Computer Systems Performance Analysis and Benchmarking (37-235)

Analytic Modelling

Simulation

Measurements / Benchmarking

Lecture/Assignments/Projects:

Dipl. Inf. Ing. Christian Kurmann

Textbook:

Raj Jain, “The Art of Computer Systems Performance Analysis”, 1991 Wiley & Sons, New York

Topic of Today:

- **Popular Benchmarks**
- **Workload Selection**

Benchmarking and Benchmarks

;-)

to benchmark *v. trans.*
To subject (a system) to a series of tests in order to obtain prearranged results not available on competitive systems.

S. Kelly-Bootle
The Devil's DB Dictionnary

Benchmarking:

The process of performance comparison for two or more systems by measurement.

Benchmarks:

Workload used in the measurements.

Popular Benchmarks

Sieve of Erathostenes:

- Brain dead way to find primes
- Depends on:
  - Memory Systems Performance
  - Implementation Array/Bitset
  - Working Set in Memory

Ackermann Function:

- Silly recursive Funktion
- Product of Infinite Log Shaving by Theoreticians. Notion of log* ~ const. but not quite constant.
- Inverse of x=log*(y)

```
A(1,j)=2^j
A(i,1)=A(i-1,2)
A(i,j)=A(i-1,A(i,j-1))
```

see also: Cormen, Leiserson, Rivest: Algorithms

Popular Benchmarks

Dhrystone

- Reinhold Weicker at Siemens [1984]
- lots of procedure calls, integer perf.
- systems programming workload
- in C, Ada and Pascal
- Integer only: DIPS

Whetstone

- British central computer agency [1975], 11 modules designed after 949 Algol programs.
- Fixed and floating point arithmetic: KWIPS
- compiler sensitive

both will fit in caches...

Popular Benchmarks

LINPACK

- Jack Dongarra (1983), Argonne Nat.Lab
- Subroutines to solve dense linear systems.
- Most time consumed in BLAS (Basic Linear Algebra Subroutines)
- MFLOPs

  Popular for:
  - 100x100 system of equations
  - 1000x1000 system of equations

  Represent:
  - finite element analysis, simulation
  - calls for high computation speed
  - graphics processing
- Engineering/Scientific applic. perf.

Popular Benchmarks

Debit Credit

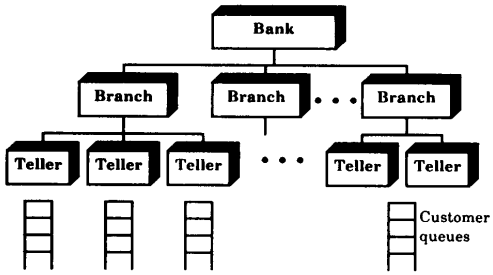


FIGURE 4.4 Banking environment.

Begin-Transaction	
Read message from the terminal	(100 bytes)
Rewrite account	(100 bytes, random)
Write history	(50 bytes, sequential)
Rewrite teller	(100 bytes, random)
Rewrite branch	(100 bytes, random)
Write message to the terminal	(200 bytes)
Commit-Transaction	

FIGURE 4.5 Debit-credit transaction pseudo-code.

TPC-A Benchmark, August 1988

(Transaction Processing Performance Council)

Popular Benchmarks

SPEC Suite 1992, 1995, 2000...

Programs submitted by scientists and engineers.

- GCC
- Espresso Electronic Design Automation
- Spice 2g6 Electronic Design Automation
- Doduc Monte Carlo Simulation Nuke
- NASA7 Matrix OPs submitted by NASA
- LI Do 9 queens on 8x8 board in LISP
- Eqntott Boolean Eqn to Truth Table
- Matrix300 LINPACK 300x300 doubles
- Fpppp Quantum Chemistry integral on double floats FORTRAN
- Tomcatv Vectorized Mesh Generator

Popular Benchmarks

- **CPU and FPU**
- Two copies run on each processor
- Compared to Reference System a VAX 11/780 Release 1.0 ~ 1 MIPS
- Ratio of each benchmarks
- Spec Througput considering the number of processors #cpu@ratio.
- Geometric Mean leads to SPECmark.
- New trend: Industry standard benchmarks

Workload Selection (More an Art than a Science)

Very important part of every performance evaluation project.

