

IOmark:一种精确的存储系统性能测试工具^①

宋丽丽^{②*} 张广艳^{③**} 蔡 涛^{*} 郑纬民^{**}

(* 江苏大学计算机科学与通信工程学院 镇江 212013)

(** 清华大学计算机科学与技术系 北京 100084)

(*** 吉林大学符号计算与知识工程教育部重点实验室 长春 130012)

摘 要 针对已有 I/O 记录重放工具在测试存储系统性能上缺乏简单而精确地重现 I/O 负载的能力,容易给存储系统性能指标观测值带来显著误差的问题进行了测试方法研究,提出了一个精确的块级别存储系统性能测试工具 IOmark。它重放来自真实存储系统的 I/O 记录,使用 Linux 内核态异步 I/O 实现对块级别存储系统的访问,通过跟踪生成的 I/O 请求的状态和 I/O 记录中的请求时间来控制 I/O 请求的发送时间并筛选 I/O 记录中的盘号来控制负载轻重,从而实现精确地重放 I/O 负载的功能。实验表明,IOmark 能在 trace 指定时间内将 I/O 请求发送出去,发送时间误差为 0;与同类测试工具相比,引入的测试误差小,对存储系统的性能评测更准确。

关键词 块级别,存储系统,性能测试工具,I/O 记录,精确性,测量误差

0 引 言

高性能块级别存储系统是近年来的研究热点。测试其性能有助于确定影响其性能的关键因素,从而可以有针对性地进行分析和改进,也可用于比较不同存储系统,为选取或购买存储系统提供参考。面对大容量存储设备、复杂的网络存储系统及多维度的应用需求,有必要设计并实现可信、可靠、可重复的存储系统性能测试工具。

存储系统性能的高低主要通过它处理负载的能力来衡量。根据添加负载的真实程度和轻重情况的不同来划分,一般有三种方法可以实现:运行实际应用,即搭建真实的系统环境,添加用户所关心的真实的应用负载;生成人工合成负载,即针对系统的情况以及用户群体的需求等来生成合成的负载,并向该系统播放;播放 I/O 记录,即向存储系统播放已有的得到公众认可的真实的负载。存储系统性能测试的最准确方法是运行真正使用的应用程序。但是,应用程序往往难于配置,甚至由于涉密等原因而不可

使用,而且当用户需求发生变化而重新测试时,需要修改应用和重新配置实验环境,使得该方法的复杂度明显提高。人工负载生成技术具有实现复杂度低的优势,但是人工合成的负载不具有真实性,且配置和运行起来很复杂,它依赖 I/O 负载生成程序并根据用户指定的参数配置,生成用户预期的负载。已有的 I/O 负载生成程序包括 TPC 基准测试程序^[1]、Postmark^[2]、IOzone^[3]、Bonnie^[4]、SPC^[5]、SPEC SFS^[6]、Iometer^[7]等。通过播放 I/O 记录的方式进行块级别存储系统性能测试,兼具运行实际应用的真实程度高和生成人工合成负载的实现难度低的双重优势,即负载真实具有代表性,且不需要配置参数,运行起来简单方便。但是已有 I/O 记录重放工具缺乏简单而精确重现 I/O 负载的能力,容易给存储系统性能指标观测值带来显著误差。本文提出了一个块级别存储系统性能测试工具 IOmark,它能精确模拟运行环境的真实负载变化,实现对存储系统的性能测试。实验表明,IOmark 能在 trace 指定时间内将 I/O 请求发送出去,且与同类工具相比引入的测试误差最小,对存储系统的性能评测更准确,实

① 863 计划(2013AA01A210),国家自然科学基金(60903183,61170008,61272055)和吉林大学符号计算与知识工程教育部重点实验室开放基金(93K172012K12)资助项目。

② 女,1987 年生,硕士;研究方向:网络存储与性能测试;E-mail:sllforever05@163.com

③ 通讯作者,E-mail:gyzh@tsinghua.edu.cn

(收稿日期:2013-04-01)

用价值更高。

1 相关工作

块级别存储系统性能测试需要向存储系统播放真实的负载,面临的主要问题是选取有代表性的真实的负载和精确地播放 I/O 记录。Buttress^[8]在精确地播放 I/O 记录方面做出了贡献,将 I/O 请求的发出时间精确到微秒级,能在 100 μ s 内将 I/O 请求发送出去,降低了测试误差,但是设计方法较复杂;IOmark 设计方法简单,能在 trace 指定时间内将 I/O 请求发送出去,且发送时间误差为 0。

