

Folder Comparison and Synchronization

Usage:

1. Choose left and right directories.

Drag & Drop	Drag & Drop		
C:\Source 🔻 Browse	D:\Target	•	Browse

2. Compare them.



3. Select synchronization settings.



4. Press **Synchronize** to begin synchronization.



Main Dialog Overview

😂 FreeFileSync - Folder Compar	ison and Synchroni	zation			- • ×
<u>P</u> rogram <u>T</u> ools <u>H</u> elp		(3)			
Compare File time and siz) 🍸	4\$		nchronize 5
Overview ×	E Drag & dr	op		rag & drop	-
Name Size	7 🕂 C:\Source):\Target	8 Browse
	Name	Size	R 2	Name	Size ^
6	1				
	2	_			
	4				=
Configuration ×	5		9		
	6 7	_			
	8				
<last session=""></last>	9				-
Backup FreeFileSync.ffs_batch	10				
10		~	1 1		
		(11)	(1	2 * * *	0 bytes 0 0 0
	1 F	$\overline{}$			0 bytes 0 0 0

- 1. Start comparison
- 2. Change comparison settings
- 3. Include/exclude specific files
- 4. Change synchronization settings
- 5. Start synchronization
- 6. Tree overview panel
- 7. Add additional folder pairs
- 8. Select left and right folders
- 9. Synchronization preview
- 10. Save/load configuration
- 11. Select categories to show on grid
- 12. Synchronization statistics

Command Line Usage

FreeFileSync enables additional synchronization scenarios via a command line interface. To get a syntax overview, open the console, go to the directory where FreeFileSync is installed and type:

FreeFileSync -h

1. Run a FreeFileSync batch job

In order to start synchronization in batch mode, supply the path of a ffs_batch configuration file as the first argument for FreeFileSync.exe:

FreeFileSync "D:\Backup
Projects.ffs_batch"

After synchronization one of the following status codes is returned:

Return Codes

- 0 Synchronization completed successfully
- 1 Synchronization completed with warnings
- 2 Synchronization completed with errors
- 3 Synchronization was aborted

You can evaluate these codes from a script (e.g. a cmd or bat file on Windows) and check if synchronization completed successfully:

```
"C:\Program
Files\FreeFileSync\FreeFileSync.exe"
"D:\Backup Projects.ffs_batch"
if errorlevel 1 (
    ::if return code is 1 or greater,
    something went wrong, add special
```

```
treatment here
  echo Errors occurred during
synchronization...
  pause
)
```

Instead of displaying "An error occurred!" you can also send an email notification (using a third party tool).



2. Start a FreeFileSync GUI configuration

If you pass a ffs_gui file, FreeFileSync will start in GUI mode and immediately start comparison (but only if all directories exist):

FreeFileSync "D:\Manual Backup.ffs_gui"

3. Customize an existing configuration

You can replace the directories of a given ffs_gui or ffs_batch configuration file by using the -leftdir and -rightdir parameters:

```
FreeFileSync "D:\Manual Backup.ffs_gui" -
leftdir C:\NewSource -rightdir
D:\NewTarget
```

4. Merge multiple configurations

When more than one configuration file is provided, FreeFileSync will merge everything into a single configuration with multiple folder pairs and start in GUI mode:

```
FreeFileSync "D:\Manual Backup.ffs_gui"
"D:\Backup Projects.ffs_batch"
```

Comparison Settings

Select a	variant:
\bigcirc	File time and size
010110 110011 101000	File content
òymboli Exclude	
Exclude Direct	

Symbolic Link Handling

FreeFileSync offers three options to configure handling of symbolic links (also called symlinks or soft links):

- 1. **Exclude:** Skip symbolic links while scanning directories.
- 2. **Direct:** Evaluate the symbolic link object directly. Symbolic links will be shown as a separate entity on grid. Links pointing to directories are not traversed and the link object is copied directly during synchronization.
- 3. **Follow:** Treat symbolic links like the object they are pointing to. Links pointing to directories are traversed like ordinary directories and the target of each link is copied during synchronization.

