HP Smart Array 6402 controller online volume expansion with Microsoft Exchange Server 2003

Executive summary

In this whitepaper, HP demonstrates how to increase storage capacity of an existing HP Smart Array 6402 array and logical drive hosting an online Microsoft® Exchange Server 2003 mail store database without interrupting service to the Exchange Server 2003. This paper also examines performance of the array expansion process while varying database size and server load.

Target audience

This document is intended for systems administrators and integrators with experience in Exchange Server 2003 administration and Microsoft Windows® Server 2003 administration.

Overview

Many organizations face the challenge of explosive storage growth of corporate e-mail. As an Exchange Server 2003 administrator, it is probable that at some point you have been (or will be) faced with the task of increasing storage with Exchange Server 2003. This process can be an intensively manual task requiring significant downtime associated with deleting the volume, creating a new larger volume, and restoring the database(s) onto the new volume. In this paper, HP demonstrates how to expand the capacity of volumes and logical drives without incurring any downtime of the servers running Microsoft Windows® Server 2003 or Exchange Server 2003. Typically, there are two reasons to increase a storage array: to increase capacity, or increase performance. In today's messaging environment, personal mailboxes are reaching into the 500MB – 1GB size. As the propensity for conducting business via e-mail increases, users are required to retain a higher percentage of e-mail for their records. In many cases, the corporate e-mail users tend to use their e-mail inboxes almost as a file server. At some point in the lifecycle of an Exchange Server 2003, additional storage capacity will be required to support the rapidly increasing storage demands of the end users. Capacity is not the only reason an organization may need to add physical disks to an array. In some cases, it may be necessary to add physical disks to an array to increase I/O performance. With the availability of large capacity 300GB SCSI disk drives, the number of spindles required to meet the I/O demands often exceeds the number of spindles required for capacity.

Why expand?

Capacity bound vs. I/O bound

There are two basic reasons a company would be required to expand an array and logical drive: first, the disk volume is capacity bound and more disk space is required, or second, I/O performance has degraded and the volume is spindle bound, requiring additional spindles to increase I/O performance. A performance degradation problem can be resolved by adding spindles to the array and logical drive. Disk I/O bottlenecks can occur at several locations along the I/O path – including the controller, host adapter, and disk spindles. With the large capacity drives available today, it is not uncommon to find under-configured I/O subsystems. The configurations using large capacity disk drives meet the capacity requirements with too few spindles to meet the I/O demands. Adding physical disk drives (spindles) to the configuration resolves the I/O demand problem; however, it can result in an excessive amount of unused capacity. Corporations do not like to see excessive amounts of unused capacity in their disk subsystems. A good example of this would be a RAID0 array of four 300GB 15K RPM SCSI drives. This would result in 1200 GB of usable capacity. If the Exchange Server 2003 users had mailbox quotas of 100MB each, the assumption would be that this RAID0 array could host 1200 users. The reality is that this configuration would be I/O or spindle bound. Assuming that the average corporate e-mail user load equals one I/O per user per second, the array

would have to support 1200 I/Os per second. A typical 15K RPM SCSI drive can support approximately 150 I/Os per second; this array with four 15K RPM drives could only support 600 I/Os per second, approximately half of the required throughput. The result would be an array configuration that is I/O bound with too few spindles. This I/O bottleneck could be resolved by adding four more spindles to the array. The additional spindles would double the capacity of the array leaving a significant amount (1.2TB) of excess unused capacity.

Test configuration

The test configuration consisted of a single HP ProLiant DL580 G2 server and an HP Smart Array 6402 controller connected to a single HP StorageWorks 30 Modular Smart Array (MSA30) storage enclosure. The storage enclosure was populated with fourteen 36GB 15K SCSI hard disk drives. A single array was created using 10 of the 14 SCSI disk drives. A logical drive was created on the array using a RAID1+0 configuration.

Figure 1. Hardware Configuration

Volume expansion procedures

HP Smart Array controllers include the drive array capacity expansion and volume extension features. These features allow an organization to increase the capacity of an existing array and extend the logical drive without incurring any downtime. As a precaution, a full backup should be performed prior to expanding or extending an array to reduce the risk of data loss. In this configuration, the

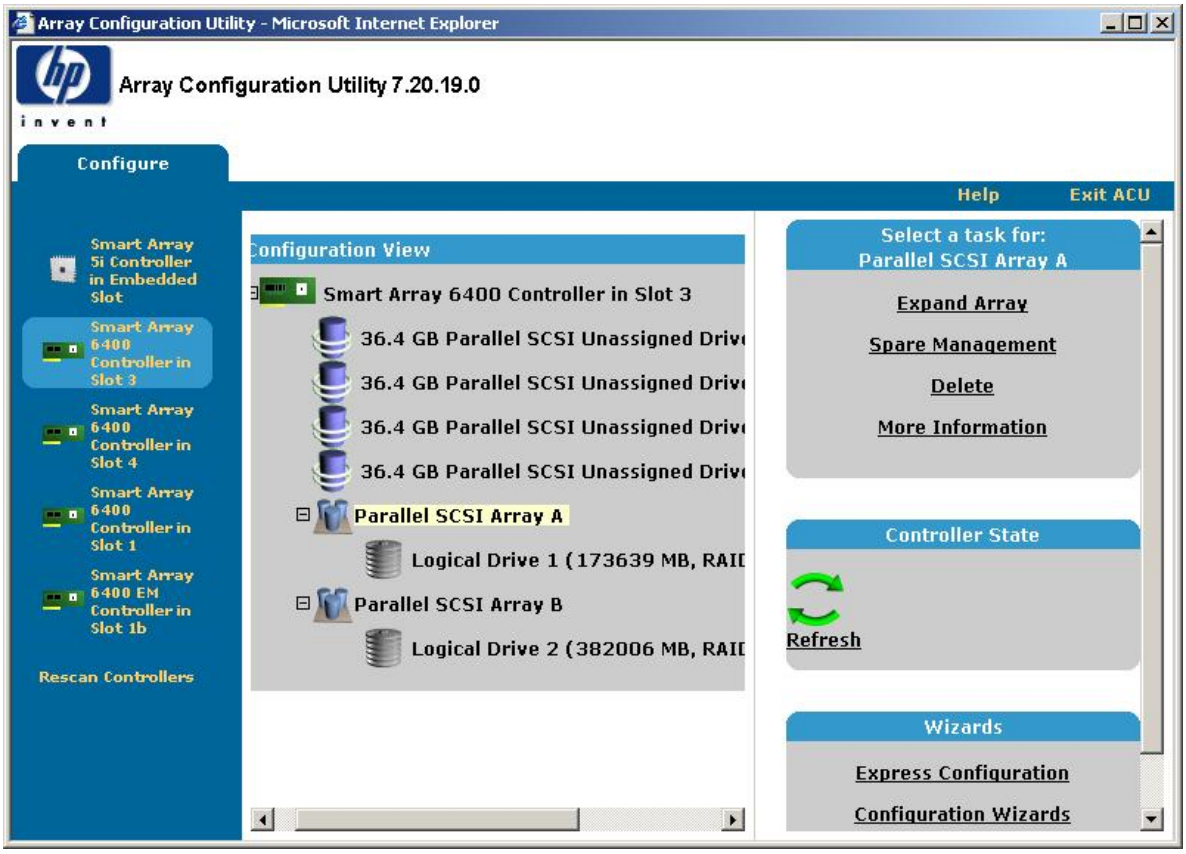
existing 10-member disk array will be expanded from ten 36GB 15K RPM SCSI disk drives to fourteen 36GB 15K RPM SCSI disk drives. The RAID1+0 logical drive will then be extended from 167GB to 237GB. The steps involved in expanding the array and extending the logical drive include the following:

