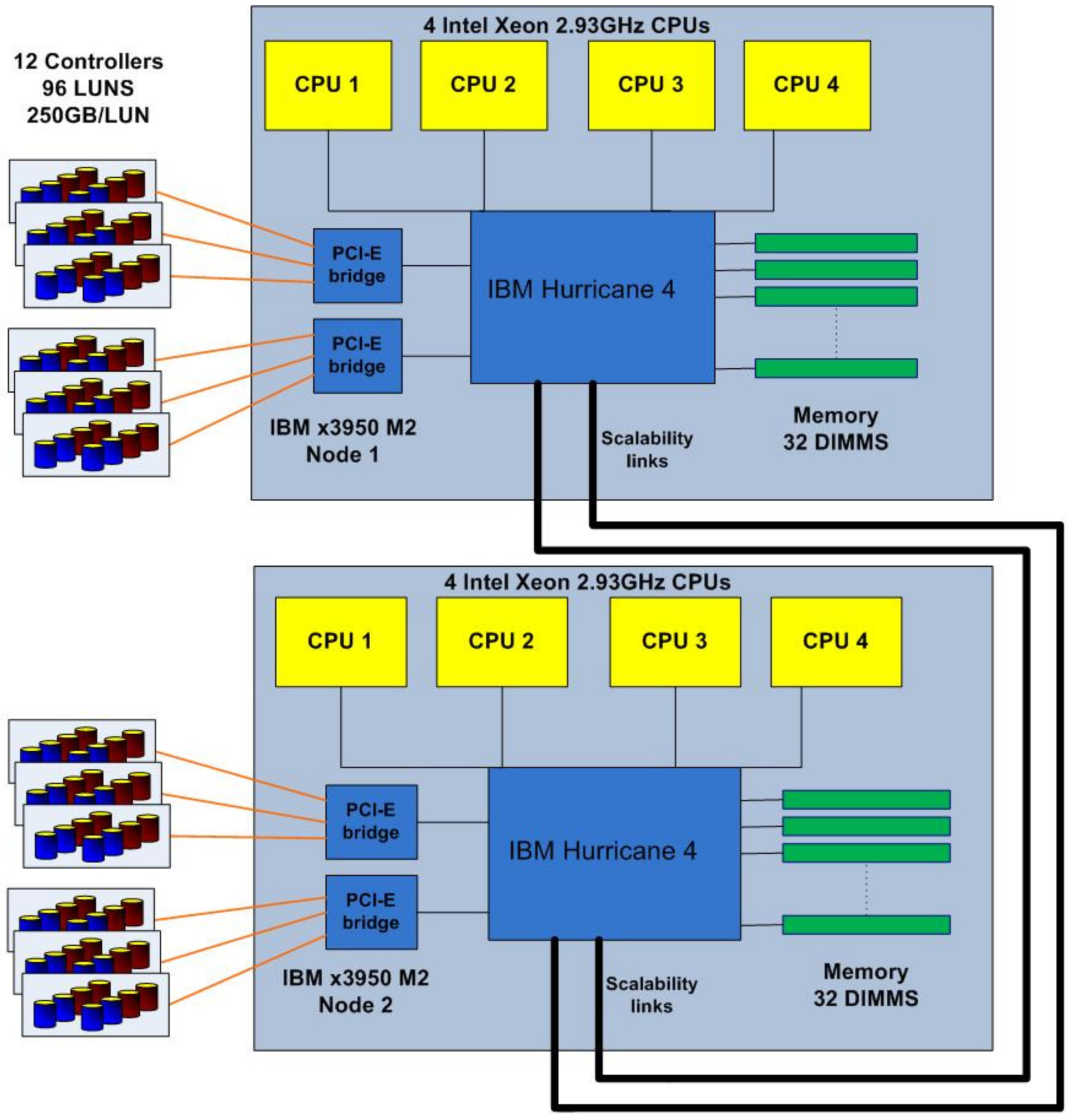

Figure 2) Detailed server and storage connection via 4Gb/sec Fibre Channel interconnects.

