# ISTITUTO NAZIONALE DI FISICA NUCLEARE

**Sezione di Padova**

**INFN/CCR-10/03**
**7 Settembre 2010**

**CCR-36/2009/P**


## HEP-SPEC06 - GUIDA ALL'USO


Alberto Crescente[1], Michele Michelotto[1]

*[1)]INFN – Sezione di Padova, Via F. Marzolo 8, Padova, Italia*



Abstract

Recentemente la comunità delle alte energie ha scelto un nuovo benchmark per misurare la potenza dei nodi di calcolo. HEP-SPEC06 è il nuovo benchmark di riferimento che sostituisce SPECINT 2000. Questa nota descrive i motivi che hanno portato alla sostituzione e spiega come va usata la nuova unità di misura.



*Published by **SIS–Pubblicazioni***
*Laboratori Nazionali di Frascati*

# 1 IL BENCHMARK SPEC CPU

SPEC[1], acronimo di Standard Performance Evaluation Corporation, è una organizzazione no-profit che mantiene e promuove un numero rilevante di benchmarks per computer ad alte prestazioni. SPEC CPU in particolare è la famiglia di benchmark pensata per misurare i carichi di lavoro cpu intensive su sistemi diversi.

Nel corso degli anni si sono avute diverse versioni del benchmark CPU: SPEC 95, SPEC 98 e SPEC 2000 e l'ultimo SPEC CPU 2006. Il benchmark misura il rapporto tra la velocità di esecuzione del benchmark della macchina che si sta misurando e quello di una macchina di riferimento (che vale quindi per definizione uno).

## 1.1 Interi e Floating Point

SPEC CPU è composto di due "suite", una per misurare le performance di applicazioni CPU intensive basate su calcoli con numeri interi (CPU INT) e l'altra per applicazioni in cui si fa più uso di operazioni in virgola mobile (CPU FP dove FP sta per Floating Point).

Ogni suite è composta di un certo numero di applicazioni che vengono eseguite in sequenza e alla fine si prende come valore del benchmark la media geometrica del risultato finale (dal momento che ci interessa il rapporto tra le velocità, inversamente proporzionali ai tempi di esecuzione).

## 1.2 Dalla misura di un core alla misura di una macchina multicpu

Il benchmark misura le prestazioni di applicazioni single-threaded. Se vogliamo conoscere il rating di una macchina con diverse CPU logiche (per esempio due o quattro cpu single core o anche una sola CPU multicore) dobbiamo girare diversi benchmark in parallelo.

L'approccio scelto da SPEC è di lanciare il benchmark in modalità RATE anziché in modalità SPEED.

Lo SPECINT RATE (lo stesso vale per lo SPECFP RATE) quindi misura il throughput complessivo della macchina composta da diverse CPU logiche (per esempio in macchine con diversi processori, con più core per processore o con diversi thread per core). Si lanciano quindi in parallelo i diversi benchmark e si misura il tempo di esecuzione nel momento in cui l'ultimo dei N processi paralleli termina.

## 1.3 La modalità HEP Multiple Speed

In ambiente HEP si è preferito invece misurare le prestazioni di throughput lanciando in parallelo diverse versioni del benchmark SPEED completo e prendere come valore di Throuphput della macchina la somma dei diversi valori individuali, senza sincronizzare i core alla fine di ogni benchmark. Questo approccio che viene chiamato MULTIPLE SPEED è stato scelto dal working group di Hepix e non è considerato uno dei possibili modi di esecuzione da SPEC.

