第1章 计算机设计基础

评价性能

- 比较机器性能的指标
  - 执行时间(响应时间，时延)：桌面机、服务器
  - 吞吐量：网站服务器
- 用程序集比较机器性能
  - 选择适当的程序评估性能
    - 基准测试程序套件（Benchmark Suites）
  - 不同平均值（Different Means）：Arithmetic, and Geometric Means

性能

- 性能与执行时间*互为倒数*

  最大的性能意味着最小的执行时间

Relative Performance

- Define Performance = 1/Execution Time
- "X is *n* time faster than Y"

$$\text{Performance}_X / \text{Performance}_Y$$
$$= \text{Execution time}_Y / \text{Execution time}_X = n$$

- Example: time taken to run a program
  - 10s on A, 15s on B
  - Execution Time_B / Execution Time_A = 15s / 10s = 1.5
  - So A is 1.5 times faster than B

性能指标—响应时间

- 墙钟时间
  - 程序开始执行到结束看钟知道的时间，就是墙钟时间，也称为响应时间或消逝时间
  - 测量用户感觉到的系统速度
- 墙钟时间的问题
  - 如果一个机器上运行多个程序 ？
  - 如果程序运行时需要用户输入？

性能指标--CPU 时间

- 测量CPU时间更具可计算性 (not waiting for I/O)
  - 测量*设计者感觉到的*CPU速度
- CPU时间进一步分为：
  - 用户CPU时间- 花费在用户模式的时间
  - 系统CPU时间- 花费在OS的时间
- Unix时间命令报告CPU时间：
  - 90.7 user CPU seconds (in the user's program)
  - 12.9 system CPU seconds (in the system calls e.g. printf)
  - 2 minutes, 39 seconds wall-clock time（159秒，103.6秒CPU）
  - 65% of the wall clock time was spent running on the CPU

性能指标----吞吐量（服务器）

- 单位时间内完成的工作总量---吞吐量
  - 测量管理员感觉到的系统性能
- 常用吞吐量测量
  - 每秒处理的事务数量，如每秒服务的网页数量
- 相比前一个时延指标
  - 等待接受网页的时间量

- 单核处理器性能一般使用时延指标
- 对于很多应用，吞吐量比时延更重要：

  金融市场，政府统计（人口普查）

响应时间与吞吐量（服务器）

- 通常改善了响应时间也会改善吞吐量
  - 处理器用更快的型号替换
- 假设任务不可分解
  - 增加额外的处理器，如用多处理器，只改善吞吐量而不改善响应时间

嵌入式系统：功耗

- 嵌入式系统的关键因素：
  - 功耗
  - 成本
  - 物理尺寸
  - 存储空间
- 例如
  - AMD ElanSC520
  - AMD K6-2E
  - IBM PowerPC 750CX
  - NEC VR 4122
  - NEC VR 5432

NEC VR 4122 的性能/瓦最好，但是它是其中性能倒数第2低的处理器。

常常误导的指标: MIPS

- MIPS ：Millions of Instructions per Second
- 用相同的指令集比较两台机器（A，B），MIPS一般是公平的
- MIPS可能是一个“无意义的性能指标”
- Pitfall: MIPS as a Performance Metric

例子: MIPS 指标常常无意义的

- 机器A有一条计算平方根的特殊指令，它执行需要100个时钟周期（设每个时钟周期1us）
- 机器B没有这种指令 ---它计算平方根用软件方式即用加、乘、移位简单指令（一般执行需要1个时钟周期，设时钟周期1us）编程实现
- 机器 A: $(1/100)*10^6$条指令/s = 0.01 MIPS
- 机器 B: $1*10^6$条指令/s = 1 MIPS
- 机器B不一定比A好

Check Yourself

- Consider the following performance measurements for a program:

Measurement	Computer A	Computer B
Instruction count	10 billion	8 billion
Clock rate	4 GHz	4 GHz
CPI	1.0	1.1

a. Which computer has the higher MIPS rating?

b. Which computer is faster?

Answer (Part 1)

Measurement	Computer A	Computer B
Instruction count	10 billion	8 billion
Clock rate	4 GHz	4 GHz
CPI	1.0	1.1

a. Which computer has the higher MIPS rating?

