

Virtualizing Microsoft[®] SQL Server 2008 on the Hitachi Adaptable Modular Storage 2000 Family Using Microsoft Hyper-V

Reference Architecture Guide

By Eduardo Freitas and Ryan Sokolowski

February 2010



Summary

Server virtualization is enjoying rapid adoption in the 21st-century data center as organizations continue to plan for, validate and realize the benefits this technology can provide. Maintaining acceptable user experience levels, meeting service level agreements and optimizing capital and operational costs are critical objectives for IT administrators. IT administrators often function in more than one role, charged with SQL Server, storage and server virtualization responsibilities. This means that businesses of all sizes need storage solutions for virtualized applications that are easy to deploy and maintain, are highly available, provide flexible scalability, deliver predictable performance, and introduce consolidation-related savings. The solution described in this paper addresses each of these issues for organizations that are challenged by managing underutilized, sprawling servers and storage silos that traditional SQL Server 2008 implementations can generate.

This white paper describes a building block architecture that helps large and enterprise deployments achieve these critical business objectives. It provides best practices required to successfully deploy Microsoft® SQL Server 2008 in a virtualized environment using Microsoft® Hyper-V configurations on the Hitachi Adaptable Modular Storage 2000 family.

For best results use Acrobat Reader 8.0.



Contributors

The information included in this document represents the expertise, feedback and suggestions of a number of skilled practitioners. The authors recognize and sincerely thank the following contributors and reviewers of this document:

- Rick Andersen
- Patricia Brailey
- Steve Burns
- Ron-An Lee
- Lisa Pampuch



Table of Contents

Solution Components	1
Hitachi Adaptable Modular Storage 2000 Family	1
Microsoft Windows Hyper-V and SQL Server 2008	3
Hitachi Performance Monitor Feature	4
Hitachi Tuning Manager Software	4
Reference Architecture	5
Storage Building Block	8
Design Goals	10
Multipathing	11
Disk Management	12
Scaling SQL Server 2008	12
Planning	12
Scaling the Storage	13
Tested Deployment	15
Software	15
Hardware	16
Path Configuration	16
Storage Configuration Standard RAID Groups and LUs	17
Storage Configuration Using Hitachi Dynamic Provisioning Software	18
Lab Validated Results	18
Best Practices	20
Conclusion	20



Virtualizing Microsoft® SQL Server on the Hitachi Adaptable Modular Storage 2000 Family using Microsoft Hyper-V

Reference Architecture Guide

Server virtualization is enjoying rapid adoption in the 21st-century data center as organizations continue to plan for, validate and realize the benefits this technology can provide. As organizations become more familiar, confident and capable with virtual machine deployments, they naturally investigate the feasibility of applying server virtualization to mission-critical applications in pursuit of similar or even enhanced benefits. Databases are arguably vital or nearly all organizations operating in today's global marketplace and Microsoft® SQL Server 2008 is the database server of choice for many of these enterprises.

Maintaining acceptable user experience levels, meeting service level agreements and optimizing capital and operational costs are critical objectives for IT administrators. IT administrators often function in more than one role, charged with SQL Server, storage and server virtualization responsibilities. This means that businesses of all sizes need storage solutions for virtualized applications that are easy to deploy and maintain, are highly available, provide flexible scalability, deliver predictable performance, and introduce consolidation-related savings. The solution described in this paper addresses each of these issues for organizations that are challenged by managing underutilized, sprawling servers and storage silos that traditional SQL Server 2008 implementations can generate.

This white paper describes a building block architecture that helps large and enterprise-scale deployments achieve these critical business objectives. It provides best practices required to successfully deploy Microsoft SQL Server 2008 in a virtualized environment using Microsoft Hyper-V configurations with the Hitachi Adaptable Modular Storage 2000 family.

The solution is intended for use by IT administrators responsible for Microsoft SQL Server 2008, virtualization or storage administration. It assumes familiarity with Microsoft Hyper-V, Hitachi Storage Navigator Modular 2 software, Microsoft Windows Server 2008 and Microsoft SQL Server 2008.

Although the documented solution in this paper uses the Hitachi Adaptable Modular Storage 2300 for testing, any member of the Hitachi Adaptable Modular Storage 2000 family provides a reliable, flexible and cost-effective storage platform for supporting demanding applications like SQL Server 2008.

Solution Components

This section describes the key components of this solution.

Hitachi Adaptable Modular Storage 2000 Family

The Hitachi Adaptable Modular Storage 2000 family is the only midrange storage product with symmetric active-active controllers that provide integrated, automated hardware-based front-to-back-end I/O load balancing. Both controllers in a 2000 family storage system are able to dynamically and automatically assign the access paths from the back of the controller to the logical unit (LU). All LUs are accessible regardless of the physical storage front-end port or the server from which the access is requested. Utilization rates of each controller are monitored so that a more even distribution of workload between the two controllers can be maintained. When coupled with MPIO or Hitachi Dynamic Link Manager's round-robin load balancing policies, the 2000 family eliminates many complex and time consuming path planning tasks that storage administrators typically face.

No other midrange storage product that scales beyond 100TB has a serial attached SCSI (SAS) drive interface. The point-to-point back end design virtually eliminates I/O transfer delays and contention associated with Fibre Channel arbitration and provides significantly higher bandwidth and I/O concurrency.

Although the Hitachi Adaptable Modular Storage 2300 was used in the testing of this solution, the information in this paper is relevant for and the building block approach can be applied to the other 2000 family members with the proper changes to account for capacity and performance differences. The 2000 family is an easy-to-use, scalable, cost effective storage system for mission-critical business applications like SQL Server 2008. It is also a top choice for tiered and standalone storage, consolidation, business continuity, data replication, backup and archiving. The Adaptable Modular Storage 2300 offers a rich set of features in a model that scales to 480 disk drives and delivers enterprise-class performance and capabilities at a modular price. Table 1 lists some of the 2300's specification options. For more information about the other member of the 2000 family, see the [Hitachi Adaptable Modular Storage 2000 family](#) Web site.

Table 1. Hitachi Adaptable Modular Storage 2300 Specification Options

Raw Capacity	472TB SATA 138TB SAS
Available Disk Drives (SAS unless otherwise noted)	146GB (15K RPM) 300GB (15K RPM) 450GB (15K RPM) 600GB (15K RPM) 1TB SATA II (7200 RPM) 2TB SATA II (7200 RPM)
Available Flash Drives (SAS unless otherwise noted)	200GB
Host Interfaces	Fibre Channel: 8Gb/sec iSCSI: GigE
Maximum Host Connections	8 Fibre Channel or 4 iSCSI
Maximum Attached Hosts Through Virtual Ports	2,048
SAS Ports	16
Maximum Number of LUs	4,096
Maximum LU Size	60TB
Controller Cache (per system)	8GB to 16GB

Hitachi Dynamic Provisioning Software

On Hitachi Adaptable Modular Storage 2000 family systems, Hitachi Dynamic Provisioning software provides a dynamic provisioning feature with thin provisioning and wide striping functionalities that provide virtual storage capacity to eliminate application service interruptions, reduce costs and simplify administration, as follows:

- Optimizes or “right-sizes” storage performance and capacity based on business or application requirements.
- Supports deferring storage capacity upgrades to align with actual business usage.
- Simplifies and adds agility to the storage administration process.
- Provides performance improvements through automatic optimized wide striping of data across all available disks in a storage pool.

For more information, see the [Hitachi Dynamic Provisioning Software datasheet](#).

