



253 36th Street  
Suite C309 #22  
Brooklyn, NY 11232  
<http://weareavp.com>  
917.475.9630  
[info@weareavp.com](mailto:info@weareavp.com)

# Fixity User Guide

## Version 1.0

**2018-03-07**

### Contact information

AVP

<http://www.weareavp.com/>

### GitHub repository

<https://github.com/WeAreAVP/fixity>

### Download pages for application

**Windows:** <https://www.weareavp.com/wp-content/uploads/2018/03/Fixity-Win.zip>

sha256 (.exe): d58119c02f1eb13b5ea740048d7fe21437ff9a65938f29e9916050ca74f5d0d2

**Mac:** <https://www.weareavp.com/wp-content/uploads/2018/03/Fixity-macOS.zip>

sha256 (.dmg): db695a5cc64e46d3eb8557c0c1e67b1b2ec675e0b1fc3fa7426138d8b0040c6e

### Fixity Google Group/Email List

<https://groups.google.com/d/forum/fixity-users>

### Feature Requests and Bug Reports

We appreciate your feedback! Please report all feature requests and bug reports to

[fixity@weareavp.com](mailto:fixity@weareavp.com)

### Change log

#### 2018-03-07 Version 1.0

- Addressed major compatibility issues with Mac Sierra (OS 10.12) and Java 1.8 which caused a variety of issues.
- Coding language changed from Python to Java.
- Added unlimited number of directories for scanning.

- Added email user authentication with Google API.
- Added “Disable Schedule” functionality.
- Force-switch algorithm bugs were fixed.
- Added custom selection of location for history and reports folders.
- Creation of history and reports folder now takes place upon initial launch.
- Password encryption bug fixed.

### **2015-03-05 Version 0.5**

- Create tasks as Administrator (Windows using Run as Admin).
- Notification window occurs when a non-admin tries to change the schedule of a project originally created by an admin. This notifies the user that the schedule cannot be changed by a non-admin. (Windows)
- Improved visual feedback of progress during manual scans (Windows/Mac).
- Launch Run process as a New thread (Windows/Mac).
- Fixed Fixity not responding/crashing issue when scanning very large directories (Windows).
- Improved labeling for scheduling section. (Windows/Mac).
- Disabled closing of console window prior to completion of scanning (Windows).
- Improved reporting (Windows/Mac)
  - Added time elapsed
  - Added algorithm used
- Greyed out Recipient Email Addresses when email settings are not filled out
- Added mouse-over functionality to the directory path; the full file path is now displayed as a tooltip.
- Email settings now allow blank passwords to accommodate internal SMTP gateways.

### **2014-07-01 Version 0.4**

- Created OSX version (64-bit systems only)
- Added MD5 checksum algorithm option
- Added ability to change checksum algorithm for a project
- Added ability to import projects between versions of Fixity
- Added checksum and filter information to manifest (.tsv files in history directory)
- Added ability to change project name
- Added ability to update the location of a scanned directory
- Added ignore hidden files option in Filter Files dialog
- Added ability to log errors and events
- Implemented SQLite database for storing projects
- Updated logic for reporting on copied and moved files
- Added ability to delete a directory from a project
- Added About Fixity
- Improved handling of special characters in file paths
- Added history directory to store snapshots of the manifest

- Changed project file extension from .csv to .tsv to indicate that the data uses tabs as the separator
- Changed report names and the subject line of emailed reports to include the project name
- Code and interface improvements

### **2014-01-13 Version 0.3**

- Changed Save Email/Time to Save Settings in File Menu. Selecting this saves all project settings.
- Changed Save and Run to Run Now and altered the way this invokes the scheduler so that it doesn't tie up the Fixity UI.
- Added Preferences > Configure Sender Email to enable ability to configure Sender Email instead of using fixityreport@WeAreAVP.com as the fixed address. This is a global preference for the application. The email and password are stored using a cryptographic hash to maintain security.
- Added Preferences > Filter Files to enable users to exclude the scanning and validation of files. This is a project level preference.
- Added Run when on battery power checkbox preference to choose whether the scanning and validation tasks should occur when the computer is on battery power/not plugged in. This is a Windows Task Scheduler preference that caused a lot of confusion previously with scheduled tasks 'failing' to run. This was made explicit and placed front and center to make it clearer. If unchecked the application will not run if running on a laptop that is not plugged in. This is a project level preference.
- Added If missed, run upon restart checkbox preference that invokes the scheduled task to run when started if it was missed for some reason (e.g., turned off or logged out). If unchecked the application missed scheduled tasks will simply be missed. This is a project level preference.
- Added Email only upon warning or failure checkbox preference to leave it up to the user when to receive emailed reports. If unchecked emails will be sent to the identified recipient email addresses every time a scan is performed regardless of the results. This is a project level preference.
- Compiled as 32-bit application so that it will run on 32-bit and 64-bit systems.
- Altered behavior so that saving does not wipe the project manifest
- Updated the reporting terms:
  - Created was changed to New
  - Moved was changed to Moved or Renamed
  - Missing was changed to Removed
  - Corrupted was changed to Changed
  - Confirmed is still Confirmed
- Altered behavior so that changed files are updated with their new information after being reported. Therefore a changed file will be reported as changed one time, and if no further changes occur it will not continually be reported as changed.
- Fixed bug that wouldn't allow adding new directories to an existing project.

- Fixed bug that maintained directories in slots 5, 6 and 7 when existent and creating a new project.
- A display issue with name of application in Title bar of application window was corrected.

