Parallel Computing of ATLAS Data with PROOF at the Leibniz-Rechenzentrum Munich

P. Calfayan, J. de Graat, G. Duckeck, J. Ebke, J. Elmsheuser, C. Galea, J. Kennedy, C. Kummer, C. Mitterer, D. Schaile, C. Serfon , and R. Walker

The PROOF (Parallel ROOT Facility) library is designed to perform parallelized ROOT-based analyses with a possibly heterogeneous cluster of computers. Its particularity lies in the fact that it allows an interactive utilization. With the forthcoming start-up of the Large Hadron Collider, the ATLAS experiment will have to take up the challenge of processing the huge expected amount of data. Since the collisions recorded are independent of one another, ATLAS analyses can highly benefit from the parallelization provided by PROOF.

A PROOF cluster hosted at the Leibniz Rechenzentrum (LRZ) and consisting of a scalable amount of up to ten worker nodes (Opteron) has been exploited in order to conduct the performance tests in the case of interactive ATLAS analyses. Each worker node provides four processing cores running at 2.6 GHz that are associated with 8 GB of RAM.

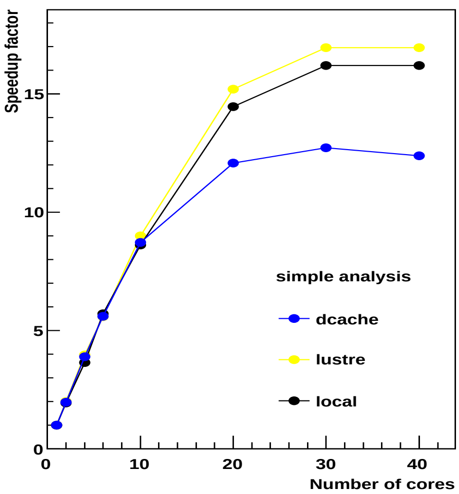

Fig. 1: *Speedup factor of an I/O-dominated (simple) analysis running over input files in native ROOT format as function of the number of cores available in the PROOF cluster. Storage strategies based on local files, Lustre and DCache are compared.*

Scenarios of various complexities have been considered to exercise PROOF with ATLAS data and evaluate its utilization in actual conditions. The investigation of the PROOF performance at LRZ described in [1] focused on: varying the number of parallelized processing units, the amount of simultaneous users, and the type of the files storage. Strategies based on local files, dCache, and Lustre had been compared. This study has been updated with a new hardware configuration for the DCache alternative, based on a RAID-6 array and a 10 GB switch. In Figure 1 the storage strategies are compared using analyses dominated by the data-transfer (I/O) rate and running over native ROOT input files which contain events with sizes from 1 to 10 KB (referred to as D^3PD files). Events are processed by PROOF using a CINT dictionary. The discrimination between the storage alternatives is represented in terms of speedup factors, as defined by the ratio between the processing time required when utilizing one single core and that required when requesting n cores. Comparable performances have been found for all the storage strategies tested, with still a slight disadvantage for DCache.

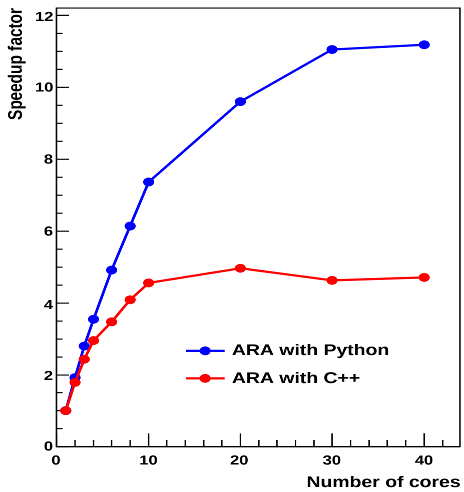

Fig. 2: *Speedup factor of a CPU-dominated (complex) analysis running over ATLAS pool input files as function of the number of cores available in the PROOF cluster. The use of either C++-compiled or Python code is compared.*

The performance of PROOF has also been tested with ATLAS pool files (AOD) as input. This format gathers events of nearly 100 KB, and can be read under ROOT through a specific wrapper from the ATLAS Athena framework, namely the package AthenaROOTAccess. A CPU-dominated analysis has been set up using either C++-compiled or Python code, and is run with PROOF over AOD files containing nearly 12500 $W \to \mu\nu$ events. The input files are stored with the Lustre filesystem. The loading of the ATLAS pool libraries with PROOF is made possible by the utilization of a REFLEX dictionary, which is able to handle advanced C++ structures. Figure 2 illustrates the comparision between both programming languages in terms of speedup factors. The C++-compiled version turns out to be more sensitive to the I/O limitation, which is interpreted by the fact that it allows a better CPU performance than its Python counterpart.

The flexibility offered by REFLEX dictionaries for the use of C++-compiled analyses and the performance of the storage strategies tested show that PROOF in association with AthenaROOTAccess is well suited for interactive analyses with typical ATLAS data formats.

References

[1] M. Schott, *et al.*, MLL Annual report 2007, p. 99.