SOLUTION
READINESS

Using VMware HA, DRS and vMotion with Exchange 2010 DAGs

VMware, Inc
3401 Hillview Ave
Palo Alto, CA 94304
www.vmware.com

Contents

1. Overview

Organizations looking to deploy or upgrade to Microsoft Exchange 2010 have many options for providing highly available service. With the latest release of Microsoft Exchange, providing high availability is easier than ever. When deploying an Exchange 2010 mailbox cluster (known as a *Database Availability Group* or DAG), almost no knowledge of the Microsoft Clustering Service is required because the installer installs and configures all required components. Shared storage is no longer required, which helps reduce the single points of failure, and databases can now fail-over individually instead of failing over the entire server (as in previous versions).

When organizations virtualize the Exchange environment on VMware vSphere™, additional options and features are available that can complement virtualized DAG nodes. As of Exchange 2010 SP1 Microsoft has added support for using hypervisor-based clustering solutions along with Exchange 2010 DAGs. Additionally, the use of live migration technology such as VMware vMotion™ is fully supported with Exchange 2010 DAGs. Microsoft support details for Exchange 2010 on hardware virtualization solutions can be found in Microsoft *Exchange 2010 System Requirements*. VMware has also published a KB article, *Microsoft Clustering on VMware vSphere: Guidelines for Supported Configurations*, to provide guidance for virtualized Microsoft clustering solutions.

Tests indicate that VMware vSphere features, including VMware HA, Distributed Resource Scheduling (DRS), and vMotion, can be used with virtualized Exchange 2010 DAG nodes. Test cases included automated DAG node recovery using VMware HA to reduce the amount of time needed to reestablish database protection, and vMotion of DAG nodes. Distributed Resource Scheduling relies on vMotion, so by testing vMotion we can confidently say that DRS can be used effectively with DAG nodes. By testing with loads similar to those found in production environments, guidance and best practices have been established for using these vSphere features in production Exchange 2010 environments.

1.1 Purpose

Tests were performed to:

- Validate the use case of combining VMware HA with Exchange DAGs to reduce the amount of time required to re-establish DAG availability.
- Validate the use of vMotion with Exchange DAGs to allow the use of DRS and proactive workload migration while maintaining database availability and integrity.
- Provide guidance and best practices for taking advantage of these vSphere features.

These tests are *not* meant to show performance characteristics of Exchange 2010 on vSphere. For performance results, see the following VMware VROOM! Blog articles:

- *Exchange 2010 Disk I/O on vSphere*
- *Scale-Out Performance of Exchange 2010 Mailbox Server VMs on vSphere 4*
- *Exchange 2010 Scale-Up Performance on vSphere*

1.2 Audience

The intended audience includes:

- VMware partners who are responsible for designing and implementing Exchange 2010 on vSphere.
- Customers who are considering using vSphere advanced features to increase the operational efficiency of their Exchange 2010 environment.

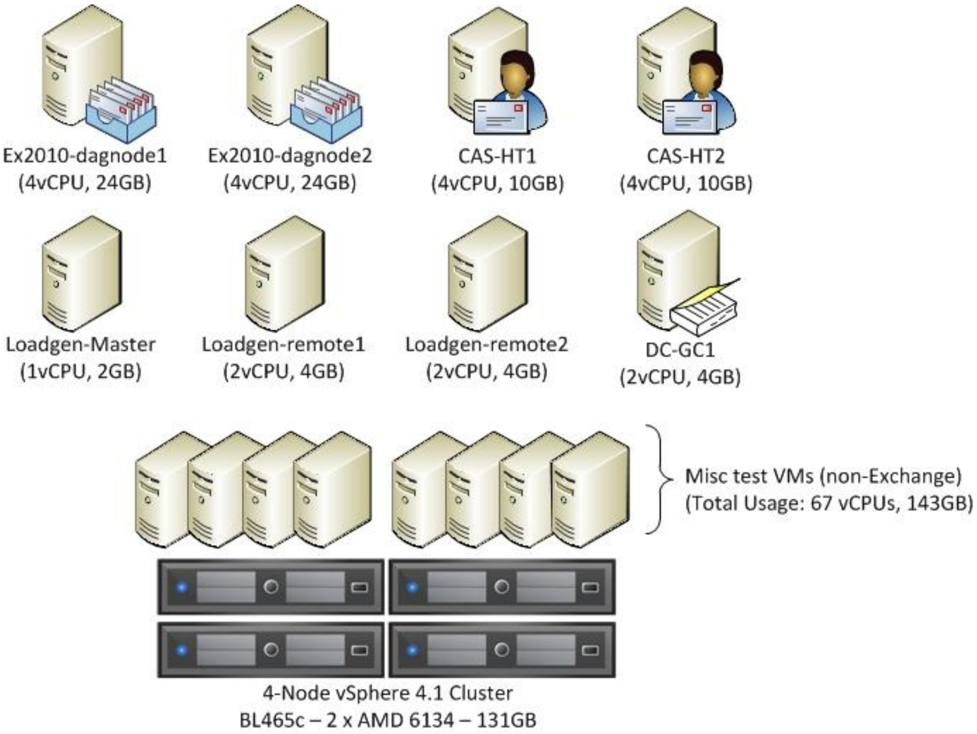
2. Testing Environment Overview

The vSphere environment used for testing was shared with multiple applications, similar to many production environments. We chose to not isolate the Exchange workload, and instead, ran these tests while other application tests were occurring to simulate more of a "real-world" environment. The following table and architecture diagram describes the testing environment.

Table 1. Hardware/Software Components

Component	Configuration
Server Model	HP BL465 G7
Platform	VMware ESXi 4.1 (260247)
CPU Type	AMD Opteron 6134
No. of CPUs per Server (physical/logical)	8/16
Memory per Server	128GB
Network	1Gbps
Shared Storage Technology	Fibre Channel
Multipathing Technology	VMware ESX™ Native Multipathing
Shared Interconnect	Intel NC364m, HP GbE2c
Guest OS Version	Windows 2008 R2 SP1
Exchange Software Version	Exchange 2010 SP1
Load Generation Software	Microsoft JetStress, LoadGen 2010
Storage	EMC NS960
Disks	(30) 2TB SATA II

Figure 1. Architecture Diagram



3. Testing Tools and Methodology

When testing against Microsoft Exchange it is best to use the tools Microsoft recommends for validating an environment before deploying to production. For these tests Microsoft Exchange Server Jetstress 2010 and Exchange Load Generator 2010 were used to validate the configurations and provide a simulated load.

3.1 Jetstress

Jetstress is used in pre-production environments to simulate the Exchange database and log file loads placed on the storage system. By using Jetstress, Performance Monitor, and esxtop we can validate that the underlying storage system allocated to the Exchange 2010 mailbox virtual machines has adequate bandwidth to support our use case. Jetstress should only be used in non-production environments. More information can be found in Microsoft's *Jetstress Field Guide*.

3.2 Load Generator 2010

LoadGen is a simulation tool used to validate the design and performance characteristics of servers within the Exchange environment. LoadGen measures the impact of various client types and helps determine if the servers can handle the load intended. LoadGen should only be used in an isolated environment. More information can be found at Microsoft *Tools for Performance and Scalability Evaluation* page.

3.3 Test Cases

Each test scenario was scoped to run with LoadGen simulating 3000 heavy users (100 messages sent/received per day). VMware vSphere performance monitoring tools and Windows Performance Monitor were used to measure the impact of the tests being performed. The client performance (LoadGen results), DAG stability, and Exchange mailbox database integrity are used to determine success or failure.

Table 2. Test and Description

vSphere Feature Tested	Task
n/a	Jetstress throughput test
n/a	LoadGen baseline test
vMotion	Test manual vMotion during LoadGen run
DRS	Test effect of DRS groups and rules
HA	Test DAG recovery time after ESX host failure

4. Testing and Validation Results

The following tests were performed:

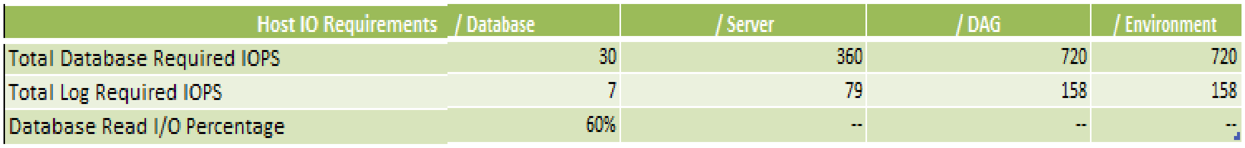
- Jetstress validation
- LoadGen validation
- vMotion
- DRS
- HA

Results are given for each test.

4.1 JetStress Validation and Results

The purpose of this test was to validate the storage design in use for the HA, DRS, and vMotion tests. To support 3000 medium-heavy users (100 messages sent/received per day) the Exchange 2010 Mailbox Storage Calculator predicted a requirement of 878 total IOPS. To achieve this IO requirement 30 2TB SATA II drives were configured as six 4+1 RAID 5 disk groups.

