ПРОГРАММНОЕ УПРАВЛЕНИЕ ADSI: LDAP

В предыдущих статьях [1, 2] были рассмотрены теоретические аспекты построения Active Directory и проведен обзор доступных провайдеров, с помощью которых можно программно управлять Active Directory, а также описаны основы программирования одного из провайдеров – WinNT. Данный материал содержит основы программирования провайдера LDAP, объектная модель которого рассмотрена на примере стандартных утилит, созданных компанией Microsoft.

ИВАН КОРОБКО

Объектная модель провайдера LDAP

Для программного управления Active Directory с помощью провайдера LDAP необходимо использовать его объектную модель. Объектная модель представляет собой совокупность объектов, которые взаимосвязаны друг с другом и образуют между собой иерархическую структуру. Каждый из этих объектов имеет набор свойств, характерных исключительно для объектов данного типа. Существует несколько типов (идентификаторов) объектов: CN, DC, OU. Расшифровка и назначение каждого объекта см. в таблице 1.

Таблица 1

Сокращение	Описание
DC (Domain Component)	Метка доменного имени.
OU (Organization Unit)	Подразделение – организационная единица.
CN (Common Name)	Идентификатор пользователя.

Имена LDAP URL

Имена LDAP URL (см. RFC 1779, RFC 2247) построены на основе протокола X.500 и используются для связывания с объектами. Идентификаторы объектов DC, OU, CN образуют полное составное имя (Distinguished Name, DN), а имя самого объекта – относительное составное имя (Relative Distinguished Name, RDN). Полное составное имя объекта включает в себя имя объекта и всех его родителей, начиная с корня домена (см. рис. 1).

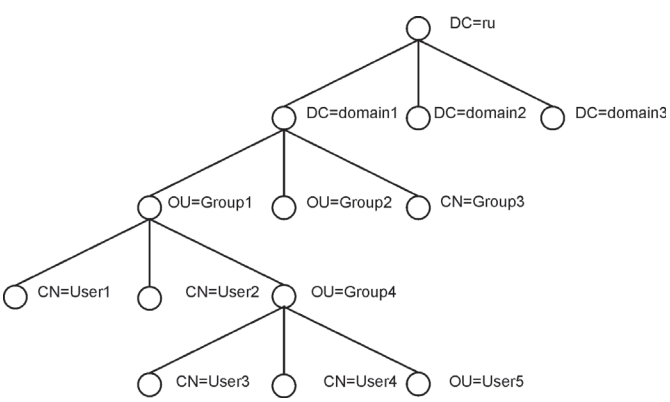

Рисунок 1

Существуют две формы доступа к ADSI: развернутая и сокращенная. Рассмотрим принципы построения путей к ресурсу двумя способами на примере домена domain.com (см. рис. 1).

Развернутая форма

При использовании этого вида формы строка связывания начинается с описания верхнего элемента структуры. Затем происходит переход вниз по иерархии. Важно помнить, что при написании пути к объекту необходимо исключать пробелы.

В качестве шаблона может служить выражение вида:

```
Set obj = GetObject ("LDAP://DC=Domain_name1/ ↵
    DC=Domain_name2/DC=Domain_name3/OU=OU_Name_Level1/ ↵
    OU=OU_Name_Level2…/OU=OU_Name_Levelμ/CN=CN_Name")
```

где DC=Domain_name1/DC=Domain_name2/DC=Domain_name3 образуют полное имя контроллера домена, элементы OU=OU_Name_Level1/OU=OU_Name_Level2…/OU=OU_Name_Levelμ представляют собой вложенные друг в друга элементы. В развернутой форме доступа объект CN является «дном колодца» в иерархии.

