



**Design and Development  
of Configuration Software**

# 组态软件设计与开发

王亚民 陈 青 刘畅生 王水平 编著



西安电子科技大学出版社

<http://www.xduph.com>

# 组态软件设计与开发

王亚民 陈 青 刘畅生 王水平 编著

西安电子科技大学出版社

2003

## 内 容 简 介

本书介绍使用 Visual C++ 开发一个实用的组态软件系统, 并按照组态软件的基本结构完成开发的全过程, 内容包括组态软件的基本概念、组态软件 HMI 矢量绘图、网络实时通信、语音报警、串口通信、数据库访问、ActiveX 控件开发及联机帮助设计等。为方便读者学习, 本书配有光盘一张。

本书注重理论分析、设计和实际编程实现相结合, 内容具有很强的实用性, 既可供 Windows 软件开发人员参考, 也可用作各类院校计算机、仪器与检测等相关专业的教学参考书。

本书主要面向相关中、高级专业人员, 要求读者了解 C++ 语言和 MFC 的有关知识。

### 图书在版编目 (CIP) 数据

组态软件设计与开发 / 王亚民等编著.

—西安: 西安电子科技大学出版社, 2003.4

ISBN 7-5606-1212-1

I. 组… II. 王… III. 软件开发 IV. TP311.5

中国版本图书馆 CIP 数据核字 (2003) 第 017030 号

责任编辑 云立实

出版发行 西安电子科技大学出版社 (西安市太白南路 2 号)

电 话 (029)8227828 邮 编 710071

<http://www.xduph.com> E-mail: [xdupfxb@pub.xaonline.com](mailto:xdupfxb@pub.xaonline.com)

经 销 新华书店

印 刷 陕西光大印务有限责任公司

版 次 2003 年 4 月第 1 版 2003 年 4 月第 1 次印刷

开 本 787 毫米×1092 毫米 1/16 印张 15.375

字 数 357 千字

印 数 1~4 000 册

定 价 28.00 元 (含光盘)

ISBN 7-5606-1212-1 / TP·0632

XDUP 1483001-1

\*\*\* 如有印装问题可调换 \*\*\*

# 前 言

工业控制组态软件的使用和开发已经有了几十年的历史,现阶段每个从事工业控制的公司,几乎都有自己的硬件产品和组态软件。早期的工业控制组态软件大多是基于 UNIX 和 DOS 平台的,随着 Windows 系列操作系统的普及和发展,越来越多的工业控制组态软件都选择基于 Windows 平台开发。然而,由于工业控制组态软件对开发者的知识面的要求较高,加之工业控制组态软件的开发一直是一个较新的领域,因而图书市场上很少有关于工业控制组态软件的著作发行。本书根据作者多年的工作经历,将开发、设计工业控制组态软件的经验介绍给读者,期望对读者有所帮助和启发。

本书共有 10 章,各章内容分别如下。

第一章介绍组态软件的基本概念、基本结构和开发环境。

第二章介绍开发平台 Microsoft Visual C++6.0 与 MFC 类库编程方法,并用 Visual C++6.0 生成组态软件基本框架。

第三章介绍组态软件的 HMI 矢量制图。一般地,组态软件都具有开发态和运行态,本章主要介绍组态软件开发态的设计方法。

第四章介绍组态软件的动画处理。组态软件的 HMI 要能够反映控制现场的 AI/AO、DI/DO 状态,如何建立现场和画面的对应关系是本章的主要内容。

第五章介绍组态软件的通信,包括板卡设备通信、串口设备通信、网络实时通信。板卡设备通信和串口设备通信用于连接现场设备,网络实时通信构成监控实时网。

第六章介绍组态软件的 ActiveX 控件设计,包括 ActiveX 控件的基本概念和作者设计的仪表 ActiveX 控件 wyMeter.ocx。

第七章介绍组态软件的语音处理,包括语音处理的基本原理和基本语音处理类。

第八章介绍组态软件的数据库操作,包括实时数据库和后台数据库的处理。

第九章介绍组态软件的联机帮助设计。

第十章介绍作者开发的一种实用组态软件。

本书配有一张光盘,其中包括了书中介绍的所有源程序。本书中的所有程序在 Visual C++6.0 和 Windows 2000 环境下调试通过,源程序也可以在 Visual C++.NET 下编译调试。

