
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
