

Contents

Currency Server API.....	2
Methods.....	3
AddPriceBuffer.....	3
BroadcastPrices.....	4
StartServer.....	5
PauseServer.....	5
StopServer.....	6
DisconnectThisSession.....	6
DisconnectThisUsername.....	7
GetAllSessionsInfo.....	8
Properties.....	9
Username.....	9
Password.....	9
Port.....	10
HighLowType.....	11
EnableMultiSession.....	11
IgnoreBidWhenGreaterThanAsk.....	12
Events.....	13
ServerStatus.....	13
SessionsChanged.....	13
Data Type.....	14
ServerStatus.....	14
HighLowType.....	14

Currency Server API

Currency Server API Introduction

VertexFX Currency Server API is an ActiveX DLL Component Object Model (COM) control that is supported in many development environments. Our ActiveX control and sample applications plug into all popular development environments.

Using the API you can build your own currency server application which takes prices from any feed, whether it is DDE supported or not. Also you can develop your filtering rules before sending prices to your system.

Methods

AddPriceBuffer

Syntax

```
Public Sub AddPriceBuffer(ByRef SymbolID As String, ByRef  
Bid As Double, ByRef Ask As Double , ByRef Optional High As  
Double = 0.0 , ByRefOptional Low As Double= 0.0, ByRef  
Optional Last As Double = 0.0, ByRef Optional Bank As  
String=" ")
```

This function used to add symbol information to buffer before calling [BroadcastPrices](#) function.

Parameters

Key	Description
SymbolID	The Symbol number to be added, value of type String.
Bid	Bid price value for added symbol, value of type is Double.
Ask	Ask price value for added symbol, value of type is Double.
High	High price value. Optional value of type is Double.
Low	Low price value. Optional value of type is Double.
Last	Last price value. Optional value of type is Double.
Bank	Bank name. Optional value of type is String.

Sample

```
Private Sub AddPrice_Click (ByVal sender As System.Object,  
ByVal e As System.EventArgs) Handles AddPrice.Click  
  
    'CurSer is predefined variable as CVertexCurSvrAPI  
    CurSer.AddPriceBuffer ("SymbolID", 1.0, 2.0, 0.0, 0.0,  
0.0, "Bank name")  
  
    CurSer.BroadcastPrices ()  
  
End Sub
```

BroadcastPrices

Syntax

```
Public function BroadcastPrices () As Boolean
```

This function is used to send the buffered symbol to connected sessions.

Return value

Returns true if price is sent otherwise returns false (if you call it after less than 250 ml seconds from the last call)

Sample

```
Private Sub AddPrice_Click (ByVal sender As System.Object,  
ByVal e As System.EventArgs) Handles AddPrice.Click  
    'CurSer is predefined variable as CVertexCurSvrAPI  
    CurSer.AddPriceBuffer ("SymbolID", 1.0, 2.0, 0.0, 0.0,  
0.0, "Bank name", 0.0)  
    CurSer.BroadcastPrices ()  
End Sub
```

StartServer

Syntax

```
Public Function StartServer () As String
```

This function is used to start listening to a specific port which should be set using **Port** property to enable gateway connection to that port.

Parameters

No parameters.

Sample

```
Private Sub start_Click (ByVal sender As System.Object, ByVal  
e As System.EventArgs) Handles start.Click  
  
    'CurSer is predefined variable as CVertexCurSvrAPI  
    CurSer.startserver ()  
  
End Sub
```

PauseServer

Syntax

```
public Function PauseServer () As String
```

This function is used to pause the server to prevent any new gateway connection

Parameters

No Parameters.

Sample

```
Private Sub pause_Click (ByVal sender As System.Object, ByVal  
e As System.EventArgs) Handles pause.Click  
  
    CurSer.pauseServer()  
  
End Sub
```

StopServer

Syntax

```
Public Function StopServer () As String
```

This function is used to stop the server and disconnect all connected gateway sessions.

Parameters

No parameters.

Sample

```
Private Sub stop_Click (ByVal sender As System.Object,  
ByVal e As System.EventArgs) Handles stop.Click  
    'CurSer is predefined variable as CVertexCurSvrAPI  
    CurSer.StopServer()  
  
End Sub
```

DisconnectThisSession

Syntax

```
Public Sub DisconnectThisSession (ByRef SID As String)
```

This method is used to disconnect the given session ID from the gateway.

Parameters

Key	Description
SID	The session ID to be disconnected, value of type is String.

Sample

```
Private Sub DisconnectThisSession_Click  
    (ByVal sender As System.Object, ByVal  
    e As System.EventArgs) Handles DisconnectThisSession.Click  
  
    'CurSer is predefined variable as CVertexCurSvrAPI  
  
    CurSer.DisconnectThisSession ("SessionID")  
  
End Sub
```

DisconnectThisUsername

Syntax

```
Public Sub DisconnectThisUsername (ByRef UN As String)
```

This method is used to disconnect the given username from gateway.

Parameters

Key	Description
UN	The username to be disconnected, value of type String.