Figure 2. Mailbox Role Calculator IO Requirement

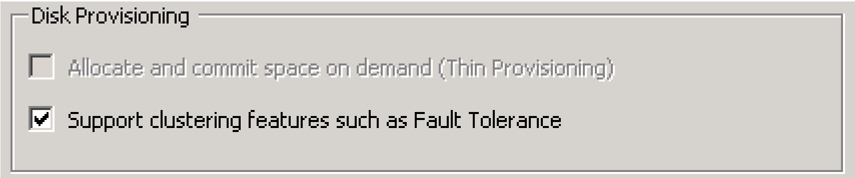


Host IO Requirements	/ Database	/ Server	/ DAG	/ Environment
Total Database Required IOPS	30	360	720	720
Total Log Required IOPS	7	79	158	158
Database Read I/O Percentage	60%	--	--	--

Each RAID 5 set held a 2TB LUN formatted as a VMFS3 datastore. Within each VMFS datastore four 270GB virtual disks (VMDK) were created using the option **Support clustering features such as Fault Tolerance**.

Note Selecting this feature is recommended when creating virtual disks for use with high IO workloads because it zeros out the entire virtual disk prior to allowing the virtual machine access. Another option is to create the virtual disk using the vSphere CLI command `vmkfstools` with the `eagerzeroedthick` option.

Figure 3. Creating Virtual Disk As eagerzeroedthick Using the vSphere Client



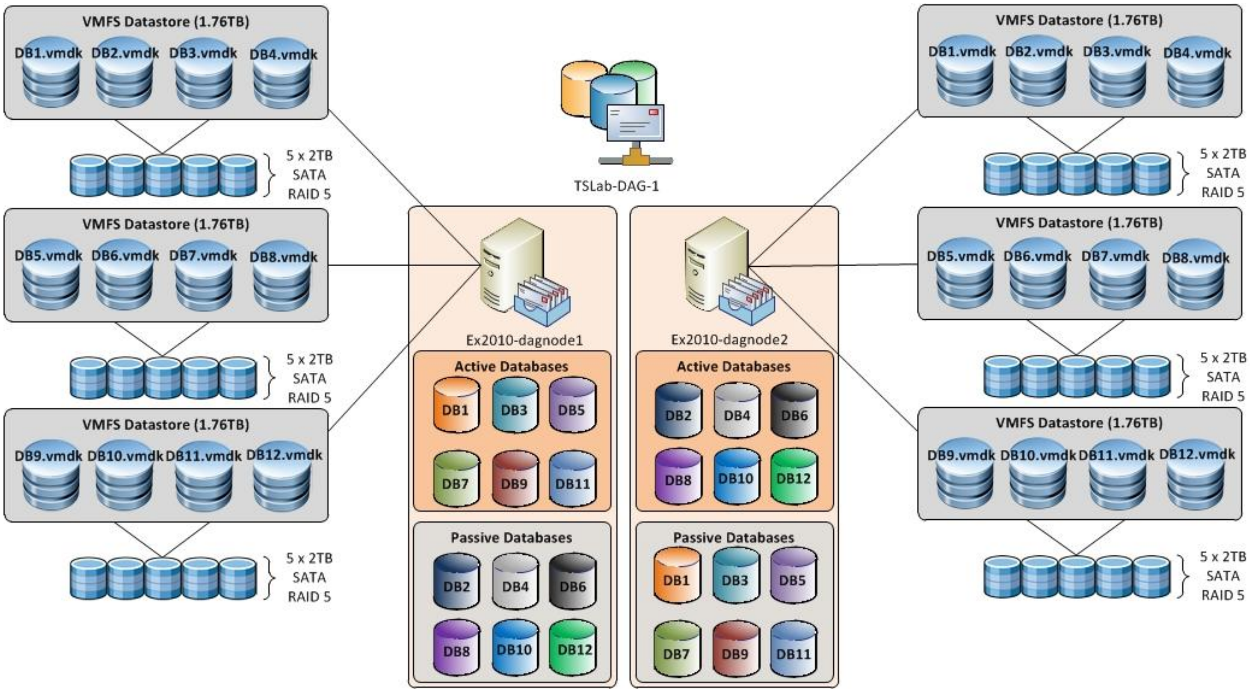
Disk Provisioning

□ Allocate and commit space on demand (Thin Provisioning)

☑ Support clustering features such as Fault Tolerance

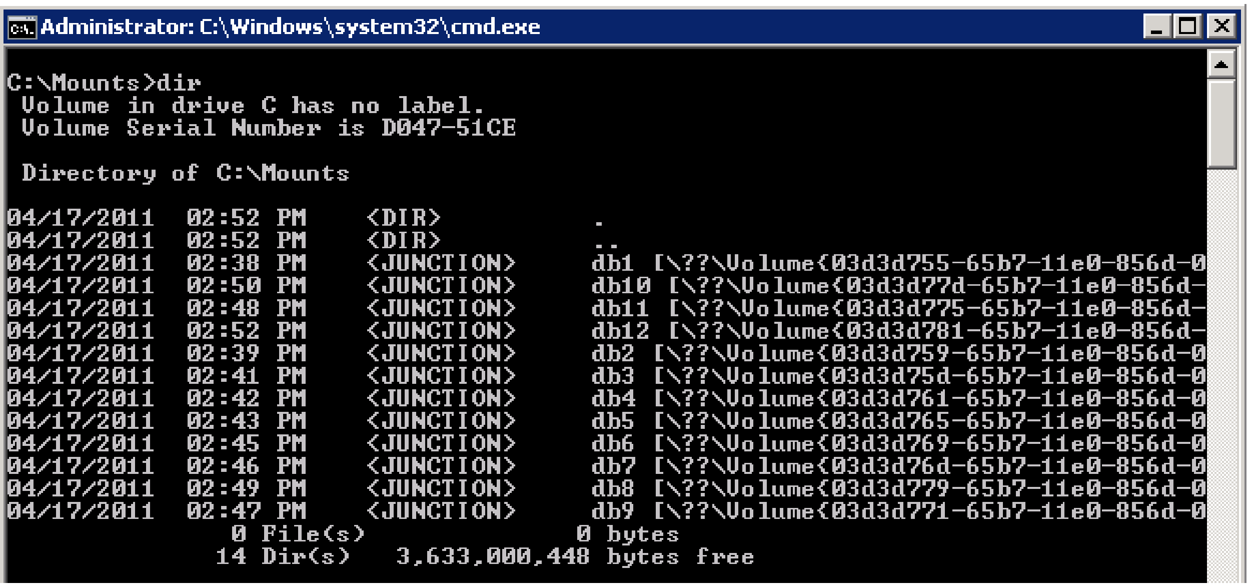
Of the four virtual disks created within the same raid set, two were used to house active mailbox databases and the other two were used to house passive mailbox databases. In all, 12 virtual disks, across three RAID sets were assigned to each DAG node. Each DAG node supports six active mailbox databases during normal operations.

Figure 4. DAG Node Database and Virtual Disk Layout



Within the virtual machine the 12 volumes were formatted as NTFS, using a 64KB file allocation unit size with the **quick format** option disabled. Each new volume was mounted as a mount point off of the directory `c:\mounts`. Because this was not a production environment it was decided to co-locate the database and log files as opposed to placing them on separate physical spindles.

