

软件性能测试面面观

赵丽莉 金学军 大连) / 文

软件测试,作为软件工程的一部分,随着软件生产的产业化运作应运而生,是软件生产的一个动态监控过程,对软件开发全过程进行检测,可以随时发现问题、报告问题,并重新评估新的风险,设置新的监控基准,并持续下去。软件测试是软件质量控制的过程,是对软件系统中潜在的各种风险进行评估的活动,其目的是监测和排除缺陷,以确保软件产品在可用性、功能性、可操作性等多方面满足质量需求。

目前,软件测试已经由被动的以监测和发现错误为目的发展到从软件质量控制(SQC, Software Quality Control)开始转移到软件质量保证(SQA, Software Quality Assurance),使软件测试从单纯的缺陷检测和发现覆盖到整个软件开发过程,避免了软件开发过程中由于软件需求和设计等方面的缺陷所带来的巨大风险。一个典型的软件过程可以分为测试需求分析、测试设计、测试执行、缺陷和配置管理过程等许多个不同的阶段。在软件测试技术方面也已经被细化为单元测试、集成测试、系统测试、用户验收测试等不同的测试技术。而在对软件产品质量呼声日高的今天,软件性能测试技术尤为重要。

软件性能测试“整体观”

软件的性能测试是为了检验系统或系统部件是否达到需求规格说明中规定的各类性能指标,并满足一些性能相关的约束和限制条件,它必须对系统或系统部件具有的性能(例如,速度、精度、频率)做出规定的要求。

性能测试通常在系统测试阶段执行,常常与强度测试结合起来,一般需要使用测试工具。评估测试对象的性能行为时,可以使用多种评测,这些评测侧重于获取与行

为相关的数据,如响应时间、计时配置文件、执行流、操作可靠性和限制。这些评测主要在评估测试活动中进行,也可以在执行测试活动中使用性能评测评估测试进度和状态。

性能需要在各种条件下测试,这些条件包括:

不同的工作量和/或系统条件。

不同的用例/功能。

不同的配置。

性能需求在补充规格或需求规格说明书中的性能描述部分中说明。

在上述条件下执行测试时,要特别注意以下信息,并为反映这些信息的每条语句生成至少一个测试需求:

时间语句,如响应时间或定时情况。

指出在规定时间内必须出现的事件数或用例数的语句。

将某一项性能的行为与另一项性能的行为进行比较的语句。

将某一配置下的应用程序行为与另一配置下的应用程序行为进行比较的语句。

一段时间内的操作可靠性(平均故障时间或MTTF)。

配置或约束

软件性能测试工作主要包括如下几个方面:

动态监测:在测试执行过程中,实时获取并显示正在执行的各测试脚本的状态。

响应时间/吞吐量:测试对象针对特定主角和/或用例的响应时间或吞吐量的评测。

百分位报告:数据已收集值的百分位评测/计算。

比较报告 :代表不同测试执行情况的两个(或多个)数据集之间的差异或趋势。
追踪报告 :主角(测试脚本)和测试对象之间的消息 / 会话详细信息。

软件性能测试“方法观”

软件性能测试的方法可以根据测试目的的不同,选择不同的方法,具体如下表:

测试目的	测试内容
测试端用户反应时间	完成一个完整的业务过程要多长时间
定义最优化硬件配置	哪种硬件配置提供最佳性能
检查可靠性	系统有错误或失效能工作多长或多坚挺
检查硬件和软件升级	升级对性能或可靠性的影响
评估新产品	应该选择哪一种服务器硬件或软件
测量系统容量	没有明显的性能衰退系统能处理多少负载
识别瓶颈	哪一种元素降低了反应时间

根据测试内容的不同,性能测试主要包括以下方面:

1、响应时间测试

响应时间测试,通常指正常单用户操作时客户端的响应时间,以及将强度测试、负载测试、压力测试结合时客户端的响应时间。

函数、方法、对象、子例程执行时间。

函数、方法调用频度及嵌套。

运行特定模块、按特定路径执行或处理特定数据所花费的时间。

处理精度。

如果两次运行时间相差三倍

以上,则可能存在问题。

2、强度测试(压力 / 负载测试)

