

基于质量属性的软件测试力系统分析架构

张洪春

(湖北职业技术学院 信息技术学院,湖北 孝感 432000)

摘要: 在深入研究软件测试过程和技术实践的基础上,提出了软件测试力概念,构建了软件测试力系统(STPS)体系结构。以此为基础,通过分析 STPS 关键需求,确立系统的质量因素,构架了基于质量属性的 STPS 结构模型。从而实践了构架技术与行业实际的有效结合,促进了软件构架技术在软硬件融合的系统构架中的应用,为进一步研究 STPS 高效的运行机理奠定了基础。

关键词: 软件测试;软件构架;软件质量;质量属性;软件测试力系统;构件

中图分类号: TP311

文献标识码: A

文章编号: 1674-7720(2011)19-0003-04

Analysis and architecture of software testing power system based on quality attribute

Zhang Hongchun

(School of Information Technology, Hubei Polytechnic Institute, Xiaogan 432000, China)

Abstract: With a thorough study on software testing process and technical practice, this paper defined the concept of software testing power, and proposed the architectural framework of STPS. Based on the preceding and critical demands of STPS arising thereof, this paper made clear the quality factor of STPS and structured the quality-based STPS model. Thus effective combination between architectural technology and industrial practice was realized to facilitate the use of technology of software architecture in systematic framework integrating with software and hardware, and a foundation was built for the further research on efficient running mechanism of STPS.

Key words: software testing; software architecture; software quality; quality attribute; STPS; component

在测试技术发展的同时,测试全过程的行为和管理显得尤为重要。一个成功的测试项目,离不开对测试过程科学的组织和监控,高质量的过程体系已成为测试成功的重要保证。但是,就目前软件工程发展的状况而言,普遍存在测试活动组织性、计划性不强,测试管理不完善,测试流程不标准,测试过程失控,测试效率低下,以致影响整个软件工程质量的现象。为此,本文分析和研究软件测试全过程活动,以可管理性、高效率、可扩充性等质量需求为核心,以最终的“规划与准备”、“测试执行”、“监督与度量”、“分析与表达”、“结果与反馈”的功能集成为主线,构造了 STPS 管理软件构架,以实现领域构架到软件构架的映射^[1],促进软件测试理论的进一步研究和软件测试体系的建构。

软件测试力是指围绕软件产品测试所拥有的独特

测试资源和整合这些资源所形成的高效持续的测试能力。软件测试力系统 STPS (Software Testing Power System) 是包括测试工程管理和测试工程技术的一个完整实体,是测试全过程活动的集成体系,是包括软件系统的测试系统。STPS 内涵丰富,影响要素众多,且各要素间存在着复杂的多层次、非线性相互作用,是一个开放式、可集成的复合系统^[2]。

测试力的概念是针对测试系统效力提出来的,其目的是整合测试所涉及的所有软硬件资源,用系统工程观点来研究测试全过程活动,以提高测试系统的效能。

软件构架是对系统整体组织结构和控制结构的刻画,它包括系统中各计算单元构件的功能分配、各单元之间的高层交互连接器说明以及软件构架的约束^[3]。

软件构架体现在系统高层次的抽象,着重解决软件系统的结构和需求向实现平坦过渡的问题^[4]。研究软件

体系结构的目的是为软件系统提供合理的构架,重点解决应用系统开发中的总体结构问题,用以实现既定的商业目标和系统的质量属性。

软件构架是软件开发领域的一门新兴学科,是软件业的一个重要的研究领域,正受到越来越多的关注。但目前构架技术及应用还处于研究发展阶段,存在很多不足:(1)无论从技术角度还是从管理角度,针对实际软件开发组织的、有关如何管理软件构架的实用研究文献十分缺乏。(2)现有的构架多侧重于软件构架,对于软件、硬件、过程、管理及目标相结合的系统构架研究较为少见,更是很少有关于如何把系统构架与行业或组织的实际情况结合起来的探讨。(3)现有构架只依据静态的系统目标来设计,没有考虑动态的过程,如人力资源、进度要求以及开发环境的满足情况等。

因此,在分析测试活动全过程的基础上,提出测试力及其系统概念,分析和研究软件测试的体系结构,并用 STPS 来描述全过程活动,用全过程构架建立 STPS 结构模型,将有助于测试全过程的行为管理,提高 STPS 的质量。同时,通过构架实践,实现了软件构架技术在结合了环境、过程、管理和目标的系统构架中的应用,推动了领域构架转化为软件构架的进一步研究,为促进测试领域向系统化、专业化、高效化方向发展。

