# SQL Server High Availability and Disaster Recovery for SAP Deployment at QR: A Technical Case Study

**Writer:** Danny Tambs (Microsoft)

**Contributor:** Sanjay Mishra (Microsoft)

**Technical Reviewers:** Prem Mehra (Microsoft), Juergen Thomas (Microsoft), Tim Robinson (QR), Chris Ludbey (QR)

**Published:** September 2010

**Applies to:** SQL Server 2005

**Summary:** This paper describes the high availability and disaster recovery solution implemented at QR to support the mission critical SAP application.

***Note****: This document is current as of its most recent publish date of September 15, 2010. Be sure to check www.sqlcat.com to obtain the most recent version of this paper.*

Copyright

The information contained in this document represents the current view of Microsoft Corporation on the issues discussed as of the date of publication. Because Microsoft must respond to changing market conditions, it should not be interpreted to be a commitment on the part of Microsoft, and Microsoft cannot guarantee the accuracy of any information presented after the date of publication.

This white paper is for informational purposes only. MICROSOFT MAKES NO WARRANTIES, EXPRESS, IMPLIED, OR STATUTORY, AS TO THE INFORMATION IN THIS DOCUMENT.

Complying with all applicable copyright laws is the responsibility of the user. Without limiting the rights under copyright, no part of this document may be reproduced, stored in, or introduced into a retrieval system, or transmitted in any form or by any means (electronic, mechanical, photocopying, recording, or otherwise), or for any purpose, without the express written permission of Microsoft Corporation.

Microsoft may have patents, patent applications, trademarks, copyrights, or other intellectual property rights covering subject matter in this document. Except as expressly provided in any written license agreement from Microsoft, the furnishing of this document does not give you any license to these patents, trademarks, copyrights, or other intellectual property.

Unless otherwise noted, the example companies, organizations, products, domain names, e-mail addresses, logos, people, places, and events depicted herein are fictitious, and no association with any real company, organization, product, domain name, e-mail address, logo, person, place, or event is intended or should be inferred.

© 2010 Microsoft Corporation. All rights reserved.

Contents

[SQL Server High Availability and Disaster Recovery for SAP: A Technical Case Study 1](#_Toc271205804)

[Introduction 5](#_Toc271205805)

[Audience 5](#_Toc271205806)

[Solution Requirements 5](#_Toc271205807)

[Goals for the Solution 6](#_Toc271205808)

[Recovery Point Objective 6](#_Toc271205809)

[Recovery Time Objective 6](#_Toc271205810)

[Site Disaster Recovery 6](#_Toc271205811)

[Reduction in Mainframe Support Costs 7](#_Toc271205812)

[Performance 7](#_Toc271205813)

[High-Level Results Post Migration 7](#_Toc271205814)

[Performance Measured Post-Migration 7](#_Toc271205815)

[Supportability and Tuning 8](#_Toc271205816)

[Flexibility 8](#_Toc271205817)

[Database Size 8](#_Toc271205818)

[Dealing with a Site Failover 8](#_Toc271205819)

[Running the Solution on One Node 9](#_Toc271205820)

[SAP Landscape 9](#_Toc271205821)

[Premigration Legacy Implementation 10](#_Toc271205822)

[New Design – Windows Server Geo-Clustering and SQL Server 10](#_Toc271205823)

[Multiple Data Centers 11](#_Toc271205824)

[Logical Cluster Design Layout 12](#_Toc271205825)

[SAN Data Replication 14](#_Toc271205826)

[Geo-Cluster Instance Failover Time 16](#_Toc271205827)

[Solution Failure Modes Sustainable 16](#_Toc271205828)

[Loss of Network Links 17](#_Toc271205829)

[Loss of a Site or Database Cluster Node 18](#_Toc271205830)

[Logical Design of SAP 4.6C on SQL Server 2005 18](#_Toc271205831)

[Logical Design of SAP ERP6 on SQL Server 2005 19](#_Toc271205832)

[Design Elements 20](#_Toc271205833)

[Clustering Specifics 20](#_Toc271205834)

[Database Server Configuration 22](#_Toc271205835)

[SAN Storage 22](#_Toc271205836)

[Post Migration Go-Live Practices 23](#_Toc271205837)

[Monitoring SQL Server and SAP Performance 23](#_Toc271205838)

[SQL Server Monitoring via System Monitor 23](#_Toc271205839)

[Monitoring Using SAP Transactions 25](#_Toc271205840)

[ST03 25](#_Toc271205841)

[ST04 or DBA Cockpit 26](#_Toc271205842)

[DBA Cockpit 26](#_Toc271205843)

[Security and Hardening 26](#_Toc271205844)

[Applying Updates to Windows Server and SQL Server 27](#_Toc271205845)

[Conclusion 28](#_Toc271205846)

[Appendix: Hardware and Software High Level Bill Of Materials 28](#_Toc271205847)

# Introduction

Queensland Rail (QR) has been running trains in the Australian state of Queensland since the 1860s. Queensland Rail undertook a migration from a z/OS mainframe running DB2 to host SAP. Early versions of SAP for example R/2 only ran on mainframes; however, newer versions of SAP R/3 run on a number of different hardware and software platforms, providing more options for SAP customers. QR engaged Microsoft Services to assist in migrating the database platform underpinning SAP to Microsoft® SQL Server® 2005 by June 2007. This timeframe would enable QR to upgrade SAP to ERP6 running on SQL Server 2005 running in a highly available configuration by 2008. The change of platform to the Windows Server® and SQL Server platforms would provide a much more cost-effective total cost of ownership, as well as the high availability between data centers that was a requirement for QR. Improved performance headroom was also critical, because the load on the system was expected to grow.

QR stressed that achieving maximum availability and near-immediate recovery from a disaster was of highest importance for QR’s operations. QR’s design goals were to eliminate all single points of failure in the architecture, and now, with well thought-out procedures and practices, QR has 99.99x percent uptime, including both planned and unplanned downtime.

This paper provides insight into the details of the highly available architecture Microsoft designed to host a mission-critical SAP solution.

# Audience

This paper is intended for architects and solution integration specialists who want to learn more about SQL Server high availability design patterns that customers use in production to host mission-critical application environments. The focus of this paper is to provide insight into high availability for a SAP implementation project; however, the best practices presented here can be reused to satisfy other database high availability project requirements, not just those that involve SAP.

# Solution Requirements

SAP is a mission-critical application within QR, second in criticality only to the systems that control signaling and junction boxes for the rail track in the field. In the premigration configuration, SAP application servers running version 4.6C were connected to DB2 v7.1 for z/OS ver. 1.4 on the mainframe. The mainframe running the database and supporting the SAP system was active in only one data center. The disaster recovery (DR) site, located several kilometers away, contained only SAP servers and a DR mainframe in a cold state. If the main data center containing the database were to fail, the entire SAP solution would be offline until the cold mainframe could be activated (started) and appropriate network changes made in order to simulate an environment in which the DR mainframe looked like the main data center mainframe. This startup procedure was labor-intensive, and it took approximately four hours when the appropriate staff members were available onsite. The QR team needed to minimize any unscheduled downtime, as well as improve performance significantly in order to cope with the increased database demands imposed by the upcoming upgrade to SAP ERP 6. The mainframe had many scheduled outages (on Sunday nights) that also caused SAP outages.

SAP also specified that support for SAP ABAP and Java work processes would end with SAP NetWeaver 2004s on the z/OS platform. For more information about SAP support for operating system platforms, see the SAP product availability matrix, which is available from SAP (<http://service.sap.com/pam>). Please note that this site requires credentials that can only be supplied by SAP.

Finally, the pricing arrangement for support for the legacy mainframe platform was very expensive for QR. A move to the Windows Server/SQL Server platform would enable the entire SAP solution to be supported on commodity hardware at a much better price point. Over time, QR had moved nearly all of the other applications off the mainframe; the SAP supporting database was one of the last applications to be migrated off so that the mainframe could be decommissioned.

# Goals for the Solution

The solution Microsoft provided addressed several key goals defined by QR for the project.

## Recovery Point Objective

The Recovery Point Objective (RPO) value essentially stipulates the amount of committed data that can be lost due to a disaster or failure scenario. In this project the RPO value was deemed to be zero, because the mission-critical SAP deployment at QR couldn’t afford to lose any data in the events of:

Loss of a database server.

Loss of storage in a data center.

Loss of an entire data center.

Downtime, planned or unplanned.

## Recovery Time Objective

The Recovery Time Objective (RTO) value stipulates the amount of time that the solution can be unavailable to users due to a disaster or other unplanned failure scenario. The RTO value defines how much time the system and operators have in order to recover from a failure situation and resume normal operation. In this project there are multiple failure modes, because the solution was also meant to provide high availability even if one of the data centers is offline. Because this was a mission-critical application for QR, the database downtime was deemed to be a total of five minutes per year for unplanned outages. It should be noted that since the database migration to SQL Server in June 2007, no unplanned outages have been caused by SQL Server database issues.

