CORE SYSTEMS WINSOCK FTP LIBRARY PACKAGE

VERSION 1.0

CONTENTS

1. OVERVIEW
- 1.1 Introduction
- 1.2 CFTP Dynamic Link Library
- 1.3 CFTP VBX Control

2. USING CFTP DYNAMIC LINK LIBRARY
- 2.1 Application Interface
- 2.2 Connection Management Commands
- 2.3 Login Commands
- 2.4 Control Session Commands
- 2.5 Data Transfer Commands
- 2.6 API Reference
  - 2.6.1 CFTPChDir
  - 2.6.2 CFTPClose
  - 2.6.3 CFTPGetCwd
  - 2.6.4 CFTPDir
  - 2.6.5 CFTPGet
  - 2.6.6 CFTPLogin
  - 2.6.7 CFTPMkDir
  - 2.6.8 CFTPOpen
  - 2.6.9 CFTPParent
  - 2.6.10 CFTPPut
  - 2.6.11 CFTPRemove
  - 2.6.12 CFTPRename
  - 2.6.13 CFTPRmDir
  - 2.6.14 CFTPReset
  - 2.6.15 CFTPSetType
  - 2.6.16 CFTPSetTrace
- 2.7 Notification Reference
  - 2.7.1 FTP_RX_CONTROL_DATA
  - 2.7.2 FTP_TX_CONTROL_DATA

3. USING CFTP VBX CONTROL
- 3.1 Application Interface
- 3.2 Connection Management
- 3.3 Login
- 3.4 Control Session
- 3.5 Data Transfer
- 3.6 Methods
- 3.6 Events
- 3.6 Attributes

APPENDICES

A. Error Codes and Header Files

A.1 Error Codes

A.2 Header Files

1. OVERVIEW

1.1 Introduction

1.2 CFTP Dynamic Link Library

1.3 The CFTP VBX Control

2. USING CFTP DYNAMIC LINK LIBARY

2.1 Application Interface

2.2 Connection Management Commands

2.3 Login Commands

2.4 Control Session Commands

2.5 Data Transfer Commands

2.6 API Reference

2.6.1 CFTPChDir()

Description The CFTPChDir changes the current working directory on the server host.

```
#include <cftp.h>

int WINAPI CFTPChDir
(
        HFTP    hFtp,
        LPCSTR      lpcszDir;
)
```

hFtp FTP session handle.
lpcszDir New directory.

Remarks The function change the current working directory on the FTP Server host. This function accepts absolute OR relative pathnames.

Return Value The function will return the FTP Server's response code if the command is successful transmitted and executed on the server host. Otherwise a value of -1 is returned.

Error Codes None.

Compatibility Standards: None
16-Bit: WIN, WINDLL

See Also

Example CFTPRemoveDir, CFTPCreateDir

2.6.2 CFTPClose()

Description The CFTPClose closes the FTP control session

```
#include <cftp.h>

int WINAPI CFTPClose
(
        HFTP    hFtp
)
```

hFtp FTP session handle.

Remarks The function close the specified *hFtp* control session.

Return Value The function will return CFTP_SERVICE_CLOSED (221) if successful. Otherwise a value of CFTP_NOT_LOGGED_IN (530) is returned.

Error Codes

Compatibility

Standards:	None
16-Bit:	DOS, QWIN, WIN, WINDLL

See Also

Example

2.6.3 CFTPGetCwd()

Description The CFTPGetCwd returns the user's current working directory on the server host.

```
#include <cftp.h>

int WINAPI CFTPGetCwd
(
        HFTP    hFtp,
        LPCSTR      lpcszDir,
        WORD        wLength
)
```

hFtp	FTP session handle.
lpcszDir	Return buffer for the current working directory.
wLength	Buffer size.

Remarks The function retrieves the user's full path current working directory on the server host and copies it into the buffer *lpcszDir*.

Return Value The function will return CFTP_CMD_OK (200) if successful. Otherwise a value of CFTP_NOT_LOGGED_IN (530) is returned if the connection is closed. A value of less than 0 is returned if the buffer size *wLength* is not enough to store the complete path.; the absolute value of the return code is the size of the expected buffer.

Error Codes

Compatibility

Standards:	None
16-Bit:	DOS, QWIN, WIN, WINDLL

See Also

Example

2.6.4 CFTPDir()

Description The CFTPDir gets the listing of the current directory.

```
#include <cftp.h>

int WINAPI CFTPDir
(
        HFTP    hFtp,
        HWND        hNotifyWnd
)
```

hFtp FTP session handle.

hNotifyWnd Window handle for sending notifications.

