

# CPU Performance Study for HEP Workloads with Respect to the Number of Single-core Slots

Matthias J. Schnepf<sup>1,\*</sup>, Max Fischer<sup>1,\*\*</sup>, and Andreas Petzold<sup>1,\*\*\*</sup>

<sup>1</sup>Karlsruhe Institute of Technology

**Abstract.** Many common CPU architectures provide simultaneous multithreading (SMT). The operating system sees multiple logical CPU cores per physical CPU core and can schedule several processes to one physical CPU core. This overbooking of physical cores enables a better usage of parallel pipelines and doubled components within a CPU core. On systems with several applications running in parallel, such as batch jobs on worker nodes, the usage of SMT can increase the overall performance. In high energy physics (HEP) batch/Grid jobs are accounted for in units of single-core jobs. One single-core job is designed to utilize one logical CPU core fully. As a result, Grid sites often configure their worker nodes to provide as many single-core slots as physical or logical CPU cores. However, due to memory and disk space constraints, not all logical CPU cores can be used. Therefore, it can be useful to configure more single-core slots than physical CPU cores but fewer than logical CPU cores per worker node. We have extensively used and studied this strategy at the GridKa Tier 1 center. In this contribution, we show benchmark results for different overbooking factors of physical cores on various CPU models for different HEP workflows and Benchmarks.

## 1 CPU Benchmarks in High Energy Physics

The various computing centers in the Worldwide LHC Computing Grid (WLCG) [1], so-called Grid sites, provide massive computing power to the experiments in high energy physics (HEP). Each Grid site pledges a given amount of computing power to the various experiments each year. HEP-specific benchmarks are used to define quantity in terms of performance provided. At the beginning of the WLCG, the HEP-SPEC06 benchmark was introduced as a measure for the computing power of a given system. [2]. However, current HEP software increasingly uses features of modern CPUs which are not accurately represented by the HEP-SPEC06 benchmark. Therefore, the HEPiX benchmark working group designed a new benchmark based on current HEP workloads to measure the computing performance of a system for HEP software. [3] The current HEP benchmark released in April 2023, the so-called HEPscore23 [4], will be used for the pledges of the Grid sites in the coming years.

At the WLCG Tier 1 center GridKa in Karlsruhe, Germany, we have run the HEPscore23 benchmark on various systems with different settings, see section 3. In contrast to merely

---

\*e-mail: [matthias.schnepf@kit.edu](mailto:matthias.schnepf@kit.edu)

\*\*e-mail: [max.fischer@kit.edu](mailto:max.fischer@kit.edu)

\*\*\*e-mail: [andreas.petzold@kit.edu](mailto:andreas.petzold@kit.edu)

benchmarking the per-slot performance of existing worker nodes as one would do for pledges, we explore the effect of simultaneous multithreading (SMT) on the performance of concurrently running tasks. This is integral to determining how to utilize computing resources best to maximize the HEP workflow performance. In section 4, we present our results of benchmarking itself. Building on this, we describe key settings that we see as beneficial in section 5.

## 2 Simultaneous Multithreading

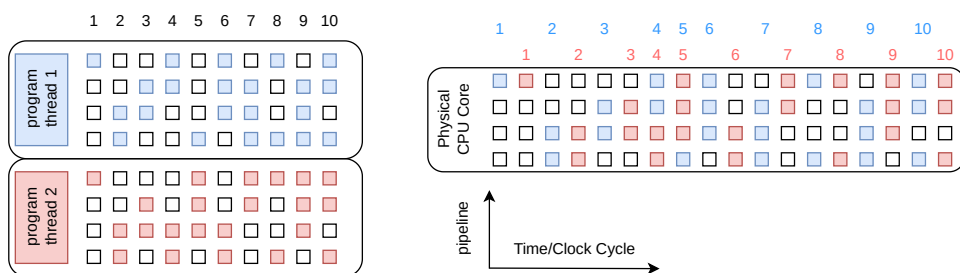
Computer programs are one or more processes running on CPUs. Current CPUs, especially server CPUs, contain several CPU cores, so different processes can run on different CPU cores in parallel. Processes, in turn, consist of one or more sequences of CPU instructions, so-called threads. Current CPU cores have several pipelines for different sets of CPU instructions. Therefore, some CPU instructions within a thread can be executed in parallel to other instructions in different pipelines. Also, several threads concurrently running on the same CPU core, so-called multithreading, can help to utilize the CPU cores better. For example, while one thread waits for data, another thread can run.