## Site Disaster Recovery

The premigration solution used a mainframe for cold disaster recovery. If the SAP servers on the primary site failed, the application could be restarted on the SAP servers in the DR environment. However, if the entire data center went offline, the SAP system would remain unavailable until the application servers and SAP database servers were brought up manually on the DR site mainframe, which, as mentioned earlier, took around four hours. As part of this migration project QR wanted to ensure that the solution was impervious to site failures.

## Reduction in Mainframe Support Costs

The maintenance costs for supporting SAP amounted to hundreds of thousands of dollars per month. This was also expected to grow significantly due to the large number of upcoming projects. The new platform needed to provide a configuration for running SAP at reduced costs, because the mainframe would be decommissioned. As a separate business exercise, the cost of the replatforming project was weighed against the running costs for the existing platform, and calculations revealed that the entire SAP migration project would cost less than six months of the support costs of the mainframe environment. The return on investment (ROI) of this database migration project was less than six months, counting both internal QR costs and external hardware and consulting costs.

## Performance

QR stipulated that performance baseline for the new solution should be the same or better. It is important to note however that the unwritten expectation was that the new platform should perform better in most areas so that QR could upgrade from SAP 4.6C to ERP6 (which imposes more load on the database and application infrastructure) without further hardware upgrades. Current-generation server hardware leverages the benefits of commodity prices as well as increased technology improvements over the years. Because the mainframe I/O path technology is an older implementation, its performance was not comparable to current-generation hardware, and it did not offer the same range of performance options. The MIPS rating of the mainframe was 1700.

The following table lists the success measures QR identified for this project.

|  |  |
| --- | --- |
| Success measure | Criteria |
| RPO | 0 |
| RTO | 5 minutes total per year for unplanned downtime |
| Availability scope | Site disaster |
| Performance | Same or better |
| Flexibility | Ability to run entire solution on one node |
| Planned downtime | Planned as needed for SAP upgrades |

**Table 1:** Summary of goals and success criteria

# High-Level Results Post Migration

After the migration project was completed, the QR team reviewed the project requirements to validate that those requirements were met.

## Performance Measured Post-Migration

QR’s management service-level agreement (SLA) reporting processes track the performance of the SAP solution via early watch reports. These reports have indicated that the dialog response times for users (at peak dialog steps per hour periods) improved from initial ranges of 0.4 to 0.55 seconds per dialog step with DB2 to a range of 0.25 to 0.35 seconds per dialog step with SQL Server. After the DB2 to SQL Server migration, response time performance improved more than 60 percent during the peak load period.

CPU utilization is on track. As of this writing, the same hardware is still in use. (The migration occurred in June 2007.) One cluster node supports mainly the SAP ERP6 load while the other cluster node supports mainly the SAP BW load. The hardware configuration is covered in more detail later in this paper. There have been a number of occasions where QR has operated all the SAP databases on the one node with more than adequate performance. Normal database load while the solution is running is between 20 and 30 percent CPU utilization on both nodes.

## Supportability and Tuning

QR expended a lot of time and effort on tuning the DB2 system and running the mainframe infrastructure. This was drastically reduced due to the introduction of SQL Server 2005, which is designed to require less hands-on tuning and support. QR estimates that this has saved almost a full-time equivalent (FTE) resource in database tuning activities alone.

## Flexibility

The SQL Server solution has greatly improved flexibility. As demonstrated in a large number of projects that have occurred since the conversion, QR can far more easily duplicate systems from a base system (either development or production) to assist in project execution or prototyping.

## Database Size

Initial estimates provided by other vendors suggested that database size would increase significantly after a move from DB2 to SQL Server. In fact, the reverse was true in QR’s case: The database shrunk considerably—from 630 GB to 490 GB (providing a 20 percent gain) at conversion time for QR.

The SAP database size in 2009 was approximately 1.1 terabytes, and there has been no discernable performance degradation since migration to SQL Server in mid-2007. Since then, QR has upgraded to SAP ERP6 and added multiple new functional rollouts, all running on the original hardware.

SAP Solution Manager and a production SAP Mobile Infrastructure system now also use the same SQL Server database node.

## Dealing with a Site Failover

The SQL Server solution handles node failure in the same way as a site failover. In a site or node failure situation, the SQL Server instance and the databases that the affected instance hosts are automatically moved over to the other site. This also means that since there are two HDS SANs replicating writes synchronously, the LUNs hosted by the failed SAN are remounted on the surviving SAN by the cluster configuration. This LUN remounting feature works automatically and was tested numerous times before the production rollout. The automatic remounting of the LUNs can be either handled by the HDS TrueCopy software or by other third-party solutions like Symantec storage foundation software.

In all cases the failover time was well within the SLA conditions specified by the business.

The synchronous data replication between the two SANs over the fiber network ensured that there was no data loss.

## Running the Solution on One Node

The solution needed to have the ability to host the entire production system on one node that was capable of running SQL Server. The servers used for the solution were sized to be capable of hosting all of instances of SQL Server while one of the two nodes was unavailable (due to planned or unplanned downtime). This was demonstrated numerous times during rolling upgrades that were used for security fixes and patches for SQL Server and Windows Server. The team also routinely performed disaster recovery testing during maintenance windows every six months to ensure that everything was running correctly. The solution used the **max\_server\_memory** and **min\_server\_memory** settings to avoid memory conflicts and memory oversubscription issues when all the instances were running on one node. The following rules were applied for the **min\_server\_memory** and **max\_server\_memory** configuration settings of the SQL Server instances:

* If the SQL Server instance hosts large SAP databases such as SAP ERP and SAP BW, accept the default value for **max\_server\_memory**, and adjust the **min\_server\_memory** value. Max\_server\_memory remains at the default maximum.
* If the SQL Server instance hosts smaller SAP databases that should have priority upon startup, adjust both the **min\_server\_memory** and **max\_server\_memory** values.
* To provide adequate memory for the Windows Server operating system, the total amount of memory set for **min\_server\_memory** for all SQL Server instances should be less than the amount of memory on the physical host minus at least 4 GB to allow for the Windows Server operating system.
* CPU affinity and I/O affinity settings where not be altered from the default values.

In situations where an instance hosting a large database moves between hosts, a memory rebalancing period occurs when the instance starts and the data cache of the new instance starts being populated. The memory rebalancing happens as existing instances running on that node relinquish memory by trimming their data caches and the new instance allocates the memory, thereby increasing its data cache. Under normal load situations, this rebalancing can take several minutes to be achieved.

# SAP Landscape

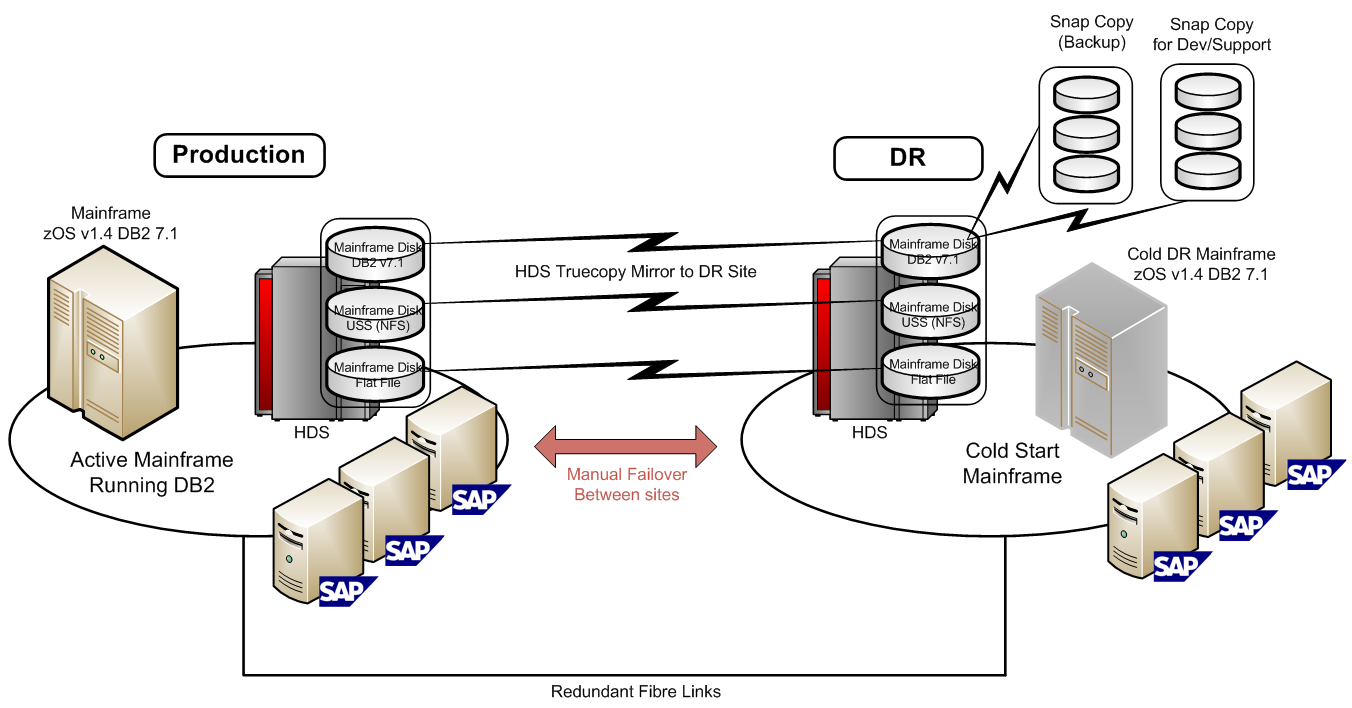
This describes the high level design of the pre and post environments supporting the SAP application

