

FTPOCX

WILL

株式会社ウィル

白紙ページ

白紙ページ

- Microsoft、Windows、Windows NT、Visual Basic、ActiveX、Office、Access、Excel は、米国 Microsoft Corporation の米国ならびに各国における登録商標です。
- その他本書に掲載されている会社名、製品名はそれぞれ各社の商標又は登録商標です。

目次

はじめに.....	5
商品に含まれるもの.....	7
動作環境について.....	7
インストール.....	8
ライセンスの登録.....	10
サンプルを見る.....	13
サポートについて(無償).....	14
バージョンアップについて(無償).....	15
再配布について.....	17
プログラミング概要.....	19
通信を開始する.....	22
通信を終了する.....	22
メソッドの発行.....	23
状態遷移図.....	25
プロパティ.....	29
Account プロパティ.....	31
Copyright プロパティ.....	32
FileType プロパティ.....	33
FileCode プロパティ.....	34
Port プロパティ.....	35
Passive プロパティ.....	36
PreCommand プロパティ.....	37
Rest プロパティ.....	39
SysCode プロパティ.....	40
State プロパティ.....	41
TransErrorEvent プロパティ.....	42
メソッド.....	43
Abend メソッド.....	45
Abort メソッド.....	46
AppendFile メソッド.....	47
Connect メソッド.....	48
Cwd メソッド.....	49

Cd メソッド	50
Delete メソッド	51
Dir メソッド	52
Disconnect メソッド	53
GetFile メソッド	54
Ls メソッド	55
IsIdle メソッド	56
Mkdir メソッド	57
Noop メソッド	58
PutFile メソッド	59
Pwd メソッド	60
Quote メソッド	61
Rename メソッド	62
Rmdir メソッド	63
Size メソッド	64
Site メソッド	65
Login メソッド	66
イベント	67
CommandDone イベント	69
Connected イベント	70
DirClose イベント	71
DirData イベント	72
DirOpen イベント	73
Disconnected イベント	74
GetOpen イベント	75
GetData イベント	76
GetClose イベント	77
PutData イベント	78
PutClose イベント	79
Trace イベント	80
TransError イベント	81
Winsock エラーコード	83
サンプル	87
WILLFTP	89
FTPEditor	92

FTP SYNC.....	94
BULK FTP	95
索 引.....	97

白紙ページ

はじめに

はじめに

白紙ページ

商品に含まれるもの

- | | |
|--------------|--|
| 1. CD-ROM | ▪ Willware.exe
▪ Cryptdll.exe
(暗号 DLL 専用・実行環境用セットアップキット)
▪ readme.txt |
| 2. フロッピーディスク | ▪ レジストリファイル
▪ readme.txt |
| 3. 使用許諾契約書 | |
| 4. マニュアル | |

動作環境について

■対応 OS

FTPOCX は、以下に示す OS で動作確認を行っております。

Microsoft Windows 95、Microsoft Windows 98、
Microsoft WindowsNT 4.0、Microsoft Windows 2000
Microsoft Windows XP、Microsoft Windows 2003

■開発に必要なソフトウェア

FTPOCX をご使用いただくには、以下のいずれかのソフトウェアが必要です。

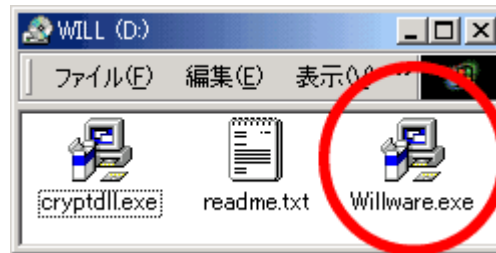
Microsoft Visual Basic Ver 5.0
Microsoft Visual Basic Ver 6.0
Microsoft Office 2000 (Access、Excel)

FTPOCX は、Microsoft Visual Basic Ver 5.0 で作成しています。サンプルは、Microsoft Visual Basic Ver 5.0 及び Ver 4.0 で作成しています。

※ 本製品は日本語環境のみの対応となります。

インストール

製品の CD-ROM に含まれているセットアップキット (Willware.exe) をダブルクリックします。

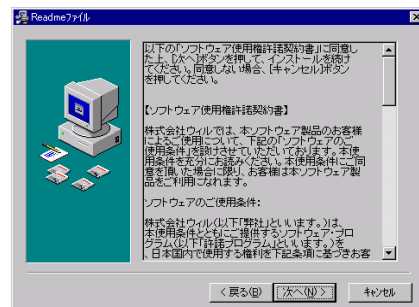


画面にしたがって、インストールを進めて下さい。

1. インストールを始めます。「次へ」をクリックして下さい。



2. 使用許諾契約書です。内容に同意される場合は「次へ」をクリックして下さい。



3. インストール先のフォルダを指定します。初期設定でよろしければ「次へ」をクリックして下さい。別のフォルダを指定したい場合は「参照」をクリックし、フォルダを指定して下さい。



4. インストール中に置換されるファイルのバックアップを作成できます。そのバックアップファイルの保存先フォルダを指定します。初期設定でよろしければ「次へ」をクリックして下さい。



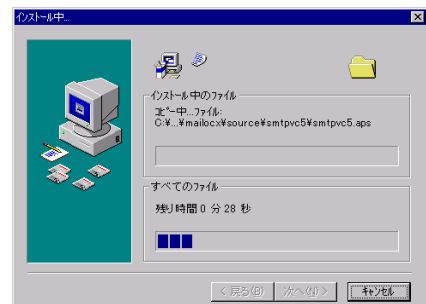
5. WILLWARE Components を登録するスタートメニュー又はプログラクマネージャのグループフォルダを指定します。初期設定では、新規に「WILLWARE Components」の名前でフォルダを作成します。特に指定する必要がなければ、初期設定をお勧めします。



6. プログラムのコピーを開始します。「次へ」をクリックして下さい。



7. プログラムのコピーをしています。中断する場合は、「キャンセル」をクリックして下さい。



8. インストールが完了しました。「完了」をクリックし、インストールを終了して下さい



はじめに

ライセンスの登録

■レジストリファイルから登録する

ライセンスを登録します。製品に含まれているフロッピーディスクのレジストリファイル (EFXXXXXXXXX.reg) をダブルクリックして下さい。(「XXXXXXXXXX」は、任意の数字がファイル名として付けられています。)

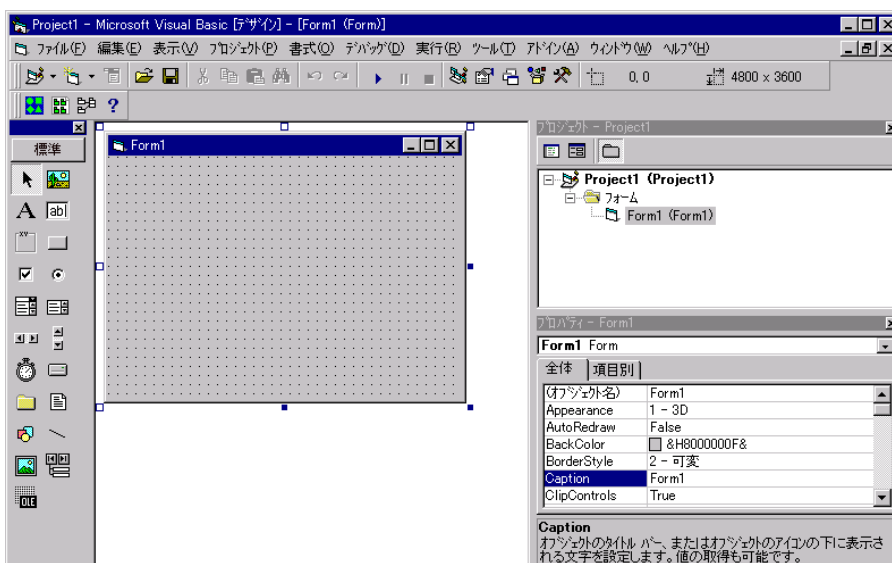


以下のメッセージボックスが表示され、ライセンスがレジストリに登録されます。

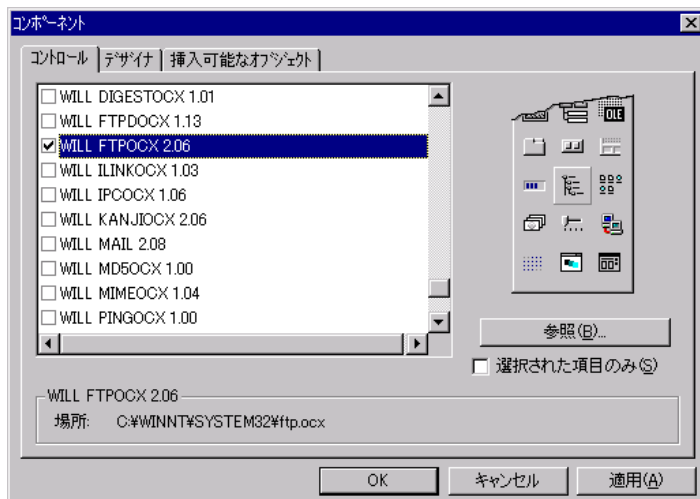


■手動で登録する

あらかじめ電子メールで通知しているライセンス情報を利用してライセンスを登録する等、レジストリファイルを利用しない場合は、VisualBasic 起動後に新規プロジェクトを選択し以下のデザイン画面を開きます。