Service Exercised:

- System under Test (SUT)
  = Service Provider
  e.g. CPU, PC
- Component under Study (CUS)
  e.g. ALU, CPU

The system determines selection of workload & metric not the component.

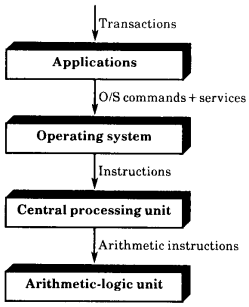
Banking system services

- transaction per second
- not instructions per second

CPU

- vice versa

Hierarchical View of a System

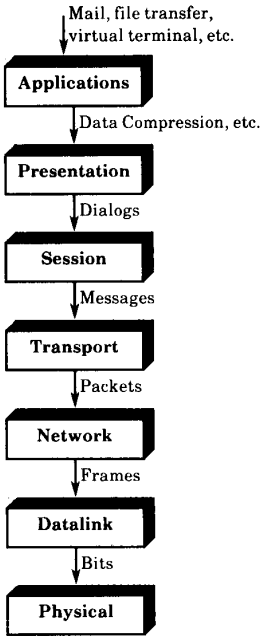


Hierarchy of interfaces at which requests on a time sharing system are serviced:

- 1. ALU: most frequent instruction
- 2. CPU: instruction mix, kernels
- 3. OS: synthetic programs with IO
- 4. Application: frequency of transactions

Determine the SUT level!

Example Network



Choice of Workload depends on layer at which networks are compared.

Example Backup System

Levels in a magnetic tape backup system:

- Backup System
- Tape Data System
- Tape Drives
- Read/Write Subsystem
- Read/Write Heads

What are the services, factors, metrics and workloads on the different levels?

Example Backup System

1. *Backup System*:
   (a) Services: Backup files, backup changed files, restore files, list backed-up files.
   (b) Factors: File system size, batch or background process, incremental or full backups.
   (c) Metrics: Backup time, restore time.
   (d) Workload: A computer system with files to be backed up. Vary frequency of backups.
2. *Tape Data System*:
   (a) Services: Read/write to the tape, read tape label, autoload tapes.
   (b) Factors: Type of tape drive.
   (c) Metrics: Speed, reliability, time between failures.
   (d) Workload: A synthetic program generating representative tape I/O requests.
3. *Tape Drives*:
   (a) Services: Read record, write record, rewind, find record, move to end of tape, move to beginning of tape.
   (b) Factors: Cartridge or reel tapes, drive size.
   (c) Metrics: Time for each type of service, for example, time to read record and to write record, speed (requests per unit time), noise, power dissipation.
   (d) Workload: A synthetic program exerciser generating various types of requests in a representative manner.
4. *Read/Write Subsystem*:
   (a) Services: Read data, write data (as digital signals).
   (b) Factors: Data-encoding technique, implementation technology (CMOS, TTL, and so forth).
   (c) Metrics: Coding density, I/O bandwidth (bits per second).
   (d) Workload: Read/write data streams with varying patterns of bits.
5. *Read/Write Heads*:
   (a) Services: Read signal, write signal (electrical signals).
   (b) Factors: Composition, interhead spacing, gap sizing, number of heads in parallel.
   (c) Metrics: Magnetic field strength, hysteresis.
   (d) Workload: Read/write currents of various amplitudes, tapes moving at various speeds. □

Level of Detail:

Level of detail in recording (and thus reproducing) the requests for a service:

- Most frequent request
- Frequency of request types
- Time-stamped sequence of requests (Trace)
- Average resource demand
- Distribution of resource demands

Workload descriptions

- executable WL: trace of user commands
- non-executable WL: modeling, simulation

Representativeness:

Test WL should be representative of the real application.

Three aspects that have to match:

- Arrival Rate: same or proportional
- Resource Demands: total demand
- Resource Usage Profile: e.g. important in multiprogramming environment

Timeliness:

WL should follow changes in usage pattern.

- demands and user behaviour change
- workloads have to change

Example: SPEC 89,92,95,2000...

Problem: caches get bigger and bigger

Other Considerations:

- Loading Level
  - best case: full capacity
  - typical case: real world
  - worst case: beyond capacity
- Impact of External Components
  - e.g. I/O in CPU benchmark
- Repeatability
  - reproduction of results is mandatory
  - when average, standard deviation