在负载方面,存储性能理事会^[5](Storage Performance Council, SPC)根据工业标准开发了 SPC-1^[5]和 SPC-2^[5]。SPC-1 用于商务应用的存储系统的测试,其负载主要由数据随机访问组成,包括大量的查询、更新等在线事务处理(on-line transaction processing, OLTP)系统的操作;SPC-2 则以数据顺序访问为主,主要用于具有大量顺序读写操作的商务应用存储系统的测试。事务处理性能委员会^[1](Transaction Processing Performance Council, TPC)系列基准程序主要用于测试系统进行事务处理的能力,其中 TPC-C^[9]模拟复杂的数据库事务交易操作,用每分钟交易次数(transactions per-minute, tpm)作为指标来评价存储系统性能。SPC^[5]和 TPC^[1]均是真实的负载,根据用途不同,IOmark 支持 SPC 和 TPC-C 负载,并兼容其它类型的 I/O 负载。

已有的存储系统性能测试工具大都采用多线程和同步 I/O 的处理方式来实现,如 Buttress^[8]、IOmeter^[6]、filebench^[10]、IOzone^[3]和 Trace^[11]等,除此之外还有单线程和异步 I/O 的处理方式可以实现,Postmark^[2]是一个单线程的基准测试程序,其主要目的是测量文件系统在大量短生命周期的小文件访问负载下的性能,主要用于 email、NetNews、电子商务等应用的存储系统测试,主要缺点是由于不能自由配置导致其可扩展性较差。

同步 I/O 操作会引起线程的阻塞,所以使用同步 I/O 处理并发操作需要发起多个线程,异步 I/O 操作不会引起线程阻塞,磁盘读写操作完毕接收结果即可。Linux 中的异步 I/O 有两种实现方式,一种是用 glibc 实现的用户态异步 I/O,另一种是由 linux 内核实现的内核态异步 I/O。对于用户态异步 I/O,底层的 I/O 调度器每次只能处理一个 I/O 请求,I/O 请求较多时效率较低。而内核态异步 I/O,可以将

所有的 I/O 请求都提交给底层内核的 I/O 调度器,当 I/O 请求较多时优势较大。

本文提出的测试工具 IOmark 使用单线程和内核态异步 I/O 实现,实验结果表明,与其它两种 I/O 处理方式实现的测试工具相比,IOmark 引入的测试误差最小,对存储系统的性能评测更准确。

2 IOmark 的设计

2.1 系统架构

IOmark 根据来自真实存储系统的 I/O 记录生成相同的 I/O 请求负载,以模拟运行环境的真实负载变化,进而达到对存储系统进行性能测试的目的。如图 1 所示,IOmark 由 5 个功能器件组成,包括 I/O 记录读取器、I/O 请求控制器、I/O 发射器、结果收集器和结果分析器。

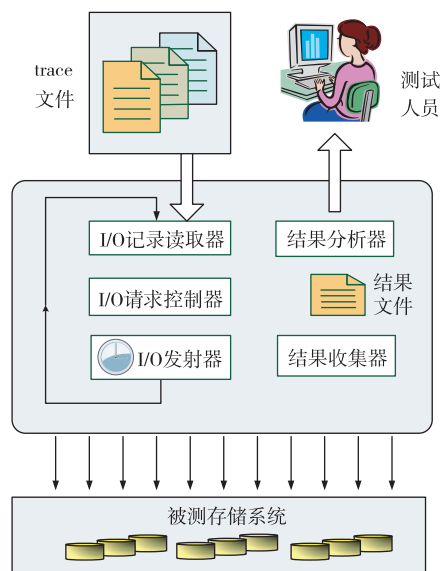


图 1 IOmark 系统架构

(1) I/O 记录读取器负责将 trace 文件中的 I/O 记录读取出来,每条 I/O 记录的内容一般包括盘号、开始字节、访问的字节总数、请求开始时间。

(2) I/O 请求控制器主要有两个功能,即控制负载的轻重和跟踪 I/O 记录的状态。

(3) I/O 发射器控制 I/O 记录的发射时间,并将 I/O 请求以异步的方式提交出去,交给系统去处理。

(4) 结果收集器在播放 I/O 记录过程中,查看 I/O 读写的状态,如果 I/O 读写完成,则将性能测试的结果数据收集到结果文件中。

(5) 结果分析器在测试结束后,按照测试人员

的要求将结果文件中的数据进行统计和分析,采用累计分布函数(cumulative distribution function, CDF)、分段求均值等统计方法,获得方便评价存储系统性能的数据,避免数据量过大造成性能评价结果不直观以及存在信息噪声等问题。

