

Filesystem Types

- Video:** Module Introduction
38 sec
- Video:** Review of Filesystems
1 min
- Reading:** Supplemental Reading for FAT32 File System
10 min
- Video:** Disk Partitioning and File System Essentials
2 min
- Video:** Windows: Disk Partitioning and Formatting a Filesystem
6 min
- Reading:** Supplemental Reading for Disk Partitioning and Formatting in Windows
10 min
- Video:** Windows: Mounting and Unmounting a Filesystem
37 sec
- Video:** Linux: Disk Partitioning and Formatting a Filesystem
5 min
- Video:** Linux: Mounting and Unmounting a Filesystem
3 min
- Reading:** Supplemental reading Mounting and Unmounting a Filesystem in Linux
10 min
- Video:** Windows: Swap
3 min
- Reading:** Supplemental Reading for Windows Paging
10 min
- Video:** Linux: Swap
2 min
- Reading:** Supplemental Reading for Linux Swap
10 min
- Video:** Windows: Files
5 min
- Reading:** Supplemental Reading on NTFS File System
10 min
- Video:** Linux: Files
2 min
- Video:** Windows: Disk Usage
3 min
- Reading:** Supplemental Reading for Windows Disk Usage
10 min
- Video:** Linux: Disk Usage
1 min
- Reading:** Supplemental reading for Linux Disk Usage
10 min
- Video:** Windows: Filesystem Repair
4 min
- Video:** Linux: Filesystem Repair
1 min
- Reading:** Supplemental Reading for Linux Filesystem Repair
10 min
- Discussion Prompt:** Disk Partitioning
10 min
- Practice Quiz:** Filesystem Types
9 questions
- Video:** Ben: The power of computers
1 min
- Reading:** Module 4 Glossary
10 min

Graded Assessments

Supplemental Reading for Windows Paging

Windows Paging Files

In this reading, you will learn about Windows paging files and their primary functions. You will also learn how to set the appropriate Windows paging file size. As an IT Support specialist, you may want to add or maintain page files to improve system performance. A paging file is an optional tool that uses hard drive space to supplement a system's RAM capacity. The paging file offloads data from RAM that has not been used recently by the system. Paging files can also be used for system crash dumps or to extend the system commit charge when the computer is in peak usage. However, paging files may not be necessary in systems with a large amount of RAM.

Page file sizing

Determining the size needed for a paging file depends on each system's unique needs and uses. Variables that have an impact on page file sizes include:

- System crash dump requirements - A system crash dump is generated when a system crashes. A page file can be allocated to accept the Memory.dmp. Crash dumps have several size options that can be useful for various troubleshooting purposes. The page file needs to be large enough to accept the size of the selected crash dump. If the page file is not large enough, the system will not be able to generate the crash dump file. If the system is configured to manage page dumps, the system will automatically size the page files based on the crash dump settings. There are multiple crash dump options. Two common options include:
 - Small memory dump:** This setting will save the minimum amount of info needed to troubleshoot a system crash. The paging file must have at least 2 MB of hard drive space allocated to it on the boot volume of the Windows system. It should also be configured to generate a new page file for each system crash to save a record of system problems. This history is stored in the Small Dump Directory which is located in the %SystemRoot%\Minidump file path.
 - To configure a small memory dump file, run the following command using the cmd utility:

Wmic recoveros set **DebugInfoType** = 3

- Alternatively, this option can be configured in the registry:

Set the **CrashDumpEnabled** DWORD value to 3

- To set a folder as the Small Dump Directory, use the following command line:

Wmic recoveros set **MiniDumpDirectory** = <folderpath>

- Alternatively, the directory option can be set in the registry:

Set the **MinidumpDir Expandable String Value** to <folderpath>

- Complete memory dump:** The option records the contents of system memory when the computer stops unexpectedly. This option isn't available on computers that have 2 or more GB of RAM. If you select this option, you must have a paging file on the boot volume that is sufficient to hold all the physical RAM plus 1 MB. The file is stored as specified in %SystemRoot%\Memory.dmp by default. The extra megabyte is required for a complete memory dump file because Windows writes a header in addition to dumping the memory contents. The header contains a crash dump signature and specifies the values of some kernel variables. The header information doesn't require a full megabyte of space, but Windows sizes your paging file in increments of megabytes.
 - To configure a complete memory dump file, run the following command using the cmd utility:

wmic recoveros set **DebugInfoType** = 1

- Alternatively, a complete memory dump file can be configured in the registry:

Set the **CrashDumpEnabled** DWORD value to 1

- To set a memory dump file, use the following command line:

wmic recoveros set **DebugFilePath** = <folderpath>

- Alternatively, the memory dump file can be set in the registry:

Set the **DumpFile Expandable String Value** to <folderpath>

- To indicate that the system should not overwrite kernel memory dumps or other complete memory dumps, which may be valuable for troubleshooting system problems, use the command:

wmic recoveros set **OverwriteExistingDebugFile** = 0

- Alternatively, the overwrite setting can be turned off in the registry:
 - Set the **Overwrite** DWORD value to 0

- Peak usage or expected peak usage of the system commit charge - The system commit limit is the total of RAM plus the amount of disk space reserved for paging files. The system commit charge must be equal to or less than the system commit limit. If a page file is not in place, then the system commit limit is less than the system's RAM amount. The purpose of these measurements is to prevent the system from overpromising available memory. If this system commit limit is exceeded, Windows or the applications in use may stop functioning properly. So, it is a best practice to assess the amount of disk storage allocated to the page files periodically to ensure there is sufficient space for what the system needs during peak usage. It is fine to reserve 128 GB or more for the page files, if there is sufficient space on the hard drive to dedicate a reserve of this size. However, it might be a waste of available storage space if the system only needs a small fraction of the reserved disk space. If disk space is low, then consider adding more RAM, more hard drive storage, or offload non-system files to network or cloud storage.
- Space needed to offload data from RAM - Page files can serve to store modified pages that are not currently in use. This keeps the information easily accessible in case it is needed again by the system, without overburdening RAM storage. The modified pages to be stored on the hard drive are recorded in the **\Memory\Modified Page List Bytes** directory. If the page file is not large enough, some of the pages added to the Modified Page List Bytes might not be written to the page file. If this happens, the page file either needs to be expanded or additional page files should be added to the system. To assess if the page file is too small, the following conditions must be true:
 - \Memory\Available MBytes indicates more physical memory is needed.
 - A significant amount of memory exists in the modified page list.
 - \Paging Files(*)% Usage (existing page files) are almost full.

Mark as completed

