

DiskMap

The need for documenting and preserving disk structures, such as the master boot record (MBR) and boot sector, in case of future disk corruption is clear. The Microsoft®Windows® 2000 tool DiskMap is the perfect tool for extracting the contents of these disk structures

DiskMap Overview

DiskMap is a command-line disk mapping tool that allows users to view the partition and logical volume structures of the disks. It displays a map of the disk and produces a report about the disk's configuration. It provides information about the disk characteristics and a description of each partition and logical volume on the disk.

DiskMap Limitations

The version of DiskMap included with the *Windows 2000 Resource Kit* does not support all the disk-based features found in Windows 2000.

No Dynamic Disk Support

Windows 2000 offers two types of disk configurations: basic disk and dynamic disk. The basic disk configuration uses the same disk structures found on all earlier versions of Windows NT, using primary partitions and logical volumes within extended partitions as containers of data. All disks are initially configured as basic disks. DiskMap fully supports basic disks.

Dynamic disk is new to Windows 2000. It uses a disk management database located at the end of the disk to manage the structure of disk volumes. Each physical disk has to be upgraded to use dynamic disk.

DiskMap cannot read the disk management database. It gathers its disk data from reading the partition table and any extended partition tables that may exist. Users who upgrade their disks to dynamic disk might not be able to use DiskMap to get an accurate portrayal of the structures on those disks.

Any partition structures that existed before disks are upgraded to dynamic disk are retained in the partition table. DiskMap is able to read and display this data. However, once dynamic disk is installed, users can add as many new volumes as space allows without the limits imposed by the partition table, and volumes can be easily extended and spanned beyond their initial configuration. None of these types of additions or changes are recorded in the partition table on dynamic disks, which inhibits DiskMap's ability to recognize new volumes, locate boot sectors, and identify the start and end sectors of individual volumes. For more information about basic and dynamic disks, refer to *Microsoft® Windows® 2000 Resource Kit Server Operations Guide*.

Running DiskMap

Table 1 shows the DiskMap command line options.

Table 1 DiskMap Options

Option	Description
/d<disk number>	Indicates the number of the physical disk to map.
/h	Indicates that the report should be generated in hexadecimal output. The default is decimal output.

Whether or not you specify the /h option, certain fields in the DiskMap report are always hexadecimal; others are always decimal as listed in Table 2. The fields in the DiskMap sample report are shown in Figure 1.

Table 2 DiskMap Report Field Format

Always hexadecimal	Always decimal
Signature	The megabyte (MB) portion of the DiskSize field
System ID	Partition Number

Not e The **Partition Number** field is a cumulative listing of all volumes on the disk, including all partitions (primary and extended) as well as all logical volumes contained within any existing extended partitions. While a disk may contain no more than four partitions, there can be as many logical volumes as disk space permits. When you run out of letters in the alphabet to assign to volumes (excluding the reserved drive letters A through C), you can use the new Windows 2000 feature volume mounting points to graft the structure and contents of one volume onto another volume's folder structure. For more information about volume mounting points, see the *Microsoft® Windows® 2000 Resource Kit Server Operations Guide*.

To run DiskMap from the Windows 2000 command prompt, type:

diskmap /d<disk number> [/h]

Not e DiskMap will not run from a Microsoft® MS-DOS® prompt.

DiskMap Report Format

Figure 1 shows a disk configuration report for a computer with one hard disk. The information under the heading MBR is from the MBR. The information under the extended boot record (EBR) headings is from the extended partition tables.

```
Cylinders  HeadsPerCylinder  SectorsPerHead  BytesPerSector  MediaType
      1757             255           63             512         12
TrackSize = 32256, CylinderSize = 8225280, DiskSize = 14451816960 (13782MB)

Signature = 0x31373132
StartingOffset      PartitionLength  StartingSector  PartitionNumber
```

*	32256	3142024704	63	1
	3142056960	1883589120	6136830	2
	5025646080	2303078400	9815715	3
	7328756736	2623832064	63	4
	9952621056	2409974784	63	5
	12362628096	2089188864	63	6

MBR:

*	Starting			Ending			System	Relative	Total
	Cylinder	Head	Sector	Cylinder	Head	Sector	ID	Sector	Sectors
	0	1	1	381	254	63	0x07	63	6136767
	382	0	1	610	254	63	0x06	6136830	3678885
	611	0	1	890	254	63	0x0b	9815715	4498200
	891	0	1	1023	254	63	0x05	14313915	13912290

EBR: (sector 14313915)

Starting			Ending			System	Relative	Total
Cylinder	Head	Sector	Cylinder	Head	Sector	ID	Sector	Sectors
891	1	1	1023	254	63	0x07	63	5124672
1023	0	1	1023	254	63	0x05	5124735	4707045
0	0	0	0	0	0	0x00	0	0
0	0	0	0	0	0	0x00	0	0

EBR: (sector 19438650)

Starting			Ending			System	Relative	Total
Cylinder	Head	Sector	Cylinder	Head	Sector	ID	Sector	Sectors
1023	1	1	1023	254	63	0x0b	63	4706982
1023	0	1	1023	254	63	0x05	9831780	4080510
0	0	0	0	0	0	0x00	0	0
0	0	0	0	0	0	0x00	0	0

EBR: (sector 24145695)

Starting			Ending			System	Relative	Total
Cylinder	Head	Sector	Cylinder	Head	Sector	ID	Sector	Sectors
1023	1	1	1023	254	63	0x06	63	4080447
0	0	0	0	0	0	0x00	0	0
0	0	0	0	0	0	0x00	0	0
0	0	0	0	0	0	0x00	0	0

Figure 1 DiskMap Report

Disk Geometry

DiskMap uses IOCTL_DISK_GET_DRIVE_GEOMETRY to determine the disk geometry. Windows 2000 uses the following disk geometry:

Cylinders	HeadsPerCylinder	SectorsPerHead	BytesPerSector	MediaType
1757	255	63	512	12

TrackSize = 32256, CylinderSize = 8225280, DiskSize = 14451816960 (13782MB)

Note The values of fields **TrackSize**, **CylinderSize**, **DiskSize**, **StartingOffset**, and **PartitionLength** are measured in bytes.