Пример: запрос к объекту CN=User3 с помощью развернутой формы доступа выглядит следующим образом (см. рис. 1):

```
Set obj = GetObject ("LDAP://DC=RU/DC=Domain1/OU=Group1/ ↵
    OU=Group4/CN=User3")
```

Сокращенная форма

Форма характеризуется тем, что строка запроса строится в соответствии с обратной иерархией структуры организации. Шаблон выглядит следующим образом:

```
Set obj=GetObject("LDAP://CN=CN_Name,OU=OU_Name_Levelμ…, ↵
    OU=OU_Name_Level2,OU=OU_Name_Level1/DC=…")
```

Соответственно запрос к объекту CN=User3 с помощью сокращенной формы доступа выглядит следующим образом:

```
Set obj=GetObject("LDAP://CN=User3,OU=Group4, ↵
    OU=Group3,DC=Domain1,dc=RU")
```

Инструменты, обеспечивающие доступ к объектной модели каталога

Существует несколько программ, предназначенных для просмотра объектной модели каталога. Остановимся лишь на двух из них: Active Directory Viewer (Microsoft) и LDAP Browser 2.5.3 (Softerra).

Active Directory Viewer (Microsoft)

Active Directory Viewer (ADV) является графической утилитой, позволяющей выполнять операции чтения, модифицирования, осуществлять поиск в любых совместимых каталогах, таких как Active Directory, Exchange Server, Netscape Directory, Netware Directory.

Active Directory Viewer входит в состав SDK для Active Directory Services Interface, который можно бесплатно загрузить с сайта Microsoft: http://www.microsoft.com/ntserver/nts/downloads/other/adsi25.

После установки SDK for ADSI утилита Active Directory Viewer (AdsVw.exe) будет находиться в c:\Program Files\Microsoft\ADSI Resource Kit, Samples and Utilities\ADsVw\.

Программа работает в двух режимах: ObjectViewer и Query (см. рис. 2). Для просмотра объектной модели какого-либо провайдера необходимо использовать режим ObjectViewer. Режим Query используется для осуществления поиска объектов в выбранной объектной модели. В данной статье режим Query рассматриваться не будет.

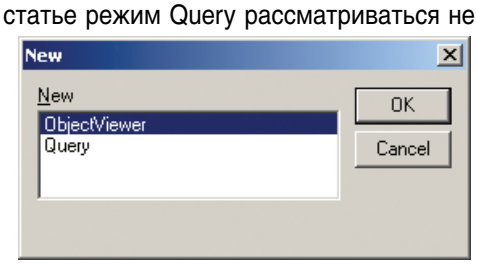

Рисунок 2

Просмотр и редактирование объектной модели программой ADV в режиме ObjectViewer

После выбора режима работы ObjectViewer появится диалоговое окно (см. рис. 3). Для получения доступа к каталогу необходимо указать путь к каталогу и параметры учетной записи, обладающей правами администратора (имя и пароль). Путь к каталогу должен быть построен в соответствии со следующим шаблоном:

```
<Provider_Name:>//<Server_Name>/<Full_Domain_Name>
```

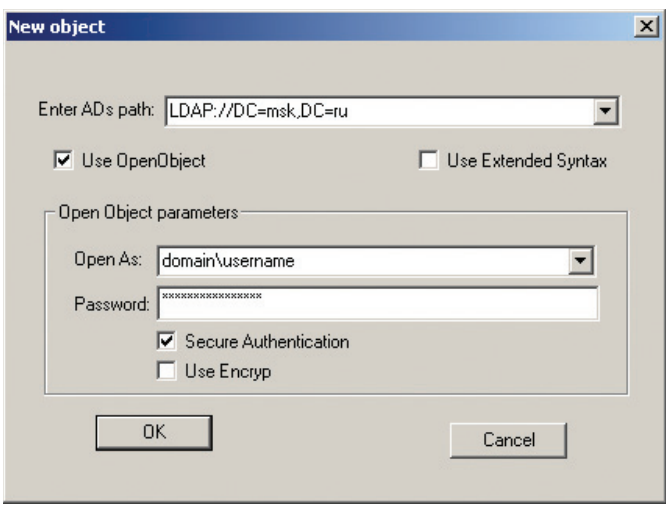

Рисунок 3

Обратите внимание на две особенности в этом шаблоне: в шаблоне не должно быть пробелов, «слэши» должны быть прямыми – «/». Невыполнение хотя бы одного из перечисленных условий приведет к ошибке в соединении с каталогом. Для доступа к серверу server домена domain.ru с помощью протокола LDAP используется следующий запрос:

```
LDAP://server/DC=domain,DC=ru;
```

После соединения с каталогом на экране будет отображена его иерархическая структура (см. рис. 4). В левой части экрана отображается иерархическая структура каталога. В правой части отображаются характеристики объекта, на котором установлен курсор. Список свойств объекта и соответствующих им значений приведен в «Properties» и «Property Value».