- **MIPS = Clock rate / (CPI $\times$ 10^6)**
- **MIPS(A) = 4 $\times$ 10^9/ (1 $\times$ 10^6) = 4000**
- **MIPS(B) = 4 $\times$ 10^9/ (1.1 $\times$ 10^6) = 3636**
- **Computer A has the higher MIPS rating.**

Answer (Part 2)

Measurement	Computer A	Computer B
Instruction count	10 billion	8 billion
Clock rate	4 GHz	4 GHz
CPI	1.0	1.1

a. Which computer has the higher MIPS rating?

b. Which computer is faster?

- **Execution Time = Instruction count $\times$CPI / Clock rate**
- **Execution time A = 10 $\times 10^9 \times$ 1.0 / (4 $\times$ 10^9) = 2.5 (seconds)**
- **Execution time B = 8 $\times 10^9 \times$ 1.1 / (4 $\times$ 10^9) = 2.2 (seconds)**
- **Computer B is faster**

比较两个机器

Machine	CPI	Clock Period	Avg Instruction Time (secs)
Machine A	1.2	2 ns	?
Machine B	2.5	1 ns	?

- 哪个机器更快?
- 快多少?

比较两个机器

Machine	CPI	Clock Period	Avg Instruction Time (secs)
Machine A	1.2	2 ns	1.2*2=2.4ns
Machine B	2.5	1 ns	2.5ns

- CPU Time = 执行指令条数 * avg instruction time
- Assume 1,000,000, 000 instructions
- Machine A: 1,000,000,000 * 2.4ns = 2.4 seconds
- Machine B: 1,000,000,000 * 2.5ns = 2.5 seconds
- 哪个机器更快？ Machine A
- 快多少？ 2.5 / 2.4 = 1.04 times faster

比较性能

- 为什么要比较不同机器或者不同程序的性能?
  - 要帮助设计者知道哪一个更好
  - 要给销售在新闻发布时一个有力的依据
  - 要帮助消费者选择满足自己需求的机器
- 性能与执行时间*互为倒数*

  最大的性能意味着最小的执行时间

常用短语

- “P_1 性能比P_2 更好”：对给定工作负载程序 L， P_1 执行L的时间比 P_2 更少

performance(P1) > Performance(P2)

⇒ Execution Time(P1, L) < Execution Time(P2, L)

- “处理器 X 比Y快n 倍”：

$$n = \frac{\text{Execution time } Y}{\text{Execution time } X}$$

通过多个程序比较机器性能

	Computer A	Computer B	Computer C
Program 1 (secs)	1	10	20
Program 2 (secs)	1000	100	20
Program 3 (secs)	1001	110	40

- A is 10 times faster than B for program 1
- B is 10 times faster than A for program 2
- A is 20 times faster than C for program 1
- C is 50 times faster than A for program 2
- B is 2 times faster than C for program 1
- C is 5 times faster than B for program 2

❖ 上述句子都是正确的，但是哪一个机器是最好的？

算术和调和平均值
（Arithmetic and Harmonic Mean）

- **总的执行时间：一个一致的总的度量**
  - **算术平均时间：**是*n*个测试程序总的执行时间的算术平均值。

$$\frac{1}{n}\sum_{i=1}^{n} Time_i$$

算术平均时间的问题

- 每个应用程序运行的概率并不相同
- 更长执行时间的程序在平均值中的份量更重
- 例如，两个机器执行两个基准测试程序的时间

	Machine A	Machine B
Program 1	2 seconds (20%)	4 seconds (20%)
Program 2	12 seconds (80%)	8 seconds (80%)

（Program 2运行的概率是 Program 1 的4倍）

- 计算算术平均值， Program 2 占的比重比Program 1更大
  - 显然，改善Program 2对平均值的影响大于Program 1

加权执行时间

- 通常，机器运行某些程序更频繁，则应该给这些程序更大的权值（weight）

$$\sum_{i=1}^{n} Weight_i * Time_i$$

使用加权和 (或加权平均)

	Machine A	Machine B
Program 1	2 seconds (20%)	4 seconds (20%)
Program 2	12 seconds (80%）	8 seconds (80%)
Total	10 seconds	7.2 seconds

