HEPiX Benchmarking Group

Michele Michelotto at
pd.infn.it

A comparison of HEP code with SPEC benchmark on multicore worker nodes

What is HEPiX?

- HEPiX:
  - An international group of Unix users and administrator from **cooperating** HEP institutions and HEP data center
- Initial focus:
  - enhance Unix in a standard way, like was done inside HEPVM in the 80's.
- Now:
  - more focus on sharing of experiences, documentation, code and best practices, in all area of computing (Linux, Windows, Mail, Spam, AAA, Security, Infrastructures)

HEPiX Meeting

- A Yearly HEPiX meeting in Spring (Europe)
- A Yearly HEPix meeting in Fall (North America)
- Most recent meeting was at ASGC, Taipei (the Taiwan Tier1)
- Next meeting at Umeå univ. (Sweden), May 25-29, 2009
- Each meeting ~100 users, ~50 talks and many open discussions
- To join:
  - Send an e-mail message to: listserv@fnal.gov
  - Leave the subject line blank
  - Type "SUBSCRIBE HEPiX-hepnt FIRSTNAME LASTNAME" (without the quotation marks) in the body of your message.

HEPiX Benchmarking WG

- Since about 2004 several HEPiX users were presenting measurements on performances and benchmarking
- Anomalies in performances between application code and SI2K
- In 2006 a Working Group, chaired by Helge Meinhard (CERN) was setup inside HEPiX to address those issues
- We requested an help from the major HEP experiments

The Group

- People from HEPiX
  - Helge Meinhard (chair, CERN IT)
  - Peter Wegner (Desy)
  - Martin Bly (RAL)
  - Manfred Alef (FZK Karlsruhe)
  - Michele Michelotto (INFN, Padova)
  - Ian Gable (Victoria CA)
  - Andreas Hirstius (CERN, OpenLab)
  - Alex Iribarren (CERN IT)

- People sent by the Experiments:
  - CMS: Gabriele Benelli
  - ATLAS: Franco Brasolin, Alessandro De Salvo
  - LHCB: Hubert Degaudenzi
  - ALICE: Peter Hristov

What is SPEC?

- SPEC
  - "www.spec.org : a non profit corporation that establish maintains and endorses a set of computer related benchmarks"
- SPEC CPU
  - "Designed to provide performance measurements that can be used to compare compute-intensive workloads on different computer systems"
- History
  - Before SPEC: CERN UNIT, MIPS, VUPS (Lep Era)
  - After SPEC: SPEC89, CPU92, CPU95, CPU2000, CPU2006

Why INT ?

- Since SPEC CPU 92 the HEP world decide to use INT as reference instead of FP (Floating Point)
- HEP programs of course make use of FP instructions but with minimal inpact on benchmarks
- I’ve never seen a clear proof of it

The mythical SI2K

- SPEC CPU INT 2000 shortened as SI2K
- The “Unit of Measure”
  - For all the LHC Computing TDR
  - For the WLCG MoU
  - For the resources pledged by the Tier [0,1,2]
  - Therefore used in tender for computer procurements

The measured SI2K

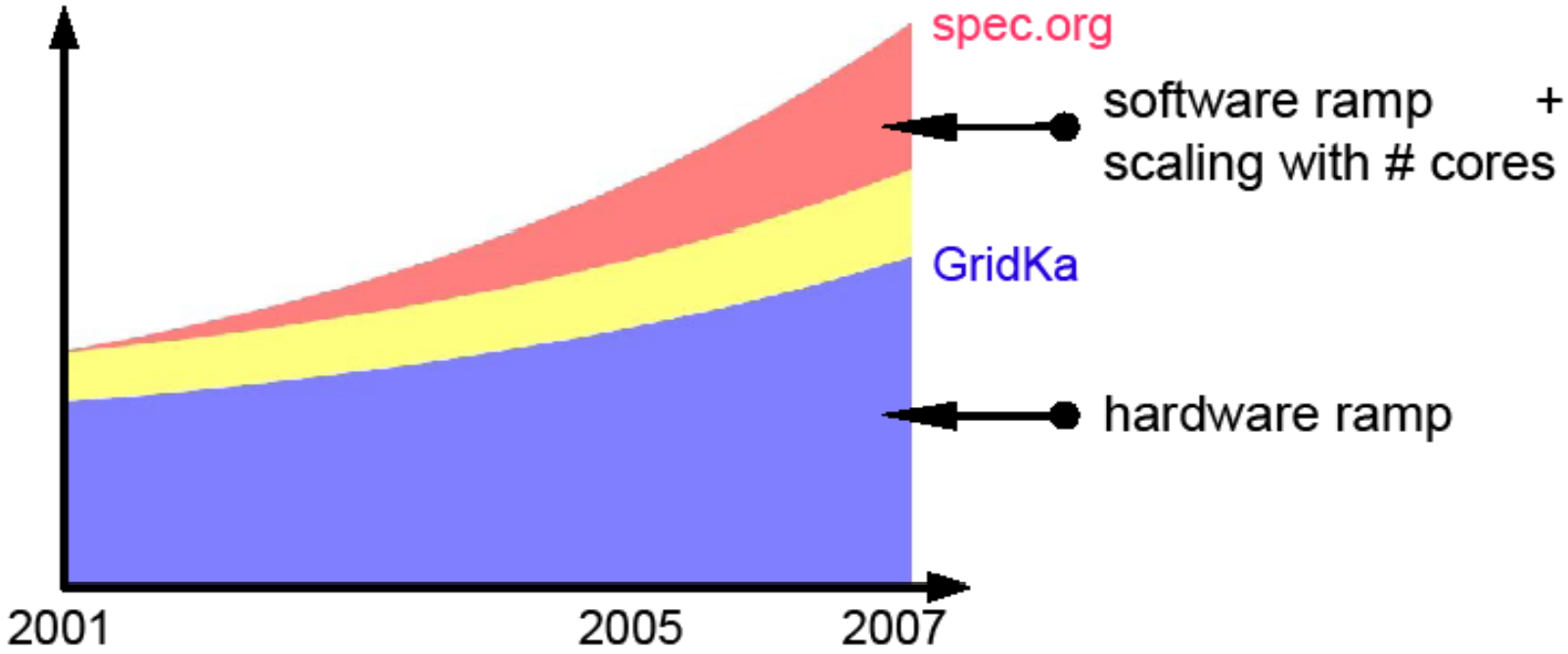
- Results taken from www.spec.org for different processors showed good linearity with HEP applications up to ~ Y2005
- HEP applications use Linux + gcc
- SPEC.org makes measurements on Linux/Win + Intel or Pathscale compiler
- If you run SPEC on Linux+gcc you obtain a smaller value (less optimization)
- Is it proportional to SPEC.org or to HEP applications?

SI2K measurement

- Take your typical WN; a dual proc with Linux + gcc
- Compile it in your typical environment with typical optimisation
  - for GridKa: “gcc –O3 –march=$ARCH”
  - for Cern (LHC): “gcc –O2 –fPIC –pthread”
- If you have N cores → Run N instances of SPEC INT in parallel
- In 2001 GridKa / Spec.org ratio was 80%
- So they needed to apply a scaling factor of +25%

The SI2K inflation

➔ Running benchmarks in GridKa environment:



- Blue is the value measured with gcc and GridKa tuning
- Yellow is the 25% scaling to normalize to 2001
- Red is the value published by spec.org

SPEC CERN and SPEC LCG

- At HEPiX meetings since 2005, people presented measurement showing the correlation of HEP application with SPEC measured
- Of course lack of linearity with spec.org
- Interim solution
  - Make measurement with Cern tuning (**gcc -O2 –fPIC –pthread**)
  - Add +50% to normalize to 2001
  - This was the SI2K LCG to be used for the pledges

Too many SI2K?

