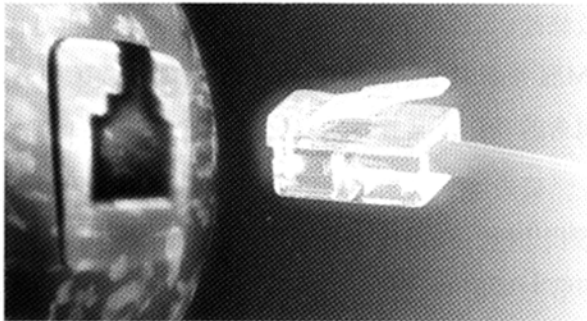


基于COM/DCOM 技术的卫星通信终端网络监控系统的实现



任志国 (太原卫星发射中心技术部通信室 030027)

摘要: 本文介绍了如何利用组件对象模型 (Component Object Model, COM) 和分布式组件对象模型 (Distributed Component Object Model, DCOM) 技术构建一个卫星通信终端设备计算机网络监控系统, 并结合一个实例, 阐明该系统的组成及工作原理。

关键词: COM/DCOM 接口 可连接对象 监视 控制

本文结合一个具体的实例介绍如何利用 COM/DCOM 技术实现一个比较通用的计算机网络监控系统。

1 系统的硬件组成

则先进行接口变换, 转换成标准 RS-232 串口后再和计算机相连。为了提高监控微机的使用效率、节约成本, 可在监控微机安装一个串口扩展卡, 这样一台计算机可以连接多台设备。由于地域特点, 各种设备不一定集中在一个地方, 此时所监控的设备可通过 Modem 以专线或拨号方式连接到监控网络中。系统硬件组成如图 1 所示:

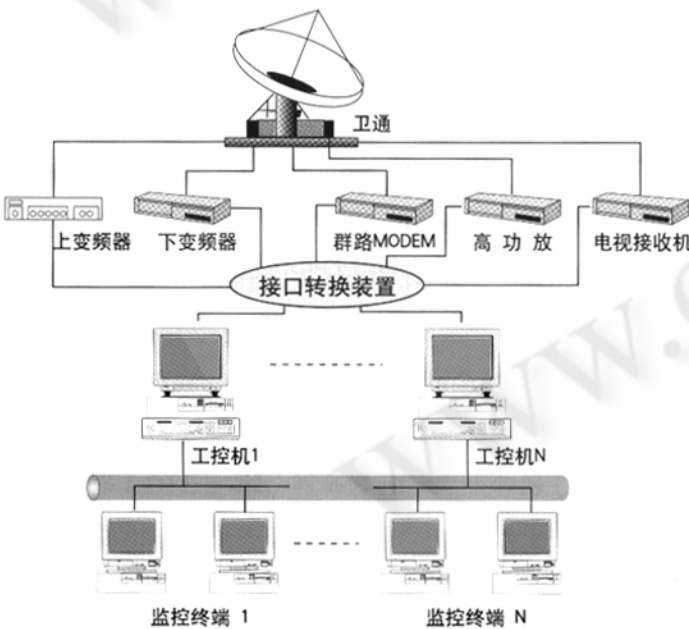


图 1 系统硬件组成

为了完成对卫通机房各个通信终端设备监视与控制, 必须通过各种接口转接装置把各个终端设备引接到监控网络中。如果设备具有标准的 RS-232 串口, 且可以直接接收控制指令, 则此类设备可以直接引接到监控微机的串口上; 如果设备具有其他类型监控接口如 RS-422/485,

2 系统的软件组成

2.1 监视功能

该系统在功能上由两大模块来完成: 监视和控制功能。监视功能是指当设备运行出现告警或故障时, 能生成一个告警信号并在网络中广播, 并由相应的接收软件处理, 最终以声光形式提醒用户。完成此功能是一个 COM 对象服务器即告警服务器来完成, 此 COM 对象接口定义如下:

```
/******IWarnNotify.IDL******/
import "oidl.idl";
import "ocidl.idl";
[
object,
uuid(D760787F-2449-469A-BBC2-A102454A7A04),
helpstring("IMakeWarn Interface")]
interface IMakeWarn : IUnknown
{
[id(1), helpstring("method Send_Warn_Msg")] HRESULT
Send_Warn_Msg([in] int DevID,[in]BSTR bstrMsg);};
```

```
[object,
uuid(85464369-2287-4FFC-B1A5-5D98707963D2),
[id(2),helpstring("method Save_Warn_Msg")]
HRESULT Save_Warn_Msg([in] int DevID,[in] BSTR
bstrMsg);
helpstring("_INotifyWarn Interface")
interface INotifyWarn : IUnknown
{
HRESULT OnRecieveWarn([in] int DevID ,[in] BSTR
bstrMsg);
};
[
uuid(10123B4B-5AAE-4815-8A46-D40F8446355E),
version(1.0),
helpstring("WarnNotify 1.0 Type Library")
]
library BILLNOTIFYLib
{
importlib("stdole32.tlb");
importlib("stdole2.tlb");
[uuid(73B7282F-19DD-4C8B-ACD6-4A4BBC51715B),
helpstring("ComMakeWarn Class")]
coclass ComMakeWarn
{
[default] interface IMakeWarn;
[default,source] interface INotifyWarn;
};
};
```

