Administrative Object API

GroupWise® Software Developer Kit

November 2012

Novell®

Legal Notices

Novell, Inc. makes no representations or warranties with respect to the contents or use of this documentation, and specifically disclaims any express or implied warranties of merchantability or fitness for any particular purpose. Further, Novell, Inc. reserves the right to revise this publication and to make changes to its content, at any time, without obligation to notify any person or entity of such revisions or changes.

Further, Novell, Inc. makes no representations or warranties with respect to any software, and specifically disclaims any express or implied warranties of merchantability or fitness for any particular purpose. Further, Novell, Inc. reserves the right to make changes to any and all parts of Novell software, at any time, without any obligation to notify any person or entity of such changes.

Any products or technical information provided under this Agreement may be subject to U.S. export controls and the trade laws of other countries. You agree to comply with all export control regulations and to obtain any required licenses or classification to export, re-export, or import deliverables. You agree not to export or re-export to entities on the current U.S. export exclusion lists or to any embargoed or terrorist countries as specified in the U.S. export laws. You agree to not use deliverables for prohibited nuclear, missile, or chemical biological weaponry end uses. Please refer to http://www.novell.com/info/exports/ (http://www.novell.com/info/exports/) for more information on exporting Novell software. Novell assumes no responsibility for your failure to obtain any necessary export approvals.

Novell, Inc.
1800 South Novell Place
Provo, UT 84606
U.S.A.
www.novell.com

Online Documentation: To access the online documentation for this and other Novell developer products, and to get updates, see the Novell Developer Web site (http://www.novell.com/developer). To access online documentation for Novell products, see the Novell Documentation Web site (http://www.novell.com/documentation).

Novell Trademarks

For Novell trademarks, see the Novell Trademark and Service Mark list (http://www.novell.com/company/legal/trademarks/tmlist.html).

Third-Party Materials

All third-party trademarks are the property of their respective owners.

Contents

About This Guide

GroupWise Administrative Object API lets you see, use, and manipulate GroupWise administration information from outside GroupWise. You can use the Administrative Object API through any programming environment supporting OLE Automation, such as Visual Basic, Delphi, and C++.

IMPORTANT: The features in GroupWise Administrative Object API work with GroupWise 8 and GroupWise 2012.

This guide contains the following sections:

- "Overview" on page 9
- "Reference" on page 11
- "Revision History" on page 91

Audience

This guide is intended for GroupWise developers.

Feedback

We want to hear your comments and suggestions about this manual and the other documentation included with this product. Please use the User Comments feature at the bottom of each page of the online documentation.

Additional Documentation

For additional GroupWise SDK documentation, see the Novell Developer Web site (http://www.novell.com/developer).

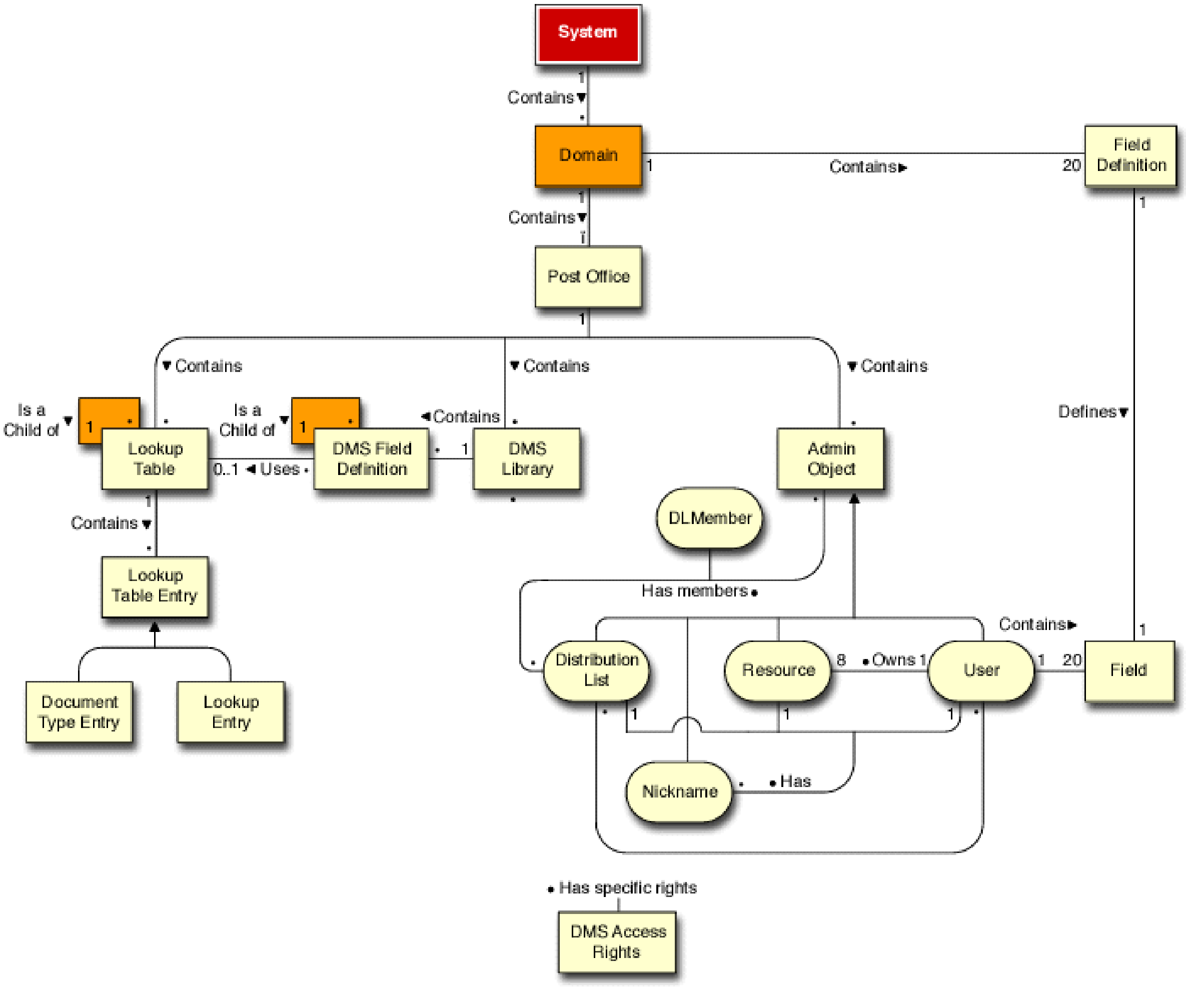
1 Overview

This section covers the following topics:

- Section 1.1, “Visual Basic,” on page 10
- Section 1.2, “Delphi,” on page 10
- Section 1.3, “C++,” on page 10

The first step to using the Administrative Object API in a third-party application is to get a handle to the GroupWise System object.

Figure 1-1 *GroupWise Admin API Object Model*



1.1 Visual Basic

To get the handle to the GroupWise System object, from Visual Basic use:

```
Set objAdminSystem = CreateObject( "NovellGroupWareAdmin" )
```

The variable objAdminSystem must be of type variant or object.

Next, log in to GroupWise by invoking the System Object Connect method. From Visual Basic, use:

```
objAdminSystem.Connect (txtPath)
```

The String txtPath is the full path to the domain database (do not enter the domain database name itself—only the path is expected).

Your application is now connected to a GroupWise domain and the System object properties are valid. Store contents are accessible through the System object.

1.2 Delphi

From Delphi, get a handle to the System object for the Administrative Object API and connect to a GroupWise domain using statements similar to Visual Basic. However, OleAuto must be declared in the Uses section and objAdminSystem must be declared variant in the Var section.

To get the handle to the GroupWise System object, from Delphi use:

```
objAdminSystem := CreateOleObject( ’NovellGroupWareAdmin’ );
```

To connect to a GroupWise domain from Delphi, use:

```
objAdminSystem.Connect( txtPath );
```

Important: Applications that use the Administrative Object API require the Novell 32-bit NetWare client. Otherwise, the application will appear to register correctly, but will not function properly.

1.3 C++

Get a pointer to the System Interface:

```
hResult = CoCreateInstance(CLSID_System, NULL, CLSCTX_INPROC_SERVER |
CLSCTX_INPROC_HANDLER | CLSCTX_LOCAL_SERVER, IID_IADSystem, (void **)
&pIADSystem);
```

Connect to the Domain:

```
hResult = pIADSystem->Connect(vDomainPath);
```

vDomainPath is the full path to the domain database (do not enter the domain database name itself, only the path)

2 Reference

This section contains information for the following components of GroupWise Administrative Object API:

2.1 Objects

AdminObject

A base class for all objects that appear in the system address book.

Properties

The following table lists the properties for this class, along with access and descriptions.

Property	Access	Description
Description	R/W	String
DistinguishedName	R/O	String
Domain	R/O	"Domain" on page 33
LastModifiedBy	R/O	String
MemberOf	R/O	"DLMembers" on page 19
Name	R/W	String
ObjectID	R/O	String
ObjType	R/O	Enum(eadObjDistributionList, eadObjNickname, eadObjResource, eadObjUser,)
PostOffice	R/O	"PostOffice" on page 54
System	R/O	"System" on page 65
Visibility	R/W	Enum("VisibilityTypeConstants" on page 79)

Methods

`Commit()`

Writes out all changed information in the object.

`Delete( StoreTypeConstants StoreType )`

Deletes this object according to the StoreType. The eadGW enumeration is valid only for the User object.

`Refresh()`

Discards information associated with the object and re-reads the object.

Remarks

None.

DistributionList

Lets you group users, resources, and other distribution lists. This object is derived from AdminObject and inherits all its properties and methods.

Properties

The following table lists the properties for this class, along with access and descriptions.

Property	Access	Description
DMSAccessRights	R/O	"DMSAccessRightsCollection" on page 22
Membership	R/O	"DLMembers" on page 19
Nicknames	R/O	"Nicknames" on page 52

Methods

`MoveWithinTree( String DN )`

Moves this object to a different Organization or Organizational Unit specified by the given DN (distinguished name).

Remarks

GroupWise does not support member roles defined as Carbon Copy (CC) or Blind Copy (BC) in eDirectory Group objects or Role objects. These type of memberships require special handling.

DistributionLists

This large collection contains DistributionList objects.

Properties

The following table lists the properties for this class, along with access and descriptions.

Property	Access	Description
_NewEnum	R/O	Enumeration object (Implements IEnumVARIANT. For Windows only.)
ObjType	R/O	Enum(eadObjDistributionLists)
System	R/O	“System” on page 65

Methods

`DistributionList Add( String ObjName, String DN, [PostOffice PO], [Domain Dom] )`

Creates a DistributionList object with the given ObjName and DN (distinguished name). See the table under Remarks.

`DistributionList Add( String ObjName, String DN, [String PO], [String Dom] )`

Creates a DistributionList object with the given ObjName and DN (distinguished name). See the table under Remarks.

`DistributionListsIterator CreateIterator()`

Creates a DistributionListsIterator object.

`DistributionList Item( String Name, [String PO], [String Dom] )`

Returns the DistributionList object with the given Name. See the table under Remarks.

`DistributionList Item( String Name, [PostOffice PO], [Domain Dom] )`

Returns the DistributionList object with the given Name. See the table under Remarks.

`DistributionList ItemByDN( String DN, [String Tree] )`

Returns the DistributionList object with the given DN (distinguished name). If no Tree is supplied, the current tree is assumed.

`DistributionList ItemByObjectID( String ObjectID )`

Returns the DistributionList object with the given ObjectID.

Remarks

This table applies to the above methods as noted. It shows the parameters you can use with the methods depending on where you obtain the DistributionLists object.

Source of DistributionLists Object	Required Parameters	Optional Parameters
PostOffice		PO, Dom
Domain	PO	Dom
System	PO, Dom	

DistributionListsIterator

Iterates over all DistributionList objects in a DistributionLists collection.

Properties

The following table lists the properties for this class, along with access and descriptions.