С помощью кнопок «Change», «Clear», «Append», «Delete» можно изменять объектную модель каталога: изменять, удалять, добавлять поля в свойствах объектов.

LDAP Browser 2.5.3 (Softerra)

LDAP Browser 2.5.3 является бесплатной программой (http://www.ldapadministrator.com).

По своим возможностям программа превосходит Active Directory Viewer, в использовании LDAP Browser гораздо удобнее. В процессе создания соединения с каталогом могут быть заданы фильтры, параметры административ-

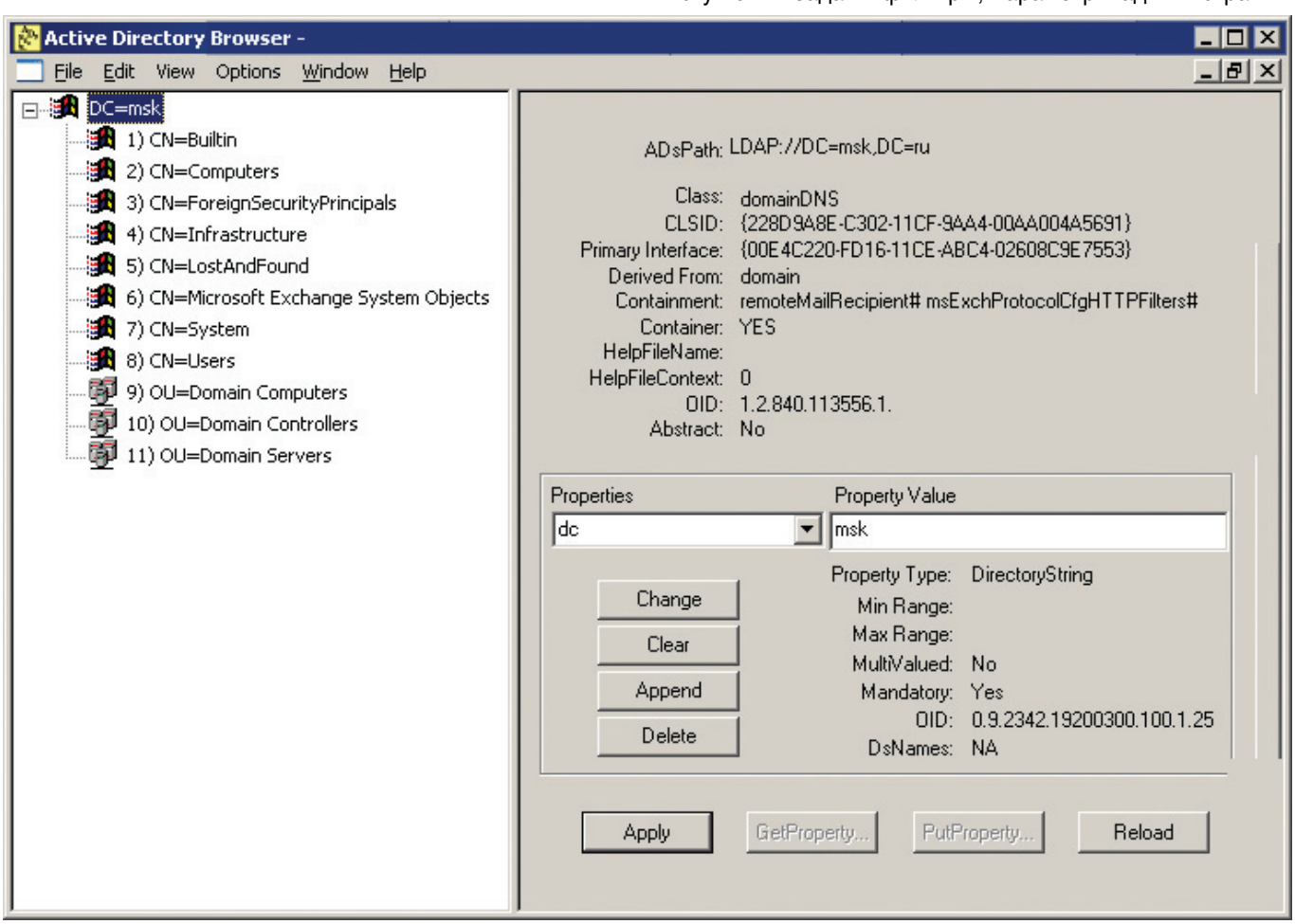

Рисунок 4

ной учетной записи, порт TCP, по которому имеет место соединение, и другие параметры. Общий вид программы приведен на рис. 5.

Различия в функционале провайдеров LDAP и WinNT

Об одном из отличий речь велась в предыдущей статье [2] – провайдер LDAP (Lightweight Directory Access Protocol) рассматривает принтер как сетевое устройство, в то время как провайдер WinNT рассматривает принтер исключительно как локальное устройство. Использование обоих провайдеров при работе с принтерами позволяет полностью управлять принтерами. Наглядный пример управления принтерами домена рассмотрен в статье «Управление сетевыми принтерами домена» [3].

Вторым принципиальным отличием провайдеров являются расширенные возможности поиска провайдера LDAP. Используя провайдер WinNT, можно было осуществлять поиск, пользуясь фильтром, который позволял осуществлять поиск всех объектов, принадлежащих к одному из классов – computer, user, service и др. Провайдер позволяет искать объект, при этом не обязательно указывать класс, к которому относится объект. Найдя объект, осуществляем чтение его свойств, включая местоположение объекта в AD, класс, к которому относится объект, и другие параметры.

