NETMANAGE

NNTP Client ActiveX Control

The following Help Topics are available:

For Help on Help, Press F1

Net*Manage*

NetManage develops, markets and supports an integrated set of TCP/IP inter-networking applications and development tools for Microsoft Windows. NetManage software facilitates communication, productivity and the administration of personal computers across dissimilar networking environments. The Company's award-winning product families include Chameleon and ECCO.

The company is located at 10725 North De Anza Blvd. Cupertino, CA 95014, USA

Phone: 408-973-7171 Fax: 408-973-8272.

International phone: +972-4-8550234 Fax +972-4-8550122

General

The Networking News Transfer Protocol (NNTP) Client Control implements the basic client NNTP Protocol as specified by RFC977, *Network News Transfer Protocol*. THE NNTP Control also implements NNTP extension commands as documented in the Internet-Draft on Common NNTP Extensions. For questions on Internet-Drafts contact, Internet-Drafts@CNRI.Reston.VA.US.

The NNTP Control provides a reusable component that allows applications to access NNTP news servers. It provides news reading and posting capabilities.

There should be no speed overhead and response delay other than the one given by the network. This control uses and is dependent on DocStream objects.

NNTP Commands

The following table summarizes the NNTP Client commands as specified in RFC977 and the NNTP extension commands as specified in Internet-Draft on Common NNTP Extensions.

NNTP Client Commands	NNTP Extension Commands
ARTICLE	AUTHINFO
GROUP	LISTGROUP
LIST	LIST OVERVIEW.FMT
NEWSGROUP	XHDR
POST	XMOTD
QUIT	XOVER

Naming Conventions

Objects described in the Properties, Methods and Events section are preceded by the required parameter: object. During execution object translates to the name of the control.

Properties

Properties set the attributes for NNTP behavior.

Note: Some common ActiveX properties of the control, such as Name, Index, About Box, and others, may appear in the Object Browser but are not documented here.

Following is a list of all properties supported by the NNTP Control.

ArticleNumbersSupported

Busy

DocInput

DocOutput

EnableTimer

Errors

LastUpdate

NotificationMode

OverviewSupported

PostingAllowed

ProtocolState

ProtocolStateString

RemoteHost

RemotePort

ReplyCode

ReplyString

State

StateString

Timeout

URL

ArticleNumbersSupported

Description

If True, the GetArticleNumbers method may be used to retrieve a list of article numbers for a newsgroup. This property has no meaning before the connection to the server has been established.

Syntax

object.**ArticleNumbersSupported**

The object placeholder is required and evaluates to the name of the relevant control or collection during object during execution.

Permission

R (Read-only).

Availability

R(Runtime).

Data Type

Boolean.

Default Value

False.

Range

True or False

Comments

Examine this property after a connection is established to determine if the server supports the GetArticleNumber method (LISTGROUP command).

Busy

Description

Indicates a command is in progress.

Syntax

object.**Busy**

The object placeholder is required and evaluates to the name of the relevant control or collection during execution.

Permission

R (Read only).

Availability

R (Runtime).

Data Type

Boolean.

Default Value

N/A.

Range

True or False

DocInput

See Also

Description

Object describing input information for the document being transferred.

Syntax

object.**DocInput**

The object placeholder is required and evaluates to the name of the relevant control or collection during execution.

Permission

R (Read-only).

Availability

R (Runtime).

Data Type

DocInput.

Default Value

N/A.

Range

N/A

Comments

The DocInput object provides a more powerful interface than the basic capabilities of the SendDoc method. However, you can use the basic functions of the control without knowledge or use of the DocInput object.

Properties of the DocInput object may be set before calling the SendDoc method or they may be passed as arguments to this method. The DocInput object is also used for conveying information about the progress of the document transfer and for data linking and streaming.

DocInputEvent
Common Control Objects

DocOutput

See Also

Description

Object describing output information for the document being transferred.

Syntax

object.**DocOutput**

The object placeholder is required and evaluates to the name of the relevant control or collection during execution.

Permission

R (Read-only).

Availability

R (Runtime).

Data Type

DocOutput.

Default Value

N/A.

Range

N/A

Comments

The DocOutput object provides a more powerful interface than the basic capabilities of the GetDoc method. However, you can use the basic functions of the control without knowledge or use of the DocInput object.

Properties of the DocOutput object may be set before calling the GetDoc method or they may be passed as arguments to this method. The DocOutput object is also used for conveying information about the progress of the document transfer, and for data linking and streaming.