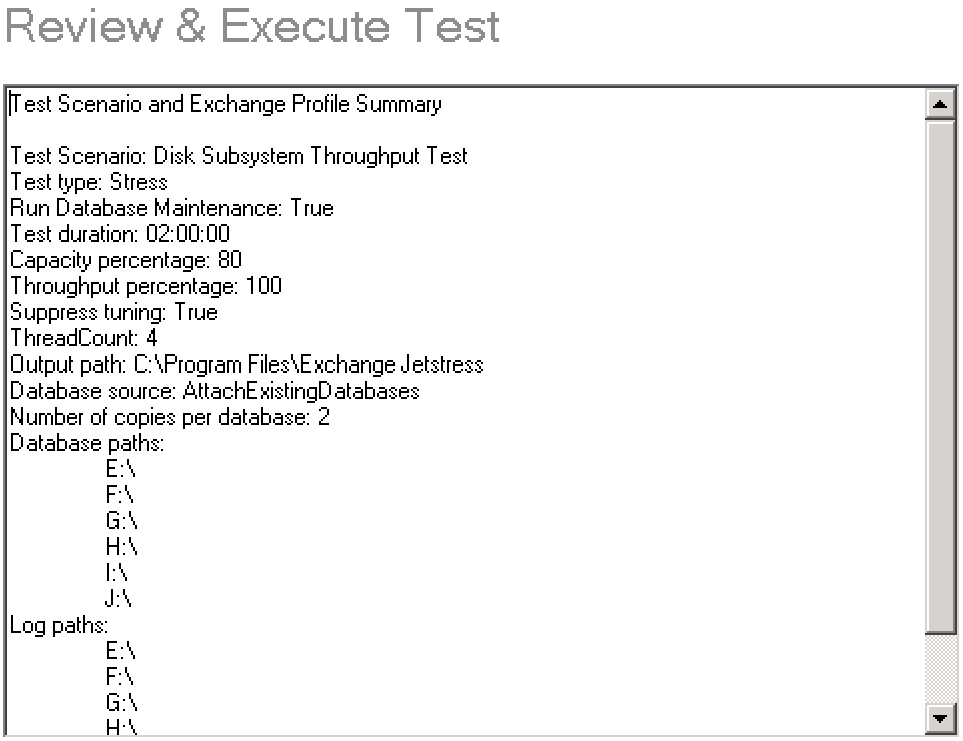
Figure 5. Database Mount Points



```
C:\Mounts>dir
 Volume in drive C has no label.
 Volume Serial Number is D047-51CE

 Directory of C:\Mounts

04/17/2011  02:52 PM    <DIR>          .
04/17/2011  02:52 PM    <DIR>          ..
04/17/2011  02:38 PM    <JUNCTION>     db1 [\??\Volume{03d3d755-65b7-11e0-856d-0
04/17/2011  02:50 PM    <JUNCTION>     db10 [\??\Volume{03d3d77d-65b7-11e0-856d-
04/17/2011  02:48 PM    <JUNCTION>     db11 [\??\Volume{03d3d775-65b7-11e0-856d-
04/17/2011  02:52 PM    <JUNCTION>     db12 [\??\Volume{03d3d781-65b7-11e0-856d-
04/17/2011  02:39 PM    <JUNCTION>     db2 [\??\Volume{03d3d759-65b7-11e0-856d-0
04/17/2011  02:41 PM    <JUNCTION>     db3 [\??\Volume{03d3d75d-65b7-11e0-856d-0
04/17/2011  02:42 PM    <JUNCTION>     db4 [\??\Volume{03d3d761-65b7-11e0-856d-0
04/17/2011  02:43 PM    <JUNCTION>     db5 [\??\Volume{03d3d765-65b7-11e0-856d-0
04/17/2011  02:45 PM    <JUNCTION>     db6 [\??\Volume{03d3d769-65b7-11e0-856d-0
04/17/2011  02:46 PM    <JUNCTION>     db7 [\??\Volume{03d3d76d-65b7-11e0-856d-0
04/17/2011  02:49 PM    <JUNCTION>     db8 [\??\Volume{03d3d779-65b7-11e0-856d-0
04/17/2011  02:47 PM    <JUNCTION>     db9 [\??\Volume{03d3d771-65b7-11e0-856d-0
               0 File(s)              0 bytes
              14 Dir(s)   3,633,000,448 bytes free
```

The Jetstress test was configured to run a two-hour disk subsystem throughput test, including database maintenance and one HA copy of the database. Four threads were used to achieve the required IOPS. The pre-test configuration is shown in the following figure.

Figure 6. Jetstress Test Configuration



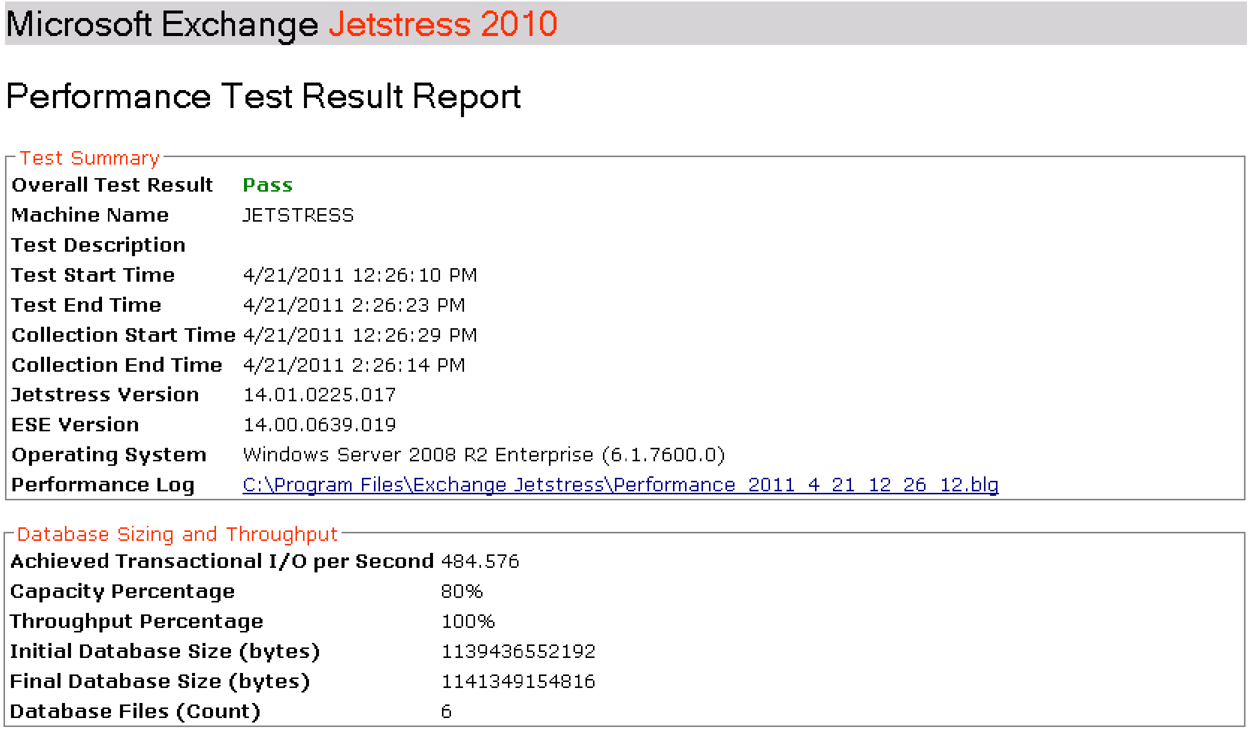
Review & Execute Test

```
Test Scenario and Exchange Profile Summary

Test Scenario: Disk Subsystem Throughput Test
Test type: Stress
Run Database Maintenance: True
Test duration: 02:00:00
Capacity percentage: 80
Throughput percentage: 100
Suppress tuning: True
ThreadCount: 4
Output path: C:\Program Files\Exchange Jetstress
Database source: AttachExistingDatabases
Number of copies per database: 2
Database paths:
        E:\
        F:\
        G:\
        H:\
        I:\
        J:\
Log paths:
        E:\
        F:\
        G:\
        H:\
```

4.1.1 Results

Based on the requirements predicted by the Exchange 2010 Storage Calculator, the storage provisioned for our 3000 test users was adequate. The following figure shows the output of the Jetstress 2010 report.

Figure 7. Jetstress Report



Microsoft Exchange Jetstress 2010

Performance Test Result Report

Test Summary

Overall Test Result	Pass
Machine Name	JETSTRESS
Test Description	
Test Start Time	4/21/2011 12:26:10 PM
Test End Time	4/21/2011 2:26:23 PM
Collection Start Time	4/21/2011 12:26:29 PM
Collection End Time	4/21/2011 2:26:14 PM
Jetstress Version	14.01.0225.017
ESE Version	14.00.0639.019
Operating System	Windows Server 2008 R2 Enterprise (6.1.7600.0)
Performance Log	C:\Program Files\Exchange Jetstress\Performance_2011_4_21_12_26_12.blg

Database Sizing and Throughput

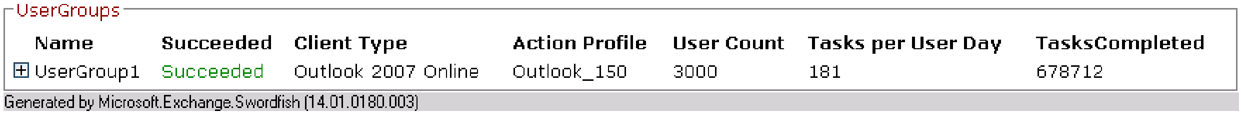
Achieved Transactional I/O per Second	484.576
Capacity Percentage	80%
Throughput Percentage	100%
Initial Database Size (bytes)	1139436552192
Final Database Size (bytes)	1141349154816
Database Files (Count)	6

4.2 LoadGen Validation and Results

The initial LoadGen run was required to validate the sizing that was recommended by the Exchange Storage Calculator and to obtain a baseline performance measurement. The original test plan called for using a medium-heavy user who would average about 100 messages sent/received per day. Initial testing showed that the virtual hardware and storage could easily handle the load placed by that profile. It was decided to use a heavier user profile to increase CPU utilization within the virtual machines and more closely replicate real world workloads. A user profile of 150 messages sent/received per day was used for all subsequent tests.