- Perform a backup of all data on the logical drive to be expanded
- Insert the new disks (if they are not already in enclosure)
- Run the HP Array Configuration Utility (ACU)
- Select the Array and choose **Expand**
- Select Logical Drive and choose **Expand**
- Wait for disk expansion to complete
- Use diskpart to select and extend partition

Array configuration utility

This section explains the procedures for expanding an existing array with 10 SCSI disk drives to 14 SCSI disk drives. Launch the HP Array Configuration Utility (ACU) and select the controller that hosts the array and logical drive that are targeted for expansion.

Figure 2. Array Configuration Utility - Array Selected



Highlight the array to be expanded and select **Expand Array** from the navigation panel titled "Select a task for: Parallel SCSI Array A", as shown in Figure 2. The ACU will then present a list of

unassigned drives that are available to expand the array. In this example, the four remaining drives in the StorageWorks MSA30 storage array were chosen to increase the size of the array.

The Expansion process can take a considerable amount of time depending upon the quantity of data residing on the drive and the current load placed on the controller. This factor will be examined in more detail later in this document. The status of the expansion process can be observed by using the ACU or the Microsoft Windows Server 2003 Event Log Monitor. Selecting **More Information** on the array during the expansion process will provide a status bar indicating the expansion progress. In Figure 3 below, the expansion process is 3% completed.

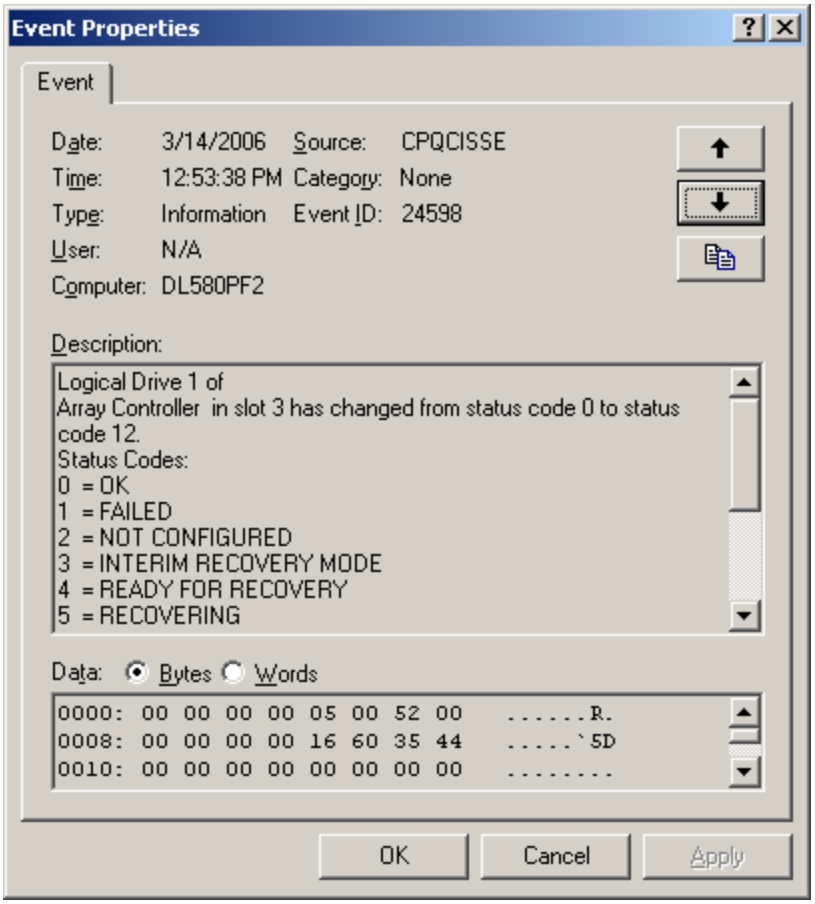
Figure 3. Array Configuration Utility - Expansion Status

The Microsoft Windows Server 2003 Event log also records controller events that document the expansion process. Figure 4 below shows that an event has been logged indicating the status of the controller has changed from code 0 to code 12. Code 12 indicates the controller is Queued for Expansion. Once the expansion process has begun, the controller will log another event indicating the status has changed to code 10, Expanding. When the expansion has successfully completed, an

event will be logged, indicating that the status has changed from code 10 to code 0. The Status Codes are listed below:

Status Codes:
0 = OK
1 = FAILED
2 = NOT CONFIGURED
3 = INTERIM RECOVERY MODE
4 = READY FOR RECOVERY
5 = RECOVERING
6 = WRONG PHYSICAL DRIVE REPLACED
7 = PHYSICAL DRIVE NOT PROPERLY CONNECTED
8 = HARDWARE IS OVERHEATING
9 = HARDWARE HAS OVERHEATED
10 = EXPANDING
11 = NOT YET AVAILABLE
12 = QUEUED FOR EXPANSION