DocOutputEvent
Common Control Objects

EnableTimer

Description

Enable timer for the specified event. The event is specified by entering:

```
EnableTimer(short event)
```

Syntax

object.**EnableTimer** *(event)* [= *Boolean*]

The object placeholder is required and evaluates to the name of the relevant control or collection during execution.

Permission

W (Write Only).

Note:This is the only control property that is Write only.

Availability

R (Runtime)

Data Type

Boolean.

Default Value

False. (The timer for this event will not be enabled.)

Range

True or False

Comments

Event is an integer value that determines the type of Timeout event that will be enabled. Constants defined for enum types for events are:

Value	Meaning
prcConnectTimeout = 0	Timeout for connect. If connection is not established within the timeout period, the Timeout event will be activated.
prcReceiveTimeout = 1	Timeout for receiving data. If no data arrives within the timeout period, the Timeout event will be activated.
prcUserTimeout= 65	Timeout for user defined event. Use prcUserTimeout + [Integer] range for custom timeout events.

Errors

Description

A collection of errors that can be accessed for details about the last error that occurred. This collection should be used within an Error event if information passed through the Error event is not sufficient. For more details, see icErrors

Syntax

object.**Errors**

The object placeholder is required and evaluates to the name of the relevant control or collection during execution.

Permission

R (Read only).

Availability

R (Runtime).

Data Type

icErrors.

Default Value

N/A.

Range

N/A

LastUpdate

See Also

Description

The default value used by the GetAdministrationFile and ListNewGroups methods.

Syntax

object.**LastUpdate** [= *String*]

The object placeholder is required and evaluates to the name of the relevant control or collection during execution.

Permission

W (Read/Write).

Availability

R (Runtime) and D (Design).

Data Type

DATE.

Default

The time the control is first created.

Range

N/A.

GetAdministrationFile
ListNewGroups

NotificationMode

Description

Determines when notification is issued for incoming data. Notification can also be suspended.

Syntax

object.**NotificationMode** [= *Integer*]

The object placeholder is required and evaluates to the name of the relevant control or collection during execution.

Permission

W (Read/Write).

Availability

R (Runtime) and D (Design).

Data Type

Integer.

Default Value

1.

Range

0-maximum unsigned long. At present , the values are:

Constant	Meaning
0	COMPLETE: notification is provided when there is a complete response.
1	CONTINUOUS: an event is repeatedly activated when new data arrives from the connection.

OverviewSupported

Description

If True, the GetOverviewFormat and GetOverview methods may be used to retrieve header information stored in the server's overview database. This property has no meaning before the connection to the server has been established.

Syntax

object.**OverviewSupported**

The object placeholder is required and evaluates to the name of the relevant control or collection during execution.

Permission

R (Read-only).

Availability

R(Runtime).

Data Type

Boolean.

Default Value

False.

Range

True or False

Comments

Examine this property after a connection is established to determine if the server supports the OVERVIEW.FMT command.

PostingAllowed

Description

If True, the current NNTP server allows posting of news articles. This property has no meaning before the connection to the server has been established.

Syntax

object.**PostingAllowed**

The object placeholder is required and evaluates to the name of the relevant control or collection during execution.

Permission

R (Read-only).

Availability

R(Runtime).

Data Type

Boolean.

Default Value

True.

Range

True or False

Comments

Examine this property after a connection is established to determine if the server supports posting.

ProtocolState

Description

This property specifies the current state of the protocol.

Syntax

object.**ProtocolState**

The object placeholder is required and evaluates to the name of the relevant control or collection during execution.

Permission

R (Read-only).

Availability

R (Runtime).

Data Type

Integer.

Default Value

prcBase.

Range

0-1. Constants defined for the enum types of ProtocolState property are:

Value	Meaning
prcBase = 0	Base state before connection to server is established.
prcTransaction = 1	Connection to server is established. This is the valid state for calling methods on the control.

ProtocolStateString

Description

String representation of ProtocolState.

Syntax

object.**ProtocolStateString**

The object placeholder is required and evaluates to the name of the relevant control or collection during execution.

Permission

R (Read-only).

Availability

R (Runtime).

Data Type

String.

Default Value

"BASE".

Range

N/A

RemoteHost

Description

The remote machine to connect to if the remoteHost parameter in the Connect method is missing. You can either provide a host name or an IP address string in dotted format. For example, 127.0.0.1.

Note:This is the default property of the control.

Syntax

object.**RemoteHost** [= *String*]

The object placeholder is required and evaluates to the name of the relevant control or collection during execution.

Permission

W (Read/Write).

Availability

D (Design) and R (Runtime).

Data Type

String.

Default Value

News.