### **2013-11-22 Version 0.2**

- Addressed bug with scheduling issue — task was not generated properly if the file path had a directory with spaces in it.
- Addressed bug with emailing of reports — corrected SMTP configuration
- Changed the code for images so that they display properly
- Moved AutoFixity.exe into schedule folder to avoid confusion
- Other minor code refinements
- Updated User Guide according to user feedback

### **Known issues**

- If a scan is scheduled and does not have access to the project directories (i.e., is not logged on to a server or a removable drive is unavailable) it will report the files as missing. When run again with access to the project directories it will report the files as new.
- When importing projects between operating systems the file-path syntax is different and will result in reporting as if files having been removed because the application is unable to reach the files with the incorrect syntax. If the file path is corrected after import but before running the scan it will work correctly.
- When importing an .fxy project file the checksum algorithm is not detected. You must set this correctly manually prior to running a scan or the application will report all files as changed. This does not apply to .tsv files.
- Certain CIFS network implementations may experience errors in Fixity's reporting.
- Networks utilizing CIFS will generate new inodes upon each connection to the network, throwing off Fixity's logic and causing unreliable reporting.
- It is currently not possible to skip the initial window to select destination directory for history and reports folders. Once these are set they cannot be changed through the user interface. As a temporary solution users can reinstall the application.
- Users running Windows 7 have experienced problems running Fixity, such as installation issues and window displays not functioning correctly.

# Table of Contents

<b>Fixity User Guide</b>	<b>1</b>
Table of Contents	5
Brief overview and background	6
Fixity folder structure setup	7
Launching Fixity	10
Windows Users	10
Mac Users	11
Fixity interface	12
Projects	12
Scheduling	12
Directories	14
Recipient Email Addresses	15
Fixity File menu	16
New Project	16
Save Settings	16
Run Now	16
Delete Project	17
Change Project Name	17
About Fixity	17
Quit Fixity	17
Fixity Preferences menu	18
Filter Files	18
Turn Debugging On/Off	19
Import Project	19
Configure Sender Email	20
Select Checksum Algorithm	22
Fixity analysis and reporting	23
Fixity report	23
Analysis and reporting logic	24
Fixity reporting and manifest updating behavior	26
History snapshots	26
Frequently Asked Questions (FAQs)	28

# Brief overview and background

Fixity is a utility for the documentation and regular review of stored files. Fixity scans a folder or directory, creating a manifest of the files, including their file paths and their checksums, against which a regular comparative analysis can be run. Fixity monitors file integrity through generation and validation of checksums, and file attendance through monitoring and reporting on new, missing, moved, and renamed files.

AVP created Fixity after coming to the realization that:

1. Checksum generation and validation are universally recognized as a primary mechanism for fulfilling the goal of fixity.
2. Generating checksums without systematically and routinely validating them is of little or no use.
3. Most organizations that “use checksums” only generate checksums but do not validate them routinely, if ever.
4. The primary reason organizations do not systematically and routinely verify checksums is that there is no tool that allows simple scheduling and reporting of validation.
5. Organizations that do routinely verify checksums do not typically monitor and report on information related to file attendance.
6. Overcoming the lack of resources and/or access to IT expertise is a critical factor in enabling organizations to fulfill the goal of fixity. Organizations need a simple, low cost tool.

Fixity was created with the sole focus of fulfilling the requirements of those concerned with monitoring and managing the fixity of a collection of files over the long term. There are many free and open-source checksum utilities out on the market, but they do not offer the feature set or simplicity necessary to fulfill the needs of many organizations. It is intended for use in monitoring collections of files that are “final state,” or ready for deposit into an archive or preservation-oriented repository.

# Fixity folder structure setup

When you download and unzip Fixity, you will find the executable file and this user guide.

When running Fixity for the first time, the application will ask the user to set a location for the reports and history folders (See Figure 1). These folders can be placed anywhere on your computer and will later contain the reports generated by the application, specifically:

**history:** the directory where a snapshot of the project manifest is stored each time a project is run. See section on [History Snapshots](#) for more details.

**reports:** the directory where reports are stored. See section on [Fixity Reports](#) for more details.



Figure 1: Select folder path for History and Reports folders.

Additionally, after running Fixity for the first time in Windows, the application will create the following additional elements:

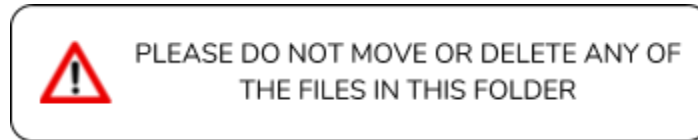
**FixityDB:** contains the database file and configuration settings for checksum and filter settings.

**debug:** contains the log reports generated when debugging is turned on. Note: the debug file in Mac can be found in the Package Contents (See Turn Debugging On/Off).

**derby:** contains a log of database activity.

**conf:** contains configuration information captured as a result of setting preferences.

**schedules:** Stores the scheduling information necessary to update Windows Task Scheduler. This folder will appear after running the first scan.



Name	Date
Fixity	3/1/2
derby	3/2/2
debug	3/2/2
conf	3/2/2
FixityDB	3/2/2

Figure 2: Fixity directory structure for Windows after running it for the first time.

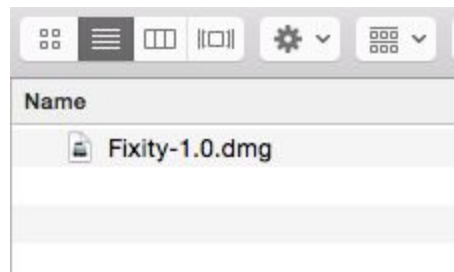


Figure 3: Fixity directory structure for Mac.





## NOTES

Moving the executable file —after setting up the folders' paths— will result in the following:

Mac: Links to report folders and application settings will remain unchanged. This means that even if the executable file is relocated Fixity will retain the locations and the application will run without issues.

Windows: All previously set parameters will be lost and folder path for reports folders will be reset.

If you move/delete the reports and/or history folders, Fixity will create new ones in the same initial location when performing a new scan.

# Launching Fixity

## Windows Users

To open Fixity in a Windows environment, you can double click the Fixity application file. However, if you want Fixity to be able to run scans when you are logged off (computer is on, but no user is logged in), you will need to right click on the Fixity.exe file. Select Run as administrator, as shown in the figure below.

