

Using RichCopy

For

Bulk Fileshare Transfer

By Don Reese



RichCopy is a GUI-based, multi-threaded file copy tool provided by Microsoft. It takes advantage of the multi-threaded Windows environment to speed up multi-file transfers.

We use RichCopy to perform ad-hoc copies as well as to set up scheduled tasks to perform folder synchronization.

The first step is to obtain and install RichCopy. The software may be provided along with this document, or you may download it directly from Joshua Hoffman's article on Microsoft TechNet:

<http://technet.microsoft.com/en-us/magazine/2009.04.utilityspotlight.aspx>

Wipro standard is to create a directory on the C:\ drive called "**Install**" (if it does not already exist) :

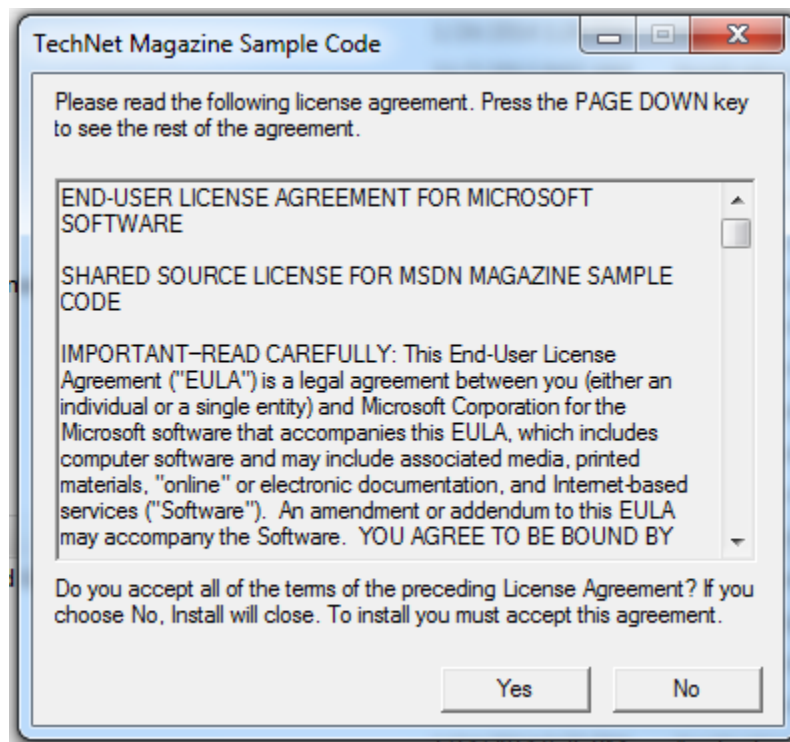
```
C:\> md Install
```

The location itself doesn't matter, but we will need to refer to the precise location later, as this is where we will also store our log files (this is configurable depending on requirements).

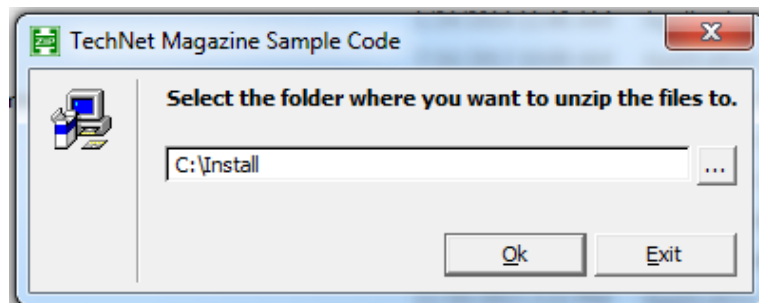
Then, copy the **RichCopy.zip** file included with this document or the **HoffmanUtilitySpotlight2009_04.exe** (which is a self-extracting archive) file downloaded from TechNet to this directory.

INSTALLATION:

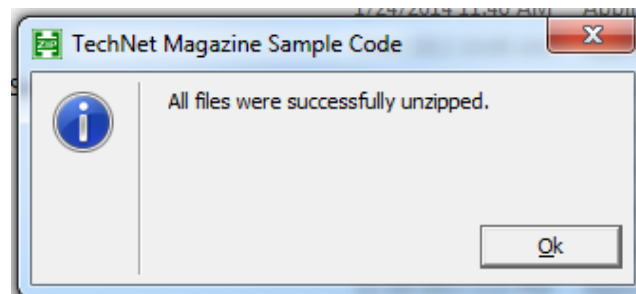
Next, extract the files from the .zip archive or run the downloaded executable file. If you downloaded the file from TechNet you will need to agree to the pop-up EULA:



Click the **Yes** button, and then specify the directory where you'd like to extract the files to:






Enter "**C:\Install**" and then click the **OK** button. Again, the location doesn't matter, but we will need to refer to it later, so it's best to keep it somewhere that is easily located.








Again click the **OK** button once the files are extracted.

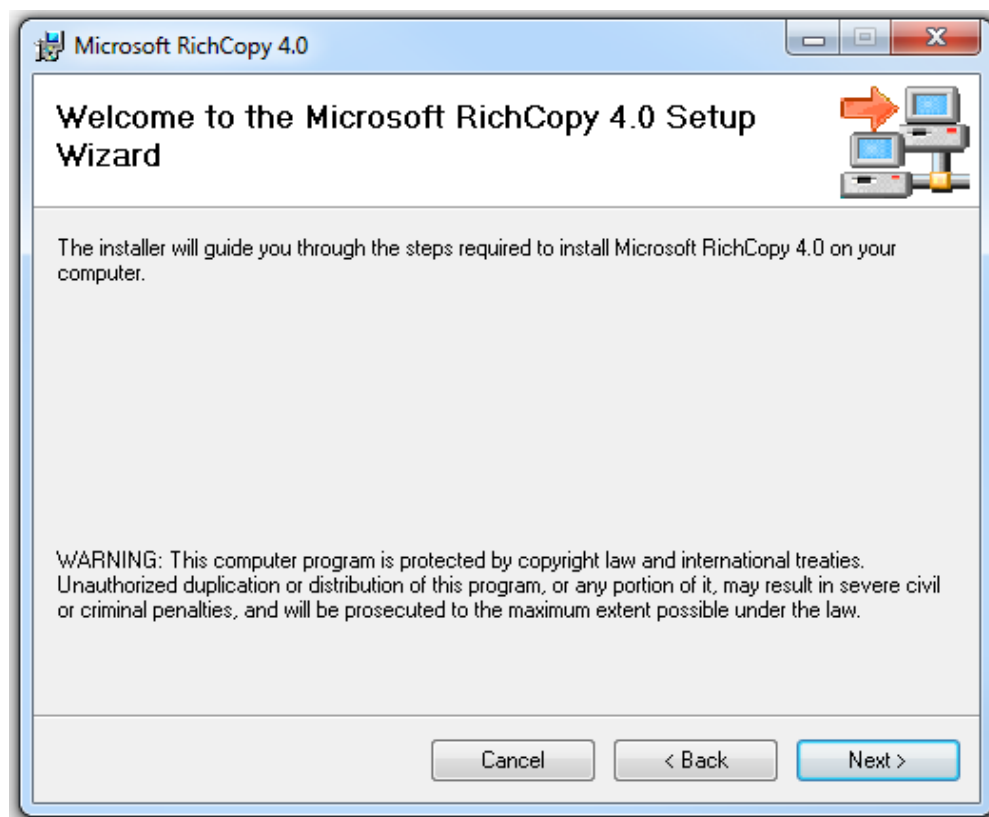
Depending on how you obtained the software, you will have either the **HoffmanUtilitySpotlight** directory or the **RichCopy .zip** file and **RichCopy** directory:

Name	Date modified	Type	Size
 HoffmanUtilitySpotlight	2/14/2014 1:14 PM	File folder	
 RichCopy	2/14/2014 12:11 PM	File folder	
 RichCopy.zip	2/14/2014 12:15 PM	Compressed (zipp...	5,804 KB

Inside the directory you will find the following files:

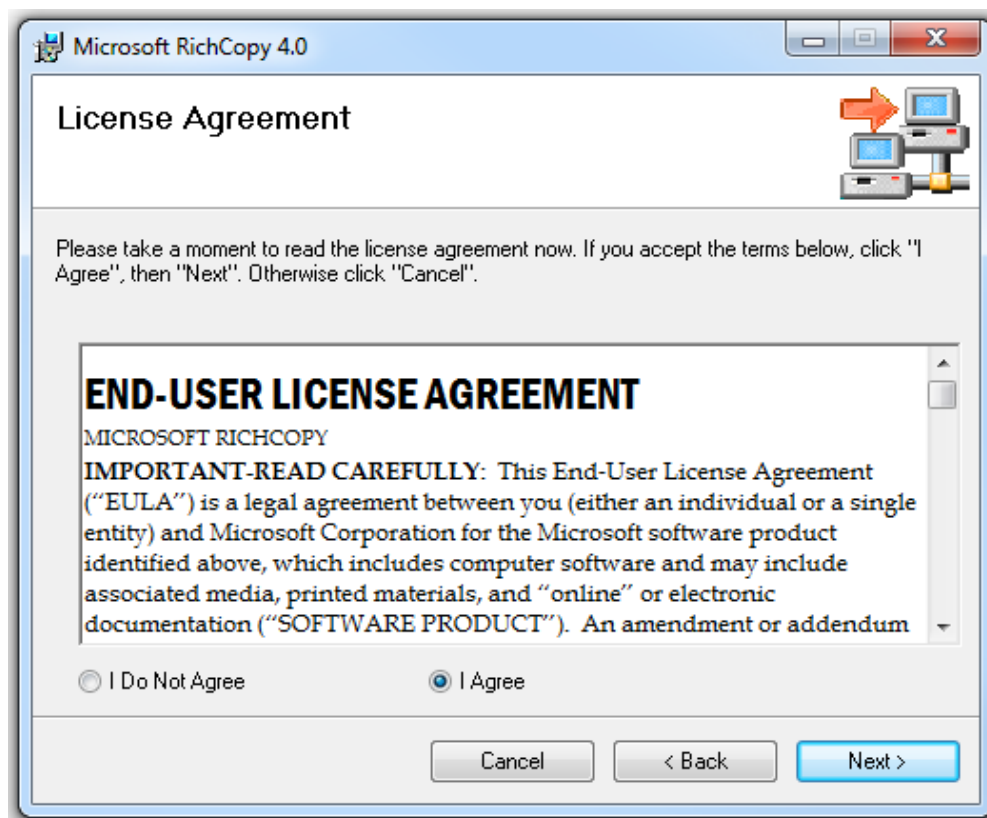
Name	Date modified	Type	Size
 EULA.doc	12/2/2002 2:59 PM	Microsoft Word 97 - 2003 Document	45 KB
 readme.txt	5/18/2009 11:40 AM	Text Document	2 KB
 RichCopy.chm	5/20/2009 8:17 PM	Compiled HTML Help file	778 KB
 RichCopySetup.msi	6/11/2009 12:32 AM	Windows Installer Package	5,229 KB
 setup.exe	6/11/2009 12:32 AM	Application	375 KB

At this point, run the “**setup.exe**” file by double-clicking on it, or by right-clicking on the file and selecting “**Open**” or “**Run**”. The setup wizard will appear:



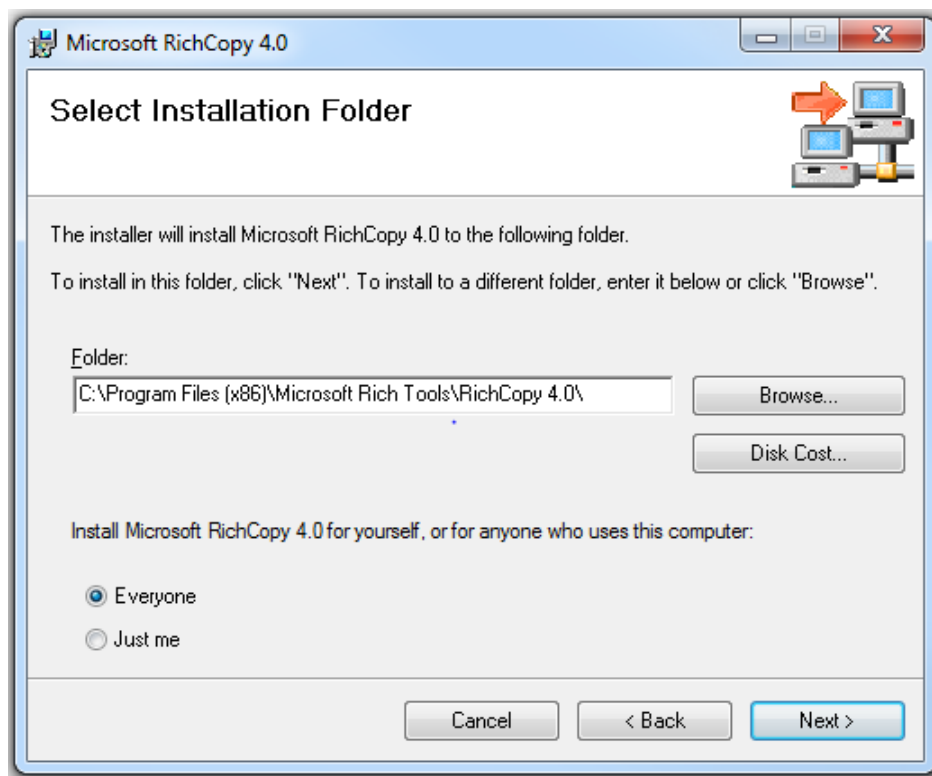
Click the **Next** button to continue.

The license agreement screen will appear:



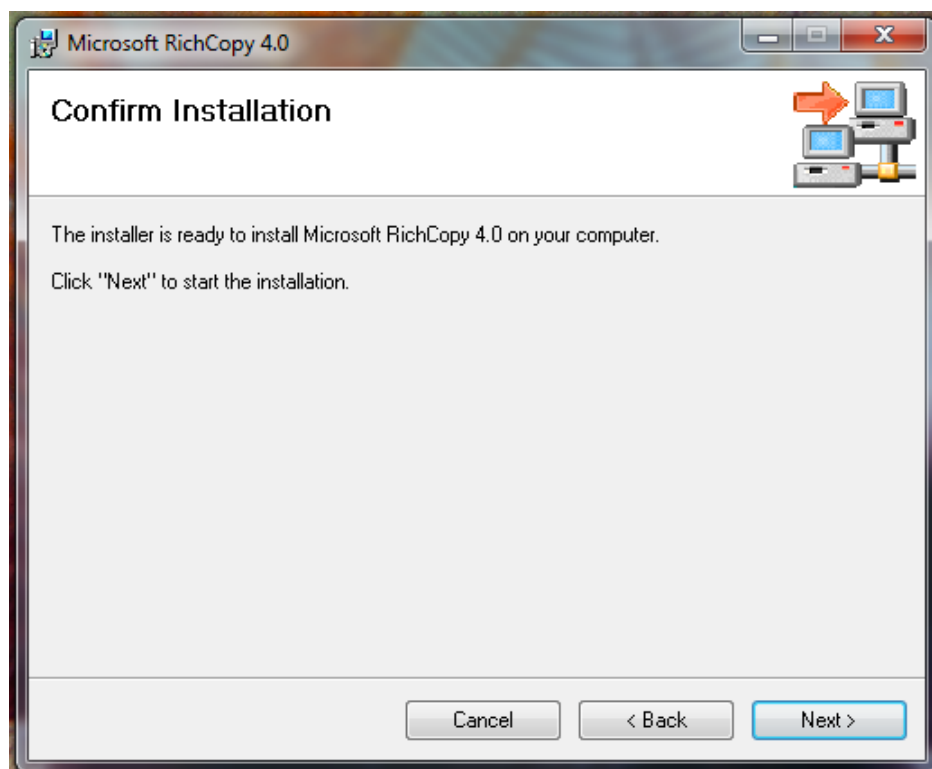
Select the "I Agree" radio control and click the **Next** button.

The installation destination screen will appear:

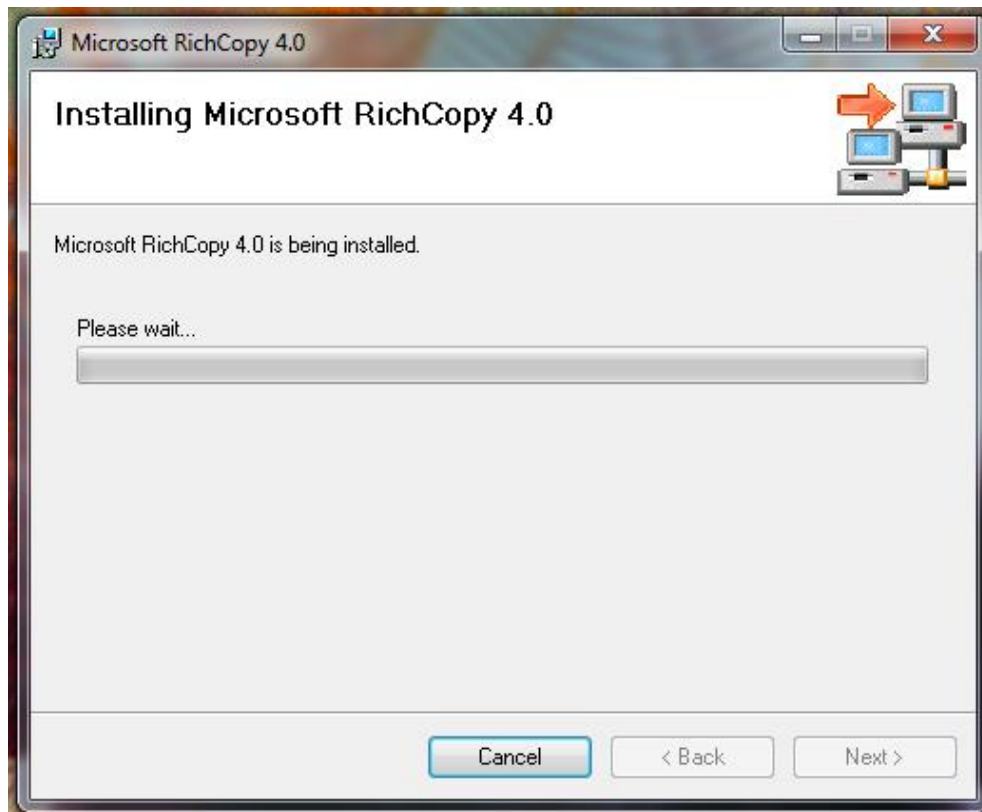


Leave the defaults as shown (unless you need to specify an alternate directory location), and click **Next**.

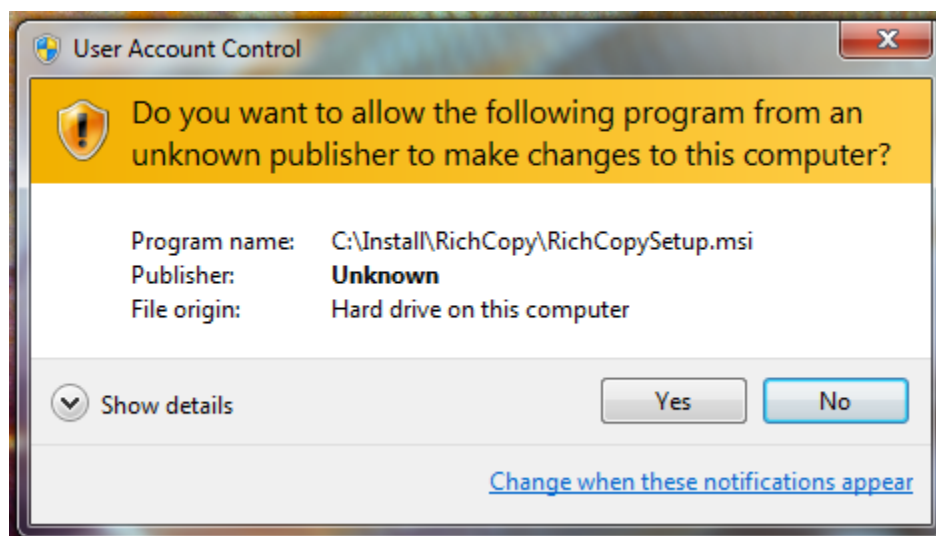
On the confirm installation screen, click the **Next** button:



RichCopy will now be installed:

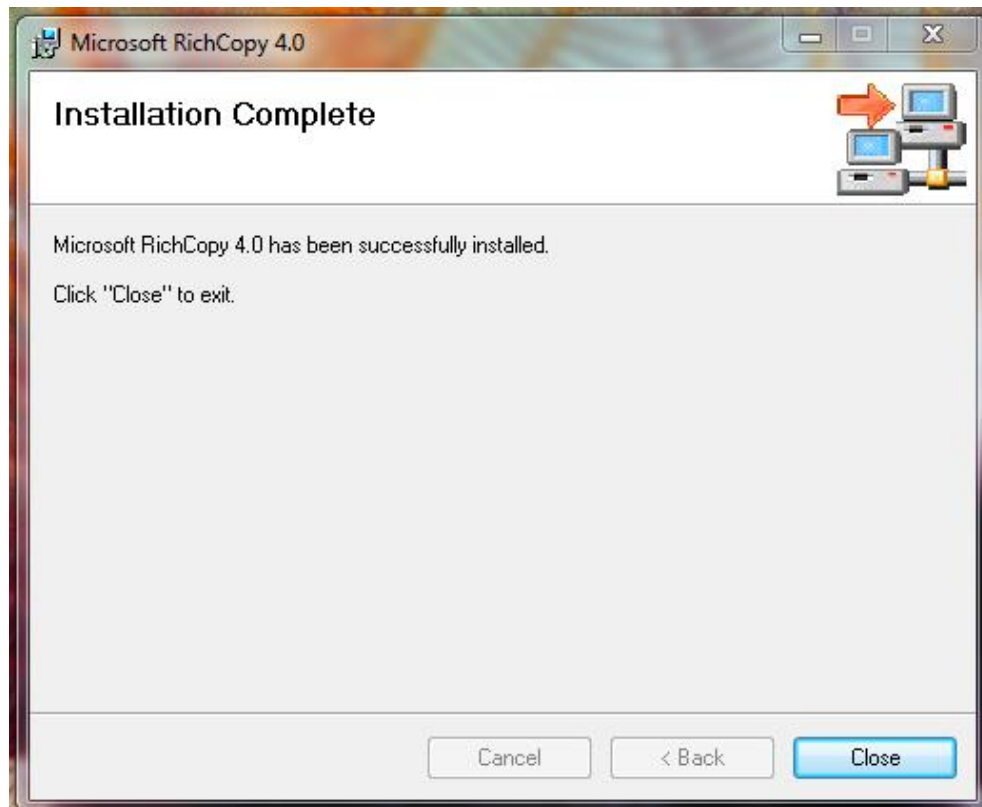


Depending on your installation, you may be presented with the UAC confirmation screen:



Click the **Yes** button to continue.

