



# F\*TRAN2007<sub>Ver.2.0</sub>

---

操作説明書／プログラム応用編

---

第1版

株式会社富士通ビー・エス・シー

## はじめに

F\*TRAN2007 V2.0をお買い上げいただき、誠にありがとうございます。

F\*TRAN2007は、汎用機やオフコンなど、ホストコンピュータの標準フロッピーディスク形式であるIBM形式フロッピー（IBMファイル）と、パソコンの標準であるWindowsファイルを相互に変換するためのソフトウェアです。

おもな用途としては、汎用機やオフコンのCOBOLデータと、パソコンのCSV形式ファイルとの交換・プリント形式ファイルとの交換・Windows COBOLデータとの交換などがあります。また、COBOLのゾーン形式・パック形式をはじめ、豊富なコード変換機能をサポートしています。

日本で販売されたほとんどのホストの漢字をサポートしており、拡張漢字テーブルも標準提供しています。

F\*TRAN2007 V2.0では、Windows 7上での動作をサポートしています。

F\*TRAN2007のマニュアルには、導入編、解説編、コマンド編、マルチレコード編、プログラム応用編（本書）があります。

2010年4月  
株式会社 富士通ビー・エス・シー

# 目 次

## 第1章 OLEオートメーション

1.1 OLEオートメーションとは？	2
--------------------	---

## 第2章 インターフェース

2.1 インターフェース	4
■ インターフェース	4
■ 全体の流れの参考図	6
■ IFtCmdのメソッド	7
■ IFtSinkのメソッド (コールバック)	13
■ IFtQuitのメソッド	18
■ データ形式	19
■ メッセージボックス、ダイアログボックスの種類	20
■ メッセージボックス、ダイアログボックスのボタンタイプ	22
■ メッセージボックス、ダイアログボックスの戻り値	22
2.2 リターンコード	23

## 第3章 オートメーションサンプル

3.1 Visual C++のサンプル	26
3.2 Visual Basicのサンプル	34

## 本書で用いる表記法

### ●本文と画面のパラメータ類の表記法

{ A   B   C }	A、B、またはCのうち、どれか1つを選択します。省略はできません。
$\left  \begin{array}{c} A \\ B \\ C \end{array} \right $	同上。
( A / B / C )	同上。
[ A ]	Aは省略できます。
[ A / <u>B</u> / C ]	A、B、またはCのうち、どれか1つを選択します。省略が可能で、その場合、下線を引いたBを選択したものとみなします。
$\left[ \begin{array}{c} A \\ \underline{B} \\ C \end{array} \right]$	同上。
( A / [ B ] / C )	同上。ただし、[ ] でくくったBを選択したものとみなします。
X . . .	X類を A B C のように列挙します。
n、n n、< n >	1 0進数を指定します。 ( < > は表記上の記号で、入力はしません)
x x H	1 6進で x x です。Hを省くこともあります。
↓	改行を意味します。リターンキーのシンボルです。
<u>a</u>	下線部を入力します。
<u>a b c</u> ↓	下線部を入力し、リターンキーを押します。
CTRL + A	コントロール ( CTRL ) キーを押しながら、Aキーを押します。コントロールAと読みます。
^ A	同上。
d :	ドライブ A : や C : など、任意のドライブ指定を表します。

### ◆注意 ----- 実画面と少し差異がある

本書に示す画面と実際の画面には、若干の差異がある場合があります。あらかじめ、ご了承ください。

## 第1章

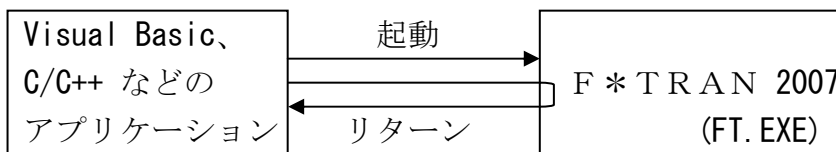


# OLEオートメーション

## 1. 1 OLEオートメーションとは？

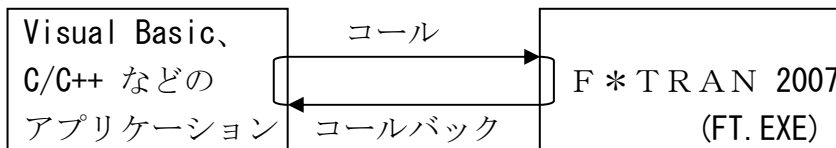
F\*TRAN2007では、OLEオートメーションのインターフェースが提供されています。このインターフェースをアプリケーションから利用することで、F\*TRAN2007をユーザアプリケーションの「部品」として使用することができ、F\*TRAN2007の処理を細かにコントロールすることができます。

### ●OLEオートメーションを使わないF\*TRAN2007の利用（従来の方法）



F\*TRAN2007を外部プログラムとして起動し、プロセス終了後のリターン（正常、エラーのどちらか）を取得します。細かいコントロールはできません。非同期の動作となります。

### ●OLEオートメーションを使ったF\*TRAN2007の利用



F\*TRAN2007の処理を自関数としてコールし、メッセージやダイアログのコールバックを受け取ることもできます。詳細なエラーコードの判定ができ、F\*TRAN2007をアプリケーションの一部として使用することができます。同期した動作となります。

## 第2章



# インターフェース

## 2. 1 インターフェース

F\*TRAN2007のOLEオートメーションのインターフェースは、つぎのとおりです。

### ■ インターフェース

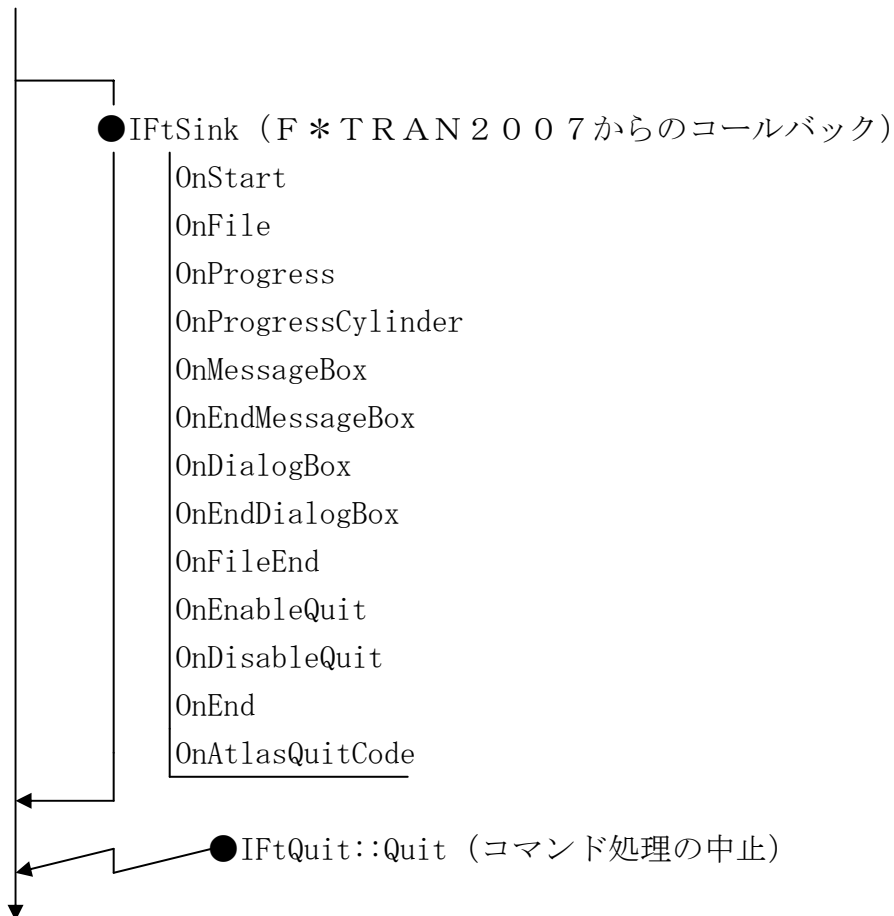
● IFtCmd	クライアントとのインタフェース
メソッド	
Command	コマンドラインを実行し、終了後に復帰する。
GetLastFtErrorCode	F*TRAN2007 のリターンコードを取得する。
GetLastWinError	Win32 エラーまたは OLE エラーの最終値を取得する。
GetLastMessage	実行ウインドウに出力された最終エラーメッセージを取得する。
SetOwnerWindow	オーナーウインドウを設定する。
MessageBoxValue	OnMessageBox の戻り値を設定する。 OnMessageBox の中で使用する。
DialogBoxValue	OnDialogBox の戻り値を設定する。 OnDialogBox の中で使用する。
SelectWFile	ファイル選択ウインドウから選択したファイルを通知する。
SetCurrentDirectory	カレントディレクトリを設定する。
GetCurrentDirectory	カレントディレクトリを取得する。
GetInstallDirectory	F*TRAN2007 のインストールディレクトリを取得する。
SelectIFile	IBM ファイル 選択ウインドウから選択した IBM ファイル を通知する。
GetList	最後に実行された iList コマンドの結果を取得する。
GetVolumeLabel	現在のボリュームラベルを CSV 形式で取得する。
GetAtlasQuitCode	最後に実行した Quit 文の値 (SysQuit) を取得する。