## Premigration Legacy Implementation

SQL Server replaced a legacy architecture in which the DB2 database ran on z/OS. This architecture hosted a number of SAP databases that supported application servers running on Windows Server 2003. QR installed numerous SAP R/3 Modules (HR/PY, MM, PM, PS, SD, FI, AR, AP, GL, CO, PP, LO, AA) to provide business functionality. To protect against disk subsystem failure, the data volumes were replicated to the DR SAN via HDS TrueCopy. The fiber links provided low latency communications to the DR SAN.

Before the migration project started, the SAP Central Instance application servers had already been moved off the z/OS mainframe due to performance problems. They were running on Windows Server 2003, Enterprise x64 Edition Intel based machines. Only the DB2 databases for SAP remained on the mainframe.

Moving forward, QR wanted to implement a modern data center architecture in which the IT infrastructure could support the business from either data center with minimal disruption. In the existing legacy environment, a failure of the main data center would also make the mainframe unavailable, thereby causing extended downtime while the cold DR mainframe was brought online.



**Figure 1:** Legacy Mainframe Environment

## New Design – Windows Server Geo-Clustering and SQL Server

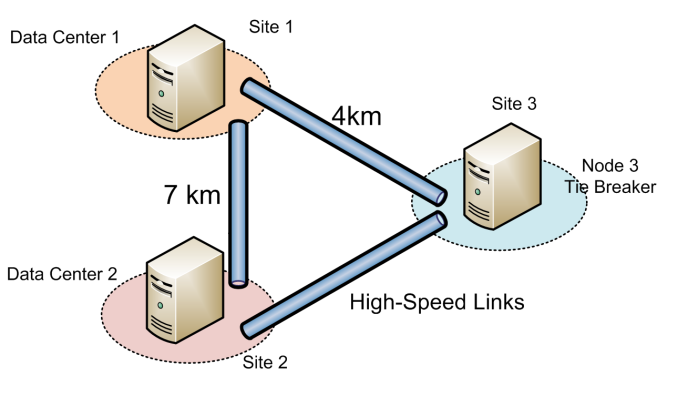
In order to meet the business requirements of both high availability and performance at a reasonable total cost of ownership, a stretch Windows Server cluster with SQL Server 2005 was chosen. Stretch clusters are also known as a geo-clusters (short for geographically dispersed clusters) because the cluster nodes are located at two data centers located typically several kilometers apart.

This design provides failover capability for SQL Server between the two data centers, which means that the system can remain online even during a data center–wide disruption.

The x64-based database servers have sufficient CPU power and performance headroom to be able to support the future needs of the QR SAP solution. After the database platform was migrated, the SAP version was also upgraded to SAP ERP 6 without requiring any further hardware investment. From a performance perspective, current generation x64 hardware coupled with 64-bit editions of Windows Server and SQL Server removed previous limitations around memory and performance. The x64 platform expanded the former 4 GB memory limit and allowed direct memory access to all the physical memory on the server. This enabled great performance improvements for both the SAP application servers and the SQL Server RDBMS.

## Multiple Data Centers

The QR IT infrastructure leveraged two data centers and the fiber links between them. QR owned the physical fiber link that was laid next to the rail tracks. The two data centers were situated in close proximity to the rail line, but they were sufficiently far apart to provide good redundancy. They were also close enough to minimize significant delays in servicing components. The data centers were powered by two separate power substations. Owning and controlling the fiber links provided some special benefits to QR, because the link is dedicated to the traffic placed upon it by the company, unlike most leased lines that are multiplexed with other unknown loads and managed by service providers.



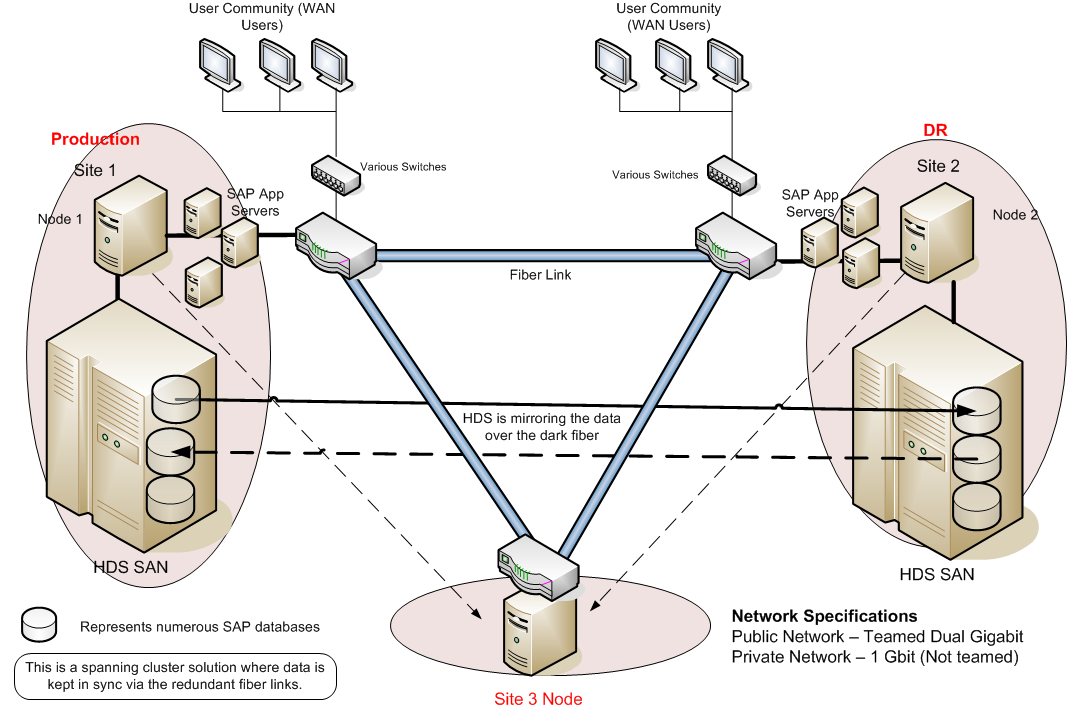
**Figure 2:** Data center locations and links

The direct benefit of the dedicated fiber translated to good performance of the link as well as the speed to implement changes without the necessary contractual negotiations and added costs.

The solution added a third site, which was a small computer room located approximately halfway between the two main data centers. This site is used for remote data storage and accessible independently by each of the other two data centers. The actual server implementation on the third site uses a virtual server with minimal configuration and load.

## Logical Cluster Design Layout

This solution used a geo-cluster. A geo-cluster is significantly different from a normal cluster configuration; in this configuration two SANs mirror the data synchronously between them. If the site fails, the instance of SQL Server as well as the mounted active disks are moved and then remounted on the remaining SAN. In a single-site cluster configuration, the SAN is shared between the server nodes. Because there is only one shared disk subsystem, it is a single point of failure.



**Figure 3:** Geo-cluster design for high availability

In a geo-cluster, special SAN-specific drivers and software are needed in order to ensure that the LUNs are mapped across to the other SAN just before the instance of SQL Server and any file shares fail over to, and then start on, the remaining node. Also, in a single-site Windows Server 2003 failover cluster, the shared disk becomes a voting member in the arbitration process to determine which node owns the disks, if the nodes can no longer communicate with each other. In a geo-cluster, this is not possible because there are two SANs and the disks must be synchronized and then mounted on the destination node before the SQL Server RDBMS is started up. In this case a witness or third node can provide this arbitration capability. The third node only exists to provide the arbitration vote and does not have to be a powerful server, because it does not need to actually host an instance of SQL Server. As an alternative to a third node, a remote file share can be used for this purpose. This enhancement option was made available in Service Pack 1 of Windows Server 2003.

