**C#的插件开发**

|  |
| --- |
| [分类 WinForm](http://www.chenjiliang.com/Article/Default.aspx?TypeID=79)2006-9-13 15:25:40 浏览 4838 回复 4    隐藏侧边栏 |
| [转自 gavinou的专栏](http://blog.csdn.net/gavinou/archive/2006/09/13/1217933.aspx) |

.

之前一直想学学插件编程, 主要的原因是感到现在的客户需求变化不定 ,如果把全部功能整合在一个执行文件中的话,修改,升级起来很 不方便, 所以想采用插件编程的方式, 只需要构建好了程序框架之后, 每完成一个功能,就可以让用户审核一 个, 化整为零,让开发团队和客户都能构掌握项目开发的进度. 同时大家通过这种方式,增强对项目按时完成的信心.   
  
以下资料主要是从haha blog中获得的, 从网上找了很多类似的资料,只有这个我觉得是比较适合初学者

复制  保存

//1 定义插件接口,将其编译成 dll,例如:

using System;

namespace PluginInterface

{

public interface IShow

{

string Show();

}

}

//2 编写插件. 新建dll工程,并引用第一步做的dll插件,实现其接口,例如:

namespace PluginA

{

public class PluginA : PluginInterface.IShow

{

public string Show()

{

return "I am plugin A";

}

}

}

收集程序集:

复制  保存

//3. 在指定目录下寻找Dll文件

private void frmMain\_Load(object sender, System.EventArgs e)

{

//获取Plugins目录中所有的DLL文件，并保存在combo中

try

{

string path = Application.StartupPath;

path = System.IO.Path.Combine(path, "Plugins");

foreach (string file in System.IO.Directory.GetFiles(path, "\*.dll"))

{

this.cmbPlugins.Items.Add(file);

}

}

catch (Exception ex)

{

MessageBox.Show(ex.Message);

}

}

使用插件

复制  保存

private void btnExecute\_Click(object sender, System.EventArgs e)

{

try

{

//1. 获得 文件名称

string asmFile = this.cmbPlugins.Text;

string asmName = System.IO.Path.GetFileNameWithoutExtension(asmFile);

if (asmFile != string.Empty)

{

//2. 利用反射,构造DLL文件的实例

System.Reflection.Assembly asm = System.Reflection.Assembly.LoadFrom(asmFile);

//3. 利用反射,从程序集(DLL)中,提取类,并把此类实例化

PluginInterface.IShow iShow = (PluginInterface.IShow)

System.Activator.CreateInstance(asm.GetType(asmName + "Namespace." + asmName + "Class"));

//4. 在主程序中使用这个类的实例

this.label2.Text = iShow.Show();

}

}

catch (Exception ex)

{

MessageBox.Show(ex.Message);

}

}

示例见附件