# Robot Framework关键字库实现模式

## 简介

Robot Framework是一个**开源**的、**通用型**的、**关键字驱动**的自动化框架。本身基于python开发，兼容cpython、jython、ironpython等不同python实现。初始的时候，是针对ATDD（Acceptance Test Driven Developement）开发模式设计，后续又添加对于BDD（Behavior Driven Developement）和RPA（Robotic Process Automation）的支持。

Robot Framework作为一个通用型的自动化框架，具备较强的扩展性，网上有大量的外部关键字库可供使用，覆盖了界面测试、接口测试、数据库、APP、通信协议等各种场景。但具体到项目中的业务测试时，还是需要有专门的自动化组、或者具备代码能力的测试人员来实现业务层面的关键字库。

Robot Framework的中文资料很多，但都是以如何使用为主，而较少介绍如何实现关键字库；而Robot Framework本身虽有完备的文档介绍，但都是英文资料，学习起来还是有些困难。因此我想基于自己的经验，向大家介绍如何基于python开发Robot Framework关键字库，同时还会介绍Remote Library接口，如有不准确的地方，请大家提供建议。

## 关键字库的三种实现模式

对于了解python，也有关键字库开发经验的人来说，会有些诧异。开发一个关键字库有什么难的，***定义一个类，实现几个方法，导入到Robot Framework的用例中，就可以使用了***。嗯，是的，确实就是这么简单，这个就是下边要讲的Static Library模式，同时还想介绍Dynamic Library模式、Hybrid Library模式，如果你没有听说过这些概念的话，不妨可以往下再看看：）。

### Static Library模式

* + 1. 基于module的关键字库
    2. 基于class的关键字库
    3. Keyword装饰器
  1. Dynamic Library模式
     1. get\_keyword\_names方法
     2. run\_keyword方法
     3. 非必须API方法
     4. Dynamic Library例子
  2. Hybrid Library模式
     1. \_\_getattr\_\_介绍
     2. Hybrid Library例子

1. Remote Library接口
2. 参考资料