Table of Contents

1 Snappy Fax Automation Server

Snappy Fax Automation Server supplies a set of properties and methods that can be called from your program to control snappy fax as an automation server much as you would control Microsoft Word or Excel through their object model.

The following is a description of the object model that is exposed by snappy fax automation server.

Sample code is given in Visual Basic only. Snappy Fax was not written in Visual Basic, the sample code is shown in Visual Basic since most programmers will be understand the syntax even if they don't use VB itself.

Note: Snappy fax automation is provided in the snappy fax desktop/client software. No automation methods are exposed in the snappy fax server software since all needed functionality can be accomplished within the desktop/client automation model, regardless of whether snappy fax is configured as a desktop application or a client to the fax server.

Registering the Automation Server

Before you can use the Automation server's object model in your program you must register it with Windows using regserver. After installing the Automation Server edition, start a command prompt running as administrator:

1. Click Start button and type cmd in search box
2. When cmd.exe appears in the list above, right click and select 'Run As Administrator'

In the command box, navigate to the program files folder where it was installed and type: sf5.exe /regserver

If the registration is successful, there will be no message

1.1 Object Model

1.1.1 Addcontact

Purpose: Add a contact to the address book indicated by the Address book name parameter.
Return value: boolean
Parameters:
Address book name, string
Contact Full Name, string
Fax Number, string
Contact Full Address, string
Group, string

Sample Usage (Visual Basic):

```
Dim ABName as string
Dim ContactName as string
Dim FaxNumber as string
Dim Address as string
Dim Group as string
Dim Snappy as Object

Snappy = CreateObject("SF5.SnappyFaxIntf")
ABName = "Default"
ContactName = "John Doe"
FaxNumber = "18007776543"
Address = "123 Elm Street" & vbcrlf & "City, State, zip"
Group = "Friends"
if Snappy.AddContact(ABName,ContactName,FaxNumber,Address,Group) then
  'success
```

1.1.2 AddressBookNames

Purpose: Obtain a list of address book names. The return value is a string with address book names separated by semi-colons
Return value: string
Parameters: None

Sample Usage (Visual Basic):

```
Dim ABNames as string
```

```
Dim Snappy as Object

Snappy = CreateObject("SF5.SnappyFaxIntf")
ABNames = ""
ABNames = Snappy.AddressBookNames
```

1.1.3 AddressBookPathOf

Purpose: Obtain the data folder location of the address book indicated by the string parameter
Return value: string
Parameters:

Address book name, string [in]

Sample Usage (Visual Basic):

```
Dim ABPath as string
Dim Snappy as Object

Snappy = CreateObject("SF5.SnappyFaxIntf")
ABPath = ""
ABPath = Snappy.AddressBookPathOf("shared")
```

1.1.4 AddressBookRecs

Purpose: Obtain the number of records in address book data table for the named address book indicated by the string parameter
Return value: long integer
Parameters: Address Book Name, string

Sample Usage (Visual Basic):

```
Dim NumABRecs as long
Dim Snappy as Object

Snappy = CreateObject("SF5.SnappyFaxIntf")
NumABRecs = Snappy.AddressBookRecs("Default")
```

Note: Snappy fax can have multiple address books, if the address book name is blank (blank string as parameter), the Default address book will be assumed.

1.1.5 AppMsgNumber

Purpose: Set the Windows message number your application will use to process status messages from snappy fax
Return value : None
Parameters: None

Sample Usage (Visual Basic):

```
Dim Snappy as Object

Snappy = CreateObject("SF5.SnappyFaxIntf")
Snappy.AppWindow = Me.Handle
Snappy.AppMsgNumber = WM_APP + 1000
```

1.1.6 AppWindow

Purpose: Set the Windows handle of your application window to receive status messages about a fax job
Return value : None
Parameters: None

Sample Usage (Visual Basic):

```
Dim Snappy as Object

Snappy = CreateObject("SF5.SnappyFaxIntf")
Snappy.AppWindow = Me.Handle
```

1.1.7 AutoJobId

Purpose: Set a job id of your choice to be used to track progress of fax job
Return value : None
Parameters: None

You will need to ensure that the AutoJobId is unique for each fax you send, otherwise the results may not be reliable when obtaining job status

See Also:

GetOutboxFaxDetailByAutoJobID

Sample Usage (Visual Basic):

```
Dim Snappy as Object
Dim JOB as integer
Snappy = CreateObject("SF5.SnappyFaxIntf")
Snappy.AppWindow = Me.Handle
Snappy.AppMsgNumber = WM_APP + 1000
JOB = 1
Snappy.AutoJobId = JOB
{prepare send}
 Snappy.SendFax
```

1.1.8 Billcode

Purpose: Set text to display in snappy fax's billing code field in the outbox data table
Return value : None
Parameters: None

Sample Usage (Visual Basic):

```
Dim Snappy as Object
Dim JOB as integer
Snappy = CreateObject("SF5.SnappyFaxIntf")
Snappy.BillCode = "B10929"
```

1.1.9 ConfigurationType

Purpose: Obtain the type of configuration of snappy fax
Return value : long integer (read only)
Parameters: None

return values:

0 = configured as desktop only
1 = configured as client to fax server

Sample Usage (Visual Basic):

```
Dim config as long
Dim Snappy as Object

Snappy = CreateObject("SF5.SnappyFaxIntf")
config = Snappy.ConfigurationType
if config = 1 then
  MessageBox("Configured as client to server")
else
  MessageBox("Configured as desktop")
end if
```

1.1.10 FaxNumber

Purpose: Set this property to fax number of the intended recipient
Return value: None
Parameters: None

Sample Usage (Visual Basic):

```
Dim FaxNumber as string
Dim Snappy as Object

Snappy = CreateObject("SF5.SnappyFaxIntf")
Snappy.FaxNumber = "18009871626"
```

You must specify this property, if not the exception 'No Fax Number Specified' will be raised when attempting to start the transmission.