During the LoadGen tests a single user group was used for all 3000 users. The users were equally spread between the two DAG nodes with 250 users per database across six databases per DAG node. The databases were equally spread between the two CAS/Hub servers by setting the `RPCClientAccessServer` parameter for each database instead of using a hardware or software load-balancer. As shown in the following screenshot, the single 3000 user group successfully passed the initial LoadGen run using a more aggressive client type, Outlook 2007 Online, versus a cached mode client type.

Figure 8. Baseline LoadGen Results



UserGroups

Name	Succeeded	Client Type	Action Profile	User Count	Tasks per User Day	TasksCompleted
⊞ UserGroup1	Succeeded	Outlook 2007 Online	Outlook_150	3000	181	678712

Generated by Microsoft.Exchange.Swordfish (14.01.0180.003)

The following table outlines the performance counters used to validate responsiveness of Exchange from a client perspective during the baseline LoadGen run.

Table 3. Baseline Performance Results

Counter	DAG Node 1	DAG Node 2	Target
Processor\% Processor Time	26%	31%	< 80%
MSExchange Database ➔ Instances\I/O Database Reads Average Latency	17 ms	16 ms	< 20 ms
MSExchange Database ➔ Instances \I/O Database Writes Average Latency	1 ms	1 ms	< 20 ms
MSExchange Database ➔ Instances \I/O Log Reads Average Latency	1 ms	1 ms	< 20 ms
MSExchange Database ➔ Instances \I/O Log Writes Average Latency	1 ms	1 ms	< 20 ms
MSExchangeIS\RPC Requests	1	1	< 70
MSExchangeIS\RPC Average Latency	1 ms	1 ms	< 10 ms

These results show that client performance was well within the thresholds recommend by Microsoft to achieve good client responsiveness.

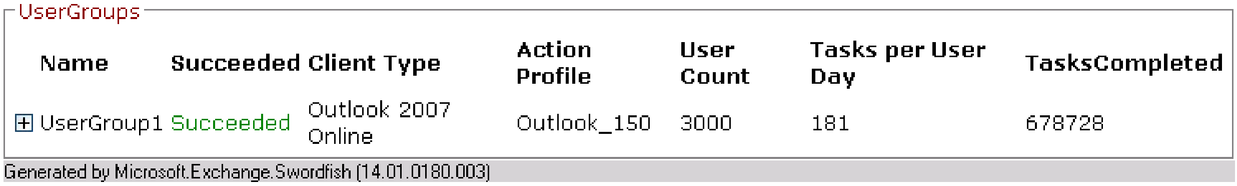
4.3 vMotion Results

Many customers rely on vMotion to quickly move workloads between vSphere hosts with no downtime. Historically, vMotion has not been supported by VMware for use with clustered virtual machines, and this continues to be the case for clustering solutions where shared storage is involved. However, for clustering solutions where there is no shared storage such as Exchange 2007 CCR and Exchange 2010 DAG, VMware does not restrict the use of vMotion. As of Exchange 2010 SP1 Microsoft has added support for using hypervisor-based clustering solutions along with Exchange 2010 SP1 DAGs. Additionally, the use of live migration technology such as VMware vMotion is fully supported with Exchange 2010 SP1 DAGs. Microsoft support details for Exchange 2010 SP1 on hardware virtualization solutions can be found in Microsoft *Exchange 2010 System Requirements*.

Note Microsoft support may differ from VMware support, so we urge customers to review Microsoft's support policies.

In this set of tests both nodes of the Database Availability Group were manually migrated using vMotion between three different vSphere hosts while running a 3000 heavy user LoadGen test. Each DAG node was migrated once per hour with database and cluster availability monitored throughout. ESXTop and Performance Monitor were used to determine the performance impact of the vMotion activity while LoadGen was used to monitor the overall client experience. At the conclusion of the LoadGen test all databases continued to be in a healthy and replicated state, and the LoadGen test was successful.

Figure 9. vMotion Testing LoadGen Results



UserGroups

Name	Succeeded	Client Type	Action Profile	User Count	Tasks per User Day	TasksCompleted
⊞ UserGroup1	Succeeded	Outlook 2007 Online	Outlook_150	3000	181	678728

Generated by Microsoft.Exchange.Swordfish (14.01.0180.003)

Table 4. vMotion Testing Performance Results

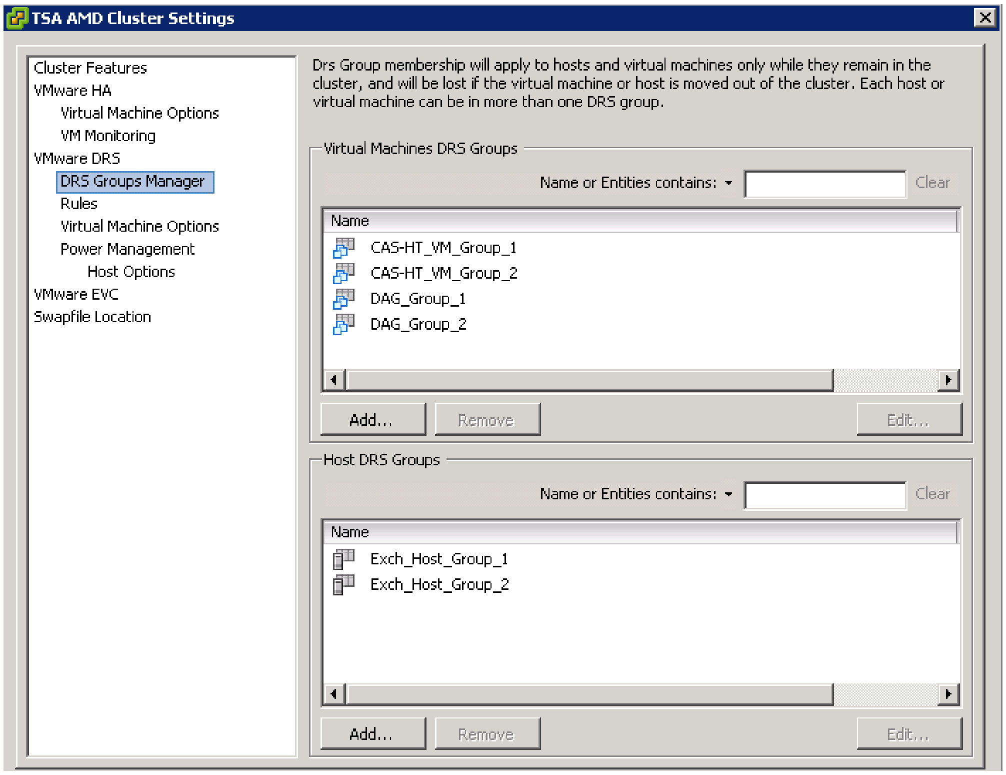
Counter	DAG Node 1	DAG Node 2	Target
Processor\% Processor Time	29%	42%	< 80%
MSExchange Database ➔ Instances\I/O Database Reads Average Latency	19 ms	17 ms	< 20 ms
MSExchange Database ➔ Instances \I/O Database Writes Average Latency	1 ms	1 ms	< 20 ms
MSExchange Database ➔ Instances \I/O Log Reads Average Latency	1 ms	1 ms	< 20 ms
MSExchange Database ➔ Instances \I/O Log Writes Average Latency	1 ms	1 ms	< 20 ms
MSExchangeIS\RPC Requests	1	1	< 70
MSExchangeIS\RPC Average Latency	1 ms	1 ms	< 10 ms

4.4 DRS Results

Using DRS for automated resource load-balancing enables administrators to offload the task of trying to move workloads as demand increases and decreases. Besides constantly monitoring the available compute resources and changing usage patterns of virtual machines, DRS allows administrators to proactively prepare the environment for maintenance tasks by evacuating virtual machines to other available hosts. Being able to use DRS for DAG nodes can eliminate the requirement for the Exchange administrator to manually initiate a failover and shutdown the virtual machine before the vSphere administrator places a vSphere host in maintenance mode.

Having tested vMotion, which DRS uses to migrate workloads, this set of tests focused on determining best practices for using DRS with Exchange 2010 virtual machines (all roles). DRS automatically places virtual machines on hosts that have adequate resources, but for clustered applications administrators might want additional control over where virtual machines are placed. In vSphere 4.1, DRS added support for creating groups for virtual machines and hosts, as well as rules for virtual machine placement and affinity. In our test cluster we created six groups as shown in the following figure.

Figure 10. DRS Groups



TSA AMD Cluster Settings

Drs Group membership will apply to hosts and virtual machines only while they remain in the cluster, and will be lost if the virtual machine or host is moved out of the cluster. Each host or virtual machine can be in more than one DRS group.

