

## 20 如何用代码框架实现接口测试?

更新时间: 2019-10-03 12:14:28



“学习从来无捷径，循序渐进登高峰。——高永祚”

在我打过交道的同学朋友中，使用工具来完成接口测试的几乎占到了八成以上，我们前边也看到了诸如 Postman、Jmeter 等工具在接口测试上有着非常强大的能力，但是实际工作中我们会发现，是所谓“成也萧何败也萧何”，使用这样界面化的工具在维护大量测试案例或者想要执行持续集成又或者希望编辑更多测试场景的时候，界面化又成为了测试的瓶颈，所以我们需要一套更优秀一点的接口测试框架。

其实相关的接口测试框架有很多，不过呢，在这篇文章里我们不谈某一个接口测试框架应该怎么用，而是通过我自己的经历来看看接口自动化测试框架的发展史。

### 一代：基于 Postman 的扩展框架

鉴于最早的时候开始进行接口测试一直在使用的工具都是 Postman，觉得这个工具真的是又便捷又好用，所以在最早想要扩展出一套接口测试框架往往考虑的问题是：能不能在现有的基础上加一点点“佐料”？

于是便有了 Jenkins+Newman+Postman 这样的一套接口自动化框架。毫无疑问 Jenkins 负责的是持续集成，而 Newman 则是作为 Postman 的命令行集合运行器，可以说 Newman 是完全为了 Postman 而开发的，所以兼容性、可扩展性都是非常 OK 的，同时也具有比较好的文件上传、测试结果展示的功能，可以算得上是开始进行接口框架建设的神器。

这样的一套框架的确在前期解决了我们很多问题，毕竟很多接口脚本就一直写在 Postman 里边的，但是当我们的接口复杂度不断增大、希望实现的功能越来越多的时候，Postman 这样非开源工具性产品就展现出了它的不足：很难扩展插件、无法与数据库配合、校验方式也比较死板、接口变动修改起来呢也很麻烦。

Postman 天然的封装让我们可以很轻松的进入接口测试，但是不开放的属性也就是让我们为之头疼。

## 一代 PLUS：基于 Jmeter 的扩展框架

提前说明一点，这里边跟大家聊一下 Jmeter 的接口测试框架，并非是基于 Postman 的升级，仅仅是以我个人经历的时间线为准，所以我的标题里加上了 PLUS。

这是由于一个比较特殊的项目，项目团队呢一直都在用 Jmeter 进行接口测试，等到我进入项目中已经使用 Jmeter 编写了超过 2000 的测试用例，所以做任何架构和工具上的调整都成为了不可能，唯一我能做的就是基于 Jmeter 之上的封装。唯一值得开心的事情是相比较 Postman 来说，Jmeter 是开源的，这样集成和修改上更容易一些。

所以由此以 Jmeter 为核心构建了 Jenkins+Maven+Jmeter 的简单结构化框架，复用 Jmeter 自身的测试结果来辅助实现。这也实实在在解决了当时 Jmeter 脚本执行的需求，但是尽管 Jmeter 的接口测试封装设计思路非常好，但是它的 UI 设计和实现逻辑实现的实在太复杂，为了通用也做了很多过度设计，所以操作、设计、维护接口上都不是太方便。

在当时我就想到，既然像 Jmeter 的 if 控制器等都已经细节化到这个程度了，我们何不通过代码来实现呢？这个时候我已经搭建了前面谈过了 Web 自动化测试框架，所以在这个基础上，进行了进一步的改进，加入了接口测试的内容。

—— 代码库构建与接口测试框架

让我们一起回忆一下前边介绍过的 UI 自动化测试框架：

Jenkins + Maven + TestNG+ Selenium + ReportNG + JAVAMAIL+ ExtentX

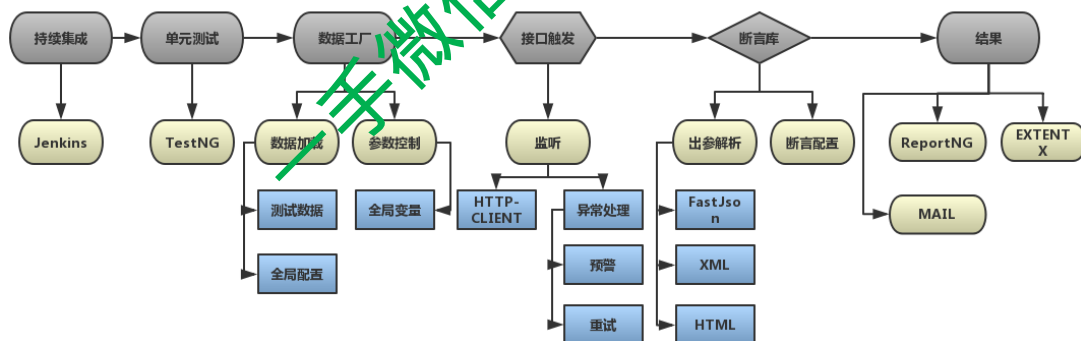
所以这里边我们可以做一个简单的改造，讲 UI 的 Selenium 替换为 API 测试的依赖。那我们在 JAVA 里，主要进行接口测试的类为 HTTPClient 或者 URLConnection，于是这个框架就改造成了

Jenkins + Maven + TestNG + HTTPClient/URLConnection + ReportNG + JAVAMAIL+ ExtentX

