

基于 LNS 和 COM 的 LonWorks 网络监控软件开发

Monitor And Control of a LonWorks Network Based On LNS And COM

TP27 A

雷刚 吕国强 洪占勇

摘要: 目前大多数 LonWorks 网络监控软件采用 DDE Server 或 ActiveX 控件开发, 与采用 COM 开发相比, 响应速度较慢。本文介绍了用 COM 开发基于 LNS 的 LonWorks 组网与监控软件的开发过程, 并给出了实例代码, 对 LNS 下的几种编程方式和网络监控方式进行了分析, 对 LNS 的几种编程环境进行了讨论。介绍了用 COM 方式开发的网络监控软件的特点。

关键词: 监控 LNS COM 组件模型 LonWorks 网络

Abstract: The monitor and control software development based on LNS and COM is introduced. Several kinds of programming manners are discussed in this article. Some sample code which can monitor LonWorks Network is shown.

Keywords: Control and Monitor COM Component Model LonWorks Network

1 LonWorks 现场总线简介

LonWorks 现场总线是 1991 年美国 ECHOLON 公司推出的一种控制局域网, 它实现了 OSI 的七层网络模型, 所以也经常叫它 LonWorks 控制网络。与其它现场总线相比, 它具有组网方便, 通讯速率快, 不局限于通讯介质等优点。日前其在国内的应用非常广泛, 工程实例遍及楼宇控制, 智能小区, 工厂, 矿井监控等场合。由于其实现了完整的 OSI 网络模型, 所以与以太网的无缝集成非常方便, 是一种具有良好发展前景的总线。

2 LNS 的简介

LNS(LonWorks Network Services)是控制网络工业界的一个支持多客户的网络操作系统, 所以说它是网络操作系统是因为 LNS 封装了基本的网络操作, 如变量连接, 消息标签连接, 监控点的建立, 网络安装, 网络维护等, 监控或组网软件只需要与 LNS 交换信息就可以完成原本很复杂的工作, 就如同应用程序与 WINDOWS 操作系统的关系一样。LNS 集成了 IP 协议, 这使得通过 Internet 来监控 LonWorks 网络非常方便。通过 LNS, 几个客户程序可以同时打开一个网络, 而无需考虑同步的问题。这大大方便了网络的安装与监控。

LNS 由如下几个组件构成: LNS Server, LNS Object Server COM Component, Data Server, Network Interface。其中, LNS Server 是核心组件, 它来完成网络操作等工作, Data Server 主要用于向应用程序提供数据封装服务, 当客户通过 Internet 监控 LonWorks 网络时, DataServer 还用来保存客户程序创建的监控集合, 注意在这种场合下, 监控集合保存在运行 LNS Server 的机器的 DataServer 里, 而不在运行监控程序的机器的 DataServer 里。

当客户程序仅需监控网络, 而不需管理和维护网络时, 客户程序可以以一种 Independent-Server 的方式运行, 在这种方式下, LNS Server 可以关闭, 客户程序只与 DataServer 通讯即可达到监控的目的。要注意的是, 远程轻量级的客户程序(通过以太网连接本地机监控网络的程序)不能以 Independent-Server 的方

式运行, 也就是说即使仅仅对网络进行监控, 也需要打开 LNS Server, Network Interface 即网络接口, 用于将运行 LNS 的机器连入 LonWorks 网络。它主要由两部份组成: 物理上的网络接口和软件上的网络接口驱动程序。

LNS 还提供一种虚拟的网络接口(VNI)。用这种网络接口客户程序可以同时打开多个网络, 就象是安装了多个网络接口下一样。由于工作在 VNI 方式下时, 很多接口卡要完成的工作都在软件上用 runtime 函数实现, 所以使用 VNI 还可以大大提高网络接口的性能, 提高数据吞吐量。因此在监控任务比较繁重的场合, 推荐使用虚拟网络接口。

