Contribution ID: **36** Type: **Oral**

CPU-level resources allocation for optimal execution of multi-process physics code

Monday, 24 October 2022 17:00 (20 minutes)

During the LHC LS2, the ALICE experiment has undergone a major upgrade of the data acquisition model, evolving from a trigger-based model to a continuous readout. The upgrade allows for an increase in the number of recorded events by a factor of 100 and in the volume of generated data by a factor of 10. The entire experiment software stack has been completely redesigned and rewritten to adapt to the new requirements and to make optimal use of storage and CPU resources. The architecture of the new processing software relies on running parallel processes on multiple processor cores and using large shared memory areas for exchanging data between them.

Without mechanisms that guarantee job resource isolation, the deployment of multi-process jobs can result in a usage that exceeds those originally requested and allocated. Internally, jobs may launch as many processes as defined in their workflow, significantly higher than the number of allocated CPU cores. This freedom of execution can be limited by mechanisms like cgroups, already employed by some Grid sites, however these are a minority. If jobs are allowed to run unconstrained, they may interfere with each other in terms of the simultaneous utilization of the resources. Constrainment mechanisms in this context improve the fairness of resource utilization, both between ALICE jobs and towards other users in general.

The efficient use of the worker nodes' cache memory is closely related to the CPU cores executing the job. An important aspect to consider is the host architecture and the cache topology, i.e. cache levels, size and hierarchical connection to individual cores. Memory usage patterns of running tasks, the memory and cache topologies and the chosen CPU cores to constrain the job to influence the overall efficiency of the execution, in terms of useful work done by unit of time.

This paper presents an analysis of the impact of different CPU pinning strategies on the efficiency of the execution of simulation tasks. The evaluation of the different configurations is performed by extracting a set of metrics tightly related to job turnaround and efficient resource utilization. The results are presented both for the execution of a single job on an idle machine and for whole node saturation, analyzing the interference between jobs. Different host architectures are studied for a global and robust assessment.

Significance

Efficient use of resources by multi-threaded physics applications on heterogeneous hardware.

References

Experiment context, if any

ALICE simulation, processing and analysis code

Primary author: BERTRAN FERRER, Marta (CERN)

Presenter: BERTRAN FERRER, Marta (CERN)

Session Classification: Track 1: Computing Technology for Physics Research

Track Classification: Track 1: Computing Technology for Physics Research