Also shown in Figure 3 are 8 LUNs (250GB each) per controller, for a total of 96 LUNs, mounted to the IBM x3950 M2 server for DSS databases and transaction logs.

The FAS3070 controllers ran version 7.3 of Data ONTAP. Each FAS3070 was configured with forty-two 300GB 15K RPM FC disks.

Note: The FAS3070 storage system described in this paper is used by the Microsoft SQL Server Performance Team for multiple parallel projects and was configured for maximum testing flexibility rather than sized for this particular test. As a result, the tested configuration can provide more I/O throughput and storage capacity than was used during this test. This design allowed the storage to be provided in such a way that no bottlenecks would be associated with the storage system or the interconnects to that storage.

This means that more controllers and more spindles were provided than would otherwise have been necessary to support the load during the test. Future technical reports will focus on deployment configurations and best practices for SQL Server 2008 with NetApp storage.

STORAGE VIRTUALIZATION

Data ONTAP 7G introduced powerful storage virtualization features that dramatically simplify storage provisioning and greatly improve storage use, flexibility, and manageability. Two of the key concepts are aggregates and FlexVol volumes. An aggregate is a logical container for pools of physical disks that are organized into one or more RAID-DP groups. An aggregate is the logical layer that decouples volumes from the underlying physical storage. A FlexVol volume is a logical entity that is separated from the physical storage that contains the associated data. One or more FlexVol volumes can reside within an aggregate. A FlexVol volume can grow or shrink in size, constrained only by the hard limits of the aggregate size or the soft limits set when the volume was first created. Each FlexVol volume leverages the performance of all of the disks in the aggregate.

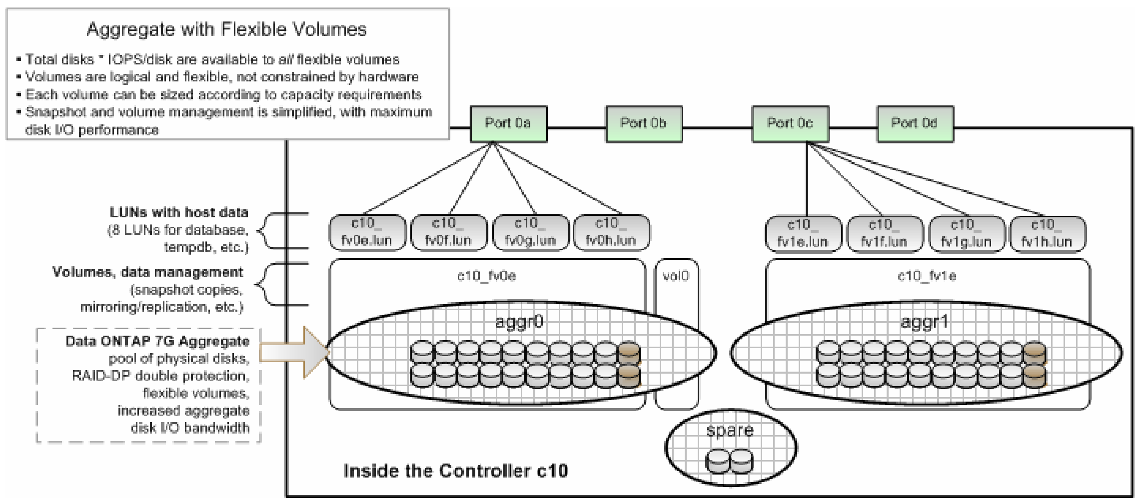

Figure 3) FAS3070 configuration.

Figure 3 shows detailed configuration of one of the FAS3070 controllers. Note that all 12 controllers are configured in identical fashion.

A total of 40 of the 42 drives within each controller were used to form 2 aggregates, each using 20 drives. The remaining 2 drives were left as hot swappable spares. In Figure 3, 2 drives within each aggregate are darker gray than the others. These 2 drives are the 2 parity drives within the RAID-DP group, which protect against double disk failure.