Note

- Under Windows the symbolic link options apply to symbolic links, volume mount points and NTFS junction points.
- Copying symbolic links requires FreeFileSync to be

started with administrator rights.

I. Compare by File time and size

This variant considers two files with the same name as equal when both modification time **and** file size match. The following categories are distinguished:

i. file exists on one side only

- 1. left only
- 2. right only
- ii. file exists on both sides
 - i. different date
 - left newer
 - right newer
 - ii. same date
 - equal
 - conflict (same date, different size)

II. Compare by File content

Two files with the same name are marked as equal if and only if they have the same content. This option is more useful for consistency checks rather than backup operations since it is naturally slower. The file modification time is not taken into account at all.

i. file exists on one side only

- 1. left only
- 2. right only
- ii. file exists on both sides
 - 1. equal
 - 2. different content

Daylight Saving Time (Windows)

A common problem synchronization software has to handle are +-1 hour file time shifts after a Daylight Saving Time (DST) switch has occurred. This can be observed for example when a FAT-formatted volume is compared against an NTFS volume as is the case when synchronizing a local disk against a USB memory stick. Files that previously appeared to be in sync are now shown with an one hour modification time offset, although they have not been modified by the user or the operating system.

The reason for this behavior lies in the way NTFS and FAT drives store file times: NTFS stores time in UTC format, while FAT uses local time.

When times of these two different formats are compared, one format has to be converted into the other first. In either way Windows uses the current DST status as well as the current time zone for its calculations. Consequently the result of this comparison is dependent from current system settings and in particular file times that used to be the same can show up as different after a DST switch or when the time zone is changed.

For a detailed discussion about this issue refer to: http://www.codeproject.com/KB/datetime/dstbugs.aspx

Solution:

FreeFileSync automatically handles this problem by adding the missing time information. Each file on a FAT volume automatically gets additional meta data encoded in its creation date that enables a correct file time calculation. This not only solves all DST issues but also time shifts that occur due to travel between different time zones.

Note

• In order for FreeFileSync to start handling DST and timezone differences, an initial full synchronization

is required. Subsequent syncs will then never show a time difference again for unchanged files.

• If a FAT volume is scanned the first time by FreeFileSync this will take longer than usual since additional meta data is written for each file.

Exclude Items

Filter		
Y	Select filter rules to exclude their corresponding folder	certain files from synchronization. Enter file paths relative to pair.
	Include:	Time span:
Ø	*	
	Exclude:	Show examples File size:
8	\file.txt \SubFolder\ *\thumbs.db \SubFolder*.tmp *temp* *.tmp; *.doc; *.bak	Minimum: 0 (None) Maximum: 0 (None) (None)
	<u>C</u> lear	OK Cancel

Files and directories are only considered for synchronization if they pass all filter rules. They have to match **at least one** entry in the include list and **none** of the entries in the exclude list as presented in the filter configuration dialog:

- Each list item must be a file or directory path **relative** to synchronization base directories.
- Multiple items must be separated by ; or a new line.
- Wild cards * and ? may be used: * means zero or more characters while ? represents exactly one character.

Example: Exclude items for mirror-sync from C:\Source to D:\Target

Description	Exclude
Single file C:\Source\file.txt	\file.txt
Single folder C:\Source\SubFolder	\SubFolder\
All files (and folders) named thumbs.db	*∖thumbs.db
All * . tmp files located in SubFolder only	\SubFolder*.tmp
Files and folders containing temp somewhere in their path	*temp*
Multiple entries separated by semicolon	*.tmp; *.doc; *.bak
Exclude all files and folders located in subdirectories of base directories	**
Exclude only *.txt files located in subdirectories of base directories	**.txt

Note

- For simple exclusions just right-click and exclude one item or a list of items directly on main grid via context menu.
- A filter phrase is compared against both file and directory paths. If you want to consider directories only, you can give a hint by appending a path separator (\).

Expert Settings

FreeFileSync has a number of special purpose settings that can only be accessed directly via the global configuration file GlobalSettings.xml. Note that this file is read once when FreeFileSync starts and saved when it closes. Therefore do only apply changes while FreeFileSync is not running.

