

FTPS プロバイダ FTPS クライアント

Version 1.1.0

ユーザーズ ガイド

October 24, 2022

【備考】

【改版履歴】

バージョン	日付	内容
1.0.0	2019-02-20	初版.
1.0.1	2020-01-20	1GB のファイルを取り扱う際の不具合を修正及び、ファイル処理の速度改善. 文言修正.
1.0.2	2020-12-24	UNIX 系 OS の FTP サーバで AddFile できない不具合修正. Execute メソッドに Raw コマンド追加. Log オプション追加. 文言修正. FTP 通信ライブラリを変更.
1.0.3	2021-03-12	Controller::Execute メソッドを空のコマンド名で実行すると落ちる不具合を修正.
	2021-12-31	Controller::AddFile メソッドの Path オプションの説明を追記.
1.1.0	2022-10-24	TLS 1.1 および TLS 1.2 に対応. SecurityProtocol オプションを追加.

【対応機器】

機種	バージョン	注意事項

【動作確認機器】

機種	バージョン	注意事項
IIS FTP	IIS10.0	
vsftpd	3.0.3	
KEYENCE XG-8000		LIST, MDTM, SIZE コマンドに対応してないため, MLSD コマンドの結果から最終更新日時とファイルサイズを取得しています.

目次

1. はじめに	5
2. プロバイダの概要	6
2.1. 概要	6
2.2. メソッド・プロパティ	7
2.2.1. CaoWorkspace::AddController メソッド	7
2.2.2. CaoController::AddFile メソッド	8
2.2.3. CaoController::Execute メソッド	9
2.2.3.1. Raw コマンド	9
2.2.4. CaoFile::AddFile メソッド	11
2.2.5. CaoFile::Copy メソッド	11
2.2.6. コピー先の指定方法	11
2.2.6.1. 絶対パスでコピー先を指定	12
2.2.6.2. 相対パスでコピー先を指定	12
2.2.6.3. ファイル名のみを指定	13
2.2.6.4. パスの最後にデリミタ	13
2.2.7. CaoFile::Delete メソッド	14
2.2.8. CaoFile::Move メソッド	14
2.2.9. CaoFile::Execute メソッド	14
2.2.9.1. Download コマンド	15
2.2.9.2. Upload コマンド	16
2.2.9.3. Raw コマンド	18
2.2.10. CaoFile::get_FileName プロパティ	18
2.2.11. CaoFile::get_FileNames プロパティ	18
2.2.12. CaoFile::get_Attribute プロパティ	18
2.2.13. CaoFile::get_Path プロパティ	18
2.2.14. CaoFile::get_Size プロパティ	18
2.2.15. CaoFile::get_Type プロパティ	18
2.2.16. CaoFile::get_DateLastModified プロパティ	18
2.2.17. CaoFile::get_Value プロパティ	19
2.2.18. CaoFile::put_Value プロパティ	19
2.3. 変数一覧	20
2.4. エラーコード	20

3. サンプルプログラム 21

1. はじめに

本書は、FTP over SSL を用いてファイルにアクセスするためのプロバイダである FTPS プロバイダのユーザーズガイドです。

CAO の CaoFile オブジェクトと実際のファイルを 1 対 1 に対応しています。これにより CaoFile オブジェクトが実際のファイルのツリー構造を再現できるように実装されています。

本書は、この FTPS プロバイダの機能と実装されているメソッドについて説明します。

本プロバイダは FTP 通信にオープンソースのライブラリを使用しています。

[FluentFTP]

Copyright (c) 2015 Robin Rodricks and FluentFTP Contributors

URL: <https://github.com/robinrodricks/FluentFTP/blob/master/LICENSE.TXT>

このプロバイダを使用するためには、「.NET Framework 4.7.2」が必要です。

2. プロバイダの概要

2.1. 概要

FTPS プロバイダは、FTP over SSL を用いてファイルとアクセスする部分を CAO のインタフェースによってラップしています。これにより、FTP サーバ上にあることを意識することなく、ローカルに存在するファイルと同様に処理していくことができます。

FTPS プロバイダを使用するには表 2-1 の方法で登録を行う必要があります。RegistAsm.bat および UnregistAsm.bat は ORiN2SDK をインストールしたフォルダの下の DotNet¥BAT フォルダにあります。

表 2-1 FTPS プロバイダ

ファイル名	CaoProvFTPS.dll
ProgID	CaoProv.FTPS
レジストリ登録 ¹	RegistAsm.bat CaoProvFTPS.dll
レジストリ登録の抹消	UnregistAsm.bat CaoProvFTPS.dll

