

Web 测试综述^{*}

许 蕾 徐宝文 陈振强

(东南大学计算机科学与工程系 南京210096) (武汉大学软件工程国家重点实验室 武汉430072)

A Survey of Web Testing

XU Lei XU Bao-Wen CHEN Zhen-Qiang

(Department of Computer Science & Engineering, Southeast University, Nanjing 210096)

(State Key Laboratory of Software Engineering, Wuhan University, Wuhan 430072)

Abstract With the extensive application of Web technology, it becomes more and more critical of the request to the quality and reliability of Web applications. Then it is crucial to test Web applications automatically, entirely and thoroughly. So we make a study of some Web testing methods and technologies. First, we discuss the necessity of Web testing, then analyze where the faults may take place based on the architecture of the Web, next discuss various methods of Web testing in details. Then, based on the ideology of Object-Oriented, we build a model for Web testing, and discuss the method of doing some pertinence testing on pages when we utilize the information of statistic. At last, we introduce some tools for white-box testing.

Keywords Web testing, Object-oriented, Web testing model, Statistic information

1 引言

Web 以其广泛性、交互性和易用性等特点迅速风靡世界,网页数量正以指数级飞速增长^[1]。能够吸引尽可能多的用户并对其进行长时间关注,是网站追求的主要目标,也是衡量一个网站是否成功的主要指标,这就对网页功能的正确性、有效性和完善性提出了较高的要求,从而 Web 测试就成为 Web 应用开发过程中的一个重要环节。通过 Web 测试尽可能多地发现浏览器端和服务器端程序中的错误并及时加以修正,以保证 Web 应用的质量。由于 Web 具有分布、异构、并发和平台无关的特性,因而 Web 测试要比普通程序的测试复杂得多^[2,3]。

目前,国内外已经开始对 Web 测试进行研究并取得了一些初步研究成果,本文的目的是对现有的一些典型 Web 测试方法与技术进行深入剖析。为此,本文首先讨论了 Web 测试的必要性及其特殊性;接着针对 Web 应用特殊的体系结构,分析故障可能产生的位置;然后详细讨论了各种具体的测试技术,给出了测试的解决办法;最后讨论了 Web 测试模型,并重点分析了运用面向对象的思想对 Web 测试建模的方法以及利用统计信息对页面进行有针对性的测试以减轻测试工作量的方法。

2 Web 应用特点及其故障源分析

与传统软件相比,Web 应用有很多独特的地方,其中对测试有影响的方面主要有以下几点:其一,Web 应用的用户数量巨大,并要求能提供对 Web 资源的跨平台全局访问,需要有并发处理事务的能力,因而需要进行多用户的性能测试。

其二,使用环境不尽相同。硬件设备、网络连接、操作系统、中间件、服务端支持、浏览器等都有所不同,形成了异构、自治的工作环境。这就需要考虑兼容性测试。其三,着重于信息的发布。大多数传统软件强调的是运算的功能,而在 Web 上,信息的搜索和获取占了很大一部分,需要通过测试来保证信息的正确显示。总之,Web 具有分布、异构、并发和平台无关的特性,传统的测试方法已经不能完全胜任对 Web 的测试,需要一套独特的测试体系。

我们可以把 Web 看成是一个使用方便、接受全局访问、具有图形化界面的大的数据库的前端,其结构示意图如图1所示^[4]。由于 Web 应用具有多层体系结构,客户、数据通信、硬件以及服务器之间的依赖关系又非常复杂,使得在每层内以及各层间都有可能发生故障。在客户端,由于浏览器的型号、版本有很大的不同,以及对应的显示技术各不相同,使得有些信息往往不能正常地显示,从而产生兼容性问题以及显示故障;在服务器端,可能存在超链接不可达或者根本不存在的问题,影响用户的使用和评价;服务器、数据库的负载能力有限,在用户访问达到高峰时,响应时间太长甚至不接受用户的访问;并发用户的行为会影响到与站点交互的情况,用户之间也可能相互干扰。