<b>●IFtSink</b>		コールバックのインタフェース
メソッド		
OnStart		コマンド開始のタイミングで呼ばれる。
OnEnd		コマンド終了のタイミングで呼ばれる。
OnProgress		処理中の%進行表示のために呼ばれる。
OnMessageBox		メッセージボックス表示の直前に呼ばれる。 MessageBoxValue で戻り値が設定されると、 F*TRAN2007 のメッセージボックスは表示されない。
OnEndMessageBox		メッセージボックス終了直後に呼び出される。
OnDialogBox		ダイアログボックス表示の直前に呼ばれる。 DialogBoxValue で戻り値が設定されると、 F*TRAN2007 のダイアログボックスは表示されない。
OnEndDialogBox		ダイアログボックス終了直後に呼び出される。
OnFile		ファイル変換開始のタイミングで呼ばれる。 CSV 形式でファイルラベル情報が渡される。
OnFileEnd		ファイル変換終了のタイミングで呼ばれる。
OnDisableQuit		IBMディスク 初期化開始のタイミングで呼ばれる。
OnEnableQuit		IBMディスク 初期化終了のタイミングで呼ばれる。
OnProgressCylinder		IBMディスク 初期化/複写時のシリンダ 進行表示のために呼ばれる。
OnAtlasQuitCode		Atlas マップの Quit 値に合わせて。
<b>●IFtQuit</b>		中止を指示するインタフェース
メソッド		
Quit		実行中のコマンドの中止を指示する。非同期。

## ■全体の流れの参考図

●IFtCmd::Command (F \* T R A N 2 0 0 7のコマンドをコール)



リターン (コマンド処理の終了)

## ■ IFtCmd のメソッド

### ● IFtCmd::Command

```
long Command(  
    BSTR pCmd  
);
```

パラメータ	pCmd	コマンドラインと同じコマンド文字列を指定する。
リターン	0	正常
	0 以外	エラー
説明		コマンドの開始点である。 実行終了後に復帰する。 実行中にコールバックが呼び出される。

### ● IFtCmd::GetLastFtErrorCode

```
long GetLastFtErrorCode(  
);
```

パラメータ	なし。
リターン	F*TRAN2007 のリターンコード。
説明	F*TRAN2007 のリターンコードが得られる。 リターンコードは Command メソッド終了後、または、OnEnd コールバック時に有効である。

### ● IFtCmd::GetLastWinError

```
long GetLastWinError(  
);
```

パラメータ	なし。
リターン	Win32 エラーまたは OLE エラー。
説明	Win32 エラーまたは OLE エラーの最終値が得られる。 リターンコードは Command メソッド終了後、または、OnEnd コールバック時に有効である。

● IFtCmd::SetOwnerWindow

```
long SetOwnerWindow(  
    long hWnd  
);
```

パラメータ hWnd オーナーウインドウのハンドル。

リターン 0 正常  
0 以外 エラー

説明 実行ウインドウのオーナーウインドウを設定する。  
実行ウインドウが表示されないときには、  
メッセージボックスやダイアログボックスのオーナーウインドウとなる。  
オーナーウインドウを設定しないと、デスクトップがオーナーウインドウ  
になる。  
Command、SelectWFile、SelectIFile の呼び出し前に設定する。  
また、hWnd が無効になったときは hWnd を 0 として呼び出す。

● IFtCmd::GetLastMessage

```
BSTR GetLastMessage(  
);
```

パラメータ なし。

リターン 実行ウインドウに出力された最終エラーメッセージ。

説明 実行ウインドウの最後のメッセージが得られる。  
最終的に正常でも途中でエラーがあったときには、そのエラーメッセージ  
になる。  
メッセージボックスに表示されたエラーメッセージを、このメソッドで  
得ることはできない。  
最終エラーメッセージは Command メソッド終了後、または、OnEnd コール  
バック時に有効である。

● IFtCmd::MessageBoxValue

```
long MessageBoxValue(
    long lId
);
```

パラメータ	lId	メッセージボックスの終了コードを指定する。 0 は、このメソッド呼び出しなしと同じである。
リターン	0	
説明		OnMessageBox 内部で使用する。 メッセージボックスの戻り値を F*TRAN2007 に知らせる。 F*TRAN2007 は、その値に対応する処理に制御を移す。 このメソッドが呼び出されないとき、または lId に 0 が設定されると、 F*TRAN2007 は、自身のメッセージボックスを表示する。

● IFtCmd::DialogBoxValue

```
long DialogBoxValue(
    long lId,
    BSTR lpEdit
);
```

パラメータ	lId	ダイアログボックスの戻り値を指定する。 0 は、このメソッド呼び出しなしと同じである。
	lpEdit	NULL は、OnDialogBox 呼び出しのときの値を意味する。 ダイアログボックスの種類（後述）参照。
リターン	0	
説明		OnDialogBox 内部で使用する。 ダイアログボックスの戻り値を F*TRAN2007 に知らせる。 F*TRAN2007 は、その値に対応する処理に制御を移す。 このメソッドが呼び出されないとき、または lId に 0 が設定されると、 F*TRAN2007 は、自身のダイアログボックスを表示する。

● IFtCmd::SelectWFile

```
BSTR SelectWFile(
    BSTR lpTitle,
    BSTR lpDrive,
    BSTR lpFilter,
    long lIndex
);
```

