

[MS-FSCC]: File System Control Codes

This topic lists the Errata found in the MS-FSCC document since it was last published. Since this topic is updated frequently, we recommend that you subscribe to these RSS or Atom feeds to receive update notifications.



Errata are subject to the same terms as the Open Specifications documentation referenced.

Errata below are for Protocol Document Version [V39.0 – 2015/10/16](#).

Errata Published *	Description
2016/03/21	<p>In Section 2.3.40.1, FILE_REGION_INFO, and Section 2.3.79 STORAGE_OFFLOAD_TOKEN, corrected two field names – DesiredUsage and TokenId.</p> <p>In Section 2.3.40.1, FILE_REGION_INFO, changed from: Usage (4 bytes): A 32-bit unsigned integer that indicates the usage for the given region of the file. The valid values are defined in section 2.3.39.</p> <p>Changed to: DesiredUsage (4 bytes): A 32-bit unsigned integer that indicates the usage for the given region of the file. The valid values are defined in section 2.3.39.</p> <p>Also, corrected the field name in the bit table.</p> <p>In Section 2.3.79 STORAGE_OFFLOAD_TOKEN, changed from: The TokenType and TokenIdLength fields of STORAGE_OFFLOAD_TOKEN structure MUST be sent in big-endian format. The TokenID field is a stream of bytes and has no endian property.</p> <p>Changed to: The TokenType and TokenIdLength fields of STORAGE_OFFLOAD_TOKEN structure MUST be sent in big-endian format. The TokenId field is a stream of bytes and has no endian property.</p>
2015/11/23	<p>In various sections, changed the normative language.</p> <p>In Section 1.7, Vendor-Extensible Fields, "should" has been changed to "MUST" in the first paragraph.</p> <p>Changed from: File system control codes that are used to set reparse point data specify a ReparseTag field value that identifies the file system filter that understands the application-specific reparse point data format. A vendor developing an application protocol that sets reparse point data should request a unique reparse tag for that application from Microsoft by following the instructions described in [WHDC-RPTR]. For more information about reparse points, see [REPARSE].</p> <p>Changed to: File system control codes that are used to set reparse point data specify a ReparseTag field value that identifies the file system filter that understands the application-specific reparse point data format. A vendor developing an application protocol that sets reparse point data MUST</p>

Errata Published *	Description
	<p>request a unique reparse tag for that application from Microsoft by following the instructions described in [WHDC-RPTR]. For more information about reparse points, see [REPARSE].</p> <p>In Section 2.1.2.1, Reparse Tags, "should" has been changed to "SHOULD" in the third paragraph.</p> <p>Changed from: The following reparse tags, with the exception of IO_REPARSE_TAG_SYMLINK, are processed on the server and are not processed by a client after transmission over the wire. Clients should treat associated reparse data as opaque data.<2></p> <p>Changed to: The following reparse tags, with the exception of IO_REPARSE_TAG_SYMLINK, are processed on the server and are not processed by a client after transmission over the wire. Clients SHOULD treat associated reparse data as opaque data.<2></p> <p>In Section 2.1.7, FILE_NAME_INFORMATION, "should not" has been changed to "MUST NOT" in the description of FileName.</p> <p>Changed from: FileName (variable): A sequence of Unicode characters containing a pathname (section 2.1.5). The meaning of the pathname depends on the operation. The name string is not null-terminated. There are scenarios where one or more padding characters may be at the end of the string due to buffer alignment requirements, but their presence and their values should not be relied upon. When working with this field, use FileNameLength to determine the length of the file name rather than assuming the presence of a trailing null delimiter.</p> <p>Changed to: FileName (variable): A sequence of Unicode characters containing a pathname (section 2.1.5). The meaning of the pathname depends on the operation. The name string is not null-terminated. There are scenarios where one or more padding characters may be at the end of the string due to buffer alignment requirements, but their presence and their values MUST NOT be relied upon. When working with this field, use FileNameLength to determine the length of the file name rather than assuming the presence of a trailing null delimiter.</p> <p>In Section 2.3.19, FSCTL_GET_OBJECT_ID Request, "should" has been changed to "SHOULD" in the second paragraph.</p> <p>Changed from: Object identifiers are 16-byte opaque values that are used to track files and directories, and they are generated by the server. File and directory object identifiers are invisible to most applications and should never be modified by applications.</p> <p>Changed to: Object identifiers are 16-byte opaque values that are used to track files and directories, and they are generated by the server. File and directory object identifiers are invisible to most applications and SHOULD never be modified by applications.</p>
2015/11/23	<p>In Section 2.4.42, FileNotifyInformation, added two missing values for the Action field.</p> <p>Changed from:</p>