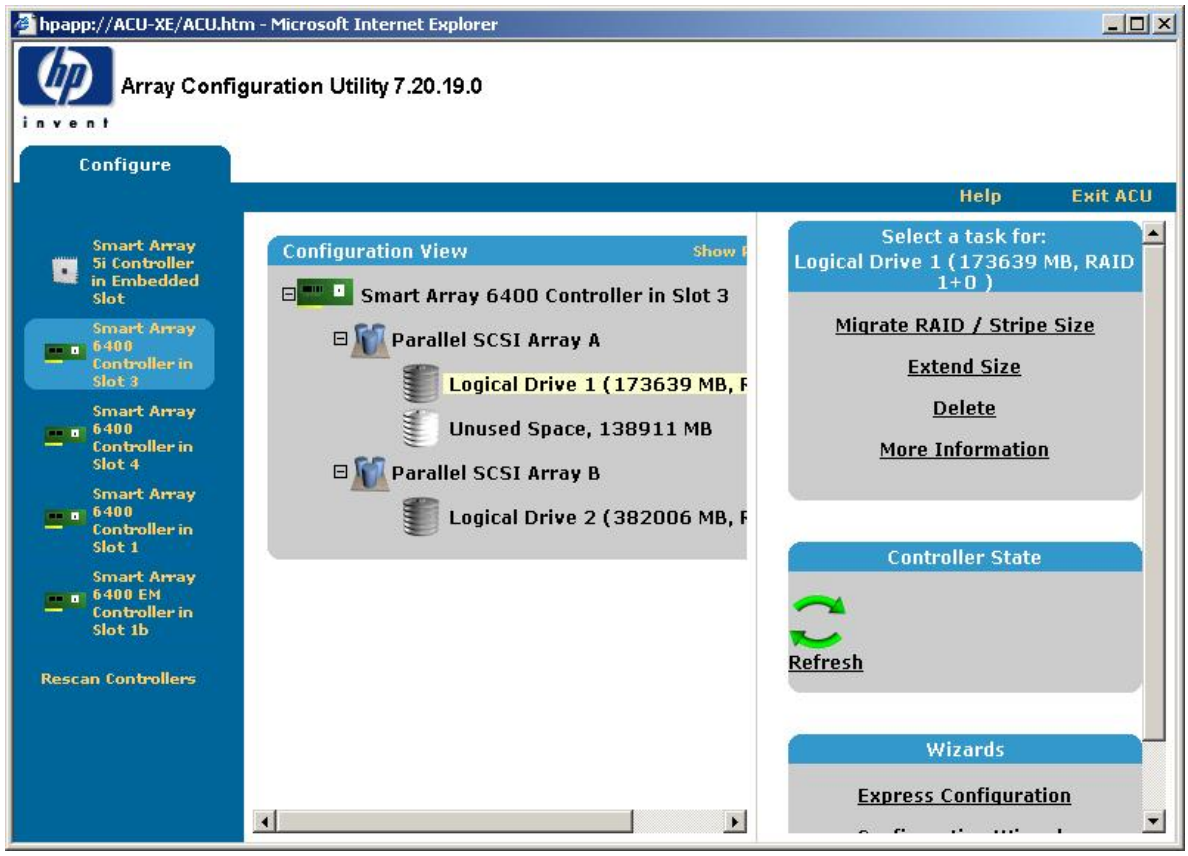
Figure 4. Microsoft Windows Server 2003 Event Log Controller Status Codes



Once the array expansion has completed, the ACU will be used to extend the logical drive. Below, in Figure 5, Logical Drive 1 is highlighted and ACU reports Parallel SCSI Array A has 138911MB of unused space. This unused space can be used to create a new logical drive or to extend an existing

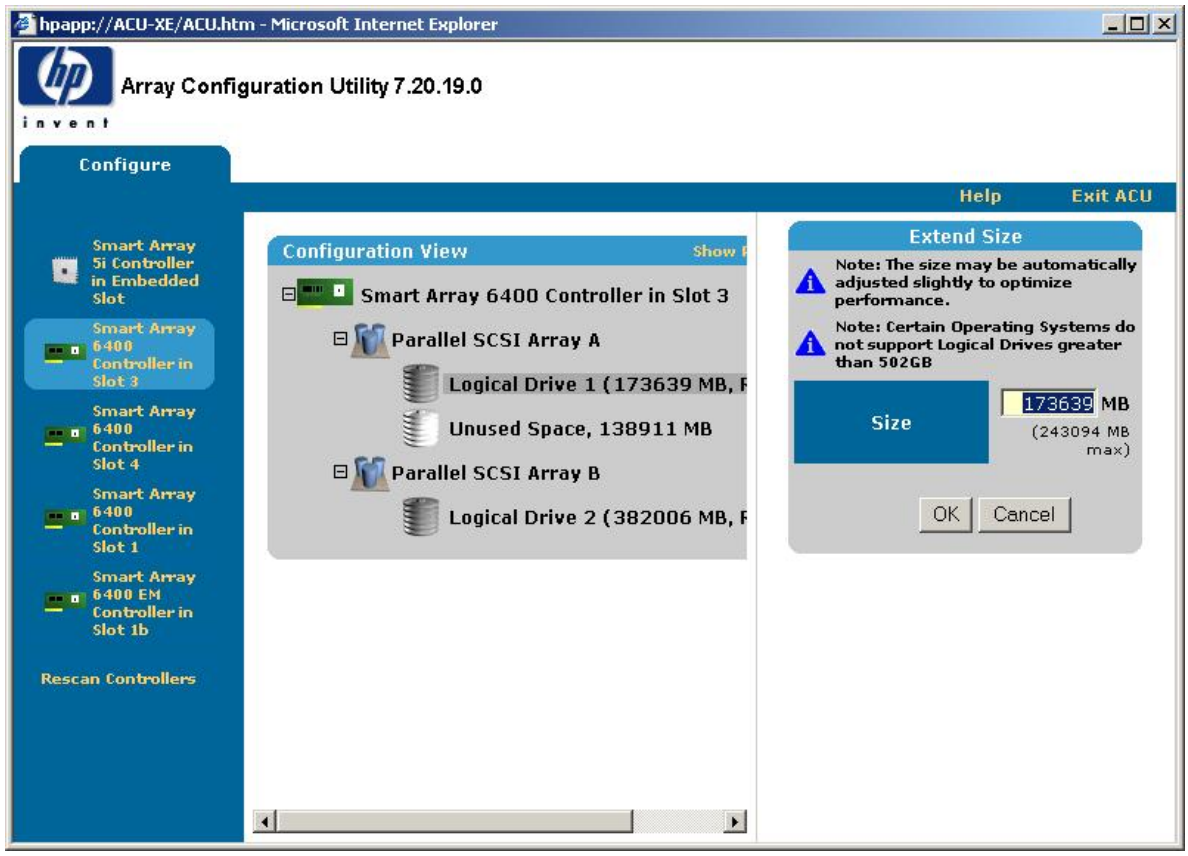
logical drive. In this document, all the unused space will be assigned to extend Logical Drive 1. In the navigational panel titled "Select a Task for: Logical Drive 1," select the **Extend Size** option.