ツールバーの「プロジェクト」から、「コンポーネント」を選択し、「コンポーネント」画面を開きます。次にコントロールタブの一覧から FTPOCX を選択して「OK」をクリックすると、FTPOCX がツールボックスに追加され、アイコンが表示されます。



ツールボックスに追加された FTPOCX を選択し、フォームにアイコンを貼り付けると、以下の「WILL LICENSE REGISTRATION」画面が表示されます。ここで、ユーザー名、シリアル番号、キーコードをそれぞれ入力してライセンス登録を行います。



はじめに

■トライアルライセンスから正規ライセンスへの移行

既にトライアルライセンスが登録されている場合には、デザイン画面にある FTPOCX のプロパティで「バージョン情報」をクリックして下さい。



「WILL LICENSE REGISTRATION」画面が表示されますので、ここで正規ライセンスを入力して下さい。



■ライセンス入力時のご注意

※ライセンスが入力できない!?

入力したライセンスにスペースが含まれていないか確認して下さい。(ライセンスに、スペースは使用していません。)

※登録したライセンスを認識しない!?

ライセンスを登録しても、オブジェクトが新規ライセンスを認識していない場合は、FTPOCX のアイコンを少し動かして下さい。この作業により、オブジェクトにライセンスが記憶されます。

※トライアルライセンスで作成したアプリケーションはどうする!?

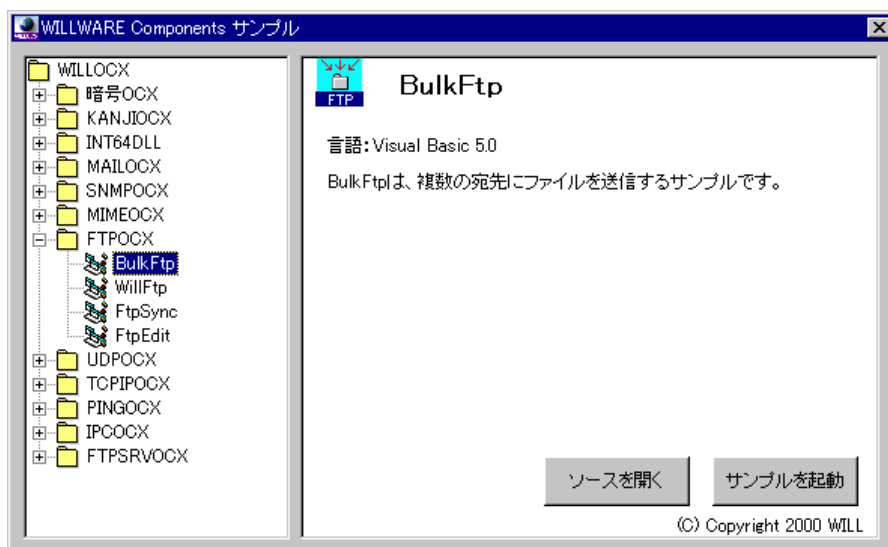
既にトライアルライセンスで作成したアプリケーションは、正規ライセンスを登録した後、再コンパイルする必要があります。

サンプルを見る

インストールが完了すると、スタートメニューに「WILLWARE Components」が追加されます。



「WILLWARE Components」の「サンプル」を起動すると「WILLWARE Components サンプル」画面が表示されます。サンプルの起動、またはそれぞれのソースを開くことができます。但し、ソースを開くにはライセンスが必要です。トライアルライセンス又は、正規ライセンスを登録してご利用下さい。(ライセンスの登録方法は前項の「ライセンスの登録」をご覧下さい。)



はじめに

サポートについて(無償)

サポートは基本的に電子メールで受け付けております。サポートは無償でご利用いただけます。

■お問い合わせの前に

サポート作業を円滑に行うために、お問い合わせの際には以下の情報をご用意下さい。

1. 製品名及びバージョン
2. 開発環境(OSの種類及びバージョン、サービスパッケージの種類)
3. 開発ツール及びバージョン
4. サーバーの種類
5. 問題点
 - (1) エラー内容又は、エラー状況のハードコピー
 - (2) 問題点となる部分のサンプルソースコード

■FAQ

弊社ホームページの「サポート」のページで、キーワードを入力して FAQ を検索できます。休業日などサポートの対応が遅れる場合もありますので、まずはこちらをご確認下さい。

■お問合せ先

info@will-ltd.co.jp

バージョンアップについて(無償)

製品のバージョンアップは、すべて無償です。

■バージョンアップ情報の入手方法

バージョンアップの情報は、弊社ホームページの新着情報で通知し、各商品のページの更新履歴で更新内容を掲示致します。

■最新バージョンの入手方法

最新バージョンのプログラムは、弊社ホームページ(<http://www.will-ltd.co.jp/>)のダウンロードのページよりダウンロードすることができます。ダウンロードするファイルは、以下のバージョンアップの目的により異なりますのでご注意ください。

- **WILLWARE Components(全製品用)セットアップキットを利用してバージョンアップ**
ファイル名 : 「Willware.exe」

WILLWARE Components(全製品用)セットアップキットは全ての製品をインストールするためのものです。そのため本製品以外の製品及びサンプル、マニュアルも同時にバージョンアップされます。

- **各コンポーネント毎のセットアップキットを利用してバージョンアップ**
ファイル名 : 「○○○ocx.exe」

各コンポーネントのファイル(ocx、dll)及び、依存ファイルのみバージョンアップされません。サンプル及びマニュアルはバージョンアップされませんのでご注意ください。

はじめに

■バージョンアップをする前に

各セットアップキットを利用してバージョンアップをする前に、以下のことにご注意ください。

● WILLWARE Components(全製品用)セットアップキットを利用してバージョンアップする場合は、古いバージョンをアンインストールしてから、最新バージョンをインストールすることをお勧めいたします。

※ アンインストールの方法は、スタートメニューから「設定」→「コントロールパネル」→「アプリケーションの追加と削除」の画面で、「WILLWARE Components」を選択し、画面の指示に従って行って下さい。

● 各コンポーネント毎のセットアップキットを利用してバージョンアップする場合は、最新バージョンをそのままインストールして下さい。古いファイルは上書きされます。

※ 弊社製品を複数ご利用いただいている場合、いずれか1つをバージョンアップしても他の製品に影響はありません。

■バージョンアップの方法

セットアップキットをダブルクリックし、画面の指示に従ってインストールを進めて下さい。

再配布について

■作成したアプリケーションの配布時

FTPOCX を利用して作成したアプリケーションの配布時のランタイムライセンスはフリーです。但し、開発ライセンスの配布はできません。

■再配布時に必要な配布可能ファイル

FTPOCX を利用して作成したアプリケーションを配布する場合には、以下のファイルを添付する必要があります。()内は推奨バージョンです。

- ・ FTP.OCX
- ・ TCPIP.OCX
- ・ KANJI.OCX
- ・ VB5JP.DLL (Ver 5.0.4319)
- ・ MSVBVM50.DLL (Ver 5.2.8244)
- ・ MFC42.DLL (Ver 4.21.7022)
- ・ MFC42LOC.DLL (Ver 4.21.7022)
- ・ MSVCRT.DLL (Ver 5.00.7022)
- ・ OLEPRO32.DLL (Ver 5.0.4118)
- ・ OLEAUT32.DLL (Ver 2.20.4118)

※ セットアップウィザードを使用する場合

FTPOCX をインストールすると、自動的に OCX の依存ファイルが以下のディレクトリにインストールされます。

C:\Windows\system (Windows 95, 98 の場合)

C:\WINNT\system32 (Windows NT, 2000, 2003 の場合)

C:\Windows\system32 (WindowsXP の場合)

セットアップウィザードを実行すると自動的にアプリケーション配布時に必要な OCX (内部で利用している OCX) と、DLL ファイルが Setup.lst ファイルに追加されます。