The wide-striping technology that is fundamental to Hitachi Data Provisioning software dramatically improves performance, capacity utilization and management of your environment. By deploying your SQL Server 2008 using V-VOLs from Hitachi Dynamic Provisioning storage pools on the 2000 family, you can expect the following benefits:

- An improved I/O buffer that allows for bursts during peak usage times or intense maintenance activities like content indexing or database integrity checks
- A smoothing effect to the SQL Server workload that can eliminate hot spots across the different RAID groups, resulting in reduce data migration moves related to performance or capacity constraints
- Elimination of excess, unutilized capacity by leveraging the combined capabilities of all disks comprising a storage pool
- Elimination of the need to manage the placement of heavy load databases

Both Hitachi Dynamic Provisioning and standard provisioning environments were tested for this solution.

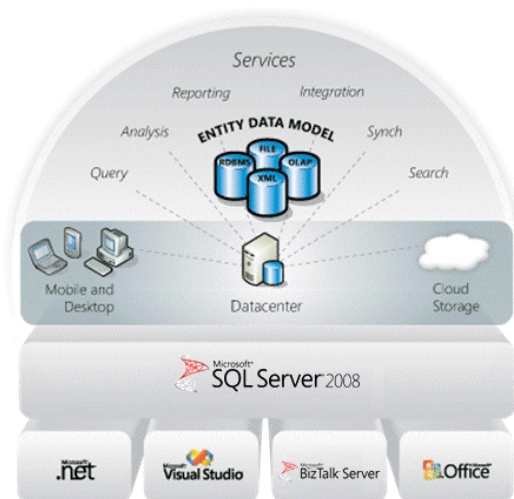
Microsoft Windows Hyper-V and SQL Server 2008

Microsoft Windows Hyper-V is a hypervisor-based virtualization technology that is integrated into Windows Server 2008 x64 and Windows Server 2008 R2 versions of the operating system. It allows for the reduction of hardware footprints and capital expenses through server consolidation.

Alongside native features such as live migration, Hyper-V provides one of the best virtualization solutions that provide high availability out of the box. One of the requirements for such environments is that the SQL Server virtual machines must be hosted in a Hyper-V Failover Cluster configuration.

Microsoft SQL Server 2008 facilitates the management of any data, any place and any time. Together with the 2000 family, SQL Server 2008 provides a scalable, high-performance database engine for any midrange to enterprise level applications that require the highest levels of availability and security, while reducing the total cost of ownership through enhanced enterprise-class manageability for all data management workload deployments at a midrange price. Figure 1 from Microsoft illustrates SQL Server 2008's integration capabilities.

Figure 1. SQL Server 2008 Integration Capabilities



© 2008 Microsoft Corporation. All rights reserved.
Microsoft Data Vision Platform diagram reprinted with permission from Microsoft Corporation.

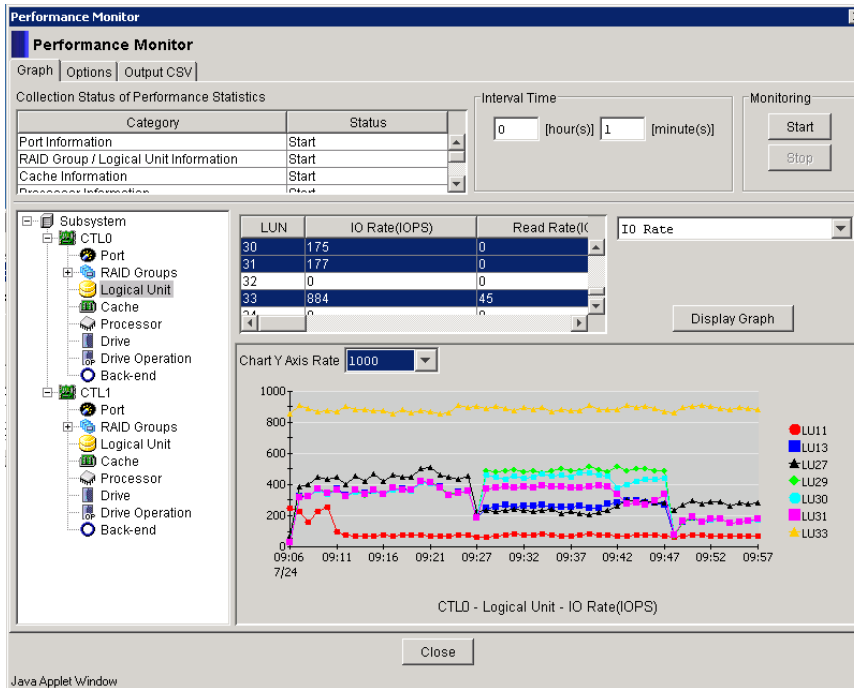
For more information about SQL Server consolidation, see the Microsoft white paper, [SQL Server Consolidation Guidance](#).

Hitachi Performance Monitor Feature

Hitachi Performance Monitor feature is part of the Storage Navigator Modular 2 software package. It acquires information on the performance of RAID groups, LUs and other elements of the storage system. It can also track the utilization rates of resources such as hard disk drives and processors. Information is displayed with line graphs in the Performance Monitor windows or saved in .csv files that you can analyze later. Use Performance Monitor to validate that the underlying storage system is not taxed by the SQL Server environment.

Figure 2 shows the Hitachi Performance Monitor Feature interface.

Figure 2. Hitachi Performance Monitor Feature Interface

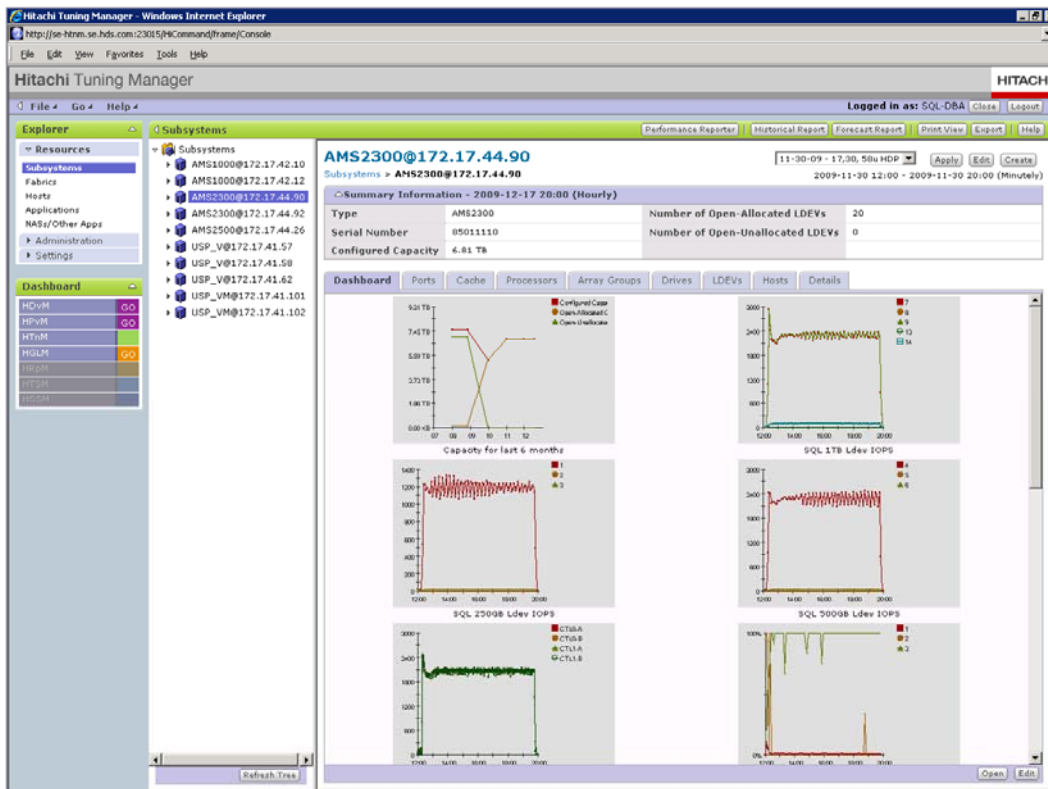


Hitachi Tuning Manager Software

Hitachi Tuning Manager software enables you to proactively monitor, manage and plan the performance and capacity for the Hitachi modular storage that is attached to your Hyper-V servers. Hitachi Tuning Manager software consolidates statistical performance data from the entire storage path. It collects performance and capacity data from the operating system, SQL Server instances, switch ports, and storage system information such as RAID groups and LUs IOPS and latency measures. Hitachi Tuning Manager software provides you with a complete performance picture of the environment. It provides historical, current and forecast views of these metrics. While Hitachi Tuning Manager software is an optional component for your solution, it greatly simplifies the monitoring of your environment by providing a central location to monitor your storage systems, hosts, and applications such as SQL Server 2008. For more information about Hitachi Tuning Manager software, see the [Hitachi Data Systems support portal](#).

