
Subject: Added FSMon - FileSystem Monitor class
Posted by [mdelfede](#) on Sat, 12 May 2012 07:32:04 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi,

I dropped a small class, FSMon, along with its demo FSMonTest, which allows monitoring a folder and its sub-folders for file changes.

Usage is quite simple :

```
FSMon fsMon;
```

```
// this will add a monitored path  
fsMon.Add(APathToMonitor);
```

```
// this will remove it  
fsMon.Remove(AMonitoredPath);
```

Catch of events can be done by polling

```
while(true)  
{  
    if(fsMon.HasChanges())  
    {  
        Vector<FSMon::Info> info = fsMon.GetChanged();  
        for(int i = 0; i < info.GetCount(); i++)  
            DoSomething(info[i].path, info[i].newPath, info[i].flags);  
    }  
    else  
        Sleep(100);  
}
```

or event driven with a callback :

```
void eventHandler(void)  
{  
    Vector<FSMon::Info> info = fsMon.GetChanged();  
    .....  
}
```

```
fsMon.EventHandler = THISBACK(eventHandler);
```

Beware, for the event-driven the behaviour is different if runnini inside a GUI app or a NON-GUI app :
for GUI apps, the call is done through a PostCallback, so runs in main thread
for NON GUI apps, the call is direct, so runs in FSMon monitoring thread, which is not good for lengthy operations.

So, for NON-GUI apps it's better to use a separate thread and polling.

FSMon can catch file/folders creation, deletion and renaming, along with attribute changes. See flags in FSMon.h and the sample FSMonDemo for details.

Ciao

Max

Subject: Re: Added FSMon - FileSystem Monitor class
Posted by [mdelfede](#) on Sat, 12 May 2012 12:33:32 GMT
[View Forum Message](#) <> [Reply to Message](#)

Added another fancy, more useful example : FSMonSync package.

It realizes a trivial path synchronization mechanics; the demo creates a couple of folders in user home folder, which are

```
/home/user/FSMonTest_A  
/home/user/FSMonTest_B
```

on Linux, and

```
c:\windows\profiles\users\user\FSMonTest_A  
c:\windows\profiles\users\user\FSMonTest_B
```

or

```
c:\documents and settings\user\FSMonTest_A  
c:\documents and settings\user\FSMonTest_A
```

on Windows (well, I guess, it depends on win version... anyways the log windows will show you both).

Whatever you do on first folder (FSMonTest_A) gets duplicated on second one (FSMonTest_B), even in subfolders, in real-time.

The app is quite trivial, don't do any security checks nor optimizations, but works and allows, for example, to have a real-time backup of your data on an external disk.

The very base for an "UCloud" app ?

Ciao

Max

Subject: Re: Added FSMon - FileSystem Monitor class
Posted by [koldo](#) on Tue, 15 May 2012 17:18:56 GMT
[View Forum Message](#) <> [Reply to Message](#)

Very interesting Massimo. Thank you .

Subject: Re: Added FSMon - FileSystem Monitor class
Posted by [Mindtraveller](#) on Wed, 05 Jun 2013 12:55:06 GMT
[View Forum Message](#) <> [Reply to Message](#)

Thank you for this package!

A little addition: in FSMonTest/main.cpp at line 15, you should add directory in system charset (not in UTF-8).

```
fsmMon.Add(ToSystemCharset(path));
```

And so it works for files and directories with localized names.