

# Addressing Storage Arrays Greater Than 2 Terabytes



*Document v1.0*

## TABLE OF CONTENTS

Abstract .....	2
Introduction .....	2
The RAID Controllers .....	3
The Host Operating Systems .....	3
Windows .....	3
Linux .....	6
Quick Look: 2 terabytes table .....	8
AMCC suggested solutions .....	9
Conclusion .....	9
Quick Summary: OS vs controllers table .....	9
Glossary .....	10

©2004 by AMCC. All rights reserved. No part of this publication may be reproduced, stored in retrieval system, or transmitted in any form by any means, electronic, mechanical, photocopying or otherwise, without the prior written agreement or permission of AMCC. 455 West Maude Avenue; Sunnyvale, CA 94085 USA. 3ware is a registered trademark of 3ware, Inc. The AMCC logo is a trademark of AMCC Corporation. All other trademarks herein are the property of their respective owners.

AMCC assumes no responsibility for errors or omissions in this document, nor does AMCC make any commitment to update the information contained herein. Any information contained herein may change at any time without prior notice.

## Abstract

With increasing hard disk drive sizes, the available array and volume capacity of a system has grown considerably. This document discusses in detail the 2 terabytes limit under some operating systems and suggests workarounds currently available to achieve volume sizes greater than 2 terabytes.

This paper has been written for system integrators and systems administrators who are attempting to create and use arrays and volume sizes over 2 terabytes under Linux and Windows. The solutions for Linux and for Windows are different.

## Introduction

### Overcoming the 2 terabyte storage barrier

This white paper is targeted towards designing and implementing a RAID solution that can meet the growing capacity requirements that converge with higher density, higher capacity hard disk drives. The paper will aggregate RAID volumes to illustrate and support the creation of capacity well beyond the current ceiling of 2 terabytes per volume. Note: A volume is defined as a hardware RAID array (which is a combination of disk drives) that appears to the operating system as a single disk.

Several key design points must coalesce to make RAID volumes greater than 2 terabytes a reality.

- The RAID controller itself needs to support LBA (Logical Block Addressing) that can access more than 2 terabytes of physical capacity per array.
- The host or operating system needs to be able to format and address a logical volume or disk greater than 2 terabytes.
- Hard disk drives with drive sizes greater than 250 GB (preferably 300 GB or 400 GB) must be included in the array with 8 or 12-port controllers.

End users have asked the question — How can I configure a hardware RAID volume greater than 2 terabytes. Why can't I configure a hardware RAID volume greater than 2 terabytes?

Currently on the AMCC 3ware 7000 and 8000 series controllers as well as with some Linux and Windows operating systems the LBA is only 32 bits in length which provides for a maximum of slightly more than 2 terabytes of storage (see Note). However, the AMCC 3ware 9000 series controllers have 64-bit LBA, which can address more than 16 terabytes.

Note: 32-bits (a bit has two possible values) is represented as  $2^{32}$ , which equals 4,294,967,296 unique addresses. To determine the maximum capacity of a drive you multiply the (LBA max. x 512 bytes per LBA) for a total capacity of 2,199,023,256,000 bytes or a little over 2 terabytes of capacity.

## The RAID Controllers

RAID controllers have adapted to the growing capacity requirements by supporting more than a 32-bit LBA to accommodate drive capacities greater than 2 terabytes.

### AMCC 7000/8000 series

The 7506 and 8506 12-port controllers will support a maximum total capacity of 2 terabytes using 250 GB drives. The limitation of using more than 2 terabytes for a single array still exists on these controllers. The firmware on the 7506/8506 series supports 32-bit LBA, which limits the capacity to a maximum of 2 terabytes per array.

### AMCC (3ware) 9000S Solution

The AMCC 3ware 9000S series RAID controller breaks the 2 terabyte limit. The hardware supports a 64-bit LBA, which can satisfy array capacities up to 9 zettabytes.  $2^{64} \times 512$  bytes = 9,444,732,965,739,290,427,392 bytes. An AMCC 3ware 9000S 12-port series controller will support a maximum total capacity of 4.8 terabytes using 400 GB drives and when 500 GB drives are available then the array will support a maximum total capacity of 6 terabytes per controller.

## The Host Operating Systems

### Windows

Windows supports many file system types. This paper focuses on two that are relevant to the 2 terabyte limit, namely FAT32 and NTFS.