1 STPS 结构与目标

STPS 涉及到环境、过程、管理、软硬件和目标等因素,是技术、商业和社会等诸多因素作用的结果。STPS 结构不同于一般的系统管理结构,它是系统中的系统,是静态和动态结合的过程管理。

1.1 STPS 结构

根据测试力的定义及参考相关文献,可将测试力分为硬力和软力两大部分^[5]。硬力由环境力、设施力、科技力、劳动力、结构力和聚集力组成;软力由文化力、制度力、管理力、发展力和秩序力组成。硬力为弓,软力为弦,测试力为箭,三方共同作用形成测试力的“弓弦箭”模型。“弓弦箭”模型的关注点是“能力”、“效力”问题,这正好体现了 STPS 的本质要求。

STPS 由基础环境要素、测试要素和目标要素构成。测试要素又由 5 个相互关联、相互作用的测试执行过程组成。具体来说,STPS 包括人员、设施环境、物资技术、管理制度、测试执行全过程以及系统目标等元素,并由数据、行动、设备、事件和线索来构造^[6]。为方便分析,将 STPS 的结构按工程的思想物化为测试输入、测试生产和测试输出等工程行为过程,与之对应,系统分为如图 1 所示的三个作用组件。

(1)输入组件:是 STPS 起到保障作用的条件域,包括人员、设施环境、管理制度、激励机制,产业发展水平、学科体系状况以及测试在软件开发过程中的地位等因素,其数据的表现形式是人力、资源力、技术力、管理力

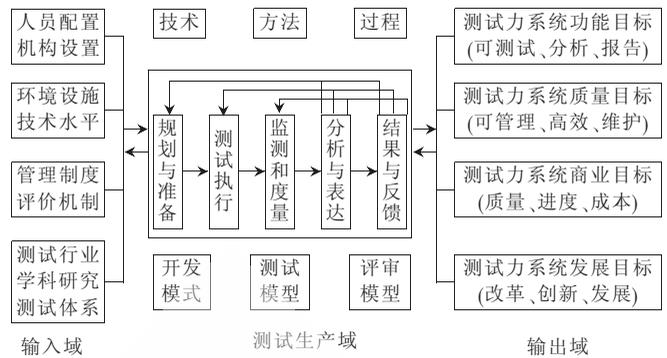


图 1 STPS 结构组成

与领域状况力。输入域是实现系统需求的条件和基础。

(2)测试组件:是决定 STPS 成效的关键过程域,包括测试规划和准备、测试执行、监测和度量、分析和表达、结果反馈五个阶段^[7-8],这五个阶段阶段相互联系又相互影响,在域内部形成一个行为闭合回路。另外,此域也有一些内部的作用子要素在起作用,如测试行为中各阶段的技术、方法的使用和过程设计;软件开发模式下对测试模型和评估测量方式的选择等。测试生产域是实现系统需求的关键域。

(3)输出组件:是 STPS 自身质量的保证体系,也是 STPS 的目标域,包括:实现对业务和系统进行正确测试、分析和报告的功能因素;可管理、高效率、正确性、可更改的系统质量因素;测试的质量、效率、进度和成本收益等测试商业因素;改革、创新和发展目标因素。每一个因素又由多个属性来刻画。追求质量更好、效率更高、进度更快和低成本、高效益以及实现商业目的是 STPS 的最终目标,也是其自身功能完善和可持续发展的内在驱动力。

1.2 STPS 目标

STPS 的作用:一是完成具体软件项目的测试,实现其功能目标;二是在不断提高系统质量目标的同时,以实现其商业目标。STPS 需求可由以下四方面来描述:

(1)功能目标:STPS 要实现的功能是要在各子系统及组件的共同作用下,有效地完成项目业务的测试过程,实现对软件制品所期望的各项测试,以实现自己的功能目标(测试、分析和报告)。

(2)质量目标:STPS 对软件制品进行测试时整体所表现出来的性能和能力,它用质量属性来度量,用构架来实现。STPS 质量目标包括可管理性、高效率 and 正确性、可扩充性、可更改性等质量属性。

