

How to copy a directory structure but only include certain files (using windows batch files)

▲ As the title says, how can I recursively copy a directory structure but only include some files. E.g given the following directory structure:

91
▼
★
42

```
folder1
  folder2
    folder3
      data.zip
      info.txt
      abc.xyz
    folder4
    folder5
      data.zip
      somefile.exe
      someotherfile.dll
```

The files **data.zip** and **info.txt** can appear everywhere in the directory structure. How can I copy the full directory structure, but only include files named data.zip and info.txt (all other files should be ignored)?

The resulting directory structure should look like this:

```
copy_of_folder1
  folder2
    folder3
      data.zip
      info.txt
    folder4
```

Join **Stack Overflow** to learn, share knowledge, and build your career.

Email Sign Up

OR SIGN IN WITH

 Google

Facebook 

edited Feb 3 at 13:10



Yvette Colomb ♦

20.5k 15 71 114

asked Jan 23 '09 at 12:18



M4N

76.1k 40 194 250

15 Answers



You don't mention if it has to be batch only, but if you can use `ROBOCOPY`, try this:

148

```
ROBOCOPY C:\Source C:\Destination data.zip info.txt /E
```



EDIT: Changed the `/S` parameter to `/E` to include empty folders.



edited May 22 '13 at 18:04

answered Jan 23 '09 at 12:25



aphoria

16.2k 6 50 63

Can I copy *.zip with this syntax? I tried just writing *.zip instead of data.zip, but it didn't copy the files, only the folders. I also tried *.zip – [Niels Brinch](#) Sep 19 '12 at 21:47 ✎

@Niels Brinch Yes, you should be able to do that. What is the exact command line you are using? – [aphoria](#) Sep 19 '12 at 23:53

1 `ROBOCOPY C:\Source C:\Destination *.zip /E` – [Niels Brinch](#) Sep 20 '12 at 7:28

Weird. What OS are you using? Do you have the necessary file/folder permissions? I just tested on Win 7 and *.zip worked for me. – [aphoria](#) Sep 20 '12 at 9:56

Really!? I am also using Win 7. When I try, it just doesn't copy any files. Only folders. I'll try again. (while we're at it, what is the syntax to include more than *.zip - let's say *.zip and *.jpg. – [Niels Brinch](#) Sep 21 '12 at 12:46 ✎

Join **Stack Overflow** to learn, share knowledge, and build your career.

Email Sign Up

OR SIGN IN WITH



Google



An alternate solution that copies one file at a time and does not require ROBOCOPY:

29

```
@echo off
setlocal enabledelayedexpansion

set "SOURCE_DIR=C:\Source"
set "DEST_DIR=C:\Destination"
set FILENAMES_TO_COPY=data.zip info.txt

for /R "%SOURCE_DIR%" %%F IN (%FILENAMES_TO_COPY%) do (
    if exist "%%F" (
        set FILE_DIR=%%~dpF
        set FILE_INTERMEDIATE_DIR=!FILE_DIR:%SOURCE_DIR%=!
        xcopy /E /I /Y "%%F" "%DEST_DIR%!FILE_INTERMEDIATE_DIR!"
    )
)
```

The outer for statement generates any possible path combination of subdirectory in `SOURCE_DIR` and name in `FILENAMES_TO_COPY`. For each existing file `xcopy` is invoked. `FILE_INTERMEDIATE_DIR` holds the file's subdirectory path within `SOURCE_DIR` which needs to be created in `DEST_DIR`.

edited Apr 30 at 14:13

answered Jul 9 '09 at 14:48



sakra

41.4k

8

122

125

1 I needed to copy all photos from folders and subfolders, this batch script helped. Just changed the line `set FILENAMES_TO_COPY=data.zip info.txt` above to `set FILENAMES_TO_COPY=*.jpg` – Kai Noack May 8 '16 at 20:48 ✎

I'm getting an error because the file path has spaces on it (`SOURCE_DIR="C:\Origin Test"`, `DEST_DIR="C:\Dest Test"`, error: `Test""` was unexpected at this moment). I've tried to change `%var%` by `!var!` everywhere, but without success. Any clues on how to solve this? The whole structure is full of folder names with spaces. – ricardo.scholz Apr 30 at 14:10

1 Try `set "SOURCE_DIR=C:\Origin Test"` and `set "DEST_DIR=C:\Dest Test"`. Note the placement of the double quotes. – sakra Apr 30 at 14:12 ✎

Thank you, @sakra. It worked! :) – ricardo.scholz Apr 30 at 15:44 ✎

Join Stack Overflow to learn, share knowledge, and build your career.

