

OPC 原理和配置说明书

OPC 原理和配置说明书

第一部分：OPC 原理

1. 概述

OPC (OLE for Process Control——用于过程控制的 OLE)是一个工业标准，它是许多世界领先的自动化和软、硬件公司与微软公司合作的结晶。这个标准定义了应用 Microsoft 操作系统在基于 PC 的客户机之间交换自动化实时数据的方法。管理该标准的组织是 OPC 基金会。该基金会的会员单位在世界范围内超过 220 个。包括了世界上几乎全部的控制系统、仪器仪表和过程控制系统的主要供应商。OPC 基金会的先驱——一支由 Fisher-Rosemount、Rockwell 软件公司、Opto 22、Intellution 和 Intuitive Technology 公司组成的“特别工作组”——在经过一年工作后，开发出一个基本的、可运行的 OPC 规范。简化的第一阶段的标准在 1996 年 8 月发布。

随着 1997 年 2 月 Microsoft 公司推出 Windows95 支持的 DCOM 技术，1997 年 9 月新成立的 OPC Foundation 对 OPC 规范进行修改，增加了数据访问等一些标准，OPC 规范得到了进一步的完善。OPC 是基于 Microsoft 公司的 Distributed Internet Application (DNA) 构架和 Component Object Model (COM) 技术的，根据易于扩展性而设计的。OPC 规范定义了一个工业标准接口，这个标准使得 COM 技术适用于过程控制和制造自动化等应用领域。OPC 是以 OLE/COM 机制作为应用程序的通讯标准。OLE/COM 是一种客户/服务器模式，具有语言无关性、代码重用性、易于集成性等优点。OPC 规范了接口函数，不管现场设备以何种形式存在，客户都以统一的方式去访问，从而保证软件对客户的透明性，使得用户完全从低层的开发中脱离出来。

2. 传统的过程控制系统结构

传统的过程控制系统是一对一的系统，任何一种 HMI 等上位监控软件或其它应用软件（如趋势图软件、数据报表与分析等）在使用某种硬件设备时都需要开发专用的驱动程序。如图 1 所示：

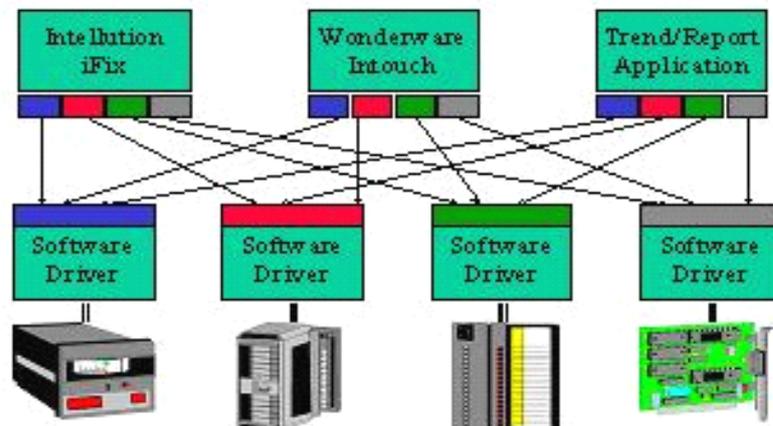


图 1

系统构建完成后的最终结果是：

1 种软件要使用 N 类硬件设备需要开发 N 个驱动程序；

M 类软件要使用 N 类硬件设备需要开发 $M*N$ 个驱动程序；

每增加 1 个新的应用软件需要另外开发 N 个硬件设备的驱动程序；

每增加 1 个新的硬件设备需要为 M 个软件开发新的设备驱动程序；

在上图所示的例子中，针对 4 种控制设备所完成的 3 个应用系统一共需要开发 $3*4=12$ 种驱动程序。新增应用软件或者硬件设备带来的只会是驱动程序种类的迅速增长。