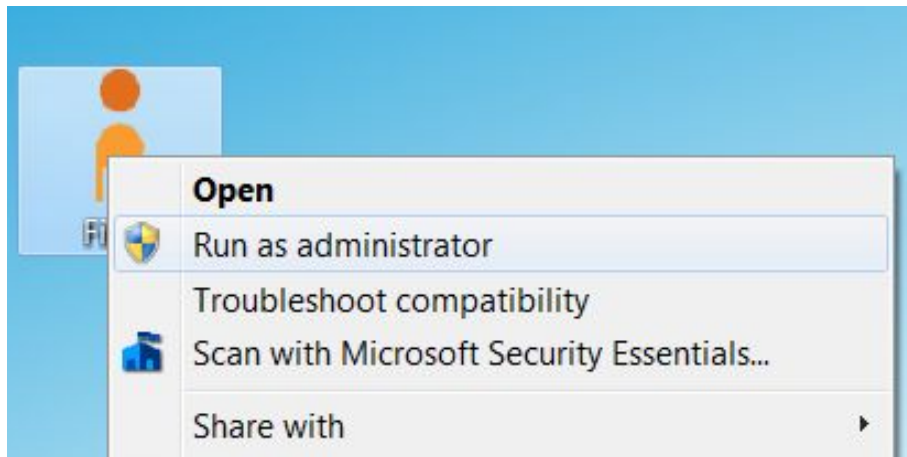


Figure 4: Right-click options in a Windows environment.



### NOTE

Fixity's ability to access network resources when logged off is dependent on settings within a given environment. If Fixity does not have access to network resources that it has documented previously, it will report them as removed.

## Mac Users

To install the program, Mac users can double click on the .dmg file. Drag and drop the Fixity application into the Applications folder, as shown in Figure 5 below. You can run the program by either going to the Applications folder and double-clicking on the app icon, or by going to the Launchpad.

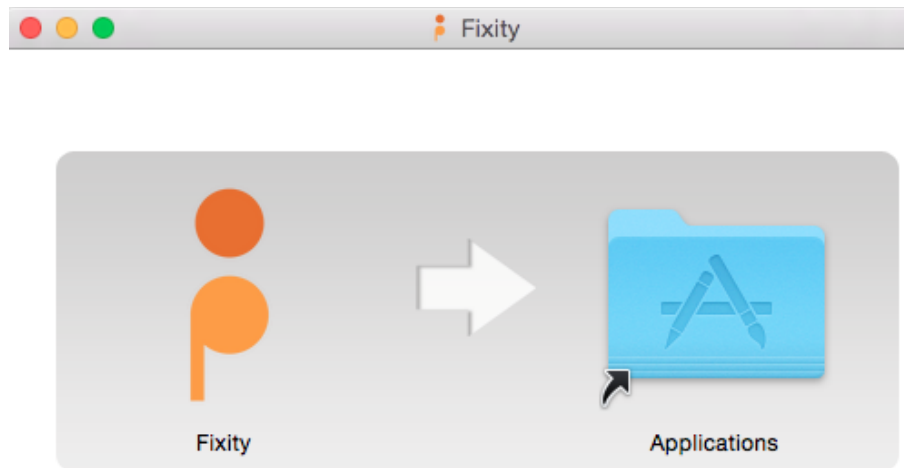


Figure 5: Fixity installation on Mac. Drag and drop Fixity into the Applications folder.

There is no option for running while logged off of your user account in your computer. Additionally, scheduled scans will run only if you are logged in.

# Fixity interface

Fixity has a graphical user interface (GUI). The following section details the behaviors of each section of the interface from left to right.

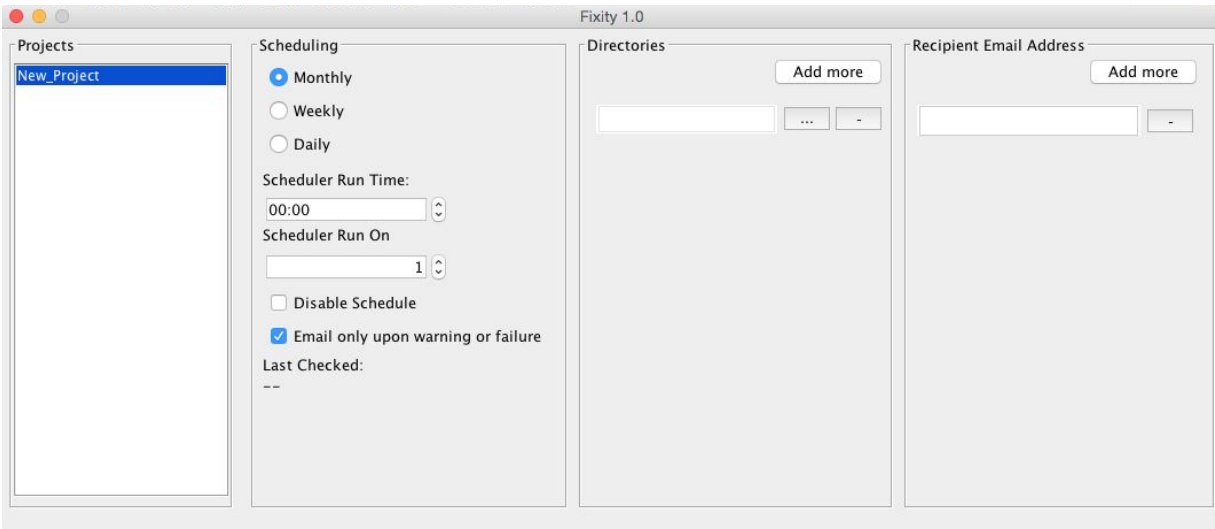


Figure 6: Screenshot of Fixity version 1.0 interface for Mac OS. Note that all options will appear greyed out until you create a project.

## Projects

As you create and save new projects, they will appear in this box. Click on a project name to view its settings in the rest of the interface.

## Scheduling

You can set the schedule for generating/validating checksums and file attendance in this section. Using the radio buttons, you can choose between a monthly, weekly, or daily scan.

- **Monthly:** If you choose this option, you will be presented with two boxes. The box containing “00:00” is where you enter the time (in military time and in reference to the date and time settings of your computer) that you want the validation to begin. The second box scrolls through numbers from 1 to 31, allowing you to choose the day of the month on which you wish the scan to run.
- **Weekly:** If you choose this option, you will be presented with two boxes. The box containing “00:00” is where you enter the time (in military time and in reference to the date and time settings of your computer) that you want the validation to begin. The

second box allows you to choose the day of the week on which you wish the scan to run.