An even better utilization of the CPU cores can be achieved by running several threads in parallel on a CPU core. This is possible via simultaneous multithreading (SMT), where the CPU cores have some parts multiple times to run several threads per CPU core in parallel, such as the "Hyper-Threading Technology" from Intel. [5] Figure 1 shows a sketch of two threads running on a CPU with and without SMT. Usually, x86-based CPUs such as the *Intel(R) Xeon(R) E5-2680 v4@2.40GHz* or the *AMD EPYC 7742 64-Core Processor* can run two threads per CPU core. As a result, the operation system (OS) sees more CPU cores, so-called logical CPU cores, than physical CPU cores. The number of logical cores corresponds to the number of threads the system can run in parallel.

It is possible to enable/disable SMT in the BIOS/EFI settings or via the OS, e.g., on Linux on runtime via

```
echo [ off | on ] > /sys/devices/system/cpu/smt/control
```

With disabled SMT, only the physical cores are available for the OS, and the number of logical CPU cores is the same as the number of physical CPU cores.



**Figure 1.** On the left are shown two program threads. Each of them needs 10 clock cycles. The colored boxes represent the used CPU pipeline in a clock cycle. On the right is a physical CPU core shown with four pipelines. Without SMT, both threads would take 20 clock cycles in total to complete. With enabled SMT, shown on the right side, both threads need 18 clock cycles in total since the 4th and 5th steps of both threads can run in parallel on different pipelines. In this example, SMT causes a performance increase of 10%.

### 3 Batch System and Benchmark

We generally define a *single-core slot* on a batch system worker node as the amount of CPU resources that one batch system job needs to utilize one logical core in CPU time. Measuring the utilization via CPU time is advantageous since the real CPU utilization is not easy to measure, and programs may not be able to use some CPU features, such as various pipelines or single instruction multiple data. A machine with 256 logical CPU cores therefore can provide up to 256 single-core slots. The logical allocation of these cores, e.g., as 32 eight-core slots or a mixture of both, is not further relevant.

A naive approach for batch system usage is to provide all logical CPU cores to batch system jobs and thus a number of single-core slots equal to the number of logical cores. However, given that a minimum amount of memory and disk space per slot has to be provided, using all the additional logical cores provided by SMT may incur increased costs without comparable performance gain. The performance increase can be up to 33% on Intel Xeon processors with Intel's Hyper-Threading technology. [5]. The actual performance gain via SMT depends on both the running programs and CPUs. Therefore, it could make sense to provide not all logical cores to the batch system but to determine the point at which performance gain and resource costs are balanced. To determine the performance gain via SMT for HEP software, we use the HEP-Score23 benchmark with and without enabled SMT. For tests without SMT, SMT was disabled on the kernel level.

The number of memory slots on a mainboard and the given sizes of memory modules limit possible memory configurations. Several Grid sites provide computing resources to the different virtual organizations (VOs). The computing resource requirements can be different for the VOs and must be considered in buying hardware. 2 GB per single-core slot is not enough for some VOs, e.g., LHCb requests up to 4 GB memory per single-core slot [7]. 4 GB per single-core slot, on the other hand, is too much for, e.g. Belle II, which requests 2 GB memory per single-core slot [8]. As an example, a system has 2 GB per logical CPU core with SMT enabled. By providing as many single-core slots as logical cores, the 2 GB per single-core slot is not enough for VOs such as LHCb, Using only the physical CPU cores to the batch system provides 4 GB per single-core slot fulfills all requirements, but some memory does not get used. By providing fewer single-core slots than logical cores, it is possible to provide 3.2 GB per single-core slot on average by providing 80 single-core slots on a machine with 128 logical cores and 2 GB per logical core. By a job mixture of 50% Belle II and 50% LHCb, 3 GB per single-core slot on average is sufficient.