从本书的酝酿到编写,始终得到了西安电子科技大学出版社云立实老师的关怀和指导,在此表示衷心的感谢。

限于作者水平所限,书中难免有不足之处,恳请读者批评指正。作者的 E-mail 地址是: yamwang@xidian.edu.cn。

作 者

2003 年 1 月

# 目 录

|                                     |    |
|-------------------------------------|----|
| 第一章 组态软件概述.....                     | 1  |
| 1.1 组态软件的基本概念.....                  | 1  |
| 1.1.1 组态软件的发展背景.....                | 3  |
| 1.1.2 组态软件的发展趋势.....                | 4  |
| 1.2 组态软件的基本结构.....                  | 6  |
| 1.2.1 组态软件的基本结构.....                | 6  |
| 1.2.2 组态软件的数据处理流程.....              | 9  |
| 1.2.3 组态软件的特点.....                  | 9  |
| 1.2.4 组态软件的性能.....                  | 10 |
| 1.2.5 组态软件在监控系统中的地位.....            | 11 |
| 1.3 组态软件开发环境.....                   | 11 |
| 1.3.1 组态软件开发环境选择.....               | 11 |
| 1.3.2 组态软件的一般使用步骤.....              | 12 |
| 第二章 Visual C++6.0 简介.....           | 14 |
| 2.1 Visual C++6.0 开发环境.....         | 14 |
| 2.1.1 开发环境简介.....                   | 15 |
| 2.1.2 MFC 类库介绍.....                 | 15 |
| 2.2 MFC 类库的基本类结构.....               | 22 |
| 2.2.1 CObject 类.....                | 22 |
| 2.2.2 应用程序结构类.....                  | 23 |
| 2.2.3 可视对象类.....                    | 23 |
| 2.2.4 绘图和打印类.....                   | 25 |
| 2.2.5 通用类.....                      | 26 |
| 2.2.6 数据库类.....                     | 27 |
| 2.2.7 Internet 和网络类.....            | 27 |
| 2.2.8 OLE 类.....                    | 27 |
| 2.3 用 Visual C++6.0 生成组态软件基本框架..... | 27 |
| 2.3.1 创建一个新的 ProjectWorkSpace.....  | 28 |
| 2.3.2 利用 AppWizard 创建程序框架.....      | 29 |
| 2.4 组态软件扉页制作.....                   | 35 |
| 第三章 组态软件的矢量制图.....                  | 37 |
| 3.1 矢量制图系统相关的基础知识.....              | 38 |
| 3.1.1 矢量图.....                      | 38 |

|                                         |            |
|-----------------------------------------|------------|
| 3.1.2 基本结构 .....                        | 38         |
| 3.1.3 存储的设计 .....                       | 38         |
| 3.1.4 图元绘制的基本实现方法 .....                 | 39         |
| 3.2 绘图程序相关的主要类及函数 .....                 | 43         |
| 3.2.1 CView 类和 CScrollView 类及成员函数 ..... | 43         |
| 3.2.2 CDocument 类及成员函数 .....            | 46         |
| 3.2.3 CDC 类和 CClientDC 类及成员函数 .....     | 48         |
| 3.2.4 CDialog 类及成员函数 .....              | 51         |
| 3.3 图元基类 CDrawObj .....                 | 53         |
| 3.3.1 CDrawObj 的构造函数 .....              | 54         |
| 3.3.2 CDrawObj 的串行化函数 .....             | 55         |
| 3.3.3 CDrawObj 的矩形跟踪函数 .....            | 55         |
| 3.3.4 CDrawObj 的热点句柄获取 .....            | 56         |
| 3.3.5 CDrawObj 的属性编辑对话框 .....           | 57         |
| 3.4 矩形作图类 CDrawRect .....               | 59         |
| 3.5 OLE 对象作图类 CDrawOleObj .....         | 61         |
| 3.6 绘图工具基类 CDrawTool .....              | 63         |
| 3.7 矩形绘图工具类 CRectTool .....             | 63         |
| 3.8 图元属性选择类 CSelectTool .....           | 64         |
| <b>第四章 组态软件的动画处理 .....</b>              | <b>67</b>  |
| 4.1 组态软件系统配置 .....                      | 67         |
| 4.2 监测点参数配置 .....                       | 68         |
| 4.3 动画连接 .....                          | 70         |
| 4.4 表达式处理 .....                         | 73         |
| 4.4.1 表达式类 CExpression .....            | 73         |
| 4.4.2 表达式类的使用 .....                     | 88         |
| <b>第五章 组态软件的通信 .....</b>                | <b>89</b>  |
| 5.1 组态软件通信软总线 .....                     | 89         |
| 5.2 板卡设备通信 .....                        | 90         |
| 5.3 串口设备通信 .....                        | 90         |
| 5.3.1 Win32 通信函数介绍 .....                | 90         |
| 5.3.2 Windows API 串口通信编程示例 .....        | 91         |
| 5.4 网络实时通信 .....                        | 111        |
| 5.4.1 TCP/IP 协议介绍 .....                 | 111        |
| 5.4.2 网络应用编程接口的原理 .....                 | 115        |
| 5.4.3 网络编程接口 Windows Socket .....       | 117        |
| 5.4.4 WinSock 程序设计实例 .....              | 132        |
| <b>第六章 组态软件的 ActiveX 控件设计 .....</b>     | <b>136</b> |
| 6.1 ActiveX 控件简介 .....                  | 136        |

