

SeaSonde[®]

Configuration Guide for FileExchange filexch_t2



CODAR Ocean Sensors, LTD.
1914 Plymouth St, Mountain View, CA 94043 USA
Tel. (408) 773-8240 FAX (408) 773-0514

www.codaros.com

e-mail: support@codaros.com

©2005 CODAR Ocean Sensors, Ltd.

Mar 15, 2005

This guide describes in detail the filexch_t2 usage and property list. For more basic understanding please read the Guide_File_Exchange first.

1. Overview

The filexch_t2.pl perl script uses Timbuktu to transfer files to or from a remote computer using either Internet or DialDirect service.

You will need to use Timbuktu 7.0.4 on the CLIENT computer. Timbuktu 7.0.3 contains a bug where a lost connection will cause Timbuktu's helper application "Timbuktu Host" to crash and you will be unable to make future connections until the computer is restarted or Timbuktu is re-launched and you enter in your password.

The filexch_t2.pl perl script transfers files in same manner that you would use Timbuktu by making a phone connection or IP address and then starting an Exchange session window. The script does this by using the Applescript commands built into Timbuktu. This causes an Exchange window to appear on machine running filexch_t2 during the transfer. If a user clicks the window close box or quits Timbuktu it will cause the script to fail for that instance.

The filexch_t2 perl script uses a supporting Applescript called T2Exchange.scpt to control Timbuktu. The T2Exchange.scpt file must be in the FileExchange/Scripts folder.

The filexch_t2 script copies files in alphabetically order. It selects files to copy base on whether they don't exist in the destination or optionally have a newer modification date. Timbuktu, unlike ftp, also copies the file's resource fork if it has one.

The filexch_t2 script expects a first parameter from the command line to be the plist (Property list) filename that exists in the ExchangeConfigs folder.

2. Configuration Property List

The default configuration file for filexch_t2.pl is template_t2.plist. As per instructions in Guide_File_Exchange, duplicate the template_t2.plist as a new file to /CODAR/SeaSonde/Configs/FileExchangeConfigs and rename it to SiteN_XXXX_t2.plist where N is the combining site number and XXXX is the Radial site code.

Then edit the new property list with either Property List Editor application or a Text editor. (If using a text editor, be careful to only edit the values and not the key words; if the format is not correct, the property list might be rendered unreadable.)

Most of the property list values from the default settings should only need changing for debugging and special situations. Here are what key values you do need to change:

Exchange_Method

Change the value to "IP" if using internet access to the remote site; otherwise leave the value at "Dial" for Timbuktu Direct Dial.

Exchange_Number

If Exchange_Method is "Dial", enter the phone number of the remote site, as you would dial it. If the Exchange_Method is "IP" enter the IP number x.x.x.x of the remote site.

T2_Login_Name

Enter the Timbuktu user name that gives you access to the remote site. This may be different than the OS X login user name. It is defined on the remote site in the Timbuktu application menu "Setup" item "Define Users".

T2_Login_Pass

Enter the password for the Timbuktu user name.

Modem_Ignore_DialTone

Set to "No" only if your country uses a different dial tone than what is expected by the modem.

Folder_Replace_With

Change the value from "Site_1" to Site_<number> where <number> is the combining site number set in your created combine grid for the remote Radial Site. This causes all the Folder_Local paths to be automatically redirected to the specific Site_# folder without having to edit each task Folder_Local path.

Enable for each task in Tasks

For task you wish to get specific files from make sure the **Enabled** value is "Yes"

By default, the RDLi task is enabled to bring back the LLUV ideal radials.

Enable RDLm task to bring back LLUV measured pattern radials.

Enable Rads task to bring back old style ideal radials.

Enable Radz task to bring back old style measured pattern radials.

You should disable the tasks you are not using especially for a slow phone line.

If the phone line is speedy and reliable you might want to enable the DiagStatFile task.

Lastly, when trying out the exchange for the first time setting these values will help to better diagnose what is happening.

Modem_Speaker_On = "Yes"

ExchangeVerbocity = "1000"

T2_Task_Finished_Chime = "Yes"

When you are finished testing the configuration, you might want to set these values back to the default so that the computer is quiet in its normal operation.

Modem_Speaker_On = "No"

ExchangeVerbosity = "999"

T2_Task_Finished_Chime = "No"

3. Property List Details

In the Root dictionary:

The key **An_Exchange_Kind** should always contain the String value "T2" to identify this property list as the kind to use for filexch_t2. This key is *required*.

The key **Exchange_Direction** needs to contain a String value of either "**pull**" to copy files *from* the remote computer or "**push**" to copy files *to* the remote computer. This key is *optional*. If this key is missing, the T2Exchange script used by filexch_t2 defaults to "pull". Optionally each Task may also have this key overriding this Root key setting.

The key **Exchange_Method** specifies the type of transfer to use it should be a String value of either "Dial" for DialDirect or "IP" for internet. Only the first character is checked so you can whatever message you want. This key is *required*.

The key **Exchange_Number** is a String value of the phone number to dial if method is "Dial" or the IP number in 0.0.0.0 format if method is "IP". This key is *required*.

The key **Exchange_Verbosity** is a Number value, which specifies how much diagnostic information gets written to the exchange log file. A value of 0 will output very little while all messages will be output with a value of 999. A value of 1000 will cause the script to tell the system to open the log file; this will typically cause the Console to show the log file, which is a convenient way of debugging the task. This key is *optional*.

The key **File_Check_Modification_Dates** is a Boolean value of "No" to only copy files that do not exist on the destination side. If "Yes", existing files for both source and destination are checked to see if the source is newer and if so it is copied. This key is *optional* and can be overridden by each Task.