■著作権

- ・ FTPOCX およびこれに付随するマニュアルの著作権は株式会社ウィル(横浜市保土ヶ谷区)にあります。
- ・ 本ソフトウェアおよびマニュアルを運用した結果については、当社は一切責任を負いません。
- ・ 本ソフトウェアの仕様またはマニュアルに記載されている事項は予告無く変更することがあります。
- ・ マニュアルなどに記載されている会社名、製品名は、各社の商標および登録商標です。

はじめに

- ・ FTPOCX を利用するアプリケーションは FTPOCX の著作権表示を行わなければなりません。Copyright プロパティに FTPOCX の著作権を示す文字列があります。アプリケーションまたはドキュメントのいずれかにこの文字列を表示して、FTPOCX を使用していることを示してください。

プログラミング概要

白紙ページ

FTPOCXは、メソッドの指定に従ってFTPサーバーと非同期に通信を行い、実行結果をイベントとして通知します。メソッドを発行すると、メソッドに応じたイベントが発生しますので、イベントの発生にしたがってプログラミングをしてください。自然とFTPクライアントプログラムを作成することができます。また、いくつかのプロパティが用意されていて、ファイルの転送モードや、漢字コードの変換方法を変更することができます。メソッドを発行する前に正しく設定してください。

通信を開始する

通信を開始するには Connect メソッドを使用します。このメソッドを用いて、FTP サーバーにログインします。ログインに失敗しても自動的に FTP セッションは切断されません。明示的に Disconnect メソッドを発行してください。

通信を終了する

通信を終了するには、Disconnect メソッドを使用します。このメソッドを用いると、サーバーに QUIT コマンドを発行して FTP セッションが閉じられるのを待ちます。正常に切断されたら DisConnected イベントが発生します。

DisConnected イベントは、Disconnect メソッドの発行だけでなく、サーバーがタイムアウトを起こして FTP セッションを切断した場合にも発生します。

メソッドの発行

メソッドは、成功、失敗の如何に関わらず、CommandDone イベントを発生します。(IsIdle メソッド、Abort メソッド、Abend メソッドを除く)。メソッドによっては、CommandDone イベントが発生する前に、メソッド特有のイベントが発生します(Ls/GetFile/PutFile メソッド)。

■Ls メソッドの発行

Ls メソッドを発行すると、DirOpen イベント、DirData イベント、DirClose イベントが順番に発生します。DirOpen でディレクトリリストの受け取りの準備を行い、DirData でディレクトリリストを 1 行ずつ受け取り、DirClose ですべてのディレクトリリストを受け取ったことを確認することができます。

■GetFile メソッドの発行

GetFile メソッドを発行すると、GetOpen イベント、GetData イベント、GetClose イベントが順番に発生します。GetOpen イベントで、ローカルファイルのオープンを行い、GetData イベントでローカルファイルにデータを追加し、GetClose でローカルファイルをクローズします。なお、GetFile メソッドで指定したリモートファイルが存在しない場合は、GetOpen/GetData/GetClose イベントは発生しません。

■PutFile メソッドの発行

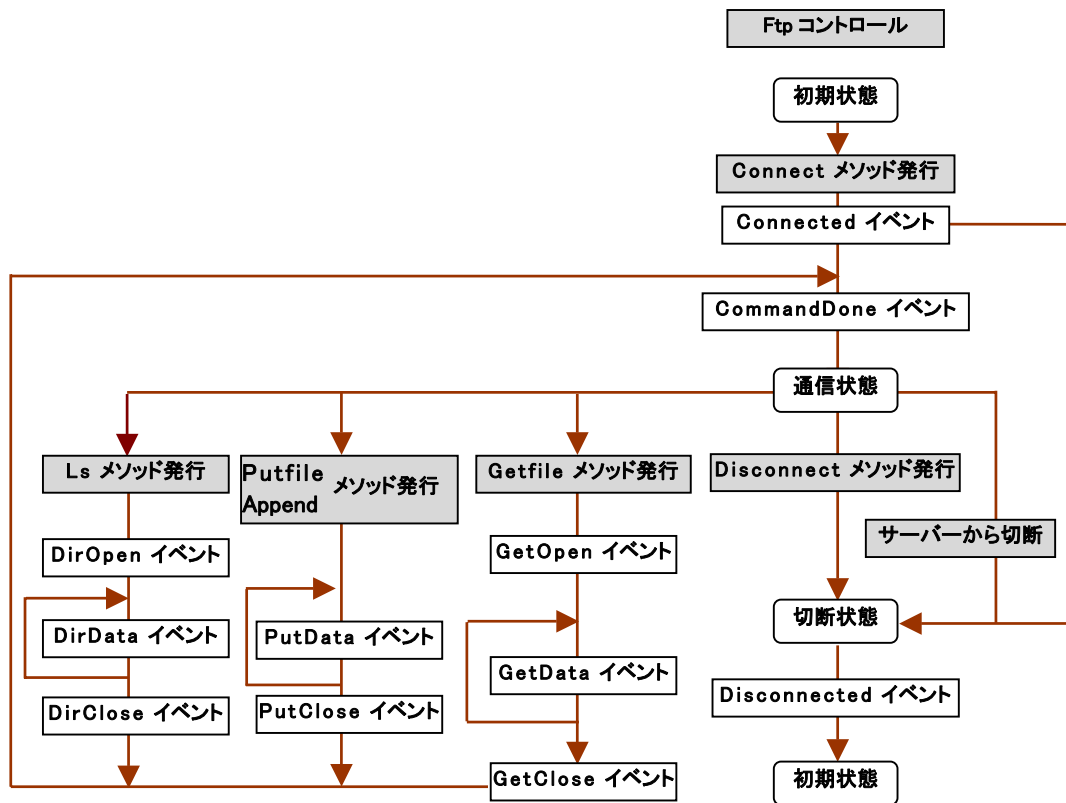
PutFile メソッドを発行すると、PutData イベント、PutClose イベントが順番に発生します。PutOpen イベントはありませんので、PutFile メソッドを発行する前に、必要なら送信するファイルをオープンしておきます。PutData イベントでData パラメーターに送信するデータを格納します。Data パラメーターになにも格納しないと、送信データの終了とみなされ、データコネクションが閉じられて、PutClose イベントが発生します

白紙ページ

状態遷移図

状態遷移図

白紙ページ



状態遷移図

白紙ページ

プロパティ

プロパティ

白紙ページ

Account プロパティ

■機能

アカウントを要求するシステムに接続するときのアカウント名の設定です。Connect メソッドを使う前に設定してください。

■構文

Object.Account[=Value]

Account プロパティの構文の指定項目は次のとおりです。

(指定項目)	(内容)
Object	FTPOCX オブジェクトです。
Value	アカウントの指定をする文字列式です。

■データ型

文字列(String)

Copyright プロパティ

■機 能

FTPOCX のコピーライト文字列。プログラム、ヘルプファイル、マニュアルなどのいずれかにこのプロパティの文字列を表示してください。この値は参照のみ可能です。

■構 文

Object. Copyright

Copyright プロパティの構文の指定項目は次のとおりです。

(指定項目)	(内 容)
Object	FTPOCX オブジェクトです。

■データ型

文字列(String)

FileType プロパティ

■機能

ファイル転送モードを指定します。アスキーモード転送であれば、“A”を、バイナリモード転送であれば、“I”をセットしてください。アスキーモードは、送信側が改行コードをCR LFに変換します。

バイナリモードは変換をしません。このプロパティを読み出すと現在設定されているモードを知ることができます。

ファイル転送モードはファイル転送指示メソッドの呼び出しにより、サーバーに通知されます。この通知を抑止するには、FileType に””を代入して下さい。この場合、ファイル転送モードは Quote メソッドを用いて指示して下さい。

■構文

Object. FileType[=Value]

FileType プロパティの構文の指定項目は次のとおりです。

(指定項目)	(内容)
Object	FTPOCX オブジェクトです。
Value	転送するファイルのモードを指定する文字列式です。

■データ型

文字列(String)

FileCode プロパティ

■機能

サーバー上でのファイルの漢字コードを指定します。“JIS”、“SJIS”、“EUC”、“BIN”を指定できます。“BIN”であれば、送受信に際して漢字コードの変換を行いません。“JIS”、“SJIS”、“EUC”であれば、送信する際には送信データを指定された文字コードに変換して送信します。受信する際は、サーバーのファイルの文字コードを指定された文字コードとして変換します。“BIN”以外では、GetData に渡されるデータは SJIS になっています。

■構文

Object.FileCode[=Value]