To determine the CPU performance of a system with SMT enabled but only a fraction of logical cores provided to the batch system, we apply the HEP-Score23 benchmarks. The HEP-Score23 benchmark runs several instances of exemplary HEP workloads in parallel. These workloads utilize one or four logical cores each, depending on the workload. Per default, as many copies are started as necessary to utilize all logical cores on the system. HEP-Score23 can be configured to run several copies to reproduce the workload of a fixed number of single-core jobs. We benchmarked several systems for different fractions of single-core job workload over the number of physical cores to get the performance for these settings. Several systems have the same configuration (CPUs, Memory, disk space, ...). Each system configuration was benchmarked at least ten times. The results shown are the average results on all machines with the same system configuration.

### 4 Results

All benchmarked systems are configured as dual socket configuration and a minimum of 2 GB memory per logical core. How much performance SMT provides is shown in Table 1. For all

**Table 1.** HEPsScore23 scores for dual socket systems with various CPU models with and without SMT. Systems are ordered by the release date of the CPU model from newest (top) to oldest (bottom). The configurations with disabled SMT are benchmarked with as many HEPsScore23 used cores as physical cores of the respective system. The configurations with enabled SMT are benchmarked with as many HEPsScore23 used cores as logical cores of the respective system.

CPU Model (Dual Socket-System)	SMT off	SMT on	increase (%)
AMD EPYC 7742 64-Core Processor	2499.8	2944.0	17.77
AMD EPYC 7702 64-Core Processor	2228.0	2537.1	13.87
AMD EPYC 7662 64-Core Processor	2182.6	2493.3	14.23
Intel(R) Xeon(R) E5-2680 v4@2.40GHz	525.9	617.3	17.39
Intel(R) Xeon(R) E5-2630 v3@2.40GHz	246.8	292.3	18.46
Intel(R) Xeon(R) E5-2660 v3@2.60GHz	339.9	391.9	15.31
Intel(R) Xeon(R) E5-2665 0@2.40GHz	189.0	221.7	17.26
Intel(R) Xeon(R) E5-2670 0@2.60GHz	202.6	238.8	17.87

Intel CPUs benchmarked, the most performance gain via SMT was observed for the systems with *Intel(R) Xeon(R) E5-2630 v3@2.40GHz* processors at 18.46% performance gain. For all AMD CPUs benchmarked, the systems with *AMD EPYC 7742 64-Core Processor* have the most significant performance gain via SMT at 17.77%. The least performance gain via SMT gets systems with *Intel(R) Xeon(R) E5-2660 v3@2.60GHz* (15.31% increase) and *AMD EPYC 7702 64-Core Processor* (13.87% increase).

