

[MS-BKUP]: Microsoft NT Backup File Structure

This topic lists the Errata found in the MS-BKUP 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 [V9.0 – 2018/09/12](#).

Errata Published*	Description
2020/04/27	<p>In Section 2.13.1, Creating an NT Backup File, a new paragraph was added to the end of the section.</p> <p>Added:</p> <p>If F has any ghosted extents, the NT backup file MUST generate one GHOSTED_EXTENT backup stream structure. During restore the GHOSTED_EXTENT backup stream structure is presented to the filesystem to recreate the file ghosted extent state.</p> <p>The following new sections were added:</p> <p>2.12 FileSystem Ghosted Extents Functionality</p> <p>Hierarchical Storage Management (HSM) solutions on top of a filesystem remove cold data and move it to the next storage tier. This movement creates sparse holes in file system data, and HSM solutions have to maintain mappings between those sparse holes and location of the data in the new tier. An implementation of the Ghosted extents feature helps this process by maintaining some token in the sparse holes, on behalf of the HSM solution. This obviates the need for the HSM solution to maintain its own mappings. When a file with such tokens, hereby referred to as ghosted extents, are backed up, the backup process should store the tokens and their locations in the backup stream state. On restore those tokens should be reinserted in the data stream in exactly the same locations to recreate the original file state. The method to query the tokens, serialize the tokens, and restore the tokens is file system implementation specific.</p> <p>2.12.1 Ghosted Extents Stream Structure</p> <p>A ghosted extent stream structure represents ghosted extents in the DATA backup stream. Ghosted extents are a kind of sparse extents, which store a GUID representing the owner of the extent and some variable-sized metadata. The structure of the data portion of this backup stream for a specific implementation is as follows:</p> <pre> 0 1 2 3 4 5 6 7 8 9 1 0 1 2 3 4 5 6 7 8 9 2 0 1 2 3 4 5 6 7 8 9 3 0 1 Count </pre>

Errata Published*	Description
	<p>TotalCount</p> <p>Data (variable)</p> <p>...</p> <p>Count (4 bytes): The number of extents in the Data portion.</p> <p>TotalCount (4 bytes): The total number of ghosted extents in the stream.</p> <p>Data (Variable): The data portion of the above structure contains a variable number of extents. The number of extents is given by Count. The structure of each Extent is described below:</p> <pre> 0 1 2 3 4 5 6 7 8 9 1 0 1 2 3 4 5 6 7 8 9 2 0 1 2 3 4 5 6 7 8 9 3 0 1 </pre> <p>Offset</p> <p>...</p> <p>Length</p> <p>...</p> <p>Guid</p> <p>...</p> <p>...</p> <p>...</p> <p>NextOffset</p> <p>Size</p> <p>Data (variable)</p> <p>...</p> <p>Offset (8 bytes): The logical byte offset in the DATA backup stream where the ghosted extent starts.</p>

Errata Published*	Description
	<p>Length (8 bytes): The logical length of the ghosted extent.</p> <p>GUID (16 bytes): The GUID identifier of the owner for the ghosted extent.</p> <p>NextOffset (4 bytes): Offset to the next Extent structure.</p> <p>Size (4 bytes): Size of the metadata of the ghosted extent.</p> <p>Data (variable): Metadata of the ghosted extent.</p>

*Date format: YYYY/MM/DD