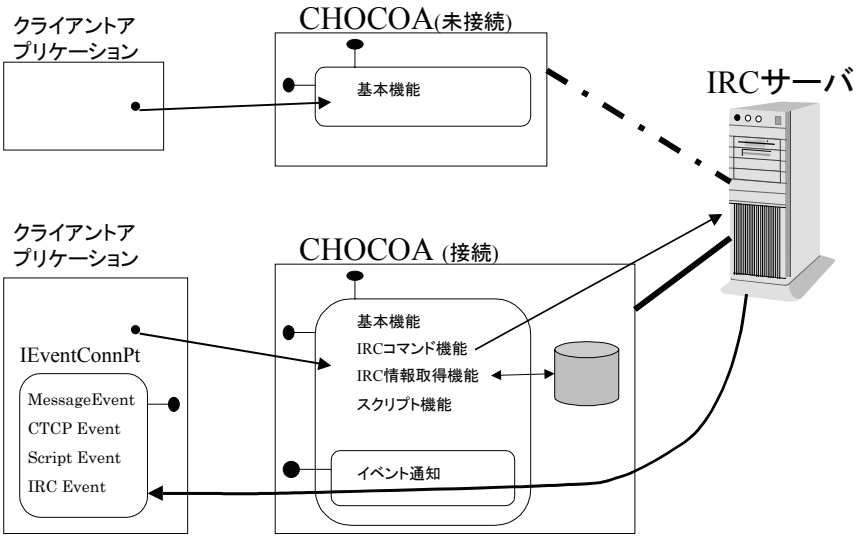
CCAPI

１．概要

CCAPI(COM 版)は、CHOCOA の機能(IRC コマンド機能、CHOCOA スクリプト機能等)を外部から操作及び取得するための COM インタフェースです。環境情報の取得・設定やウインドウの表示・非表示等の機能(基本機能)と、IRC サーバと接続した後に利用できる機能(IRC コマンド機能、IRC 情報取得機能、スクリプト機能)また、クライアントアプリケーションにイベント通知を受けるインタフェース(IEventConnPt)を実装することにより、CHOCOA で発生するイベント情報を受け取る(イベント通知機能)ことが可能になります。

- ◆ 基本機能
  IRC サーバとの接続に必要な環境情報の取得･設定、IRC サーバとの接続･切断、CHOCOA ウインドウの表示･非表示などの機能があります。
- ◆ IRC コマンド機能
  CHOCOA のコマンド機能が使用できます。
- ◆ IRC 情報取得機能
  CHOCOA の情報（ユーザーリスト、チャンネルリスト、サーバリスト）を取得することができます。
- ◆ スクリプト機能
  強力な CHOCOA スクリプトの登録、削除、実行･結果取得を行えます。
- ◆ イベント通知機能
  CHOCOA 側で発生するイベント（メッセージ（PrivMessage、CTCPMessage）･IRC 関連･スクリプト）を通知します。

※ 詳しくは、CCAPI 関数仕様書を参照してください。



© 株式会社富士通研究所., 1998,2006

２．CCAPI の利用

CCAPI は様々なメソッドが存在しますが、そのほとんどは IRC サーバと接続した後に使用可能となります。以下はＶＢで作成したサンプルソースです。

＜処理手順＞

① CCAPI インタフェースオブジェクトの作成

・インタフェースオブジェクトを生成することで、CCAPI を利用できるようになります。

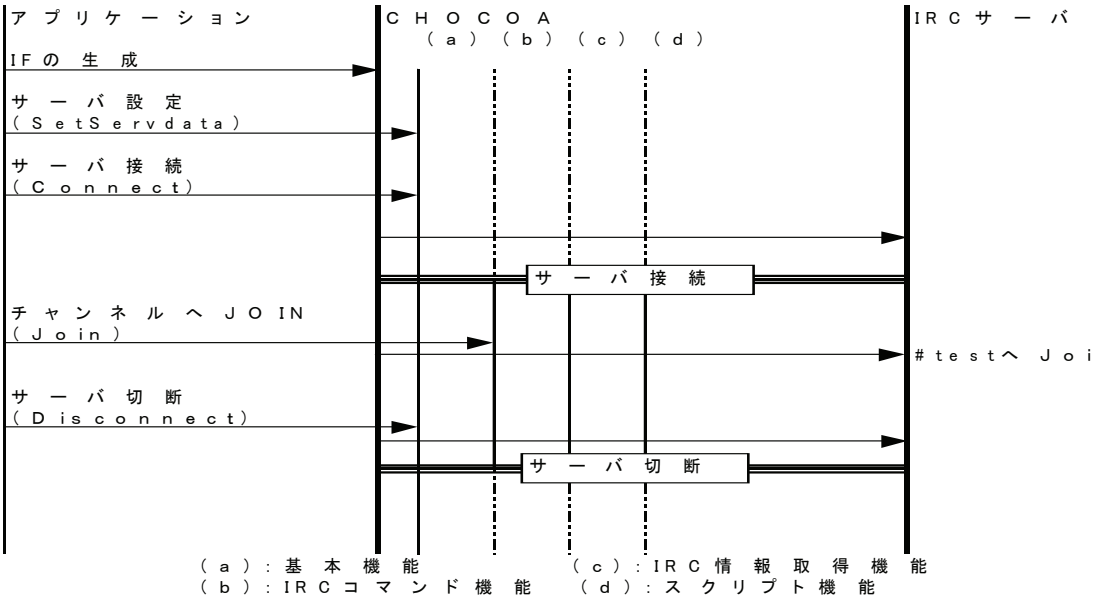
```
Dim MyObject As Object
Sub Form_Load()
        ‘ インタフェースオブジェクトを生成
        Set MyObject = CreateObject(“CHOCOA.Application”)
End Sub
```

② 接続したい IRC サーバに接続する

③ 各機能を利用する

・IRC サーバに接続するための環境設定を行い任意の IRC サーバに接続することも、また既に設定されている IRC サーバに接続することも可能です。IRC サーバと接続することで、さまざまな CCAPI の機能を利用できます。

```
Sub SetUpServer_Command()
        ‘ 下記サーバ設定で設定します。
        Dim ServerName As String
        Dim Channel As String
        ServerName = “test_server”
        SetNo = MyObject.FindServ(ServerName)
        ‘ 上記で設定したサーバに接続します。
        MyObject.Connect(SetNo)
        ‘ “#test”チャンネルに Join します。
        Channel = “#test”
        MyObject.Join(SetNo, Channel)
End Sub
```



© 株式会社富士通研究所., 1998,2006

３．WSH からの CCAPI メソッドの利用について

VBScript、JScript などのスクリプト言語（WSH : Windows Scripting Host）は、COM オブジェクトのインタフェースを利用することができます。CHOCOA が公開しているインタフェース（CCAPI）のメソッドの一部も、WSH から利用することが可能です。（利用可能メソッド一覧参照）

利用可能メソッド一覧

FindServEx	サーバ識別子の取得
IsConnect	接続状態のチェック
Connect	IRC サーバとの接続
DisConnect	IRC サーバとの切断
GetSettingCount	サーバ識別子の個数
GetSettingAtEx	識別子に対するサーバ設定名の取得
ShowWnd	ウインドウの表示（全てのウィンドウ）
ShowWndEx	ウインドウの表示（指定ウィンドウ）
HideWnd	ウインドウの非表示（全てのウィンドウ）
HideWndEx	ウインドウの非表示（指定ウィンドウ）
IsShowWnd	ウインドウ表示状態のチェック
GetVariableEx	環境変数の取得
SetVariableEx	環境変数の設定
ExecScriptEx	スクリプトコマンドの実行

※ 上記コマンドの詳細については、CCAPI 関数仕様書を参照してください。

VBScript（WSH）からの CCAPI の利用例：

以下のサンプルコードは、VBScript から CHOCOA の環境変数を変更するスクリプトです。動作としては、サーバ設定名が“WSH_TEST”で設定されているサーバ環境変数の Socks 関連の情報を設定するスクリプトです。

```
Dim Obj
Dim bRet
Dim SettingID

'CCAPI オブジェクトの Create
Set Obj = CreateObject("CHOCOA.Application")

'サーバ識別子の取得
SettingID = Obj.FindServEx("WSH_TEST")
if SettingID = 0 Then
        str = "この設定はありません"
        Counter = 1
        MsgBox str, vbOKOnly, "イントラ用設定"
End if

Do While Counter = 0
        str = "イントラ用の設定が正常に終了しませんでした"
        'サーバ環境の設定
        'Socks 使用の設定
```


© 株式会社富士通研究所., 1998,2006

```
        bRet = Obj.SetVariableEx(SettingID, "$usesocks", "1")
        if bRet = False Then
                MsgBox str, vbOKOnly, "イントラ用設定($usesocks)"
                Exit Do
        End if
        'Socks のバージョン設定
        bRet = Obj.SetVariableEx(SettingID, "$socksversion", "5")
        if bRet = False Then
                MsgBox str, vbOKOnly, "イントラ用設定($socksversion)"
                Exit Do
        End if
        'Socks サーバ名
        bRet = Obj.SetVariableEx(SettingID, "$socksserver", "socks.server.co.jp")
        if bRet = False Then
                MsgBox str, vbOKOnly, "イントラ用設定($socksserver)"
                Exit Do
        End if
        'Socks サーバのポート番号
        bRet = Obj.SetVariableEx(SettingID, "$socksport", "1080")
        if bRet = False Then
                MsgBox str, vbOKOnly, "イントラ用設定($socksport)"
                Exit Do
        End if

        Counter = 1
Loop
```


