# 组件封装环境组件创建说明文档

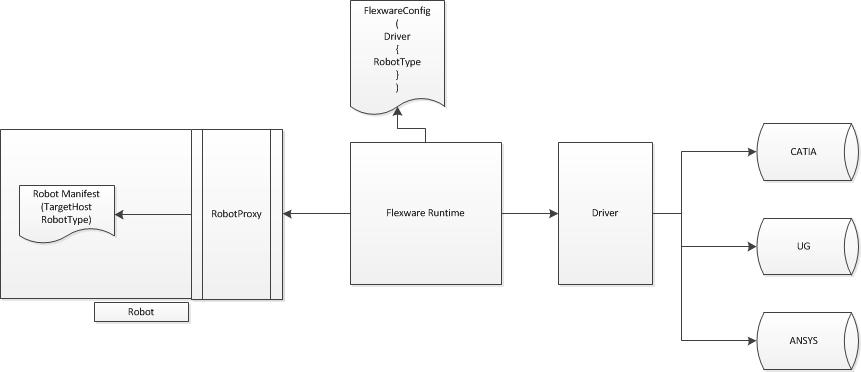
说明：1、本文档解释组件如何在pera13.1版本封装环境中封装新类型组件。

2、pera13.1中组件不向前兼容。

3、pera13.1中单独功能性组件不存在独立的表单设计功能。

开发步骤：

1. Flexware标准设计执行过程
2. 创建组件
3. 提取参数
4. 编译运行
5. Flexware的runtime调用关系图



1. 组件实现步骤
2. 在Flexware.Controls解决方案中创建参数编辑控件，并实现IPickupParameters接口。

IPickupParameters接口说明：

设计时中打开组件与单击保存参数都会调用该接口。

单击保存参数按钮调用组件接口顺序：

1、Parameters属性

2、GetAddtionalFiles()保存至manifest中Reference下

3、GetAddtionalParameters()保存至manifest中Runtime下

4、SaveState（Mainifest文件路径）

打开组件工程调用顺序：

ScriptType会调用OpenState(Mainifest文件路径)方法。

EntityType会调用该组件对应驱动的GetParameters()方法并将返回值赋值给IPickupParameters.PickupSpurce属性。

在实现对应的方法中要增强对Minifest文件的控制可在SaveState与OpenState中创建RobotManifestAccess实例、只要将SaveState或OpenState方法的Manifest文件路径传入构造方法即可。

1. 在Flexware.BizBase解决方案中创建新工程，引入Flexware.BizBase.Base.dll与Flexware.BizBase.BizDriver.dll、Flexware.FrameBase.Services.dll

I、脚本类型实现

Driver类 继承BaseBizDriver

Parser类 继承BaseRobotParser

II、实体类型实现

Driver类 继承BaseBizEntityDriver

Parser类 继承BaseRobotParser

1. 配置Flexware安装目录下bin\Config\Services.config

添加配置段

<component

id="\*BizDriver"

service="Flexware.BizBase.Driver.IBizDriver,Flexware.BizBase.Base"

type=" Driver Dll名+driver类名, Driver Dll名">

</component>

1. 在Flexware安装目录下Addins\BackendBindings\FlexwareBinding\Templates添加模板文件，修改相应字段，如下例

<?xml version="1.0"?>

<Template originator = "Li Donghui"

created = "06/10/2001"

lastModified = "02/01/2003">

<!-- Template Header -->

<TemplateConfiguration>

<Name>显示的名称</Name>

<Category>类别名称</Category>

<Icon>Flexware.ProjectBrowser.Project</Icon>

<Description>描述</Description>

</TemplateConfiguration>

<!-- Actions -->

<Actions>

</Actions>

<!-- Template Content -->

<Project language = "Flexware">

<PropertyGroup>

<TargetHost>驱动目标</TargetHost>

<RobotType>ScriptType</RobotType>

<Driver>驱动实现类名, 驱动实现空间</Driver>

<ParameterControl>取参控件类名, 驱动取参控件空间名</ParameterControl>

<BaseInfoEditControl>Flexware.Controls.PickupEditBaseInfo, Flexware.Controls.PickupBaseInfo</BaseInfoEditControl>

</PropertyGroup>

<Files>

<File name="Manifest.xml" src="模板Manifest相对地址" />

</Files>

</Project>

</Template>

1. Mainfest.xml新结构

<?xml version="1.0" encoding="utf-8" ?>

<Manifest xmlns="">

<Component RobotName="" RobotType="GUID字符串">

<Properties>

<Property name="ID"></Property>

<Property name="InstanceID"></Property>

<Property name="Version"></Property>

<Property name="CreateTime"></Property>

<Property name="ModifyTime"></Property>

<Property name="Author"></Property>

<Property name="Description"></Property>

<Property name="FormType">NullForm</Property>

</Properties>

<Parameters>

</Parameters>

<Forms>

</Forms>

<Reference>

</Reference>

<RunTime>

</RunTime>

</Component>

</Manifest>

Mainfest结构说明：

Property存放组件基本信息

Parameters存放组件参数

Reference存放组件文件

RunTime存放组件文本信息

1. 在封装环境安装目录下Config文件夹下Tag\_TargetHost\_Mapping.config添加配置字段

<Mapping>

<Tag>Mainfest中RobotType对应值</Tag>

<TargetHost>步骤3中添加的Id字段</TargetHost>

<RobotType>ScriptType</RobotType>

</Mapping>