---

[1] www.spec.org

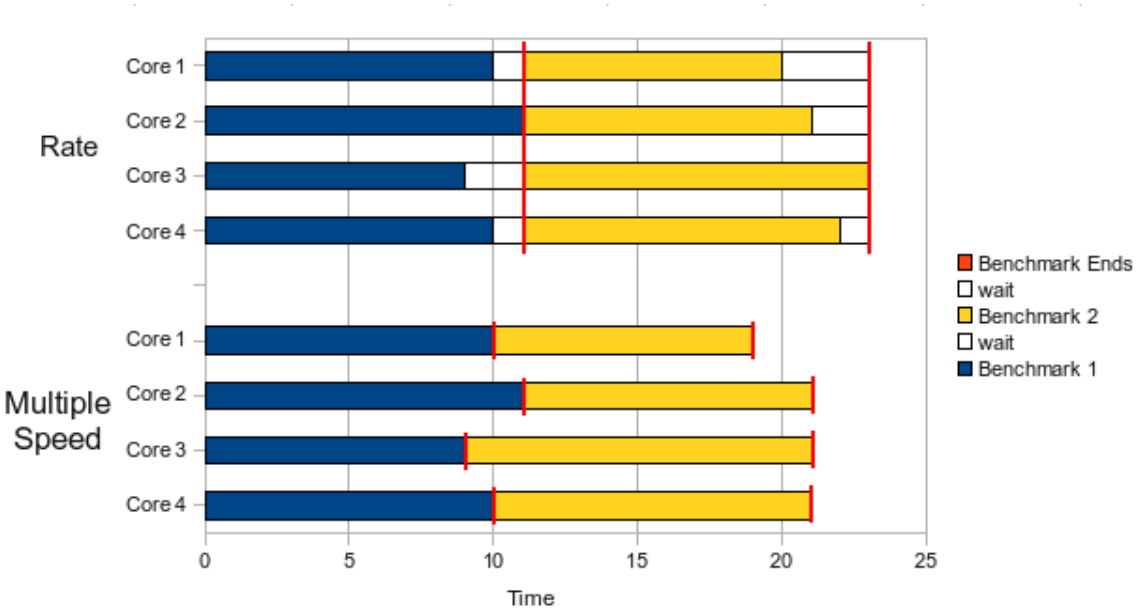


**Fig. 1**: La differenza tra una misura SPEED in parallelo (Multiple Speed) e una misura RATE

Nel caso RATE ogni volta è necessario aspettare che finiscano tutti i processi prima di dichiarare concluso un test e quindi per brevi periodi si hanno alcuni cores inattivi e appena prima del completamento dell'ultimo degli N cores si hanno N-1 cores inattivi.

Inoltre il tempo di esecuzione viene preso sull'ultimo core che si è concluso per cui anche agli altri core viene assegnato il tempo del più lento.

Questo approccio ha senso per misurare le prestazioni di una macchina che gira applicazioni multithreaded perché in effetti il risultato è disponibile solo nel momento in cui anche l'ultimo thread si è concluso.

Nel caso MULTIPLE SPEED si cerca di replicare la modalità batch dei nodi di calcolo HEP per cui non appena un core ha finito l'esecuzione, subito viene impegnato con un nuovo task. Alla fine si prende come risultato la SOMMA dei punteggi dei singoli JOB di tipo SPEED.

Per valutare una farm poi si sommano di nuovo i valori MULTIPLE SPEED delle singole macchine. In pratica si considera la farm come una somma dei cores di tutte le macchine.

Questo approccio è stato preferito perché in ambito HEP non importa avere un risultato nel tempo minimo ma importa il Throughput complessivo della farm, cioè il numero di eventi di fisica elaborati nell'unità di tempo.

## 2 LA STORIA DI SPEC IN AMBITO HEP

SPEC CPU 2006 è l'ultima evoluzione della SPEC CPU. Andando indietro nel tempo abbiamo avuto SPEC CPU 2000 INT famigliarmente conosciuto come SI2K e prima SPEC 98 e SPEC 95. Negli ultimi anni i Computing TDR (Technical Design Report) degli esperimenti hanno fatto uso di SI2K o anche di kSI2K. Di conseguenza anche le potenze di calcolo dei centri di calcolo veniva espressa in termini di kSI2K.