© 株式会社富士通研究所., 1998,2006

CCAPI 関数仕様書
（本仕様書のサンプルソースは、Visual Basic V5.0 で作成したものです）

＜基本機能＞
CHOCOA を利用する上で必要な基本的な機能を提供します。

名 称：
GetVersion
書 式：
BOOL GetVersion (BSTR* ***Version*****)**
機 能：
CHOCOA のバージョンを取得します。
パラメータ：
BSTR * *Version* OUT CHOCOA のバージョン
戻り値：
正常：TRUE

© 株式会社富士通研究所., 1998,2006

名 称：

FindServ

書 式：

long FindServ(BSTR* *SettingName*)

機 能：

サーバ設定名に対応するサーバ識別子を返します。他のメソッドで使用する識別子は、このメソッドで取得しておく必要が有ります。

パラメータ：

BSTR **SettingName*　　IN　　サーバ設定名

戻り値：

正常：サーバ識別子（1 以上の値）

異常：0

© 株式会社富士通研究所., 1998,2006

名 称：

FindServEx

書 式：

long FindServEx(BSTR *SettingName*)

機 能：

サーバ設定名に対応するサーバ識別子を返します。他のメソッドで使用する識別子は、このメソッドで取得しておく必要が有ります。

（FindServ と同じ結果を返します。）

パラメータ：

BSTR *SettingName*　　IN　　サーバ設定名

戻り値：

正常：サーバ識別子（1 以上の値）

異常：0

© 株式会社富士通研究所., 1998,2006

名 称：

SetServdata

書 式：

long SetServdata(BSTR* *SettingName*, **BSTR*** *Server*, **long** *port*, **BSTR*** *Nick*, **BSTR*** *Password*, **BSTR*** *RealName*, **BSTR*** *UID*, **BOOL** *Saveflg*)

機 能：

サーバ設定情報を登録し、サーバ識別子を戻り値として返します。サーバ設定名が既に使用されている場合は、エラーとなります。

パラメータ：

BSTR*	*SettingName*	IN	サーバ設定名
BSTR*	*Server*	IN	サーバアドレス
long	*port*	IN	ポート番号
BSTR*	*Nick*	IN	ニックネーム
BSTR*	*Password*	IN	パスワード（パスワード不必要時は NULL を指定）
BSTR*	*RealName*	IN	本名
BSTR*	*UID*	IN	ユーザＩＤ（Email アドレス）
BOOL	*Saveflg*	IN	True:サーバ設定名を保存、False:サーバ設定未保存

戻り値：

正常：サーバ識別子

異常：サーバ設定名が使用済みの場合（=0）

© 株式会社富士通研究所., 1998,2006

名 称：

IsConnect

書 式：

BOOL IsConnect (long *SettingNo*)

機 能：

指定したサーバとの接続状態をチェックする。

パラメータ：

long *SettingNo*　　IN　　サーバ識別子（FindServ()の戻り値）

戻り値：

正常：TRUE

異常：FALSE（指定したサーバと未接続の場合）

備 考：

接続後有効なメソッドについては、この機能を使用し接続状態をチェックする必要があります。

© 株式会社富士通研究所., 1998,2006

名 称：

Connect

書 式：

BOOL Connect (long *SettingNo*)

機 能：

指定したサーバ設定で IRC サーバと接続する。

パラメータ：

long *SettingNo*　　IN　　サーバ識別子（FindServ()の戻り値）

戻り値：

正常：TRUE

異常：FALSE（接続できない場合、又は、既に接続されている場合）

例：

```
Dim obj As Object

Private Sub Form_Load()
    'COM サーバの起動
    Set obj = CreateObject("CHOCOA.Application")
    obj.ShowWnd
 End Sub

Private Sub ConnectCmd_Click()
    Dim servid As Long
    Dim ServName As String
    ServName = “test_server”
    servid = FindServ(ServName)          ‘ “test_server”のサーバ識別子取得
    'IRC サーバとの接続
    If obj.Connect(servid) Then          ‘ サーバとの接続
        MsgBox ("正常終了しました")
    End If
End Sub

Private Sub DisconnectCmd_Click()
    Dim servid As Long
    Dim ServName As String
    ServName = “test_server”
    servid = FindServ(ServName)          ‘ “test_server”のサーバ識別子取得
    'IRC サーバとの切断
    If obj.Disconnect(servid) Then       ‘ サーバとの切断
        MsgBox ("正常終了しました")
    End If
End Sub
```


© 株式会社富士通研究所., 1998,2006

名 称：

DisConnect

書 式：

BOOL DisConnect (long *SettingNo*)

機 能：

指定された IRC サーバを切断します。

パラメータ：

long *SettingNo*　　IN　　サーバ識別子（FindServ()の戻り値）

戻り値：

正常：TRUE

異常：FALSE

© 株式会社富士通研究所., 1998,2006

名 称：

GetSettingCount

書 式：

long GetSettingCount (void)

機 能：

サーバ識別子の個数を返します。

パラメータ：

なし

戻り値：

サーバ設定登録数を返します。

備　考：

使用方法は、GetSettingAt の例参照してください。

© 株式会社富士通研究所., 1998,2006

名 称：

GetSettingAt

書 式：

BOOL GetSettingAt (long *SettingNo*, BSTR* *SettingName*)

機 能：

サーバ識別子に対応するサーバ設定名を返します。
手動による削除などで使用できないサーバ識別子を指定した場合は、SettingName＝NULL となります。

パラメータ：

long	*SettingNo*	IN	サーバ識別子（0 < SettingNo <= GetSettingCount の戻り値）
BSTR*	*SettingName*	OUT	サーバ設定名

戻り値：

正常：TRUE
異常：FALSE（“num<0 or num>GetSettingCount での取得値”の場合。取得データが存在しない場合。）

例：

```
Option Explicit
Dim obj As Object

Private Sub Form_Load()
    Dim count As Long
    Dim cnt As Integer
    Dim tmp As String

    ‘Active オブジェクトの生成
    Set obj = CreateObject("CHOCOA.Application")

    ‘ 登録個数の取得
    count = obj.GetSettingCount
    For cnt = 1 To count
        ‘ サーバ設定名の取得
        obj.GetSettingAt cnt, tmp
        Combo1.AddItem (tmp)
    Next

End Sub
```


© 株式会社富士通研究所., 1998,2006

名 称：

GetSettingAtEx

書 式：

BSTR GetSettingAtEx (long *SettingNo*)

機 能：

サーバ識別子に対応するサーバ設定名を返します。
（GetSettingAt で返されるサーバ設定名と同じ値を返します。）

パラメータ：

long	*SettingNo*	IN	サーバ識別子（0 < SettingNo <= GetSettingCount の戻り値）

戻り値：

正常：サーバ設定名
異常：””

© 株式会社富士通研究所., 1998,2006

名 称：

ShowWnd

書 式：

void ShowWnd (void)

機 能：

非表示の全てのウインドウを表示します。

パラメータ：

なし

© 株式会社富士通研究所., 1998,2006

名 称：

HideWnd

書 式：

void HideWnd (void)

機 能：

表示されている全てのウインドウを非表示にします。

パラメータ：

なし

© 株式会社富士通研究所., 1998,2006

名 称：

IsShowWnd

書 式：

BOOL IsShowWnd (long *SettingNo*)

機 能：

指定したサーバと接続している CHOCOA ウインドウが、表示状態にあるかをチェックします。サーバ識別子に 0 を設定した場合は、全てのウインドウの状態をチェックします。1 つ以上のウインドウが表示されている場合、戻り値は“TRUE”になります。

パラメータ：

long	*SettingNo*	IN	サーバ識別子

戻り値：

正常：TRUE

異常：FALSE（非表示、または、サーバ接続がされていない場合。）

© 株式会社富士通研究所., 1998,2006

名 称：

GetVariable

書 式：

BOOL GetVariable (long *SettingNo*, BSTR* *Type*, BSTR* *Value*)

機 能：

指定したサーバ設定の指定情報を返します。取得できるデータについては、『データタイプ一覧』を参照してください。

（取得データタイプにパスワードを指定することはできません。）

パラメータ：

long	*SettingNo*	IN	サーバ識別子
BSTR*	*Type*	IN	取得データタイプ（『データタイプ一覧』参照）
BSTR*	*Value*	OUT	取得データ