Property	Access	Description
ObjType	R/O	Enum(eadObjDistributionListsIterator)
System	R/O	“System” on page 65

Methods

`DistributionListsIterator Clone()`

Creates a new, independent DistributionListsIterator object. The clone starts at the original’s current position in the collection. Clones are useful for saving the current position of an iterator.

`DistributionList Next()`

Returns the next DistributionList object, or nothing if the last DistributionList object has already been returned.

`Reset()`

Resets the DistributionListsIterator to its intitial state.

`Skip( Long NumItems )`

Skips past the next NumItems DistributionList objects, or as many as possible if there are no remaining NumItems to skip. Returns an error if NumItems < 0.

Remarks

None.

DLMember

Associates the owning DistributionList with an AdminObject (DistributionList, Resource, or User).

Properties

The following table lists the properties for this class, along with access and descriptions.

Property	Access	Description
DistributionList	R/O	"DistributionList" on page 14
LastModifiedBy	R/O	String
Member	R/O	"AdminObject" on page 13
MemberType	R/O	Enum(eadObjDistributionList, eadObjResource, eadObjUser)
ObjType	R/O	Enum(eadObjDLMember)
Participation	R/W	Enum("ParticipationTypeConstants" on page 78)
System	R/O	"System" on page 65

Methods

`Commit()`

Writes out all changed information in the object.

`Delete()`

Deletes this object.

Remarks

GroupWise does not support member roles defined as Carbon Copy (CC) or Blind Copy (BC) in eDirectory Group objects or Role objects. These type of memberships require special handling.

DLMembers

This large collection contains DLMember objects.

Properties

The following table lists the properties for this class, along with access and descriptions.

Property	Access	Description
NewEnum	R/O	Enumeration object (implements IEnumVARIANT. For Windows only).
ObjType	R/O	Enum(eadObjDLMembers)
System	R/O	“System” on page 65

Methods

`DLMember Add( AdminObject Member, [ParticipationTypeConstants Participation], [DistributionList DistList] )`

Creates a DLMember object from the given Member object. Member is a DistributionList, Nickname, Resource, or User object.

- If Participation is omitted, To is assumed.
- If DLMembers is obtained through the MemberOf method of Nickname, Resource, or User, DistList is required.

`DLMembersIterator CreateIterator()`

Creates a DLMembersIterator object.

`DLMember Item( AdminObject Member, [ DistributionList DistList ] )`

Returns the DLMember object represented by Member. Member is a DistributionList, Nickname, Resource, or User object.

- If DLMembers is obtained through the MemberOf method of Nickname, Resource, or User, DistList is required.

Remarks

None.

DLMembersIterator

Iterates over all DLMember objects belonging in a DLMembers collection.

Properties

The following table lists the properties for this class, along with access and descriptions.

Property	Access	Description
ObjType	R/O	Enum(eadObjDLMembersIterator)
System	R/O	“System” on page 65

Methods

`DLMembersIterator Clone()`

Creates a new, independent DLMembersIterator object. The clone starts at the original’s current position in the collection. Clones are useful for saving the current position of an iterator.

`DLMember Next()`

Returns the next DLMember object, or nothing if the last DLMember object has already been returned.

`Reset()`

Resets the DLMembersIterator to its intitial state.

`Skip( Long NumItems )`

Skips past the next NumItems DLMember objects, or as many as possible if there are no remaining NumItems to skip. Returns an error if NumItems < 0.

Remarks

None.

DMSAccessRights

A document management library access rights object. This object allows you to change the rights of a User or DistributionList to a DMSLibrary.

Properties

The following table lists the properties for this class, along with access and descriptions.

Property	Access	Description
BitMask	R/W	Long. A bit mask containing one bit for each Boolean property. See the Add method in DMSAccessRightsCollection for the bit definitions. This property is synchronized with the Boolean properties (such as, CanAdd and CanDelete) and is an alternate way to access the same information.
CanAdd	R/W	Boolean
CanDelete	R/W	Boolean
CanDesignateOfficialVersion	R/W	Boolean
CanEdit	R/W	Boolean
CanResetInUseFlag	R/W	Boolean
CanView	R/W	Boolean
DMSLibrary	R/O	"DMSLibrary" on page 30
IsLibrarian	R/W	Boolean
Member	R/O	See "AdminObject" on page 13.
MemberType	R/O	Enum(MemberTypeConstants) See "Remarks" on page 21.
ObjType	R/O	Enum(eadObjDMSAccessRights)
System	R/O	"System" on page 65

Methods

`Commit()`

Writes out all changed information in the object.

`Delete()`

Deletes this object.

`Refresh()`

Discards information associated with the object and re-reads the object.

Remarks

If MemberType is set to eadNoMember, the object is the DefaultRights retrieved from DMSLibrary.

DMSAccessRightsCollection

This large collection contains DMSAccessRights objects.

Properties

The following table lists the properties for this class, along with access and descriptions.

Property	Access	Description
_NewEnum	R/O	Enumeration object. Implements IEnumVARIANT. (For Windows only.)
ObjType	R/O	Enum(eadObjDMSAccessRightsCollection)
System	R/O	"System" on page 65

Methods

DMSAccessRights Add(Long RightsBitMask, AdminObject Member)

Creates a DMSAccessRights object with the given information and object properties. The value for RightsBitMask is derived from the enumerated values listed in the following table under Remarks, combined with the bitwise inclusive OR operator. This method is to be used when the collection is obtained from a DMSLibrary. The only valid objects for Member are a DistributionList or User.

DMSAccessRights Add(Long RightsBitMask, String Member, Enum(eadObjDistributionList, eadObjUser) MemberType, [String PO], [String Dom])

Creates a DMSAccessRights object with the given information and object properties. The value for RightBitMask is derived from the enumerated values listed in the following table under Remarks, combined with the bitwise inclusive OR operator. This method is to be used when the collection is obtained from a DMSLibrary. The only valid objects for Member are a DistributionList or User. If PO or Dom are empty, the Library's PostOffice and Domain will be used as the default values.

DMSAccessRights Add(Long RightsBitMask, DMSLibrary Library)

Creates a DMSAccessRights object with the given information and object properties. The value for RightsBitMask is derived from the enumerated values listed in the following table under Remarks, combined with the bitwise inclusive OR operator. This method is to be used when the collection is obtained from a DistributionList or User.

`DMSAccessRights Add( Long RightsBitMask, String Library, [String PO], [String Dom] )`

Creates a DMSAccessRights objects with the given information and object properties. The value for RightsBitMask is derived from the enumerated values listed in the following table under Remarks, combined with the bitwise inclusive OR operator. This method is to be used when the collection is obtained from a DistributionList or a User. If PO or Dom are empty, the DistributionList's PostOffice and Domain will be used as the default values.

`DMSAccessRightsCollectionIterator CreateIterator()`

Creates a DMSAccessRightsCollectionIterator object.

`DMSAccessRights Item( Variant Object )`

Returns the DMSAccessrights object for the given Object. If this collection is obtained from a DMSLibrary, Object must be a DistributionList or User. If this collection is obtained from a DistributionList or User, Object must be a DMSLibrary.

`DMSAccessRights Item( String ObjectName, Enum( eadObjDistributionList, eadObjDMSLibrary, eadObjUser ) ObjectType )`

Returns the DMSAccessrights object with the given ObjectName. If this collection is obtained from a DMSLibrary, ObjectName must be the name of a DistributionList or User. If this collection is obtained from a DistributionList or User, ObjectName must be the name of a DMSLibrary, and ObjType must be eadObjDMSLibrary.

Remarks

The following enumerated values apply to the DMSAccessRightsCollection methods indicated above.

Value	Description
eadAddRight	The user can add documents.
eadDeleteRight	The user can delete documents.
eadDesignateOfficialRight	The user can designate a document as the official version.
eadEditRight	The user can edit documents.
eadLibrarianRight	The user has Librarian rights.
eadResetInUseFlagRight	The user can reset a document’s In Use flag.
eadViewRight	The user can view documents.

DMSAccessRightsCollectionIterator

This object iterates over all DMSAccessRights objects in a DMSAccessRightsCollection, belonging to a DistributionList, DMSLibrary, or User.

Properties

The following table lists the properties for this class, along with access and descriptions.

Property	Access	Description
ObjType	R/O	Enum(eadObjDMSAccessRightsCollectionIterator)
System	R/O	"System" on page 65

Methods

`DMSAccessRightsCollectionIterator Clone()`

Creates a new, independent DMSAccessRightsCollectionIterator object. The clone starts at the original's current position in the collection. Clones are useful for saving the current position of an iterator.

`DMSAccessRights Next()`

Returns the next DMSAccessRights object, or nothing, if the last DMSAccessRights object has already been returned.

`Reset()`

Resets the DMSAccessRightsCollectionIterator to its initial state.

`Skip( Long NumItems )`

Skips past the next NumItems DMSAccessRights objects, or as many as possible if there are no remaining NumItems to skip. Returns an error if NumItems < 0.

Remarks

None.

DMSFieldDefinition

Allows modifications to field definitions in a DMSLibrary.

Properties

The following table lists the properties for this class, along with access and descriptions.

Property	Access	Description
DataType	R/W	Enum(DataTypeConstants)
Description	R/W	String
DMSLibrary	R/O	DMSLibrary
HasLookupTable	R/O	Boolean
Hidden	R/W	Boolean
LookupTable	R/W	LookupTable
MaximumLength	R/W	Long
MaximumValue	R/W	Long
MinimumValue	R/W	Long
Name	R/W	String
ObjType	R/O	Enum(eadObjDMSFieldDefinition)
ParentFieldDefinition	R/W	DMSFieldDefinition
ReadOnly	R/W	Boolean
Required	R/W	Boolean
StringCase	R/W	Enum(StringCaseTypeConstants)
System	R/O	"System" on page 65
Type	R/O	Enum(FieldDefinitionTypeConstants)

Methods

`Commit()`

Writes out all changed information in the object.

`Delete()`

Deletes this object.

`Refresh()`

Discards information associated with the object and re-reads the object.

Remarks

The LookupTable property is valid only if the HasLookupTable property is TRUE.

The following system-defined DMSFieldDefinition objects are ReadOnly: Author, Creator, Current Version #, Date Created, Document Number, Document Type, Official Version #, Subject.

DMSFieldDefinitions

This collection contains DMSFieldDefinition objects.

Properties

The following table lists the properties for this class, along with access and descriptions.

Property	Access	Description
Count	R/O	Long
_NewEnum	R/O	Enumeration object. Implements IEnumVARIANT. (For Windows only.)
ObjType	R/O	Enum(eadObjDMSFieldDefinitions)
System	R/O	“System” on page 65

Methods

`DMSFieldDefinition Add( String FieldName, [DataTypeConstants DataType] )`

Adds a DMSFieldDefinition with the given FieldName.

`DMSFieldDefinition Item( Long Index )`

Returns the DMSFieldDefinition object at the specified Index in the collection.

`DMSFieldDefinition Item( String Name )`

Returns the DMSFieldDefinition with the given Name.

Remarks

None.

DMSLibraries

This large collection contains DMSLibrary objects.

Properties

The following table lists the properties for this class, along with access and descriptions.

Property	Access	Description
_NewEnum	R/O	Enumeration object. Implements IEnumVARIANT. (For Windows only.)
ObjType	R/O	Enum(eadObjDMSLibraries)
System	R/O	"System" on page 65

Methods

`DMSLibrary Add( String LibraryName, StringDN, [PostOffice PO], [Domain Dom] )`

Creates a DMSLibrary with the given LibraryName and DN (distinguished name). If you are adding a DMSLibrary to a PostOffice in the current Domain, only PO is required. If you are adding a DMSLibrary to a PostOffice in another Domain, PO and Dom are required.

`DMSLibrary Add( String LibraryName, String DN, [String PO], [String Dom] )`

Creates a DMSLibrary with the given LibraryName and DN (distinguished name). If you are adding a DMSLibrary to a PostOffice in the current Domain, only PO is required. If you are adding a DMSLibrary to a PostOffice in another Domain, PO and Dom are required.