パラメータ	lpTitle	ウィンドウのタイトルを指定する。
	lpDrive	ファイルのあるドライブのパス名を指定する。
	lpFilter	選択するファイルの種類をファイルフィルタで指定する。 例: "F*TRAN2007 スクリプトファイル(*.ft5) *.ft5 "
	lIndex	ファイルフィルタの何番目を規定値とするかを指定する。
リターン		選択されたファイル名。 キャンセルまたはエラーのときには、""となる。
説明		ファイル選択ウィンドウが表示される。 オペレータが選択したファイル名が復帰情報となる。 復帰情報として返されたファイルが必ず存在するとは限らないので注意すること。

● IFtCmd::SetCurrentDirectory

```
long SetCurrentDirectory(
    BSTR lpPath
);
```

パラメータ	lpPath	カレントとして設定するパス名
リターン		正常終了の場合は 0 。 エラーのときには、GetLastError の値。
説明		Command メソッド呼び出し前に設定する。実行途中で変更したときの結果は予想できない。

● IFtCmd::GetCurrentDirectory

```
BSTR GetCurrentDirectory(
);
```

パラメータ		なし。
リターン		取得したカレントディレクトリ。 エラーのときには、"" を返す。

● IFtCmd::GetInstallDirectory

```
BSTR GetInstallDirectory(  
);
```

パラメータ なし。

リターン 取得した F\*TRAN2007 のインストールディレクトリ。

● IFtCmd::SelectIFile

```
BSTR SelectIFile(  
    BSTR lpTitle,  
    BSTR lpDrive  
);
```

パラメータ lpTitle ウィンドウのタイトルを指定する。

lpDrive フロッピーディスクのあるドライブの文字列を指定する。

リターン 選択されたファイル名。

キャンセルまたはエラーのときには、""となる。

説明 IBMファイル選択ウィンドウが表示される。

オペレータが選択したファイル名が復帰情報となる。

● IFtCmd::GetList

```
BSTR GetList(  
);
```

パラメータ なし。

リターン 実行結果が文字列であるコマンドの実行結果。

説明 最後に実行された iList コマンドの結果を獲得する。

● IFtCmd::GetVolumeLabel

```
BSTR GetVolumeLabel(  
);
```

パラメータ なし。

リターン ボリュームラベルの文字列表示。

説明 現在のボリュームラベルを C S V 形式で獲得する。

● IFtCmd::GetAtlasQuitCode

```
long GetAtlasQuitCode(  
);
```

パラメータ なし。

リターン 最後に実行した Quit 文の戻り値。

説明 最後に実行した Quit 文の値 (SysQuit) を取得する。



## ■ IFtSink のメソッド (コールバック)

### ● IFtSink::OnStart

```
void OnStart(  
);
```

パラメータ なし。

リターン なし。

説明 Command 呼び出し直後に呼び出される。  
実行ウインドウが表示される前のタイミングである。

### ● IFtSink::OnEnd

```
void OnEnd(  
    long lReturnCode  
);
```

パラメータ lReturnCode Command のリターンコードと同じ。

リターン なし。

説明 Command の終了時に呼び出される。

### ● IFtSink::OnProgress

```
void OnProgress(  
    long lPercent  
);
```

パラメータ lPercent 1-100 までの進行状況である。

リターン なし。

説明 処理中の%進行表示のために呼ばれる。  
呼ばれる率を制御することはできない。

● IFtSink::OnMessageBox

```
void OnMessageBox(
    long lFtFlag,
    long hWnd,
    BSTR lpText,
    BSTR lpCaption,
    long lType
);
```

パラメータ	lFtFlag	メッセージボックスの種類
	hWnd	オーナーウィンドウ
	lpText	テキストへのポインタ
	lpCaption	タイトルへのポインタ
	lType	メッセージボックスのボタンタイプ

リターン

なし。

説明

F\*TRAN2007 がメッセージボックスを表示する直前に呼び出される。MessageBoxValue を呼び出し、メッセージボックスの戻り値を設定すると、F\*TRAN2007 はメッセージボックスを表示しない。  
C 言語のクライアントで Win32 関数::MessageBox() を lFtFlag を除いたパラメータで呼び出せば、F\*TRAN2007 と同じメッセージボックスになる。

● IFtSink::OnEndMessageBox

```
void OnEndMessageBox(
    long lId
);
```

パラメータ	lId	メッセージボックスの戻り値である。
-------	-----	-------------------

リターン

なし。

説明

メッセージボックス終了直後に呼び出される。

●IFtSink::OnDialogBox

```
void OnDialogBox(
    long lFtFlag,
    long hWnd,
    BSTR lpText,
    BSTR lpCaption,
    long lType
);
```

パラメータ	lFtFlag	ダイアログボックスの種類
	hWnd	オーナーウィンドウ
	lpText	テキストへのポインタ
	pCaption	タイトルへのポインタ
	lType	ダイアログボックスのボタンタイプ
リターン	なし。	
説明	ダイアログボックス表示の直前に呼ばれる。 DialogBoxValue を呼び出し、ダイアログボックスの戻り値を設定すると、 F*TRAN2007 はダイアログボックスを表示しない。	

●IFtSink::OnEndDialogBox

```
void OnEndDialogBox(
    long lId
);
```

パラメータ	lId	ダイアログボックスの戻り値である。
リターン	なし。	
説明	ダイアログボックス終了直後に呼び出される。	

●IFtSink::OnFile

```
void OnFile(
    BSTR lpFileLabel
);
```

パラメータ	lpFileLabel	C S V形式でファイルラベル情報が渡される。 内容は iList コマンドに準じる。
リターン	なし。	
説明	ファイル変換開始のタイミングで呼ばれる。 ファイル変換を伴わないコマンドでは、呼び出しはない。	

● IFtSink::OnFileEnd

```
void OnFileEnd(  
);
```

パラメータ なし。

リターン なし。

説明 ファイル変換終了のタイミングで呼ばれる。  
ファイル変換を伴わないコマンドでは、呼び出しはない。

● IFtSink::OnDisableQuit

```
void OnDisableQuit(  
);
```

パラメータ なし。

リターン なし。

説明 I B Mディスク初期化開始のタイミングで呼ばれる。  
F\*TRAN2007 は、I B Mディスク初期化中、Quit を無効にする。

● IFtSink::OnEnableQuit

```
void OnEnableQuit(  
);
```

パラメータ なし。

リターン なし。

説明 I B Mディスク初期化終了のタイミングで呼ばれる。

● IFtSink::OnProgressCylinder

```
void OnProgressCylinder(  
    long lCylinder,  
    long lHead  
);
```

パラメータ lCylinder シリンダ番号

lHead ヘッド番号

リターン なし。

説明 I B Mディスク初期化／複写中のシリンダ進行表示のために呼ばれる。

● IFtSink::OnAtlasQuitCode

```
void OnAtlasQuitCode (  
    long lValue  
);
```

パラメータ lValue Quit 文の戻り値 (SysQuit)

リターン なし。

説明 Atlas マップの中で Quit 文が呼ばれるたびに実行される。

## ■ IFtQuit のメソッド

### ● IFtQuit::Quit

```
long Quit(  
);
```

パラメータ    なし。

リターン        0

説明            中止を指示する。

このメソッドは中止を指示するだけなので、実際にコマンドが中止されるのは、このメソッドから復帰したときではない。

## ■ データ形式

### ● 一般規則

データは全体、""内部ともに常に固定長である。

""内部の空白は詰めない。

""内部以外に空白は置かない。

""内部の情報はすべて半角文字であり、漢字は含まれない。

### ● iList のデータ形式 (( )の数値は""内部の長さ)

"IBMファイル名", "ボリューム番号", "レコード長", "ブロック長", "形式", "フラグ",

(17) (4) (5) (5) (4) (4)