下面是与用户关系最密切的 LNS 组件—LNS ObjectServer COM Component。用户要与这个组件打交道才能通过 LNS Server 和 DataServer 来管理、安装和监控 LonWorks 网络。说得通俗些, LNS ObjectServer COM Component 相当于 LNS Server 和 DataServer 向外提供一个接口, 应用程序必须要通过 LNS ObjectServer COM Component 才能利用 LNS Server 来安装管理网络, 必须要通过 LNS ObjectServer COM Component 才能利用 DataServer 来监控网络。最后让我们来看看 LNS 的结构框图。

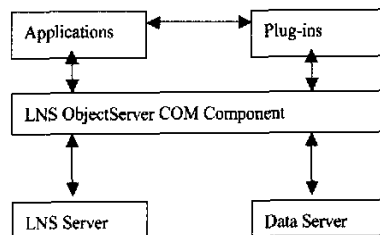


图 1 LNS 结构框图

对 LNS 有了一个比较清晰的认识之后。在正式编程之前, 让我们先来了解一下 COM。

3 COM 组件编程

COM 是微软公司推出的软件编程规范。它是一个和语言无关的软件组件编程模型, 它使得运行在 Windows 平台上的软件组件能与应用程序实现交互。COM 的关键特性在于通过定义清晰的接口使得客户与服务器之间能够通讯。接口提供了一个方法用于客户在运行时查询服务器所支持的功能。如果服务器还要提供其他的功能, 只需要在服务器上添加与其功能相对应的接口即可。

应用程序可以访问位于同一台计算机上的 COM 组件接口, 也可以通过 DCOM 来访问其它计算机上的组件接口。COM 技术也是 ActiveX, ASP 技术的基础。

用 ActiveX 控件可以很容易的访问服务器的接口, 调用接口的方法。这种方法用的是 COM 的调度接口 (Dispatch Interface), 虽然使用起来很方便, 但与直接使用 COM 的虚函数表接口这种方式相比, 速度较慢。调度接口主要是供 VB, Java 等不支持指针的语言使用的接口, 对于 C++ 这种语言, 既可以使用调度接口, 也可以使用虚函数表 (Vtable) 接口。

4 LNS 下的网络监控

LNS 提供了两种系统级的网络监控方式,一种是监控集合,一种是监控点。监控集合一建立就可以永久存在,下次仍然可以使用,因为它一经建立就被保存在 Data Sever 中。而监控点则不会永久存在,下次使用需要重建。由此可见,监控集合这种方式适合于监控任务比较频繁的场合,而监控点这种方式适合监控任务较轻,点数较多的情况下。

无论是监控集合或是监控点,他们都要用到监控点。监控点有如下几种监控方式:显式读写,隐式绑定,查询,显示绑定。

显示读写这种方式下,客户可以直接读取和写入监控点的 Value 属性。隐式绑定这种方式下,可以将上位机的一个监控点与节点上的网络变量绑定,当网络变量变化时,会以 NvMonitorUpdateEvent 通知客户,客户获取此事件的句柄就可以取得更新过的数据。当使用 poll 即查询方式时,无论被监控值是否变化,隔一段时间将会触发一个 NvMonitorUpdateEvent 事件,用户同样可以用这个事件句柄来取得监控点的数据。当使用显式连接这种方式时,先在主机上建立一个网络变量,这个网络变量可以连接多个监控点,从而达到一对多监控这种效果。在某些场合,需要用到显式连接这种方式,如消防,任何一个探测头有情况都应该立刻反映出来。

综上所述,查询这种方式适合于那些监控任务比较频繁,但对数值的变化不太敏感的场合。隐式绑定适用于监控任务不频繁,但数值一变化就要马上知道的场合,如安防等。

5 用 COM 来实现网络监控

由于监控的特殊性,网络监控软件需要一个相当快的响应速度,被监控点的状态变化要能马上反映到上位机上,以利于决策者尽快作出决策。

现在很多 LonWorks 网络监控软件都用 Echelon 提供的 lnsobjectserver 这个 ActiveX 控件进行开发,虽然开发周期短,开发很方便,但由于是使用调度接口,在调用其方法前需要绑定,所以速度比较慢。

