

# 通过 ASP 和 ADSI 维护系统帐号信息

## Maintaining System Account Information by ASP and ADSI

胡定奇 刘瑞芳 (深圳市宝安区中等专科学校 518000)



**摘要:** 本文介绍了一种新的维护系统帐号的方法: 通过使用 ASP 技术, 以及 ADSI 显露的 COM 接口, 采用 WWW 方式更新系统帐号信息。这种方法将维护帐号的任务分散给了各个用户, 极大的减轻了管理员的负担, 而且还增强了系统帐号信息的准确性, 有效性。

**关键词:** 活动服务页面 活动目录服务接口 活动目录

### 1 引言

随着计算机在各行各业的广泛使用, 系统管理员面临的帐号管理压力越来越大。因为计算机帐号随着公司员工的加入和离去也必须做相应的增加或删除, 而且帐号信息保存的不仅仅是用户名和口令, 许多诸如用户联系方式, 办公地点, 职位等信息也是非常关键的。同时由于社会生活节奏的加快, 员工的调动也相当频繁, 帐号信息必须及时反应这些变化, 保证信息的有效性。如果这些全部更新必须全部由系统管理员来处理的话, 系统管理员会有点力不从心, 特别是对规模比较大的组织或者公司来说, 如果想保证计算机帐号信息的及时更新, 同时又不增加系统管理员的负担, 最好的方式是将更新帐号信息的任务分散给用户本人, 而管理员负责检查帐号信息的有效性。

随着 WIN2000 SERVER 逐步成为了各公司服务器的主要操作系统, WIN2000 内置的活动目录服务接口 (ADSI) 为系统管理员简化帐号管理提供技术条件。本文将介绍一种利用 ASP 和 ADSI 来实现通过 WWW 方式维护系统帐号信息的方法, 这种方法可以做到由用户自己管理维护自己的帐号信息, 减轻了系统管理员的负担, 保证了信息的准确性, 有效性。本文后面将先介绍 ASP 基本原理, 然后对 ADSI 进行简要的介绍, 接着介绍通过 ASP 和 ADSI 维护系统帐号信息的具体实现。最后将对本文做简要的总结。

### 2 活动服务页面 (ASP) 原理

Active Server Pages (ASP) 是一个“服务器端脚本编写”环境, 可用于创建和运行动态的交互式 Web 服务器应用程序。使用 ASP, 可以合并 HTML 页、脚本命令以及 COM 组件创建交互式 Web 页和功能强大的 Web 应用程序, 它们都可以轻松地开发并修改。当浏览器向 Web 服务器请求 .asp 文件时, 服务器端脚本便开始运行。Web 服务器于是调用 ASP, 用它从头至尾处理所请求的文件, 执行脚本命令, 并将 Web 页发送到浏览器。

因为脚本运行于服务器而不是客户端, 所以生成发送到浏览器的 HTML 页等工作便由 Web 服务器负责。服务器端脚本无法被预先复制, 因为返回到浏览器的只是脚本的运行结果。用户无法得知创建其所查看的页面使用的脚本命令, 保证网站的安全性。

现在可以将商业逻辑脚本过程转换为可重复使用的 COM 组件, 该组件可用于 Web 应用程序和其他组件对象模型 (COM) 适用的程序。组件提供了在脚本中用来执行专门任务的功能, 如执行金融交易或验证数据等。ASP 还提供有一套用以大大增强脚本性能的基本组件。组件是包含动态链接库 (.dll) 或可执行 (.exe) 文件的可执行代码。组件可提供一个或多个“对象”, 这是一种自包含的代码单元, 可以在组件中执行指定的功能。每个对象都具有方法 (编程过程) 和属性 (行为属性)。要使用由组件提供的对象, 可以创建对象实例并为新的实例分配变量名称。使用 ASP 的 Server.

