# Robot结构改造设计文档

## 概述

针对原Robot结构文件存在信息冗余、结构不紧凑的情况，对Fiper、ModelCenter的模型文件进行了充分调研与讨论，制定了Robot文件结构，主要有两个目录，组件封装生成目录、编译产出物目录。

组件封装生成目录：将原方案此目录下的Additional、Bodies、data、Forms、Interfproxy、Logic、MainBody、Output、Resources、Robots目录及其所包含的内容均删除，并删除此目录下的图片资源文件：Robot\_Image16.png、Robot\_Image42.png；改造后方案仅仅存在一个Reference目录（存在组件引用时，生成此目录）。

编译产出物目录：原方案下的InDefaultFile目录及其所包含的内容均删除，删除此目录下的图片资源文件：Robot\_Image16.png、Robot\_Image42.png，将input.xml、output.xml文件重命名为in.xml、out.xml。

此次改造最大的亮点在于组件封装生成目录下的Manifest.xml，形式更紧凑，内容更饱满，涵盖了组件类型、基本属性、参数定义、组件表单、组件引用、组件运行时设置，原本分散存储的组件状态信息，现以分门别类的方式更清晰地展现出来，具有良好的通用性、扩展性、可维护性。

## Robot结构设计



Robot结构

封装环境组件保存编译产出物的bot文件结构如上图所示，由三大部分组成。

Manifest.xml：组件模型文件，具体描述见下文

\*.flexproj：组件工程文件

Reference：组件关联资源目录



Manifest.xml

组件基于模板创建，因此作为组件模板的重要组成部分Manifest.xml文件，必须具有足够的概括性、通用性、可扩展性。以公式组件的Manifest.xml文件为例，进行详细说明。

组件类型：

RobotType

基本属性：

ID为组件标识符，InstanceID为组件实例ID，Version为组件版本，CreateTime为组件创建时间，ModifyTime为组件最后一次保存时间，Author为作者，Description为组件描述，FormType为表单属性

参数定义：

Id为参数唯一标识符，name为参数名称，direction为参数输入输出特性，type为参数类型，desc为参数描述，format为参数格式化字段，value为参数值，listname为参数列表名称，group为参数组名称。

组件表单：

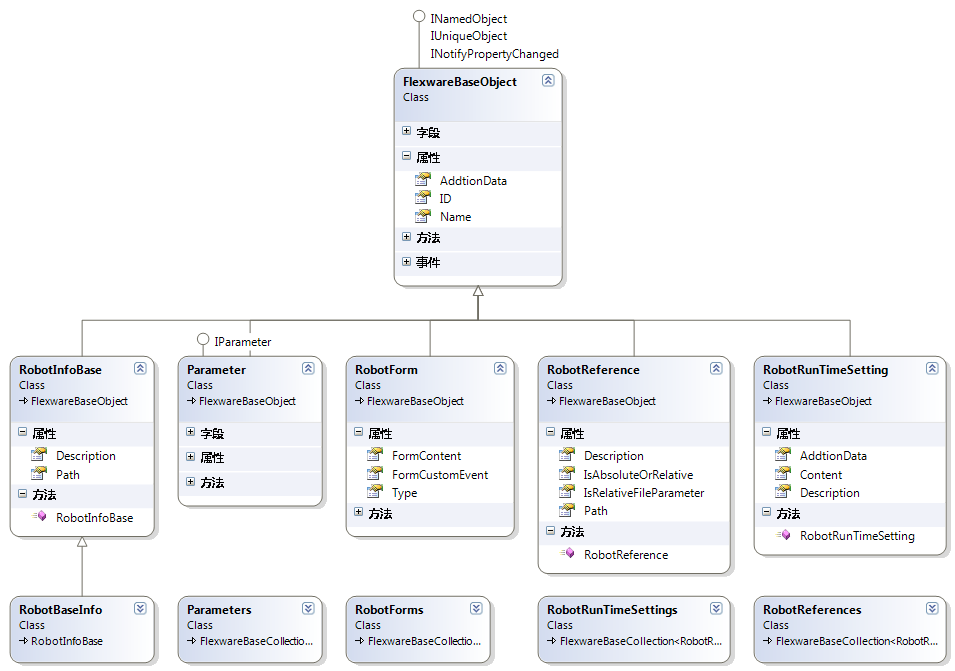
主要用于存储表单内容及用户事件代码等

组件引用

主要用于存储组件引用的外部文件信息等

组件运行时设置

主要用于存储组件内部的状态信息及其运行特性设置等



Manifest.xml关联类图设计

## Robot创建



Robot创建时序图