- Too many definition of SI2K around
- E.g. take a common processor like an Intel Woodcrest dual core 5160 at 3.06 GHz
- SI2K spec.org: 2929 – 3089 (min – max)
- SI2K sum on 4 cores: 11716 - 12536
- SI2K gcc-cern: 5523
- SI2K gcc-gridka: 7034
- SI2K cern + 50%: 8284

Transition to CPU 2006

- The use of the SI2K-LCG was a good INTERIM solution
- In 2006 SPEC published CPU 2006 and stopped the maintenance on CPU 2000
- Impossibile to find SI2000 from SPEC for the new processor
- Impossibile to find SI2006 for old processor
- Time to move to a benchmark of CPU 2006 family?

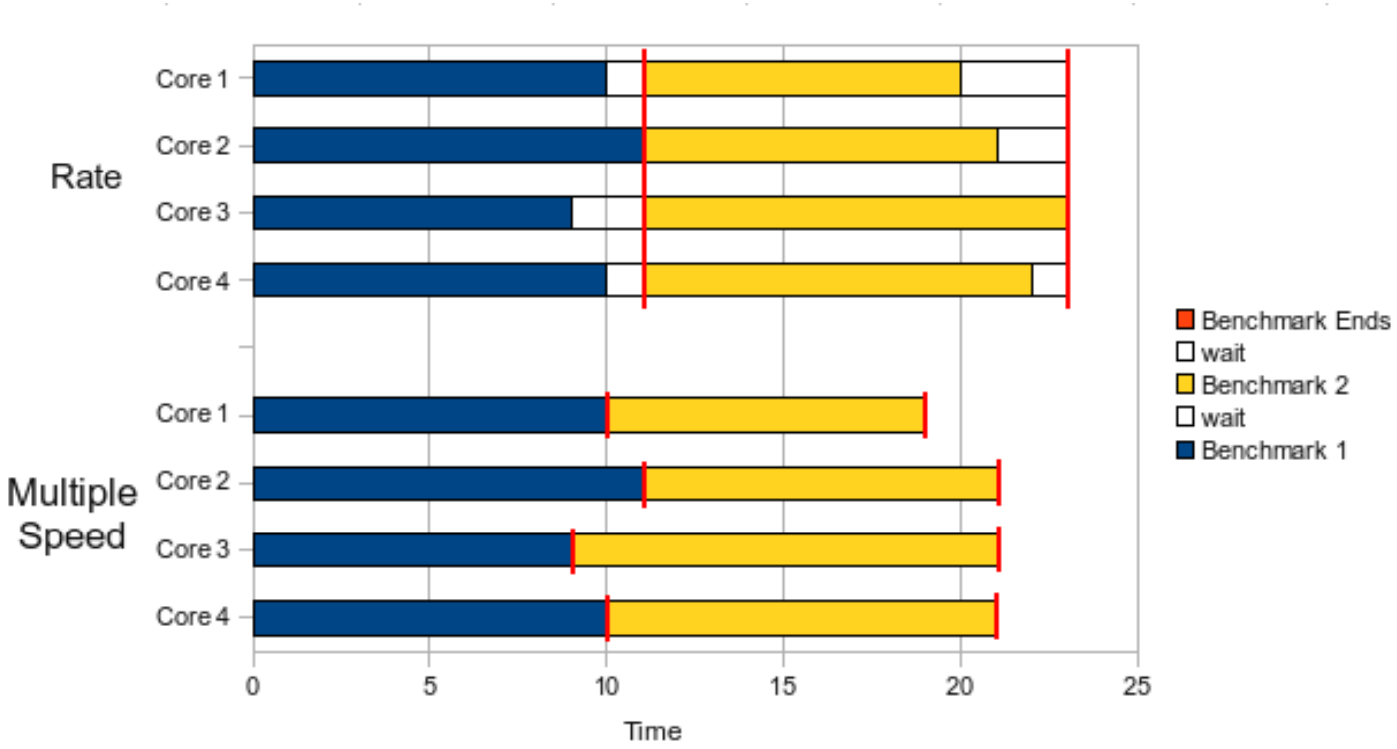
CPU 2006

- What's new:
  - Larger memory footprint: from ~200MB per core to about 1GB per core in 32bit environment
  - Run longer (1 day vs 1 hour)
  - CPU 2000 fitted too much in L2 caches
  - INT: 12 CPU intensive applications written in C and **C++**
  - FP: 17 CPU intensive applications written in C, **C++** and Fortran

The HEPiX WG

- In the HEPiX Fall 2006 meeting at JLAB a group, chaired by H.Meinhard (CERN-IT) started a detailed study of CPU2006
- We needed to compare CPU 2000 and CPU 2006 with HEP applications
- We found a good collaboration with LHC experiments thank to the push of WLCG Grid Deployment Board

SPEC rate vs parallel



- SPEC Rate syncronizes all the cores at the end of each test
- We preferred to emulate the batch-like environment of our farms using multiple parallel run
- Noticeable effect if the WN has four or more cores

Lxbench cluster

- We needed and obtained a set of dedicated Worker Nodes at CERN
  - To measure SI2000, 32 and 64 bit
  - To measure CPU 2006, INT and FP, 32 and 64 bit
  - To measure on EXACTLY the same machines the LHC applications performances
  - All dual processor, both Intel and Amd, single core, dual core, quad core
  - Plus other “control” machines from INFN, DESY, GridKa, RAL

HEP Applications

- Atlas provided results for:
  - Event Generation, Simulation, Digitization, Reconstruction, Total (Full chain production)
- Alice:
  - Gen+Sim, Digitization, Reconstruction and Total
- LHCB:
  - Gen+Sim
- CMS
  - Gen+Sim, Digitization, Reconstruction and Total
  - For several Physics Processes (Minimum Bias, QCD Jets, TTbar, Higgs in 4 lepton, single particle gun events) to see if some physics channel would produce something different

Results

- Very good correlation (>90%) for all experiments
- Both SI2006 and SFP2006 (multiple parallel) could be good substitute for SI2000
- Interesting talk from Andreas Hirstius from CERN-IT Openlab at HEPiX Spring 08 on “perfmon”

