

基于软组件技术的 LonWorks 现场总线 网络管理软件研究

卫立新 邢建春 王 平 解放军理工大学工程兵工程学院国防工程自动化研究所(210007)

Abstract

In this paper, we discuss the COM component technology. Then at the basis of analyzing the function of LonWorks Fieldbus Network Management Software, we design and implement the components providing the specified services. Finally, with the help of detailed examples, we provide several methods in which client's programs access components in VB.

Keywords: fieldbus, component, network management, COM, CBD, ActiveX

摘 要

本文首先阐述了 COM 组件技术, 然后在对 LonWorks 现场总线网络管理软件功能分析的基础上, 设计和实现出了针对各种功能服务的 COM 组件。最后结合具体例子介绍了在 VB 环境下客户程序访问组件对象的几种方法。

关键词: 现场总线, 组件, 网络管理, COM, CBD, ActiveX

0 引言

现场总线网络管理软件是对具体的现场总线网络系统进行统一管理和配置的软件系统, 主要提供了对现场网络运行调试、故障诊断及测控模块(节点)的工作模式设置、功能组态、运行监视等功能。对特定的现场总线系统需要开发特定的网络管理软件。

传统的现场总线网络管理软件在设计和开发时通常被分割成文件、模块或类, 然后被编译并链接成一个铁板一块的二进制文件。采用这种方法开发出的管理软件缺乏开放性和可扩展性, 随着时间的推移其会日益“老化”。如果在设计和实现现场总线网络管理软件时将其分割成不同的组件, 那么随着技术的发展我们可以用新的组件取代已有的组件, 这样会给已有的现场总线网络管理软件不断注入新的活力, 使之趋于完善。

1 LonWorks 网络管理软件实现的功能

由于 LonWorks 总线技术相对于其它现场总线的诸多优点, 我们研制了基于 LonWorks 的 893—LM 系列分布式测控网络产品, 主要应用于智能建筑, 工业控制等领域。893—LM 网络管理软件一般运行于 Windows 95/98/Me 和 Windows NT/2000 操作系统, 主要用于网络运行调试、故障诊断及测控模块(节点)的工作模式设置、功能组态、运行监视等。具体功能如下:

1) 网络管理, 主要进行网络检测, 查看网络通信有无故障。

2) 网络变量管理, 主要包括网络变量浏览和网络变量连接。

3) 路由器管理, 主要包括路由器安装、路由器替换和路由器删除。

4) 节点管理, 在网络管理软件中, 我们对节点的管理主要包括:

- 自动搜索网络中的节点, 并将节点按类型列表显示。
- 节点基本参数显示: 对于网络中的节点, 能自动显示其基本参数, 例如: 节点的地址号、产品序列号、芯片 ID 号, 型号, 电子标签等。

· 节点通道模式设置。节点一般有模拟量和开关量两种通道类型。模拟量通道的工作模式一般有直流电压、电阻、热电阻和热电偶四种方式。根据需要还可以进行相应的函数运算处理及越限报警设置等。开关量通道的工作模式一般有计数和测频两种方式。

· 节点组态程序的上载、下载。节点组态程序主要是指节点各个通道的工作模式, 组态程序的上载是指将节点各个通道的工作模式保存到网络数据库; 组态程序的下载是指将节点各个通道的工作模式从网络数据库提取出来, 发送到节点各个通道。

· 节点运行状态监视。对于模拟量节点, 我们可以查看所有通道的采集数值及通道工作是否正常; 还可以查看指定通道的采集数值及工作状况。对于开关量节点, 我们可以查看通道当前是处在开状态还是闭状态。

2 LonWorks 网络管理软件的 COM 组件设计

CBD (Component Based Development, 基于组件的软件开发技术) 将一个软件的开发分为三个阶段: 应用系统的分析和设计; 组件的开发及组件的装

配。其中,系统的分析和设计是一项系统工程。首先根据应用需求建立系统模型,然后由分析及设计人员按照组件开发规则来定义系统所需的所有组件。组件开发也叫组件生产,可以重新设计,可以将现有的软件封装成组件,也可以从外界(市场或互联网)直接获得。组件的装配则按照应用系统设计中提供的结构,从组件库中选取合适的组件用适当的编程工具完成应用系统的连接与合成,最后对系统进行各种测试,这种开发过程如图1所示。

