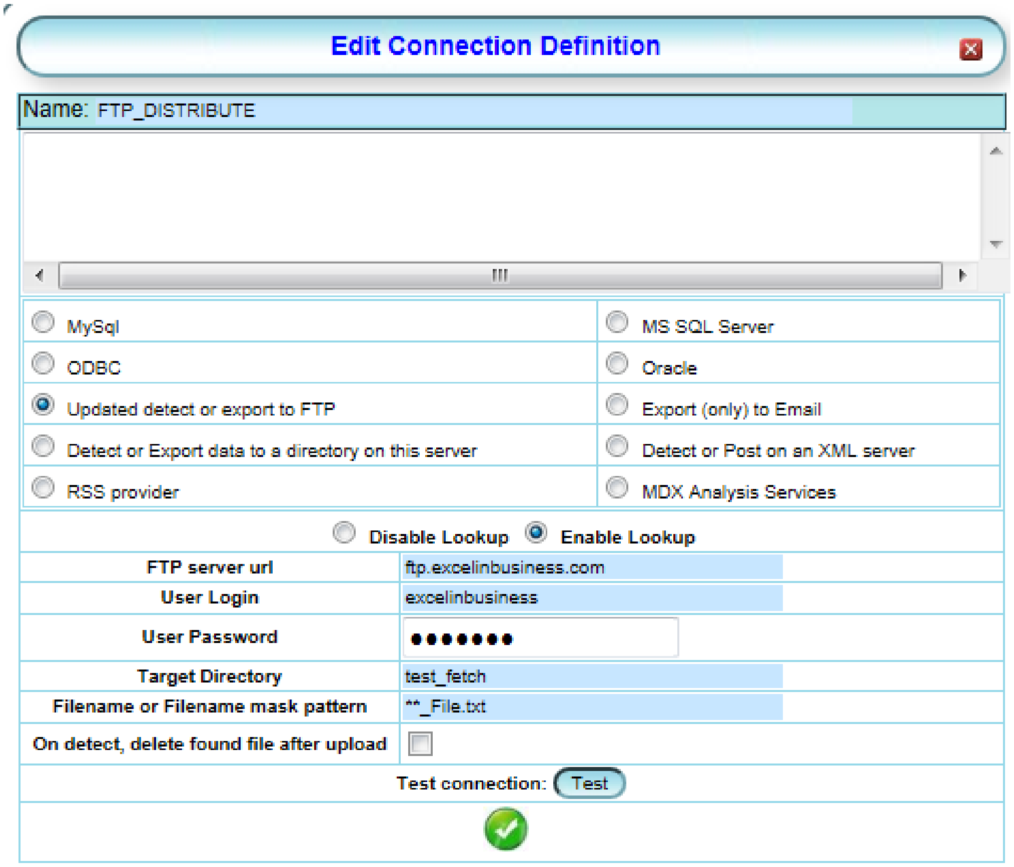
File Management

This document describes the system functions available to manage files within ALCW. Essentially the requirement is move files (read and write and copy), fetching files from external sources on FTP and PSFTP sites or external mapped drives and placing them (copying or moving) into known locations for subsequent processing.

It is recommended that file or ftp references use the ALCW dictionary rather than explicit (hard-coded) file-path or 'url' strings.
Setting up a dictionary entry is found for admin/system users under this icon in Profiles: 🌐
A file or ftp reference looks like this:



Edit Connection Definition

Name: FTP_DISTRIBUTE

○ MySql	○ MS SQL Server
○ ODBC	○ Oracle
● Updated detect or export to FTP	○ Export (only) to Email
○ Detect or Export data to a directory on this server	○ Detect or Post on an XML server
○ RSS provider	○ MDX Analysis Services

○ Disable Lookup ● Enable Lookup

FTP server url	ftp.excelinbusiness.com
User Login	excelinbusiness
User Password	●●●●●●●
Target Directory	test_fetch
Filename or Filename mask pattern	**_File.txt
On detect, delete found file after upload	□

Test connection: Test

✓

This is an ftp reference that includes a filename mask, this can be over-ridden when calling a fetch function, so is not explicitly required.

Detecting and Fetching Files:

```
DETECT_LOOKUP(dictionary_entry_name [, filename/mask]);
```

This function looks in the location specified by the dictionary entry, using either the file mask/name in the entry or the optional parameter filename/mask. If found the file(s) are moved to the working directory of the running profile, registered in the audit log, and returns the list of files found and their paths. So this call produces this output:

```
$detected = DETECT_LOOKUP('FTP_DISTRIBUTE');
dump($detected);
console("Specific File");
$detected = DETECT_LOOKUP('FTP_DISTRIBUTE','AA_File.txt');
dump($detected);
```

```
.[0].[FileName] AA_File.txt
.[0].[FilePath]
CUSTOM/PF510142af682dd/FilesToProcess/UP5105337a5ccde.txt
.[0].[EffectiveDate] 2013-01-23 15:40:03
.[1].[FileName] BB_File.txt
.[1].[FilePath]
CUSTOM/PF510142af682dd/FilesToProcess/UP5105337c0cea2.txt
.[1].[EffectiveDate] 2013-01-23 15:40:05

Specific File
.[0].[FileName] AA_File.txt
.[0].[FilePath]
CUSTOM/PF510142af682dd/FilesToProcess/UP5105337eb9f95.txt
.[0].[EffectiveDate] 2013-01-23 15:40:03
```