戻り値：

正常：TRUE

異常：FALSE（取得データタイプに、タイプ以外を指定した場合。又は、使用不可タイプを指定した場合）

備 考：

・ 取得データには、設定した取得データタイプの型に示される値がストリング型で設定されます。

・ 以下のタイプを指定することはできません

$password（パスワード）

例：

・“test_server”に接続している CHOCOA で使用しているニックネームを取得する場合

（以下は、ＶＢでのサンプルです）

```
Private Sub cmdOK_Click()
    Dim value As String
    Dim SettingNo As Integer
    Dim ret As Integer
    Dim ServName As String
    Dim tp As String
    ServName = “test_server”
    tp = “$initnick”
   SettingNo = objct.FindServ(ServName)
   ret = objct.GetVariable(SettingNo, tp, value)
   If ret Then
        Text1 = value
   End If
End Sub
```


© 株式会社富士通研究所., 1998,2006

名 称：

GetVariableEx

書 式：

BSTR GetVariableEx (long *SettingNo*, BSTR *Type*)

機 能：

指定したサーバ設定の指定情報を返します。取得できるデータについては、『データタイプ一覧』を参照してください。

（取得データタイプにパスワードを指定することはできません。）

パラメータ：

long	*SettingNo*	IN	サーバ識別子
BSTR	*Type*	IN	取得データタイプ（『データタイプ一覧』参照）

戻り値：

正常：取得データ

異常：”（NULL）”

備 考：

・ 取得データには、設定した取得データタイプの型に示される値がストリング型で設定されます。

・ 以下のタイプを指定することはできません

$password（パスワード）

例：

・Windows Scripting Host（VBScript）での例

```
Dim strRet
Dim obj

‘CHOCOA オブジェクトの作成
Set obj = CreateObject(“CHOCOA.Application”)

strRet = obj.GetVariableEx(1, “$serverlist”)

if strRet <> “” Then
    MsgBox(strRet)
End if
```


© 株式会社富士通研究所., 1998,2006

名 称：

SetVariable

書 式：

BOOL SetVariable (long *SettingNo*, BSTR ** *Type*, BSTR ** *Value*)

機 能：

指定したサーバの指定情報のデータを変更します。変更できるデータについては、『データタイプ一覧』を参照してください。

パラメータ：

long	*SettingNo*	IN	サーバ識別子
BSTR*	*Type*	IN	変更データタイプ（『データタイプ一覧』参照）
BSTR*	*Value*	IN	変更データ

戻り値：

正常：TRUE

異常：FALSE（取得データタイプに、タイプ以外を指定した場合。又は、使用不可タイプを指定した場合）

備 考：

・設定データには、『取得データタイプ一覧』で示される型をストリング型に変換して設定してください。

・以下のタイプについては設定できません。

$serversettingname（サーバ設定名）
$browserpath（ブラウザの実行形式）
$channellogpath （チャンネルログ自動保存ディレクトリ）
$urllogpath（URL ログ自動保存ディレクトリ
$dccdir（ファイル保存ディレクトリ）

例：

・“test_server”に接続している CHOCOA で、ニックネームの表示色を設定する場合
（以下は、ＶＢでのサンプルです）

```
Private Sub cmdOK_Click()
    Dim value As String
    Dim SettingNo As Integer
    Dim UseCol As String
    Dim ServName As String
    Dim tp As String
    ServName = “test_server”
    tp = “$usenickcr”
    value = Str(RGB(0, 0, 255))        ‘ 青色を設定
    UseCol = True                      ‘ “ニックネームに色を付ける”を“ＯＮ”に設定
    SettingNo = objct.FindServ(ServName)
    objct.SetVariable(SettingNo, tp, UseCol)
    tp = “$nickcr”
    objct.SetVariable(SettingNo, tp, value)
End Sub
```


© 株式会社富士通研究所., 1998,2006

名 称：

SetVariableEx

書 式：

BOOL SetVariableEx (long *SettingNo*, BSTR *Type*, BSTR *Value*)

機 能：

指定したサーバの指定情報のデータを変更します。変更できるデータについては、『データタイプ一覧』を参照してください。

パラメータ：

long	*SettingNo*	IN	サーバ識別子
BSTR	*Type*	IN	変更データタイプ（『データタイプ一覧』参照）
BSTR	*Value*	IN	変更データ

戻り値：

正常：TRUE

異常：FALSE（取得データタイプに、タイプ以外を指定した場合。又は、使用不可タイプを指定した場合）

備 考：

・設定データには、『取得データタイプ一覧』で示される型をストリング型に変換して設定してください。

・以下のタイプについては設定できません。

$serversettingname（サーバ設定名）

$browserpath（ブラウザの実行形式）

$channellogpath （チャンネルログ自動保存ディレクトリ）

$urllogpath（URL ログ自動保存ディレクトリ

$dccdir（ファイル保存ディレクトリ）

例：

・Windows Scripting Host（VBScript）での例

```
Dim ServerList
Dim obj

‘CHOCOA オブジェクトの作成
Set obj = CreateObject(“CHOCOA.Application”)

‘ ”chocoa.ne.jp”,“test.ne.jp”,“temp.ne.jp”をサーバ群として登録
ServerList = “chocoa.ne.jp” + vbCrLf + “test.ne.jp” + vbCrLf + “temp.ne.jp”
obj.SetVariableEx(1, “$serverlist”, ServerList)
```


© 株式会社富士通研究所., 1998,2006

設定 → IRC

$serversettingname （＊1）	STRING	サーバ設定名
$servername	STRING	IRC サーバ名
$serverlist（＊8）	LIST	サーバリスト
$port	INT	ポート番号
$initnick	STRING	起動時に設定されるニックネーム
$password （＊2）	STRING	パスワード
$autoconnectonboot	BOOL	起動時に自動接続する
$autochannel （＊3）	STRING	自動接続チャンネルリスト

設定 → 一般 → ユーザ

$realname	STRING	ユーザの名前
$emailaddr	STRING	e-mail アドレス
$userinfo	STRING	ユーザ情報情報
$quitmessage	STRING	デフォルト QUIT メッセージ

設定 → 一般 → ブラウザ

$browsername （＊4）	INT	ブラウザの選択
$browserpath	STRING	ブラウザの実行形式
$autoopenurl	STRING	自動的にページを開く
$notopenprivurl	BOOL	1 対 1 で発言された URL はブラウザ起動
$browsenewwindow	BOOL	新規ウインドウに開く
$ignoreownurl	BOOL	自分の発言中の URL は抽出しない

設定 → 一般 → Telephony

$iphone	BOOL	デフォルトは Iphone
$iphonepath	STRING	InternetPhone の実行形式（ファイルパス）
$netmeetingpath	STRING	NetMeeting の実行形式（ファイルパス）

設定 → 一般 → システム

$maxbuffer	INT	バッファサイズ（最大）
$thresholdbuf	INT	バッファサイズ（削除後）
$maxhistory	INT	ヒストリの数
$pasteconfirm	BOOL	ペースト時に確認ダイアログを出す

設定 → 一般 → ディレクトリ

$autochannellog	BOOL	チャンネルログ 自動保存
$channellogpath	STRING	チャンネルログ 自動保存ディレクトリ
$autourllog	BOOL	URL ログ 自動保存
$urllogpath	STRING	URL ログ 自動保存ディレクトリ
$dccdir	STRING	ファイル受信ディレクトリ
$autoconsolelog	BOOL	コンソールのログを保存する

© 株式会社富士通研究所., 1998,2006

設定 → 右クリック

$popupmenu	STRING	メインウィンドウの右クリックメニュー
$popupmenuuuser	STRING	ユーザウィンドウの右クリックメニュー
$popupmenuchnnel	STRING	チャンネルウィンドウの右クリックメニュー

設定 → キーワード設定

$beepon	BOOL	BEEP を鳴らす
$usewavefile	BOOL	WAVE ファイルを使用する
$wavefilename	STRING	WAVE ファイルのファイル名
$raiseon	BOOL	全面に出す
$deiconify	BOOL	アイコン時は通常のサイズに戻す
$flashwindow	BOOL	アイコン時は通常のサイズに戻す
$beepignorecase	BOOL	大文字小文字の区別をしない
$keyword （＊5）	STRING	キーワード文字列

設定 → 画面スタイル → 色

$timestamp	BOOL	時刻表示に色を付ける
$timecr	COLORREF	時刻表示の表示色
$usenickname （＊6）	BOOL	ニックネームに色を付ける
$nicknamecr （＊6）	COLORREF	ニックネームの表示色
$usesyscr	BOOL	システムメッセージに色を付ける
$systemcr	COLORREF	システムメッセージの表示色
$busewindowscr	BOOL	テキスト背景を Windows の設定に依存する
$crtext	COLORREF	フォアグランドカラー
$crback	COLORREF	バックグランドカラー
$joincr	COLORREF	Join メッセージの表示色
$usejoincr	BOOL	Join メッセージに色を付ける
$partcr	COLORREF	Part メッセージの表示色
$usepartcr	BOOL	Part メッセージに色を付ける
$quitcr	COLORREF	Quit メッセージの表示色
$usequitcr	BOOL	Quit メッセージに色を付ける
$nickcr	COLORREF	Nick メッセージの表示色
$usenickcr	BOOL	Nick メッセージに色を付ける
$topiccr	COLORREF	Topic メッセージの表示色
$usetopiccr	BOOL	Topic メッセージに色を付ける
$namescr	COLORREF	Names メッセージの表示色
$usenamescr	BOOL	Names メッセージに色を付ける
$modecr	COLORREF	Mode メッセージの表示色
$usemodecr	BOOL	Mode メッセージに色を付ける
$kickcr	COLORREF	Kick メッセージの表示色
$usekickcr	BOOL	Kick メッセージに色を付ける
$invitecr	COLORREF	Invite メッセージの表示色
$useinvitecr	BOOL	Invite メッセージに色を付ける
$notifycr	COLORREF	Notify メッセージの表示色
$usenotifycr	BOOL	Notify メッセージに色を付ける

