Ms Net插件技术在继电保护整定计算软件中的应用研究

卓 越, 吕飞鹏, 黄 斌, 易 雷, 胡鹏飞
(四川大学电气信息学院, 四川 成都 610065)

摘要: 针对当前继电保护整定计算软件存在的不足,论述 Ms Net组件技术相对于传统组件技术的优势,提出了利用 Ms Net组件技术开发插件的方案,给出了基于 Ms Net插件技术的整定计算软件体系架构,并详细介绍了用 C#语言设计开发继电保护整定计算软件插件的要点以及代码发现、插件动态加载的方法,最后阐述了 Ms Net插件的安全性问题。

关键词: 继电保护; 整定计算; Ms Net; 插件技术

中图分类号: TM77 **文献标识码**: A **文章编号**: 1003-4897(2005)21-0021-04

0 引言

继电保护装置是保障电力系统安全稳定运行的重要基础设施,随着继电保护整定计算及定值管理软件的广泛应用,现有的软件存在的问题越来越突出,其中包括:用户界面与业务逻辑的紧耦合,导致系统扩展、维护和重用困难;整定计算大多数只考虑原理级的主定值,未考虑装置级的大量辅助定值计算,或者仅考虑具体应用单位的当前保护装置或市场上常用的装置;当出现新原理的装置、新的计算原则或计算需求以及由于不同地域的电网特点、考虑的计算原则、工作流程和习惯造成的差异时,需要进行大量的修改维护工作。这样造成现有软件的许多代码失去其生命力,造成大量的重复项目开发和大量的人力、物力的重复投入,从而造成软件开发周期长、成本高的恶性循环。

鉴于上述问题,并利用 Microsoft公司最新推出的. Net软件开发平台在开发组件及插件方面的强大功能,本文提出了基于. Net平台开发继电保护应用插件的新技术以及基于. Net插件的分布式多层应用架构,详细介绍了用 C#语言开发. NET插件的设计思路和实现方法及插件的动态发现加载的原理和安全性的考虑。

1 基于. Net平台的插件技术

1.1 . Net插件技术的引入

插件的原理就是通过统一的程序接口来调用不同的功能模块,以达到实现不同的功能调用,扩充主程序功能的目的。从结构上看,一个插件实际上就是一个组件,插件的开发从某种意义上说即是组件的开发技术,组件技术即将单独、庞大、复杂的应用程序分成多个模块,这些模块不再是简单的代码集,而是自给自足的组件。这些组件模块可运行在同一机器上,也可运行在局域网、广域网甚至 Internet上的不同机器中。

目前,组件技术主要有:OMG起草与颁布的 CORBA; SUN 发表的 JavaBeans; Microsoft推出的 COM/DCOM/COM+3个规范。与前两个规范相比较,COM标准具有这些优点[1]:

1) COM定义了一个对象必须支持的二进制接口,开发者可以使用不同的语言来编写支持该接口的 COM的对象和调用该对象的客户。

2) COM的版本管理功能使得有新功能的新版本软件替换旧版本时,不必更改已有的客户程序,因为 COM对象具有支持多个接口的能力,新增功能时可用添加新接口的形式来实现,因而可不修改原有的接口,这样已有的客户程序就不会受影响,但是不同版本的 COM组件可能使依赖旧版本的应用不能正常工作,存在版本冲突,即所谓的"DLL HELL"问题。

3) COM能够实现软件功能的模块化,每个功能块提供各自的服务,开发者能够使用面向对象的方法去设计和开发程序,简化了系统复杂性。

4) COM提供了访问软件服务的一致性,无论访问的服务存在于动态连接库、另一进程或系统软件中,均可将它们当成 COM对象,使用同一种方法去访问。

而 Microsoft最近推出的基于. Net Framework组件技术更是维护了核心的 COM思想,同时还改善了 COM组件的一些不足,如:通过套间实现的难以使用的并发管理;由引用计数的缺陷导致的内存和资源泄露;COM组件将依赖于注册表,导致庞大的注

册表文件;需要开发者提供代理存根对;需要在与代码分离的 IDL文件中定义接口和类型等等。与 COM组件技术相比,. Net Framework开发组件主要有以下这些优越性[2]:

1). Net Framework简化了组件开发的过程[3]。它没有派生出所有组件的正式基本接口,所有的组件都从 System. object类派生;它没有类工厂;没有对象的引用计数;. Net编程的基本单位程序集 (Assembly)是自描述的,因为存于程序集中的信息即元数据全面地描述了每一模块的内容,这样,就不需要描述接口和定制类型的 IDL文件或类型库;没有 GUID,用命名空间和程序集名称来确定类型范围的方法使类型 (类或接口及其方法)具有唯一性;没有套间,默认情况下,所有. Net组件都在自由线程环境中执行,开发者负责同步组件的访问。. Net组件的内存分配和回收由. Net公共语言运行时 CLR (Common Language Runtime)自动负责管理。

2)与 COM组件相比,. Net Framework不会大量地有求于 Windows注册表或 Win\system32系统目录,大多数. Net组件对注册表没有影响,安装. Net组件就是把它简单地复制到主程序所在的目录,对多个应用程序公用的组件,则可以通过对组件签名后使其具有强名 (Strong Name),然后发布到系统全局程序集缓冲区 GAC (Global Assembly Cache)。

3)跨语言的兼容性。COM的语言互用性仅在开发时有效,而. Net在开发期间允许使用一种语言开发的组件无缝地从另一种语言开发的组件派生,这样,代码的复用率会大大提高。

4). Net严格维护版本控制,允许相同计算机上相同组件的多个新旧版本共存同时运行,进而摆脱了“DLL Hell”的困境,原有应用程序可以使用安装时的版本或经用户修改配置文件使用新版本。

5). Net的代码访问安全性 CAS (Code Access Security)功能允许为指定代码段配置许可并提供证据,从而证明它拥有访问资源或执行敏感操作需要的正确凭证。组件可以要求执行许可检查,以便在继续执行自己的操作之前首先验证调用链中的所有调用都拥有正确的许可。这就实现了 COM+基于角色的安全性和调用身份验证,它为应用程序管理员提供了细致的控制,不仅控制允许用户执行的操作,而且控制允许组件执行的操作。

基于以上这些特点,我们采用. Net Framework和它的核心语言 C#来开发组件以及插件。

现有的插件开发的构架,大致上可分为:

1)脚本式,即使用某种语言把插件的程序逻辑写成脚本代码。但用户容易自行修改脚本,安全性不高。

2)聚合式,就是把插件功能直接写成 EXE,主程序负责调度这些插件,但在这种形式中插件与插件之间,主程序与插件之间的信息交流十分困难。

3)组件技术,我们的插件需要做的只是实现程序定义的接口。主程序不需要知道插件怎样实现预定的功能,它只需要通过接口访问插件,并提供主程序相关对象的接口。这样一来,主程序与各插件之间的信息交流就变得异常简单。并且,插件对于主程序来说是完全透明的。

1.2 . Net插件技术的实现

1)建立接口

在一个独立的程序集中,接口定义继承该接口的插件所实现的功能、方法、属性、事件,接口是通过 C#中的 interface关键字定义的,具体的功能实现通过相关插件完成。

2)通过插件实现接口

在程序运行的时候,主程序将可以利用反射 (reflection)来访问插件中定义的类、方法及其属性, System. Reflection命名空间也提供了访问外部程序集的一系列相关对象。

3)插件的加载

为了实现插件的动态加载,主程序使用了. Net反射机制。在. Net反射机制的帮助下,插件类型信息将被主程序加载和查看,这样就可以通过检查这个类型以判断插件是否有效,如果类型通过了检查,那么插件就可以被添加到主程序的界面中,就可以被用户操作。

4)版本控制

由于. Net框架允许相同计算机上新旧版本的组件的同时运行,所以主程序与插件间没有必然的版本制约关系,在更新设计主程序时也会尽量地兼容低版本的插件。

2 . Net插件技术在整定计算软件中的应用

对于国内外有关单位开发的继电保护整定计算程序存在的一些问题,采用基于. Net的组件和插件技术都可以迎刃而解。下面我们就结合一个具体的继电保护整定计算软件来详细介绍插件技术在整定软件中的开发和实现过程。