$2 \times 20\% + 12 \times 80\% = 10$ $4 \times 20\% + 8 \times 80\% = 7.2$

计算相对性能：10/7.2 = 1.38

--> Machine B is 1.38 times faster than Machine A

但是，加权和存在一个重要问题：

- SPEC 不同厂商对应用程序权重的选取有冲突

加权平均性能比较

两个程序在A、B、C三台机器上的执行时间

	机器A	机器B	机器C	W（1）	W（2）	W（3）
程序1	1.00	10.00	20.00	0.50	0.909	0.999
程序2	1000.00	10.00	20.00	0.50	0.091	0.001
加权算术平均值A_m（1）	500.50	10.00	20.00			
加权算术平均值A_m（2）	91.91	10.00	20.00			
加权算术平均值A_m（3）	2.00	10.00	20.00			

选择程序评估机器性能

- 理想的性能评估：
  - 运行随机取样的用户的程序和OS命令
- 不同类型的基准测试程序（benchmarks）

不同类型的基准测试程序（benchmarks）

- 核心测试程序：从实际的程序中抽取少量较短的关键程序框架代码构成，这些代码的执行直接影响程序总的执行时间。如Linpack。
- 小测试程序：代码在10~100行，具有特定目的的测试程序。如Quicksort。
- 综合测试程序：对一大套应用程序中的操作和操作数的执行频率进行统计，得到平均执行频率，再按这个频率编制的模拟测试程序。如Dhrystone。
- <u>基准测试程序集：</u>选择一组有代表性的不同类型应用程序，集中起来构成基准测试程序集，以有效评测计算机处理各种应用的性能。这种测试程序集合也称为测试程序组件（benchmark suites）。如SPEC。

注意：基准测试程序的局限

- 基准测试程序可以针对一个系统的某些方面
  - Integer & floating point运算, memory system, I/O, OS
- 通用基准测试程序有时可能会误导，因为硬件和编译器的供应商或许会仅仅对这些基准程序优化他们的设计。
- 系统可能运行某些应用程序性能好，而另一些则性能差
- 编译利用结构的特点可以提高性能。应用程序特定的编译器优化已普遍采用。
- 最好的基准测试程序应该是用户实际使用的应用程序，因为它们反应了终端用户的需要。

SPEC—实用基准测试程序集

- SPEC - The System Performance Evaluation Cooperative一个开放性的非赢利组织
  - 1988年由工作站厂商HP，DEC，MIPS，SUN共同发起， 以满足市场迫切需要的标准化性能测试。
  - 成功的性能测试标准化组织，有几十个公司成员。
  - http://www.spec.org
- SPEC's Philosophy
  - 目标：保证市场有一套公平和实用的指标来区分不同的候选系统。
  - 基本方法：提供基于现有应用程序的一套标准化源代码作为基准测试程序集。

SPEC benchmarks
Desktop Benchmarks

- **CPU-intensive benchmarks**
  - SPEC89
  - SPEC92
  - SPEC95
  - SPEC2000
  - SPEC CPU2006 (12 CINT2006, 17 CFP2006)
- **graphics-intensive benchmarks**
  - SPEC2000
    - SPECviewperf
      - is used for benchmarking systems supporting the OpenGL graphics library
    - SPECapc
      - consists of applications that make extensive use of graphics.

Server Benchmarks

- **SPEC**
  - SPECrate—processing rate of a multiprocessor

    由SPEC CPU2000构建实现多个CPU基准测试程序副本
  - SPECSFS--file server benchmark
  - SPECWeb--Web server benchmark
- **TPC benchmark—Transaction Processing Council**
  - **TPC-A，1985**
  - **TPC-C，1992，**
  - **TPC-H→ TPC-R→TPC-W**

事务处理（**TP**）：数据库访问与更新。

典型**TP**系统：机票预订系统，银行**ATM**系统

评测指标：每秒钟处理的事务数。对响应时间也有要求。

Embedded Benchmarks

- EDN Embedded Microprocessor Benchmark Consortium (or EEMBC, pronounced “embassy”).

运行基准测试程序（ Benchmarks ）

- 关键：可重现性（Reproducibility ）
- 尽量多的细节

  列出实验时所有的假定和条件

  - 如：program input, version of the program, version of the compiler, optimization level, OS version, main memory size, disk types, etc.