|            |                        |            |
|------------|------------------------|------------|
| 6.1.1      | ActiveX 控件             | 136        |
| 6.1.2      | ActiveX 控件容器           | 136        |
| 6.1.3      | 事件                     | 137        |
| 6.1.4      | 方法                     | 138        |
| 6.1.5      | 属性                     | 138        |
| 6.1.6      | 持久性和串行化                | 139        |
| 6.1.7      | 属性页                    | 140        |
| 6.1.8      | 许可                     | 140        |
| 6.2        | ActiveX 控件的基本组成        | 140        |
| 6.2.1      | 基类 COleControl         | 140        |
| 6.2.2      | 事件激发函数                 | 140        |
| 6.2.3      | 派遣映射表                  | 141        |
| 6.3        | wyMeter 控件的设计          | 141        |
| 6.3.1      | 创建基本的程序框架              | 141        |
| 6.3.2      | 程序框架组成                 | 144        |
| 6.4        | wyMeter 控件的属性和方法       | 152        |
| 6.4.1      | 创建控件属性                 | 152        |
| 6.4.2      | 添加方法                   | 156        |
| 6.4.3      | 绘制控件                   | 159        |
| 6.5        | wyMeter 控件的属性页         | 167        |
| 6.6        | wyMeter 控件在组态软件中的应用    | 169        |
| 6.6.1      | 测试控件                   | 169        |
| 6.6.2      | 组态软件中使用 wyMeter 控件     | 172        |
| <b>第七章</b> | <b>组态软件的语音处理</b>       | <b>173</b> |
| 7.1        | 语音处理的基本原理              | 173        |
| 7.1.1      | 高层音频服务                 | 173        |
| 7.1.2      | 低层音频服务                 | 176        |
| 7.2        | 实时语音报警类                | 181        |
| 7.2.1      | 音波处理类 CWave            | 181        |
| 7.2.2      | 音波输出设备类 CWaveOutDevice | 183        |
| 7.2.3      | 语音处理窗口类 CWaveDevWnd    | 187        |
| 7.3        | 语音参数的设定和处理             | 189        |
| 7.3.1      | 语音参数的设定                | 189        |
| 7.3.2      | 语音报警在组态软件中的处理          | 189        |
| <b>第八章</b> | <b>组态软件的数据库操作</b>      | <b>192</b> |
| 8.1        | 数据库的构造                 | 192        |
| 8.1.1      | 关系数据库的概念               | 192        |
| 8.1.2      | 关系数据库的基本术语             | 194        |
| 8.1.3      | SQL 语言简介               | 194        |

|                               |            |
|-------------------------------|------------|
| 8.2 ODBC 简介 .....             | 195        |
| 8.2.1 ODBC .....              | 195        |
| 8.2.2 ODBC 驱动程序监管器 .....      | 196        |
| 8.2.3 ODBC 驱动程序管理器 .....      | 196        |
| 8.2.4 ODBC 驱动程序 .....         | 197        |
| 8.2.5 ODBC API 的编程顺序 .....    | 197        |
| 8.3 创建 ODBC 数据源 .....         | 199        |
| 8.3.1 VC++ 中使用 ODBC 的方法 ..... | 199        |
| 8.3.2 创建 ODBC 数据源 .....       | 199        |
| 8.4 数据库操作 .....               | 203        |
| <b>第九章 组态软件的联机帮助 .....</b>    | <b>206</b> |
| 9.1 联机帮助的主要功能 .....           | 206        |
| 9.1.1 联机帮助的形式 .....           | 206        |
| 9.1.2 联机帮助的使用方式 .....         | 206        |
| 9.1.3 建立带联机帮助的应用程序框架 .....    | 208        |
| 9.1.4 联机帮助的设计思路 .....         | 209        |
| 9.2 联机帮助的机理及文件格式 .....        | 209        |
| 9.2.1 联机帮助的机理 .....           | 209        |
| 9.2.2 超文本文件 .....             | 210        |
| 9.2.3 RTF 文件 .....            | 210        |
| 9.2.4 CNT 文件 .....            | 211        |
| 9.2.5 HPJ 文件 .....            | 212        |
| 9.3 联机帮助的实现 .....             | 215        |
| 9.3.1 创建带有联机帮助的应用程序 .....     | 215        |
| 9.3.2 编写 CNT 目录文件 .....       | 216        |
| 9.3.3 编写 RTF 文件 .....         | 216        |
| 9.3.4 编译 HPJ 文件 .....         | 217        |
| 9.4 联机帮助的编译运行 .....           | 218        |
| 9.5 HTML 帮助 .....             | 218        |
| <b>第十章 光盘演示系统说明 .....</b>     | <b>220</b> |
| 10.1 系统概要 .....               | 220        |
| 10.1.1 系统说明 .....             | 220        |
| 10.1.2 系统的安装 .....            | 221        |
| 10.1.3 系统的启动 .....            | 222        |
| 10.1.4 系统配置方式 .....           | 222        |
| 10.2 组态软件使用说明 .....           | 223        |
| 10.2.1 组态主画面介绍 .....          | 223        |
| 10.2.2 文件菜单说明 .....           | 225        |
| 10.2.3 编辑菜单说明 .....           | 226        |

|                             |            |
|-----------------------------|------------|
| 10.2.4 查看菜单说明 .....         | 227        |
| 10.2.5 对象菜单说明 .....         | 228        |
| 10.2.6 终端菜单说明 .....         | 228        |
| 10.2.7 系统数据菜单说明 .....       | 230        |
| 10.2.8 参数菜单说明 .....         | 231        |
| 10.2.9 系统菜单说明 .....         | 234        |
| 10.2.10 如何设计画面并建立动画连接 ..... | 234        |
| <b>参考文献</b> .....           | <b>235</b> |

# 第一章

## 组态软件概述

### 1.1 组态软件的基本概念

组态软件是监控系统的指挥中心,是监控系统发展的产物。监控系统经历了下面五种发展形式。

(1) 集中式监控系统。集中式监控系统以单板机为控制装置,与上位机通过串行口相连,所有模块集中在若干个机柜中。这类系统技术比较落后,不符合分布、开放趋势。

(2) 基于模拟仪表的监控系统。这类系统以“集散式控制系统 DCS”为理论基础,采用成熟的模拟仪表,以“传感器、变送器、控制器和执行机构”为系统模型构成。一个变送器或执行机构用一对传输线来单向传送一个  $4\sim 20\text{ mA}$  或  $1\sim 5\text{ V}$  模拟信号。控制器可以是工控机(IPC)、可编程控制器(PLC)、直接现场控制器(DDC)和其他 A/D 采集设备。