## 2.1 La crisi di SI2K

SI2K è entrato in crisi per due motivi: il primo sta nel fatto che alcuni siti riportavano i valori di SI2K misurati con gcc mentre altri portavano i valori riportati dal sito www.spec.org.

La figura seguente illustra bene la differenza tra i due approcci.

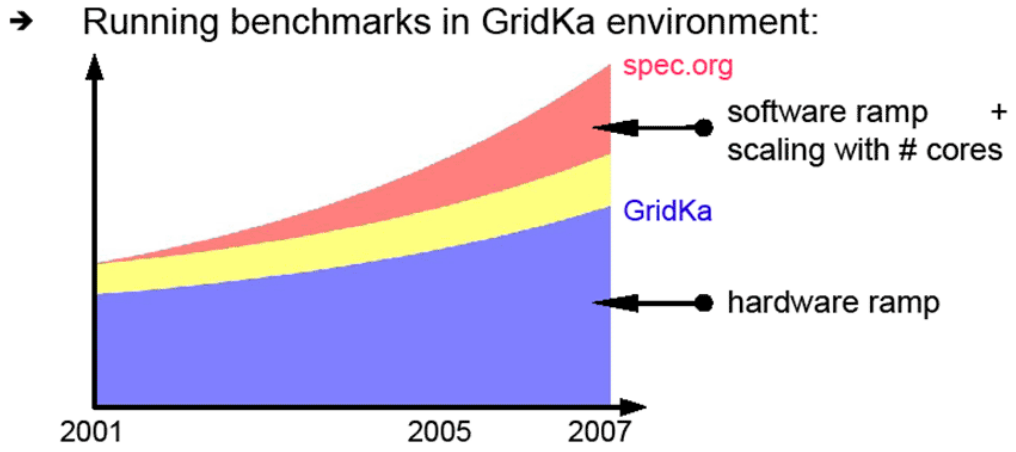


Figura 2 – Il rapporto tra valori misurati e valori ufficiali cambia nel tempo. È necessaria una rinormalizzazione al valore del 2001

La fascia inferiore blu rappresenta il valore misurato a GridKa [2] con gcc nel 2001, con un'ottimizzazione migliore di quella usato normalmente in ambiente HEP dava un valore di CPU INT pari all'80% circa di quello che si vedeva su www.spec.org (fascia gialla intermedia) Era sufficiente quindi fare una misura e aumentarla del 25% per ottenere il valore del sito ufficiale di spec. Nel 2007 invece il rapporto è cambiato. L'aumento di prestazioni dell'hardware ha portato un aumento di prestazioni visibile nella zona inferiore blu, ma i compilatori usati nel sito www.spec.org hanno dato un incremento (descritto nella zona superiore rossa come software ramp) che in ambienti con il compilatore gcc non si vede. Usando le ottimizzazioni meno spinte tipiche definite da WLCG Architect Forum la figura sarebbe leggermente diversa. Nel 2001 il valore misurato era circa il 65% di quello del sito www.spec.org per cui nel 2007 il valore misurato va incrementato del 50% per rinormalizzarlo al 2001.

Abbiamo visto dunque come il rapporto tra le due misure che si pensava fosse costante in realtà è cambiato nel tempo.

Inoltre questi rapporti cambiavano anche per architetture diverse (INTEL vs AMD o anche solo tra vecchi processori INTEL e nuovi processori INTEL).

---

[2] grid.fzk.de

La cura temporanea fu la definizione di un protocollo di misura all'interno della comunità WLCG per la misura di SPEC CPU INT 2000 con una misura che chiameremo SI2K-LCG. La procedura stabilisce che SI2K venga ottenuto con il test CPU INT compilato con il compilatore "*gcc*", con gli switch di compilazione standard definito da WLCF Architect Forum "*-O2 -fPIC -pthread*" e rivalutata del 50%.

In questo modo veniva recuperata nel 2006 il rapporto tra SI2K-LCG e il valore pubblicato.