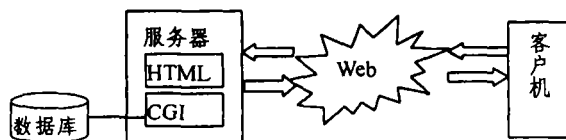


图1 Web 应用的体系结构图

^{*})本研究得到国家自然科学基金(60073012)、江苏省自然科学基金(BK2001004)、江苏省科技攻关项目(BE2001025)、教育部高校骨干教师资助计划、江苏省三三三人才基金、高等学校重点实验室访问学者基金、南京大学软件新技术国家重点实验室基金、武汉大学软件工程国家重点实验室开放基金、江苏省重点学科(K0015)、东南大学科学基金(XJ000928)资助。徐宝文 教授,博士生导师,主要从事程序设计语言、软件工程、并行与网络软件等方向的教学与科研工作。许 蕾 博士研究生,主要从事 Web 软件分析、测试与应用方面的研究工作。陈振强 博士研究生,主要从事软件分析、理解与测试方面的研究工作。

总之,由于诸多问题的存在,必须做功能测试、负载和强度测试、浏览器兼容性测试以及安全性测试等^[5~7],各种基本的测试技术将在下部分详述。

3 Web 测试基本技术

Web 应用的结构特点决定了各种 Web 故障的分布情况,要发现、分析、排除故障,通常需要进行多方面的测试,以保证 Web 应用具有完善良好的功能、性能、兼容性和安全性。

3.1 功能测试

功能测试就是结合规格说明的要求,保证功能上正确无误,其内容包括 HTML 语法检查、链接检查、表格测试、端到端交互验证、Java 组件测试等。根据测试要求的难易程度的不同,功能测试可分为:简单功能测试、任务特征测试、边界测试、强制错误情况测试、探测性测试等,以确保不同层次上的网站或网络应用程序运行的质量。简单功能测试主要做一些链接可达性的检查工作。任务特征测试是根据任务的交互性、不确定性等不同特征,进行有针对性的测试。边界测试是在输入数据域的边界抽取数据进行测试^[8,9]。强制错误情况测试是根据设计时的规格说明,人为输入明显错误的数据,然后观测系统的运行情况,主要测试系统的容错性。探测性测试就是边设计边执行测试,试探性地前进几步并及时调整。

与普通程序相比,Web 的分布性和复杂性造成了 Web 程序执行时的不确定性:由于运行环境是时刻变化的,对同样的输入,输出往往会有很大的不同。顾成在他的博士论文^[6]中提出了这样的方法:一是不确定性测试,即在给定输入后就让程序自由运行,然后观察输出结果,需要执行若干次测试,然

后进行综合分析、比较。二是确定性测试,即在给定输入后,人为干预程序的执行,使其向预想的状态转移或者执行预定的动作,以检测可能的错误。第一种方法能够自动实现,但结果不一定很全面;第二种方法人为干预太多,效率不是很高。这样,需要根据具体的测试要求来决定采用哪种测试方法或者是取两种方法的结合。

3.2 负载测试

负载测试主要是确定在用户可接受的响应时间内,系统能够承担的并发用户的数量。由于成千上万的用户可能在同一时刻访问一个网站,为了保证系统的正常运行,必须测试在最重的工作负载下网站的运行情况。

可以通过脚本来生成成千上万的“虚拟用户”,同时访问站点并与站点交互。这些虚拟用户执行各种典型的任务,如浏览网页、购买商品、搜索数据库等。在虚拟用户执行这些任务的同时,记录下服务器的响应时间。当测试执行完成以后,分析通过负载测试得到的数据,如在不同交互情况下的网页传送所需时间、网页传输出错信息等,经过一定的分析、运算,得出网站能同时支持的用户数目、交互数目等,并尽可能找出多用户访问的瓶颈,最后以报告和图表的形式来显示测试情况下网站的执行情况以及潜在问题存在的方