- **Daily:** If you choose this option, you will be presented with one box. This box is where you enter the time (in military time and in reference to the date and time settings of your computer) that you want the scan to begin each day.

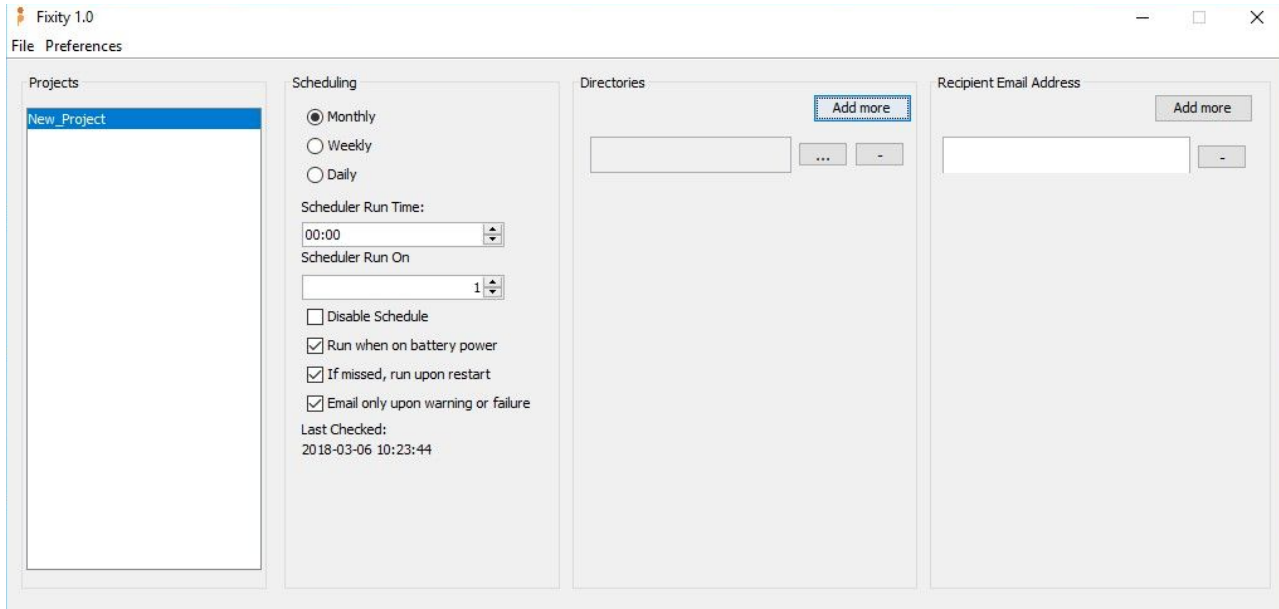


Figure 7: Screenshot of Fixity version 1.0 interface for Windows. Note that all options will appear greyed out until you create a project.

The Scheduling section also contains settings for the behavior of the scheduled task.

- **Run when on battery power:** (Windows only) If Fixity is being run on a laptop, check this option to allow Fixity to run when operating on battery power. Note that this option does not affect non-battery powered systems. On a Mac, Fixity will always run regardless of whether the computer is operating on battery power or not.
- **If missed, run upon restart:** (Windows only) If this is selected, Fixity will run upon startup if a scheduled scan is missed (e.g., due to the system being powered off). If this is not selected, Fixity will run at the next scheduled time. On a Mac, Fixity will not run a scheduled scan if it was missed. It will run at the next scheduled time. Note that Fixity will not run immediately upon startup — please wait 15-20 minutes to allow for it to run.
- **Disable Schedule:** when this option is checked, Fixity will ignore set schedule for this project, no scans will be performed automatically. The user can run a manual check if desired.
- **Email only upon warning or failure:** If selected, Fixity will only send email reports when a change is detected during a scan. Otherwise, it will send an email following each and every completed scan. See Fixity File Report section below for more details.

Lastly in the Scheduling column, Fixity displays the date and time of your project's most recent scan.

## Directories



Click on this button next to the text field to select the specific directory that you wish to scan. You can select unlimited directories per project. Mousing over the file path in the Fixity 1.0 interface will cause the full file path to display as a tooltip.

If you need to change the location or name of one of the directories that is included in your Fixity project (for instance, you move the directory from one drive to another drive of a different name, or you need to rename the directory), you can update the Fixity project accordingly without negatively impacting your monitoring and reporting. You can do this by selecting the file path that you wish to change and selecting the new location for the directory. When selecting "Save Settings," you will be prompted with the window shown in Figure 8. Selecting "Change Path" will update the Fixity project with the new path and treat all files within the selected parent directory in the same way it would have if nothing had changed.



Be careful when changing a directory to a directory containing different files than the first. As Fixity will look for the files from the first directory, it will report that all prior files were removed and all new files were added

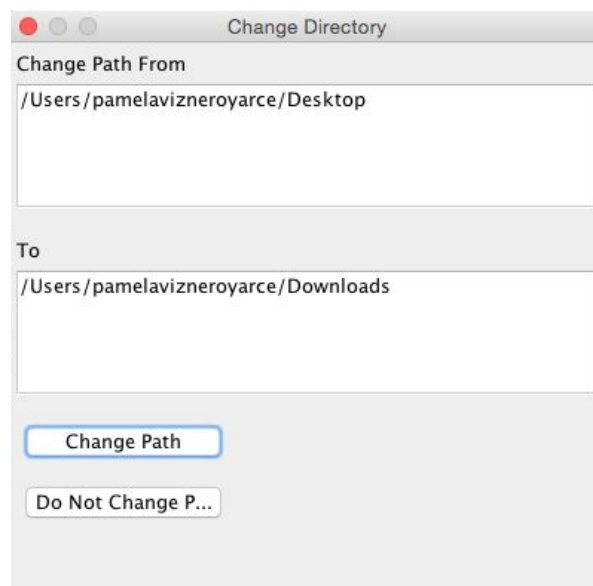


Figure 8: Dialog to update the file path for a directory.

By clicking on “Add More,” you can add more directories to the project to be scanned.