Note: The fax number should always be specified exactly as it should be dialed. Snappy fax does not use dialing rules.

1.1.11 CoverAction_Comment

Purpose: Set to true to have snappy fax check the 'Comment' cover page action item
Return value : None
Parameters: None

Sample Usage (Visual Basic):

```
Dim Snappy as Object
Dim JOB as integer
Snappy = CreateObject("SF5.SnappyFaxIntf")
```

```
Snappy.UseCoverPage = true
Snappy.FileToFax = "ThursdayMenu.tif"
Snappy.CoverPageTemplate = "thursdaymenu.fct"
Snappy.CoverPageMemo = "This is the Bistro's menu for Thursday, Oct. 31"
Snappy.CoverPageSubject = "New Jersey Bistro Menu"
Snappy.CoverAction_Comment = true
```

1.1.12 CoverAction_Recycle

Purpose: Set to true to have snappy fax check the 'Recycle' cover page action item
Return value : None
Parameters: None

Sample Usage (Visual Basic):

```
Dim Snappy as Object
Dim JOB as integer
Snappy = CreateObject("SF5.SnappyFaxIntf")
Snappy.UseCoverPage = true
Snappy.FileToFax = "ThursdayMenu.tif"
Snappy.CoverPageTemplate = "thursdaymenu.fct"
Snappy.CoverPageMemo = "This is the Bistro's menu for Thursday, Oct. 31"
Snappy.CoverPageSubject = "New Jersey Bistro Menu"
Snappy.CoverAction_Recycle = true
```

1.1.13 CoverAction_Reply

Purpose: Set to true to have snappy fax check the 'Reply' cover page action item
Return value : None
Parameters: None

Sample Usage (Visual Basic):

```
Dim Snappy as Object
Dim JOB as integer
Snappy = CreateObject("SF5.SnappyFaxIntf")
Snappy.UseCoverPage = true
Snappy.FileToFax = "ThursdayMenu.tif"
Snappy.CoverPageTemplate = "thursdaymenu.fct"
Snappy.CoverPageMemo = "This is the Bistro's menu for Thursday, Oct. 31"
```

```
Snappy.CoverPageSubject = "New Jersey Bistro Menu"
Snappy.CoverAction_Reply = true
```

1.1.14 CoverAction_Review

Purpose: Set to true to have snappy fax check the 'Review' cover page action item
Return value : None
Parameters: None

Sample Usage (Visual Basic):

```
Dim Snappy as Object
Dim JOB as integer
Snappy = CreateObject("SF5.SnappyFaxIntf")
Snappy.UseCoverPage = true
Snappy.FileToFax = "ThursdayMenu.tif"
Snappy.CoverPageTemplate = "thursdaymenu.fct"
Snappy.CoverPageMemo = "This is the Bistro's menu for Thursday, Oct. 31"
Snappy.CoverPageSubject = "New Jersey Bistro Menu"
Snappy.CoverAction_Review = true
```

1.1.15 CoverAction_Urgent

Purpose: Set to true to have snappy fax check the 'Urgent' cover page action item
Return value : None
Parameters: None

Sample Usage (Visual Basic):

```
Dim Snappy as Object
Dim JOB as integer
Snappy = CreateObject("SF5.SnappyFaxIntf")
Snappy.UseCoverPage = true
Snappy.FileToFax = "ThursdayMenu.tif"
Snappy.CoverPageTemplate = "thursdaymenu.fct"
Snappy.CoverPageMemo = "This is the Bistro's menu for Thursday, Oct. 31"
Snappy.CoverPageSubject = "New Jersey Bistro Menu"
Snappy.CoverAction_Urgent = true
```

1.1.16 CoverOnlyFax

Purpose: Set to true if the fax will consist of a cover page only
Return value : None
Parameters: None

Sample Usage (Visual Basic):

```
Dim Snappy as Object
Dim JOB as integer
Snappy = CreateObject("SF5.SnappyFaxIntf")
Snappy.UseCoverPage = true
Snappy.CoverOnlyFax = true
Snappy.CoverPageTemplate = "mytemplate.fct"
Snappy.CoverPageMemo = "This is the Bistro's menu for Thursday, Oct. 31"
Snappy.CoverPageSubject = "New Jersey Bistro Menu"
```

1.1.17 CoverPageMemo

Purpose: Set to the desired text to appear in the memo area of the cover page
Return value : None
Parameters: None

Sample Usage (Visual Basic):

```
Dim Snappy as Object
Dim JOB as integer
Snappy = CreateObject("SF5.SnappyFaxIntf")
Snappy.UseCoverPage = true
Snappy.FileToFax = "ThursdayMenu.tif"
Snappy.CoverPageTemplate = "thursdaymenu.fct"
Snappy.CoverPageMemo = "This is the Bistro's menu for Thursday, Oct. 31"
Snappy.CoverPageSubject = "New Jersey Bistro Menu"
```

1.1.18 CoverPageSubject

Purpose: Set to the desired text to appear on the subject line of the cover page
Return value : None
Parameters: None

