

Recursively delete zero-byte files (Windows)

A common problem with data migration at the file level is that you can end up with users complaining of zero-length files.

The problems come when you re-run the copy in the hope that the zero-byte files in the destination folder structure are updated with healthy ones from the source. Since the zero-byte files will have a newer time stamp than the source data, they become impossible to “fix” without potentially affecting surrounding files.

The best solution I’ve found is to identify and delete any files that are zero-bytes in length before re-running another copy.

To recursively delete zero-byte files in your folder structure, use the following command

```
for /r %F in (*) do if %~zF==0 del "%F"
```

If you wish to include the command in a batch file, you’ll need to double up on the % characters,

```
for /r %%F in (*) do if %%~zF==0 del "%%F"
```

This will remove the zero-byte files, allowing you to subsequently re-copy the now missing files from the source

```
robocopy.exe "%source%" "%dest%" *.* /S /X0 /FFT /MT:10
```

Note: to set your source and destination paths in a batch file, use the following syntax

```
SET "source=R:\MOVED\Mydata"  
SET "dest=T:\Mydata"
```

To simply find zero-byte files and append their full pathnames to a logfile, use

```
for /r %F in (*) do if %~zF==0 echo "%F" >
e:\logs\zerobytefiles.log
```