# Testflow平台的开发说明

Testflow的开发从功能上分为两部分：Testflow框架和Testflow软件。

Testflow的开发基于依赖倒置的范式：首先定义Testflow框架内部和对外的接口，然后框架和软件可以同时基于这些定义的接口各自进行上层的开发：框架开发时需要实现接口定义的功能，软件开发时，需要依赖接口定义的组件实现功能。最终两个部分通过TestflowRunner作为实例化的接口，实现功能的对接。

# Testflow框架的开发设计

Testflow框架需要完成的组件主要包括：Testflow框架组件、Testflow设计时组件、Testflow运行时组件、TestflowRunner、Testflow用户组件接口及内置组件。

## Testflow框架组件

Testflow框架组件提供最底层的功能实现，主要包括：

* 日志模块，模块控制接口为Testflow.Modules.ILogService，该模块可能被所有框架接口和用户组件依赖。
* 组件接口管理模块，模块控制接口为Testflow.Modules.IComInterfaceManager
* 序列管理模块，模块控制接口为Testflow.Modules.ISequenceManager
* 参数检查模块，模块控制接口为Testflow.Modules.IParameterChecker
* 引擎控制模块，模块控制接口为Testflow.Modules.IEngineController
* 数据维护模块，模块控制接口为Testflow.Modules.IDataMaintainer，该接口为引擎模块和结果管理模块依赖(因为涉及到具体细节，暂时还没定义完成)
* 结果管理模块，模块接口为Testflow.Modules.IResultManager，该接口涉及到细节，暂时还未定义完成。

在框架组件中除了需要实现模块控制接口，额外需要实现的接口包括：

|  |  |  |
| --- | --- | --- |
| 接口 | 关联模块 | 说明 |
| Testflow.Data.Sequence命名空间里所有接口 | 序列管理模块 | 所有和序列数据相关的接口定义 |
| Testflow.Data.IAssemblyInfo  Testflow.Data.IAssemblyInfoCollection  Testflow.Data.ITypeData | 序列管理模块 | 组件和类型的相关信息 |
| Testflow.Data.Description所有接口 | 组件接口管理模块 | 所有和组件描述信息相关的接口定义 |

各个模块需要完成的功能及实现说明如下所述：

### 日志模块

日志模块提供在设计时的框架日志和运行时的运行时日志，两个相互独立。

框架日志为普通的日志模块，提供LogLevel的配置，

### 组件接口模块

组件接口加载模块提供加载组件接口描述的功能，维护当前所有被加载的描述接口和索引号的映射。使用Assembly加载程序集时最好在另外的AppDomain执行(如果加载到当前的AppDomain会无法即时卸载，会严重增加整个应用的资源占用)。

需要实现的方法及说明如下：

* GetComInterfaceById：根据ID号获取对应的组件接口描述信息，如果没有返回null
* GetComponentInterface：根据路径或者程序集信息获取组件接口描述信息，如果没有返回null
* GetComponentInterfaces：根据路径集合和程序集信息集合获取一组组件接口描述信息，返回的List长度和路径集合长度相同，如果某个未加载成功，对应位置返回null
* GetPropertyType：根据ITypeData和属性名获取对应属性的类型信息。
* GetTypeProperties：获取某个类型的所有属性名

接口描述信息需要包括的内容如下，这些内容和接口定义一一对应：

* 该组件接口在全局的唯一ID
* 组件的签名，该字符串格式为“程序集名称-版本号”
* 程序集信息
  + 程序集名称
  + 程序集描述信息
  + 程序集所在路径，包括文件名
  + 程序集加载是否成功
  + 程序集版本号
* 类接口列表
  + 类名
  + 类描述信息
  + 类所在组件接口的全局索引号
  + 类在当前组件中唯一的类ID
  + 类的类型信息
    - 类的类型信息
    - 所在程序集名称
    - 命名空间
    - 类名
  + 方法接口列表
    - 方法名
    - 方法描述信息
    - 方法类型(构造方法/实例方法/静态方法)
    - 方法所在组件的接口信息的全局索引
    - 类在组件的ID号
    - 是否泛型方法
    - 方法返回值信息
    - 入参信息列表
    - 方法签名，格式为“类名.方法名(入参类型列表)”

## Testflow设计时组件

设计时服务提供序列编辑服务，这些功能承载于底层组件之上，实现测试序列的直接设计与增删操作。设计时组件需要实现Testflow.DesignTime命名空间下的所有接口，主要包括：

* IDesignTimeService：设计时服务，完成设计时TestProject级别的控制和功能管理
* IDesignTimeSession：单个测试序列组的设计时服务，完成设计时SequenceGroup级别的控制和功能管理
* IDesignTimeContext：单个DesignTimeSession的上下文信息。

## Testflow运行时组件

运行时服务提供序列运行服务，这些功能承载于底层组件之上，实现测试序列组的运行控制、运行时调试、状态维护、结果处理。运行时服务需要实现Testflow.Runtime命名空间下的所有接口，主要的三个类包括：

* IRunTimeService：运行时服务，完成运行时TestProject级别的控制、状态维护和结果处理。
* IRunTimeSession：单个测试序列组的运行时服务，完成运行时SequenceGroup级别的控制、状态维护。
* IRunTimeContext：单个RunTimeSession的上下文信息。

## TestflowRunner

Testflow平台运行器，维护一个Testflow运行实例的所有功能的实例，包括所有组件和服务的创建/获取/销毁操作。该类为抽象单例类，在单个AppDomain中通过GetInstance方法可以获取唯一的实例，该方法的入参是TestflowRunnerOptions参数类，用以配置平台参数。该类中包括的主要属性有：

|  |  |
| --- | --- |
| 属性名 | 说明 |
| RuntimeService | 运行时服务 |
| DesignTimeService | 设计时服务 |
| ComInterfaceLoader | 组件接口加载器 |
| DataMaintainer | 数据维护组件，实现数据的持久化等功能 |
| EngineController | 运行引擎控制器 |
| ParameterChecker | 参数校验组件控制 |
| ResultManager | 结果管理组件 |
| SequenceManager | 序列管理组件 |
| LogService | 日志服务 |
| I18n | 国际化组件 |

该类需要依赖所有已实现的组件，所以GetInstance方法需要待所有组件和服务开发完成后实现。

## Testflow内置组件及用户组件接口

# Testflow框架的功能规划

# Testflow软件的设计规划

Testflow软件需要完成的组件主要包括：序列管理面板、设计时面板、运行时面板、结果管理面板。