Sample Usage (Visual Basic):

```
Dim Snappy as Object
Dim JOB as integer
Snappy = CreateObject("SF5.SnappyFaxIntf")
Snappy.UseCoverPage = true
Snappy.FileToFax = "ThursdayMenu.tif"
Snappy.CoverPageTemplate = "thursdaymenu.fct"
Snappy.CoverPageMemo = "This is the Bistro's menu for Thursday, Oct. 31"
Snappy.CoverPageSubject = "New Jersey Bistro Menu"
```

1.1.19 CoverPageTemplate

Purpose: Set to the file name of the desired cover page template
Return value : None
Parameters: None

Sample Usage (Visual Basic):

```
Dim Snappy as Object
Dim JOB as integer
Snappy = CreateObject("SF5.SnappyFaxIntf")
Snappy.UseCoverPage = true
Snappy.FileToFax = "ThursdayMenu.tif"
Snappy.CoverPageTemplate = "thursdaymenu.fct"
Snappy.CoverPageMemo = "This is the Bistro's menu for Thursday, Oct. 31"
Snappy.CoverPageSubject = "New Jersey Bistro Menu"
```

Note: if CoverPageTemplate is left blank then snappy fax will use the default cover page template defined in its settings. If left blank and no default template is defined in its settings, the exception "No Cover Page Template defined" will be raised. If the template designated here does not exist in snappy fax's cover page folder the exception 'cover page template does not exist' will be raised.

1.1.20 FaxRecipient

Purpose: Set this property to name of the fax recipient
Return value: None
Parameters: None

Sample Usage (Visual Basic):

```
Dim FaxRecipient as string
Dim Snappy as Object

Snappy = CreateObject("SF5.SnappyFaxIntf")
Snappy.FaxRecipient = "John Doe"
```

You must specify a recipient name, if this property is not specified, the exception 'No Fax Recipient Name Specified' will be raised then attempting to start the fax transmission

1.1.21 FaxRouting

Purpose: Set this property to indicate the fax routing of the fax file to be transmitted
Return value: None
Parameters: None

Sample Usage (Visual Basic):

```
Dim Faxrouting as string
Dim Snappy as Object

Snappy = CreateObject("SF5.SnappyFaxIntf")
Snappy.FaxRouting = "R"
```

When starting a fax job via automation, set the FaxRouting to either "R" or "S".

Fax routing of "R" indicates send the fax normally (Regular fax) via modem. Fax routing of "S" is only valid if snappy fax is configured as a client to the fax server software.

If you set this value to "S" and snappy fax is not configured as a client to the fax server, the exception 'Server Routing not supported in configuration' will be raised.

If you set this value to anything other than "S" or "R" then the exception 'Invalid Fax routing Method specified' will be raised

1.1.22 SendASAP

Purpose: Set this property to indicate the whether the fax should be sent as soon as possible
Return value: None

Parameters: None

Note this property should only be used when the Faxrouting is set to 'S'.

Sample Usage (Visual Basic):

```
Dim Faxrouting as string
Dim Snappy as Object

Snappy = CreateObject("SF5.SnappyFaxIntf")
Snappy.FaxRouting = "S"
Snappy.SendASAP = true
```

If nothing is specified for this property, true will be assumed

1.1.23 SendDate

Purpose: Set this property to indicate the target date when a fax job is not to be sent immediately
Return value: None
Parameters: None

Note this property should only be used when the Faxrouting is set to 'S' and the SendASAP property has been set to false

Sample Usage (Visual Basic):

```
Dim Faxrouting as string
Dim DateToSend as Date
Dim TimeToSend as Date
Dim Snappy as Object

Snappy = CreateObject("SF5.SnappyFaxIntf")
Snappy.FaxRouting = "S"
Snappy.SendASAP = false
DateToSend = #4/30/2015#
TimeToSend = #5:00:00 PM#
Snappy.SendDate = DateToSend
Snappy.Sendtime = TimeToSend
```

If you set SendASAP to false and do not set the SendDate *and* SendTime properties, SendASAP will revert to true

1.1.24 SendTime

Purpose: Set this property to indicate the target date when a fax job is not to be sent immediately
Return value: None
Parameters: None

Note this property should only be used when the Faxrouting is set to 'S' and the SendASAP property has been set to false

Sample Usage (Visual Basic):

```
Dim Faxrouting as string
Dim DateToSend as Date
Dim TimeToSend as Date
Dim Snappy as Object

Snappy = CreateObject("SF5.SnappyFaxIntf")
Snappy.FaxRouting = "S"
Snappy.SendASAP = false
DateToSend = #4/30/2015#
TimeToSend = #5:00:00 PM#
Snappy.SendDate = DateToSend
Snappy.Sendtime = TimeToSend
```

If you set SendASAP to false and do not set the SendDate *and* SendTime properties, SendASAP will revert to true

1.1.25 FileToFax

Purpose: Set this property to indicate the file that is to be transmitted
Return value: string (read, write)
Parameters: None

Sample Usage (Visual Basic):

```
Dim FaxFile as string
Dim Snappy as Object

Snappy = CreateObject("SF5.SnappyFaxIntf")
Snappy.FileToFax = "c:\FaxImage.tif"
```

When starting a fax job via automation, set the FileToFax to the file you want to transmit.

Supported file types:

.txt (text files)
.tif, .tiff (tif files)
.jpg, .jpeg (jpg files)
.bmp (bitmap files)
.pdf (pdf files)
.doc, .docx (Microsoft Word Files) Note: Word must be installed on system
.xls, .xlsx (Microsoft Excel Files) Note: Excel must be installed on system
.odt (Open Office document files) Note: Open Office must be installed on system
.ods (Open Office spreadsheet files) Note: Open Office must be installed on system

Note: Attempting to set the FileToFax to any other file type will result in an exception 'File Type is not Supported for Fax Automation'. If the file does not exist the exception 'File Specified does not exist' will be raised.

