

用于 Intranet 分布计算的 ActiveX 构件特性

李威 杨乔林 (中科院计算所 100080)

摘要: ActiveX 构件是微软分布计算、构造“三级服务体系结构”的构件软件开发模型程序单元。本文全面介绍了 ActiveX 构件的概念、种类、运行方式以及在 Internet/Intranet 环境中的引用和部署方法。本文也是“Intranet 中基于软件构件的企业分布计算”[2]一文的进一步说明,对其中使用的 ActiveX 技术作了深入的介绍。

1. 引言

ActiveX 既是微软的一个商标,同时也代表了基于微软的构件对象模型 COM(Component Object Model)的松散定义的技术集合。ActiveX 是以前称为 OLE 自动化技术的扩展,至于 ActiveX 到底包含哪些技术,目前处于不断的变化之中,对其发展也很难预测,但有一点是确定的,那就是 ActiveX 所包含的所有技术都将基于 COM。一个 ActiveX 构件是一个遵循 ActiveX 规范(Specification)的一个可执行代码单元(Unit),诸如一个 .EXE、.DLL 或者 .OCX 文件。ActiveX 技术允许程序员开发、组装、部署和共享这些可重用的 ActiveX 构件以提供特定的服务。由于 ActiveX 构件基于 COM 模型,所以 ActiveX 构件可以用各种语言来书写(如 C++, JAVA 和 VB 等);可以以 DLL 或 EXE 方式来实现以及用一致的(Consistent)的面向对象的方式相互交互。

2. ActiveX 构件对象的运行方式

(In-Process 方式:如果 ActiveX 构件与调用该构件的客户应用在同一网络结点,那么构件对象装入调用该构件的客户应用进程中运行(如图 1(a)所示),这使得客户应用对构件接口的访问和对其自身包含的对象的访问几乎没有区别,所以有较高的访问效率;如果 ActiveX 构件位于与调用该构件的客户应用不同的远程服务器结点,那么构件对象装入远程服务器结点的宿主进程(Host

Process)中运行(如图 1(b)所示)。尽管远程客户应用是从宿主进程间接得到对构件的引用(Reference),但是一旦得到了这个引用,那么以后对构件的引用都将是直接的,所以构件的访问性能可以和具有独立进程的构件访问性能相比。由于这种方式不要求完全装载构件对象进程便可以运行,那么当系统有许多构件运行时,可以降低远程连接时间及操作系统资源的消耗。

(Out-of-Process 方式:如果 ActiveX 构件与调用该构件的客户应用在同一网络结点,那么构件对象与客户应用在同一结点的不同进程中运行(如图 1(c)所示)。在这种方式下对构件的访问性能虽然不如 In-Process 方式,但是构件进程的独立性使得它可以被多个客户应用所并发共享,降低系统资源消耗。另外,如果构件用来完成相对长时间的任务,它允许客户应用采用异步调用方式,而不必因等待任务的完成而阻塞;如果 ActiveX 构件位于与调用该构件的客户应用不同的远程服务器结点,那么构件对象与客户应用在不同结点的不同进程中运行(如图 1(d)所示)。虽然构件的访问性能较低,但却可以利用远程服务器的强大功能,从整体性能上看还是较好的,且构件的远程集中管理大大降低了维护与更新的费用,并可为多个客户机上的多个客户进程所共享。这种远程运行方式是降低计算机系统总拥有成本的有效方法^[3]。