The key **File_Copy_Timeout** is a Number value that sets the time out in seconds to copy a file. This key is *optional* and can be overridden by each Task.

The key **Folder_Replace_Find** along with key **Folder_Replace_With** allow for a quick way to control the folder paths in the Task list. Any **Folder_Local** or **Folder_Remote** paths will be searched to see if they contains the **Folder_Replace_Find** string and have the match replaced with the **Folder_Replace_With** string. For SeaSonde use, if you just change the **Folder_Replace_With** to the "Site_#" where # is the site number, so you don't need to change each **Folder_Local** of every Task. These keys are *optional*.

The key **Modem_Ignore_DialTone** is a Boolean value if "Yes" tells Timbuktu to ignore checking for the dial tone before dialing. The helps with some countries where the dial tone is different than what the modem expects. The key is *optional* and if it is missing the T2Exchange script defaults to "No".

The key **Modem_Speaker_On** is a Boolean value if "Yes" tells Timbuktu to output the modem connection sound to the computer's speaker. This is an invaluable tool to those who well know what a modem connection sounds like. This key is *optional* and if missing the T2Exchange script defaults to "No".

The key **T2_Login_Name** is the String value of the Timbuktu login user name to use. This key is *required*.

The key **T2_Login_Pass** is the String value of the Timbuktu login password to use. This key is *required*.

The key **T2_Task_Finished_Chime** is the Boolean value, which tells Timbuktu to make its chime noise at the end of the transfers for each task. This key is *optional*.

The key **T2_Timeout_Connection** is the Number value, which sets the number of seconds to wait when establishing a connection to the remote computer. This key is *optional*.

The key **T2_Timeout_Session** is the Number value, which sets the number of seconds to wait when establishing a file transfer session to the remote computer. This key is *optional*.

The key **Tasks** is a dictionary that contains a list of dictionaries where each dictionary in Tasks contains the controls keys for each file transfer to/from different folders. The dictionaries in Tasks may have any key name you wish to describe the transfer. You should prefix your name with a_ b_ c_ and so on, so that using the Property List Editor application to edit the plist causes the Tasks to performed in alphabetical order. This key is *required* and must contain at least one task dictionary. (*Don't use numbers in the key name for the Task as the reading in chadsmo.d.pl cannot currently handle this.*)

The keys for each dictionary in Tasks:

The key **Enabled** is a Boolean value if "No" means the task is disabled and will be skipped. If "Yes", the task will be performed by filexch_t2. This key is *required*.

The key **File_Copy_Grouping** is a Number value which sets how many files can be copied each Timbuktu transfer. Multiple transfers are used to copy enough files to satisfy the **File_Copy_Maximum** or number of available source files. Timbuktu can transfer more than one file at a time saving overhead, which makes the transfers more efficient in bandwidth, but if one file fails, all do for the transfer. For the IP method, it makes sense to set the grouping value to 10 or higher. For a phone connection with high risk of drop out, it makes more sense to set the grouping to 1, so that there is a higher chance at least one file will successfully transfer. This key is *optional* and if missing then a file grouping of 1 will be used. Note, If the **File_Copy_Grouping** is greater than two and there is only one or two more files left to copy after this group, then the group will temporarily increase by the one or two files left so that the last grouping will never be less than three files.

The key **File_Copy_Maximum** is a Number value to limit the maximum number of possible files to copy for this task. This prevents overwhelming a single transfer especially on a phone connection with lots of files, however, if this value is smaller than the number files be created on the source side between each run time of this task then the copy will never get to the latest files. This key is *optional*, and if missing, then the maximum files copied are 9999.

The key **File_Filter_Names_Startingwith** is a String value to filter files to copy that only have filenames that start with this value ignoring case. To allow all files in the source folder to be copied either delete this key or set it to empty "". This key is *optional*.

The key **Folder_Local** is String value of the POSIX style path of local folder to copy files to/from depending on whether the Exchange_Direction is pull/push. A path of "/Codar/SeaSonde/Data" would be the Data folder in the SeaSonde folder in the Codar folder of the startup disc. If any of the folders in path are missing they will be created. This key is *required*.

The key **Folder_Remote** is String value of the POSIX style path of remote folder to copy files from/to depending on whether the Exchange_Direction is pull/push. If any of the folders in the path are missing the task will fail with an error in the log file. *(If anyone can figure out how to AppleScript Timbuktu to create folders, I will add it. Netopia Timbuktu support says they do not support AppleScript)*. This key is *required*.

The following task keys are optional copies of Root keys, which if used, override the Root key:

The key **Exchange_Direction** needs to contain a string value of either "**pull**" to copy files *from* the remote computer or "**push**" to copy files *to* the remote computer. This key is *optional*.

The key **File_Check_Modification_Dates** is a Boolean value of "**No**" to only copy files that do not exist on the destination side. If "**Yes**", existing files for both source and destination are checked to see if the source is newer and if so it is copied. This key is *optional*. Note, files like Radials, CSS are usually only created once and never change unless reprocessed. Files like wave history lists and diagnostics, are updated with each new data and require that the modification date is checked so that the newer file is copied over.

The key **File_Copy_Timeout** is a Number value that sets the time out in seconds to copy a file. This key is *optional* and can be overridden by each Task. When **File_Copy_Grouping** is greater than one, the **File_Copy_Timeout** is multiplied by the actual number of grouped files currently being transferred, and if it exceeds this time to copy the current group then the transfer will fail. If this Task key and the root key **File_Copy_Timeout** are missing then the T2Exchange script defaults to 180seconds.