Remarks The function requests a listing of the current working directory on the server host. If a window handle *hNotifyWnd* is specified, the function will operate in the asynchronous (non-blocking) mode, and the directory listing data messages will be sent to this window handle. When operating in the asynchronous mode, the all data displays will be handled by the window handle, and not be handle by the function. If the window handle *hNotifyWnd* is not provided (NULL), the function will operate in the synchronous mode and the data displays will be done by the function in a list box control.
If the function operates in the asynchronous mode, following directory listing data messages will be sent:

CFTP_NOTIFY_RX_DIRECTORY_BEGIN
wParam Total number of records.
lParam 0L
CFTP_NOTIFY_RX_DIRECTORY_RECORD
wParam Buffer size
lParam Buffer pointer, NULL terminated string.
CFTP_NOTIFY_RX_DIRECTORY_END
wParam -1 indicates abort condition, 0 if transfer complete
lParam 0L

Even in the synchronous mode, these notification message also be sent to the application window, but only for information purposes and no file I/O action is required from the application.

Return Value If the function is operate in asynchronous mode, it will return CFTP_DIRECTORY_OK (150) if successful, and the data transfer is to be started asynchronously. If the function is operate in synchronous mode, it will return CFTP_DIRECTORY_COMPLETED (250) if successful. Otherwise a value of CFTP_DIRECTORY_UNAVAILABLE (550) if the requested directory listing is unavailable or can not be accessed.

Error Codes

Compatibility Standards: None
16-Bit: DOS, QWIN, WIN, WINDLL

See Also

Example

2.6.5 CFTPGet()

Description The CFTPGet retrieves a file from the server host

```
#include <cftp.h>

int WINAPI CFTPChDir
(
        HFTP        hFtp,
        LPCSTR      lpcszRemoteFile,
        LPCSTR      lpcszLocalFile,
        HWND        hNotifyWnd
)
```

hFtp FTP session handle.

lpcszRemoteFile File name on the server host.

lpcszLocalFile File name on the local host to be created.

hNotifyWnd Window handle for sending notifications.

Remarks The function requests data transfer from the file with file name *lpcszRemoteFile* on the server host to the local file name *lpcszLocalFile*. If a window handle *hNotifyWnd* is specified, the function will operate in the asynchronous (non-blocking) mode, and the data transfer operation messages will be sent to this window. When operating in the asynchronous mode, all file I/O will not be handle by the function. If the window handle *hNotifyWnd* is not provided (NULL), the function will operate in the synchronous mode and the data transfer will be done by the function.
If the function operates in the asynchronous mode, following data transfer operation messages will be sent:

CFTP_NOTIFY_RX_DATA_BEGIN
- *wParam* Total file size in bytes.
- *lParam* Remote file name

CFTP_NOTIFY_RX_DATA
- *wParam* Buffer size
- *lParam* Buffer pointer

CFTP_NOTIFY_RX_DATA_END
- *wParam* -1 indicates abort condition, 0 if transfer complete
- *lParam* 0L

Even in the synchronous mode, these notification message also be sent to the application window, but only for information purposes and no file I/O action is required from the application.

Return Value If the function is operate in asynchronous mode, it will return CFTP_FILE_STATUS_OK (150) if successful, and the data transfer is to be started asynchronously. If the function is operate in synchronous mode, it will return CFTP_FILE_TRANSFER_COMPLETED (250) if successful. Otherwise a value of CFTP_FILE_UNAVAILABLE (550) if the requested file is unavailable or can not be accessed.

Error Codes

Compatibility

Standards:	None
16-Bit:	DOS, QWIN, WIN, WINDLL

See Also

Example

2.6.6 CFTPLogin()

Description

```
#include <cftp.h>

int WINAPI CFTPLogin
(
        HFTP      hFtp,
        LPCSTR    lpcszUser,
        LPCSTR    lpcszPasswd,
        LPCSTR    lpcszAccount
)
```

hFtp	FTP session handle.
lpcszUser	User name.
lpcszPasswd	User password.
lpcszAccount	User account

Remarks The function proceeds with the login sequence. It will first present the user login name, then if the host requires a password, the function will send the user's password. If the account is further required, the function will sends the user's account, as well. The function only checks for the user login name prior to commence the login process, if the user login name is not provided the function returns immediately, with error. The password and account will be send as is (without checking).

Return Value The function returns CFTP_USER_LOGGED_IN (230) if successful. Otherwise a value of CFTP_NOT_LOGGED_IN (530) if the login process failed.