1.1.26 FirstAddressBookKey

Purpose: Obtain the Key of the first (oldest) record in the address book indicated by the string parameter
Return value: long integer
Parameters: Address book name, string

Sample Usage (Visual Basic):

```
Dim Key as integer
Dim Snappy as Object

Snappy = CreateObject("SF5.SnappyFaxIntf")
Key = Snappy.FirstAddressBookKey("Default")
```

Note: if the address book name is passed as a blank string, the Default address book will be assumed

1.1.27 FirstInBoxKey

Purpose: Obtain the key of the first (oldest) record in the inbox data table.
Return value: long integer
Parameters: None

Sample Usage (Visual Basic):

```
Dim FirstInKey as long
Dim Snappy as Object

Snappy = CreateObject("SF5.SnappyFaxIntf")
FirstInKey = Snappy.FirstInboxKey
```

1.1.28 FirstOutBoxKey

Purpose: Obtain the key of the first (oldest) record in the outbox data table.
Return value: long integer
Parameters: None

Sample Usage (Visual Basic):

```
Dim FirstOutKey as long
Dim Snappy as Object

Snappy = CreateObject("SF5.SnappyFaxIntf")
FirstOutKey = Snappy.FirstOutboxKey
```

1.1.29 FirstServerInBoxKey

Purpose: Obtain the key of the first (oldest) record in the fax Server's inbox data table.
Return value: long integer
Parameters: None

Sample Usage (Visual Basic):

```
Dim FirstInKey as long
Dim Snappy as Object

Snappy = CreateObject("SF5.SnappyFaxIntf")
FirstInKey = Snappy.FirstServerInboxKey
```

Note: Only applies if snappy fax is configured as a client to snappy fax server, otherwise the exception 'Client not configured for fax server' will be raised

1.1.30 FirstServerJobKey

Purpose: Obtain the key of the first (oldest) record in the fax Server's pending jobs data table.
Return value: long integer
Parameters: None

Sample Usage (Visual Basic):

```
Dim FirstKey as long
Dim Snappy as Object

Snappy = CreateObject("SF5.SnappyFaxIntf")
FirstKey = Snappy.FirstServerJobKey
```

Note: Only applies if snappy fax is configured as a client to snappy fax server, otherwise the exception 'Client not configured for fax server' will be raised

1.1.31 FirstServerOutBoxKey

Purpose: Obtain the key of the first (oldest) record in the fax Server's outbox data table.
Return value: long integer
Parameters: None

Sample Usage (Visual Basic):

```
Dim FirstKey as long
Dim Snappy as Object

Snappy = CreateObject("SF5.SnappyFaxIntf")
FirstKey = Snappy.FirstServerOutboxKey
```

Note: Only applies if snappy fax is configured as a client to snappy fax server, otherwise the exception 'Client not configured for fax server' will be raised

1.1.32 GetAddressBookDetails

Purpose: Retrieve detail information of a record in the outbox data table
Return value: None
Parameters:

- ABname : string [in]
- Key : integer [in]
- sFirstName : string [out]
- sLastName : string [out]
- sFullName : string [out]
- sFaxNumber : string [out]
- sAddress : string [out]
- sEmail : string [out]
- NextKey : long integer [out]

Abname is the name of the desired address book, if left blank the default address book is assumed. Key is the key of the record in the address book table you are interested in.

Sample Usage (Visual Basic):

Refer to the example for GetInboxFaxDetails

1.1.33 GetInboxFax

Purpose: Retrieve the fax image of a record in the inbox data table and save to a .tif file or a .pdf file
Return value: boolean (true if successful)
Parameters:

- Key : integer
- sFile : string

Key is the key of the record in the inbox data table you want to retrieve the fax image for sFile is the desired file name to save the fax image to. The file extension must either be .tif or .pdf, if a file with another file extension is passed in this parameter an exception will be raised.

Sample Usage (Visual Basic):

```
Dim Key as long
Dim sFile as string
Dim Snappy as Object

Snappy = CreateObject("SF5.SnappyFaxIntf")
Key = 1909
sFile = "c:\MyTifFile.tif"
if Snappy.GetInboxFax(Key,sFile) then
  'do something with image file
```

1.1.34 GetCoverPageTemplateNames

Purpose: Obtain a list of cover page template file names. The return value is a string with the names separated by semi-colons
Return value: string
Parameters: None

Sample Usage (Visual Basic):

```
Dim TemplateNames() as string
Dim TemplateName as string
Dim i as integer
Dim Snappy as Object

Snappy = CreateObject("SF5.SnappyFaxIntf")

TemplaateName = Snappy.GetCoverPageTemplateNames
TemplateNames = Split(TemplateName,";")
for i = 0 to TemplateNames.Length - 1
  cbDropDownListBox.Items.Add(TemplateNames(i))
next
```

Note: Above example shows how to populate a drop down list box with cover page template names

1.1.35 GetDefaultCoverPageTemplateName

Purpose: Obtain the default cover page template file name.
Return value: string
Parameters: None

Sample Usage (Visual Basic):

```
Dim TemplateName as string
Dim i as integer
Dim Snappy as Object

Snappy = CreateObject("SF5.SnappyFaxIntf")

TemplaateName = Snappy.GetDefaultCoverPageTemplateName
```

1.1.36 GetInboxFaxDetails

Purpose: Retrieve detail information of a record in the inbox data table
Return value: None
Parameters:

- Key : integer [in]
- Pages : integer [in]
- Date : Date [out]
- sFax : string [out]
- sFaxResult : string [out]

- NextKey : long integer [out]

Key is the key of the record in the inbox data table you want to retrieve the details for The Number of pages in the fax, the date /time of the fax, the fax number of the caller, the result of the fax and the key of the next record in the inbox data table are returned in the Pages, Date, sFax, sFaxResult and NextKey parameters.