perfmon

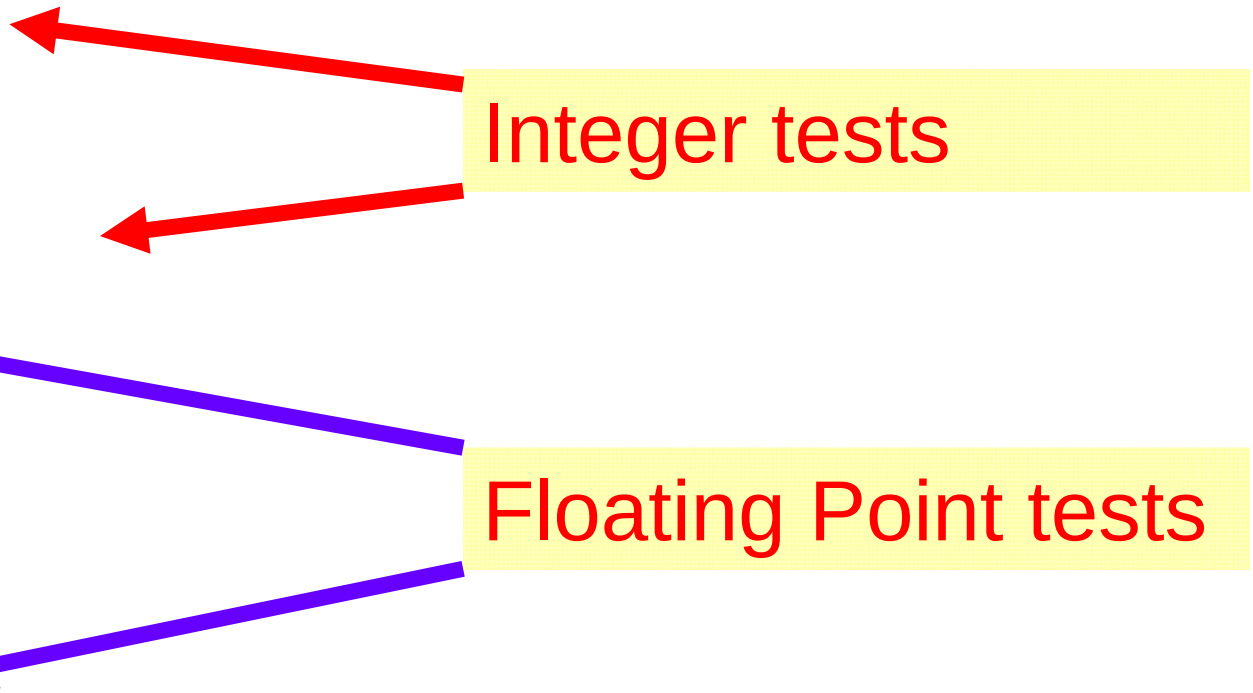
- Measure a large number of hardware performance counter events
- ~100 events/4-5 counters on Intel/Amd
- Very little overhead
- What do we measure:
  - Cycle per instruction, Load/Store inst., x87 or SIMD inst., % of mispredicted branches, L2 cache misses, data bus utilization, resource stall…

Perfmon on lxbatch

- Perfom was run on 5 nodes of lxbatch for one month to measure the average behaviour of real HEP applications
- Compared with SPEC CPU: 2000 and 2006 Int, Fp and CPP
- CPP is the subset of all CPP test in CPU 2006
- CPP showed a good match with average lxbatch e.g. for FP+SIMD, Loads and Stores and Mispredicted Branches

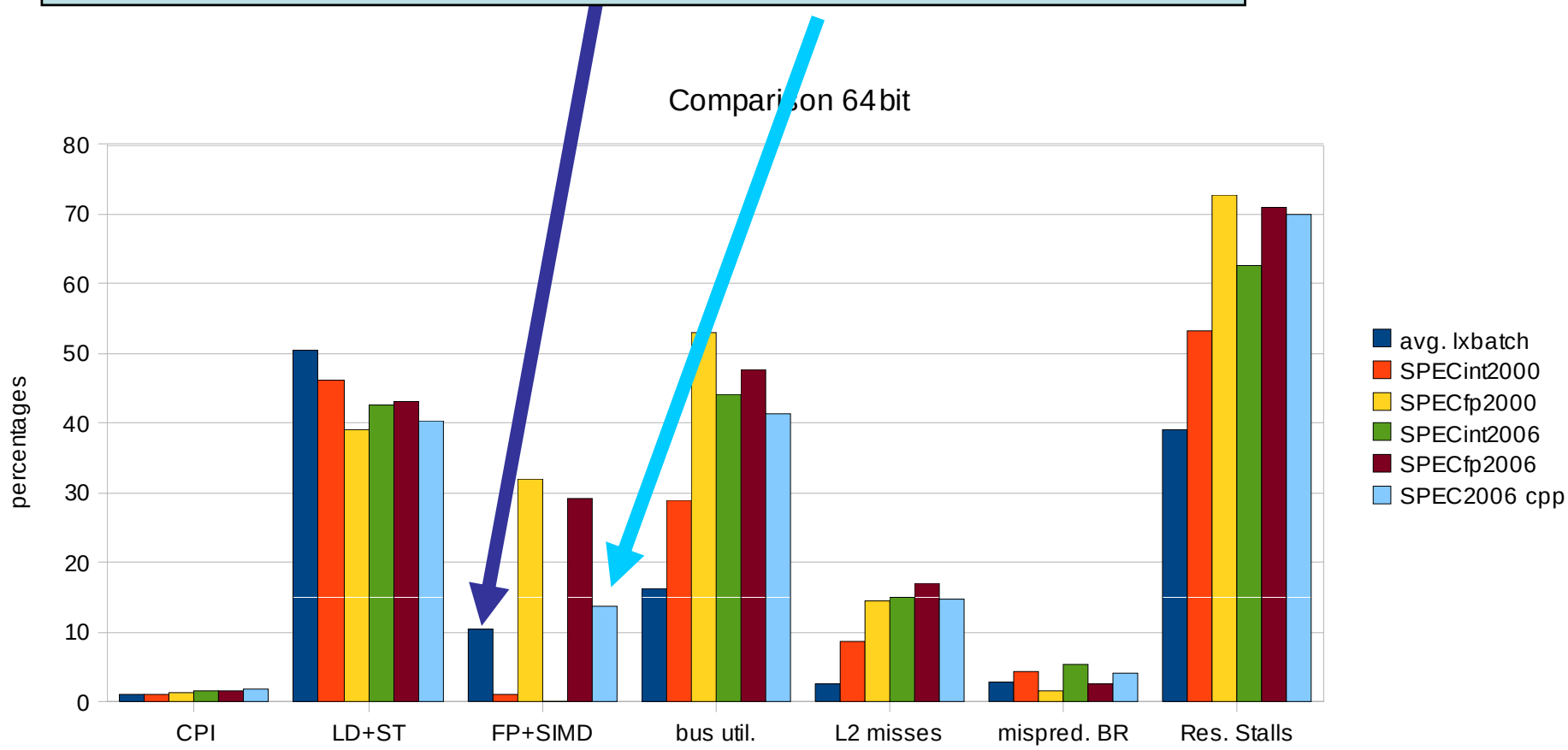
SPEC CPP

- 471.omnetpp
- 473.astar
- 483.xalancbmk
- 444.amd
- 447.dealII
- 450.soplex
- 453.povray