© 株式会社富士通研究所., 1998,2006

設定 → 画面スタイル → スタイル

$notdisplay	BOOL	Join,part,quit,nick メッセージを表示しない
$autofreeze	BOOL	画面停止を自動にする
$autousersort	BOOL	ユーザリストの自動ソート
$inputwindow	BOOL	入力ウインドウを下にする
$hidechanname	BOOL	現在のバッファにチャンネル名を表示しない
$hideinitmsg	BOOL	入力ウインドウに「ここに・・・」を表示しない
$memberlistmode	INT	ユーザリストを詳細表示に
$channellistmode	INT	チャンネルリストを詳細表示に
$usemembericon	BOOL	ユーザリストでアイコン表示を有効に
$usechannelicon	BOOL	チャンネルリストでアイコン表示を有効に
$topmost	BOOL	最前面表示
$usesystemmenu	BOOL	メニューをシステムメニューに収容

設定 → イベントスイッチ

$joinshowuserhost	BOOL	Join してきたユーザ名とホスト名を表示する
$privdonotautoactive	BOOL	バッファを自動的にアクティブにしない
$privbeep	BOOL	個人宛のメッセージが届いたら音を鳴らす
$privbeepusewave	BOOL	個人宛のメッセージが届いたときに WAVE ファイルを使用する
$privbeepwavefile	STRING	WAVE ファイルのパス
$inviteautojoin	BOOL	Invite されたチャンネルに自動的に入る
$modehidekeyword	BOOL	チャンネルキーをタイトルバーに表示しない
$autoreconnect	BOOL	自動再接続をする
$floodprotection	BOOL	フラッドプロテクションを利用する
$dontremovebufonclose	BOOL	接続が切れたときにバッファを消さない
$useautonickselect	BOOL	ニックネームの自動変更を行う。
$nickselectschema	BOOL	変更方法の指定（0:末尾に数字、1:末尾に"_"を追加）
$hideconfirmexit	BOOL	終了確認ダイアログの非表示
$hidekeyconfirm	BOOL	キーワード確認ダイアログの非表示
$usenotify	BOOL	Notify 機能を使用する
$notifyfreq	INT	Notify のニックネームをチェックする間隔
$notifynick	LIST	Notify のニックネームリスト

設定 → チャンネルエイリアス設定

$channelaliasdef	STRING	デフォルトマスク
$aliaslist（＊6）	STRING	チャンネルエイリアスのリスト
$channelnumarray（＊7）	STRING	チャンネル番号予約

Ignore 関係

$ignorelist	LIST	無視するニックネームのリスト

© 株式会社富士通研究所., 1998,2006

SJIS 関連

$usesjis	BOOL	SJIS の使用

Socks 関係

$usesocks	BOOL	Socks の使用
$socksversion	INT	Socks のバージョン
$socksserver	STRING	Socks のサーバ名
$socksport	INT	Socks のポート番号

NAT 関連

$usenat	BOOL	NAT 機能を使用する
$natip	INT	NAT の IP 番号
$natport	INT	NAT のポート番号

型 S：string 型
N：unsigned int 型
B：BOOL 型（TRUE＝0 以外、 FALSE＝0）
C：COLORREF 型の値として使われる Long 値
L：リスト形式（＜データ＞ CRLF ＜データ＞ CRLF ・・・・・）
＊１：このタイプは、データ設定時には使用できません
＊２：このタイプは、データ取得時には使用できません
＊３：チャンネルリストは、以下の形式に なります。
チャンネル１,チャンネル２,チャンネル３・・・
各チャンネル間は“ ”（ブランク）で区切られます。（登録時は、各チャンネル間は改行で区切って下さい）
＊４：この値はブラウザの種類を表しています。（意味は以下）
“0”：ブラウザの自動選択
“1”：Netscape
“2”：Internet explorer
＊５：キーワードリストは、以下の形式になります。
キーワード１“CR＋LF”キーワード２“CR+LF”キーワード３…
＊６：チャンネルエイリアスリストは以下の形式になります。
＜チャンネルの本名＞ “,” ＜チャンネルの別名＞ “ＣＲ＋ＬＦ”・・・・
＊７：チャンネル番号予約のデータは設定が無い場合にも“”（空のデータ）を送信します。
例：１→#test1 ２→なし ３→#test3 ４～１８→なし １９→#test19 の場合
#test1”CRLF””CRLF”***#test3***”CRLF””CRLF””CRLF””CRLF””CRLF””CRLF””CRLF””CRLF””CRLF””CRLF””CRLF””CRLF””CRLF””CRLF””CRLF””CRLF”***#test19***
＊８：サーバリスト登録時には、１番目のサーバがデフォルトサーバとして登録されます。

© 株式会社富士通研究所., 1998,2006

名 称：

ShowWndEx

書 式：

void ShowWndEx (long *SettingNo*)

機 能：

指定したサーバと接続しているウインドウを表示します。

パラメータ：

long	*SettingNo*	IN	サーバ識別子

© 株式会社富士通研究所., 1998,2006

名 称：

HideWndEx

書 式：

void HideWndEx (long *SettingNo*)

機 能：

指定したサーバと接続しているウインドウを非表示します。

パラメータ：

long　*SettingNo*　IN　サーバ識別子

© 株式会社富士通研究所., 1998,2006

名 称：

GetCStatus

書 式：

void GetCStatus (long *SettingNo*, BSTR* *tp*, long* *num*, VARIANT* *vt*)

機 能：

CHOCOA の状態を取得します。

パラメータ：

long	SettingNo	IN	サーバ識別子
BSTR *	tp	IN	データタイプ（備考参照）
long *	num	IN/OUT	インデックス ／ データ数
VARIANT *	vt	OUT	取得データ（備考参照）

戻り値：

正常：TRUE

異常：FALSE

備 考：

・データ内容

データタイプ	情報内容	通知データ		
		long num	VARIANT vt	型
WINDOW	CHOCOAウィンドウの現在位置	データ個数：OUT	ウィンドウの状態（*1） ウィンドウの左上X座標 ウィンドウの左上Y座標 ウィンドウの右下X座標 ウィンドウの右下Y座標	Long
KEYWORDCOUNT	キーワードが発言された個数 キーワードを含むメッセージリスト	データ個数：OUT	キーワード文字列(*2)	文字列
KEYWORD	キーワードを含むメッセージ	インデックス：IN(*3) データ個数：OUT	チャンネル ニックネーム メッセージ	文字列
URL	URL個数とURLリスト	データ個数：OUT	取得URL	文字列

© 株式会社富士通研究所., 1998,2006

＊１：ウインドウ状態の内容

0	CHOCOAウインドウが非表示の場合
1	CHOCOAウインドウが通常サイズで表示された場合
2	CHOCOAウインドウが最小化された場合
3	CHOCOAウインドウが最大化された場合
4	CHOCOAウインドウが最大化（または最小化）表示される直前のサイズと位置で表示された場合
5	CHOCOAウインドウが表示された場合
6	CHOCOAウインドウが最小化された場合（システムリストのトップレベルウインドウがアクティブになります）
7	CHOCOAウインドウが最小化された場合（CHOCOAウィンドウがアクティブ場合は、アクティブなままです）
8	CHOCOAウインドウが現在の状態で表示された場合（現在アクティブなウインドウはアクティブなままです）
9	CHOCOAウインドウがアクティブ表示された場合（最小化または最大化表示から復元します）

＊２ 取得できるキーワード文字列は、文字列型の配列になっています。
＊３ ０以上、KEYWORDCOUNT で取得したデータ個数以下の値

例：

```
Public obj As Object
Private Sub Button1_Click()
    Dim tp As String
    Dim SettingNo As Long
    Dim num As Long
    Dim ret As Long
    Dim ldata(5) As Long
    Dim vaData As Variant

    ‘ CHOCOA オブジェクトの作成
    Set obj = CreateObject(“CHOCOA.Application”)

    SettingNo = 1
    tp = “WINDOW”
    vaData = ldata
    If tp = "WINDOW" Then
        ‘ CHOCOA 状態の取得
        ret =obj.GetCStatus(SettingNo, tp, num, vaData)
        ‘          状態          左上 X          左上 Y          右下 X          右下 Y
        viewdata = Str(vaData (0)) + Str(vaData (1)) + Str(vaData (2)) + Str(vaData (3)) + Str(vaData (4))
    End if

End Sub
```


© 株式会社富士通研究所., 1998,2006

＜IRC コマンド機能＞

IRC コマンド機能は、CHOCOA が提供しているコマンドの機能を使用することができます。使用時には IRC サーバと接続されていることが必要です。接続されていない状態で使用した場合にはエラーとなります。