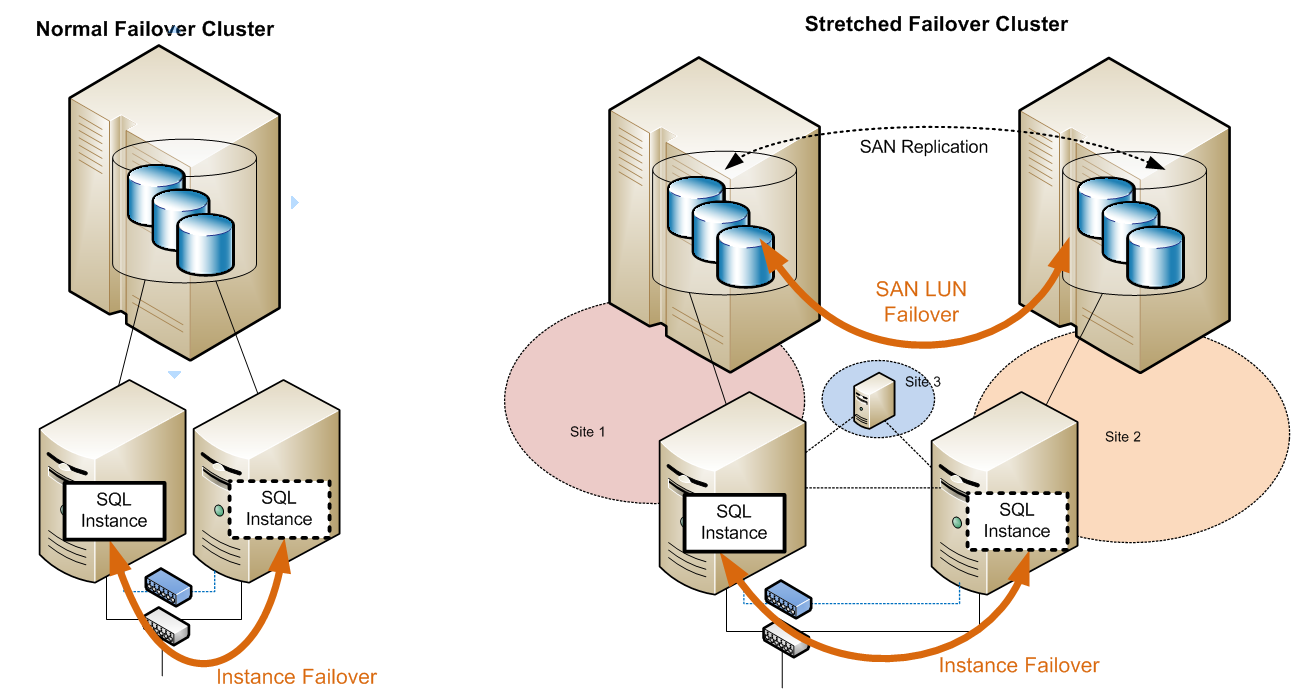
#### Related Article

Article ID: 921181 An update is available that adds a file share witness feature and a configurable cluster heartbeats feature to Windows Server 2003 Service Pack 1-based server clusters.

<http://support.microsoft.com/kb/921181>

On the three-node cluster, two nodes are designated as preferred owners and have identical memory and CPU configuration. The third node, used primarily as a remote host for the quorum, has less memory.

Failover between cluster nodes is automatic, but the cluster is set to not automatically fail back when the preferred node is available. The optional failback to the preferred node can then be performed manually when convenient, enabling administrators to perform the operation at an off-peak time to avoid further disruption during normal operations.



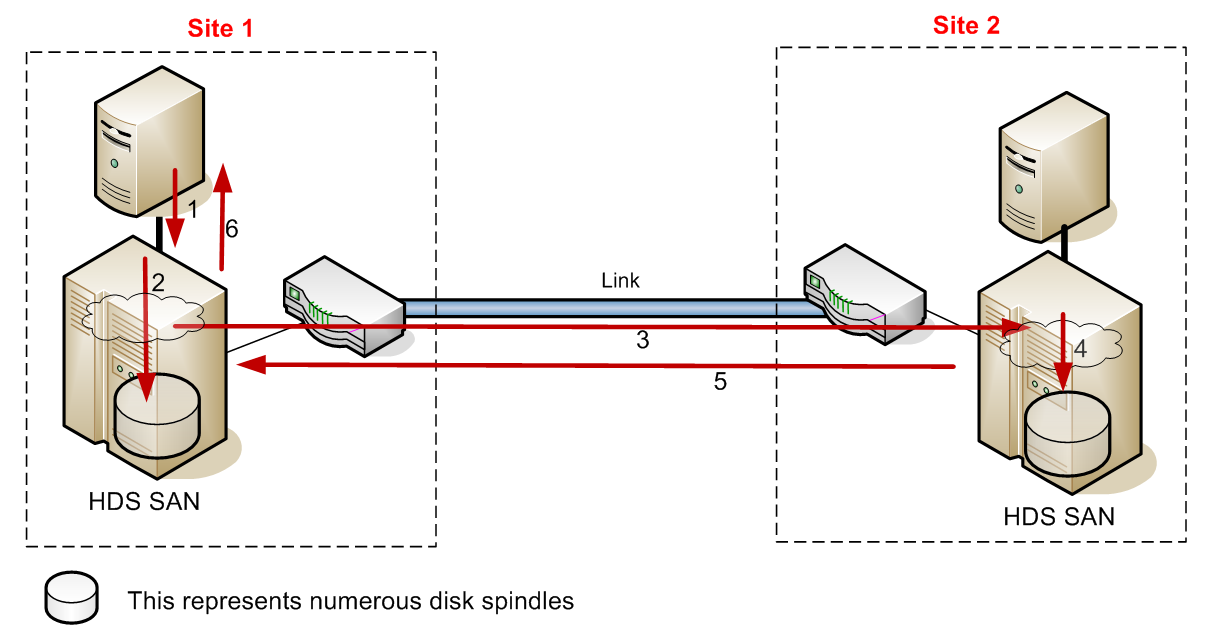
**Figure 4:** Normal Windows Server failover cluster vs. stretched failover cluster design

The failure of either any two network links or a single server failure can be mitigated in this design as the arbitration between the server nodes is maintained. Network links to user and application connections must also be maintained, however this is beyond the cluster and not covered in this document. The network connectivity between all of the sites was 3 by 14 gigabits because the company already owned the sufficient fiber strands. It should be noted however that gigabit speed for the connectivity to the third site is not required. However, the network latency must be less than 500 ms. (500 ms is an upper limit, and it is specific to Windows Server 2003 failover cluster network I/O, not to the SAN site-to-site fiber synchronous latency that is covered later in this paper.) We recommend that network I/O latency under normal situations be at least an order of magnitude better than the 500-ms limit.

## SAN Data Replication

SAN data replication between sites is a critical part of this solution. SAN vendors typically support two replication models - asynchronous and synchronous data replication. A geo-cluster implementation requires synchronous data replication to ensure no data loss. Asynchronous data replication cannot be used if the expected RPO value is zero, because if any data has not yet been sent to the remote site at the time of failure, it will be lost.

The main difference between the two SAN replication models is in the latency of the data that is persisted on the remote site. Asynchronous replication does not wait for the data to be persisted on the remote SAN, and the data latency depends on the link speed. That means that uncommitted I/Os between the SANs are allowed, and the server does not have to pause I/O throughput. On the other hand, synchronous data replication must ensure that every I/O is committed on the remote site before the whole I/O operation is deemed successful. The implication here is that there is no data loss during a site failure and that the I/O response times may suffer if the site to site bandwidth is not sufficient. This can be seen in the following figure, which shows the I/O path diagram.



**Figure 5**: Synchronous I/O replication between SANs

1 –The I/O write request is passed to the local SAN.

2, 3 – These two steps happen together. I/O is persisted to the local SAN cache or disk and also sent to the remote SAN.

4 – I/O is persisted on the remote SAN cache or disk.

5 – Acknowledgement that the I/O was persisted successfully is sent to the originating SAN.

6 – The local SAN acknowledges that the I/O has been persisted on the SAN fabric.

The project team at QR tested the link between sites to ensure that the speed and bandwidth was sufficient for even the most intensive tasks. The team used tools like SQLIO and SQLIOSim, which can simulate random and sequential I/O, with similar I/O sizes to what SQL Server can generate to test the I/O throughput of the solution. These test programs are available from Microsoft to stress-test disk I/O subsystems.