This button will remove a directory from a project. Once this is done, Fixity will no longer scan it as part of a project, and its file information will be removed from the Database.

## Recipient Email Addresses

Every time Fixity scans a directory, a report is generated. In each of your projects, you can input the email addresses (unlimited, one per text box) to have this report sent to based on the preference “Email only upon warning or failure.” Each email address entered will receive the same report for all directories selected as part of the project. These email addresses can be changed at any time. By clicking on “Add More,” you can add more email addresses to the project. The [-] button will remove an email address from a project. To be able to send emails, you first have to configure the Email Settings (See Configure Sender Email)

# Fixity File menu

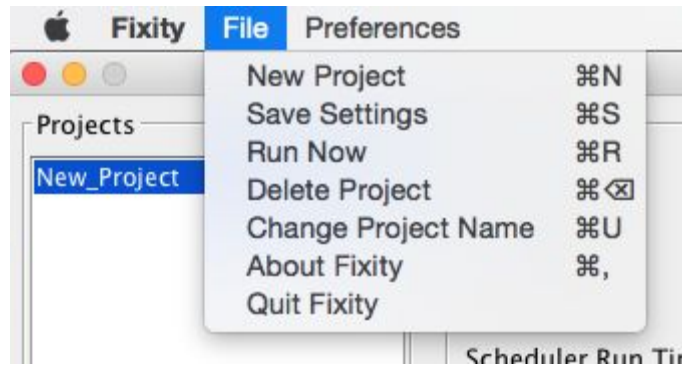


Figure 9: Detail of Fixity File menu (Mac OS).

## New Project

Choose this option each time you wish to set up a new project. A project contains the settings for the scheduling, directories, and recipient email address boxes. You will be prompted to name your project. Project names cannot contain spaces or characters disallowed in filenames by the operating system (such as “ and <).

## Save Settings

When you are satisfied with your project's parameters, click “Save Settings” to save it to the project folder. The project file contains the settings for the scheduling, directories, and recipient email address boxes. Note that unlike previous versions, saving settings now does not invoke a fixity check scan at the time of saving.

## Run Now

“Run Now” saves the current project settings and invokes a fixity scan. If it is the first time a scan is being run for a project it will perform the initial scan. If a scan has already been performed it will validate against the existing data.



If Fixity is closed during a Run Now scan, it will not complete the scan. Please also note that Fixity will not allow for multiple scans to be run at the same time.



## Delete Project

Choose this option if you wish to completely delete a project.

## Change Project Name

Select this to rename a project.

## About Fixity

Opens the “About Fixity” dialog window containing a description of the application, licensing information, and contact information.

## Quit Fixity

Closes the application.

# Fixity Preferences menu

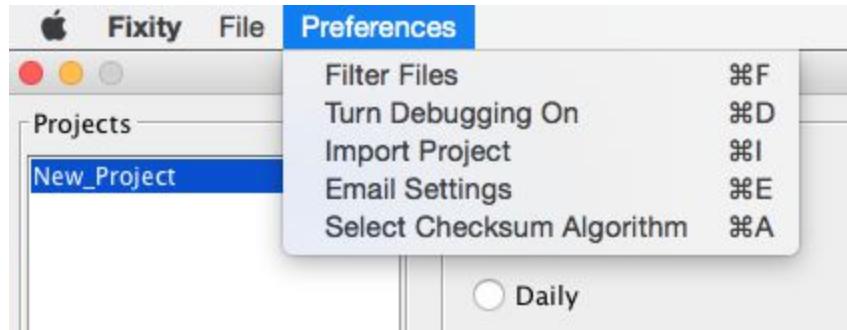


Figure 10: Detail of Fixity Preferences menu (Mac OS).

## Filter Files

The Filter File window (Figure 11) allows the user to create pattern-matching filters to prevent Fixity from scanning certain files. For example, a repository may not want to include in their scans the DS\_Store or Thumbs.db files created by default by certain operating systems. These can be blocked from scans by adding them to the filter.

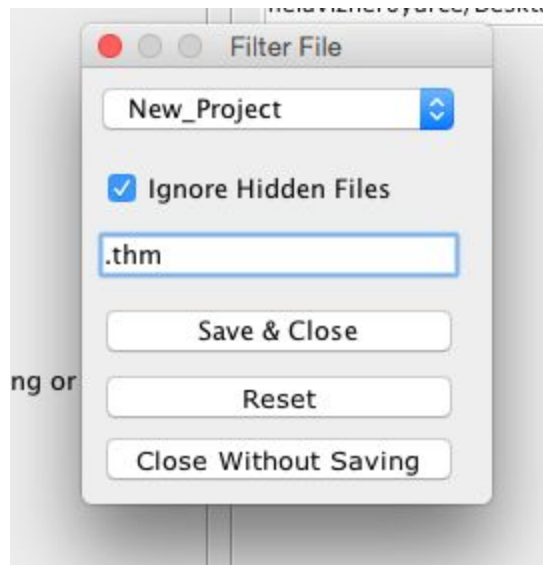


Figure 11: Fixity Filter File detail with "Ignore Hidden Files" selected and sample file format pattern to block.

The filter interface contains a drop-down menu of projects, from which the user can select a project before setting filters. A project must be saved before it will appear in the dropdown list.

In the text box, a comma-separated list of case-sensitive patterns to block can be set. For example, the settings in Figure 11 will not scan hidden files or files containing .thm. When you are satisfied with the filters, click “Save & Close.”

## Turn Debugging On/Off

With debugging on, Fixity generates log files in order to assist in troubleshooting issues. Turning debugging off disables logging. In Windows, the debug file can be found in the same folder as the executable file. In Mac, the debug file can be found by right-clicking on the executable file and selecting “Show Package Contents>Java.”

## Import Project

This option allows for the importing of project information. Import Project will accept .fxy files from older versions of Fixity and .tsv files generated by newer versions of Fixity in the history directory. Importing a .fxy or .tsv file into Fixity imports the following information:

- Directories
- Recipient email addresses
- File paths
- Checksum values
- Filesystem index location
- Scheduling information
- Checksum algorithm type (.tsv files only)
  - For .fxy files, this must be manually configured. If the wrong algorithm is used, Fixity will not be able to confirm any files.
- Filter file information (.tsv files only)
  - For .fxy files, this must be manually configured for each project.

