Benchmark 性能测试综述

王 良
(中国人民大学计算机科学与技术系,北京 100872)

摘 要 基准(Benchmark)测试是一种应用广泛、内容繁杂的测试技术,也是目前最主要的信息系统性能测试技术。文章对 Benchmark 测试的规范和测试方法做了归纳总结,给出了选用 Benchmark 测试的建议和开发 Benchmark 测试规范需要解决的问题。最后介绍了有代表性的 Benchmark 测试规范和程序集。

关键词 基准测试 测试 性能

文章编号 1002-8331-(2006)15-0045-04 **文献标识码** A **中图分类号** TP311

Summa of Benchmark Performance Test

Wang Liang
(Department of Computer Science and Technology, Renmin University of China, Beijing 100872)

Abstract: As a testing technology widely applied and included multifarious contents, a benchmark testing is primary importance for evaluating performance of information systems. This paper summarizes specifications and testing methods of benchmark testing, and puts forward for selecting it from a number of benchmark tests. Finally, typical benchmark specifications and testing software tools are introduced.

Keywords: benchmark test, test, performance

1 引言

Benchmark 作为一种评价方式,在计算机领域有着长期的应用。Benchmark,一般译成基准或标杆,按牛津百科全书的解释,Benchmark 是指测试人员在岩石、混凝土立柱等上面刻下的标记,用以测量相对高度等,也称(供比较参照之用)样板或参照点。Benchmark 测试的着眼点是测试结果的可比性,即按照统一的测试规范(test specification)对被测试系统进行测试,测试结果之间具有可比性,并可再现测试结果。

1.1 应用领域

Benchmark 测试在计算机领域中最广泛和最成功的应用是性能测试,主要测试响应时间、传输速率和吞吐量等。此外,它也用于功能、可操作性和数据处理开发易用性等方面的测试[1]。按照 Benchmark 的思想,它还可以有更广泛的用途,但目前性能以外有影响的 Benchmark 测试很少。

Benchmark 测试有些偏重于硬件,有些偏重于软件,还有些注重整个系统。在硬件方面广泛应用于评价 CPU、内存、I/O 接口和外围设备的性能,主要测试两个方面性能指标:一是硬件传输数据的带宽,称为带宽基准测试(Bandwidth benchmark);二是数据传输的延迟,称为延迟基准测试(Latency benchmark)。在软件方面,它用于评价操作系统、数据库和中间件以及应用软件的数据处理能力。

1.2 作用

Benchmark 测试对生产商和用户都很有价值。对生产厂商的作用是为产品进行市场宣传和发现系统的瓶颈;对用户的作用是指导产品的选择。Benchmark 测试最具吸引力的特点就是一个好的 Benchmark 测试对于某一领域的技术发展有积极的导向作用,它会引导生产厂商采用新技术改进产品。

选择 Benchmark 测试时需要有明确的目的,当用于产品宣传时,就应该选用权威机构的 Benchmark 测试,并且结果得到其认可。而用于指导产品选择的 Benchmark 测试,则需要清楚 Benchmark 测试的结果是否与应用的特性有密切的关系。Benchmark 只能模拟一定的应用环境,不可能适用所有情况。

Benchmark 测试也会带来消极的影响,如生产厂商可能会相互攀比测试结果,一味地追求高指标,而忽略了实际应用的需要。因此,好的 Benchmark 测试就是引导厂家和用户向正确的方面努力。

2 选用 Benchmark 测试规范

2.1 规范来源

Benchmark 测试发展了 20 多年,至今仍方兴未艾,许多组织和个人从事这方面的研究和开发。众多的 Benchmark 规范或测试程序集为了不同的目的、产生于不同的背景,归纳起来有如下形式:

(1)权威组织制定的测试标准或开发的测试程序集。它们的测试标准、测试程序和测试参数及测试报告都是公开的,如 SPEC、Linpark 和 TPC 组织的 TPC 系列等。