One FlexVol volume was then created within each aggregate, resulting in 2 volumes per controller. Note also that residing in aggr0 is an additional volume, vol0, which is needed by Data ONTAP for controller management.

Within each FlexVol volume, 4 LUNs (250GB) were created. Thus 4 LUNs per aggregate, 8 LUNs per controller (96 LUNs total on all 12 controllers) were created and then mapped to the IBM x3950 M2 server for storing DSS databases and transaction logs.

Figure 3 also illustrates that 4 LUNs residing within an aggregate were physically connected to the IBM x3950 M2 server using one 4Gb/sec FC port. Therefore 2 ports per controller were used in this test environment. Table 2 summarizes the storage configuration information.

Table 2) Summary of NetApp storage used in the 1TB data warehouse.

Storage Entity	Quantity
NetApp FAS3070 controllers	12
300GB 15K 4Gb/sec FC disks	504 (42 per controller)
Aggregates	24 (2 per controller)
FlexVol volumes	24 (2 per controller, 1 per aggregate)
LUNs	96 (8 per controller, 4 per aggregate, 4 per volume)
4Gb/sec FC links	24 (2 per controller, 1 per aggregate)

2.3 SQL SERVER 2008 DATABASE

The type of DSS database tested is representative of databases found in many customer environments and is designed for tracking sales, customer, supply-chain, and product lifecycle trends. The central charter of a DSS database is to help organizations increase profitability by analyzing trends and correlations over long periods of time.

The test database was fully normalized and fully indexed on primary and foreign keys. The size of the database, including tables, indexes, and backup, is 3.7TB on disk.

DATABASE LAYOUT

Among the 96 LUNs mounted to the IBM server, 48 contain databases, shown in Figure 2 in blue. The other 48 LUNs contain tempdb and backup files created by SQL Server (brown in Figure 2). The transaction log was placed on one of the 48 LUNs shared with tempdb and backup files.

From Figure 2, it is apparent that the large-scale DSS database was evenly distributed across all 12 controllers, available spindles, and FC interconnects.

SQL SERVER 2008 TUNING OPTIONS

SQL Server 2008 performs most of the necessary tuning automatically and dynamically configures its parameters based on usage and availability of system resources. In addition, affinity mask was used to associate SQL Server threads to all 32 cores on the IBM x3950 M2 server.

3 DSS TESTING QUERIES

A number of typical decision support queries were run to stress the system and to evaluate the performance of the system. These queries were chosen for their complexity and diversity in terms of data access patterns and query parameters, and because they access a large proportion of the available data. All queries were run as separate job requests. The set of queries is described in the following sections.

3.1 PROFIT ENHANCEMENT (PE)

This query provides the total increase in profits, if certain discounts had not been offered on products sold during a specified time period. The PE query assists in determining future product discounts.

Stress characteristics:

- I/O intensive
- Intensive indexing scan required

3.2 ADVERTISEMENT PROFIT (AP)

This query measures the percent of profit that was made as result of advertising for a given time period. The AP query helps companies gauge their return on advertising.

Stress characteristics:

- Hash joins and nested loop joins are critical

3.3 UNSHIPPED ORDER SUMMARY (UO)

This query extracts a summary of a specified number of highest revenue generating orders not shipped by specified date. For example, if N=20, the query returns the highest 20 rows (unless fewer than 20 rows qualify for the SQL statement, in which case all rows are returned). This query assists in identifying the shipment priority to meet company revenue goals.

Stress characteristics:

- Complex query
- Query plan and optimization critical
- Intensive sorting

3.4 COUNTRY TRADE SUMMARY (CT)

This query calculates the total revenue of products sold between certain countries during a specified time period. It also measures the trade volume and assists in renegotiation of shipping contracts.

Stress characteristics:

- Complex query
- Query plan and optimization critical

3.5 PRICING SUMMARY (PS)

This query provides a total count and total price of all products sold during a specified time period. The time period was selected so that approximately 95% of the table was scanned. The PS query measures the total amount of business sales during the specified time frame.