DCS(Distributed Control System)主要有四层结构: I/O 层、控制器层、人机接口层和信息系统层。后两层之间采用以太网,其他层采用专用网络、专用控制设备、专用软件。这类系统开放性不足,系统维护和升级不便。

传统 PLC 由电源模块、CPU 模块、I/O 模块、槽板及扩展机架组成,使用自定义梯形图逻辑语言编程。I/O 模块和扩展机架通过槽板的自定义封闭总线连接或扩展,开放性不足。

先进 PLC 的优点是:支持远程 I/O 及现场总线网络,具有向下分布式 I/O 能力;支持高速网络(如以太网等),具有向上高速连接能力;支持 OPC(Object Linking and Embedding for Process Control)标准,具有软件开放性。PLC 比 DCS 便宜,但可靠性和电气性能不如 DCS。

(3) 基于智能仪表的监控系统。智能仪表是集遥测、遥控、遥信于一体的多功能监控装置,由软件算法和硬件电路组成,具有完备的通信联网功能,能通过网络传输数据。

该类系统一般由监控主机、CPU 模块、I/O 模块、智能仪表、带电执行机构、MODBUS 网络等组成。

(4) 基于现场总线的监控系统。现场总线控制系统(FCS)运用数字通信、计算机、自动控制、网络、智能仪表等先进技术,突破传统“点对点”模拟信号控制的局限性,具有全分散、开放性、全数字化、智能、双向、互联、多变量、多接点、多分支等特点。现场总线标准有 Profibus、CAN、Lonworks、WorldFip、FF 等。智能仪表、现场总线是 FCS 的技术关键。

现场总线智能断路器具有计算、保护、控制及对话功能,它与监控主机双向通信可构

成智能化的监控、保护、信息网络系统。

(5) 基于计算机网络的监控系统。虽然 PLC、DCS、FCS 也采用了工控机、OPC 和以太网技术,但由于这些技术被镶嵌在传统的系统结构中,因此,它们的基本构成无本质改变。

基于计算机网络的监控系统采用“TCP/IP 以太网协议标准”,系统变得结构清晰、简单,开放性强,建模和仿真容易。在该类系统中,直接将 I/O 设备连到以太网或连到与以太网兼容的 I/O 数据集中器上,保留以太网的物理层及数据链路层协议,应用层使用现场总线定义的协议。

现代的以太网采用星型连接及交换式 Hub。交换式 Hub 提供数据缓冲并具有确定接收数据的网段智能逻辑,使数据冲撞及重发机会最小化。千兆以太网的发展及网络冗余技术使以太网更适应于实时应用。以太网控制器不仅可以简便地接入,与主控室的计算机进行控制信息交换,并能在企业 Intranet 上进行类似 PLC 的“硬实时”控制。工业以太网日渐可靠并有取代现场总线的趋势。

“组态”的概念是伴随着集散式控制系统(Distributed Control System, DCS)的出现才开始被广大的生产过程自动化技术人员所熟悉的。

组态软件(有时也称为监控组态软件或工控组态软件)为自动化工程技术人员提供了一种采用搭积木的方式制作现场控制过程和控制界面的工具。

20 世纪 70 年代中期,随着微处理器的出现,诞生了第一代 DCS。到目前,DCS 和其他控制设备在全球范围内得到了广泛应用。计算机控制系统的每次大发展和更新换代,都是以微处理器技术、超大规模集成电路技术、计算机网络技术和计算机软件技术的发展为基础的。

由于每一套 DCS 都是比较通用的控制系统,可以应用到很多的领域中,为了使用户在不需编写代码程序的情况下,便可生成适合自己需求的应用系统,每个 DCS 厂商在 DCS 中都预装了系统软件和应用软件。而其中的应用软件,实际上就是组态软件,但一直没有人给出明确定义,只是将使用这种应用软件设计生成目标应用系统的过程称为“组态(configure)”。

组态的概念最早来自英文 configuration,其含义是使用软件工具对计算机及软件的各种资源进行配置,达到使计算机或软件按照预先设置,自动执行特定任务,满足使用者要求的目的。组态软件是面向监控与数据采集(Supervisory Control and Data Acquisition, SCADA)的软件平台工具,具有丰富的设置项目,使用方式灵活,功能强大。组态软件最早出现时,人机接口 HMI(Human Machine Interface)或 MMI(Man Machine Interface)是其主要内涵,即主要解决人机图形界面问题。随着它的快速发展,实时数据库、实时控制、SCADA、通信及联网、开放数据接口、对 I/O 设备的广泛支持已经成为它的主要内容。随着技术的发展,组态软件将会不断被赋予新的内容。

直到现在,大部分 DCS 厂家的组态软件仍是专用的(即与硬件相关的),不可相互替代。现在,国内已出现了多家独立软件商,专门从事工业控制组态软件的开发。他们都在设法提供不同厂家、不同设备的对应 I/O 驱动模块,使组态软件越来越趋于通用。

