



工业和信息化部“十二五”规划教材
“十二五”国家重点图书出版规划项目

软件测试与质量保证

Software Testing and Quality Assurance

● 朱东杰 孙玉山 主编



哈尔滨工业大学出版社
HARBIN INSTITUTE OF TECHNOLOGY PRESS



工业和信息化部“十二五”规划教材
“十二五”国家重点图书出版规划项目

软件测试与质量保证

Software Testing and Quality Assurance

● 朱东杰 孙玉山 主编



哈尔滨工业大学出版社
HARBIN INSTITUTE OF TECHNOLOGY PRESS

内 容 简 介

本书分为两部分:第一部分先介绍软件测试的概念,然后按照软件测试过程,分别对单元测试、集成阶段、系统测试、回归测试、验收测试进行介绍,最后介绍自动化测试工具的使用以及与现阶段热门的“互联网+”相关的 App 测试方法和工具的使用方法;第二部分介绍软件质量保证的相关概念,重点介绍关键型软件的质量标准,软件质量保证团队与计划,当代软件质量管理与标准,统计软件质量保证等方面的相关概念与技术。

本书可供高等院校计算机与软件工程专业高年级本科生使用,也可供相关专业科技人员参考。

图书在版编目(CIP)数据

软件测试与质量保证/朱东杰,孙玉山主编. —哈尔滨:哈尔滨工业大学出版社,2017.6
ISBN 978-7-5603-5924-3

I. ①软… II. ①朱… ②孙… III. ①软件-测试
②软件质量-质量管理 IV. ①P311.5

中国版本图书馆 CIP 数据核字(2016)第 062162 号

策划编辑 王桂芝 张 荣
责任编辑 刘 瑶 王桂芝
出版发行 哈尔滨工业大学出版社
社 址 哈尔滨市南岗区复华四道街 10 号 邮编 150006
传 真 0451-86414749
网 址 <http://hitpress.hit.edu.cn>
印 刷 哈尔滨市工大节能印刷厂
开 本 787mm×1092mm 1/16 印张 14.25 字数 350 千字
版 次 2017 年 6 月第 1 版 2017 年 6 月第 1 次印刷
书 号 ISBN 978-7-5603-5924-3
定 价 32.00 元

(如因印装质量问题影响阅读,我社负责调换)

前 言

随着软件规模的不断扩大,软件设计的复杂程度不断提高,软件开发中出现错误或缺陷的机会越来越多,同时,市场对软件质量重要性的认识逐渐增强,由此软件测试在软件项目实施过程中的重要性日益突出。但现实情况是,与软件编程比较,软件测试的地位和作用还没有真正得到重视。

本书按照软件测试过程,针对软件测试过程中的各个阶段分析总结出相关技术,并结合实际示例,使软件测试更加容易理解,并且能够利用所学技术,采用合适的自动化测试工具对系统进行测试。通过本书的学习,读者可以了解为什么需要软件测试,软件测试都测些什么,由谁来进行软件测试,软件测试可以使用哪些自动化工具,采用何种技术和手段来保证软件质量。

由于软件测试与质量保证理论较为抽象,而在校学生往往无实践经验,因此本书将软件测试理论与实例相结合,通过精心设计的实验,引导学生掌握软件测试自动化的相关内容,这正是本书的特色所在。

本书的主要特点如下:

(1)对软件测试与质量保证的理论和技術做深入、详尽的阐述。

(2)本书贯彻“教你怎样做”的原则,包含了软件测试自动化工具使用的例子。

(3)为了培养学生的动手能力,本书在第6、8、9章以业界实际使用的软件测试自动化工具为例,讲述如何使用实际的工具对软件系统进行功能测试、性能测试和回归测试等。

(4)移动互联网大潮正以前所未有之势席卷传统行业,整个社会也在不知不觉中迈入以各种平板、智能手机等移动终端为信息传播主导媒介的移动互联网时代,本书第9章根据移动App的特点分析对移动App进行测试的方法和技术,并对业界使用的App自动化测试工具进行介绍。

