上海交通大学软件学院硕士论文开题报告

**面向Web应用的动态负载生成工具设计与实现**

**Design and Implementation of a Dynamic Workload Generation Tool for Web Applications**

|  |  |
| --- | --- |
| 系 别 | **软件学院** |
| 学科专业 | **软件工程** |
| 研究方向 | **软件工程** |
| 姓 名 | **李梦萍** |
| 导 师 | **吴刚** |

上海交通大学软件学院

2013年 5 月 9 日

|  |  |
| --- | --- |
| **课 题 来 源** |  |
| **一、研究背景**  随着计算机网络的发展及Web应用的普及，用户对系统响应的实时性、吞吐量、安全性等性能的要求越来越高, Web应用必须能够处理大量用户的并发请求；同时，云计算的方法对软件和硬件的设计和实现方式产生了巨大的影响，Web应用可以部署在云平台，根据用户需求动态提供资源，在满足服务水平协议(SLA)的基础上，提高了系统的可伸缩性和可靠性。因此，随着Web应用的规模和复杂度的逐渐增加，对其进行性能测试和分析的难度和代价也大大增加。  Web应用的负载特征是动态变化的。对于不用的应用，用户的操作行为不同；例如，Web应用程序从1.0逐渐发展到2.0[1] [2]，用户的行为发生了很大的改变。而对于同一个Web应用，在不同的时间段，并发的用户数目及用户操作行为也处于动态变化之中。  由于一个Web应用常常是大量用户同时使用，人工测试无法完成大量用户的并发请求，因此必须借助自动化测试工具。为了提高测试的效率，降低成本，Workload生成工具被广泛地应用在Web应用的测试过程中。Workload生成工具可以针对特定的Web应用，模拟终端用户的行为，向服务器发送和接收请求，并且获得服务器的响应时间及吞吐量等信息。  目前针对Web应用的Workload生成工具有LoadRunner、Httperf、JMeter、Faban等，它们可以编写或者录制测试脚本、运行测试脚本、最后对测试结果进行分析，并生成测试报告。  本课题的主要目的是希望实现一个针对部署在云平台的Web应用的动态负载生成工具，对Web服务器的可伸缩性进行有效地测试，检测和报告可能出现的问题。 | |

|  |
| --- |
| **二、研究目标**   1. 用户行为模型的研究。在这部分研究中，针对特定的Web应用，分析用户行为，提取用户操作。 2. 研究Faban框架，分析其Workload生成工具的优缺点，以及如何对其进行改进。 3. 分析常用网站的用户行为规律，提取出Workload模型，并利用Faban框架实现这些模型。 4. 用户给定一个函数，针对某个Web应用，可生成符合该函数形状的Workload。 5. 结合Faban框架及Workload模型的研究，实现一个针对Web应用的动态负载生成工具。 |
| **三、关键技术**   1. 针对Web应用的Workload生成工具，包括各种Workload生成的方法及它们之间的优缺点，重点研究Faban框架。 2. 分析常见用户行为规律，提取Workload动态变化模型。 3. 根据用户给定的函数，找出该函数中的特征点，用来定义Workload的变化情况。 4. 在Faban框架的基础上，实现一个动态的Workload生成工具。 |

|  |
| --- |
| **四、课题创新**  首先，由于Web应用的负载是动态变化的。在不同的Web应用中，用户操作行为有很大的不同。因此分析用户行为，对用户的行为变化建模，并在Faban框架的基础上动态生成Workload。  其次，现有的Workload生成工具主要针对Web应用的功能和压力进行测试，本课题目标是对Web服务器的可伸缩性进行有效地测试，能够根据用户给定的函数，生成用户期望的动态负载。 |
| **五、研究进度**   * 2013年2月-2013年4月：收集整理文献资料，确定具体研究方向和可行性。 * 2013年4月-2013年5月：研究Faban框架以及如何利用其Driver Framework生成测试负载。 * 2013年5月-2013年6月：提取并定义用户行为，初步建立Workload模型,并生成动态变化的Workload。 * 2013年7月-2013年8月：完成课题的实验验证，并根据测试结果调整程序，完成动态负载生成工具的开发。 * 2013年9月-2013年10月：完成课题相关论文撰写工作。 |

|  |
| --- |
| **六、参考文献**   1. Web 2.0 on wiki. <http://en.wikipedia.org/wiki/Web_2>. 2. O’reilly T. What is web 2.0[J]. 2005. 3. Httperf project web site. <http://code.google.com/p/httperf/>. 4. Mosberger D, Jin T. httperf: A tool for measuring web server performance. ACM SIGMETRICS Performance Evaluation Review 1998. 5. Nevedrov D. Using JMeter to Performance Test Web Services[J]. Published on dev2dev (http://dev2dev. bea. com/), 2006. 6. YANG P, LI J. Using LoadRunner to Test Web's Load Automatically [J][J]. Computer Technology and Development, 2007, 1: 080. 7. SPEC Benchmarks. <http://www.spec.org/benchmarks.html#web>. 8. Rubis project web site. <http://rubis.ow2.org/>. 9. Sobel W, Subramanyam S, Sucharitakul A, et al. Cloudstone: Multi-platform, multi-language benchmark and measurement tools for web 2.0[C]//Proc. of CCA. 2008. 10. Bianca Schroeder, Adam Wierman, Mor Harchol-Balter. Open vs. Closed: A Cautionary Tale. Proc. NSDI 2006. 11. D. Krishnamurthy et al., A Synthetic Workload Generation Technique for Stress Testing Session-Based Systems, IEEE Technique for Stress Testing Session-Based Systems, IEEE Trans. on Software Eng. 32(11), Nov. 2006 12. Balachander Krishnamurthy, Graham Cormode (02 June 2008). "Key differences between Web 1.0 and Web 2.0". First Monday, Volume 13 Number 6. 13. Hashemian R, Krishnamurthy D, Arlitt M. Web workload generation challenges–an empirical investigation[J]. Software: Practice and Experience, 2012, 42(5): 629-647. 14. Mills Jr D L. Workload Generation for Scalable Social Computing Applications[J]. 15. Anderson K S, Bigus J P, Bouillet E, et al. SWORD: Scalable and flexible workload generator for distributed data processing systems[C]//Proceedings of the 38th conference on Winter simulation. Winter Simulation Conference, 2006: 2109-2116. 16. Armbrust M, Fox A, Griffith R, et al. A view of cloud computing[J]. Communications of the ACM, 2010, 53(4): 50-58. |