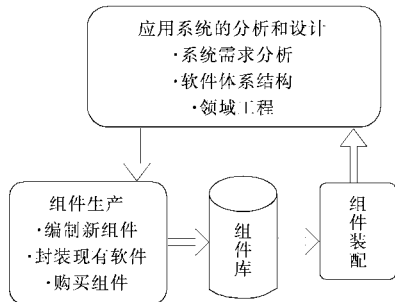


图1 组件化软件开发模型

通过对上述网络管理软件的功能分析,我们可以开发出三种组件:节点组件,路由器组件,网络变量组件,如图2所示。下面我们以节点组件为例来说明组件的开发过程。

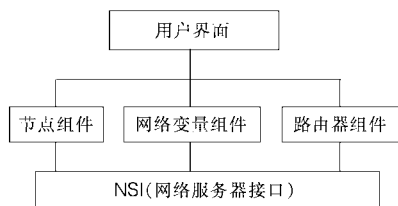


图2 LonWorks网络管理软件的组件类型

节点组件用来访问节点信息和对节点进行各种设置。每一类节点对应一种节点组件,不同类型的节点有不同的节点组件。在节点组件中只包含一个COM对象,即节点对象。它主要有三个接口:节点基本参数接口;节点工作模式接口;节点通道状态接口。每个接口是一组逻辑上相关的函数集合(属性或方法)。如图3所示。

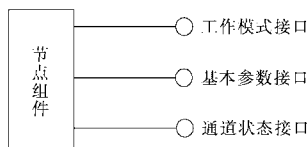


图3 节点组件的接口类型

节点基本参数接口:用来访问节点的基本参数,例如:地址号、序列号、型号,电子标签等。

节点工作模式接口:用来访问节点的工作模式,有两种类型:设置节点的通道工作模式;读取节点的通道工作模式。

节点通道状态接口:用来访问节点的通道状态,例如:通道的当前数值,通道数值是否越限;通道工作是

否正常等。

虽然COM组件具有语言无关性,但COM所使用的一些数据类型在VB或者更弱的高级语言(如VBA或者SCRIPT语言)中很难表达,使得这些高级语言很难使用COM组件,自动化技术的出现为这些高级语言直接使用组件模块提供了一条有效的途径。

自动化技术建立在COM技术基础之上,但它简化了COM的一些底层操作。自动化技术主要是通过IDispatch接口来体现的。所谓的自动化对象,实际上就是实现了IDispatch接口的COM对象。自动化对象的核心就是IDispatch接口。IDispatch接口有四个成员函数:GetTypeInfoCount;GetTypeInfo;GetD-OfNames和Invoke。其中Invoke成员函数是IDispatch接口的关键函数,客户程序必须通过Invoke函数才能访问属性或方法。

因此,要想在VB等弱的高级语言中使用组件模块,组件就必须实现IDispatch接口,即组件的各种接口必须从IDispatch接口继承而来。

由于组件在设计过程中相互独立,因而为它们的重复使用奠定了良好的基础。需要注意的是:组件只是能够提供应用服务的一个部件,需要在应用环境中获取各种参数,由应用环境来驱动。对组件的编写我们通常采用VC++语言,因为VC++提供的工程向导使我们能很容易地开发出符合COM规范的组件。

组件开发完毕后,将开发的组件进行安装和注册,以后用户在利用编程工具(VB或VC++)完成网络管理软件开发的过程中,可根据需求直接引用合适的组件。

3 基于组件技术的LonWorks网络管理软件的实现

LonWorks网络管理软件的开发主要包括两部分:前台用户界面的设计和后台应用程序的编写(见图4)。虽然VB和VC++都可以用来开发网络管理软件,但是选用VB作为开发工具更好一些,除了VB界面设计方便外,更主要的原因是VB对自动化COM组件的访问非常方便。

3.1 设计前台用户界面。

良好的用户界面是一个好的软件所必不可少的。因此在开发网络管理软件时,必须对用户界面的设计加以重视。例如对节点,我们需要设计出如下的用户界面:基本参数窗口,工作模式窗口,通道状态窗口等。在设计用户界面时,我们发现许多窗体中控件的使用几乎完全一致。例如对于各种类型的节点,其基本参数窗口中都包括许多Label控件和Textbox控件,用来显示节点的基本参数名称和数值。对于这些窗体,如果我们按照通常的方法去设计,那么需要花费大量的时间去进行重复性的工作;如果我们能开发一些自定义的ActiveX控件来

封装那些在界面设计时重复使用的控件组合,那么在设计时就可以大大简化用户界面的开发工作。

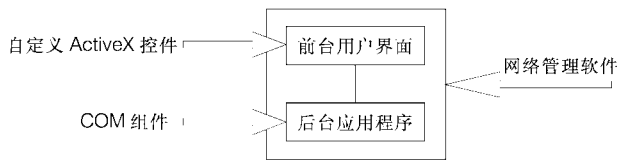


图 4 网络管理软件的组件实现

ActiveX 控件是一门综合技术,它涉及到 COM 和 OLE 的许多技术精华,是应用最为广泛的 COM 组件。它与普通 COM 组件的主要区别在于具有用户界面特性,通常是一个 DLL 或 OCX 程序。