"BOE", "EOE", "EOD", "作成日", "レコード件数", "満了日" CR(0DH) LF(0AH)

(5) (5) (5) (6) (6) (6)

⋮

### ● GetVolumeLabel のデータ形式 (( )の数値は""内部の長さ)

"ボリュームID", "所有者名", "媒体種別", "タイプ"

(6) (14) (11) (12)

## ■メッセージボックス、ダイアログボックスの種類

FT_MB_QUERY	一般的な問い合わせ。(注：参照)
FT_MB_RETRY	リトライの問い合わせの表示。
FT_MB_FATAL	処理中断の表示。
FT_MB_MV_CHANGE_VOLUME	メディアの交換指示 (マルチボリューム)。
FT_MB_CHANGE_VOLUME	メディアの交換指示 (マルチボリューム以外)。
FT_DLG_GETPUT_SKIP	変換時の問合わせ・確認。
FT_DLG_DELETE_SKIP	I B Mファイル削除の問合わせ・確認。
FT_DLG_DCFORMAT_SKIP	I B Mディスク複写時の OK、初期化スキップ、キャンセル。
FT_DLG_DCFORMAT_ERROR	I B Mディスク複写時の再試行、初期化スキップ、キャンセル。
FT_DLG_FORMAT_INPUT	I B Mディスク初期化時のパラメータ入力。

lpText: "RTEXT", "ドライブ(&D) : "CR LF  
 "COMBOBOX", "A", "B"CR LF  
 "RTEXT", "タイプ' (&T) : "CR LF  
 "COMBOBOX", "一般", "東芝"CR LF  
 "RTEXT", "メディア(&M) : "CR LF  
 "COMBOBOX", "2HD-256", "2HD-512", "2HD-512", ..., CR LF  
 "RTEXT", "ボリューム I D (&V) : "CR LF  
 "EDITTEXT", "FTRAN"CR LF  
 "RTEXT", "所有者名(&O) : "CR LF  
 "EDITTEXT", ""CR LF  
 "RTEXT", "ベリファイ(&E) : "CR LF  
 "CHECKBOX", "0"CR LF  
 "RTEXT", "クイックフォーマット(&Q) : "CR LF  
 "CHECKBOX", "0"CR LF

lpEdit: "COMBOBOX", "0"CR LF  
 "COMBOBOX", "0"CR LF  
 "COMBOBOX", "0"CR LF  
 "EDITTEXT", "FTRAN"CR LF  
 "EDITTEXT", ""CR LF  
 "CHECKBOX", "0"CR LF  
 "CHECKBOX", "0"CR LF

FT\_DLG\_FORMAT\_OK\_CANCEL I B Mディスク初期化時のパラメータ確認。

lpText: "RTEXT", "ドライブ : "CR LF  
 "LTEXT", "A"CR LF  
 "RTEXT", "タイプ : "CR LF  
 "LTEXT", "一般"CR LF  
 "RTEXT", "メディア : "CR LF



```
"LTEXT", "2HD-256" CR LF
"RTEXT", "ボリューム I D : " CR LF
"LTEXT", "FTRAN" CR LF
"RTEXT", "所有者名 : " CR LF
"LTEXT", "" CR LF
"RTEXT", "ベリファイ : " CR LF
"LTEXT", "OFF" CR LF
"RTEXT", "クイックフォーマット : " CR LF
"LTEXT", "OFF" CR LF
```

lpEdit: NULL

FT\_DLG\_FORMAT\_OK IBMディスク初期化時のパラメータ表示。

```
lpText: "RTEXT", "ドライブ : " CR(ODH) LF(OAH)
"LTEXT", "A" CR(ODH) LF(OAH)
"RTEXT", "タイプ : " CR(ODH) LF(OAH)
"LTEXT", "一般" CR(ODH) LF(OAH)
"RTEXT", "メディア : " CR(ODH) LF(OAH)
"LTEXT", "2HD-256" CR(ODH) LF(OAH)
"RTEXT", "ボリューム I D : " CR(ODH) LF(OAH)
"LTEXT", "FTRAN" CR(ODH) LF(OAH)
"RTEXT", "所有者名 : " CR(ODH) LF(OAH)
"LTEXT", "" CR(ODH) LF(OAH)
```

lpEdit: NULL

FT\_DLG\_W\_RERERENCE SelectWFile で表示されるダイアログ。

lpEdit: ファイルの絶対パス。

FT\_DLG\_I\_RERERENCE SelectIFile で表示されるダイアログ。

lpText: iList の形式。

lpEdit: ファイル順番 (10 進)。

注) FT\_MB\_QUERY には、つぎのようなものがある。

- 1) オペレーションの要求、MB\_OKCANCEL
- 2) エラーの表示、MB\_OK
- 3) 進行確認、MB\_OKCANCEL
- 4) 進行確認、MB\_YESNO

## ■メッセージボックス、ダイアログボックスのボタンタイプ

Win32 定義(メッセージボックス、および、ダイアログボックス)

MB_OK	戻り値	IDOK
MB_OKCANCEL	戻り値	IDOK、IDCANCEL
MB_RETRYCANCEL	戻り値	IDRETRY、IDCANCEL
MB_YESNO	戻り値	IDYES、IDNO

F\*TRAN2000 の追加定義(ダイアログボックスのみ)

FT_DLG_ALLCANCEL	戻り値	IDOK、FT_IDALL、IDCANCEL、FT_IDALLCANCEL
------------------	-----	---------------------------------------

注) OnMessageBox の lType は上記の値にメッセージボックスのスタイルのフラグがORされている。上記の値を抜き出すためにはつぎのように Win32 定義の値でマスクする必要がある。(MB\_TYPMASK & lType)

## ■メッセージボックス、ダイアログボックスの戻り値

Win32 定義(メッセージボックス、および、ダイアログボックス)

IDOK  
IDCANCEL  
IDABORT  
IDRETRY  
IDIGNORE  
IDYES  
IDNO

F\*TRAN2007 の追加定義(ダイアログボックスのみ)