Figure 3 shows the Hitachi Tuning Manager software interface.

Figure 3. Hitachi Tuning Manager Software Interface



Reference Architecture

For ease of management, scalability, and to provide predictable performance, this solution uses a building block approach, a familiar concept that Hitachi uses for current and previous generations of SQL Server and the Adaptable Modular Storage 2000 family. A single virtual machine (VM) running Windows Server 2008 and SQL Server 2008 with underlying storage from the Hitachi Adaptable Modular Storage 2300 make up a given building block for purposes of this paper. The 2000 family takes advantages of Hyper-V features such as live migration to establish a reliable and highly available virtualized SQL Server solution.

LUs from the 2300 are allocated to the Hyper-V servers and made available to the virtual machines by using the pass-through disks approach. These disks are automatically made available to the Windows Server 2008 guest operating system and can be partitioned and used in NTFS file systems for the SQL Server 2008 databases, tempdb, transaction logs and BLOB storage, although BLOB storage is outside of this paper's scope. The solution building block supports databases starting at 250GB capacity and scales up to 1TB database capacity. Nevertheless, this is not a limitation of the building block approach as more resources can be assigned to a given environment, both from storage and host perspectives.

Table 2 describes an environment built using the building block architecture that supports three SQL Server instances. This reference architecture uses a 250GB database, a 500GB database and a 1TB database in a standard provisioned environment. The databases all use RAID-1+0 with different drive combinations

depending on their capacity needs. The databases were also built by using a scalable approach from a virtual machine perspective.

Table 2. Reference Architecture Supporting Three Different Size Databases Using Standard LUs

<i>Resource</i>	<i>Details for 250GB</i>	<i>Details for 500GB</i>	<i>Details for 1TB</i>
Virtual machines	16GB memory 1 virtual processor	32GB memory 2 virtual processors	64GB memory 4 virtual processors
Storage system - Hitachi Adaptable Modular Storage 2300	10 x 146GB 15K RPM SAS disks	16 x 146GB 15K RPM SAS disks	30 x 146GB 15K RPM SAS disks
RAID groups	1 x RAID-1+0 (3D+3D) 1 x RAID-1+0 (2D+2D)	1 x RAID-1+0 (6D+6D) 1 x RAID-1+0 (2D+2D)	2 x RAID-1+0 (6D+6D) 1 x RAID-1+0 (3D+3D)
LUs	1 x 300GB on RAID-1+0 (3D+3D) for database files 1 x 30GB on RAID-1+0 (3D+3D) for tempdb files 1 x 80GB on RAID-1+0 (2D+2D) for transaction logs*	1 x 625GB on RAID-1+0 (6D+6D) for database files 1 x 62.5GB on RAID-1+0 (6D+6D) for tempdb files 1 x 160GB on RAID-1+0 (2D+2D) for transaction logs	2 x 625GB one on each RAID-1+0 (6D+6D) for database files 2 x 62.5GB one on each RAID-1+0 (3D+3D) for tempdb files 1 x 250GB on RAID-1+0 (3D+3D) for transaction logs

*Although a 2D+2D RAID group was assigned for transaction logs, the RAID group capacity might not be fully utilized until the user grows the database by another 250GB, in which case an additional LU can be provisioned out of the RAID group. For example, note how the 500GB architecture uses 160GB of the 2D+2D RAID group for transaction logs.

Additional storage was also provisioned to serve as the OS LUs for each of the SQL Server hosts. For that purpose, each SQL Server virtual server was provisioned on a 150GB LU from a common RAID-5 (4+1) group.

When building the same type of environment while using Hitachi Dynamic Provisioning software, a single pool of 14 RAID-1+0 (2+2) groups was created for the SQL Server database, transaction logs and tempdb volumes. Dedicated volumes within the pool simplify backup operations due to their granularity and also provide an easier way to monitor the capacity utilization of each of the volume types. Hitachi recommends using a RAID-1+0 (2D+2D) configuration as the incremental growth for your Dynamic Provisioning pool as it provides good reliability, performance, as well as a good capacity increment for your pool.

Table 3 describes the same configuration described in Table 2 when deployed on a Hitachi Dynamic Provisioning software environment. This configuration supports the same database capacity described in Table 1 while providing the best performance by striping the LUs across all disks in the single Dynamic Provisioning pool. Wide striping reduces or eliminates hot spots caused by skewed workloads while ensuring the same level of RAID protection. In cases where log LUs are less busy than database LUs, I/Os are distributed across all disks and therefore more IOPS capability is available for a given SQL Server instance. RAID protection is provided to the Dynamic Provisioning pool through the use of multiple RAID-1+0 (2D+2D) groups.

Table 3. Reference Architecture Supporting Three Different Size Databases Using Hitachi Dynamic Provisioning Software

<i>Resource</i>	<i>Details for 250GB</i>	<i>Details for 500GB</i>	<i>Details for 1TB</i>
Virtual machines	1 16GB memory 1 virtual processors	1 32GB memory 2 virtual processors	1 64GB memory 4 virtual processors
Storage system - Hitachi Adaptable Modular Storage 2300	11 x RAID-1+0 (2D+2D), 44 x 146 15K RPM SAS disks for the database and tempdb Dynamic Provisioning pool 3 x RAID-1+0 (2D+2D), 12 x 146 15K RPM SAS disks for the transaction log Dynamic Provisioning pool		
LUs	1 x 300GB DP-VOL for database files 1 x 30GB DP-VOL for tempdb files 1 x 80GB DP-VOL for transaction logs	1 x 625GB DP-VOL for database files 1 x 62.5GB DP-VOL for tempdb files 1 x 160GB DP-VOL for transaction logs	1 x 1.2TB DP-VOL for database files 1 x 125GB DP-VOL for tempdb files 1 x 250GB DP-VOL for transaction logs

Notice that the same number of disks was used for both the standard (Table 2) as well as the Dynamic Provisioning configuration (Table 3). Also notice that the Dynamic Provisioning pool was created utilizing RAID-1+0 (2+2) groups. While it is possible to create a pool with other RAID group types supported by the 2000 family, the RAID-1+0 (2+2) provides the required protection level that Microsoft recommends for a SQL Server environment and a way to provide a small scalability increment for the pool. When increasing the amount of SQL Server virtual machines you have the choice to either create another pool or to increase the capacity of the existing pool. It is important to note that when increasing the capacity of a given pool, it is beneficial to optimize the LUs on that given pool. Optimizing a pool ensures that all previously written data within a pool is appropriately distributed across all of the available spindles within a pool. It is also important to note that this action may cause increased latency between the storage and the host as the storage system is moving data within the pool to a new location.

For scalability purposes on Hitachi Dynamic Provisioning environments, one can also create a third pool that can host the virtual machine OS. As with the data pools, this approach enables you to more efficiently scale up your architecture while requiring only the utilized space and making use of all the spindles within a pool.

