Benchmarking Working Group

Status Report
and
Introduction to Benchmarking pre-GDB

2017-02-07

Manfred Alef (KIT)
manfred.alef@kit.edu

Benchmarking Working Group

Mandate:

➔ **Fast benchmark**
to estimate the performance of the provided job slot (in traditional batch farms) or VM instance (in cloud environments)

- Job matching / masonry
(e.g. “can a pilot run another payload with the resources left?”)
- Accounting if HS06 score is not available
- ...

➔ Next generation of long-running benchmark for installed capacities, accounting, procurements aso. in WLCG (successor of HS06)

Benchmarking Working Group

Organization:

- ➔ Mailing list (hepix-cpu-benchmark@hepix.org):
  ~50 subscribers
- ➔ Meetings:
  - Kick-off at HEPiX Zeuthen
  - Biweekly Vidyo meetings
  - 6 … 16 attendees per meeting

Benchmarking Working Group

Fast benchmarks:

➔ Started with 5 candidates:

- Atlas KV (default workload: 100 single muon event generations)
- DIRAC Benchmark 2012 (DB12, Python script)
  - Aka FastBmk, LHCbMarks, ...
- ROOT stress test

- Whetstone, Dhrystone

Benchmarking Working Group

Fast benchmarks:

➔ Tools used for analysis:

- CERN Cloud Benchmarking Suite
  - Framework to run benchmarks
  - Optional sending results to central ES for later analysis
  - See talks by Cristovao Cordeiro and Domenico Giordano

Benchmarking Working Group

Fast benchmarks:

- ➔ Good correlation of DB12 and Atlas KV with single-core experiment applications
- ➔ Analysis by Costin Grigoras (Alice) has shown better scaling of Alice applications with DB12 than with ROOT
- ➔ See talks by Alice, Atlas, and LHCb at GDB Sept 2016 for details (https://indico.cern.ch/event/394786/)

Benchmarking Working Group

Fast benchmarks:

➔ Pro and con:

- DB12 much faster than KV
- License issue with Athena framework of KV
- Two flavors of DIRAC benchmark (different conversion factors)

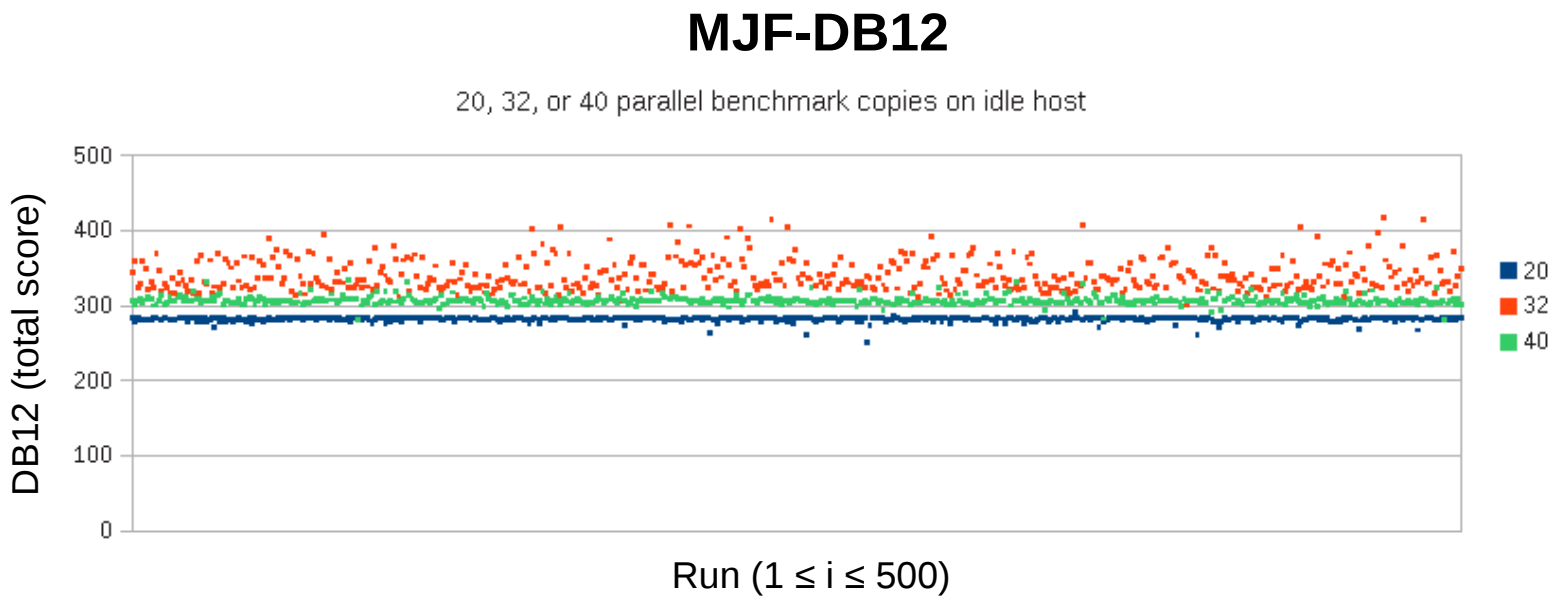
➔ **DB12 as well as KV are suitable candidates to estimate the performance of a job slot**

Benchmarking Working Group

Fast benchmarks:

➔ Still open question:

- Estimation of ‘whole node’ performance, e.g. of VM’s



- See also talk by Andrew McNab about LHCb+DB12

Benchmarking Working Group

Fast benchmarks:

➔ Still open question:

- Estimation of ‘whole node’ performance, e.g. of VM’s
  - See also talk by Domenico Giordano (VM studies)
- How are applications scaling with ‘whole node’ fast benchmark results?
- Work still in progress
  - Batch farm at GridKa reconfigured, there are now WNs with 1, 1.5, 1.6, or 2 job slots per physical core, and the corresponding static benchmark scores available from MJF
  - See experiment reports

Benchmarking Working Group

HS06 issues:

➔ Previous talks, e.g. GDB Sept 2015, have demonstrated broken scaling of application performance (events/s) with HS06 when running on latest processor generations, e.g. Intel Haswell

- What‘s causing this magic performance boost?
- Probable causes are new hardware features like AVX2, hardware random number generator, ...
- See talk by Marco Guerri for details

Benchmarking Working Group

Questions?