FT\_IDALL  
FT\_IDALLCANCEL

## 2. 2 リターンコード

- 正常終了コード
  - 0 : 正常終了コード (E\_FT\_NORMAL)
- 再試行が可能な終了コード
  - 1001 : メディアがセットされていない (E\_FT\_DEVICE\_NOT\_READY)
  - 1002 : メディアが書き込み禁止 (E\_FT\_PROTECTED\_DISK)
- 再試行が不可能な終了コード
  - 9001 : 入力パラメータエラー (E\_FT\_PARAMETER)
  - 9002 : 使用できないときに使用した (E\_FT\_PROTOCOL)
  - 9003 : オペレータによるキャンセル (E\_FT\_CANCELED)
  - 9004 : システムのリソース不足、メモリ獲得失敗など (E\_FT\_INSUFFICIENT\_RESOURCES)
  - 9005 : システムのリソース不足以外のエラー発生 (E\_FT\_WIN\_ERROR)
  - 9006 : システムのリソース不足のエラー発生 (E\_FT\_LIMIT\_EXCEEDED)
  - 9007 : 特定できないエラー発生 (E\_FT\_UNKNOWN)
  - 9008 : 情報通知の事象発生 (E\_FT\_INFORMATION)
  - 9009 : フロッピーメディアエラー(読み込み時) (E\_FT\_NO\_DATA)
  - 9010 : フロッピーメディアエラー(読み込み時) (E\_FT\_CRC\_ERROR)
  - 9011 : フロッピードライブのエラー(I/Oエラー) (E\_FT\_IO\_ERROR)
  - 9012 : フロッピーメディアのエラー(読み込み時) (E\_FT\_INCORRECT\_DRIVE)
  - 9013 : フロッピーに入力ファイルがない (E\_FT\_NO\_SUCH\_FILE)
  - 9014 : フロッピーに出力ファイルがすでにある (E\_FT\_ALREADY\_EXISTS)
  - 9015 : ファイルがライトプロテクトである (E\_FT\_PROTECT\_FILE)
  - 9016 : フロッピーのインデックスがすでいっぱい (E\_FT\_INDEX\_FULL)
  - 9017 : フロッピーの空き領域がない (E\_FT\_NO\_SPACE)
  - 9018 : ファイルが異常な形式である (E\_FT\_INVALID\_FILE)
  - 9019 : F\*TRAN でサポートされない形式のファイル (E\_FT\_UNSUPPORTED\_FILE)
  - 9020 : フロッピーが異常な形式である (E\_FT\_INVALID\_DISK)
  - 9021 : F\*TRAN でサポートされない形式のフロッピー (E\_FT\_UNSUPPORTED\_DISK)
  - 9022 : ボリューム番号が正しくないフロッピー (E\_FT\_VOLUME\_SEQUENCE)
  - 9023 : Windows ファイルの作成でエラー発生 (E\_FT\_CREATE\_FAILED\_WIN)
  - 9024 : Windows ファイルのオープンでエラー発生 (E\_FT\_OPEN\_FAILED\_WIN)
  - 9025 : Windows ファイルのクローズでエラー発生 (E\_FT\_CLOSE\_FAILED\_WIN)
  - 9026 : Windows ファイルのリードでエラー発生 (E\_FT\_READ\_FAILED\_WIN)
  - 9027 : Windows ファイルのライトでエラー発生 (E\_FT\_WRITE\_FAILED\_WIN)

9028 : HOST	ファイルの作成でエラー発生	(E_FT_CREATE_FAILED_HOST)
9029 : HOST	ファイルのオープンでエラー発生	(E_FT_OPEN_FAILED_HOST)
9030 : HOST	ファイルのクローズでエラー発生	(E_FT_CLOSE_FAILED_HOST)
9031 : HOST	ファイルのリードでエラー発生	(E_FT_READ_FAILED_HOST)
9032 : HOST	ファイルのライトでエラー発生	(E_FT_WRITE_FAILED_HOST)
9033 :	設定ファイルの作成でエラー発生	(E_FT_CREATE_FAILED_SET)
9034 :	設定ファイルのオープンでエラー発生	(E_FT_OPEN_FAILED_SET)
9035 :	設定ファイルのクローズでエラー発生	(E_FT_CLOSE_FAILED_SET)
9036 :	設定ファイルのリードでエラー発生	(E_FT_READ_FAILED_SET)
9037 :	設定ファイルのライトでエラー発生	(E_FT_WRITE_FAILED_SET)
9038 :	設定ファイルまたは MAP 指定エラー	(E_FT_INVALID_DATA)
9039 :	設定ファイルやパラメータ設定に矛盾がある	(E_FT_INCONSISTANT)

●メッセージボックス、ダイアログボックスの類別コード

1 :	一般的な問い合わせ	(FT_MB_QUERY)
2 :	リトライの問い合わせ	(FT_MB_RETRY)
3 :	重要障害による処理中断の表示	(FT_MB_FATAL)
4 :	メディアの交換指示 (マルチボリューム)	(FT_MB_MV_CHANGE_VOLUME)
5 :	メディアの交換指示 (マルチボリューム以外)	(FT_MB_CHANGE_VOLUME)
6 :	複数ファイル変換時のスキップ確認ダイアログ	(FT_DLG_GETPUT_SKIP)
7 :	複数ファイル削除時の確認ダイアログ	(FT_DLG_DELETE_SKIP)
8 :	ディスクコピー時のフォーマット確認ダイアログ	(FT_DLG_DCFORMAT_SKIP)
9 :	ディスクコピー時のフォーマットのエラーダイアログ	(FT_DLG_DCFORMAT_ERROR)
10 :	フォーマットの確認ダイアログ	(FT_DLG_FORMAT_INPUT)
11 :	フォーマットの確認ダイアログ	(FT_DLG_FORMAT_OK_CANCEL)
12 :	フォーマットの確認ダイアログ (確認のみ)	(FT_DLG_FORMAT_OK)
13 :	Windows ファイル 一覧表示ダイアログ (SelectWFile)	(FT_DLG_W_REFERENCE)
14 :	IBM ファイル 一覧表示ダイアログ (SelectIFile)	(FT_DLG_I_REFERENCE)

●ダイアログボックスのタイプ

15 :	はい、すべて変換、いいえ、キャンセル	(FT_DLG_ALLCANCEL)
16 :	OK、すべて削除、いいえ、キャンセル	(FT_DLG_SKIPCANCEL)

●ダイアログボックスの終了コード

1006 :	すべて変換、すべて削除	(FT_IDALL)
1008 :	キャンセル	(FT_IDALLCANCEL)
1282 :	スキップ	(FT_IDSKEEP)

## 第3章



# オートメーションサンプル

## 3. 1 Visual C++のサンプル

Visual C++からF\*TRAN2007のOLEオートメーション機能を使用したサンプルプログラムについて説明します。

このサンプルプログラムはVisual C++バージョン6以上で動作します。F\*TRAN2007インストールディレクトリの中にある¥Samples¥Ole¥Vc¥OleSampleCB¥のワークスペースを開きます。

以下、サンプルソース重要部分の抜粋です。

```

////////////////////////////////////
// このサンプルプロジェクトの作成手順 (Visual C++ 6.0)

// このプロジェクトでは、F*TRAN2007のマンマシンインターフェイスを完全に隠した形
// で変換処理を行うプログラムを作成しています。エラーメッセージもF*TRAN2007のも
// のをマスクして表示しているため、ユーザーはF*TRAN2007の存在を意識することがあ
// りません。

// このプロジェクトは以下の手順で作成されました。
// 1) Microsoft Developer Studio を起動する。
// 2) 「ファイル」メニューから「新規作成」をクリックする。
// 3) 「プロジェクト」タブから「MFC AppWizard (exe)」を選択する。
// 4) 「プロジェクト名」に「OleSampleCB」と入力し「OK」を押す。
// 5) 「ダイアログベース」ラジオボタンをクリックし「次へ」を押す。
// 6) 「オートメーション」チェックボックスを有効にし「次へ」を押す。
// 7) 「次へ」「終了」「OK」の順に押し、プロジェクトを新規作成する。
// 8) 「表示」メニューから「ClassWizard」をクリックする。
// 9) 「オートメーション」タブから「クラスの追加」を押す。
// 10) 「タイプライブラリから...」をクリックし、提供テーブルファイル(ftwin.tlb)
//     をインポートする。
// 11) 「クラスの確認」ダイアログで、生成したいクラスを選択し「OK」を押す
// 12) 「クラス名」から「CFtSink」を選択し、「メソッドの追加」をクリックし、
//     すべてのコールバック定義を以下の順番通り作成する。
//     (1) void OnStart();
//     (2) void OnEnd(long lReturnCode);
//     (3) void OnProgress(long lPercent);