强度测试需要在反常数量、频率或资源的方式下运行系统,以检验系统能力的最高实际限度,它要求软件必须被强制在它的设计能力的极限状态下运行。

3、软件可靠性测试

这种测试经常发现的错误包括越界指针,内存泄露、栈溢出、超过两个特性之间的错误交互等,也称

长序列测试(long sequence testing)、持久测试(duration testing)、耐力测试(endurance testing)。测试持续时间较长,目标是发现程序测试遗漏的错误。

可靠性差的软件,如执行时总是频繁地、重复地失败,软件不能稳定地工作。

软件可靠性测试的目的是给出可靠性的定量估计值。

软件性能评测“指标观”

一般来说衡量软件性能测试的标准可以采用以下曾用的方法:

1、软件可靠度(R)指标

一种定量描述软件可靠性的方法,是指在规定的条件下和规定的时间内,软件在其运行剖面的某时刻正确地实现规定功能的概率。

2、软件平均故障前工作时间(MTTF)

一种定量描述软件可靠性的方法,是指一定配置状态下的软件产品在其规定的运行剖面中故障工作时间的期望值,以及软件故障强度。

3、软件平均无故障工作时间(MTBF)

计算机系统或子系统实现其功能的能力。一般以对计算机系统或子系统执行其功能的能力的度量。例如,响应时间、吞吐能力、事务处理数、占用率。

软件性能测试“实例观”

为了让读者对性能测试有更深刻的认识,下面以手机中运行的软件为例,说明在实际的软件开发过程中性能测试方法的运用:

手机性能测试的方法可分为手工测试和自动测试。

手工测试主要是通过测试人员手动操作,并借助某些监测仪器和工具来验证手机性能。但由于手机功能众多,很多性能测试需要重复性地进行,工作量很大,这需要耗费大量测试时间同时也容易造成测试的遗漏,不能保证性能测试的准确性和高效率。

借助PC平台,目前已经有很多功能强大且通用的自动测试工具,如比较典型的有Winrunner、Robot、Loadrunner等等,但这些工具必须进行二次开发,才能将自动测试工具兼容到手机这种嵌入式系统中来。

手机的自动化性能测试一般分为以下几个步骤进

行:

1、系统分析

将系统的性能指标转化为性能测试的具体目标。通常在这一步骤里,要分析被测系统结构,结合性能指标,制定具体的性能测试实施方案。要求测试人员对被测系统结构和实施业务全面掌握。

2、建立虚拟用户脚本

将业务流程转化为测试脚本,通常指的是虚拟用户脚本或虚拟用户。虚拟用户通过驱动一个真正的客户程序来模拟真实用户,要将各类被测业务流程从头至尾进行确认和记录,分析每步操作的细节和时间,将其精确地转化为脚本。类似制造一个能够模仿人的行为和动作的机器人过程。

3、根据用户性能指标创建测试场景

根据真实业务场景,对生成的测试脚本进行复制和控制,对脚本的执行制定规则和约束关系,转化为满足性能测试指标的测试用例集。这个步骤需要结合用户性能指标进行细致地分析。

4、运行测试场景,同步监测应用性能

在性能测试运行中,实时监测能使测试人员在测试过程中的任何时刻都可以了解应用程序的性能优劣。系统的每一部件都需要监测:协议栈、MMI、应用程序、内存占用情况、驱动程序运行状态等。实时监测可以在测试执行中及早发现性能瓶颈。

5、性能测试的结果分析和性能评价

结合测试结果数据,分析出系统性能行为表现的规律,并准确定位系统的性能瓶颈所在。可以利用数学手段对大批量数据进行计算和统计,使结果更加具有客观性。在性能测试中,需要注意的是,能够执行的性能测试方案并不一定是成功的,成败的关键在于其是否精确地对真实世界进行了模拟。

在整个性能测试过程中,自动化测试工具的选择只能影响性能测试执行的复杂程度,但人的分析和思考却会直接导致性能测试的成败。

总之,软件性能测试的方法很多,不同方法的评测指标也是不同的。针对用户的需求、开发模式以及开发过程的不同,灵活采用全面、实用、高效率的性能测试,才是永恒不变的测试之道。IT