`DMSLibrariesIterator CreateIterator()`

Creates a DMSLibrariesIterator object.

`DMSLibrary Item( String Name, [String PO], [String Domain] )`

Returns the DMSLibrary with the given Name.

`DMSLibrary Item( String Name, [PostOffice PO], [Domain Dom] )`

Returns the DMSLibrary with the given Name.

`DMSLibrary ItemByDN( String DN, [String Tree] )`

Returns the DMSLibrary with the specified DN (distinguished name). If no Tree is supplied, the current tree is assumed.

`DMSLibrary ItemByObjectID( String ID )`

Returns the DMSLibrary with the specified object ID.

Remarks

None.

DMSLibrariesIterator

This object iterates over all DMSLibrary objects in a DMSLibraries collection that belong to a PostOffice.

Properties

The following table lists the properties for this class, along with access and descriptions.

Property	Access	Description
ObjType	R/O	Enum(eadObjDMSLibrariesIterator)
System	R/O	“System” on page 65

Methods

`DMSLibrariesIterator Clone()`

Creates a new, independent DMSLibrariesIterator object. The clone starts at the original’s current position in the collection. Clones are useful for saving the current position of an iterator.

`DMSLibrary Next()`

Returns the next DMSLibrary object, or nothing if the last DMSLibrary object has already been returned.

`Reset()`

Resets the DMSLibrariesIterator to its initial state.

`Skip( Long NumItems )`

Skips past the next NumItems DMSLibrary objects, or as many as possible if there are no remaining NumItems to skip. Returns an error if NumItems < 0.

Remarks

None.

DMSLibrary

A document management library object.

Properties

The following table lists the properties for this class, along with access and descriptions.

Property	Access	Description
CurrentArchiveBytes	R/O	Long
CurrentArchiveLocation	R/O	String
DefaultRights	R/O	“DMSAccessRights” on page 21
Description	R/W	String
DisplayName	R/W	String
DistinguishedName	R/O	String
DMSAccessRights	R/O	“DMSAccessRightsCollection” on page 22
DocumentTypeFieldDefinition	R/O	“DMSFieldDefinition” on page 25
Domain	R/O	“Domain” on page 33
FieldDefinitions	R/O	DMSFieldDefinition Collection
LastModifiedBy	R/O	String
LocalID	R/O	Long
MaxArchiveBytes	R/O	Long
Name	R/W	String
NextDocumentNumber	R/W	Long
ObjectID	R/O	String
ObjType	R/O	Enum(eadObjDMSLibrary)
PartitionCount	R/O	Long
PostOffice	R/O	“PostOffice” on page 54
StartingVersionNumber	R/W	Long
System	R/O	“System” on page 65

Methods

`Commit()`

Writes out all changed information in the object.

`Delete( StoreTypeConstants StoreType )`

Deletes this object according to the StoreType.

`MoveWithinTree( String DN )`

Moves this object to a different Organization or Organizational Unit as specified by the DN (distinguished name).

`Refresh()`

Discards information associated with the object and re-reads the object.

Remarks

None.

DocumentTypeEntry

A document management document type entry object. This object is derived from the LookupTableEntry object and inherits all of its properties and methods.

Properties

The following table lists the properties for this class, along with access and descriptions.

Property	Access	Description
DocumentLife	R/W	Long
ExpirationAction	R/W	Enum(ExpirationTypeConstants)
MaxVersions	R/W	Long

Methods

None.

Remarks

None.

Domain

Allows modification to an existing GroupWise domain.

Properties

The following table lists the properties for this class, along with access and descriptions.

Property	Access	Description
AddressFormat	R/W	Enum(AddressFormatConstants)
AdminDefLabels	R/O	"FieldDefinitions" on page 42
AllowedAddressFormat	R/W	Enum(AllowedAddressFormatsConstants)
Description	R/W	String
DistinguishedName	R/O	String
DistributionLists	R/O	"DistributionLists" on page 15
DomainType	R/O	Enum(DomainTypeConstants)
FullLicenseMailboxCount	R/O	Long. Returns the total count of mailboxes in the domain where a full-license client is used. The accuracy of this count depends on the Audit process which is performed by the Post Office Agent. **---GroupWise 8.0 SP1 and later.**
InactiveMailboxCount	R/O	Long. Returns the total count of inactive mailboxes in the domain. The accuracy of this count depends on the Audit process which is performed by the Post Office Agent. **--GroupWise 8.0 SP1 and later.**
InternetDomainName	R/O	String **--GroupWise 8.0 and later.**
InternetDomainNameExclusive	R/O	Boolean **--GroupWise 8.0 and later.**
LastModifiedBy	R/O	String
LimitedLicenseMailBoxCount	R/O	Long. Returns the total count of mailboxes in the domain where a limited-license client is used. The accuracy of this count depends on the Audit process which is performed by the Post Office Agent. **--GroupWise 8.0 SP1 and later.**
Linux65XClientCount	R/O	Long. Returns the count of mailboxes in the domain where the client type being used is the GroupWise 6.5 Linux Client. The accuracy of this count depends on the Audit process which is performed by the Post Office Agent. **--GroupWise 8.0 SP1 and later.**

Property	Access	Description
Linux70XClientCount	R/O	Long. Returns the count of mailboxes in the domain where the client type being used is the GroupWise 7.0 Linux Client. The accuracy of this count depends on the Audit process which is performed by the Post Office Agent. **--GroupWise 8.0 SP1 and later.**
Linux80XClientCount	R/O	Long. Returns the count of mailboxes in the domain where the client type being used is the GroupWise 8.0 Linux Client. The accuracy of this count depends on the Audit process which is performed by the Post Office Agent. **--GroupWise 8.0 SP1 and later.**
Name	R/O	String
NDSTree	R/O	String
Nicknames	R/O	"Nicknames" on page 52
ObjectID	R/O	String
ObjType	R/O	Enum(eadObjDomain)
Mac5XClientCount	R/O	Long. Returns the count of mailboxes in the domain where the client type being used is the GroupWise 5.X Mac Client. The accuracy of this count depends on the Audit process which is performed by the Post Office Agent. **--GroupWise 8.0 SP1 and later.**
Mac6XClientCount	R/O	Long. Returns the count of mailboxes in the domain where the client type being used is the GroupWise 6.X Mac Client. The accuracy of this count depends on the Audit process which is performed by the Post Office Agent. **--GroupWise 8.0 SP1 and later.**
Mac70XClientCount	R/O	Long. Returns the count of mailboxes in the domain where the client type being used is the GroupWise 7.0 Mac Client. The accuracy of this count depends on the Audit process which is performed by the Post Office Agent. **--GroupWise 8.0 SP1 and later.**
Mac80XClientCount	R/O	Long. Returns the count of mailboxes in the domain where the client type being used is the GroupWise 8.0 Mac Client. The accuracy of this count depends on the Audit process which is performed by the Post Office Agent. **--GroupWise 8.0 SP1 and later.**
OutlookConnector65XClientCount	R/O	Long. Returns the count of mailboxes in the domain where the client type being used is the GroupWise 6.5 Outlook Connector. The accuracy of this count depends on the Audit process which is performed by the Post Office Agent. **--GroupWise 8.0 SP1 and later.**

Property	Access	Description
OutlookConnector70XClientCount	R/O	Long. Returns the count of mailboxes in the domain where the client type being used is the GroupWise 7.0 Outlook Connector. The accuracy of this count depends on the Audit process which is performed by the Post Office Agent. **--GroupWise 8.0 SP1 and later.**
OutlookConnector80XClientCount	R/O	Long. Returns the count of mailboxes in the domain where the client type being used is the GroupWise 8.0 Outlook Connector. The accuracy of this count depends on the Audit process which is performed by the Post Office Agent. **--GroupWise 8.0 SP1 and later.**
Path	R/O	String
PostOffices	R/O	"PostOffices" on page 59
Resources	R/O	"Resources" on page 62
System	R/O	"System" on page 65
TotalGatewayCount	R/O	Long. This returns the count of gateways defined in the domain. **--GroupWise 8.0 SP1 and later.**
TotalGroupCount	R/O	Long. Returns the count of distribution lists assigned to post offices in the domain. **--GroupWise 8.0 SP1 and later.**
TotalLibraryCount	R/O	Long. Returns the count of libraries assigned to post offices in the domain. **--GroupWise 8.0 SP1 and later.**
TotalMailboxCount	R/O	Long. Returns the total count of mailboxes in the domain. The accuracy of this count depends on the Audit process which is performed by the Post Office Agent. **--GroupWise 8.0 SP1 and later.**
TotalPostOfficeCount	R/O	Long. Returns the count of post offices defined in the domain. **--GroupWise 8.0 SP1 and later.**
TotalResourceCount	R/O	Long. Returns the count of resources assigned to post offices in the domain. **--GroupWise 8.0 SP1 and later.**
TotalUserCount	R/O	Long. Returns the count of users assigned to post offices in the domain. **--GroupWise 8.0 SP1 and later.**
Windows65XClientCount	R/O	Long. Returns the count of mailboxes in the domain where the client type being used is the GroupWise 6.5 Windows Client. The accuracy of this count depends on the Audit process which is performed by the Post Office Agent. **--GroupWise 8.0 SP1 and later.**

Property	Access	Description
Windows70XClientCount	R/O	Long. Returns the count of mailboxes in the domain where the client type being used is the GroupWise 7.0 Windows Client. The accuracy of this count depends on the Audit process which is performed by the Post Office Agent. **--GroupWise 8.0 SP1 and later.**
Windows80XClientCount	R/O	Long. Returns the count of mailboxes in the domain where the client type being used is the GroupWise 8.0 Windows Client. The accuracy of this count depends on the Audit process which is performed by the Post Office Agent. **--GroupWise 8.0 SP1 and later.**
WinX65XClientCount	R/O	Long. Returns the count of mailboxes in the domain where the client type being used is the GroupWise 6.5 WinX Client. The accuracy of this count depends on the Audit process which is performed by the Post Office Agent. **--GroupWise 8.0 SP1 and later.**
WinX70XClientCount	R/O	Long. Returns the count of mailboxes in the domain where the client type being used is the GroupWise 7.0 WinX Client. The accuracy of this count depends on the Audit process which is performed by the Post Office Agent. **--GroupWise 8.0 SP1 and later.**
WinX80XClientCount	R/O	Long. Returns the count of mailboxes in the domain where the client type being used is the GroupWise 8.0 WinX Client. The accuracy of this count depends on the Audit process which is performed by the Post Office Agent. **--GroupWise 8.0 SP1 and later.**
UncategorizedMailboxCount	R/O	Long. Returns the total count of mailboxes in the domain where the client type could not be determined., The accuracy of this count depends on the Audit process which is performed by the Post Office Agent. **--GroupWise 8.0 SP1 and later.**
Users	R/O	"Users" on page 72
Version	R/O	Long

Methods

`Boolean IsUserNameUnique( String Name )`

Returns TRUE if there is no user in the domain with the given Name. Otherwise, returns FALSE.

`ClearAddressFormat(THIS)`

Removes the Internet Address Format setting for the domain.

`ClearAllowedAddressFormat(THIS)`

Removes the Allowed Internet Address Formats setting for the domain.

`ClearInternetDomainName(THIS)`

Removes the Internet Domain name setting for the domain.

`Commit()`

Writes out all changed information in the object.

`Refresh()`

Discards information associated with the object and re-reads the object.

`SetInternetDomainName(THIS_BSTR bstrIDomName, VARIANT_BOOL bIDomExclusive)`

Set the Internet Domain Name and the Known Exclusively By The Internet Domain Name setting for the domain.

Remarks

None.

Domains

This large collection contains Domain objects.

Properties

The following table lists the properties for this class, along with access and descriptions.

Property	Access	Description
_NewEnum	R/O	Enumeration object. Implements IEnumVARIANT. (For Windows only.)
ObjType	R/O	Enum(eadObjDomains)
System	R/O	"System" on page 65

Methods

`DomainsIterator CreateIterator()`

Creates a DomainsIterator.

