

1 Change Request

Short Description

“Skip all” for errors that occur during a file transfer operation

Long Description

As suggested in the [<http://www.mucommander.com/forums/viewtopic.php?f=2&t=938> forums]: adding a “Skip all” button when an error occurs in a multiple file move / copy operation would be a nice feature to have.

2 Feature Description

ファイル管理ソフトウェアである muCommander は、ユーザが選択したファイルをまとめてコピーあるいは移動する機能を持っています。ユーザがファイルを選択してコピーの実行を指示すると、muCommander はダイアログを表示し、ファイルのコピー先ディレクトリの入力を要求します。ダイアログの Copy ボタンを押すとファイルのコピーが開始されます。ファイルのコピー途中でエラーが起きた場合は、エラーの内容を告げるダイアログを表示し、ユーザにそのファイルを飛ばす (skip) か、再試行 (retry) か、処理全体の中止 (cancel) かを問い合わせます。

この機能に対して、“Skip all” という機能を追加しました。ファイルのコピー先ディレクトリを指定するダイアログに “Skip errors” というチェックボックスを表示し、ここにチェックが行われた場合は、エラーが発生してもユーザにダイアログを表示せず、自動で “Skip” ボタンが押されたものとみなして処理を実行します。また、エラーメッセージのダイアログにおいても “Skip all” ボタンを表示し、このボタンが押されたときも同様に、それ以降はダイアログを表示せず、自動で “Skip” ボタンが押されたものとして扱います。

この機能追加によって追加あるいは修正されたすべてのメソッドを特定してください。クラス宣言そのものに対する修正、インタフェースにおけるメソッド宣言（本体を持たないもの）、クラスの static 初期化子は含みません。

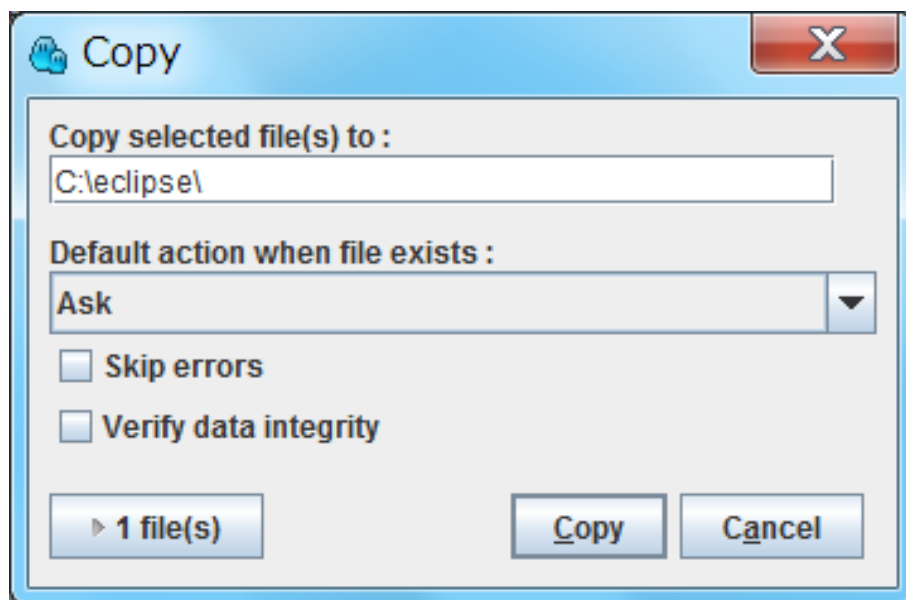


图 1 参考图

-
- `com.mucommander.job.FileJob.setAutoSkipErrors(boolean)`
 - `com.mucommander.io.StreamUtils.skipFully(InputStream, long)`
 - `com.mucommander.io.CounterInputStream.skip(long)`
 - `com.mucommander.io.BoundedReader.skip(long)`
 - `com.mucommander.io.BoundedInputStream.skip(long)`
 - `com.mucommander.job.TransferFileJob.tryCopyFile(AbstractFile, AbstractFile, boolean, String)`
 - `com.mucommander.job.FileJob.showErrorDialog(String, String, String[], int[])`
 - `com.mucommander.job.FileJob.showErrorDialog(String, String)`
 - `com.mucommander.ui.dialog.file.TransferDestinationDialog.startJob(PathUtils.ResolvedDestination)`
 - `com.mucommander.ui.dialog.file.TransferDestinationDialog.TransferDestinationDialog(MainFrame, FileSet, String, String, String, String)`

Eclipse & FL-Player 使い方まとめ

対象	機能	右クリックによる操作	ショートカットキー
メソッド	呼び出し先の表示	宣言を開く	F3
	メソッドの呼び出し元の位置表示	呼び出し階層を開くまたは参照 - ワークスペース	CTRL+ALT+H CTRL+SHIFT+G
	FL-Player のリストに追加	Add to FL Player	F10
クラス	宣言位置の表示	宣言を開く	F3
	親子関係にあるクラスを表示する	型階層を開く	F4

FL-Player

- ▶ ダブルクリックで行を開く
- ▶ 項目のクリックで編集
- ▶ 別項目やウィンドウをクリックして編集モードを抜けると自動保存される
 - 時間がある限り何度でも修正可能

“Relevant” の基準

- ▶ メソッド、コンストラクタが対象
 - 匿名クラスが機能に対応する場合、その匿名クラスを含むメソッドをリストに含める
 - static { ... } (static initializer) は含まない
 - クラス宣言やフィールドは含まない
- ▶ 調べている機能に1行でも関係していれば “Relevant”
- ▶ ベースとなっている機能の実装であれば “Irrelevant”
- ▶ 無関係なメソッドをリストに追加してしまった場合は “Irrelevant” を設定してください(削除はできません)