本书由朱东杰、孙玉山任主编;由高晨光、董爽爽、王大顺、房笑棣、乔学明任副主编;由王宇颖、李东主审。哈尔滨工业大学(威海)计算机学院的朱东杰对全书进行整体策划和统稿。具体分工如下:哈尔滨工业大学(威海)计算机学院孙玉山编写第10~14章,并对全书进行了修改与校对;牡丹江医学院高晨光编写第1、2章;哈尔滨工业大学(威海)王大顺编写第3章;中国建设银行山东省威海市支行的房笑棣编写第4章;国网山东省电力公司威海供电公司乔学明编写第5章,并对实验部分提出了宝贵的意见;哈尔滨工业大学(威海)董爽爽编写第8、9章并对本书的源代码进行调试;哈尔滨工业大学研究生彭暄、孙昊和杜海

文参与编写第8、9章的实验部分。哈尔滨工业大学王宇颖教授和李东教授审阅了书稿,提出了很多宝贵的修改意见。权光日教授和张廷斌教授也对本书的出版提出了宝贵的建议。

由于编者水平有限,书中难免有不妥之处,恳请读者批评指正,作者将不胜感激。作者联系方式:zhudongjie@hit.edu.cn。

本书实验相关要求及代码可以从<http://software.hitwh.edu.cn/moodle> 下载或者直接向作者索取。

编 者

2017年1月

目 录

第一部分 软件测试

第 1 章 软件测试技术相关概念	3
1.1 软件测试概述	3
1.2 软件测试现状	6
1.3 软件测试前景	7
1.4 软件测试相关术语	8
习题	10
第 2 章 软件测试过程模型的分类及其流程	12
2.1 软件测试过程模型	12
2.2 软件测试的分类	17
2.3 软件测试的流程	20
习题	25
第 3 章 单元测试技术	27
3.1 引言	27
3.2 单元测试的定义及范畴	28
3.3 单元测试的主要内容	30
3.4 单元测试的主要技术	35
3.5 单元测试工具	43
3.6 单元测试工具 JUnit 实验	52
习题	55
第 4 章 集成测试技术	57
4.1 集成测试概述	57
4.2 集成测试的策略	58
4.3 集成测试的实施	63
4.4 集成测试工具及框架	66
习题	84
第 5 章 系统测试技术	86
5.1 系统测试概述	86
5.2 系统测试的基本方法	89

5.3	系统测试的相关工具介绍	101
	习题	104
第 6 章	回归测试技术	106
6.1	回归测试概述	106
6.2	回归测试的基本方法	110
6.3	回归测试的相关工具介绍	113
6.4	自动化测试工具 QTP 使用实验	116
	习题	123
第 7 章	验收测试技术	124
7.1	验收测试的定义	124
7.2	验收测试的主要内容	124
7.3	α 测试与 β 测试	126
7.4	验收测试流程	129
	习题	130
第 8 章	自动化测试工具 LoadRunner	131
8.1	LoadRunner 简介	131
8.2	LoadRunner 的功能	133
8.3	LoadRunner 实验	139
	习题	140
第 9 章	App 测试工具 MonkeyRunner	141
9.1	移动 App 测试点	141
9.2	移动 App 测试工具 MonkeyRunner	151
	习题	156

第二部分 软件质量保证

第 10 章	软件质量保证相关概念	159
10.1	软件质量保证中的基本概念	159
10.2	软件质量保证中的重要概念	160
10.3	关键型软件与其质量标准	165
第 11 章	软件质量保证的内容	168
11.1	软件产品质量与过程质量	168
11.2	软件质量保证的组成	171
11.3	软件质量控制	175
11.4	软件配置管理	179

第 12 章 软件质量保证团队与计划	184
12.1 软件质量保证团队的组成	184
12.2 软件质量保证团队的活动	185
12.3 软件质量保证实例	186
12.4 软件质量保证计划	187
第 13 章 当代软件质量管理与标准	190
13.1 ISO 9000 质量保证体系	190
13.2 软件成熟度模型	191
13.3 戴明质量管理的 14 点原则	192
13.4 戴明质量循环模型	195
第 14 章 统计软件质量保证	199
14.1 Pareto 原理	199
14.2 统计软件质量保证	200
14.3 Fast 方法	205
附录 1 软件测试计划模板	206
附录 2 软件测试用例模板	212
附录 3 软件测试报告模板	214
参考文献	216