The installation process will complete:



Click the **Close** button.

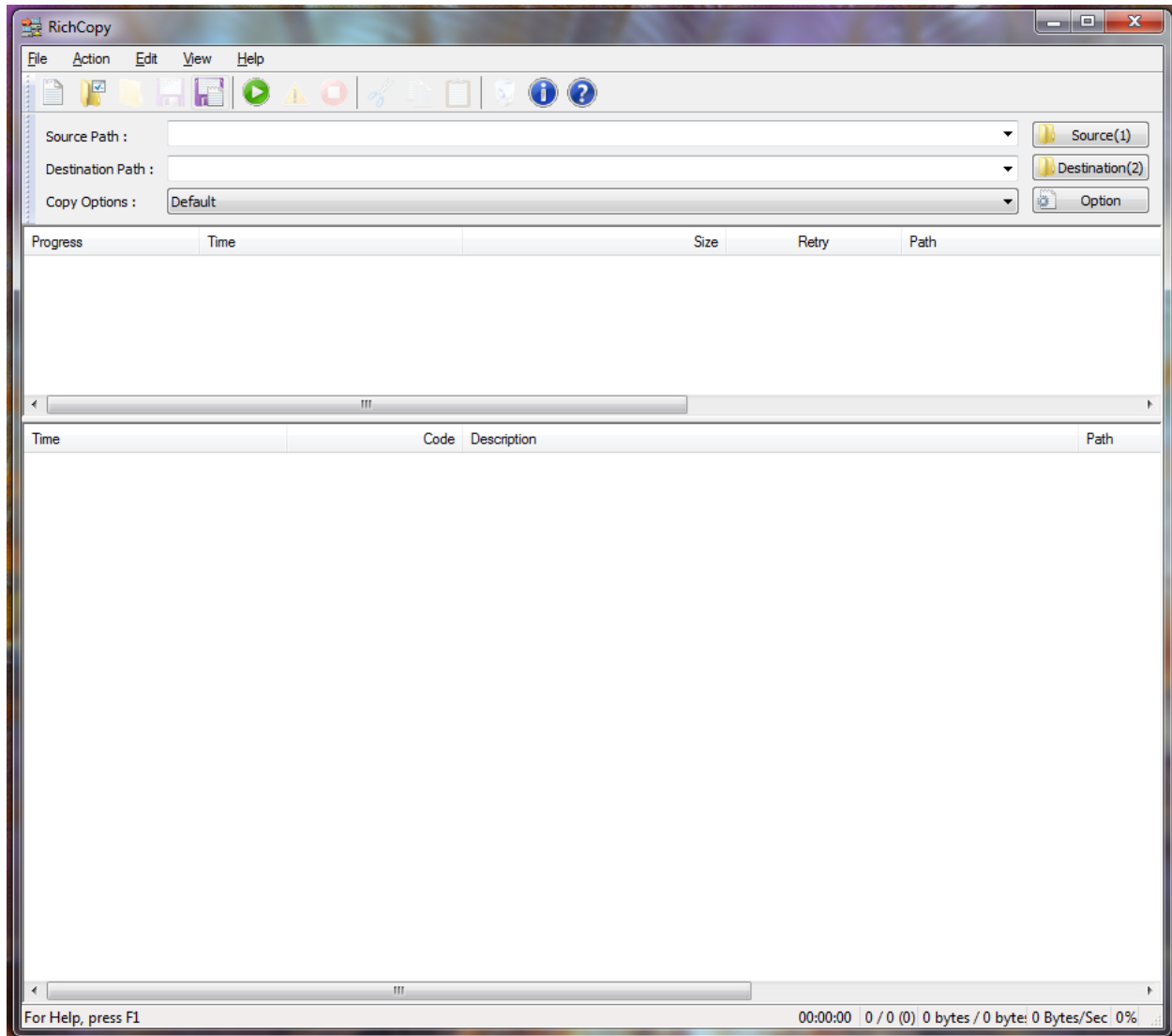
RichCopy is now installed! On to the hard part!

Note: Before you start blindly copying files across the ether, save yourself some grief. First, set up the destination with the proper permissions. Create a folder at the destination, and set the share permissions to the same as those on the source. For the file permissions, first disable the inherited permissions, and choose to copy them instead of removing them. Then, adjust the permissions to match the source. Doing this now instead of mid- or post-copy will eliminate both extraneous permissions flowing into your newly-copied file structure that you may not want there, and enable the file and directory permissions to flow properly at the destination. It will save you work.

CONFIGURATION:

Select “Start” or the Windows Start Button, then select “Programs” or “All Programs” (depending on your Windows version).

Open the “Microsoft Rich Tools” folder, and then open the “RichCopy 4.0” folder. Finally, select “RichCopy 4.0” to open RichCopy.

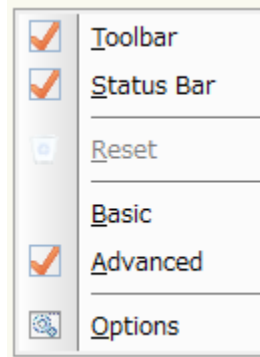


This is the main RichCopy screen, where we will set the Source and Destination paths, and view the copy status.

We'll explore RichCopy in greater detail soon, but for now please take notice of the **Source(1)**, **Destination(2)** and **Option** buttons in the upper right-hand corner of the screen. You will use these buttons to set your source and destination directories and access the option screens to configure how RichCopy works.

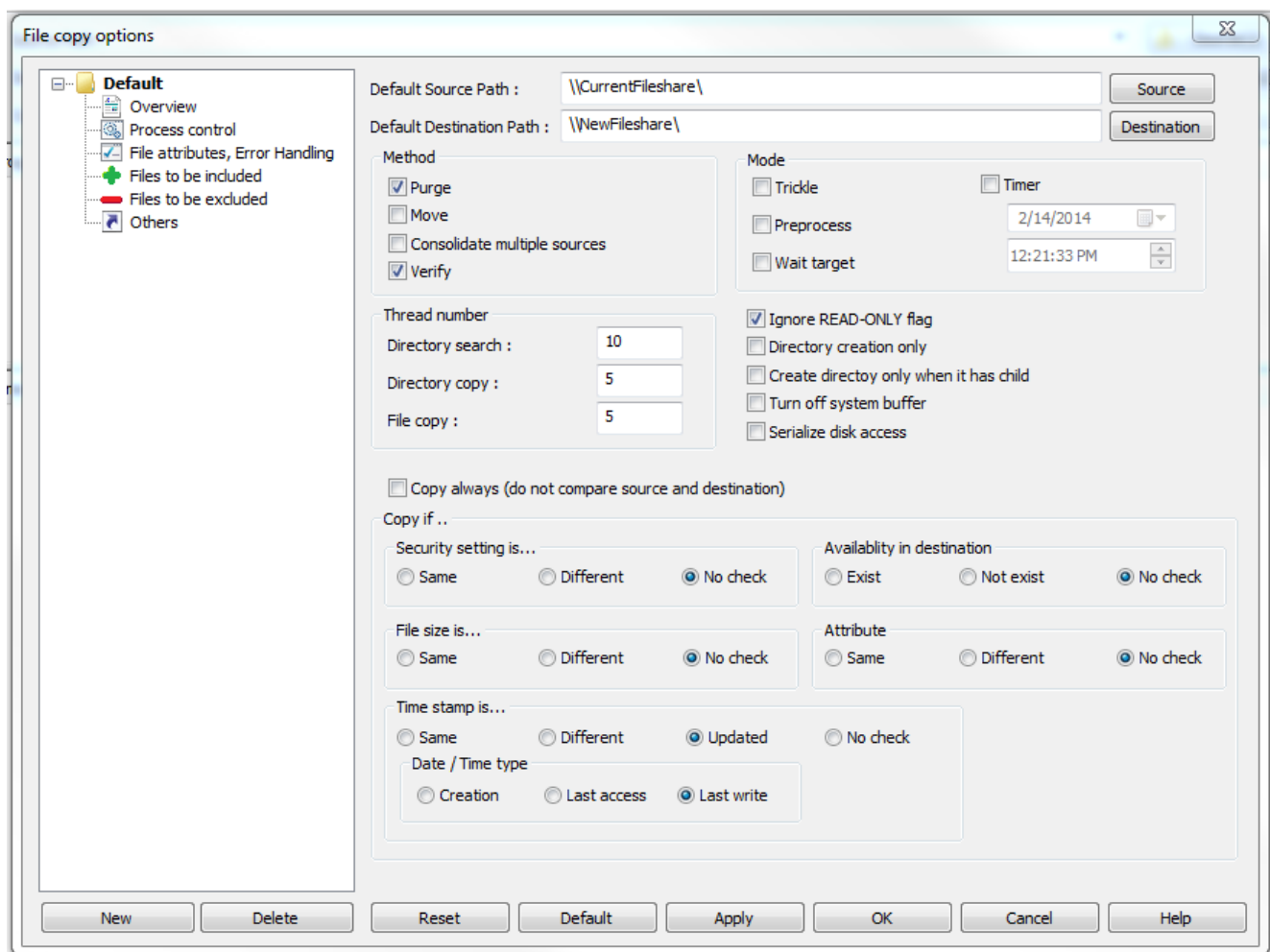
Also notice the green 'Play' VCR-like button just under the 'Help' menu item. You will also see a yellow 'Pause' and red 'Stop' button, although they are greyed out at the moment.

The first thing we need to do is change our view to **Advanced** mode, because that is where all the good stuff is:



You will find the selection for Advanced mode under the '**View**' menu item. Click on **View** in the menu bar, then click on **Advanced**.