为实现最快的响应速度,有必要直接用最低层的 COM 虚函数表接口来开发网络监控软件。

本人的开发环境为 C++ Builder 5.0,所使用的类型库为 lcaobjsv.ocx,有了这两个东西就可以进行开发了。



图2 lcaobjsv.ocx 被类型库编辑器导入时的界面

首先我们需要导入类型库,导入类型库之后就可以看到 LNS 所提供的接口了(见图2),在这些接口中 _DlcaObjectServer 为主接口,我们可以看到这个接口里有很多建立网络所要用的方法和属性,要使用这些方法和属性,就要先获取这个接口的指针。可初始化它工厂类获得这个接口的指针,具体代码如下:

```
CoInitialize(NULL); //工厂类初始化
_DlcaObjectServer *pobject=NULL; //定义工厂类上接口
//_DlcaObjectServer 的接口指针 pobject
HRESULT hr = CoCreateInstance(CLSID_LcaObjectServer,
NULL,CLSCTX_INPROC_SERVER,IID__DlcaObjectServer,
(void**)&pobject);
if(SUCCEEDED(hr))//判断成功与否
ShowMessage("success to hold the handle of lcaobjectserver");
如果返回成功信息则表明成功取得 _DlcaObjectServer 接口
的指针 pobject,接下来通过 pobject 来调用 _DlcaObjectServer 接
口的方法和属性。
pobject->set_SingleUserMode(false); //设为多人模式
pobject->set_RemoteFlag(false); //本地模式
if(pobject->get_IsOpen()==false)//如果 lcaobjectserver 的 global
数据库没打开,则打开它
pobject->Open();
IDispatch *p=NULL;
p=pobject->get_Networks(); //取得 networks 的接口指针
如果要用到另外一个接口,比如我现在想增加一个网络,而
_DlcaObjectServer 接口里没有这个方法,那么我需要转到其它接
口中,这个接口为 ILcaNetworks,这时可以用 QueryInterface 这个
方法。每一个接口都支持 QueryInterface 方法。这样,只要取得一
个接口指针,我就可用 QueryInterface 来取得其它接口的指针。
ILcaNetworks *p1=NULL;
HRESULT hr1=p->QueryInterface (IID_ILcaNetworks,(void**)&p1); //取得 ILcanetworks 的指针 p1
现在,我就可以通过这个指针 p1 来调用 ILcaNetworks 的
Add 方法来增加一个网络。
OLECHAR FAR *a=L"lei22";
OLECHAR FAR *b=L"e:lei22";
IDispatch *p2;
p2=p1->Add(a,b,true); //增加网络
用类似的方法还可以取得类型库里定义的其它接口,调用
其它方法和属性来进行组网和监控,在这里就不一一列举了。
上面的例子虽然是在 C++ Builder 环境下开发的,但由于
COM 只是一种编程规范,所以也适合在 Visual C++ 下开发,事实
上,只要所用开发语言支持指针,支持虚函数,都可以用这种语
言,采用本文所述的方法,直接通过虚函数表接口来调用方法和
属性。
```

6 结束语

介绍了 LNS 的编程架构,讨论了在 LNS 环境下网络监控的各种方式,并比较了其优缺点。本文还介绍了一种直接利用最底层的 COM 接口的方法来开发网络监控软件的方法,与采用控件开发的软件相比,在响应速度方面有比较大的优势。

参考文献:

[1]COM 技术内幕.[美]Dale Rogerson 著.杨秀章,江英译.1997年.

[2]Lon 控制网络及应用.杨育红著.1999年.

作者简介:雷刚,出生年月:1977年9月,现在合肥工业大学检测技术研究所,在读硕士研究生,研究方向:现场总线在检测技术中的应用。tel:0551-2901511

(230009 合肥工业大学仪器仪表学院检测技术研究所)

雷刚 吕国强 洪占勇

(收稿日期:2002.6.25)