Copy & Mask Files (CAMF)

This was created in response to the User Story: [47178 - Prepare modified copies of WSU files for SAP HANA Proof of Concept](https://dev.azure.com/watech-vsts/HRLabor/_workitems/edit/47178)

**Description:**

CAMF is a console CLI app. It copies all files in a Source location, checks it they exist in a Destination location. If the Source files are not in the Destination location, then they are copied to the Destination location. Once the files have been copied to the Destination location, then each file is evaluated to for the fields to be masked. Each file is masked appropriately. If a file does not have a column specified to be mask then it is not updated.

**Arguments:**

These are the values passed into the app when executed. To see example of how to call the app see below.

**CopyAndMaskFiles** SourcePath DestinationPath FieldsToMask LoggingOptions ManualRun

If the arguments have spaces in them, then it is required that you surround the argument with double quotes (“”). If an argument does not have a space, then you do not need the double quotes, but it will work with them. I would suggest always surrounding arguments with the double quotes to ensure more consistent execution.

|  |  |
| --- | --- |
| **SourcePath** | The path to the folder that has the files to be copied |
| **DestinationPath** | The path to the folder that will have the file to be masked. If the file(s) have already been copied, then they will not be copied again (unless the file name is changed or is removed from the destination location. |
| **FieldsToMask** | A comma separated list of field names and masked lengths, which are bar (‘|’) separated:  Ex: fieldName1|5.  In the example above, ‘fieldName1’ is the name of the field and the ‘5’ is the number characters to be mask the values within.  If 0 (zero) is passed in for the length to mask, then the entire length of the values within the column will be masked.  If the length to mask is greater than the value(s) within the column then length of the value is masked. |
| **FilesToMaskSearchPattern** | A comma separated list of file name parts to be searched for to determine which files to mask and copy to a separate folder location. This means that first these files will be copied to the destination folder, then masked, and then finally this modified file is copied to the separate folder. So these file will have two copies – one in the destination and one in the separate folder. |
| **LoggingOptions** | This argument controls the logging to the screen and to a file. These switches are separated by a space.  The first switch can be “on” or “off”. This turns logging to the screen on or off.  The second switch can be “on” or “off”. This turns logging to a file on or off.  The log file is stored in the directory of where the app is executed from.  If the log file exceeds 1 MB (1000000 bytes) then it will be backed up to the same location with the file name: LogFile.<DateTime>.txt. All backup log file will be automatically removed once they have become older than 3 months old. |
| **ManualRun** | This arguments controls how the user interaction will be enabled from the console screen.  This switch can be “T” or “F”.    If “T” is set, then user interaction is required. The user will have to “press enter” to allow the process to continue to the next step.  This should be set to “F” when running as an automated process.  This switch is ignored if logging to the screen is turned off (see LoggingOptions above). I.e. If logging to the screen is turned off, and ManualRun is set to “T” then the user will not need to “press enter” to continue the process. |

**Example:**

**CopyAndMaskFiles** "\\ENADBSHRdv01\Dataload\SWHR\HRHIED\Archive\365" "\\filedepot\app\S-ETL\DEV\SWHR\HANA\_POC\_SWHR\_HEFiles" "Old Social Security Number|7,New Social Security Number|7,Person Last Name|0,Person Birth Date|0" "on on" "T"

* **“**[**\\ENADBSHRdv01\Dataload\SWHR\HRHIED\Archive\365**](file:///\\ENADBSHRdv01\Dataload\SWHR\HRHIED\Archive\365)**”:** This is the SourcePath. It has the files that need to be copy to the DestinationPath.
* **“**[**\\filedepot\app\S-ETL\DEV\SWHR\HANA\_POC\_SWHR\_HEFiles**](file:///\\filedepot\app\S-ETL\DEV\SWHR\HANA_POC_SWHR_HEFiles)**”:** This is the DestinationPath. It will have the files copied from the SourcePath.
* **"Old Social Security Number|7,New Social Security Number|7,Person Last Name|0,Person Birth Date|0":** This is the list of FieldsToMask. It includes the name of the column(s) to be mask and the length of characters to be mask. If the column name is found in any file it will be masked. It is **not** an option mask a certain column in one file, but not in another. All column with the given named will be mask, not matter the file. The length of characters to be mask needs to be a positive whole number. If the number is negative or decimal, the app with throw an error and stop (if logging to a file is turned on, then this error will stored in the log file).
* **".person.dat|.job.dat":** This argument specifies the files that need to be masked and copied to an additional folder. The folder these files will be copied to will be a subfolder of the DestinationPath and will be named “Person&JobSubsets”
* **"on on":** This is the two switches that control logging. The first on is for the screen and the second is for a file. Valid entries are: "on on", "on off", "off on", or "off off". Any other specified value will throw an error and the app will stop (if logging to a file is turned on, then this error will stored in the log file).
* **"T":** This turns the ManualRun on. If this is set to “T” then after each step the app executes the user must press enter for the execution to continue. Do not release the app with this value set if you expect the app to run autonomously.

**Notes about logging to the screen:**

To have the app log to the screen, you must set the LoggingOptions “on xx”. If you would like to control the pace of the execution of the app then make sure the LogginOptions for the screen is turned on and ManualRun is set to “T”. With these options set the app will not continue to the next step until you press enter. This can allow you to review the action the app just took to validate the data or state of the app.

While displaying the process to the screen, you may notice the text in certain colors. Here is the definition of the colors used within the app:

Red: Something has gone wrong. An error has been thrown, and most likely the app will stop executing. In this case the error will be saved to a log file, if the log file option is turned on the LoggingOptions.

Yellow: This is simply a warning that something isn’t quite right, but the app can continue to execute. You may want to review the details of the log file if you see a warning.

Green: Means that action was completed successfully.

White: Normal activity