CreateObject 方法或在 HTML 中使用 <OBJECT> 标记可创建对象实例。使用脚本语言变量赋值语句可为对象实例指定名称。创建对象实例时，必须提供其注册名 (PROGID)。

本文描述的维护系统帐号信息的方法，将利用 ASP 使用 COM 组件来和 ADSI 数据库联系，达到提取用户帐号信息和修改用户帐号信息的目的。

### 3 活动目录服务接口 (ADSI) 介绍

随着网络技术的迅速发展，出现了当今最为流行的分布式网络环境。识别和定位网络资源是分布式网络环境中的难题之一。目录服务的出现解决了这个难题，它提供了定位和识别用户及系统中可用资源的方法。由于各个网络资源的差异，在大多数网络中使用了许多不同的目录，例如网络操作系统、电子邮件系统都有它们自己的目录。当一个网络分散拥有多个目录时，管理员要面对管理多个目录的复杂性而无法配置大范围的目录。终端用户要面对多个登录和跨越多个目录的各种信息接口，开发人员难以选用网络中已有的目录服务。ADSI 从分布式计算环境的不同网络提供者中抽取了目录服务能力，为管理和使用多种网络及多种目录环境中的网络资源提供了单一的、一致的、开放的接口集。管理员和开发者只需要处理单一的目录服务接口集，而不管安装了哪些目录服务。他们能够利用 ADSI 服务去枚举和管理目录服务中的资源而不管这些资源是在哪一个网络环境中。网络管理员能够利用 ADSI 和诸如 Visual Basic 及 Java 之类兼容 Automation 的语言自动完成“添加用户、管理打印机、设置网络资源的权限”等常见的管理任务。终端用户能够在单一登录的条件下定位整个分布式计算环境中的资源。开发者能够将 ADSI 用于支持目录的产品和应用，并且它们能成功地运行在多种网络与目录的环境中。ADSI 为系统管理和开发提供了许多优点：开放任何一个目录提供者能够实现一个 ADSI 提供者，用户获得了选择目录服务的自由而不用牺牲可管理性。

ADSI 使得管理型应用独立于特定开发者的目录服务，在不需理解开发者的目录 API 的条件下，就可以开发管理型应用及其他支持目录的应用。ADSI 支持 Visual Basic、VBScript、Perl 和 C/C++ 等语言编程。任何兼容 Automation 的语言都可以用于开发目录服务的应用程序。目录提供者、ISV 和最终用户可以用新对象和函数集扩展 ADSI，而且 ADSI 提供了一个 OLE DB 接口。这些优点对于支持多种目录管理和开发是十分重要的，也给 ADSI 编程提供了方便。今天，目录服务已经广泛应用到分布式网络环境中，ADSI 的出现为管理各种目录服务提供了统一的接口集，它必将大大改善网络中目录的管理和开发。但目前大家对 ADSI 的了解还不很深入，本

文将从利用 ADSI 维护帐号信息入手介绍 ADSI，但需要声明的是 ADSI 功能远不止这些。

### 4 通过 ASP 和 ADSI 维护系统帐号信息的实现

ADSI 向用户提供丰富的编程接口，给程序员和开发者极大的便利。通过 ADSI 中的 IADsUser 接口可以全面而方便的维护系统帐号信息。下面就演示一下如何综合 ASP 和 ADSI 来实现系统帐号信息的维护。

首先讲 ADSI 和 LDAP 的关系。我们的计算机帐号信息保存在活动目录数据库中，ADSI 定义了一个目录服务模型和一套 COM 接口，通过这些接口，程序员可以开发访问目录服务的应用程序。而 ADSI 则主要是通过 LDAP 目录服务提供商和活动目录 (Active Directory) 联系的，事实上通过 ADSI 还可以访问 Novell 目录服务。

其次，需要说明的是 Adspath，在活动目录 (Active Directory) 中，所有的对象都是通过 Adspath 来表示的。比如：在服务器中注册的一个用户，一个用户组或者是一台打印机或者扫描仪。比如“LDAP://CNSSL/CN=TopHat,DC=DEV,DC=MSFT,DC=COM”表示一个注册用户。

ADSI 使用 IADsUser 接口来表示在网络注册中一个终端用户帐号，开发者可以通过调用其中的函数来操纵和用户帐号相关的信息。这些信息包括：用户名、电话号码、头衔、办公地点、传真号码等等。这个接口也同时支持设置用户的组属关系。通过这个接口还可以设置或者修改用户口令。通过 IADsUser 接口可以更新 50 种左右的和帐号相关的信息，在这里的例子中将只演示其中的几种重要的常用信息，具体可参考 MSDN 帮助。