Virtual Machines DRS Groups

Name
CAS-HT_VM_Group_1
CAS-HT_VM_Group_2
DAG_Group_1
DAG_Group_2

Host DRS Groups

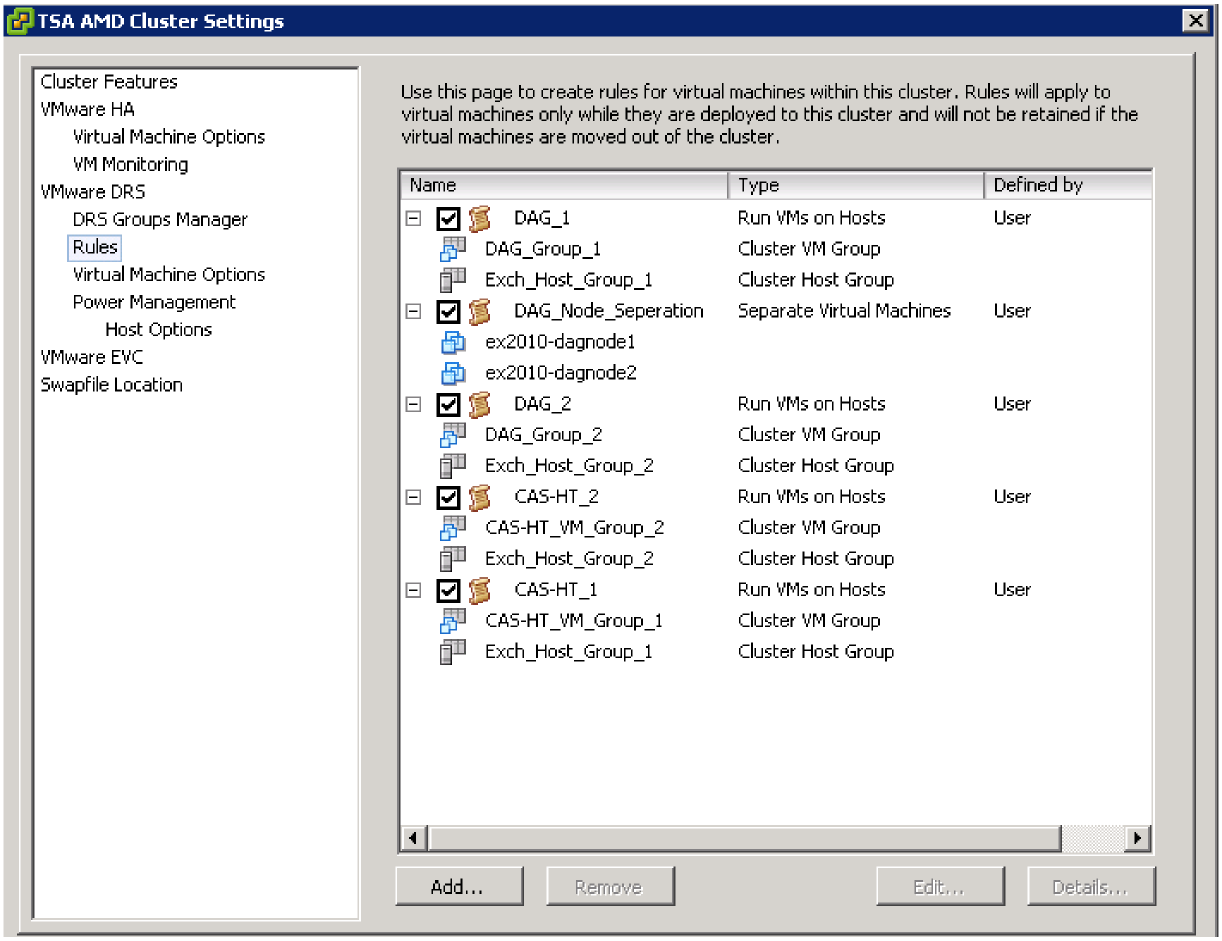
Name
Exch_Host_Group_1
Exch_Host_Group_2

Each virtual machine group contains half of the virtual machines that support that role (CAS, Hub, or DAG). Each host group contains half of the hosts in the cluster. By splitting the resources into groups we can create DRS rules to define where virtual machines are allowed to run. Table 5 shows the relationship of virtual machines and hosts to the groups.

Table 5. Virtual Machine and Host Groups

Group	Members
CAS-HT_VM_Group_1	Ex2010-casht1
CAS-HT_VM_Group_2	Ex2010-casht2
DAG_Group_1	Ex2010-dagnode1
DAG_Group_2	Ex2010-dagnode2
Exch_Host_Group_1	Tsa-bl465-1 Tsa-bl465-2
Exch_Host_Group_2	Tsa-bl465-3 Tsa-bl465-4

Figure 11. DRS Rules Assigned to Groups



Use this page to create rules for virtual machines within this cluster. Rules will apply to virtual machines only while they are deployed to this cluster and will not be retained if the virtual machines are moved out of the cluster.

Name	Type	Defined by
☑ DAG_1	Run VMs on Hosts	User
DAG_Group_1	Cluster VM Group	
Exch_Host_Group_1	Cluster Host Group	
☑ DAG_Node_Seperation	Separate Virtual Machines	User
ex2010-dagnode1		
ex2010-dagnode2		
☑ DAG_2	Run VMs on Hosts	User
DAG_Group_2	Cluster VM Group	
Exch_Host_Group_2	Cluster Host Group	
☑ CAS-HT_2	Run VMs on Hosts	User
CAS-HT_VM_Group_2	Cluster VM Group	
Exch_Host_Group_2	Cluster Host Group	
☑ CAS-HT_1	Run VMs on Hosts	User
CAS-HT_VM_Group_1	Cluster VM Group	
Exch_Host_Group_1	Cluster Host Group	

As shown in Table , five rules were created in the DRS cluster.

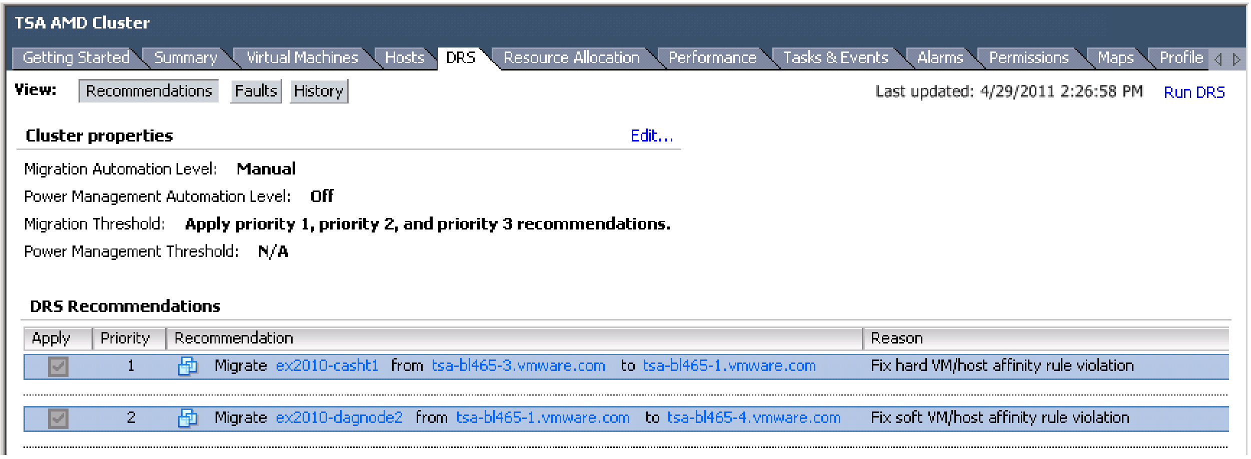
Table 6. DRS Rules

Rule Name	Type	Details
DAG_1	Run virtual machines on hosts	Virtual machines in DAG_Group_1 should run on hosts in Exch_Host_Group_1
DAG_2	Run virtual machines on hosts	Virtual machines in DAG_Group_2 should run on hosts in Exch_Host_Group_2
CAS-HT_1	Run virtual machines on hosts	Virtual machines in CAS-HT_VM_Group_1 must run on hosts in Exch_Host_Group_1
CAS-HT_2	Run virtual machines on hosts	Virtual machines in CAS-HT_VM_Group_2 must run on hosts in Exch_Host_Group_2
DAG_Node_Seperation	Separate virtual machines	Keeps virtual machines listed in the rule from running on the same host

The following tests were performed to validate the usage of DRS groups and rules:

1. Test DRS placement of virtual machines when rules are defined.
   a. Place virtual machines on hosts that violate DRS rules.
   b. Change the DRS automation level of virtual machines to **Fully Automated**.
   c. Validate placement of virtual machines.