例如为了查看节点的基本参数,我们可以开发出名为 NodeBase 的自定义 ActiveX 控件(用 VB 作为开发工具):在 NodeBase 控件的 UserControl 上添加一些 Label 控件和 Textbox 控件,Label 控件用来标注节点的基本参数名称,例如地址号、序列号、型号、芯片 ID 号、电子标签等,对应的 Textbox 控件则用来显示具体的参数值,同时为 NodeBase 控件添加一些属性,分别映射到各 Textbox 控件的 text 属性。具体开发过程在此不再叙述。

ActiveX 控件开发完毕并进行安装和注册后,就可以引用到应用工程中。用户在设计程序界面时,将这些 ActiveX 控件置于窗体容器上,然后修改控件的属性以满足设计要求。

在设计应用程序界面时采用将多种控件封装成一个 ActiveX 控件的处理方法大大简化了用户界面的设计工作、缩短了开发时间,也有效保证了软件的健壮性。

3.2 编写后台应用程序。

由于采用了 COM 组件技术,使得网络管理软件的应用程序编写非常方便,只需要掌握如何访问组件的接口就可以很容易地完成编写工作。

客户程序访问组件对象一般有两种方式:早绑定和晚绑定。如果客户程序在编译时能找到组件对象的方法信息,这种技术叫作“早绑定”;如果客户程序在编译时不知道组件对象的信息,而是在运行过程中查询组件对象,找到它所支持的方法和所需要的参数,这种技术叫作“晚绑定”。

VB 是通过晚绑定方式访问 COM 对象的,有三种方法。下面我们基于 VB 的客户程序访问节点组件 node.dll 为例,来说明这三种方法的区别。(假设 node.dll 的基本参数接口为 INode1)

1) 晚绑定与早绑定相比效率较低,因为后者在编译时就确定了接口的地址。但是可以使用如下语句来优化晚绑定,以允许 VB 在编译时而不是运行时就查看接口的 ID。

```
Dim nd1 as new INode1
Dim j as long
j = nd1.addr
```

访问节点的地址

2) 如果编译时还不知道对象的类型,比如由用户的动作来确定使用对象的类型,那么就必须使用下面这种方法,它强制 VB 在运行时根据需要寻找 COM 对象的接口的 ID。

```
Dim nd1 as object
Set nd1 = New INode1
Dim j as long
j = nd1.addr
```

3) 访问 COM 对象时,甚至可以不必要在工程中引用组件,只需在创建语句中指定它的 ProgID (Program Identifier, 程序标示符)即可。下面代码中,参数 Nodesample.Node1.1 是接口 INode1 的 ProgID。

```
Dim nd1 as object
Set nd1 = CreateObject ("Nodesample.Node1.1")
Dim j as long
j = nd1.addr
```

在以上三种情形中,客户程序都不必关心 COM 对象的释放,VB 会自动为你完成这一工作。

4 结束语

组件的二进制兼容性、平台独立性和网络透明性,使得其可以用于各种不同的软硬件环境;用组件封装好业务逻辑并为各种客户程序所调用,这已成为当前软件开发的一种模式,这种思想使得软件的设计、扩展和维护非常方便。在 LonWorks 现场总线网络管理软件中引入组件技术,将相对独立的功能单元以组件的形式进行封装,使得该软件设计思路更加清晰,有利于软件开发过程中的工作分工,优化了该管理软件的结构,提高了开发效率。

参考文献

- 1 吕涌,皇甫正贤.组件化结构的组态软件研究与开发.工业控制计算机,2000(5)
- 2 邢建春,王平,仲未央,等.工业控制软件互操作标准 OPC 综述.工业控制计算机,2000(1)
- 3 吴刚,孙家启.基于组件的 WEB 数据库互连研究.微型机与应用,2000(11)
- 4 卿芳会,汤荷美,黄维通.COM/DCOM 技术的结构及特点分析.计算机应用与软件,2000(7)
- 5 王双庆,邢建春.OPC—现代工业控制软件的互操作标准.测控技术,2000(3)
- 6 阳宪惠.现场总线技术及其应用.清华大学出版社,1999
- 7 杨育红.LON 网络控制技术及应用.西安电子科技大学出版社,1999
- 8 潘爱民.COM 原理与应用.清华大学出版社,1999
- 9 余英,梁刚.Visual C++ 实践与提高——COM 和 COM+ 篇.中国铁道出版社,2001

[收稿日期:2002.2.2]