## 开篇词 | 从“小工”到“专家”，我的软件测试修炼之道

第一，自动化测试在软件质量工程中的地位发生了质的变化，从原本的“以自动化测试为辅”变成了“以自动化测试为主”。

所以，你不仅需要从业务本身出发来对软件进行手工测试验证，还需要掌握完整的自动化测试开发技术来设计自动化测试用例。

第二，传统软件企业的产品发布通常以“月”为单位，因此，测试执行总时间不会成为关键问题。但是，对于互联网企业，尤其是大型电商网站，产品上线周期通常都是以“天”甚至是以“小时”为单位，留给测试的时间非常有限，这也就对测试执行总时间提出了极大的挑战。为了解决这个难题，你需要一套完善的高并发测试执行基础架构的支持。

所以，作为测试工程师，你就必须掌握设计开发测试基础架构的关键技术。

第三，随着自动化测试的规模化，测试数据准备的各种问题被逐渐暴露并不断放大，成为影响自动化测试效率以及稳定性的“拦路虎”。早期的传统测试数据准备方法，无论是从测试数据准备的时间成本，还是从测试数据的稳定性和测试数据创建的便利性上看，都已经很难适应大规模自动化测试的要求。

所以，你必须系统性地思考如何才能将测试数据的准备工具化，服务化，最终实现平台化。

第一步，成为互联网时代合格的测试工程师。

如果你是入行不满 3 年的测试工程师，一定对此有迫切需求。此时，你必须具有快速学习的能力，能迅速掌握被测软件的业务功能与内部架构，并在此基础上运用各种测试方法，尽可能多地发现潜在缺陷，并能够在已知缺陷的基础上进一步发现相关的连带缺陷。

从知识体系上看，你需要有比开发人员更全面的计算机基础知识，还需要了解互联网的基础架构、安全攻击、软件性能、用户体验和常见缺陷等知识。从测试技术上看，你需要能够使用常见的测试框架或者工具，需要具有一定的自动化测试脚本的开发能力，这可以把你从大量重复的工作中解放出来，然后你才能有时间去做更有意思的工作。

第二步，成为互联网时代优秀的测试工程师。

如果你想从“合格”变为“优秀”，那必须先认识到两者的差距在哪里。

首先，合格的测试工程师关注的是纯粹的测试，而优秀的测试工程师关注更多的是软件整体的质量，需要根据业务风险以及影响来制定测试策略，有效控制测试的时间和成本，并且能够对测试框架以及工具做出适合项目需求的选型。

以新房装修为例，合格的测试工程师就是各个工序的装修师傅，他们只管按照设计要求做好自己的工序，而优秀的测试工程师更像是个包工头，他们关心的是整体交付的质量。

其次，优秀的测试工程师不仅可以娴熟地运用各类测试工具，还非常清楚这些测试工具背后的实现原理，以及多个同类测试工具各自的优缺点和适用场景。

在遇到问题时，你还需要能够通过二次开发解决工具和框架层面的问题，对于没有合适可用工具的场景，可以自行设计开发一些小工具来更好地展开测试工作。

当然这个阶段，你很有可能会接触到一些代码级的测试，这就要求你具有一定的开发背景，并能够很好地理解代码级的测试技术。

最后，随着自动化测试用例的不断增长，自动化测试的关注点也从原本的“如何把手工测试步骤用自动化脚本实现”变成了“如何构建低维护成本，可以灵活组装的自动化脚本”，这就要求你理解自动化脚本的分层设计、页面对象模型以及业务流程模型，并且能够把这些设计应用到你的测试框架里。

第三步，成为互联网时代的测试架构师。

当你经历了各种类型的测试项目，就会发现这些项目本身虽然差异巨大，但是有很多东西是相通的。

比如，面对大量测试用例的执行，无论是 GUI 还是 API，都需要一套高效的能够支持高并发的测试执行基础架构；再比如，面对测试过程中的大量差异性数据要求，需要统一的测试数据准备平台；再比如，为了可以更方便地和持续集成与发布系统（CI/CD）以解耦的形式做集成，需要统一发起测试执行的接口。

这样的例子还有很多，如果你已经能够站在这样的高度看待软件测试，那么恭喜你，你已经具备了测试架构师的视野。当然，你还必须对一些前沿的测试方法和技术有自己的理解，并能够在恰当的时候、因地制宜地把它们应用到实际项目中。

## 01 | 你真的懂测试吗？从“用户登录”测试谈起

作为测试工程师，你的目标是要保证系统在各种应用场景下的功能是符合设计要求的，所以你需要考虑的测试用例就需要更多、更全面

等价类划分方法，是将所有可能的输入数据划分成若干个子集，在每个子集中，如果任意一个输入数据对于揭露程序中潜在错误都具有同等效果，那么这样的子集就构成了一个等价类。后续只要从每个等价类中任意选取一个值进行测试，就可以用少量具有代表性的测试输入取得较好的测试覆盖结果。

边界值分析方法，是选取输入、输出的边界值进行测试。因为通常大量的软件错误是发生在输入或输出范围的边界上，所以需要对边界值进行重点测试，通常选取正好等于、刚刚大于或刚刚小于边界的值作为测试数据。

初级：