2.2 I/O 请求控制策略

2.2.1 负载轻重控制

向存储系统播放 I/O 记录需要根据该系统承受负载的能力来选取合适的负载,以免负载过轻无法体现系统的优势,或者过重导致系统崩溃;为了测量系统的最大负载承受能力,有时需要进行过载测试,即加重负载。可见控制负载轻重对于存储系统的性能测试是至关重要的^[12]。

IOmark 通过筛选盘号来控制负载的轻重。从 trace 文件中读取的每条 I/O 记录的目标设备是用盘号标记的,而盘号的分布是不均匀的。IOmark 通过选择带有某个或某些盘号的 I/O 记录进行播放来控制负载的轻重。

2.2.2 记录 I/O 请求状态

IOmark 使用循环队列记录 I/O 请求的状态信息。该队列不遵守先进先出规则,用来存储一些基本信息。比如,在 I/O 请求发射之后记录发射时间,用于 I/O 完成后统计 I/O 延迟时间。在播放 I/O 记录进行测试之前,IOmark 为循环队列预分配内存空间。在整个测试过程中,循环队列反复使用。当性能测试完成之后,再将申请的内存空间释放。

IOmark 使用占用标识来表示队列项是否正被使用。当一个 I/O 操作开始之前,在循环队列中循环查找一个空闲的数组项,并将该项的占用标识置 1。当某个 I/O 操作完成后,从 bio 结构体的数据项找到对应的数组项,并将该数组项的占用标识置 0。

2.3 I/O 请求发射机制

2.3.1 发射时间的控制

理论上,向系统发送 I/O 请求的发送时间应该为 I/O 记录中的请求时间。但是,当 trace 文件中某一时间段的 I/O 记录数量巨大时,发送 I/O 请求的速度会大于系统处理 I/O 请求的速度,使得请求时间不一定恰好为发送时间。当请求时间大于当前时间时,线程需要睡眠一段时间,直至当前时间增长到请求时间;相反,当请求时间小于当前时间时,则产生了延迟,IOmark 立刻将该 I/O 请求发射出去。

2.3.2 I/O 请求的发射方式

I/O 记录读取器读取负载信息,通过 I/O 请求控制器进行相应的控制。最后是由 I/O 发射器负责

将 I/O 记录发射出去。本文采用内核态异步 I/O 的访问方式来处理 I/O 请求,将 I/O 请求提交之后,不需要等待 I/O 读写操作完成。这样,I/O 记录读取器、I/O 请求控制器和 I/O 发射器不断循环执行,不停地向系统提交 I/O 请求。I/O 请求发射之后,linux 内核的底层通过一系列的系统调用,实现对磁盘的读写操作。

3 IOmark 的特点

IOmark 具有以下 4 个特点。

(1) **精确性**。理想情况下,向系统发送 I/O 请求的发送时间应该为 I/O 记录中的请求时间,但是这种理想情况是较难实现的,因为 trace 文件的负载瞬时较重等原因,很难保证每个 I/O 请求都能按时发送出去。IOmark 能在规定时间内将 I/O 请求发送出去,发送时间误差为 0,即可以精确地重放 I/O 记录,接近理想情况。

(2) **真实性**。IOmark 根据来自真实存储系统的 I/O 记录文件,在适当时间点发射 I/O 请求从而生成 I/O 负载,保证了负载的真实性。块级别存储设备以 O_DIRECT 方式打开,所以 I/O 请求不通过缓存,而直接对存储系统进行 I/O 读写,这就确保了 I/O 读写的真实性。可见,IOmark 从读取 I/O 记录到进行实际读写操作的整个过程,保证了存储系统性能测试的真实性和可信性。

(3) **测量误差小**。IOmark 采用 Linux 内核态异步 I/O 读写方式,可以在 trace 文件指定时间内将 I/O 请求提交给内核底层的 I/O 调度器。实验结果表明,与同类测试工具相比,IOmark 具有平均延迟时间小、延迟时间波动幅度小等特征,表明 IOmark 引入测量误差最小^[8]。IOmark 可以更加真实地重现用户的读写请求,对存储系统性能的评测更准确,更具实用价值。