上面文件是事例工程中告警服务器接口定义文件,此COM服务器对象是一个可连接对象,首先通过IMakeWarn接口生成一个告警事件,然后通过连接点接口IMakeWarnEvnts进行事件广播。接口INotifyWarn是一个源接口(source),其包含在服务器中,但它的完成在客户端(这里具体是指告警接收软件)。接口IMakeWarn包含两个方法,Send_Warn_Msg方法用于生成一条告警消息,Save_Warn_Msg用于保存告警消息;接口INotifyWarn只有一个方法既OnRecieveWarn。参数nDevID为设备标识符,代表发生告警的设备,bstrMsg代表具体的告警消息。当然通过进一步扩展接口监视内容可进一步增多。这里客户作为告警信息接收器,要完成COM服务器的回调功能,要遵守“从连接点容器到连接

点到枚举器再到接口指针”这一顺序,虽然这一过程很复杂,但很通用。有关技术可参阅参考文献2第十三章有关内容。此书已翻译成中文,由清华大学出版社出版,书名为《COM/DCOM编程指南》。

2.2 控制功能

控制功能是指在监控网络上任何一台监控终端对设备进行参数修改、工作状态的切换等工作。控制功能也是由一个COM服务器即控制服务器完成的,主要完成ICtrlSvr接口所包含的功能。其接口定义文件包含的方法如下:

```
...
[id(1), helpstring("method SaveOperateLog")] HRESULT
SaveOperateLog([in] bstrOperateParam);
[id(2),helpstring("method Is_A_Authority")] HRESULT
Is_A_Authority([in] BSTR bstrUserName,[in] BSTR
bstrPassWord,[out] int* pnResult);
[id(3), helpstring("method Open_Port")] HRESULT
Open_Port([in] int nPort,[out] int* pnResult);
[id(4), helpstring("method Close_Port")] HRESULT
Close_Port([in] int nPort,[out] int* pnesult);
[id(5), helpstring("method Send_Cmd")] HRESULT
Send_Cmd([in] int nPort,[in]BSTR bstrCmd,[out]BSTR*
pbstrEcho,[out] int* pnResult);
[id(6), helpstring("method Modify_Port_Param")]
HRESULT Modify_port_Param([in] int ort,[in] BSTR
bstrParam,[out] int* pnResult);
[id(7), helpstring("method Query_Port")] HRESULT
QueryPort([in] BSTR bstrName,[out] int* pnPort,[out]
int* pnResult);
}
```

...

该接口包含了七个方法,每个方法的名称基本上说明了该方法所完成的功能。SaveOperateLog方法用于把控制设备的各种操记录保存到数据库,以便实现对历史操作进行查询;Is_A_Authority是服务器验证客户的身份,以验证操作员对设备进行操作是合法的,同时也把在线控制信息显示在前端服务器的用户界面上。Modify_Port_Param方法修改相应串口的通信参数。Query_Port方法是根据设备名称返回该设备连接的串口号,以便客户操作该端口。Open_Port打开相应的串口。Send_Cmd和Open_Port配合使用,它向设备发送控制命

令并且一直等到设备的响应信息返回。Close_Port 关闭相应串口。以上介绍了两个服务器的所完成的功能,并给出相应的接口介绍,到此读者可能问为什么没有相应实现的代码,“只看接口、不看实现”这才真正是 COM 技术核心所在,只要给出一个接口,就可以用不同的语言来实现它。