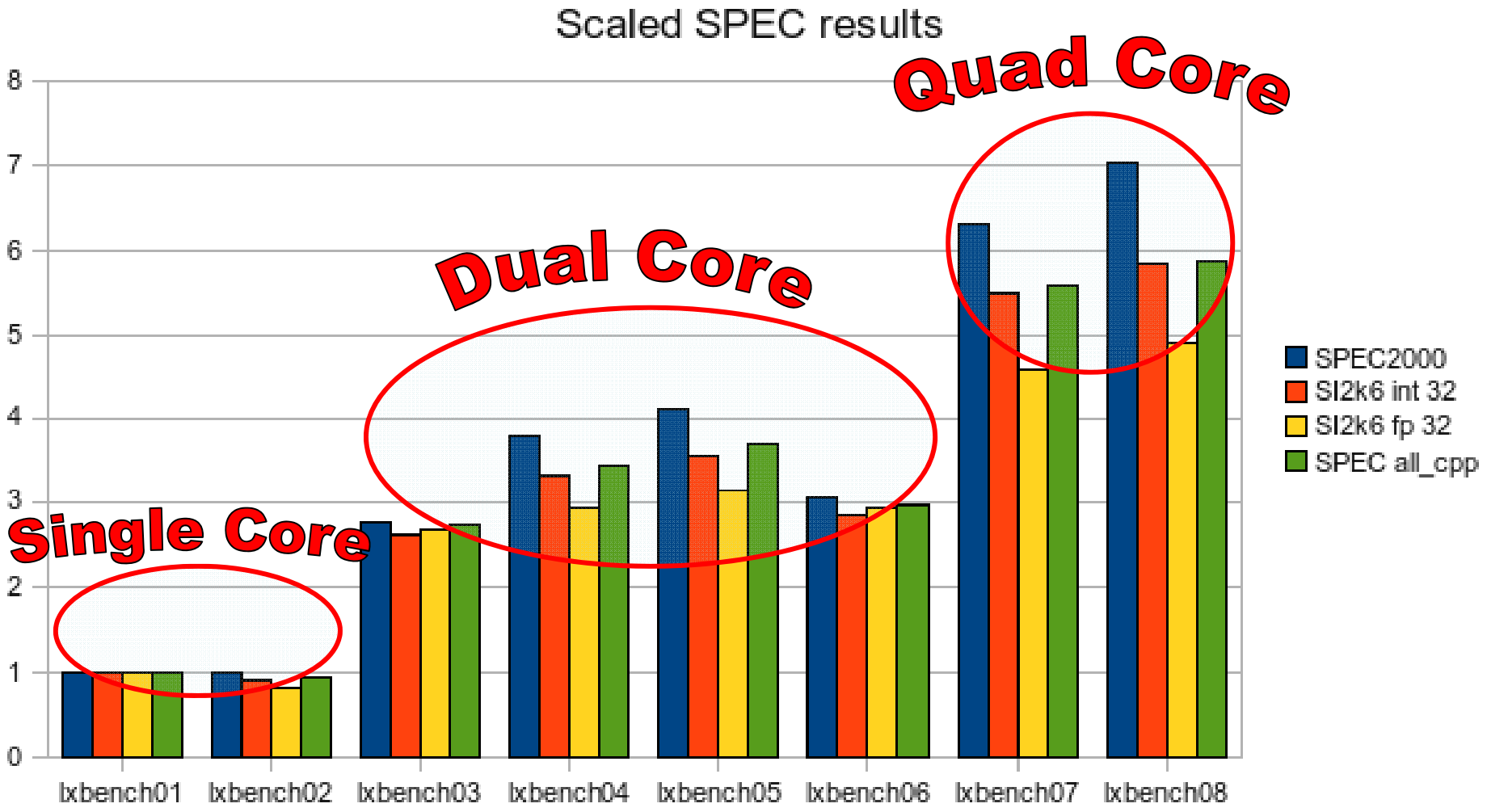


FP usage in cpp_all

Negligible usage of FP instructions in INT 2000 and 2006 similar in “average lxbatch” and “cpp_all”.



Relative performances



The choice

- SPECint2006 (12 applications)
  - Well established, published values available
  - HEP applications are mostly integer calculations
  - Correlations with experiment applications shown to be fine
- SPECfp2006 (17 applications)
  - Well established, published values available
  - Correlations with experiment applications shown to be fine
- SPECall_cpp2006 (7 applications)
  - Exactly as easy to run as is SPECint2006 or SPECfp2006
  - No published values (not necessarily a drawback)
  - Takes about 6 h (SPECint2006 or SPECfp2006 are about 24 h)
  - Best modeling of FP contribution to HEP applications
  - Important memory footprint
- Proposal to WLCG to adopt SPECall_cpp 2006, in parallel and to call it HEP SPEC06

Hep-Spec06

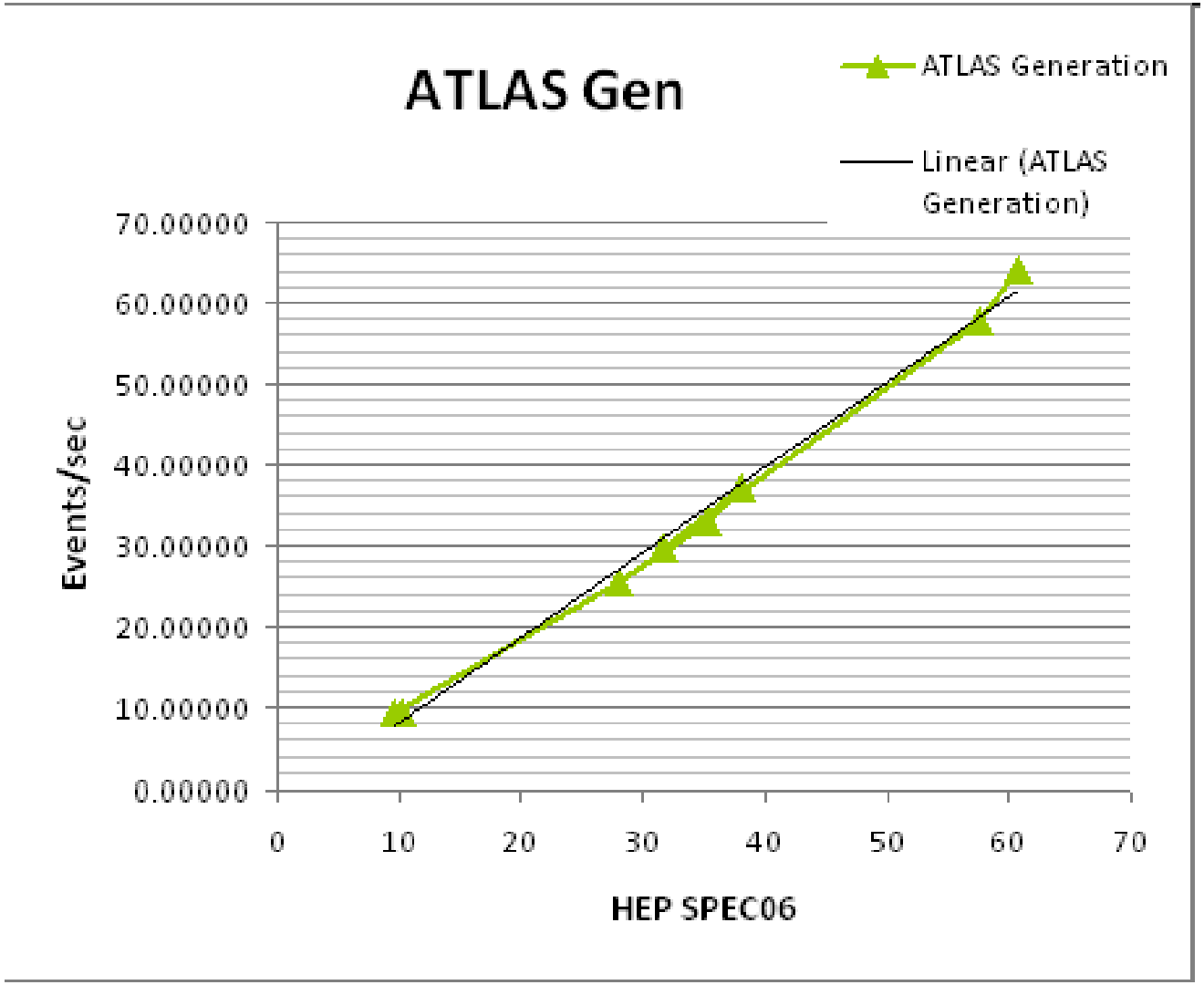
Machine	SPEC2000	SPEC2006 int 32	SPEC2006 fp 32	SPEC2006 CPP 32
lxbench01	1501	11.06	9.5	10.24
lxbench02	1495	10.09	7.7	9.63
lxbench03	4133	28.76	25.23	28.03
lxbench04	5675	36.77	27.85	35.28
lxbench05	6181	39.39	29.72	38.21
lxbench06	4569	31.44	27.82	31.67
lxbench07	9462	60.89	43.47	57.52
lxbench08	10556	64.78	46.48	60.76

