Atitit 分布式对象体系

.2.1 分布式对象体系  
在基于对象的分布式系统中，对象的概念在分布式实现中起着极其关键的作用。从原理上来讲，所有的一切都被作为对象抽象出来，而客户端将以调用对象的方式来获得服务和资源。分布式对象之所以成为重要的范型，是因为它相对比较容易地把分布的特性隐藏在对象接口后面。此外，因为对象实际上可以是任何事务，所以它也是构建系统的强大范型。面向对象技术于20 世纪80 年代开始用于开发分布式系统。同样，在达到高度分布式透明性的同时，通过远程服务器宿主独立对象的理念构成了开发新一代分布式系统的稳固的基础。在分布式对象体系架构中，比较有代表性的技术有 DCOM、CORBA 及 RMI。  
1. DCOM（COM ）  
1992 年4 月，微软发布 Windows 3.1 ，包括一种被称为 OLE（Object Linking and Embedding）的机制。这允许一个程序动态链接其他库来支持其他功能，如将一个电子表格嵌入 Word 文档。OLE演变成了 COM （Component Object Model）。一个 COM 对象是一个二进制文件。使用 COM 服务的程序来访问标准化接口的 COM 对象，而不是其内部结构。COM 对象用全局唯一标识符（GUID）来命名，用类的 ID 来识别对象的类。可以有多种方法来创建一个 COM 对象，如 CoGetInstance-FromFile。COM 库在系统注册表中查找相应的二进制代码（一个 DLL 或可执行文件）来创建对象，并给调用者返回一个接口指针。COM 的着眼点是在同一台计算机上不同应用程序之间的通信需求。  
DCOM（Distributed Component Object Model）是 COM 的扩展，它支持不同的两台机器上组件间的通信，而且无论它们是运行在局域网、广域网，还是 Internet 上。借助 DCOM 的应用程序将能够进行任意空间分布。DCOM 于1996 年在 Windows NT 4.0 中引入，后来更名为 COM 。由于DCOM 是为了支持访问远程 COM 对象，需要创建一个对象的过程，此时需要提供服务器的网络名及类 ID。微软提供了一些机制来实现这一点。最透明的方式是远程计算机的名称固定在注册表（或DCOM 类存储）里，与特定类 ID 相关联。采用这种方式之后，应用程序便不知道它正在访问一个远程对象，并且可以使用与访问本地 COM 对象相同的接口指针。另外，应用程序也可指定一个机器名作为参数。  
由于 DCOM 是 COM 这个组件技术的无缝升级，所以能够从现有的有关 COM 的知识中获益，以前在 COM 中开发的应用程序、组件、工具都可以移入分布式的环境中。DCOM 将屏蔽底层网络协议的细节，你只需要集中精力于应用。  
DCOM 最大的缺点是，这是微软独家的解决办法，但在跨防火墙方面的工作做得不是很好（大多数RPC 系统也有类似的问题），因为防火墙必须允许某些端口来让 ORPC 和 DCOM 通过。