```

```
// (4) void OnMessageBox(long lFtFlag, long hWnd, BSTR lpText, BSTR lpCaption, long lType);
// (5) void OnEndMessageBox(long lId);
// (6) void OnDialogBox(long lFtFlag, long hWnd, BSTR lpText, BSTR lpCaption, long lType);
// (7) void OnEndDialogBox(long lId);
// (8) void OnFile(BSTR lpFileLabel);
// (9) void OnFileEnd();
// (10) void OnDisableQuit();
// (11) void OnEnableQuit();
// (12) void OnProgressCylinder(long lCylinder, long lHead);
// (13) void OnAtlasQuitCode(long lValue);
// 13) 「OK」を押し MFC ClassWizard を閉じる。
// 14) FtGuid.cpp, Ftguid.h, FtDef.h をプロジェクトに追加する。
// 15) //BEGIN_EDIT~//END_EDIT の間のコードを加える。
// 16) OnOK()、ThreadFunc() 内に以下のようにコードを実装する。
// 17) FtSink.cpp に、コールバックによる処理を実装する。
// 18) 「追加コード」の部分を記述する
// 19) 必要に応じてリソースを編集する。
// 20) 必要に応じてダイアログに機能を追加する。
// 21) 「ビルド」メニューから「リビルド」を選択し EXE を作成する。
```

```
void COleSampleCBDlg::OnOK()
{
// BEGIN_EDIT
    UINT ThreadFunc(LPVOID);

    AfxBeginThread(ThreadFunc, NULL, 0, 0, 0, 0);
    return;
// if (CanExit())
//     CDialog::OnOK();
// END_EDIT
}
```

```
void COleSampleCBDlg::OnCancel()
{
    if (CanExit())
        CDialog::OnCancel();
}
```

```
BOOL COleSampleCBDlg::CanExit()
{
    // プロキシ オブジェクトがまだ残っている場合、オートメーション
    // コントローラはこのアプリケーションをまだ保持しています。
    // ダイアログの周囲は残しますが UI は非表示になります。
    if (m_pAutoProxy != NULL)
    {
        ShowWindow(SW_HIDE);
        return FALSE;
    }
    return TRUE;
}

void COleSampleCBDlg::OnStop()
{
    // TODO: この位置にコントロール通知ハンドラ用のコードを追加してください
// BEGIN_EDIT
    long    IRc;
    IftQuit FtQuit;
    BOOL    bBool;
    CString strCode;

    ////////////////////////////////////////////////////////////////////
    // サーバとの接続処理 (必須)
    // サーバーオブジェクトを作成する
    bBool = FtQuit.CreateDispatch("Ftwin.FtQuit", 0);
    if(bBool == FALSE) {
        // エラーメッセージを表示して戻る
        MessageBox("FtQuit.CreateDispatch にてエラーが発生しました。");
        return;
    }

    ////////////////////////////////////////////////////////////////////
    // FtQuit.Quit() の呼び出し
    IRc = FtQuit.Quit();
    if(IRc != 0) {
        // サーバーオブジェクトへの接続を開放する
        FtQuit.ReleaseDispatch();
    }
}
```



```

// エラーメッセージを表示して戻る
strCode.Format("FtQuit.Quitにてエラーが発生しました。 ¥n エラー番号 : %d", IRc);
MessageBox(strCode);
return;
}

////////////////////////////////////
// サーバとの切断処理 (必須)
// サーバオブジェクトへの接続を開放する
FtQuit.ReleaseDispatch();
return;
// END_EDIT
}

// BEGIN_EDIT
LONG COleSampleCDBlg::OnProgressSet(UINT mParam, LONG lParam)
{
    UpdateData(TRUE);
    m_progressctl.SetPos(lParam);
    UpdateData(FALSE);
    return 0L;
}

UINT ThreadFunc(LPVOID lpParam)
{
    long IRc;
    long IFtErr=0;
    long IWinErr=0;
    BOOL bBool;
    DWORD dwCookie;
    DWORD dwRc;
    LPUNKNOWN pUnkSink;
    CftSink FtSink;
    CString msg;
    CString strCode;
    CString lpDir;

```

```

////////////////////////////////////
// サーバとの接続処理（必須）

// COM ライブラリの初期化（スレッドにしない場合は必要なし）
dwRc = CoInitialize(0);
if(dwRc != S_OK) {
    // エラーメッセージを表示して戻る
    MessageBox(ghWnd, "CoInitialize にてエラーが発生しました.", "メッセージ", MB_ICONINFORMATION|MB_OK);
    return FALSE;
}

// サーバーオブジェクトを作成する
bBool = FtCmd.CreateDispatch("Ftwin.FtCmd", 0);
if(bBool == FALSE) {
    // エラーメッセージを表示して戻る
    MessageBox(ghWnd, "FtCmd.CreateDispatch にてエラーが発生しました.", "メッセージ", MB_ICONINFORMATION|MB_OK);
    return FALSE;
}

pUnkSink = FtSink.GetIDispatch(FALSE);

// FtCmd.m_lpDispatch と pUnkSink の間で接続を確立します。

bBool=AfxConnectionAdvise(FtCmd.m_lpDispatch, IID_IFtSink, pUnkSink, FALSE, &dwCookie);
if(bBool == FALSE) {
    // サーバーオブジェクトへの接続を開放する
    FtCmd.ReleaseDispatch();
    // エラーメッセージを表示して戻る
    MessageBox(ghWnd, "AfxConnectionAdvise にてエラーが発生しました.", "メッセージ", MB_ICONINFORMATION|MB_OK);
    return FALSE;
}

////////////////////////////////////
// F*TRAN2007 の処理（任意）