Figure 12. DRS Recommendations



Last updated: 4/29/2011 2:26:58 PM

Cluster properties

Migration Automation Level: **Manual**
Power Management Automation Level: **Off**
Migration Threshold: **Apply priority 1, priority 2, and priority 3 recommendations.**
Power Management Threshold: **N/A**

DRS Recommendations

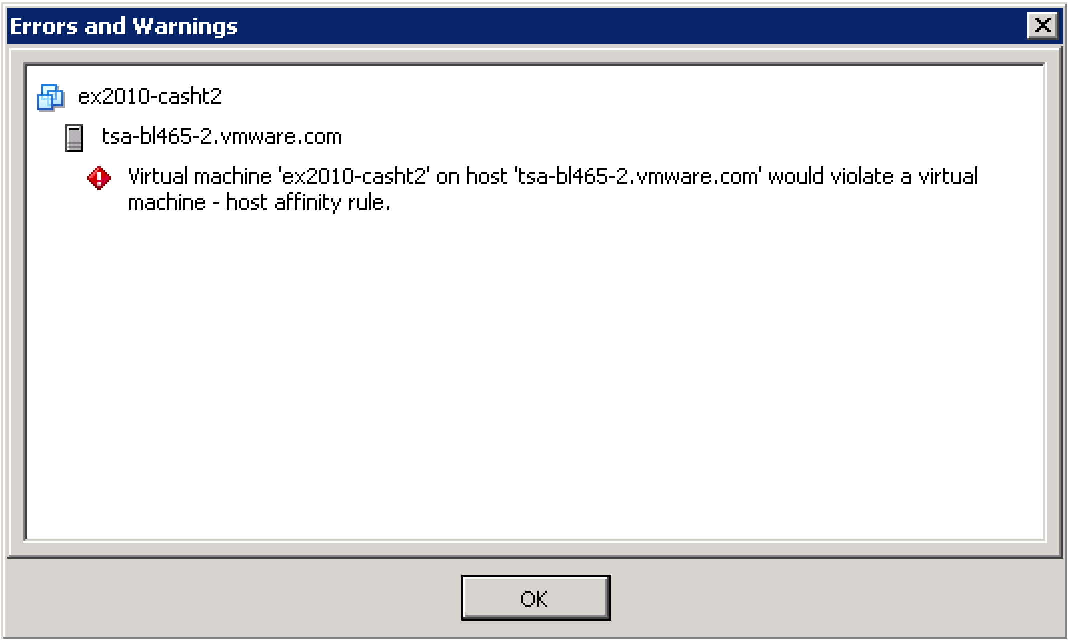
Apply	Priority	Recommendation	Reason
☑	1	Migrate ex2010-casht1 from tsa-bl465-3.vmware.com to tsa-bl465-1.vmware.com	Fix hard VM/host affinity rule violation
☑	2	Migrate ex2010-dagnode2 from tsa-bl465-1.vmware.com to tsa-bl465-4.vmware.com	Fix soft VM/host affinity rule violation

After manually placing the virtual machines on hosts that violated the DRS rules recommendations were made to fix the virtual machine/host affinity rule violations.

2. Attempt to manually violate DRS rules.
   a. Migrate a virtual machine that must run in a host group to another host group.

Note The use of a **Must run on hosts in group** DRS rule in this test was for demonstration purposes. In most cases the best practice recommendation would be to use a **Should run on hosts in group** DRS rule.

Figure 13. Manual vMotion Attempt



Errors and Warnings

ex2010-casht2

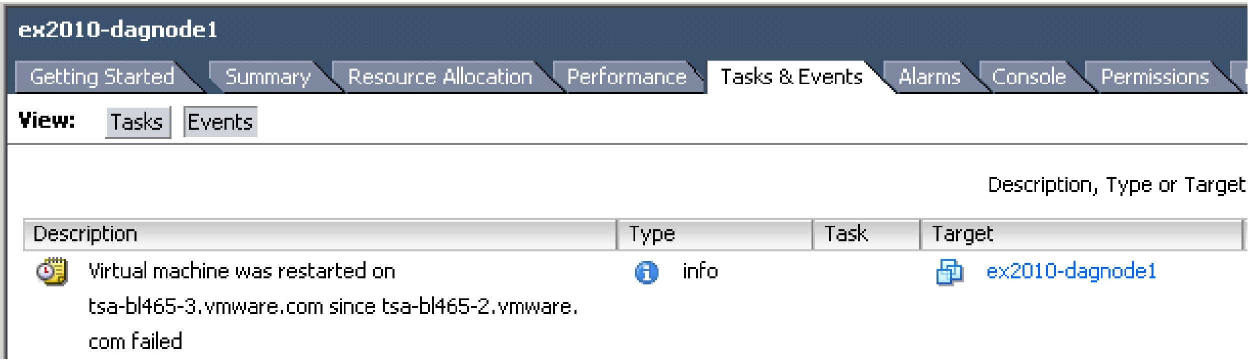
tsa-bl465-2.vmware.com

Virtual machine 'ex2010-casht2' on host 'tsa-bl465-2.vmware.com' would violate a virtual machine - host affinity rule.

The vMotion is blocked due to the **Must run on hosts in group** DRS rule.

3. Initiate a host failure for every host in a host group.
   a. Shut down both hosts in the host group.
   b. Verify that virtual machines are powered on.
   c. Verify that any anti-affinity rules are enforced.

Figure 14. HA Obeying Anti-Affinity Rules and Restarting Virtual Machine on Next Host



ex2010-dagnode1

Description	Type	Task	Target
Virtual machine was restarted on tsa-bl465-3.vmware.com since tsa-bl465-2.vmware.com failed	info		ex2010-dagnode1

The **Should run on hosts in group** rule allows ex2010-dagnode1 to be rebooted on a host in a different group, while keeping both DAG nodes separate.

4.5 HA Results

VMware HA is often the only line of defense for many applications against hardware or even guest OS failure. The simplicity of HA configuration and its OS and application-agnostic nature make it the perfect complement to any virtualized application. When deploying a clustered application on vSphere it may seem that there is no added benefit to combining VMware HA with the clustered application, but HA automates tasks that otherwise must be performed manually. This test addresses the benefits VMware HA can provide to clustered applications.

In this scenario, Exchange 2010 DAG nodes are enabled for VMware HA. After a host failure the failed DAG node automatically powers on and re-establishes Exchange level high availability. Without VMware HA an administrator would have to be notified of the failure, log in, and manually power on the virtual machine. The test results show that from the time the host shutdown is initiated to the time the Exchange databases are back in a protected state is about three minutes. The scenario was tested as follows:

1. Automated power-on of DAG node.
    a. HA enabled for both DAG nodes.
    b. vSphere host powered off.
    c. DAG fails over. Exchange service is functional for users.
    d. Failed DAG node is automatically powered back on by VMware HA.
    e. Databases are brought back into synch
    f. DAG-level redundancy is restored although a host is still down.

The following table shows the tasks and their duration during the test.

Table 7. VMware HA Timeline

Task	Time (Elapsed Time)
vSphere host powered off initiated	3:29:34 PM (0:00:00)
Host failure detected	3:30:06 PM (0:00:32)
Ex2010-dagnode1 fails	3:30:14 PM (0:00:40)
Ex2010-dagnode1 restarted by VMware HA	3:30:24 PM (0:00:50)
DAG node failure detected by ex2010-dagnode2	3:30:34 PM (0:01:00)
Databases begin to mount on ex2010-dagnode2	3:30:41 PM (0:01:07)
All databases mounted	3:30:43 PM (0:01:09)
Ex2010-dagnode1 OS running	3:31:07 PM (0:01:33)
Ex2010-dagnode1 begins copying and replaying log files	3:32:15 PM (0:02:41)
Ex2010-dagnode1 completes replication and replay of logs for all databases	3:32:36 PM (0:03:02)

5. Guidance and Best Practices

We tested various configurations both at the vSphere level and at the guest OS level. A number of settings were found to be beneficial and may be considered best practices.

5.1 Jumbo Frames

Standard Ethernet frames are limited to a length of 1500 bytes (slightly larger actually, but that's outside the scope of this paper). Jumbo frames can contain a payload of up to 9000 bytes. Support for jumbo frames on vmkernel ports was added to vSphere 4.0 for both ESX and ESXi. This added feature means that large frames can be used for all vmkernel traffic including vMotion.

Using jumbo frames reduces the processing overhead to provide the best possible performance by reducing the number of frames that must be generated and transmitted by the system. Though not part of the formal test plan, we had an opportunity to test vMotion of DAG nodes with and without jumbo frames enabled. Results showed that, with jumbo frames enabled for all vmkernel ports and on the vNetwork Distributed Switch, vMotion migrations completed successfully. During these migrations no database failovers occurred and there was no need to modify the cluster heartbeat setting.

The use of jumbo frames requires that all network hops between the vSphere hosts support the larger frame size; this includes the systems and all network equipment in between. Switches that do not support or are not configured to accept large frames will drop them. Routers and L3 switches may fragment the large frames into smaller frames that must then be reassembled and can cause performance degradation.

More information on vSphere and support for jumbo frames can be found in the VMware Networking Blog article *Jumbo Frames in vSphere 4.0*.