Stress characteristics:

- Floating point calculations
- Expression evaluation performance critical

3.6 LARGEST BUYERS (LB)

This query extracts a specified number of large buyers whose purchases have exceeded certain transaction thresholds. For example, if N=50, the 50 largest buyer would be returned, along with information about the customers and associated transactions. The EB query helps companies identify the customers they should value the most.

Stress characteristics:

- Intensive index scans and lookups required
- Hash joins and nested loop joins are critical
- Intensive sorting

3.7 MARKET SHARE MOVEMENT (MS)

This query calculates the market share movement for a part in a particular nation in two years.

Stress characteristics:

- Complex query
- Query plan and optimization critical
- Hash Joins critical

3.8 SHIPPING MODES (SM)

This query determines whether using cheaper shipping modes means delaying delivery of high-priority orders until after the committed date.

Stress characteristics:

- Hash joins are critical
- Intensive index scans

3.9 LATE SHIPPING SUPPLIERS (LS)

This query lists suppliers that ship late.

Stress characteristics:

- I/O intensive
- Intensive scan indexing required

4 RESULTS AND ANALYSIS

For each DSS query, tests were run using the configurations described in the preceding sections. Baseline measurements were first taken on SQL Server 2008, without compression; then comparison measurements were taken on the same SQL Server 2008 build, with page compression.

This section shows the test results in terms of execution times, read Mbytes per query, and server CPU consumption per query.

4.1 QUERY EXECUTION TIME

Figure 4 shows the relative query response times (blue bars) for nine typical DSS queries: Profit Enhancements, Advertisement Profit, Unshipped Orders, Country Trade Summary, Pricing Summary, Largest Buyers, Market Share Movement, Shipping Modes, and Late Shipping Suppliers. Each query's execution time was normalized using the baseline execution time (shown as the horizontal line at 100%) in SQL Server 2008 without compression enabled. For example, Figure 4 shows that the PE query with compression enabled executed in roughly half the amount of time as the baseline test without compression. In all cases, response times for queries run on SQL Server 2008 with compression were faster than those run without compression, indicating significant performance improvements.