To locate this file on Windows enter %appdata%\FreeFileSync in the Windows Explorer address bar or go to the FreeFileSync installation folder if you are using the portable installation. On Linux you can find the file in ~/.FreeFileSync for the Launchpad release and in the installation folder for the portable version. On OS X go to ~/Library/Application Support/FreeFileSync.

```
<?xml version="1.0" encoding="UTF-8"?>
<FreeFileSync XmlType="GLOBAL">
<Shared>
<FileTimeTolerance Seconds="2"/>
<RunWithBackgroundPriority
Enabled="false"/>
<LockDirectoriesDuringSync
Enabled="true"/>
<VerifyCopiedFiles
Enabled="false"/>
<LastSyncsLogSizeMax
Bytes="100000"/>
```

FileTimeTolerance:

By default file modification times are allowed to have a 2 second difference while still being considered equal. This is required by FAT/FAT32 file systems which store file times with a 2 second precision only.

This setting can also be used to simulate a "compare by file size", ignoring last modification times: Set tolerance to -1 which will be

considered as an unlimited file time tolerance. Changed files will then be detected as a conflict (same date, different file size) and the synchronization direction for conflicts can be used accordingly.

RunWithBackgroundPriority:

While synchronization is running, other applications which are accessing the same data locations may experience a noticeable slowdown. Enable this setting to lower FreeFileSync's resource consumption at the cost of a significantly slower synchronization speed.

LockDirectoriesDuringSync:

In order to avoid race conditions of multiple FreeFileSync instances writing to the same folder at the same time, accesses are serialized by lock files (sync.ffs_lock). This allows to operate FreeFileSync with an arbitrary number of users in a network out of the box.

VerifyCopiedFiles:

If active, FreeFileSync will binary-compare source and target files after copying and report verification errors. Note that this may double file copy times and is no guarantee that data has not already been corrupted prior to copying and corruption is not hidden by deceptively reading valid data from various buffers in the application and hardware stack. Does the CopyFile function verify that the data reached its final

destination successfully?

LastSyncsLogSizeMax:

The progress logs of the most recent synchronizations (for both GUI and batch jobs) are collected automatically in the file LastSyncs.log. The maximum size of this log file can be set here.

External Applications

When you double-click on one of the rows on the main dialog, FreeFileSync opens the operating system's file browser by default. On Windows it calls explorer /select, "%item_path%", on Linux xdg-open "%item_folder%" and on OS X open -R "%item_path%".

You can customize this behavior and integrate other external applications into FreeFileSync: Navigate to **Menu** \rightarrow **Tools** \rightarrow **Global settings: Customize context menu** and add or replace a command. The first entry is executed when double-clicking a row on main grid or pressing ENTER while all other entries can be accessed via the context menu shown after a right mouse click. The following macros can be used:

%item_path%	- full file or folder name
%item_folder%	- folder part only
%item2_path%	- Counterpart of %item_path% on
the opposite grid	
%item2_folder%	 Counterpart of %item_folder%
on the opposite grid	

Examples:

- Start visual difference tool: "C:\Program Files (x86)\WinMerge\WinMergeU.exe" "%item_path%" "%item2_path%"
- Show file in Windows Explorer: explorer /select, "%item_path%"
- Open file in associated application: cmd /c start "" "%item_path%" or simply "%item_path%"
- Open console dialog:

cmd /k cd /D "%item_folder%"

Note

You need to protect macros with quotation marks if they can resolve to a file path with space characters.

Macros

All directory names may contain macros that are expanded during synchronization. Begin and end of each macro is marked by a % character. Besides special macros handling time and date, the **operating system's environment variables** can also be used.