(2)媒体机构开展的测试。一般由媒体机构建立测试实验室,组织测试。测试对象一般是大众电子类产品,测试规范和测试程序不一定由测试机构开发,测试结果发布在媒体的专栏上,如 PC Magazine 采用 Futuremark 3DMark05 测试图像和声音处理的性能,采用 NetIQ's Chariot 测试 VoIP 的性能[2]。

(3)研究机构以研究为目的开发的测试规范或测试程序。

基金项目:国家 863 高技术研究发展计划资助项目(编号:2003AA4Z3030);国家教育部 211 工程资助项目
作者简介:王良(1963-),男,副教授,主要研究领域为软件工程,数据库。

这种测试结果不会由权威机构予以确认,测试程序可以免费得到。如 Wisconsin 大学开发的非常有影响的 Wisconsin Benchmark; 先由 Sun 后来是 SGI 支持的 LMbench;IOzone 组织开发的文件系统测试程序集 IOzone Filesystem Benchmark。

(4)开源测试项目开发的测试规范和测试程序。如开源的 OSDB(Open Source Database Benchmark)[3]。

(5)专业咨询公司开发的测试规范和测试程序。这类测试以盈利为目的,如 Doculabs Web 服务基准测试 @Bench[4]。

(6)生产厂家自行开发的测试规范和测试程序。如 NC&AC 的磁盘 Benchmark 测试系列工具集。

尽管有众多的 Benchmark 测试规范和程序集,但目前还没有官方标准化组织发布的用于计算机系统的 Benchmark 测试标准,生产厂商联合成立的机构和科研机构是 Benchmark 的最主要发源地。在使用 Benchmark 测试时,应根据测试的目的,搞清楚测试规范的权威性。

2.2 测试分类

Benchmark 测试根据被测试对象的不同可分为两类:组件测试和系统测试。组件测试是指测试的重点是针对信息系统中的某一部件或某一子系统,如 CPU、内存、磁盘、总线、文件系统、网络设备等。系统测试则是对整个计算机系统或信息系统进行测试。在系统测试中,由于关注点不同,使用的 Benchmark 测试规范就不同,则测试作用和度量指标也不同。不论是哪种 Benchmark 测试,都必须在一个完整的计算机系统上进行,因此,整个系统中的所有部分都可能对 Benchmark 测试结果产生影响,特别是硬件的配置水平、操作系统、编译器和数据库管理系统。

使用不同的 Benchmark 测试规范评价同一个被测试系统时,可能出现不一致的测试结果。在进行同类系统比较时,可能出现差异较大的结果。造成这一现象的原因复杂,但主要原因是所有的 Benchmark 测试规范有各自的侧重点,揭示出的系统瓶颈存在差异。

2.3 测试成本

Benchmark 测试有简单与复杂之分,所要付出的代价有大有小,需要注意的是大型 Benchmark 测试成本非常高。Benchmark 测试的成本构成如下:

(1)测试程序开发成本。它包括编写测试代码和测试数据。测试代码的开发需要对测试问题有深入的理解,并找到有效的测试方法。

(2)硬件资源。所有的 Benchmark 测试都需要计算机系统,甚至还要包括网络设备或一些大容量存储设备,如磁盘阵列、大规模机群系统和网络系统等。

(3)测试管理和维护成本。

(4)测试执行成本。Benchmark 测试是一个循序渐进的过程(可能延续数个月),需要优化系统、优化测试程序。这部分的成本可概括为调优成本、学习成本、加载数据成本以及测试系统运行成本等。

(5)测试系统占用成本。是指被测试系统因测试而不能用于实际工作的成本。

3 Benchmark 测试的共性特征

3.1 规范的主要内容

Benchmark 测试规范众多,复杂程度差别很大,但是作为一种被广泛应用的测试思想,它们通常表现出许多共性。Benchmark 测试规范一般应具有的特性如下:

(1)有一个公开的测试规范。这个规范一般包括测试目的、测试模型描述、测试环境配置要求、度量指标定义和测试量方法、测试结果发布方式。

(2)一般提供可执行程序或源程序,测试程序又可分两部分:一部分是测试数据装载程序,或者直接提供测试数据;另一部是测试执行程序,为被测试系统提供测试负载。也有些 Benchmark 测试规范不提供测试程序,如 TPC-C 和 TPC-W 等,这就需要测试者自行开发测试程序。

(3)提供度量指标的测量方法或计算方法的详细说明。度量指标要求在不同的被测试系统间具有可比性,如性能/价格比。度量指标可能很简单,如数据传输率;也可能很复杂,要求多个指标同时满足规范的要求,如 TPC-C 中的指标 tpmC 是每秒钟新事务的数量,但同时还要求按比例提交 5 种类型其它事务,每 10 个 New-Order 事务要伴随有 10 个 Payment 事务、1 个 Delivery 事务、1 个 Order-Status 事务和 1 个 Tack-Level 事务,即 New_Order 占总事务数的 43.7%。

(4)测试结果可重现。即在相同的测试环境下,可重现测试结果。这一点很重要,Benchmark 测试的精髓就是提供一个可比较的结果。

(5)测试结果公开。一个 Benchmark 测试结果是否公开取决于测试目的,公开程度一般应达到按公开的测试方法可再现测试结果。

3.2 规范必须解决的问题

(1)目的:所有的 Benchmark 测试都必须有明确的目的性,它应能够回答"为什么设计这个 Benchmark 测试?"、"它用于测试什么?"这样的问题。

(2)度量指标:测试的结果是通过度量指标来表示的,一般 Benchmark 测试有一个主要的指标和若干个辅助指标,辅助指标用于约束主指标的测量过程。

(3)负载:Benchmark 测试负载可归纳为三种,即处理大量数据、做高强度计算和传输大量数据。

(4)约束:包括对被测系统的优化约束、负载的配比及测试量指标之间的关系等。

(5)测试方法:或称测试程序的使用方法,一般包括步骤、测试持续时间和指标测量方法。

(6)测试结果发布形式:Benchmark 测试公开报告有严格的要求,对需要公开的内容有具体而明确的规定,一般要求其他人按公开报告的测试方法可再现测试结果。如 TPC-C 测试的公开报告(Full Disclose Report)包括:处理器数目、操作系统、Cache 大小、内存容量、磁盘控制器类型、磁盘容量等等。

4 常用 Benchmark 测试介绍

Benchmark 测试规范和测试程序集非常多,以下只选取部分有代表性的、应用较广泛的 Benchmark 测试进行介绍。

4.1 TPC 测试集[5]

TPC(Transaction Processing Performance Council)在 1988 年 8 月由 Omri Serlin 和 Tom Sawyer 创建,最初有 8 个成员,目前发展为 24 个,其中包括国际知名厂商 bea、HP、IBM、Intel、

Microsoft 和 Fujitsu 等。从 1989 年发布了第一个 Benchmark 标准至今,TPC 总共发布了 8 个标准,它们是 TPC-A、TPC-B、TPC-C、TPC-D、TPC-R、TPC-H、TPC-W 和 TPC-App,其中 TPC-A、TPC-B、TPC-D 和 TPC-R 标准已经被 TPC 组织宣布淘汰。最新发布的标准是 2004 年 12 月 5 日的 TPC-App。TPC 测试从 DBMS 的 ACID、查询时间和联机事务处理能力等方面对 DBMS 进行性能测试。

TPC 所制定的 Benchmark 测试标准可以从 http://www.tpc.org 网站下载。TPC 组织提供详细测试指导和测试结果的通过标准,在 TPC-R、TPC-H、TPC-W 和 TPC-App 标准中提供装载数据的代码,但不提供测试程序。TPC 测试的报告要求完全公开,包括测试的源代码。