Figure 5. Array Configuration Utility - Logical Drive Selected



The ACU will then present the operator with the Extend Size menu, as shown in Figure 6. The current size is displayed in the input box with a maximum size illustrated below the input box. Enter the size that the logical drive will be extended to. The extension process completes in seconds.

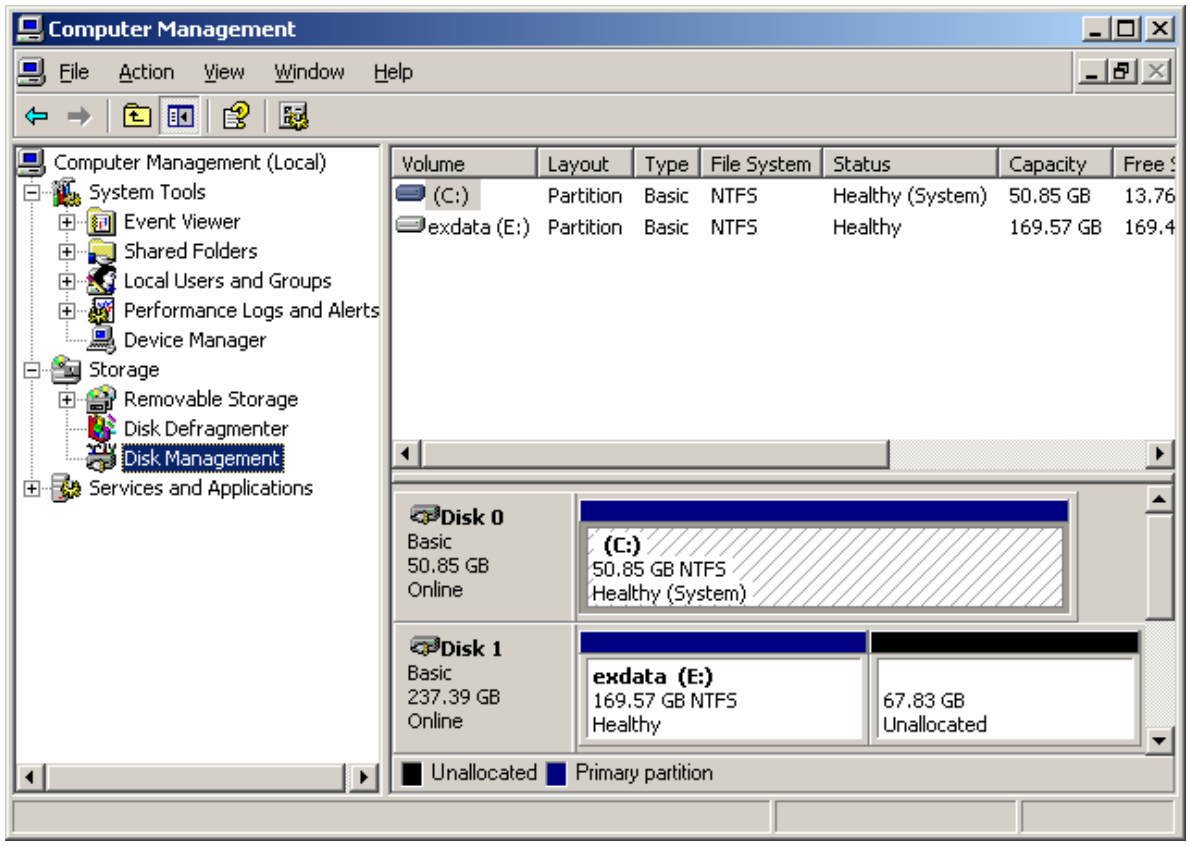
Figure 6. Array Configuration Utility - Logical Drive Extension