- 系统软件的配置会有效地影响benchmark的性能结果。

SPEC 性能评价

SPEC采用比选择权值更好的方法，就是选择一个统一的参考计算机，给出各测试程序在参考机上的执行时间，然后将被测机器的执行时间与之比较来评价不同机器的性能。

SPEC的评价指标有2个：

SPEC率（SPEC Ratio）

SM（Spec Mark）：采用SPEC率的几何平均值

- SPEC率（SPEC Ratio）是一个测试程序在参考计算机上的执行时间与在被测计算机上的执行时间的比值，可以表示为：

$$SPECRatio_A = \frac{\text{参考计算机的执行时间}}{A\text{机的执行时间}}$$

显然，SPEC Ratio比值越高，说明被测计算机的性能越高。

例如，假设针对同一个基准测试程序A计算机的SPEC Ratio是B计算机的1.3倍，则可表示为：

$$\frac{SPECRatio_A}{SPECRatio_B} = \frac{\frac{\text{参考计算机的执行时间}}{A\text{机的执行时间}}}{\frac{\text{参考计算机的执行时间}}{B\text{机的执行时间}}} = \frac{B\text{机的执行时间}}{A\text{机的执行时间}} = \frac{A\text{机的性能}}{B\text{机的性能}} = 1.3$$

- SM（Spec Mark）是被测试计算机执行n个基准测试程序分别得到的SPEC率的几何平均值。若某被测计算机的SPEC率有n个数值，则该计算机的SPEC率的几何平均值的计算公式为：

$$SM = \sqrt[n]{\prod_{i=1}^{n} SPECRatio_i}$$

SM是一个综合指标以衡量不同计算机的性能。

但是为了完整的表示出系统的性能特征，通常也将n个基准程序的SPEC率列出，以分项比较分析。

算术平均与几何平均

- **算术平均值**
  - 算术平均值的权重与特定程序在特定机器上的执行时间成正比，所以权重不仅与该程序的执行频度有关，也与运行该程序的机器特性和输入数据相关。可能导致测试者把自己机器上运行最快的程序的输入量增到最大，以提高计算该程序平均值时的权重，干扰了真实结果
- **几何平均值**
  - 小于算术平均值、与参考机无关、强调性能平衡
  - 可能导致硬件和软件设计者集中精力提高最易提高速度的软件的性能，而不是提高速度最慢的软件的性能。

SPEC 95

处理机	SPECint’95	SPECfp’95
PentiumII 400	18.5	13.3
PentiumII 450	18.7	13.7
PentiumIII 500	20.6	14.7
PentiumIII 550	22.3	15.6
Celeron 300A	12.0	9.66
Celeron 333	13.1	10.20
Celeron 366	14.1	10.70
Celeron 400	15.1	11.20
Celeron 433	16.1	11.60
Celeron 466	17.0	12.00

Figure 1.17 Ultra 5、Itanium 2、Opteron的SPECfp2000执行时间和*SPECRatio*

基准 测试程序	Ultra 5 时间（秒）	Opteron 时间（秒）	Opteron *SPECRatio*	Itanium 2 时间（秒）	Itanium 2 *SPECRatio*	Itanium / Opteron *SPECRatio*
wupwise	1600	51.5	31.06	56.1	28.53	0.92
swim	3100	125.0	24.73	70.7	43.85	1.77
mgrid	1800	98.0	18.37	65.8	27.36	1.49
applu	2100	94.0	22.34	50.9	41.25	1.85
mesa	1400	64.6	21.67	108.0	12.99	0.60
galgel	2900	86.4	33.57	40.0	72.47	2.16
art	2600	92.4	28.13	21.0	123.67	4.40
equake	1300	72.6	17.92	36.3	35.78	2.00
facerec	1900	73.6	25.80	86.9	21.86	0.85
ammp	2200	136.0	16.14	132.0	16.63	1.03
lucas	2000	88.8	22.52	107.0	18.76	0.83
fma3d	2100	120.0	17.48	131.0	16.09	0.92
sixtrack	1100	123.0	8.95	68.8	15.99	1.79
apsi	2600	150.0	17.36	231.0	11.27	0.65
几何平均值SM			20.86		27.12	1.30

Discussion 1.6

- Figure 1.24 gives a comparison of power and performance for several benchmarks comparing two servers: Sun Fire T2000 (which uses Niagara) and IBM x346 (using Intel Xeon processors). This information was reported on a Sun Web site. There are two pieces of information reported: power and speed on two benchmarks. For the results shown, the Sun Fire T2000 is clearly superior. What other factors might be important and thus cause someone to choose the IBM x346 if it were superior in those areas?