Sample Usage (Visual Basic):

Here is a detailed example showing how to iterate through all inbox records and populate a grid with the results...

```
Sub MyGetInboxFaxDetails(Key As Integer, ByRef Pages As Integer, ByRef sDate As
Date, ByRef sFax As String, ByRef sResult As String, ByRef NextKey As Integer)
        Snappy.getInboxFaxDetails(Key, Pages, sDate, sFax, sResult, NextKey)
End Sub
Private Sub btnGetLastDetail_Click(sender As Object, e As EventArgs) Handles
btnGetLastDetail.Click
        Dim Pages As Long
        Dim Key As Long
        Dim LastKey As Long
        Dim FirstKey As Long
        Dim count As Long
        Dim NumRecs As Long
        Dim sFax As String
        Dim sResult As String
        Dim NextKey As Integer
        Dim fDate As Date

        sTime = ""
        sFax = ""
        Snappy = CreateObject("SF5.SnappyFaxIntf")
        NumRecs = Snappy.InboxRecs
        LastKey = Snappy.LastInboxKey
        FirstKey = Snappy.FirstInboxKey
        count = 1
        Key = FirstKey
        NextKey = 0
        MyGetInboxFaxDetails(Key, Pages, fDate, sFax, sResult, NextKey)
        Grid1.Rows.Add(Key, sFax, fDate, Pages, sResult)
        Do While NextKey <> -1
            count = count + 1
            Key = NextKey
            NextKey = -1
            sTime = ""
            sFax = ""
            sResult = ""
            MyGetInboxFaxDetails(Key, Pages, fDate, sFax, sResult, NextKey)
            Grid1.Rows.Add(Key, sFax, fDate, Pages, sResult)
            inProgressBar.Value = (count * 100) / NumRecs
        Loop
    End Sub
```

1.1.37 GetServerInboxFax

Purpose: Retrieve the fax image of a record in the fax server's inbox data table and save to a .tif file or a .pdf file
Return value: boolean (true if successful)
Parameters:

- Key : integer
- sFile : string

Key is the key of the record in the inbox data table you want to retrieve the fax image for
sFile is the desired file name to save the fax image to. The file extension must either be .tif or .pdf, if a file with another file extension is passed in this parameter an exception will be raised.

Sample Usage (Visual Basic):

```
Dim Key as long
Dim sFile as string
Dim Snappy as Object

Snappy = CreateObject("SF5.SnappyFaxIntf")
Key = 1909
sFile = "c:\MyTifFile.tif"
if Snappy.GetInServerInboxFax(Key,sFile) then
  'do something with image file
```

1.1.38 GetServerInboxFaxDetails

Purpose: Retrieve detail information of a record in the fax server's inbox data table
Return value: boolean
Parameters:

- Key : integer [in]
- Date : Date [out]
- sFax : string [out]
- sFaxResult : string [out]
- sCheckedOutby : string [out]
- Pages : integer [out]
- NextKey : integer [out]
- CheckedOut : boolean [out]

The key of the desired record is passed in the Key parameter. Snappy fax will pass back the other parameters from the fax detail record. The CheckedOut parameter will indicate if the record is checked out and the sCheckedOutby parameter will contain the client computer name that has checked out the record if it is checked out.

Refer to the example visual basic code under the topic GetInBoxFaxDetails for proper usage of this method

1.1.39 GetOutboxFax

Purpose: Retrieve the fax image of a record in the outbox data table and save to a .tif file
Return value: boolean (true if successful)
Parameters:

- Key : integer
- sFile : string

Key is the key of the record in the outbox data table you want to retrieve the fax image for
sFile is the desired file name to save the fax image to, *must* be a .tif file

Sample Usage (Visual Basic):

```
Dim Key as long
Dim sFile as string
Dim Snappy as Object

Snappy = CreateObject("SF5.SnappyFaxIntf")
Key = 1909
sFile = "c:\MyTifFile.tif"
if Snappy.GetOutboxFax(Key,sFile) then
  'do something with image file
```

1.1.40 GetOutboxFaxDetails

Purpose: Retrieve detail information of a record in the outbox data table
Return value: None
Parameters:

- Key : integer [in]
- Pages : integer [out]
- Date : Date [out]
- sFax : string [out]
- sSentTo : string [out]
- sFaxResult : string [out]
- NextKey : long integer [out]

Key is the key of the record in the inbox data table you want to retrieve the details for
The Number of pages in the fax, the date /time of the fax, the fax number called, the recipient's name to whom the fax was sent, the result of the fax and the key of the next record in the inbox data table are returned in the Pages, Date, sFax, sFaxResult and NextKey

parameters.