DISKPART and Disk Management utilities

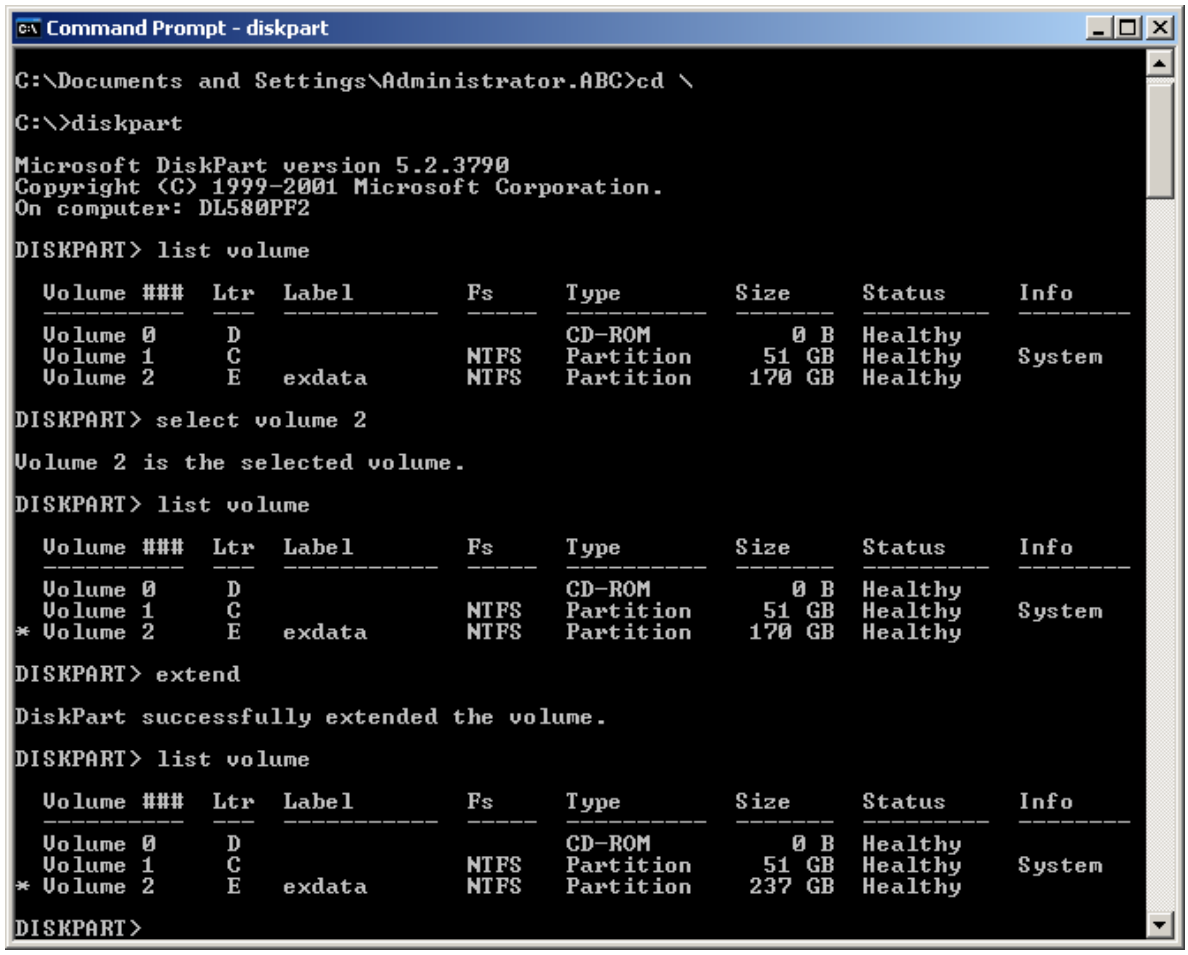
In the previous section, the ACU was used to expand the array from ten 36GB 15K RPM SCSI drives to fourteen 36GB 15K RPM SCSI drives. The ACU was then used to extend the size of the logical drive that is presented to the operating system (Microsoft Windows Server 2003 in this example). The next step is to determine how Microsoft Windows Server 2003 is going to allocate the additional capacity of the logical drive. Using the Disk Management utility, the additional capacity is viewed as unallocated space on Disk 1 as shown below in Figure 7. At this point, you could choose to create a new partition on Disk 1 using the Disk Management utility. However, this example will be to extend the existing partition to increase the size of the partition that hosts the Exchange Server 2003 Information store. The DISKPART utility will be used to perform the partition extension.

Figure 7. Disk Management before partition extension



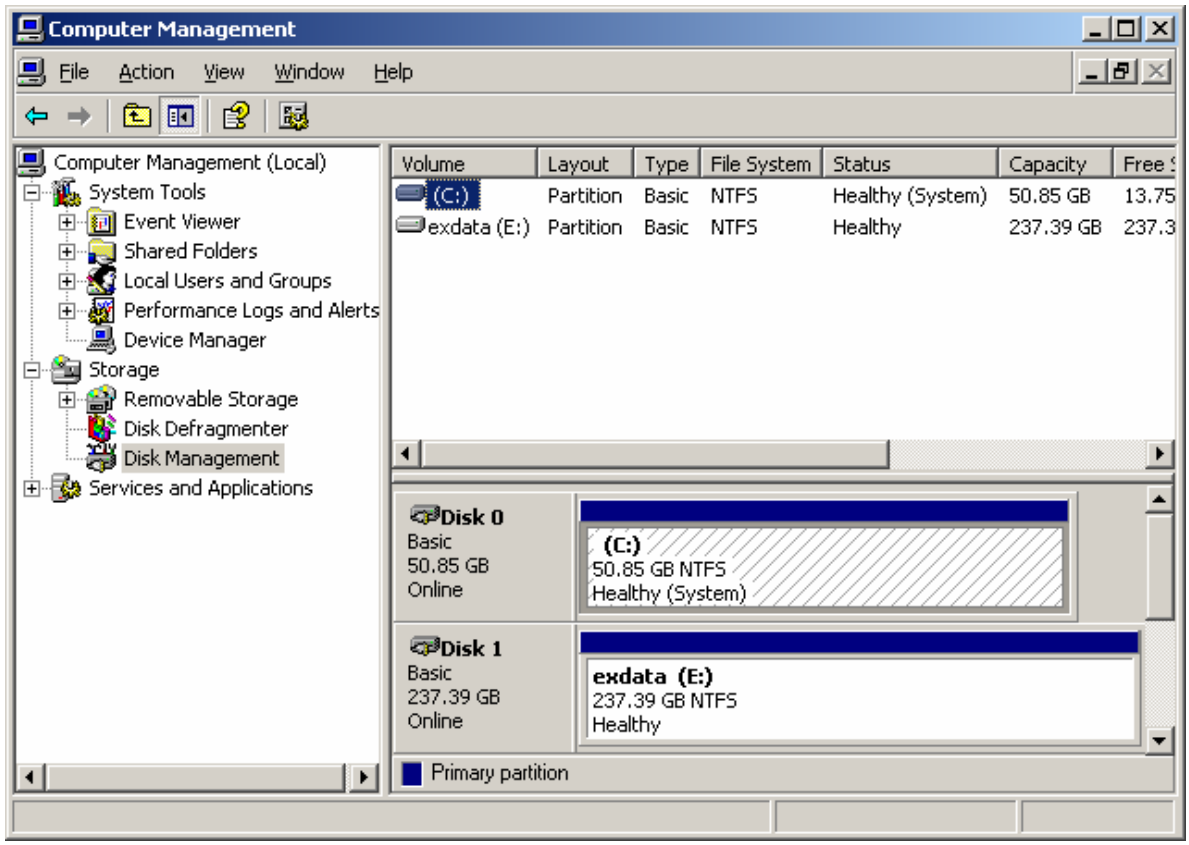
The DISKPART utility is a command line utility that will be used to extend the NTFS partition labeled exdata on Disk 1. Launch the DISKPART utility by typing **diskpart** at the command line; this will start the DISKPART utility shell shown below in Figure 8. Within the DISKPART command shell, enter **list volume** to display a list of volumes known to the system. Next, select the volume that will be extended by entering **select volume 2**. In the example in this paper, volume 2 is the volume that hosts the Exchange Server 2003 information store and is the volume that will be extended. Once a volume has been selected, it is designated by an asterisk ("*") to the left of the entry. Finally, to extend the selected volume, enter **extend** at the command line. The extension should occur immediately and the new size will be displayed in the size column.