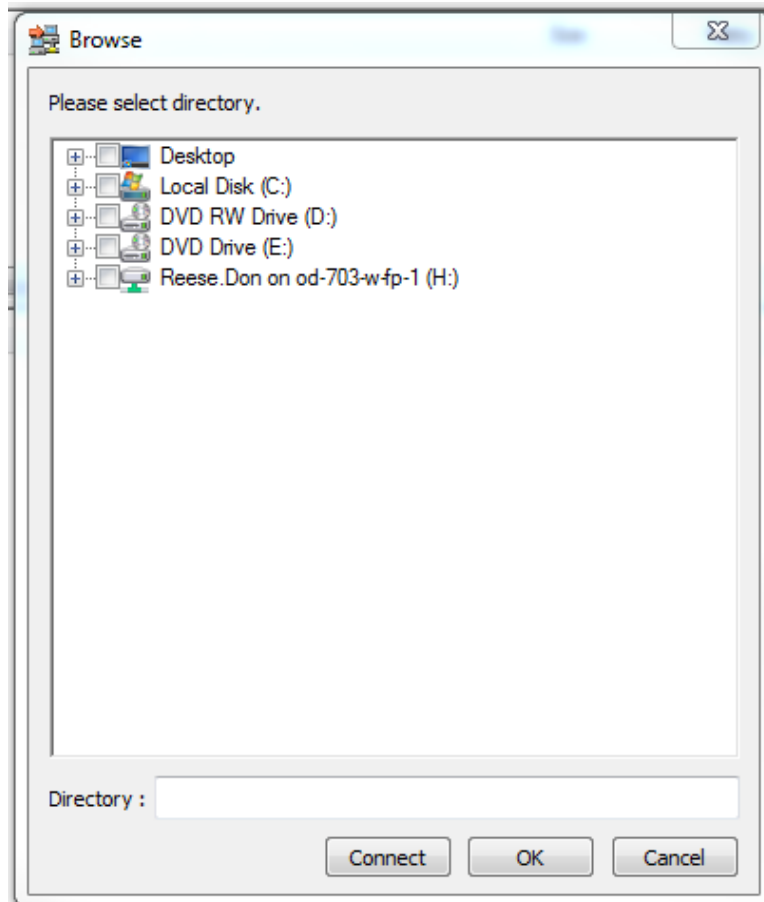
Next, click on the **Option** button to select the file copy options. You will be see the **Overview** option page:



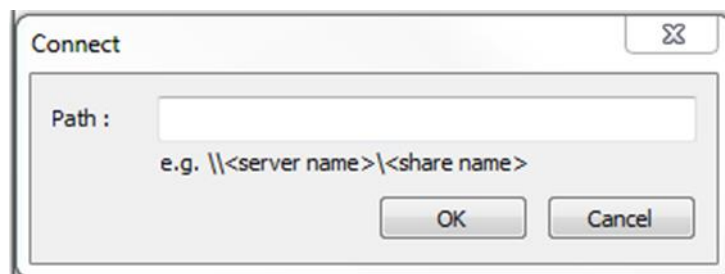
Note: If your screen shows only the '**Overview**' option page and not the others shown in the picture above, you are still in '**Basic**' mode. You can do , well, basic copies in '**Basic**' mode, but we typically want a little more

control over what gets copied and how it gets copied, so we will be using '**Advanced**' mode more often than not. So, please make sure you are in '**Advanced**' mode before proceeding.

Start by specifying your Source and Destination paths. You can directly type in a local (e.g., "C:\Install") or UNC (e.g., "\\fileserv\Install") file path, or you can click the **Source** or **Destination** buttons and browse for a location:



The **Connect** button will allow you to specify a direct UNC file path that you may not be able to easily browse to:



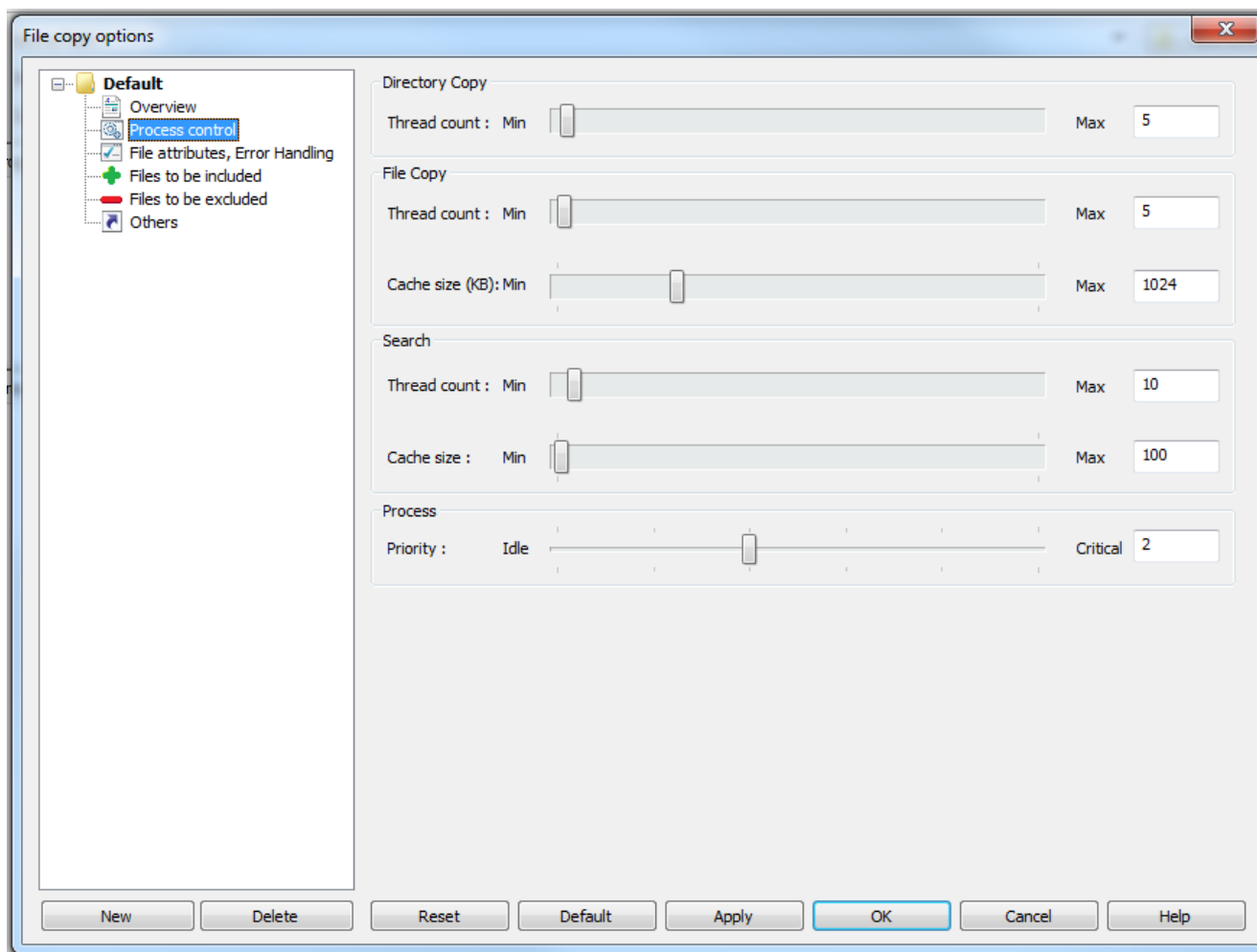
Enter your source path, then click the **OK** button; then enter your destination path, and again click the **OK** button.

Under the **Method** section, select the checkboxes labeled '**Purge**' and '**Verify**'. **Purge** will remove any files previously copied to the destination that have since been removed in the source. This is useful when you are

doing iterative copies prior to a final cut-over (This is the **'/P'** code you will see later). The **Verify** option (**'/V'**) will, of course, verify that the file we copied to the destination is the same as the source file.

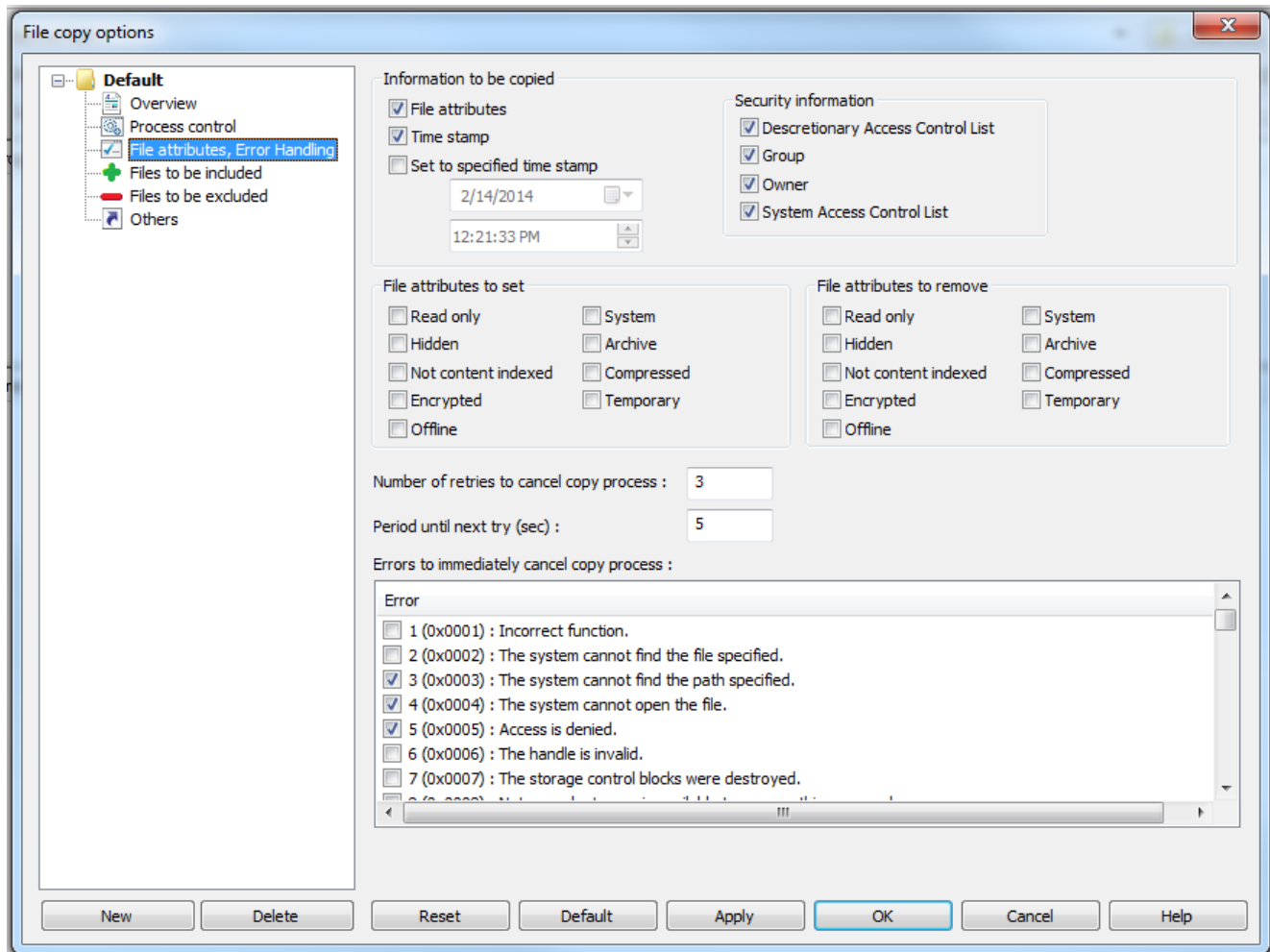
In the **Thread number** section, enter **'10'** for **Directory search** (**'/TS 10'**), **'5'** for **Directory copy** (**'/TD 5'**), and **'5'** for **File copy** (**'/TP 5'**). This controls how many simultaneous threads we will allow. Typically these numbers work best, as specifying too many threads can be counter-productive. You can experiment with higher values if you have a robust system, but these are typically what are recommended.

Click on the **Process Control** selection:



The values we entered on the Overview page under Thread number affect this page. You will see the numbers entered for **Directory Copy**, **File Copy** and **Directory Search**. Now you start to see why working in **Advanced** mode is helpful, as there are several other parameters we can change that are not available from the **Overview** page. For our purposes, the only one we really need to change is the **Cache size** under **File Copy**. Change that value to **"1024"** (**'/FC 1024'**). This value adjusts the amount of memory RichCopy can use as a buffer while copying files. This can greatly enhance the speed of the file copy process. You can specify more if you have abundant system resources available to you, but specifying more than 2048 will have less dramatic impact on performance.

