**Understanding the archive bit and backup methods**

Whenever a file is created or changed, the operating system activates the Archive Bit or modified bit. Unless you select to use backup methods that depend on a date and time stamp, ECM Library uses the archive bit to determine whether a file has been backed up, which is an important element of your backup strategy. It is dangerous if other archive methods, processes or tools access an identified directory.

Selecting the following backup methods can affect the archive bit:

* Full - Back up files - Using archive bit (reset archive bit)
* Differential - Back up changed files since last full - Using archive bit (does not reset archive bit)
* Incremental - Back up changed files since last full or incremental - Using archive bit (reset archive bit)

Whenever a file has been backed up using either the Full - Back up files - Using archive bit (reset archive bit) or Incremental - Changed Files - Reset Archive Bit backup method, Backup Exec turns the archive bit off, indicating to the system that the file has been backed up. If the file is changed again prior to the next full or incremental backup, the bit is turned on again, and Backup Exec will back up the file in the next full or incremental backup. Backups using the Differential - Changed Files backup method include only files that were created or modified since the last full backup. When this type of differential backup is performed, the archive bit is left intact.

Consider the following backup strategy scenario:

Fred wants to implement a backup strategy for the office fileserver. Fred knows that all backup strategies begin with a full backup (backup of an entire device using the full backup method), so he creates a Selection List for his server and submits the job to run at the end of the day on Friday.

Since most files on the server, such as operating system files and application files, seldom change, Fred decides that he can save time and media by incorporating incremental or differential backups in his media rotation scheme. Fred opts to use incremental backups, so he schedules the script to run at the end of the day, Monday through Thursday, with the incremental backup method.

Here’s what happens: Fred’s Friday tape contains all of the data on the fileserver and Backup Exec changes all of the files’ statuses to “BACKED UP”. At the end of the day on Monday, the incremental job runs and only the files that were created or changed (had the archive bit re-set by the operating system) are backed up. When the incremental job completes, Backup Exec will turn the archive bit off, showing that the files have been backed up. On Tuesday through Thursday, the same events happen.

If Fred’s fileserver crashed on Thursday morning, after he got it running, he would restore each backup in the order in which it was created (for example, Friday, Monday, Tuesday, and so forth).

If Fred had decided to perform differential backups on Monday through Thursday, he would have only needed Friday’s and Wednesday’s tapes: Friday’s tape because it included all of the data, and Wednesday’s tape because it included every file that had been created or changed since Friday’s backup.