第一部分 软件测试

第1章 软件测试技术相关概念

1.1 软件测试概述

1.1.1 软件测试的定义

软件测试(Software Testing)是描述一种用来促进鉴定软件的正确性、完整性、安全性和质量的过程。换句话说,软件测试是一种实际输出与预期输出间的审核或者比较过程。软件测试的经典定义是:在规定的条件下对程序进行操作,以发现程序错误,衡量软件质量,并对其是否能满足设计要求进行评估的过程。软件测试是软件质量保证(Software Quality Assurance, SQA)的重要子域。

Glenford J. Myers 曾对软件测试的目的提出以下观点:

- (1) 测试是为了发现程序中的错误而执行程序的过程。
- (2) 好的测试方案是极可能发现迄今为止尚未发现的错误的测试方案。
- (3) 成功的测试是发现了至今为止尚未发现的错误的测试。
- (4) 测试并不仅仅是为了找出错误。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。
- (5) 这种分析也能帮助测试人员设计出有针对性的测试方法,改善测试的效率和有效性。
- (6) 没有发现错误的测试也是有价值的,完整的测试是评定软件质量的一种方法。
- (7) 根据测试目的的不同,可分为回归测试、压力测试和性能测试等,分别是为了检验修改或优化过程是否引发新的问题、软件所能达到处理能力和是否达到预期的处理能力等。

1.1.2 软件测试的原则

从不同的角度出发,软件测试会派生出两种不同的测试原则。从用户的角度出发,就是希望通过软件测试能充分暴露软件中存在的问题和缺陷;从开发者的角度出发,就是希望通过测试能表明软件产品不存在错误,已经正确地实现了用户的需求,因此提出这样一组测试原则:

- (1) 所有软件测试都应追溯到用户需求。
- (2) 应当把“尽早和不断地测试”作为开发者的座右铭。
- (3) 完全测试是不可能的,测试需要终止。
- (4) 制订严格的测试计划,并把测试时间安排得尽量宽松,不要希望在极短的时间内完成一个高水平的测试。
- (5) 应按测试对象所处环境来设计测试用例。

(6) 测试错误结果,应按测试对象所处环境进行举一反三,看其具有普遍性还是唯一性。

(7) 回归测试的关联性一定要引起充分的注意,修改一个错误而引起更多错误出现的现象并不少见。

(8) 妥善保存一切测试过程文档,意义是不言而喻的,测试的重现性往往要靠测试文档。

(9) 测试无法显示软件潜在缺陷。

(10) 充分注意测试中的群集现象,根据 80/20 原则,80% 的错误与系统 20% 的程序模块有关。

(11) 程序员应避免检查自己的程序。

(12) 避免测试的随机性,要有组织、有计划、有步骤地测试。

1.1.3 软件测试的目标

软件测试的目标如下:

- (1) 发现一些可以通过测试避免的开发风险。
- (2) 实施测试来降低所发现的风险。
- (3) 确定测试何时可以结束。
- (4) 在开发项目过程中将测试看作是一个标准项目。

1.1.4 软件测试的过程

软件测试的过程如下:

(1) 对要执行测试的产品/项目进行分析,确定测试策略,制订测试计划。该计划被审核批准后转向下一步。测试工作启动前一定要确定正确的测试策略和指导方针,这些是后期工作的基础。只有将本次的测试目标和要求分析清楚,才能决定测试资源的投入。

(2) 设计测试用例。要根据测试需求和测试策略来设计测试用例,如果进度压力不大,应进行详细设计;如果进度、成本压力较大,则应保证测试用例覆盖到关键性的测试需求。该用例被批准后转向下一步。

(3) 如果满足“启动准则”(Entry Criteria),那么执行测试。执行测试主要是搭建测试环境,执行测试用例。执行测试时要进行进度控制、项目协调等工作。

(4) 提交缺陷。进行缺陷审核和验证等工作。

(5) 消除软件缺陷。在通常情况下,开发经理需要审核缺陷,并进行缺陷分配。程序员修改自己负责的缺陷。程序员修改完成后,进入到回归测试阶段。如果满足“完成准则”(Exit Criteria),则正常结束测试。