Range

N/A.

RemotePort

Description

The remote port number to which to connect.

Syntax

object.**RemotePort** [= *Long*]

The object placeholder is required and evaluates to the name of the relevant control or collection during execution.

Permission

W (Read/Write).

Availability

R (Runtime) and D (Design).

Data Type

Long.

Default Value

119.

Range

1-65535.

ReplyCode

Description

The value of the reply code is a protocol specific number that determines the result of the last request, as returned in the ReplyString property.

Syntax

object.**ReplyCode**

The object placeholder is required and evaluates to the name of the relevant control or collection during execution.

Permission

R (Read-only).

Availability

R (Runtime).

Data Type

Long.

Default Value

0

Range

See RFC 977 for a list of valid reply codes.

ReplyString

Description

Lists the last reply string sent by the FTP Server to the client as a result of a request.

Syntax

object.**ReplyString**

The object placeholder is required and evaluates to the name of the relevant control or collection during execution.

Permission

R (Read-only).

Availability

R (Runtime).

Data Type

String.

Default Value

Empty.

Range

N/A.

State

Description

This property specifies the connection state of the control.

Syntax

object.**State**

The object placeholder is required and evaluates to the name of the relevant control or collection during execution.

Permission

R (Read-only).

Availability

R (Runtime).

Data Type

Integer.

Default Value

prcDisconnected.

Range

1-6. Constants defined for enum types of State property are:

Value	Meaning
prcConnecting = 1	Connecting. Connect has been requested, waiting for connect acknowledge.
prcResolvingHost = 2	Resolving Host. Occurs when RemoteHost is in name format rather than dot-delimited IP format.
prcHostResolved = 3	Resolved the host. Occurs only if ResolvingHost state has been entered previously.
prcConnected = 4	Connection established.
prcDisconnecting = 5	Connection closed. Disconnect has been initiated.
prcDisconnected = 6	Initial state when protocol object is instantiated, before Connect has been initiated, after a Connect attempt failed or after Disconnect performed.

StateString

Description

A string representation of State.

Syntax

object.**StateString**

The object placeholder is required and evaluates to the name of the relevant control or collection during execution.

Permission

R (Read-only).

Availability

R (Runtime).

Data Type

String.

Default Value

"Disconnected".

Range

N/A

Timeout

Description

Timeout value for the specified event. The event is specified by entering:

```
Timeout(short event)
```

Syntax

object.**Timeout** (*event*) [= *Long*]

The object placeholder is required and evaluates to the name of the relevant control or collection during execution.

Permission

W (Read/Write).

Availability

R (Runtime) and D (Design).

Data Type

Long.

Default Value

0.

Range

0-maximum unsigned long. Constants defined for enum types for events are:

Value	Meaning
prcConnectTimeout = 0	Timeout for connect. If connection is not established within the timeout period, the Timeout event will be activated.
prcReceiveTimeout = 1	Timeout for receiving data. If no data arrives within the timeout period, the Timeout event will be activated.
prcUserTimeout= 65	Timeout for user defined event. Use prcUserTimeout + [Integer] range for custom timeout events.

URL

Description

URL string identifying the current document being transferred. The valid URL formats are:

news:<newsgroupname>
news:<messageid>

Syntax

object.**URL** [= S*tring*]

The object placeholder is required and evaluates to the name of the relevant control or collection during execution.

Permission

W (Read/Write).

Availability

R (Runtime).

Data Type

String.

Default Value

Empty string.

Range

Valid URL.

Comments

URL may be set before calling the GetDoc or SendDoc method of the control, or it may be passed as an argument to these methods. If it is passed as an argument, the URL property will be set to the argument value.

The URL type (first part up to the colon) may be omitted. In this case, it will default to the correct type for this control. For example, the `nntp:` string may be omitted when using the NNTP control.

Methods

Methods are called to perform a particular operation. The methods performed by the NNTP Client control are:

Cancel

Connect

GetAdministrationFile

GetArticleByArticleNumber

GetArticleByMessageID

GetArticleHeaders

GetArticleNumbers

GetBodyByArticleNumber

GetBodyByMessageID

GetDoc

GetHeaderByArticleNumber

GetHeaderByMessageID

GetOverview

GetOverviewFormat

GetStatByArticleNumber

ListGroupDescriptions

ListGroups

ListNewGroups

Quit

SelectGroup

SendDoc

SetLastArticle

SetNextArticle

Cancel

Description

Cancels a pending request.

Return Value

Void.

Syntax

object.**Cancel**

The object placeholder is required and evaluates to the name of the relevant control or collection during execution.