多用户访问的瓶颈可能发生在 Web 服务器上,或者是在应用的服务器上,也可能在数据库服务器上。就目前技术来看,要查明瓶颈发生的确切地点还是非常困难的。另外,要确定同时访问网站的用户数目并不容易。可以参考系统给出的参数结合实际情况进行估计,也可以选择有代表性的分布算法(如指数分布、常量分布、泊松分布等算法)来估计。

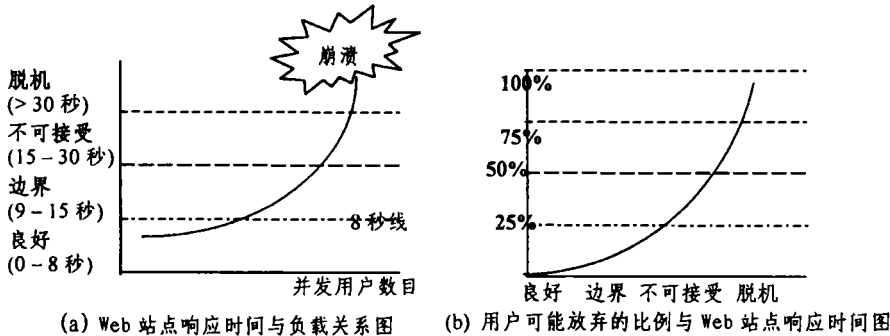


图2 多用户性能测度中部分数据关系图

图2显示了经统计而得的 Web 站点负载、Web 站点响应时间与用户可能放弃的比例之间的关系。从图2中可以看出,并发用户数目越多,Web 站点的响应时间越长,从而用户放弃执行该动作的可能性越大。因而必须从网站的实际负载能力出发,确定合适的并发用户的数目,以先保证服务质量;然后在系统性能稳定的范围内,再考虑为更多的用户提供优质服务。

3.3 浏览器兼容性测试

浏览器兼容性测试主要是在不同的浏览器环境下对 Web 应用的显示作测试,使得 Web 应用的功能能够正确运行于广泛的操作平台之上,保证某些代码(如 Java Applets、ActiveX Controls、JavaScript、CSS、HTML)的功能在不同配置上都具有正确性。

浏览器有各种版本的 Netscape、Internet Explorer、AOL、Opera 等,可应用于 Windows、Macintosh、Unix、Linux 平台,因而要在不同浏览器、操作系统和硬件设置的结合下进行网

站测试,在不同的客户硬件配置、客户操作系统、浏览器类型和版本以及浏览器插件的组合使用情况下,观察浏览器显示的所有元素,检察浏览器命令,检验内容设置,确认浏览器安全设置,检查高级设置。这就使得测试工作非常繁重,用人工实现上述测试过程是很难想象的,而使用相应的测试工具(如 SilkTest^[10])就能够大大提高工作效率。

3.4 安全性测试

对一个网站而言,安全性是至关重要的。尤其对网上交易来说,如果安全得不到保障,那么交易很难进行。通常的安全机制包括信息存取控制、用户身份校验以及对机密信息进行加密等,同时要能根据用户的访问情况判断出该用户是正常的用户还是蓄意的破坏者。因而,安全性测试所做的工作也是围绕这几方面进行的。对信息存取控制进行测试,采用的方法类似于功能测试,选择的测试用例应该能够反映不同的控制级别。对用户身份校验的方法有密码设置、数字签名等,相应的测试方法是在输入密码或数字签名后,看输出的结果正确

与否。对加密信息进行测试,主要是看解密后的信息是否正确。另外还要检查其是否具有识别恶意访问者的功能。就目前情况来看,这部分的工作还要加强。

3.5 回归测试

在测试过程中,根据检测到的错误,可能需要进行一定的程序修改。由于修改后的程序仍可能存在错误,并且修改过程中可能会带来新的错误,因此还需要进行有针对性的测试,也称回归测试。在进行回归测试时,只需要测试与修改有关的部分,这可通过分析程序中定义-使用等关系来确定。具体来说就是:如果在定义部分做了改动,那么所有使用到这一定义的部分必须重新测试;如果被依赖部分有改动,所有相应的依赖部分也必须重测。在我们已有的工作中,我们曾对程序中各种依赖关系进行过比较深入的研究^[11~14],在此不再赘述。当然,这一过程的执行与否,由具体测试的实际情况而定,并且在修改过程中,应尽量避免引发新的错误。