名 称：

Join

書 式：

BOOL Join (long *SettingNo*, BSTR* *Channel*)

機 能：

指定サーバに接続している CHOCOA 上で、新しいチャンネル（Channel）に入ります。既に指定したチャンネルに入っている場合は、指定チャンネルをアクティブチャンネルにします。

パラメータ：

long	*SettingNo*	IN	サーバ識別子
BSTR*	*Channel*	IN	新規チャンネル

戻り値：

正常：TRUE

異常：FALSE（サーバ識別子が無効、or 接続できない場合）

例：

```
Dim obj As Object

Private Sub Form_Load()
    'COM サーバの起動
    Set obj = CreateObject("CHOCOA.Application")
    obj.ShowWnd

End Sub

Private Sub JoinCmd_Click()
    Dim Chan As String
    Chan = “#test”
    If obj.Join(1, Chan) Then                    ‘ チャンネルへ入る
        MsgBox ("#test チャンネルに入りました")
    End If
End Sub

Private Sub PartCmd_Click()
    Dim Chan As String
    Chan = “#test”
    If obj.Part(1, Chan) Then                    ‘ チャンネルから抜ける
        MsgBox ("#test チャンネルから抜けました")
    End If
End Sub
```


© 株式会社富士通研究所., 1998,2006

名 称：

Part

書 式：

BOOL Part (long *SettingNo*, BSTR* *Channel*)

機 能：

指定サーバに接続している CHOCOA 上で入っているチャンネルから抜けます。

パラメータ：

long	*SettingNo*	IN	サーバ識別子
BSTR*	*Channel*	IN	新規チャンネル

戻り値：

正常：TRUE

異常：FALSE（サーバ識別子が無効、or 抜けられない場合）

© 株式会社富士通研究所., 1998,2006

名 称：

PrivMessage

書 式：

BOOL PrivMessage (long *SettingNo*, BSTR* *Channel*, BSTR* *Message*)

機 能：

指定サーバに接続している CHOCOA 上で、指定チャンネルに対しプライベートメッセージを送信します。CHOCOA 入力ウインドウからメッセージを送信する場合と同じです。

パラメータ：

long	*SettingNo*	IN	サーバ識別子
BSTR*	*Channel*	IN	チャンネル
BSTR*	*Message*	IN	送信メッセージ

戻り値：

正常：TRUE
異常：FALSE

例：

```
Private Sub PrivCmd_Click()
    Dim Chan As String
    Dim Mess As String
    Dim obj As Object

    'COM サーバの起動
    Set obj = CreateObject("CHOCOA.Application")

    Chan = “#test”
    Mess = “Private メッセージのテスト”
    If obj.PrivMessage(1, Chan, Mess) Then
        MsgBox ("Private メッセージを送信しました")
    End If

End Sub
```


© 株式会社富士通研究所., 1998,2006

名 称：

NoticeMessage

書 式：

BOOL NoticeMessage (long *SettingNo*, BSTR* *Channel*, BSTR* *Message*)

機 能：

指定サーバに接続している CHOCOA 上で、指定チャンネルに対し Notice メッセージを送信します。

パラメータ：

long	*SettingNo*	IN	サーバ識別子
BSTR*	*Channel*	IN	チャンネル
BSTR*	*Message*	IN	送信メッセージ

戻り値：

正常：TRUE

異常：FALSE（Notice メッセージ送信に失敗した場合）

例：

```
Private Sub NoticeCmd_Click()
    Dim Chan As String
    Dim Mess As String
    Dim obj As Object

    'COM サーバの起動
    Set obj = CreateObject("CHOCOA.Application")

    Chan = “#test”
    Mess = “Notice メッセージのテスト”
    If obj. NoticeMessage (1, Chan, Mess) Then
        MsgBox ("Notice メッセージを送信しました")
    End If

End Sub
```


© 株式会社富士通研究所., 1998,2006

名 称：

Nick

書 式：

BOOL Nick (long *SettingNo*, BSTR* *NewNick*)

機 能：

現在使用しているニックネームを変更します。サーバ設定情報のニックネームを変更することはできません。

パラメータ：

long	*SettingNo*	IN	サーバ識別子
BSTR*	*NewNick*	IN	新しいニックネーム

戻り値：

正常：TRUE

異常：FALSE（ニックネームが変更できない場合）

例：

```
Private Sub Nick_Click()
    Dim Nick As String
    Dim obj As Object

    'COM サーバの起動
    Set obj = CreateObject("CHOCOA.Application")

    Nick = “NewNick”
    If obj.Nick(1, Nick) Then
        MsgBox ("NewNick に変更しました")
    End If

End Sub
```


© 株式会社富士通研究所., 1998,2006

名 称：

Ignore

書 式：

BOOL Ignore (long *SettingNo*, BSTR* *IrcUser*, BOOL *OnOff*)

機 能：

無視するユーザを設定することができます。

パラメータ：

long	*SettingNo*	IN	サーバ識別子
BSTR*	*IrcUser*	IN	IRC ユーザ
BOOL	*OnOff*	IN	状態（TRUE：無視する、FALSE：無視を解除）

戻り値：

正常：TRUE

異常：FALSE

備 考：

IrcUser の設定は以下の形式で設定してください。

“ニックネーム”!”マシンアドレス”

例：hoge!hogehoge.fujitsu.co.jp

設定にはワイルドカードを使用することもできます。

例：nickname!*@* 、a*!*@*、*!*@*.jp

例：

```
Private Sub IgnoreCmd_Click()
    Dim Mess As String
    Dim flg As Integer
    Dim obj As Object

    'COM サーバの起動
    Set obj = CreateObject("CHOCOA.Application")

    ‘ 設定／解除の選択
    If OnOption.Value = True Then
        flg = True                          ‘ 設定
    ElseIf OffOption.Value = True Then
        flg = False                         ‘ 解除
    End If

    Mess = “test!*@*”                       ‘ 全てのマシン上の”test”ユーザ
    If obj.Ignore(1, Mess, flg) Then
        MsgBox ("設定／解除しました")
    End If

End Sub
```


© 株式会社富士通研究所., 1998,2006

名 称：

SetAway

書 式：

BOOL SetAway (long *SettingNo*, BSTR* *AwayMessage*)

機 能：

サーバ識別子で示されたサーバに Away メッセージを登録します。

パラメータ：

long	*SettingNo*	IN	サーバ識別子
BSTR*	*AwayMessage*	IN	Away メッセージ

戻り値：

正常：TRUE

異常：FALSE（登録できなかった場合）

例：

```
Private Sub SetAwayCmd_Click()
    Dim away As String
    Dim obj As Object

    'COM サーバの起動
    Set obj = CreateObject("CHOCOA.Application")

    away = “ちょっと離れます”
    If obj.SetAway(1, away) Then
        MsgBox ("設定ました")
    End If

End Sub
```


© 株式会社富士通研究所., 1998,2006

名 称：

DelAway

書 式：

BOOL DelAway (long *SettingNo*)

機 能：

サーバ識別子で示されたサーバ上の全ての Away メッセージを解除します。

パラメータ：

long　*SettingNo*　IN　サーバ識別子

戻り値：

正常：TRUE

異常：FALSE（解除できなかった場合）

例：

```
Private Sub DelAwayCmd_Click()
    Dim away As String
    Dim obj As Object

    'COM サーバの起動
    Set obj = CreateObject("CHOCOA.Application")

    away = “ちょっと離れます”
    If obj.DelAway(1) Then
        MsgBox ("解除しました")
    End If

End Sub
```


© 株式会社富士通研究所., 1998,2006

名 称：

Invite

書 式：

BOOL Invite (long ***SettingNo,*** **BSTR*** ***Nick,*** **BSTR*** ***Channel*****)**

機 能：

サーバ識別子で指定されたサーバ上 **Invite** コマンドメッセージを発行する。

パラメータ：

long	*SettingNo*	IN	サーバ識別子
BSTR*	Nick	IN	ニックネーム
BSTR*	Channel	IN	チャンネル名

戻り値：

正常：TRUE

異常：FALSE

例：

```
Private Sub InviteMessageCmd()
        Dim SettingNo As Long
        Dim strNick As String
        Dim strChannel As String
        Dim ret As Boolean
        Dim obj As Object

        SettingNo = 1
        strNick = “naka”
        strChannel = “#test”

       'COM サーバの起動
       Set obj = CreateObject("CHOCOA.Application")

        ret = obj.Invite(SettingNo, strNick, strChannel)
        If ret Then
            MsgBox ("Invite コマンドは正常終了しました")
        Else
            MsgBox ("Invite コマンドはエラーをかえしました")
        End If

        Exit Function

ErrExecScript:
        MsgBox (CmdType + "のコマンドは異常終了しました")

End Function
```


© 株式会社富士通研究所., 1998,2006

＜IRC 情報の取得＞

CHOCOA が持っている IRC の情報を取得することができます。使用時には IRC サーバと接続されていることが必要です。接続されていない状態で使用した場合にはエラーとなります。

名 称：

GetChanUserList

書 式：

BOOL GetChanUserList (long *SettingNo*, BSTR* *Channel*, BSTR* *UserList*, long* *count*)