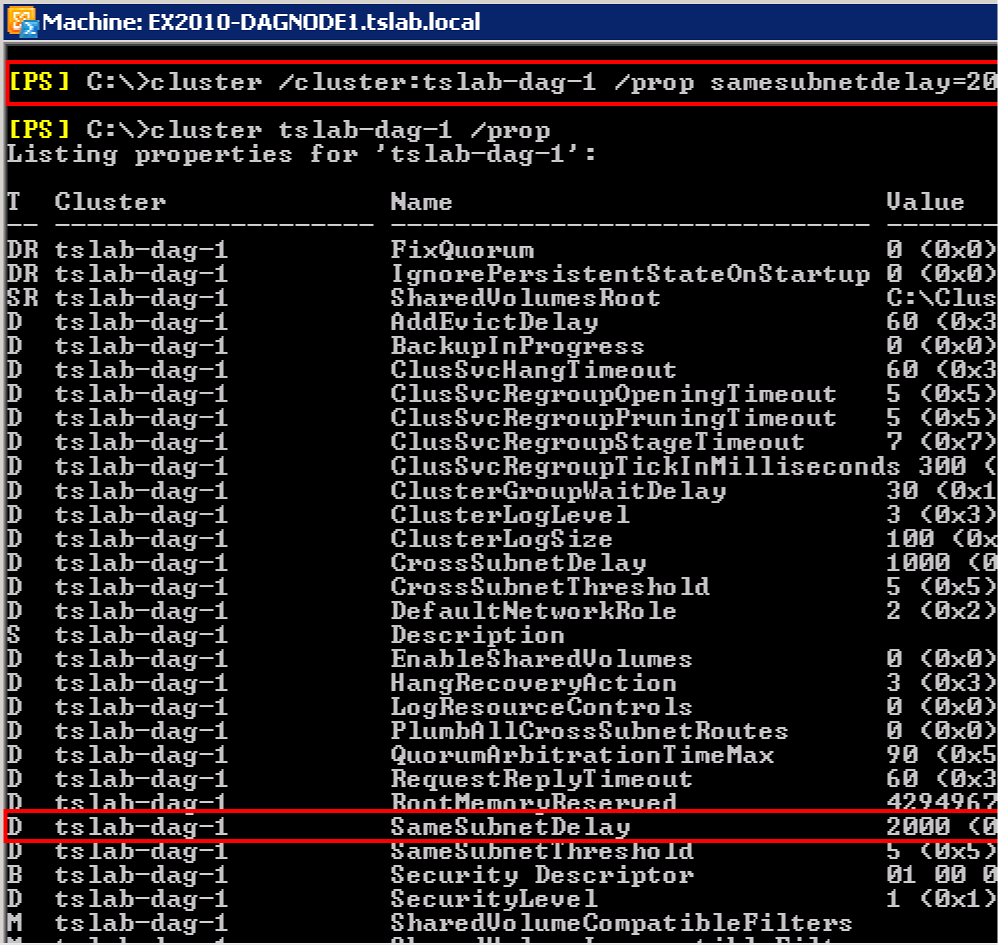
5.2 Cluster Heartbeat Settings

To establish a baseline during the vMotion tests no changes were made to the default configuration of the database availability group. Each virtual machine was configured with 24GB of memory, and with the change rate of memory pages, due to the load generation tool running, the application stun time was just enough to cause database failovers when jumbo frames were not used. However, the vMotion migrations completed 100% of the time and database replication continued uninterrupted. It was clear that the cluster heart beat interval was too short to support no-impact vMotion migrations in our test environment.

To support vMotion migrations during heavy usage we adjusted the `samesubnetdelay` parameter of the cluster from the default of 1000ms to 2000ms. This parameter controls how often cluster heartbeat communication is transmitted. The default threshold for missed packets is five, after which the cluster service determines that the node failed. By increasing the transmission period to two seconds and keeping the threshold at five intervals we were able to perform all of the vMotion tests (when jumbo frames were not used) with no database failovers.

Note Microsoft recommends using a maximum value of 10 seconds for cluster heartbeat timeout. In our configuration the maximum recommended value is used by configuring a heartbeat interval of two seconds (2000 milliseconds) and a threshold of five (default).

Figure 15. Windows Failover Cluster Heartbeat settings



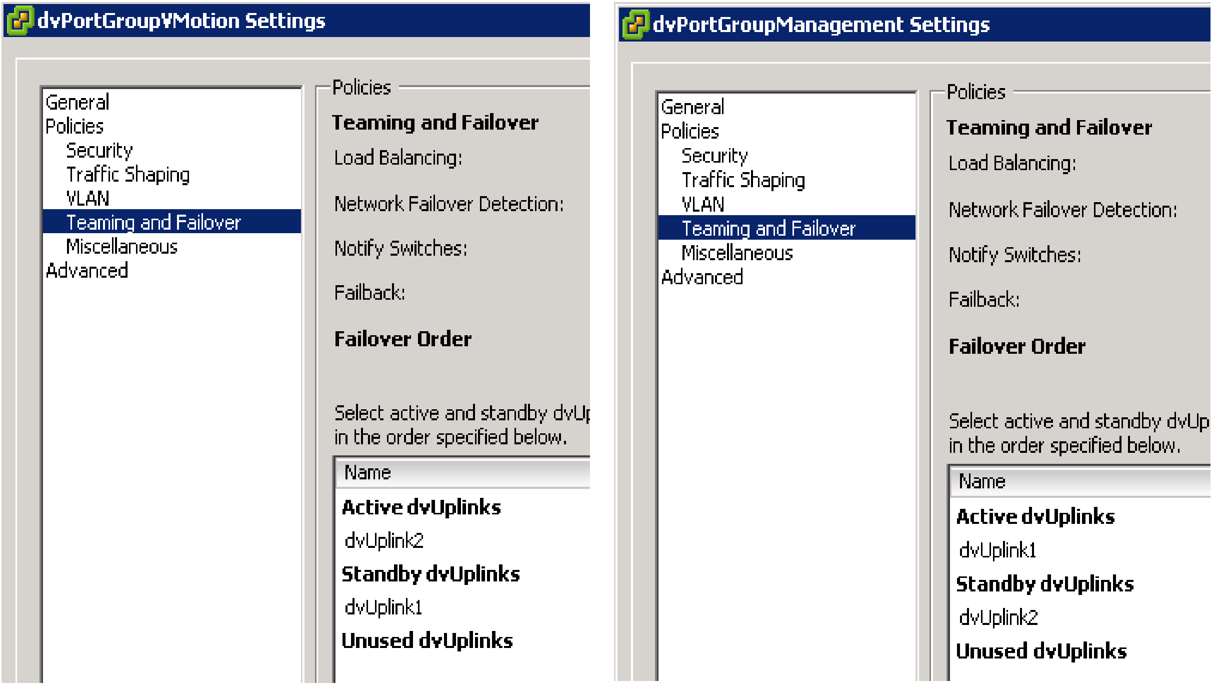
Machine: EX2010-DAGNODE1.tslab.local

```
[PS] C:\>cluster /cluster:tslab-dag-1 /prop samesubnetdelay=20

[PS] C:\>cluster tslab-dag-1 /prop
Listing properties for 'tslab-dag-1':

T  Cluster              Name                           Value
-- -------------------- ------------------------------ -------
DR tslab-dag-1          FixQuorum                      0 (0x0)
DR tslab-dag-1          IgnorePersistentStateOnStartup 0 (0x0)
SR tslab-dag-1          SharedVolumesRoot              C:\Clus
D  tslab-dag-1          AddEvictDelay                  60 (0x3
D  tslab-dag-1          BackupInProgress               0 (0x0)
D  tslab-dag-1          ClusSvcHangTimeout             60 (0x3
D  tslab-dag-1          ClusSvcRegroupOpeningTimeout   5 (0x5)
D  tslab-dag-1          ClusSvcRegroupPruningTimeout   5 (0x5)
D  tslab-dag-1          ClusSvcRegroupStageTimeout     7 (0x7)
D  tslab-dag-1          ClusSvcRegroupTickInMilliseconds 300 (
D  tslab-dag-1          ClusterGroupWaitDelay          30 (0x1
D  tslab-dag-1          ClusterLogLevel                3 (0x3)
D  tslab-dag-1          ClusterLogSize                 100 (0x
D  tslab-dag-1          CrossSubnetDelay               1000 (0
D  tslab-dag-1          CrossSubnetThreshold           5 (0x5)
D  tslab-dag-1          DefaultNetworkRole             2 (0x2)
S  tslab-dag-1          Description
D  tslab-dag-1          EnableSharedVolumes            0 (0x0)
D  tslab-dag-1          HangRecoveryAction             3 (0x3)
D  tslab-dag-1          LogResourceControls            0 (0x0)
D  tslab-dag-1          PlumbAllCrossSubnetRoutes      0 (0x0)
D  tslab-dag-1          QuorumArbitrationTimeMax       90 (0x5
D  tslab-dag-1          RequestReplyTimeout            60 (0x3
D  tslab-dag-1          RootMemoryReserved             4294967
D  tslab-dag-1          SameSubnetDelay                2000 (0
D  tslab-dag-1          SameSubnetThreshold            5 (0x5)
B  tslab-dag-1          Security Descriptor            01 00 0
D  tslab-dag-1          SecurityLevel                  1 (0x1)
M  tslab-dag-1          SharedVolumeCompatibleFilters
```

5.3 vMotion Interfaces

A general best practice for vMotion is to have dedicated interfaces because of the heavy traffic that is sent over the vMotion network. In our lab, we used a distributed virtual switch with a port group dedicated to management traffic and one dedicated to vMotion traffic. The switch was configured with two uplink interfaces which are physically connected to two separate Ethernet switches. Using this configuration we found that about 50% of the vMotion traffic had to traverse multiple physical switches. During vMotion migration we typically saw a steady receive/transmit rate of 900 Mbps. Having dedicated NICs gives the best performance while making sure we don't interfere with other management traffic. By explicitly configuring dvUplink2 (vmnic0) as active to the vMotion port group and configuring dvUplink1 (vmnic1) as standby we can be sure that all vMotion traffic stays local to the switch. The opposite was performed on the management port group with dvUplink1 being configured as active and dvUplink2 as standby.