Errata Published *	Description																
	<p>Action (4 bytes): The changes that occurred on the file. This field MUST contain one of the following values.</p> <table border="1" data-bbox="391 394 1414 680"> <thead> <tr> <th>Values</th> <th>Meaning</th> </tr> </thead> <tbody> <tr> <td>...</td> <td>...</td> </tr> <tr> <td>FILE_ACTION_REMOVED_BY_DELETE 0x00000009</td> <td>An object ID was removed because the file the object ID referred to was deleted. This notification is only sent when the directory being monitored is the special directory "\$Extend\ObjId:\$O:\$INDEX_ALLOCATION".<125></td> </tr> </tbody> </table> <p>Changed to:</p> <p>Action (4 bytes): The changes that occurred on the file. This field MUST contain one of the following values.</p> <table border="1" data-bbox="391 848 1414 1659"> <thead> <tr> <th>Values</th> <th>Meaning</th> </tr> </thead> <tbody> <tr> <td>...</td> <td>...</td> </tr> <tr> <td>FILE_ACTION_REMOVED_BY_DELETE 0x00000009</td> <td>An object ID was removed because the file the object ID referred to was deleted. This notification is only sent when the directory being monitored is the special directory "\$Extend\ObjId:\$O:\$INDEX_ALLOCATION".<125></td> </tr> <tr> <td>FILE_ACTION_ID_NOT_TUNNELLED 0x0000000A</td> <td>An attempt to tunnel object ID information to a file being created or renamed failed because the object ID is in use by another file on the same volume. This notification is only sent when the directory being monitored is the special directory "\$Extend\ObjId:\$O:\$INDEX_ALLOCATION".<126></td> </tr> <tr> <td>FILE_ACTION_TUNNELLED_ID_COLLISION 0x0000000B</td> <td>An attempt to tunnel object ID information to a file being renamed failed because the file already has an object ID. This notification is only sent when the directory being monitored is the special directory "\$Extend\ObjId:\$O:\$INDEX_ALLOCATION".<127></td> </tr> </tbody> </table> <p><126> Only NTFS supports this special directory. <127> Only NTFS supports this special directory</p>	Values	Meaning	FILE_ACTION_REMOVED_BY_DELETE 0x00000009	An object ID was removed because the file the object ID referred to was deleted. This notification is only sent when the directory being monitored is the special directory "\$Extend\ObjId:\$O:\$INDEX_ALLOCATION".<125>	Values	Meaning	FILE_ACTION_REMOVED_BY_DELETE 0x00000009	An object ID was removed because the file the object ID referred to was deleted. This notification is only sent when the directory being monitored is the special directory "\$Extend\ObjId:\$O:\$INDEX_ALLOCATION".<125>	FILE_ACTION_ID_NOT_TUNNELLED 0x0000000A	An attempt to tunnel object ID information to a file being created or renamed failed because the object ID is in use by another file on the same volume. This notification is only sent when the directory being monitored is the special directory "\$Extend\ObjId:\$O:\$INDEX_ALLOCATION".<126>	FILE_ACTION_TUNNELLED_ID_COLLISION 0x0000000B	An attempt to tunnel object ID information to a file being renamed failed because the file already has an object ID. This notification is only sent when the directory being monitored is the special directory "\$Extend\ObjId:\$O:\$INDEX_ALLOCATION".<127>
Values	Meaning																
...	...																
FILE_ACTION_REMOVED_BY_DELETE 0x00000009	An object ID was removed because the file the object ID referred to was deleted. This notification is only sent when the directory being monitored is the special directory "\$Extend\ObjId:\$O:\$INDEX_ALLOCATION".<125>																
Values	Meaning																
...	...																
FILE_ACTION_REMOVED_BY_DELETE 0x00000009	An object ID was removed because the file the object ID referred to was deleted. This notification is only sent when the directory being monitored is the special directory "\$Extend\ObjId:\$O:\$INDEX_ALLOCATION".<125>																
FILE_ACTION_ID_NOT_TUNNELLED 0x0000000A	An attempt to tunnel object ID information to a file being created or renamed failed because the object ID is in use by another file on the same volume. This notification is only sent when the directory being monitored is the special directory "\$Extend\ObjId:\$O:\$INDEX_ALLOCATION".<126>																
FILE_ACTION_TUNNELLED_ID_COLLISION 0x0000000B	An attempt to tunnel object ID information to a file being renamed failed because the file already has an object ID. This notification is only sent when the directory being monitored is the special directory "\$Extend\ObjId:\$O:\$INDEX_ALLOCATION".<127>																
2015/10/26	In Section 2.1.2.6, Network File System (NFS) Reparse Data Buffer, the composition of the DataBuffer field in the descriptions of NFS_SPECFILE_CHR and NFS_SPECFILE_BLK has been corrected.																