Parameters

None.

Connect

Description

Initiates a Connect request. The control calls the StateChanged event if a connection is established.

Return Value

Void.

Syntax

object.**Connect** [*RemoteHost*,] [*RemotePort*]

The object placeholder is required and evaluates to the name of the relevant control or collection during execution.

Parameters

RemoteHost

Optional. Remote host to which to connect. If this parameter is missing, the control connects to the host defined in the RemoteHost property
Data Type: String
Param: IN
Default Value: N/A

RemotePort

Optional. Remote port to which to connect. If this parameter is missing, the control connects to the port defined in the RemotePort property.
Data Type: Long
Param: IN
Default Value: N/A

Comments

Optional arguments to this method override the values from corresponding RemoteHost and RemotePort properties. The values of the properties will not change. If no argument is given, the values from the properties will be used to establish the connection.

GetAdministrationFile

Description

Sends the NNTP XMOTD command to the server. This command retrieves the news server administrator's information if the information is newer than the value of lastUpdate.

Return Value

Void.

Syntax

object.**GetAdministrationFile** [*lastUpdate*]

The object placeholder is required and evaluates to the name of the relevant control or collection during execution.

Parameters

lastUpdate

Optional. Indicates time of last update from the server. If the lastUpdate argument is not given, the Control uses the value of the lastUpdate property.
Data Type: DATE
Param: IN
Default Value: None

GetArticleByArticleNumber

Description

Sends the NNTP ARTICLE command with articleNumber to the NNTP server. Upon successful completion, this method causes the DocOutput event to be activated.

Return Value

Void.

Syntax

object.**GetArticleByArticleNumber** [*articleNumber*]

The object placeholder is required and evaluates to the name of the relevant control or collection during execution.

Parameters

articleNumber

Optional. The article number of an article in the current newsgroup. The article number must be chosen from the range of articles numbers provided when the newsgroup was selected. If it is omitted, the current article is assumed.
Data Type: VARIANT. Valid Variant types for articleNumber are String and Integer.
Param: IN
Default Value: None

GetArticleByMessageID

Description

Sends the NNTP ARTICLE command with articleID to the server. When this method reaches a successful completion, the DocInput event is activated.

Return Value

Void.

Syntax

object.**GetArticleByMessageID** *messageID*

The object placeholder is required and evaluates to the name of the relevant control or collection during execution.

Parameters

messageID

Specifies the article's unique messageID for the current NNTP server. The client may obtain the message-id from references contained within another article or from the message-id provided in the response to some other command.
Data Type: String
Param: IN
Default Value: None

GetArticleHeaders

Description

Sends the NNTP XHDR command to the server. Upon successful completion, this method causes the DocOutput event to be activated.

Return Value

Void.

Syntax

object.**GetArticleHeaders** *header*, [*firstArticle*,] [*lastArticle*]

The object placeholder is required and evaluates to the name of the relevant control or collection during execution.

Parameters

header

The name of a header line (e.g.,"subject") in a news group article. This parameter is required. See RFC-1036 for a list of valid header lines.
Data Type: String
Param: IN
Default Value: None

firstArticle, lastArticle

Optional. If firstArticle and lastArticle are given, they indicate a range of article numbers. If lastArticle is 0, the range is all headers following firstArticle.
If no lastArticle argument is given, then the firstArticle indicates a message-id.
If neither firstArticle or lastArticle is given then information for the current article is retrieved.
Data Type: Long
Param: IN
Default Value: None

GetArticleNumbers

Description

Sends the NNTP command LISTGROUP to the server. Upon successful completion, this method causes the DocOutput event to be activated.

Return Value

Void.

Syntax

object.**GetArticleNumbers** [*groupName*]

The object placeholder is required and evaluates to the name of the relevant control or collection during execution.

Parameters

groupName

Optional. If the groupName argument is given, a list of article numbers for that group is retrieved and the group becomes the selected group. If the groupName argument is not given, a list of article numbers for the selected news group is retrieved.
Data Type: String.
Param: IN
Default Value: None

Comments

Use the ArticleNumbersSupported property after connection to determine if the current NNTP server supports this command.

GetBodyByArticleNumber

Description

Sends the NNTP BODY command with articleNumber to the NNTP server. Upon successful completion, this method causes the DocOutput event to be activated.

Return Value

Void.

Syntax

object.**GetBodyByArticleNumber** [*articleNumber*]

The object placeholder is required and evaluates to the name of the relevant control or collection during execution.