Поиск объектов осуществляется с помощью OLE Distributed Query (DB) интерфейса, который вызывается прямо из интерфейса службы активного каталога (Active Directory Service Interface – ADSI). Поиск осуществляется на основании запроса и его параметров. В качестве параметров запроса могут быть: уровень поиска, диапазон поиска, ограничение по размеру, сортировка и т. д. Форма запроса (Distributed Query) заимствована из Microsoft SQL Server.

Запрос строится по шаблону:

```
SELECT поле1,поле2,полеn FROM путь ↵
    WHERE objectClass="тип_объекта"
```

где путь – путь к интересующему объекту AD в формате LDAP URL.

На практике поиск объектов осуществляется следующим образом:

```
Пример 1:

Set objNameSpace = GetObject("WinNT:")
  For Each Domain in objNameSpace
    DomainName=Domain.Name
  Next

Set objConnection = CreateObject("ADODB.Connection")
Set objCommand = CreateObject("ADODB.Command")
objConnection.Provider = "ADsDSOObject"
objConnection.Open "Active Directory Provider"
Set objCommand.ActiveConnection = objConnection
objCommand.CommandText = "SELECT printerName, serverName ↵
    FROM " _ & " 'LDAP://"& DomainName & "' ↵
    WHERE objectClass='printQueue'"
objCommand.Properties("Cache Results") = False
Set objRecordSet = objCommand.Execute
objRecordSet.MoveFirst

Do Until objRecordSet.EOF
  temp=temp & "Printer Name: " ↵
    & objRecordSet.Fields("printerName").Value ↵
    & " Server Name: " ↵
    & objRecordSet.Fields("serverName").Value & chr(13)
  objRecordSet.MoveNext
Loop

wscript.echo temp
```

В приведенном примере осуществляется поиск всех зарегистрированных в AD-принтеров. У найденных прин-