3. 基于 OPC 技术的过程控制系统结构

基于 OPC 技术的过程控制系统则可以完美地解决传统方式种的上述问题。其基本系统结构如图 2 所示：

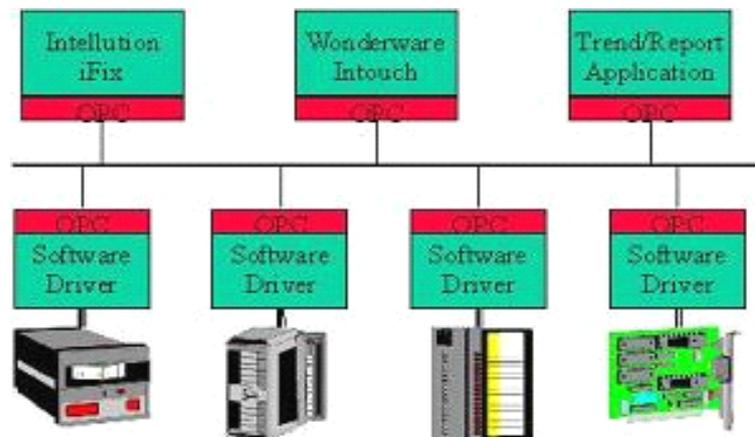


图 2

任何一种设备只需要提供一种驱动就可以供任何软件系统使用，系统构建完成后的最终结果是：

- M 类软件要使用 N 类硬件设备只需要开发 N 个驱动。
- 每增加 1 个新的应用软件不需要另外开发硬件设备的驱动程序；
- 每增加 1 个新的硬件设备只需要为开发 1 个新设备的驱动程序；

在上图所示的例子中，针对 4 种控制设备所完成的 3 个应用系统一共仅需要开发 4 种驱动程序。新增应用软件或者硬件设备可以轻松的扩展您的系统。

由于 OPC 规范基于 OLE/COM 技术，同时 OLE/COM 的扩展远程 OLE 自动化与 DCOM 技术支持 TCP/IP 等多种网络协议，因此可以将 OPC 客户、服务器在物理上分开，分布于网络不同节点上。

OPC 规范可以应用在许多应用程序中，如它们可以应用于从 SCADA 或者 DCS 系统的物理设备中获取原始数据的最底层，它们同样可以应用于从 SCADA 或者 DCS 系统中获取数据

到应用程序中。实际上，OPC 设计的目的就是 from 网络上某节点获取数据。OPC 的客户/服务器关系图同样描述了 OPC 在 SCADA 系统的应用。

4. 采用 OPC 规范设计系统的好处：

采用标准的 Windows 体系接口，硬件制造商为其设备提供的接口程序的数量减少到一个，软件制造商也仅需要开发一套通讯接口程序。即有利于软硬件开发商，更有利于最终用户。

OPC 规范以 OLE/DCOM 为技术基础，而 OLE/DCOM 支持 TCP/IP 等网络协议，因此可以将各个子系统从物理上分开，分布于网络的不同节点上。

OPC 按照面向对象的原则，将一个应用程序（OPC 服务器）作为一个对象封装起来，只将接口方法暴露在外面，客户以统一的方式去调用这个方法，从而保证软件对客户的透明性，使得用户完全从低层的开发中脱离出来。

OPC 实现了远程调用，使得应用程序的分布与系统硬件的分布无关，便于系统硬件配置，使得系统的应用范围更广。

采用 OPC 规范，便于系统的组态，将系统复杂性大大简化，可以大大缩短软件开发周期，提高软件运行的可靠性和稳定性，便于系统的升级与维护。

OPC 规范了接口函数，不管现场设备以何种形式存在，客户都以统一的方式去访问，从而实现系统的开放性，易于实现与其它系统的接口。

5. OPC 的效率

从 OPC 标准的制定到现在已历经了 5 年的时间。在这过去的 5 年中，众多业界领先的制造商已开发了多种 OPC 服务器和客户机应用。在实际工程中也历经了多方面的测试和考验。以瑞士的 TetraPak 为例，其基于 OPC 服务器技术的数据采集系统，保证了对超过 500 个数据点的更新时间为 200 毫秒。

在 Softing 公司的 OPC 服务器性能测试中，分别对基于本地的 OPC 应用和基于 DCOM 的分布式应用进行了测试。

为了保证测试的可信程度，特别选用了两台低档配置的计算机。其配置如下：

硬件：

CPU：奔腾 90MHz 处理器

RAM：48 或 64M

软件：

Windows NT 4.0

在本地测试中,OPC 客户机测试应用程序和 OPC 服务器安装在同一台 PC 上。在基于 DCOM 的分布式 OPC 应用测试中, OPC 客户机测试应用程序在第二台远程 PC 上运行。

测试结果:无论是在本地还是两台计算机之间,5000 个过程变量的变化值能在 1 秒内在 OPC 服务器和客户机之间传送。对于只有 500 个过程变量的情况,只需 100 毫秒时间。因而,OPC 被认为是非常适合于在很短的更新速率内采集大量的动态过程变量的应用场合。

第二部分：环境配置说明

1.将下列文件拷贝至要运行 OPC 服务器和 OPC 客户端的机器上

```
copy opcproxy.dll c:\winnt\system32
copy opccomn_ps.dll c:\winnt\system32
copy opc_aeps.dll c:\winnt\system32
copy opchda_ps.dll c:\winnt\system32
copy aprxdist.exe c:\winnt\system32
copy opcenum.exe c:\winnt\system32
```

2.注册这些 dll 文件

```
REGSVR32 opcproxy.dll
REGSVR32 opccomn_ps.dll
REGSVR32 opc_aeps.dll
REGSVR32 opchda_ps.dll
```