#### Maximum sizes on FAT32 volumes

Microsoft first introduced FAT32 with Windows 95 OSR2. The FAT32 file system can support arrays and volumes up to 2 terabytes in size because of the type of clusters used.

**Table: FAT32 Size Limits**

Description	Limit
Maximum file size	4 GB minus 1 byte ( $2^{32}$ bytes minus 1 byte)
Maximum volume size	32 GB (implementation)
Files per volume	4,177,920
Maximum number of files and subfolders within a single folder	65,534 (The use of long file names can significantly reduce the number of available files and subfolders within a folder.)

Although the FAT32 file system can support drives up to a standard theoretical size of 2 terabytes Windows 2000 and Windows XP cannot FORMAT a volume larger than 32 GB in size using the native FAT32 file system.

**Maximum sizes on NTFS volumes**

The maximum NTFS volume size as implemented in Windows is  $2^{32}$  clusters. For example, using 64-KB clusters, the maximum NTFS volume size possible in theory is 256 terabytes. Using the default cluster size of 4 KB for large volumes, the maximum NTFS volume size is 16 terabytes.

**Table: NTFS Size Limits**

Description	Limit
Maximum file size	Theory: 16 exabytes minus 1 KB ( $2^{64}$ bytes minus 1 KB) Implementation: 16 terabytes minus 64 KB ( $2^{44}$ bytes minus 64 KB)
Maximum volume size	Theory: $2^{64}$ clusters minus 1 cluster Implementation: 256 terabytes minus 64 KB ( $2^{32}$ clusters minus 1 cluster)
Files per volume	4,294,967,295 ( $2^{32}$ minus 1 file)

**Current AMCC Solution for Windows based systems**

**Recommended hardware for test set up:** AMCC 3ware 9500S-12 controller, twelve 300 GB or 400 GB or higher size SATA hard disk drives. (Note: 8506 has a hard limit at 2 terabytes, the 9500S does not.)  
System Requirements: Set up two, hardware RAID 5 arrays (6 drives each)  
Operating System Requirements (for single volume configurations): Convert to dynamic volumes.  
Use host based software striping across multiple 2 terabyte 3ware RAID units.

**Illustrated Examples**

NOTE: Total formatted (useable) capacity is always slightly lower than the theoretical projected value.

**Configuration 1:**

(Using SATA 300GB or 400GB Hard Disk Drives with the AMCC 3ware 9000S-12 Series RAID Controller —Hardware RAID 5 configuration)

1	2	3	4	5	6	1st H/W RAID 5 array
7	8	9	10	11	12	2nd H/W RAID 5 array

Array Capacity for 300GB drives =  $300 * (6 - 1) = 300 * 5 = 1.5$  terabytes approximately

Array Capacity for 400GB drives =  $400 * (6 - 1) = 400 * 5 = 2.0$  terabytes approximately

### Steps to create the volume:

1. Create two, hardware RAID 5 (6 drive) arrays.
2. Boot Windows Server 2003 and Launch LDM (Logical Disk Manager).
3. Convert the new disks e.g. Disk 0 and Disk 1 from "basic" to "dynamic".
4. Right click in the partition area and select "Create Partition".
5. Next select "Spanning"
6. Note: Add both Disk 0 and Disk 1 (both arrays) here.
7. Click next, to format the volume for a total capacity of over 2 terabytes.

1	2	3	4	5	6
7	8	9	10	11	12

Total Volume Capacity for 300 GB drives = 1.5 terabytes \* 2 = 3 terabytes approximately  
Total Volume Capacity for 400 GB drives = 2.0 terabytes \* 2 = 4 terabytes approximately

### Configuration 2 :

(using SATA 300 GB or 400 GB Hard Disk Drives with the AMCC 3ware 9000S-12 Series RAID Controller—Hardware RAID 0 configuration)

1	2	3	4	5	6	1st H/W RAID 0 array
7	8	9	10	11	12	2nd H/W RAID 0 array

Array Capacity for 300 GB drives = 300 \* (6) = 1.8 terabytes approximately  
Array Capacity for 400 GB drives = 400 \* (4) = 1.6 terabytes approximately  
(Note: Since Windows cannot fdisk beyond 2 terabytes, limit array capacity to or below 2 terabytes.)