Era una misura dichiaratamente temporanea dal momento che SI2K era comunque basato su di una suite che è stata ritirata da SPEC per obsolescenza tecnica nel 2006. Certo si sarebbe potuto usare ancora per qualche anno con nuove rinormalizzazioni ma andava indagata la possibilità di usare un nuovo benchmark dal momento che CPU 2000 era progettato per misurare bene i computer del 2001 e occupava in memoria una piccola parte della memoria in uso nei processori moderni (circa 200MB) e per gran parte del tempo il codice riusciva a rimanere in cache.

## 3 HEP-SPEC06

Il benchmark proposto dal gruppo di lavoro HEPIX si chiama HEP-SPEC06 (HS06) ed è composto da tutti e soli i test di CINT e CFP scritti in C++. Il motivo della scelta sta nell'ottimo accordo dei valori di HEP/SPEC06 con le prestazioni degli esperimenti LHC, ATLAS, CMS, LHCB e ALICE, sia dal lato delle simulazioni che nel codice di ricostruzione. Inoltre viene garantito un buon mix di benchmark INTEGER e FLOATING POINT.

Come abbiamo detto SPEC CPU è progettato per comparare carichi di lavoro CPU intensive su sistemi di calcolo diversi. Contiene due suites: CINT2006 per misurare e confrontare carichi di lavoro in cui sono importanti le prestazioni su dati INTEGER e CFP2006 per i calcolo di tipo FLOATING POINT.

SPEED e RATE

La misura di default, SPEED riguarda le prestazioni di una unica CPU e determina la velocità di esecuzione, espressa come rapporto tra il tempo di esecuzione di una macchina di riferimento e il tempo misurato per la macchina sotto esame. CPU più veloci hanno valori più alti.

Ogni test viene eseguito tre volte e si prende come valore quello mediano. Poi si esegue la media geometrica di tutti i rapporti per trovare il valore finale per una determinata suite.

La tabella 1 illustra i test di tipo Intero e di tipo Floating Point.

| Test di CPU INT 2006 | | Test di CPU FP 2006 | |
|---|---|---|---|
| Nome del test | Linguaggio | Nome del Test | Linguaggio |
| 400.perlbench | C | 410.bwaves | Fortran |
| 401.bzip2 | C | 416.gamess | Fortran |
| 403.gcc | C | 433.milc | C |
| 429.mcf | C | 434.zeusmp | Fortran |
| 445.gombk | C | 435.gromacs | C/Fortran |
| 456.hemmer | C | 436.cactusADM | C/Fortran |
| 458.sjeng | C | 437.leslie3d | Fortran |
| 462.libquantum | C | 444.namd | C++ |
| 464.h264ref | C | 447.dealII | C++ |
| 471.omnetpp | C++ | 450.soplex | C++ |
| 473.astar | C++ | 453.povray | C++ |
| 483.xalancbmk | C++ | 454.calculix | C/Fortran |
| | | 459.GemsFDTD | Fortran |
| | | 470.lbm | C |
| | | 481.wrf | C/Fortran |
| | | 482.shinx3 | C |

**Tabella 1** – I test di INT e FP di CPU 2006

La scelta dei soli test C++ permette di avere un benchmark che gira in poche ore nei processori moderni, riconosce l'elevato uso dei compilatori C++ nello sviluppo dei programmi HEP e aumenta l'importanza della valutazione delle prestazioni in virgola mobile (FP) che non erano usate in quantità troppo ridotta in CPU INT 2000 o in CPU INT 2006.