¹ ORiN SDK でインストールした場合は手動で登録/抹消する必要はありません。

2.2. メソッド・プロパティ

2.2.1. CaoWorkspace::AddController メソッド

このメソッドは、引数にコントローラ名と FTP サーバとの接続パラメータを指定することで、FTPS プロバイダの起動と FTP サーバへの接続を行います。以下にこのメソッドの引数仕様を示します。

```

AddController
(
    "<コントローラ名>", // コントローラ名
    "GaoProv. FTPS ", // プロバイダ名. 固定.
    "<オプション>" // オプション文字列
)
    
```

以下にオプション文字列に指定するリストを示します。

表 2-2 CaoWorkspace::AddController のオプション文字列

オプション	意味										
Server[=<サーバ名>]	FTP サーバ名. (デフォルト: "localhost")										
User[=<ユーザ名>]	FTP サーバにログインするユーザ名. (デフォルト: 空文字列)										
Password[=<パスワード>]	ユーザ名に対応するパスワード. (デフォルト: 空文字列)										
Port[=<ポート番号>]	FTP のポート番号. (デフォルト: 21) 0 を指定した場合自動判定モードとなり, 21 を指定したと判断します.										
Passive[=<True/False>]	パッシブモードの設定. True: パッシブモード False: アクティブモード (デフォルト)										
SSL[=<True/False>]	暗号化接続の設定. True: SSL/TLS 接続 (デフォルト) False: 平文で接続										
SecurityProtocol[=<使用する SSL/TLS プロトコル>]	暗号化接続を行う際に使用するプロトコル. 使用するプロトコルの値の合計を指定してください. 省略時は 15 が指定されたと判断します.										
	<table border="1"> <thead> <tr> <th>プロトコル</th> <th>値</th> </tr> </thead> <tbody> <tr> <td>SSL 3.0</td> <td>1</td> </tr> <tr> <td>TLS 1.0</td> <td>2</td> </tr> <tr> <td>TLS 1.1</td> <td>4</td> </tr> <tr> <td>TLS 1.2</td> <td>8</td> </tr> </tbody> </table>	プロトコル	値	SSL 3.0	1	TLS 1.0	2	TLS 1.1	4	TLS 1.2	8
プロトコル	値										
SSL 3.0	1										
TLS 1.0	2										
TLS 1.1	4										
TLS 1.2	8										
ReferencePath [=<相対パスの基準パス>]	Download, Upload コマンドを実行する際のローカル PC の基										

	<p>準パスを指定します。詳細はそれぞれのコマンドを参照してください。(P15, P16)</p> <p>(デフォルト:FTPS プロバイダの dll が配置されているディレクトリ)</p>
Log[=<True/False>]	<p>FTP 通信時のログ出力の設定。</p> <p>True: ログ出力する</p> <p>False: ログ出力しない(デフォルト)</p>

2.2.2. CaoController::AddFile メソッド

このメソッドではファイルオブジェクトを作成します。そのオブジェクトに対応させるファイルをファイル名として指定します。ここで指定したファイルは、ファイルオブジェクトによって形成されるツリー構造のルートディレクトリとなります。このためオプションでこのファイルの位置を示すパスを指定する必要があります。以下に AddFile の引数仕様を示します。

```

AddFile
(
    "<ファイル名>",           // ファイル名
    "<オプション>"           // オプション文字列
)
    
```

以下にオプション文字列に指定するリストを示します。

表 2-3 CaoController::AddFile のオプション文字列

オプション	意味
Path[=<パス名>]	<p>ファイル名として指定したファイルのあるディレクトリのパスを記述します。(デフォルト:FTP ルートディレクトリ)</p> <p>パスの指定は、絶対パスで指定してください。相対パスを指定した場合、AddFile メソッドは失敗します。このとき、パスのデリミタは"¥", "/"のどちらでも使用できます。</p> <p>例) ルート直下の"Sample"フォルダを指定する場合</p> <p>Path=/Sample: 成功します。</p> <p>Path=Sample: 失敗します。</p>
@Create[=<0~3>]	<p>指定したファイルがないとき、このオプション値に従ってファイルを作成します。</p> <p>0:ファイルを作成しません。(デフォルト)</p> <p>1:ファイルを作成します。</p> <p>2:ディレクトリを作成します。</p> <p>3:拡張子があればファイルを、拡張子がなければディレクトリを作成します。</p> <p>指定したファイルが存在するときはこのオプションは無視されます。</p>