Sample Usage (Visual Basic):

Here is a detailed example showing how to iterate through all outbox records and populate a grid with the results...

```
Sub MyGetOutboxFaxDetails(Key As Integer, ByRef Pages As Integer, ByRef sDate As
Date, ByRef sFax As String,ByRef sSentTo As String, ByRef sResult As String,
ByRef NextKey As Integer)
        {Note: Snappy has already been instantiated to an object with global
scope}
        Snappy.getOutboxFaxDetails(Key, Pages, sDate, sFax,sSentTo, sResult,
NextKey)
End Sub
Private Sub btnGetLastDetail_Click(sender As Object, e As EventArgs) Handles
btnGetLastDetail.Click
        Dim Pages As Long
        Dim Key As Long
        Dim LastKey As Long
        Dim FirstKey As Long
        Dim count As Long
        Dim NumRecs As Long
        Dim sSentTo As String
        Dim sFax As String
        Dim sResult As String
        Dim NextKey As Integer
        Dim fDate As Date
        sFax = ""
        sResult = ""
       Snappy = CreateObject("SF5.SnappyFaxIntf")
        NumRecs = Snappy.InboxRecs
        LastKey = Snappy.LastInboxKey
        FirstKey = Snappy.FirstInboxKey
        count = 1
        Key = FirstKey
        NextKey = 0
        MyGetInboxFaxDetails(Key, Pages, fDate, sFax,sSentto, sResult, NextKey)
        Grid1.Rows.Add(Key, sFax,sSentto, fDate, Pages, sResult)
        Do While NextKey <> -1
            count = count + 1
            Key = NextKey
            NextKey = -1
            sFax = ""
            sResult = ""
            MyGetOutboxFaxDetails(Key, Pages, fDate, sFax,sSentto, sResult,
NextKey)
            Grid1.Rows.Add(Key, sFax,sSentTo, fDate, Pages, sResult)
            OutProgressBar.Value = (count * 100) / NumRecs
        Loop
    End Sub
```

1.1.41 GetOutboxFaxDetailByAutoJobID

Purpose: Retrieve detail information of a record in the outbox data table referencing it by the AutoJobId
Return value: None
Parameters:

- Key : integer [in]
- Pages : integer [out]
- Date : Date [out]
- sFax : string [out]
- sSentTo : string [out]
- sFaxResult : string [out]

Key is the AutoJobID used when you originally made a call to SendFax. This number (integer) should always be unique, it is up to you to enforce that. If the number is not unique then the call to this method will produce unreliable results.
The Number of pages in the fax, the date /time of the fax, the fax number called, the recipient's name to whom the fax was sent, the result of the fax and the key of the next record in the inbox data table are returned in the Pages, Date, sFax, sFaxResult parameters.

Here is a detailed example showing how to iterate through all outbox records and populate a grid with the results...

```
Sub MyGetOutboxFaxDetailByAutoJobID(Key As Integer, ByRef Pages As Integer, ByRef
sDate As Date, ByRef sFax As String,ByRef sSentTo As String, ByRef sResult As
String)
        {Note: Snappy has already been instantiated to an object with global
scope}
        Snappy.getOutboxFaxDetailByAutoJobID(Key, Pages, sDate, sFax,sSentTo,
sResult)
End Sub
Private Sub btnGetLastDetail_Click(sender As Object, e As EventArgs) Handles
btnGetLastDetail.Click
        Dim Pages As Long
        Dim Key As Long
        Dim sSentTo As String
        Dim sFax As String
        Dim sResult As String
        Dim fDate As Date
        sFax = ""
        sResult = ""
       Snappy = CreateObject("SF5.SnappyFaxIntf")
        MyGetInboxFaxDetailByAutoJobID(Key, Pages, fDate, sFax,sSentto, sResult)
        Grid1.Rows.Add(Key, sFax,sSentto, fDate, Pages, sResult)
    End Sub
```

1.1.42 GetServerOutboxFax

Purpose: Retrieve the fax image of a record in the fax server's outbox data table and save to a .tif file or a .pdf file
Return value: boolean (true if successful)
Parameters:

- Key : integer
- sFile : string

Key is the key of the record in the inbox data table you want to retrieve the fax image for sFile is the desired file name to save the fax image to. The file extension must either be .tif or .pdf, if a file with another file extension is passed in this parameter an exception will be raised.

Sample Usage (Visual Basic):

```
Dim Key as long
Dim sFile as string
Dim Snappy as Object

Snappy = CreateObject("SF5.SnappyFaxIntf")
Key = 1909
sFile = "c:\MyTifFile.tif"
if Snappy.GetInServerOutboxFax(Key,sFile) then
  'do something with image file
```

1.1.43 GetServerJobDetails

Purpose: Retrieve detail information of a record in the fax server's pending jobs data table
Return value: boolean
Parameters:

- Key : integer [in]
- sSubmitter : string [out]
- sClient : string [out]
- sSendto : string [out]
- DateTimeToSend : Date [out]
- SendImmediately : boolean [out]
- InProgress : boolean [out]
- NextKey : integer [out]

The key of the desired record is passed in the Key parameter. Snappy fax will pass back the other parameters from the job detail record.

See Also:

GetServerOutBoxFaxDetails
GetServerInBoxFaxDetails

1.1.44 GetServerOutboxFaxDetails

Purpose: Retrieve detail information of a record in the fax server's outbox data table
Return value: boolean
Parameters:

- Key : integer [in]
- Date : Date [out]
- sSentto : string [out]
- sFax : string [out]
- sFaxResult : string [out]
- sSubmitter : string [out]
- Pages : integer [out]
- NextKey : integer [out]

The key of the desired record is passed in the Key parameter. Snappy fax will pass back the other parameters from the fax detail record. The sSubmitter parameter will contain the network name of the client that submitted the job to the fax server.