在上述各种具体测试的过程当中,要完成这些任务,仅靠手工实现是难以想象的,必须要有工具的帮助。目前已开发了一些实用的工具,如 Weblint^[15]、W3C Validator^[16]、Net Mechanic^[17]、Doctor HTML^[18]、TCAT for Java^[19]、SilkTest^[19]、Visual Test^[20]等,但在具体的测试环境中,需求不尽相同,因而需要针对实际情况进行调整、改进,甚至重新编写。

4 典型 Web 应用测试模型

为了对 Web 应用进行深入分析,精确定位各类错误,除了上述的黑箱测试以外,还必须进行白箱测试,以深入了解其内部结构,方便查错、排错,这就需要建立 Web 测试模型。

最先对 Web 站点演化和 Web 维护进行系统研究的是 Durham 大学的 P. Warren、C. Boldyreff 和 M. Munro^[21,22],他们指出 Web 站点开发过程中产生各种问题的根源在于没有按照软件工程的规律办事,强调了 Web 维护的重要性,提出用度量技术来表征 Web 站点的演化。目前比较有影响的 Web 测试模型有 Conallen 等提出的可以用 UML 形式化描述的 Web 模型^[23]、F. Ricca 等提出的强调链接与交互等动态内容的结构模型^[24]以及 D. C. Kung、C. -H. Liu 等提出的 Web 应用测试模型^[25,26]等。

由于 Web 应用很直观地与类有对应关系(如超链接对关联关系元素,页对应类,页的脚本对应类的操作,脚本中的所有页范围内的变量对应类的属性),这些模型基本上都是基于面向对象思想而建立的,只是在具体表述上有所侧重。这里以第三种模型为例,讨论分析 Web 测试模型的相关方面。要体现 Web 应用的结构和行为,必须通过一系列的图表来表示,从而实现对 Web 应用的分析和测试。我们可以从三个不同的角度来讨论 Web 测试模型。

4.1 对象模型

将 Web 应用的实体建模成有属性和动作的对象并描述它们之间的关系。根据 Web 页所处位置的不同,形成了三种不同类型的对象:客户端页、服务器端页和组件,其中,客户端页是嵌入脚本的 HTML 文档,在客户端由 Web 浏览器触发;服务器端页是 CGI 脚本、ASP、JSP 或者 servlet,在服务器端由 Web 服务器执行;组件可以是 HTML 元素、Java Applet、ActiveX 控件、Java Bean 或者与客户端页、服务器端页以及其它组件交互的程序模块。对象的属性为程序变量或者是 HTML 文档元素,动作即为以脚本或编程语言书写的函数。

为了表示 Web 应用的实体以及它们之间的关系,提出对

象关系图^[27]的概念,定义如下:

定义1 对象关系图 $ORD=(V,L,E)$ 是一个有向图, V :表示对象的节点的集合; L :表示关系类型的标记的集合; $E\subseteq V\times V\times L$,表示对象之间关系的边的集合。

由于 Web 应用的特殊性,除了传统的面向对象体系中的继承(Inherit)、聚合(Aggregate)、联合(Associate)等关系以外,还增加了请求(Request)、响应(Response)、导航(Navigate)和重定向(Redirect)关系,以分别表示客户端页与服务器端页之间 HTTP 请求/响应、客户端页之间的导航以及服务器端页之间的重定向关系。这样,记 $L=\{I, Ag, As, Req, Rs, N, Rd\}$,相应的边记为 $E_I, E_{Ag}, E_{As}, E_{Req}, E_{Rs}, E_N, E_{Rd}$ 。若 $v_1, v_2 \in V$,则 $(v_1, v_2, Req) \in E_{Req}$ 就表示客户端页 v_1 向服务器端页 v_2 发出请求。其它关系可以类似地进行表述。