DETECT_LOOKUP will also have the same functionality where the dictionary entry is a mapped file directory reference. Like this:

```
$detected = DETECT_LOOKUP('Test FILE','ABC_*.txt');
dump($detected);
```

```
.[0].[FileName] ABC_File.txt
.[0].[FilePath]
CUSTOM/PF510142af682dd/FilesToProcess/UP510538bd1fb88.txt
```

FTP_READ

```
FTP_READ(into_file, dictionaty_lookup_name[,
                                       optional_file_name]);
```

This function uses ftp to fetch a specific file into the target file (in this profiles working directory). Like this:

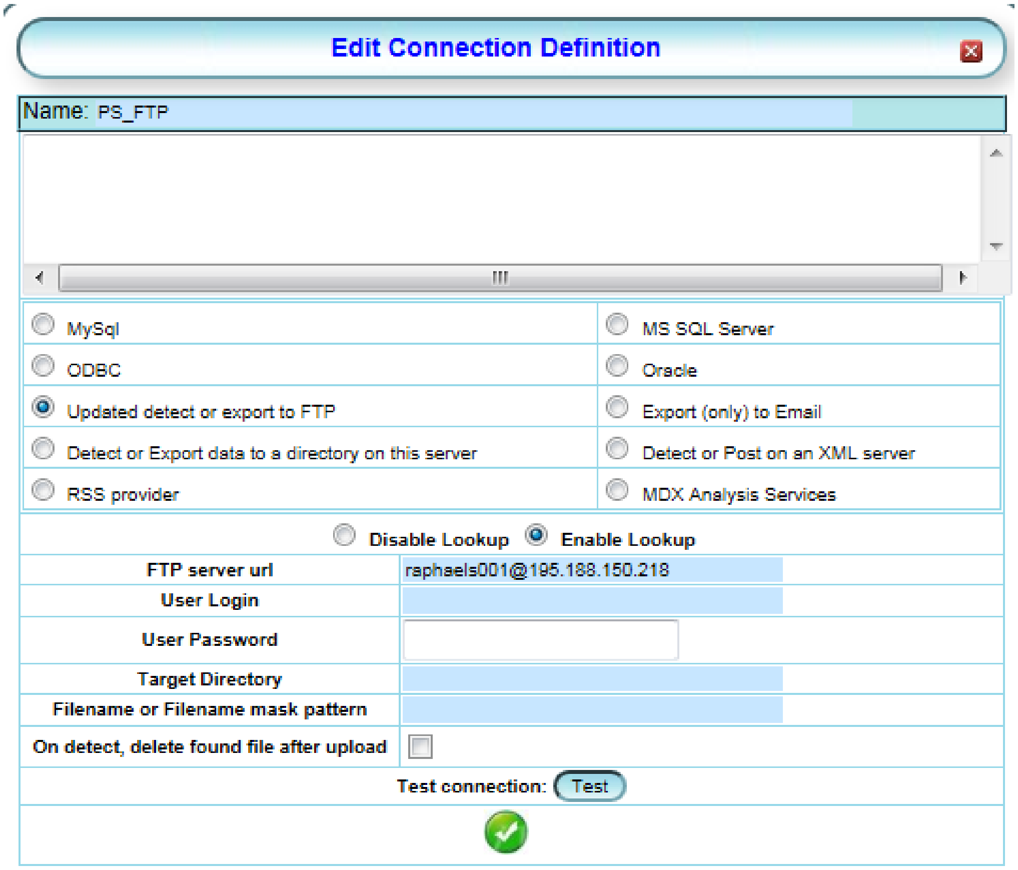
```
$readfile =
FTP_READ('MyFile.txt','FTP_DISTRIBUTE','BB_File.txt');
dump($readfile);
$s = FILE_GET_CONTENTS('MyFile.txt');
dump($s);
```

```
.[FilePath] CUSTOM/PF510142af682dd/MyFile.txt
.[FileName] BB_File.txt

 This is BB File
```

Using PSFTP

The ftp method with a key. The directory entry is set up like this:



Edit Connection Definition

Name: PS_FTP

○ MySql	○ MS SQL Server
○ ODBC	○ Oracle
● Updated detect or export to FTP	○ Export (only) to Email
○ Detect or Export data to a directory on this server	○ Detect or Post on an XML server
○ RSS provider	○ MDX Analysis Services

○ Disable Lookup ● Enable Lookup

FTP server url	raphaels001@195.188.150.218
User Login	
User Password	
Target Directory	
Filename or Filename mask pattern	
On detect, delete found file after upload	□

Test connection: Test

✓

That is, the only necessary setting is the FTP server url. This process requires that the binary exe psftp.exe is in the binary directory along with the private key file named thus: psftp_key.ppk.