Conversion factor

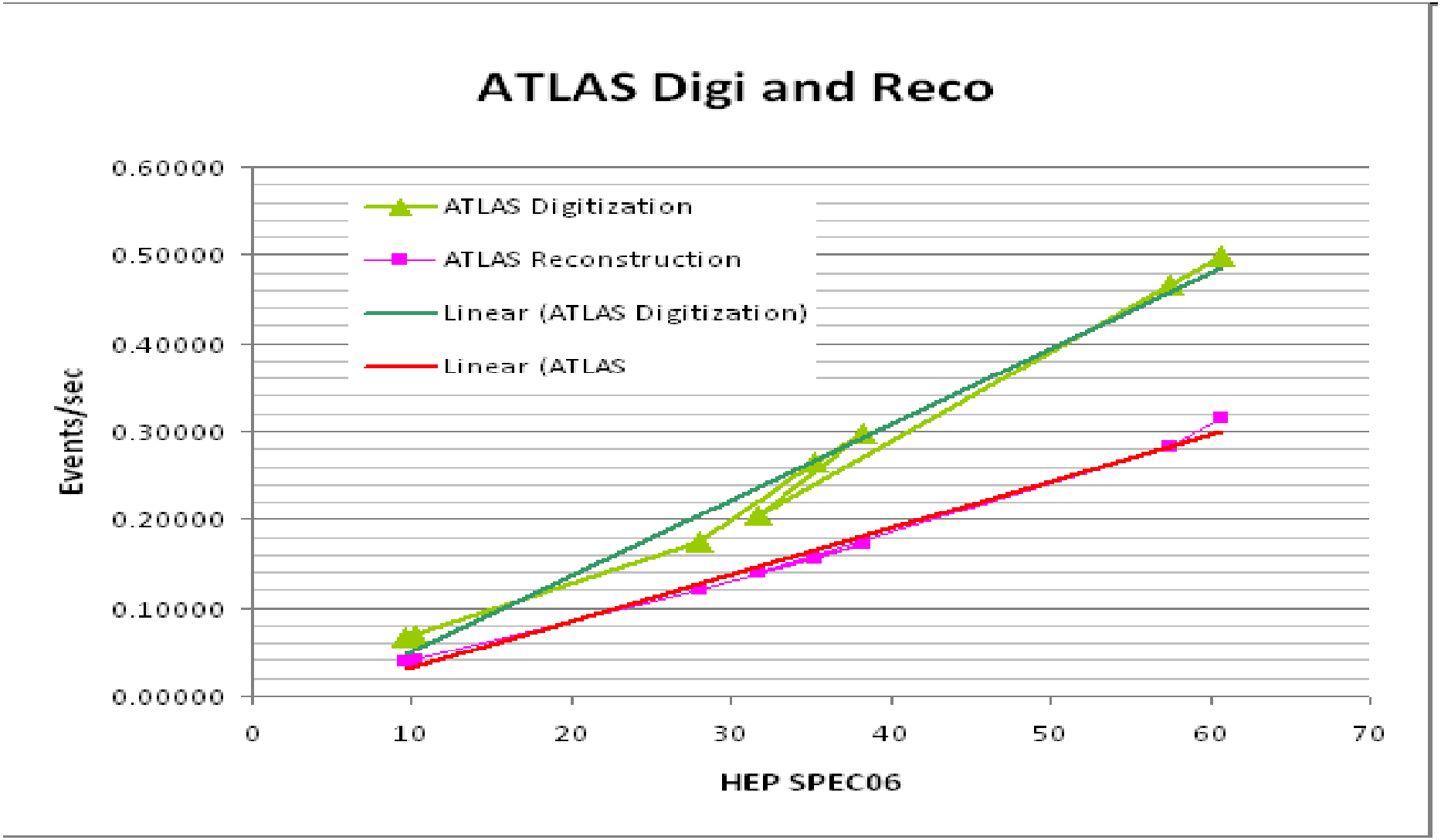
- Choose an approximate conversion factor (~5%)
- Give more weight to modern processors
- We choose a ratio of “4” to stress that we care more easiness of portability than extreme precision
- To validate we measured the whole GridKa and found the same number

Hostname	Processor type	HEP-SPEC06 (new)	KSI2K (old)	Ratio new/old
lxbench01	Intel Xeon 2.8 GHz	10,24	2,25	4,55
lxbench02	Intel Xeon 2.8 GHz	9,63	2,24	4,29
lxbench03	AMD Opteron 275	28,03	6,20	4,52
lxbench04	Intel Xeon 5150	35,58	8,51	4,18
lxbench05	Intel Xeon 5160	38,21	9,27	4,12
lxbench06	AMD Opteron 2218	31,67	6,85	4,62
lxbench07	Intel Xeon E5345	57,52	14,19	4,05
lxbench08	Intel Xeon E5410	60,76	15,83	3,84