機 能：

サーバ識別子で示されたサーバ上で、指定チャンネルに入っている全てのユーザのニックネームリストと、ニックネームの数を返します。

パラメータ：

long	*SettingNo*	IN	サーバ識別子
BSTR*	*Channel*	IN	チャンネル
BSTR*	*UserList*	OUT	ニックネームリスト
Long*	*count*	OUT	ニックネームの数

戻り値：

正常：TRUE

異常：FALSE（取得できない場合 or 指定サーバが接続されていない場合）

備 考：

UserList はデリミタとして“ ”（ブランク）が入っています。

例：ニックネーム１“ ” ニックネーム２“ ” ニックネーム３

例：

```
End FunctionPrivate Sub GetChanUserCmd_Click()
    Dim UserList As String
    Dim Chan As String
    Dim count As Long
    Dim obj As Object

    'COM サーバの起動
    Set obj = CreateObject("CHOCOA.Application")

    Chan = “#test”
    ‘ “#test”チャンネルに Join しているユーザ一覧取得
    If obj.GetChanUserList(1, Chan, UserList, count) Then
        MsgBox(UserList)
    End If

End Sub
```


© 株式会社富士通研究所., 1998,2006

名 称：

GetChanList

書 式：

BOOL GetChanList (long *SettingNo*, BSTR* *ChanList*, long* *count*)

機 能：

サーバ識別子で示されたサーバ上で、自分が入っているチャンネルのリストとチャンネル数を返します。

パラメータ：

long	*SettingNo*	IN	サーバ識別子
BSTR*	*ChanList*	OUT	チャンネルリスト
Long*	*count*	OUT	チャンネル数

戻り値：

正常：TRUE

異常：FALSE（取得できない場合 or 指定サーバが接続されていない場合）

備 考：

ChanList はデリミタとして“ ”（ブランク）が入っています。

例：チャンネル１“ ”チャンネル２“ ”チャンネル３

例：

```
Private Sub GetChanCmd_Click()
    Dim ChanList As String
    Dim count As Long
    Dim obj As Object

    'COM サーバの起動
    Set obj = CreateObject("CHOCOA.Application")

    ‘ 指定サーバで Join しているチャンネルのリスト取得
    If obj.GetChanList(1, ChanList, count) Then
    If ret = True Then
        MsgBox(ChanList)
    End If

End Sub
```


© 株式会社富士通研究所., 1998,2006

名 称：

GetServList

書 式：

BOOL GetServList (BSTR* *ServList*, long* *count*)

機 能：

現在接続されているサーバの設定名リストと設定名の個数を返します。

パラメータ：

BSTR*	*ServList*	OUT	サーバ設定名リスト
Long*	*count*	OUT	サーバ設定名数

戻り値：

正常：TRUE

異常：FALSE（取得できない場合 or 指定サーバが接続されていない場合）

備 考：

・ ServList は区切り文字として“ ”（ブランク）が入っています。

例：サーバ設定名１“ ”サーバ設定名２“ ” サーバ設定名３

・ サーバ設定名に“ ”（ブランク）が使用されている場合には、ブランクは“%20”で表されます。“%”が使用されている場合には、“%%”で表されます。

例： 変換前 変換後

TEST %1 → TEST%20%%1

例：

```
Private Sub GetServListCmd_Click()
    Dim ServList As String
    Dim count As Long
    Dim obj As Object

    'COM サーバの起動
    Set obj = CreateObject("CHOCOA.Application")

    ‘ 接続しているサーバのリスト取得
    If obj.GetServList(ServList, count) Then
        MsgBox(ServList)
    End If

End Sub
```


© 株式会社富士通研究所., 1998,2006

＜CHOCOA スクリプト関連＞
　CHOCOA スクリプトの作成（登録）、実行、結果取得ができます。使用時には IRC サーバと接続されていることが必要です。接続されていない状態で使用した場合にはエラーとなります。

名 称：
　　ChkServiceName
書 式：
　　BOOL ChkServiceName (long *SettingNo*, BSTR* *ServiceNM*)
機 能：
　　サーバ識別子で示されたサーバと接続している CHOCOA 上で、現在そのサービス名が使用されているかどうかチェックします。
パラメータ：

long	*SettingNo*	IN	サーバ識別子
BSTR*	*ServiceNM*	IN	サービス名

戻り値：
　　TRUE　：使用済みの場合
　　FALSE：未使用の場合
例：

```
Private Sub ChkServiceNameCmd_Click()
    Dim ServiceNM As String
    Dim ret As Integer
    Dim obj As Object

    'COM サーバの起動
    Set obj = CreateObject("CHOCOA.Application")

    ServiceNM = “TestServiceName”
    ‘ サービス名のチェック
    If obj.ChkServiceName(1, ServiceNM) Then
        MsgBox(“使用済みです")
    Else
        MsgBox(“使用されていません")
    End If

End Sub
```


© 株式会社富士通研究所., 1998,2006

名 称：

GetScriptFile

書 式：

BOOL GetScriptFile (long *SettingNo*, BSTR* *ServiceNM*, BSTR* *Path*)

機 能：

サーバ識別子で示されたサーバと接続している CHOCOA 上で、現在使用されているサービスが設定されているスクリプトファイル（フルパス）を取得します。

パラメータ：

long	*SettingNo*	IN	サーバ識別子
BSTR*	*ServiceNM*	IN	サービス名
BSTR*	*Path*	OUT	ファイルパス

戻り値：

正常：TRUE
異常：FALSE

例：

```
Private Sub GetScriptFileCmd_Click()
    Dim ServiceNM As String
    Dim FilePath As String
    Dim obj As Object

    'COM サーバの起動
    Set obj = CreateObject("CHOCOA.Application")

    ServiceNM = “TestServiceName”
    ‘ スクリプトファイルパスの取得
    If obj.GetScriptFile(1, ServiceNM, FilePath) Then
        MsgBox(FilePath)
    End If

End Sub
```


© 株式会社富士通研究所., 1998,2006

名 称：

GetServiceName

書 式：

BOOL GetServiceName (long *SettingNo*, BSTR* *Path*, BSTR* *ServiceNM*)

機 能：

サーバ識別子で示されたサーバと接続している CHOCOA 上で、現在使用されているサービスが設定されているスクリプトファイルからサービス名を取得します。

パラメータ：

long	*SettingNo*	IN	サーバ識別子
BSTR*	*Path*	IN	ファイルパス。
BSTR*	*ServiceNM*	OUT	サービス名

戻り値：

正常：TRUE

異常：FALSE

例：

```
Private Sub GetServiceNameCmd_Click()
    Dim ServiceNM As String
    Dim FilePath As String
    Dim obj As Object

    'COM サーバの起動
    Set obj = CreateObject("CHOCOA.Application")

    FilePath = “c:¥temp¥cscript.csf”
    If obj.GetServiceName(1, FilePath, ServiceNM) Then
        MsgBox(ServiceNM)
    End If

End Sub
```


© 株式会社富士通研究所., 1998,2006

名 称：

RegScriptFile

書 式：

BOOL RegScriptFile (long *SettingNo*, BSTR* *ScriptFile*, BSTR* *ServiceNM*, BOOL *Saveflg*)

機 能：

サーバ識別子で示されたサーバと接続している CHOCOA に、指定されたスクリプトファイル(~.csf)を登録します。

パラメータ：

long	*SettingNo*	IN	サーバ識別子
BSTR*	*ScriptFile*	IN	スクリプトファイル（フルパス指定）
BSTR*	*ServiceNM*	OUT	サービス名
BOOL	*Saveflg*	IN	保存指定（TRUE:保存、FALSE:未保存）

戻り値：

正常：TRUE

異常：FALSE

備 考：

・CHOCOA からスクリプトファイルをオープンできる必要が有ります。

・未保存を選択した場合、一時的なスクリプトファイルが Windows テンポラリディレクトリ上に作成されます。

・指定したサービス名が既に使用済みの場合、サービス名は以下の形式で自動的に決定されサービス名（ServiceNM）に設定されます。スクリプトを実行するときは、そのサービス名を指定してください。

例： “指定サービス名_管理番号”

例：

```
Private Sub RegScriptFile_Click()
    Dim File As String
    Dim service As String
    Dim obj As Object

    'COM サーバの起動
    Set obj = CreateObject("CHOCOA.Application")

    File = “c:¥temp¥cscript.csf”
    '保存 or 未保存のチェック
    If Saveflg = "未保存" Then
        ‘ チョコアスクリプトファイルの登録（未保存）
        obj.RegScriptFile(GetServID, File, service, False)
    Else
        ‘ チョコアスクリプトファイルの登録（保存）
        obj.RegScriptFile(GetServID, File, service, True)
    End If
End Sub
```


© 株式会社富士通研究所., 1998,2006

名 称：

RegScript

書 式：

BOOL RegScript (long *SettingNo*, BSTR* *Script*, BSTR* *ServiceNM*, BOOL *Saveflg*)