```

```
// F*TRAN2007 のインストールディレクトリを取得する
lpDir = FtCmd.GetInstallDirectory();
if(lpDir.IsEmpty()) {
    // FtCmd.m_lpDispatch と pUnkSink の間で接続を切断します。

AfxConnectionUnadvise(FtCmd.m_lpDispatch, IID_IFtSink, pUnkSink, FALSE, dwCookie);
    // サーバーオブジェクトへの接続を開放する
    FtCmd.ReleaseDispatch();
    // エラーメッセージを表示して戻る
    strCode.Format("FtCmd.GetInstallDirectory にてエラーが発生しました。 %n エラー
番号 : %d", IRc);
    MessageBox(ghWnd, strCode, "メッセージ", MB_ICONINFORMATION|MB_OK);
    return FALSE;
}

// 取得したインストールディレクトリをカレントディレクトリとして設定する
// 変換コマンド(ここでは"putdata")において、変換元・先ファイルを絶対パスで
// 指定しない場合は、カレントディレクトリをあらかじめ指定しておく必要がある
IRc = FtCmd.SetCurrentDirectory(lpDir);
if(IRc != 0) {
    // FtCmd.m_lpDispatch と pUnkSink の間で接続を切断します。

AfxConnectionUnadvise(FtCmd.m_lpDispatch, IID_IFtSink, pUnkSink, FALSE, dwCookie);
    // サーバーオブジェクトへの接続を開放する
    FtCmd.ReleaseDispatch();
    // エラーメッセージを表示して戻る
    strCode.Format("FtCmd.SetCurrentDirectory にてエラーが発生しました。 %n エラー
番号 : %d", IRc);
    MessageBox(ghWnd, strCode, "メッセージ", MB_ICONINFORMATION|MB_OK);
    return FALSE;
}

// ダイアログのボタンの選択可/不可の切り替え
gDlg->m_okctl.EnableWindow(FALSE);
gDlg->m_cancelctl.EnableWindow(FALSE);

// F*TRAN2006 におけるオーナーウィンドウを設定する
IRc = FtCmd.SetOwnerWindow((long)ghWnd);
```

```
if(IRc != 0) {
    // FtCmd.m_lpDispatch と pUnkSink の間で接続を切断します。

AfxConnectionUnadvise(FtCmd.m_lpDispatch, IID_IFtSink, pUnkSink, FALSE, dwCookie);
    // サーバーオブジェクトへの接続を開放する
    FtCmd.ReleaseDispatch();
    // エラーメッセージを表示する
    strCode.Format("FtCmd.SetOwnerWindow にてエラーが発生しました。 ¥n エラー番号 : %d", IRc);
    MessageBox(ghWnd, strCode, "メッセージ", MB_ICONINFORMATION|MB_OK);
    // ダイアログのボタンの選択可／不可の切り替え
    gDlg->m_okctl.EnableWindow(TRUE);
    gDlg->m_cancelctl.EnableWindow(TRUE);
    gDlg->m_stopctl.EnableWindow(FALSE);
    return FALSE;
}

// ダイアログのボタンの選択可／不可の切り替え
gDlg->m_stopctl.EnableWindow(TRUE);

// 変換コマンドの呼び出し
IRc = FtCmd.Command("/nwd /wc/ putdata test.dat A: /Query");
if(IRc != 0) {
    // F*TRAN2007 の内部エラー情報を取得する
    msg = FtCmd.GetLastMessage();
    IFtErr = FtCmd.GetLastFtErrorCode();
    IWinErr = FtCmd.GetLastWinError();
    // FtCmd.m_lpDispatch と pUnkSink の間で接続を切断します。

AfxConnectionUnadvise(FtCmd.m_lpDispatch, IID_IFtSink, pUnkSink, FALSE, dwCookie);
    // サーバーオブジェクトへの接続を開放する
    FtCmd.ReleaseDispatch();
    // エラーメッセージを表示する
    if(msg.IsEmpty()) {
        if(IFtErr == E_FT_CANCELED) {
            strCode.Format("処理を中断しました。 (%d, %d)", IFtErr, IWinErr);
        }
        else {
```

```

        strCode.Format("FtCmd.Command にてエラーが発生しました。
¥n[(%d,%d)]", IFtErr, IWinErr);
    }
}
else {
    strCode.Format("%s (%d,%d)", msg, IFtErr, IWinErr);
}
MessageBox(ghWnd, strCode, "メッセージ", MB_ICONINFORMATION|MB_OK);
// ダイアログのボタンの選択可/不可の切り替え
gDlg->m_okctl.EnableWindow(TRUE);
gDlg->m_cancelctl.EnableWindow(TRUE);
gDlg->m_stopctl.EnableWindow(FALSE);
return FALSE;
}

// ダイアログのボタンの選択可/不可の切り替え
gDlg->m_okctl.EnableWindow(TRUE);
gDlg->m_cancelctl.EnableWindow(TRUE);
gDlg->m_stopctl.EnableWindow(FALSE);

////////////////////////////////////
// サーバとの切断処理 (必須)
// FtCmd.m_lpDispatch と pUnkSink の間で接続を切断します。

bBool=AfxConnectionUnadvise(FtCmd.m_lpDispatch, IID_IFtSink, pUnkSink, FALSE, dwCookie);
if(bBool == FALSE) {
    // サーバーオブジェクトへの接続を開放する
    FtCmd.ReleaseDispatch();
    // エラーメッセージを表示して戻る
    MessageBox(ghWnd, "AfxConnectionUnadvise にてエラーが発生しました。 ", "メッセージ", MB_ICONINFORMATION|MB_OK);
    return FALSE;
}

// サーバーオブジェクトへの接続を開放する
FtCmd.ReleaseDispatch();
return TRUE;
}
// EDN_EDIT

```

## 3. 2 Visual Basicのサンプル

Visual BasicからF\*TRAN2007のOLEオートメーション機能を使用したサンプルプログラムについて説明します。

このサンプルプログラムはVisual Basicバージョン6以上で動作します。F\*TRAN2007をインストールした¥Samples¥Ole¥Vb¥ディレクトリの中にツギファイルがありますので、プロジェクトを開いてください。

### Project1.vbp

以下、サンプルソースです。

- ' このサンプルプログラムはVisual Basic 6.0以上で動作します。
- ' プログラムを実行すると、F\*TRANを隠した状態で変換を実行することができます。
- ' 変換中にF\*TRANからコールバックを受け取り、独自の処理（進行状況を表示する、
- ' メッセージボックスを出す、など）をしています。また、時間のかかる変換の途中では
- ' キャンセルができるように、ここでは"Timer"を使って変換コマンドを呼び出しています。

' ----- このサンプルプロジェクトの作成手順 -----

- ' ① Visual Basic を起動する。
- ' ② 「ファイル」メニューの「新しいプロジェクト」をクリックして、
- ' 「新規作成」タブの「標準 EXE」をダブルクリックする。
- ' ③ 「プロジェクト」メニューから「参照設定」をクリックして「参照」ボタンを押し、
- ' F\*TRANの提供テーブルファイル(Ftwin.tlb)を開く。
- ' リストボックスに「Ftwin」が表示され、チェックが着いていることを確かめて、
- ' 「OK」ボタンを押す。
- ' ④ 必要に応じてフォームを編集する(ここではForm1にコマンドボタン、タイマー、
- ' ラベルを追加している)。
- ' ⑤ 以下のようにコードを実装し、「ファイル」メニューの「Project1.exeの作成」を
- ' クリックする。

' ----- コード -----

- ' F\*TRANオブジェクトへの参照を格納するための変数を宣言する
- ' F\*TRANからの通知を受け取って処理をするため、 WithEvents 変数にする必要がある

```

Private WithEvents FtCmd As Ftwin.FtCmd
Private FtQuit As Ftwin.FtQuit
Private bActive As Boolean

' F*TRAN で定義されているコードを宣言する
' コードの詳細は、添付ファイルの"FtDef.txt"を参照
Private Const FT_MB_MV_CHANGE_VOLUME As Long = 4
Private Const FT_MB_CHANGE_VOLUME As Long = 5
Private Const E_FT_CANCELED As Long = 9003

Private Sub Command1_Click()
    Dim IRc As Long
    If bActive Then
        ' 変換中止コマンドの呼び出し
        IRc = FtQuit.Quit()
        If IRc <> 0 Then
            ' エラーメッセージを表示する
            msg = "FtQuit.Quitにてエラーが発生しました。" & Chr(13) & "エラー番号:" &
IRc
            MsgBox msg, vbOKOnly, "メッセージ"
            Exit Sub
        End If
    Else
        ' 変換中か否かの切り替え
        bActive = True
        Command2.Enabled = False
        Command1.Caption = "変換中止"
        ' タイマーコントロールをオンにする
        Timer1.Interval = 10
        Timer1.Enabled = True
    End If
End Sub

Private Sub Command2_Click()
    ' サーバーオブジェクトへの接続を解放する
    Set FtQuit = Nothing
    Set FtCmd = Nothing
End Sub
End Sub