Different hardware might have different physical geometry because translation can be performed at either the hardware or the device-driver layer. It is important to know the geometry that Windows 2000 uses. You can use this information to repair a damaged partition table. For more information about running this procedure in DiskProbe, see the document titled “Troubleshooting with DiskProbe,” Dskprtrb.doc, which is installed with the Support Tools from the *Windows 2000* product CD into C:\Program Files\Support Tools.

The MBR and EBR sections on the report describe the contents of their respective partition tables. The MBR section describes the partition table contained on the first sector of the disk. Disks with no extended partitions have no EBR sections. Otherwise, there is one EBR section showing the extended partition table for each logical volume in the extended partition. For descriptions of the fields in each of these sections, refer to the *Microsoft® Windows® 2000 Resource Kit Server Operations Guide*.

The partition with an ASTERISK (*) is the active, primary partition and contains the files required to load an operating system such as Windows 2000. On a computer running Windows 2000 configured with basic disk, each disk can have up to four partitions, but only one primary partition will be marked active.

Hardcopy Reports

DiskMap can be used to make hardcopy reports for each disk. You can create a record of the DiskMap display by redirecting the output to a printer or file. Keep the record of the DiskMap display with the other configuration information that you maintain for your computer.

To create a record of the DiskMap display as a text file, at the command line type:

```
diskmap /d0 > disk0map.txt
```

Note Be sure to use a descriptive file name in order to note at a glance which disk the map represents.

DiskMap Errors

DiskMap might display various error messages. Some are easily corrected; others might indicate a hardware problem or data corruption. Table 5.3 describes the error messages that you might see. The *N* and *E* in the messages indicate numbers.

Table 5.3 DiskMap Tool Error Messages

Error Message	Explanation
BLOCKED: CreateFile() Failed /dN [Error E] Ensure that you have selected a valid drive number and that the selected drive is not	The attempt to open the physical disk <i>N</i> failed. Either another program, such as Disk Management or DiskProbe, has the disk open, or the value following / <i>d</i> is not valid (out of range or not a number).

locked by another process.	
Unable to get drive geometry using IOCTL_DISK_GET_DRIVE_GEOMETRY. [Error E]. This is a fatal error.	The call to DeviceIoControl() with the parameter IOCTL_DISK_GET_DRIVE_GEOMETRY failed. This error occurs when you remove a disk from a removable media device. Otherwise, the message might indicate a hardware failure (for example, a device that was available at startup is no longer available).
Unable to allocate memory.	DiskMap tried to allocate memory for a temporary buffer but no memory was available. Quit other running programs and try again.
Unable to get drive layout using IOCTL_DISK_GET_DRIVE_LAYOUT. Error: [Error E].	The call to DeviceIoControl() with the parameter IOCTL_DISK_GET_DRIVE_LAYOUT failed. When walking the partition table chain, the utility encountered an error. The error might indicate partition table corruption, hardware RAID array corruption, hardware failure, or a change in apparent disk geometry that makes portions of the volume inaccessible. For more information, refer to the <i>Microsoft® Windows® 2000 Resource Kit Server Operations Guide</i> .
Unable to read next EBR. Error: [Error E].	A call to ReadFile() failed. This is usually caused when DiskMap attempts to read past the end of the disk (for example, when the pointer to the next EBR is not valid). It can also indicate a bad sector at the EBR location.
Next EBR failed to pass sanity check.	This error indicates the Ending Head for each of the four partition table entries is not 0 or not equal to HeadsPerCylinder-1, or the last two bytes in the sector are not 0x55AA. The problem is either an error in the chain of EBRs or a bad sector at the EBR location.
Detected partition corruption.	The value in the Relative Sector field for a partition table entry is not equal to the SectorsPerHead value.

In general, if the partition size and layout displayed by DiskMap looks right, and the utility does not display any error messages, the partition tables are correct. When a partition table is written incorrectly, and the last EBR referenced is past the end of the disk, IOCTL_DISK_GET_DRIVE_LAYOUT fails. In this situation, DiskMap displays an error about IOCTL_DISK_GET_DRIVE_LAYOUT and the EBRs that it can read prior to the point of failure. Comparing a hard copy of a DiskMap that has no errors with the incorrect DiskMap display on your computer terminal can show you where the two EBR chains differ. You can restore the original partition information by correcting the EBR that has incorrect values by using a low-level disk editor such as DiskProbe. Typically, the remaining original EBRs are not damaged by incorrect data in the EBR chain. However, you might need to repair more than one EBR.

For questions or feedback concerning this tool, please contact rkinput@microsoft.com.
© 1985-2000 Microsoft Corporation. All rights reserved.