Parameters

articleNumber

Optional. The article number of an article in the current newsgroup. The article number must be chosen from the range of articles numbers provided when the newsgroup was selected. If it is omitted, the current article is assumed.
Data Type: VARIANT. Valid Variant types for articleNumber are String and Integer.
Param: IN
Default Value: None

GetBodyByMessageID

Description

Sends the NNTP BODY command with messageID to the server. Upon successful completion, this method causes the DocOutput event to be activated.

Return Value

Void.

Syntax

object.**GetBodyByMessageID** *messageID*

The object placeholder is required and evaluates to the name of the relevant control or collection during execution.

Parameters

messageID

Specifies the article's unique messageID for the current NNTP server. The client will probably obtain the message-id from references contained within another article or from the message-id provided in the response to some other commands.
Data Type: String
Param: IN
Default Value: None

GetDoc

See Also

Description

A DocOutput related method that requests retrieval of a document identified by a URL.

Return Value

Void.

Syntax

object.**GetDoc** [*URL*,] [*Headers*,] [*OutputFile*]

The object placeholder is required and evaluates to the name of the relevant control or collection during execution.

Parameters

URL

Optional. The URL identifying the remote document to be retrieved.
Data Type: String
Param: IN
Default Value: DocInput.URL

Headers

Optional. Headers used for requesting the document. This argument only applies to protocols where request headers can be specified (for example, HTTP).
Data Type: DocHeaders
Param: IN
Default Value: DocInput.Headers

OutputFile

Optional. A local file to which the retrieved document will be written.
Data Type: String
Param: IN
Default Value: DocOutput.Filename

Comments

The GetDoc method in NNTP means retrieving an article from the NNTP server.

The URL and (for some controls) Headers are used as inputs specifying which document is to be retrieved. The OutputFile argument indicates where the retrieved document should be written locally.

The URL type (first part up to the colon) may be omitted and will default to the correct type for this control. For example, when using the NNTP control, the "nntp" string may be omitted.

For basic use of this control, arguments should be passed to GetDoc to describe the document transfer. For more powerful use of this control, the DocInput and DocOutput objects can be used in conjunction with the DocInput and DocOutput events. The arguments of GetDoc correspond to properties in the DocInput and DocOutput objects of this control. DocInput and DocOutput properties can be set before calling GetDoc to avoid passing arguments. The DocInput and DocOutput events can also be used for transferring data using streaming rather than local files.

DocInput
DocOutput Property
DocInput, DocOutput Events
Common Control Objects

GetHeaderByArticleNumber

Description

Sends the NNTP HEAD command with messageNumber to the NNTP server. Upon successful completion, this method causes the DocOutput event to be activated.

Return Value

Void.

Syntax

object.**GetHeaderByArticleNumber** [*articleNumber*]

The object placeholder is required and evaluates to the name of the relevant control or collection during execution.

Parameters

articleNumber

Optional. The article number of an article in the current newsgroup. The article number must be chosen from the range of articles numbers provided when the newsgroup was selected. If it is omitted, the current article is assumed.
Data Type: VARIANT. Valid Variant types for articleNumber are String and Integer.
Param: IN
Default Value: None

GetHeaderByMessageID

Description

Sends the NNTP HEAD command with messageID to the server. Upon successful completion, this method causes the DocOutput event to be activated.

Return Value

Void.

Syntax

object.**GetHeaderByMessageID** *messageID*

The object placeholder is required and evaluates to the name of the relevant control or collection during execution.

Parameters

messageID

Specifies the article's unique messageID for the current NNTP server. The client will probably obtain the message-id from references contained within another article or from the message-id provided in the response to some other commands.
Data Type: String
Param: IN
Default Value: None

GetOverview

Description

Sends the XOVER command to the server. Use the OverSupported property after connection to determine if the current NNTP server supports this command. When this method reaches a successful completion, the DocInput event is activated.

Return Value

Void.

Syntax

object.**GetOverview** [*firstArticle*,] [*lastArticle*]

The object placeholder is required and evaluates to the name of the relevant control or collection during execution.

Parameters

firstArticle, lastArticle

Optional. If firstArticle and lastArticle are given, they indicate a range of article numbers. If lastArticle is 0, the range is all headers following firstArticle.
If neither firstArticle or lastArticle is given then information for the current article is retrieved.
Data Type: String.
Param: IN
Default Value: N/A

Comment

The XOVER command returns information from the overview database for the article(s) specified.

GetOverviewFormat

Description

Sends the LIST OVERVIEW.FMT command to the server. Use the OverViewSupported property after connection to determine if the current NNTP server supports this command. When this method reaches a successful completion, the DocInput event is activated.