尽管看起来框架的改造并不太大，然而实际上思路上的变化还是有的，我们把框架定位成为：

- 数据驱动设计，无需编写代码实现接口自动化
- 自动化测试用例完全用数据文件编写，简单易用高效

我们来看看这样一套框架的整体架构设计思路：



这就是很多公司内部应用的接口自动化测试框架模式，相对比较成熟，当然，针对不同的公司业务，会有不同的业务上下文和业务属性，所以在这个比较成熟的测试框架基础之上来封装属于自己公司的接口测试框架，让接口测试更加灵活和适配。

我在慕课上也非常详细地分享过这套 UI 框架的代码思路，对于接口也需要一定的修改，然而涉及代码的讲解，可能不是一篇文章可以快速讲清楚的，所以延续着本专栏的模式，更多还是给大家这样的指引。如果有同学说：网落，我不想用 JAVA 啊，只会 PYTHON 怎么办？其实加个框架

想是不变的，只是组件上会发生一些变化，将上述的核心组件变为 Unittest+Requests，有兴趣的同学可以研究一下哈。

## 三代：开源框架平台的利用

能够自主搭建一套框架当然是很好的，但是一套优秀的框架，从诞生到改进、完善还有很长的路要走，所以搭建框架并不是“来之能战”的，最近几年开源的优秀接口框架非常多，也可以利用这样的开源框架进行搭建、改进甚至二次开发，避免重复造轮子，也成功借鉴了前人的经验，同时也可以对自己搭建框架有更强的指导意义，更可以取长补短。

### HTTPRUNNER

这是一个非常简洁的 Python 接口自动化测试框架，易用性非常高，用例维护主要是基于 **YAML/JSON** 脚本，不仅可以实现接口自动化测试、还包含性能测试、线上监控等多种测试需求。给大家一个官方传送门：

在我接触过的接口框架中，HttpRunner 是最容易上手的，二次开发也比较容易。

### Rest-Assured

如果要我在所有接触过的接口测试框架中只能选择一个的话，那么 Rest-Assured 一定是我的不二之选。由于越来越多的 Web 应用转向了 RESTful 的架构，很多产品和应用暴露给用户的往往就是一组 REST API。所以针对于 REST API 测试的框架也应运而生，它可以直接编写代码向服务器端发起 HTTP 请求，并验证返回结果；它的语法非常简洁，是一种专为测试 REST API 而设计的 DSL。使用 Rest-Assured 测试 REST API，就和真正的用户使用 REST API 一样，只不过 Rest-Assured 让这一切变得自动化了。直接上传送门：

之所以推崇这样一个框架，主要在于其支持的功能非常完整，几乎可以满足我们对于接口测试 90% 以上的需求。在我自己设计框架的过程中，有不少的功能也借鉴了 Rest-Assured 框架的理念。

有同学要说了：不是说不要重复造轮子么，风落你打自己脸啊！！NoNoNo，作为一名程序猿，就是喜欢重复造轮子，这样才能提高我的代码能力，当然，如果要造，一定要造一个比它更好用更适合你的轮子 O (∩\_∩) O

## 平台

更有一些界面管理化的平台可以支持我们进行更好更容易的接口自动化实施，更进一步减少了对代码编写的需求。例如前边聊过的开源的 AutoLink、LuckyFrame 都集成了接口测试，同样还有针对于接口测试更专业化的 YApi，思路与我们前边聊到的自动化平台建设并无太大的不同。

## 不是结束的开始

接口测试当然越来越重要，所以未来也许还会有四代、五代，也可能会融入人工智能甚至是我们现在看来还不能想象到的技术。无论你是开始基于工具来扩展一套框架，或者是基于开源的框架平台进行应用，又或者是开始自己编写一套适合自己的新“轮子”，你都要先这么问自己：

- 我的框架目的是什么？
- 我或者用户想要怎么使用？
- 这是不是最适合我的改进？

然后再动手，自由发挥。没有哪种是最好的解决方案，只有最适合你的方式，所以我在这里并没有指给你哪个最好，也没有给你分析框架应该怎么写，开源框架 ABCD 应该如何去用，而是展示给大家要设计一个框架，应该怎么去思考。也许今天在看这篇文章的时候你还没有完全理解，没关系，做下去，等到困惑的时候再回来看看，想想，希望能对你有所帮助。

← 19 从零开始了解接口测试与工具

21 求同存异：论移动测试与传统测试 →

### 精选留言 4

欢迎在这里发表留言，作者筛选后可公开显示

#### 简单随风

目前公司项目用的Springboot+Vue搭的接口自动化平台。想请教一下，因为之后想压测和服务器的实时监控(Vue上渲染)，这个监控有什么现成的框架可以用的么？

👍 0 回复

2019-11-15

回复

2019-11-18 14:39:01

**风落几番**

我先补发一下传送门，再问问传送门被吃的真相~~ [HttpRunner]: <https://github.com/httprunner> [Rest-Assured]: <https://github.com/rest-assured/rest-assured> [Rest-Assured中文文档]: <https://github.com/RookieTester/rest-assured-doc>

👍 2 回复

2019-10-09

**风落几番**

我先补发一下传送门哈~~再问问为啥我这么简单的传送门被吃掉了~~~ [HttpRunner]: <https://github.com/httprunner> [Rest-Assured]: <https://github.com/rest-assured/rest-assured> [Rest-Assured中文文档]: <https://github.com/RookieTester/rest-assured-doc>

👍 0 回复

2019-10-09

**慕丝8558034**

官方传送门去哪了???

👍 0 回复 举报

2019-10-08

**风落几番** 回复 **慕丝8558034**

要传送哪个? 我帮你手动传送下~~

回复

2019-10-09 10:37:43

**土豆稀饭** 回复 **风落几番**

rest-assured 和 httprunner 的官方传送门，写了给传送门，但是没给地址

回复

2019-10-09 15:12:21

干学不如一看，干看不如一练