The Hyper-V servers used in this test environment contain 128GB of memory and four quad-core AMD Opteron CPUs. If you require a high availability environment, use a minimum of two Hyper-V servers. Additional SQL Server VMs can be added to the Hyper-V server along with additional RAID groups on a 2000 family storage system to scale up to larger databases. To scale beyond the amount of resources a given Hyper-V server can provide to an environment, additional Hyper-V servers must be added, ensuring that sufficient Hyper-V servers are employed to achieve redundancy and high availability.

This paper includes details regarding the test environment and methods used to validate the SQL Server 2008 instances and storage infrastructure. For more information about how to deploy and configure the environment, see the *Virtualizing SQL Server on the Hitachi Adaptable Modular Storage 2000 Family Using Hyper-V Implementation Guide* on the Hitachi Data Systems [SQL Solutions Web site](#).

This solution focuses specifically on best practices for deploying SQL Server OLTP databases on the 2000 family. However, you can deploy other types of databases in this environment after conducting testing to ensure all components are provisioned appropriately to ensure performance.

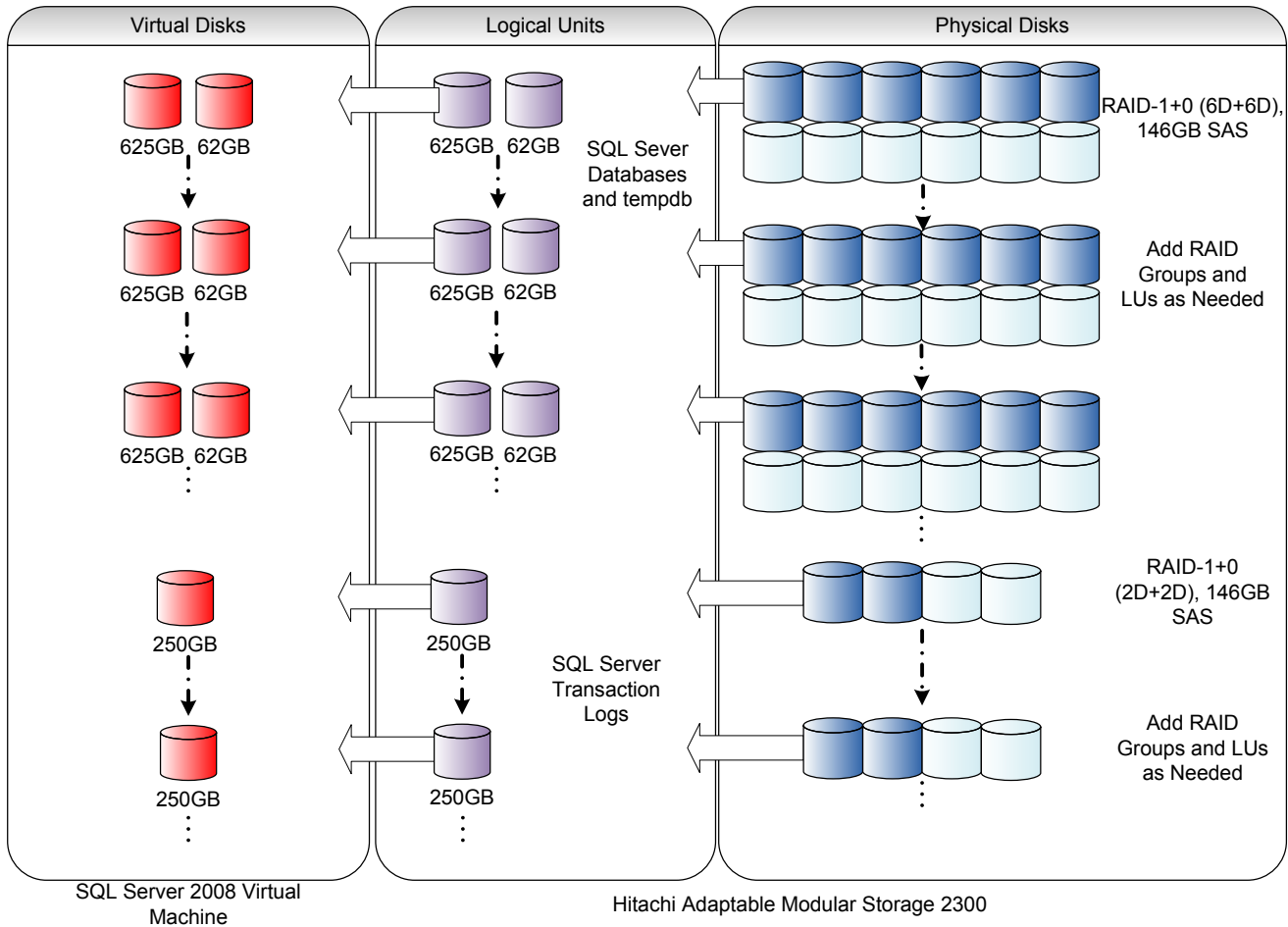
Storage Building Block

From a storage perspective, designing a SQL Server implementation in a virtual server environment is no different than performing the same activities in a non-virtualized environment. However, from a server perspective, you are required to perform a few additional tasks when deploying a virtual SQL Server environment compared to a non-virtualized SQL Server environment. Nevertheless, after the virtualized environment is set up, it is easy to manage using the native Hyper-V Manager or System Center Virtual Machine Manager 2008 R2 on more complex environments. Deploying SQL Server 2008 using a building block approach allows you to easily manage and scale your environment. Additional VMs are easy to deploy using templates, and storage can be provisioned on a per-RAID group basis. This paper provides a building block approach that can be utilized depending on the capacity needed for your environment. While the three reference architectures differ in capacity, they all use the same underlying building block. All the RAID groups and LUs for the environment were built following these guidelines:

- For standard provisioned environments, place database and log files on physically separate RAID groups.
- For standard provisioned environments tempdb files can be placed on the same RAID group as the database files. However, place them on a separate LU within the RAID group for performance and ease of monitoring. Sharing the same LU for data files and tempdb can negatively affect performance.
- For the best performance and availability, use RAID-1+0 for the RAID groups for your SQL environment. Let your performance requirements for your environment determine the best combination of drives for a given RAID group, followed by the specific capacity.
- From a capacity perspective, always account for approximately 20 percent additional overhead for the database, tempdb, and transaction logs LUs. Additional capacity may also be required for your transaction logs depending on your backup recovery model.
- Ensure that any additional capacity added to the environment follows the same RAID-1+0 drive combination. This ensures that performance requirements continue to be met as additional capacity is needed.
- SQL Server virtual machines CPU and memory requirements must be appropriately scaled for the expected database workloads.

With these guidelines in mind, you can build a scalable architecture that meets your performance and capacity requirements and ensures that you have a scalable and highly available environment. Figure 4 shows how an architecture built on an Adaptable Modular Storage 2300 that starts with a 500GB capacity can scale up while following the building block guidelines.

Figure 4. Physical to Virtual Disk Mapping of the 500GB Building Block Architecture




LUs are presented to the Hyper-V server and made available to the VMs through the pass-through disk capability. Note that when deploying environments that utilize Microsoft Cluster Services, Hitachi recommends that you use the LUs' GUIDs when mapping them to the virtual machines, rather than using the disk number. For more information on best practices and deployment guidelines see the [Optimizing High Availability and Disaster Recovery with Hitachi Storage Cluster in Microsoft® Virtualized Environments with Live Migration](#) white paper.

Note that a minimum of 20 percent additional capacity is accounted for on each of the LUs, as per Microsoft's best practice recommendations. Additional capacity is also provided to ensure that the architecture continues to meet performance requirements for the environment. Also note that as capacity increases, it might be necessary to increase the amount of virtual processors and memory for a SQL Server virtual machine to maintain the same performance levels achieved for your initial building block.