Refer to the example visual basic code under the topic GetOutboxFaxDetails for proper usage of this method

1.1.45 InboxRecs

Purpose: Obtain the number of records in the inbox data table.
Return value : long integer
Parameters: None

Sample Usage (Visual Basic):

```
Dim NumInboxRecs as long
Dim Snappy as Object

Snappy = CreateObject("SF5.SnappyFaxIntf")
NumInboxRecs = Snappy.InBoxRecs
```

1.1.46 IsInboxImageValid

Purpose: Test if the image data that is stored in the inbox record indicated by Key is valid
Return value: boolean
Parameters:

Key , integer [in]

Sample Usage (Visual Basic):

```
Dim Key as long
Dim Snappy as Object

Snappy = CreateObject("SF5.SnappyFaxIntf")
Key = 10902
if Snappy.IsInboxImageValid(Key) then
  'yes do something
```

1.1.47 IsOutboxImageValid

Purpose: Test if the image data that is stored in the outbox record indicated by Key is valid
Return value: boolean
Parameters:

Key , integer [in]

Sample Usage (Visual Basic):

```
Dim Key as long
Dim Snappy as Object

Snappy = CreateObject("SF5.SnappyFaxIntf")
Key = 10902
if Snappy.IsOutboxImageValid(Key) then
  'yes do something
```

1.1.48 IsServerInboxImageValid

Purpose: Test if the image data that is stored in the fax server's inbox record indicated by Key is valid
Return value: boolean
Parameters:

Key , integer [in]

Sample Usage (Visual Basic):

```
Dim Key as long
Dim Snappy as Object
```

```
Snappy = CreateObject("SF5.SnappyFaxIntf")
Key = 10902
if Snappy.IsServerInboxImageValid(Key) then
  'yes do something
```

1.1.49 IsServerOutboxImageValid

Purpose: Test if the image data that is stored in the fax server's outbox record indicated by Key is valid
Return value: boolean
Parameters:

Key , integer [in]

Sample Usage (Visual Basic):

```
Dim Key as long
Dim Snappy as Object

Snappy = CreateObject("SF5.SnappyFaxIntf")
Key = 10902
if Snappy.IsServerOutboxImageValid(Key) then
  'yes do something
```

1.1.50 IsValidInboxKey

Purpose: Test if Key is a valid index key for the inbox table
Return value: boolean
Parameters:

Key , integer [in]

Sample Usage (Visual Basic):

```
Dim Key as long
Dim Snappy as Object

Snappy = CreateObject("SF5.SnappyFaxIntf")
Key = 10902
if Snappy.IsValidInBoxKey(Key) then
  'yes do something
```

1.1.51 IsValidOutboxKey

Purpose: Test if Key is a valid index key for the outbox table
Return value: boolean
Parameters:

Key , integer [in]

Sample Usage (Visual Basic):

```
Dim Key as long
Dim Snappy as Object

Snappy = CreateObject("SF5.SnappyFaxIntf")
Key = 10
if Snappy.IsValidOutBoxKey(Key) then
  'yes do something
```

1.1.52 OutboxRecs

Purpose: Obtain the number of records in the outbox data table.
Return value: long integer
Parameters: None

Sample Usage (Visual Basic):

```
Dim NumOutboxRecs as long
Dim Snappy as Object

Snappy = CreateObject("SF5.SnappyFaxIntf")
NumOutboxRecs = Snappy.OutBoxRecs
```

1.1.53 LastAddressBookKey

Purpose: Obtain the Key of the last (newest) record in the address book indicated by the string parameter
Return value: long integer
Parameters: Address book name, string

Sample Usage (Visual Basic):

```
Dim Key as integer
Dim Snappy as Object
```

```
Snappy = CreateObject("SF5.SnappyFaxIntf")
Key = Snappy.LastAddressBookKey("Default")
```

Note: if the address book name is passed as a blank string, the Default address book will be assumed

1.1.54 LastInboxKey

Purpose: Obtain the key of the last (most recent) record in the Inbox data table.
Return value: long integer
Parameters: None

Sample Usage (Visual Basic):

```
Dim LastInKey as long
Dim Snappy as Object

Snappy = CreateObject("SF5.SnappyFaxIntf")
LastInKey = Snappy.LastInKey
```

1.1.55 LastOutboxKey

Purpose: Obtain the key of the last (most recent) record in the outbox data table.
Return value: long integer
Parameters: None

Sample Usage (Visual Basic):

```
Dim LastOutKey as long
Dim Snappy as Object

Snappy = CreateObject("SF5.SnappyFaxIntf")
LastOutKey = Snappy.LastOutboxKey
```

1.1.56 LastServerInboxKey

Purpose: Obtain the key of the last (most recent) record in the fax Server's inbox data table.
Return value: long integer
Parameters: None

Sample Usage (Visual Basic):

```
Dim LastInKey as long
```

```
Dim Snappy as Object

Snappy = CreateObject("SF5.SnappyFaxIntf")
LastInKey = Snappy.LastServerInboxKey
```

Note: Only applies if snappy fax is configured as a client to snappy fax server, otherwise the exception 'Client not configured for fax server' will be raised

1.1.57 LastServerJobKey

Purpose: Obtain the key of the last (most recent) record in the fax Server's pending jobs data table.
Return value: long integer
Parameters: None

Sample Usage (Visual Basic):

```
Dim LastKey as long
Dim Snappy as Object

Snappy = CreateObject("SF5.SnappyFaxIntf")
LastKey = Snappy.LastServerJobKey
```