2.1 系统架构及工作流程

如图 1,采用. Net组件技术将整定软件分解为

用户表示层、中间业务逻辑层和数据访问层三层结构。利用. Net组件模型的自描述反射功能,实现各层组件的动态发现、装配和插拔。

当用户启动整定程序时,主程序会搜索插件目录下的每一个文件,如果文件为插件,就用 Assembly对象加载它。然后通过插件管理器调用不同的插件,如果增加了新的保护装置,就可以调用新的装置级插件;如果整定原理变了,就可以调用原理级插件。这样通过调用插件更新了故障计算组件、装置级整定计算组件、原理级整定计算组件等等逻辑处理模块,然后将更新了的组件集成之后实现整定计算、数据传递、数据显示等功能,这样就以最小的系统维护代价来适应用户需求的变化。

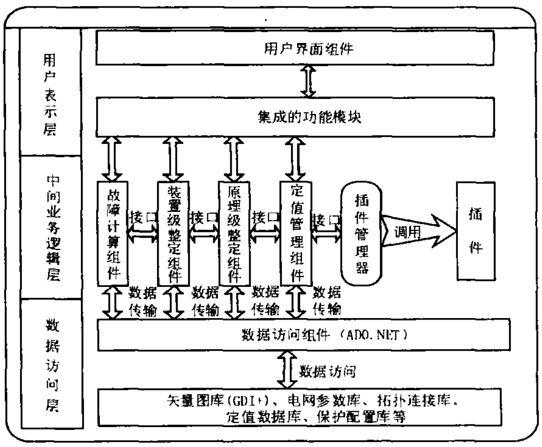

图 1　整定计算软件系统架构

Fig 1　Framework of relay setting calculation software

2.2　插件实现要点

1) 接口定义

在软件开发的过程中,我们可以在故障计算组件、装置级整定计算组件、原理级整定计算组件、定值管理组件等等地方定义接口,在接口中定义需要的定值数据、整定计算原则、定值表单等属性。以一个简单的装置级整定计算组件的接口为例:

```
public interface DevicePlugin
{
  //访问装置定值
  DataSet GetDeviceData();
  //更新装置定值
  bool UpdateDevice(DataSet changes) ;
  … … … … … … … … … … … …
}
```

DevicePlugin接口定义了访问和更新装置定值两个方法,这些定义并没有描述具体是怎么完成的,DevicePlugin接口通过继承它的类来实现这些方法。

2) 插件开发

根据接口的定义和实际的情况来开发新插件,如果当用户的实际应用条件改变如新增加了保护装置时,就可以根据接口定义的方法和新增装置的数据开发新的插件。以新增保护装置 WXH - 110为例,设计一个名为"Wxh110Plugin. dll"的插件,该插件中描述 Wxh - 110装置的类 Wxh110plugin实现了 DevicePlug接口:

```
public class Wxh110Plugin: DevicePlug
{
  public DataSet GetDeviceData()
  {
   //实现访问定值方法的代码
   }
  public UpdateDevice(DataSet changes)
  {
   //实现更新定值方法的代码
  }
}
```

3) 插件加载和代码发现

插件开发完成后被编译为程序集,主程序运行时通过迟绑定 (late binding)动态加载插件。迟绑定即系统在运行而不是编译的时候发现和执行代码,在. NET框架中,迟绑定通过反射机制完成,主要调用了以下三个重要的类,即: System. AppDomain, System. Reflection. Assembly和 System. Type。系统使用静态方法 Assembly. Load将程序集加载到现有的应用程序域 AppDomain中,并返回程序集对象,这样系统就能发现包含在程序集中的类型、创建实例、调用方法,这样就完成了插件的动态加载和实例化,下面为一个具体的加载过程的代码:

```
using System;
using System. IO;
using System. Reflection;
public Class PluginsLoadDemo
{
  static void Main(string[ ] args)
{
  //使用插件文件名称
  //"d: \plugins\Wxh110plugin. dll"
    //创建程序集实例对象
   Assembly a = Assembly. LoadFile(
  @"d: \plugins\Wxh110plugin. dll");
//获取程序集中定义的类型
```

```
Type[ ]pluginTypes = a. GetTypes();
//搜索插件中的类是否继承了预定义的接口
// Deviceplugin
foreach (Type pluginType in pluginTypes) {
   Type typeInterface =
         pluginType. GetInterface(
              " DevicePlugin", true);
          if (typeInterface ! = null) {
         //动态创建 Wxh110plugin类实例对象
         //并转换为 DevicePlugin类型
          DevicePlugin wxh110plugin =
         Activator. CreateInstance(type) as DevicePlugin;
         //通过接口变量调用接口方法
         wxh110plugin. GetDeviceData();
          }
   }
}
}
```