Here are some high level data points from I/O performance validation testing using SQLIO that QR conducted before they commissioned the solution.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Log File I/O Testing | | |  |  |  |  |  |  |  |  | |  |  |
|  | Type | 32KB sequential I/O Write | | |  |  |  |  |  |  | |  |  |
|  | No of Files | 1 |  |  |  |  |  |  |  |  | |  |  |
|  | Threads | 1 |  |  |  |  |  |  |  |  | |  |  |
|  | IOs/sec | 4428.9 |  |  |  |  |  |  |  |  | |  |  |
|  | MBs/sec | 138.4 |  |  |  |  |  |  |  |  | |  |  |
|  | Avg. Latency(ms) | 1 |  |  |  |  |  |  |  |  | |  |  |
|  | IO Histogram: |  |  |  |  |  |  |  |  |  | |  |  |
|  | Latency (ms) | 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24+ | | | | | | | | |
|  | % of IO requests | 34 36 21 6 2 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 | | | | | | | | |
|  |  |  |  |  |  |  |  |  |  |  | |  |  |
| Data File I/O Testing | |  |  |  |  |  |  |  |  |  | |  |  |
|  | Type | 8KB random I/O Write | | |  |  |  |  |  |  | |  |  |
|  | No of Files | 8 |  |  |  |  |  |  |  |  | |  |  |
|  | Threads | 8 |  |  |  |  |  |  |  |  | |  |  |
|  | IOs/sec | 4138.9 |  |  |  |  |  |  |  |  | |  |  |
|  | MBs/sec | 32.33 |  |  |  |  |  |  |  |  | |  |  |
|  | Avg. Latency(ms) | 14 |  |  |  |  |  |  |  |  | |  |  |
|  | IO Histogram: |  |  |  |  |  |  |  |  |  | |  |  |
|  | Latency (ms) | 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24+ | | | | | | | | |
|  | % of IO requests | 51 23 2 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 22 | | | | | | | | |
|  |  |  |  |  |  |  |  |  |  |  | |  |  |
|  |  |  |  |  |  |  |  |  |  |  | |  |  |
|  | Type | 8KB random I/O Read | | |  |  |  |  |  |  | |  |  |
|  | No of Files | 8 |  |  |  |  |  |  |  |  | |  |  |
|  | Threads | 8 |  |  |  |  |  |  |  |  | |  |  |
|  | IOs/sec | 4930.6 |  |  |  |  |  |  |  |  | |  |  |
|  | MBs/sec | 38.52 |  |  |  |  |  |  |  |  | |  |  |
|  | Avg. Latency(ms) | 12 |  |  |  |  |  |  |  |  | |  |  |
|  | IO Histogram: |  |  |  |  |  |  |  |  |  | |  |  |
|  | Latency (ms) | 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24+ | | | | | | | | |
|  | % of IO requests | 0 0 0 0 1 3 5 8 10 10 9 9 8 7 6 5 4 3 3 2 2 1 1 1 5 | | | | | | | | |

**Related Articles**

Article ID: 231619: How to use the SQLIOSim utility to simulate SQL Server activity on a disk subsystem

<http://support.microsoft.com/kb/231619>

SQLIO Disk Subsystem Benchmark Tool

<http://www.microsoft.com/downloads/details.aspx?familyid=9a8b005b-84e4-4f24-8d65-cb53442d9e19&displaylang=en>

## Geo-Cluster Instance Failover Time

The instance failover time in this case is defined as the time it takes for an instance of SQL Server to failover between cluster nodes. The overall failover time depends on a number of factors:

* Machine and SAN hardware vendor specifics, firmware, BIOS, and so forth
* Hardware failure detection timeouts
* HBA failure detection timeouts
* Heartbeat network failure detection timeouts
* SQL Server database recovery time
* The number of LUNs connected to that instance

The solution developed by the QR team employed a number of instances, databases, and clusters. The hardware in all cases was the same, and the main factor that influenced the failover time was the number of LUNs that the database was spread across. When the number of LUNs was less than two, the failover time was less than one minute. As the number of LUNs increased, the failover time took longer. The SAP CRM and Solution Manager systems failed over in much less than one minute. SAP R/3 and BW systems that employed many LUNs failed over in about two and a half minutes. This increasing failover time makes sense, because each LUN must be unmounted and then remounted on the remote site after the relevant SAN cache is cached. This must happen before the SQL Server service can restart.

The SQL Server database recovery time did not seem to play a large role in the overall failover time.

|  |  |
| --- | --- |
| **LUNs on HDS SAN** | **Failover time** |
| 1 to 4 | <1 minute |
| >8 | approx. 2.5 minutes |

**Table 2:** Failover Times

It should be noted that the results listed here are specific to this solution hardware and configuration. Also, failover comparison times are not applicable between normal failover cluster and geo-cluster configurations because the geo-cluster setup is substantially more complex.

## Solution Failure Modes Sustainable

The network and cluster nodes make up the solution. Each node communicates with the others over the network links to ensure that in the case of any single failure the solution can still maintain up-time and availability. In order for the cluster to be available, quorum must be maintained, which means that any two out of the three nodes must always be online and able to communicate with the other node or nodes.



**Figure 6:** Network links and sites

### Loss of Network Links

In this configuration, the solution can remain available even if any two network links are offline. If one network link is removed, as long as that traffic can be passed between sites, the solution can maintain high availability (abbreviated to HA in the following table). If two separate intra-site network failures occur, the solution will still be available but not able to sustain high availability if a further failure occurs. The following table illustrates how different types of failures are handled.

|  |  |  |
| --- | --- | --- |
| Failure (of) | Quorum | Result |
| Network A | Maintained between nodes 1, 2, and 3 | Online - HA maintained |
| Network B | Maintained between nodes 1,2, and 3 | Online - HA maintained |
| Network C | Maintained between node 1, 2, and 3 | Online - HA maintained |
| Networks A and B | Maintained between nodes 1 and 2 | Online - No further failure allowed |
| Networks A and C | Maintained between nodes 2 and 3 | Online - No further failure allowed |
| Networks B and C | Maintained between nodes 1 and 3 | Online - No further failure allowed |
| Networks A, B, and C | No quorum | Offline - No communication |

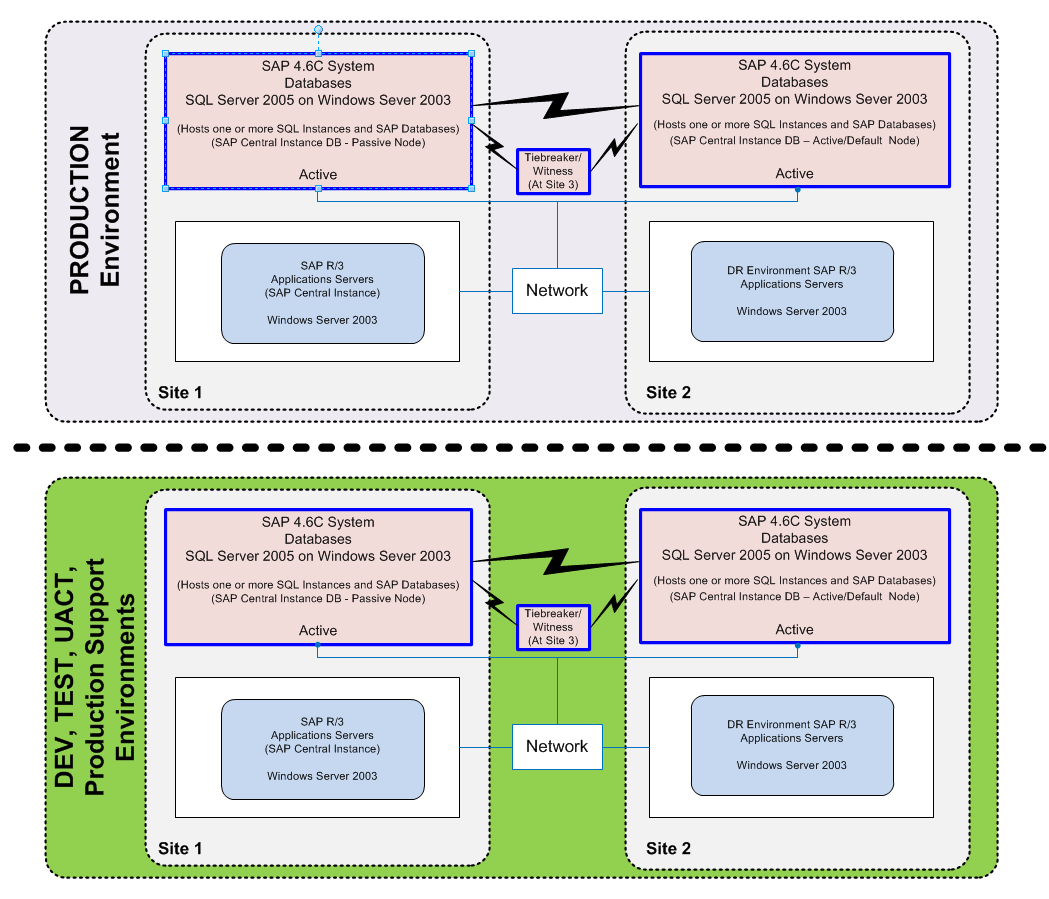
**Table 3:** Network failure modes

### Loss of a Site or Database Cluster Node

Similarly the cluster can sustain a loss of one server node and still remain active. This is because the cluster is built upon the principle of a shared-nothing cluster where a majority of the nodes must be available and able to communicate in order to form quorum. In this case, if any node fails the database is still available. However, if more than one node fails, the entire cluster will go offline, because the majority of nodes are now offline.