(6) 撰写测试报告。对测试进行分析,总结本次的经验教训,在下次工作中进行改进。

软件测试过程管理主要包括软件测试是什么样的过程,如何评价一个软件测试过程,如何进行配置管理以及测试风险分析与测试成本管理。

1.1.5 软件测试的内容

软件测试的内容包括验证(Verification)和确认(Validation)。

1. 验证

验证是保证软件正确实现一些特定功能的一系列活动,即保证软件以正确的方式做了这个事件(Do it right)。

(1)确定软件生存周期中的一个给定阶段的产品是否达到前阶段所确立的需求的过程。

(2)程序正确性的形式证明,即采用形式理论证明程序符合设计规约规定的过程。

(3)评审、审查、测试、检查和审计等各类活动,或对某些项处理、服务或文件等是否和规定的需求相一致,进行判断和提出报告。

2. 确认

确认是一系列的活动和过程,目的是想证实在一个给定的外部环境中软件的逻辑正确性,即保证软件做了你所期望的事情(Do the right thing)。

(1)静态确认。不在计算机上实际执行程序,通过人工或程序分析来证明软件的正确性。

(2)动态确认。通过执行程序做分析,测试程序的动态行为,以证实软件是否存在问题。

软件测试的对象不仅仅是程序测试,还应包括整个软件开发期间各个阶段所产生的文档,如需求规格说明、概要设计文档及详细设计文档。当然,软件测试的主要对象还是源程序。

1.1.6 软件测试的分类

软件测试从不同角度考虑,其分类也不同。

按是否关心软件内部结构和具体实现的角度划分为:

(1)白盒测试。

(2)黑盒测试。

(3)灰盒测试。

按是否执行程序的角度划分为:

(1)静态测试。

(2)动态测试。

按软件开发的过程阶段划分为:

(1)单元测试。

(2)集成测试。

(3)系统测试。

(4)验收测试。

(5)回归测试。

(6) α 测试。

(7) β 测试。

具体来说,单元测试是集中对用源代码实现的每个程序单元进行测试,检查各个程序模块是否正确地实现了规定的功能;集成测试把已测试过的模块组装起来,主要对与设计相关

的软件体系结构进行测试;系统测试是把已经经过确认的软件纳入实际运行环境中,与其他系统组合在一起进行测试;验收测试是系统开发生命周期方法论的一个阶段,相关的用户和独立测试人员根据测试计划和结果对系统进行测试和接收;回归测试是指修改了旧代码后,重新进行测试以确认修改没有引入新的错误或导致其他代码产生错误; α 测试是由一个用户在开发环境下进行的测试,也可以是公司内部用户在模拟实际操作环境下进行的受控测试; β 测试是由软件的多个用户在一个或多个实际使用环境下进行的测试,开发者通常不在现场。

本书将从软件开发过程按阶段划分为读者详细介绍软件测试的方法、过程及相关工具。

1.2 软件测试现状

由于软件开发中会出现很多错误或缺陷,而且市场对软件质量重要性的认识逐渐增强,因此软件测试在软件项目实施过程中的重要性日益突出。但现实情况是,与软件编程比较,软件测试的地位和作用还没有真正受到重视,很多人(甚至是软件项目组的技术人员)还存在对软件测试的认识误区,这进一步影响了软件测试活动的开展和软件测试质量的真正提高。

1. 误区之一:软件开发完成后进行测试

一般认为,软件开发要经过以下几个阶段:需求分析、概要设计、详细设计、软件编码、软件测试及软件发布。据此,认为软件测试只是软件开发的一个过程。这是不了解软件测试周期的错误认识。软件测试是一系列过程活动,包括软件测试需求分析、测试计划设计、测试用例设计及执行测试。因此,软件测试贯穿于软件项目的整个生命过程。在软件项目的每个阶段都要进行不同目的和内容的测试活动,以保证各个阶段的正确性。软件测试的对象不仅仅是软件代码,还包括软件需求文档和设计文档。软件开发与软件测试应该是交互进行的,例如,单元编码需要单元测试,模块组合阶段需要集成测试。如果等到软件编码结束后才进行测试,那么测试的时间将会很短,测试的覆盖面将很不全面,测试的效果也会大打折扣。更严重的是,如果此时发现软件需求阶段或概要设计阶段出错,要修复该类错误则会耗费大量的时间和人力。

2. 误区之二:软件发布后如果发现质量问题,则是软件测试人员的错

这种认识很打击软件测试人员的积极性。软件中的错误可能来自软件项目中的各个过程,软件测试只能确认软件存在错误,不能保证软件没有错误,因为从根本上讲,软件测试不可能发现全部的错误。从软件开发的角度看,软件的高质量不是软件测试人员测出来的,而是靠软件生命周期的各个过程设计出来的。软件出现错误,不能简单地归结为某一个人的责任,有些错误的产生可能不是技术原因,而是来自于混乱的项目管理。应该分析软件项目的各个过程,从过程改进方面寻找产生错误的原因和改进的措施。