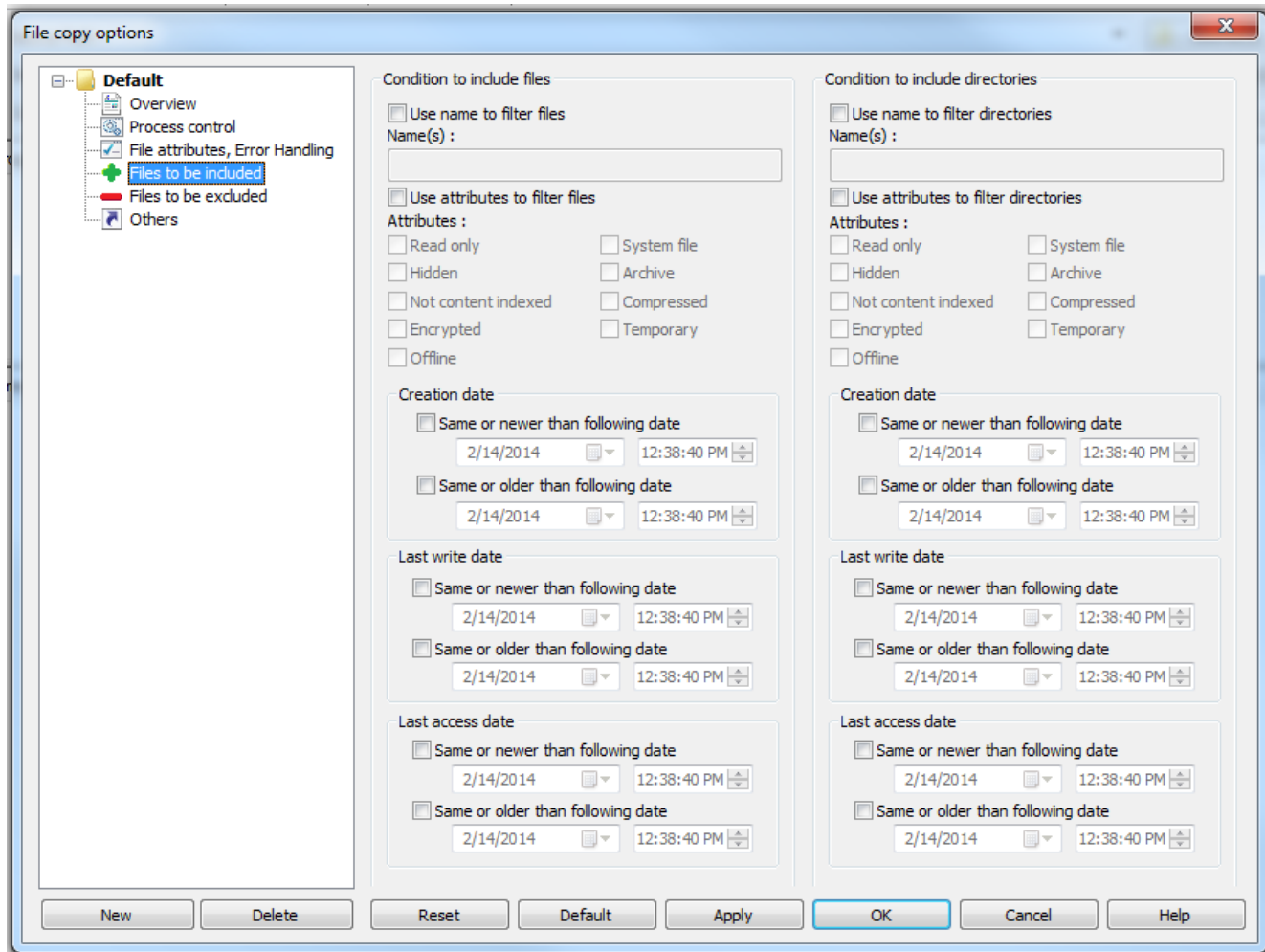
Click on the **File attributes, Error Handling** selection:



Here we want to select the options as shown above:

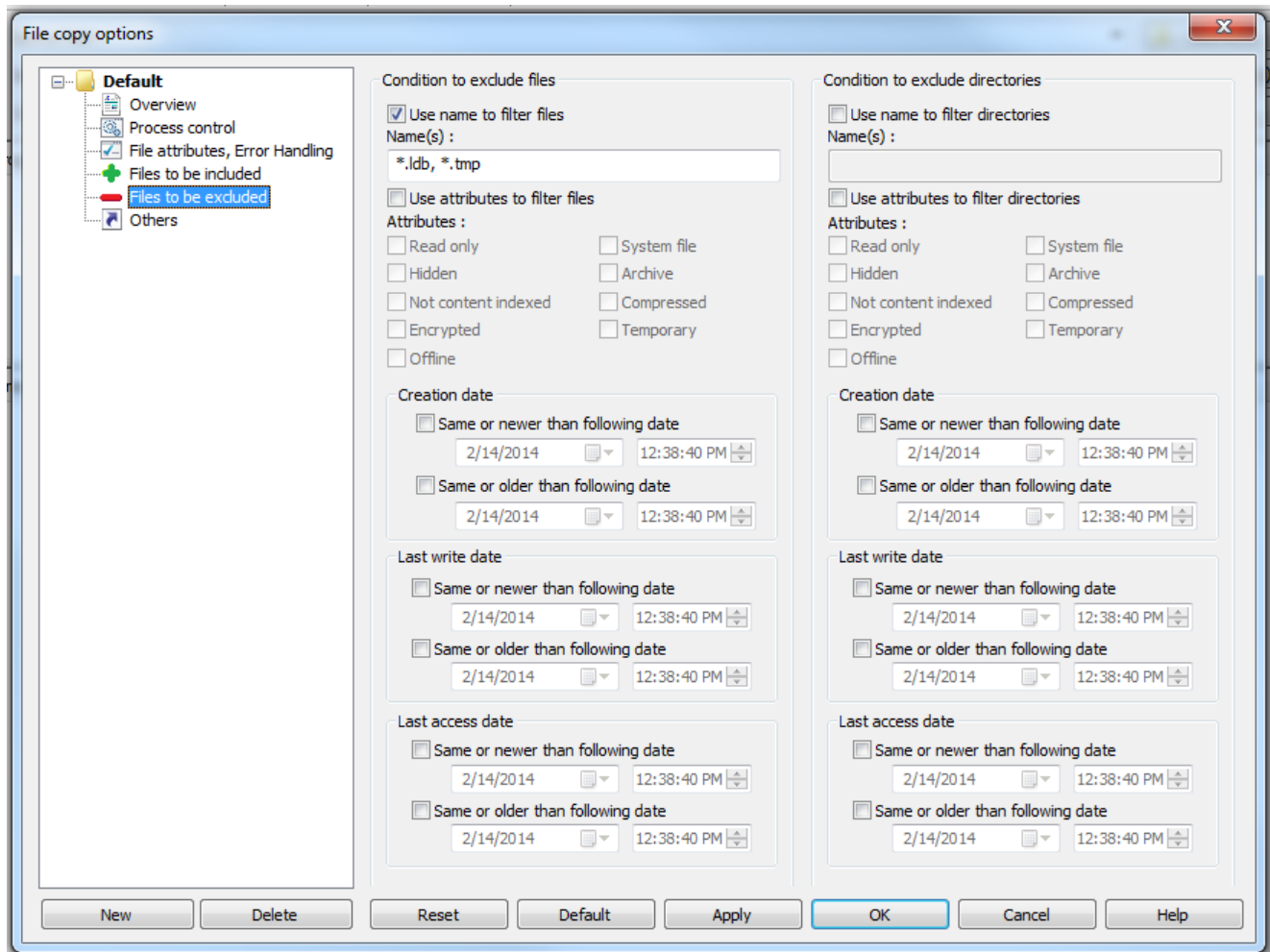
- **Information to be copied**
 - **File attributes ('/CA')**
 - **Time stamp ('/CT')**
- **Security information**
 - **Discretionary Access Control List ('/CSD')**
 - **Group ('/CSG')**
 - **Owner ('/CSO')**
 - **System Access Control List ('/CSS')**

Click on the **Files to be included** selection:



Generally, we make no modifications to this section, because we're already copying pretty much everything as it is. However, if you only wanted to copy certain files, such as .pst or .mp3 files (say, if you wanted to remove said files from your network file server, being in violation of corporate computer usage policies), you would do so here.

Click on the **Files to be excluded** selection:

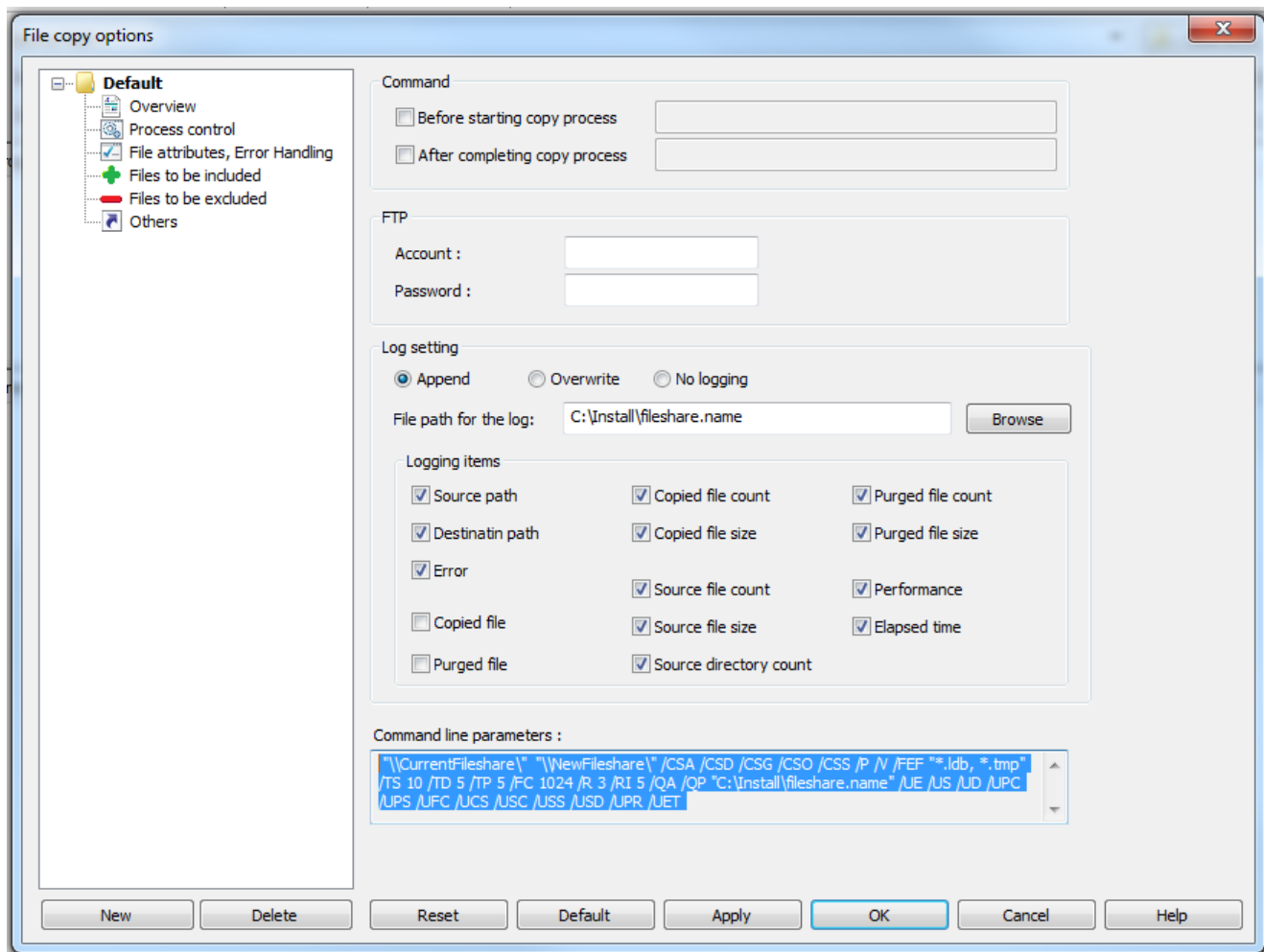


As on the Files to be included page, we will normally make very few, if any, modifications in this section, as in general we really want to copy every file from source to destination.