### Steps to create the volume:

1. Create two, hardware RAID 0 (6 drive) arrays with 300GB drives or create three, RAID 0 (4 drive) arrays with 400GB drives.
2. Boot Windows Server 2003 and Launch LDM (Logical Disk Manager).
3. Convert the new disks e.g. Disk 0, Disk 1 and Disk 2 from "basic" to "dynamic".
4. Right click in the partition area and select "Create Partition".
5. Next select "Spanning"
6. Note: Add all Disk 0, Disk 1 and Disk 2 (all arrays) here.
7. Click next, to format the volume for a total capacity of over 2 terabytes.

1	2	3	4	5	6
7	8	9	10	11	12

Total Volume Capacity for 300 GB drives = 1.8 terabytes \* 2 = 3.6 terabytes approximately  
Total Volume Capacity for 400 GB drives = 1.6 terabytes \* 3 = 4.8 terabytes approximately

**Configuration 3:**

(A common configuration, NOT recommended by AMCC, since it does NOT provide the solution.)

1. Create a 12 drive, hardware RAID 5 array in the AMCC 3ware BIOS for a total array capacity of  $300 * (12 - 1) = 3.3$  terabytes or  $400 * (12 - 1) = 4.4$  terabytes.
2. Boot Windows Server 2003 and Launch LDM (Logical Disk Manager).
3. LDM will recognize the 3.3 or 4.4 terabyte array as one disk. E.g. Disk 0.
4. Windows limit on fdisk will not recognize over 2 terabytes although the controller provides arrays of over 2 terabytes to the operating system. This is a Windows limitation with a possible feature upgrade in SP1 or higher. Till then, AMCC recommends the "spanning" workaround or solution to higher capacity volumes.

**Configuration 4:**

(A common configuration, NOT recommended by AMCC, since it is another software RAID solution.)

1. Create each drive as a single mode (independent disk) in a 12 drive system/configuration.
2. Boot Windows Server 2003 and Launch LDM (Logical Disk Manager).
3. LDM will recognize all 12 drives. E.g. Disk 0 through Disk 11.
4. Convert all drives to Dynamic disk.
5. Software span all 12 drives for a total volume capacity over 2 terabytes.

Note: This example provides only software spanning and no hardware fault tolerance. Hence, this is not a recommended solution by AMCC.

Total Volume Capacity for 250 GB drives =  $250 \text{ GB} * 12 = 3.0$  terabytes approximately

Total Volume Capacity for 300 GB drives =  $300 \text{ GB} * 12 = 3.6$  terabytes approximately

Total Volume Capacity for 400 GB drives =  $400 \text{ GB} * 12 = 4.8$  terabytes approximately

**Linux**

Linux versions with kernels older than 2.4.18 will have a 1 terabyte limit on disk array size. If you create an array of 1 terabyte or larger, you may have difficulty installing Linux versions with kernels older than 2.4.18. Linux 2.4.18 and newer supports up to 2 terabytes and Linux kernel 2.6 compiled for AMD and Itanium 64-bit architectures breaks the 2 terabyte limit.

**Will Linux 2.4.18 kernels support capacities greater than 2 terabytes?**

Linux kernel 2.4.18 supports capacities up to 2 terabytes per disk target. Disk targets over 1 terabyte in capacity will be shown to have a negative (erroneous) capacity value at driver load time. This cannot be fixed by 3ware and is not a 3ware bug. Customers should check for kernel updates for fixes to these anomalies.

### **Will Linux kernels compiled for 32-bit architecture support capacities greater than 2 terabytes?**

Linux kernel 2.4.x if compiled for 32-bit architectures will NOT support capacities greater than 2 terabytes for a single software volume.

Linux kernel 2.6.x will support capacities greater than 2 terabytes when used with the “parted” tool. Please review the AMCC/3ware knowledge base article Q 11920 for more technical details on how to create over 2 terabyte arrays, since Linux partitioning tool “fdisk” does not properly support volumes larger than 2 terabytes. <http://www.3ware.com/kb/article.aspx?id=11920>

### **Will Linux kernels compiled for 64-bit architectures support capacities greater than 2 terabytes?**

Linux kernel 2.4.x if compiled for 64-bit architectures will NOT support capacities greater than 2 terabytes. Linux kernel 2.6.x will support capacities greater than 2 terabytes when used with the “parted” tool. (A native 64-bit architecture will support greater than 2 terabytes. A 64-bit extension architecture/platforms will have the same limitation as the 2.4.x kernel. These limitations are overcome by using GPT disk labels with the parted tool as explained in the knowledge base article Q 11920. <http://www.3ware.com/kb/article.aspx?id=11920>)

### **Booting to a disk array that is larger than 1 terabyte**

To boot from a disk array that is larger than 1 terabyte in capacity you need to install Linux to a disk drive or disk array smaller than 1 terabyte. After installation is complete, transfer the boot files from the smaller disk drive or array to the larger array and update your LILO or GRUB configuration. Consult your Linux documentation for more information.