(3)商业目标:STPS 商业目标需求要用对某具体业务项目测试来衡量,即希望测试某具体项目时测试的质量、进度和效率要好,以此实现 STPS 系统的整体工作质量、效力和效益。

(4)发展目标:STPS 的改革、创新和发展。

《微型机与应用》2011 年第 30 卷第 19 期

2 STPS 构架因素

2.1 构架商业周期

构架是技术、商业和社会等诸多因素作用的结果,而构架又能返回并影响到开发环境,这种相互影响的周期,称为构架商业周期(ABC)^[9]。ABC 陈述了当设计师开始构建系统时的各种影响因素,并指出了特定的质量属性需求通常产生于组织的业务目标。STPS 系统根据涉及及质量属性所构架的商业周期如图 2 所示。这里的最终用户是测试人员,用户则是测试组织。

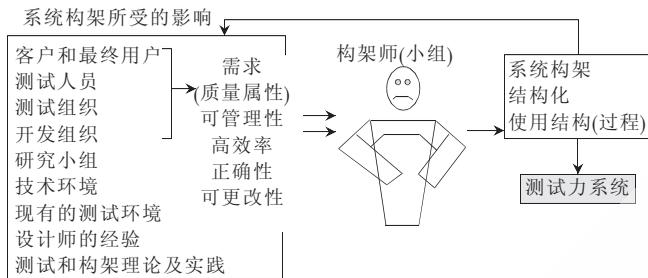


图 2 STPS 构架商业周期

2.2 需求与质量

2.2.1 STPS 质量需求

软件质量体现了所希望的各种属性组合的程度^[10-11],通过质量属性来度量。质量属性是系统在其生命周期过程中所表现出来的各种特征。在构架设计过程中应考虑质量属性,并通过构架实现质量需求。质量需求是系统对软件制品进行测试时整体所表现出来的性能。

本文树立了一个重要的理念:商业目标决定质量目标。提高系统质量的最终目的是为了赢利,而不是创造完美无缺的产品。因此对于普通商业软件而言,并不是“质量越高越好”,而是恰好让广大用户满意,并且将提高质量所付出的代价控制在预算之内。因此,STPS 质量属性通过以下顺序获得:商业目标→用户需求→构架需求→质量属性。STPS 系统的四个最重要质量属性需求是:

(1)可管理性:通常是指各组件的工作协调和通信的有效管理。它要求测试工作便于人员分工(模块独立性、测试工作的负载均衡、进度安排优化、预防人员流动对测试的影响等);测试过程便于资源的配置管理,便于控制系统运行、监视系统状态、错误处理及使模块间通信的简单等;测试结构要求模块要大小合理和适度的复杂等。

(2)高效率:效率包括时间效率与空间效率。在时间效率方面,用户的响应时间要小、吞吐量要大;空间效率方面,运行过程中占用资源要少^[12]。高效率通常是指系统的响应能力,即对外部刺激(事件)做出反应时所需要的时间或在某段时间内所处理的事件个数。这是一个关键的质量目标,是通过更小的成本实现更有效测试来度量的。

(3)正确性:是指系统按照需求正确执行任务的能力。正确性的语义涵盖了精确性,正确性无疑是很重要的软件质量属性。

(4)可更改(可维护)性:它不仅反映了系统适应变化的能力,即要求进行快速修改系统并使修改代价尽可能低的能力;而且也反映了系统进行扩展的能力,即它能适应更多的测试环境要求并使结构重新调整操作的能力。

2.2.2 STPS 功能性需求

功能是指系统所能完成的业务工作。功能性需求是指系统能够完成所期望的工作的能力。功能需求往往通过技术需求分析得到,并由用例来规范、检验和驱动,由开发过程及软件设计来实现。STPS 要实现的功能是要在各子系统及组件的共同作用下,有效地完成测试过程,实现对软件制品进行所期望的测试。通过分析,STPS 必须完成以下功能:

(1)按规格说明书,完成对软件制品进行所期望的测试。

(2)对业务测试过程进行监督,测试结果进行分析并输出测试报告。

(3)反馈业务的测试结果,提出改进建议。

质量需求与功能需求是正交的,质量需求高于功能需求,即质量属性要求高于系统功能(系统能力、服务和行为)的基本要求。两者的共同作用实现 STPS 的目标需求并完成其商业目标。

