FTPUpload

FTPUpload [***flags***] ***urlStr, localPathStr***

The FTPUpload operation uploads a file or a directory to an FTP server on the Internet.

Warning: When you upload a file or directory to an FTP server, all previous contents of the server file or directory are obliterated.

For background information on Igor's FTP capabilities and other important details, see **File Transfer Protocol (FTP)** on page IV-272.

FTPUpload sets a variable named V_flag to zero if the operation succeeds and to nonzero if it fails. This, in conjunction with the /Z flag, can be used to allow procedures to continue to execute if a FTP error occurs.

If the operation succeeds, FTPUpload sets a string named S_Filename to the full file path of the uploaded file or, if the /D flag was used, to the full path to the base directory that was uploaded. This is useful in conjunction with the /I flag.

If the operation fails, S_Filename is set to "".

Parameters

urlStr specifies the file or directory to create. It consists of a naming scheme (always "`ftp://`"), a computer name (e.g., "`ftp.wavemetrics.com`" or "`38.170.234.2`"), and a path (e.g., "`/Test/TestFile1.txt`"). For example: "`ftp://ftp.wavemetrics.com/pub/test/TestFile1.txt`".

urlStr must always end with a file name if you are uploading a file or with a directory name if you are uploading a directory, in which case *urlStr* must not end with a slash.

To indicate that *urlStr* contains an absolute path, insert an extra '/' character between the computer name and the path. For example:

```
ftp://ftp.wavemetrics.com//pub/test
```

If you do not specify that the path in *urlStr* is an absolute path, it is interpreted as a path relative to the FTP user's base directory. Since pub is the base directory for an anonymous user, this URL references the same directory:

```
ftp://ftp.wavemetrics.com/test
```

Special characters such as punctuation that are used in *urlStr* may be incorrectly interpreted by the operation. If you get unexpected results and *urlStr* contains such characters, you can try percent-encoding the special characters. If you get unexpected results and *urlStr* contains such characters, you can try percent-encoding the special characters. See **Percent Encoding** on page IV-268 for additional information.

localPathStr and *pathName* specify the name and location on your hard disk of the local file to be uploaded. If you use a full or partial path for *localPathStr*, see **Path Separators** on page III-451 for details on forming the path.

localPathStr must always end with a file name if you are uploading a file or with a directory name if you are uploading a directory. In the case of a directory, *localPathStr* must not end with a colon or backslash.

FTPUpload displays a dialog that you can use to identify the file or directory to be uploaded in the following cases:

1. You used the /I (interactive) flag.
2. You did not completely specify the location of the file or folder to be uploaded via *pathName* and *localPathStr*.
3. There is an error in *localPathStr*. This can be either a syntactical error or a reference to a nonexistent directory.

See **Examples** for examples of constructing a URL and local path.

Flags

/D	Uploads a complete directory. Omit it if you are uploading a file.
/I	Interactive mode which displays a dialog for choosing the local file or directory to be uploaded.

/M=*messageStr* Specifies the prompt message used by the dialog in which you choose the local file or directory to be uploaded.

/N=*portNumber* Specifies the server's TCP/IP port number to use (default is 21). In almost all cases, this will be correct so you won't need to use the /N flag.

/O[=*mode*] Overwrite. FTPUpload *always* overwrites the specified server file or directory, whether /O is used or not.

If /O=2 is *not* used, all files and subdirectories in the destination directory on the server are first deleted and then the local files and directories are uploaded to the server.

If /O=2 *is* used, the existing contents the contents of the local source directory are merged into the remote directory instead of completely overwriting it.

/P=*pathName* Contributes to the specification of the file or directory to be uploaded. *pathName* is the name of an existing symbolic path. See **Examples**.

/S=*showProgress* Determines if a progress dialog is displayed.

0: No progress dialog.

1: Show a progress dialog (default).

/T=*transferType* Controls the FTP transfer type.

0: Image (binary) transfer (default).

1: ASCII transfer.

See **FTP Transfer Types** on page IV-275 for more discussion.

/U=*userNameStr* Specifies the user name to be used when logging in to the FTP server. If this flag is omitted or if *userNameStr* is "", you will be logged in as an anonymous user. Use this flag if you have an account on the FTP server.

/V=*diagnosticMode* Determines what kind of diagnostic messages FTPUpload will display in the history area. *diagnosticMode* is a bitwise parameter, with the bits defined as follows:

Bit 0: Show basic diagnostics. Currently this just displays the URL in the history.

Bit 1: Show errors. This displays additional information when errors occur.

Bit 2: Show status. This displays commands sent to the server and the server's response.

The default value for *diagnosticMode* is 3 (show basic and error diagnostics). If you are having difficulties, you can try using 7 to show the commands sent to the server and the server's response.

See **FTP Troubleshooting** on page IV-275 for other troubleshooting tips.

/W=*passwordStr* Specifies the password used when logging in to the FTP server. Use this flag if you have an account on the FTP server.

If this flag is omitted, "nopassword" will be used for the login password. This will work with most anonymous FTP servers. Some anonymous FTP servers request that you use your email address as a password. You can do this by including the /W="<your email address>" flag.

If /W is omitted, the login is done using a default password that will work with most anonymous FTP servers.

See **Safe Handling of Passwords** on page IV-270 for information on handling sensitive passwords.