La tabella 2 riporta di nuovo i nomi dei test in C++ con una breve descrizione del task eseguito.

| | |
|---|---|
| 471.omnetpp | Discrete Event Simulation |
| 473.astar | Path-Finding Algorithms |
| 483.xalancbmk | XML Processing |
| 444.namd | Biology/Molecular Dynamics |
| 447.dealII | Finite Element Analysis |
| 450.soplex | Linear Programming Optimization |
| 453.povray | Image Ray-tracing |

**Tabella 2** - I test in C++

## 4 COME POSSO FARE UNA MISURA DI HEP-SPEC06 SULLA MIA MACCHINA?

Prima di tutto bisogna procurarsi una **versione di SPEC CPU 2006**. L'INFN ha una licenza. Contattare Michelotto AT pd.infn.it per scaricare una copia del kit.

Serve un **file di configurazione per SPEC**, che per Sistemi Operativi simili a RedHat, quindi anche SL4 oSL5, viene fornito da WLCG.

Serve un **compilatore gcc**. La versione 4.x di default su Scientific Linux 5 va bene ma è possibile usare anche gcc3.4.3 che è di default su Scientific Linux 4.

Gli switch di compilazione sono sempre gli stessi suggeriti da LCG Architects Forum: ***"gcc -O2 -fPIC -pthread -m32"***. Lo switch ***–m32*** serve per forzare la compilazione di codice a 32 bit anche su macchine con architettura a 64 bit (x86_64)

Il comando da dare per ogni core è il seguente:

***# runspec.sh -d "HEP-SPEC06 32-bit" -a 32 -b all_cpp***

Alla fine bisogna fare la media geometrica dei 7 risultati per ogni core e sommare sul numero dei core (in realtà delle cpu logiche).

Lo si può fare con un comando unix del tipo ***scale=8; e(($partial)/$count)" | bc -l'*** dove *$partial* è la somma dei logaritmi dei risultati e *$count* il numero di cpu logiche. Il comando citato permette quindi di ottenere ***(x * y * z)**1/3*** sotto forma di ***exp( ( ln(x) + ln(y) + ln(z) ) / 3 )*** usando il comando unix ***bc(1)***

Hepix mantiene una pagina web che fornisce lo script che si occupa di lanciare correttamente il benchmark e di calcolare la media finale. Fornisce inoltre le istruzioni e il file di configurazione.

Le istruzioni sono disponibili sul sito del gruppo benchmarking di HEPIX:
http://hepix.caspur.it/benchmarks/doku.php?id=bench:howto
Altre istruzioni in inglese si trovano nel sito del gruppo server di CCR:
http://www.infn.it/CCR/server/

## 5 VALORI TIPICI DI HEPSPEC

Prima di tutto si deve osservare che si SPEC CPU e quindi HS06 misurano le prestazioni di una macchina e non di un processore. Il processore è solo una delle componenti, diciamo pure la più importante, ma non è l'unica componente che influenza il risultato.

- Memoria. Il tipo di memoria non è molto importante. Il test non sembra essere molto sensibile al numero di banchi di memoria, alla banda passante verso la memoria o alla latenza. È importante però che la memoria totale non sia troppo bassa altrimenti il sistema comincia a paginare su disco e le prestazioni crollano.
- Compilatore. Il compilatore è molto importante. Il passaggio da gcc3 a gcc4 permette di migliorare di qualche percento, soprattutto per codice compilato a 64bit. Nell'attuale definizione di HS06 a 32 bit comunque il compilatore influisce visibilmente sul risultato. Le misure di CPU 2006 con compilatori più ottimizzati permettono di migliorare le prestazioni anche del 30-40%.
- Sistema Operativo. Le prestazioni non dipendono troppo dal sistema operativo. Tuttavia al cambio di sistema operativo corrisponde spesso un cambio di compilatore di default, per cui un apparente miglioramento nel passaggio da SL4 a SL5 potrebbe in realtà essere dovuto al passaggio del compilatore da gcc3 a gcc4.
- Altre variabili. Chiaramente tutto il resto della macchina può influire, chipset e dischi per esempio. Anche le condizioni ambientali possono influire perché le CPU moderne

aggiustano dinamicamente il clock al variare delle temperatura misurata dai sensori interni.