	また, @Create のみを指定した場合は, 3 を指定されたと判断します.
Mode[=<0~2>]	<p>get_Value, put_Value プロパティの転送モードを指定します.</p> <p>0: 自動設定 (詳細 0, 2.2.18) (デフォルト)</p> <p>1: ASCII モード</p> <p>2: BINARY モード</p>

使用例

```
Dim engine As caoEngine
Dim workspace As caoWorkspace
Dim ctrl As CaoController
Set engine = New caoEngine
Set workspace = engine.AddWorkspace("NewWrks", "")
Set ctrl = workspace.AddController("sample", _
    "CaoProv.FTPS", _
    "", _
    "Server=FTPServer, User=User, Password=Passwd")

' ルート要素を追加
Dim root As CaoFile
Set root = ctrl.AddFile("/")

' 子ディレクトリを CaoController に追加
Dim childDir As CaoFile
Set childDir = ctrl.AddFile("childDir")

' 孫ディレクトリを CaoController に追加
Dim grandsonDir As CaoFile
Set grandsonDir = ctrl.AddFile("grandsonDir", "Path=/childDir/grandsonDir")

' 存在しないファイルをルートに追加して CaoController に追加
Dim nonFile As CaoFile
Set nonFile = ctrl.AddFile("nonFile.txt", "@Create")
```

2.2.3. CaoController::Execute メソッド

CaoController の拡張コマンドを実行します. 以下に, Execute の仕様を示します.

```
Execute
(
    "<コマンド名>", // コマンド名
    "<引数>" // 引数
)
```

以下に, Execute で指定できる拡張コマンド一覧を示します. 使用例は拡張コマンドの詳細で記述してあります.

コマンド	説明	参照
Raw	引数で渡された文字列を FTP コマンドとしてサーバに送信します.	P.9

2.2.3.1. Raw コマンド

引数で渡された文字列を FTP コマンドとして送信します. この時カレントディレクトリは FTP ルートディレクト

リです。戻り値として FTP コマンドの成功/失敗と実行時の応答メッセージを取得することができますが、ファイル一覧などの応答データを取得することはできません。

表 2-4 Raw コマンドの引数

項目	型説明	
引数	VT_BSTR	FTP コマンド文字列を指定します。
戻り値	VT_BOOL	FTP コマンドの実行結果を返します。 TRUE:FTP コマンド実行成功 FALSE:FTP コマンド実行失敗
	VT_BSTR	FTP サーバからの応答メッセージを返します。

2.2.4. CaoFile::AddFile メソッド

前述 2.2.2 と同様にファイルオブジェクトを作成します。ファイル名には、このメソッドを実行する CaoFile オブジェクトと対応しているディレクトリ内のファイルのみを指定することができます。このため前述 2.2.2 のオプションである Path オプションは無視されます。このメソッドは実行する CaoFile オブジェクトがディレクトリと対応していないときは失敗します。

2.2.5. CaoFile::Copy メソッド

このメソッドを実行すると、ファイルを指定した FTP サーバ上の場所にコピーします。オブジェクトに対応しているファイルの種類によって表 2-5 に示すように、動作が変わります。

表 2-5 対応ファイルごとの Copy メソッドの動作

対応しているファイル	Copy メソッドの動作
ファイル	ファイルをコピーします。
ディレクトリ	ディレクトリとその中のファイル、サブディレクトリをコピーします。
FTP ルートディレクトリ	コピーしないで、エラーを返します。

以下に Copy の引数仕様を示します。

```
Copy
(
    "<コピー先>",           // ファイルのコピー先
    "<オプション>"         // オプション文字列
)
```

表 2-6 CaoFile::Copy のオプション文字列

オプション	意味
Overwrite[=<True/False>]	コピー先にファイルがあるときの上書き設定を指定します。省略すると上書きモードで動作します。 True: 上書きします。 False: 自動でユニーク名前を設定します。

2.2.6. コピー先の指定方法

プロバイダで接続している FTP サーバの構成が図 2-1 の時に、/TestDir/Test.txt をそのファイルに対応する CaoFile オブジェクトの Copy メソッドを使用してコピーすることを考えます。なお、コピー先に以下の条件のものを入力したときはエラーを返します。