Errata Published *	Description																		
	<p>Changed from:</p> <p>Type (8 bytes): A 64-bit unsigned integer value describing the type and format of the data stored in the DataBuffer field. The valid values for this field are:</p> <table border="1" data-bbox="436 527 1414 989"> <thead> <tr> <th>Value</th> <th>Meaning</th> </tr> </thead> <tbody> <tr> <td>...</td> <td>...</td> </tr> <tr> <td>NFS_SPECFILE_CHR 0x000000000524843</td> <td>Indicates that the DataBuffer field has two 16-bit integers that contain the major and minor numbers for the character special device created by the Network File System client.</td> </tr> <tr> <td>NFS_SPECFILE_BLK 0x0000000004b4c42</td> <td>Indicates that the DataBuffer field has two 16-bit integers that contain the major and minor numbers for the block special created by the Network File System client.</td> </tr> <tr> <td>...</td> <td>...</td> </tr> </tbody> </table> <p>...</p> <p>DataBuffer (variable): A variable buffer that has the following formats depending upon the Type field defined earlier.</p> <ul style="list-style-type: none"> ▪ NFS_SPECFILE_CHR and NFS_SPECFILE_BLK: The DataBuffer field contains two 16-bit integers that represent major and minor device numbers. <p>Changed to:</p> <p>Type (8 bytes): A 64-bit unsigned integer value describing the type and format of the data stored in the DataBuffer field. The valid values for this field are:</p> <table border="1" data-bbox="436 1444 1414 1854"> <thead> <tr> <th>Value</th> <th>Meaning</th> </tr> </thead> <tbody> <tr> <td>...</td> <td>...</td> </tr> <tr> <td>NFS_SPECFILE_CHR 0x000000000524843</td> <td>Indicates that the DataBuffer field has two 32-bit integers that contain the major and minor device numbers for the character special device created by the Network File System client.</td> </tr> <tr> <td>NFS_SPECFILE_BLK 0x0000000004b4c42</td> <td>Indicates that the DataBuffer field has two 32-bit integers that contain the major and minor device numbers for the character special device created by the Network File System client.</td> </tr> </tbody> </table>	Value	Meaning	NFS_SPECFILE_CHR 0x000000000524843	Indicates that the DataBuffer field has two 16-bit integers that contain the major and minor numbers for the character special device created by the Network File System client.	NFS_SPECFILE_BLK 0x0000000004b4c42	Indicates that the DataBuffer field has two 16-bit integers that contain the major and minor numbers for the block special created by the Network File System client.	Value	Meaning	NFS_SPECFILE_CHR 0x000000000524843	Indicates that the DataBuffer field has two 32-bit integers that contain the major and minor device numbers for the character special device created by the Network File System client.	NFS_SPECFILE_BLK 0x0000000004b4c42	Indicates that the DataBuffer field has two 32-bit integers that contain the major and minor device numbers for the character special device created by the Network File System client.
Value	Meaning																		
...	...																		
NFS_SPECFILE_CHR 0x000000000524843	Indicates that the DataBuffer field has two 16-bit integers that contain the major and minor numbers for the character special device created by the Network File System client.																		
NFS_SPECFILE_BLK 0x0000000004b4c42	Indicates that the DataBuffer field has two 16-bit integers that contain the major and minor numbers for the block special created by the Network File System client.																		
...	...																		
Value	Meaning																		
...	...																		
NFS_SPECFILE_CHR 0x000000000524843	Indicates that the DataBuffer field has two 32-bit integers that contain the major and minor device numbers for the character special device created by the Network File System client.																		
NFS_SPECFILE_BLK 0x0000000004b4c42	Indicates that the DataBuffer field has two 32-bit integers that contain the major and minor device numbers for the character special device created by the Network File System client.																		

Errata Published *	Description
	<div data-bbox="436 296 1414 348" style="border: 1px solid black; padding: 2px; margin-bottom: 5px;"> <div style="display: flex; justify-content: space-between;"> </div> </div> <p>...</p> <p>DataBuffer (variable): A variable buffer that has the following formats depending upon the Type field defined earlier.</p> <ul style="list-style-type: none"> ▪ NFS_SPECFILE_CHR and NFS_SPECFILE_BLK: The DataBuffer field contains two 32-bit integers that represent major and minor device numbers.

*Date format: YYYY/MM/DD