从 20 世纪 80 年代末开始,由于个人计算机的普及,国内开始有人研究如何利用 PC 进行工业监控,同时开始出现基于 PC 总线的 A/D、D/A、计数器、DIO 等各类 I/O 板卡。应该说,国内组态软件的研究起步是不晚的。当时有人在 MS-DOS 的基础上用汇编语言或 C

语言编制带后台处理能力的组态软件,有实力的研究机构则在实时多任务操作系统 iRMX86 上做文章,但均未形成有竞争力的产品。笔者曾经用 iRMX86 和 DOS 相结合的方式参与设计过电力调度自动化软件,其中以 DOS 设计前台人机界面,以实时性强的 iRMX86 设计后台处理程序,还使用过 iRMX86 for Windows。随着 MS-DOS 和 iRMX86 用户数量的萎缩和微软公司 Windows 操作系统的普及,基于 PC 的组态软件才迎来了发展机遇。世界上第一个把组态软件作为商品进行开发、销售的专业软件公司是美国的 Wonderware 公司,它于 20 世纪 80 年代末率先推出第一个商品化的组态软件 Intouch。Intellution 公司的 FIX 软件在国内也有很大的市场。此后,组态软件在全球得到了蓬勃发展。目前,世界上有影响的组态软件有几十种之多,总装机量有几十万套。伴随着信息化社会的到来,监控组态软件在社会信息化进程中将扮演越来越重要的角色,每年的市场增幅都会有较大增长,未来的发展前景十分看好。

### 1.1.1 组态软件的发展背景

监控组态软件是伴随着计算机技术的突飞猛进发展起来的。在 20 世纪 60 年代,计算机开始涉足工业过程控制,但由于计算机技术人员缺乏工厂仪表和工业过程的知识,导致计算机工业过程控制系统在各行业的推广速度比较缓慢。20 世纪 70 年代初期,微处理器的出现,促进了计算机控制技术走向成熟。微处理器在提高计算能力的基础上,大大降低了计算机的硬件成本,缩小了计算机的体积,很多研究、生产控制仪表和工业控制计算机的公司先后推出了新型控制系统。这一时期较有代表性的产品是 1975 年美国 Honeywell 公司推出的世界上第一套 DCS TDC-2000。而随后的 20 年间,DCS 及其计算机控制技术日趋成熟,得到了广泛应用。此时的 DCS 已具有较丰富的软件,包括计算机系统软件(操作系统)、组态软件、控制软件、操作站软件以及其他辅助软件(如通信软件)等。

在这一阶段,虽然 DCS 技术、市场发展迅速,但软件仍是专用和封闭的。除了在功能上不断加强外,由于开发成本高,用户数量少,软件成本一直居高不下,DCS 在中小型项目上的单位成本过高,使一些中小型应用项目不得不放弃使用 DCS。20 世纪 80 年代中后期,随着个人计算机的普及和开放系统(Open System)概念的推广,基于个人计算机的监控系统开始进入市场,并发展壮大。组态软件作为个人计算机监控系统的重要组成部分,比 PC(个人计算机)监控的硬件系统具有更为广阔的发展空间。第一,很多 DCS 和 PLC 厂家主动公开通信协议,加入“PC 监控”的阵营。目前,几乎所有的 PLC 和一半以上的 DCS 都使用 PC 作为操作站。第二,PC 监控大大降低了系统成本,使得市场空间得以扩大,从无人值守的远程监视(如防盗报警、江河汛情监视、环境监控、电信线路监控、交通管制与监控、矿井报警等)、数据采集与计量(如居民水、电、气表的自动抄表、铁道信号采集与记录等)、数据分析(如汽车和机车自动测试、机组和设备参数测试、医疗化验仪器设备实时数据采集、虚拟仪器、生产线产品质量抽检等)到过程控制,几乎无处不在。第三,各类智能仪表、调节器和 PC-based 设备由于增加了公开协议的通信接口,可与组态软件构筑完整的低成本自动化系统,具有广阔的市场空间。第四,各类嵌入式系统和现场总线的异军突起,把组态软件推到了自动化系统主力军的位置,组态软件越来越成为工业自动化系统中的灵魂。

组态软件之所以同时得到用户和 DCS 厂商的认可,主要有以下三个原因:

① 个人计算机操作系统日趋稳定可靠,实时处理能力增强且价格便宜。

② 个人计算机的软件及开发工具丰富,使组态软件的功能强大,开发周期相应缩短,软件升级和维护也较方便。

③ 可重用的第三方组件功能强大,集成方便。

目前,多数组态软件都是在 Windows 3.1 或 3.2 操作系统下逐渐成熟起来的,国外少数组态软件可以在 OS/2 或 UNIX 环境下运行。绝大多数组态软件都运行在 Windows 98/NT/2000/XP 环境下。较理想的运行环境是 Windows NT 或 Windows 2000 操作系统,因为这两种操作系统的内核是原来的 VMS 的变种,可靠性和实时性都好于 Windows 98。

## 1.1.2 组态软件的发展趋势

### 1. 组态软件作为单独行业出现是历史的必然

市场竞争的加剧使行业分工越来越细,“大而全”的企业将越来越少(企业集团除外),每个 DCS 厂商必须把主要精力用于他们本身所擅长的技术领域,巩固已有优势。如果他们还是软、硬件一起做,就很难在竞争中取胜。今后,社会分工会更加细化。表面上看来功能较单一的组态软件,其市场才初步形成,今后的成长空间还相当广阔。