3. 如果在 windows 系统 (\winnt\system32) 目录下不存在 actxprxy.dll, 运行 aprxdist.exe

```
aprxdist
```

4. 安装 opcenum.exe

```
opcenum /regserver
```

一、NT/2000 服务器 DCOM 配置

1. 在 WINNT\system32 目录下找到：dcomcnfg.exe，并运行它。
2. 在默认属性页面不作改动:默认选项是：选择在这台计算机上启动分布式 COM(E)选项、默认身份验证级别是：连接，默认模拟级别是：标识。如图 3 所示。

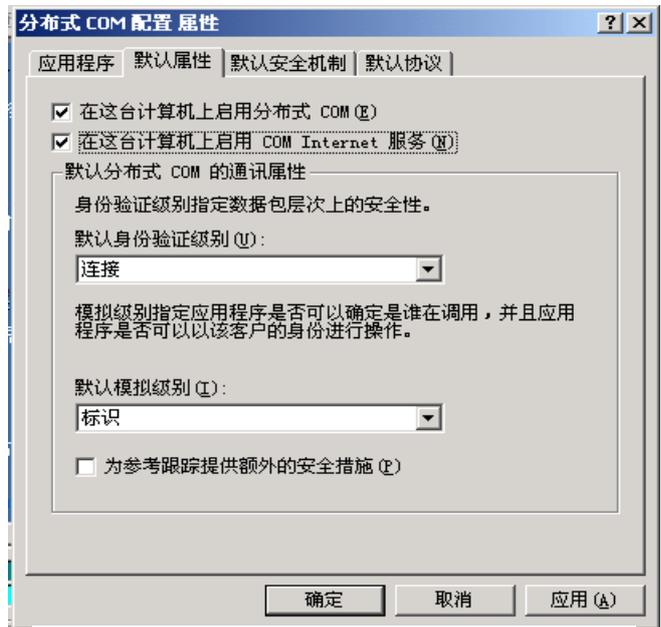


图 3

3. 在默认安全机制页面
 - a. 按默认访问权限框里的(编辑默认值)按钮，如果无用户：Everyone，按(添加)按钮，选择：Everyone，按(添加)按钮，在访问类型框里选择：允许访问，按(确定)按钮,如图 4 所示。再按(确定)按钮回到默认安全机制页面。

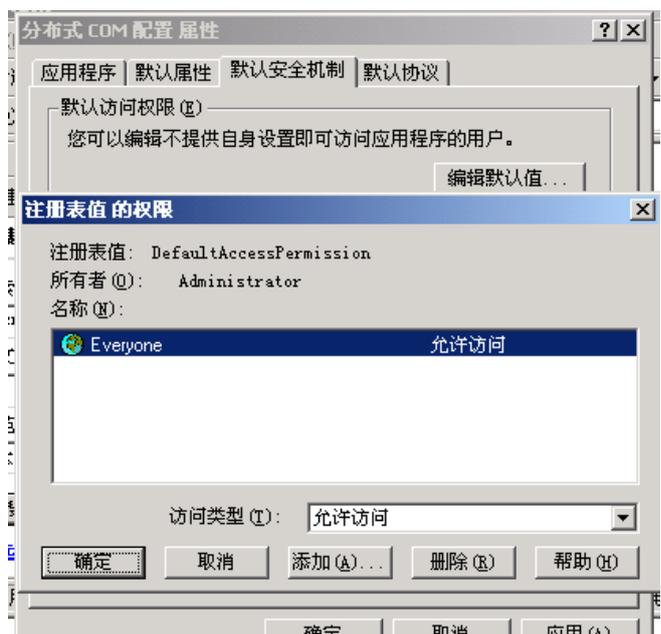


图 4

- b. 按默认启动权限框里的(编辑默认值)按钮, 如果无用户: Everyone, 按(添加)按钮, 选择: Everyone, 按(添加)按钮, 在访问类型框里选择: 允许调用, 按(确定)按钮, 如图 5 所示。再按(确定)按钮回到默认安全机制页面。



图 5

- c. 按默认配置权限框里的(编辑默认值)按钮, 按(添加)按钮, 选择: Everyone, 按(添加)按钮, 在访问类型框里选择: 完全控制, 按(确定)按钮, 如图 6 所示。再按(确定)按钮回到默认安全机制页面。