Figure 16. vMotion and Management Portgroup Uplink Preference



dvPortGroupVMotion Settings

General
Policies
Security
Traffic Shaping
VLAN
Teaming and Failover
Miscellaneous
Advanced

Policies

Teaming and Failover

Load Balancing:

Network Failover Detection:

Notify Switches:

Failback:

Failover Order

Select active and standby dvUp
in the order specified below.

Name

Active dvUplinks
dvUplink2
Standby dvUplinks
dvUplink1
Unused dvUplinks

dvPortGroupManagement Settings

General
Policies
Security
Traffic Shaping
VLAN
Teaming and Failover
Miscellaneous
Advanced

Policies

Teaming and Failover

Load Balancing:

Network Failover Detection:

Notify Switches:

Failback:

Failover Order

Select active and standby dvUp
in the order specified below.

Name

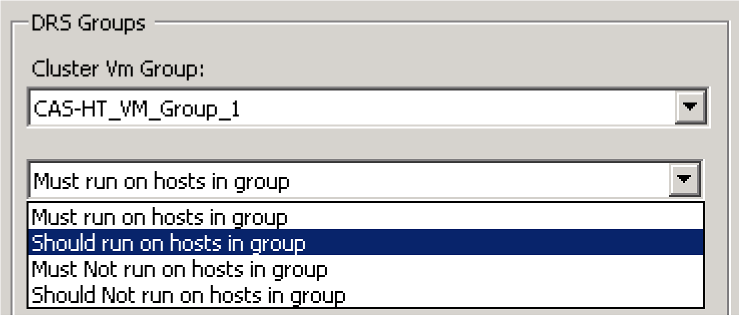
Active dvUplinks
dvUplink1
Standby dvUplinks
dvUplink2
Unused dvUplinks

For applications that are sensitive to the short stun time that a vMotion migration creates while migrating the virtual machine to the new host it is recommended that vMotion traffic be as contained as possible. By adjusting our primary vMotion NICs we were able to keep all vMotion traffic within the same physical switch, ensuring that the migration was occurring in as short a time frame as possible.

5.4 Use DRS "Should run on hosts in group" Rules

DRS affinity rules that link virtual machine groups to host groups are used to make sure virtual machines stay on a particular set of hosts; for example, when virtual machines are on separate blade chassis or racks. In the **Must run on hosts in group** mode, if all hosts in a host group fail the virtual machines will stay powered off. When configured as a **Should run on hosts in group** rule DRS attempts to keep the virtual machines on the desired hosts. If all hosts in the host group fail, the virtual machines can be restarted on hosts that are not in the host group. In most cases this is the preferred approach for creating rules so that even if there is a complete failure of hosts in a group all virtual machines can be restarted (assuming there are sufficient resources available).

Figure 17. DRS Rules Selection



DRS Groups

Cluster Vm Group:

CAS-HT_VM_Group_1

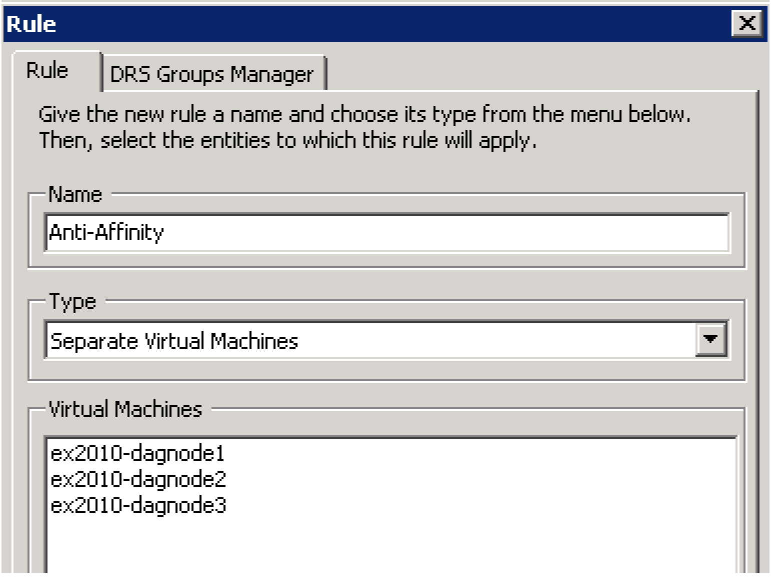
Must run on hosts in group

Must run on hosts in group
Should run on hosts in group
Must Not run on hosts in group
Should Not run on hosts in group

5.5 Anti-Affinity Rules

When deploying clustered virtual machines consider creating anti-affinity rules to keep members of the cluster on separate hosts. When combined with virtual machine and host groups, all resources in the vSphere cluster can be fully utilized while making sure that no single host failure causes a prolonged outage.

Figure 18. DRS Affinity Rules



Give the new rule a name and choose its type from the menu below.
Then, select the entities to which this rule will apply.

Name: Anti-Affinity

Type: Separate Virtual Machines

Virtual Machines:
ex2010-dagnode1
ex2010-dagnode2
ex2010-dagnode3

6. Conclusion

The availability requirements of every Exchange deployment vary based on the business requirements and SLAs. Database availability groups provide the highest level of protection, including data redundancy and application awareness. When combined with vSphere features such as vMotion, DRS, and HA, Exchange uptime can be increased and management of the vSphere environment can remain consistent. Benefits of these features include:

- Workload mobility – vMotion provides the ability to migrate running workloads with no user downtime. vSphere administrators use vMotion proactively to migrate virtual machines during maintenance windows. Application administrators do not need to be involved as the migration is transparent to the running applications and operating systems.
- Increased uptime – VMware HA provides application- and operating system-agnostic availability. Within a vSphere cluster host monitoring adds a layer of protection not available in physical deployments. If a vSphere host fails, the virtual machines running on that host restart on another host. This can dramatically shorten the time to resolution for both standalone and clustered virtual machines.
- Enforced virtual machine placement – Using DRS rules enables administrators to define where a virtual machine should and shouldn’t run, and which virtual machines must be kept together or separate. Using a **Should run on hosts in group** rule provides the best level of availability, keeping redundant virtual machines separated, but allowing them to run on the same set of hosts in case of a more extensive failure.

By following the recommended best practices and guidance customers can be confident that the same capabilities they are accustomed to in other vSphere environments are available in their Exchange 2010 virtualized environment.

7. References

- *VMware.com: Exchange 2010 on VMware – Best Practices*

  http://www.vmware.com/files/pdf/exchange-2010-on-vmware-best-practices-guide.pdf
- *VMware.com: Exchange 2010 on VMware – Availability and Recovery Options*

  http://www.vmware.com/files/pdf/exchange-2010-on-vmware-availability-and-recovery-options.pdf
- VMware Networking Blog: *Jumbo Frames in vSphere 4.0*

  http://blogs.vmware.com/networking/2010/03/jumbo-frames-in-vsphere-40.html
- VMware KB: *Microsoft Clustering on VMware vSphere: Guidelines for Supported Configurations*

  http://kb.vmware.com/kb/1037959
- MSFT Exchange Team Blog: *Announcing Enhanced Hardware Virtualization Support*

  http://blogs.technet.com/b/exchange/archive/2011/05/16/announcing-enhanced-hardware-virtualization-support-for-exchange-2010.aspx
- MSFT TechNet: *Exchange 2010 System Requirements*

  http://technet.microsoft.com/en-us/library/aa996719.aspx
- MSFT TechNet: *Configure Heartbeat and DNS Settings in a Multi-Site Failover Cluster*

  http://technet.microsoft.com/en-us/library/dd197562(WS.10).aspx