Internal macros

%date% [YYYY-MM-DD]	e.g. 2012-12	2-22	format:
%time% [hhmmss]	e.g. 123044		format:
%timestamp% [YYYY-MM-DD h	•	2-22 123044	format:
%year%	e. g. 2012		
%month%	e. g. 12		
%day%	e. g. 22		
%hour%	e. g. 12		
%min%	e. g. 30		
%sec%	e. g. 44		
%weekday% %week%	e. g. Monday e. g. 28	(day of the we (calendar wee	,

Environment variables (Windows)

%AllUsersProfile% e.g.C:\ProgramData %AppData% e.g.C:\Users\ <username>\AppData\Roaming %ComputerName% e.g.Zenju-PC %LocalAppData% e.g.C:\Users\

<username>\AppData\Local %ProgramData% e.g.C:\ProgramData %ProgramFiles% e.q.C:\Program Files %ProgramFiles(x86)% e.g.C:\Program Files (x86) e.g.C:\Users\Public %Public% e.g.C:\Windows\Temp %Temp% %UserName% e. q. Zenju e.g.C:\Users\ %UserProfile% <username> %WinDir% e.g.C:\Windows

Special folder locations (Windows)

%csidl_Desktop% e.g.C:\Users\ <username>\Desktop %csidl Downloads% e.g.C:\Users\ <username>\Downloads %csidl Favorites% e.g.C:\Users\ <username>\Favorites %csidl_MyDocuments% e.g.C:\Users\ <username>\Documents %csidl_MyMusic% e.g.C:\Users\ <username>\Music %csidl_MyPictures% e.g.C:\Users\ <username>\Pictures %csidl MyVideos% e.g.C:\Users\ <username>\Videos e.q.C:\Users\ %csidl Nethood% <username>\AppData\Roaming\Microsoft\Windows\Net Shortcuts e.g.C:\Users\ %csidl Programs% <username>\AppData\Roaming\Microsoft\Windows\Sta Menu\Programs %csidl Quicklaunch% e.g.C:\Users\ <username>\AppData\Roaming\Microsoft\Internet

Explorer\Quick Launch %csidl Resources% e. q. C:\Windows\Resources e.q.C:\Users\ %csidl StartMenu% <username>\AppData\Roaming\Microsoft\Windows\Sta Menu %csidl_Startup% e.g.C:\Users\ <username>\AppData\Roaming\Microsoft\Windows\Sta Menu\Programs\StartUp %csidl_Templates% e. q. C:\Users\ <username>\AppData\Roaming\Microsoft\Windows\Ten Note: Most macros listed here also have a variant for public folders. E.g. csidl_MyMusic \rightarrow

csidl_PublicMusic

Hint: You can add a great amount of flexibility to a ffs_batch configuration file by creating new temporary environment variables in a bat or cmd file that are evaluated by FreeFileSync at runtime.

Example: The FreeFileSync batch file C:\SyncJob.ffs_batch contains macro %MyVar% instead of an absolute target folder and is invoked by a cmd file:

set MyVar=C:\Target
"C:\Program
files\FreeFileSync\FreeFileSync.exe"
C:\SyncJob.ffs_batch
::%MyVar% is resolved as C:\Target during
synchronization

Note

Temporary environment variables created with the set command are only valid if the synchronization is started by calling the FreeFileSync executable directly. Using start /wait creates a new program context without these temporal variables.

Schedule a Batch Job

- 1. Create a new batch job via FreeFileSync's main dialog: Menu \rightarrow Program \rightarrow Save as batch job...
- 2. If the batch job shall run without user interaction or as part of an unattended batch script, make sure that no popup dialog stops the progress:

Disable checkbox **Show progress dialog** to avoid blocking while showing the result after synchronization. Alternatively you can leave this option checked and select the *On completion* action **Close progress dialog** located in synchronization settings. FreeFileSync will then show a progress dialog but close it automatically when it is finished.

Note: Even if the progress dialog is not shown at the beginning, you can make it visible **during** synchronization by double-clicking the FreeFileSync notification area icon.

		Show progress dialog
Ignore	Pop-up Stop	On completion: Close progress dialog
Save log:		1
Save log.		Browse Limit: 100

3. In order to prevent error or warning popup messages from stopping progress, set **Handle errors** to either **Ignore** or **Stop**.