`Domains Find( DomainTypeConstants DomainType )`

Finds all the Domain objects with the given DomainType.

`Domain Item( String Name )`

Returns the Domain object with the given Name.

`Domain ItemByDN( String DN, [String Tree] )`

Returns the Domain object with the given DN (distinguished name). If no Tree is supplied, the current tree is assumed.

`Domain ItemByObjectID( String ObjectID )`

Returns the Domain object with ObjectID.

Remarks

None.

DomainsIterator

This object iterates over all DomainsIterator belonging to an owning collection.

Properties

The following table lists the properties for this class, along with access and descriptions.

Property	Access	Description
ObjType	R/O	Enum(eadObjDomainsIterator)
System	R/O	"System" on page 65

Methods

`DomainsIterator Clone()`

Creates a new, independent DomainsIterator object. The clone starts at the original's current position in the collection. Clones are useful for saving the current position of an iterator.

`Domain Next()`

Returns the next Domain object, or nothing if the last Domain object has already been returned.

`Reset()`

Resets the DomainsIterator to its initial state.

`Skip( Long NumItems )`

Skips past the next NumItems Domain objects, or as many as possible if there are no remaining NumItems to skip. Returns an error if NumItems < 0.

Remarks

None.

Field

Allows modification of admin-defined fields.

Properties

The following table lists the properties for this class, along with access and descriptions.

Property	Access	Description
Definition	R/O	"FieldDefinition" on page 41
Name	R/O	String
ObjType	R/O	Enum(eadObjField)
System	R/O	"System" on page 65
Value	R/W	Returns only strings.
ValueType	R/O	Enum(FieldTypeConstants)

Methods

`Commit()`

Writes out all changed information in the object.

`Refresh()`

Discards information associated with the object and re-reads the object.

Remarks

Currently, the only ValueType available is eadString.

FieldDefinition

Allows modifications to field definitions on a Domain.

Properties

The following table lists the properties for this class, along with access and descriptions.

Property	Access	Description
FieldType	R/O	Enum(FieldTypeConstants)
Name	R/W	String
ObjType	R/O	Enum(eadObjFieldDefinition)
System	R/O	“System” on page 65

Methods

`Commit()`

Writes out all changed information in the object.

`Delete()`

Deletes the object.

`Refresh()`

Discards information associated with the object and re-reads the object.

Remarks

None.

FieldDefinitions

This collection contains FieldDefinition objects. There is currently a limit of 20 fields per user.

Properties

The following table lists the properties for this class, along with access and descriptions.

Property	Access	Description
Count	R/O	Long
_NewEnum	R/O	Enumeration object Implements IEnumVARIANT. (For Windows only.)
ObjType	R/O	Enum(eadFieldDefinitions)
System	R/O	"System" on page 65

Methods

`FieldDefinition Item( String Name )`

Returns the FieldDefinition object with the given Name.

`FieldDefinition Item( Long Index )`

Returns the FieldDefinition object at the specified Index in the collection.

Remarks

None.

Fields

This collection contains Field objects. There is currently a limit of 20 fields per user.

Properties

The following table lists the properties for this class, along with access and descriptions.

Property	Access	Description
Count	R/O	Long
_NewEnum	R/O	Enumeration object. Implements IEnumVARIANT. (For Windows only.)
ObjType	R/O	Enum(eadObjFields)
System	R/O	“System” on page 65

Methods

`Field Item( String Name )`

Returns the Field object with the given Name.

`Field Item( Long Index )`

Returns the Field object at the specified Index in the collection.

Remarks

None.

LookupEntry

A document management lookup entry object. This object is derived from the LookupTableEntry object and inherits all of its properties and methods.

Properties

The following table lists the properties for this class, along with access and descriptions.

Property	Access	Description
ParentValue	R/W	Variant
ParentValueType	R/O	Enum(DataTypeConstants). Defaulted to eadDataString, but meaningless if ParentValue does not exist.

Methods

None.

Remarks

None.

LookupTable

A document management lookup table object.

Properties

The following table lists the properties for this class, along with access and descriptions.

Property	Access	Description
DataType	R/O	Enum(DataTypeConstants)
Description	R/O	String
Domain	R/O	Domain
HasParentTable	R/O	Boolean
LookupTableEntries	R/O	LookupTableEntries
Name	R/O	String
ObjType	R/O	Enum(eadObjLookupTable)
ParentTable	R/O	LookupTable
PostOffice	R/O	"PostOffice" on page 54
System	R/O	"System" on page 65
TableType	R/O	Enum(TableTypeConstants)

Methods

`Delete()`

Deletes this object.

Remarks

None.

LookupTableEntries

This large collection contains LookupTableEntry objects.

Properties

The following table lists the properties for this class, along with access and descriptions.

Property	Access	Description
_NewEnum	R/O	Enumeration object. Implements IEnumVARIANT. (For Windows only.)
ObjType	R/O	Enum(eadObjLookupTableEntries)
System	R/O	"System" on page 65

Methods

`LookupTableEntry Add( Variant Value, [Variant ParentValue], [Long DocLife], [ExpirationTypeConstants ExpAction], [Long MaxVers] )`

Creates a LookupTableEntry with the given information and object properties. Value can be either a String or a numeric value. If the owning table has a ParentTable, ParentValue is required. DocLife, ExpAction, and MaxVers are only applicable if the LookupTable is the DocumentType Table.

`LookupTableEntriesIterator CreateIterator()`

Creates a LookupTableEntriesIterator object.

`LookupTableEntry Item( Variant Value, [Variant ParentValue] )`

Returns the LookupTableEntry with the given Value. Value can either be a String or a numeric value. ParentValue is only valid if this collection LookupTable has a parent table.

Remarks

None.

LookupTableEntriesIterator

This object iterates over all LookupTableEntry objects in a LookupTableEntries collection that belong to a PostOffice.

Properties

The following table lists the properties for this class, along with access and descriptions.

Property	Access	Description
ObjType	R/O	Enum(eadObjLookupTableEntriesIterator)
System	R/O	"System" on page 65

Methods

`LookupTableEntriesIterator Clone()`

Creates a new, independent LookupTableEntriesIterator object. The clone starts at the original's current position in the collection. Clones are useful for saving the current position of an iterator.

`LookupTableEntry Next()`

Returns the next LookupTableEntry object, or nothing if the last LookupTableEntry object has already been returned.

`Reset()`

Resets the LookupTableEntriesIterator to its initial value.

`Skip( Long NumItems )`

Skips past the next NumItems LookupTableEntry objects, or as many as possible if there are no remaining NumItems to skip. Returns an error if NumItems < 0.

Remarks

None.

LookupTableEntry

A document management lookup table entry.

Properties

The following table lists the properties for this class, along with access and descriptions.

Property	Access	Description
DataType	R/O	Enum(DataTypeConstants)
Domain	R/O	Domain
ObjType	R/O	Enum(eadObjDocumentTypeEntry, eadObjLookupEntry)
PostOffice	R/O	“PostOffice” on page 54
System	R/O	“System” on page 65
Table	R/O	“LookupTable” on page 45
Value	R/W	Variant

Methods

`Commit()`

Writes out all changed information in the object.

`Delete()`

Deletes this object.

`Refresh()`

Discards information associated with the object and re-reads the object.

Remarks

None.

LookupTables

This collection contains LookupTable objects.

Properties

The following table lists the properties for this class, along with access and descriptions.

Property	Access	Description
Count	R/O	Long
_NewEnum	R/O	Enumeration object. Implements IEnumVARIANT. (For Windows only.)
ObjType	R/O	Enum(eadObjLookupTables)
System	R/O	"System" on page 65

Methods

`LookupTable Add( String Name, [String Description], [LookupTable ParentTable], [DataTypeConstants DataType], [Long Value1], VARIANT Value2] )`

Creates a LookupTable with the given Name. Description and ParentTable are optional. The only valid LookupTables for ParentTable are LookupTables that do not have a parent themselves. Also, the ParentTable must be on the same PostOffice as the LookupTable you are adding. Currently, eadDataString and eadDataNumeric are the only valid values for DataType. If DataType is omitted, eadDataString is assumed.

- If DataType is eadDataString, Value1 is the maximum String length and Value2 is a StringCaseTypeConstants. If Value1 is omitted, 65535 is assumed. Note that Value1 must be between 1 and 65535. If Value2 is omitted, eadMixed is assumed.
- If DataType is eadDataString, Value1 is the maximum String length and Value2 is the maximum value. If Value1 is omitted, -2147483648 is assumed. If Value2 is omitted, 2147483627 is assumed. Valid ranges for Value1 and Value2 are: -2147483648 <= Value1 <= Value2 <= 2147483647.

`LookupTable Add( String Name, [String Description], String ParentTable], [DataTypeConstants DataType], [Long Value1], [VARIANT Value2] )`

Creates a LookupTable with the given Name. Description and ParentTable are optional. The only valid LookupTables for ParentTable are LookupTables that do not have a parent themselves. Also, the ParentTable must be on the same PostOffice as the LookupTable you are adding. Currently, eadDataString and eadDataNumeric are the only valid values for DataType. If DataType is omitted, eadDataString is assumed.

- If DataType is eadDataString, Value1 is the maximum String length and Value2 is a StringCaseTypeConstants. If Value1 is omitted, 65535 is assumed. Note that Value1 must be between 1 and 65535. If Value2 is omitted, eadMixed is assumed.
- If DataType is eadDataString, Value1 is the maximum String length and Value2 is the maximum value. If Value1 is omitted, -2147483648 is assumed. If Value2 is omitted, 2147483627 is assumed. Valid ranges for Value1 and Value2 are: -2147483648 <= Value1 <= Value2 <= 2147483647.

`LookupTable Item( Long Index )`

Returns the LookupTable object at the specified Index in the collection.

`LookupTable Item( String Name )`

Returns the LookupTable with the given Name.

Remarks

None.

Nickname

Lets you modify an existing GroupWise nickname. This object is derived from the base class AdminObject.

Properties

The following table lists the properties for this class, along with access and descriptions.

Property	Access	Description
Owner	R/W	"AdminObject" on page 13
OwnerType	R/O	Enum(eadObjDistributionList, eadObjResource, eadObjUser)

Methods

None.

Remarks

The Owner of a Nickname must exist in the same Domain and PostOffice as the Nickname.

Nicknames

This collection contains Nickname objects.

Properties

The following table lists the properties for this class, along with access and descriptions.

Property	Access	Description
_NewEnum	R/O	Enumeration object Implements IEnumVARIANT. (For Windows only.)
ObjType	R/O	Enum(eadObjNicknames)
System	R/O	"System" on page 65

Methods

`Nickname Add( String ObjName, AdminObject Owner, [String PO], [String Dom] )`

Creates a GroupWise Nickname object with the given ObjName and Owner. Owner may not be a Nickname object. See the table under Remarks.

`Nickname Add( String ObjName, AdminObject Owner, [PostOffice PO], [Domain Dom] )`

Creates a GroupWise Nickname object with the given ObjName and Owner. Owner may not be a Nickname object. See the table under Remarks.

`NicknamesIterator CreateIterator()`

Creates a NicknamesIterator object.

`Nickname Item( String Name, [String PO], [String Dom] )`

Returns the object with the given Name. See the table under Remarks.

`Nickname Item( String Name, [PostOffice PO], [Domain Dom] )`

Returns the object with the given Name. See the table under Remarks.

`Nickname ItemByObjectID( String ObjectID )`

Returns the Nickname object with the given ObjectID.

Remarks

The Owner of a Nickname must exist in the same Domain and PostOffice as the Nickname.

This table applies to the above methods as noted. It shows the parameters you can use with the methods depending on where you obtain the Nicknames object.

Source of Nicknames Object	Required Parameters	Optional Parameters
PostOffice		PO, Dom
Domain	PO	Dom
System	PO, Dom	

NicknamesIterator

Iterates over all Nickname objects in a Nicknames collection.

Properties

The following table lists the properties for this class, along with access and descriptions.