4.2 行为模型

Web 应用的动态行为包括通过超链接从一个页面跳转到另一个页面的导航以及各实体之间状态依赖的行为。前一种行为本质上不很复杂,但由于网页数量巨大,使得导航关系看起来错综复杂;后一种行为牵涉到有依赖关系的各个具体对象,需要通过深入内部分析才能得到它们之间的状态依赖关系,因而比较复杂。为了对 Web 应用的动态行为建模,可以通过两种图(页面导航图和对对象状态图)来描述。

定义2 页面导航图 $PND=(P, HL, \delta_0, q_0, F)$, 是一个有限状态自动机, P :所有页面的集合; HL :所有超链接的集合; δ_0 :转移函数, $\delta_0=P\times HL\rightarrow P$; q_0 :起始页, $q_0\in P$; F :终止页集合, $F\subseteq P$ 。

这样,在这个有限状态自动机中,每一个状态表示一个页面,状态之间的变迁表示超链接,并用该超链接的 URL 来标示,从而可以形式化地描述页面间的导航关系。要得到这个导航图并不复杂,也有很多现成的工具(如在 FrontPage 中就能生成这样的图);然后,将这个图改写成一棵树,根节点为 q_0 ,每个节点的子节点为其通过超链接能够到达的页面,同时注意避免产生回路;从而从根节点到每个叶节点之间的路径就是测试路径,这样就能够检查超链接的可达性,保证导航行为是合适的并且是满足需求的。

定义3 对象状态图^[28] $OSD=(S, \sigma, \delta, q_i, q_f)$, 也是一个有限状态自动机, S :状态空间; σ :Web 页面的集合; q_i :初始状态; q_f :终结状态; δ :变迁函数, $\delta=S\times\sigma\rightarrow S$,也可以将其表示为一个三元组 (s, d, s_b) ,其含义为:当系统处于状态 s 时,运行页面 d 会使系统的状态变为 s_b 。

每个对象实体的状态是状态空间的向量,各对象不同状态的每一次可能组合便构成了系统的状态,我们比较深入地研究过状态依赖问题^[28],在此不再赘述。类似地,在生成对象状态图的基础上,将其改造成测试树,最终得到测试路径。这样就能够保证 Web 应用的实现在状态依赖行为方面也满足其规格说明需求,即在一些应用领域,对页面的访问存在着一定的先后顺序,如网上购物,一定是要经历先登录、再经过身份验证、然后才能购买、最后付款、确认等一系列应用流程,也就是说这些状态之间的依赖关系是非常重要的,必须进行细致、完整的测试。

4.3 结构模型

为了进一步深入了解 Web 应用,必须掌握其内部的控制流和数据流信息,通过对这些信息的分析和测试,确保 Web 应用的可靠性。其做法与传统软件测试的白箱方法基本相同,即从变量出发,找出其定义-使用链,进而得到关于这个变量

的测试路径,进行基于覆盖率的测试。

如果在运算语句或输出语句中使用变量,则称为该变量的 C-use (computation use),主要反映的是数据流信息;如果在谓词语句中使用变量,则称为该变量的 P-use (predicate use),主要反映的是控制流信息;从包含变量定义的语句到包含对该变量进行运算使用的语句的路径即为 C-use 的定义-使用链;从包含变量定义的语句到包含对该变量进行断言使用的语句的两个后继语句的路径为 P-use 的定义-使用链。变量的定义-使用链便构成了测试路径。

这样,在变量的定义-使用链的基础上,可以由低到高地分五个等级来描述不同范围内的数据流测试,具体介绍如下:

最低级是函数级,即描述一个函数内部的数据流信息,可以用控制流图 CFG 来表示,其节点表示语句块,边表示语句块之间的控制流;其次是函数群级,用来描述一个对象内部的几个相关函数之间的数据流信息,即存在函数调用关系,或者说,一个变量在一个函数中定义,而在另一个函数中使用,这用过程间控制流图 ICFG 表示,是几个有关联的 CFG;再次是对象级,描述的是一个对象的所有函数之间的数据流信息,由于在 Web 应用中,一个对象就代表相应的页面,这样一个页面内的所有数据流、控制流信息就能够得以表述,用对象控制流图 OCFG 来表示,它包含了这个对象内部所有函数的 CFG 或者 ICFG;再其次是对象群级,由于页面之间也有联系,有的是直接消息传递,即调用关系,用类似于 ICFG 的图表示,有的是间接消息传递,即通过超链接或者是提交表格从一个页面跳转到另一个页面,用类似于合成控制流图 CCFG 的图来表示,从而可以反映出相关页面之间的关系;最高级是应用级,即各不同用户的具体访问情况,需要考虑多用户访问可能引起的各种后果,类似于并发程序的结构测试。

这样,在不同的等级上,我们可以根据进行不同粒度的测试,从而得到比较完整、全面的信息,最大程度地找出潜在错误,使 Web 应用的质量得到保证。

4.4 基于统计信息的 Web 测试

由于 Web 页面和 Web 用户的数量特别巨大,如果完全按照上述方法进行测试,测试工作将繁重不堪。而实际上,因为各个页面的重要程度是不相同的,只需要进行有针对性的测试即可,重点测试关键页面,使测试效率达到最高。

由于 Web 应用是最终提供给用户的,用户的看法才是 Web 应用好坏的评判标准,所以需要找出用户访问的关键页

面,再对这些页面进行重点测试^[29]。为了找到用户认为重要的页面,需要对用户的行为进行统计分析。通常的做法就是对日志信息进行分析,通过数据挖掘得到用户访问页面的具体情况,从而根据这些历史信息来预测用户的未来行为,找出存在的以及可能的关键页面^[30]。日志文件通常包括提出请求的用户 IP 地址、提出请求的日期和时间、用户请求的文件名、该文件的大小以及 HTTP 状态代码。具体的统计、分析步骤如下:

首先,对一段时间内的日志文件进行分析,统计出访问 Web 页面的实际情况,如各页面的用户访问次数、从一个页面跳转到另一个页面的概率、页面是否能正常响应等信息。

其次,根据上述信息构造页面访问情况图 PVG,其定义如下:

定义4 页面访问情况图 $PVG=(V,E,V_weight,V_error,E_weight)$,是一个有向图,V:代表一个被访问的页面,E:连接两个顶点的边,V_weight:该页面的访问次数,V_error:访问该页面时出错的次数,E_weight:对应两个顶点间的迁移概率。

再次,根据 PVG 图,可以得知用户访问的关键页面,一般认为,V_weight 或 V_error 值较大的页面和 E_weight 值较大的边所对应的两个页面为关键页面,将其提取出来,进行重点测试。

上述过程与 Web 测试相辅相成,经过一定测试的 Web 页面才能提交给用户使用,用户的使用信息决定了页面重要程度的不同,从而可以有选择地进行下一步测试。用户的使用情况不停变动,测试的着重点也相应变化。这两个过程应该很好地配合,以取得良好效果。

5 Web 测试工具

由于 Web 页数目巨大且变动频繁,对其测试如果单靠手工是无法进行的,必须有测试工具的参与。前面我们提到了一些测试工具,它们在保障 Web 应用的功能、性能方面起到了重要的作用。在对 Web 应用建模以后,为了深入内部进行白箱测试以对故障精确定位,也需要一定的工具支持,F. Ricca 等^[24]提出了两种辅助测试工具,ReWeb 和 TestWeb,使用这两种工具可以实现半自动化的 Web 测试,其测试流程图如图 3 所示。