Figure 8. DISKPART console commands



The partition has now been extended. Using the Disk Management utility, shown in Figure 9 below, the partition on Disk 1 now consumes the entire disk.

Figure 9. Disk Management after partition extension



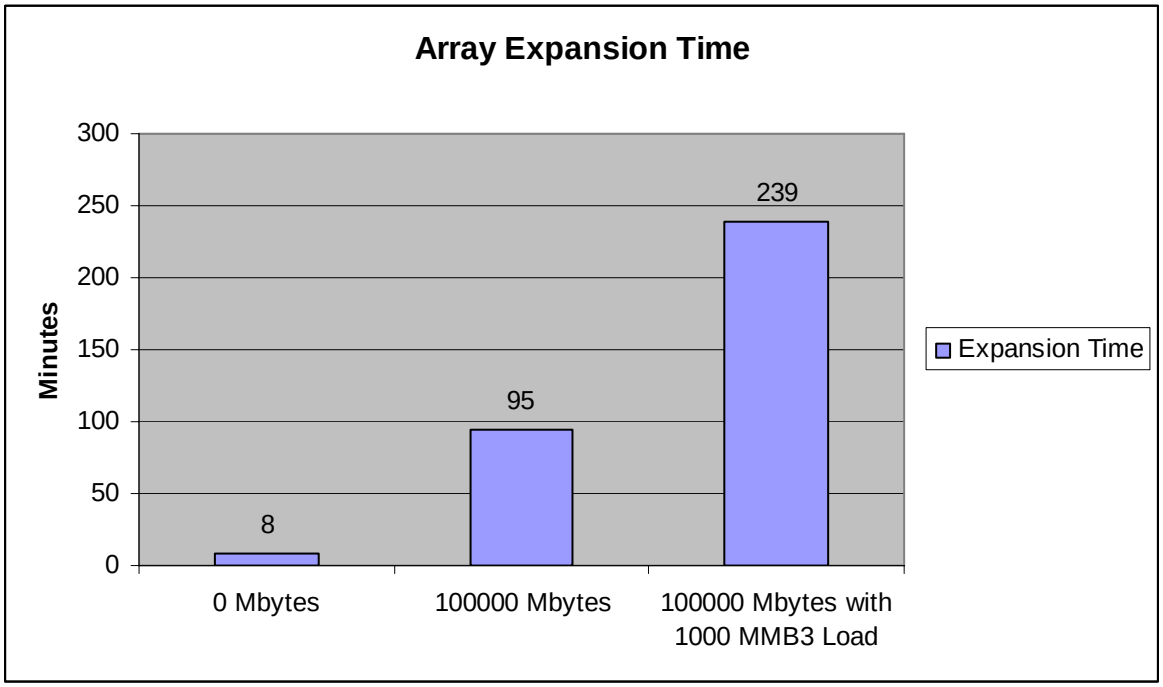
Expansion performance

Best practices would dictate that a volume maintenance or expansion be performed during off peak hours to minimize impact on the end user production environment. While this paper states that volume expansion can be performed online, there is a performance cost associated with the expansion process. This section will help identify the performance impact that an online expansion can have on the Exchange Server 2003 server and the expansion process. The length of time to perform the array expansion can vary depending upon the quantity of data on the array and I/O activity during the expansion process. HP performed a test with a 100GB Exchange Server 2003 Information store database. If the purpose of the expansion is to improve performance by adding additional spindles to the array, then the expansion process should be completed during off peak hours when the server is idle. The expansion process can also impact the performance of the logical disk. In order to quantify the performance impact of the expansion process, HP conducted a 1000 user LoadSim MMB3 test.

The length of time to complete the expansion process is the data point that is perhaps the most relevant to an administrator. As discussed earlier, the array expansion is the most time consuming process of the procedure; the ACU logical drive extension and partition extension both complete in just a few seconds. The graph in Figure 10 below illustrates the length of time to expand the array from 10 disk drives to 14 disk drives. The first data point shows the time to complete the expansion with no existing data on the drives; the array expansion is completed in only 8 minutes. The second data point shows the time required to complete the array expansion with 100GB of data on the drive; in this case, the array expansion process is completed in 95 minutes. Finally, the time required to complete the array expansion is measured with 100GB of data and an active load of 1000 LoadSim