Return Value

Void.

Syntax

object.**GetOverviewFormat**

The object placeholder is required and evaluates to the name of the relevant control or collection during execution.

Parameters

None.

Comment

The LIST OVERVIEW.FMT command is used to retrieve a list of headers in the order they appear in the servers overview database.

GetStatByArticleNumber

Description

Sends the NNTP STAT command with articleNumber to the NNTP server. When this method reaches a successful completion, the StatArticle event is activated.

Return Value

Void.

Syntax

object.**GetStatByArticleNumber** [*articleNumber*]

The object placeholder is required and evaluates to the name of the relevant control or collection during execution.

Parameters

articleNumber

Optional. The article number of an article in the current newsgroup. The article number must be chosen from the range of articles numbers provided when the newsgroup was selected. If it is omitted, the current article is assumed.
Data Type: VARIANT. Valid Variant types for articleNumber are String and Integer.
Param: IN
Default Value: None

ListGroups

Description

Sends NNTP LIST command to the server. The server responds with a list of all news groups. Upon successful completion, this method causes the DocOutput event to be activated.

Return Value

Void.

Syntax

object.**ListGroups**

The object placeholder is required and evaluates to the name of the relevant control or collection during execution.

Parameters

None.

ListGroupDescriptions

Description

Sends the NNTP LIST NEWSGROUPS command to the server. Upon successful completion, this method causes the DocOutput event to be activated.

Return Value

Void.

Syntax

object.**ListGroupDescriptions**

The object placeholder is required and evaluates to the name of the relevant control or collection during execution.

Parameters

None.

ListNewGroups

Description

Sends NNTP NEWGROUPS command to server. Upon successful completion, this method causes the DocOutput event to be activated.

Return Value

Void.

Syntax

object.**ListNewGroups** [*lastTime*]

The object placeholder is required and evaluates to the name of the relevant control or collection during execution.

Parameters

lastTime

Optional. Indicates the last time articles were retrieved by the client. If the lastTime parameter is not given, the Control uses the value of the lastUpdate property.
Data Type: DATE.
Param: IN
Default Value: lastUpdate property

Quit

Description

Sends NNTP QUIT command and disconnects from the NNTP server. When this method reaches a successful completion, the StateChanged event is activated.

Return Value

Void.

Syntax

object.**Quit**

The object placeholder is required and evaluates to the name of the relevant control or collection during execution.

Parameters

None.

SelectGroup

Description

Sends NNTP GROUP command to the server. On successful completion, the SelectGroup event is activated.

Return Value

Void.

Syntax

object.**SelectGroup** *groupName*

The object placeholder is required and evaluates to the name of the relevant control or collection during execution.

Parameters

groupName

The name of the group of articles to be selected.
Data Type: String.
Param: IN
Default Value: None

SendDoc

See Also

Description

A DocInput related method that requests sending a document identified by a URL.

Return Value

Void.

Syntax

object.**SendDoc** [*URL*,] [*InputData*,] [*InputFile*,] [*OutputFile*]

The object placeholder is required and evaluates to the name of the relevant control or collection during execution.

Parameters

URL

Optional. The URL identifying the remote document to be sent. If specified, the URL property will be set to this value.
Data Type: String
Param: IN
Default Value: DocInput.URL

InputData

Optional. A data buffer containing the document to be sent.
Data Type: VARIANT
Param: IN
Default Value: DocInput.SetData

InputFile

Optional. A local file containing the document to be sent.
Data Type: String
Param: IN
Default Value: DocInput.Filename

OutputFile

Optional. A local file to which a reply document is written. This argument only applies for protocols that return a reply document (for example, HTTP).
Data Type: String
Param: IN
Default Value: DocOutput.Filename

Comments

The SendDoc method in NNTP means posting an article to the NNTP server.

The URL and (for some controls) Headers are used as inputs describing the document to be sent. The InputData and InputFile arguments (only one can be specified) contain the document to be sent. For controls such as HTTP that return a reply document, the OutputFile argument indicates where the reply document should be written locally.

The URL type (first part up to the colon) may be omitted and will default to the correct type for this control. For example, when using the NNTP control, the "nntp:" string may be omitted .

For basic use of this control, arguments should be passed to SendDoc to describe the document transfer. For more powerful use of this control, the DocInput and DocOutput objects can be used in conjunction with the DocInput and DocOutput events. The arguments of SendDoc correspond to properties in the DocInput and DocOutput objects of this control. DocInput and DocOutput properties can be set before calling SendDoc to avoid passing arguments. The DocInput and DocOutput events can also be used for transferring data using streaming rather than local files.