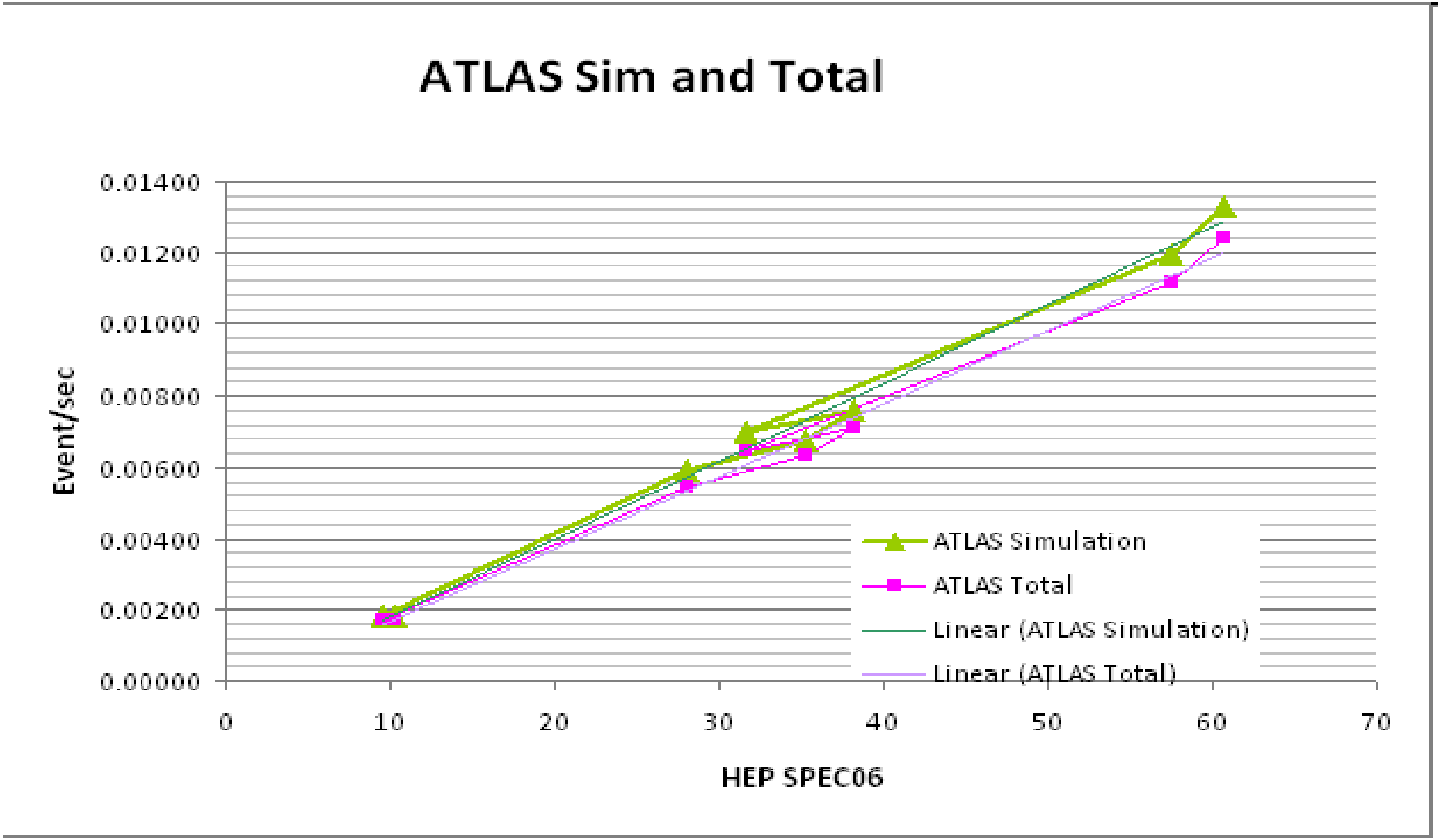
Atlas Generation



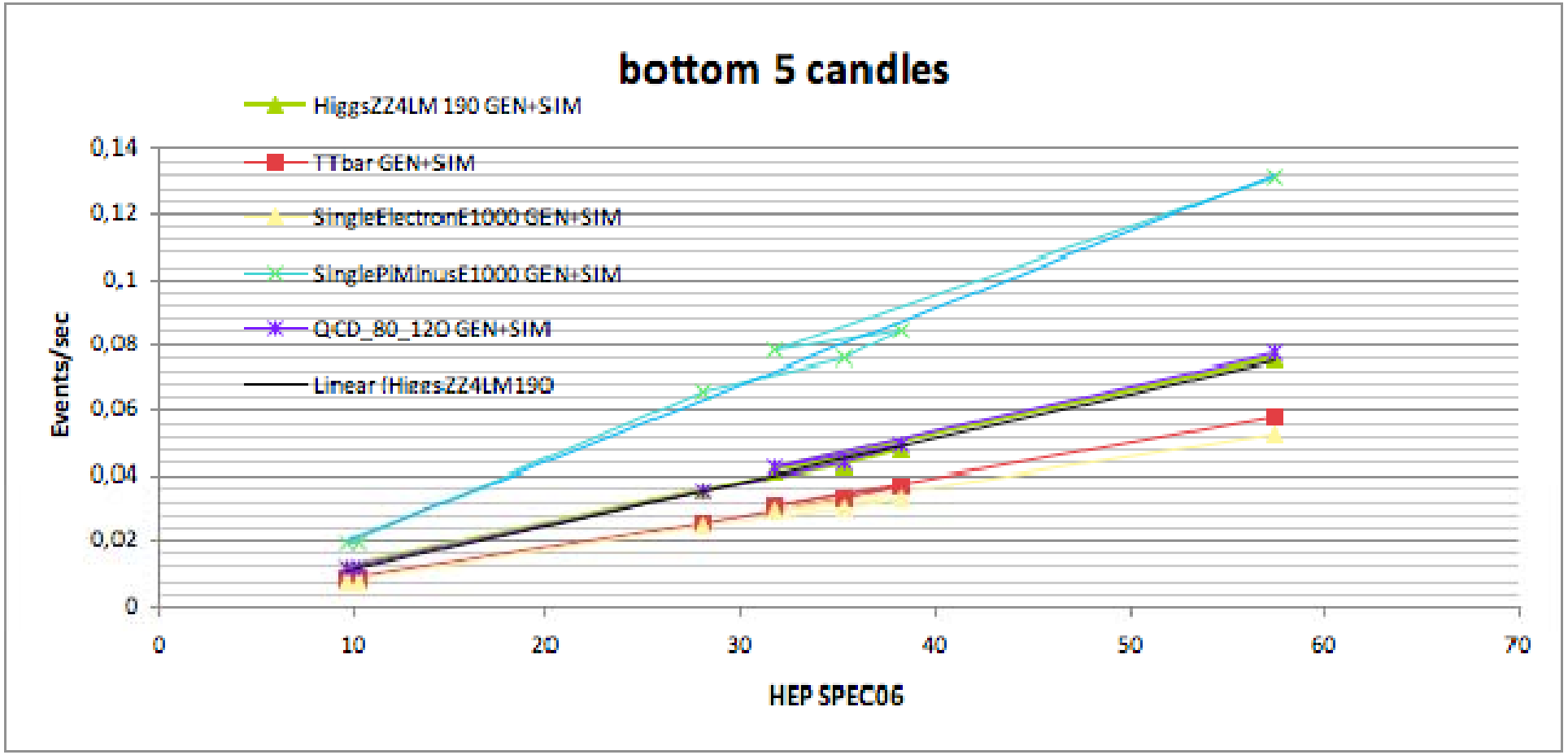
Atlas Digi and Reco



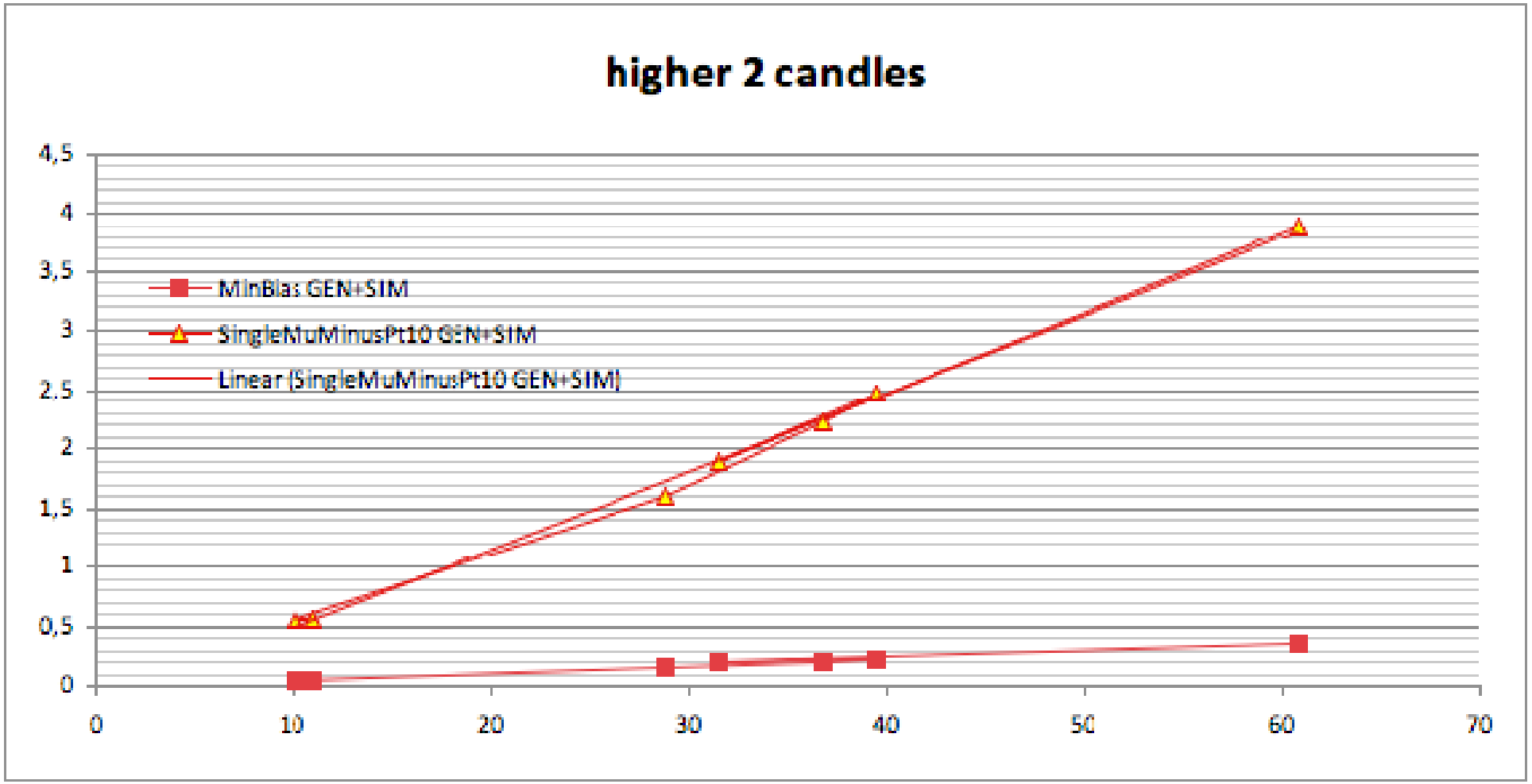
Atlas Sim and Total



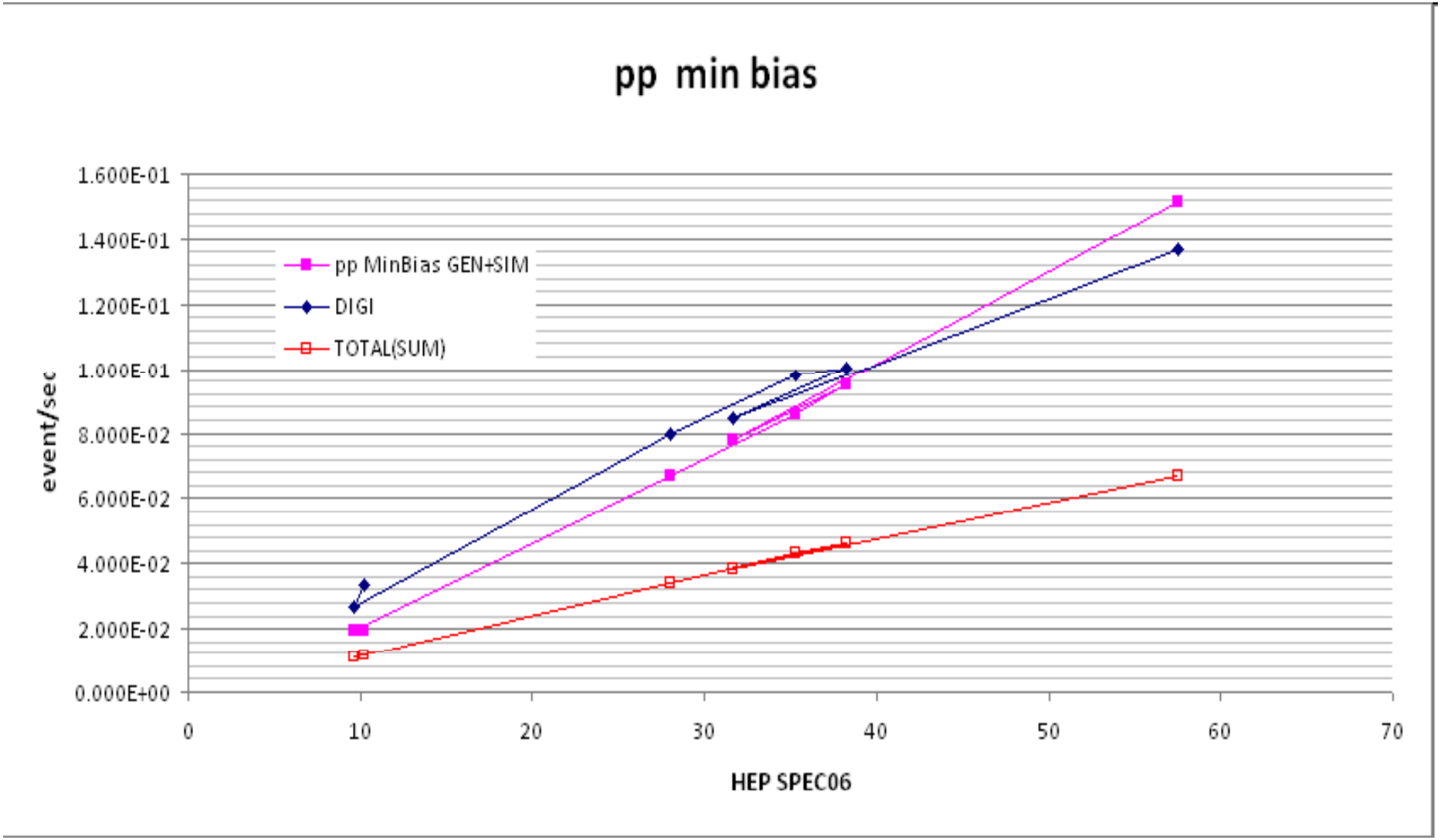
CMS slowest 5 candles



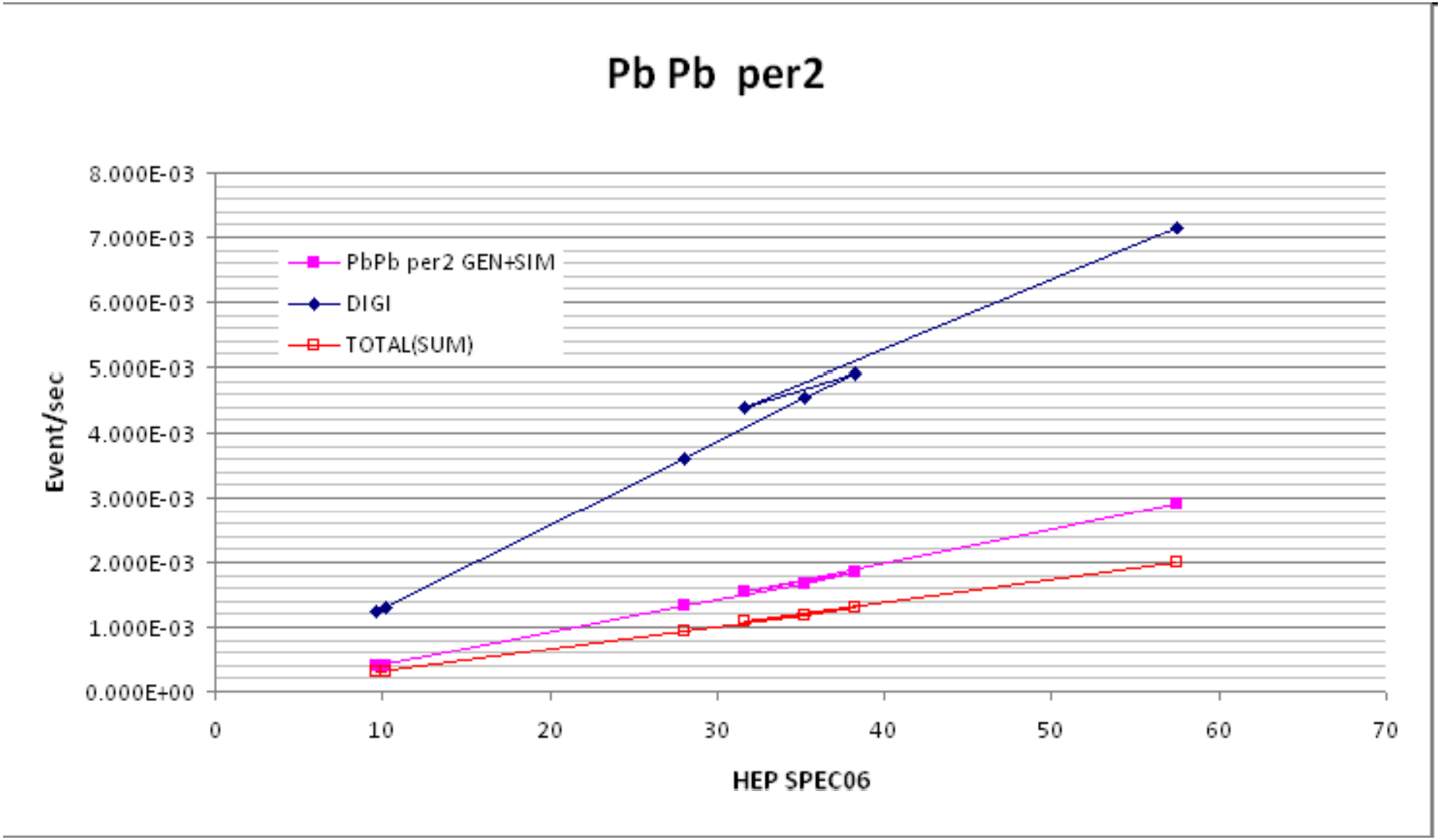
CMS fastest 2 candles