The Hitachi Dynamic Provisioning software architecture uses 14 RAID-1+0 (2D+2D) groups in two separate Dynamic Provisioning pools. One pool is used for the user database and tempdb files, the other for the transaction log files. Each pool has independent LUs for each of the file types for the maximum protection level. Although a single Dynamic Provisioning pool can be created for database and transaction log files, a dual Dynamic Provisioning pool architecture ensures that the same level of protection you have on a standard provisioned environment at the RAID group level is also provided on a Hitachi Dynamic Provisioning software environment. For more information on how the two pools are configured for an environment using Hitachi Dynamic Provisioning software, see Table 3.

The Hitachi Dynamic Provisioning building block is also built following the guidelines described in the previous section.



Storage for the SQL Server databases is configured first for performance and capacity requirements for the medium to high performance levels that a given SQL Server VM can handle. This is no different than what is done when sizing SQL Server storage for physical server deployments. The architectures described in this section require the following calculations be derived from the requirements for an existing or planned SQL Server 2008 environment:

- Overall capacity needed for user databases and transaction logs including planned growth for SQL Server instances
- IOPS needed for SQL Server user databases and transaction logs
- SQL Server instance host CPU and memory requirements for expected database workloads

Design Goals

This solution's building block architecture achieves the following design goals for an OLTP workload:

- Reach and maintain a minimum of 50 percent CPU utilization throughout the OLTP workload applied to the provisioned SQL Server 2008 VMs.
- Optimize storage configuration on the 2000 family storage system for best I/O throughput and ease of management for both standard and Hitachi Dynamic Provisioning configurations.
- Deliver sustainable and acceptable levels of IOPS falling within the 1ms to 20ms response time range for data and transaction logs transactions file I/O. Testing was conducted through the use of an OLTP workload environment simulating a stock brokerage scenario and measures were reported by the Windows Performance Monitor as well as SQL Server Management Studio reports accommodating burst and peak activities. SQL Server Dynamic Management Views (DMVs) were used to determine if the SQL Server database configuration was able to meet Microsoft's performance recommendations.
- Deliver at least 80 percent disk capacity utilization for the database volumes.

Several other factors contributed to the considerations and trade-offs to arrive at the standard provisioned building block architecture, including these:

- For a very heavy user profile, you might need additional disks available to support the required IOPS in standard provisioned environments. As with any SQL Server deployment, make sure to test your environment and ensure the proper amount of disks are provisioned to the database from an IOPS perspective.
- Granularity of scale is determined by the database size along with the SQL Server virtual server provisioned hardware; additional storage can be provisioned by adding RAID-1+0 groups for database and transaction logs. The specific drive combination for each RAID group depends on the initial building block environment that meets both performance and capacity requirements for your environment.

For example, in a 500GB database configuration, a RAID-1+0 (6D+6D) group is used to host both the database as well as the tempdb LUs while a RAID-1+0 (2D+2D) group is used to host the transaction logs LU. To increase the capacity of the database by 500GB in a standard provisioned environment, create an additional RAID-1+0 (6D+6D) group and provision LUs to host the additional database files and possibly additional tempdb volumes, if the number of tempdb files are not aligned with the number of CPU cores due to the increase of resources made available to a SQL Server VM. In this case, an additional RAID group for the transaction logs is not necessary due to the unused capacity still available from the original RAID group provisioned for this purpose. An additional increase of 500GB to the database capacity requires two additional RAID groups, one RAID-1+0 (6D+6D) group for the database volumes and one RAID-1+0 (2D+2D) group for the transaction logs.

In a dynamic provisioned configuration, Dynamic Provisioning pool space can be allocated by adding one RAID-1+0 group to the Dynamic Provisioning pool. After that, the required LUs can be created and appropriately assigned to the Hyper-V server.

- Devise an appropriate database growth plan Remember that each increment includes one LU for the database files, one LU for the transaction logs, along with possibly one LU for tempdb depending on your workload profile and usage of tempdb.

Multipathing

To maintain a constant connection between the Hyper-V server and storage, Windows 2008 supports multipathing. Multipathing allows multiple physical or logical connections from the host to the storage. To support multipathing, Hitachi Data Systems recommends that the physical host contain at least two HBAs that connect to at least two Fibre Channel ports, one on each storage system controller.

Several types of multipathing policies are available through the Windows Multi Path I/O software:

- **Failover only** — Uses a single active path, and the rest of the paths are standby paths. Normally used on storage systems that does not possess the active/active capability.
- **Round robin** — Allows the Device Specific Module (DSM) to use all available paths for MPIO in a balanced way. This is the default MPIO policy and it can be used with the 2000 family no matter what controller/ports are mapped to the Hyper-V server due to the storage's active/active capability.
- **Round robin with subset** — Allows the application to specify a set of paths to be used in a round robin fashion, and with a set of standby paths.
- **Least queue depth** — Sends I/O down the path with the fewest currently outstanding I/O requests.
- **Weighted paths** — Assigns a weight to each path. The weight indicates the relative priority of a given path. The larger the number, the lower ranked the priority
- **Least blocks** — Sends I/O down the path with the least number of data blocks currently being processed.

For more information about MPIO and its load balancing policies, see the Microsoft TechNet article, [“Understanding MPIO Features and Components.”](#)


Hitachi Dynamic Link Manager software possesses the same capabilities as the native MPIO with the addition of some algorithms that you may find to be more suitable for your environment such as the extended round-robin algorithm. Hitachi Dynamic Link Manager Advanced software provides the ability to manage and monitor your links from a central location.

Hitachi Dynamic Link Manager software has two categories of algorithms which differ in their processing method. The following describes both of these processing methods:

- Round robin, least I/Os and least blocks algorithms select which path to use every time an I/O is issued. The path that is used is determined by the following:
 - **Round robin** — The paths are simply selected in order from among all the connected paths.
 - **Least I/Os** — The path that has the least number of I/Os being processed is selected from among all the connected paths.
 - **Least blocks** — The path that has the least number of I/O blocks being processed is selected from among all the connected paths.
- Extended round robin, extended least I/Os and extended least blocks algorithms determine which path to allocate based on whether the data of the I/O to be issued is sequential with the data of the I/O that was issued immediately beforehand.

If the data is sequential, the path used is the one to which the data of the I/O that was issued immediately beforehand was distributed. However, if a specified number of I/Os have been issued to a path, processing switches to the next path. If the data is not sequential, these algorithms select the path to be used each time an I/O request is issued. The path that is used is determined by the following:

- **Extended round robin** — The paths are simply selected in order from among all the connected paths.
- **Extended least I/Os** — The path that has the least number of I/Os being processed is selected from among all the connected paths.
- **Extended least blocks** — The path that has the least number of I/O blocks being processed is selected from among all the connected paths.



Hitachi Data Systems testing shows that when using either MPIO or Hitachi Dynamic Link Manager software, the round-robin policy or algorithm is the best choice for a SQL Server environment.

Disk Management

Hyper-V servers can access LUs in two ways, through the pass-through disks approach, or through the use of virtual hard disks (VHDs) made available on a given LU. Pass-through disks enable the LUs to be available to a virtual machine as if they were directly connected to it. The pass-through disks are presented to the virtual machine and used as containers to house the virtual machine OS VHD file or additional data files.

Pass-through disks have the following characteristics:

- Must be in the offline state from the Hyper-V parent perspective, except in the case of clustered or highly available virtual machines.
- Presented as raw disk to the Hyper-V host partition.
- Cannot be dynamically expanded, meaning that the disks cannot be dynamic.
- Do not allow the capability to take snapshots or utilize differencing disks.
- Easier to scale to a larger number of virtual machines utilizing pass-through disks because pass-through disks do not require drive letters. The raw disk is formatted and assigned a volume label and drive letter in the guest virtual machine partition.

With VHDs, all I/O goes through two complete storage stacks, once in the guest (or child) virtual machine partition and once in the Hyper-V host (or parent) partition. The guest application disk I/O request goes through the storage stack within the guest OS and onto the Hyper-V parent partition file system. Microsoft recommends the use of fixed VHDs because the use of dynamic VHDs increases the likelihood of negative performance effects.