機 能：

サーバ識別子で示されたサーバと接続している CHOCOA に、指定されたスクリプトソースを登録します。

パラメータ：

long	*SettingNo*	IN	サーバ識別子
BSTR*	*Script*	IN	スクリプトソース
BSTR*	*ServiceNM*	OUT	サービス名
BOOL	*Saveflg*	IN	保存指定（TRUE:保存、FALSE:未保存）

戻り値：

正常：TRUE

異常：FALSE　　設定できない場合

備　考：

・保存を選択した場合、スクリプトファイルがテンポラリディレクトリ(“CHOCOA のディレクトリ”¥ScriptTmp)に作成されます（作成されるファイル名は、“_CCAPI_SCRIPT_FILE_[管理番号].csf” になります)。次回 CHOCOA 起動時にも設定したスクリプトを使用することができます。

・指定したサービス名が既に使用済みの場合、サービス名は以下の形式で自動的に決定されます。スクリプトを実行するときは、そのサービス名を指定してください。

例：　“指定サービス名_管理番号”

・スクリプトソースは、下記フォーマットを使用してください。

例：スクリプトソース作成時のフォーマット

```
/Service [サービス名]
{
 ;Alias Script 設定
 /Function
 {
 }
 ;Remote Script 設定
 /Event
 {
 }
 :アクセスコントロール設定
 /Access
 {
 }
 :右クリックメニュー設定（メインウィンドウ用）
 /PopupMain
 {
 }
 :右クリックメニュー設定（ユーザーウィンドウ用）
 /PopupUser
 {
```


© 株式会社富士通研究所., 1998,2006

```
    }
    }
```

例：

```
Private Sub RegSourceCmd_Click()
    Dim Script As String
    Dim service As String
    Dim flg As Integer
    Dim obj As Object

    'COM サーバの起動
    Set obj = CreateObject("CHOCOA.Application")

    Script = ScriptSrc

    '保存 or 未保存のチェック
    If Saveflg = "未保存" Then
        flg = False
    Else
        flg = True
    End If

    If obj.RegScript(1, Script, service, flg) Then
        MsgBox(service + “名で登録しました”
    End If

End Sub
```

※ ScriptSrc ＝

```
    /Service テストスクリプト 1
    {
     ;Alias Script 設定
     /Function
     {
        /test1
        {
            /comresult $me
        }
      }
        .
        .
        .
      }
```


© 株式会社富士通研究所., 1998,2006

名 称：

RegScriptEx

書 式：

BOOL RegScriptEx (long *SettingNo*, **long** *Count*, **long** *Total*, **BSTR*** *Script*, **BSTR*** *ServiceNM*, **BOOL** *Saveflg*)

機 能：

サーバ識別子で示されたサーバと接続している CHOCOA に、指定されたスクリプトを登録します。スクリプトソースが膨大な場合は、分割して送信することが可能です。登録途中で登録操作をキャンセルしたい場合には、“Count＝0”を指定することで行えます。登録途中に違うスクリプトを登録した場合には、前に登録を開始していたスクリプトソースは無効となります。

パラメータ：

long	*SettingNo*	IN	サーバ識別子
long	*Count*	IN	登録個数（1 以上を指定）
long	*Total*	IN	分割総数
BSTR*	*Script*	IN	スクリプトソース
BSTR*	*ServiceNM*	OUT	サービス名
BOOL	*Saveflg*	IN	保存指定（TRUE:保存、FALSE:未保存）

戻り値：

正常：TRUE
異常：FALSE

備 考：

・RegScript の備考参照

例：

スクリプトソースを 3 分割で送信する場合の例

```
Private Sub Sample
    ・・・・・
    Script2 = ScriptSource              ‘スクリプトソース（ScriptSource）

    total = 3                           ‘3 分割で送信する
    length = Len(Script2) / total
    For count = 1 To total
        If count = total Then
            ret = Object.RegScriptEx(ServName, count, total, Script2, Service, 0)
        Else
            Script = Left(Script2, length)
            ret = Object.RegScriptEx(ServName, count, total, Script, Service, 0)
        End If

        Script2 = Mid(Script2, length + 1)
    Next
    ・・・・・
End Sub
```


© 株式会社富士通研究所., 1998,2006

名 称：

DelScript

書 式：

BOOL DelScript (long *SettingNo*, BSTR* *ServiceNM*)

機 能：

サーバ識別子で示されたサーバと接続している CHOCOA 上で、指定されたサービス名を持つスクリプトを削除します。

パラメータ：

long	*SettingNo*	IN	サーバ識別子
BSTR*	*ServiceNM*	IN	サービス名

戻り値：

正常：TRUE

異常：FALSE

例：

```
Private Sub DelScriptCmd_Click()
    Dim service As String
    Dim obj As Object

    'COM サーバの起動
    Set obj = CreateObject("CHOCOA.Application")

    service = “TestServiceName”
    ‘ スクリプトの削除
    If obj.DelScript(1, service) Then
        MsgBox ("削除しました")
    End If
End Sub
```


© 株式会社富士通研究所., 1998,2006

名 称：

RegScriptFunc

書 式：

BOOL RegScriptFunc (long *SettingNo*, BSTR* *Script*, BSTR* *ServiceNM*)

機 能：

サーバ識別子で示されたサーバと接続している CHOCOA に、スクリプトコマンドを登録します。登録したスクリプトコマンドは一時的な使用のみ可能です。（CHOCOA を終了した時点で、登録は削除されます。）

パラメータ：

long	*SettingNo*	IN	サーバ識別子
BSTR*	*Script*	IN	スクリプトソース
BSTR*	*ServiceNM*	OUT	サービス名

戻り値：

正常：TRUE

異常：FALSE

備 考：

・サービス名は自動的に設定されます。設定されるサービス名は以下の形式です。

例： “_CCAPI_TMP_SERVICE_[管理番号]”

・このサービス名は、スクリプト実行時に使用します。

例：

ExecScript の例参照

© 株式会社富士通研究所., 1998,2006

名 称：

ExecScript

書 式：

BOOL ExecScript (long *SettingNo*, BSTR* *Script*, BSTR* *Result*)

機 能：

登録した（又は既に登録済み）のスクリプトコマンドを実行します。以下の形式でスクリプトコマンドを作成すると、実行結果が Result に設定されます。

例：現在のニックネームを取得する

```
/Sample_Script
{
    /comresult $me
}
```

パラメータ：

long	*SettingNo*	IN	サーバ識別子
BSTR*	*Script*	IN	スクリプトコマンド
BSTR*	*Result*	OUT	実行結果

戻り値：

TRUE	登録できた場合
FALSE	登録できない場合

例：

```
Dim service As String
Dim obj As Object

Private Sub RegFuncCmd_Click() ()
    Dim Script As String
    'COM サーバの起動
    Set obj = CreateObject("CHOCOA.Application")
    Script = FuncSrc                              ‘ 機能参照
    ‘ スクリプトコマンドの登録
    If obj.RegScriptFunc(1, Script, service) Then
        MsgBox ("登録しました（サービス名 = " + service + “ ）”)
    End If
End Sub

Private Sub ExecScriptCmd_Click()
    Dim Result As String
    Dim command As String
    ‘ スクリプトコマンドの編集(“/サービス名.関数名”)
    command = “/” + service + “.” + “Sample_Script”
    ‘ スクリプトコマンドの実行
    If obj.ExecScript(1, command, Result) Then
        MsgBox(Result)
    End If
End Sub
```


© 株式会社富士通研究所., 1998,2006

名 称：

ExecScriptEx

書 式：

BSTR ExecScriptEx (long *SettingNo*, BSTR *Script*)

機 能：

登録した（又は既に登録済み）のスクリプトコマンドを実行します。

“/comresult”コマンドを実行すると、実行結果をリターン値で返します。（実行されない場合には、リターン値は”（NULL）”を返します。

例：現在のニックネームを取得する

```
/Sample_Script
{
    /comresult $me
}
```

パラメータ：

long	*SettingNo*	IN	サーバ識別子
BSTR	*Script*	IN	スクリプトコマンド

戻り値：

正常： 実行結果　又は　“”（/comresult でリターン値が設定され無かった場合）

異常： “”

© 株式会社富士通研究所., 1998,2006

＜CHOCOA イベントの取得＞

COM クライアントは、下記シンクインタフェースを実装し、CHOCOA（COM サーバ）とコネクションを確立することで、CHOCOA 側で発生した各種イベントを取得することができます。（取得するためには、欲しいイベント情報のプロパティを設定する必要があります。）

MessageEvent：	メッセージイベントを取得
CTCPMessageEvent：	CTCP メッセージイベントを取得
IRCEvent：	IRC イベントの取得
ScriptEvent：	スクリプトイベントの取得
ChocoaEvent	Chocoa の状態取得

名 称：

MessageEvent

書 式：

BOOL MessageEvent (BSTR ***SettingNM*****, BSTR** ***Channel*****, BSTR** ***Nick*****, BSTR** ***Address*****, Short** ***MessageType*****, BSTR** ***Message*****)**