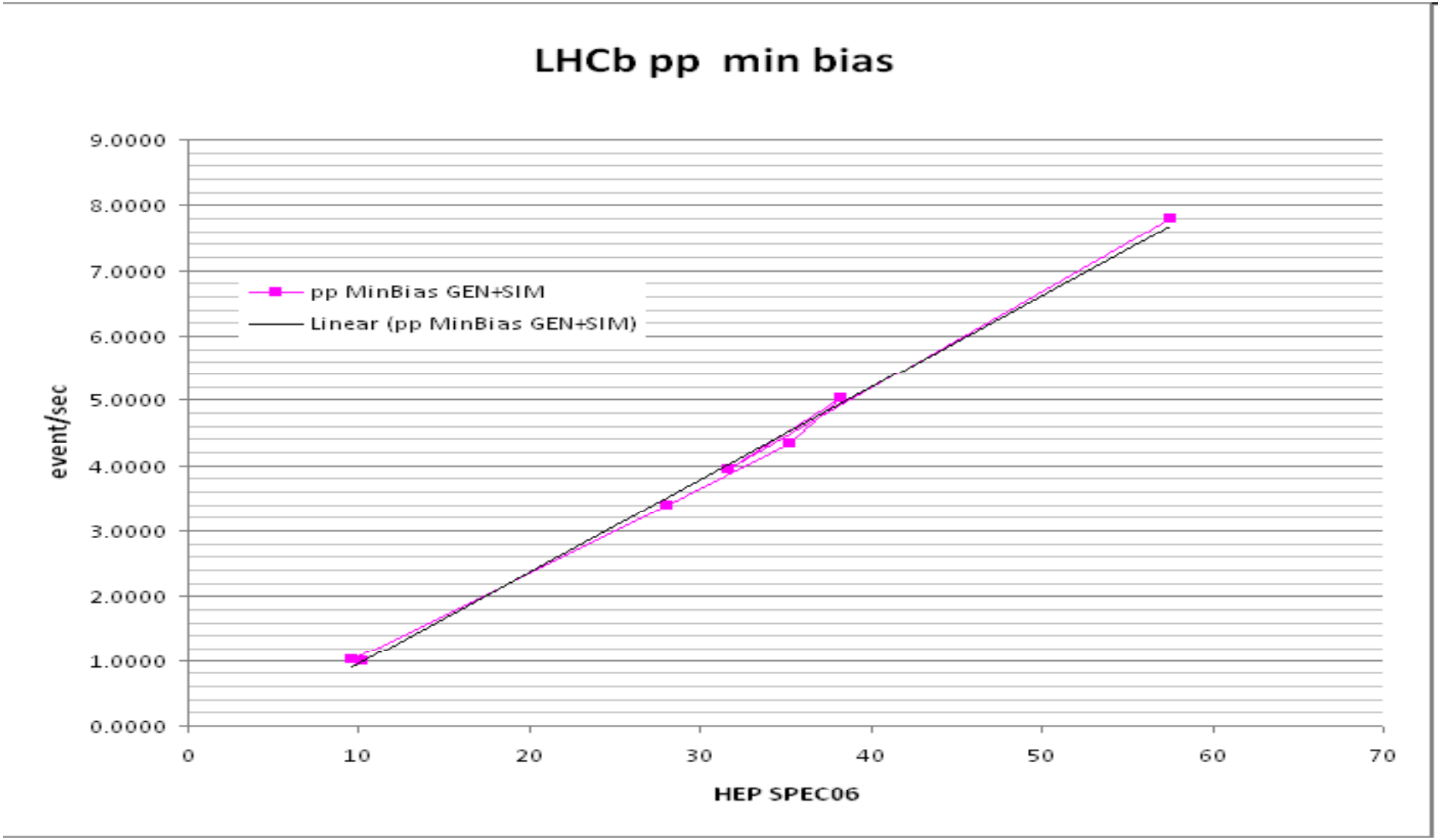
Alice pp


pp min bias
1.600E-01
1.400E-01
1.200E-01
1.000E-01
8.000E-02
6.000E-02
4.000E-02
2.000E-02
0.000E+00
event/sec
pp MinBias GEN+SIM
DIGI
TOTAL(SUM)
0
10
20
30
40
50
60
70
HEP SPEC06

Alice PbPb


Pb Pb per2
8.000E-03
7.000E-03
6.000E-03
5.000E-03
4.000E-03
3.000E-03
2.000E-03
1.000E-03
0.000E+00
Event/sec
PbPb per2 GEN+SIM
DIGI
TOTAL(SUM)
0
10
20
30
40
50
60
70
HEP SPEC06

LHCb pp


LHCb pp min bias
9.0000
8.0000
7.0000
6.0000
5.0000
4.0000
3.0000
2.0000
1.0000
0.0000
event/sec
pp MinBias GEN+SIM
Linear (pp MinBias GEN+SIM)
0
10
20
30
40
50
60
70
HEP SPEC06