4. Setup your operating system's scheduler

A. Windows 7 Task Scheduler:

- Go to Start and run taskschd.msc.
- Create a new basic task and follow the wizard.
- Make Program/script point to the location of FreeFileSync.exe and insert the ffs_batch file into Add arguments.
- Use quotation marks to protect spaces in path names, e.g.
 "D:\Backup Projects.ffs_batch"

Create Basic Task Wizard		Y
5tart a Program		
Create a Basic Task Trigger Daily Action	Program/script: "C:\Program Files\FreeFileSync\FreeFileSync.exe"	Browse
Start a Program Finish	Add arguments (optional): Start in (optional):	"D:\Backup Projects.ffs_batch"
	<	Back Next > Cancel

Note

Beginning with Windows Vista the *Program/script* always needs point to an executable file like FreeFileSync.exe even if ffs_batch file association is set. If a ffs_batch file is entered instead the task will return with error code 0xC1, "%1 is not a valid Win32 application".

B. Windows XP Scheduled Tasks:

- Go to Start → Control Panel → Scheduled Tasks and select Add Scheduled Task.
- Follow the wizard and select FreeFileSync.exe as program to run.
- Fill the input field **Run:** < FreeFileSync installation

folder>\FreeFileSync.exe <job name>.ffs_batch

FreeFileSync ? 🔀
Task Schedule Settings
C:\WINDOWS\Tasks\FreeFileSync.job
Bun: FileSync\FreeFileSync.exe" "D:\Backup Projects.ffs_batch"
Browse
Start in: "C:\Program Files\FreeFileSync"
Comments:
R <u>u</u> n as: Zenju <u>S</u> et password
Run only if logged on Enabled (scheduled task runs at specified time)
OK Cancel Apply

C. Ubuntu Linux Gnome-schedule:

- Install Gnome-schedule, if necessary: sudo apt-get install gnome-schedule
- Go to System → Preferences → Scheduled tasks
- Enter the command: <FreeFileSync installation folder>/FreeFileSync <job name>.ffs_batch

& Create	a New Scheduled Task	
Descri Comm	ption: Test Task hand: ("/usr/bin/FreeFileSync" "/home/zenju/Desktop/SyncJob.ffs ☑ <u>N</u> o output (>/dev/null 2>&1)	_batch"
Time & Date		
Basic	Every hour	\$
O Advanced		
Minute:	0	<u>E</u> dit
Hour:	*	Edit
Day:	*	Edit
Month:	*	<u>E</u> dit
Weekday:	*	Edit
Preview		
At every	full hour	
	Add as template	-∯ <u>A</u> dd

Synchronize with FTP/WebDAV (Windows)

FreeFileSync does not support accessing FTP volumes directly. But this functionality can be integrated by mapping the FTP web space to a drive letter:

Example: Use the free utility NetDrive (<u>http://www.netdrive.net</u>)

- Add a **New Site** and specify site name, site URL, drive letter, account and password.
- Use the newly created drive as if it were a normal hard disk.

NoteMost FTP drives set a file's time stamp to the current time when synchronizing ignoring the source file's time and date. As a workaround you can do a <u>Compare by File Size</u>.

Synchronize with SFTP (Linux)

An SFTP share can be easily mapped onto a local folder for use with FreeFileSync:

Install: sudo apt-get install sshfs
Mount SFTP share: sshfs ssh-account@ssh-server:<path> mountpoint
Unmount: fusermount -u mountpoint

Variable Drive Letters

USB memory sticks or external hard disks often get different drive letters assigned when plugged into distinct computers. FreeFileSync offers two solutions to handle this problem:

Option 1: Specify a folder path by using the volume name:

Use [ZENJU-USB]\folder instead of G:\folder
where ZENJU-USB is the volume name of the USB
stick which is currently mounted in drive $G: \$.

Note			
It is not required to look up and ente	r tł	ne volum	e name
manually. Just select the correspond			
drop down menu.			
Drag & drop			
G:\folder	-	Browse	
G:\folder			
[ZENJU-USB]\folder			
	_		

Option 2: Use a relative directory name:

- Use \folder instead of G: \folder
- Save and copy synchronization settings to the USB stick: G:\Backup.ffs_gui
- Start FreeFileSync by double-clicking on G:\Backup.ffs_gui

The working directory is then automatically set to G:\ by the operating system so that the relative path \folder will be resolved as G:\folder during synchronization.

