

如何保证软件测试的质量

学院：	软件学院
专业：	软件工程
姓名：	马晨夕
学号：	14126126

在生活中，我们常常听说：某某软件好用，某某软件功能全、结构合理、层次分明、语言流畅。这些模模糊糊的语言实在不能算是软件质量评价，特别不能算是软件质量一个科学的、定量的评价。但是，软件质量，乃至任何产品质量，都是一个很复杂的事物性质和行为。

随着计算机和智能终端的普及和深入，使得我们整个社会的经济体系，以至人们的日常生活的各个层面都面对着计算机和手机，特别是人们对软件的依赖性越来越大。互联网的蓬勃发展带动了一系列传统行业纷纷与其结合，走上了 O2O 的道路，人们购物、订餐、打车、洗衣甚至是做美甲，通过一款小小的软件便都能办到。然而软件引起的故障缺越来越多，Bug 已经不再是 IT 行业术语，“启动速度”、“闪退”、“用户体验差”这些词汇对用户来说已经习以为常。软件不同于硬件，软件不会用坏，不存在零件更换问题，但不允许存在误差，不能发生错误，否则后果十分严重。不只是在平时生活方面，由软件质量问题造成巨大经济损失，引发危害人的生命和社会安全事故的例子不胜枚举。医疗系统中的软件错误可能造成生命危险，银行系统中的软件错误会使金融混乱，铁路管理系统中的错误会造成火车追尾。2011 年 7 月 23 日晚上 20 点 30 分左右，北京南站开往福州站的 D301 次动车组列车运行至甬温线上海铁路局管内永嘉站至温州南站间双屿路段，与前行的杭州站开往福州南站的 D3115 次动车组列车发生追尾事故，后车四节车厢从高架桥上坠下。这次事故造成 40 人死亡，约 200 人受伤。经调查认定，导致事故发生的原因是：由于信号系统遭到雷击，产生通信故障，采集数据不再更新，错误地控制轨道电路发码及信号显示，导致了列车追尾。因此，我们需要在软件交付前对软件进行系统的测试，以保证软件的安全性、可用性、稳定性、可靠性。软件测试是保证软件质量的主要手段。

一提到软件测试，很多人会想到反复使用软件，试图在频繁操作中寻找错误发生的场景。其实这是一种错误的概念，软件测试不只是通过用户使用来发现 Bug 的初级阶段。软件测试是在规定的条件下，通过人工或者自动测试等手段，对程序进行操作，以发现程序错误，衡量软件质量，鉴定软件的正确性、完整性、安全性，检验其是否满足规定的需求，或弄清预期结果与实际结果之间的差别的过程。软件测试是为了发现程序中的错误而执行程序的过程。通过分析错误产生的原因和错误的发生趋势，可以帮助项目管理者发现当前软件开发过程中的缺陷，以便及时改进。由于软件测试环节是软件质量的保证，如何保证软件测试的质量就变得至关重要。好的测试方案可能发现迄今为止尚未发现的错误，不好的测试方案可能无法有效的排除和防止软件缺陷与故障，从而给软件带来很多隐患。

为了保证软件测试的质量，有一些基本原则我们必须遵守。首先就是保证软件测试的客观性。对软件测试和软件中的错误必须持客观的态度，避免先入为主，所以一般的软件公司都有第三方或者独立的测试部门，这样既能以揭露软件中错误的态度工作，也能不受发现的错误的影响。其次是要保证软件测试的专业性。软件测试是技术含量很高的工作，既要有软件相关知识，还要具备相关管理能力，这就需要有专业队伍，并经过大量的测试实践。专业化分工是提高测试水平、保证测试质量、充分发挥测试效应的必然途

径。例如：手工测试、自动化测试。由于专业优势，独立测试和分工合作得到的测试结果更具信服力，而测试结果常常与软件的质量评价联系在一起，专业化的独立测试机构的评价更客观、公正和具有权威性。因此我们要保证测试资源，具体体现在经费、人力和时间等方面。一个项目在开发的过程中以及开发结束后都需要投入足够的资源进行测试，以保证如期的交付使用。没有足够的测试资源，软件质量以及交付时间必然会受到很大影响。不能够因为注重软件开发而减少对测试的投入，进而降低测试的有效性。