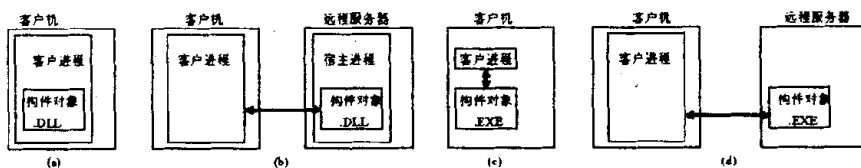


图 1 ActiveX 构件的运行方式

3. ActiveX 构件的种类及其引用方式

·ActiveX Document: 在 Visual Basic 中构造一个 ActiveX Document 与构造一个 VB 应用没有太大的差别,但它却与通常的应用特性完全不同。它既具有文档的灵活性,可被不同的容器软件打开,如微软的 Internet Explorer 或 Office Binder,又具有应用程序的功能性,所以它是动态的文档。ActiveX Document 由文档文件(.vbd)和服务文件(.dll 或 .exe)两部分组成。点取文档界面上嵌入

```
<OBJECT -----> 指示浏览器下载、注册 ActiveX Document
  CLASSID="CLASSID:2F390484-1C7D-11D0-8990-00A0C90395F4" -----> ActiveX Document 的 GUID
  CODEBASE="ActXDocSam.cab#Version=1,0,0,0" -----> ActiveX Document 的压缩文件包
</OBJECT>
<SCRIPT LANGUAGE="VBScript"> -----> 用 VBScript 指示浏览器打开相应的 ActiveX Document
Sub Window_OnLoad
  Document.Open
  Document.Write "<FRAMESET>"
  Document.Write "<FRAME SRC="ActXDocSam.vbd">"
  Document.Write "</FRAMESET>"
  Document.Close
End Sub
</SCRIPT>
```

·ActiveX Control: ActiveX Control 是可以提供各种服务的标准视觉界面元素。ActiveX Control 的最常用的使用方式是:构造一个 ActiveX Control (.ocx 文件),将其注册于 Windows Registry,然后就可以在多种软件开发工具如 VB、VC++ 中,以可视化的方式引用。点取相应的

```
<OBJECT -----> 运行许可对象,浏览器用它来验证 ActiveX Control 的运行许可
  CLASSID="CLASSID:5220CB21-C88D-11CF-B347-00AA00A28331" ----->
  <PARAM NAME="LPKPath" VALUE="LPKfilename.LPK" -----> 运行许可包(Run-time License Package)文件
</OBJECT>
<OBJECT ID="MainCtrl" WIDTH=90 HEIGHT=50 指示浏览器下载、注册和实例化 ActiveX Control ActiveX Control 的引用 ID
  CLASSID="CLASSID:E1366928B-3A08-11D1-BB4A-0000B436EB37" -----> ActiveX Control 的 GUID
  CODEBASE="ActXCtrlSam.cab#Version=1,0,0,0" -----> ActiveX Control 的压缩文件包
</OBJECT>
```

的控制或者与容器软件菜单相融合的某一菜单项,便可以获得相应的服务。由于 ActiveX Document 支持超链对象等 Internet 特性,所以它主要用于嵌入 HTML 页面来构造 Internet/Intranet 应用。ActiveX Document 既可以以 In-Process 方式运行,也可以以 Out-of-Process 方式运行。一个在 HTML 页面上引用 ActiveX Document 对象的例子如下:

ActiveX Control,便可以获得其代码提供的相应的服务。ActiveX Control 的另一种使用方式是嵌入 HTML 页面,用来构造 Internet/Intranet 应用。ActiveX Control 只以 In-Process 方式运行。一个在 HTML 页面上引用 ActiveX Control 对象的例子如下:

·ActiveX DLL 和 ActiveX EXE: ActiveX DLL 和 ActiveX EXE 统称为 ActiveX 代码构件,它们以代码库(Code Library)的方式存在和引用。产生一个代码构件并将其注册于 Windows Registry,程序员便可以在其应用中用代码构件提供的类来产生对象,使用对象的属性,调用对象的方法和响应对象的事件。ActiveX DLL 和 ActiveX EXE 的主要区别是 ActiveX DLL 以 In-Process 方式运行,而 ActiveX EXE 以 Out-of-Process 方式运行。将代码构件嵌入 HTML 页面便可以在 Internet/Intranet

环境中获得代码构件提供的服务,一个在 HTML 页面上引用 ActiveX EXE 代码对象的例子如下:

4. ActiveX 构件在网络上的部署(Deployment)

ActiveX 构件在网络上的灵活部署方式主要是针对 ActiveX 的代码构件 ActiveX DLL 和 ActiveX EXE 而言。当构件和调用构件的客户用位于不同的网络结点时,主要是通过远程自动化(Remote Automation)和分布式 COM(Distributed COM)来提供构件访问的网络透明性,而无需对客户应用或构件代码做任何修改。

```

<FORM NAME="TESTFORM">
<INPUT NAME="QUERY" TYPE="BUTTON" VALUE="DB QUERY">
</FORM>
<OBJECT ID="QueryClass" -----> 指示浏览器下载、注册和实例化 ActiveX EXE ActiveX EXE 的引用 ID
    CLASSID="CLASSID:DD2C1BDD-4242-11D1-9DC4-0000B436EB37" -----> ActiveX EXE 的 GUID
    CODEBASE="DBQuery.cab" #Version=1,0,0,0" -----> ActiveX EXE 的压缩文件包
</OBJECT>
<SCRIPT LANGUAGE="VBScript"> -----> 用 VBScript 引用 ActiveX EXE 对象
DIM ob
Sub QUERY_OnClick
    Set ob=QueryClass
    ob.QueryMethod -----> 调用对象提供的方法
End Sub
</SCRIPT>