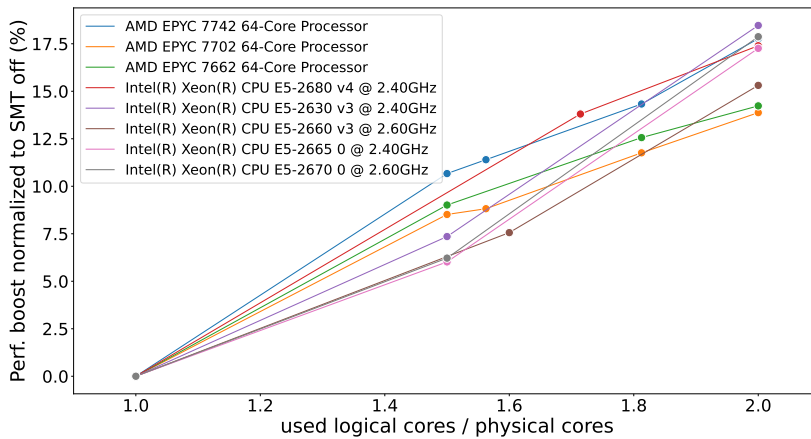
Figure 2 shows the performance gain per logical CPU cores used in percent of the HEPsScore23 result relative to the result with disabled SMT. The tested AMD CPU systems gain the most performance when using 1.5 logical cores per physical core instead of 1.0. The Intel CPU systems show a different behavior. The performance gain from 1.0 to 1.5 used logical cores per physical core is smaller than in the range from 1.5 to 2.0. It is unclear if that depends on the manufacturer or the release date of the CPU since the newer systems in the benchmark pool are equipped with AMD CPUs while the older systems have Intel CPUs.

## 5 Example: GridKa

GridKa provides resources to various VOs with different requirements, which have a fixed share of the computing resources, and the number of running jobs per VO varies over time. Thus, such a usage scenario must be prepared for some uncertainty in the momentary resource demand and should account for some buffer. During March 2023 and June 2023, the average requested memory per single-core slot was 2.4 GB. The current worker nodes at GridKa have 576 GB memory and 128 physical (256 logical) CPU cores; on average, this amounts to 2.25 GB per logical CPU core. However, we provide only 192 single-core slots per worker node, which results in 3.0 GB memory per single-core slot on average. Therefore, we provide, on average, enough memory per single-core slot with some buffer for the variance of job mix and other services on the worker nodes, such as configuration management and monitoring.

## 6 Conclusion

Enabling SMT results in additional CPU performance between 13% to 18% for HEP applications. Depending on the memory and disk requirements, it is more useful to provide as many single-core slots as logical cores or disable SMT completely. Some Grid sites provide as many single-core slots as logical cores to provide all the performance to the jobs; other



**Figure 2.** HEPsScore23 scores performance gain relative to the HEPsScore23 scores with disabled SMT over the ratio of used logical cores by the HEPsScore23 benchmark and physical cores. The benchmarked systems have SMT enabled and thus all logical cores available; only the fraction of logical cores, as shown, was assigned to the benchmark processes themselves.

sites disable SMT completely because the additional costs for memory and disk space do not match the performance gain via SMT. The configuration with fewer single-core jobs than logical CPU cores provides some additional CPU power for other system services, e.g., monitoring, and provides a good method to adjust the average memory per single-core slot. This paper provides some benchmark results from the current HEP benchmark HEPsScore23 for different scenarios.

## References

- [1] WLCG, The Worldwide LHC Computing Grid <http://wlcg.web.cern.ch> accessed 28.8.2023
- [2] HEPsSpec06 specifications, <http://w3.hepfix.org/benchmarking/HS06.html>, accessed 28.8.2023
- [3] Giordano, D., Alef, M., Atzori, L. et al. *HEPiX Benchmarking Solution for WLCG Computing Resources* Computing and Software for Big Science **5**, 28 (2021). <https://doi.org/10.1007/s41781-021-00074-y>
- [4] Randall J. Sobie, *HEPsScore: A new CPU benchmark for WLCG computing*, Proceeding of 26th International Conference on Computing in High Energy & Nuclear Physics, to be published
- [5] Intel *Hyper-Threading Technology*, Intel Technology Journal **6** Issue 01, 1-66 (2002) ISSN 1535766X
- [6] AMD, <https://www.amd.com/en/products/cpu/amd-epyc-7742> (2023)
- [7] European Grid Infrastructure, VO Id Card : lhcb <https://operations-portal.egi.eu/vo/view/voname/lhcb> accessed 28.8.2023
- [8] European Grid Infrastructure, VO Id Card : belle <https://operations-portal.egi.eu/vo/view/voname/belle> accessed 28.8.2023