



## SISMO SFTP FILE UPLOADER

### License

This project is licensed under the MIT License – see the [LICENSE](#) file for details. You are free to use, modify, and distribute it, but it comes with no warranty and the authors take no liability for any damages.

---

### SftpClient Setup and Usage Guide (macOS)

This document provides all the necessary information to prepare, configure, and run SftpClient.jar on macOS. It also explains how to schedule the application to run daily at 05:00 AM and how to remove the scheduled task.

The SftpClient is a Java-based application that connects to an SFTP server and uploads it to a specified directory on the server.

---

### Prerequisites

#### 1. Java Runtime Environment (JRE):

- Ensure the jre folder is included in the same directory as the SftpClient.jar.

#### 2. Environment Variables:

- The run\_macos.sh script requires several environment variables to be set. These variables must be adjusted to match your environment before running the script.

#### 3. Required Folders:

- The following folders in the root directory of the application can be used as default to upload files on the sftp server::
    - user\_indicator: Used for storing user indicator files.
    - macro\_indicator: Used for storing macro indicator files.
    - portfolio: Used for storing portfolio files.
  - These folders are automatically created during the zip creation process.
  - You can create your own folders and set the environment variables accordingly in the run\_macos.bat.
- 

### Environment Variables

#### Examples and Explanations

##### *SFTP\_PORTFOLIO\_FILE\_MAPPER*

- **Purpose:** Maps external IDs to portfolio file names. This is used to rename files before uploading them to the SFTP server.
  - **Example:** PortfolioExternalId1-Portfolio1,PortfolioExternalId2-Portfolio2
    - PortfolioExternalId1: External ID of the first portfolio.
-



## SISMO SFTP FILE UPLOADER

- Portfolio1: Substring in the file name within the portfolio folder that matches the first portfolio.
- PortfolioExternalId2: External ID of the second portfolio.
- Portfolio2: Substring in the file name within the portfolio folder that matches the second portfolio.
- **Mock File Name:** Portfolio1\_Report.csv
  - The substring Portfolio1 in the file name matches the mapping PortfolioExternalId1-Portfolio1. The file is renamed to PortfolioExternalId1-[timestamp]-[operation].csv before being uploaded.

### *SFTP\_PORTFOLIO\_FILE\_MAPPER\_DEFAULT\_OPERATION*

- **Purpose:** Specifies the default operation for portfolio files. F means replace the file, and M means modify the file.
- **Example:** F
  - Files will be uploaded with the “replace” operation unless specified otherwise.

### *SFTP\_USER\_INDICATOR\_FILE\_MAPPER*

- **Purpose:** Maps external IDs to user indicator file names. This is used to rename files before uploading them to the SFTP server.
- **Example:** UserExternalId1-UserIndicator1,UserExternalId2-UserIndicator2
  - UserExternalId1: External ID of the first user indicator.
  - UserIndicator1: Substring in the file name within the user indicator folder that matches the first user indicator.
  - UserExternalId2: External ID of the second user indicator.
  - UserIndicator2: Substring in the file name within the user indicator folder that matches the second user indicator.
- **Mock File Name:** UserIndicator1\_Data.csv
  - The substring UserIndicator1 in the file name matches the mapping UserExternalId1-UserIndicator1. The file is renamed to UserExternalId1-[timestamp]-[operation].csv before being uploaded.

### *SFTP\_USER\_INDICATOR\_FILE\_MAPPER\_DEFAULT\_OPERATION*

- **Purpose:** Specifies the default operation for user indicator files. F means replace the file, and M means modify the file.
- **Example:** M
  - Files will be uploaded with the “modify” operation unless specified otherwise.

### *SFTP\_MACRO\_INDICATOR\_FILE\_MAPPER*

- **Purpose:** Maps external IDs to macro indicator file names. This is used to rename files before uploading them to the SFTP server.