Property	Access	Description
ObjType	R/O	Enum(eadObjNicknamesIterator)
System	R/O	"System" on page 65

Methods

`NicknamesIterator Clone()`

Creates a new, independent NicknamesIterator object. The clone starts at the original's current position in the collection. Clones are useful for saving the current position of an iterator.

`Nickname Next()`

Returns the next "Nickname" on page 51 object, or nothing if the last Nickname has already been returned.

`Reset()`

Resets the NicknamesIterator to its intitial state.

`Skip( Long NumItems )`

Skips past the next NumItems Nickname objects, or as many as possible if there are no remaining NumItems to skip. Returns an error if NumItems is < 0.

Remarks

None.

PostOffice

Lets you modify an existing GroupWise PostOffice.

Properties

The following table lists the properties for this class, along with access and descriptions.

Property	Access	Description
AccessMode	R/W	Enum(AccessModeTypeConstants)
AddressFormat	R/W	Enum(AddressFormatConstants)
AllowedAddressFormat	R/W	Enum(AllowedAddressFormatsConstants)
DefaultSecurityLevel	R/W	Enum(SecurityTypeConstants)
DeliveryMode	R/W	Enum(DeliveryTypeConstants)
Description	R/W	String
DisableLogins	R/W	Boolean
DistinguishedName	R/O	String
DistributionLists	R/O	"DistributionLists" on page 15
Domain	R/O	"Domain" on page 33
External	R/O	Boolean
FullLicenseMailboxCount	R/O	Long. Returns the total count of mailboxes in the post office where a full-license client is used. The accuracy of this count depends on the Audit process which is performed by the Post Office Agent.
InactiveMailboxCount	R/O	Long. Returns the total count of inactive mailboxes in the post office. The accuracy of this count depends on the Audit process which is performed by the Post Office Agent. **--GroupWise 8.0 SP1 and later.**
InternetDomainName	R/O	String
InternetDomainNameExclusive	R/O	Boolean
LastAuditDate	R/O	Long. Returns the date the last audit process was run for the Post Office. **--GroupWise 8.0 SP1 and later.**
LastModifiedBy	R/O	String
Libraries	R/O	"DMSLibraries" on page 28
LimitedLicenseMailboxCount	R/O	Long. Returns the total count of mailboxes in the post office where a limited-license client is used. The accuracy of this count depends on the Audit process which is performed by the Post Office Agent. **--GroupWise 8.0 SP1 and later.**

Property	Access	Description
Linux65XClientCount	R/O	Long. Returns the total count of mailboxes in the post office where the client type being used is the GroupWise 6.5 Linux Client. The accuracy of this count depends on the Audit process which is performed by the Post Office Agent. **--GroupWise 8.0 SP1 and later.**
Linux70XClientCount	R/O	Long. Returns the total count of mailboxes in the post office where the client type being used is the GroupWise 7.0 Linux Client. The accuracy of this count depends on the Audit process which is performed by the Post Office Agent. **--GroupWise 8.0 SP1 and later.**
Linux80XClientCount	R/O	Long. Returns the total count of mailboxes in the post office where the client type being used is the GroupWise 8.0 Linux Client. The accuracy of this count depends on the Audit process which is performed by the Post Office Agent. **--GroupWise 8.0 SP1 and later.**
LookupTables	R/O	"LookupTables" on page 49
Mac5XClientCount	R/O	Long. Returns the total count of mailboxes in the post office where the client type being used is the GroupWise 5.X Mac Client. The accuracy of this count depends on the Audit process which is performed by the Post Office Agent. **--GroupWise 8.0 SP1 and later.**
Mac6XClientCount	R/O	Long. Returns the total count of mailboxes in the post office where the client type being used is the GroupWise 6.X Mac Client. The accuracy of this count depends on the Audit process which is performed by the Post Office Agent. **--GroupWise 8.0 SP1 and later.**
Mac70XClientCount	R/O	Long. Returns the total count of mailboxes in the post office where the client type being used is the GroupWise 7.0 Mac Client. The accuracy of this count depends on the Audit process which is performed by the Post Office Agent. **--GroupWise 8.0 SP1 and later.**
Mac80XClientCount	R/O	Long. Returns the total count of mailboxes in the post office where the client type being used is the GroupWise 8.0 Mac Client. The accuracy of this count depends on the Audit process which is performed by the Post Office Agent. **--GroupWise 8.0 SP1 and later.**
Name	R/O	String
Nicknames	R/O	"Nicknames" on page 52
ObjectID	R/O	String
ObjType	R/O	Enum(eadObjPostOffice)

Property	Access	Description
OutlookConnector65XClientCount	R/O	Long. Returns the total count of mailboxes in the post office where the client type being used is the GroupWise 6.5 Outlook Connector. The accuracy of this count depends on the Audit process which is performed by the Post Office Agent. **--GroupWise 8.0 SP1 and later.**
OutlookConnector70XClientCount	R/O	Long. Returns the total count of mailboxes in the post office where the client type being used is the GroupWise 7.0 Outlook Connector. The accuracy of this count depends on the Audit process which is performed by the Post Office Agent. **--GroupWise 8.0 SP1 and later.**
OutlookConnector80XClientCount	R/O	Long. Returns the total count of mailboxes in the post office where the client type being used is the GroupWise 8.0 Outlook Connector. The accuracy of this count depends on the Audit process which is performed by the Post Office Agent. **--GroupWise 8.0 SP1 and later.**
Path	R/O	String
Resources	R/O	"Resources" on page 62
System	R/O	"System" on page 65
TotalGroupCount	R/O	Long. Returns the count of distribution lists assigned to the Post Office. **--GroupWise 8.0 SP 1 and later.**
TotalLibraryCount	R/O	Long. Returns the count of resources assigned to the post office. **--GroupWise 8.0 SP1 and later.**
TotalMailboxCount	R/O	Long. Returns the total count of mailboxes in the Post Office. The accuracy of this count depends on the Audit process which is performed by the Post Office Agent. **--GroupWise 8.0 SP1 and later.**
TotalResourceCount	R/O	Long. Returns the count of users assigned to the post office. **--GroupWise 8.0 SP1 and later.**
TotalUserCount	R/O	Long. Returns the count of users assigned to the post office. **--GroupWise 8.0 SP1 and later.**
UncategorizedMailboxCount	R/O	Long. Returns the count of mailboxes in the post office where the client type could not be determined. The accuracy of this count depends on the Audit process which is performed by the Post Office Agent. **--GroupWise 8.0 SP1 and later.**
Users	R/O	"Users" on page 72

Property	Access	Description
Version	R/O	Long
Windows65XClientCount	R/O	Long. Returns the total count of mailboxes in the post office where the client type being used is the GroupWise 6.5 Windows Client. The accuracy of this count depends on the Audit process which is performed by the Post Office Agent. **--GroupWise 8.0 SP1 and later.**
Windows70XClientCount	R/O	Long. Returns the total count of mailboxes in the post office where the client type being used is the GroupWise 7.0 Windows Client. The accuracy of this count depends on the Audit process which is performed by the Post Office Agent. **--GroupWise 8.0 SP1 and later.**
Windows80XClientCount	R/O	Long. Returns the total count of mailboxes in the post office where the client type being used is the GroupWise 8.0 Windows Client. The accuracy of this count depends on the Audit process which is performed by the Post Office Agent. **--GroupWise 8.0 SP1 and later.**
WinX65XClientCount	R/O	Long. Returns the total count of mailboxes in the post office where the client type being used is the GroupWise 6.5 WinX Client. The accuracy of this count depends on the Audit process which is performed by the Post Office Agent. **--GroupWise 8.0 SP1 and later.**
WinX70XClientCount	R/O	Long. Returns the total count of mailboxes in the post office where the client type being used is the GroupWise 7.0 WinX Client. The accuracy of this count depends on the Audit process which is performed by the Post Office Agent. **--GroupWise 8.0 SP1 and later.**
WinX80XClientCount	R/O	Long. Returns the total count of mailboxes in the post office where the client type being used is the GroupWise 8.0 WinX Client. The accuracy of this count depends on the Audit process which is performed by the Post Office Agent. **--GroupWise 8.0 SP1 and later.**

Methods

`AdminObject FindObject( String MailboxID )`

Returns the Resource or User object specified by MailboxID.

`Boolean IsUserNameUnique( String Name )`

Returns TRUE if there is no User in the PostOffice with the given Name. Otherwise, returns FALSE.

`ClearAddressFormat(THIS)`

Removes the Internet Address Format setting for the post office.

`ClearAllowedAddressFormat(THIS)`

Removes the Allowed Internet Address Formats setting for the post office.

`ClearInternetDomainName(THIS)`

Removes the Internet Domain name setting for the post office.

`Commit()`

Writes out all changed information in the object.

`Refresh()`

Discards information associated with the object and re-reads the object.

`SetInternetDomainName(THIS_BSTR bstrIDomName, VARIANT_BOOL bIDomExclusive)`

Set the Internet Domain Name and the Known Exclusively By The Internet Domain Name settings for the post office.

Remarks

None.

PostOffices

This collection contains PostOffice objects.

Properties

The following table lists the properties for this class, along with access and descriptions.

Property	Access	Description
_NewEnum	R/O	Enumeration object. Implements IEnumVARIANT. (For Windows only.)
ObjType	R/O	Enum(eadObjPostOffices)
System	R/O	"System" on page 65

Methods

`PostOfficesIterator CreateIterator()`

Creates a PostOfficesIterator object.

`PostOffice Item( String Name, [String Domain] )`

Returns the PostOffice object with the given Name. If PostOffices is obtained from System, Dom is required.

`PostOffice Item( String Name, [Domain Dom] )`

Returns the PostOffice object with the given Name. If PostOffices is obtained from System, Dom is required.

`PostOffice ItemByDN( String DN, [String Tree] )`

Returns the PostOffice object with the given DN (distinguished name). If no Tree is supplied, the current tree is assumed.

PostOffice ItemByObjectID(String ObjectID)

Returns the PostOffice object with the given ObjectID.

Remarks

None.

PostOfficesIterator

Iterates over all PostOffice objects in a PostOffices collection.

Properties

The following table lists the properties for this class, along with access and descriptions.

Property	Access	Description
ObjType	R/O	Enum(eadObjPostOfficesIterator)
System	R/O	"System" on page 65

Methods

`PostOfficesIterator Clone()`

Creates a new, independent PostOfficesIterator object. The clone starts at the original's current position in the collection. Clones are useful for saving the current position of an iterator.

`PostOffice Next()`

Returns the next PostOffice object, or nothing if the last PostOffice object has already been returned.

`Reset()`

Resets the PostOfficesIterator to its intitial state.

`Skip( Long NumItems )`

Skips past the next NumItems PostOffice objects, or as many as possible if there are no remaining NumItems to skip. Returns an error if NumItems < 0.

Remarks

None.

Resource

Lets you modify an existing GroupWise Resource. This object is derived from AdminObject and inherits all its properties and methods.

Properties

The following table lists the properties for this class, along with access and descriptions.

Property	Access	Description
MailboxID	R/O	String
Nicknames	R/O	"Nicknames" on page 52
Owner	R/W	"User" on page 70
ResourceType	R/W	Enum(ResourceTypeConstants)

Methods

`MoveWithinTree( String DN )`

Moves this object to a different Organization or Organizational Unit specified by the given DN (distinguished name).

Remarks

The Owner of a Resource must exist in the same Domain and PostOffice as the Resource.

Resources

This collection contains Resource objects.

Properties

The following table lists the properties for this class, along with access and descriptions.

Property	Access	Description
_NewEnum	R/O	Enumeration object. Implements IEnumVARIANT. (For Windows only.)
ObjType	R/O	Enum(eadObjResources)
System	R/O	"System" on page 65

Methods

`Resource Add( String ObjName, String DN, User Owner, [String MailboxID], [String PO], [String Dom] )`

Creates a Resource object with the given ObjName, DN (distinguished name), and Owner. See Remarks below.