Sample

```
Private Sub Disconnect_Click (ByVal sender As System.Object,  
    ByVal e As System.EventArgs) Handles Disconnect.Click  
  
    'CurSer is predefined variable as CVertexCurSvrAPI  
  
    CurSer.DisconnectThisusername ("username")  
  
End Sub
```

GetAllSessionsInfo

Syntax

```
Public function GetAllSessionsInfo(ByRef vUsernames As System.Array,ByRef vLoginTimes As System.Array, ByRef vIPs As System.Array, ByRef vSessionIDs As System.Array) As Boolean
```

This function is used to get the all connected sessions info.

Parameters

Key	Description
vUsernames	Server result buffer that contain session usernames, value of type is Array
vLoginTimes	Server result buffer that contain session login times, value of type is Array
vIPs	Server result buffer that contain session IPs address, value of type is Array
vSessionIDs	Server result buffer that contain session IDs, value of type is Array

Return value

Returns true if get information for connected session was successful, otherwise returns false.

Sample

```
Private Sub SessionsInfo_Click  
(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles SessionsInfo.Click  
  
    Dim vusername As System.Array  
    Dim loginTime As System.Array  
    Dim IPs As System.Array  
    Dim vSessionID As System.Array  
    Dim Result As Boolean  
  
    Result = server.GetAllSessionsInfo (vusername,  
loginTime, IPs, vSessionID)  
End Sub
```


Properties

Username

Syntax

```
Property Username As String
```

This property is used to set or get the username which gateway should use in connecting to this currency server.

Parameters

No Paramters.

Sample

```
Private Sub username_Click (ByVal sender As System.Object,  
ByVal e As System.EventArgs) Handles username.Click  
  
    'CurSer is predefined variable as CVertexCurSvrAPI  
    CurSer.username = " username"  
    Dim username As String = CurSer.username  
  
End Sub
```

Password

Syntax

```
Property Password As String
```

This property is used to set or get password that gateway should use in connecting to this currency server.

Parameters

No Paramaters.

Sample

```
Private Sub password_Click (ByVal sender As System.Object,  
ByVal e As System.EventArgs)Handles password.Click  
  
    'CurSer is predefined variable as CVertexCurSvrAPI  
    CurSer.Password = "Password"  
    Dim pass As String = CurSer.Password.  
  
End Sub
```

Port

Syntax

```
public property port () As Integer
```

This property is used to set and get port number that server will start listening to using [StartServer](#) method.

Parameters

No Paramters.

Sample

```
Private Sub port_Click (ByVal sender As System.Object,  
ByVal e As System.EventArgs)Handles port.Click  
  
    'CurSer is predefined variable as CVertexCurSvrAPI  
    CurSer.port = 4412  
    CurSer.BroadcastPrices ()  
    Dim port As Integer = CurSer.port  
  
End Sub
```

HighLowType

Syntax

```
public property port () As Integer
```

This property is used to set or get the high and low fields.

Parameters

No Parameters.

Sample

```
Private Sub HighLowType_Click(ByVal sender As System.Object,  
ByVal e As System.EventArgs) Handles HighLowType.Click  
  
    'CurSer is predefined variable as CVertexCurSvrAPI  
    CurSer.HighLowType =HighLowType.ComuteItLocally  
    Dim Type As HighLowType = CurSer.HighLowType  
  
End Sub
```

EnableMultiSession

Syntax

```
public property port () As Integer
```

This property is used to set or get whether to connect multi sessions with one username.

Parameters

No Parameters.

Sample

```
Private Sub MultiSession_Click(ByVal sender As System.Object,  
ByVal e As System.EventArgs) Handles MultiSession.Click  
  
    'CurSer is predefined variable as CVertexCurSvrAPI  
    CurSer.EnableMultiSession = True  
    Dim multiSession As Boolean = CurSer.EnableMultiSession  
    ()' it will return true.  
End Sub
```

IgnoreBidWhenGreaterThanAsk

Syntax

```
Property IgnoreBidWhenGreaterThanAsk () As Boolean
```

This property is used to set or get whether to filter the quote when the bid is greater than ask.

Parameters

No Parameters.

Sample

```
Private Sub IgnoreBid_Click (ByVal sender As System.Object,  
ByVal e As System.EventArgs) Handles IgnoreBid.Click  
  
    'CurSer is predefined variable as CVertexCurSvrAPI  
    CurSer.IgnoreBidWhenGreaterThanAsk = True  
    Dim result As Boolean = CurSer.IgnoreBidWhenGreaterThanAsk.  
  
End Sub
```

Events

ServerStatus

Syntax

```
Public Event ServerStatus(ByRef Status As ServerStatus)
```

This event Fire when the server status is changed.

Parameters

Key	Description
Status	Server status , value of type ServerStatus

SessionsChanged

Syntax

```
public Event SessionsChanged(ByRef Sessions As Integer)
```

Fire when the connected session's number changed because of new gateway connection or gateway disconnection.

Parameters

Key	Description
Sessions	Number of connected Sessions, value of type Integer

Data Type

ServerStatus

Represent Server status.

Value	Constant	Description
1	ServerOn	Server on
2	ServerOff	Server off
3	ServerPause	Server pause

HighLowType

Represents High and Low Types.

Value	Constant	Description
1	FromFeed	Exactly as what have been set in the AddPriceBuffer function.
2	ComputeItLocally	The API will compute the values within the day using the BID price.