Importing a project will not import:

- Sender Email Configuration: This is a preference that is set at the global level and is not part of the project import information.

Importing projects can also be used for reverting back to a previous snapshot by importing an older .tsv file from the history folder.



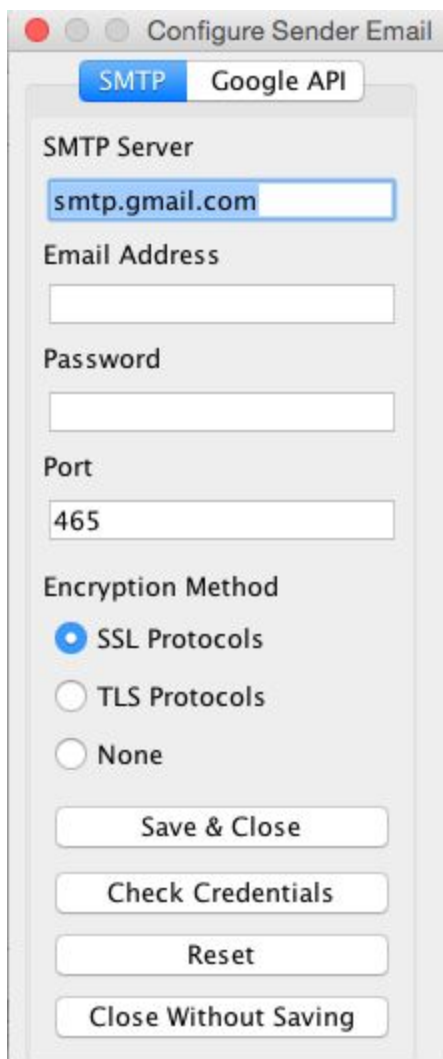
Due to how the filesystem structures of different operating systems track files, the index location and file path for each file will change when importing projects between a Mac OS and Windows. File paths will need to be updated when moving between operating systems, and all files will be reported as being New Files during the first scan.

## Configure Sender Email

Users can configure email settings either manually (SMTP) or via the Google API (only for Google users). Without setting this preference, Fixity will not be able to send report emails. However, running scans is possible without setting a sender email. Note that Fixity will encrypt the credentials before storing them in order to provide account security.

### SMTP Configuration

Figure 12 demonstrates how Fixity allows you to configure the email settings associated with the account used for emailing reports.



The screenshot shows a window titled "Configure Sender Email" with two tabs: "SMTP" (selected) and "Google API". The "SMTP" tab contains the following fields and options:

- SMTP Server:** A text field containing "smtp.gmail.com".
- Email Address:** An empty text field.
- Password:** An empty text field.
- Port:** A text field containing "465".
- Encryption Method:** Three radio button options: "SSL Protocols" (selected), "TLS Protocols", and "None".
- Buttons:** "Save & Close", "Check Credentials", "Reset", and "Close Without Saving".

By default, Fixity populates the server, port, and security fields with Gmail default settings. If these values are different for your email account (for example, if your email account is provided through your organization), you will need to provide the appropriate information. This is the same information you would find in your email application configuration settings, so you may be able to obtain the information by looking at your email application account settings. If not, the information needed can be obtained from your email provider or by contacting your IT administrator.

The email account can be tested with "Check Credentials," which will send an email to the account provided. If the email is received, then Fixity is able to successfully use the email address. Be sure to check your spam folder if the email is not received in your inbox before performing troubleshooting.

Select "Save & Close" to save changes.

Figure 12: SMTP Email Settings window

## Google API Configuration

Figure 13 shows the settings window to connect with your Google account via the Google API. To connect Fixity to your Gmail account, simply click on “Connect with Google API.” This will open a new browser window where you can authorize the operation. The email address of the sender will show up in Fixity. Once authorized, you must check the “Send Email Using Google API” checkbox. You can “Send Test Email” to check settings.

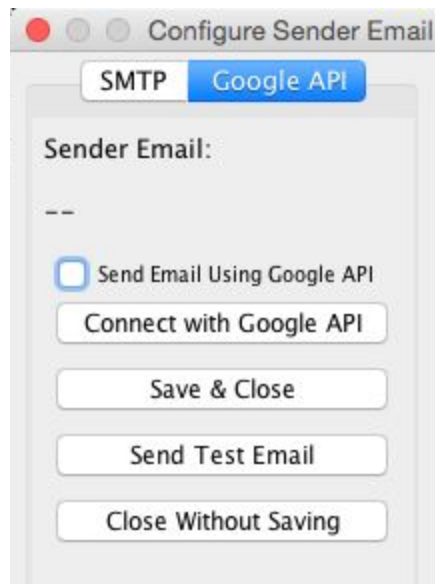


Figure 13: Google API Email Settings window



### NOTES

- The sender email settings are global for all projects in Fixity. It only needs to be set once and will work on all projects in which recipient email addresses have been filled in.
- If you're using Google and you are connecting to your account via SMTP, It is possible that Fixity gets blocked for security reasons. Make sure you add Fixity to the list of third-party apps that can access your account. More information here:  
<https://support.google.com/accounts/answer/3466521?hl=en>

## Select Checksum Algorithm

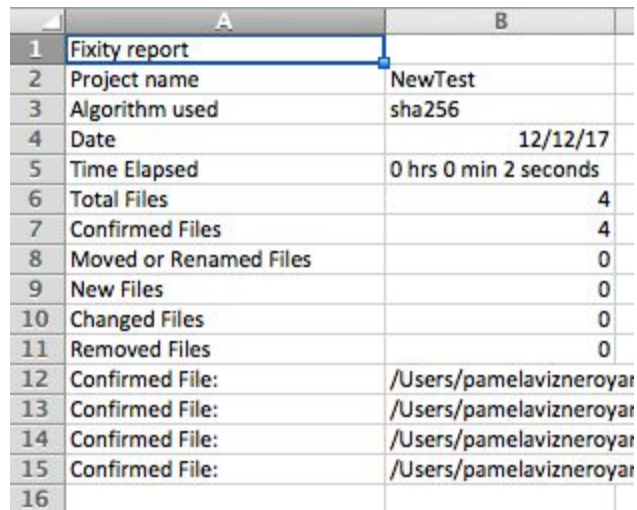
Fixity defaults to using the SHA256 algorithm to generate its checksums. This setting allows you to select either the SHA256 or MD5 algorithm to calculate checksums for a given project. If the algorithm is changed after checksums have already been generated for a project, Fixity will do the following:

