

Exiv2 - Bug #527

Thumbnail extract fails when location ends in directory separator

28 Aug 2007 17:18 - Jeff Woehler

Status:	Closed	Start date:	
Priority:	Normal	Due date:	
Assignee:	Robin Mills	% Done:	100%
Category:	not-a-bug	Estimated time:	0.00 hour
Target version:	0.26		
Description			
When the -l switch is used and a directory is specified that ends in the directory separator "\" the extraction fails with error code 22. It appears that this sequence of parameters causes the thumbnail filename to be incorrectly generated.			
Supplying a directory name without the ending "\" functions properly.			
Additional information:			
C:\Documents and Settings\jeff.WOEHLER\My Documents\utilities\exiv2>exiv2 -et DS C00498.JPG -l "C:\Documents and Settings\jeff.WOEHLER\Local Settings\Temp"			
C:\Documents and Settings\jeff.WOEHLER\My Documents\utilities\exiv2>exiv2 -et DS C00498.JPG -l "C:\Documents and Settings\jeff.WOEHLER\Local Settings\Temp"			
Exiv2 exception in extract action for file DSC00498.JPG:			
C:\Documents and Settings\jeff.WOEHLER\Local Settings\Temp\DSC00498-thumb.jpg: Failed to open file (wb): Invalid argument (22)			

History

#1 - 28 Aug 2007 19:00 - Jeff Woehler

This issue also only occurs with quoted directories. Unquoted locations function properly. For example:

```
C:\Documents and Settings\jeff.WOEHLER\My Documents\utilities\exiv2>exiv2 -et DS C00498.JPG -l C:\Temp\
```

#2 - 22 Aug 2015 15:33 - Robin Mills

- Description updated
- Category changed from jpeg parser to not-a-bug
- Status changed from New to Closed
- Assignee set to Robin Mills
- Target version set to 0.26

This is an artefact of cmd.exe's command to argv/argc parsing and has nothing to do with Exiv2(.exe).

I have a very useful little program "args.cpp" for debugging shell arguments. (args is a command on the Apollo Domain OS).

```
#include <stdio.h>
int main(int argc, char* argv[])
{
    int i = 1 ;
    while ( i < argc ) {
        printf("%-2d: %s\n",i,argv[i]) ;
        i++;
    }
    return 0 ;
}
```

args.exe enumerates and echoes arguments passed to it. Using this on your command line reveals:

```
C:\>args exiv2 -et DS C00498.JPG -l "C:\Documents and Settings\jeff.WOEHLER\Local Settings\Temp\"
1: exiv2
2: -et
3: DS
```

```
4: C00498.JPG
5: -l
6: C:\Documents and Settings\jeff.WOEHLER\Local Settings\Temp"
```

```
C:\>
```

You can see that argument 6 has the " appended. cmd.exe has passed the token \" as literal ". The correct syntax is to double the \ (i.e. escape the escape. The token \\ is passed as \ and the " closes the quoted string.

```
C:\>args exiv2 -et DS C00498.JPG -l "C:\Documents and Settings\jeff.WOEHLER\Local Settings\Temp\\" another-arg
1: exiv2
2: -et
3: DS
4: C00498.JPG
5: -l
6: C:\Documents and Settings\jeff.WOEHLER\Local Settings\Temp\
7: another-arg
```

Horrible? You betcha.

#3 - 22 Aug 2015 16:17 - Robin Mills

- % Done changed from 0 to 100

#4 - 23 Aug 2015 01:19 - Jeff Woehler

Thanks for the update Robin.