为了保证软件测试的质量，我们还需要遵循严格的步骤。首先是建立测试计划，即测试“什么”。根据用户需求报告中关于功能要求和性能指标的规格说明书，定义相应的测试需求报告，即制订黑盒测试的最高标准；同时，适当选择测试内容，合理安排测试人员、测试时间及测试资源等。其次进行测试案例设计，即“怎样”测试。测试案例是按一定顺序执行的与测试目标相关的一系列测试步骤。将测试计划阶段制订的测试需求分解、细化为若干个可执行的测试过程，设计将产生测试步骤的输入数据要求、期望输出及其他任何运行测试的有关信息，如环境要求，并为每个测试过程选择适当的测试用例。其次进行测试案例建立，即测试开发。建立一个可重复使用的自动测试过程。包括准备测试脚本、测试具体输入数据、详细的期望输出。测试脚本是具有正规语法的数据和指令的程序，一个测试脚本可以实现一个或多个测试案例。输入数据可以放在脚本中，也可存放在另外的文件中；期望输出也是存放在文件中。接下来就是测试的执行。启动测试脚本，将结果保留在文件中，并对发现的缺陷进行跟踪管理，及时上报。最后一步是测试结果分析。将测试结果撰写成测试报告，与预期的测试结果相比较，并于开发团队进行沟通，解决软件缺陷。

软件测试的根本任务就是通过测试案例的执行来发现软件中的错误。软件的质量依赖于软件测试的质量，软件测试的质量依赖于测试案例的质量。软件测试可以是高质或劣质的，测试案例设计是保证测试质量的关键技术。为保证测试案例的设计质量，有一些原则我们必须遵守。首先是规范性。测试案例设计的工作量是非常巨大的，特别在大系统中，有成百上千个模块，需要成千上万个测试案例。测试案例设计内容的规范性是保证测试案例正确执行的前提，是保证整个测试质量的基础。案例规范包括执行环境、测试目标、执行的前置条件、输入数据要求和期望的输出等。其次就是要保证有效性。测试案例的设计应贯穿到软件开发的各个环节。通常认为测试是在软件已编写完成后进行的。当然没有软件运行测试不可能执行。但是软件的开发不仅仅是程序的编写，而是经历了软件生命周期的各个阶段。由于原始问题的复杂性，软件的复杂性和抽象性，软件开发各个阶段工作的多样性，以及参加开发各种层次人员之间工作的配合关系等因素，使得开发的每个环节都可能产生错误。因此软件测试不是仅仅针对源程序进行，而是应该针对软件开发各个阶段。在需求分析阶段，测试部门根据需求文档编写相应的测试用例；与此同时，在项目原型绘制完毕后邀请用户进行可用性测试，以进一步确认需求。在开发阶段，开发人员针对每一个完成的模块进行相应的单元测试，以保证功能的正确运行和性能。开发完成之后，由测试部门进行集成测试和系统测试，以确保项目质量达到预期标准。第三，测试案例应该具有较高的覆盖率。覆盖率指每个测试案例所覆盖

的测试目标程度，覆盖程度高的测试案例能测试多项的内容，提高单个测试案例的测试效率，减少测试案例的数量。测试案例数量的减少可以避免大量重复性的工作从而节省了时间和人力。第四是经济性。指测试案例的执行、结果分析及程序调试所需要的工作量。如果测试需要大量的人力及时间来进行，并且在测试结束后需要大量的人力和时间来分析处理测试的结果，不能算是一种高质量的测试案例。因此我们需要适当缩减测试案例的数量，同时提高单个测试案例的效率，以实现较高的“性价比”。最后，我们需要保证测试案例的可维护性。软件需要维护，相应的测试案例为了跟上软件的变化也需要维护。维护的工作量是非常巨大的，因此在测试案例设计时要考虑将来的维护工作量。我们可以在将测试内容中的共性内容提取出来进行封装，同时多设接口，让不同的测试内容尽可能的相互独立，这样在以后维护时会节省很多时间和工作量。

在软件越来越普及的今天，对软件产品质量的度量、评估和保证，成了用户和软件开发企业都十分关注的问题。软件测试作为保证软件质量的重要手段，能够找出软件缺陷，确保软件满足用户需求。软件测试只有经过有序地组织、管理，并在各个阶段的每一测试过程中，运用合理的测试原则、方法，才能做到测试最优，从而保证软件质量最优。