- コピー先とコピー元が同じとき。
- コピー先がコピー元のサブディレクトリであるとき。

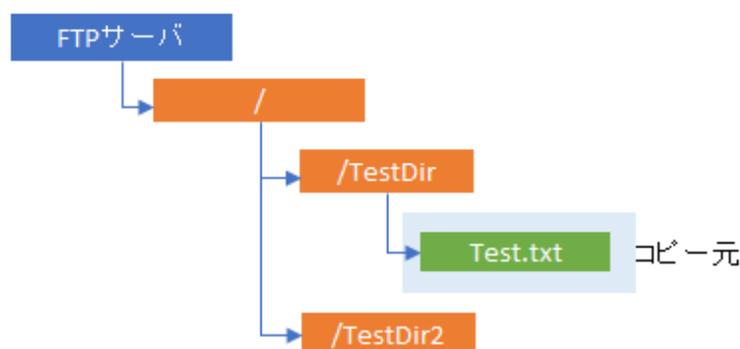
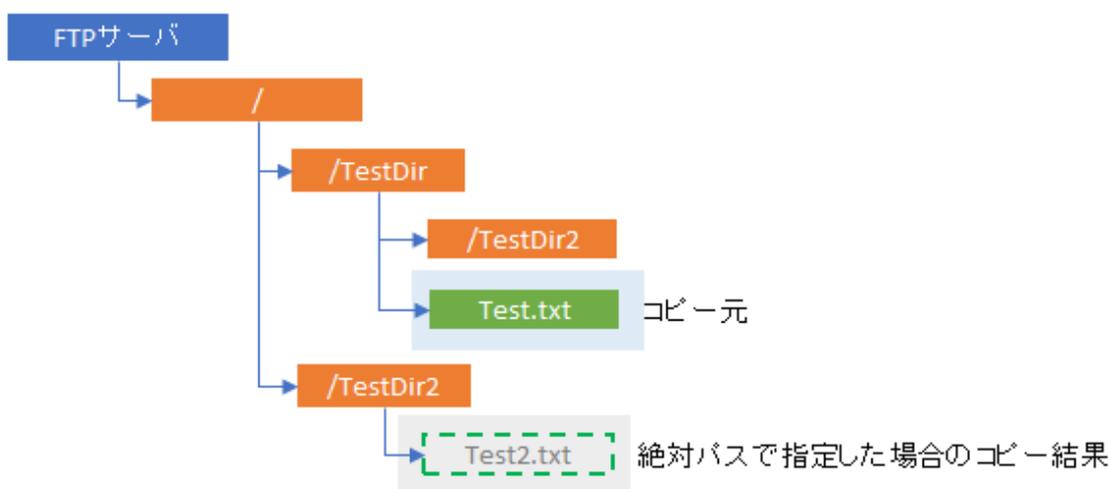


図 2-1 FTP サーバの構成

2.2.6.1. 絶対パスでコピー先を指定

絶対パスで指定した場合、コピー先に指定した場所に指定した名前でもコピーします。図 2-2 はコピー先に `/TestDir2/Test2.txt` を指定した場合のコピー結果を表しています。

図 2-2 コピー先に `"/TestDir2/Test2.txt"` を指定したコピー結果

2.2.6.2. 相対パスでコピー先を指定

コピー元のファイル・ディレクトリのあるディレクトリを基準とした場所にコピーします。図 2-3 はコピー先に `./TestDir2/Test.txt` を指定した場合のコピー結果を表しています。/TestDir の子ディレクトリである /TestDir/TestDir2/ディレクトリにファイルがコピーされます。

