Platform SDK: Active Directory Service Interfaces

ADSI Attribute Syntax

Each attribute in the directory has an associated syntax. For example, integer, string, numeric, and so on. ADSI defines its own syntax that maps to the native directory syntax. This section describes the types of attribute syntaxes in ADSI. The next section contains a table that maps NDS and Active Directory syntax.

Distinguished Name String

```
Syntax Type: ADSTYPE_DN_STRING
```

The distinguished name is useful for linking two objects together. For example, it can create a link that makes the Alice object a manager of the Bob object. If the Alice object moves to different place, the manager link between Alice and Bob is updated automatically.

The distinguished name must contain a valid distinguished name object. If the distinguished name does not correspond to a valid existing object, most servers reject the request and return a constraint violation error.

Examples:

```
Set x = GetObject("LDAP://CN=Bob, OU=Sales,DC=Fabrikam, DC=com)
x.Put "manager", "CN=Alice, OU=Sales, DC=Fabrikam, DC=COM"
x.SetInfo

PADS_ATTR_INFO pInfo;
// .. IDirectoryObject::GetObjectAttribute
printf("%S\n", pInfo->pADsValues->DNString );
```

Case Exact String and Case Ignore String

```
Syntax Types: ADSTYPE_CASE_IGNORE_STRING, ADSTYPE_CASE_EXACT_STRING.
```

Case Exact String is a case-sensitive string while Case Ignore String is a case-insensitive string. A large percentage of attributes in the directory use this syntax.

> **Note** The directory may or may not store this as a Unicode string. However, ADSI accepts and returns Unicode strings.

Example:

```
Dim propList As IADsPropertyList
Set propList = GetObject("LDAP://DC=Fabrikam,DC=com")
Set propVal = New PropertyValue

' --- Property Value ---
propVal.CaseIgnoreString = "Fabrikam, Inc  - Seattle, WA"
propVal.ADsType = ADSTYPE_CASE_IGNORE_STRING
```

Printable String

```
Syntax Type: ADSTYPE_PRINTABLE_STRING
```

This syntax is used for attributes with string values where uppercase and lowercase are considered unequal for comparisons, for example, "FABRIKAM" and "Fabrikam" do not match. ADSI accepts any contents for a Printable-String; it does not attempt to verify that they are indeed printable.

Numeric String

```
Syntax Type: ADSTYPE_NUMERIC_STRING
```

In this syntax, strings match as in Printable String, except that all space characters are ignored in comparisons. ADSI does not perform value checking to ensure that only numerals and spaces appear in values of this syntax. Active Directory accepts any content for a numeric string; it does not verify that the characters are numeric.

UTC Time

```
Syntax Type: ADSTYPE_UTC_TIME
```

This syntax stores the date and time in a single string. The string format consists of three concatenated parts: (1) YYMMDD; (2) HHMM or HHMMSS (both are acceptable); and (3) "Z" to indicate that the time given is Greenwich Mean Time (GMT), or "+/-HHMM" to indicate that the time given is local time with the given differential from GMT. The differential is based on the formula: GMT=Local+differential.

Note The first two digits of the year are not stored in this string.

Some examples of legal values are "9101311455Z", "910131145503Z", "9101314455-0500", "910131145503+0130". This string is stored as single-byte ASCII characters, and no code page number is stored with it.

Although ordering is supported, it is done only as an ASCII case-insensitive string sort, not by properly interpreting the meaning of the strings.

Any valid string value is accepted. No attempt is made to ensure that the string contains a valid time string.

Generalized Time

```
Syntax Type: ADSTYPE_UTC_TIME
```

If a new attribute to store time values is being defined, the GeneralizedTime syntax should be used. GeneralizedTime syntax uses four characters to represent the year instead of two as with UTCTime.

The format for the GeneralizedTime syntax is "YYYYMMDDHHMMSS.0Z". An example of an acceptable value is "20010928060000.0Z". The "Z" indicates no time differential. Active Directory stores date/time as Greenwich Mean Time (GMT). If no time differential is specified, GMT is the default.

If the time is specified in a time zone other than GMT, the differential between the time zone and GMT is appended to the string instead of "Z" in the form "YYYYMMDDHHMMSS.0[+/-]HHMM". An example of an acceptable value is "20010928060000.0+0200".

The differential is based on the formula: GMT=Local+differential.

Boolean

```
Syntax Type: ADSTYPE_BOOLEAN
```

Active Directory accepts only a signed 32-bit value for this syntax. It handles zero as FALSE and all nonzero values as TRUE.

Integer

```
Syntax Type: ADSTYPE_INTEGER
```

A 32-bit signed numeric value.

Large Integer

```
Syntax Type: ADSTYPE_LARGE_INTEGER
```

A 64-bit signed numeric value. Large integers are actually implemented as COM objects on the IADsLargeInteger interface. The **HighPart** and **LowPart** methods are used to access the two 32-bit halves of the large integer value.

Example:

```
Dim x as IADsLargeInteger
Set o = GetObject("LDAP://DC=Fabrikam,DC=com")
Set x = o.Get("UsnCreated")
Debug.Print x.HighPart
Debug.Print x.LowPart
```

Octet String

```
Syntax Type: ADSTYPE_OCTET_STRING
```

An octet string is returned as a variant array of bytes. This consists of a size count (number of octets) followed by a series of octets. An octet is an 8-bit byte, so a series of octets is a string of binary data.

Object Class

```
Syntax Type: ADSTYPE_CASE_IGNORE_STRING
```

Object Class is a unique object identifier for a given schema class. The class of each object instance is identified by the **objectClass** attribute. When created, you can never change an object class. **objectClass** is a multiple valued attribute. It lists the specific class of the object, and the classes of all structural or abstract classes from which the specific class was derived. This includes Top, the class from which all other classes are ultimately derived. Active Directory does not list auxiliary classes in the **objectClass** attribute.

Security Descriptor

```
Syntax Type: ADSTYPE_NT_SECURITY_DESCRIPTOR
```

Access rights define what abilities a security principal has when it attempts to perform an operation on an Active Directory object. A security descriptor describes the access control information associated with an object.

The security descriptor is stored as a property of a directory object in the nTSecurityDescriptor property. When an authenticated user attempts to access a directory object, the directory server determines the access granted or denied to the user based on the object security descriptor.

The ADS_SD_CONTROL_ENUM enumeration specifies control flags for a security descriptor.

The following code example shows how to obtain a security descriptor.

```
' Obtain a security descriptor.
Dim x as IADs
Dim sd as IADsSecurityDescriptor
Dim acl as IADsAccessControlList

Set x = GetObject("LDAP://DC=Fabrikam, DC=com")
Set sd = x.Get("nTSecurityDescriptor")

Debug.Print sd.Control
Debug.Print sd.Group
Debug.Print sd.Owner
Debug.Print sd.Revision

Set acl = sd.DiscretionaryAcl
Set sacl = sd.SystemAcl
```

See Also

Syntaxes for Active Directory Attributes, Choosing a Syntax, How to Specify Comparison Values

Last updated: December 2004 | What did you think of this topic? | Order a Platform SDK CD