The failover cluster availability is governed by the principles of node majority. That means that the cluster can tolerate the failure of half the number of nodes minus one.

## Logical Design of SAP 4.6C on SQL Server 2005



**Figure 7:** Logical design

The diagram shows the logical design of the SAP production and nonproduction environments. Following best practices, the Development, Test, and UACT environments are also based on a geo-cluster configuration so that code changes, SAP transports, and upgrades are promoted logically into the production environment and change control is maintained. This also enables production support staff to test all proposed production changes in an environment that exactly mirrors production and also gain intimate familiarity with Windows Server and SQL Server features that the production environment uses.

The solution uses a number of instances of SQL Server that can be running on either of the two cluster nodes hosting multiple SAP databases. The SAP central instance provides critical SAP Message and Enqueue Server functionality for the other SAP application servers. The SAP central instance process is a single point of failure because it is not a clustered resource. This situation has been improved with the upgrade to SAP ERP6 from SAP 4.6C and using ASCS that provides SAP high availability at a SAP application level. This configuration is further discussed later in this paper.

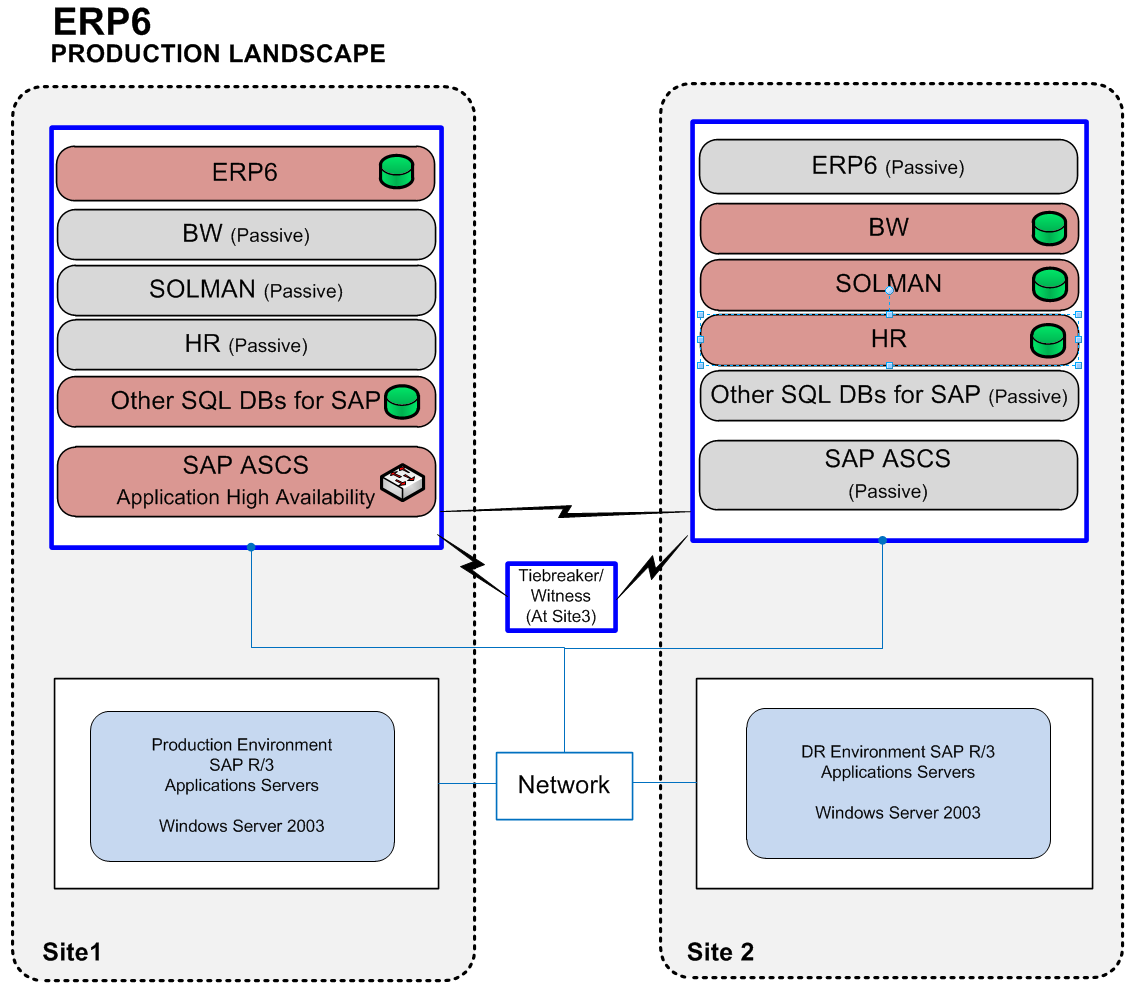
The database nodes can host the application load from all of the SAP application servers. The tiebreaker server at Site 3 is only used to maintain quorum and cast a vote to determine which node will be used to create quorum in the event of a system failure of one of the nodes. The node at Site 3 is a smaller server and not configured to host SQL Server.

In this configuration, the cluster nodes have the following software installed:

* Windows Server 2003, Enterprise x64 Edition
* SQL Server 2005 (64-bit) Enterprise Edition

## Logical Design of SAP ERP6 on SQL Server 2005

After upgrading the database infrastructure to SQL Server 2005, the QR team upgraded SAP from 4.6C to ERP6. In this configuration a number of instances of SQL Server host a number of databases. The ERP6 database is hosted on a specific instance of SQL Server. The SAP Business Warehouse (BW) database is hosted by a different instance of SQL Server, and the ERP6 and BW instances are configured to use different preferred cluster nodes so that the load is distributed among different cluster servers. There are a number of other related systems in that make up the total solution. Each of these systems has a database that is kept highly available by the geo-cluster configuration. The SAP Enqueue Server is made highly available via the installation of SAP ASCS on the cluster. This enables the SAP Enqueue Server service to start and run on either of the two clustered nodes. This solution provides a highly available SQL Server database environment and also hosts critical SAP services to provide SAP application high availability.



**Figure 8:** ERP 6 production cluster environment

# Design Elements

The team that designed and implemented this geo-cluster followed a number of Microsoft best practices. This section lists the critical areas that the team considered.

## Clustering Specifics

* The cluster operating system version used was Windows Server 2003 Enterprise Edition. As the company needs grow, the Windows Server Enterprise Edition cluster can be configured to host additional server nodes to provide extra functionality for SAP.
* Network testing showed that the fiber network latency was less than 1 ms even under load. Correct failover cluster setup requires that the network latency between nodes be less than 500 ms in order to avoid unnecessary failovers. The team specifically load tested both the fiber links and the heartbeat network infrastructure to validate expected results.
* Redundancy was considered. Each cluster node hosts two network cards for redundancy. The heartbeat network can also use the public network as a backup.
* Teaming is supported only on the public network interfaces. It is not supported for heartbeat network communications.
* Hardware compatibility was considered. Computer hardware that is used to assemble the geo-cluster is on the spanning cluster hardware compatibility list if Windows Server 2003 is used. This includes the SAN as well as the HBA and machine firmware. It is critical that the firmware, BIOS, and versions of components making up the cluster align with the hardware manufacturer’s recommendations. Windows Server 2008 clusters introduce a new concept called *validation*. The validation testing process built into Windows Server performs various tests on the underlying hardware subsystems to ensure that they meet the minimum hardware specifications and requirements to support a reliable cluster solution.
* Numerous hardware failover validation tests were done to ensure that the solution failed over according to the vendor expectations during different load scenarios.
* SQL Server and LUN dependency checks were carried out to ensure that the SQL Server service in the clustering setup was dependent on the disk and IP address resources being available first. If drive letters are used to assign to disk volumes, the SQL Server cluster setup program assigns the dependencies automatically. However, if mount points are used to define the disk volumes, or if drive letters are changed after setup, the mount point dependencies must be set up manually as a post-installation step.
* SAN vendors were involved in the project. The team found it very beneficial to involve the hardware partners during the planning and building phase of the project in order to ensure that the configuration was supported.
* SAN I/O and system performance testing was conducted to validate that the drivers, SAN multi-path, and host bus adapter configurations, as well as SAN settings, provided optimal performance.
* Validation of new drivers and hardware was considered. Drivers and hardware used in the solution must be validated against the Windows logo hardware compatibility testing results site. This validation ensures that the solution as a whole has been stress-tested by the hardware vendors and Microsoft.
* A stretched VLAN was used to accommodate the cluster nodes. SQL Server 2005 requires that the various nodes of the cluster be on the same subnet. A stretched VLAN was set up between the affected sites.
* As mentioned earlier, the function of Site 3 was arbitration, so it only needed a host and a quorum file share. Only nodes in sites 1 and 2 were configured to host SQL Server.