4.2 SPEC[6]

SPEC(Standard Performance Evaluation Corporation)是一家非盈利公司,致力于建立、维护和认可与高性能计算机 Benchmark 标准。SPEC 开发了 7 个方面的 Benchmark 测试集,发布来自于成员单位和授权单位的测试结果。

SPEC 在美国加州注册,由 45 家成员、31 家伙伴和 2 家支持成员构成,囊括了欧美和日本的主要计算机厂商,如 IBM、Microsoft、Oracle、HP、Fujitsu、NEC、Dell、Hitachi、Intel、Sun 等。SPEC 结果是具有权威性的测试结果。SPEC 的测试集包括 CPU、图形/应用处理、高性能计算机/消息传递接口(MPI)、Java 客户机/服务器、邮件服务器、网络文件系统、Web 服务器等。

4.3 LINPACK[7]

LINPACK(Linear system Package)是在高性能计算机领域中最具影响的 Benchmark 测试,它使用线性代数方程组,利用选主元高斯消去法按双精度(64 位)算法测量求解线性方程的稠密系统所需的时间。LINPACK 的结果按每秒浮点运算次数(Flops)表示。LINPACK 源于 1974 年 4 月美国 Argonne 国家实验室,该实验室的应用数学所主任 Jim Pool 提出 LINPACK 计划,并得到美国 NSF 的支持,LINPACK 计划由 Jack Dongarra 主持实施。Jack Dongarra 教授不定期地发布报告《使用标准线性方程软件的各种计算机性能》。

LINPACK 在 HPC(High Performance Computer)领域的测试最具权威性,TOP500 就是采用 LINPACK 测试结果进行性能排序。在 TOP500 的网页 http://www.top500.org/lists/linpack.php 上可以找到它选择 LINPACK 的理由是"它被广泛应用和性能数值几乎对所有领域都有效"。为适应计算机系统体系的发展,LINPACK 中又发展出两个项目,即 LAPACK(Linear Algebra PACKage)和 EISPACK,这样可以更好地运行在共享内存的向量超级计算机上。

LINPACK 由一组 Fortran 程序组成,测试分为三种情况:

(1)使用 LINPACK 标准程序,处理 100×100 矩阵,不允许对程序做任何修改。

(2)使用 LINPACK 标准程序,允许修改测试算法,追求尽可能高的性能。

(3)针对大规模并行计算系统的测试。

4.4 IDC 平衡评价指标[8]

IDC 平衡分级(Balanced Rating)指标是由 IDC 公司与圣迭哥超级计算机中心联合为 HPC 测试而提出的指标体系,与其它许多 Benchmark 测试的区别是:它不使用峰值指标作为评价被测系统性能的指标,而是试图用 4 个独立的分级表来更好地满足 HPC 用户的个性需求。

IDC Balanced Rating 用于测试三个领域的性能:

(1)处理器性能;

(2)内存系统的能力;

(3)可伸缩能力。

IDC 平衡分级指标由 4 个分级列表组成,即 1 个综合列表和 3 个分别对应处理器、内存和可伸缩性(scaling)的分级列表。每个列表从 0 到 100 进行分级,100 是最好得分。

4.5 NPB[9]

NPB(Nasa Parallel Benchmark)是由 NASA 开发的一个用于评价并行超级计算机性能的小型程序集,又称为 NAS(Numerical Aerodynamic Simulation)Benchmark 测试。这个 Benchmark 测试来源于流体力学计算应用。NAS 发布了 5 个 NPB 规范,它们是 NPB1、NPB2、NPB3、GridNPB3 和 NPB3Multi-zone versions:

(1)NPB1:这是 NPB 最基础的版本,由 5 个内核和 3 个模拟应用程序组成,以 NASA Ames 研究中心的研究为基础,模拟大规模计算和数据传输的流体动力学(CFD,Computational Fluid Dynamics)应用。