当然你也可给用户分配权限, 不过麻烦, 一个用户的权限不对, 就不能连接到服务器。

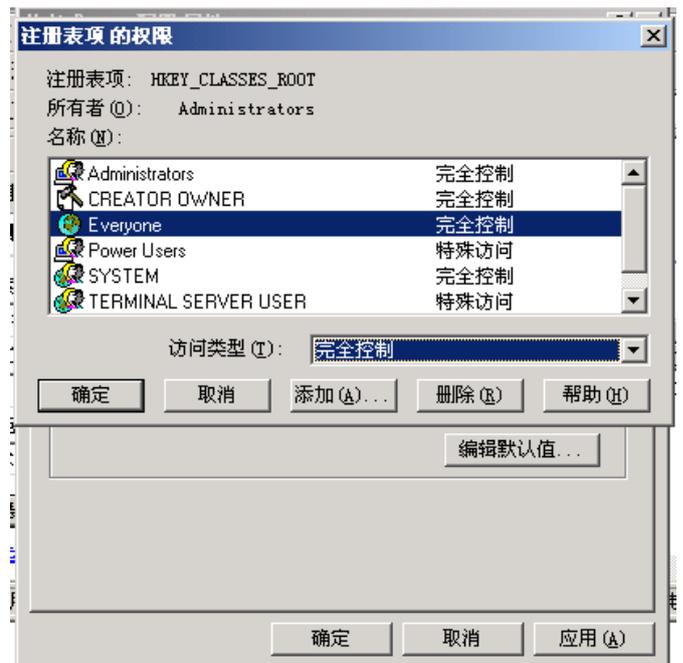


图 6

4. 默认协议页面不作改动, 回到在应用程序页面。

5. 在应用程序页面，用鼠标左键点击选择：NSPS.OPC.Server.VC8.0，如图 7 所示



图 7

6. 按(属性)按钮

- a.在常规页面不作改动，身份验证级别是：默认值。如图 8 所示。

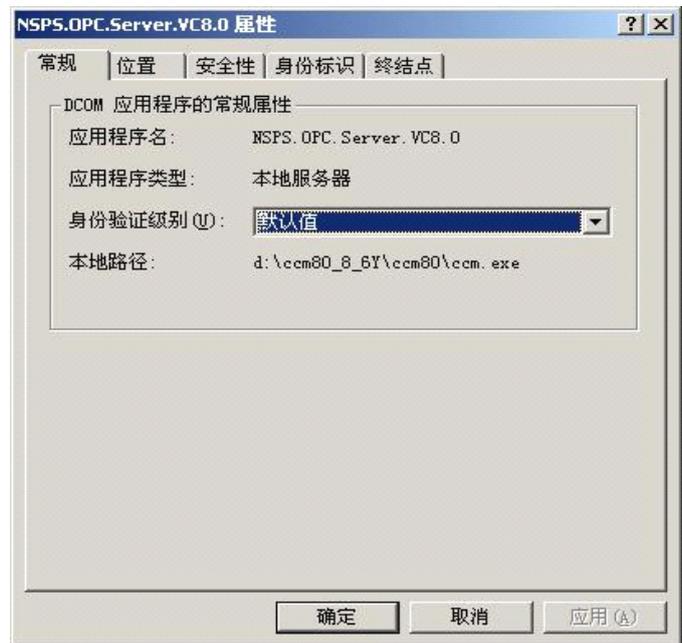


图 8

b.在位置页面不作改动，默认选项是：在这台计算机上运行应用程序。如图 9 所示。

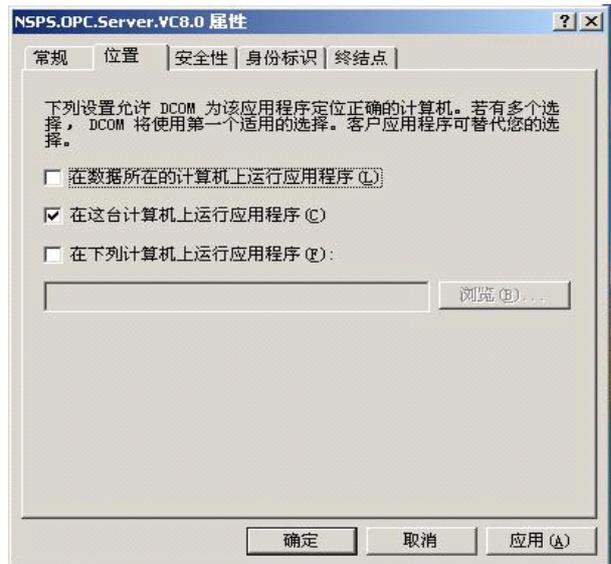


图 9

c.在安全性页面不作改动，全部选择：使用默认选项。如图 10 所示。