FileCode プロパティの構文の指定項目は次のとおりです。

(指定項目)	(内容)
Object	FTPOCX オブジェクトです。
Value	サーバー上でのファイルの漢字コードを指定する文字列式です

■設定値

Value の設定値は次のとおりです。

(値)	(説明)
JIS	送受信する際にデータを JIS に変換します。
SJIS	送受信する際にデータを SJIS に変換します。
EUC	送受信する際にデータを EUC に変換します。
BIN	送受信する際に漢字コードの変換を行いません。

■データ型

文字列(String)

Port プロパティ

■機 能

接続する FTP サーバーのポート番号です。デフォルトは 21 です。

■構 文

Object. Port[=Value]

Port プロパティの構文の指定項目は次のとおりです。

(指定項目)	(内 容)
Object	FTPOCX オブジェクトです。
Value	接続する FTP サーバーのポート番号を指定する整数式または文字列式(サービス名称)です。

■データ型

文字列(String)

Passive プロパティ

■機 能

PASSIVE モードでデータ転送を行ないます。

■構 文

Object. Passive [=Boolean]

Passive プロパティの構文の指定項目は次のとおりです。

(指定項目)	(内 容)
Object	FTPOCX オブジェクトです。
Boolean	PASSIVE モードを利用するかどうかを決めます。

■設定値

Value の設定値は次のとおりです。

(値)	(説 明)
True	PASSIVE モードを利用します。
False	PASSIVE モードを利用しません。(初期値)

■データ型

ブール型(Boolean)

PreCommand プロパティ

■機能

ログイン処理の前に発行するコマンドを記述します。
(ファイヤーウォール対応に利用します。)

■構文

Object. PreCommand [=Text]

PreCommand プロパティの構文の指定項目は次のとおりです。

(指定項目)	(内容)
Object	FTPOCX オブジェクトです。
Text	vbCrLf で区切られた FTP コマンド

■設定値

Text の設定値は次のとおりです。

(パターン)	(コーディング方法)
ファイヤーウォールなし	Ftp1.PreCommand = "" Ftp1.Connect FtpServer,User,Pass
ファイヤーウォールサーバーに接続し、OPEN コマンドを発行して FTP サーバーに接続する。その後、通常 LOGIN する。ファイヤーウォールのユーザー名、パスワードは用いない。	Ftp1.PreCommand = "OPEN " & FtpServer Ftp1.Connect FireWall,User,Pass
ファイヤーウォールサーバーに接続し、ファイヤーウォールのユーザー名、パスワードを用いてログインする。そして SITE コマンドを発行して FTP サーバーに接続し、通常 LOGIN する。	Ftp1.PreCommand = "USER " & FireWallUser & vbCrLf _ "PASS " & FireWallPass & vbCrLf _ "SITE " & FtpServer Ftp1.Connect FireWall,User,Pass
ファイヤーウォールサーバーに接続し、通常 LOGIN する。ユーザー名は、'FTP ユーザー名 @FTP サーバー名' の形式。ファイヤーウォールのユーザー名、パスワードは用いない。	Ftp1.PreCommand = "" Ftp1.Connect FireWall, _ User & "@" & FtpServer, _Pass

<p>ファイヤーウォールサーバーに接続し、ファイヤーウォールのユーザー名,パスワードを用いてログインしたのち、通常 LOGIN する。ユーザー名は、'FTP ユーザー名@FTP サーバー名'の形式。ファイヤーウォールのユーザー名,パスワードは用いない。</p>	<pre>Ftp1.PreCommand = _ "USER " & FireWallUser & vbCrLf _ "PASS " & FireWallPass Ftp1.Connect FireWall, _ User & "@" & FtpServer, _Pass</pre>
--	--

■変数

(変数名)	(意味)
FtpServer	目的とする FTP サーバーのホスト名
User	FTP サーバーのユーザー名
Pass	FTP サーバーのパスワード
FireWall	ファイヤーウォールのホスト名
FireWallUser	ファイヤーウォールでのユーザー名
FireWallPass	ファイヤーウォールでのパスワード

■データ型

文字列(String)

Rest プロパティ

■機能

このプロパティが0以上にセットされていると、GetFile メソッドを行う際に、RETR する直前に REST を発行します。初期値は、-1 で、GetFile が完了すると-1に戻ります。

■構文

Object.Rest[=Value]

Rest プロパティの構文の指定項目は次のとおりです。

(指定項目)	(内容)
Object	FTPOCX オブジェクトです。
Value	0以上にセットされていると、GetFile メソッドを行う際に、RETR する直前に REST を発行します。

■解説

REST コマンドは、パラメータで指定した位置までリモートファイルのポインタを進めます。この機能を使うためには REST コマンドが FTP サーバーでサポートされていることが必要です。

■データ型

長整数(Long)

SysCode プロパティ

■機能

サーバー上でのファイル名およびエラーメッセージで使用される漢字コードの取り扱いを指定します。通常は自動判別を行うが、SJISとEUCで自動判別できない場合があるので、自動判別できないときのヒントを指定します。“EUC”または“SJIS”を指定する。JIS やアルファベットのみの場合には自動判別ができるので、そのような場合は、このプロパティは参照されません。

■構文

Object.SysCode[=Value]

SysCode プロパティの構文の指定項目は次のとおりです。

(指定項目)	(内容)
Object	FTPOCX オブジェクトです。
Value	自動判別できないときのヒントを指定します。

■設定値

Value の設定値は次のとおりです。

(値)	(説明)
EUC	ファイル名およびエラーメッセージで使用される漢字コードを EUC に指定します。
SJIS	ファイル名およびエラーメッセージで使用される漢字コードを SJIS に指定します。

■データ型

文字列(String)

State プロパティ

■機能

現在実行されているメソッドが格納されます。参照のみ可能です。

■構文

Object.State

State プロパティの値は次のとおりです。

(定数)	(説明)
Abort	現在実行中のメソッドは Abort です。
AppendFile	現在実行中のメソッドは AppendFile です。
Connect	現在実行中のメソッドは Connect です。
Cwd	現在実行中のメソッドは Cwd です。
Cd	現在実行中のメソッドは Cd です。
Delete	現在実行中のメソッドは Delete です。
Dir	現在実行中のメソッドは Dir です。
DisConnect	現在実行中のメソッドは DisConnect です。
GetFile	現在実行中のメソッドは GetFile です。
Ls	現在実行中のメソッドは Ls です。
MkDir	現在実行中のメソッドは MkDir です。
Noop	現在実行中のメソッドは Noop です。
PutFile	現在実行中のメソッドは PutFile です。
Pwd	現在実行中のメソッドは Pwd です。
Quote	現在実行中のメソッドは Quote です。
Rename	現在実行中のメソッドは Rename です。
RmDir	現在実行中のメソッドは RmDir です。
Size	現在実行中のメソッドは Size です。
Site	現在実行中のメソッドは Site です。
Login	現在実行中のメソッドは Login です。

■データ型

文字列(String)

TransErrorEvent プロパティ

■機 能

Ture のとき、エラーをイベントとして報告します。
False のとき、エラーを MsgBox で報告します。

■構 文

Object.TransErrorEvent

TransErrorEvent プロパティの構文の指定項目は次のとおりです。

(指定項目)	(内 容)
Object	FTPOCX オブジェクトです。

■データ型

ブール型(Boolean)

メソッド

メソッド

白紙ページ

Abend メソッド

■機 能

通信を強制的に切断します。通信中であれば、Disconnected イベントが発生します。通信中でなければ何も起こりません。

■構 文

Object.Abend()

Abend メソッドの構文の指定項目は次のとおりです。

(指定項目)	(内 容)
Object	FTPOCX オブジェクトです。

■戻り値

なし。

Abort メソッド

■機 能

Ls/Dir/GetFile/PutFile によるデータ転送を中断します。これはいつでも発行できます。中断されたメソッドは処理が中断され、CommandDone イベントが発生します。転送中でない場合は、何もしません。

■構 文

Object.Abort()

Abort メソッドの構文の指定項目は次のとおりです。

(指定項目)	(内 容)
Object	FTPOCX オブジェクトです。

■戻り値

なし。

AppendFile メソッド

■機 能

APPE コマンドを用いてリモートファイルに格納します。FileType プロパティ、FileCode プロパティに応じてコード変換が行われます。IsIdle メソッドを使って、コマンド発行可能状態であるかどうか調べてから呼び出してください。コマンド発行可能状態で無い場合に呼び出すとエラー（エラー番号“17”）になります。このメソッドが成功すると、PutData/PutClose イベントが発生します。転送が終わると CommandDone イベントが発生します。

■構 文

Object.AppendFile(File As String)