1. Perform a Fixity scan and generate a report using the original algorithm.
2. Confirmation:
  - a. If all files are confirmed, another scan will be performed immediately generating checksums using the new algorithm, and generating a report. Future scans will use the new algorithm.
  - b. If all files are not confirmed, a report will be generated and the process will be halted after the initial scan. A prompt will be generated in the application and an email will be sent to recipient email addresses with notification of the failure and a full report. The checksum algorithm for the project will not be changed if all files are not confirmed. However, the user can perform the operation again (i.e. retry to change the algorithm) to force Fixity to complete the process, in which case a new report will be generated and the algorithm will be changed.

# Fixity analysis and reporting

## Fixity report

A Fixity report is saved to the reports folder and emailed to the recipient email addresses identified in the project according to the “Email Only Upon Warning or Failure” preference. The Fixity report is named `fixity_yyyy-mm-dd-hhmmss_projectname.tsv` where `hhmmss` is in military time.



	A	B
1	Fixity report	
2	Project name	NewTest
3	Algorithm used	sha256
4	Date	12/12/17
5	Time Elapsed	0 hrs 0 min 2 seconds
6	Total Files	4
7	Confirmed Files	4
8	Moved or Renamed Files	0
9	New Files	0
10	Changed Files	0
11	Removed Files	0
12	Confirmed File:	/Users/pamelavizneroyar
13	Confirmed File:	/Users/pamelavizneroyar
14	Confirmed File:	/Users/pamelavizneroyar
15	Confirmed File:	/Users/pamelavizneroyar
16		

Figure 14: Detail of a sample Fixity report.

When opening Fixity reports in spreadsheet applications, it is important to open them by importing the `.tsv` file, identifying the delimiter as a Tab, and treating all data as text (see FAQ for details). This will ensure maximum functionality and keep spreadsheet applications from altering the data in unintended ways. Fixity reports contain the following information:

**Fixity report:** Identifies the report as a Fixity report.

**Project name:** Provides the name of the project for which the report was generated

**Algorithm used:** Identifies the hash algorithm used to compute the checksums for the files scanned.

**Date:** The date that the report was generated.

**Total Files:** Total files scanned and/or removed. It is the sum of all of the below categories in any single scan.

**Confirmed Files:** A file is confirmed if nothing has changed since the last scan. In Figure 14, the number of files scanned that exist in the manifest where the file path, hash value, and filesystem index location are the same as they were in the last scan.

**Moved or Renamed Files:** A file has been renamed and/or moved to another location being scanned within the same Fixity project. When files are moved or renamed they will usually have a different file path but retain the same hash value and index location. If a file is copied and pasted to a new location and the original one is deleted this will result in the hash value remaining the same but the filesystem index location and file path changing, resulting in a report of one new file and one removed file.

**New Files:** The number of files detected by Fixity that do not currently exist in its manifest. If a file is found that does not have a matching hash, filesystem index location, or file path, it is listed as a new file. Files are also listed as new if they share a hash value with a previously scanned file, but have a new filesystem index location and file path.

**Changed Files:** The number of files that have a different hash value since the last scan.

**Removed Files:** The number of files that existed in the manifest previously, which are no longer present.

Each row beneath “Removed Files” provides a file path and the associated status of the identified file for all files scanned and/or removed in a project. Files that have been filtered out using the “Filter File” preference will not appear because they will not have been scanned.

## Analysis and reporting logic

Figure 15 aims to support users in interpreting reporting by presenting various scenarios and results in an easy-to-read table with Y[es] or N[o] indicators and the outcomes.

Figure 15 denotes:

- If a file that is present in the manifest is present in a target directory,
- If a file has the same hash value, file path, and filesystem index location upon comparison, and
- How Fixity reports each scenario.



File present?	Same hash?	Same file path?	Same index location?	Result
Y	N	Y	Y	Changed File
Y	N	N	Y	Changed File
Y	N	Y	N	Changed File
Y	Y	Y	Y	Confirmed File
Y	Y	Y	N	Confirmed File
Y	Y	N	Y	Moved or Renamed File
Y	None	None	None	New File
Y	Y	N/None	N/None	New File
N	Existing	Existing	Existing	Removed File

Figure 15: Fixity reporting chart.



#### NOTES

- If a file is moved to a new directory, it will have the same index location value and same hash value, but a new path. This will result in reporting that the file has been Moved or Renamed. Some versions of some operating systems will perform a copy operation even when moving files, producing the same results as if a file was copied and pasted (described below).
- Due to how Fixity tracks files, there are no conflicts or confusion when two files share a checksum value. It will properly track them as two distinct objects.
- If a file is copied to a new location and the original file is deleted (effectively moving a file but using different operations), this will result in the "moved" file having a different index location value but the same hash value. This will result in reporting that there is one new file and one removed file (two actions for the original file).



## MORE NOTES

- If a file is renamed, it will have the same index location and hash value, and a new path. As such, Fixity will report that it was Moved or Renamed.
- If a file is copied to a new location it will have a different index location value and path but the same hash value. This will result in reporting that there is one New File and one Confirmed File (the original file).
- Most files maintain their index location value when they are edited. Certain applications (such as many text editors) will instead edit a temporary version of a file and copy that over the original upon saving. This will change the checksum and the index location, which will report correctly as a changed file. However, if this were coupled with a rename or move it would result in a report of 1 removed file and 1 new file.