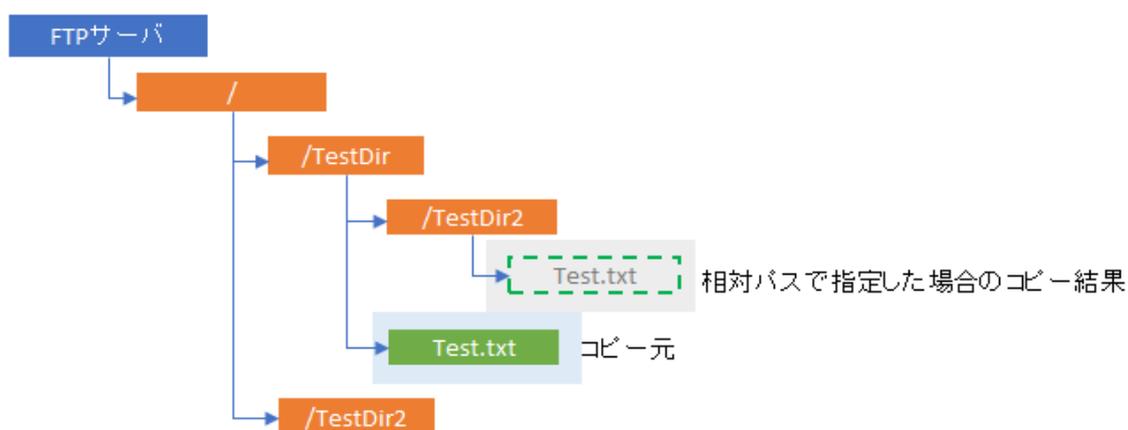


図 2-3 コピー先に“./TestDir2/Test.txt”を指定したコピー結果

2.2.6.3. ファイル名のみを指定

ファイル名のみを指定した場合は、コピー元と同じディレクトリにコピーを行います。図 2-4 はコピー先に Test2.txt を指定した場合のコピー結果を表しています。同一ディレクトリ内に Test2.txt ファイルが作成され、そこにコピーされます。

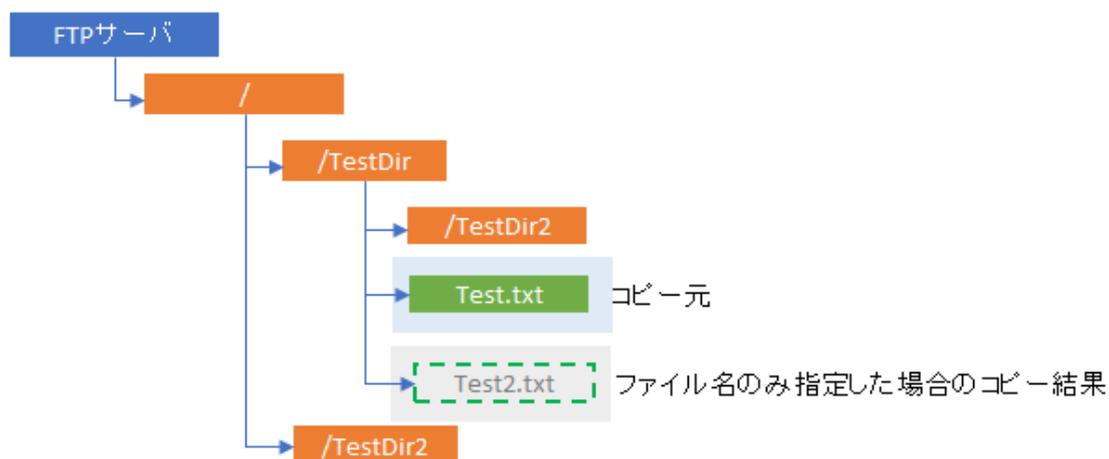


図 2-4 コピー先に“Test2.txt”を指定したコピー結果

2.2.6.4. パスの最後にデリミタ

パスの最後にデリミタを指定すると、指定されたパス以下にコピー元と同一名称でディレクトリ/ファイルを作成します。図 2-5 はコピー先に /TestDir2/ を指定した場合のコピー結果を表しています。

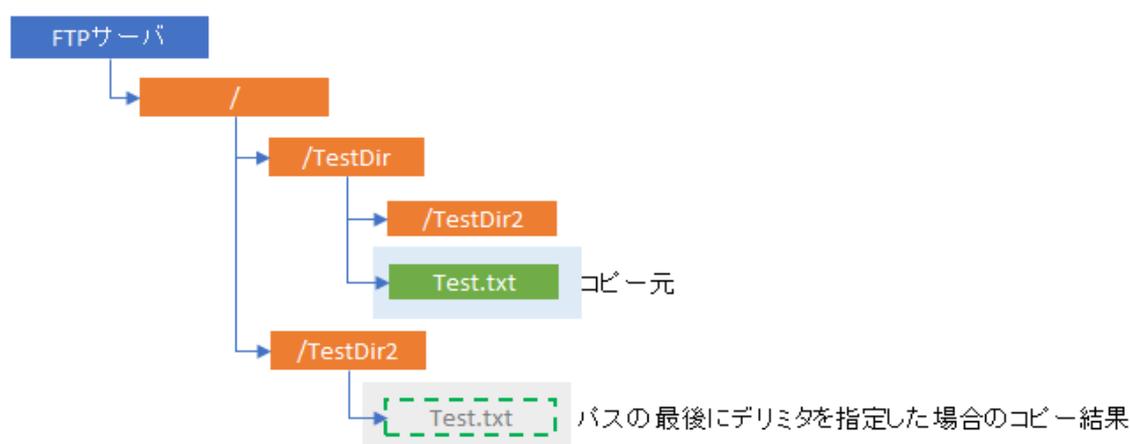


図 2-5 コピー先に"/TestDir2/"を指定したコピー結果

2.2.7. CaoFile::Delete メソッド

オブジェクトに対応しているファイルを削除します。ファイルを削除した後もオブジェクトは消去されないの
で、不必要なときはクライアントでオブジェクトを消す必要があります。