3 监控过程的实现

该监控系统是由两层来完成的,前置服务层主要完成对设备运行进行监视和控制,如发生告警消息则向监控网络进行告警事件广播,同时保存到数据库或文件中;如果收到控制指令,则把相应的控制指令送往相应的串口。客户层主要完成接收告警消息和发送控制指令两大功能。当收到告警消息时,客户将显示醒目界面提示工作人员,并辅助以声音告警;当需要对设备进行控制时,操作人员在客户层选中相应的设备,通过建立一个远程控制组件,并通过网络对设备进行远程控制操作。整个系统的流程图如图(2)所示。客户层由 Client.exe 完成,前置服务层由 CtrlServ.exe 和 WarnNotify.exe 组成。CtrlServ.exe 必须在图(1)中所示的工控机上运行,因为该程序包含了控制服务器,它将对工业监控机上串口进行操作。当用 Windows NT4.0 以上版本组网时,每台监控机可以扩展到 255 个串口,所以一般情况下有一台工控机就可以完成对整个系统设备进行监视和控制。

4 COM 服务器完成与使用

4.1 系统软件的完成

由图 2 可知,前置服务层包含两个 COM 服务器即控制服务器和告警服务器。COM 服务器有多种形式,进程内服务器 (.DLL)、本地服务器和远程服务器 (.EXE)。由于客户端和前置服务层分别在不同的机器上运行,所以采用远程服务器完成两个服务器比较适合的。服务器完成的方法也有多种,包括 MFC、ATL、甚至 VB,笔者偏爱使用 MFC;一方面介绍 MFC 的书籍较普遍,且 MFC 对 COM 进行了包装,用户使用 COM 时可以省略很多细节;另一方面 MFC 直接支持 WIN API 函数调用,对串口操作也很方便。在控制服务器程序 CtrlServ.exe 中,笔者使用了一个对串口打包的 C++ 类 CComm。上述简要介绍了系统的设计思想和主要结构,由于篇幅有限,源代码不再附上。

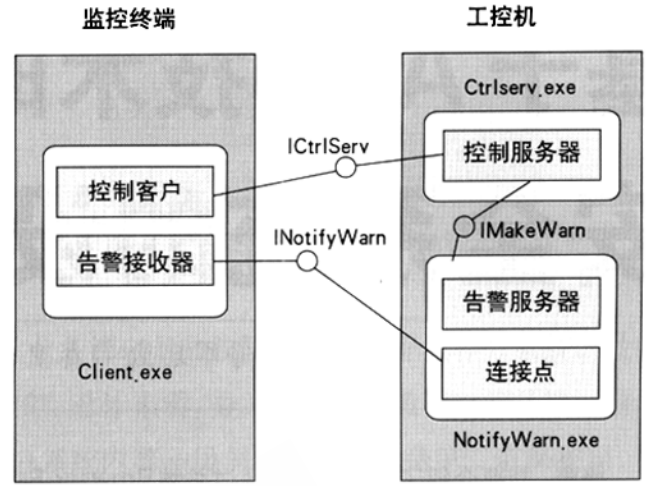


图 2 系统软件组成

4.2 系统软件的使用

CtrlServ.exe 和 WarnNotify.exe 是两个远程服务器,所示首次运行必须以自注册方式运行即在以命令行参数“/REGSERVER”运行两个可执行文件。由于接口 ICtrlServ 和 IMakeWarn 是两个自定义接口(直接派生于 IUnknown),所以为了远程运行两个服务器,必须在运行该系统的微机上注册它们的代理,即 CtrlServPS.dll 和 WarnNotifyPS.dll,注册方法用 REGSVR32 程序。代理主要作用是实现远程调用过程中参数的编排,其代码在编译 IDL 文件自动生成,你只需写出 DLL 要求的 DEF 文件即可。最后出于 COM/DCOM 安全的需要,必须对 COM 对象服务器进行配置,以便系统能正确的运行。注册完以后通过 DCOMCNFG.EXE 程序进行服务器配置。运行 DCOMCNFG.EXE 后,在应用程序列表框中找到 CtrlServ.Ctrl,点击“属性”按钮,在身份标识中选中“交互式”用户按钮,按确定按钮返回。用同样的方法对 NotifyWarn.exe 配置,它的服务器标识为 WarnNotify.Ctrl。启动了 CtrlServ.exe 后,NotifyWarn.exe 也就自动运行了,此时工作人员可在监控网上任一监控终端运行 Client.exe,就可实现对监控网内任一通信终端设备运行状态进行监视和控制了!对设备控制是一个向导式应用过程,维护人员只需按提示输入必要信息就可进入一个虚拟的网络终端,此时就可以象一个和设备直连的超级终端对设备进行控制和参数设置。■

参考文献

- 1 Dale Regerson, Inside COM(Microsoft Press, 1997)。
- 2 The Waite group's ,COM/DCOM PRIMER PLUS, Sams Publishing, 1999. 5.