	Sun Fire T2000	IBM x346
Power (watts)	298	438
SPECjbb (operations/sec)	63,378	39,985
Power (watts)	330	438
SPECWeb (composite)	14,001	4348

Figure 1.24 Sun power/performance comparison as selectively reported by Sun.

Discussion 1.6

- Figure 1.24 gives a comparison of power and performance for several benchmarks comparing two servers: Sun Fire T2000 (which uses Niagara) and IBM x346 (using Intel Xeon processors). This information was reported on a Sun Web site. There are two pieces of information reported: power and speed on two benchmarks. For the results shown, the Sun Fire T2000 is clearly superior. What other factors might be important and thus cause someone to choose the IBM x346 if it were superior in those areas?

	Sun Fire T2000	IBM x346
Power (watts)	298	438
SPECjbb (operations/sec)	63,378	39,985
Power (watts)	330	438
SPECWeb (composite)	14,001	4348

Figure 1.24 **Sun power/performance comparison as selectively reported by Sun.**

The IBM x346 could take less space, which would save money in real estate.

The racks might be better laid out. It could also be much cheaper.

In addition, if we were running applications that did not match the characteristics of these benchmarks, the IBM x346 might be faster.

Finally, there are no reliability numbers shown. Although we do not know that the IBM x346 is better in any of these areas, we do not know it is worse, either.

Discussion(1-11)(from COD_5e)

1.11 The results of the SPEC CPU2006 bzip2 benchmark running on an AMD Barcelona has an instruction count of 2.389E12, an execution time of 750 s, and a reference time of 9650 s.

Discussion(1-2)

1.11 The results of the SPEC CPU2006 bzip2 benchmark running on an AMD Barcelona has an instruction count of 2.389E12, an execution time of 750 s, and a reference time of 9650 s.

- **Find the CPI if the clock cycle time is 0.33333 ns.**
- **Find the SPECratio.**

1. CPI = clock rate $\times$ CPU time/instr. count

   clock rate = 1/cycle time = 3 GHz

   CPI(bzip2) = $3 \times 10^9 \times 750/(2389 \times 10^9)$= 0.9418
2. SPEC ratio = ref. time/execution time

   SPEC ratio(bzip2) = 9650/750 = 12.8667

Discussion(3)

1.11 The results of the SPEC CPU2006 bzip2 benchmark running on an AMD Barcelona has an instruction count of 2.389E12, an execution time of 750 s, and a reference time of 9650 s.

- Find the increase in CPU time if the number of instructions of the benchmark is increased by 10% without affecting the CPI and clock.

3. CPU time = No. instr. $\times$ CPI/clock rate

   If CPI and clock rate do not change, the CPU time increase is equal to the increase in the of number of instructions, that is 10%.

Discussion(4-5)

1.11 The results of the SPEC CPU2006 bzip2 benchmark running on an AMD Barcelona has an instruction count of 2.389E12, an execution time of 750 s, and a reference time of 9650 s.

- **Find the increase in CPU time if the number of instructions of the benchmark is increased by 10% and the CPI is increased by 5%.**
- **Find the change in the SPECratio for this change**

4. CPU time(before) = No. instr. × CPI/clock rate

   CPU time(after) = 1.1 × No. instr. × 1.05 × CPI/clock rate

   CPU time(after)/CPU time(before) = 1.1 × 1.05 =1.155. Thus, CPU time is increased by 15.5%.

5. SPECratio = reference time/CPU time

   SPECratio(after)/SPECratio(before) = CPU time(before)/CPU time(after) = 1/1.155 = 0.8658. The SPECratio is decreased by 13.42%.

Discussion(6)

1.11 The results of the SPEC CPU2006 bzip2 benchmark running on an AMD Barcelona has an instruction count of 2.389E12, an execution time of 750 s, and a reference time of 9650 s.