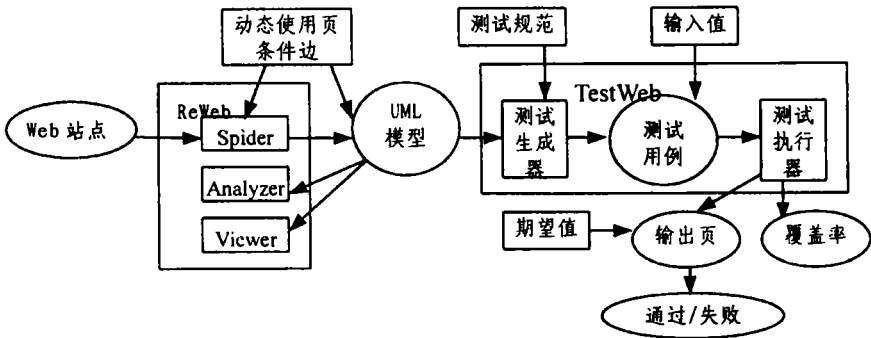


图3 应用 ReWeb 和 TestWeb 的测试流程图

ReWeb 是用来从网站收集页面以生成 UML 模型的工具,其中 Spider 通过请求-响应模式从服务器获得网页,由于 Spider 不能判断页面如何生成,因而需要人工提供动态页、条件边等信息,这样综合而得的信息就能形成用 UML 表示的

Web 模型,如对象关系图、页面导航图、对象状态图等,包括了页面之间的链接信息、页面之内的结构、数据流信息等。可以用 Analyzer 对这个 UML 模型做一些分析,找出它们之间的潜在联系,也可以根据统计信息对模型做适当的简化,如根

据页面访问情况图找出关键页,以减轻测试工作量。用 Viewer 来显示这个模型,从而给测试者较为直观的印象。

在 Web 模型的基础上,用 TestWeb 来实现测试。结合 UML 模型和手工输入的特定测试规范,测试生成器产生了具体的测试用例,如根据页面导航图生成链接可达性测试用例、根据对象状态图生成状态依赖测试用例,以及根据变量的定义-使用链而生成的不同测试等级内的结构测试用例等。在具体测试时,可能会要求手工输入一些值才能运行测试用例,从而运行测试执行器,然后输出一些页面并得到一些覆盖率信息。输出页面与期望值相比较,得到测试的最终结果:成功或失败。

这样,在不多的人为干预下,Web 测试半自动化地运行起来。这样的测试能够包括 Web 测试的各个方面,并可以反复进行,从而能够全面、高效、集中地保证 Web 应用的质量,达到 Web 测试的目的。

结束语 随着 Web 应用种类的不断增长,作为保证 Web 质量和可靠性的重要手段,Web 测试受到人们越来越多的重视。本文首先讨论了对 Web 进行测试的必要性以及 Web 测试不同于传统软件测试的特殊性,接着针对 Web 应用的特定结构,分析可能出现故障的位置,然后详细讨论、分析了现有

的各种测试技术,并总结了面向对象的 Web 测试建模以及利用统计信息对页面进行有针对性的测试,最后以两个测试工具进行 Web 测试的实例来说明、实现这样的思想。这样本文对现有的各种 Web 测试思想、技术和工具进行了一定的分析、综合、比较,用户可以根据实际应用情况来选择使用的测试模型、方法和工具。

参考文献

- 1 张卫丰,徐宝文,周晓宇,李东,许蕾. 元搜索引擎研究. 计算机科学,2001,28(8):36~41
- 2 Gao J, Chen C, Toyoshima Y, Leung D. Engineering on the Internet for Global Software Production. IEEE Computer, May 1999. 38~47
- 3 Powell T A, et al. Web Site Engineering: Beyond Web Page Design. Prentice Hall, 1998
- 4 Jamsa K, 王玉琳, et al. 译. WEB 程序设计教程. 电子工业出版社, 1997
- 5 Automated Web Testing. <http://www.internetqa.com>
- 6 Web Site Testing. <http://www.telsoft-inc.com>
- 7 Software QA and Testing Frequently-Asked-Questions. <http://www.softwareqatest.com>
- 8 顾庆. 分布式程序测试研究. [南京大学博士生毕业论文]. 2001
- 9 Binder R V, 华庆一等译. 面向对象系统的测试. 人民邮电出版社, 2001
- 10 Silk Test. <http://www.segure.com/html/s-solutions/s-silk-test/s-silktest-toc.html>

(下转第110页)

(上接第107页)