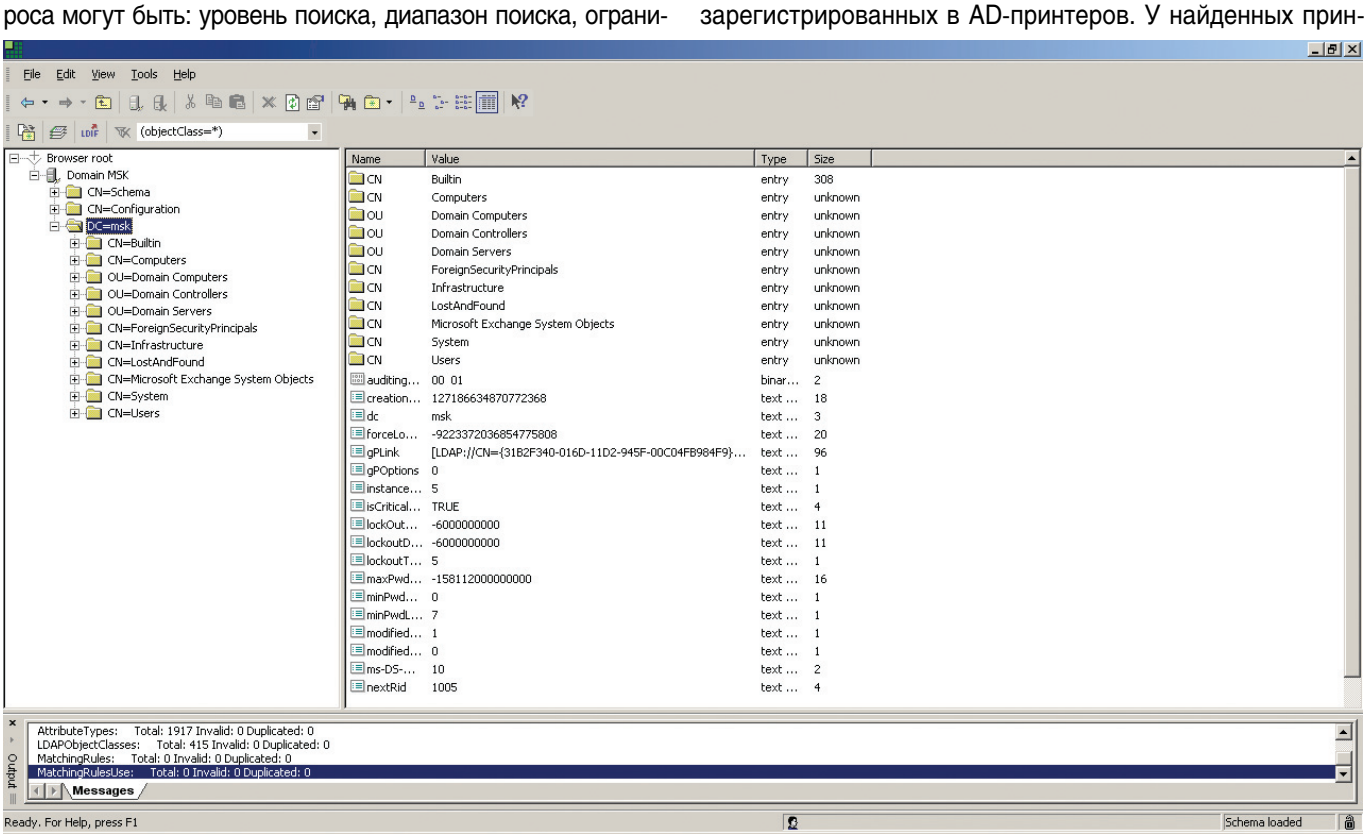

Рисунок 5

теров происходит чтение двух полей: название принтера и название сервера печати.

Поиск объектов с помощью провайдера LDAP осуществляется по следующему шаблону:

- устанавливается соединение с Active Directory Provider через ADODB;
- составляется запрос, на основе которого будет осуществляться поиск;
- осуществляется поиск по заданным критериям.

В том случае если искомые объекты найдены, то осуществляется чтение указанных в запросе полей. Результат выводится на экран. В качестве объектов может быть строка или массив.