(4) **负载轻重可调节**。IOmark 通过筛选 trace 文件中的磁盘号来控制负载的轻重,使得存储系统的性能测试更具灵活性。IOmark 不仅可以按照存储系统的实际情况选取适合系统性能指标的负载进行播放,还可以对系统的最大负载承受能力进行测试,使其应用范围更加广泛。

4 实验评价

本文通过在典型负载下对比 IOmark 与另外两

种测试工具,定量刻画了 IOmark 的精确性高和引入测量误差小等性能特征。

4.1 评价方法

本文评价方法使用 trace 文件生成有代表性的 I/O 负载,对典型的块级别存储系统进行性能测试,并通过对比 IOmark 与另外两种测试工具的表现来展示 IOmark 的性能特征。用来对比的两种测试工具 UserAIO 和 MT 均采用常见的性能测试工具实现方法。其中 UserAIO 采用单线程用户态异步 I/O 读写方式、MT 采用多线程同步 I/O 读写方式实现。

实验选取 Financial 和 TPC-C 两个 trace 文件来生成负载。Financial 来自存储性能理事会 (Storage Performance Council, SPC)^[13,14],一个商家中立的标准组织。它是从运行在一个大型金融机构的 OLTP 应用程序上收集的,该 trace 的写比率较高。TPC-C 收集了有 20 个数据仓库的 TPC-C 数据库评测工具的磁盘访问^[15],每个客户端运行 20 个循环。

使用 3 个工具分别对 3 个块级别存储系统进行测试。3 个存储系统分别为单盘系统、RAID0 系统和 RAID5 系统。其中,RAID0 系统由 2 块硬盘创建,RAID5 系统由 3 块硬盘创建。每块盘的容量均

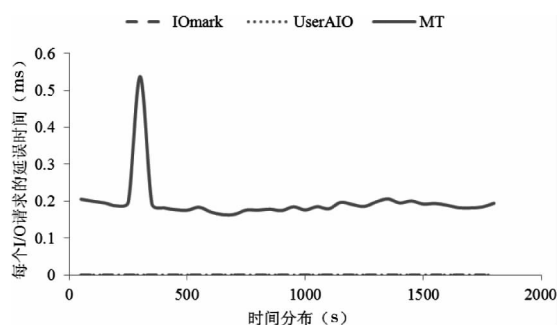
为 500G。对 RAID0 和 RAID5 系统的测试,是在 RAID 阵列创建结束之后进行的。

测试过程中记录的内容包括每个 I/O 请求的响应时间和发送延迟时间,发送延迟时间是指实际发送 I/O 请求的时间相对于 trace 文件中给定的发送请求时间的差值。所有实验数据都记录在相应的结果文件中。最后,由结果分析器对结果文件中的数据进行统计,得到最终的方便比较的结果数据。

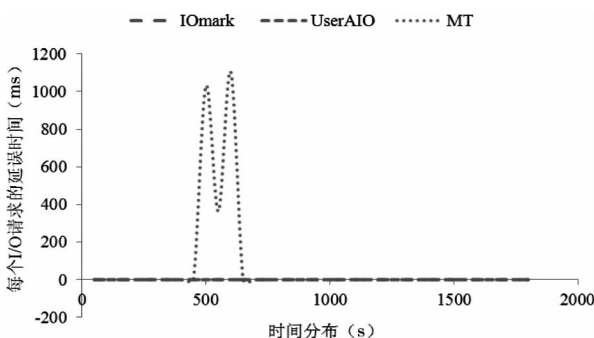
本文采用的实验平台信息描述如下:操作系统为 Ubuntu 10.04,内核版本为 2.6.32.9, GCC 版本为 4.4.3。实验中采用的机器的配置为 Intel Xeon 5606 2.13GHz 四核处理器和 8GB 内存,文件系统为 EXT4,并通过 6GB/s 的 SATA 扩展卡,将 12 块希捷 ST500DM002 的 SATA 硬盘连接到这台机器上。

4.2 精确性测试

对 IOmark 进行精确性测试,需要测试其延迟时间的大小。延迟时间越小,且发送 I/O 请求的时间越接近系统开始处理该请求的时间,则精确性越高。图 2 - 图 4 列出了 IOmark 与其它两种测试工具在 3 种系统环境下,分别播放 Financial 和 TPC-C 两种负载的延迟时间分布。从图中可以看出,在所有测试