AppendFile メソッドの構文の指定項目は次のとおりです。

(指定項目)	(内 容)
Object	FTPOCX オブジェクトです。
File	ファイル名を指定する文字列式です。

■戻り値

なし。

Connect メソッド

■機 能

RemoteHost で指定される FTP サーバーに接続します。すでに接続済みである場合は、エラーになります。上手く接続できたら、UserName、Password を用いてログインを試みます。ログインに失敗しても、コネクションは切断しません。明示的に Disconnect メソッドを呼び出してください。処理がおわると、CommandDone イベントが発生します。

■構 文

Object.Connect (RemoteHost As String, Username As String, Password As String)
Connect メソッドの構文の指定項目は次のとおりです。

(指定項目)	(内 容)
Object	FTPOCX オブジェクトです。
RemoteHost	FTP サーバーを指定する文字列式です。
UserName	ユーザー名を指定する文字列式です。
Password	パスワードを指定する文字列式です。
注意: ログイン時に、Account 情報が必要な場合は Account プロパティで設定をしてください。	

■戻り値

なし。

Cwd メソッド

■機 能

CWD コマンドだけを発行するメソッドです。従来の Cd メソッドは、FTP の CWD コマンドを発行したあと、PWD コマンドを発行していました。システムの設定によっては、PWD コマンドに失敗することもあり、そのため、CWD コマンドは成功したが、PWD コマンドに失敗し結果として、Cd メソッドに失敗するということがありました。このような状況に対応するには、このメソッドを使ってください。

ただし、IsIdle メソッドを使って、コマンド発行可能状態であるかどうか調べてから呼び出してください。コマンド発行可能状態で無い場合に呼び出すと、エラー(エラー番号 "17")になります。処理がおわると、CommandDone イベントが発生します。

■構 文

Object.Cwd(Directory As String)

Cwd メソッドの構文の指定項目は次のとおりです。

(指定項目)	(内 容)
Object	FTPOCX オブジェクトです。
Directory	ディレクトリのパスを指定する文字列式です。

■戻り値

なし。

Cd メソッド

■機 能

CWD コマンドを発行して、カレントディレクトリを変更します。Directory に空文字列"" を与えると失敗します。空文字列の場合は、"."を与えてください。CWD に成功したら、PWD コマンドが続いて発行されます。IsIdle メソッドを使って、コマンド発行可能状態であるかどうか調べてから呼び出してください。コマンド発行可能状態で無い場合に呼び出すと、エラー(エラー番号"17")になります。処理がおわると、CommandDone イベントが発生します。CommandDone イベントの Msg パラメータにカレントディレクトリのパスが格納されています。パスは、"で囲まれています。

■構 文

Object.Cd(Directory As String)

Cd メソッドの構文の指定項目は次のとおりです。

(指定項目)	(内 容)
Object	FTPOCX オブジェクトです。
Directory	ディレクトリのパスを指定する文字列式です。

■戻り値

なし。

Delete メソッド

■機 能

DELE コマンドを発行して、リモートファイルを削除します。Delete メソッドはファイルにのみ適用できます。ディレクトリを削除するには RmDir メソッドを使ってください。IsIdle メソッドを使って、コマンド発行可能状態であるかどうか調べてから呼び出してください。コマンド発行可能状態で無い場合に呼び出すと、エラー(エラー番号"17")になります。処理がおわると、CommandDone イベントが発生します。

■構 文

Object.Delete (File As String)

Delete メソッドの構文の指定項目は次のとおりです。

(指定項目)	(内 容)
Object	FTPOCX オブジェクトです。
File	削除するファイルを指定する文字列式です。

■戻り値

なし。

Dir メソッド

■機 能

LIST コマンドを発行して、ディレクトリリストをとります。File はディレクトリまたはファイル名を指定します。省略すると、カレントディレクトリを指定したことになります。IsIdle メソッドを使って、コマンド発行可能状態であるかどうか調べてから呼び出してください。コマンド発行可能状態で無い場合に呼び出すと、エラー(エラー番号"17")になります。このメソッドが成功すると、DirOpen/DirData/DirClose イベントが発生します。処理がおわると、CommandDone イベントが発生します。

■構 文

Object.Dir([File As String])

Dir メソッドの構文の指定項目は次のとおりです。

(指定項目)	(内 容)
Object	FTPOCX オブジェクトです。
Files	リストをとりたいディレクトリまたはファイル名を指定する文字列式です。

■戻り値

なし。

Disconnect メソッド

■機 能

QUIT コマンドを発行して FTP セッションを切断します。IsIdle メソッドを使って、コマンド発行可能状態であるかどうか調べてから呼び出してください。コマンド発行可能状態で無い場合に呼び出すと、エラー（エラー番号“17”）になります。処理がおわると、CommandDone イベントが発生します。また、FTP セッションが切断されると、Disconnected イベントが続いて発生します。abend メソッドを発行した場合の Code は -1 となります。

■構 文

Object. Disconnect()

Disconnect メソッドの構文の指定項目は次の通りです。

(指定項目)	(内 容)
Object	FTPOCX オブジェクトです。

■戻り値

なし。

GetFile メソッド

■機 能

RETR コマンドを用いてリモートファイルを取り出します。FileType プロパティ、FileCode プロパティに応じてコード変換が行われます。IsIdle メソッドを使って、コマンド発行可能状態であるかどうか調べてから呼び出してください。コマンド発行可能状態で無い場合に呼び出すと、エラー(エラー番号"17")になります。このメソッドが成功すると、GetOpen/GetData/GetClose イベントが発生します。処理がおわると、CommandDone イベントが発生します。現バージョンでは、LocalFile は GetOpen イベントの SaveFileName パラメータとして使われるだけで、これを指定しても、このファイルに自動的にセーブされるわけではありません。

■構 文

Object.GetFile(File As String, [LocalFile As String])
GetFile メソッドの構文の指定項目は次のとおりです

(指定項目)	(内 容)
Object	FTPOCX オブジェクトです。
File	取り出したいリモートファイルを指定する文字列式です。
LocalFile	GetOpen イベントの SaveFileName パラメータとして使われる文字列式です。

■戻り値

なし。

Ls メソッド

■機 能

NLST コマンドを発行して、ディレクトリリストをとります。File はディレクトリまたはファイル名を指定します。省略すると、カレントディレクトリを指定したことになります。IsIdle メソッドを使って、コマンド発行可能状態であるかどうか調べてから呼び出してください。コマンド発行可能状態で無い場合に呼び出すと、エラー（エラー番号“17”）になります。このメソッドが成功すると、DirOpen/DirData/DirClose イベントが発生します。処理がおわると、CommandDone イベントが発生します。

■構 文

Object.Ls([File As String])

Ls メソッドの構文の指定項目は次のとおりです。

(指定項目)	(内 容)
Object	FTPOCX オブジェクトです。
File	リストをとりたいディレクトリまたはファイル名を指定する文字列式です。

■戻り値

なし。

IsIdle メソッド

■機 能

メソッドが実行中かどうか調べます。True であれば、メソッドは実行中ではありません。False であれば、接続していないか、メソッドが実行中ですので、新たにメソッドを実行することはできません。CommandDone イベントを待ってからメソッドを発行してください。

■構 文

Value=Object.IsIdle()

IsIdle メソッドの構文の指定項目は次のとおりです。

(指定項目)	(内 容)
Object	FTPOCX オブジェクトです。
Value	メソッドが実行中かどうか示すブール式です。

■戻り値

戻り値(Boolean)が示す値は次のとおりです。

(値)	(説 明)
True	メソッドは実行中ではありません。
False	接続していないか、メソッドが実行中です。 いずれにせよメソッドを実行することはできません。

MkDir メソッド

■機 能

MKD コマンドを発行して、ディレクトリを作成します。IsIdle メソッドを使って、コマンド発行可能状態であるかどうか調べてから呼び出してください。コマンド発行可能状態で無い場合に呼び出すと、エラー(エラー番号"17")になります。処理がおわると、CommandDone イベントが発生します。

■構 文

Object.MkDir(Directory As String)

MkDir メソッドの構文の指定項目は次のとおりです。

(指定項目)	(内 容)
Object	FTPOCX オブジェクトです。
Directory	作成するディレクトリを指定する文字列式です。

■戻り値

なし。

Noop メソッド

■機 能

NOOP コマンドを発行します。FTP コネクションがタイムアウトで切れないようにする為に用います。IsIdle メソッドを使って、コマンド発行可能状態であるかどうか調べてから呼び出してください。コマンド発行可能状態で無い場合に呼び出すと、エラー(エラー番号"17")になります。処理がおわると、CommandDone イベントが発生します。

■構 文

Object.Noop()