When using the pass-through disk feature, the NTFS file system on the parent partition can be bypassed during disk operations, minimizing CPU overhead and maximizing I/O performance. With pass-through disks, the I/O traverses only one file system, the one in the child partition, and offer higher throughput because only one file system is traversed, thus requiring less code execution.

Hitachi Data Systems recommends using pass-through disks when hosting applications with high storage scalability and performance requirements.


Scaling SQL Server 2008

This section describes planning and deployment considerations to keep in mind when scaling your SQL Server environment.

Planning

Scaling SQL Server databases and instances requires planning and testing both from a server as well as a storage perspective. When scaling up, both capacity and performance are concerns; any production environment must be properly tested to ensure it satisfies end-user requirements. From a Hyper-V perspective, you must calculate the maximum number of VMs that a single Hyper-V server can support by noting the total number of CPUs and total memory requirements. Best practice is to not over-allocate the resources available on your physical Hyper-V server. Microsoft recommends deploying a single instance of SQL Server per virtual machine for improved performance and security isolation. Deploy additional virtual machines if you need more SQL Server instances to support your environment. Note that at this point, Hyper-V is limited to 64 VMs per clustered node and 384 VMs per host in standalone environments.

Note that resources that are allocated to a given VM are allocated regardless of whether or not they are fully utilized. Also keep in mind that the host operating system also requires an additional allocation of resources, although it is minimal. Assign one virtual processor per physical CPU core. If you decide to over-commit your physical processors, frequently monitor the performance of your Hyper-V server. As in a physical server, take both CPU and memory requirements into consideration when planning to deploy SQL Server on Hyper-V. Keep



in mind that while you are able to allocate additional memory and virtual processors for a given VM, you might need to shut down the VM prior to doing so.

Scaling the Storage

When starting with a database in which the performance and capacity requirements fit within the described 500GB reference architecture, RAID-1+0 (6D+6D) groups can be added to the environment as a scaling step. The same can be done from the reference architectures for the 250GB and 1TB, although the 250GB architecture is based on a RAID-1+0 (3D+3D) group. From a database perspective, Hitachi Data Systems lab tests show that each of the reference architectures described on this paper appropriately scales the amount of IOPS from one configuration to the next while maintaining latency values at or below levels where Microsoft suggests an I/O bottleneck might affect performance.

You can mix various drive combinations and capacity values with this architecture as long as you follow the storage block building guidelines described in the “Storage Building Block” section of this paper.

In a dynamic provisioned environment, additional LUs can be provisioned and assigned to a given port as with the standard provisioned environment. In cases where the overall Dynamic Provisioning pool might be nearing its limit, Hitachi Data Systems recommends creating a second Dynamic Provisioning pool to ensure full utilization of the wide striping capabilities of Hitachi Dynamic Provisioning software. Nevertheless, if a database that is known to be less active is being deployed in the same Hyper-V environment, you can choose to increase the capacity of the pools currently being used. This can provide the ability to diminish hot spots within the pool while maintaining the performance levels for both the previously deployed environment as well as the new one.

A Dynamic Provisioning pool is composed of one or more dynamic provisioning RAID groups. When increasing your pool capacity, you increase it by a dynamic provisioning RAID group unit at a time. Because new added dynamic provisioning RAID groups are not used at first, the Dynamic Provisioning pool is unbalanced, as the data stripes are not spread throughout all the spindles within the pool.

Hitachi provides a Dynamic Provisioning optimization feature that solves unbalanced conditions within a pool by optimizing the Dynamic Provisioning pool and reclaiming zero pages within that pool. Execute the optimization carefully as it can affect performance depending on how heavily the pool is currently being used. This is due to the nature of the optimization process, which moves data from their original location to spread the data across all available spindles within a Dynamic Provisioning pool.

Note that transaction log LUs also follow the same growth factor. You typically need to account for the sum of the transaction log LUs' capacity to be between five and 10 percent of the space provisioned for database LUs.

Table 4 shows the resources used for the building block architecture with a 500GB capacity.

Table 4. Building Block Resources for 500GB SQL Server

<i>Resource</i>	<i>Details</i>
Virtual machines	1 32GB memory 2 x virtual processors
Storage systems	Hitachi Adaptable Modular Storage 2300 23 x 146GB 15K RPM SAS disks Note: 2 disks are used as hot spares
RAID groups	1 x RAID-1+0 (6D+6D) 1 x RAID-1+0 (2D+2D) 1 RAID-5 (4D+1P)
LUs	1 x 625GB on RAID-1+0 (6D+6D) for database files 1 x 62.5GB on RAID-1+0 (6D+6D) for tempdb files 1 x 160GB on RAID-1+0 (2D+2D) for transaction logs 1 150GB on RAID-5 (4D+1P) for VM OS*

* The remainder of the space within the RAID-5 (4D+1P) group was utilized for additional virtual machines that were hosted on the same Hyper-V server.

Like any other SQL Server deployment, always test before deploying your production environment. Remember that additional adjustments for things such as unanticipated growth, protection methods and service level agreements might become necessary. Hitachi Data Systems used industry standard OLTP workloads that pushed the virtual servers to 60 percent or more of their “% Processor time” in Windows Performance Monitor while maintaining latency levels at or below Microsoft suggests an I/O bottleneck might affect performance.

For more information about monitoring your SQL Server environment and Adaptable Modular Storage 2000 family, see the Performance Monitoring section of the [Hitachi Adaptable Modular Storage 2000 Family Best Practices with Microsoft® SQL Server for OLTP Applications](#) white paper.

A second 2300 might be needed for more complex environments in which a failover capability is necessary to meet your business and regulatory requirements. Hitachi Storage Cluster, Failover Clustering, and Hyper-V quick or live migration allow you to automatically or manually failover SQL Servers or other virtual machines to an additional Hyper-V Cluster node and Hitachi Adaptable Modular Storage 2300 system. Note that live migration is only available on Windows Server 2008 R2.

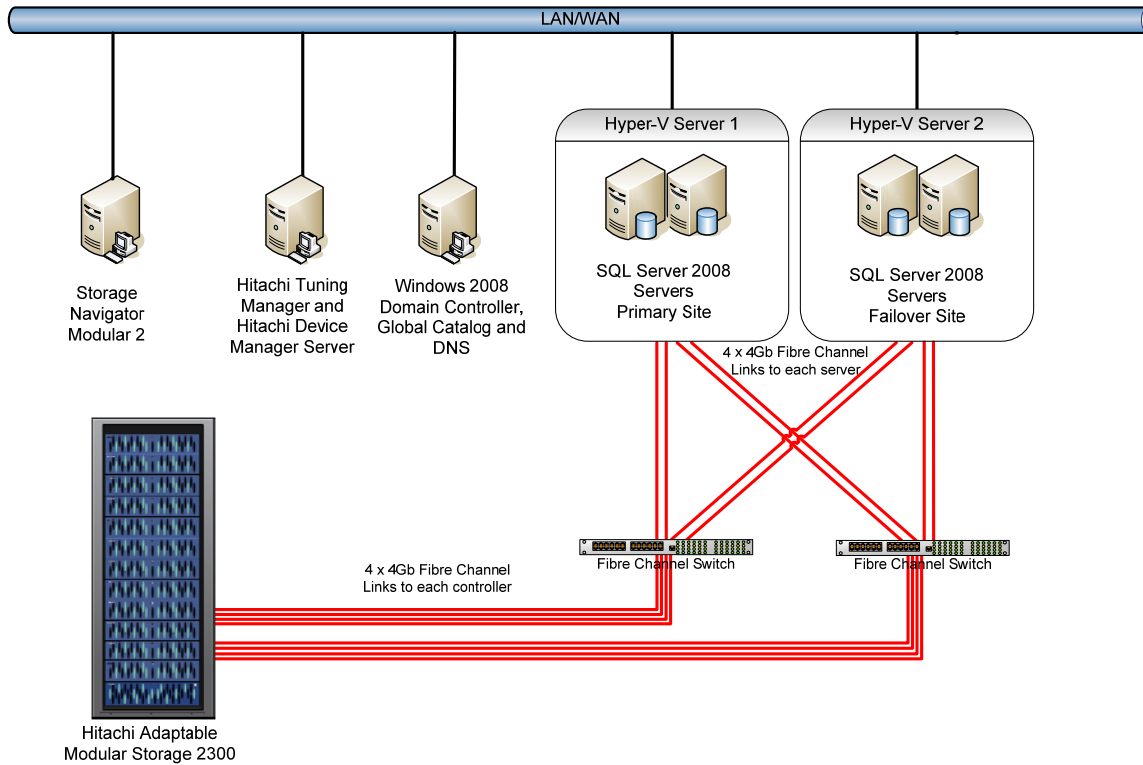
For more information, see the [Optimizing High Availability and Disaster Recovery with Hitachi Storage Cluster in Microsoft® Virtualized Environments with Live Migration](#) white paper.

