
Subject: File/Folder deleting, Recycle Bin and Read only files

Posted by [koldo](#) on Tue, 05 Jan 2010 09:02:21 GMT

[View Forum Message](#) <> [Reply to Message](#)

Hello all

Presently we have this basic functions to delete files and folders:

```
bool FileDelete(const char *filename)
```

```
bool DirectoryDelete(const char *dirname)
```

They delete forever a file/folder if it is not read-only (and of course if it is available).

It would be great if they would manage as transparently as possible the Recycle Bin and the Read-Only files.

Some possible options would be:

1.- To add to this functions new parameters. For example:

```
bool FileDelete(const char *filename, bool recycleBin = false, bool delReadOnly = false)
```

```
bool DirectoryDelete(const char *dirname, bool recycleBin = false, bool delReadOnly = false)
```

or

```
enum DeleteFlags {DEL_FOREVER = 0, DEL_RECYCLE = 1, NOT_DEL_READONLY = 0, DEL_READONLY = 2};
```

```
bool FileDelete(const char *filename, int flags = 0)
```

```
bool DirectoryDelete(const char *dirname, int flags = 0)
```

2.- To add functions to handle this:

```
void SetDeleteToRecycleBin(bool val = false);
```

```
void SetDeleteReadOnly(bool val = false);
```

or

```
enum DeleteFlags {DEL_FOREVER = 0, DEL_RECYCLE = 1, NOT_DEL_READONLY = 0, DEL_READONLY = 2};
```

```
void SetDeleteFlags(bool val = false);
```

What do you think about it ?

Best regards

Koldo

Subject: Re: File/Folder deleting, Recycle Bin and Read only files

Posted by [dolik.rce](#) on Tue, 05 Jan 2010 12:56:57 GMT

[View Forum Message](#) <> [Reply to Message](#)

koldo wrote on Tue, 05 January 2010 10:02It would be great if they would manage as transparently as possible the Recycle Bin and the Read-Only files.

Hi Koldo,

It's interesting idea, but it's definitely not multiplatform. On Linux filesystems read-only usually means you simply don't have the permission to delete the file. Also recycle bin might not be available and even when it is then its behaviour and implementation differs in each desktop environment.

On windows, it might be useful. If I remember correctly, winXP show a dialog if you really wan't to delete read-only or system file, not sure about vista or win7 (but I heard their permissions system are better now)

Thinking about it, another way to achieve this would be to implement function that manages file attributes. (Or is there something like that in upp already?) Then you could just delete the read-only flag and delete it. For recycle bin I would suggest special function to. It feels "cleaner" to me to add

```
#ifdef PLATFORM_WIN32
```

```
SetAttributes(int Attr);
```

```
MoveToRecycleBin(const char* File);
```

```
#endif
```

instead of changing the existing functions in win32 specific way...

Best regards

Honza

Subject: Re: File/Folder deleting, Recycle Bin and Read only files

Posted by [koldo](#) on Tue, 05 Jan 2010 14:11:13 GMT

[View Forum Message](#) <> [Reply to Message](#)

Hello Honza

Thank you for reading the long post.

Some comments:

Quote:It's interesting idea, but it's definitely not multiplatform. On Linux filesystems read-only usually means you simply don't have the permission to delete the file.

If my user have the permission, it can change the file attributes using `chmod()` in Linux.

See `bool FileSetReadOnly(String fileName, bool readOnly)` implementation in Functions4U Bazaar package..

Quote:Also recycle bin might not be available

As far as I know Gnome and Kde desktops have Recycle Bin (or Trash Bin). At least in case of Gnome and Ubuntu derivatives, Trash Bin is located in a fixed folder.

See `bool FileToTrashBin(const char *path)` implementation in Functions4U Bazaar package.

Quote:and even when it is then its behaviour and implementation differs in each desktop environment.

No problem, Upp packages have some internal `#ifdef PLATFORM_POSIX` and many other so that the final user does not have to worry about OS/Desktop particular behaviours.