またオブジェクトに対応しているファイルが FTP ルートディレクトリのときは、その中のファイルとサブディレク
トリーのみ削除します。

FTP サーバによりファイル削除の仕様が異なるため、Delete 処理の途中で失敗することがあります²。このと
き、既に削除したファイルは戻しません。

2.2.8. CaoFile::Move メソッド

このメソッドを実行すると、ファイルを指定した FTP サーバ上の場所に移動させます。このとき移動先のパス
指定方法およびオプション文字列は Copy メソッドと同じです。

2.2.9. CaoFile::Execute メソッド

CaoFile の拡張コマンドを実行します。以下に、Execute の仕様を示します。

```
Execute
(
  "<コマンド名>",           // コマンド名
  "<引数>"                  // 引数
)
```

以下に、Execute で指定できる拡張コマンド一覧を示します。使用例は拡張コマンドの詳細で記述していま
す。

² ファイルが読み取り専用である、フォルダがワーキングディレクトリに設定されている場合に削除できないことがあります。
後者は Execute メソッドの RAW コマンドにてパラメータを"CWD 親ディレクトリのパス"として実行することで回避できます。

コマンド	説明	参照
Refresh	ファイルの属性、サイズ、更新日時の情報を更新します。更新時に実際のファイルがなくなっている場合にはエラーを返します。本コマンドには引数は必要ありません。	--
Download	CaoFile に対応したディレクトリ/ファイルを指定したパスにダウンロードします。	P.15
Upload	CaoFile に対応したディレクトリに指定したディレクトリ/ファイルをアップロードします。	P.16
Raw	引数で渡された文字列を FTP コマンドとしてサーバに送信します。ファイルに対応している CaoFile の場合にはエラーを返します。	P.18

2.2.9.1. Download コマンド

CaoFile オブジェクトに対応しているディレクトリ/ファイルを指定したローカルパスのディレクトリにダウンロードします。ディレクトリに対応している CaoFile オブジェクトの場合はそのディレクトリではなく、ディレクトリ内のすべてのサブディレクトリ、ファイルがダウンロードされます。

表 2-7 Download コマンドの引数

項目	型説明	
引数	VT_BSTR	ダウンロード先のディレクトリのローカル PC のパスを指定します。パスの指定は Copy メソッドと同一です。相対パスの基準パスは、CaoWorkspace::AddController で指定した ReferencePath オプションの値に従います。
	VT_BOOL	同一ファイルが存在していた場合に上書き保存を行うかを指定します。省略した場合は上書きモードで動作します。 False を指定しダウンロード先に同名のファイルが既に存在していた場合はダウンロードに失敗します。
戻り値	なし	

使用例

接続先の FTP サーバの構成が図 2-6 の時に、/TestDir/ディレクトリを、同名でローカルフォルダ C:\Download¥にダウンロードする例を記述します。

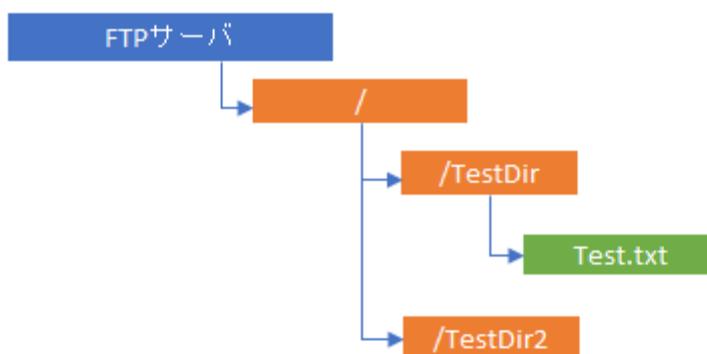


図 2-6 FTP サーバの構成

```

Set ctrl = workspace.AddController("sample", _
    "CaoProv.FTPS", _
    "" _
    "Server=FTPServer, User=User, Password=Passwd, ReferencePath=C:¥Download¥")
    
```

' 子ディレクトリを CaoController に追加

```

Dim childDir As CaoFile
Set childDir = ctrl.AddFile("childDir")
    
```

' C:¥Download¥に同名でダウンロード

```

Call childDir.Execute("Download", ". /")
    
```

2.2.9.2. Upload コマンド

CaoFile オブジェクトに対応しているディレクトリに、指定したローカルパスのディレクトリ/ファイルをアップロードします。ファイルに対応している CaoFile オブジェクトでは実行できません。

