

谈自动化测试框架思想与构建

序言：也许到现在大家对所谓的“**自动化测试框架**”仍然觉得是一种神秘的东西，仍然觉得其与各位很远；其实不然，“**自动化测试框架**”从理念来说，并不复杂，但其之所以神秘，是因为其运用起来很是复杂，每个公司，每个部门其产品线，其运作流程都是不同的，所以就导致了在想运用“**自动化测试框架**”去完成自动化测试时产生了很多不定因素，导致了很多自动化测试项目的失败，让人对“**自动化测试框架**”开始敬而远之。

而自动化测试发展也有一段时间了，为什么到现在虽见其火热，但难见其规模，关键是大家对其的定位，很多公司以及很多人都知道做好自动化测试不简简单单的靠一个工具，而更需要一个框架，但其总是对“**自动化测试框架**”缺乏清晰的定位，很容易将其定位成了一个固定的框架，其实个人理解不然，自动化测试框架不是一个模式，而是一系列思想的集合，是将各种自动化测试框架思想集合应用去搭建成的一个分层组织。

一、简述自动化测试框架

也许很多人印象里的自动化测试框架就是一个能够进行自动化测试的程序似的。其实这不全面，真正的自动化测试框架可以不是一个程序，它仅仅是一种思想和方法的集合，说白了，就是一个架构，大家应该都知道操作系统其实也是一个架构吧，你可以把其理解成一个基础的自动化测试框架为一个简单的操作系统，它定义了几层架构，定义了各层互相通信的方式。通过这个架构我们才能在上面进行拓展我们的测试对象(核心体)、测试库(链接库)、测试用例集(各个 windows 进程)、测试用例(线程)，而其之间的通过参数的传递进行通信(即相当于系统中的消息传递)。

二、自动化测试框架思想

接触过自动化测试的，一定不会对以下几种“**自动化测试框架思想**”陌生吧。

- 模块化思想
- 库思想
- 数据驱动思想
- 关键字驱动思想

很多人都将以上定义为“**框架**”，而我却觉得它们都只是代表了一种自动化测试的思想，不能以纯粹的**框架**定义。

首先，我们来看看自动化测试的一个发展，就能更加明白这些思想的真谛了。

a) 第一代自动化测试，即自动化测试思想刚开始诞生时，依靠的是传统的“录制-回放”技术，这种技术与现在的工具的“录制-回放”思想不一样，其其实

就是一个“模拟”的过程，即模拟你对 PC 的操作而形成的，其基于你对键盘的输入与对鼠标的操作，原理与按键精灵等类似，这种机制对环境的依赖性太强，对变化性太过于敏感，因此不可能发展成一种规模。

b) 第二代自动化测试，即脚本化的自动化测试，利用脚本进行结构化的自动化测试，此可以应用于 CLI 与 API 的自动化测试，在其就开始集成了模块化与库思想。

c) 第三代自动化测试，开始产生了各种自动化测试思想，包括数据驱动与关键字驱动思想，其伴随着对象化思想的产生，而且也造就了现在一系列的自动化测试软件，其实其中都集成了这些思想，从这时候开始，自动化就开始实现了一定的规模，开始运用在各个行业，并且发展趋势越来越快。

现在将一一根据自己的个人理解来介绍这些“自动化测试框架思想”：

1、所谓模块化思想，就是将一个测试用例中的几个不同的测试点拆分并且将其单个点的测试步骤进行了封装，形成了一个模块。

例如：一个测试用例要对一个登录程序进行测试，其中包括：用户名输入、密码输入、以及确定登录；

那么就可以将用户名输入、密码输入、确定登录、取消登录四个操作分别封装在四个不同的模块中。测试时，只需调用其模块即可。这样的话，当一个模块有变化，你只需单独维护那个模块即可，也可以根据模块的不同组合成不同的测试用例。

2、所谓测试库思想，就是模块化思想的升华，其为应用程序的测试创造了库文件（可以是 APIs、DLLs 等），这些库文件为一系列函数的集合。其与模块化思想不同的是，其拓展了接口思想，即可以通过接口去传递参数，而不是一个封死的模块，可以说是一个多了一个“门”的交互型模块。

例如：还是以上那个测试用例，只是将用户名输入、密码输入、确定登录、取消登录封装成一个库，这个库含有一个函数 Login，这个函数 Login 接收两个参数“用户名、密码”，对输入不同的用户名和密码可以进行不同的测试用例。也可以另外一个函数 Cancele。

3、所谓数据驱动思想，众说纷纭，很多人都觉仅仅依靠用 EXCLE 表进行不同数据的读取仅是一个高级的参数化，其实怎么理解并不重要，关键是其思想能够好的应用到你的框架中。而我的理解就是变量不变，数据驱动结果，不同的数据导致了不同的结果的产生。而对于数据的导入，可以通过很多方式，例如：EXCLE 表、XML（用在 WEB 中）、数据库（DB）、CSV 文件、TXT 等都可以。

4、所谓关键字思想，这个思想，我曾经一直思考，它与面向对象的关系，与交互模块化思想的区别。后来个人理解，其实关键字驱动就是一种面向对象的思想，例如：QTP、RFT 中，对象可以为一个数据或者一个关键字，对对象的抓取，