## SISMO SFTP FILE UPLOADER

- **Example:** MacroExternalId1-MacroIndicator1,MacroExternalId2-MacroIndicator2
  - MacroExternalId1: External ID of the first macro indicator.
  - MacroIndicator1: Substring in the file name within the macro indicator folder that matches the first macro indicator.
  - MacroExternalId2: External ID of the second macro indicator.
  - MacroIndicator2: Substring in the file name within the macro indicator folder that matches the second macro indicator.
- **Mock File Name:** MacroIndicator1\_Stats.csv
  - The substring MacroIndicator1 in the file name matches the mapping MacroExternalId1-MacroIndicator1. The file is renamed to MacroExternalId1-[timestamp]-[operation].csv before being uploaded.

### *SFTP\_MACRO\_INDICATOR\_FILE\_MAPPER\_DEFAULT\_OPERATION*

- **Purpose:** Specifies the default operation for macro indicator files. F means replace the file, and M means modify the file.
- **Example:** F
  - Files will be uploaded with the “replace” operation unless specified otherwise.

---

### Notes for Users

- Ensure all environment variables are set correctly before running the application.
- The examples provided demonstrate how the variables are used to process, rename, and upload files to the SFTP server.
- If you encounter issues, check the log file (log.txt) in the application’s directory for details.

Ensure these variables are configured to match your environment in the ‘run\_macos.sh’ script before running it.

---

### Steps to Run SftpClient.jar Manually using Finder

1. **Make the File Executable:**
  - Open the terminal and navigate to the directory containing run\_macos.command.
  - Run the following command to ensure it is executable:  
`chmod +x run_macos.command`
2. **Run the File:**
  - Double-click the run\_macos.command file in Finder.
  - A terminal window will open, and the script will execute.



## SISMO SFTP FILE UPLOADER

### 3. Check Logs:

- The script writes logs to /tmp/sftpclient\_debug.log.
- Open the log file to verify the execution or debug any issues:

```
cat /tmp/sftpclient_debug.log
```

### Steps to Run SftpClient.jar Manually using Terminal

#### 1. Adjust Environment Variables:

- Open run\_macos.sh in a text editor.
- Update the environment variables with the correct values for your environment.

#### 2. Run the Script:

- Make the script executable:

```
chmod +x run_macos.sh
```

- Execute the script:

```
./run_macos.sh
```

#### 3. Verify Execution:

- The script will check for the presence of the Java runtime and execute SftpClient.jar.

---

### Schedule SftpClient.jar to Run Daily at 05:00 AM

#### 1. Prepare the Script:

- Ensure schedule\_macos.sh is in the same directory as run\_macos.sh.

#### 2. Run the Scheduler Script:

- Execute schedule\_macos.sh to create a launchd task:

```
./schedule_macos.sh
```

#### 3. Verify the Task:

- Check if the task is loaded:

```
launchctl list | grep com.user.sftpclient
```

---

### Remove the Scheduled Task

#### 1. Run the Removal Script:

- Execute remove\_schedule\_macos.sh to remove the launchd task:

```
./remove_schedule_macos.sh
```



## SISMO SFTP FILE UPLOADER

### 2. Verify Removal:

- Check if the task is unloaded:  
`launchctl list | grep com.user.sftpclient`

---

### Notes

- Ensure all .sh files (run\_macos.sh, schedule\_macos.sh, remove\_schedule\_macos.sh) are in the same directory as SftpClient.jar and the jre folder.
- If any issues occur, check the logs or error messages displayed in the terminal.
- The folders user\_indicator, macro\_indicator, and portfolio are automatically created during the zip creation process or specified for the application in the run\_macos.sh to function correctly.

---

### File Structure

The directory should have the following structure:

```
/your-directory
├── run_macos.sh
├── schedule_macos.sh
├── remove_schedule_macos.sh
├── SftpClient.jar
├── README_MACOS.md
├── README_MACOS.pdf
├── jre/
├── user_indicator/
├── macro_indicator/
└── portfolio/
```

---

### Troubleshooting

- **Java Runtime Not Found:**
  - Ensure the jre folder is present and contains the required Java runtime files.
- **Launchd Errors:**
  - Ensure you have the necessary permissions to create or remove launchd tasks.
- **Environment Variable Issues:**
  - Double-check the values of the environment variables in run\_macos.sh for correctness.