(2)NPB2:基于 MPI(Message Passing Interface),共有 4 个版本,即 NPB-MPI 2.0/2.2/2.4 和 2.4 I/O,较 NPB1 有三个突出变化:一是为提高可移植性,采用 Fortran-77 开发测试程序集;二是只实现 NPB1 中的 5 个测试程序;三是在已有"class A"和"class B"的基础上,增加了"class C",从而扩大了测试规模。其中 NPB-MPI 2.4 I/O 利用 BT(Block-Tri_diagonal)问题测试高性能计算系统的输出能力。

(3)NPB3:为适应 HPC 的体系结构变化又分为 3 个子版本,NPB-OpenMP3.0 针对 ccNUMA(cache coherent Non-Uniform Memory Access)体系结构,NPB-Java 3.0 是用 Java 语言实现的 NPB3.0,而 NPB-High Performance Fortran 3.0 是用 HPF(High Performance Fortran)实现的 NPB3.0。

(4)GridNPB3:是 NPB3 为适应网格(Grid)而开发的测试程序集,又称为 NGB(NAS Grid Benchmark),目前只有 2002 发布的 NGB1.0。

(5)NPB3 Multi-zone versions:即 NPB3.0-MZ,是 NPB 的最新版本,于 2003 年发布,目的是解决细粒度、混合型、多级并行计算机系统的测试问题,它是 NPB 的扩展。

4.6 HPPC[10]

HPCC(HPC Challenge)是由美国 DARPA(Defense Advanced Research Projects Agency)、NSF 和 DoD 通过 DARPA HPCS(High Productivity Computing Systems)计划资助的项目,目的是为了有助于定义未来 Petascale 规模超级计算机系统的性能范围。HPCC 测试程序集由 7 个著名计算内核(STREAM、HPL、矩阵乘-DGEMM、并行矩阵转置-PTRANS、FFT、Random-Access、带宽和延迟测试-b_eff)组成,设计的目的是测试真实应用的性能,如内存访问、时空局部性等。

(1)HPL(High Performance LINPACK)是 LINPACK 的 TPP(Toward Peak Performance)的变种版本,通过求解线性方程组

测试系统的浮点计算能力。

(2)STREAM 测试系统的内存持续访问带宽和响应计算的速度。

(3)RandomAccess 测量内存随机修改速度。

(4)PTRANS 测量多处理器系统的内存中大数据量数组的传输率。

(5)DGEMM 通过执行双精度实数矩阵乘法,测量浮点数的执行速度。

(6)FFT 通过执行一维双精度离散傅立叶变换,测量浮点运算的速度。

(7)通讯带宽和延迟测试是基于同时通讯的 b_eff(effective bandwidth benchmark)。

4.7 IOzone[11]

IOzone 由 Oracle 的 William D.Norcott 发起,之后 HP 公司的 Don Capps 和 Tom McNeal 对其进行了完善。它是一个可运行在 Linux、HP-UX、Solaris 和 Windows 系统上的文件系统 Benchmark 测试工具。IOzone 的开发目的是分析计算机平台生产厂商的文件系统的 I/O 性能,为用户选择系统提供参考。IOzone 将文件系统的 I/O 作为基本负载对文件系统进行 Benchmark 测试,允许测试者调整参数,包括从很小到非常大的文件和不同的访问方式。IOzone 可测试本地系统,也可测试客户机/服务器环境下的 NFS(Network File System)客户端文件访问。

IOzone 的测试项包括:读/写、重复读/写、后向读、跨越读、流文件读/写系统库函数(fread/fwrite)、随机读/写、偏移量读/写库函数(pread/pwrite)、POSIX 异步读/写和文件映射内存的系统库函数 mmap。

4.8 LMbench[12]

Lmbench 是由 SGI 公司的 Larry McVoy 和 HP 公司的 Carl Staelin 设计开发的一组小型 Benchmark 测试程序集,它可以测试处理器、内存、网络、文件系统和磁盘中的数据传输和数据带宽。LMbench 的作者希望能够在广泛的应用领域中发现被测试系统的性能瓶颈,并能识别、隔离和再现这些瓶颈。