可以将其测试对象封装为一个关键字（即可以将 gui 元素封装成了一个关键字），这样可以对其关键对象进行各种操作了，不同的对象可以驱动不同的测试流向与结果。

简单的应用的方式可以用一个 EXCEL 表，里面包括“对象类型”“对象名称”“对象操作名称”“判断方式”“预期结果”。这样的话，可以通过导入不同的对象类型和名称、不同的对象操作来构建成了一个测试用例表了。

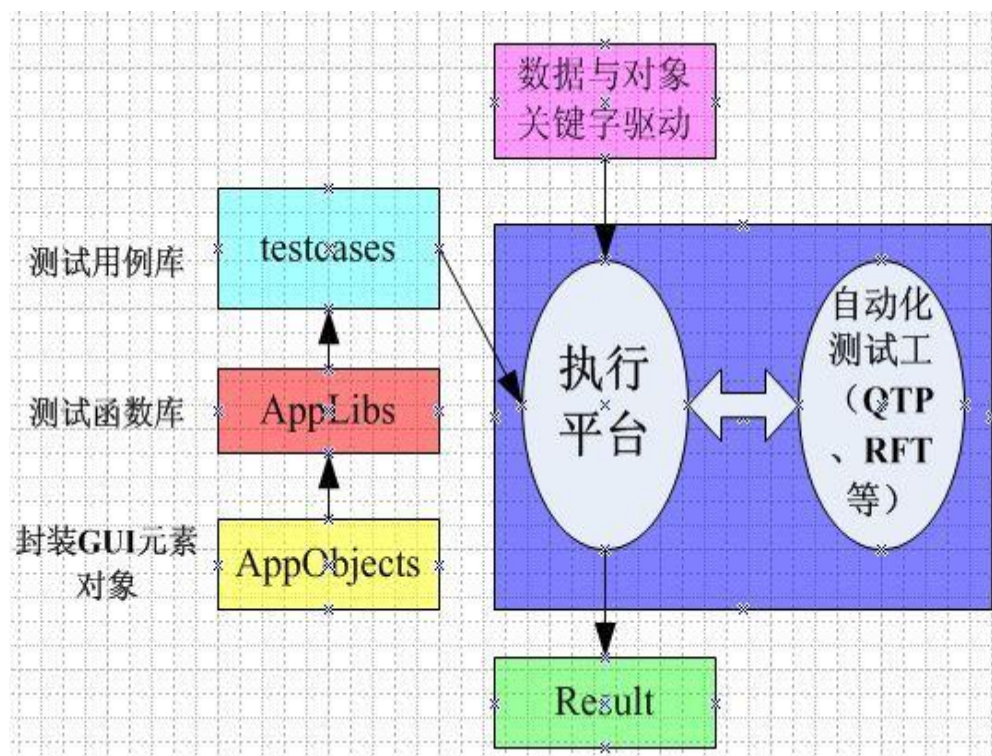
以上只是对这些思想的个人理解，做好自动化测试，不是说你掌握了一个框架，而是要掌握其自动化的思想，然后根据这些思想，结合你不同的测试环境和流程来构建你自己的自动化测试框架。

三、构建自动化测试框架的策略

1、永远记住，你的“自动化测试框架”是给测试人员用的，如果你真的想把自动化测试做成一个规模，那么你需要将测试工程师当做你的用户，你不能指望他们有耐心的去编写测试脚本或者指望他们能够像你一样对这些思想有良好的掌握。你要将他们当成什么都不懂的用户，因此你的框架必须是“一切简单化”的化身，简单的操作、简单的维护、简单的拓展。

2、做一个自动化测试框架主要是从分层上去考虑，而不是简简单单的应用一种思想，它是各种思想的集合体。

例如，做 GUI 自动化测试，简单的一般就将其分为三层，其框架如下图所示：



而其中，可以贯穿着自动化测试的各种思想，例如：对象层中有关键字的思想、可以将对象库标示在 Excel 表中进行管理，或者应用动态搜索的方式传递对象识别参数。tasks 层中可以封装各种方法，形成一个大型的方法库，而每个方法中可以应用上数据驱动的思想。

3、真正的自动化测试框架是与流程上结合的，而不简简单单的靠技术实现，技术其实不是很复杂，关键就在于对其架构和流程的深刻把握，而这需要很长的一段时间，所以不要指望一口气能吃成胖子，只能一步一步按需求来，需求指导思想的应用。

四、自动化测试框架的发展趋势

个人认为，自动化测试从初始诞生到至今，已经经过了一段漫长的日子，而其仍处于上升期，特别是现在软件大爆炸、敏捷模式、云端的开始热门，测试难度和质量保证的难度开始越来越大，自动化测试的比重也会越来越大，而单存的自动化测试是无法实现规模化的，因此，自动化测试框架热门化的趋势化的必然的，那是，在各种框架思想的集合中，各种框架将散发出各自的璀璨，来帮助我们快速的完成各种测试。

以上仅仅是至今，个人对“自动化测试框架”的理解，也许在以后的日子，因为认识的加深而会有不同的火花蹦出，但至少觉得现在的框架对自己的项目能够进行应用，也许某一天，需求饱和时，那么，新一轮的远征探索就又要开始……

希望，我们大家在自动化测试的征程上能越走越远，也希望自动化测试能真正成为测试流程中“不可缺少”的一部分。共勉之。