理量减少。另外,对每个模块都以数据块形式进行操作可充分利用高速缓冲存储器,避免引起不期望的等待延迟。

5. VDR 的典型应用

虚拟无线电的应用依赖于高性能的 A/D 转换器,但并不采用 DSP,而是选择快速发展的工作站硬件。下面以软件蜂窝移动电话接收机为例介绍一个虚拟无线电的具体应用。

软件蜂窝移动电话宽带数字接收机工作在美国“A-side”蜂窝移动电话频段^[3]。接收机可连续地监视 10MHz 宽的蜂窝频带,可解调其间任何一处 FM 信号。图6为蜂窝移动电话接收机方框图。前端是 RF 接收机,其功能是把 825~835MHz 带宽的信号下变频到基带信号,然后以 25.6MSPS 采样信号速率对信号进行采样。12bit 采样流流入 GuPPI 卡后,所有信号处理都在软件中执行。

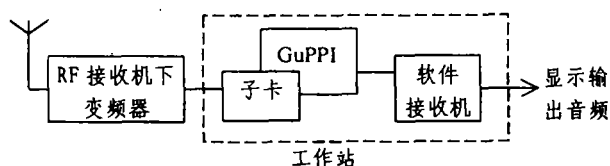


图6 蜂窝移动电话接收机框图

宽带软件接收机处理步骤如图7所示。值得注意的是所有系统都由软件进行控制。这样即使接收机正在工作,这些参数也可以很容易地被修改。



图7 软件结构框图

处理过程的第一步是信道选择滤波。该模块的任务是从 10MHz 频带中抽取窄带 FM 信号 (AMPS 信道带宽为 30kHz)。这个步骤采用了新型滤波设计,把 3 个传统步骤——下变频信号到基带、低通滤波和抽选采样速率结合在一起。信

道选择滤波采用 RF 前端 $R_s = 25.6$ MSPS 的原始采样作为输入,并以一个可变中间采样速率来产生一个复合基带信号。在复合 FM 信号中,所需信息(在这里指语音)由连续频率携带,它是复合信号相位的时间微分。因此,第二步信号用简单正交解调算法解调,用连续采样间相位微分估计信号相位微分。第三步处理是作为限制冲击响应 (FIR) 滤波完成的。首先简化低通抽选滤波,当采样频率降到音频速率时,把可能引起偏移的高频成份除掉。最后一步是低通滤波,其目的是把带外噪声从语音信号中除去。

类似这样的软件接收机具有开发快捷,在开发期间充分利用大量容易修改的滤波算法进行实验等特点,并可利用现有的器件或软件,这不但有利于缩短开发周期,也有利于增加新功能。

此外,虚拟无线电也可以应用于实现软件无线网络接口卡,使网络接口卡功能可动态调整。软件无线网络接口卡系统比普通的网络接口卡更具有动态灵活性、便携性、兼容性强和软件重复使用率高等优势。

结束语 软件无线电技术是当今计算机技术、超大规模集成电路和数字信号处理技术在无线电通信中应用的产物。它已与第三代移动通信系统紧密的结合起来,第三代移动通信系统的研究推动了软件无线电技术的发展,而软件无线电技术又促使第三代移动通信系统更加灵活的实现。它在我国提出的第三代移动通信系统 SCDMA 中,应用就更广泛了,SCDMA 系统的基站和终端都采用了高速数字处理器和高速 A/D 变换器,处理速度高于 5000 万次/秒,全部基带信号处理和变换都用软件来完成。虚拟无线电作为新一代的软件无线电技术将在未来发挥着巨大的作用。

参考文献

- 1 Mitola J. Cognitive Radio: Making Software Radio More Personal. IEEE Personal Communications, 6(4)
- 2 王际兵,赵明,姚彦. 软件无线电发展动态. 清华大学学报(自然科学版), 1999, 39(9)
- 3 Software Special Issue. IEEE Communications Magazine, Feb. 1999
- 4 吴伟陵. 移动通信中的关键技术. 北京: 邮电大学出版社, 2002