

IA039: Architektura superpočítačů a náročné výpočty

Profiling and Benchmarking

Luděk Matyska

Fakulta informatiky MU

Jaro 2015

Měření času výpočtu

- Optimalizace není možná bez znalosti, co optimalizovat
- Potřebujeme znát údaje o běhu programu
 - Čas výpočtu celého programu: příkaz **time**
 - Čas výpočtu částí programů: profilování
 - Srovnání systémů: benchmarking

Příkaz `time`

- User time
 - Čas procesoru strávený uživatelskými procesy
- System time
 - Čas procesoru strávený obsluhou funkcí jádra
- Elapsed time
 - Celkový čas výpočtu

$\text{CPU time} = \text{user time} + \text{system time}$

Příkaz **time** – další údaje

- Dostupné pro **time** v prostředí C shellu
- Sdílený paměťový prostor
- Privátní (unshared) paměťový prostor
- Počet block input operací
- Počet block output operací
- Počet page faultů
- Počet swapů

Profiling

viz http://www.site.uottawa.ca/~mbolic/elg6158/Subhasis_profiling.pdf

Zjišťování výkonu – benchmarking

- Snaha o porovnání *systémů*
 - Hardware i software společně
- Neexistuje žádné „zázračné“ řešení
- Základní přístupy
 - Průmyslové („profesionální“) benchmarky
 - Porovnatelnost, nezávislost na výrobcích
 - „Privátní“ benchmarky
 - Konkrétní (specifické) požadavky

Mysteriózní MIPS a MFLOPS

- Srovnání na základě počtu instrukcí vykonaných za sekundu
- MIPS – milion celočíselných instrukcí za sekundu
- MFLOPS – milion operací s pohyblivou řádovou čárkou za sekundu
- Problémy
 - Jaké instrukce?
 - V jaké posloupnosti?
- Umělé, nevypovídající

Celočíselné benchmarky

- VAX MIPS
- Dhrystones

Benchmarky s pohyblivou řádovou čárkou

- Whetstone (umělý mix, skalární)
- Linpack (daxpy, vektorizace)
 - 100*100
 - 1000*1000

SPEC benchmarks

- Nezávislá organizace
 - Standard Performance Evaluation Corporation
- Standardizované benchmarky pro různé architektury
- Vychází z tzv. *kernel kódů*
 - Celý nebo část existujícího programu
 - Dostupné ve zdrojovém kódu
 - Je možno „doladit“

- Open Systems Group (OSG)
- High Performance Group (HPG)
- Graphics Performance Characterization Group (GPC)

SPEC OSG podskupiny

- CPU
 - SPECmarks a CPU benchmarks
- JAVA
 - JVM98, JBB2005, Java client a server benchmarky
- MAIL
 - SPECmail2001
- SFS
 - Systémy souborů (SFS97)
- WEB
 - WEB99, WEB99_SSL, WEB2005

- Aktuální CPU benchmark
- Dělení
 - CINT2006 – celočíselné výpočty
 - CFP2006 – výpočty s pohyblivou řádovou čárkou

- Jednotlivé součásti

164.gzip	Compression
175.vpr	FPGA Circuit Placement and Routing
176.gcc	C Programming Language Compiler
181.mcf	Combinatorial Optimization
186.crafty	Game Playing: Chess
197.parser	Word Processing
252.eon	Computer Visualization
253.perlbnk	PERL Programming Language
254.gap	Group Theory, Interpreter
255.vortex	Object-oriented Database
256.bzip2	Compression
300.twolf	Place and Route Simulator

- Jednotlivé součásti

400.perlbench	C	PERL Programming Language
401.bzip2	C	Compression
403.gcc	C	C Compiler
429.mcf	C	Combinatorial Optimization
445.gobmk	C	Artificial Intelligence: go
456.hmmer	C	Search Gene Sequence
458.sjeng	C	Artificial Intelligence: chess
462.libquantum	C	Physics: Quantum Computing
464.h264ref	C	Video Compression
471.omnetpp	C++	Discrete Event Simulation
473.astar	C++	Path-finding Algorithms
483.xalancbmk	C++	XML Processing

- Jednotlivé součásti

168.wupwise	Physics / Quantum Chromodynamics
171.swim	Shallow Water Modeling
172.mgrid	Multi-grid Solver: 3D Potential Field
173.applu	Parabolic / Elliptic Partial Differential Equations
177.mesa	3-D Graphics Library
178.galgel	Computational Fluid Dynamics
179.art	Image Recognition / Neural Networks
183.quake	Seismic Wave Propagation Simulation
187.facerec	Image Processing: Face Recognition
188.amp	Computational Chemistry
189.lucas	Number Theory / Primality Testing
191.fma3d	Finite-element Crash Simulation
200.sixtrack	High Energy Nuclear Physics Accelerator Design
301.apsi	Meteorology: Pollutant Distribution

CFP2006

410.bwaves	Fortran	Fluid Dynamics
416.gamess	Fortran	Quantum Chemistry
433.milc	C	Physics: Quantum Chromodynamics
434.zeusmp	Fortran	Physics/CFD
435.gromacs	C/Fortran	Biochemistry/Molecular Dynamics
436.cactusADM	C/Fortran	Physics/General Relativity
437.leslie3d	Fortran	Fluid Dynamics
444.namd	C++	Biology/Molecular Dynamics
447.dealll	C++	Finite Element Analysis
450.soplex	C++	Linear Programming, Optimization
453.povray	C++	Image Ray-tracing
454.calculix	C/Fortran	Structural Mechanics
459.GemsFDTD	Fortran	Computational Electromagnetics
465.tonto	Fortran	Quantum Chemistry
470.lbm	C	Fluid Dynamics
481.wrf	C/Fortran	Weather Prediction

Transakční benchmarky

- Výkon databází
 - TPC-A
 - Testuje interakci s ATM (6 požadavků za minutu)
 - 1 TPS znamená, že 10 ATM současně vydá požadavek a výsledek dostane do 2 s (s 90 % účinností)
 - TPC-B
 - Jako TPC-A, ale testuje se přímo, ne přes (pomalou) síť
 - TPC-C
 - Komplexní, transakce jsou objednávky, platby, dotazy s jistým procentem záměrných chyb vyžadujících automatickou korekci

Síťové benchmarky

- netperf
- iperf
- End to end měření
- Pozor na to, co skutečně v síti měříme

Vlastní benchmarky

- Specifické (konkrétní) požadavky
- Důležité parametry:
 - Co testovat
 - Jak dlouho testovat
 - Požadavky na velikost paměti
- Typy benchmarků
 - Jednoduchý proud (opakování)
 - Propustnost (benchmark stone wall)

- Nezbytná součást používání benchmarků
 - Měříme opravdu to, co chceme?
- Možné příčiny ovlivnění
 - Použitá optimalizace
 - Velikost paměti
 - Přítomnost jiných procesů
- Co je třeba explicitně kontrolovat
 - CPU čas a čas nástěných hodin
 - Výsledky!
 - Srovnání se „známým“ standardem