`Resource Add( String ObjName, String DN, User Owner, [String MailboxID], [ PostOffice PO], [Domain Dom] )`

Creates a Resource object with the given ObjName, DN (distinguished name), and Owner. See Remarks below.

`ResourcesIterator CreateIterator()`

Creates a ResourcesIterator object.

`Resource Item( String Name, [String PO], [String Dom] )`

Returns object with the given Name.

`Resource Item( String Name, [PostOffice PO], [Domain Dom] )`

Returns object with the given Name.

`Resource ItemByDN( String DN, [String Tree] )`

Returns the Resource object with the given DN (distinguished name). If no Tree is supplied, the current tree is assumed.

`Resource ItemByObjectID( String ObjectID )`

Returns the Resource object with the given ObjectID.

Remarks

The MailboxID parameter in the Add method is an optional three-character, alphanumeric file ID that is used to define the Resource in the User database. It is highly recommended that you do not specify a Mailbox ID, in which case the system will assign one for you. However, if you do specify a MailboxID, and it conflicts with an existing one, the system will automatically assign a different MailboxID.

This table applies to the above methods as noted. It shows the parameters you can use with the methods depending on where you obtain the Users object.

Source of Resources Object	Required Parameters	Optional Parameters
PostOffice		PO, Dom
Domain	PO	Dom
System	PO, Dom	

ResourcesIterator

Iterates over all Resources in a Resources collection.

Properties

The following table lists the properties for this class, along with access and descriptions.

Property	Access	Description
ObjType	R/O	Enum(eadResourcesIterator)
System	R/O	"System" on page 65

Methods

`ResourcesIterator Clone()`

Creates a new, independent ResourcesIterator object. The clone starts at the original's current position in the collection. Clones are useful for saving the current position of an iterator.

`Resource Next()`

Returns the next Resource , or nothing if the last Resource object has already been returned.

`Reset()`

Resets the ResourceIterator to its intitial state.

`Skip( Long NumItems )`

Skips past the next NumItems Resource objects, or as many as possible if there are no remaining NumItems to skip. Returns an error if NumItems < 0.

Remarks

None.

System

All objects are directly or indirectly contained by the System object.

Properties

The following table lists the properties for this class, along with access and descriptions.

Property	Access	Description
APIMajorVersion	R/O	Long. The Object API major version number.
APIMinorVersion	R/O	Long. The Object API minor version number.
ConnectedDomain	R/O	“Domain” on page 33
Description	R/W	String
DistributionLists	R/O	“DistributionLists” on page 15
Domains	R/O	“Domains” on page 38
FullLicenseMailboxCount	R/O	Long. Returns the total count of mailboxes in the system where a full-license client is used. The accuracy of this count depends on the Audit process which is performed by the Post Office Agent. **--GroupWise 8.0 SP1 and later.**
InactiveMailboxCount	R/O	Long. Returns the total count of inactive mailboxes in the system. The accuracy of this counts depends on the Audit process which is performed by the Post Office Agent. **--GroupWise 8.0 SP1 and later.**
LastModifiedBy	R/O	String
LimitedLicenseMailboxCount	R/O	Long. Returns the total count of mailboxes in the system where a limited-license client is used. The accuracy of this count depends on the Audit process which is performed by the Post Office Agent. **--GroupWise 8.0 SP1 and later.**
Linux65XClientCount	R/O	Long. Returns the total count of mailboxes in the system where the client type being used is the GroupWise 6.5 Linux Client. The accuracy of this count depends on the Audit process which is performed by the Post Office Agent. **--GroupWise 8.0 SP1 and later.**
Linux70XClientCount	R/O	Long. Returns the total count of mailboxes in the system where the client type being used is the GroupWise 7.0 Linux Client. The accuracy of the count depends on the Audit process which is performed by the Post Office Agent. **--GroupWise 8.0 SP1 and later.**

Property	Access	Description
Linux80XClientCount	R/O	Long. Returns the total count of mailboxes in the system where the client type being used is the GroupWise 8.0 Linux Client. The accuracy of this count depends on the Audit process which is performed by the Post Office Agent. **--GroupWise 8.0 SP1 and later.**
Mac5XClientCount	R/O	Long. Returns the total count of mailboxes in the system where the client type being used is the GroupWise 5.X Mac Client. The accuracy of this count depends on the Audit process which is performed by the Post Office Agent. **--GroupWise 8.0 SP1 and later.**
Mac6XClientCount	R/O	Long. Returns the total count of mailboxes in the system where the client type being used is the GroupWise 6.X Mac Client. The accuracy of this count depends on the Audit process which is performed by the Post Office Agent. **--GroupWise 8.0 SP1 and later.**
Mac70XClientCount	R/O	Long. Returns the total count of mailboxes in the system where the client type being used is the GroupWise 7.0 Mac Client. The accuracy of this count depends on the Audit process which is performed by the Post Office Agent. **--GroupWise 8.0 SP1 and later.**
Mac80XClientCount	R/O	Long. Returns the total count of mailboxes in the system where the client type being used is the GroupWise 8.0 Mac Client. The accuracy of this count depends on the Audit process which is performed by the Post Office Agent. **--GroupWise 8.0 SP1 and later.**
Name	R/O	String
Nicknames	R/O	"Nicknames" on page 52
NWLogin	R/W	Boolean. When TRUE and an unauthenticated NDS™ tree is accessed, a dialog box prompts for a password. When FALSE, the function accessing the tree fails.
ObjType	R/O	Enum(eadObjSystem)
OutlookConnector65XClientCount	R/O	Long. Returns the total count of mailboxes in the system where the client type being used is the GroupWise 6.5 Outlook Connector. The accuracy of this count depends on the Audit process which is performed by the Post Office Agent. **--GroupWise 8.0 SP1 and later.**
OutlookConnector70XClientCount	R/O	Long. Returns the total count of mailboxes in the system where the client type being used is the GroupWise 7.0 Outlook Connector. The accuracy of this count depends on the Audit process which is performed by the Post Office Agent. **--GroupWise 8.0 SP1 and later.**

Property	Access	Description
OutlookConnector80XClientCount	R/O	Long. Returns the total count of mailboxes in the system where the client type being used is the GroupWise 8.0 Outlook Connector. The accuracy of this count depends on the Audit process which is performed by the Post Office Agent. **--GroupWise 8.0 SP1 and later.**
PostOffices	R/O	"PostOffices" on page 59
Resources	R/O	"Resources" on page 62
TotalExternalDomainCount	R/O	Long. Returns the count of external domains defined in the system. **--GroupWise 8.0 SP1 and later.**
TotalExternalGatewayCount	R/O	Long. Returns the count of external gateways defined in the system. **--GroupWise 8.0 SP1 and later.**
TotalExternalGroupCount	R/O	Long. Returns the count of distribution lists assigned to external post offices in the system. **--GroupWise 8.0 SP1 and later.**
TotalExternalLibraryCounts	R/O	Long. Returns the count of libraries assigned to external post offices in the system. **--GroupWise 8.0 SP1 and later.**
TotalExternalPostOfficeCount	R/O	Long. Returns the count of external post offices defined in the system. **--GroupWise 8.0 SP1 and later.**
TotalExternalResourceCount	R/O	Long. Returns the count of resources assigned to external post offices in the system. **--GroupWise 8.0 SP1 and later.**
TotalExternalUserCount	R/O	Long. Returns the count of users assigned to external post offices in the system. **--GroupWise 8.0 SP1 and later.**
TotalInternalDomainCount	R/O	Long. Returns the count of internal domains defined in the system. **--GroupWise 8.0 SP1 and later.**
TotalInternalGatewayCount	R/O	Long. Returns the count of internal gateways defined in the system. **--GroupWise 8.0 SP1 and later.**
TotalInternalGroupCount	R/O	Long. Returns the count of distribution lists assigned to internal post offices in the system. **--GroupWise 8.0 SP1 and later.**

Property	Access	Description
TotalInternalLibraryCount	R/O	Long. Returns the count of libraries assigned to internal post offices in the system. **--GroupWise 8.0 SP1 and later.**
TotalInternalResourceCount	R/O	Long. Returns the count of resources assigned to internal post offices in the system. **--GroupWise 8.0 SP1 and later.**
TotalInternalPostOfficeCount	R/O	Long. Returns the count of internal post offices defined in the system. **--GroupWise 8.0 SP1 and later.**
TotalInternalUserCount	R/O	Long. Returns the count of users assigned to internal post offices in the system. **--GroupWise 8.0 SP1 and later.**
TotalMailboxCount	R/O	Long. Returns the total count of mailboxes in the system. The accuracy of this counts depends on the Audit process which is performed by the post office agent. **--GroupWise 8.0 SP1 and later.**
UncategorizedMailboxCount	R/O	Long. Returns the total count of mailboxes in the system where the client type could not be determined. The accuracy of this count depends on the Audit process which is performed by the Post Office Agent. **--GroupWise 8.0 SP1 and later.**
Users	R/O	"Users" on page 72
Windows65XClientCount	R/O	Long. Returns the total count of mailboxes in the system where the client type being used is the GroupWise 6.5 Windows Client. The accuracy of this count depends on the Audit process which is performed by the Post Office Agent. **--GroupWise 8.0 SP1 and later.**
Windows70XClientCount	R/O	Long. Returns the total count of mailboxes in the system where the client type being used is the GroupWise 7.0 Windows Client. The accuracy of this count depends on the Audit process which is performed by the Post Office Agent. **--GroupWise 8.0 SP1 and later.**
Windows80XClientCount	R/O	Long. Returns the total count of mailboxes in the system where the client type being used is the GroupWise 8.0 Windows Client. The accuracy of this count depends on the Audit process which is performed by the Post Office Agent. **--GroupWise 8.0 SP1 and later.**
WinX65XClientCount	R/O	Long. Returns the total count of mailboxes in the system where the client type being used is the GroupWise 6.5 WinX Client. The accuracy of this count depends on the Audit process which is performed by the Post Office Agent. **--GroupWise 8.0 SP1 and later.**

Property	Access	Description
WinX70XClientCount	R/O	Long. Returns the total count of mailboxes in the system where the client type being used is the GroupWise 7.0 WinX Client. The accuracy of this count depends on the Audit process which is performed by the Post Office Agent. **--GroupWise 8.0 SP1 and later.**
WinX80XClientCount	R/O	Long. Returns the total count of mailboxes in the system where the client type being used is the GroupWise 8.0 WinX Client. The accuracy of this count depends on the Audit process which is performed by the Post Office Agent. **--GroupWise 8.0 SP1 and later.**

Methods

`Commit()`

Writes out all changed information in the object.

`Connect( String Path )`

Connects to the domain database located at Path. Path may be specified using either a mapped path or the Universal Naming Convention (UNC).

`ConnectByDN( String DN, [String Tree] )`

Connects to the Domain specified by the given DN (distinguished name). If no Tree is supplied, the current tree is assumed.

`Boolean IsUserNameUnique( String Name )`

Returns TRUE if there is no User in the System with the given Name. Otherwise, returns FALSE.

`Object ItemByObjectID( String ObjectID )`

Returns the object with the given ObjectID.

`Refresh()`

Discards information associated with the object and re-reads the object.

Remarks

None.

User

Lets you modify an existing GroupWise user. It is derived from "AdminObject" on page 13 and inherits all its properties and methods.

Properties

The following table lists the properties for this class, along with access and descriptions.

Property	Access	Description
AccountID	R/W	String
AddressFormat	R/W	Enum(AddressFormatConstants)
AdminDefined	R/O	"Fields" on page 43
AllowedAddressFormat	R/W	Enum(AllowedAddressFormatsConstants)
Department	R/W	String
DisableLogin	R/W	Boolean
DMSAccessRights	R/O	"DMSAccessRightsCollection" on page 22
External	R/O	Boolean
ExternalEntity	R/O	Boolean
FaxNumber	R/W	String
GatewayAccess	R/W	String
GivenName	R/W	String
InternetDomainName	R/O	String
InternetDomainNameExclusive	R/O	Boolean
LastClientType	R/O	Enum(ClientTypeConstants)
LDAPAuthentication	R/W	String
MailboxExpDate	R/W	Date
MailboxID	R/O	String
MailboxLastLoginDate	R/O	Date.Returns the date of the last time the mailbox was logged into and updated.
MailboxSize	R/O	Long
NetID	R/W	String
Nicknames	R/O	"Nicknames" on page 52
PhoneNumber	R/W	String
PrefEMailID	R/W	String
Resources	R/O	"Resources" on page 62
Surname	R/W	String
Title	R/W	String

Methods

`ClearAddressFormat(THIS)`

Removes the Internet Address Format settings for the user.