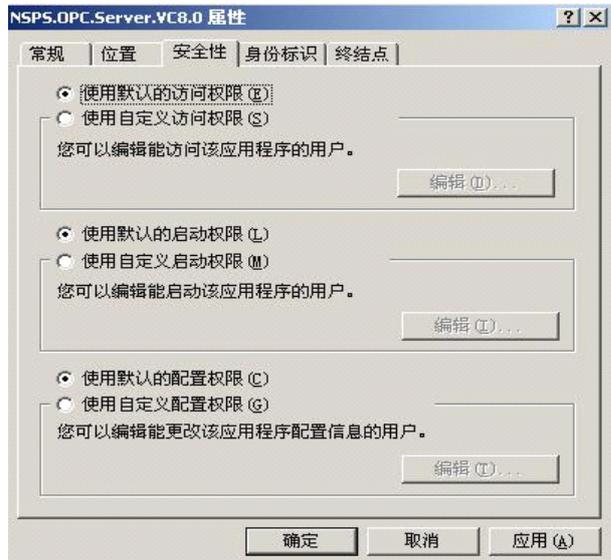
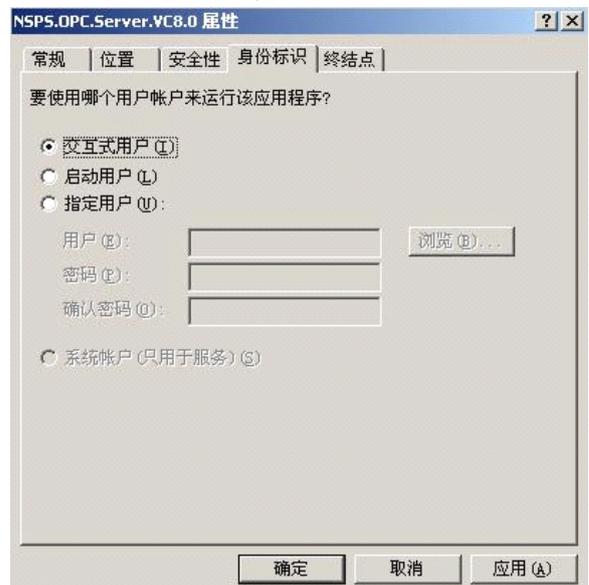


图 10

d.在身份标示页面，选择：交互式用户。如图 11 所示。



e. 终结点页面不作改动。
f. 按(确定)按钮，回到，在应用程序页面

图 11

7.在应用程序页面,用鼠标左键点击选择: OpcEnum, 如图 12 所示。点击属性页, 参照步骤 6 设置各项属性。

8.按(确定)按钮, 关闭程序



图 12

二、XP 服务器 DCOM 配置

1. 运行 dcomcnfg.exe(在 WINNT\system32 目录下), 可在 “开始->运行” 中输入 “dcomcnfg”, 按确认键运行, 出现如图 13 所示的界面。