组态软件的发展与成长和网络技术的发展与普及密不可分。曾有一段时期,各 DCS 厂商的底层网络都是专用的,现在则使用国际标准协议,这在很大程度上促进了组态软件的应用。有不少用户的监控点分布在上百甚至上千平方公里的范围内,要想把这些装置的实时数据进行联网共享,在几年前是不可想象的,而目前通过公众电话网,用 MODEM、ISDN、光纤或 ATM 将各 DCS 装置连起来,通过 TCP/IP 协议完成实时数据采集和远程监控就是一种可行方案。

### 2. 现场总线技术的成熟促进了组态软件的应用

应该说现场总线是一种特殊的网络技术,其核心内容一是工业应用,二是完成从模拟方式到数字方式的转变,使信息和供电同在一根双线电缆上传输,还要满足许多技术指标。同其他网络一样,现场总线的网络系统也具备 OSI 的若干层协议。从这个意义上讲,现场总线与普通的网络系统具有相同的属性,但现场总线设备的种类多,同类总线的产品也分现场设备、耦合器等多种类型。在未来几年,现场总线设备将大量替代现有现场设备,给组态软件带来更多机遇。

### 3. 能够同时兼容多种操作系统平台是组态软件的发展方向之一

可以预言,微软公司在操作系统市场上的垄断迟早要被打破,未来的组态软件也要求跨操作系统平台,至少要同时兼容 Windows 和 LINUX/UNIX。

UNIX 是唯一可以在微、超微、小、超小型工作站和大型机、中型机、小型机上“全谱系通用”的系统。由于 UNIX 的特殊背景和它强有力的功能,特别是它的可移植性以及目前硬件突飞猛进的发展形势,吸引了越来越多的厂家和用户。

UNIX 在多任务、实时性、联网方面的处理能力优于 Windows NT/2000,但它在图形界面、即插即用、I/O 设备驱动程序数量方面赶不上 Windows NT/2000。20 世纪 90 年代以来,UNIX 的这些缺点已得到改进,现在 UNIX 的图形界面 Xwindow 和 UNIX 的变种——Linux

已经具备了较好的图形环境。

#### 4. 组态软件在嵌入式整体方案中将发挥更大作用

前面已讲过, 微处理器技术的发展会带动控制技术 & 监控组态软件的发展, 目前, 嵌入式系统的发展速度极为迅猛, 但相应的软件尤其是组态软件滞后较严重, 制约着嵌入式系统的发展。从使用方式上把嵌入式系统分为两种: 带显示器/键盘和不带显示器/键盘。

(1) 带显示器/键盘的嵌入式系统。这种系统又可分为带机械式硬盘和带电子盘的嵌入式系统两种。

带机械式硬盘(如 PC/104 可外接硬盘)的嵌入式系统, 可装 Windows 98/NT 等大型操作系统, 对组态软件没有更多的要求。

不带机械式硬盘(带电子盘)的嵌入式系统, 由于电子盘的容量受限(也可以安装大容量电子盘, 但造价太高), 因此只能安装 Windows CE、DOS 或 LINUX 操作系统。目前, 支持 Windows CE 或 LINUX 的组态软件很少, 用户一般或自己亲自编程, 或使用以前的 DOS 环境软件。此类应用规模都不大, 但数量却有很大潜力。另外, 价格是一个重要因素, 如果嵌入式系统的软、硬件价格得到进一步降低, 其市场规模将是空前的。

(2) 不带显示器/键盘的嵌入式系统。这种嵌入式系统一般都使用电子盘, 只能安装 Windows CE、DOS 或 LINUX 操作系统。此类应用有的会带外部数据接口(以太网、RS232/485 等)。目前, 面向此类应用的组态软件市场潜力巨大。

#### 5. 组态软件在 CIMS 应用中将起到重要作用

美国的 Harrington 博士于 1973 年提出了计算机集成制造系统(Computer Integrated Manufacturing System, CIMS)的概念。CIMS 的主要内容有: 企业内部生产各环节密不可分, 需统筹协调; 工厂的生产过程实质就是对信息的收集、传递、加工和处理的过程。CIMS 所追求的目标是使工厂的管理、生产、经营、服务全自动化、科学化、受控化, 最大限度地发挥企业中人力、资源、信息的作用, 提高企业运转效率和市场应变能力, 降低成本。CIMS 的概念不仅适用于离散型生产流程的企业, 同样适用于连续型生产流程的企业。也有人把 CIMS 叫做计算机集成流程系统(Computer Integrated Process System, CIPS)。

自动化技术是 CIMS 的基础。目前多数企业对生产自动化都比较重视, 他们或采用 DCS(含 PLC)或采用以 PC 总线为基础的工控机构成简易的分散型测控系统。但现实当中的自动化系统都是分散在各装置上的, 企业内部的各自动化装置之间缺乏互联手段, 不能实现信息的实时共享, 这从根本上阻碍了 CIMS 的实施。

组态软件在企业 CIMS 发展过程中能够发挥以下作用:

① 充当 DCS(含 PLC)的操作站软件, 尤其是 PC-based 监控系统。

② 以往各企业只注重在关键装置上投资, 引进自动化控制设备, 而在诸如公用工程(如能源监测、原材料管理、产成品管理、产品质量监控、自动化验分析、生产设备状态监视等)生产环节方面则重视程度不够。这种一个企业内部各部门间自动化程度的不协调也会影响 CIMS 的进程, 受到损失的将是企业本身。组态软件在这方面(即技术改造方面)也会发挥更大的作用, 促进企业以低成本、高效率实现企业的信息化建设。

③ 组态软件具有丰富的 I/O 设备接口, 能与绝大多数控制装置相联, 具有分布式实时数据库, 可以解决分散的“自动化孤岛”互联问题, 大幅度节省 CIMS 建设所需的投资。伴随着 CIMS 技术的推广与应用, 组态软件将逐渐发展成为大型平台软件, 以原有的图形

用户接口、I/O 驱动、分布式实时数据库、软逻辑等为基础,将派生出大量的实用软件组件,如先进控制软件包、数据分析工具等。

### 6. 信息化社会的到来为组态软件拓展了更多的应用领域

组态软件的应用不仅仅局限在工业企业,在农业、环保、邮政、电信、实验室、医院、金融、交通、航空等各行各业均能找到使用组态软件的实例。

随着社会进步和信息化速度的加快,组态软件将赢得巨大的市场空间。这将极大地促进国产优秀组态软件的应用,为国产优秀组态软件创造良好的环境,促进国产软件品牌的成长和参与国际竞争。

组态软件的发展也加剧了对从事组态软件开发与研制的人才需求。以往,国内的组态软件经历了从无到有的曲折过程,而目前则面临着如何在未来的竞争中取胜,如何制定未来的发展战略,如何开拓国际市场等一系列新的课题。组态软件涉及自动控制理论及技术、计算机理论及技术、通信及网络技术、人机界面技术等多个学科,对开发人员的软件设计、理论及实践经验都有很高的要求,广大在校的相关专业大学生、研究生,面临着从事该项工作的难得机遇,盼望更多的人才加入到组态软件的开发队伍中来,为我国的国民经济信息化做出历史性贡献。

## 1.2 组态软件的基本结构

### 1.2.1 组态软件的基本结构

工程应用中,组态软件一般是由系统开发环境和系统运行环境两大部分构成的。

(1) 系统开发环境。它是自动化工程设计工程师为实施其控制方案,在组态软件的支持下进行应用程序的系统生成工作所必须依赖的工作环境。通过建立一系列用户数据文件,生成最终的图形目标应用系统,供系统运行环境运行时使用。系统开发环境由若干个组态程序组成,如图形界面组态程序、实时数据库组态程序等。

(2) 系统运行环境。在系统运行环境下,目标应用程序被装入计算机内存并投入实时运行。系统运行环境由若干个运行程序组成,如图形界面运行程序、实时数据库运行程序等。

组态软件支持在线组态技术,可在不退出系统运行环境的情况下直接进入组态环境并使修改后的组态直接生效。

进行自动化工程设计最先接触的—定是系统开发环境,再通过一定工作量的系统组态和调试,最终将目标应用程序在系统运行环境投入实时运行,完成一个工程项目。

一般工程应用必须有一套开发环境,可以有多套运行环境。在本书的例子中,为了方便,我们将开发环境和运行环境放在一起,通过菜单限制编辑修改功能而实现运行环境。

本书介绍组态软件的设计与开发,所以我们应该了解其功能结构。组态软件因为功能强大,而每个功能相对来说又具有一定的独立性,因此其组成形式是一个集成软件平台,由若干程序组件构成。

基本组态软件必备的功能组件包括如下四个部分。

(1) 应用程序管理器。应用程序管理器是提供应用程序的搜索、备份、解压缩、建立新

应用等功能的专用管理工具。在自动化工程设计工程师应用组态软件进行工程设计时,经常会遇到下面一些烦恼:经常要进行组态数据的备份;经常需要引用以往成功项目中的部分组态成果(如画面);经常需要迅速了解计算机中保存了哪些应用项目。虽然这些工作可以用手工方式实现,但效率低下,极易出错。有了应用程序管理器的支持,这些工作将变得非常简单。

(2) 图形界面开发/运行程序。它是一个进行图形系统生成工作所依赖的开发环境。通过建立一系列用户数据文件,生成最终的图形目标应用系统。生成的图形目标应用系统可在图形运行环境中运行。

构成现场各过程图形的画面被划分成三类简单的对象:线、填充形状和文本。每个简单的对象均有影响其外观的属性。对象的基本属性包括:线的颜色、填充颜色、高度、宽度、取向、位置移动等。这些属性可以是静态的,也可以是动态的。静态属性在系统投入运行后保持不变,与原来组态时一致。而动态属性则与表达式的值有关。表达式可以是来自 I/O 设备的变量,也可以是由变量和运算符组成的数学表达式。这种对象的动态属性随表达式值的变化而实时改变。例如,用一个矩形填充体模拟现场的液位,在组态这个矩形的填充属性时,指定代表液位的工位号名称、液位的上、下限及对应的填充高度,就完成了液位的图形组态。这个组态过程通常叫做动画连接。

在图形界面上还具备报警通知及确认、报表组态及打印、历史数据查询与显示等功能。各种报警、报表、趋势都是动画连接的对象,其数据源都可以通过组态来指定。这样,每个画面的内容就可以根据实际情况由工程技术人员灵活设计,每幅画面中的对象数量均不受限制。