Note: Only applies if snappy fax is configured as a client to snappy fax server, otherwise the exception 'Client not configured for fax server' will be raised

1.1.58 LastServerOutboxKey

Purpose: Obtain the key of the last (most recent) record in the fax Server's outbox data table.
Return value: long integer
Parameters: None

Sample Usage (Visual Basic):

```
Dim LastKey as long
Dim Snappy as Object

Snappy = CreateObject("SF5.SnappyFaxIntf")
LastKey = Snappy.LastServerOutboxKey
```

Note: Only applies if snappy fax is configured as a client to snappy fax server, otherwise the exception 'Client not configured for fax server' will be raised

1.1.59 MemoField

Purpose: Set text to display in snappy fax's memo field in the outbox data table
Return value : None
Parameters: None

Sample Usage (Visual Basic):

```
Dim Snappy as Object
Dim JOB as integer
Snappy = CreateObject("SF5.SnappyFaxIntf")
Snappy.MemoField = "Project 7"
```

1.1.60 NextInboxKey

Purpose: Obtain the next key that will be assigned to a new inbox record
Return value: long integer
Parameters: None

Sample Usage (Visual Basic):

```
Dim NextKey as long
Dim Snappy as Object

Snappy = CreateObject("SF5.SnappyFaxIntf")
NextKey = Snappy.NextInBoxKey
```

1.1.61 NextOutboxKey

Purpose: Obtain the next key that will be assigned to a new outbox record
Return value: long integer
Parameters: None

Sample Usage (Visual Basic):

```
Dim NextKey as long
Dim Snappy as Object

Snappy = CreateObject("SF5.SnappyFaxIntf")
NextKey = Snappy.NextOutBoxKey
```

1.1.62 SendFax

Purpose: Call this method to start the fax transmission after setting the required property values
Return value: None
Parameters: None

Sample Usage (Visual Basic):

```
Dim FaxNumber as string
Dim FaxRecipient as string
Dim FaxFile as string
Dim Snappy as Object

Snappy = CreateObject("SF5.SnappyFaxIntf")
Snappy.FaxNumber = "18009871626"
Snappy.FaxRouting = "R"
Snappy.FiletoFax = "c:\mytiffile.tif"
Snappy.FaxRecipient = "John Doe"
Snappy.AutoJobId = 121098  {some unique integer value}
Snappy.SendFax
```

1.1.63 UseCoverPage

Purpose: Set to true if the fax will contain a cover page
Return value : None
Parameters: None

Sample Usage (Visual Basic):

```
Dim Snappy as Object
Dim JOB as integer
Snappy = CreateObject("SF5.SnappyFaxIntf")
Snappy.UseCoverPage = true
Snappy.FileToFax = "ThursdayMenu.tif"
Snappy.CoverPageTemplate = "mytemplate.fct"
Snappy.CoverPageMemo = "This is the Bistro's menu for Thursday, Oct. 31"
Snappy.CoverPageSubject = "New Jersey Bistro Menu"
```

1.2 Advanced Topics

1.2.1 Tracking Fax Progress

You can track the progress of a fax that is submitted to snappy fax for processing by using the AppWindow, AppMsgNumber and AutoJobID properties.

- Set the AppWindow property to the window handle of your application's window.
- Set the AppMsgNumber to your desired message number, usually WM_APP + (some number of your choosing)
- Set the AutoJobID property to a unique integer id that you can use internally in your application to know which fax job is being referred to in the status messages snappy fax sends to your application.
- Setup a message handler in your application to receive and process the message number set in AppMsgNumber sent from snappy fax to your window handle specified in AppWindow.

When a fax job has been ***started*** by snappy fax it will send a message to your application as follows:

PostMessage(AppWindow,AppMsgNumber,1,LParam(AutoJobId))

Note the wParam is 1 indicates that the job is now being started (transmission will begin), the job being referred to is sent in the lparam

When the fax job has ***completed*** with or without success, snappy fax will send a message to your application as follows:

PostMessage(AppWindow,AppMsgNumber,ErrorCode,LParam(AutojobId))

Note that in the completion message the wParam is the ErrorCode which can never be 1, so a 1 in this message parameter indicates the start of the fax job transmission. Any other value indicates this is a completion message and the wParam contains the error code. Refer to the topic Error Codes for a complete listing of all error codes.

See Also: GetOutboxFaxDetailsByAutoJobID

1.2.2 Error Codes

All possible error codes and their meaning are presented below. Error code is the wParam sent by snappy fax in the completion message when a fax job has been completed either successfully or not.

Error Code	Meaning
0	Fax was Successful
6006	Fatal Time Out
6014	Abort Carrier Lost
6017	Remote Send Disconnect
6018	Remote did not reply to MCF
6021	Prepare fax job failed
6023	Remote did not send CFR
6028	Fax Session in hung state, aborted
6030	Unexpected Disconnect

6049	Training Failed
8060	Not compatible with remote fax
8061	Modem reported ERROR session cannot continue
8062	Training could not be completed
8063	Modem reported ERROR during initialization
8064	Called fax number BUSY
8065	Called fax number answered with voice
8067	Modem reported NO DIAL TONE
8068	Failed to connect to remote fax
8069	Fax failed in mid-session
8070	Fax failed at page end
8074	Remote Fax did not answer
8076	Modem did not respond during initialization

As you can see anything other than 0 is a failure.

Index

- 6 -

- 8 -

- A -

- B -

- C -

- E -

- F -

- G -

- I -

- L -

- M -

- N -

- O -

- P -

- R -

- S -

- T -

- U -