

性能测试专家分析系统^①

张永祥, 路红英, 肖俊

(北京交通大学 计算机科学与技术学院, 北京 100044)

摘要: 为了降低性能测试结果分析的难度, 提高性能测试结果分析的准确性, 提出了针对性能测试结果分析过程的专家系统. 首先对性能分析的对象及方法进行了归纳总结, 通过进一步研究人工分析过程以及分析知识结构提出了对结果数据以及分析知识的结构化表示方法, 并依此简要讨论了测试结果分析专家系统的结构.

关键词: 性能测试; 测试结果分析; 专家系统; 测试工具; 知识表示

Expert Analysis System for Performance Test

ZHANG Yong-Xiang, LU Hong-Ying, XIAO Jun

(School of Computer Science and Technology, Beijing Jiaotong University, Beijing 100044, China)

Abstract: To reduce difficulties of performance analysis and improve performance test quality, the expert analysis system for performance test is researched. The performance analysis objects and methods are first summarized. By the analysis of manual method and knowledge prototype, a structured representation method is studied. Based on these studies, the structure of expert analysis system is simply designed.

Key words: performance test; test results analysis; expert system; test tools; knowledge representation

性能测试属于软件测试中的系统级测试, 它针对软件在继承系统中运行的性能指标进行测试, 旨在及早确定和消除软件中的性能瓶颈^[1]. 相对功能测试性能测试具有更高的专业性和复杂度, 这一点尤其体现在测试结果分析过程中. 性能测试结果的数据分析要求测试人员具备丰富的专业知识以及敏锐的洞察力. 只有专业的测试人员才能通过测试数据分析快速准确的发现系统性能缺陷并对系统进行正确的评价. 性能测试的分析包括了操作系统、数据库、应用软件等不同方面的内容, 由于对知识储备以及工作经验的需求, 性能测试的结果分析往往成为测试实施过程的瓶颈. 测试人员往往以小组讨论的形式进行数据分析, 而缺乏丰富经验的测试人员通常无法有效的发现系统缺陷甚至可能对系统做出错误的判断. 如何有效的降低结果分析的难度成为了性能测试执行过程的一个重要问题.

分析型专家系统是一种在特定领域内具有专家水平解决问题能力的程序系统. 专家系统将应用专家知识通过模拟测试专家的推理过程对测试结果进行分析.

在性能测试的分析阶段, 分析型专家系统能够扮演指导者以及辅助者的角色. 测试人员通过应用分析专家系统对数据进行初步分析并对分析结果进行筛选, 从而得出对于系统的综合评价. 因此测试结果分析专家系统将有效的降低性能测试的执行难度并提高测试质量以及测试效率.

1 分析模型

性能测试分析围绕着系统软件、硬件两方面进行. 测试人员在测试执行过程中监控并记录系统各项指标参数. 测试执行后通过分析不同执行场景下的软硬件参数信息对系统进行综合分析及评价. 而对于不同指标测试人员有着不同的评定方式.

例如:应用 LoadRunner 执行测试场景并针对 LoadRunner 获得的内存参数对系统进行分析, 测试人员会考查其记录的以下各项参数:

- ① Memory/Available Mbytes 系统可用内存数量.
- ② Memory/Page Reads/sec 内存中读操作数量,

^① 收稿时间:2012-09-28;收到修改稿时间:2012-11-11

不包含读操作的页面数.

③ Memory/Pool NonPaged 非换页池中的字节数, 非换页池是指系统内存中可供对象使用的一个区域.

④ Memory/Cache Bytes 静态文件缓存大小.

⑤ Memory/Cache Faults/sec 系统在文件系统缓存中查找数据失败的次数.

⑥ Server/Pool Nonpaged Failures 从未分页池中分配内存是出错的次数.

⑦ Memory/Pages/sec 页面读取或者写入磁盘以解决硬页面错误的比率.

测试人员将分别考察不同参数的值从而确定是否存在内存问题. 而这个过程是测试人员有针对性的提出问题并根据记录数据回答问题的过程. 例如:

① 提问: Memory/Available Mbytes 少于 20%吗?

回答:少于. 推论: 测试过程系统内存不足.

② 提问: Memory/Page Reads/sec 很高吗?

回答: 高数值. 推论: 表明存在内存瓶颈.

回答: 底数值. 推论: 进一步提问, Physical

Disk/%Disk Time 和 Physical Disk/Avg.Disk Queue Length 很高吗?