在图形界面中,各类组态软件普遍提供了一种类似 Basic 语言的编程工具——脚本语言来扩充其功能。用脚本语言编写的程序可由事件驱动或周期性地执行,是与对象密切相关的。例如,当按下某个按钮时可指定执行一段脚本语言程序,完成特定的控制功能,也可以指定当某一变量的值变化到关键值以下时,马上启动一段脚本语言程序完成特定的控制功能。

本书第三章介绍基本图形界面开发工具。

(3) 实时数据库系统组态/运行程序。有的组态软件只在图形开发环境中增加了简单的数据管理功能,因而不具备完整的实时数据库系统。目前,比较先进的组态软件都有独立的实时数据库组件,以提高系统的实时性,增强处理能力。实时数据库系统组态程序是建立实时数据库的组态工具,可以定义实时数据库结构、数据来源、数据连接、数据类型及相关的各种参数,生成目标实时数据库。生成的目标实时数据库可在实时数据库运行环境中运行。

实时数据库是更为重要的一个组件。因为 PC 的处理能力强大,因此,实时数据库更加充分地表现出了组态软件的长处。实时数据库可以存储每个工艺点的多年数据,用户既可浏览企业当前的生产情况,又可回顾过去的生产情况。可以说,实时数据库对于企业来说,就如同飞机上的“黑匣子”。企业的历史数据是很有价值的,实时数据库具备数据档案管理功能。企业的实践告诉我们,现在很难知道将来进行分析时哪些数据是必需的。因此,保存所有的数据是防止信息丢失的最好方法。

本书第八章介绍相关内容。

(4) I/O 驱动程序。它是组态软件中必不可少的组成部分,用于和 I/O 设备通信,互相交换数据。DDE 和 OPC Client 是两个通用的标准 I/O 驱动程序,用来支持 DDE 标准和 OPC 标准的 I/O 设备通信。多数组态软件的 DDE 驱动程序被整合在实时数据库系统或图形系统中,而多数 OPC Client 则单独存在。

本书第五章介绍相关内容。

典型的组态软件还应包括下列功能组件。

(1) 通用数据库接口(ODBC 接口)组态/运行程序。通用数据库接口组件用来完成组态软件的实时数据库与通用数据库(如 Oracle、Sybase、Foxpro、DB2、Informix、SQL Server 等)的互联,实现双向数据交换。通用数据库既可以读取实时数据,又可以读取历史数据;实时数据库也可以从通用数据库实时地读入数据。通用数据库接口(ODBC 接口)组态环境用于指定要交换的通用数据库的数据库结构、字段名称及属性、时间区段、采样周期、字段与实时数据库数据的对应关系等。

(2) 策略(控制方案)编辑/生成组件。策略编辑/生成组件是以 PC 为中心实现低成本监控的核心软件,具有很强的逻辑、算术运算能力和丰富的控制算法。策略编辑/生成组件通过 IEC61131-3 标准为使用者提供标准的编程环境,共有四种编程方式:梯形图,结构化编程语言,指令助记符,模块化功能块。使用者一般都习惯于使用模块化功能块,根据控制方案进行组态,结束后系统将保存组态内容并对组态内容进行语法检查、编译。

编译生成的目标策略代码既可以与图形界面同在一台计算机上运行,也可以下装到目标设备(如 PC/104、Windows CE 系统等 PC-based 设备)上运行。

控制功能组件以基于 PC 的策略编辑/生成组件(也有人称之为软逻辑或软 PLC)为代表,是组态软件的主要组成部分。虽然脚本语言程序可以完成一些控制功能,但还是不很直观,对于用惯了梯形图或其他标准编程语言的自动化工程师来说,是太不方便了,因此,目前的多数组态软件都提供了基于 IEC61131-3 标准的策略编辑/生成控制组件。它也是面向对象的,但不唯一地由事件触发,它像 PLC 中的梯形图一样按照顺序周期地执行。策略编辑/生成组件在基于 PC 和现场总线的控制系统中是大有可为的,可以大幅度地降低成本。

(3) 实用通信程序组件。实用通信程序极大地增强了组态软件的功能,可以实现与第三方程序的数据交换,是组态软件成为开放系统的标志。实用通信程序具有以下功能:

① 用于双机冗余系统中主机与从机间的通信。

② 通信实用程序可以使用以太网、RS-485、RS-232、PSTN、GSM 等多种通信介质或网络实现其功能。实用通信程序组件可以划分为 Server 和 Client 两种类型。Server 是数据提供方,Client 是数据访问方。一旦 Server 和 Client 建立起了连接,二者间就可以实现数据的双向传送,构成分布式 HMI/SCADA 监控实时网。

③ 在基于 Internet 或 Browser/Server(B/S)的应用中实现通信功能。

在单任务操作系统环境下(例如 MS-DOS),要想让组态软件具有很强的实时性,就必须利用中断技术。这种环境下的开发工具较简单,软件编制难度大。目前,运行于 MS-DOS 环境下的组态软件基本上已退出市场。

在多任务环境下,由于操作系统直接支持多任务,组态软件的性能得到了全面加强。因此,组态软件一般都由若干组件构成,而且组件的数量在不断增长,功能不断加强。各组态软件普遍使用了“面向对象”(Object Oriented)的编程和设计方法,使软件更加易于学