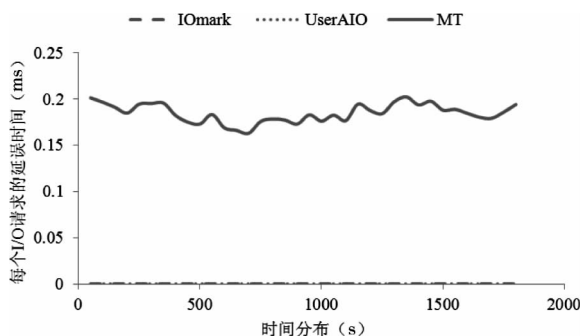


(a) 播放 Financial 负载

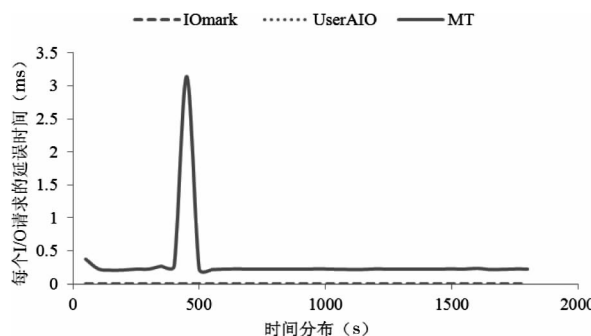


(b) 播放 TPC-C 负载

图 2 单盘系统下播放 Financial 和 TPC-C 负载的延迟时间分布比较

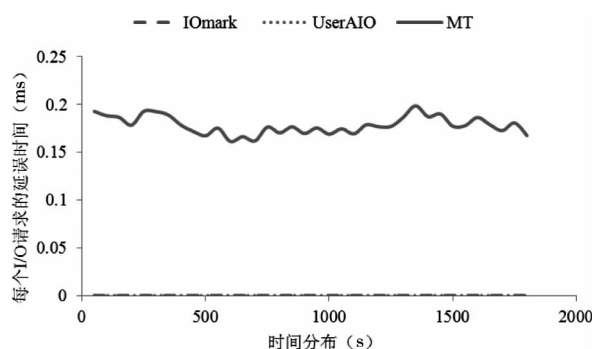


(a) 播放 Financial 负载

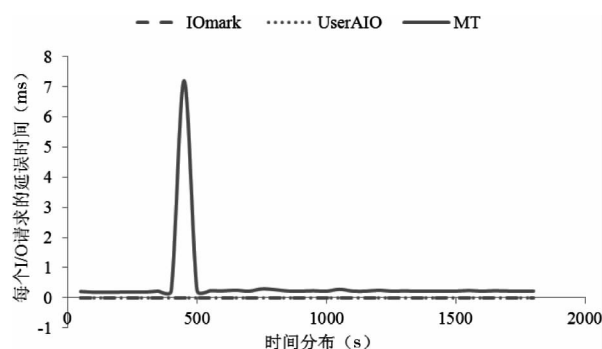


(b) 播放 TPC-C 负载

图 3 raid0 系统下播放 Financial 和 TPC-C 负载的延迟时间分布比较



(a) 播放 Financial 负载



(b) 播放 TPC-C 负载

图4 raid5 系统下播放 Financial 和 TPC-C 负载的延迟时间分布比较

情况下,IOmark 与 UserAIO 的延迟时间恒为 0,而 MT 的延迟时间较大,且在播放 TPC-C 负载情况下延迟时间波动幅度较大。

UserAIO 与 IOmark 的延迟时间相同,并不代表二者的精确性相同,还需要考虑发送 I/O 请求的时间是否为系统开始处理该请求的时间。因为 UserAIO 采用用户态异步 I/O 的读写方式实现,底层的 I/O 调度器每次只能处理一个 I/O 请求,所以 I/O 请求发送出去之后,需要排队等待,直至一个 I/O 请求处理完毕,I/O 调度器才会处理下一个请求。因此,UserAIO 虽然在负载指定时间将 I/O 请求发送出去,却无法保证将该 I/O 请求立即提交给系统进行处理。IOmark 采用内核态异步 I/O 读写方式,I/O 请求发送出去之后,立即提交给 I/O 调度器进行处理,所以与 UserAIO 相比,IOmark 的精确性更高。