## Database Server Configuration

Several best practice configuration items apply to SQL Server in an SAP environment. Microsoft has published specific guidance in this area. Documentation is available on the Microsoft/SAP Information Center website (<http://www.microsoft.com/isv/sap/>) or on <http://www.sdn.sap.com/irj/sdn/sqlserver>. Configuration best practices are listed here:

* SQL Server should be configured to use a number of equal-sized data files for the SAP database. The number of files typically corresponds to the number of CPU cores in the database server. This applies to both the SAP database and **tempdb**. When the database size is extended, the database files making up the database should each be increased by the same amount at the same time.
* The operating system and database software should be x64 or IA64 versions. Use of x86 versions of software in a production environment is not supported.
* MSDTC is not required for SAP. Therefore it should not be installed.
* The SQL Server management tools should ideally not be installed on the database cluster. The database failover cluster should be managed through remote desktop servers on the perimeter.
* The ERP SQL Server database instance should be configured with **max degree of parallelism** set to 1.
* The BW SQL Server database instance should be configured with **max degree of parallelism** set to 1.

## SAN Storage

* The data file system should be formatted with allocation units 64K in size.
* The LUNs should be partitioned with an offset based on a multiple of 64K to ensure proper disk alignment. It should be noted that this applies to systems that are based on Windows Server 2003. Disks configured with Windows Server 2008 automatically create partitions correctly aligned with a 1 MB offset.
* The SAN performance should be validated with the SQLIO tool using both random and sequential I/O patterns. Ideally LUNs that host database data files should have a latency of 10 ms or better. The LUNs that host database log files should have a latency of 5 ms or better.
* SAN and instance failover should be extensively tested during the project to validate the solution. For this test case, disasters were simulated by physically unplugging servers at random while they were running and by disconnecting fiber cables while the system was under a test load.
* SAN connections that use Multipath I/O (MPIO), which provides I/O path redundancy as well as path load balancing, should be tested to verify that the I/O paths work correctly and also provide the extra I/O bandwidth.
* SAP solutions should be scoped for disk I/O performance rather than disk space.
* HBA settings that relate to the number of outstanding I/Os, sometimes called *queue depth*, should be revised according to performance testing results to ensure that the HBA is performing as expected.

# Post Migration Go-Live Practices

## Monitoring SQL Server and SAP Performance

The QR IT team monitors the database and SAP environment using SAP and Microsoft tools as well as third-party tools like Spotlight from Quest Software. There is tight integration of system monitoring and performance data gathering between SAP and SQL Server. Many of the performance statistics that SQL Server captures are available and can be accessed via the SAP administration transactions.

### SQL Server Monitoring via System Monitor

Post go-live, QR gathered a number of metrics on a routine basis in order to be able to detect trends and perform troubleshooting analysis as required. Data captured at a frequency of 15 seconds has insignificant impact on the overall system performance. The following list is an extensive list of counters that was implemented went the project went live.

|  |  |  |
| --- | --- | --- |
| **Performance Object** | **Counter** | **Instance** |
|  |  |  |
| Processor | %Processor Time | Total |
|  |  | All Processors |
| %Privileged Time | \_Total |
| %User time | \_Total |
|  |  |  |
| Memory | Available Kbytes |  |
|  | Cache Bytes |  |
| Pages Per Sec |  |
| Pool Nonpaged Bytes |  |
| Pool paged Bytes |  |
| Page faults/sec |  |
|  |  |  |
| Paging File | % Usage | Total |
|  | % Usage Peak | Total |
|  |  |  |
| Network Interface | Bytes Total/sec | For all NICs |
|  | Output Queue Length | For all NICs |
| Packet Received Errors |  |
| Packet Outbound Errors |  |
|  |  |  |
| Physical Disk | Average Disk Queue Length | All Database Drives Selected |
|  | Disk Read Bytes/sec | All Database Drives Selected |
| Disk Write Bytes/sec | All Database Drives Selected |
| Disk Sec/Read | All Database Drives Selected |
| Disk Sec/Write | All Database Drives Selected |
| Disk Sec/Transfer | All Database Drives Selected |
|  |  |  |
| Process | %processor Time | SQLserver |
|  | % Privileged Time | SQLserver |
| %User time | SQLserver |
| Thread Count | SQLserver |
| Virtual Bytes | SQLserver |
| Virtual Bytes Peak | SQLserver |
| Working Set | SQLserver |
| Working Set Peak | SQLserver |
|  |  |  |
| Server | Bytes Total/sec |  |
|  |  |  |
| System | Context Switches/sec |  |
|  |  |  |
| SQLServer: Databases | Transactions /sec |  |
|  | Backup/Restore Throughput/sec |  |
|  |  |  |
| SQLServer: General Statistics | Logins/sec |  |
|  | Logouts/sec |  |
| User Connections |  |
|  |  |  |
| SQLServer: Latches | Latch Waits /sec | Total |
|  |  |  |
| SQLServer: Locks | Lock waits/sec |  |
|  | Number of Deadlocks /sec |  |
| Avg. Wait Time (ms) | Select object type |
|  |  |  |
| SQLServer: SQL Statistics | Batch Requests /sec |  |
|  | SQL Compilations/sec |  |
| SQL Re-Compilations/sec |  |
| Auto-Param Attempts/Sec |  |
|  |  |  |
| SQL Server: Access Methods | Full Scans/sec |  |
|  | Index Searches/sec |  |
|  |  |  |
| SQL Server: Buffer Manager | Cache Hit Ratio |  |
|  | Page Life Expectancy |  |
| Checkpoint Pages/sec |  |
| Page reads/sec |  |
| Page writes/sec |  |
|  |  |  |
| SQL Server: Plan Cache | Cache Hit Ratio |  |
|  | Cache Object Counts |  |
| Cache objects in use |  |
| Cache Pages |  |
|  |  |  |
| SQL Server: Wait Statistics | Lock waits |  |
|  | Log buffer waits |  |
| Log Write Waits |  |
| Memory grant queue waits |  |
| Network IO waits |  |
| Non-Page latch waits |  |
| Page IO latch waits |  |
| Wait for the worker |  |
|  |  |  |