3. 误区之三:软件测试要求不高,随便找个人做就行

很多人都认为软件测试就是安装和运行程序,点点鼠标、按按键盘的工作,这是由于不了解软件测试的具体技术和方法而造成的。随着软件工程学的发展和软件项目管理经验的提高,软件测试已经形成了一个独立的技术学科,演变成了一个具有巨大市场需求的行业。

软件测试技术不断更新和完善,新工具、新流程和新测试设计方法都在不断更新,需要掌握和学习很多测试知识。所以,具有编程经验的程序员不一定是一名优秀的软件测试工程师。软件测试包括测试技术和管理两个方面,完全掌握这两个方面的内容,需要具有很多测试实践经验和不断的学习。

4. 误区之四:软件测试是测试人员的事情,与程序员无关

开发和测试是相辅相成的过程,需要软件测试人员、程序员和系统分析师等保持密切的联系,需要更多的交流和协调,以便提高测试效率。另外,对于单元测试主要由程序员完成,必要时测试人员可以帮助设计测试用例。对于测试中发现的错误,很多需要程序员通过修改编码才能修复。程序员可以通过有目的地分析软件错误的类型、数量,找出产生错误的位置和原因,以便在今后的编程中避免发生同样的错误,积累编程经验,提高编程能力。

5. 误区之五:项目进度吃紧时少做些测试,时间富裕时多做些测试

这是不重视软件测试的表现,也是软件项目过程管理混乱的表现,必然会降低软件测试的质量。一个软件项目的顺利实现需要有合理的项目进度计划,其中包括合理的测试计划,对项目实施过程中的任何问题,都要有风险分析和相应的对策,不要因为开发进度的改变而影响测试时间、人力和资源。因为缩短测试时间所带来的测试不完整,对项目质量的下降具有潜在风险,往往造成更大的浪费。克服这种现象的最好办法是加强软件过程的计划和控制,包括软件测试计划、测试设计、测试执行、测试度量和测试控制。

6. 误区之六:软件测试是没有前途的工作,只有程序员才是软件高手

由于我国软件整体开发能力比较低,软件开发过程很不规范,很多软件项目的开发都还停留在“作坊”式和“垒鸡窝”式阶段。项目的成功往往靠个别全能程序员决定,他们负责总体设计和程序详细设计,认为软件开发就是编写代码,给人的印象往往是程序员是真正的牛人,具有很高的地位和待遇。因此,在这种环境下,软件测试很不受重视,软件测试人员的地位和待遇自然就很低了,甚至软件测试变得可有可无。随着市场对软件质量要求的不断提高,软件测试将变得越来越重要,相应的软件测试人员的地位和待遇也会逐渐提高。在软件开发过程比较规范的大公司,软件测试人员的数量和待遇与程序员没有多大差别,优秀测试人员的待遇甚至比程序员还要高。软件测试是一个具有很大发展前景的行业,市场需要更多具有丰富测试技术和管理经验的测试人员,他们同样是软件专家。

1.3 软件测试前景

随着软件产业的发展,软件产品的质量控制与质量管理正逐渐成为软件企业生存与发展的核心。几乎每个大中型IT企业的软件产品在发布前都需要大量的质量控制、测试和文档工作,而这些工作必须依靠拥有娴熟技术的专业软件人才来完成。软件测试工程师就在企业中扮演着这样一个重头角色。据招聘网站51job数据显示,软件测试工程师将成为最紧缺的人才,该类职位的需求主要集中在发达城市,其中北京、上海的需求量分别占33%和29%。同一时间中华英才网发布了最新一期的IT职场人气排行榜,IT人才仍是企业需求量最大的人群,其中软件测试工程师、高级程序员、产品项目经理等高级职位进入“三甲”,成为IT就业市场最新风向标。作为软件开发流程中的重要一环,软件测试岗位渐渐“浮出水