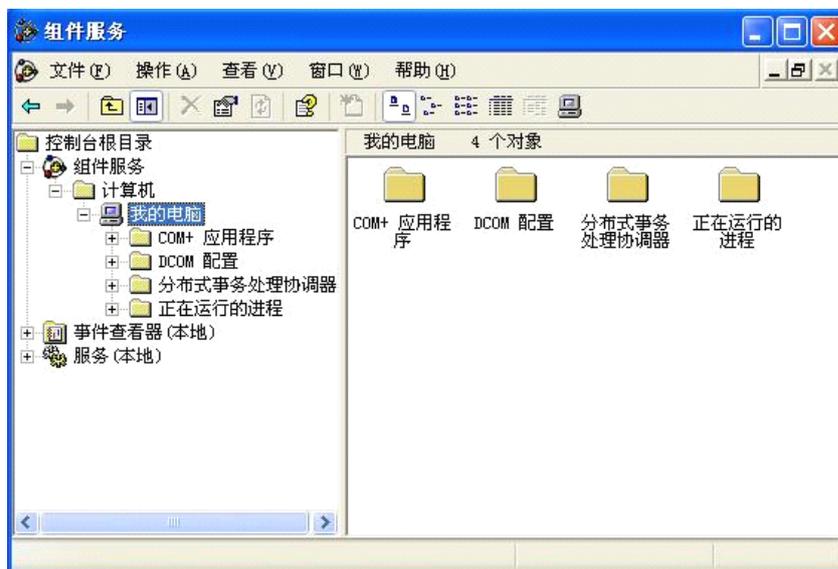


图 13

2. 右键点击“我的电脑”，选择“属性”出现如图 2.2 所示的对话框。

3. 在“默认属性”设置：选择在此计算机上启用分布式 COM(E)选项，“默认身份验证级别”是：连接，“默认模拟级别”是：标识。如图 14 所示。

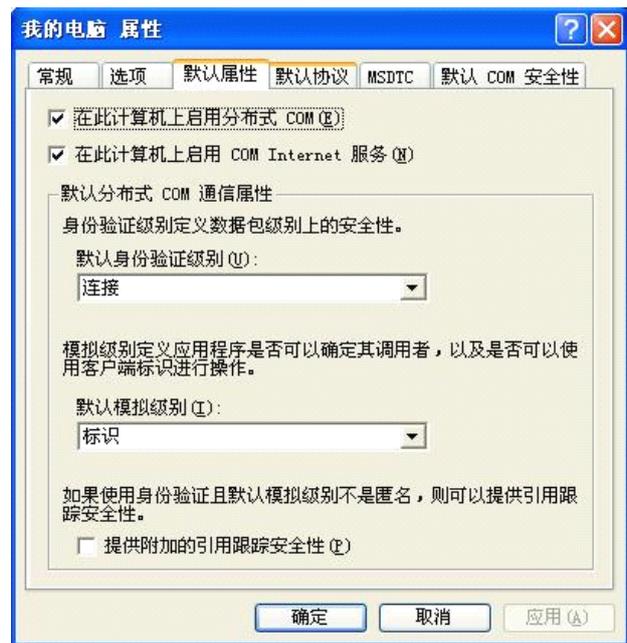


图 14

4. 在“默认 COM 安全性”页面中

a. 在“访问权限”栏里，按“编辑默认值”按钮，添加“Everyone”，访问权限设为“允许”。如图 15 所示。



图 15

b.在“启动权限”栏里，按“编辑默认值”按钮，添加“Everyone”，启动权限设为“允许”。如图 16 所示。

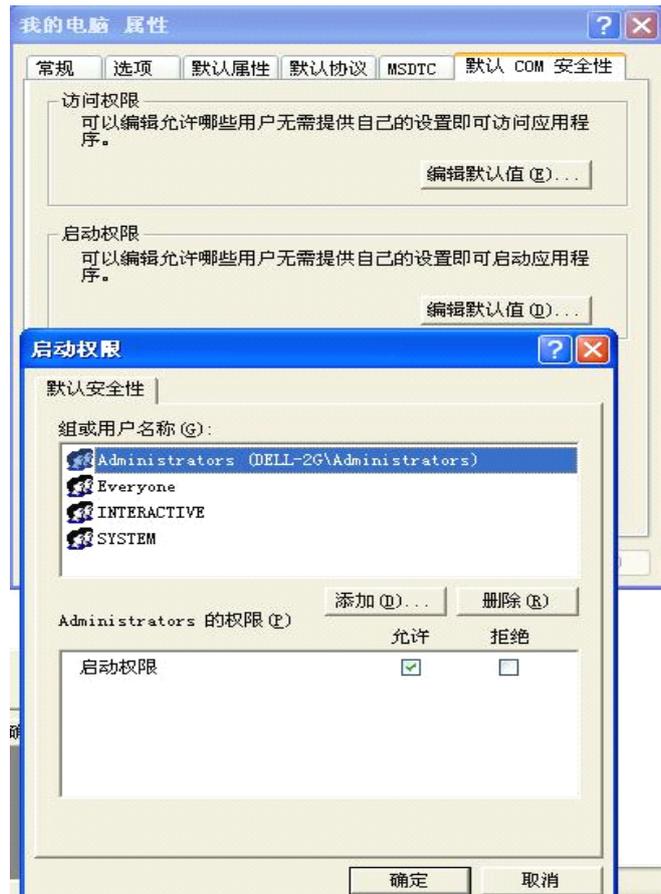


图 16

5.按“确定”按钮回到如图 2.1 所示的 dcomcnfg 程序界面，展开 DCOM 配置，右键点击 NSPS.OPC.Server.VC8.0，如图 17 所示。