回答: 很高. 推论: 应该是硬盘瓶颈.

③ 提问: Server/Pool Nonpaged Failures 是非 0 正值吗?

回答: 是. 推论: 则表示存在内存瓶颈有可能是内存泄露引起的.

④ 提问: Memory/Pool NonPaged 比系统启动时增加了 10%以上吗?

回答: 是. 推论: 则存在很严重的内存泄露.

综上所述可以看出测试人员对于数据的分析过程也就是针对所收集数据提问和回答的过程. 而如何针对参数内容提出问题并能够给问题以正确的回答和推论是测试人员将要面对的主要困难. 提出问题以及回答问题的能力则来源于测试人员的经验以及知识积累. 测试人员提出以及解决问题的方式可以表示为如图 1 所示的树形结构.

2 专家系统模块结构

由于应用的复杂程度和功能的不同, 专家系统在设计过程中结构往往有着一定的差异. 但从概念出发大部分专家系统具有相似的体系结构. 根据测试分析专家系统的信息输入、信息的处理过程, 测试结果分析型专家系统的可以根据处理对象划分为三大模块,

这三个模块分别为:

测试数据数据输入及存储模块. 这个模块主要为专家系统的分析对象即测试数据提供输入、处理和存储功能.

专家知识获取及存储模块. 这个模块主要为专家系统的专家知识提供输入和存储功能.

分析专家系统综合处理模块(推理机和解释器). 这个模块是专家系统的工作核心, 主要完成推理和解释工作.

测试分析专家系统的结构为图 2 所示.

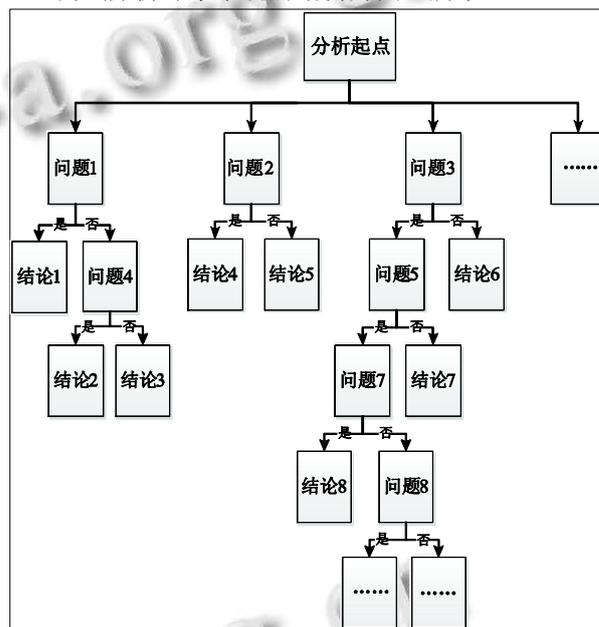


图 1 人工分析模型

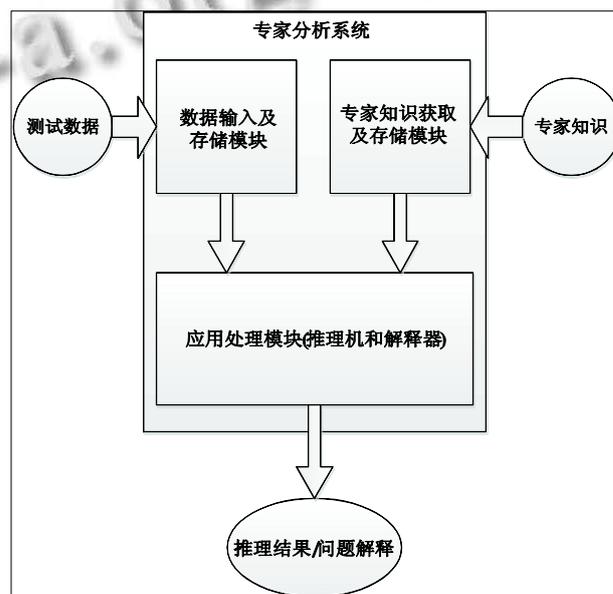


图 2 专家分析系统功能结构

3 测试场景的对象表示

测试执行过程中所获得参数(测试数据)是专家系统的分析对象,也就是专家系统的输入信息.在 CPU、内存、IO 的分析过程中往往会应用到许多不同的分析参数.这些参数在每一次测试执行过程都会予以记录和分析.如何有效存储和组织这些参数直接关系到专家系统的实现.