**Table 4:** Performance Counters

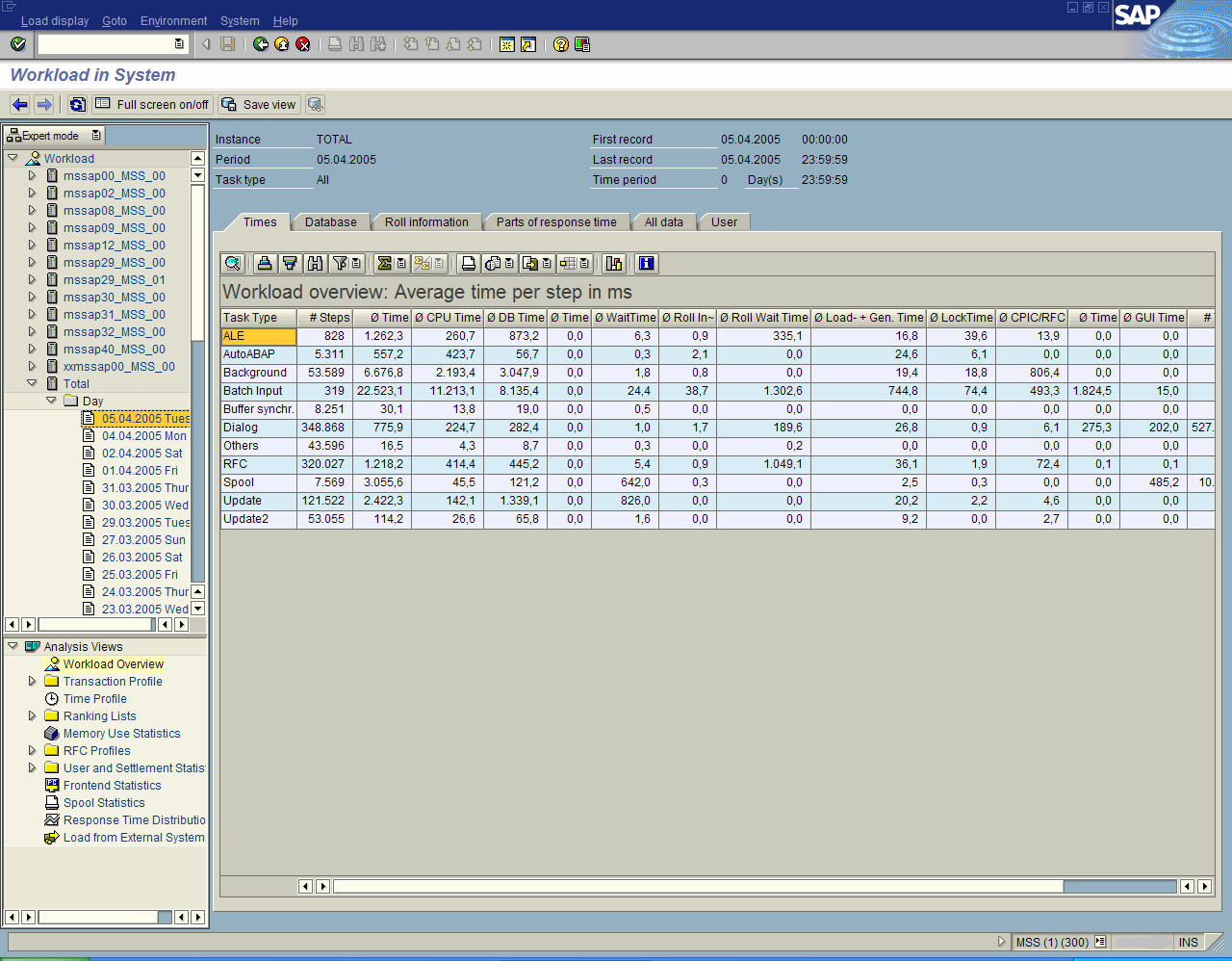
### Monitoring Using SAP Transactions

The QR SAP basis team also monitors the SQL Server and SAP Server from an SAP perspective.

The following SAP system transactions are typically used by the SAP team.

### ST03

This transaction will provide information about performance as the SAP application server experiences it. The listing shows the different workloads (dialog, batch, and so forth) and the average time spent in each step. The column DB Time is the average time a call spends processing in the database in milliseconds.

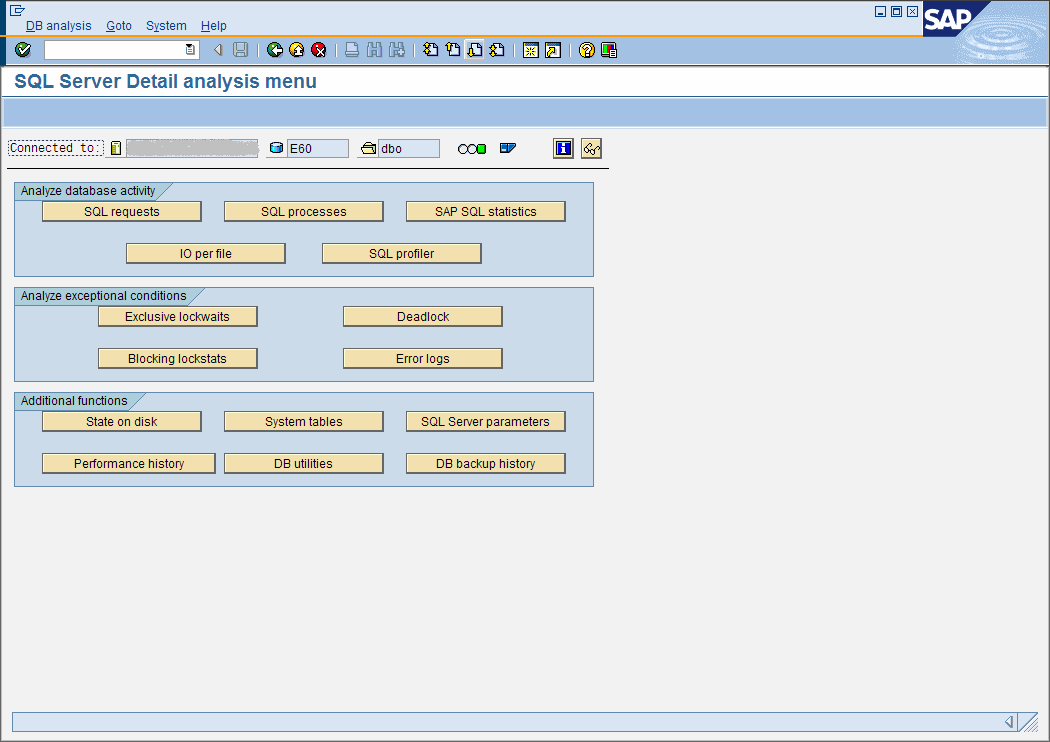


**Figure 9:** Example of as ST03 system transaction

Note: This screenshot is an example only and not a screenshot of the QR SAP system.

### ST04 or DBA Cockpit

This transaction provides both an overview of the system and the ability to drill into specific performance items. You can use this transaction to examine specific I/O performance over time, and you can also review other things like SQL Server error logs.



**Figure 10:** Example of an ST04 or DBA Cockpit system transaction

Note: This screenshot is an example only and not a screenshot of the QR SAP system.

### DBA Cockpit

Database Administration Cockpit (DBA Cockpit) is a new feature first provided in SAP support pack 12. This tool enables visibility and access to more SQL Server performance statistics and dynamic management views. This is an update to SAP transactions ST04 and DB02. For more information about DBA Cockpit, see SAP OSS note 1072066.

## Security and Hardening

There are a number of best practice guidelines around hardening SQL Server and Windows Server environments for SAP. The QR team is looking into this and planning on carrying out this system work in the future.

Some of the fundamental security and hardening tasks are:

* Moving the environment into a secure separate organizational unit (OU) in Active Directory® Domain Services (AD DS).
* Turning on and configuring the firewall features in Windows Server 2003.
* Stopping services that are not required for a SAP installation.
* Removing or disabling Internet Explorer® on the Windows Server environment.
* Configuring SQL Server and Windows Server appropriately.

System hardening guides and tools are available from Microsoft.

**Windows Server 2003 Security Baseline**

<http://technet.microsoft.com/en-us/library/cc163140.aspx>

**Windows Server 2003 Security Guide**

<http://www.microsoft.com/downloads/details.aspx?familyid=8A2643C1-0685-4D89-B655-521EA6C7B4DB&displaylang=en>

## Applying Updates to Windows Server and SQL Server

When updates needed to be applied to Windows Server and SQL Server, QR generally used a rolling upgrade method in order to minimize downtime. The new solution still calls for a maintenance window, but the work is done in less than an hour in most cases. The process involves failing over all the resources to another node and then running the update setup program on the current node. After the first node is upgraded, the resources are all moved back to the now upgraded server and the update setup is repeated on the remaining node. Then the third site is updated.

Service packs and updates are dealt with on a case-by-case basis; some updates support a rolling upgrade and some do not. For example, SQL Server 2005 Service Packs 2 and 3 do not support a rolling upgrade. In these cases a maintenance window is arranged for the upgrade after the system has been backed up.

This rolling upgrade process is described in the following public article.

Article: How Rolling Upgrades Work (Server Clusters: Rolling Upgrades. Upgrading to Windows Server 2003)

<http://technet.microsoft.com/en-us/library/cc779712(WS.10).aspx>

# Conclusion

QR has been able to build a highly available SQL Server database platform spanning multiple data centers and has upgraded to SAP ERP6 without needing any further hardware investment. SQL Server has been running with no unplanned downtime for numerous years and performance well exceeded the needs of the business. In some cases users had to run the reports twice and have the results validated by IT because they could not believe that the month-end reports could run that fast.

In order to enjoy a lower TCO, the QR IT team followed best practices and validated the architecture with Microsoft and SAP during the project. QR continues to enjoy reliable database infrastructure and has since built several more spanning cluster solutions in order to host other critical applications like the train management system and corporate email.

# Appendix: Hardware and Software High Level Bill Of Materials

* Dell Power Edge rack mount server hardware
* Hitachi Data Systems TagmaStore Universal Storage Platform
* Hitachi Data Systems TrueCopy Synchronous Remote Replication Software
* Brocade SilkWorm FC Switches
* FC HBAs for Dell PowerEdge Servers
* Symantec Veritas Volume Manager
* Microsoft Windows Server 2003 Enterprise Edition x64
* Microsoft SQL Server 2005 Enterprise Edition x64

**For more information:**

<http://www.microsoft.com/sqlserver/>: SQL Server Web site

<http://technet.microsoft.com/en-us/sqlserver/>: SQL Server TechCenter

<http://msdn.microsoft.com/en-us/sqlserver/>: SQL Server DevCenter

Did this paper help you? Please give us your feedback. Tell us on a scale of 1 (poor) to 5 (excellent), how would you rate this paper and why have you given it this rating? For example:

* Are you rating it high due to having good examples, excellent screen shots, clear writing, or another reason?
* Are you rating it low due to poor examples, fuzzy screen shots, or unclear writing?

This feedback will help us improve the quality of white papers we release.

[Send feedback.](mailto:sqlfback@microsoft.com?subject=White%20Paper%20Feedback:%20[SQL%20Server%20High%20Availability%20Architecture%20for%20SAP])