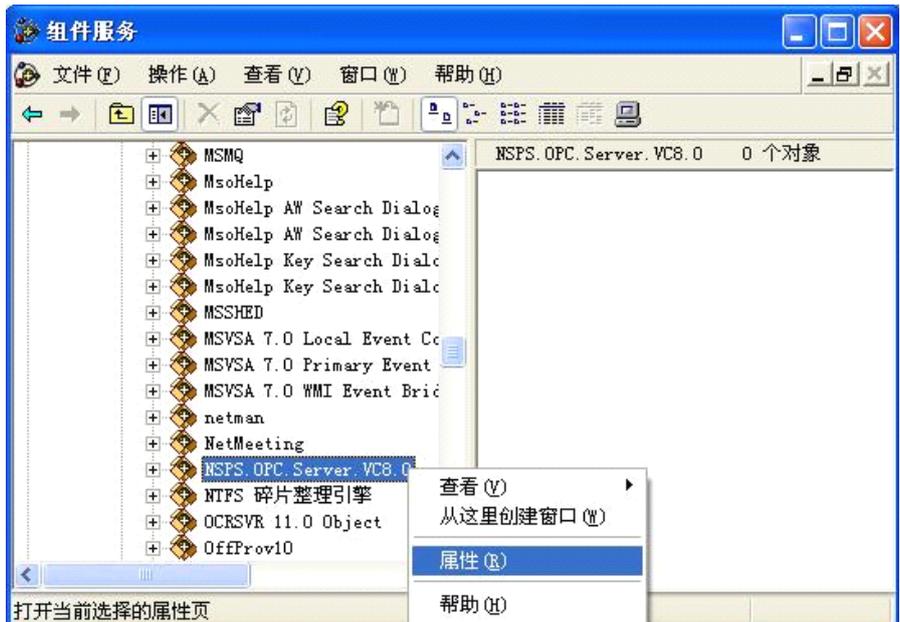


图 17

6.在图 2.5 所示的弹出菜单中点击“属性”项。

a. “常规”页面中的“身份验证级别”设为“默认”。如图 18 所示。



图 18

b.在“位置”页面中，选择“在此计算机上运行应用程序”。如图 19 所示。

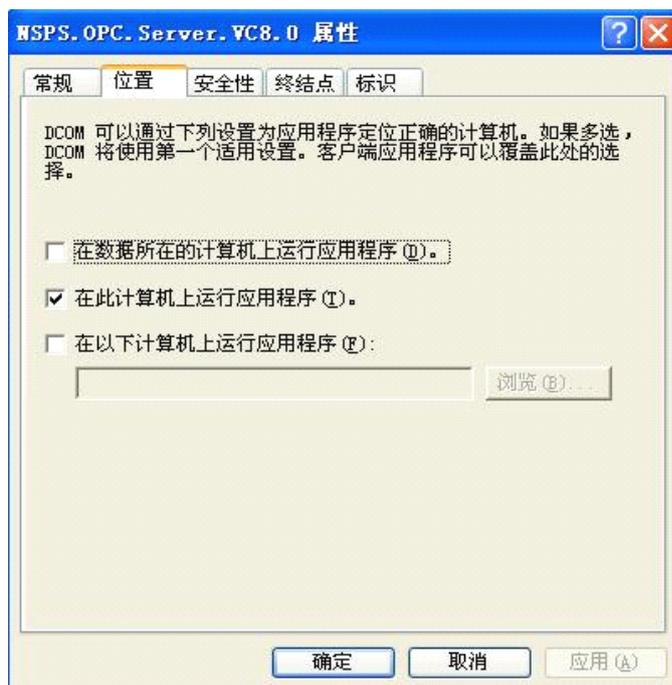


图 19

c.在“安全性”页面中，“启动权限”设置为“使用默认值”，“访问权限”设置为“使用默认值”，“配置权限”设置为“自定义”，并按“编辑”按钮，添加“Everyone”，“完全控制”置为允许。如图 20 所示。