测试场景是性能测试的基本执行单位,场景设计信息中包含大量测试基本信息.例如:系统配置、负载压力、执行时间等.以测试场景为对象,将测试数据组织到场景对象中会对数据进行有效的分类管理.我们以 XML 形式对测试场景进行存储.其基本结构如图 3 所示.

Scenario 节点代表了一个测试场景,每个测试场景的属性都以 param 节点进行存储,同时参数内容也以 param 节点信息存储到测试场景中.在对场景执行情况进行记录的过程中,测试人员可根据不同的内容定义自己的 param 节点.系统将以节点的 name 属性对其进行区分.

```
<scenario>
  <param name=" 场景名称 " >xxx压力测试</param>
  <param name=" 版本编号 " >v1.0</param>
  <param name=" 场景编号 " >xxx-xxx</param>
  <param name=" 使用到脚本编号 " >01</param>
  <param name=" 测试目的 " >... ..</param>
  <param name=" Memory/Available Mbytes " >15%</param>
  <param name=" Memory/Page Reads/sec " >... ..</param>
  <param name=" Memory/Pool NonPaged " >... ..</param>
  ... ..
</scenario>
```

图 3 场景信息 XML 结构

4 知识表示方法

结果分析专家系统由测试知识驱动,知识的表述过程是一个提问与回答的过程,因此针对部分内存分析问题我们提出了如图 4 形式的 XML 知识表示形式.

在 XML 文件中, knowledge 节点代表分析知识.其 id 属性代表了知识类型,例如,对于 CPU,内存,以及 IO 的分析性知识.每条知识节点中包含了若干的分析规则,即对应的 rule 节点.分析规则包含了相应的评测方法,即专家系统要执行的分析动作 action,分析动作可以为系统提供的方法或者人工驱动, input 以及 result 子节点表示方法的输入以及结果,结果动作可以是得出结论信息或者执行下一个分析规则 rule.

根据 XML 的结构推理机调用知识内容进行推理时,将依照如下的步骤执行.

① 执行规则 1. 调用方法 below 进行检查. below 的输入参数为场景的 Memory/Available Mbytes 及 0.2, 并判断第一个参数是否小于第二个参数.如果 below 返回为 true 则推理机产生消息“系统执行过程中内存不足”.

② 顺序执行规则 2. 规则将调用方法 over 方法进行检查. over 的传入对象为场景的 Memory/Page Reads/sec, 200, 并判断第一个数是否大于第二个数. over 返回结果为 true 则产生消息“存在内存瓶颈”,若返回 false 继续调用规则,同时检查 Physical Disk/%Disk Time 及 Physical Disk/Avg.Disk Queue Length, 返回为 true 则产生消息“存在内存瓶颈”.

```
<knowledge id = "memory">
  <rule name="check Memory/Available Mbytes">
    <action>below</action>
    <input> Memory/Available Mbytes , 20%</input>
    <result type=" true " >message(系统内存不足)</result>
  </rule>
  <rule name="check Memory/Page Reads/sec">
    <action>over</action>
    <input>Memory/Page Reads/sec,200</input>
    <result type=" true " >message(存在内存瓶颈)</result>
    <result type=" false " >
      <rule>
        <action>over</action>
        <input>Physical Disk/%Disk Time,50 </input>
        <input>Physical Disk/Avg.Disk Queue Length,15 </input>
        <result type = " true " >message(存在硬盘瓶颈)</result>
      </rule>
    </result>
  </rule>
  .....
</knowledge>
```

图 4 知识 XML 表述形式

5 推理机工作机制

作为专家系统核心部件,推理机负责完成专家系统的推理工作.在上一章中我们讨论了知识的表示方法以及解析方法.推理机主要完成的工作也就是通过知识的结构驱动完成推理.其基本结构如图 5 所示.

① 获取场景信息并保存. 场景信息中包含了性能测试的分析对象,场景信息经过处理后以 XML 形式存储于专家分析系统的综合数据库中.推理机逐个抽取每个场景的对象信息,并解析 XML 文件,最终将其实例化到内存中.

② 获取分析知识. 分析知识信息包含于专家系统给的知识库中,并以 XML 形式存储.推理机在获取场景信息同时,也将逐条抽取知识信息,并解析相应的知识信息,将其实例化到内存之中.