Noop メソッドの構文の指定項目は次のとおりです。

(指定項目)	(内 容)
Object	FTPOCX オブジェクトです。

■戻り値

なし。

PutFile メソッド

■機 能

STORコマンドを用いてリモートファイルに格納します。FileType プロパティ、FileCode プロパティに応じてコード変換が行われます。IsIdle メソッドを使って、コマンド発行可能状態であるかどうか調べてから呼び出してください。コマンド発行可能状態で無い場合に呼び出すと、エラー（エラー番号“17”）になります。このメソッドが成功すると、PutData/PutClose イベントが発生します。処理がおわると、CommandDone イベントが発生します。

■構 文

Object. PutFile(File As String)

PutFile メソッドの構文の指定項目は次のとおりです。

(指定項目)	(内 容)
Object	FTPOCX オブジェクトです。
File	格納するリモートファイル名を指定する文字列式です。

■戻り値

なし。

Pwd メソッド

■機 能

カレントディレクトリを絶対パスで返します。IsIdle メソッドを使って、コマンド発行可能状態であるかどうか調べてから呼び出してください。コマンド発行可能状態で無い場合に呼び出すと、エラー(エラー番号"17")になります。処理がおわると、CommandDone イベントが発生します。

■構 文

Object.Pwd()

Pwd メソッドの構文の指定項目は次のとおりです。

(指定項目)	(内 容)
Object	FTPOCX オブジェクトです。

■戻り値

なし。

Quote メソッド

■機 能

FTPOCX がサポートしていない FTP プロトコルコマンドを FTP サーバーに送出することができます。IsIdle メソッドを使って、コマンド発行可能状態であるかどうか調べてから呼び出してください。コマンド発行可能状態で無い場合に呼び出すと、エラー(エラー番号"17")になります。正しく送出されると、CommandDone イベントが発生します。

■構 文

Object.Quote(ExtraCommand As String)

Quote メソッドの構文の指定項目は次のとおりです。

(指定項目)	(内 容)
Object	FTPOCX オブジェクトです。
ExtraCommand	FTPOCX がサポートしていない FTP プロトコルコマンドを指定する文字列式です。
注意: データ転送を行うことはできません。	

■戻り値

なし。

Rename メソッド

■機 能

RNFR/RNTO コマンドを発行して、ファイル名を変更します。FileTo が存在すると名前の変更はできません。IsIdle メソッドを使って、コマンド発行可能状態であるかどうか調べてから呼び出してください。コマンド発行可能状態で無い場合に呼び出すと、エラー（エラー番号“17”）になります。処理がおわると、CommandDone イベントが発生します。

■構 文

Object.Rename(FileFrom As String, FileTo As String)

Rename メソッドの構文の指定項目は次のとおりです。

(指定項目)	(内 容)
Object	FTPOCX オブジェクトです。
FileFrom	変更したいファイル名を指定する文字列式です。
FileTo	変更するファイル名を指定する文字列式です。

■戻り値

なし。

Rmdir メソッド

■機 能

RMD コマンドを発行して、ディレクトリを削除します。削除できるのは、ディレクトリが空である場合だけです。IsIdle メソッドを使って、コマンド発行可能状態であるかどうか調べてから呼び出してください。コマンド発行可能状態で無い場合に呼び出すと、エラー(エラー番号"17")になります。処理がおわると、CommandDone イベントが発生します。

■構 文

Object.Rmdir(Directory As String)

Rmdir メソッドの構文の指定項目は次のとおりです。

(指定項目)	(内 容)
Object	FTPOCX オブジェクトです。
Directory	削除するディレクトリ名を指定する文字列式です。

■戻り値

なし。

Size メソッド

■機 能

SIZE コマンドを発行して、File のファイルサイズを調べます。ファイルのサイズは、FileType に依存しますので、このメソッドを発行する前に、FileType プロパティを設定しておいてください。IsIdle メソッドを使って、コマンド発行可能状態であるかどうか調べてから呼び出してください。コマンド発行可能状態で無い場合に呼び出すと、エラー（エラー番号"17"）になります。処理がおわると、CommandDone イベントが発生します。Msg パラメータにファイルサイズが格納されています。

■構 文

Object. Size(File As String)

Size メソッドの構文の指定項目は次のとおりです。

(指定項目)	(内 容)
Object	FTPOCX オブジェクトです。
File	サイズを調べるファイル名を指定する文字列式です。

■戻り値

なし。

Site メソッド

■機 能

SITE コマンドを発行して、FTP サーバーに固有のコマンドを実行するよう指定します。UNIX では、chmod がサポートされることがあります。IsIdle メソッドを使って、コマンド発行可能状態であるかどうか調べてから呼び出してください。コマンド発行可能状態で無い場合に呼び出すと、エラー（エラー番号“17”）になります。処理がおわると、CommandDone イベントが発生します。

■構 文

Object.Site(LocalCommand As String)

Site メソッドの構文の指定項目は次のとおりです。

(指定項目)	(内 容)
Object	FTPOCX オブジェクトです。
LocalCommand	サイト固有のコマンドを指定する文字列式です。

■戻り値

なし。

Login メソッド

■機 能

Connect メソッドでサーバーに接続したが、ログインに失敗した場合に、接続を切らずに再度ログインするためのメソッドです。IsIdle メソッドを使って、コマンド発行可能状態であるかどうか調べてから呼び出してください。コマンド発行可能状態で無い場合に呼び出すと、エラー(エラー番号"17")になります。処理がおわると、CommandDone イベントが発生します。

■構 文

Object.Login(Username As String, Password As String)

Login メソッドの構文の指定項目は次のとおりです。

(指定項目)	(内 容)
Object	FTPOCX オブジェクトです。
Username	ユーザ名を指定する文字列式です。
Password	パスワードを指定する文字列式です。

■戻り値

なし。

イベント

イベント

白紙ページ

CommandDone イベント

■機 能

メソッドにより指定された一連の FTP コマンドを送信し終わったときに発生します。Code が 0 であれば、正常に処理されています。Code が 0 以外であれば、コマンドがエラーになっています。Code は、FTP コマンドのエラーコードで 400 番台および 500 番台の値を取ります。

(注: 1.03 では、Code に 0 以外の数値があるとすべてエラーとして、処理を中断していました。1.05 からは、実行したメソッドによっては、エラーがあっても後の処理を継続するようにしました。この修正により、LIST コマンドの実装されていない、メインフレーム系の FTP サーバーや、カレントディレクトリの読み取りを禁止しているシステムでも WILLFTP を利用することができるようになりました。)

■構 文

Private Sub Object_CommandDone(Code As Integer, Stat As String, Msg As String)
CommandDone イベントの構文の指定項目は次のとおりです。

(指定項目)	(内 容)
Object	FTPOCX オブジェクトです。
Code	実行したメソッドが成功したか失敗したかを示す整数式です。
State	実行したメソッドが格納される文字列式です。
Msg	発行したメソッドのステータスの文字列式が格納されます。

Connected イベント

■機 能

FTP サーバーと接続した時に発生します。

■構 文

Private Sub Object_Connected()

Connected イベントの構文の指定項目は次のとおりです。

(指定項目)	(内 容)
Object	FTPOCX オブジェクトです。

DirClose イベント

■機 能

すべてのディレクトリリスト受信したときに発生します。

■構 文

Private Sub Object_DirClose()

DirClose イベントの構文の指定項目は次のとおりです。

(指定項目)	(内 容)
Object	FTPOCX オブジェクトです。

DirData イベント

■機 能

ディレクトリリストを 1 行受け取ったときに発生します。data には改行コードは含まれません。

■構 文

Private Sub Object_DirData(Data As String)

DirData イベントの構文の指定項目は次のとおりです。

(指定項目)	(内 容)
Object	FTPOCX オブジェクトです。
Data	ファイル名やバイト数、日時等の情報が格納されています。(情報の内容は FTP サーバーによって異なります。)

DirOpen イベント

■機 能

Dir/ls メソッドが成功して、ディレクトリリストの送信準備が出来た時に発生します。

■構 文

Private Sub Object_DirOpen()

DirOpen イベントの構文の指定項目は次のとおりです。

(指定項目)	(内 容)
Object	FTPOCX オブジェクトです。

DisConnected イベント

■機 能

FTP セッションが切断されたときに発生します。Code が 0 であれば正常終了です。Code が 0 以外であれば、Winsock でエラーが発生しました。Code と Msg を表示してください。デバッグに役立ちます。Code は Winsock のエラーコードです。

■構 文

Private Sub Object_DisConnected(Code As Long, Msg As String)