Error Codes

Compatibility

Standards:	None
16-Bit:	DOS, QWIN, WIN, WINDLL

See Also

Example

2.6.7 CFTPMkDir()

Description The CFTPMkDir creates a directory on the server host.

```
#include <cftp.h>

int WINAPI CFTPMkDir
(
        HFTP        hFtp,
        LPCSTR      lpcszDir,
)
```

hFtp FTP session handle.

lpcszDir New directory name to be created on the server host.

Remarks The function requests the server host to create a directory at the current working directory.

Return Value The function will return CFTP_CMD_OK (200) if successful. Otherwise a value of CFTP_CMD_FAILED (550) is returned if the request is unavailable or can not be be done due to accessed violation.

Error Codes

Compatibility Standards: None
16-Bit: DOS, QWIN, WIN, WINDLL

See Also

Example

2.6.8 CFTPOpen()

Description The CFTPOpen function establish a connection to the FTP server.

```
#include <cftp.h>

HFTP WINAPI CFTPOpen
(
        LPCSTR      lpcszHost
)
```

lpcszDir Server host name

Remarks The function allocates memory for the FTP session and tries to establish a FTP connection to the host.

Return Value The function will return the FTP session handle if successful. Otherwise a value of 0 is returned.

Error Codes

Compatibility Standards: None
16-Bit: DOS, QWIN, WIN, WINDLL

See Also

Example

2.6.9 CFTPParent()

Description The CFTPParent function changes the directory to the parent directory.

```
#include <cftp.h>

int WINAPI CFTPParent
(
        HFTP    hFtp
)
```

hFtp FTP session handle.

Remarks The CFTPParent function changes the current working directory to the parent directory on the server host.

Return Value The function will return CFTP_CMD_OK (200) if successful. Otherwise a value of CFTP_CMD_FAILED (550) is returned if the request is unavailable or can not be be done due to accessed violation.

Error Codes

Compatibility

Standards:	None
16-Bit:	DOS, QWIN, WIN, WINDLL

See Also

Example

2.6.10 CFTPPut()

Description The CFTPPut places a file on the server host.

```
#include <cftp.h>

int WINAPI CFTPPut
(
        HFTP        hFtp,
        LPCSTR      lpcszLocalFile,
        LPCSTR      lpcszRemoteFile,
        HWND        hNotifyWnd
)
```

hFtp FTP session handle.

lpcszLocalFile File name on the local host.

lpcszRemoteFile File name on the server host to be placed.

hNotifyWnd Window handle for sending notifications.

Remarks The function requests data transfer from a file with the file name *lpcszLocalFile* on the local host to be place on the server host with file name *lpcszRemoteFile*. If a window handle *hNotifyWnd* is specified, the function will operate in the asynchronous (non-blocking) mode, and the data transfer operation messages will be sent to this window. When operating in the asynchronous mode, all file I/O will not be handle by the function. If the window handle *hNotifyWnd* is not provided (NULL), the function will operate in the synchronous mode and the data transfer will be done by the function.
If the function operates in the asynchronous mode, following data transfer operation messages will be sent:

CFTP_NOTIFY_TX_DATA_BEGIN
wParam Total file size in bytes.
lParam Remote file name
CFTP_NOTIFY_TX_DATA
wParam Buffer size
lParam Buffer pointer
CFTP_NOTIFY_TX_DATA_END
wParam -1 indicates abort condition, 0 if transfer complete
lParam 0L

Even in the synchronous mode, these notification message also be sent to the application window, but only for information purposes and no file I/O action is required from the application.

Return Value If the function is operate in asynchronous mode, it will return CFTP_FILE_STATUS_OK (150) if successful, and the data transfer is to be started asynchronously. If the function is operate in synchronous mode, it will return CFTP_FILE_TRANSFER_COMPLETED (250) if successful. Otherwise a value of CFTP_FILE_UNAVAILABLE (550) if the requested file is unavailable or can not be accessed.

Error Codes

Compatibility

Standards:	None
16-Bit:	DOS, QWIN, WIN, WINDLL

See Also

Example

2.6.11 CFTPRemove()

Description The CFTPRemove function deletes a file.

```
#include <cftp.h>

int WINAPI CFTPRemove
(
        HFTP        hFtp,
        LPCSTR      lpcszFile,
)
```

hFtp FTP session handle.

lpcszFile File name to be removed..

Remarks The CFTPRemove function removes a file with the specified file name *lpcszFile* in the current working directory of the server host.