表 2-8 Upload コマンドの引数

項目	型説明	
引数	VT_BSTR	アップロードしたいディレクトリ/ファイルのローカル PC のパスを指定します。パスの指定には絶対パスと相対パスの両方が指定できます。相対パスの基準パスは、CaoWorkspace::AddController で指定した ReferencePath オプションの値に従います。
	VT_BOOL	同一ファイルが存在していた場合に上書き保存を行うかを指定します。省略した場合は上書きモードで動作します。 False を指定した場合はユニークな名前を自動で設定するように動作します。
戻り値	なし	

アップロード元の指定方法でアップロードする場所は以下のようになります。

表 2-9 アップロード元の指定方法

アップロード元の指定方法	結果	例:指定方法	アップロード結果
絶対パス	絶対パスで指定したディレクトリ/ファイルを CaoFile に対応するディレクトリに追加します。	C:/SourceDir	/TestDir/SourceDir
相対パス	現在のディレクトリからの相対パスで指定したディレクトリ/ファイルを CaoFile に対応するディレクトリに追加します。	./SourceFile.txt	/TestDir/SourceFile.txt

このとき CaoFile に対応しているディレクトリのパスは"/TestDir"とします。

使用例

接続先の FTP サーバの構成が図 2-7 の左辺時に、C:¥Download¥TestDir¥Test.txt ファイルを、/UploadedDir/ディレクトリにアップロードする例を記述します。サンプルコードを実行すると FTP サーバの構成は図 2-7 の右辺のようになります。

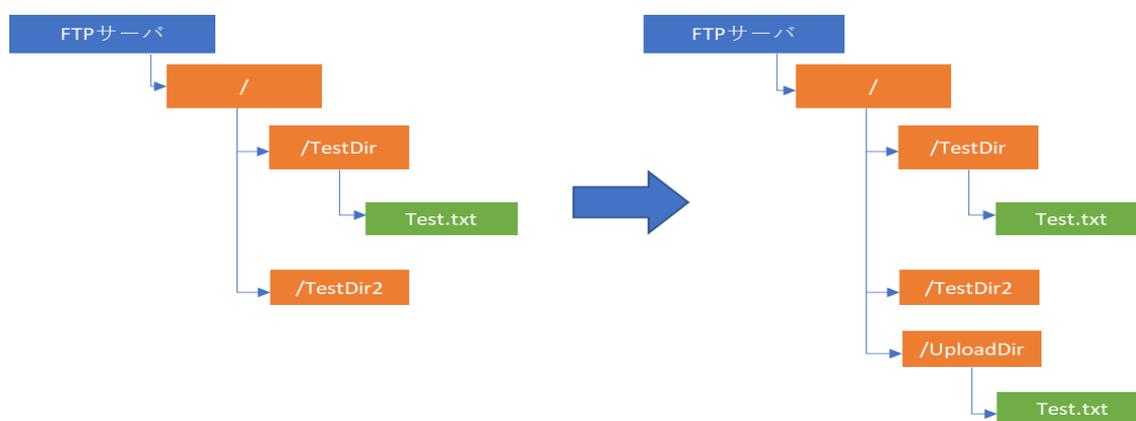


図 2-7 Upload 実行前(左)と Upload 実行後(右)の FTP サーバの構成

```

Set ctrl = workspace.AddController("sample", _
    "CaoProv.FTPS", _
    "", _
    "Server=FTPServer, User=User, Password=Passwd, ReferencePath=C:¥Download¥")

' UploadDir を作成して CaoController に追加
Dim uploadDir As CaoFile
Set uploadDir = ctrl.AddFile("UploadDir", "@Create")

' UploadDir に Test.txt をアップロード
Call uploadDir.Execute("Upload", ".¥TestDir¥Test.txt")
    
```

2.2.9.3. Raw コマンド

引数で渡された文字列を FTP コマンドとして送信します。ファイルに対応している CaoFile の場合にはエラーを返します。

FTP コマンド送信時のカレントディレクトリは CaoFile に対応したディレクトリです。引数と戻り値は CaoController::Execute メソッドの Raw コマンドと同じです(2.2.3.1)。

2.2.10. CaoFile::get_FileName プロパティ

CaoFile オブジェクトに対応しているディレクトリ/ファイル名を取得します。

2.2.11. CaoFile::get_FileNames プロパティ

CaoFile オブジェクトに対応しているディレクトリの中にあるサブディレクトリ/ファイル名リストを取得します。CaoFile オブジェクトに対応しているのがファイルのときはこのプロパティは失敗します。