MT 采用多线程、同步 I/O 读写方式,I/O 请求发送出去即通过调用函数对磁盘进行读写操作。MT 与 IOmark 相同,发送 I/O 请求的时间即为系统开始处理该请求的时间,但是 IOmark 延迟时间恒为 0,而 MT 的延迟时间较大,所以 IOmark 的精确性更高。综合考虑,与其它两种工具相比,IOmark 的精

确性最高,即 IOmark 可以精确地发送 I/O 请求,且发送误差为 0。

4.3 误差测试

不精确地播放 trace 文件会给存储系统带来测量误差,主要表现在平均响应时间大小和响应时间曲线波动幅度等性能指标方面。平均响应时间越大、响应时间曲线波动幅度越大,则引入的测量误差越大^[8]。

图 5 和图 6 分别列出了在三种系统环境下,IOmark 与其它两种测试工具播放 Financial 和 TPC-C 两种负载的平均响应时间分布情况,可以看出,在所有测试情况下,与其它两种测试工具相比,IOmark 的平均响应时间最小。

图 7 - 图 9 给出了 IOmark 与其它两种测试工具在 3 种系统环境下,分别播放 Financial 和 TPC-C 两种负载的响应时间分布。可以看到,IOmark 的曲线波动幅度最小,而其它两种测试工具的曲线波动幅度明显偏大,且曲线峰值很大,例如图 7(b)中 MT 的峰值甚至超过了 0.2s。