`ClearAllowedAddressFormat(THIS)`

Removes the Allowed Internet Address Formats setting for the user.

`ClearInternetDomainName(THIS)`

Removes the Internet Domain name setting for the user.

`ClearPassword( [StoreTypeConstants StoreType], [String CurrentPassword] )`

Clears the user's NetWare password, GroupWise password, or both, according to the specified StoreType. If no StoreType is specified, eadGW is assumed. For GroupWise, CurrentPassword is ignored. For NDS, CurrentPassword is required if the administrator does not have Write rights to the User object.

`Move( PostOffice PO )`

Moves this User to the PostOffice specified by PO.

If you call Move, you should not call User.Commit, as derived from AdminObject (page 13), afterward. When an object is moved, an internal operation is performed that changes the distinguished name of the User. Any subsequent Commit method on the original User object rewrites the original distinguished name and causes errors.

`MoveWithinTree( String DN )`

Moves this User to a different Organization or Organizational Unit specified by the given DN (distinguished name).

If you call MoveWithinTree, you should not call User.Commit, as derived from AdminObject (page 13), afterward. When an object is moved, an internal operation is performed that changes the distinguished name of the User. Any subsequent Commit method on the original User object rewrites the original distinguished name and causes errors.

`SetInternetDomainName(THIS_BSTR bstrIDomName, VARIANT_BOOL bIDomExclusive)`

Set the Internet Domain Name and the Known Exclusively By The Internet Domain Name settings for the user.

```
SetPassword( String Password, [StoreTypeConstants StoreType],
[String CurrentPassword] )
```

Sets the user's NetWare password, GroupWise password, or both. This is not a valid method for external users. For external entities, eadGW is the only valid StoreType. For GroupWise, CurrentPassword is ignored. For NDS, CurrentPassword is required if the administrator does not have Write rights to the User object.

Remarks

ClearPassword() and SetPassword() need to check NDS for rights to change GroupWise passwords.

Users

This collection contains User objects.

Properties

The following table lists the properties for this class, along with access and descriptions.

Property	Access	Description
_NewEnum	R/O	Enumeration object. Implements IEnumVARIANT. (For Windows only.)
ObjType	R/O	Enum(eadObjUsers)
System	R/O	"System" on page 65

Methods

`User Add( String UserName, String SurName, String DN, [String Password], [String PO], [String Dom], [String MailboxID] )`

Adds a User with the given UserName, Surname, and DN (distinguished name) to NDS and GroupWise. Password will apply to both NDS and GroupWise. See Remarks below.

`User Add( String UserName, String SurName, String DN, [String Password], [PostOffice PO], [Domain Dom], [String MailboxID] )`

Adds a User with the given UserName, Surname, and DN (distinguished name) to NDS and GroupWise. Password will apply to both NDS and GroupWise. See Remarks below.

`User AddExistingUser( String UserName, String DN, [String Password], [String PO], [String Dom], [String MailboxID] )`

Adds an existing NDS User with the given UserName and DN (distinguished name) to GroupWise. Password will apply to GroupWise only. See Remarks below.

`User AddExistingUser( String UserName, String DN, [String Password], [PostOffice PO], [Domain Dom], [String MailboxID] )`

Adds an existing NDS User with the given UserName and DN (distinguished name) to GroupWise. Password applies to GroupWise only. See Remarks below.

`User AddExternalEntity( String UserName, String Surname, String DN, [String NetID], [String Password], [String PO], [String Dom], [String MailboxID] )`

Adds an external entity with the given UserName, Surname, and DN (distinguished name) to NDS and GroupWise. Password applies to GroupWise only. See Remarks below.

`User AddExternalEntity( String UserName, String Surname, String DN, [String NetID], [String Password], [PostOffice PO], [Domain Dom], [String MailboxID] )`

Adds an external entity with the given UserName, Surname, and DN (distinguished name) to NDS and GroupWise. NetID is the User's ID to the Network Operating System (NOS). Password will apply to GroupWise only. See Remarks below.

`User AddExternalUser( String UserName, [String PO], [String Dom] )`

Adds an external User with the given UserName to GroupWise. Dom, if supplied, must be an external domain. See Remarks below.

`User AddExternalUser( String UserName, [PostOffice PO], [Domain Dom] )`

Adds an external User with the given UserName to GroupWise. Dom, if supplied, must be an external domain. See Remarks below.

`UsersIterator CreateIterator()`

Creates a UsersIterator object.

`Users Find( String GivenName, String Surname )`

Finds all User objects with the specified GivenName and Surname. Partial names are signified by ending the String with the "*" character.

`User Item( String Name, [String PO], [String Dom] )`

Returns the User object with the given Name. See Remarks below.

`User Item( String Name, [PostOffice PO], [Domain Dom] )`

Returns the User object with the given Name. See Remarks below.

`User ItemByDN( String DN [String Tree] )`

Returns the User object with the given DN (distinguished name). If no Tree is supplied, the current tree is assumed.

`User ItemByObjectID( String ObjectID )`

Returns the object with the given ObjectID.

Remarks

The MailboxID parameter in the above methods is an optional three-character, alphanumeric file ID that is used to define the User in the User database. It is highly recommended that you do not specify a Mailbox ID, in which case the system will assign one for you. However, if you do specify a MailboxID, and it conflicts with an existing one, the system will automatically assign a different MailboxID.

This table applies to the above methods as noted. It shows the parameters you can use with the methods depending on where you obtain the Users object.

Source of Users Object	Required Parameters	Optional Parameters
PostOffice		PO, Dom
Domain	PO	Dom
System	PO, Dom	

UsersIterator

This object iterates over all User objects in a Users collection.

Properties

The following table lists the properties for this class, along with access and descriptions.

Property	Access	Description
ObjType	R/O	Enum(eadObjUsersIterator)
System	R/O	“System” on page 65

Methods

`UsersIterator Clone()`

Creates a new, independent UsersIterator object. The clone starts at the original’s current position in the collection. Clones are useful for saving the current position of an iterator.

`User Next()`

Returns the next User object, or nothing if the last User object has already been returned.

`Reset()`

Resets the UsersIterator to its intitial state.

`Skip( Long NumItems )`

Skips past the next NumItems User objects, or as many as possible if there are no remaining NumItems to skip. Returns an error if if NumItems < 0.

Remarks

None.

2.2 Constants

This section contains constants and their values for the Administrative Object API.

“AccessModeTypeConstants” on page 75

“DataTypeConstants” on page 76

“DeliveryTypeConstants” on page 76

“DomainTypeConstants” on page 76

“ExpirationTypeConstants” on page 76

“FieldDefinitionTypeConstants” on page 76

“FieldTypeConstants” on page 76

“MemberTypeConstants” on page 77

“ObjTypeConstants” on page 77

“ParticipationTypeConstants” on page 78

“ResourceTypeConstants” on page 78

“SecurityTypeConstants” on page 78

“StoreTypeConstants” on page 78

“StringCaseTypeConstants” on page 78

“TableTypeConstants” on page 79

“VisibilityTypeConstants” on page 79

2.2.1 AccessModeTypeConstants

eadCSOnly = 0
eadCSDirect = 1
eadDirectOnly = 2

2.2.2 AddressFormatConstants

eadAddrFmtFull = 0
eadAddrFmtPO = 1
eadAddrFmtUserID = 2
eadAddrFmtLastFirst = 3
eadAddrFmtFirstLast = 4
eadAddrFmtLastNone =5
eadAddrFmtFLast = 6
eadAddrFmtForceFull = 7

2.2.3 AllowedAddressFormatsConstants

eadAllowedAddrFmtPO = 2
eadAllowAddrFmtUser = 4
eadAllowAddrFmtLastFirst = 8
eadAllowAddrFmtFirstLast = 16
eadAllowAddrFmtFLast = 32

2.2.4 ClientTypeConstants

eadClientUnknown = 0
eadClientOther = 1
eadClientWin32_65X = 2
eadClientWin32_70X = 3
eadClientWin32_80X = 4
eadClientLinux_65X = 5
eadClientLinux_70X = 6
eadClientLinux_80X = 7
eadClientMac_5X = 8
eadClientMac_6X =9
eadClientMac_70X = 10
eadClientMac_80X = 11
eadClientOutlook_65X =12
eadClientOutlook_70X = 13
eadClientOutlook_80X = 14
eadClientWinX_65X = 15
eadClientWinX_70X = 16
eadClientWinX_80X = 17

2.2.5 DataTypeConstants

eadDataString = 0
eadDataNumeric = 1
eadDataDate = 2
eadDataBinary = 3

2.2.6 DeliveryTypeConstants

eadAppThreshold = 0
eadAgentAlways = 1
eadClientLocal = 2

2.2.7 DomainTypeConstants

eadSecondary = 1
eadPrimary = 2
eadExternal = 3
eadNonGroupWise = 4

2.2.8 ExpirationTypeConstants

eadArchive = 0
eadDelete = 1
eadRetain = 2

2.2.9 FieldDefinitionTypeConstants

eadCustomFieldDefinition = 0
eadSystemFieldDefinition = 1

2.2.10 FieldTypeConstants

eadString = 0
eadNumeric = 1
eadDate = 2
eadBinary = 3

2.2.11 get_LastClientTypeConstants

eadClientUnknown = 0
eadClient Other = 1
eadClientWin32_65X = 2
eadClientWin32_70X = 3
eadClientWin32_80X = 4
eadClientLinux_65X = 5
eadClientLinux_70X = 6

eadClientLinux_80X = 7
eadClientMac_5X = 8
eadClientMac_6X = 9
eadClientMac_70X = 10
eadClientMac_80X = 11
eadClientOutlook_65X = 12
eadClientOutlook_70X = 13
eadClientOutlook_80X = 14
eadClientWinX_65X = 15
eadClientWinX_70X = 16
eadClientWinX_80X = 17

2.2.12 MemberTypeConstants

eadDistributionListMember = 0
eadUserMember = 1
eadNoMember = 2

2.2.13 ObjTypeConstants

eadObjDistributionList = 2
eadObjNickname = 3
eadObjResource = 5
eadObjUser = 6
eadObjDistributionLists = 7
eadObjDistributionListsIterator = 8
eadObjDLMember = 9
eadObjDLMembers = 10
eadObjDLMembersIterator = 11
eadObjDomain = 12
eadObjDomains = 13
eadObjDomainsIterator = 14
eadObjField = 15
eadObjFields = 16
eadObjFieldDefinition = 17
eadObjFieldDefinitions = 18
eadObjNicknames= 19
eadObjNicknamesIterator = 20
eadObjPostOffice = 21
eadObjPostOffices = 22
eadObjPostOfficesIterator = 23
eadObjResources = 24
eadObjResourcesIterator = 25
eadObjSystem = 26
eadObjUsers = 27
eadObjUsersIterator = 28
eadObjDMSFieldDefinition = 29

eadObjDMSFieldDefinitions = 30
eadObjDMSLibrary = 31
eadObjDMSLibraries = 32
eadObjDMSLibrariesIterator = 33
eadObjDMSAccessRights = 34
eadObjDMSAccessRightsCollection = 35
eadObjDMSAccessRightsCollectionIterator = 36
eadObjDocumentTypeEntry = 37
eadObjLookupEntry = 38
eadObjLookupTable = 39
eadObjLookupTables = 40
eadObjLookupTableEntry = 41
eadObjLookupTableEntries = 42
eadObjLookupTableEntriesIterator = 43

2.2.14 ParticipationTypeConstants

eadTo = 0
eadCC = 1
eadBC = 2

2.2.15 ResourceTypeConstants

eadResPlace = 4
eadResResource = 5

2.2.16 SecurityTypeConstants

eadLow = 0
eadHigh = 1
eadEDirectoryAuthentication = 2
eadLDAPAuthentication = 4

NOTE: The Security type constants are bit values. If Low security is on, the enumeration will only contain the eadLow value. If High security is turneed on, the value may contain a combination of the eadHigh, eadEDirectoryAuthentication, and eadLDAPAuthentication values.