機 能：

全ての接続されているサーバ内の Join しているチャンネルで交わされる会話内容を通知します。

パラメータ：

BSTR	*SettingNM*	OUT	サーバ設定名
BSTR	*Channel*	OUT	チャンネル
BSTR	*Nick*	OUT	ニックネーム
BSTR	*Address*	OUT	マシンアドレス
Short	*MessageType*	OUT	メッセージタイプ （0:Priv Message、1:NoticeMessage）
BSTR	*Message*	OUT	メッセージ内容

戻り値：

正常：TRUE

異常：FALSE

備考：

・自 CHOCOA ウインドウからメッセージを送信した場合にはニックネームは空になります。

© 株式会社富士通研究所., 1998,2006

名 称：

CTCPMessageEvent

書 式：

BOOL CTCPMessageEvent (BSTR *SettingNM*, **BSTR** *Channel*, **BSTR** *Nick*, **BSTR** *Address*, **Short** *MessageType*, **BSTR** *Message*)

機 能：

全ての接続されているサーバ内の Join しているチャンネルで交わされる CTCP メッセージを通知します。

パラメータ：

BSTR	*SettingNM*	OUT	サーバ設定名
BSTR	*Channel*	OUT	チャンネル
BSTR	*Nick*	OUT	ニックネーム
BSTR	*Address*	OUT	マシンアドレス
Short	*MessageType*	OUT	メッセージタイプ （0:CTCP Message、1:CTCP Reply Message）
BSTR	*Message*	OUT	メッセージ内容

戻り値：

正常：TRUE

異常：FALSE

© 株式会社富士通研究所., 1998,2006

名 称：

ScriptEvent

書 式：

BOOL ScriptEvent (BSTR *SettingNM*, **Short** *EventType*, **BSTR** *ServiceNM*, **BSTR** *FuncNM*, **BSTR** *ReValue*)

機 能：

スクリプトソース内で“/comresult”コマンドが指定されていると、結果を接続中の COM クライアントにイベントとして通知します。

パラメータ：

BSTR	*SettingNM*	OUT	サーバ設定名
Short	*EventType*	OUT	通知イベント
BSTR	*ServiceNM*	OUT	サービス名
BSTR	*FuncNM*	OUT	コマンド名
BSTR	*ReValue*	OUT	実行結果

戻り値：

正常：TRUE

異常：FALSE

© 株式会社富士通研究所., 1998,2006

名 称：

IRCEvent

書 式：

BOOL IRCEvent (BSTR *SettingNM*, long *EventType*, BSTR *ReValue*)

機 能：

全ての接続されているサーバ内の Join しているチャンネルで交わされる IRC メッセージを通知します。

パラメータ：

BSTR	*SettingNM*	OUT	サーバ設定名
long	*EventType*	OUT	通知イベント
BSTR	*ReValue*	OUT	通知内容（各イベント毎に設定される）

戻り値：

正常：TRUE

異常：FALSE

備考：

IRC イベントとして通知するのは以下のイベントです。(通知データは、” ”(ブランク)で区切られます)

イベントタイプ	タイプ値	通知データ
NICKイベント	4	新ニックネーム、旧ニックネーム
JOINイベント	5	ニックネーム、ホスト名、Joinチャンネル
TOPICイベント	8	ニックネーム、ホスト名、チャンネル、トピック
NAMESイベント	9	チャンネル、ユーザリスト(ユーザ間は″ ″で区切る)
PARTイベント	10	ニックネーム、ホスト名、Partチャンネル、コメント
QUITイベント	11	ニックネーム、ホスト名、Quitメッセージ
MODEイベント	12	ニックネーム、ホスト名、チャンネル、モード
KICKイベント	14	ニックネーム、ホスト名、チャンネル、蹴り出される人、蹴り出される理由
INVITEイベント	18	ニックネーム、ホスト名、チャンネル
ACTIONイベント	22	ニックネーム、ホスト名、チャンネル、メッセージ
ISONイベント	24	ニックネーム、ホスト名、ニックネームリスト
WHOISイベント	25	ニックネーム、ホスト名、本名
WHOイベント	27	ニックネーム、ホスト名、本名

© 株式会社富士通研究所., 1998,2006

名 称：

NumericMessageEvent

書 式：

BOOL NumericMessageEvent (BSTR *SettingNM*, long *Numeric*, BSTR *ReValue*)

機 能：

全ての接続されているサーバ内のニュメリックリプライイベントを通知します。

パラメータ：

BSTR	*SettingNM*	OUT	サーバ設定名
long	*Numeric*	OUT	ニュメリックリプライ値
BSTR	*Message*	OUT	通知メッセージ

戻り値：

正常：TRUE

異常：FALSE

備考：

ニュメリックリプライ値については、RFC1459 を参照してください。

© 株式会社富士通研究所., 1998,2006

名 称：

ChocoaEvent

書 式：

BOOL ChocoaEvent (BSTR *SettingNM*, BSTR *tp*, VARIANT *vaData*)

機 能：

CHOCOA ウインドウの状態、キーワード関連（キーワードを含むメッセージを受け取った時、キーワード確認ボタンを押した時）の状態、URL ボタンの状態の変化を通知します。

パラメータ：

BSTR	*SettingNM*	OUT	サーバ設定名
BSTR	*tp*	OUT	イベントタイプ
VARIANT	*vaData*	OUT	データ

戻り値：

正常：TRUE

異常：FALSE

備考：

・データ内容

通知イベント名	イベントタイプ	通知データ	型	備考
CHOCOAウィンドウの移動時	MOVE	ウィンドウの状態 (*1) ウィンドウの左上X座標 ウィンドウの左上Y座標 ウィンドウの右下X座標 ウィンドウの右下Y座標	Long	
CHOCOAウィンドウサイズ変更時	SIZE	ウィンドウの状態 (*1) ウィンドウの左上X座標 ウィンドウの左上Y座標 ウィンドウの右下X座標 ウィンドウの右下Y座標	Long	
CHOCOAウィンドウのアクティブ時	ACTIVEWINDOW	ウィンドウの状態 (*1) ウィンドウの左上X座標 ウィンドウの左上Y座標 ウィンドウの右下X座標 ウィンドウの右下Y座標	Long	現在、このイベントはフォーカスがあたったウィンドウに発生します。
キーワードを含むメッセージが発言された時	KEYWORD	チャンネル名 ニックネーム マシンアドレス キーワードを含む文字列	文字列	
キーワード確認ボタンを押した時	KEYWORDBUTTON	チャンネル名 ニックネーム キーワードを含む文字列	文字列	キーワード確認ダイアログの『昔へ』、『今へ』ボタン押下時も、このイベントは発生する。
URL取得ボタンを押した時	URLBUTTON	取得URL	文字列	
アクティブチャンネルが切り替えた時	ACTIVECHANNEL	OLDアクティブなチャンネル名 NEWアクティブなチャンネル名	文字列	

© 株式会社富士通研究所., 1998,2006

＊１：ウインドウ状態の内容

0	CHOCOAウインドウが非表示の場合
1	CHOCOAウインドウが通常サイズで表示された場合
2	CHOCOAウインドウが最小化された場合
3	CHOCOAウインドウが最大化された場合
4	CHOCOAウインドウが最大化（または最小化）表示される直前のサイズと位置で表示された場合
5	CHOCOAウインドウが表示された場合
6	CHOCOAウインドウが最小化された場合（システムリストのトップレベルウインドウがアクティブになります）
7	CHOCOAウインドウが最小化された場合（CHOCOAウィンドウがアクティブ場合は、アクティブなままです）
8	CHOCOAウインドウが現在の状態で表示された場合（現在アクティブなウインドウはアクティブなままです）
9	CHOCOAウインドウがアクティブ表示された場合（最小化または最大化表示から復元します）

例：

```
Public WithEvents obj As CHOCOA

Sub obj_ChocoaEvent(ByVal SettingNM As String, ByVal tp As String, ByVal Data)
    Dim tmp As String
    Dim result As String

    tmp = tp
    If StrComp(tmp, "MOVE") = 0 Then
      ‘ 座標表示(左上 X，左上 Y，右下 X，右下 Y)
      result = “( “+Str(Data(1)) + “,” + Str(Data(2)) + “,” + Str(Data(3)) + “,” + Str(Data(4)) + “ )”
      MsgBox（result）
    End If

End Sub
```


© 株式会社富士通研究所., 1998,2006

CHOCOA からのイベントを取得するには、取得したいイベントのプロパティを ON（TRUE）に設定する必要があります。プロパティを ON に設定した後は、イベントハンドラにイベントが通知されるようになります。

以下に、VB5 でのイベント取得方法をいかに示します。

```
Option Explicit

Public WithEvents Obj As CHOCOA

Private Sub Form_Load()
    'COM オブジェクトの生成
    Set Obj = CreateObject("CHOCOA.Application")

    'IRC イベント取得するため、IRC イベントプロパティを True に設定する
    Obj.IRCEvent = True

End Sub

' IRC イベント用のイベントハンドラ
Public Sub Obj _IRCEvent(ByVal SettingNM As String, ByVal MessageType As Long, ByVal ReValue As String)
    Dim str As String

    if MessageType =   4 Then
        str = "ニックネームが変更されました[NewNick = " + ReValue + " ]"
        MsgBox(str)
    End if

End Sub
```


© 株式会社富士通研究所., 1998,2006