Windows 2008 and Hitachi tools enable you to fully monitor the resource utilization on your environment. Windows Performance Monitor allows you to monitor the CPU and memory utilization of the Hyper-V hosts as well as the guest machines through the use of Performance Monitor on the virtual machine’s OS. On the storage side, Hitachi Tuning Manager provides a holistic view of all the Adaptable Modular Storage 2000 family performance related counters. This enables you to closely monitor the system to ensure that your environment is utilizing all available resources. You can also identify potential bottlenecks that might require you to add disks to the environment.

Tested Deployment

The following sections describe the hardware and software used in the deployment tested by Hitachi Data Systems. Figure 5 illustrates the test environment topology.

Figure 5. Test Environment Topology



Software

Table 5 describes the software required for the tested deployment described in this white paper.

Table 5. Software Resources

<i>Software</i>	<i>Version</i>
Windows Server 2008 (for Hyper-V server)	Datacenter edition, R2
Windows Server 2008 (for SQL Server virtual machines)	Enterprise edition, Service Pack 2
SQL Server 2008	Enterprise edition, Service Pack 1
Hitachi Storage Navigator Modular 2	7.0
Hitachi Dynamic Provisioning license*	Microcode dependent
Hitachi Device Manager**	6.2.0-00
Hitachi Tuning Manager**	6.2.0-00

*Note: To use Hitachi Dynamic Provisioning software in this solution, the license for the Hitachi Dynamic Provisioning software is also required. The license can be installed using Hitachi Storage Navigator Modular 2 software's GUI or command-line interface.

**Note: While Hitachi Device Manager software and Hitachi Tuning Manager software are not required for deployment of this solution, they provide features that enable you to more effectively manage the storage system and monitor the performance of your environment.

Hardware

The solution building block is designed to optimize resource utilization and throughput of the 2300 and the Hyper-V server. Table 6 lists the hardware used in testing this architecture.

Table 6. Hardware Resources

<i>Hardware</i>	<i>Configuration</i>
Hitachi Adaptable Modular Storage 2300 storage system	Microcode 0883A-M 2 controllers 4 disk trays 15 146GB 15K RPM SAS disks per tray 16GB cache per controller
Two Brocade 5100 SAN switches	FOS 6.21* 16 4Gb Fibre Channel ports
Two Dell R905 servers	BIOS firmware 4.0.3** 4 quad core AMD Opteron 1.9GHz processors 128GB memory 2 Emulex LPe11002 4Gb Fibre Channel HBAs

*Use the latest or the most recent Hitachi supported levels of microcode and FOS levels

**Use the latest available Hitachi and Microsoft supported BIOS firmware level.

Path Configuration

All LUs are masked and zoned to all four HBAs on each of the Hyper-V servers using four dedicated Fibre Channel ports on the 2300. Note that while four dedicated ports from the host and four from the storage system were utilized for each host to storage connection on the test, it is possible to use two ports from the host and two from the storage system without reaching the IOPS capability of a given port and therefore not creating a bottleneck in the environment.

Table 7 lists the connections between the Hyper-V servers and the storage system ports.

Table 7. Path Configuration

<i>Hyper-V Server</i>	<i>Host HBA Number</i>	<i>Switch</i>	<i>Zone Name</i>	<i>Storage System Port</i>	<i>Storage System Host Group</i>
Hyper-V Server 1	HBA 1 port 1	Brocade 5100 - 1	HyperV_1_HBA1_1_AMS2300_0A	0A	HyperV_1_HBA1_1
Hyper-V Server 1	HBA 1 port 2	Brocade 5100 - 2	HyperV_1_HBA1_2_AMS2300_1A	1A	HyperV_1_HBA1_2
Hyper-V Server 1	HBA 2 port 1	Brocade 5100 - 1	HyperV_1_HBA2_1_AMS2300_0B	0B	HyperV_1_HBA2_1
Hyper-V Server 1	HBA 2 port 2	Brocade 5100 - 2	HyperV_1_HBA2_2_AMS2300_1B	1B	HyperV_1_HBA2_2
Hyper-V Server 2	HBA 1 port 1	Brocade 5100 - 1	HyperV_2_HBA1_1_AMS2300_0C	0C	HyperV_2_HBA1_1
Hyper-V Server 2	HBA 1 port 2	Brocade 5100 - 2	HyperV_2_HBA1_2_AMS2300_1C	1C	HyperV_2_HBA1_2
Hyper-V Server 2	HBA 2 port 1	Brocade 5100 - 1	HyperV_2_HBA2_1_AMS2300_0D	0D	HyperV_2_HBA2_1
Hyper-V Server 2	HBA 2 port 2	Brocade 5100 - 2	HyperV_2_HBA2_2_AMS2300_1D	1D	HyperV_2_HBA2_2

On each Hyper-V server, the round robin multipath policy was used on MPIO for all LUs presented to the hosts. Note that you can also use Hitachi Dynamic Link Manager software with round robin algorithm. Hitachi Dynamic Link Manager Advanced software provides a central location for management and monitoring of all of your Hyper-V host links to the 2000 family.

Storage Configuration Standard RAID Groups and LUs

This section describes how storage was configured for a single building block for the standard provisioned tested deployment.

Figure 6 shows the building block for the 500GB capacity environment.

Figure 6. Standard Provisioned 500GB Storage Building Block

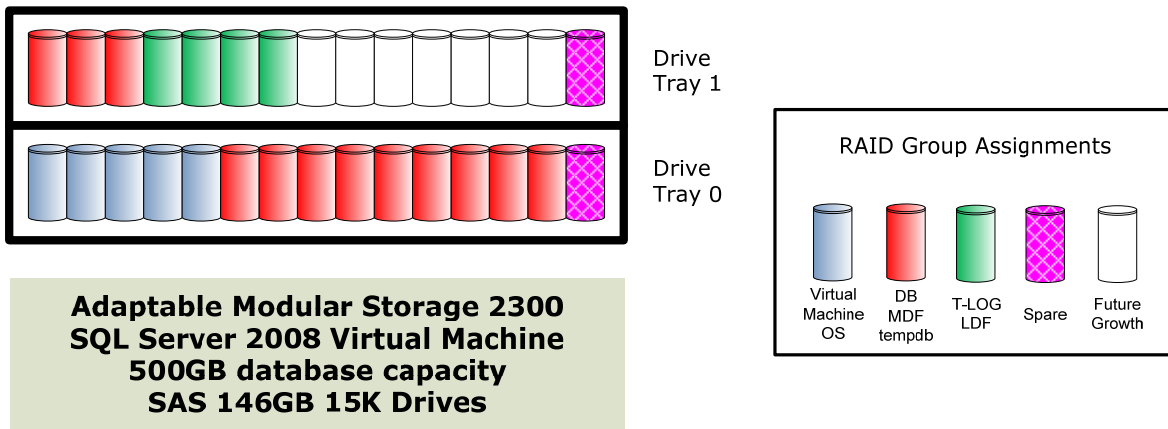


Table 8 shows the building block configuration for the 500GB environment.