DocInput
DocOutput Property
DocInput, DocOutput Events
Common Control Objects

SetLastArticle

Description

Sends NNTP LAST command to the server. On successful completion, the LastArticle event is activated.

Return Value

Void.

Syntax

object.**SetLastArticle**

The object placeholder is required and evaluates to the name of the relevant control or collection during execution.

Parameters

None.

SetNextArticle

Description

Sends NNTP NEXT command to the server. On successful completion, the NextArticle event is activated.

Return Value

Void.

Syntax

object.**SetNextArticle**

The object placeholder is required and evaluates to the name of the relevant control or collection during execution.

Parameters

None.

Events

Events are used for NNTP client notification. They indicate that an action has been requested and processed. Any errors which occur during command processing result in the Error event being called with appropriate error codes. Error codes, state changes, and protocol return values are usually checked during event processing.

AuthenticateRequest

AuthenticateResponse

Banner

Busy

Cancel

DocInput

DocOutput

Error

LastArticle

NextArticle

ProtocolStateChanged

SelectGroup

StatArticle

StateChanged

TimeOut

AuthenticateRequest

Description

This event is activated when the connected NNTP server requests authentication.

Syntax

*object*_**AuthenticateRequest** (*UserID* **As String,** *Password* **As String**)

The object placeholder is required and evaluates to the name of the relevant control or collection during execution.

Parameters

UserId

Optional. User identification string to use for authentication.
Data Type: String
Param: IN
Default Value: N/A

Password

Optional. Password to use for authentication.
Data Type: String
Param: IN
Default Value: N/A

Comments

If the UserID and Password arguments are specified, their values are used instead of the UserID and Password properties.

AuthenticateResponse

Description

This event is activated when an authentication response is received from the server.

Syntax

*object*_**AuthenticateResponse** (*Authenticated* **As Boolean**)

The object placeholder is required and evaluates to the name of the relevant control or collection during execution.

Parameters

Authenticated

Indicates if the authentication is successful. If this argument is True, the authentication has succeeded.
Data Type: Boolean

Banner

Description

This event is activated when the server responds with its sign-on banner after a connection is established.

Syntax

*object*_**Banne**r (*Banner* **As String**)

The object placeholder is required and evaluates to the name of the relevant control or collection during execution.

Parameters

Banner

The sign-on message returned by the news server.
Data Type: String
Param: IN
Default Value: N/A

Busy

Description

This event is activated when a command is in progress or when a command has completed.

Syntax

*object*_**Busy** (*Busy* **As Boolean**)

The object placeholder is required and evaluates to the name of the relevant control or collection during execution.

Parameters

Busy

Indicates whether or not a command is in progress.
Data Type: Boolean. If the argument is True, a command is in progress.

Cancel

Description

This event is activated after a cancellation request has been completed and satisfied. After this event the object's state changes to idle.

Syntax

*object*_**Cancel**

The object placeholder is required and evaluates to the name of the relevant control or collection during execution.

Parameters

None.

DocInput

See Also

Description

A DocInput related event that indicates the input data has been transferred.

Syntax

*object*_**DocInput** (*DocInput* **As DocInput**)

The object placeholder is required and evaluates to the name of the relevant control or collection during execution.

Parameters

DocInput

Object describing document input data for the current transfer.
Data Type: DocInput
Param: IN
Default Value: N/A

Comments

The DocInput event can be used in its basic form for notification of transfer progress, (for example, for updating a progress bar). The DocInput.BytesTotal, DocInput.BytesTransferred and DocInput.State properties can be examined to determine the current status of the transfer. This event can be ignored if no progress information is needed.

To gain more power from this control, you can also use the DocInput event for data streaming.

Common Control Objects

DocOutput

See Also

Description

A DocOutput related event indicating that output data has been transferred.

Syntax

*object*_**DocOutput** (*DocOutput* **As DocOutput**)

The object placeholder is required and evaluates to the name of the relevant control or collection during execution.

Parameters

DocOutput

Object describing document output data for the current transfer.
Data Type: DocOutput
Param: IN
Default Value: N/A

Comments

The DocOutput event can be used in its basic form to notify the user of transfer progress, (for example, for updating a progress bar). The DocOutput.BytesTotal, DocOutput.BytesTransferred and DocOutput.State properties can be examined to determine the current status of the transfer. This event can be ignored if no progress information is needed.

To gain more power from this control, you can also use the DocOutput event for data streaming.

Common Control Objects

Error

Description

This event is activated when an error occurs in background processing (for example, failed to connect or failed to send or receive in the background).