综合考虑平均响应时间和响应时间曲线波动幅度,可以发现:与其它两种测试工具相比,IOmark 平

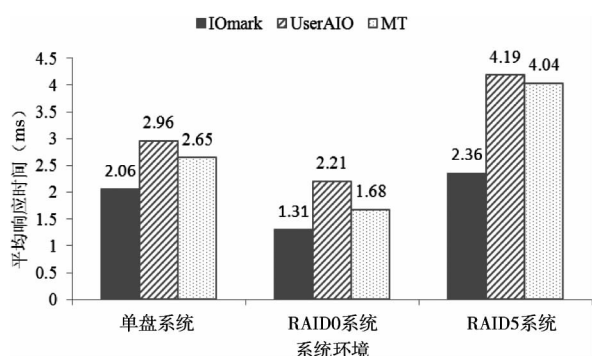


图5 播放 Financial 负载的平均响应时间

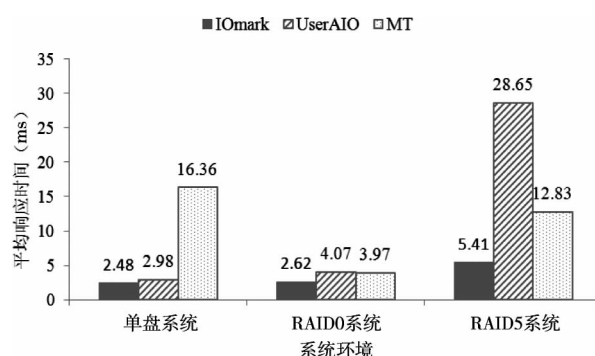


图6 播放 TPC-C 负载的平均响应时间

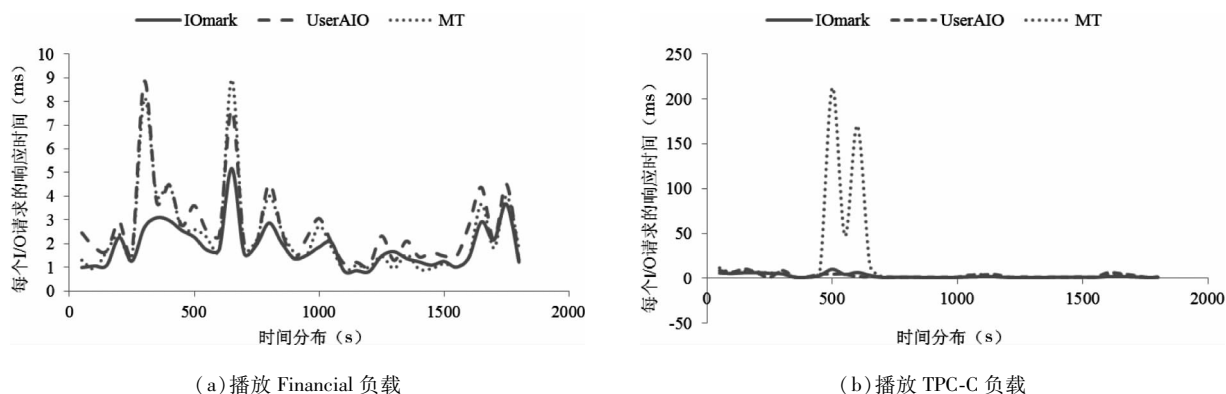


图 7 单盘系统下播放 Financial 和 TPC-C 负载的响应时间分布比较

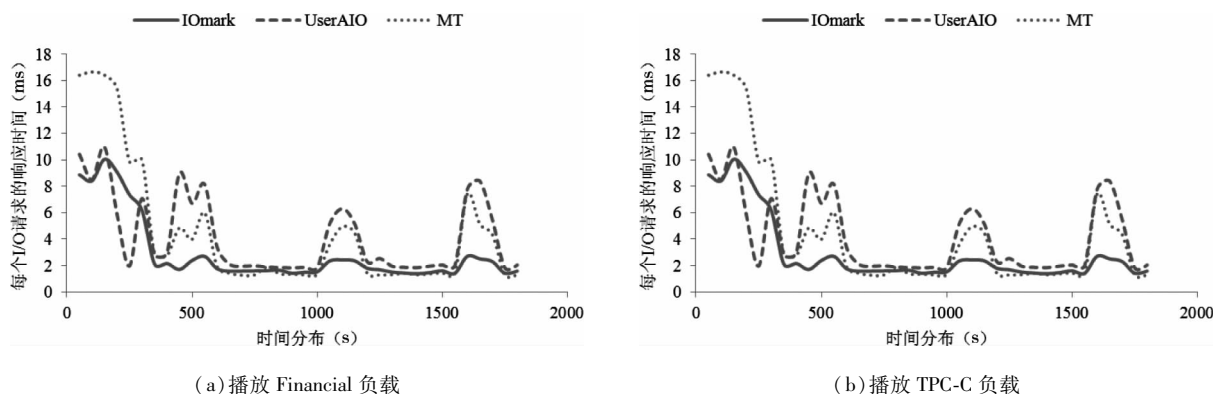


图 8 raid0 系统下播放 Financial 和 TPC-C 负载的响应时间分布比较

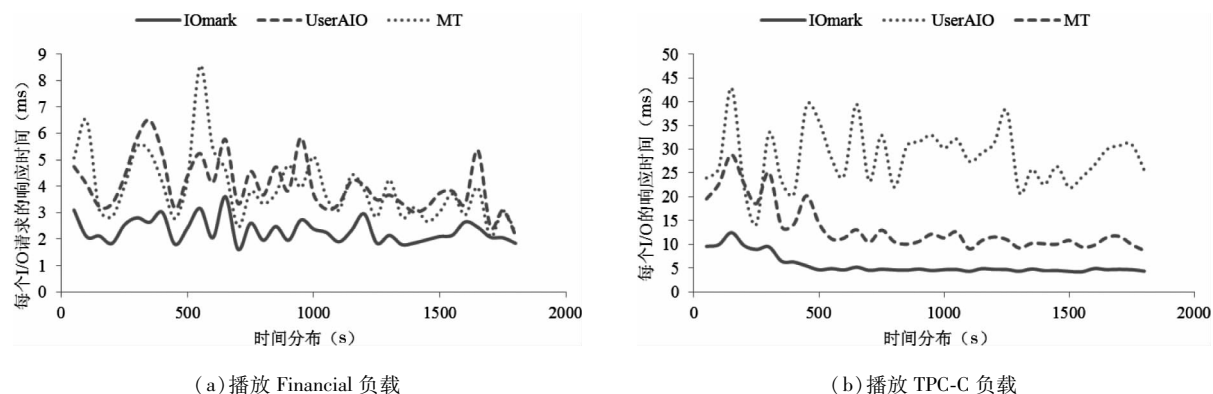


图 9 raid5 系统下播放 Financial 和 TPC-C 负载的响应时间分布

均响应时间最小,响应时间曲线波动幅度最小,即引入的测量误差最小,对存储系统的性能评测结果最准确。

5 结 论

通过播放 I/O 记录的方式进行块级别存储系统性能测试,兼具运行实际应用的真实程度高和生成人工合成负载的实现难度低的双重优势。但是,已

有的 I/O 记录重放工具缺乏精确地播放 I/O 负载的能力,容易给存储系统性能指标测量值带来显著误差。本文提出了一个精确的存储系统性能测试工具 IOmark,它使用 Linux 内核态异步 I/O 实现对块级别存储系统的访问;通过控制 I/O 请求的发送时间以及负载的轻重,模拟运行环境的真实负载变化,进而达到对存储系统性能进行测试的目的。