Следует отметить, что вместо названия свойства, которое необходимо прочитать, можно указать порядковый номер поля, под которым оно обозначено в запросе. Поля отсчитываются с нуля.

Таким образом, основываясь на приведенном примере, вместо objRecordSet.Fields(«serverName»).Value можно записать objRecordSet.Fields(1).Value.

Третье отличие – это управление коммерческими продуктами, поддерживающими LDAP. Множество коммерческих продуктов использует каталог LDAP для хранения информации. Используя ADSI, можно управлять коммерческими продуктами, в число которых входят: Microsoft Windows 2000, 2003; Microsoft Exchange 5.5, 2000, 2003; Microsoft Site Server 3.0 + SP2; Netscape Directory Server и др. Кроме того, компании Cisco и Microsoft предложили стандарт сети на основе каталога, в котором описывается интеграция сетевых устройств в каталоге LDAP.

В настоящей статье будет рассмотрен вопрос, касающийся управления Microsoft Windows 200x, а именно Active Directory. Рассмотрим следующие вопросы, касающиеся управления AD через провайдер LDAP: просмотр атрибутов записи от анонимной и конкретной учетной записи; изменение атрибутов учетной записи; создание и удаление учетной записи.

Просмотр атрибутов записи от анонимной и конкретной учетной записи

Просмотр атрибутов объектов осуществляется с помощью функции Get(), вызову которой предшествует вызов функции GetObject(). Для получения анонимного доступа к объектам необходимо указать путь к объекту, начиная с контроллера домена. В приведенном примере читается идентификационный номер учетной записи User3 (см. рис. 2) – UserID, которому соответствует поле uid. Объектная модель провайдера LDAP будет рассмотрена позже. Необходимо отметить, что свойства имеют все типы объектов – OU, CN и другие.

```
Пример 2:

  Set obj=GetObject("LDAP://CN=User3, OU=Group1, ↵
   OU=Users, o=domain.ru")
    For Each U_obj In obj
      wscript.echo  "UserUID: " & U_obj.Get("uid")
    Next
```

В том случае если доступ анонимным пользователям к записям блокирован, то необходимо читать свойства объектов от имени учетной записи, которая имеет права на чтение свойств. Пусть пользователь User1 имеет право на чтение всех полей объектов. Пароль пользователя User1 – 1234567. Осуществим чтение идентификационного номера учетной записи User3 от имени User1. Чтение параметров от имени другого пользователя осуществляется с помощью функции OpenDSObject():

```
Пример 3:

  Set PreObj= GetObject("LDAP:")
Set obj= PreObj.OpenDSObject("LDAP://CN=User3, OU=Group1, ↵
    OU=Users, o=domain.ru", "CN=User1","1234567",0)
     For Each U_obj In obj
       wscript.echo  "UserUID: " & U_obj.Get("uid")
     Next
```

Изменение атрибутов учетной записи

Модификация атрибутов учетной записи осуществляется с помощью функции Put() и метода SetInfo, служащего для сохранения внесенных изменений в Active Directory. В приведенном ниже примере атрибут учетной записи User3 – Canonical Name (CN) будет изменен с User3 на User4.

```
Пример 4:

  Set PreObj= GetObject("LDAP:")
Set obj= PreObj.OpenDSObject("LDAP://CN=User3, OU=Group1, ↵
    OU=Users, o=domain.ru", "CN=User1","1234567",0)
  Set U_obj=obj.GetObject("InetOrgPerson","CN=User3")
  U_obj.Put "CN","User4"
  U_obj.SetInfo
  MsgBox "Параметр CN изменен."
```

InetOrgPerson – тип учетной записи.