Table 8. Detailed Storage Configuration for a Single 500GB Building Block

<i>RAID Group</i>	<i>LUN</i>	<i>Size (GB)</i>	<i>RAID Level</i>	<i>RAID Type</i>	<i>Disk Spec</i>	<i>Description</i>
1	1	150	RAID-5	4+1	146GB 15K	OS LU
2	1	625	RAID-1+0	6+6	146GB 15K	Database LU
2	2	62.5	RAID-1+0	6+6	146GB 15K	tempdb LU
3	3	160	RAID-1+0	2+2	146GB 15K	Transaction logs LU

Storage Configuration Using Hitachi Dynamic Provisioning Software

This virtualized SQL Server solution uses one Hitachi Adaptable Modular Storage 2300 with the Hitachi Dynamic Provisioning software configuration for three different size databases provisioned to three different SQL Server VMs. Each VM was also configured in a building block approach. The RAID group that hosts the VM's OS resides on a separate RAID-5 (4+1) group.

Using Hitachi Dynamic Provisioning software, the I/O workload is distributed across all drives in the Dynamic Provisioning pool, which consists of 14 RAID-1+0 (2+2) groups. All LUs for the building block using Hitachi Dynamic Provisioning software are allocated from two Dynamic Provisioning pools, one for database and tempdb LUs and another for the transaction logs LUs.

Table 9 lists the configuration of RAID groups and LUs for a single building block using Hitachi Dynamic Provisioning software. Note that all storage is in Dynamic Provisioning pool 0.

Table 9. Detailed Storage Configuration for a Single Building Block using Hitachi Dynamic Provisioning Software

<i>LU</i>	<i>Size</i>	<i>Description</i>
4	300GB	Database LU for SQL Server 1
5	30GB	tempdb for SQL Server 1
6	80GB	Transaction logs for SQL Server 1
7	625GB	Database for SQL Server 2
8	62.5GB	tempdb for SQL Server 2
9	160GB	Transaction logs for SQL Server 2
10	1.2TB	Database for SQL Server 3
11	125GB	tempdb for SQL Server 3
12	250GB	Transaction logs for SQL Server 3

Note that this table does not include the RAID group used to store the virtual machine's OS, which was located on a RAID-5 (4D+1P) group.

Lab Validated Results

An industry standard OLTP workload that simulates a stock brokerage scenario was exercised on each of the building block architectures and then additional SQL Server VMs and storage building blocks were added to scale up the configuration. Results were collected and the number of building blocks increased until the design goal was achieved. The results show that as each architecture was scaled up, the IOPS levels at a minimum

doubled at each increment while maintaining the latency levels at or below Microsoft best practice recommendations.

Table 10 lists the OLTP workload parameters used as a baseline for the tests.

Table 10. OLTP Workload Test Parameters

<i>Parameter</i>	<i>Description</i>
Test scenario	OLTP workload for a stock brokerage firm
Virtual machine CPU utilization	≥50%
Minimum database LU capacity usage	80%
Minimum individual disk busy rate	50%
Number of active users	Variable, scaling with architecture size
Test type	Performance
Test duration	≥3 hours

Table 11 describes the success criteria established by Microsoft’s best practice recommendations and monitored using Windows Performance Monitor.

Table 11. OLTP Workload Success Criteria and Test Results

<i>Description</i>	<i>Criteria</i>	<i>Results</i>
Average achieved database IOPS	Varies	
For non dynamic provisioning environment		
250GB architecture		1083IOPS
500GB architecture		2100IOPS
1TB architecture		4177IOPS
For Dynamic Provisioning environment		
250GB architecture		1240IOPS
500GB architecture		2310IOPS
1TB architecture		4660IOPS
Database Avg. Disk sec/read	≤ 20ms	Average 10-20ms
Disk sec/Write	≤ 20ms	Maximum 20ms
Transaction Log Avg. Disk sec/Read	≤ 5ms	Average 1-5ms
Avg. Disk sec/Write	≤ 5ms	Maximum 5ms

Testing shows that this solution meets or exceeds all design goals for the building architectures.

Best Practices

These best practices for the design of the virtualized SQL Server 2008 environment are based on Hitachi Data Systems testing of the building block architecture. For a deployment of SQL Server 2008 using Microsoft Hyper-V and one of the Adaptable Modular Storage 2000 family storage systems, follow these guidelines:

- For enhanced storage utilization and usability, use Hitachi Dynamic Provisioning software.
- When using standard LUs, use RAID-1+0 (6D+6D) for SQL Server database LUs and RAID-1+0 (2D+2D) for SQL Server transaction logs LUs for the a 500GB building block.
- Set tempdb file growth to a reasonable value as per Microsoft's best practices. Setting the file growth to small values might negatively affect the performance of your environment. For more information, see the [Optimizing tempdb performance](#) Books Online Web site.
- When using dynamically provisioned LUs:
 - Use RAID-1+0 for all pools.
 - Adjust the size of the RAID group for scalability. When using 146GB disks, 2D+2D RAID groups provide the best scaling option for SQL Server 2008 environments.
 - Create separate pools for database and transaction log LUs.
- Use a minimum of two Hyper-V servers to take full advantage of Hyper-V features such as Hitachi Storage Cluster with live migration.
- Configure at least two redundant paths to the Adaptable Modular Storage 2000 from each Hyper-V server.
- Use the building block architecture for easy scaling.
- Use round robin multipathing policy with MPIO or the round robin algorithm with Hitachi Dynamic Link Manager to take best advantage of the Adaptable Modular Storage 2000's symmetric active-active front end ports.
- Use pass-through disks for your SQL Server data.

Conclusion

This white paper describes building block architecture that leverages the Hitachi Adaptable Modular Storage 2000 family and Microsoft Hyper-V to enable business of all sizes obtain storage solutions for virtualized applications that are easy to deploy and maintain, are highly available, provide flexible scalability, deliver predictable performance, and introduce consolidation-related savings. The solution described in this paper addresses each of these issues for organizations that are challenged by managing underutilized, sprawling servers and storage silos that traditional SQL Server 2008 implementations can generate.

For more information about Hitachi products, contact your sales representative or channel partner or visit the [Hitachi Data Systems Web site](#).



Corporate Headquarters 750 Central Expressway, Santa Clara, California 95050-2627 USA
Contact Information: + 1 408 970 1000 www.hds.com / info@hds.com

Asia Pacific and Americas 750 Central Expressway, Santa Clara, California 95050-2627 USA
Contact Information: + 1 408 970 1000 www.hds.com / info@hds.com

Europe Headquarters Sefton Park, Stoke Poges, Buckinghamshire SL2 4HD United Kingdom
Contact Information: + 44 (0) 1753 618000 www.hds.com / info.uk@hds.com

Hitachi is a registered trademark of Hitachi, Ltd., in the United States and other countries. Hitachi Data Systems is a registered trademark and service mark of Hitachi, Ltd., in the United States and other countries.

All other trademarks, service marks and company names mentioned in this document or Web site are properties of their respective owners.

Notice: This document is for informational purposes only, and does not set forth any warranty, expressed or implied, concerning any equipment or service offered or to be offered by Hitachi Data Systems. This document describes some capabilities that are conditioned on a maintenance contract with Hitachi Data Systems being in effect and that may be configuration dependent, and features that may not be currently available. Contact your local Hitachi Data Systems sales office for information on feature and product availability.

© Hitachi Data Systems Corporation 2010. All Rights Reserved.
AS-033-00 February 2010