Some files we will want to exclude, however, are things like system-generated temporary files, database lock files, and similar files. We don't have to copy these files, as they are only temporary, after all, and they will be recreated if and as needed. Many applications will create temporary files. They won't always have a .tmp extension. Database lock files are created to show that particular users are accessing a database or record exclusively, and are deleted when the user exits the program. It helps to understand the type of data files in the source in order to determine what files, if any, should rightfully be excluded from the file copy process. Use your best judgment here. If you run an iterative copy process, the log files generated by RichCopy will alert you to files it was unable to copy, for one reason or another. This is a good place to look for candidates for exclusion.

Check the box **Use name to filter files**, and in the text box enter your file exclusions, in this case **`"*.ldb, *.tmp"`**, separated by commas (**`'/FEF'`**).

Click on the **Others** selection:



On this page we have a lot more going on than you might expect. First, we have the log settings, which are extremely important if we are doing iterative copies. I suggest you select everything, with the exception of **'Copied file'** and **'Purged file'**. These two options will unnecessarily increase the size of the log files in most cases, although there may be specific instances when you would want to use them, i.e., documenting your work. Typically the summary result produced and added to the end of the log at the completion of the file copy is sufficient:

```
5/21/2012 3:36:38 PM,0,Source path : \\source\f$\sourcefiles,
5/21/2012 3:36:38 PM,0,Destination path : F:\Shares\TSHomeshares-APAC,
5/21/2012 3:36:38 PM,0,Source directory count : 11,461 directories,
5/21/2012 3:36:38 PM,0,Source file count : 50,937 files,
5/21/2012 3:36:38 PM,0,Source file size : 2,043,450,537 bytes,
5/21/2012 3:36:38 PM,0,Copied file count : 1,483 files,
5/21/2012 3:36:38 PM,0,Copied file size : 154,733,839 bytes,
5/21/2012 3:36:38 PM,0,Purged file count : 161 files,
5/21/2012 3:36:38 PM,0,Purged file size : 109,936,039 bytes,
5/21/2012 3:36:38 PM,0,Elapsed time : 00:36:37,
5/21/2012 3:36:38 PM,0,Average performance : 70,429 bytes / sec,
5/21/2012 3:36:38 PM,0,Average performance : 0 files / sec,
5/21/2012 3:36:38 PM,0>Error is occurred during the copy process.,
5/21/2012 3:36:38 PM,0,Copy complete,\\destination\f$\destinationfiles
```


The most important section of the page is highlighted in blue, **Command line parameters**. This is the meat of RichCopy, and the part that lets us do iterative copies by creating a Scheduled Task in Windows. Remember all those codes I kept drawing you attention to? This is where they are hiding!

```
"\\CurrentFileshare\" "\\NewFileshare\" /CSA /CSD /CSG /CSO /CSS /P /V /FEF
\"*.ldb, *.tmp\" /TS 10 /TD 5 /TP 5 /FC 1024 /R 3 /RI 5 /QA /QP
\"C:\Install\fileshare.name\" /UE /US /UD /UPC /UPS /UFC /UCS /USC /USS /USD
/UPR /UET
```

You could run RichCopy right now and you will copy the files you have specified. Once. Then, you could run it again. Once more. Wouldn't it be nice to be able to run iterative copies of a filesahre that is being actively used, whose contents may be changing before the date and time we will actually cut-over and start using the new fileshare? There is!

You can copy and paste the entire box contents to a text file, which will become our script that we will create a scheduled task to run for us. However, it's not yet complete; we still need to add something to it, namely, the program that will interpret this weird string of seemingly nonsensical characters: **RichCopy!**

```
"C:\Program Files (x86)\Microsoft Rich Tools\RichCopy 4.0\RichCopy64.exe"
"\\Current Fileshare\" "\\New Fileshare\" /CSA /CSD /CSG /CSO /CSS /P /V
/FEF \"*.ldb, *.tmp\" /TS 10 /TD 5 /TP 5 /FC 1024 /R 3 /RI 5 /QA /QP
\"C:\Install\fileshare.name\" /UE /US /UD /UPC /UPS /UFC /UCS /USC /USS /USD
/UPR /UET
```

Notice I have added the complete pathname to where RichCopy resides, and surrounded it in quotes, which is necessary for the command-line to interpret a pathname with spaces. Likewise the source and destination paths must have double quotes for the same reason. If you don't have any spaces in your pathnames, you can probably safely leave them off, but it won't hurt to include them even if you don't need them. It also helps to make the important bits of this script stand out a bit more.

If necessary, you can also make quick tweaks to the options you specified, instead of going all the way through the RichCopy GUI again. Once you have the basic script, you can adjust as necessary.

Now, you take the textfile you just populated with your RichCopy script, and save it back into an easily-accessed folder, preferably one we already have, like '**C:\Install**'. The filename extension doesn't really matter, although by convention I like to save them as *.cmd files, for 'command', since we're running a command-line command...

Now time to turn this into a scheduled task that will run while we sleep!

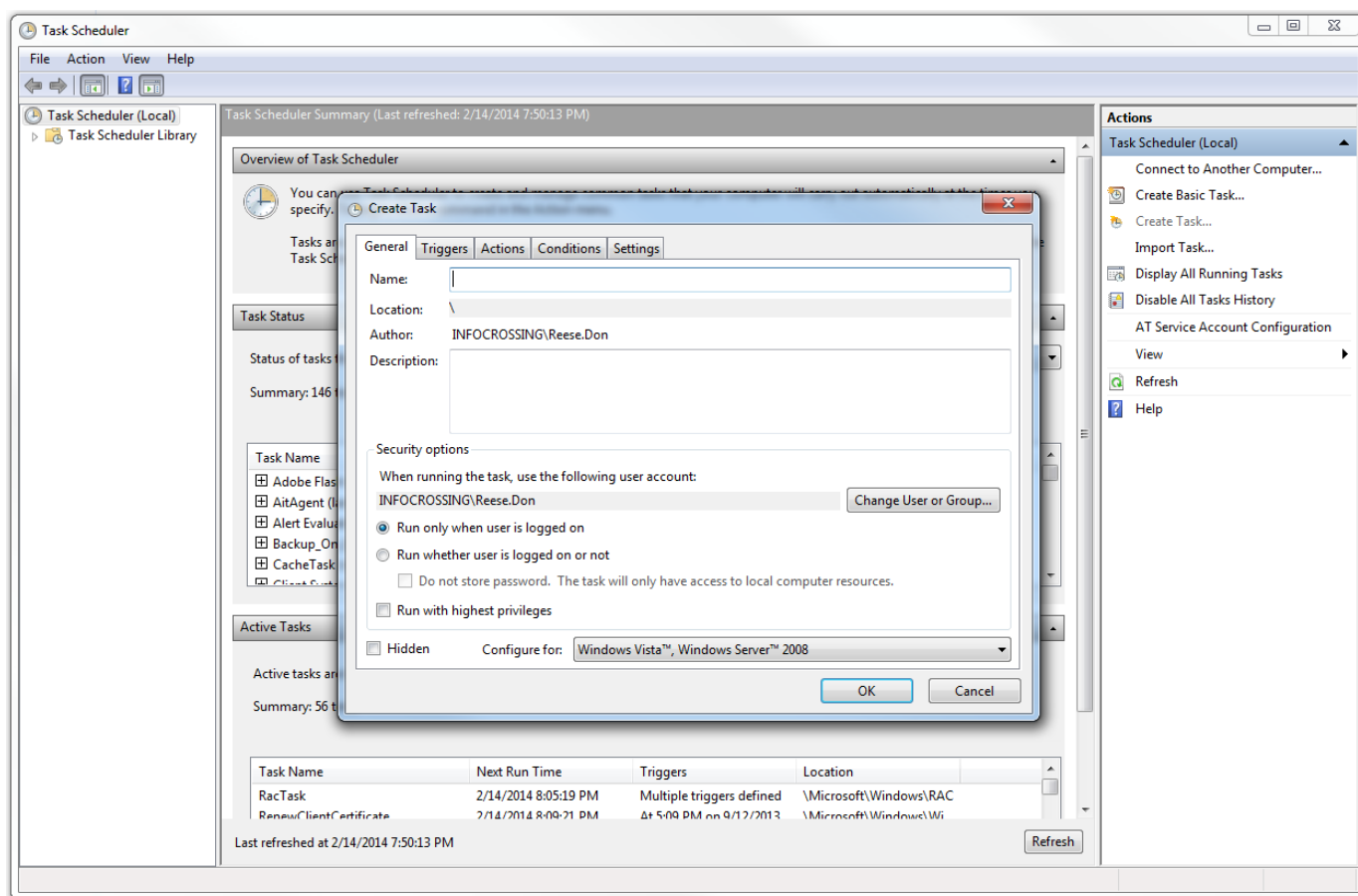
SCHEDULED TASK CONFIGURATION:

Setting up a scheduled task has changed considerably from Windows 2000/2003 to Windows 7/2008, however, the basic requirements are still the same:

- An application or script to run
- A user context in the form of a username and password to run it under
- A schedule

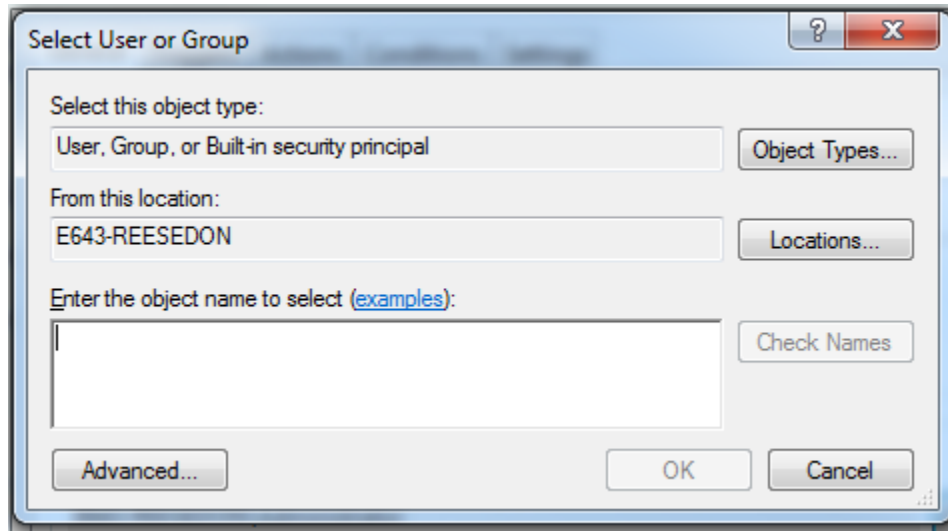
The following screenshots show a Windows 7 workstation, but Windows 2008 server is functionally identical. Remember the screens may look slightly different, but the basic setup is the same. First, open up the **Task Scheduler** program in your version of Windows. This may be found either in the Control Panel in older versions of Windows, or in Administrative Tools.