```

```
Private Sub FtCmd_OnStart()
```

```
    ' 変換開始時の処理があれば、ここに追加する
```

```
End Sub
```

```
Private Sub FtCmd_OnEnd(ByVal IReturnCode As Long)
```

```
    ' 変換終了時の処理があれば、ここに追加する
```

```
End Sub
```

```
Private Sub FtCmd_OnProgress(ByVal IPercent As Long)
```

```
    ' 進行表示のための処理があれば、ここに追加する
```

```
    ' --> Example
```

```
    ' ここではフォーム上に進行状況をパーセンテージで表示している
```

```
    Label1.Caption = "    ただいま変換中です。 " & IPercent & "%"
```

```
    ' 変換開始後にキャンセルできるように、制御をオペレーティングシステムに渡している
```

```
    DoEvents
```

```
End Sub
```

```
Private Sub FtCmd_OnMessageBox(ByVal IFtFlag As Long, ByVal hWnd As Long, ByVal lpText
As String, ByVal lpCaption As String, ByVal IType As Long)
```

```
    ' メッセージボックス表示前の処理があれば、ここに追加する
```

```
    ' --> Example
```

```
    ' ここではF*TRANのメッセージボックスの代わりに、独自のメッセージボックスを表示して
    いる
```

```
    Dim IRc As Long
```

```
    If IFtFlag = FT_MB_MV_CHANGE_VOLUME Then
```

```
        IRc = MsgBox(lpText, IType, "フロッピー交換(Multi Volume)")
```

```
    ElseIf IFtFlag = FT_MB_CHANGE_VOLUME Then
```

```
        IRc = MsgBox(lpText, IType, "フロッピー交換")
```

```
    Else
```

```
        IRc = MsgBox(lpText, IType, "メッセージ")
```

```
    End If
```

```
    FtCmd.MessageBoxValue (IRc)
```

```
End Sub
```

```
Private Sub FtCmd_OnEndMessageBox(ByVal IId As Long)
```

```
    ' メッセージボックス終了時の処理があれば、ここに追加する
```

```
End Sub
```



```
Private Sub FtCmd_OnDialogBox(ByVal IFtFlag As Long, ByVal hWnd As Long, ByVal lpText As String, ByVal lpCaption As String, ByVal IType As Long)
```

```
    ' ダイアログボックス表示前の処理があれば、ここに追加する
```

```
End Sub
```

```
Private Sub FtCmd_OnEndDialogBox(ByVal IId As Long)
```

```
    ' ダイアログボックス終了時の処理があれば、ここに追加する
```

```
End Sub
```

```
Private Sub Timer1_Timer()
```

```
    Dim IRc As Long
```

```
    Dim IFtErr As Long
```

```
    Dim IWinErr As Long
```

```
    Dim msg As String
```

```
    ' タイマーコントロールをオフにする
```

```
    Enable = False
```

```
    Interval = 0
```

```
    If bActive = True Then
```

```
        ' 変換コマンド(ここでは"putdata")において、変換元・先ファイルを絶対パスで
```

```
        ' 指定しない場合は、カレントディレクトリをあらかじめ指定しておく必要がある
```

```
        IRc = FtCmd.SetCurrentDirectory(FtCmd.GetInstallDirectory)
```

```
        If IRc <> 0 Then
```

```
            ' エラーメッセージを表示する
```

```
            msg = "FtCmd.SetCurrentDirectoryにてエラーが発生しました。" & Chr(13) & "
```

```
エラー番号:" & IRc
```

```
            MsgBox msg, vbOKOnly, "メッセージ"
```

```
            Exit Sub
```

```
        End If
```

```
        ' F*TRANにおけるオーナーウィンドウを設定する
```

```
        IRc = FtCmd.SetOwnerWindow(Form1.hWnd)
```

```
        If IRc <> 0 Then
```

```
            ' エラーメッセージを表示する
```

```
            msg = "FtCmd.SetOwnerWindowにてエラーが発生しました。" & Chr(13) & "エラ
```

```
ー番号:" & IRc
```

```
            MsgBox msg, vbOKOnly, "メッセージ"
```

```
            Exit Sub
```

```
        End If
```

```

' 変換コマンドの呼び出し
IRc = FtCmd.Command("/nwd /wc/ putdata test.dat a: /query")
If IRc <> 0 Then
    ' F*TRAN の内部エラー情報を取得する
    msg = FtCmd.GetLastMessage()
    IFtErr = FtCmd.GetLastFtErrorCode()
    IWinErr = FtCmd.GetLastWinError()
    ' エラーメッセージを表示する
    If msg = "" Then
        If IFtErr = E_FT_CANCELED Then
            msg = "処理を中断しました。" & "(" & IFtErr & ", " & IWinErr & ")"
        Else
            msg = "FtCmd.Commandにてエラーが発生しました。" & Chr(13) & "(" &
IFtErr & ", " & IWinErr & ")"
        End If
    Else
        msg = msg & "(" & IFtErr & ", " & IWinErr & ")"
    End If
    MsgBox msg, vbOKOnly, "メッセージ"
Else
    ' フォーム上に終了メッセージを表示する
    Label1.Caption = "    変換が終了しました。"
End If

' 変換中か否かの切り替え
Command1.Caption = "変換開始"
Command2.Enabled = True
bActive = False
End If
End Sub

```

```
Private Sub Form_Load()
```

```
    ' サーバーオブジェクトを作成する
```

```
    Set FtCmd = New Ftwin.FtCmd
```

```
    Set FtQuit = New Ftwin.FtQuit
```

```
    ' 時間のかかる変換中に、他の操作（キャンセルなど）を実行しようとする
```

```
    ' と、[コンポーネントを使用できません]ダイアログボックスが出て実行できない。
```

```
    ' このため、ダイアログボックスが出るまでの時間を 86400000 ミリ秒（1日）という
```

```
    ' 非現実的な値に設定しておくことによって回避している
```

```
    App.OleServerBusyTimeout = 86400000
```

```
    App.OleRequestPendingTimeout = 86400000
```

```
    ' 変換中か否かのフラグ
```

```
    bActive = False
```

```
End Sub
```

```
Private Sub Form_Unload(Cancel As Integer)
```

```
    ' サーバーオブジェクトへの接続を解放する
```

```
    Set FtQuit = Nothing
```

```
    Set FtCmd = Nothing
```

```
End Sub
```

F\*TRAN2007 V2.0 操作説明書／プログラム応用編

---

2010年 4月 第1版発行

編集・著作 株式会社 富士通ビー・エス・シー  
所在地 〒135-8300 東京都港区台場 2-3-1 トレードピアお台場

---

- Windows、MS-DOS、Visual Basic、Access、Visual C++、Excel は  
米国 Microsoft Corp.の米国およびその他の国における登録商標または商標です。
- Acrobat は Adobe Systems Inc. (アドビシステムズ社) の商標です。
- F\*TRAN は富士通ビー・エス・シーの登録商標です。
- 会社名および製品名はそれぞれ各社の商標または登録商標です。
- 本書およびシステムは、改善のため事前連絡なしに変更することがあります。
- 無断複製、および転載を禁じます。