## Fixity reporting and manifest updating behavior

When changes are identified, Fixity behaves in the following ways upon finding changes and after reporting them:

- **New files** are added to the manifest and checksums are generated for them
- **Removed files** are removed from the manifest
- **Moved files** are updated with their new location in the manifest
- **Renamed files** are updated with their new name in the manifest
- **Changed files** are updated with new hashes in the manifest

## History snapshots

Each time a project is run, a snapshot of the manifest data is saved to the history folder. This contains the hash value, file path, and index location value for every file scanned. The snapshot is created after the project is run. This provides an audit trail over time and the ability to revert to a previous snapshot using the Import Project function. A Fixity report represents the differences/similarities between two history snapshots. The file is named `projectname_yyyy-mm-dd-hhmmss.tsv` where `hhmmss` is in military time.

When opening history snapshots in spreadsheet applications, it is important to open them by importing the `.tsv` file and identifying the delimiter as a Tab and treating all data as text (see

FAQ for details). This will ensure maximum functionality and keep spreadsheet applications from altering the data in unintended ways.

1	/Users/pamelavizneroyarce/Documents/All documents/Test;	
2		
3	99 09:00:00 99 1	
4		12/12/17 14:19
5	-  0	
6	sha256	
7	d6dfef6f4294f1f63447b01ffbe71775684905824a2c147a7984594ac0378e87	/Users/pam
8	1d4a940b5f529a289177ef830ec7ec0b2e658c12074a6efc1420ee745abc6f33	/Users/pam
9	d65165279105ca6773180500688df4bdc69a2c7b771752f0a46ef120b7fd8ec3	/Users/pam
10	20e3ee7098014b91f20bddcd53bc39d5f9cfa9d3036336847b3867542f331866	/Users/pam
11		

Figure 16: Detail of a sample history snapshot

History snapshots contain the following information:

- Directories scanned:** The directories selected for scanning for a given project.
- Email addresses:** The recipient email addresses for a given project.
- Schedule information:** Codes identifying project schedule settings.
- Date and time of run:** The date that the project was run and the file was created.
- Filter Files Preference:** Any preferences that have been saved for the project regarding filtering of files.
- Checksum Algorithm Preference:** The checksum algorithm preference last saved for the project.

Each line following the checksum algorithm contains the following columns:

- Hash Value:** The checksum value for each file scanned.
- file path:** The file path for each file scanned.
- Filesystem Index Location:** The location in the filesystem index for each file scanned.

# Frequently Asked Questions (FAQs)

**Q: Can I request new features to add to Fixity?**

A: Yes, please send requests to [fixity@WeAreAVP.com](mailto:fixity@WeAreAVP.com) and let us know what you would like to do. We will add it to our backlog of issues. The more requests we get for the same feature the more likely we are to prioritize it. If you really want us to develop a specific feature you can also commission us to do the work.

**Q: Will Fixity perform a scheduled scan if the application is not open?**

A: Yes.

**Q: Will Fixity perform a scheduled scan if the computer is turned off?**

A: No. On Windows, if the project is set to scan on restart it will attempt to perform the scan 15-20 minutes after restarting. On a Mac, and if Run Upon Restart is not selected on Windows, an automated scan will not occur until the next scheduled time.

**Q: What is a .tsv file?**

A: The .tsv file found in the reports and history directories are text files that use tab as a delimiter for structured data.

**Q: What is the best way to open a .tsv file for manual review?**

A: The .tsv file can be opened with any text editor. However, for simple viewing of the data in the .tsv files using a spreadsheet application will be the easiest. The best way to open a .tsv (or .csv file for that matter) is by importing, instead of opening.

Instructions on how to do this can be found here:

[https://www.weareavp.com/wp-content/uploads/2017/08/Spreadsheet\\_Workshop\\_Materials.zip](https://www.weareavp.com/wp-content/uploads/2017/08/Spreadsheet_Workshop_Materials.zip)

**Q: How is scheduling performed within Windows?**

A: Fixity uses Windows' SCHEDULE\_TASKS tasks with AutoFixity to set/update/delete scheduled tasks. As such, its behavior is best explored via the SCHEDULE\_TASKS documentation.

**Q: How is scheduling performed within a Mac OS?**

A: Fixity uses the Mac OS's launchd to run Fixity at scheduled intervals. As such, its behavior is best explored via the launchd documentation.

**Q: Will Fixity for Mac run on a 32-bit system?**

A: No, only 64-bit systems. Fixity will run on 32-bit Windows.

**Q: Will Fixity for Windows run on a 32-bit system?**

A: Yes, Fixity will run on 32-bit Windows.

**Q: What happened to the .fxy files in previous versions of Fixity?**

A: In version 0.4, Fixity was updated to use a SQLite database instead of the .fxy file in combination with other files to serve the role of a database. However, users of Fixity were interested in accessing the data in the .fxy files. For this reason we created the history directory, where we output .tsv files that contain the same data that was in the .fxy files. In addition to making this data accessible, we decided to have this data output each time a project scan is run so that a history of snapshots is maintained for auditing and reversion purposes.

**Q: Where can I find out more about filesystem index locations?**

A: A free technical white paper on file system index locations, Fixity and File systems: Enhanced System Monitoring via inodes, is available at

<https://www.weareavp.com/fixity-and-file-systems-enhanced-system-monitoring-via-inodes-2/>

**Q: What file types does fixity scan?**

A: All files, including hidden files.

**Q: Can Fixity run on networked environments?**

A: Yes, Fixity will run on any files that can be mounted on your desktop.

**Q: How deep in a hierarchical directory structure will Fixity work?**

A: Fixity is recursive and can run on hierarchical directory structures as deep as the file system and networking protocols allow.

**Q: How many files can Fixity scan in one project?**

A: The number of files Fixity can scan in one project is only limited by network capabilities. As a point of reference, we scan an 18 TB NAS device - we break that scan into 10-12 projects.

**Q: How large can directories and files be?**

A: Fixity has been successfully tested on 10 TB directories and files that are 100s of GB. Any limitation will be based on file system, network, or bandwidth limitations.

**Q: Can fixity access LTO tape?**

A: Fixity can access files on LTO tape that utilizes LTFS.

**Q: Can scans run simultaneously?**

A: Yes, but multiple processes may impact performance/speed.