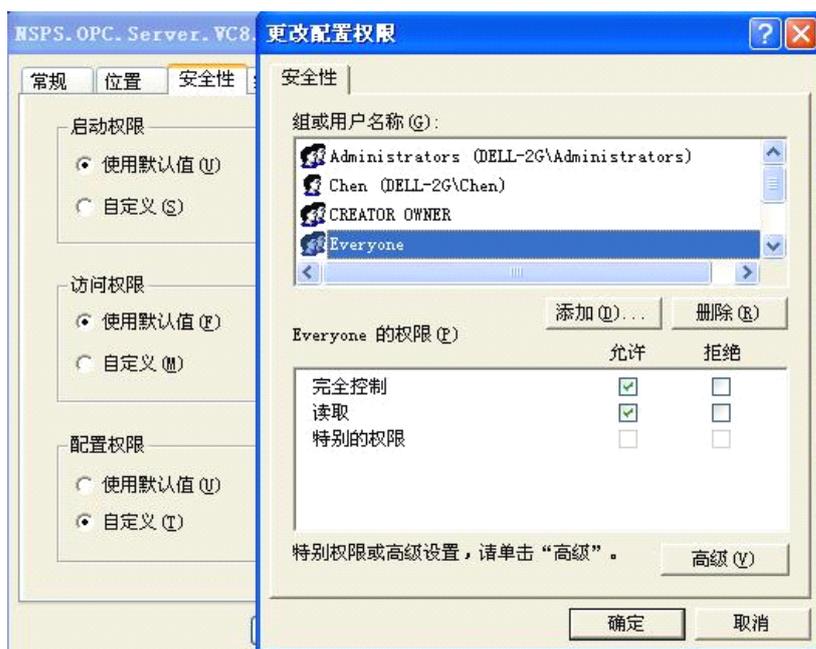


图 20

d.在“标识”页中，选择“交互式用户”，如图 21 所示。

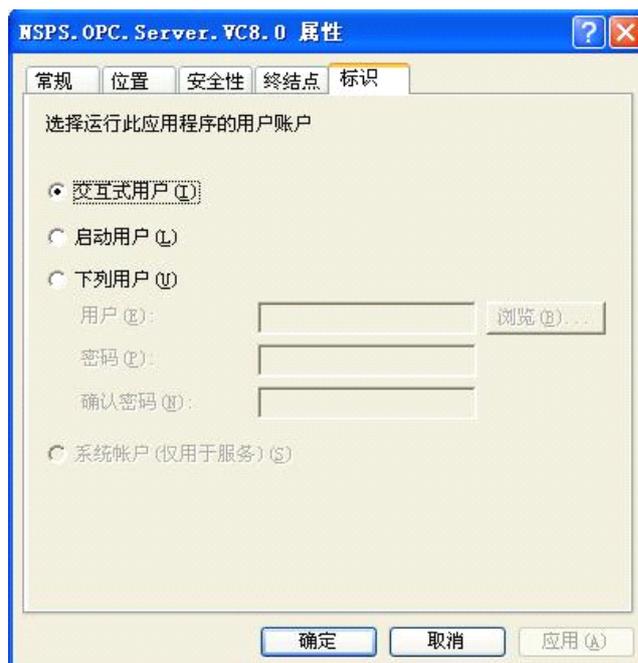


图 21

7. 按“确认”返回到 dcomcnfg 程序主界面，选中“OpcEnum”项（如图22所示），打开“属性”页面，做与 NSPS.OPC.Server.VC8.0 相同的配置。

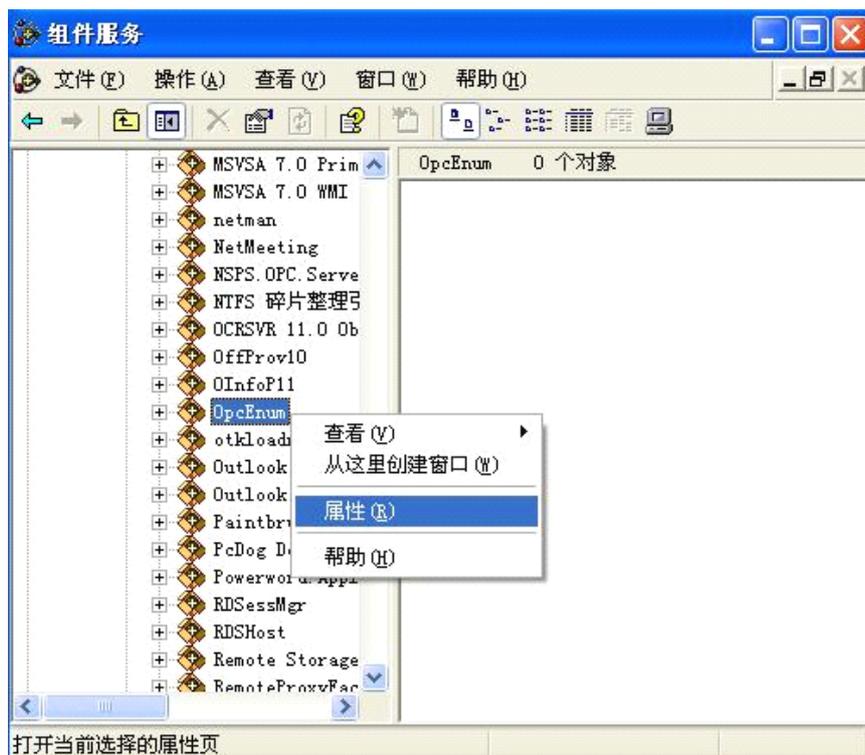


图 22

8. 最后后关闭 dcomcnfg 程序，配置全部完成。

三、在运行 OPC 服务器的主机中添加 OPC 客户端主机的登录帐户

如果 OPC 客户端需要远程访问 OPC 服务器，运行 OPC 服务器的主机操作系统无论是 windows 2000 还是 windows xp，都需要将 OPC 客户端的登录帐号添加到 OPC 服务器的主机中。

在 OPC 服务器主机中运行开始/设置/控制面板/管理工具/计算机管理/本地用户和组/用户/操作/新用户，将客户端机器的登录帐号添加进去即可，如果有多台客户端访问该服务器，就将这些客户端机器的登录帐号都添加进来。如果多台客户机的登录帐号一样（指的是用户名和密码都一样），那么只需在 OPC 服务器主机中添加一个就可以了。

总之，OPC 客户端主机的登录帐号，在 OPC 服务器主机的计算机管理的用户列表中必须找到。

四、特殊情况

当运行 OPC 服务器的主机操作系统是 windows xp，按上文配置后客户端依然无法访问 OPC 服务器，那么请将**管理工具》本地安全策略》本地策略》安全选项》网络访问》本地帐户的共享和安全模式**置为**经典——本地用户以自己的身份验证**。