DisConnected イベントの構文の指定項目は次のとおりです。

(指定項目)	(内 容)
Object	FTPOCX オブジェクトです。
Code	セッションが正常終了したか、異常終了したかを示す整数式が格納されます。
Msg	発行したメソッドのステータスの文字列式が格納されます。

GetOpen イベント

■機 能

GetFile メソッドが成功し、転送が始まったときに発生します。SaveFileName は、GetFile メソッドで指定された LocalFile です。通常、ファイルをここで Open します。

■構 文

Private Sub Object_GetOpen(SaveFileName As String)

GetOpen イベントの構文の指定項目は次のとおりです。

(指定項目)	(内 容)
Object	FTPOCX オブジェクトです。
SaveFileName	GetFile メソッドで指定された LocalFile 名を指す文字列式が格納されます。

GetData イベント

■機 能

データを受信したときに発生します。FileCode に“BIN”以外を指定している場合、data に含まれるのは、SJIS コードの文字列です。VB のコントロールに格納するには Strconv(data,VbUnicode)を用いて Unicode に変換してください。

■構 文

```
Private Sub Object_GetData(Data As String)
```

GetData イベントの構文の指定項目は次のとおりです。

(指定項目)	(内 容)
Object	FTPOCX オブジェクトです。
data	受信したデータの文字列式が格納されます。

GetClose イベント

■機 能

データ転送が終了した時に発生します。通常、ここでファイルを閉じます。

■構 文

Private Sub Object_GetClose()

GetClose イベントの構文の指定項目は次のとおりです。

(指定項目)	(内 容)
Object	FTPOCX オブジェクトです。

PutData イベント

■機 能

PutFile メソッドが成功し、相手が受信状態になっているのでデータをこのイベントで送信します。FileCode に“BIN”以外を指定してコード変換を行う場合は、SJIS コード、JIS コード、EUC コードのいずれかを渡してください。Unicode の場合は、data=StrConv(your-data,VbFromUnicode)を行ってください。“BIN”を指定した場合は、文字列の中味がそのまま送信されます。(Unicode の場合は Unicode のまま送信されます)送信するデータが無くなったら、data に何もセットしないでください。data に何もセットしないとデータの終わりと判断され、送信セッションが切断されます。

■構 文

Private Sub Object_PutData(Data As String)
PutData イベントの構文の指定項目は次のとおりです。

(指定項目)	(内 容)
Object	FTPOCX オブジェクトです。
data	送信するデータの文字列式を設定する変数です。

PutClose イベント

■機 能

送信が完了した時に発生します。通常ファイルの Close を行います。

■構 文

Private Sub Object_PutClose()

PutClose イベントの構文の指定項目は次のとおりです。

(指定項目)	(内 容)
Object	FTPOCX オブジェクトです。

Trace イベント

■機 能

FTP サーバーとのやり取りがあったときに発生します。これを記録しておけば、何処でエラーになったかわかり、デバッグに役立ちます。Msg は Unicode なので、そのまま VB のコントロールなどに格納できます。

■構 文

Private Sub Object_Trace(Msg As String)

Trace イベントの構文の指定項目は次のとおりです。

(指定項目)	(内 容)
Object	FTPOCX オブジェクトです。
Msg	発行したコマンドと受信したステータスの文字列式が格納されます。

TransError イベント

■機 能

TransErrorEvent=True のときに発生します。

■構 文

```
Private Sub Object_TransError (ByVal Ecode As Long, ByVal Description As String,  
ByVal Where As String)
```

TransError イベントの構文の指定項目は次のとおりです。

(指定項目)	(内 容)
Object	FTPOCX オブジェクトです。
Ecode	エラーコード
Description	説明
Where	エラーの発生した場所

イベント

白紙ページ

Winsock エラーコード

Winsock エラーコード

白紙ページ

10004	システムコールの割り込みが発生
10009	無効なソケット番号を指定した
10013	アクセスが拒否された
10014	間違ったアドレスを指定した
10022	無効な引数
10024	開いているファイル数が多すぎる
10035	ブロッキングモードなら操作がブロックした
10036	ブロッキング処理中に別の API を呼び出した
10037	非ブロッキング処理中に同じ API を呼び出した
10038	ソケット以外に対してソケット操作が行われた
10039	宛先のアドレスが必要
10040	メッセージが長すぎる
10041	ソケットに対するプロトコルの種類が不正
10042	プロトコルが使用できない
10043	プロトコルがサポートされていない
10044	ソケットの種類がサポートされていない
10045	操作がソケットでサポートされていない
10046	プロトコルファミリがサポートされていない
10047	アドレスファミリがサポートされていない
10048	アドレスがすでに使用中
10049	要求されたアドレスを割り当てられない
10050	ネットワークが落ちている
10051	ネットワークが到達できない
10052	リセットによりネットワーク接続が落とされた
10053	ホスト内部で接続を中断した
10054	相手が接続を強制的に切った
10055	利用可能なバッファスペースが存在しない
10056	ソケットがすでに接続されている
10057	ソケットが接続されていない
10058	ソケットの遮断後であるため送信できない
10059	参照数が過大:結合不可能
10060	時間切れで connect または send に失敗した
10061	接続が拒否された
10062	シンボリックリンクが多すぎる
10063	ファイル名が長すぎる
10064	ホストが落ちている
10065	ホストへの到達経路が存在しない
10091	ネットワークサブシステムが使用不能
10092	Winsock がサポートしていないバージョンを指示した
10093	Winsock が初期化されていない
10101	接続が切れている
11001	ホストが見つからない
11002	やり直し可能なエラー
11003	回復不可能なエラー
11004	データがない

Winsock エラーコード

白紙ページ

サンプル

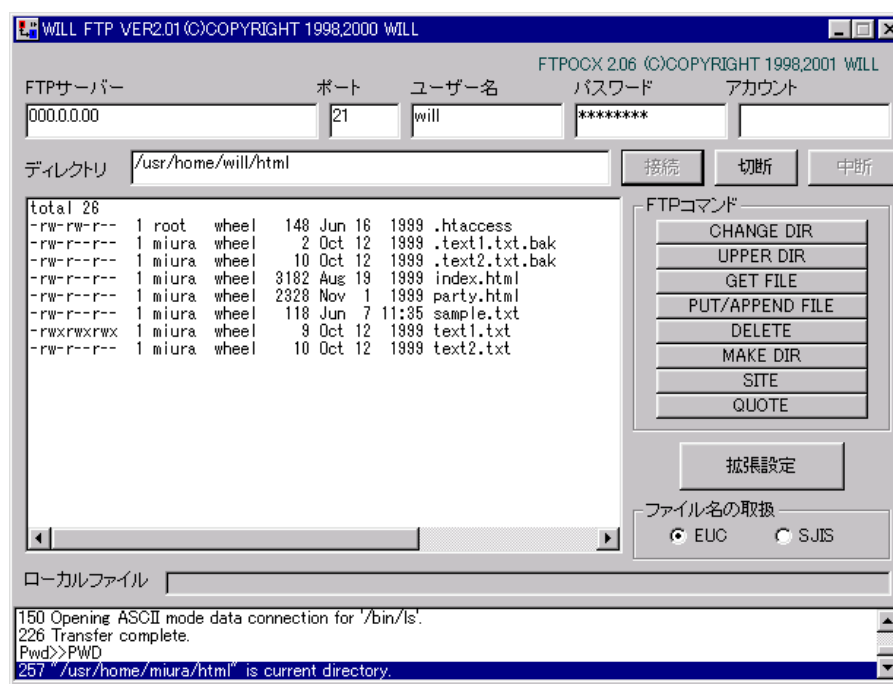
サンプル

白紙ページ

WILLFTP

WILLFTP は、FTP クライアントのプログラムです。いくつかのメソッドの機能を盛り込んだサンプルです。ファイル内容及びファイル名は送信時と受信時で別々に指定の漢字コード(SJIS/JIS/EUC)に変換可能です。

尚、WILLFTP.EXE を実行するには、FTPOCX の内部で利用する TCPIPOCX 及び KANJIPOCX が必要です。(FTPOCX のライセンスで、この2つの OCX を内部で利用することが出来ます。但し単独で利用することは出来ません。)