```

·远程自动化:如图 2 所示,远程自动化的核心是具有使用 Windows NT 远程过程调用 RPC(Remote Procedure Call)协议能力而进行网络通信的远程自动化管理程序(Remote Automation Manager)。运行于客户机上的远程自动化管理程序代理(Proxy)对象,将客户应用的构件调用要求包装成网络传输信息并发送给运行于远程服务器上的远程自动化管理程序存根对象(Stub),存根对象将包装打开并将其发送给构件进程的存根对象;调用的返回值传送过程与之正好相反。Windows Registry 中的设置确定了构件之间的通信是通过 COM 提供的标准的代理/存根对象还是使用远程自动化代理/存根对象。这种设置完全是在 ActiveX 构件安装时确定的,而运行时是完全网络透明的。作为服务器的 ActiveX 构件在安装注册时,只要被指定是通过远程自动化方式引用即可;而在客户机的客户应用安装时要将其引用的远程 ActiveX 构件在本地 Windows Registry 设置为远程自动化对象,并同时指出远程 ActiveX 构件的网络地址、构件间通信所使用的网络协议及授权级别(Authentication Level)

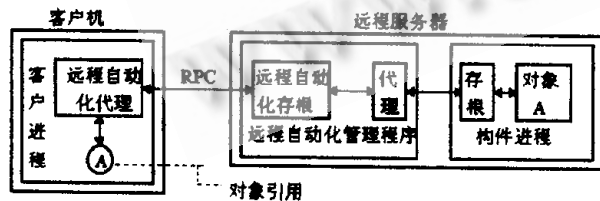


图 2 远程自动化构件对象引用

·分布式 COM:DCOM 扩展了 COM,DCOM 的代理(Proxy)/存根(Stub)对象本身具有利用 RPC 进行网络通

信的能力,而不需要象远程自动化管理程序这样的中间环节的存在。客户应用将对远程构件的调用要求首先提供给内置于 Windows NT 的 COM 核心服务,由其创建被调构件的运行进程,并建立客户应用与被调构件之间的网络连接。随之以后对构件的引用便由 DCOM 的代理/存根对象来完成,如图 3 所示。ActiveX 构件间 DCOM 访问方式的确立是在构件的安装及注册时完成的,而在运行时是完全网络透明的。使用 DCOM 构件访问方式要将远程 ActiveX 构件在其服务器上注册为 DCOM 引用,而在客户机的客户应用安装时要将其引用的远程 ActiveX 构件在本地 Windows Registry 设置为 DCOM 对象,并同时指出远程 ActiveX 构件的网络地址即可。

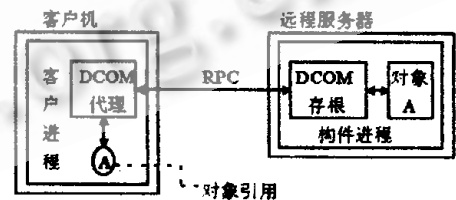


图 3 DCOM 构件对象访问

DCOM 是当前微软推荐的构件网络部署模型,而远程自动化则是为了向后兼容和 16 位平台应用。远程自动化的效率也不如 DCOM。ActiveX 构件在网络上的远程自动化和 DCOM 部署策略具有以下优点:

- 构件访问的网络透明性,从而为程序员提供了同一的程序设计界面;
- 构件是只有一份拷贝的共享构件,这就大大降低了对系统存储资源的消耗,同时也维护了代码的一致性;

·更新共享构件时,只要设定新老版本的二进制兼容性,系统中的所有客户应用可以使用这个新的版本而不必重新编译;