2.2.12. CaoFile::get_Attribute プロパティ

GetAttribute メソッドでは、オブジェクトに対応しているのがファイルか、ディレクトリかを判別します。

表 2-10 GetAttribute で取得できる属性とその値

属性名	値
ファイル	0x080
ディレクトリ	0x010

2.2.13. CaoFile::get_Path プロパティ

オブジェクトに対応しているファイルのパスを取得します。取得した値にファイル名は含まれません。

2.2.14. CaoFile::get_Size プロパティ

オブジェクトに対応しているファイルのサイズをバイト単位で取得します。ディレクトリの時は失敗します。

2.2.15. CaoFile::get_Type プロパティ

オブジェクトに対応しているファイルの拡張子を取得します。ディレクトリの時は失敗します。

2.2.16. CaoFile::get_DateLastModified プロパティ

オブジェクトに対応しているファイルの最終更新日時を取得します。対応ファイルがディレクトリの時は失敗します。

2.2.17. GaoFile::get_Value プロパティ

オブジェクトに対応しているファイルの内容を取得します。ファイルの拡張子が"txt"のときは文字列、それ以外のときはバイナリ配列で内容を取得します。ディレクトリのときは失敗します。また、本プロパティでは、文字列取得の際、ファイルサイズが 100MB 以下、バイナリ配列で取得の際は、ファイルサイズが 200MB 以下で取得してください。ファイルサイズが大きいとメモリアオーバーしますのでご注意ください。

FTP サーバとの転送モードは AddFile メソッドのオプション文字列で Mode オプションを指定します。Mode オプションの値と転送モードの関係は表 2-11 のようになります。

表 2-11 "Mode"プロパティと GetValue の転送モードの動作

"Mode"プロパティ値	転送モード
0	対象ファイルのファイルタイプにより転送モードを設定します。 テキストファイル :ASCII モード テキストファイル以外 :BINARY モード
1	ASCII モード
2	BINARY モード

2.2.18. GaoFile::put_Value プロパティ

オブジェクトに対応しているファイルの内容を設定します。クライアントはプロバイダに対し文字列、バイナリ配列のどちらでデータを送っても設定することができます。ディレクトリのときは失敗します。

FTP サーバとの転送モードは AddFile メソッドのオプション文字列で Mode オプションを指定します。Mode オプションの値と転送モードの関係は以下ようになります。

表 2-12 "Mode"プロパティと PutValue の転送モードの動作

"Mode"プロパティ値	転送モード
0	クライアントからのデータ型により転送モードを設定します。 文字列型 :ASCII モード バイナリ配列:BINARY モード
1	ASCII モード
2	BINARY モード

2.3. 変数一覧

FTPS プロバイダ固有の変数はありません。

2.4. エラーコード

FTPS プロバイダでは、以下の固有エラーコードが定義されています。ORiN2 共通エラーについては、[「ORiN2 プログラミングガイド」](#)のエラーコードの章を参照してください。

表 2-13 独自エラーコード一覧

エラー名	エラー番号	説明
E_CAOP_FILENOTFOUND	0x80110002	ファイルが見つかりません。
E_CAOP_SOURCEDIRECTORY	0x80110003	コピー先とコピー元が同じとき。 移動先と移動元が同じとき。 コピー先がコピー元のサブディレクトリであるとき。 移動先が移動元のサブディレクトリであるとき。

3. サンプルプログラム

以下に FTP サーバからファイル内容の取得, 設定をするサンプルを示します.

サーバ名 : "FTPServer"
ファイルパス : "/SampleDir/Sample.txt"
ユーザ名 : "User"
パスワード : "Password"

List 3-1 Sample.frm

```
Private eng As CaoEngine
Private ctrl As CaoController
Private file As CaoFile

Private Sub Form_Load()

    Dim ws As CaoWorkspace

    Set eng = New CaoEngine
    Set ws = eng.Workspaces(0)

    ' FTP サーバとの接続
    Set ctrl = ws.AddController("sample", _
                               "CaoProv.FTPS", _
                               "", _
                               "Server=FTPServer, User=User, Password=Password")

    ' ファイルの追加
    Set file = ctrl.AddFile("Sample.txt", "Path=/SampleDir")
End Sub

' ファイル内容の設定
Private Sub Command1_Click()
    file = Text1.Text
End Sub

' ファイル内容の取得
Private Sub Command2_Click()
    Text2.Text = file
End Sub
```