Создание и удаление учетной записи

Для создания учетной записи в Active Directory необходимо задать несколько обязательных параметров, относящихся к учетной записи и ее родительскому контейнеру:

- создание учетной записи должно выполняться пользователем, имеющим административные привилегии;
- путь к родительскому контейнеру, в котором необходимо создать учетную запись;
- класс создаваемого объекта;
- соответствующие свойства создаваемого класса объекта записи.

```
Пример 5:

  Set PreObj= GetObject("LDAP:")
Set obj= PreObj.OpenDSObject("LDAP:// OU=Group1, OU=Users, ↵
    o=domain.ru", "CN=User1","1234567",0)
  Set U_obj=obj.Create("InetOrgPerson","CN=User3")
  ClassArray=Array("InetOrgPerson","person","top","organizationPerson")
  U_obj.Put "objectClass", ClassArray
  U_obj.Put "cn", "User_Name_3"
  U_obj.Put "sn", "Second_Name_3"
  U_obj.SetInfo
  MsgBox "Учетная запись создана."
```

Для удаления учетных записей используется метод Delete («InetOrgPerson», object_name).

Заключение

На практике управление Active Directory преимущественно осуществляется с помощью провайдера WinNT или в совокупности WinNT с LDAP. В «чистом» виде программирование LDAP используется очень редко. Основная причина заключается в том, что для доступа к любому объекту с помощью провайдера LDAP необходимо знать полный путь к этому объекту. Эта проблема легко решается: происходит осуществление поиска объекта, затем чтение его свойств. Приведем пример чтения поля FullName для пользователя USER1 с помощью провайдеров LDAP и WinNT.

```
Пример 6 а) - WinNT:

Set obj=GetObject("WinNT:")
  For Each str In obj
  DomainName=str.Name
  Next
Set UserName="Value"
Set element=GetObject("WinNT://" & DomainName & "/ USER1")
Msgbox "FullName: "+ cstr(element.FullName)

Пример 6 б) - LDAP:

set rootDSE_ = GetObject("LDAP://RootDSE")
DomainName = rootDSE_.Get("defaultNamingContext")

UserLogonName="USER1"
Const ADS_SCOPE_SUBTREE = 2
Set objConnection = CreateObject("ADODB.Connection")
Set objCommand = CreateObject("ADODB.Command")
objConnection.Provider = "ADsDSOObject"
objConnection.Open "Active Directory Provider"
Set objCommand.ActiveConnection = objConnection
objCommand.CommandText = "SELECT name, sAMAccoutName ↵
    FROM " _ & " 'LDAP://"& DomainName & "' ↵
    WHERE objectClass='users'"
objCommand.Properties("Searchscope") = ADS_SCOPE_SUBTREE
objCommand.Properties("Cache Results") = False
Set objRecordSet = objCommand.Execute
objRecordSet.MoveFirst
Do Until objRecordSet.EOF
If objRecordSet.Fields("sAMAccoutName").Value=UserLogonName
msgbox   "FullName: "+ objRecordSet.Fields("name").Value
end if
Loop
```

Как видно, листинг примера 6 б) в несколько раз больше, чем листинг примера 6 а). За счет того, что в примере 6 б) «просматривается» весь массив пользователей, сценарий будет отрабатываться в несколько раз медленнее, чем сценарий 6 а). Скорость выполнения сценария напрямую зависит от количества объектов в просматриваемом массиве. Очевидно, что чем больше размер массива, тем медленнее будет работать скрипт.

Многократное использование данного механизма в одном скрипте дополнительно уменьшит скорость выполнения скрипта и увеличивает его размер по сравнению с аналогичным скриптом, в котором доступ к объектам осуществляется с помощью провайдера WinNT. Однако отказаться от доступа к AD с помощью провайдера LDAP невозможно, поскольку существует много функций, которые реализованы только в провайдере LDAP. Оптимальным вариантом является совместное использование провайдеров LDAP и WinNT. Ярким примером является чтение свойств сетевого принтера: с точки зрения провайдера WinNT принтер – локальное устройство, с точки зрения LDAP – сетевое.

Литература:

1. Коробко И. Active Directory – теория построения. – // Журнал «Системный администратор», №1(14), январь 2004 г. – 90-94с.
2. Коробко И. Программное управление ADSI: WinNT. – // Журнал «Системный администратор», №2(15), февраль 2004 г. – 66-74с.
3. Коробко И. Управление сетевыми принтерами домена. – // Журнал «Системный администратор», №10(11), октябрь 2003 г. – 38-46с.