La tabella 3 illustra valori tipici di HS06 per macchine disponibili fino alla fine del 2008 permettendo di confrontare i valori del nuovo benchmark con il vecchio SI2K (non rivalutato).

| Worker Node, cpu, clock, L2 cache, - Main Memory | SPECint2000 gcc | SPEC CPU2006 int - gcc | HEP-SPEC | site measured |
|---|---|---|---|---|
| 2 x Nocona/Irvindale 2.8 GHz/1 MB - 2GB | 1501 | 11.06 | 10.24 | CERN |
| 2 x Nocona/Irvindale 2.8 GHz/2 MB - 4GB | 1495 | 10.09 | 9.63 | CERN |
| 2 x Nocona/Irvindale 2.8 GHz/2 MB - 2GB | 1673 | 11.87 | | CERN |
| 2 x Nocona/Irvindale 2.8 GHz/2 MB - 2GB | 1703 | 12.26 | | CERN |
| 2 X Opteron 275 2.2 GHz/2 MB - 4GB | 4133 | 28.76 | 28.03 | CERN |
| 2 x Woodcrest 2.66 GHz/4 MB - 8GB | 5675 | 36.77 | 35.58 | CERN |
| 2 x Woodcrest 3.00 GHz/4 MB - 8GB | 6181 | 39.39 | 38.21 | CERN |
| 2 x Opteron 2218 (Rev. F) 2.6 GHz/2 MB- 8GB | 4569 | 31.4 | 31.67 | CERN |
| 2 x Clovertown 2.33 GHz/2x4MB - 16GB | 9462 | 60.89 | 57.52 | CERN |
| 2 x Harpertown (E5410) 2.33 GHz/2x6 MB - 16GB | 10556 | 64.78 | 60.76 | CERN |
| 2 x Harpertown (5440) 2.83 GHz/2x6MB - 16GB | 11850 | 73.32 | | DESY |
| 2 x Harpertown (5410) 2.33 GHz/2x6MB - 16GB | 11164 | 65.93 | 62.12 | INFN-PD |
| 2 x Barcelona (2352) 2.10 GHz/2x4MB - 16GB | 8488 | 56.23 | | INFN-PD |
| 2 x Barcelona (2360) 2.50 GHz/2x4MB - 16GB | 9939 | 63.75 | 63.19 | INFN-PD |
| 2 x Barcelona (2356) 2.30 GHz/4x512 KB - 16GB | 9565 | 61.05 | 59.74 | GridKa |
| 2 x Shanghai (2376) 2.30 GHz/4x512KB+6MB L3 - 16GB | 10962 | 66.88 | 65.85 | GridKa |
| 2 x Harpertown (E5430) 2.66 GHz/2x6MB -16GB | 12122 | 72.14 | 68.04 | GridKa |
| 2 x Opteron (2216) 2.4GHz -8GB | | 21.86 | 22.22 | RAL |
| 2 x Barcelona (2354) 2.2 GHz -16GB | | 58.17 | 58.1 | RAL |
| 2 x Clovertown (E5335) 2.00 GHz/2x4 MB - 16GB | | 54.42 | 52.31 | RAL |
| 2 x Harpertown (L5410) 2.33 GHz/2x6MB -16GB | | 68.84 | 62.23 | RAL |
| 2 x Harpertown (E5410) 2.33 GHz/2x6MB -16GB | | 65.73 | 62.21 | RAL |
| 2 x Harpertown (L5420) 2.50 GHz/2x6MB - 16GB | | 68.73 | 62.11 | RAL |
| 2 x Harpertown (E5420) 2.50 GHz/2x6MB - 16GB | | 69.13 | 65.11 | RAL |
| 2 x Harpertown (E5420) 2.50 GHz/2x6MB, 16GB | | 66.83 | 64.85 | RAL |
| 2 x Harpertown (E5440) 2.83 GHz/2x6MB, 16GB | | 74.92 | 68.51 | RAL |

**Tabella 3**