2.2.17 StoreTypeConstants

eadNDS = 0
eadGW = 1
eadBothNDSandGW = 2

2.2.18 StringCaseTypeConstants

eadUpper = 1
eadLower = 2

eadMixed = 3

2.2.19 TableTypeConstants

eadAddressBook = 0
eadDocumentTypeTable = 1
eadLibraryRegistry = 2
eadLookupTable = 3
eadSystem = 4

2.2.20 VisibilityTypeConstants

eadVisPostOffice = 1
eadVisSystem = 2
eadVisDomain = 3
eadVisNone = 4

2.3 Exceptions

This section contains exceptions returned from the Administrative Object API, and possible reasons why the exceptions occurred.

2.3.1 Admin commit failed

- The user has insufficient NDS™ rights to the object.
- The disk is full.

2.3.2 Already connected

- The user is trying to connect to a domain to which he or she is already connected.

2.3.3 An error occurred in allocating memory

- There is not enough memory to allocate the object.

2.3.4 An invalid argument was passed in the function call

- One or more required arguments were not supplied.
- One or more arguments were out of range or otherwise invalid.

2.3.5 An invalid object was referenced

- The user is trying to access an object that has already been deleted from memory.

2.3.6 An invalid pointer was used

- The returned value was assigned to a NULL pointer.

2.3.7 Can't move Nickname objects

- Nickname objects can be created and deleted, but not moved.

2.3.8 Cannot add DMSAccessRights

- The user has insufficient NDS rights to the object.
- The disk is full.

2.3.9 Cannot change property. Object is marked as ReadOnly.

- The user is trying to change a property on a DMSFieldDefinition object that is ReadOnly.

2.3.10 Cannot commit a system defined DMSFieldDefinition

- System defined DMSFieldDefinition objects cannot be modified.

2.3.11 Cannot delete a system defined DMSFieldDefinition

- System defined DMSFieldDefinition objects cannot be deleted.

2.3.12 Could not create the User in NDS

- The user supplied an invalid distinguished name.
- A User object with the same distinguished name already exists in the post office.
- A User object with the same name already exists in the post office.

2.3.13 Couldn’t add User

- The user supplied an invalid distinguished name.

2.3.14 Couldn’t move object within the tree

- The user has insufficient NDS rights to the object.

2.3.15 Couldn’t move User

- A User with the same name already exists in the post office.

2.3.16 DistributionList not found

- The user supplied an invalid distinguished name, name, or object ID.

2.3.17 DMSLibrary commit failed

- Temporarily lost connection to the domain database.
- The domain database is corrupt.
- The disk is full.
- Too many files are open at the same time.
- Another user is currently updating the domain database—try the action again later.
- Not enough memory to perform the operation.
- The user has insufficient rights to the domain database directory.
- The user has insufficient NDS rights to the object.

2.3.18 DMSLibrary not found

- The user supplied an invalid distinguished name, name, or object ID.

2.3.19 Domain commit failed

- The user has insufficient NDS rights to the object.
- The disk is full.

2.3.20 Domain Distinguished Name does not exist in the current tree

- The user supplied an invalid distinguished name.

2.3.21 Domain not found

- The user supplied an invalid name or object ID.
- The domain database is corrupt.

2.3.22 Error adding Library. Library name already exists.

- A DMS library with the same name already exists in the post office.

2.3.23 Error adding LookupTable object to the collection

- Temporarily lost connection to the domain database.
- The domain database is corrupt.
- The disk is full.
- Too many files are open at the same time.
- Another user is currently updating the domain database—try the action again later.
- Not enough memory to perform the operation.
- The user has insufficient rights to the domain database directory.

2.3.24 Error adding LookupTableEntry object to the collection

- Temporarily lost connection to the domain database.
- The domain database is corrupt.
- The disk is full.
- Too many files are open at the same time.
- Another user is currently updating the domain database—try the action again later.
- Not enough memory to perform the operation.
- The user has insufficient rights to the domain database directory.

2.3.25 Error adding object to DLMembers collection

- An object with the same name already exists in the collection.

2.3.26 Error committing DMSFieldDefinition to guardian database

- Temporarily lost connection to the domain database.The domain database is corrupt.
- The disk is full.
- Too many files are open at the same time.
- Another user is currently updating the domain database—try the action again later.
- Not enough memory to perform the operation.
- The user has insufficient rights to the domain database directory.
- The user has insufficient rights to the domain database directory.

2.3.27 Error committing Nickname changes

- The disk is full.

2.3.28 Error deleting DistributionList

- The user has insufficient NDS rights to the object.
- The Domain database is corrupt.

2.3.29 Error deleting DMSFieldDefinition from guardian database

- Temporarily lost connection to the domain database.
- The domain database is corrupt.
- The disk is full.
- Too many files are open at the same time.
- Another user is currently updating the domain database—try the action again later.
- Not enough memory to perform the operation.
- The user has insufficient rights to the domain database directory.

2.3.30 Error deleting LookupTable

- Temporarily lost connection to the domain database.
- The domain database is corrupt.
- The disk is full.
- Too many files are open at the same time.
- Another user is currently updating the domain database—try the action again later.
- Not enough memory to perform the operation.
- The user has insufficient rights to the domain database directory.
- The lookup table is a parent of another lookup table. You must delete the child lookup table first.
- The lookup table is used by DMSFieldDefinition.

2.3.31 Error deleting LookupTableEntry

- Temporarily lost connection to the domain database.
- The domain database is corrupt.
- The disk is full.
- Too many files are open at the same time.
- Another user is currently updating the domain database—try the action again later.
- Not enough memory to perform the operation.
- The user has insufficient rights to the domain database directory.

2.3.32 Error deleting Nickname

- The user has insufficient NDS rights to the object.
- The domain database is corrupt.

2.3.33 Error deleting Resource

- The user has insufficient NDS rights to the object.
- The domain database is corrupt.

2.3.34 Error deleting User

- The user has insufficient NDS rights to the object.
- The domain database is corrupt.

2.3.35 Error occurred opening the database. Check the path.

- The user supplied an invalid path or distinguished name.
- The user has insufficient rights to the domain database directory.

2.3.36 Error retrieving Domain Object

- The domain database is corrupt.

2.3.37 Error retrieving PostOffice Object

- The domain database is corrupt.

2.3.38 External Users don’t have passwords

- The user tried to change the password of an external user.

2.3.39 Field not found

- The user supplied an invalid name.

2.3.40 FieldDefinition not found

- The user supplied an invalid name.

2.3.41 Invalid tree

- The user supplied an invalid tree name while trying to connect to a domain database.

2.3.42 Library cannot be added. Error unknown.

- The user supplied an invalid distinguished name or name.

2.3.43 LookupEntry refresh failed

- Not enough memory to complete the operation.

2.3.44 LookupTable not found

- The user supplied an invalid name.

2.3.45 LookupTableEntry commit failed

- Not enough memory to complete the operation.

2.3.46 LookupTableEntry refresh failed

- Not enough memory to complete the operation.

2.3.47 Member object or DMSLibrary could not be found

- The user supplied an invalid domain name, distribution list name, DMS library name, post office name, or user name while trying to create a DMSAccessRights object.

2.3.48 Missing Language DLL

- The language resource file is missing or has an invalid entry in the Registry.

2.3.49 NDS Tree could not be set

- The domain doesn't exist or is invalid.

2.3.50 Nickname not found

- The user supplied an invalid name or object ID.

2.3.51 Not an external domain

- The user tried to add an external user to a domain that isn't an external domain.

2.3.52 Not connected

- The user is trying to access System properties before connecting to a domain.

2.3.53 Object not found

- The user supplied an invalid object ID while trying to obtain an object within the system.

2.3.54 Only External Users can be added to an External Domain

- The user tried to add a user object that is not an external user to an external domain.

2.3.55 PostOffice commit failed

- The user has insufficient NDS rights to the object.
- The disk is full.

2.3.56 PostOffice not found

- The user supplied an invalid distinguished name, mailbox ID, name, or object ID.

2.3.57 Property put allowed only on a User object

- User is the only subtype of AdminObject in which the name can be changed.

2.3.58 Resource not found

- The user supplied an invalid distinguished name, name, or object ID.

2.3.59 The current user does not have rights for the operation

- The user has insufficient NDS rights to the object.

2.3.60 The Domain parameter must be supplied

- The user tried to obtain a User object from the system without supplying the required domain.

2.3.61 The IsLibrarian right cannot be set within the library's default rights

- The IsLibrarian property of a DMSAccessRights object cannot be changed if that DMSAccessRights object corresponds to the DefaultRights property of the DMSLibrary object.

2.3.62 The library's default access rights cannot be deleted

- The DMSAccessRights object, that corresponds to the DefaultRights property of a DMSLibrary, cannot be deleted.

2.3.63 The PostOffice parameter must be supplied

- The user tried to obtain a User object from the system or domain without supplying the required post office.

2.3.64 The Tree parameter needs to be supplied

- The user is trying to obtain a DistributionList, DMSLibrary, PostOffice, Resource, or User object with an ambiguous distinguished name. The Tree parameter is needed to obtain the object.

2.3.65 This LookupEntry doesn't have a ParentValue

- The user is trying to access the ParentValue property of a lookup entry that doesn't have a parent value.

2.3.66 User commit failed

- The user has insufficient NDS rights to the object.
- The disk is full.

2.3.67 User not found

- The user supplied an invalid distinguished name, name, or object ID.
- The user supplied an invalid distinguished name while tying to add an existing NDS user.
- The domain database is corrupt.

2.3.68 You can only set the NetID for External Entities

- The user was trying to set the NetID of a User object that is not an external entity.

2.4 Sample Applications

The following two sample applications are included with GroupWise Administrative Object API:

- Section 2.4.1, "Admn_Api," on page 88
- Section 2.4.2, "GwAdmin," on page 89

2.4.1 Admn_Api

A small application that prompts you to enter a domain name, connects to the domain, and then extracts information from it.

Using this sample application, you can extract the domain description, the NDS™ distinguished name, and the user ID of the person who last modified the domain.

This sample application also shows how to add a new user to the current domain.

2.4.2 GwAdmin

A Visual Basic application designed to exercise most of the properties and methods of the GroupWise Administrative Object API.

After connecting to a specified domain, you will be presented with a tree view display of your system. Navigating the tree view will allow you to see the different objects related to your system. Context menus are available by right-clicking on each of the different object types in the tree view. These menus will allow you to perform different tasks on the selected object.

A Revision History

The following table lists changes made to the GroupWise Administrative Object documentation (in reverse chronological order):

Release Date	Changes
November 2012	Reviewed and updated for use with GroupWise 2012.
March 2009	Added support for GroupWise 7.0.3, and 8.x to documentation.
March 2006	Added PrefEMailID property to User (page 70). Removed the Delete method from Field (page 40) because it was never implemented. Added navigational links.
October 2005	Transitioned to revised Novell documentation standards.
February 2004	Added the Member property to DMSAccessRights (page 21), two parameters to the Add method of DMSFieldDefinitions (page 27), the Delete method to Field (page 40) and FieldDefinition (page 41), and a second Item method to Resources (page 62). Updated the description of the Value property of Field (page 40).
December 2003	Added information about the Move and MoveWithinTree methods to the User (page 70) object.
September 2001	Added support for GroupWise 6.x to documentation.
June 2001	Changed description of Fields (page 43) to contain Field objects. Made minor formatting changes to improve readability. Added table headings.
July 1998	Documentation added to the Novell Developer Kit.