Syntax

*object*_**Error** (*ErrCode* **As Integer**, *Description* **As String**, *Scode* **As Long**, *Source* **As String**, *HelpFile* **As String**, *HelpContext* **As Long**, *CancelDisplay* **As Boolean**)

The object placeholder is required and evaluates to the name of the relevant control or collection during execution.

Parameters

ErrCode

The short error code. For a list of NNTP error codes see NNTP Error Codes

Description

String containing error information.

sCode

The long Scode.

Source

Error source.

HelpFile

Help file name.

HelpContext

Help file context.

CancelDisplay

Indicates whether to cancel the display. The default is TRUE (no display of the default error message box). If you do want to use the default message box, set CancelDisplay to FALSE.

LastArticle

Description

This event is activated after a successful completion of the LastArticle method.

Syntax

*object*_**LastArticle** (*ArticleNumber* **As Long,** *MessageID* **As String**)

The object placeholder is required and evaluates to the name of the relevant control or collection during execution.

Parameters

ArticleNumber

The article number of the selected article.
Data Type: Long
Param: IN
Default Value: N/A

MessageID

The message id of the selected article.
Data Type: String
Param: IN
Default Value: N/A

NextArticle

Description

This event is activated after a successful completion of the NextArticle method.

Syntax

*object*_**NextArticle** (*ArticleNumber* **As Long,** *MessageID* **As String**)

The object placeholder is required and evaluates to the name of the relevant control or collection during execution.

Parameters

ArticleNumber

The article number of the selected article.
Data Type: Long
Param: IN
Default Value: N/A

MessageID

The message id of the selected article.
Data Type: String
Param: IN
Default Value: N/A

ProtocolStateChanged

Description

This event is activated whenever the protocol state changes.

Syntax

*object*_**ProtocolStateChanged** (*State* **As Integer**)

The object placeholder is required and evaluates to the name of the relevant control or collection during execution.

Parameters

Refer to the ProtocolState property and ProtocolStateString for possible values of the state parameter.

SelectGroup

Description

This event is activated after a successful completion of the SelectGroup method.

Syntax

*object_*(*groupName* **As String,** *firstArticleNumber* **As Long,** *lastArticleNumber* **As Long**)

The object placeholder is required and evaluates to the name of the relevant control or collection during execution.

Parameters

groupName

The name of the group of articles to be selected.
Data Type: String

firstArticleNumber

The number of the first article in the selected news group.
Data Type: Long

lastArticleNumber

The number of the last article in the selected news group.
Data Type: Long

StatArticle

Description

This event is activated after a successful completion of the GetStatByArticleNumber method.

Syntax

*object*_**StatArticle** (*ArticleNumber* **As Long,** *MessageID* **As String**)

The object placeholder is required and evaluates to the name of the relevant control or collection during execution.

Parameters

ArticleNumber

The article number of the selected article.
Data Type: Long
Param: IN
Default Value: N/A

MessageID

The message id of the selected article.
Data Type: String
Param: IN
Default Value: N/A

StateChanged

Description

This event is activated whenever the state of the transport state changes.

Syntax

*object*_**StateChanged** (*State* **As Integer**)

The object placeholder is required and evaluates to the name of the relevant control or collection during execution.

Parameters

Refer to the State property and StateString for possible values of the state parameter.

TimeOut

See Also

Description

This event is activated when the timer for the specified event expires. See Timeout property for pre-defined events.

Syntax

*object*_**TimeOut** (ByVal *Event* **As Integer,** *Continue* **As Boolean**)

The object placeholder is required and evaluates to the name of the relevant control or collection during execution.

Parameters

Event

Defines the event to which the time interval applies.
Data Type: Integer

Continue

Determines if the timer is active or not. Set *Continue* to TRUE to keep the timer active.
Data Type: Boolean
Default Value: False

Comments

Event is an integer value that determines the type of Timeout event that will be enabled. Constants defined for enum types for events are:

Value	Meaning
prcConnectTimeout = 0	Timeout for connect. If connection is not established within the timeout period, the Timeout event will be activated.
prcReceiveTimeout = 1	Timeout for receiving data. If no data arrives within the timeout period, the Timeout event will be activated.
prcUserTimeout= 65	Timeout for user defined event. Use prcUserTimeout + [Integer] range for custom timeout events.

Timeout Property

ActiveX

ActiveX is a trademark of Microsoft Corporation.

NNTP Error Codes

The following error codes apply only to the NNTP ActiveX Control.

Error Code	Error Message
2203	NNTP server does not allow posting.