■使い方

＜サーバーの設定＞

- サーバー

FTP サーバーの IP アドレスを入力します。

- ユーザー名、パスワード、アカウント

サーバーログイン時のユーザー名、パスワード、アカウントを入力します。

<拡張機能の設定>

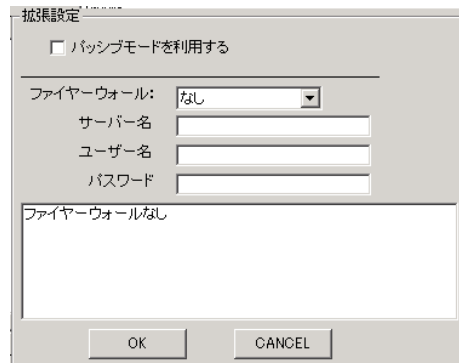
- **パッシブモードの利用**

通信を行う際にパッシブモードを利用する場合はチェックボックスにチェックをつけます。

- **ファイヤーウォールの設定**

デフォルトは、「なし」です。

設定を行う場合は、「OPEN」、「SITE」、「USER タイプ1」、「USER タイプ2」の4種類のリストの中から選択してください。それぞれの動作内容は、下のテキストボックスに表示されます。表示された内容にしたがって、「サーバー名」、「ユーザー名」、「パスワード」を指定してください。



<接続・切断・中断>

サーバーの設定が完了したら、「接続」ボタンをクリックしてサーバーに接続します。切断する場合は「切断」ボタンを、処理を中断する場合は「中断」ボタンをクリックしてください。

<FTP コマンド>

- **CHANGE DIR**

「ディレクトリ」のテキストボックスに指定したパスに移動します。

- **UPPER DIR**

開いているディレクトリの1つ上の階層に移動します。

- **GET FILE**

選択中のファイルをローカルマシンの指定したディレクトリに保存します。「LOCAL FILE DIALOG」画面が表示されますので、「ファイル名」、「転送タイプ」、「漢字コード」などを指定してください。

- **PUT/APPEND FILE**

ローカルマシンにあるファイルを、選択中のディレクトリに名前をつけて保存します。「LOCAL FILE DIALOG」画面が表示されますので、該当するファイルを選択し「ファイル名」、「転送タイプ」、「漢字コード」などを指定してください。

- **DELETE**

選択中のファイル及びディレクトリを削除します。

- **MAKE DIR**

開いているディレクトリの中に新規にディレクトリを作成します。「Make Directry」

画面が表示されますので、名前を指定してください。

- **SITE**

SITE コマンドを指定します。「SITE」画面が表示されますので、接続しているサーバー固有のコマンドを指定してください。

- **QUOTE**

FTPOCX でサポートしていない FTP プロトコルのコマンドを指定できます。「QUOTE」画面が表示されますのでコマンドを指定してください。

FTPEDITOR

FTPEDITOR は FTP クライントのプログラムです。サーバー内のファイルを検索したり、検索したテキストファイルを編集したりすることができます。

ファイル内およびファイル名は送信時と受信時で別々に指定の漢字コード(SJIS、JIS、EUC)に変換可能です。

■使い方

〈サーバーの設定〉

- **サーバー**

FTP サーバーの IP アドレスを入力します。

- **ユーザー名、パスワード**

サーバーログイン時のユーザー名、パスワードを入力します。

〈ファイルタイプの指定〉

受信するファイル内の「本文」および「ファイル名」の漢字コードを指定します。

(JIS の場合は自動判別 します。EUC,SJIS の場合は自動判別できない場合に変換するデフォルトの漢字コードとなります)。

- **ファイル名がわかっている場合**

1. 「ファイル」にファイル名を指定し、編集画面 で GET ボタンをクリックします。
2. ファイル内の本文が表示されるため、編集後 PUT ボタンをクリックします。

- **ファイル名がわからない場合**

「ファイル」にディレクトリ名を指定し、検索画面で検索ボタンをクリックします。

1. ディレクトリ名またはファイル名の一覧が表示されます。選択したい項目をダブルクリックするか、クリックして選択ボタンを押します。
2. 選択した項目がファイル名の場合、自動的に編集画面 となりファイル内の本文が表示されます。(→「ファイル名がわかっている場合」の 2.へ)
選択した項目がディレクトリ名の場合、その下の層のディレクトリ名またはファイル名の一覧が表示されます。
(→該当のファイル名が表示されまで 1.2.を繰り返す。)

<検索画面>



<編集画面>



FTP SYNC

FTP SYNC は FTPOCX のメソッドの発行を同期モードでコーディングするためのサンプルです。「Rename メソッド」を利用して、サーバーにあるファイル名の変更をします。

■使い方

<サーバーの設定>

- **サーバー**

FTP サーバーの IP アドレスを入力します。

- **ユーザー名、パスワード**

サーバーログイン時のユーザー名、パスワードを入力します。



The screenshot shows a dialog box titled "Rename (C)COPYRIGHT 1999 WILL". It contains several input fields: "サーバー" (Server) with "000.0.0.00", "ユーザー名" (Username) with "will", "パスワード" (Password) with "****", "ディレクトリ" (Directory) with "/usr/home/will/html", "元ファイル名" (Original filename) with "sample.html", and "新ファイル名" (New filename) with "sample2.txt". A "GO" button is located at the bottom right.

<ファイルの指定>

- **ディレクトリ**

名前を変更するファイルがあるディレクトリのパスを指定します。

- **元ファイル名**

該当するファイル名を指定します。

- **新ファイル名**

新規のファイル名を指定します。

<送信>

設定が完了したら、「GO」ボタンをクリックして、ファイル名を変更します。

BULK FTP

BULK FTP は、複数の宛先にファイルを送信するサンプルです。右の設定欄で設定した内容を左の一覧に追加し、一括で処理を行います。

■使い方

<サーバーの設定>

- **FTP サーバー**

FTP サーバーの IP アドレスを入力します。

- **ポート番号**

ポート番号を指定します。デフォルトでは「21」が指定されています。

- **ユーザー名、パスワード**

サーバーログイン時のユーザー名、パスワードを入力します。



<ファイルの指定>

- **送信元ファイル名**

送信元のファイルのパスを指定します。

- **送信先ファイル名**

送信先のファイルのパスを指定します。

- **ファイル種別**

送信するファイルの種別を選択します。

<設定内容の追加・更新・削除>

上記の「サーバーの設定」と、「ファイルの指定」が終わったら、「追加」ボタンをクリックして左の一覧に追加します。

また、一覧の中にある設定内容を「更新」ボタンで変更できます。該当するものを選択すると、右の設定欄に設定内容が表示されますので、内容を変更し「更新」ボタンをクリックしてください。

設定内容を削除する場合は、一覧の中から該当するものを選択し「削除」ボタンをクリックして削除します。

<転送>

それぞれの設定が完了したら、「送信」ボタンをクリックしてファイルを一括で転送します。

サンプル

白紙ページ

索引

<A>

Abend メソッド.....	45
Abort メソッド.....	46
Account プロパティ.....	31
AppendFile メソッド.....	47

<C>

Cd メソッド.....	50
CommandDone イベント.....	69
Connected イベント.....	70
Connect メソッド.....	48
Copyright プロパティ.....	32
Cwd メソッド.....	49

<D>

Delete メソッド.....	51
DirClose イベント.....	71
DirData イベント.....	72
DirOpen イベント.....	73
Dir メソッド.....	52
DisConnected イベント.....	74
Disconnect メソッド.....	53

<F>

FileCode プロパティ.....	34
FileType プロパティ.....	33

<G>

GetClose イベント.....	77
GetData イベント.....	76
GetFile メソッド.....	54
GetOpen イベント.....	75

<I>

IsIdle メソッド.....	56
------------------	----

索引

<L>

Login メソッド	66
Ls メソッド	55

<M>

Mkdir メソッド	57
------------------	----

<N>

Noop メソッド	58
-----------------	----

<P>

Passive プロパティ	36
Port プロパティ	35
PreCommand プロパティ	37
PutClose イベント	79
PutData イベント	78
PutFile メソッド	59
Pwd メソッド	60

<Q>

Quote メソッド	61
------------------	----

<R>

Rename メソッド	62
Rest プロパティ	39
Rmdir メソッド	63

<S>

Site メソッド	65
Size メソッド	64
State プロパティ	41
SysCode プロパティ	40

<T>

Trace イベント	80
TransErrorEvent プロパティ	42
TransError イベント	81

FTPOCX マニュアル

1998年1月31日 初版第1版

1999年6月1日 第2版

1999年7月1日 第3版

2001年6月15日 第4版

2003年3月10日 第5版

発行所 株式会社ウィル

住所 神奈川県横浜市保土ヶ谷区西久保町15

グランディシンヤ 302

〒240-0022

TEL: 045-338-3525

FAX: 045-338-3526

Mail-Address: info@will-ltd.co.jp

URL: <http://www.will-ltd.co.jp/>

発行者 小川 史彦

本紙の内容を許可なく複写、転載、データファイル化することを禁じます。

本紙の内容に関するご質問は、上記のメールアドレス宛にお問い合わせください。