5. ActiveX 构件的安全性

从 Internet 上下载执行代码是很危险的,而 ActiveX 构件便是一种可以从 Internet 上下载、注册和执行的代码段。ActiveX 使用数字签名来保证安全性。微软的授权编码(Authenticode)技术允许你验证 ActiveX 构件的来源,进而确定它的可靠性和安全性。授权编码技术基于微软的代码签名建议(Code - signing Proposal),授权编码使用 X.509 v3 加密标准(Cryptograph Certificates)及 PKCS#7 和 #10 签名标准。数字签名使用公开(Public)和私有(Private)一对密钥。只有私有密钥的所有者唯一知道这个密钥,而公开密钥是全球公开的。私有密钥用来产生签名,公开密钥用来验证这个签名。数字签名是 1024 位(Bit)的,所以是难于破译的。

6. ActiveX 平台

ActiveX 平台由 ActiveX 服务器(ActiveX Server)和 ActiveX 客户(ActiveX Client)两部分组成。

·ActiveX 服务器是中间级,为事务逻辑(Business Logical)及使用 ActiveX 构件的应用提供定位服务。ActiveX 服务器的核心技术包括 Windows NT 服务器,微软事务服务器 MTS(Microsoft Transaction Server),数据管理服务,目录服务,Web 服务及网络服务。MTS 是传统的线程池(Thread Polling)、数据库多路复用(Database Multiplexing)与基于构件的程序设计模型的结合,用于协调多个构件的运行,以形成一个满足用户要求的完整的开始、更新、提交或取消的事务过程(Transaction)。数据管理使得 OLE DB 和 ODBC 可以访问诸如 DB2, Oracle 和 SQL Server 这样的数据源。目录服务是围绕 DCOM 提供目录服务层,使得远程对象在网络上彼此定位。Web 服务围绕 Internet Information Server(IIS)提供用于服务器端(Server - side) Web 应用开发的脚本机制(Scripting Mechanism)。网络服务围绕 DCOM 为构件通过远程过程调用 RPC 提供连接服务。

·ActiveX 客户平台的目标是跨平台的(Cross - Platform),微软期望 ActiveX 技术用于广泛的操作系统。这

个计划使用标准的 HTML, JAVA 虚拟机和脚本语言,如 VBScript, JScript (JAVA Script 的微软版本)。目前 ActiveX 客户平台是依赖于微软的 Internet Explorer 3.0 或 4.0 来驱动。尽管 IE 已经在一些非 Windows(non - Windows)的平台上出现,但还是 Windows 平台得到最好的支持和最新的版本。

7. ActiveX 工具

微软的 Visual C++ 是第一个有能力产生 ActiveX 控制的开发工具。现在有众多的开发工具可以产生 ActiveX 构件。如 Borland 的 Delphi 3.0 和 C++ Builder, Powersoft 的 PowerBuilder 6.0, 微软的 Visual C++, Visual J++ 和 Visual Basic 5.0。也许 Visual Basic 是产生 ActiveX 构件最好的工具,它允许程序员使用友好的视觉界面快速产生 ActiveX 构件。另外 VBScript, JAVAScript 以及 JScript 都是用于在 Web 上引用 ActiveX 构件的脚本工具。

8. 结束语

ActiveX 技术不仅可以用来开发通常的客户/服务器应用,更重要的是 ActiveX 是微软整个 Internet 战略的中心环节。ActiveX 帮助应用开发者、Web 网页制作者建立动态的 Internet 工具、网页和站点。ActiveX 可能在建立 Internet/Intranet 应用方面成为 JAVA 势均力敌的竞争对手。

参考文献

- [1] David Chapped and David S. Linthicum, ActiveX DEMYSTIFIED, BYTE, SEPTEMBER 1997 pp. 56 - 64
- [2] 李威 杨乔林, Intranet 中基于软件构件的企业分布计算, 计算机系统应用, 1998.2
- [3] 杨乔林 李威, 饶上荣, 张颖, 用于企业计算的 Intranet 规划与设计, 计算机系统应用, 1998.2
- [4] 微软, Visual Basic 5.0 Books Online
- [5] 刘东波, Web 数据库进展, 中国计算机用户, 1997 10 pp. 45 - 49
- [6] Brian Farrer, ActiveX 使用指南, 机械工业出版社, 1997.6

(来稿时间:1997年11月)