Return Value The function will return CFTP_CMD_OK (200) if successful. Otherwise a value of CFTP_CMD_FAILED (550) is returned if the request is unavailable or can not be be done due to accessed violation.

Error Codes

Compatibility Standards: None
16-Bit: DOS, QWIN, WIN, WINDLL

See Also

Example

2.6.12 CFTPRename()

Description The CFTPRename function change the file name.

```
#include <cftp.h>

int WINAPI CFTPRename
(
        HFTP      hFtp,
        LPCSTR    lpcszOldName,
        LPCSTR    lpcszNewName
)
```

hFtp FTP session handle.

lpcszOldFile . Old file name to be changed.

lpcszNewFile . New file name.

Remarks The CFTPRename function renames a file with the specified file name *lpcszOldName* to the new filename *lpcszNewName*, in the current working directory of the server host.

Return Value The function will return CFTP_CMD_OK (200) if successful. Otherwise a value of CFTP_CMD_FAILED (550) is returned if the request is unavailable or can not be be done due to accessed violation.

Error Codes

Compatibility Standards: None
16-Bit: DOS, QWIN, WIN, WINDLL

See Also

Example

2.6.13 CFTPRmDir()

Description The CFTPRmDir function deletes a directory

```
#include <cftp.h>

int WINAPI CFTPRmDir
(
        HFTP        hFtp,
        LPCSTR      lpcszDir,
)
```

hFtp FTP session handle.

lpcszDir Directory name.

Remarks The CFTPRmDir function removes an empty directory of the specified directory name *lpcszDir* in the current working directory of the server host.

Return Value The function will return CFTP_CMD_OK (200) if successful. Otherwise a value of CFTP_CMD_FAILED (550) is returned if the request is unavailable or can not be be done due to accessed violation.

Error Codes

Compatibility Standards: None
16-Bit: DOS, QWIN, WIN, WINDLL

See Also

Example

2.6.14 CFTPReset()

Description The CFTPReset resets the FTP control session

```
#include <cftp.h>

void WINAPI CFTPReset
(
        HFTP    hFtp,
)
```

hFtp FTP session handle.

Remarks The CFTPReset function resets an existing FTP control connection. Any data transfer connection will be terminated and the FTP handle will no longer valid after this function call.

Return Value None.

Error Codes

Compatibility

Standards:	None
16-Bit:	DOS, QWIN, WIN, WINDLL

See Also

Example

2.6.15 CFTPSetTransferType()

Description The CFTPSetTransferType sets the data transfer type.

```
#include <cftp.h>

int WINAPI CFTPSetTransferType
(
        HFTP    hFtp,
        WORD        wTransferType,
)
```

hFtp FTP session handle.

wTransferType Data transfer type

Value		Description
FTP_DATA_TYPE_IMAGE	Binary	data representation files
FTP_DATA_TYPE_ASCII	ASCII	data representation files

Remarks The CFTPSetTransferType function sets the transfer type for all the subsequence data transfers. Two types of data transfers are supported: the ASCII text files and the binary files.

Return Value The function will return CFTP_CMD_OK (200) if successful. Otherwise a value of CFTP_CMD_NOT_SUPPORT (504) is returned if the request feature is unavailable at this server host.

Error Codes

Compatibility Standards: None
16-Bit: DOS, QWIN, WIN, WINDLL

See Also

Example

2.6.16 CFTPSetCapture()

Description The CFTPSetCapture sets the data capture mode.

```
#include <cftp.h>

void WINAPI CFTPSetCapture
(
        HFTP    hFtp,
        BOOL    bCaptureMode
)
```

hFtp FTP session handle.

bCaptureMode Data capture flag

Value	Description
TRUE	Turn on data capture mode
FALSE	Turn off data capture mode

Remarks The CFTPSetCapture function sets the data capture mode. Data transfer in the control connection will be send to the parent window with the notifications FTP_RX_CONTROL_DATA and FTP_TX_CONTROL_DATA. This function enables/disables the generation of these notifications to the parent window.

Return Value None.

Error Codes

Compatibility Standards: None
16-Bit: DOS, QWIN, WIN, WINDLL

See Also

Example

2.7 Notification Reference

2.7.1 FTP_RX_CONTROL_DATA

2.7.2 FTP_TX_CONTROL_DATA

3. USING CFTP VBX CONTROL

3.1 Application Interface

3.2 Connection Management

3.3 Login

3.4 Control Session

3.5 Data Transfer

3.6 Methods

3.6 Events

3.6 Attributes

APPENDICES

A. Error Codes and Header Files

A.1 Error Codes

A.2 Header Files