3 构架解决方案

构架是从系统的总体结构入手,是系统整体组织结构和控制结构的刻画,能最终实现对不同质量属性的满足。

3.1 参考模型

通过分析 STPS 组成、工作过程和质量需求,可知系统应由环境组件、测试组件和管理控制组件三个相互独立又相互联系的组件(角色)构成。每个组件都要被分配功能责任,并使用其他组件的运行结果。图 3 为 STPS 参考模型,展示了组件与测试人员和测试过程的交互以及系统的目标构成。

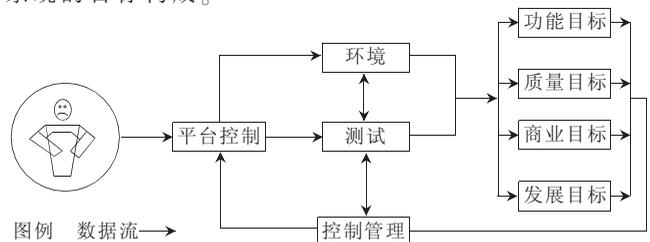


图 3 STPS 参考模型

3.2 构架结构

构架由构件组成,是多种结构的体现。结构是系统构架从不同角度观察所产生的视图。通过图 3 的 STPS 参考模型,结合其工作过程,可分析出 STPS 构架应由下列结构来体现。

(1)结构化模型。STPS 结构化展示了软件元素与外部环境中的元素之间的关系,它是一种调用和返回的体系结构^[13]。其中,测试结构由多个子模块和子过程组成,各个部分是一种使用、依赖的过程关系;环境与管理控制结构,由多个要素组成,各要素也必须相互协调并共同作用。

(2)使用结构。使用结构的思想是:如果过程 A 的运行必须以过程 B 的正确运行为前提,则认为过程 A 使用过程 B。使用结构是模块中的过程,描述了各组件之间在运行时使用关系,展示了各组件在运行过程中是如何交互来完成任务的。通过对 STPS 测试行为过程分析,可知 STPS 是一种顺序批处理构架,构件及各子组件存在着明显的使用关系。

从整体来说,STPS 中的测试结构模型是一种分区结构,是对构架的纵向划分;环境与目标结构模型是分层结构,是对构架的横向划分。因此,在物理上,STPS 是一种“分层+分区+分层”结构;在逻辑上,STPS 又是一种结构化下的使用关系。

3.3 系统构架

3.3.1 STPS 管理模块和应用模块

结构化模型就是构架模式。STPS 可通过二个相互关联但又不相同的结构进行描述,以满足其质量属性要求。在粗粒度上,可以把 STPS 结构化模型架构样式分成管理和应用两大部分^[14],如图 4 所示。

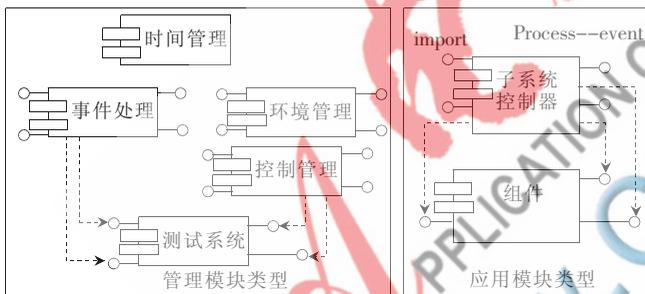


图4 测试力系统构架样式

管理部分:用以处理协调问题,包括组件的实时调度、组件之间的同步、从指挥控制台上对事件进行管理、数据共享、数据完整性等。管理模块类型中的时间管理是调度机制的基础,它负责同步、提供时钟;事件处理器负责协调模块及子系统工作;控制管理则负责测试全过程质量管理;环境管理则负责资源的调度。

应用部分:处理各个系统的运算,对测试系统建模等。其功能由各个子系统和构件完成。应用模块类型中的子系统控制器用以将相关子模块的一组功能联系起来,实现对子系统整体的控制、逻辑交互和通信。子系统控制器必须完成如下的功能:

(1)按照某个事件的响应,根据一组初始化条件,完成自身及各个组件的初始化。

(2)按照对各组件功能的了解,把对故障及参数的请

求路由给各个组件。

3.3.2 组群划分及分解

将 STPS 建模问题划分成更易管理的单位的过程中,测试结构系统成为问题的焦点。由于测试系统受到多个组群的影响,通过做最粗粒度的分解,可将 STPS 划分为环境组群、测试组群、管理组群和控制组群等具体组群,然后再进一步将组群分解为系统,以满足其质量要求。例如通过 STPS 的结构组成和参考模型,可将测试组群进一步分解成如下的系统:规划与准备系统、测试执行系统、监督与度量系统、分析与表达系统、结果与反馈系统。

3.3.3 系统分解为子系统

最后,可再将系统分解为子系统,以满足其功能要求,实现 STPS。例如测试执行系统到子系统的进一步的分解如图 5 所示。



图5 测试执行子系统

通过上面的构架过程,已经确定了功能的划分、子系统到子系统之间的控制分配,也确定了子系统之间的联系。要完成 STPS 框架,还需要做如下的工作:

(1)确定测试执行系统的组件实例。

(2)采用类似的方法对其他组群、系统和子系统进行分解。

本文定义了软件测试力概念,提出和分析了 STPS 体系结构。在软件过程理论和构架技术的指导下,从质量属性的角度出发,结合 STPS 项目实际,探讨了 STPS 的过程构架和软件构架,实现了软件构架技术在涉及环境、过程、管理和目标等软硬件结合系统领域构架中的具体应用,推动了软件测试过程及其研究方法的创新实践。

对 STPS 进行全面、系统的分析是一个十分复杂的问题,模型的构架也因为涉及到软硬件结合的复杂环境和较多的目标需求而变得困难。因此,本文的研究也显露了一定的局限性,比如未能构架 STPS 体系中除测试系统以外的其他部分组件及它们之间的联系成分;未能对系统构架在实现系统质量、进度、成本和效率等方面的现实意义作出论证等。但是,通过建立 STPS 构架模型,促进了软件测试全过程活动的实际研究,为测试过程向系统化、专业化、高效化方向发展奠定了基础。

参考文献

[1] 徐铭,万麟瑞.基于多 Agent 的 JIT 软件构架研究[J].计

- 计算机工程与设计, 2006, 27(21): 4052-4053.
- [2] 郑睿, 李汉铃. 基于多属性的城市竞争力系统结构分析[J]. 系统工程理论方法应用, 2006, 15(2): 170-171.
- [3] 何坚, 覃征. 软件构架动态行为建模与检测[J]. 计算机研究与发展, 2005, 42(11): 2018-2024.
- [4] 卓家靖, 孟晨, 方丹. 并行自动测试系统软件体系结构建模[J]. 计算机工程, 2009, 35(18): 72-78.
- [5] 郑睿, 朱清香, 李汉铃. 城市竞争力系统构成要素的定量化解析[J]. 燕山大学学报, 2004(5): 58-64.
- [6] JORGENSEN P C. Software testing: a craftsman's approach (Second Edition)[M]. 韩柯, 杜旭涛译. 北京: 机械工业出版社, 2005.
- [7] 陆永忠, 宋骏礼, 谷希谦. 基于行为的软件测试过程模型及研究[J]. 计算机应用, 2007, 27(5): 1239-1240.
- [8] 樊庆林, 吴建国. 提高软件测试效率的方法研究[J]. 计算机技术与发展, 2006, 16(10): 52-54.
- [9] BASS L, CLEMENTS P, KAZMAN R. Software architecture in practice[M]. 孙学涛译. 北京: 清华大学出版社, 2004.
- [10] 李庆义, 岳俊梅, 王爱乐. 软件测试技术[M]. 北京: 中国铁道出版社, 2006.
- [11] 杜文浩. 软件测试教程[M]. 北京: 清华大学出版社, 2008.
- [12] 梅宏, 王千祥, 张路. 软件分析技术进展[J]. 计算机学报, 2009, 32(9): 1704-1706.
- [13] PRESSMAN R S. Software engineering a practitioner's Approach[M]. 梅宏译. 北京: 机械工业出版社, 2005.
- [14] 谢新华. 软件架构设计的思想与模式[EB/OL]. <http://wenku.baidu.com/view/>, 2010.
- [15] Mei Hong, Zhang Lu, Yang Fuqing. A component-based software configuration management model and its supporting system [J]. Computer Science and Technology, 2002, 17(4): 432-441.

(收稿日期: 2011-06-01)

作者简介:

张洪春, 男, 1963年生, 硕士, 副教授, 主要研究方向: 软件工程。