在代码的发现过程中主要两种方法:

a.最简单的方法是用 XML(可扩展标记文件)配置文件形式将程序集的名称和类型表示出来,如:

```
<plugins>
<plugin type=" Wxh110Plugin,Wxh110Plugin" />
</plugins>
```

这个 XML 文件表示了我们需要加载的 Wxh110Plugin插件的类名称“Wxh110Plugin”和程序集名称“Wxh110Plugin”,代码只需通过插件配置文件中发现我们需要加载的程序集名称和类名称即可动态加载和实例化插件。这种方法容易实现,但要更改应用程序的插件配置文件来增加新的插件。

b.基于反射机制的发现方法是通过主程序搜索相关的子目录(如 Plugins)自动发现代码。用户只需要将程序集拷贝到主程序插件目录下中,当主程序启动时它将会自动绑定新代码。由此可见,这个方法更具有实用性,故在本方案中采用这一方法。

4)插件的安全性[4]

当保护整定程序被广泛应用,插件可以被多方开发时,就必须考虑其安全性,. Net Framework使代码的保护变得更加容易。. Net Framework能授予代码不同的权限,将代码分为完全信任代码(Full trusted)和部分信任代码(Partially trusted)。完全信任代码具有访问本地系统等资源的权限,而部分信任代码则只有用户赋予的部分权限。当 CAS执行部分信任代码时,因为托管代码的处理是实时的,公共语言运行时(CLR)能够判断代码未经许可的部分并拒绝运行它,如果部份信赖的代码尝试不允许的操作,CLR就会抛出一个能被应用程序捕捉到的安全异常 SecurityException,通过. Net Framework的异常(Exception)处理机制解决问题。CLR允许一个应用程序由多个程序集构成,各个程序集具有不同的安全许可,这对系统的安全性具有相当的好处。

在实际应用中,首先为插件创建 FT和 PT两个子目录,然后调整 PT目录的本地安全策略,一旦做好这一代码集合,从 PT子目录中加载插件,CLR将会自动地减少插件的许可,从而防止了一些具有恶意代码的插件进行访问系统文件和注册文件等等影响系统安全的操作。

3 总结

利用本文所提出的开发继电保护整定计算插件的新技术,现已开发出了针对南瑞、南自、许继等厂家的 110 kV线路保护装置的插件,结合用 C#语言开发的“继电保护整定计算专家系统”,该技术已在实际应用中取得一定的效果,它提高了软件代码的复用性,降低了软件的开发周期和成本,. Net的 CAS功能又为插件代码的安全性提供了充分的保证。这样就使整定计算软件能够实现真正的“即插即用”,大大提高了软件的可扩展性和重用性。

参考文献:

[1] 邓建,宋玮,张明霞. 基于组件技术的继电保护整定计算软件的设计[J]. 电力科学与工程, 2003, (4): 54-57.
DENG Jian, SONG Wei, ZHANG Ming-xia. Software Design for Relay Protection Settings Based on Component Techniques[J]. Power Science and Engineering, 2003, (4): 54-57.

[2] Lowy J.. COM与. NET[M]. 常晓波,朱剑平,译. 北京:中国电力出版社, 2002.
Lowy J.. COM and . NET[M]. CHANG Xiao-bo, ZHU Jian-ping, Trans. Beijing: China Electric Power Press, 2002.

[3] Faison T. Visual C#基于组件的开发[M]. 战晓苏,译. 北京:清华大学出版社, 2003.
Faison T. Component-based Development with Visual C# [M]. ZHAN Xiao-su, Trans. Beijing: Tsinghua University Press, 2003.

[4] Clark J. Let Users Add Functionality to You[M].. NET Applications with Macros and Plug-Ins MSMSDN library.