Email Sign Up

OR SIGN IN WITH

 Google

Facebook 

cpio checks timestamp on target files -- so its safe and fast.

remove -v for faster op, once you get used to it.

edited Dec 18 '12 at 21:26

answered Dec 18 '12 at 20:40



Gabe Rainbow

2,368 3 24 34

2 @Christian it has a dependency on cpio which isn't part of standard install. – [Crashworks](#) May 2 '14 at 21:42

Thanks To Previous Answers. :)

7

This script named "r4k4copy.cmd":

```
@echo off
for %p in (SOURCE_DIR DEST_DIR FILENAMES_TO_COPY) do set %p=
cls
echo :: Copy Files Including Folder Tree
echo :: http://stackoverflow.com
rem    /questions/472692/how-to-copy
rem    -a-directory-structure-but-only
rem    -include-certain-files-using-windows
echo :: ReScripted by r4k4
echo.
if "%1"==" " goto :NoParam
if "%2"==" " goto :NoParam
if "%3"==" " goto :NoParam
setlocal enabledelayedexpansion
set SOURCE_DIR=%1
set DEST_DIR=%2
set FILENAMES_TO_COPY=%3
for /R "%SOURCE_DIR%" %%F IN (%FILENAMES_TO_COPY%) do (
if exist "%%F" (
set FILE_DIR=%%~dpF
set FILE_INTERMEDIATE_DIR=!FILE_DIR:%SOURCE_DIR%=!
xcopy /E /I /Y "%%F" "%DEST_DIR%!FILE_INTERMEDIATE_DIR!"
)
```

Join **Stack Overflow** to learn, share knowledge, and build your career.

Email Sign Up

OR SIGN IN WITH



Facebook

```
echo Eg.      : %0 D:\Root E:\Root\Lev1\Lev2\Lev3 *.JPG
echo Means   : Copy *.JPG from D:\Root to E:\Root\Lev1\Lev2\Lev3
```

It accepts variable of "Source", "Destination", and "FileName". It also can only copying specified type of files or selective filenames.

Any improvement are welcome. :)

answered Nov 10 '11 at 4:26



[Rhak Kahr](#)

650 7 10

▲ If Powershell is an option, you can do this:

7

```
Copy-Item c:\sourcePath d:\destinationPath -filter data.zip -recurse
```

▼ The main disadvantage is it copies all folders, even if they will end up being empty because no files match the filter you specify. So you could end up with a tree full of empty folders, in addition to the few folders that have the files you want.

answered Mar 16 '15 at 17:12



[Neil](#)

5,421 4 36 41

▲ With find and cp only:

5

```
mkdir /tmp/targetdir
cd sourcedir
find . -type f -name '*.zip' -exec cp -p --parents {} /tmp/targetdir ";"
find . -type f -name '*.txt' -exec cp -p --parents {} /tmp/targetdir ";"
```

edited Jan 9 '14 at 15:42

answered Jan 9 '14 at 15:34



Join **Stack Overflow** to learn, share knowledge, and build your career.

Email Sign Up

OR SIGN IN WITH



Google

Facebook

4

`@echo OFF`

```
:: Replace c:\temp with the directory where folder1 resides.
cd c:\temp
```

```
:: You can make this more generic by passing in args for the source and destination
folders.
```

```
for /f "usebackq" %%I in (`dir /b /s /a:-d folder1`) do @echo %%~nxi | find /V
"data.zip" | find /v "info.txt" >> exclude_list.txt
xcopy folder1 copy_of_folder1 /EXCLUDE:exclude_list.txt /E /I
```

answered Jan 23 '09 at 14:10

[Patrick Cuff](#)**23.8k** 8 57 92

3

That's only two simple commands, but I wouldn't recommend this, unless the files that you DON'T need to copy are small. That's because this will copy ALL files and then remove the files that are not needed in the copy.

```
xcopy /E /I folder1 copy_of_folder1
for /F "tokens=1 delims=" %i in ('dir /B /S /A:-D copy_of_files ^| find /V "info.txt" ^|
find /V "data.zip"') do del /Q "%i"
```

Sure, the second command is kind of long, but it works!

Also, this approach doesn't require you to download and install any third party tools (Windows 2000+ BATCH has enough commands for this).

answered Jan 23 '09 at 13:33

[Paulius](#)**4,559** 7 36 45

Join **Stack Overflow** to learn, share knowledge, and build your career.

Email Sign Up

OR SIGN IN WITH



2

Under Linux and other UNIX systems, using the `tar` command would do this easily.

```
$ tar cvf /tmp/full-structure.tar *data.zip *info.txt
```

Then you'd `cd` to the target and:

```
$ tar xvf /tmp/full-structure.tar
```

Of course you could pipe the output from the first `tar` into the 2nd, but seeing it work in steps is easier to understand and explain. I'm missing the necessary `cd /to/new/path/` in the following command - I just don't recall how to do it now. Someone else can add it, hopefully.

```
$ tar cvf - *data.zip *info.txt | tar xvf -
```

`Tar` (gnutar) is available on Windows too, but I'd probably use the `xcopy` method myself on that platform.

edited Nov 20 '17 at 15:28



StackzOfZtuff

1,005 12 15

answered Apr 18 '13 at 14:56



JohnP

129 2

`tar cvf /tmp/full-structure.tar **/data.zip **/info.txt` worked for me – [ssinganamalla](#) May 21 '13 at 13:25