面”，并凭借其庞大的人才需求和广阔的职场发展前景日渐成为 IT 职场就业的大热门。测试工程师的工作是利用测试工具按照测试方案和流程对产品进行功能和性能测试，甚至根据需要编写不同的测试用例，设计和维护测试系统，对测试方案可能出现的问题进行分析和评估。对软件测试工程师而言，必须具有高度的工作责任心和自信心。任何严格的测试必须是一种实事求是的测试，因为它关系到一个产品的质量，而测试工程师则是产品面世前的把关人，所以，没有专业的技术水准是无法胜任这项工作的。同时，由于测试工作一般由多个测试工程师共同完成，并且测试部门一般要与其他部门的人员进行较多的沟通，因此要求测试工程师不但要有较强的技术能力，而且要有较强的沟通能力。

1.4 软件测试相关术语

1. 软件测试

软件测试是根据软件开发各阶段的规格说明和程序的内部结构而精心设计一批测试用例，并利用这些测试用例运行软件，以发现软件错误的过程。

2. 测试用例

测试用例是指对一项特定的软件产品进行测试任务的描述，体现测试方案、方法、技术和策略的文档，内容包括测试目标、测试环境、输入数据、测试步骤、预期结果及测试脚本等。

3. 测试计划

测试计划是指对软件测试的对象、目标、要求、活动、资源及日程进行整体规划，以保证软件系统的测试能够顺利进行的计划性文档。

4. 测试对象

测试对象是指在特定环境下运行的软件系统和相关的文档。作为测试对象的软件系统可以是整个业务系统，也可以是业务系统的一个子系统或一个完整的部件。

5. 测试流程

测试流程是指为了保证测试质量而精心设计的一组科学、合理、可行的有序活动。比较典型的测试流程一般包括制订测试计划、编写测试用例、执行测试、跟踪测试缺陷、编写《测试报告》等活动。

6. 测试评估

测试评估是指对测试过程中的各种测试现象和结果进行记录、分析和评价的活动。

7. 《测试报告》

《测试报告》是一份有关本次测试的总结性文档，主要记录有关本次测试的目的、测试结果、评估结果及测试结论等信息。

8. 测试环境

测试环境是指对软件系统进行各类测试时基于的软、硬件设备和配置，一般包括硬件环境、网络环境、操作系统环境、应用服务器平台环境、数据库环境及各种支撑环境等。

9. 白盒测试

白盒测试又称结构测试，是指基于一个应用代码的内部逻辑知识，即基于覆盖全部代

码、分支、路径、条件的测试。

10. 黑盒测试

黑盒测试又称功能测试,是指基于需求和功能性的测试,而不是基于内部设计和代码的任何知识的测试。

11. 单元测试

单元测试又称模块测试,是指针对程序模块(软件设计的最小单位)来进行正确性检验的测试工作。

12. 集成测试

集成测试又称组装测试,是指对程序模块采用一次性或增值方法组装起来,对模块间接口进行正确性检验的测试工作。

13. 系统测试

系统测试是指将通过集成测试的软件系统或子系统,作为基于计算机系统的一个元素,与计算机硬件、外设、某些支持软件、数据和人员等其他系统元素组合在一起所进行的测试工作,目的在于通过与系统的需求定义做比较,发现软件与系统定义不符合或与之矛盾的地方。

14. 确认测试

确认测试又称有效性测试,是指在模拟(或正式)的生产环境下,运用黑盒测试方法,验证所测软件是否满足用户需求说明书中所列出的需求。

15. 功能测试

功能测试是指为了保证软件系统功能实现的正确性、完整性及其他特性而进行的测试。

16. 性能测试

性能测试是指为了评估软件系统的性能状况和预测软件系统的性能趋势而进行的测试和分析。

17. 并发

狭义的并发指所有用户在同一时刻做的同一件事情或者操作。广义的并发指尽管多个用户对系统发出了请求或进行了操作,但这些操作可以是相同的,也可以是不同的。

18. 并发用户数量

并发用户数量是指在同一时刻与服务器进行交互的在线用户数量。其公式为

$$\text{并发用户数量} = \text{在线用户数量} \times (5\% \sim 20\%)$$

19. 响应时间

响应时间是指从客户端发送一个请求开始计时,到客户端接到从服务器端返回的响应结果结束计时所经历的时间。响应时间由网络传输时间、服务器处理时间和浏览器显示时间组成。

20. 吞吐量及吞吐率

吞吐量是一次性能测试过程中网络上传输的数据量的总和。吞吐率是吞吐量与传输时