File Versioning

When you need to preserve files that have been deleted or overwritten it's often sufficient to select **Recycle bin** in synchronization settings. However this is only available for local drives and offers little control on how to store and how long to keep the files. FreeFileSync therefore has an additional option, **Versioning**.

1. Keep all versions of old files

In synchronization settings set deletion handling to **Versioning** and naming convention to **Time stamp**. FreeFileSync will move deleted files into the provided folder and add a time stamp to each file name. The structure of the synchronized folders is preserved so that old versions of a file can be conveniently accessed via a file browser.

Example: A file Folder\File.txt was updated three times and old versions were moved to folder C:\Revisions

C:\Revisions\Folder\File.txt 2012-12-12 111111.txt C:\Revisions\Folder\File.txt 2012-12-12 122222.txt C:\Revisions\Folder\File.txt 2012-12-12 133333.txt

2. Save only the most recent version

Set deletion handling to **Versioning** and naming convention to **Replace**. Deleted files will be moved to the specified folder without any decoration and will replace already existing older versions.

3. Save versions at certain intervals

With naming convention **Replace** it is possible to refine the granularity of versions to keep by adding <u>macros</u> to the versioning folder path. For example you can save deleted files on a per sync session basis by adding the **%timestamp%** macro:

Example: Using the dynamically generated folder name C:\Revisions\%timestamp%

C:\Revisions\2012-12-12 111111\Folder\File.txt C:\Revisions\2012-12-12 122222\Folder\File.txt C:\Revisions\2012-12-12 133333\Folder\File.txt

This allows for a simple manual undo by moving the deleted files from the last synchronization session back to their original folders. Other macros like **%date%** or **%weekday%** can be used to reduce the granularity to days and weeks.

Shadow Copy Service (Windows only)

FreeFileSync supports copying locked or shared files by creating a Volume Shadow Copy of the source drive. This feature can be configured via Menu \rightarrow Tools \rightarrow Global settings: Copy locked files.

	he volume snapshot created by the Volume hadow Copy Service is used when copying locked
fil • A re	les only. ccessing the Volume Shadow Copy Service equires FreeFileSync to be started with dministrator rights.

Troubleshooting

If you experience problems using the Volume Shadow Copy Service a renewal of registration might help. Create and execute a cmd batch file and insert the following lines or enter directly via command line:

```
cd /d %windir%\system32
Net stop vss
Net stop swprv
regsvr32 ole32.dll
regsvr32 oleaut32.dll
regsvr32 vss_ps.dll
Vssvc /register
regsvr32 /i swprv.dll
regsvr32 /i eventcls.dll
regsvr32 es.dll
regsvr32 stdprov.dll
regsvr32 vssui.dll
regsvr32 msxml.dll
regsvr32 msxml3.dll
regsvr32 msxml4.dll
```

Reference: <u>http://support.microsoft.com/kb/940032</u>



Automated Synchronization

The primary purpose of RealtimeSync is to execute a command line each time a directory becomes available (e. g. insert of a USB-stick) **or** when it detects changes in one of the monitored directories. Usually this command line will trigger a FreeFileSync batch job.

Example: Real time synchronization - in combination with FreeFileSync

Start RealtimeSync.exe located in FreeFileSync's installation directory and enter all folders you want to monitor. Instead of doing this manually you can import a ffs_batch file via **Menu** \rightarrow **Program** \rightarrow **Open**. This not only extracts all directories relevant for synchronization but also sets up the command line to execute the ffs_batch file each time changes are detected. Now press **Start** to begin monitoring.