实验结果表明,IOmark 能精确地将 I/O 请求发送出去,且发送时间误差为 0;与同类测试工具相

比,引入的测试误差最小,对存储系统的性能评测更准确,实用价值更高。

参考文献

- [1] TPC. Transaction processing performance council. www.tpc.org;2005
- [2] Katcher J. Postmark: a new file system benchmark: [Technical Report], TR-3022, Network Appliance, Oct 1997
- [3] Oracle Corp. Willaim D. Norcott. Iozone file system benchmark white paper. Available from <http://www.iozone.org>
- [4] Bray T. Bonnie benchmark. <http://www.textuality.com/bonnie>;1988
- [5] Storage Performance Council (SPC). Storage performance council. <http://www.storageperformance.org>;2007
- [6] Standard Performance Evaluation Corporation. SPEC SFS97 v3. 0. <http://www.storageperformance.org>; December 1997
- [7] OSDL. Iometer project. <http://www.iometer.org/>;2004
- [8] Anderson E, Kallahalla M, Uysal M, et al. Buttress: A tool-kit for flexible and high fidelity I/O benchmarking. In: Proceedings of the 3rd Conference on File and Storage Technologies. USENIX Association, 2004. 45-58
- [9] Transaction Processing Council. TPC Benchmark C Standard Specification, Revision 5.2. Technical Report, 1992
- [10] McDougall R. A new methodology for characterizing file system performance using a hybrid of analytic models and a synthetic benchmark. In: Work in Progress Session in 3rd Usenix Conference on File and Storage Technologies, 2004
- [11] Mesnier MP, Wachs M, Sambasivan RR, et al. Trace: Parallel trace replay with approximate causal events. In: Proceedings of the 5th USENIX Conference on File and Storage Technologies. USENIX Association, 2007. 153-167
- [12] Sivathanu S, Kim J, Kulkarni D et al. Load-Aware Replay of I/O Traces (WIP and Poster). In: Proceedings of the 9th USENIX Conference on File and Storage Technologies, Feb 2011
- [13] OLTP Application I/O and Search Engine I/O. UMass Trace Repository. <http://traces.cs.umass.edu/index.php/Storage/Storage>; June 2007
- [14] Storage Performance Council. <http://www.storageperformance.org/home>. last accessed on December, 2010
- [15] TPC-C. Postgres. 20 iterations. DTB v1. 1. Performance Evaluation Laboratory, Brigham Young University. Trace distribution center. <http://tds.cs.byu.edu/tds/>. last accessed on December, 2010

IOmark: a high-accuracy performance testing tool for storage systems

Song Lili*, Zhang Guangyan**,***, Cai Tao*, Zheng Weimin**

(* School of Information Science and Technology, Jiangsu University, Zhenjiang 212013)

(** Department of Computer Science and Technology, Tsinghua University, Beijing 100084)

(*** Key Laboratory of Symbol Computation and Knowledge Engineering (Jilin University),
Ministry of Education, Changchun 130012)

Abstract

In consideration of the problem that existing trace replaying tools for testing the performance of storage systems are unable to reproduce I/O workloads simply and accurately and can easily result in obvious deviation in storage performance observation, the IOmark, a new tool for testing the performance of a block-level storage system, is proposed. The IOmark replays I/O records from the real storage system, and accesses a block-level storage system by using the asynchronous I/O of the Linux kernel. By tracking the generated I/O request and the request time in an I/O record, the transmission time of I/O requests is controlled and the disk number in I/O records is filtered so as to control the load intensity, thereby replaying I/O traces accurately. The experimental results demonstrate that the IOmark can issue I/O requests within the time specified by trace, and there is no issuing time error. Compared with other similar testing tools, the test errors caused by the IOmark are small. In other words, the IOmark can evaluate a storage system's performance more accurately.

Key words: block-level, storage system, performance testing tool, I/O trace, accuracy, measurement error