To obtain a list of files use this call:

```
$ps = PSFTP_READ('PS_FTP','LIST','cimb');
dump($ps);
```

Returns a structure with Status (zero is success), Output from the batch run, Request the request used for this command, and (in this case for the action ‘LIST’) the structure of the files in the directory.

```
.[Status] 0
.[Output] Connected to 195.188.150.218
Remote working directory is /
Remote directory is now /cimb
Listing directory /cimb
drw-rw-rw-   3 user   group   40960 Jan 26 23:54 .
-rw-rw-rw-   1 user   group   98369 Oct 15 21:52
2012_1015_FA21A - ATM Cardholder Dispenser Detail per
ATM(MEPS).pdf
```

```
.[Request] D:\ALCW\ALCWCodeRoot\binary/psftp.exe -i
D:\ALCW\ALCWCodeRoot\binary/psftp_key.ppk -P 2222 -batch -b
"D:\ALCW\ALCWLIVE/CUSTOM/PF510295ccb866b/tmp\psftp_batch.bat" -
v raphaels001@195.188.150.218
.[List].[0]  .
.[List].[1]  2012_1015_FA21A - ATM Cardholder Dispenser Detail
per ATM(MEPS).pdf
```

To fetch a file using psftp, then the call is of the form:

```
PSFTP_READ(dictionary_entry_for url, ‘GET’,
           ‘directory on source’,
           ’file name to fetch’);
```

The directory on the source can be a single string if the directory structure is just a single level from the root, or a structured array that will be iterated through doing cd at each level.

Like this:

```
$dir.1 = 'cimb';
$ps = PSFTP_READ('PS_FTP','GET',$dir,"2012_1015_FA21A - ATM
Cardholder Dispenser Detail per ATM(MEPS).pdf");
dump($ps);
Could also have been (in this case):
$ps = PSFTP_READ('PS_FTP','GET',’cimb’,"2012_1015_FA21A - ATM
Cardholder Dispenser Detail per ATM(MEPS).pdf");
```

Will produced this output (Status, Output and Request as before) with the properties FileName and FilePath.

```
.[Status] 0
.[Output] Connected to 195.188.150.218
Remote working directory is /
Remote directory is now /cimb
New local directory is
D:\ALCW\ALCWLIVE\CUSTOM\PF510295ccb866b\tmp
remote:/cimb/2012_1015_FA21A - ATM Cardholder Dispenser Detail
per ATM(MEPS).pdf => local:2012_1015_FA21A - ATM Cardholder
Dispenser Detail per ATM(MEPS).pdf

.[Request] D:\ALCW\ALCWCodeRoot\binary/psftp.exe -i
D:\ALCW\ALCWCodeRoot\binary/psftp_key.ppk -P 2222 -batch -b
"D:\ALCW\ALCWLIVE/CUSTOM/PF510295ccb866b/tmp\psftp_batch.bat" -
v raphaels001@195.188.150.218
.[FilePath]
D:\ALCW\ALCWLIVE/CUSTOM/PF510295ccb866b/tmp\2012_1015_FA21A -
ATM Cardholder Dispenser Detail per ATM(MEPS).pdf
.[FileName] 2012_1015_FA21A - ATM Cardholder Dispenser Detail
per ATM(MEPS).pdf
```

Placing Files:

Use COPY_FILE to move files once fetched (must be mapped drives, not FTP):

```
For explicit copies (not recommended):
COPY_FILE(‘source path’, ‘destination_path’);

Using directory look up.
COPY_FILE(‘source_path’, ‘dictionary_name_lookup’,
‘destination_file_name’);
```

For example:

```
$ok = COPY_FILE('MyFile.txt','c:\\FTP_ATM\\MyFile.txt');

$ok = COPY_FILE('MyFile.txt','FTP_ATM', 'InFile.txt');
```

Using FTP to write a file:
FTP is also available with this function:

```
FTP_FILE_WRITE(‘source_path’,’dictionary_ftp_entry’,
                             ’destination_file_name’);
```

Working Example:

This example reads a file from PSFTP and writes to a mapped drive/directory location:

```
$ps = PSFTP_READ('PS_FTP','GET','cimb',"2012_1015_FA21A - ATM
Cardholder Dispenser Detail per ATM(MEPS).pdf");
if $ps is false exit_with_error("Unable to obtain file from
PS_FTP");
$ok = COPY_FILE($ps.FilePath,'FTP_ATM', $ps.FileName);
if $ps is false exit_with_error("Unable to copy file from
PS_FTP to FTP_ATM");
console("Copied {$ps.FileName} ok");
exit true;
```

Other File Management Functions:

See also: FILE_EXISTS, FILE_PATH_EXISTS, FILE_DELETE