Both "`tar cvf /tmp/full-structure.tar **/data.zip **/info.txt`" and "`tar cvf /tmp/full-structure.tar *data.zip *info.txt`" gave me 3 errors: (1) "`tar: *data.zip: Cannot stat: No such file or directory`", (2) "`tar *info.txt: Cannot stat: No such file or directory`", (3) "`tar: Exiting with failure status due to previous errors`". From which folder do I do all this? I do not understand this method. – [user1271772](#) Oct 5 '16 at 18:45

The question says WINDOWS – [Anfelipe](#) Jun 13 '18 at 15:58

Win10 has WSL, which makes this valid. `tar` has been available on Windows for 20+ yrs and the use of a pipe has worked since the MS-DOS days. In any shell environment, the `PWD` is important. `PWD` - present working directory. I will admit this solution is less for end users and more for power-users or programmers. – [JohnP](#) Jun 14 '18 at 16:47

Join **Stack Overflow** to learn, share knowledge, and build your career.

[Email Sign Up](#)

OR SIGN IN WITH



Google



edited Jan 23 '09 at 12:45

answered Jan 23 '09 at 12:32



gkrogers

6,958 2 25 33

This give an error: "File not found - data.zip". – M4N Jan 23 '09 at 12:43

1 it doesn't preserve the folder structure. – Paulius Jan 23 '09 at 13:38

The /E option preserves the folder structure, including creating empty folders if necessary. The filename also works with a wildcard (e.g. folder1*.zip). I only get the "File not found error" when I pass it a full filename that doesn't exist (in which case I get "File not found - xxx.xxx 0 File(s) copied". When I give it a pattern that returns no files (e.g. folder1*.zzz), I just get "0 File(s) copied". – gkrogers Jun 14 '12 at 0:18

Using WinRAR command line interface, you can copy the file names and/or file types to an archive. Then you can extract that archive to whatever location you like. This preserves the original file structure.

1

I needed to add missing album picture files to my mobile phone without having to recopy the music itself. Fortunately the directory structure was the same on my computer and mobile!

I used:

```
rar a -r C:\Downloads\music.rar X:\music\Folder.jpg
```

- C:\Downloads\music.rar = Archive to be created
- X:\music\ = Folder containing music files
- Folder.jpg = Filename I wanted to copy

This created an archive with all the Folder.jpg files in the proper subdirectories.

This technique can be used to copy file types as well. If the files all had different names, you could choose to extract all files to a single directory. Additional command line parameters can archive multiple file types.

More information in this very helpful link <http://cects.com/using-the-winrar-command-line-tools-in-windows/>

Join **Stack Overflow** to learn, share knowledge, and build your career.

Email Sign Up

OR SIGN IN WITH



Google



For those using [Altap Salamander](#) (2 panels file manager) : in the Options of the Copy popup, just specify the file names or masks. Easy.

1

edited Jan 13 '18 at 22:26



Michael Haephрати

1,639 17 29

answered Feb 14 '14 at 13:57



Hugo P

64 3

I am fine with regular expressions, lazy and averse to installs, so I created a batch file that creates the directory and copies with vanilla DOS commands. Seems laborious but quicker for me than working out robocopy.

0

1. Create your list of source files with complete paths, including drive letter if nec, in a text file.
2. Switch on regular expressions in your text editor.
3. Add double quotes round each line in case of spaces - search string `(.*)` replace string `"\1"` , and click replace all
4. Create two lines per file - one to create the directory, one to copy the file (qqq will be replaced with destination path) - search string `(.*)` replace string `md qqq\1\nxcopy \1 qqq\1\n` and click replace all
5. Remove the filename from the destination paths – search `\\([^\\""]+)"\n` replace `\\\"\\n`
6. Replace in the destination path (in this example `A:\src` and `B:\dest`). Turn OFF regular expressions, search `qqq"A:\src\` replace `B:\dest\` and click replace all.

md will create nested directories. copy would probably behave identically to xcopy in this example. You might want to add /Y to xcopy to suppress overwrite confirms. You end up with a batch file like so:

```
md "B:\dest\a\b\c\"
xcopy "C:\src\a\b\c\*.xyz" "B:\dest\a\b\c\*.xyz"
```

repeated for every file in your original list. Tested on Win7.

edited Feb 22 '15 at 23:03

answered Feb 22 '15 at 22:58



Chris

1,523 1 20 28

Join **Stack Overflow** to learn, share knowledge, and build your career.

Email Sign Up

OR SIGN IN WITH



Google

Facebook

-1

answered Sep 10 '11 at 11:10



Gnoy Dav

1

Thanks, but I was looking for a solution to be used from script (batch) files. So, no drag&drop... — [M4N](#) Sep 10 '11 at 19:33

▲ To copy all text files to G: and preserve directory structure:

-1

xcopy *.txt /s G:

edited Mar 1 '18 at 1:44



Cœur

20.4k

10

119

159

answered Feb 28 '18 at 21:18



Osbert Snudge

1

Join **Stack Overflow** to learn, share knowledge, and build your career.

Email Sign Up

OR SIGN IN WITH



Google

Facebook