Quote:Thinking about it, another way to achieve this would be to implement function that manages file attributes. (Or is there something like that in upp already?) Then you could just delete the read-only flag and delete it. For recycle bin I would suggest special function to. It feels "cleaner" to me to add

```
#ifdef PLATFORM_WIN32
SetAttributes(int Attr);
MoveToRecycleBin(const char* File);
#endif
```

instead of changing the existing functions in win32 specific way...

These functions are already implemented in Functions4U Bazaar package as:

```
bool FileToTrashBin(const char *path);
int64 TrashBinGetCount();
bool TrashBinClear();
bool FileSetReadOnly(String fileName, bool readOnly);
```

They could be reimplemented, renamed and transfered to Core package... the problem is that derived functions would have to be changed like actualbool DeleteFolderDeep(const char *dir)...

I would not like to have a
bool FileToTrashBinDeep(const char *dir)

SetAttributesDeep(String folder, int Attr)

and different version handling Trash Bin and Read permissions for the rest of functions that use FileDelete or DirectoryDelete

Best regards
Koldo

Subject: Re: File/Folder deleting, Recycle Bin and Read only files

Posted by [mirek](#) on Wed, 06 Jan 2010 06:49:23 GMT

[View Forum Message](#) <> [Reply to Message](#)

koldo wrote on Tue, 05 January 2010 04:02Hello all

Presently we have this basic functions to delete files and folders:

```
bool FileDelete(const char *filename)
bool DirectoryDelete(const char *dirname)
```

They delete forever a file/folder if it is not read-only (and of course if it is available).

It would be great if they would manage as transparently as possible the Recycle Bin and the Read-Only files.

Some possible options would be:

1.- To add to this functions new parameters. For example:

```
bool FileDelete(const char *filename, bool recycleBin = false, bool delReadOnly = false)
bool DirectoryDelete(const char *dirname, bool recycleBin = false, bool delReadOnly = false)
or
enum DeleteFlags {DEL_FOREVER = 0, DEL_RECYCLE = 1, NOT_DEL_READONLY = 0,
DEL_READONLY = 2};
bool FileDelete(const char *filename, int flags = 0)
bool DirectoryDelete(const char *dirname, int flags = 0)
```

2.- To add functions to handle this:

```
void SetDeleteToRecycleBin(bool val = false);
void SetDeleteReadOnly(bool val = false);
or
enum DeleteFlags {DEL_FOREVER = 0, DEL_RECYCLE = 1, NOT_DEL_READONLY = 0,
DEL_READONLY = 2};
void SetDeleteFlags(bool val = false);
```

What do you think about it ?

Best regards
Koldo

I do not think doing it this way is a good idea.

Path.h functions are meant to be base level API encapsulation. At these levels, OS has no notion about recycle bin.

Recycle bin, OTOH, is a high level, basically OS shell function.

Mirek

Subject: Re: File/Folder deleting, Recycle Bin and Read only files
Posted by [koldo](#) on Wed, 06 Jan 2010 10:51:15 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hello Mirek

I understand you but, how did you implement this ?

Did you do a new set of FileDelete() or DeleteFolderDeep() functions between others including Trash Bin support or RW permissions ?.

Best regards

Koldo

Subject: Re: File/Folder deleting, Recycle Bin and Read only files

Posted by [mirek](#) on Wed, 06 Jan 2010 20:59:55 GMT

[View Forum Message](#) <> [Reply to Message](#)

koldo wrote on Wed, 06 January 2010 05:51Hello Mirek

I understand you but, how did you implement this ?

Did you do a new set of FileDelete() or DeleteFolderDeep() functions between others including Trash Bin support or RW permissions ?.

Best regards

Koldo

I would just stay with FileDeletToBin...

Subject: Re: File/Folder deleting, Recycle Bin and Read only files

Posted by [koldo](#) on Wed, 06 Jan 2010 22:10:57 GMT

[View Forum Message](#) <> [Reply to Message](#)

Quote:I would just stay with FileDeletToBin...

Well I will prepare extended versions with progress (Gate) support.

Best regards

Koldo