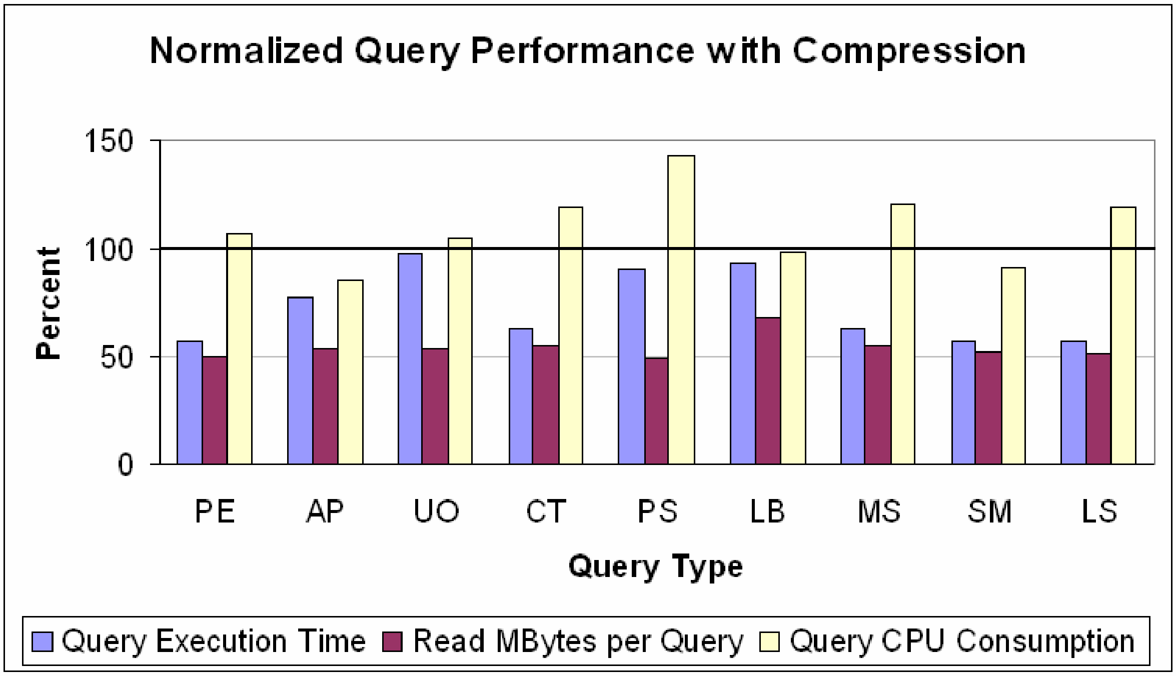

Figure 4) Comparison of SQL Server 2008 DSS query performance with and without compression. The horizontal line at 100% is the baseline performance with no compression. The blue, burgundy, and yellow bars show normalized query execution time, read Mbytes per query, and query CPU consumption with compression relative to the results without compression, respectively.

4.2 DISK READ PERFORMANCE

The burgundy bars in Figure 4 show total disk Megabytes (MB) read per query for SQL Server 2008 with compression. The results were normalized to the SQL Server 2008 baseline results without compression (shown as the line at 100% in the chart). For example, Figure 4 shows that the PE query with compression enabled read about half the amount of data from disk as the baseline test without compression. For all queries, with compression, the amount of data read per query is lower (better) than that of SQL Server 2008 without compression.

The decrease in disk MB read is an indication of the effectiveness of SQL Server 2008 compression, which improves query performance by reducing the amount of data read from disk. This reduces the total volume of data transferred to satisfy the query and reduces the overall load on the storage system. .

4.3 QUERY CPU CONSUMPTION

The yellow bars in Figure 4 show the relative query CPU consumption (processor utilization * query execution time) with compression. The results were normalized to the baseline of SQL Server 2008 with no compression. The majority of the queries with compression showed a minor increase or decrease in query CPU consumption. Only four of the queries have a substantial increase with compression. The increase in query CPU consumption could be attributed to decompression costs. Overall processor costs can also decrease if the query reduces I/O so that buffer pool management costs decrease.

4.4 STORAGE AND SPACE REDUCTION

So far we have highlighted some performance benefits of using the SQL Server 2008 database compression feature. Database compression also enables disk space reduction; therefore it is a space-saving feature. Table 3 shows the space reduction using page compression. Moreover, database compression can also reduce the space needed for database backup. Backup compression can reduce the storage costs for backups significantly and further reduce the backup size of an already compressed database. Table 4 shows the backup compression results.

Table 3) Compression: Space reduction for compressed versus uncompressed databases.

Database Schema	Database Size in GB	Space Reduction
Default schema	1,665	
Page compression	955	43%

Table 4) Backup compression: Comparing backup space reduction for compressed versus uncompressed backups.

Data Compression	Backup Type	Backup Size in GB	Backup Compression Ratio	Space Reduction
Uncompressed	Default backup	1,665		
Page compression	Default backup	955	1.7	43%
Page compression	Compressed backup	593	2.8	64%

5 CONCLUSION

This is the second in a series of papers that look at the performance, scalability, and deployment of SQL Server 2008. This paper focuses on the performance benefits of SQL Server 2008 database compression with DSS workloads.

The test results show that SQL Server 2008 compression can effectively reduce the I/O bandwidth needed and can also improve overall query performance.

The NetApp FAS3070 storage systems selected for this SQL Server 2008 data warehouse workload provided more than the required performance for this workload type, with the storage use under 50% of available throughput and capacity.

The IBM x3950 M2 server demonstrated that it embodies the processing power, memory, and I/O bandwidth performance needed to support this large-scale data warehouse workload.

6 ACKNOWLEDGEMENTS

The authors wish to thank IBM and NetApp for providing the hardware equipment that made this work possible. Thanks to the Microsoft SQL Performance Tam for the opportunity to work jointly on SQL Server 2008 prior to its shipping.