On the right hand side of the Task Scheduler window, in the **Actions** pane, select '**Create Task...**' (If you don't see the Action Pane, it may be hidden; you can get to it from the **Action** menu item, or select '**View**', select '**Customize View**', and check the box next to '**Action pane**', then click '**OK**');



Type in a name for your scheduled task, something like 'FileshareCopyProcess' in the box marked '**Name:**'. Optionally enter a short description of what the task does in the box helpfully designated '**Description:**'. In the section marked '**Security options**', you will need to select a user account for the scheduled task to run under. I suggest using a domain administrator account, but any account with full read permissions on the source and full read/write permissions on the destination should do.

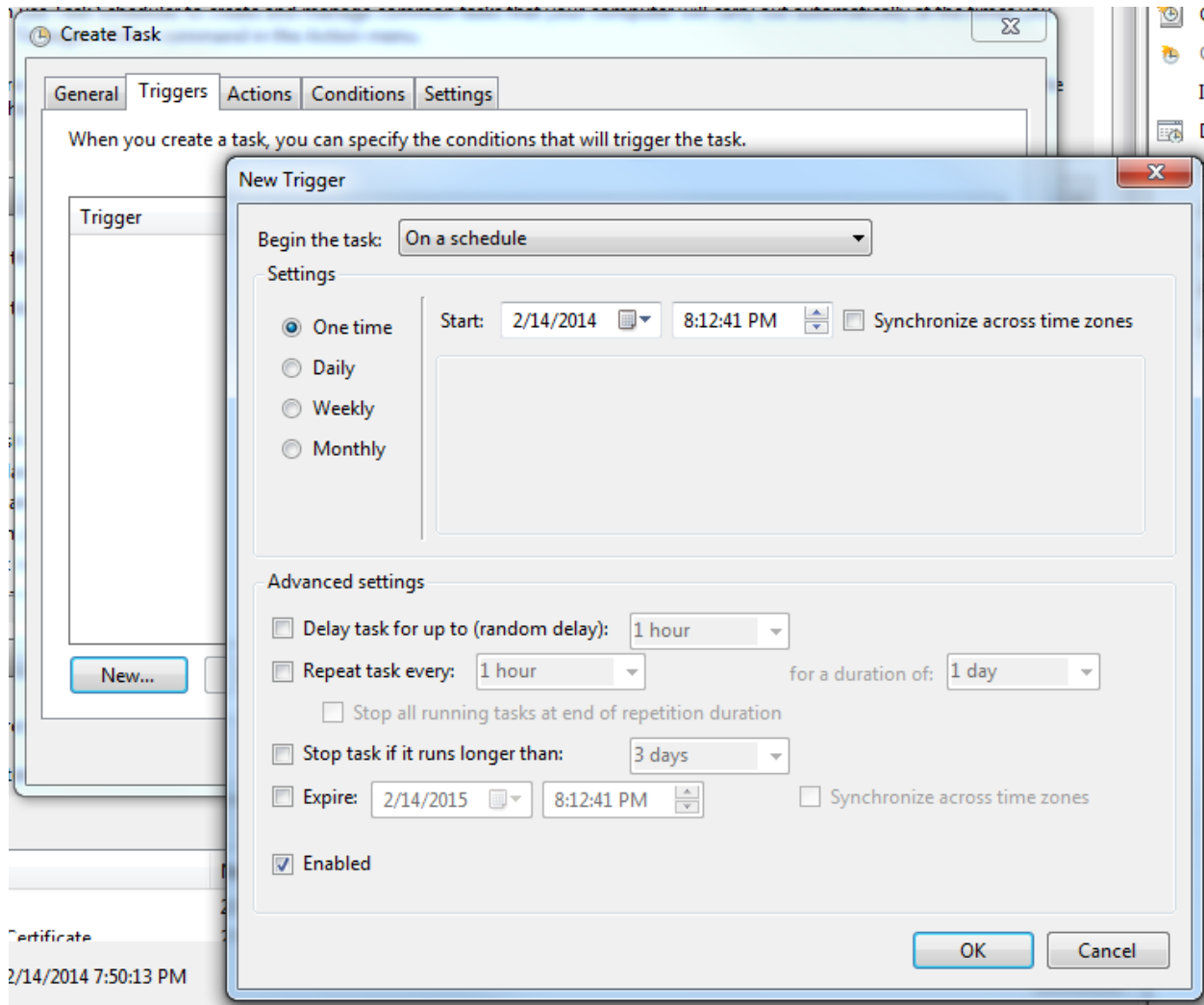
Click on the '**Change User or Group...**' button:



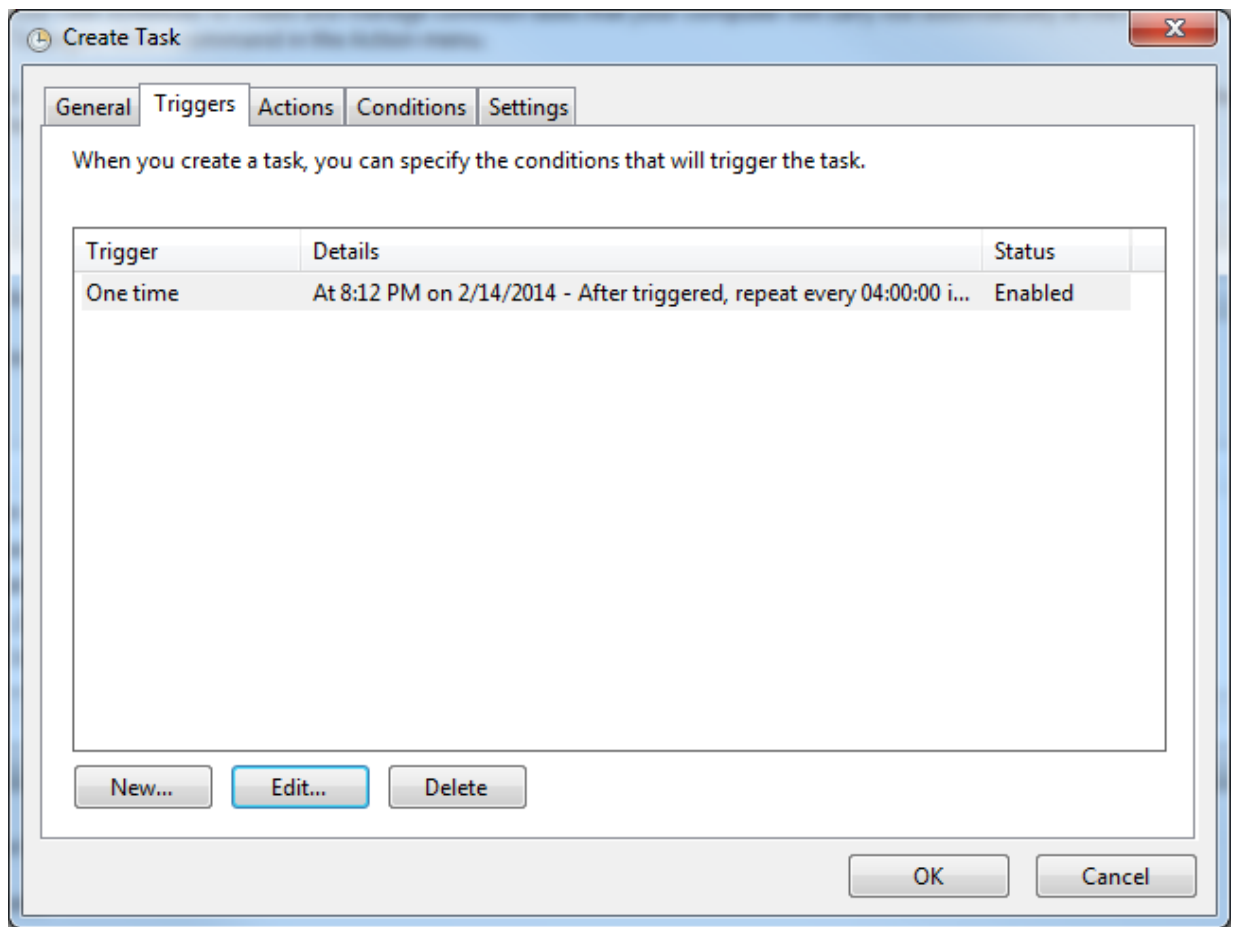
Enter a user account name in the box and click on the '**Check Names**' button. It will validate that the account exists either locally or on the domain. Click on the '**OK**' button to close the '**Select User or Group**' box.

Next, Select the '**Run whether user is logged on or not**' radio control, and the '**Run with highest privileges**' checkbox for good measure. You may select the '**Do not store password option**', but you will have to make sure the user account has the '**Logon as batch job**' policy assigned. I recommend making your life easier and go with the first choice. Finally, you should adjust the '**Configure for:**' drop-down list box for the lowest version operating system you'll be working with.

Select the '**Triggers**' tab and click the '**New...**' button:

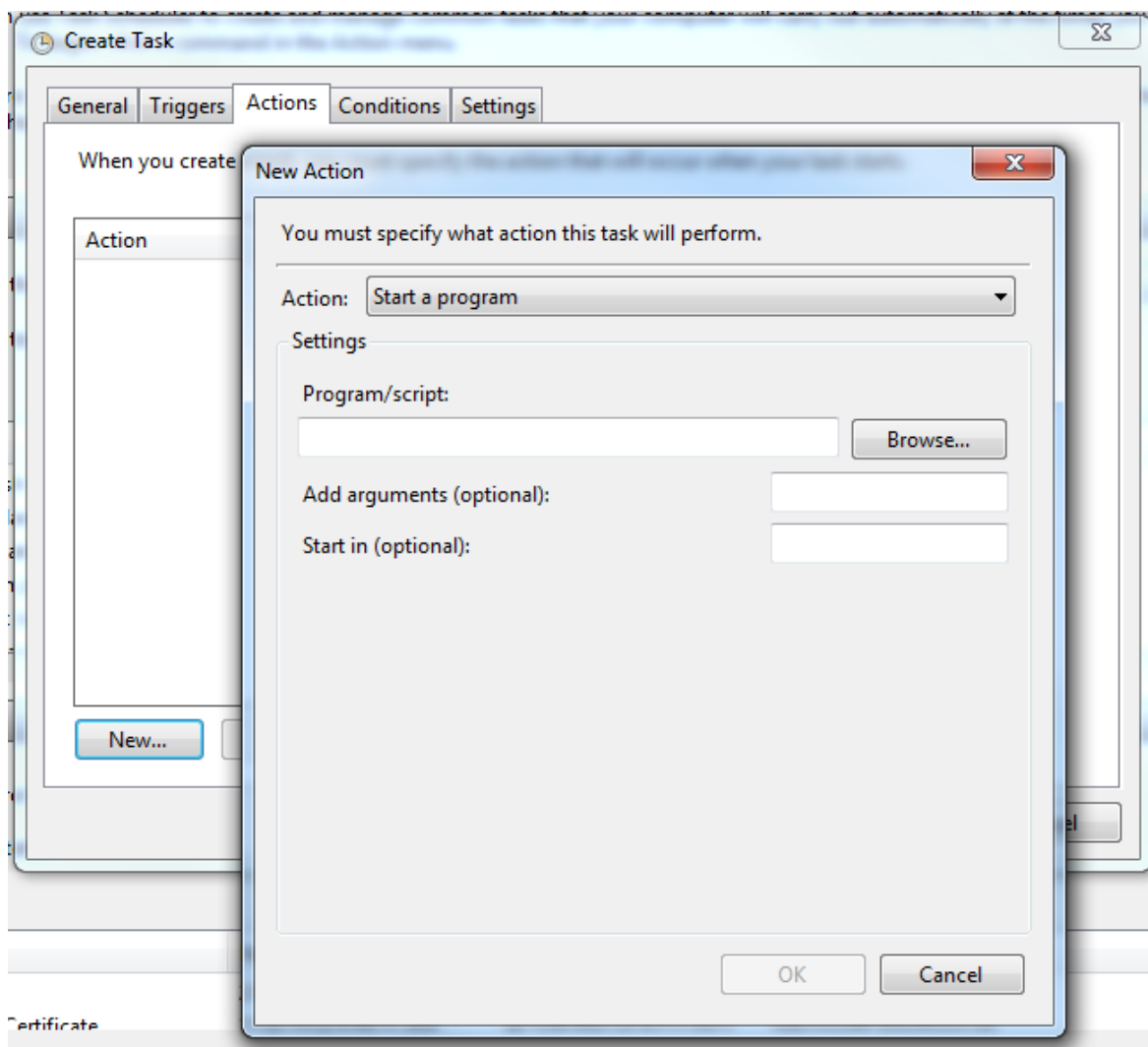


Here is where we'll setup the task schedule, or 'trigger'. You can set the date and time you'd like the task to start. I like to set things up initially at even hour intervals. Click on the '**Repeat task every:**' checkbox. You'll notice that in the drop-down list box a few default selections from 5 minutes up to 1 hour. Don't worry, those are not your only options. Simply type your desired interval into the list box where it says '1 hour'. It can be a number of minutes, such as '**120**' for two hours, or you can simply type in '**2 hours**'. Remember, the first time you run the script, it may take a very long time to complete, as it really is copying everything. Subsequent runs will usually take far less time, so you can gradually adjust this downward if warranted to get more frequent cycles. Again, it depends on the amount of data you're trying to copy, but '**4 hours**' may be a good place to start. Leave everything else as default and then click the '**OK**' button:



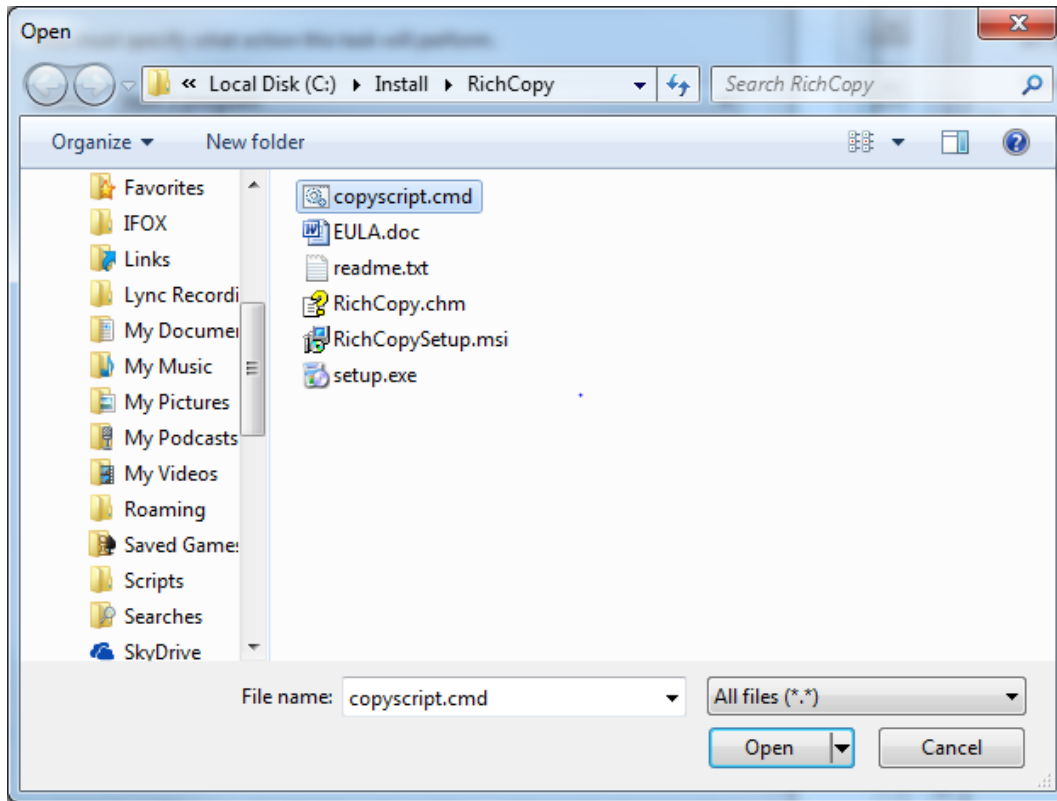
Now you can see our scheduled task is well, scheduled. Now we have to tell it what we want it to actually do.

Select the '**Actions**' tab, and click the '**New...**' button:

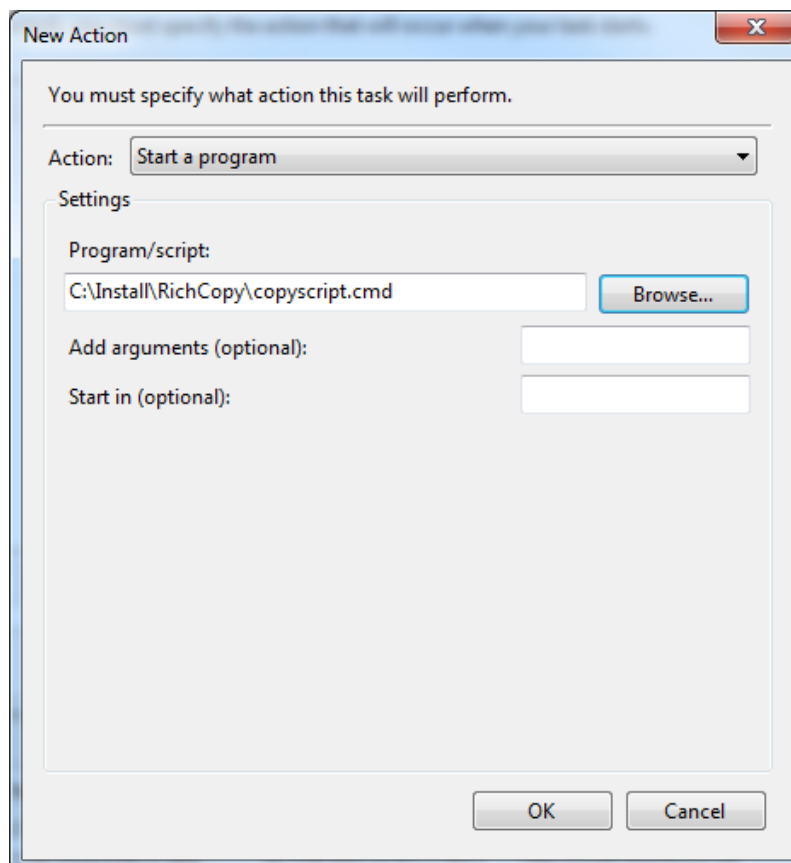


Starting a program is exactly what we want to do; we want to start our nifty little script, to be more precise. Click the '**Browse...**' button and locate the script file we saved (hopefully in 'C:\Install!').

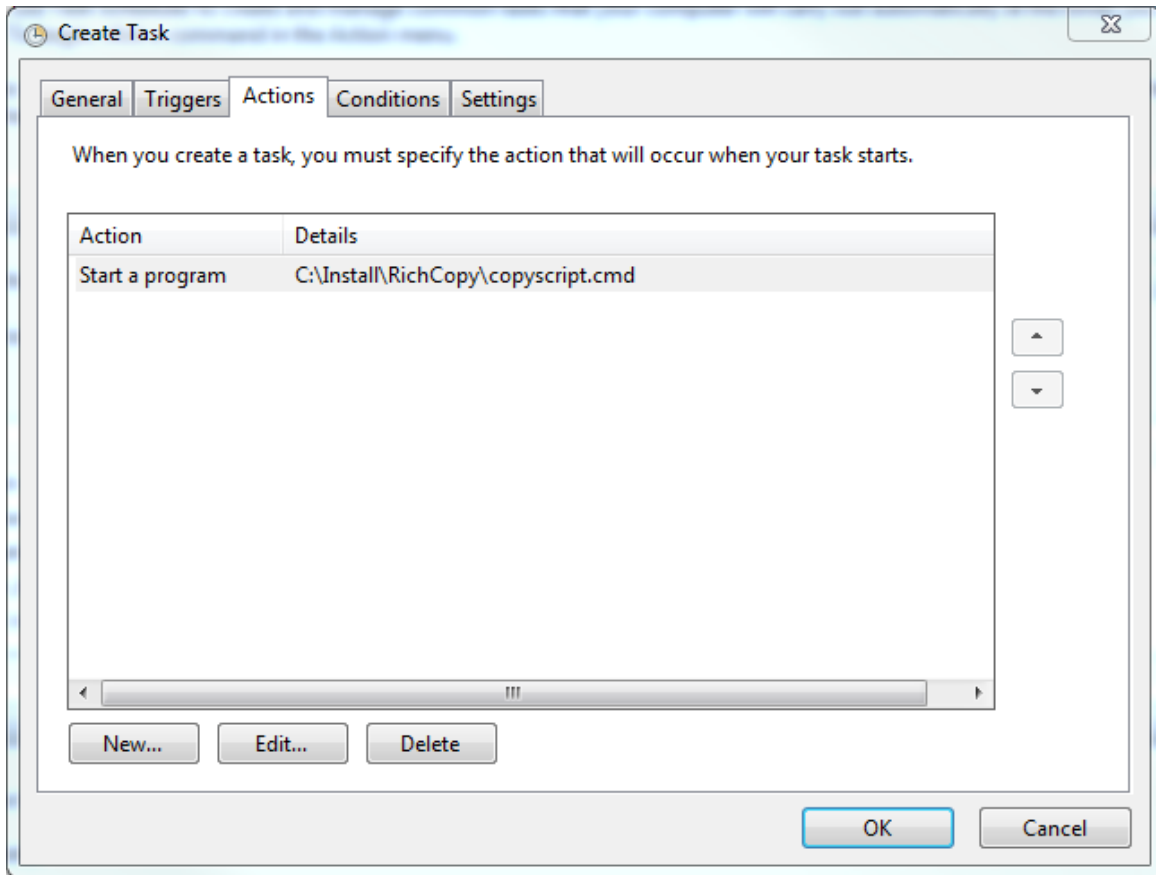
When you've found it, select it, then click the '**Open**' button.



Then, Click the '**OK**' button. That's about it!



Click the '**OK**' button once again:



If all went according to plan, you should now see the new scheduled task in the list of Active Tasks. This is the trickiest part of the whole process, so you may have to tinker with it a bit. On Windows 2000/2003/XP systems, setting up a task is a bit simpler, but you need the same 3 basic pieces of information, as outlined previously.