在获取指定帐号的 IADsPath 后，就可以在绑定到这个帐号后，通过 IADsUser 来维护帐号信息。

```
Set ?objUser = GetObject("WinNT://DCD-ELS/Administrator"),  
objUser.EmailAddress=" relyfang@sohu.com"  
objUser.FirstName=" Liu"  
objUser.SetInfo
```

注意：在修改了帐号的信息之后，必须使用 SetInfo 函数来确保所有修改在服务器上保存了，因为我们前面做的修改都是保存在本地 CACHE 中的，只有在执行了 SetInfo 之后，这些修改才会起作用。

另外，请注意出于安全性的考虑，IADsUser 接口并没有包含帐号的口令信息，这样就不可能直接检索 IADsUser 各项属性来获取用户的口令。

下面给出具体的实现代码，第一个是直接提供给用户的 EDITINFO.HTML，它显示一个表单，让用户输入自己的更新信息。然后这个表单的结果会提交给 UPDATE.ASP，ASP 使用前面描述的方法来更新帐号信息。

EDITINFO.HTML:

```
<html>
<head> <title>修改用户信息</title> </head>
<body>
<p>请及时更新你的帐号信息, 否则你的朋友或同事无法及时和你联系。</p>
<form method="POST" action="update.asp">
  <p>用户名<input type="text" name="UserName" size="20" ></p>
  <p>口令<input type="password" name="Password" size="20" ></p>
  <p>电子邮件地址<input type="text" name="Email" size="20" ></p>
  <p>个人主页地址<input type="text" name="HomePage" size="20" ></p>
  <p>办公地址<input type="text" name="Office" size="20" ></p>
  <p>传真号码<input type="text" name="Fax" size="20" ></p>
  <p>所属部门<input type="text" name="Department" size="20" ></p>
  <p>
    <input type="submit" value=" 开始更新 " name="B1">
      <input type="reset" value=" 重新再填 " name="B2"></p>
  </p>
</form>
</body>
</html>
```

用户通过浏览 EDITINFO.HTML 页面, 输入自己的信息然后提交, 所有信息将由 UPDATE.ASP 来处理。它将先创建一个 ADSI 对象, 然后通过调用前面提到的各种接口函数来实现用户信息的更改。

```
<html>
<head> <title>感谢您及时更新帐号信息</title> </head>
<body>
<%
Dim objUser
Dim hReturn
Set objUser = GetObject("LDAP://CNSSL.DHS.ORG/CN=" & Request
("UserName")
& ",OU=Floor4,OU=Cnssl Lab,DC=cnssl,DC=dhs,DC=org ")
hReturn = objUser.ChangePassword(Request("Password"), Request
("Password"))
if (bReturn = 0) then
```

```
objUser.Email = Request("Email")
objUser.HomePage = Request("HomePage")
objUser.OfficeLocations = Request("Office")
objUser.FaxNumber = Request("Fax")
objUser.Department = Request("Department")
objUser.SetInfo
else
  Response.Write("<p>你的口令正确,无法为您更新帐号信息, 请确认后
再试。</p>")
endif
%>
<p>感谢您及时更新帐号信息! 祝您工作愉快! </p>
<p>
</body>
</html>
```

## 5 结束语

随着计算机用户的增加, 以及用户信息变动性的增大, 系统管理员面临的帐号信息维护压力越来越大。本文提出一种新的方法来维护帐号信息, 它通过 WWW 方式, 可以让用户自己维护自己的帐号信息, 这样别人就可以及时了解自己的更新信息, 同时也不会增加系统管理员额外的负担; 这种方法通过 ASP 使用 ADSI 中的各种接口特别是 IADsUser 接口, 实现了远程更改帐号信息。本文最后还给出了具体的实现代码。但需指出的是, ADSI 在如今分布式网络环境中, 其作用远不止维护帐号这么简单, 它还有很多功能待我们去开发利用。 ■

## 参 考 文 献

- 1 Microsoft ADSI Tutorial. Platform SDK, MSDN, 2001.
- 2 Ken Spencer. Changing Passwords over the Web. Windows 2000 Magazine. January.
- 3 郭健, 许占文. Windows 2000 活动目录结构. 辽宁大学学报, 2001 年, 第 28 卷, 第 3 期.
- 4 闫瑞锋, 闫瑞华. NT 网域用户帐号的远程管理. 佳木斯大学学报, 2001 年, 第 19 卷, 第 3 期.