### **Using Linux versions that use Anaconda**

Anaconda is a Linux installation program used by Red Hat and other Linux distributions. Anaconda that is standard with some 2.4 kernel distributions currently does not support disk arrays larger than 1 terabyte and fails to install properly. To overcome this limitation, install Linux to a RAID array or partition smaller than 1 terabyte. After installation is complete, then you can add the 1 terabyte or larger array to the system.

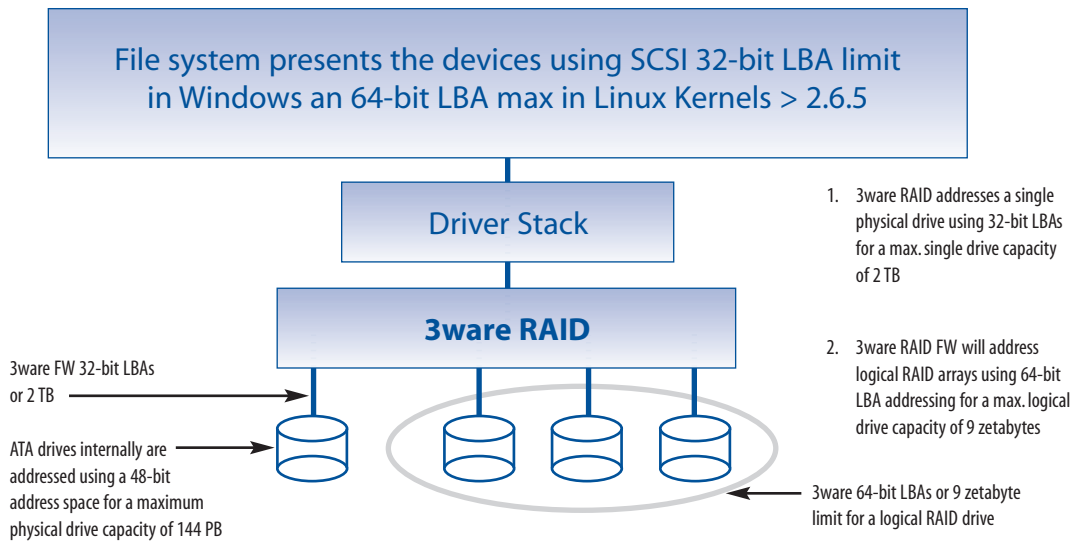
Make sure to check all releases and Linux distributions that use Anaconda prior to installation.

Note: Linux distributions that do not use Anaconda, such as SuSE, will also have this issue. This has been fixed in SuSE v8.0 and higher, which installs fine with disk arrays larger than 1 terabyte.

### **What about the ATA-6 48-bit requirement?**

This is included in the ATA-6 specification (circa 2000) and is required for drives that are greater than 137 GB.

ATA drives attached to an AMCC (3ware) controller are addressed as either a single physical drive or a logical RAID drive. The illustration below differentiates LBA usage.



### Quick Look: 2 terabytes table

AMCC/3ware RAID Controller	OS	Max Hardware Array Capacity	Operating System Volume Capacity
7000 / 8000 series 7.7.1 FW	W2K/XP (FAT File system)	2 TB	32 GB (Operating system hard limit)
7000 / 8000 series 7.7.1 FW	W2003/W2K/XP (Basic disk)	2 TB	2 TB (Operating system hard limit)
7000 / 8000 series 7.7.1 FW	W2003/W2K/XP (Dynamic disk)	2 TB	> 2 TB with AMCC Soln. (TBD: possible fix from Microsoft with SP1 or higher)
7000 / 8000 series 7.7.1 FW	Linux kernel 2.4 (2.4.17 or older)	1 TB	1 TB (Operating system hard limit)
7000 / 8000 series 7.7.1 FW	Linux kernel 2.4 (2.4.18 or newer)	2 TB	2 TB (Operating system hard limit)
7000 / 8000 series 7.7.1FW	Linux kernel 2.6.x	2 TB	> 2 TB Using "parted" tool with software RAID 0.
9000 series 9.0.2 FW	W2K/XP (Basic disk)	> 2 TB	2 TB (Operating system hard limit)
9000 series 9.0.2 FW	W2K/XP (Dynamic disks)	> 2 TB	> 2 TB with AMCC Soln. (TBD: possible fix from Microsoft with SP1 or higher)
9000 series 9.0.2 FW	Windows 2003 (Basic disk)	> 2 TB	2 TB (Operating system hard limit)
9000 series 9.0.2 FW	Windows 2003 (Dynamic disks)	> 2 TB	> 2 TB with AMCC Soln. (TBD: possible fix from Microsoft with SP1 or higher)
9000 series 9.0.2 FW	Linux kernel 2.4.x	> 2 TB	2 TB (Operating system hard limit)
9000 series 9.0.2 FW	Linux kernel 2.6.x	> 2 TB	> 2 TB Using "parted" tool.