😂 RealtimeSync - Automated Synchronization		x		
Program Help				
Usage: 1. Select folders to watch. 2. Enter a command line. 3. Press 'Start'. To get started just import a .ffs_batch file.				
Folders to watch:				
Drag & drop				
C:\Source	Brow	/se		
D:\Target	Brow	/se		
Idle time (in seconds): 10				
Command line:				
"C:\Program Files\FreeFileSync\FreeFileSync.exe" "D:\Backup Pro	ojects.ffs_ba	atch"		
Start				



Example: Automatic synchronization when a USB stick is inserted

Save a ffs_batch configuration in the USB stick's root directory, e.g. $H: \$, to let RealtimeSync call it when the stick is mounted. Configure RealtimeSync as follows:

S RealtimeSync - Automated Synchronization	
<u>P</u> rogram <u>H</u> elp	
Usage: 1. Select folders to watch. 2. Enter a command line. 3. Press 'Start'.	
To get started just import a .ffs_batch file.	
Folders to watch:	
Drag & drop	
H:\Data	Browse
Idle time (in seconds): 10	
Command line:	
"C:\Program Files\FreeFileSync\FreeFileSync.exe" "H:\Backup	Projects.ffs_batc
Start	

Whenever directory H:\Data becomes available, RealtimeSync executes the command line which starts the batch job located on the stick. RealtimeSync will also trigger each time files are modified in H:\Data.

Note

The full path of the last changed file and the action that triggered the change notification (create, update or delete) are written to the environment variables **%change_path%** and **%change_action%**.

Example: Log names of changed files and directories (Windows)

Show which file or directory has triggered a change. Enter command line: cmd /c echo %change_action% "%change_path%" & pause

Write a list of all changes to a log file: cmd /c echo %change_action% "%change_path%" >> C:\log.txt

Note

When RealtimeSync executes a Windows batch file (bat or cmd) a black console window is shown. You can hide it using the Visual Basic script HideConsole.vbs located in FreeFileSync's installation directory:

wscript "C:\Program
files\FreeFileSync\HideConsole.vbs"
C:\MyBatchFile.cmd

Limitations:

- If multiple changes happen at the same time, only the name of the first file is written to variable **%changed_file%**.
- While RealtimeSync is executing the command line, monitoring is inactive and changes occurring during this time are lost.

Run as Service (Windows)

RealtimeSync is designed to run as a background process which does not need further attention. Depending on your requirements there are a number of ways you can start it automatically. Generally the goal is to execute a command line of the form:

```
<FreeFileSync installation
folder>\RealtimeSync.exe <path to
*.ffs_real or *.ffs_batch file>
```

Example:

```
"C:\Program
Files\FreeFileSync\RealtimeSync.exe"
"D:\Backup Projects.ffs_real"
```

1. RealtimeSync should be monitoring only while a specific user is logged in: Create a new shortcut, enter the command line from above as target and place it into the user's autostart folder.



Security	Details	Previous Versions
General	Shortcut	Compatibility
R R	ealtimeSync.exe - Shortcu	ut
arget type:	Application	
Target location	: FreeFileSync	
Target:	neSync.exe" "D:\Back	up Projects.ffs_real"
<u>S</u> tart in:	"C:\Program Files\Free	FileSync"
Shortcut <u>k</u> ey:	None	
Run:	Normal window	
Comment:		
Open <u>File</u> L	ocation Change Ic	on Advanced
Open <u>Fi</u> le L	ocation Change Ic	on <u>Ad</u> vanced
	ОК	Cancel Appl

2. RealtimeSync should be monitoring while Windows is running irrespective of currently logged in users: Create a new task in your operating systems's task scheduler and have it execute the command line above when the system starts. See <u>Schedule a Batch</u> Job for an example how to add a task. Then change the user which runs the task to **SYSTEM** - a special user account always running in the background.

Create Basic Task Wizard	
5 Start a Program	
Create a Basic Task Trigger Daily Action	Program/script: "C:\Program Files\FreeFileSync\RealtimeSync.exe" Browse
Start a Program Finish	Add arguments (optional): Start in (optional):
	Sack Next > Cancel

FreeFileSync Links

Homepage: <u>http://freefilesync.sf.net/</u>

Project on SourceForge: feedback, suggestions and bug-reports http://sourceforge.net/projects/freefilesync/

If you like FreeFileSync: Support the project by a donation