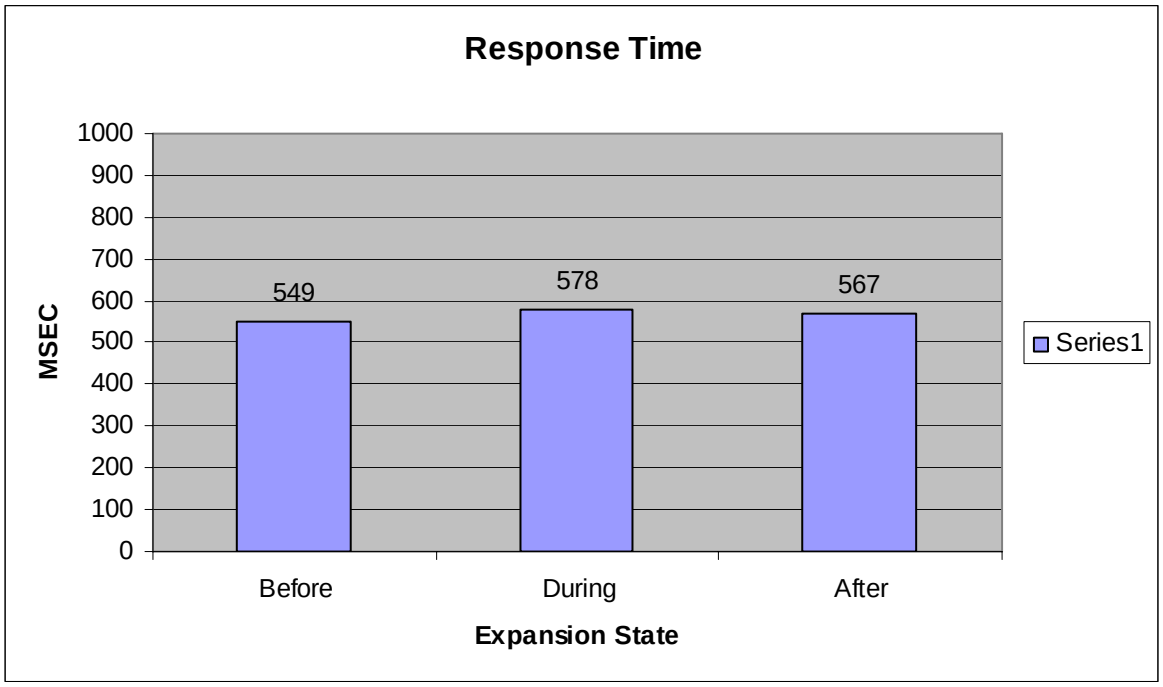
MMB3 users; the increased load and existing data resulted in the expansion process requiring 239 minutes to complete.

Figure 10. Array Expansion Time



Next, this document examines the impact that the expansion process has on the perceived client response time. As illustrated in the chart in Figure 11, the expansion process had little impact on response time. This system was configured with sufficient resources capable of handling the load with minimal impact on client response time.

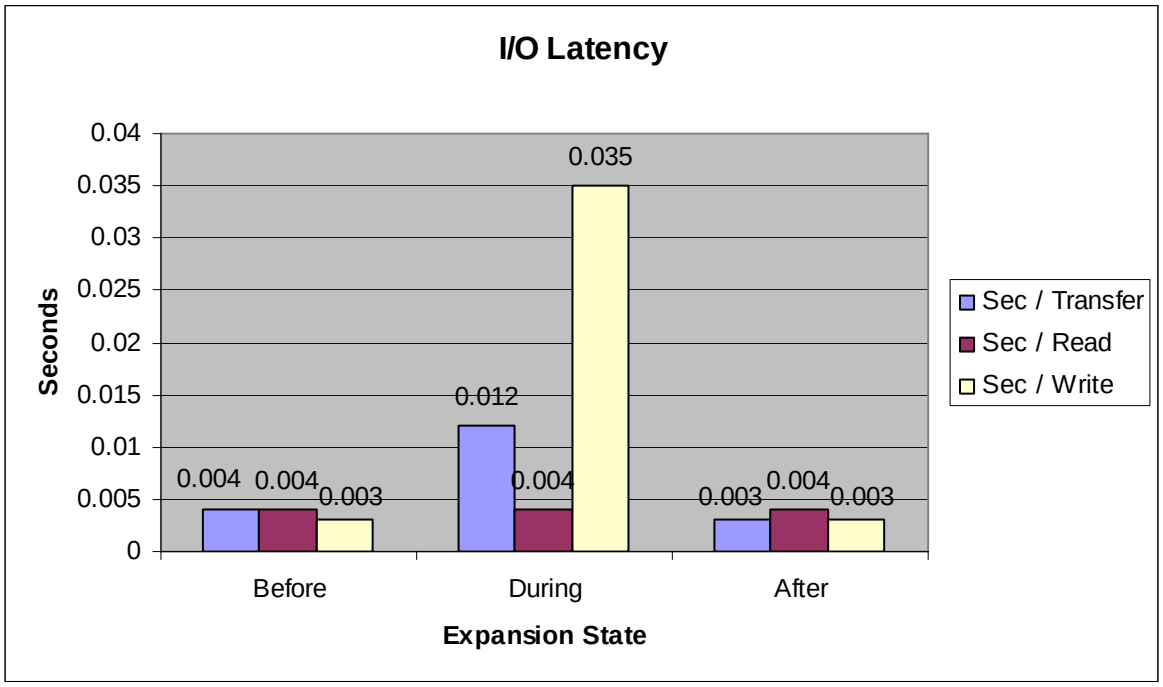
Figure 11. LoadSim Client Response Time



The following charts in Figures 12 - 14 examine I/O performance of the server while running LoadSim 1000 MMB3 simulated user profiles. The first test (Before) was conducted on the 10 disk member array. The second test (During) was conducted while the array was expanding from a 10 disk member array to a 14 disk member array, and the third test (After) was performed after the expansion process was completed on the 14 disk member array. I/O performance data was captured for each of these tests and is represented in the following graphs as Before, During, and After Expansion.