收稿日期: 2005-02-22; 修回日期: 2005-05-12

(下转第 44页 continued on page 44)

CAO Guo-chen, ZHANG Gong, et al The GeneralMethodology for Calculating Distribution Network Fault [J]. Proceedings of the CSEE, 1999, 19 (7) : 41-44.

[3] 邓佑满, 蔡乐, 赵大溥. 提高配电网可恢复供电能力的算法 [J]. 电力系统自动化, 2002, 26 (11) : 19-22.
DENG You-man, CAI Le, ZHAO Da-bo. The Algorithm of Improving Power Supply Ability of Distribution Network Resumed [J]. Automation of Electric Power Systems, 2002, 26 (11) : 19-22.

[4] 谈伟, 张焰, 蓝毓俊, 等. 配电网故障后的供电恢复研究 [J]. 中国电力, 2000, 33 (12) : 32-34.
TAN Wei, ZHANG Yan, LAN Yu-jun, et al The Research of Power Supply Resumed after Distribution Network Fault[J]. Electric Power, 2000, 33 (12) : 32-34.

[5] 陈根军, 王磊, 唐国庆. 基于蚁群最优的配电网络重构算法 [J]. 电力系统及其自动化学报, 2001, 13 (2) : 48-53.
CHEN Gen-jun, WANG Lei, TANG Guo-qing The Algorithm of Optimal Distribution Network Reconfiguration Based on Ants [J]. Proceedings of the EPSA, 2001, 13 (2) : 48-53.

[6] 刘伟. 配电网安全性分析研究 [D]. 哈尔滨: 哈尔滨工业大学, 2003. 64-67.
LIU Wei The Research on Security Analysis of Distribution Network [D]. Harbin: Harbin Institute of Technology, 2003. 64-67.

[7] 刘莉. 基于模糊遗传算法的配电网络优化方法的研究 [D]. 哈尔滨: 哈尔滨工业大学, 2000.
LIU Li The Research of Optimal Distribution Network Based on the Fuzzy and Genetic Algorithms [D]. Harbin: Harbin Institute of Technology, 2000.

收稿日期: 2005-03-02; **修回日期**: 2005-04-14

作者简介:
康巍晶 (1966 -), 女, 硕士研究生, 研究方向为配电网自动化; E-mail: kangwj@263. com
刘 伟 (1977 -), 男, 博士研究生, 研究方向为配电网自动化;
李海城 (1976 -), 男, 助理工程师, 从事调度自动化方面的工作。

The research of restorative control in distribution network

KANG Wei-jing[1], LIU Wei[1], LIU Hai-cheng[2]
(1. Harbin Institute of Technology, Harbin 150001, China; 2. Rural Power Bureau, Harbin 150076, China)

Abstract: According to the differences between distribution network and transmission network, the mathematical model of restorative control is built The method of restorative control in distribution network is realized The basic means of implementing the method include network reconfiguration and capacitor compensation after fault Meanwhile, the load-cutting strategy with customer's power supply PRI is proposed The validity of the method is proved by IEEE 69 bus system.

Key words: distribution network; restoration; control; fault

(上接第 24页 continued from page 24)

作者简介:
卓 越 (1980 -), 男, 硕士研究生, 主要从事继电保护及其在微机应用方面的研究; E-mail: zhuoailan@yahoo. com. cn
吕飞鹏 (1968 -), 男, 教授, 硕士生导师, 从事电力系统继电保护和自动化的研究;
黄 斌 (1977 -), 男, 硕士研究生, 主要从事继电保护及其在微机应用方面的研究。

Study on the application of Ms Net based plug-in technology in relay protection setting calculation software

ZHUO Yue, LÜ Fei-peng, HUANG Bin, YI Lei, HU Peng-fei
(School of Electrical Engineering and Information, Sichuan University, Chengdu 610065, China)

Abstract: By taking account of the existing deficiency in application software available for relay protection setting calculation, this paper discusses the advantages of the Ms Net component technique through comparing it with traditional component technique The solution of developing plug-in based on Ms Net component technique is proposed, and the system architecture of relay setting calculation software based on plug-in technique is presented It details the key points of developing plug-in for setting calculation software and the method of dynamically finding code and loading plug-in In the end, the security of the plug-in is set forth

Key words: relay protection; setting calculation; Ms Net; plug-in technique