③ 调用方法表中的方法并依据知识结构进行推理. 方法可以分为两种类型, 即“机器提问人为回答”方式或“机器提问机器回答”方式. 推理机根据通过解析的知识结构信息查找并调用“方法表”中定义的方法. 进而根据方法返回参数进行推理. 推理机针对场景内容引用知识库中的知识进行推理. 直到知识扫描完毕.

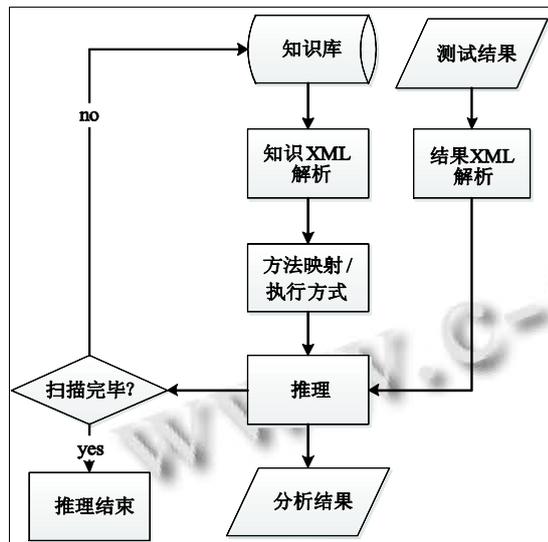


图5 推理机结构

6 专家系统要面对的问题

在专家系统的建立及应用过程中主要面对的问题有以下几点:

① 知识储备

知识是专家系统运行的基础, 知识储备量决定了专家系统的分析能力. 如何有效的扩充分析专家系统

的知识量, 以及确保专家系统的知识质量是专家系统在日常应用中要面对的重要问题.

② 分析方法的扩充

分析方法是值推理机在工作工程中应用到的推理动作. 每项知识内容都需要应用相应的推理方法. 对于新知识可能要应用到新的推理方法. 所以专家系统需要开发人员进行维护, 以确保能够适应提交的知识内容.

7 结论

本文提出了针对性能测试数据分析过程的专家系统. 并分别从数据表示方法、知识表示方法、推理机结构三方面简单的讨论了一种性能测试结果分析专家系统的构建和实现方式. 最后依据构建方式进一步讨论了专家系统的特性以及面临的问题.

参考文献

- 李怡,周国祥.基于 Load Runner 的一种性能测试流程方案研究与设计.计算机应用研究,2009,26(11):4143-4145.
- IEEE Standards Board. IEEE Standard for Software Unit Testing: An American National Standard, ANSI/IEEE Std 1008-1987. IEEE Standards: Software Engineering, Volume Two: Process Standards, 1999 Edition.
- Leszak M, Perry DE, Stoll D. A case study in root cause defect analysis. Proc. of the 22nd International Conference on Software Engineering (ICSE'00). 2000: 428-437.
- 620734263.hadoop 和 hbase 分布式配置及整合 eclipse 开发. [2011-07-20].http://wenku.baidu.com/view/8712a661caaed3383c4d392.html
- White T. Hadoop: The Definitive Guide: O'Reilly Media, 2009.
- George L. HBase: The Definitive Guide: O'Reilly Media, 2011.
- 项斌.网络舆情检测系统设计与实现.成都:电子科技大学, 2010.
- 陈旭.基于社会网络的 WEB 舆情系统的研究与实现.成都: 电子科技大学,2010.
- 何忠育.分布式社会网络分析支撑系统研究与应用.广州: 广东工业大学,2011.

(上接第 22 页)

报,2011,51(10).

6 梁斌.走进搜索引擎.北京:电子工业出版社,2007.

7 McCallum A, Nigam K, Lyle H. Ungar: efficient clustering of high-dimensional data sets with application to reference matching. Proc. of the 6th ACM SIGKDD. 2000. 169-178.

8 McQueen J. Some methods for classification and analysis of multivariate observations. Proc. of the 5th Berkeley Symp. on Math. Stat. and Prob. 1967,1:281-296.

9 周涛,柏文洁,汪秉宏,等.复杂网络研究概述.物理,2005, 34(1):31-36.

10 陈旭.基于用户行为及关系的社交网络节点影响力评价——以微博研究为例.北京:北京邮电大学,2011.