## AMCC Suggested Solutions

Short Term Solution: (due November, 2004) AMCC is working on providing a new feature called “Auto Carving”. This is a straight non-user intervention method, which will transparently export modules of 2 terabyte LUNs to the operating system. This will help to maximize drive capacity back to the volume in a Windows operating system.

Long Term Solution: (due Q2, 2005) AMCC will provide a LUN management system that will provide flexibility with user intervention and options in a future release.

Another advantage of “Auto Carving” with respect to RAID 0 is that only one RAID 0 array creation is sufficient. Capacities beyond 2 terabytes will be reported in 2 terabyte LUNs to the operating system. Multiple RAID 0 array creations are not required as hard disk drive capacity increases.

## Conclusion

File systems over 2 terabytes are possible if all the right parts are in place. AMCC controllers currently support greater than 2 terabyte arrays and are prepared to accommodate more operating systems when they become available to support this feature. In summary, Windows has a 2 terabyte limit with fdisk. Hence, hardware arrays of over 2 terabytes can be recognized but will NOT be formatted by the operating system. This is an operating system limitation with a possible feature enhancement in SP1 or higher. Until then, AMCC recommends the “disk spanning” solution to achieve over 2 terabyte capacity volumes. The only drawback when using RAID 5 is the loss of capacity equal to one disk, due to the addition of the 2nd RAID 5 array. If capacity is of immediate concern, use the new “Auto Carving” feature.

Under the Linux operating system, the above is not an issue since the “parted” tool will help create partition sizes greater than 2 terabytes and the operating system will recognize them. Note: The “parted” tool is supported under Linux kernel version 2.6.x and 2.4.x but will help to format partitions over 2 terabytes only under Linux kernel version 2.6.x.

The following table reviews in brief, the operating systems and controllers that not only support but also can create volume sizes greater than 2 terabytes.

### Quick Summary: OS vs controllers table

	Linux kernel 2.4.x	Linux kernel 2.6.x	Windows 2000 SP4	Windows 2003 SP1	Windows 2003 SP2
7506/8506	No	No	Apply AMCC Solution	Apply AMCC Solution	TBD
95005	No	Use parted tool with GPT	Apply AMCC Solution	Apply AMCC Solution	TBD

## Glossary

### Cluster

In data storage, the smallest amount of disk space that can be allocated to hold a file. All file systems used by Windows organize hard disks based on clusters, which consist of one or more contiguous sectors. The smaller the cluster size, the more efficiently a disk stores information. A cluster is also called an allocation unit.

### Dynamic Disk

A physical disk that can be accessed only by Windows 2000 and Windows XP. Dynamic disks provide features that basic disks do not, such as support for volumes that span multiple disks.

### Dynamic Volume

A volume that resides on a dynamic disk. Windows supports five types of dynamic volumes: simple, spanned, striped, mirrored, and RAID-5.

### LBA

Logical Block Addressing (LBA) is a method of accessing hard disk drives. LBA stands for "logical block addressing". Instead of referring to locations by passing to the disk a cylinder, head and sector number (CHS addressing), the sectors are serialized so that each just has an integer number; 0, 1, 2, etc. up to the total number of sectors on the disk.

### LUN

Stands for Logical Unit Number. It is a unique identifier, which is used to distinguish between devices that share the same bus. Devices that request I/O transactions are considered as initiators. Devices that perform operations requested by initiators are called targets. Every target has the capacity to service up to eight other devices also known as logical units. These logical units are assigned numbers and hence, the term LUNs. Commands, which are sent to the controller, identify devices based on this number, the LUN

