[MS-FSA]: File System Algorithms

Errata below are for Protocol Document Version V30.0 – 2020/03/04.

Errata Published*	Description
2020/07/06	In Section 2.1.5.9.22, FSCTL_QUERY_FILE_REGIONS, added a new behavior note. Changed from: ● Set InputRegion.DesiredUsage = FILE_REGION_USAGE_VALID_CACHED_DATA for NTFS or Set InputRegion.DesiredUsage = FILE_REGION_USAGE_VALID_NONCACHED_DATA for ReFS Changed to: ● Set InputRegion.DesiredUsage = FILE_REGION_USAGE_VALID_CACHED_DATA for NTFS or Set InputRegion.DesiredUsage = FILE_REGION_USAGE_VALID_NONCACHED_DATA for ReFS<99> <99> Section 2.1.5.9.22: In Windows Server 2012 R2, InputRegion.DesiredUsage is set to FILE_REGION_USAGE_VALID_CACHED_DATA for ReFS.
2020/06/22	In Section 2.1.1.1, Per Volume, added definition for the NumberOfDataCopies field. Changed from: ● FreeSpareBlocks: A 32-bit unsigned integer indicating the available number of spare blocks. The following fields are specific to the ReFS object store: Changed to: ● FreeSpareBlocks: A 32-bit unsigned integer indicating the available number of spare blocks. ● NumberOfDataCopies: A 32-bit unsigned integer indicating the number of copies of redundant data that are available on this volume. A volume with redundant copies of data MUST set this to 2 or greater. A volume without redundancy MUST have a value of 1. For example, a 2-way mirrored volume would have 2 copies and a 3-way mirrored volume would have 3 copies. Volumes configured with RAID should have a value of 2 or larger depending on which raid configuration is used. The following fields are specific to the ReFS object store: In Section 2.1.1.6, Per Open, added definition for the ReadCopyNumber field. Changed from: ● UserSetAccessTime: A Boolean that is TRUE if a user has explicitly set File.LastAccessTime through this Open.

Errata Published*	Description
	● NextEaEntry: Contains a reference to the next FILE_FULL_EA_INFORMATION entry in File.ExtendedAttributes to be returned the next time FileFullEaInformation is called using this Open as defined in section 2.1.5.11.12.<39>

Changed to:

● UserSetAccessTime: A Boolean that is TRUE if a user has explicitly set File.LastAccessTime through this Open.

● ReadCopyNumber: A 32-bit unsigned integer which is initialized to a value of 0XFFFFFFFF. Identifies which copy of data should be read from a volume with redundant data (where Volume.NumberOfDataCopies > 1). The CopyNumber is zero based, meaning zero reads the 1st copy, 1 reads the 2nd copy, etc.

● NextEaEntry: Contains a reference to the next FILE_FULL_EA_INFORMATION entry in File.ExtendedAttributes to be returned the next time FileFullEaInformation is called using this Open as defined in section 2.1.5.11.12.<39>

In Section 2.1.5.2, Server Requests a Read, revised/added processing for the BytesToRead.

Changed from:

    ● Set BytesToRead to BlockAlign(ByteCount,Open.File.Volume.LogicalBytesPerSector).

    ● Read BytesToRead bytes from the disk at offset ByteOffset for this stream into OutputBuffer. If the read from the disk failed, the operation MUST be failed with the same error status.

    ● If RequestedByteCount > ByteCount, zero out OutputBuffer between ByteCount and RequestedByteCount.

Changed to:

    ● Set BytesToRead to BlockAlign(ByteCount,Open.File.Volume.LogicalBytesPerSector).

    ● Read BytesToRead bytes from the disk at offset ByteOffset for this stream into OutputBuffer. If Open. ReadCopyNumber != 0XFFFFFFFF then include this information in the read request to the disk to indicate which copy the data should be read from. If the read from the disk failed, the operation MUST be failed with the same error status.

    ● If RequestedByteCount > ByteCount, zero out OutputBuffer between ByteCount and RequestedByteCount.

Added a new Section 2.1.5.9.17, FSCTL_MARK_HANDLE.

2.1.5.9.17 FSCTL_MARK_HANDLE

The server provides:

● Open: An Open of a DataFile.

● InputBufferSize: The byte count of the InputBuffer.

● InputBuffer: A buffer of type MARK_HANDLE_INFO as defined in [MS-FSCC] section 2.3.31.

Upon completion, the object store MUST return:

● Status: An NTSTATUS code that specifies the result.