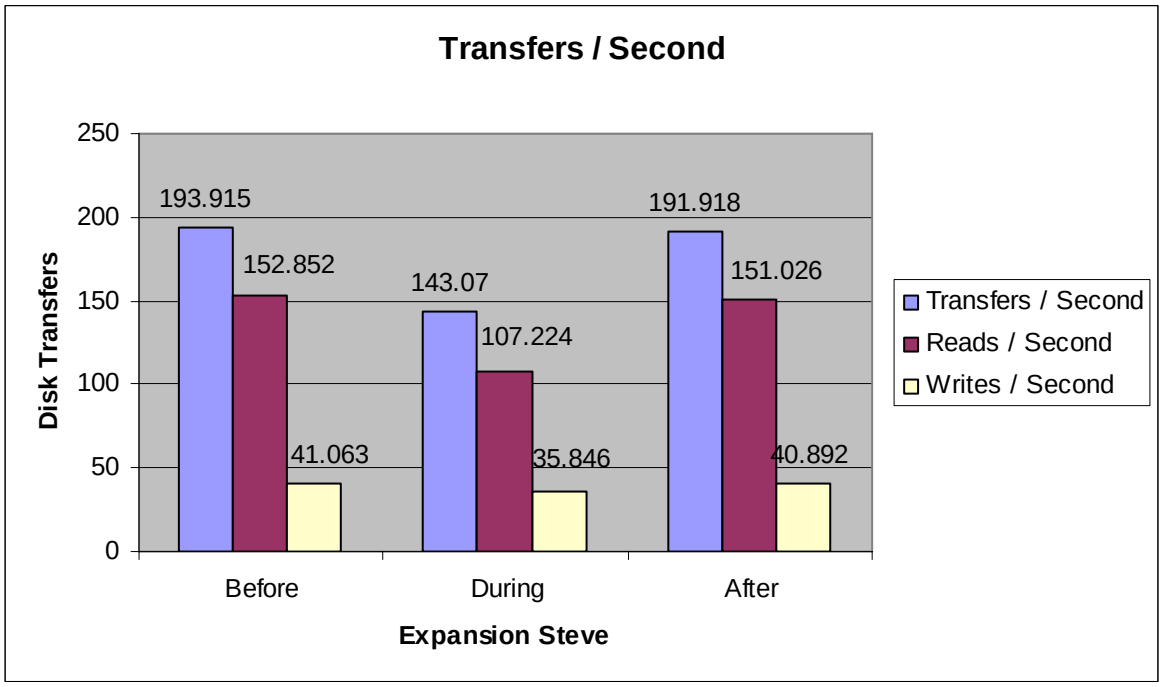
Low I/O latency is critical to a high performance Exchange Server 2003. Microsoft recommends that the average I/O latency not exceed 0.020 seconds with spikes remaining below 0.050 seconds. The chart in Figure 12 illustrates the fact that during the expansion process, the I/O write latency averaged 0.035 seconds; this is above the recommended best practice for a disk array hosting an Exchange Server 2003 information store database.

Figure 12. I/O Latency



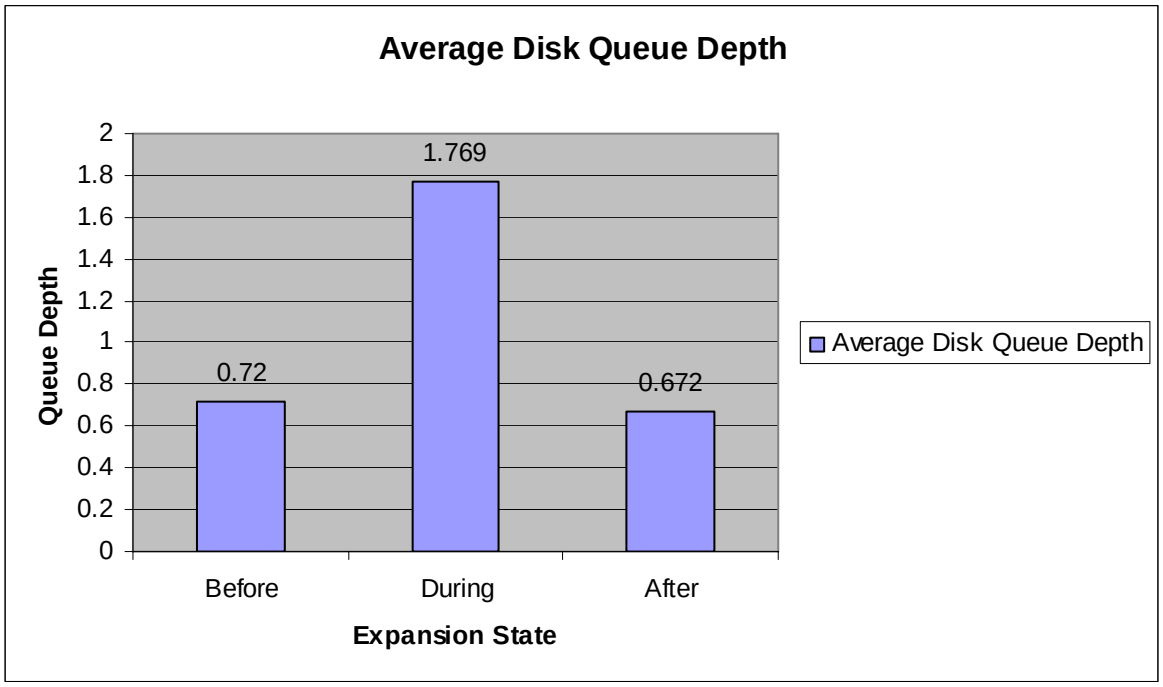
Increased I/O latency is typically associated with an increased I/O transfer rate. However, as illustrated in Figure 13 below, the transfer rate actually decreased during the expansion process. The transfer rate was measured at the OS level and the activity associated with expansion occurs at the controller card and disk subsystem level. The result of expanding the array as observed at the OS level is increased I/O latency and decreased throughput to the disks.

Figure 13. Disk Transfers per Second



The disk queue depth also increased during the expansion process. The average queue depth for the information store logical drive increased from 0.72 to 1.77 while the expansion process was in progress.

Figure 14. Disk Queue Depth



Summary

Volume expansion can occur while an Exchange Server 2003 server is online and does not require a reboot to recognize the additional capacity. An organization might consider expanding an array to increase capacity and improve I/O performance. The length of time required for the expansion process can vary from a few minutes to several hours depending upon the quantity of data on the drive and the I/O load. To complete the expansion process in the least amount of time, schedule the expansion for off hours when the server is lightly loaded. The expansion process has the greatest impact on write performance. Increased latency and disk queues were observed during the expansion process. Due to the significant activity at the controller level, I/O throughput was significantly reduced during the expansion process. Acceptable I/O latency levels for Exchange Server 2003 were observed during the expansion process. Increased I/O latency and reduced I/O throughput can result in increased user response time and poor Exchange Server 2003 server performance. Extending the logical drive via the ACU and extending the partition using DISKPART both complete in seconds with negligible impact on performance.

For more information

To learn more about HP products and services, please visit the HP website at www.hp.com

For more information about HP ProLiant servers, please visit www.hp.com/go/servers

For more information about HP Microsoft Exchange Server 2003 solutions, please visit: www.hp.com/solutions/exchange

HP ActiveAnswers website: www.hp.com/solutions/activeanswers/

6/2006-1

invent