- **Suppose that we are developing a new version of the AMD Barcelona processor with a 4 GHz clock rate. We have added some additional instructions to the instruction set in such a way that the number of instructions has been reduced by 15%. The execution time is reduced to 700 s and the new SPECratio is 13.8. Find the new CPI.**

6. CPI = (CPU time $\times$ clock rate)/No. instr.

   CPI = $700 \times 4 \times 10^9/(0.85 \times 2389 \times 10^9) = 1.3789$

Discussion(7-8)

1.11 The results of the SPEC CPU2006 bzip2 benchmark running on an AMD Barcelona has an instruction count of 2.389E12, an execution time of 750 s, and a reference time of 9650 s.

- This CPI value is larger than obtained in 1.11.1 as the clock rate was increased from 3 GHz to 4 GHz. Determine whether the increase in the CPI is similar to that of the clock rate. If they are dissimilar, why?
- By how much has the CPU time been reduced?

7. Clock rate ratio = 4 GHz/3 GHz = 1.3333

   CPI @4 GHz = 1.3789, CPI @3 GHz = 0.9418, ratio = 1.4641

   CPU time @ 4 GHz = 700 s , CPU time @ 3GHz = 750 s

   They are different because, although the number of instructions has been reduced by 15%, the CPU time has been reduced by a lower percentage.

8. 700/750 = 0.9333. CPU time reduction: 6.67%

Discussion(9)

1.11 The results of the SPEC CPU2006 bzip2 benchmark running on an AMD Barcelona has an instruction count of 2.389E12, an execution time of 750 s, and a reference time of 9650 s.

- **For a second benchmark, libquantum, assume an execution time of 960 ns, CPI of 1.61, and clock rate of 3 GHz. If the execution time is reduced by an additional 10% without affecting to the CPI and with a clock rate of 4 GHz, determine the number of instructions.**

9. No. instr. = CPU time × clock rate/CPI

No. instr. = $960 \times 0.9 \times 4 \times 10^9 / 1.61 = 2147 \times 10^9$

Discussion(10-11)

1.11 The results of the SPEC CPU2006 bzip2 benchmark running on an AMD Barcelona has an instruction count of 2.389E12, an execution time of 750 s, and a reference time of 9650 s.

- Determine the clock rate required to give a further 10% reduction in CPU time while maintaining the number of instructions and with the CPI unchanged.
- Determine the clock rate if the CPI is reduced by 15% and the CPU time by 20% while the number of instructions is unchanged.

10. Clock rate = No. instr. × CPI/CPU time.

    Clock rate$_{new}$ = No. instr. × CPI/(0.9 × CPU time) = (1/0.9) × clock rate$_{old}$

    =3.3333 (GHz)

11. Clock rate = No. instr. × CPI/CPU time.

    Clock rate$_{new}$ = No. instr. × 0.85× CPI/(0.80 × CPU time)

    = (0.85/0.80) × Clock rate$_{old}$ = (0.85/0.80) × 3

    = 3.1875(GHz)

Classroom Test

- The ratio of the geometric means is equal to the geometric mean of the performance ratios, and that the reference computer of SPECRatio matters not.（中文：几何平均值之比等于性能比值的几何平均值，且SPECRatio基准计算机的选择无关紧要。）

  A. True
  B. False

- 根据下表填写算术平均值和几何平均值

Performance	Computer A	Computer B
Program 1	1	4
Program 2	100	100
算术平均	(1) ______	(2) ______
几何平均	(3) ______	(4) ______

Classroom Test 1

- The ratio of the geometric means is equal to the geometric mean of the performance ratios, and that the reference computer of SPECRatio matters not.（中文：几何平均值之比等于性能比值的几何平均值，且SPECRatio基准计算机的选择无关紧要。）

A. True
B. False

Classroom Test 2

- 根据下表填写算术平均值和几何平均值

Performance	Computer A	Computer B
Program 1	**1**	**4**
Program 2	**100**	**100**
算术平均	**(1)** ________	**(2)** ________
几何平均	**(3)** ________	**(4)** ________

性能指标总结

- 响应 (执行) 时间
  - 系统性能
  - 仅有的各方都认可的性能测量指标
- CPU 时间
  - CPU 性能
- 吞吐量