LMbench 由许多小测试程序组成,每个测试程序能够捕获应用中的某些特定性能问题。它遵守 GPL 许可协议,可获得源代码。它可以运行在 AIX、BSDI、HP-UX、IRIX、Linux、FreeBSD、NetBSD、OSF/1、Solaris 和 SunOS 系统上。

LMbench 集中在测试带宽、延时和这两者的组合问题上:

(1)带宽基准测试,可细分为:被缓冲的文件读,利用系统调用 bcopy 的内存复制、内存读、内存写、管道(Pipe)和 TCP 传输。

(2)延时基准测试,可细分为:进程上下文切换、组网(其中包括:建立连接、管道、TPC、UDP 和 RPC)、文件系统的建立和删除、进程创建、信号操作、系统调用代价和内存读延时。

(3)杂项,只有处理器时钟速度测试一项。

5 结语

本文在对大量 Benchmark 测试进行研究分析的基础上,对 Benchmark 测试的规范和一般测试方法做了归纳总结。Benchmark 测试在信息系统的性能评价方面最为成熟、应用最为广泛,其它方面的研究难度较大,至今缺少有影响的 Benchmark 测试规范或测试程序。大型系统的 Benchmark 测试非常复杂,技术难度很大,成本较高。

随着信息技术的发展,Benchmark 测试的应用领域也在不断地拓展。除性能测试外,Benchmark 测试在计算机安全性[13]、可伸缩性、可靠性[14]和故障恢复能力[15]等领域的研究也在不断地深入。

致谢 在此,向对本文的工作给予支持和建议的同行,尤其是北京人大金仓信息技术有限公司表示衷心的感谢。

(收稿日期:2006 年 1 月)

参考文献

1.Jim Gray,Omri Serlin,Carrie Ballinger at el.The Benchmark Handbook.http://http://www.informatik.uni-trier.de/~ley/db/books/collections/gray91.html,2004-02

2.Wireless Testing.PC Magazine.http://www.pcmag.com/category2/0,4148,1938,00.asp,2005-09

3.D Bitton,C Turbyfill.Open Source Database Benchmark Project at Compaq Computer Corporation.http://osdb.sourceforge.net/index.php?page=faq,2005-10

4.http://www.doculabs.com/Downloads/WebServicePerformance_04-03.pdf,2005-09

5.Omri Serlin,Tom Sawyer.TPC Benchmarks.http://www.tpc.org/information/benchmarks.asp,2005-10

6.Kaivalya Dixit,Tom Skornia.Standard Performance Evaluation Corporation(SPEC).http://www.spec.org/osg/web99/,2005-10

7.Jack Dongarra,Jim Bunch,Cleve Moler et al.Netlib Index for LINPACK.http://www.netlib.org/linpack,2005-10

8.BENCHMARKS.http://www.hpcuserforum.com/benchmark/,2005-10

9.Rupak Biswas.http://www.nas.nasa.gov/Software/NPB,2005-09

10.Piotr Luszczek,Jack J Dongarra,David Koester et al.Introduction to the HPC Challenge Benchmark.http://icl.cs.utk.edu/hpcc/Suite,hpcc-challenge-benchmark.pdf,2005-03

11.William D Norcott,Don Capps.IOzone Filesystem Benchmark.http://www.iozone.org,2005-09

12.Larry McVoy,Carl Staelin.LMbench-Tools for Performance Analysis.http://www.bitmover.com/lmbench/,2005-11

13.Securis Inc.Security Benchmark.com.http://www.securitybenchmark.com/,2005-10

14.M Dal Cin.Dependability Benchmarking.http://www.esat.kuleuven.ac.be/electa/dbench,2003-09

15.Dave Patterson,Armando Fox.The Berkeley/Stanford Recovery-Oriented Computing(ROC) Project.http://roc.cs.berkeley.edu/,2005-11