Support for this operation is optional. If the object store does not implement this functionality, the operation MUST be failed with STATUS_INVALID_DEVICE_REQUEST. <93>

Pseudocode for the operation is as follows:

● If InputBufferSize is less than the size of the MARK_HANDLE_INFO structure, the operation MUST be failed with STATUS_BUFFER_TOO_SMALL.

● If Open.Stream.StreamType == DirectoryStream, the operation MUST be failed with STATUS_DIRECTORY_NOT_SUPPORTED.

● STATUS_INVALID_PARAMETER is returned if: |

Errata Published*	Description
	● InputBuffer.HandleInfo contains any flag other than one and only one of either MARK_HANDLE_READ_COPY or MARK_HANDLE_NOT_READ_COPY. ● Open.Mode.FILE_NO_INTERMEDIATE_BUFFERING was not specified at open time, meaning the file was opened for cached IO operations. ● If InputBuffer.CopyNumber > (Open.File.Volume.NumberOfDataCopies – 1). ● If Open.Stream.StreamType != DataStream. ● If InputBuffer.HandleInfo has MARK_HANDLE_READ_COPY set: ● If Open.File.Volume.NumberOfDataCopies < 2, the operation MUST be failed with STATUS_NOT_REDUNDANT_STORAGE. ● If Open.Stream.IsCompressed is TRUE, the operation MUST be failed with STATUS_COMPRESSED_FILE_NOT_SUPPORTED. ● Set Open.ReadCopyNumber = InputBuffer.CopyNumber. ● Else If InputBuffer.HandleInfo has MARK_HANDLE_NOT_READ_COPY set: ● Set Open.ReadCopyNumber = 0xffffffff. ● EndIf Upon successful completion of the operation, the object store MUST return: ● Status set to STATUS_SUCCESS. In Section 6, Appendix A: Product Behavior, added product behavior note to support new section 2.1.5.9.17. <93> Section 2.1.5.9.17: This operation is only supported on the NTFS and ReFS file systems. This feature is supported in Windows Server 2019 and later.
2020/04/27	In Section 2.1.5.1, Server Requests an Open of a File, the following was changed from: ● If Link.File.FileType is not DirectoryFile, the operation MUST be failed with STATUS_NOT_A_DIRECTORY. Changed to: ● If Link.File.FileType is not DirectoryFile, the operation MUST be failed with STATUS_OBJECT_PATH_NOT_FOUND.
2020/04/27	In Section 2.1.4.12, Algorithm to Check for an Oplock Break, the following has been added: ● OPERATION_MASK – a constant that MUST contain the following value: ● (LEVEL_ONE_OPLOCK|LEVEL_TWO_OPLOCK|BATCH_OPLOCK|READ_CACHING|WRITE_CACHING|HANDLE_CACHING) The following was changed from: ● If OpParams.DesiredAccess contains no flags other than FILE_READ_ATTRIBUTES, FILE_WRITE_ATTRIBUTES, or SYNCHRONIZE, the algorithm returns at this point. Changed to: ● If (((OpParams.DesiredAccess contains no flags other than FILE_READ_ATTRIBUTES, FILE_WRITE_ATTRIBUTES, READ_CONTROL, or SYNCHRONIZE) and (Oplock.State anded with

Errata Published*	Description
	OPERATION_MASK) contains no flags other than READ_CACHING, WRITE_CACHING, or HANDLE_CACHING)) or ((OpParams.DesiredAccess contains no flags other than FILE_READ_ATTRIBUTES, FILE_WRITE_ATTRIBUTES or SYNCHRONIZE) and (Oplock.State anded with OPERATION_MASK) contains no flags other than LEVEL_TWO_OPLOCK, LEVEL_ONE_OPLOCK or BATCH_OPLOCK))), the algorithm returns at this point.
2020/04/27	In Section 2.1.5.9.21, FSCTL_QUERY_FILE_REGIONS, the following was added:

Support for this operation is optional. If the object store does not implement this functionality, this operation MUST be failed with STATUS_INVALID_DEVICE_REQUEST.<97>

<97> Section 2.1.5.9.21: This operation is only supported by the NTFS and ReFS file systems. |
| 2020/03/30 | Section 2.1.5.9.33   FSCTL_SET_SHORT_NAME_BEHAVIOR has been removed from the document.

Removed:

2.1.5.9.33   FSCTL_SET_SHORT_NAME_BEHAVIOR

This control code is reserved for the WinPE<118> environment; the object store MUST return STATUS _INVALID_DEVICE_REQUEST.

<118>WinPE stands for the windows Preinstallation Environment. For more information please see [MSFT-WinPE]. |

*Date format: YYYY/MM/DD