

自动化测试概述(二)-自动化测试人员构成

为什么要分构成？

脚本执行工程师

脚本开发工程师

脚本接口(DLL)工程师

工具开发工程师

生命线

为什么要分构成？

很多人看到标题一定很疑问，自动化测试人员还有不同的构成么？不都是代码一敲，脚本一跑，等着收结果么？前文已经说过，当前软件开发已经从"小大小闹的作坊式"变更为"大规模的工业式"开发，各个开发人员各司其职，对应的自动化测试技术和模式也不断向前发展，因此人员其实也有不同的定位和区别。

按照一份脚本的生命周期，从被创造到被执行，根据职责，我大致分为4类(我自己分的):

1. 脚本执行工程师
2. 脚本开发工程师
3. 脚本接口(DLL)工程师
4. 工具开发工程师

脚本执行工程师

处于脚本周期的生命末端，其职责包含环境配置、脚本执行、脚本结果收取、脚本结果分析、问题汇报等；当脚本脱离了创造对象(开发者)后，后续所有的任务均是脚本执行工程师的职责。

脚本开发工程师

处于脚本周期的中端，脚本的执行编写和验证者；其职责包含测试脚本的开发、脚本库的管理，环境配置脚本的开发、优化；是脚本的直接创造者，在一定的规则之下编写自动化测试脚本。

脚本接口(DLL)工程师

处于脚本周期的后端，定义脚本内使用的方法、函数、属性、对象、接口等；开发完整的脚本方法库；

脚本本质上只是一种逻辑表现，其呈现的仅仅是测试逻辑,某种程度说脚本自身就是纯粹的文本，无论是用何种语言编写；

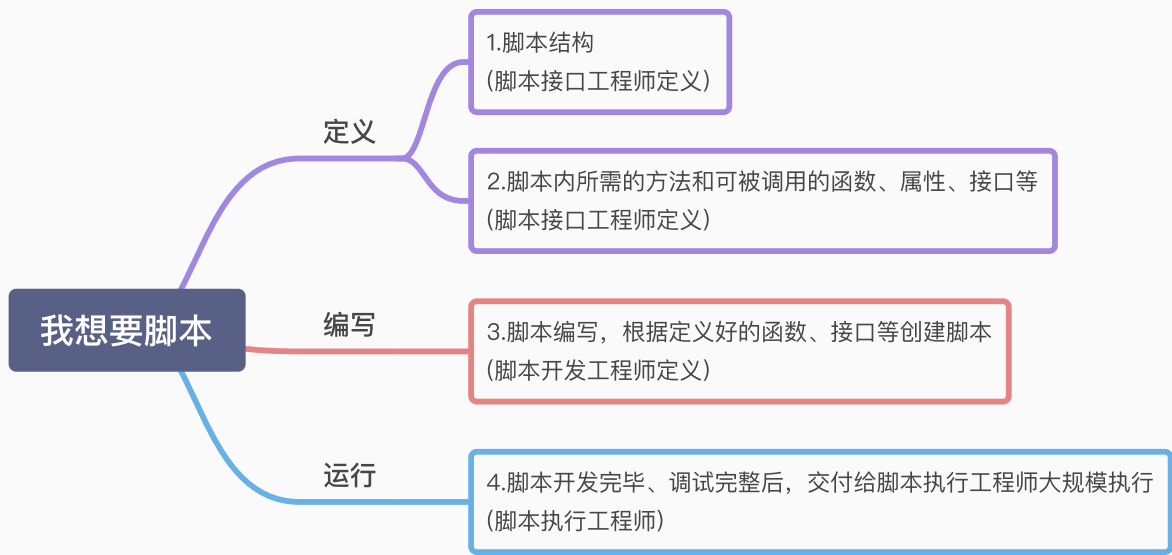
脚本开发工程师是编写的测试逻辑；脚本接口工程师则是编写的内在的调度方法！

工具开发工程师

完整的测试工具开发者，包含执行工具，结果工具以及其他；脚本即使被开发出来，也仅仅是一个纯粹的文本，因此需要工具进行执行，这时就需要通过工具来执行脚本并获取结果；
工具开发工程师的职责就是开发与创造相应的执行工具。

生命线

以上就是我定义的人员构成；实际在生产过程中，公司不允许或不可能单独为一个自动化测试小组划分这么多的人员，因此往往脚本执行与开发合并；脚本接口和工具开发合并；例如经费不足的小企业、可能四个职责都是一个人做；
无论有多少人构成的自动化测试小组，人员构成往往都是这四类；



从图内看到，脚本周期从定义到运行，工具开发工程师并没有参与，工具开发工程师更像一个"乙方"，由其他三方提出自己的需求，来创建测试工具，可以说是一个更加偏向纯粹的开发。

一个脚本的生命从被创造到被执行，其自身周期已经完结，对应的是尾端的结果分析和不断迭代循环。

