



Contribution ID: 278

Type: Oral presentation to parallel session

ATLAS Offline Software Performance Monitoring and Optimization

Thursday 17 October 2013 13:53 (22 minutes)

In a complex multi-developer, multi-package software environment, such as the ATLAS offline Athena framework, tracking the performance of the code can be a non-trivial task in itself. In this paper we describe improvements in the instrumentation of ATLAS offline software that have given considerable insight into the performance of the code and helped to guide optimisation.

Code can be instrumented firstly using the PAPI tool, which is a programming interface for accessing hardware performance counters. PAPI events can count floating point operations, cycles and instructions and cache accesses. Triggering PAPI to start/stop counting for each algorithm and processed event gives a good understanding of the whole algorithm level performance of ATLAS code.

Further data can be obtained using pin, a dynamic binary instrumentation tool. Pintools can be used to obtain similar statistics as PAPI, but advantageously without requiring recompilation of the code. Fine grained routine and instruction level instrumentation is also possible. Pintools can additionally interrogate the arguments to functions, like those in linear algebra libraries, so that a detailed usage profile can be obtained.

These tools have characterised the extensive use of vector and matrix operations in ATLAS tracking. Currently, CLHEP is used here, which is not an optimal choice. To help evaluate replacement libraries a testbed has been setup allowing comparison of the performance of different linear algebra libraries (including CLHEP, Eigen and SMatrix/SVector). Results are then presented via the ATLAS Performance Management Board framework, which runs daily with the current development branch of the code and monitors reconstruction and Monte-Carlo jobs. This framework analyses the CPU and memory performance of algorithms and an overview of results are presented on a web page.

These tools have provided the insight necessary to plan and implement performance enhancements in ATLAS code by identifying the most common operations, with the call parameters well understood, and allowing improvements to be quantified in detail.

Author: MANDRYSCH, Rocco (University of Iowa (US))

Co-authors: SALZBURGER, Andreas (CERN); RITSCH, Elmar (University of Innsbruck (AT)); STEWART, Graeme Andrew (CERN); KABRA, Gunjan (Mody Institute of Technology and Science (IN)); CHAUHAN, Neelima (Mody Institute of Technology and Science (IN)); VAN ELDIK, Niels (CERN); LANGENBERG, Robert Johannes (Technische Universitaet Muenchen (DE)); VITILLO, Roberto Agostino (Lawrence Berkeley National Lab. (US)); SEUSTER, Rolf (TRIUMF (CA)); Dr KITTELMANN, Thomas (ESS - European Spallation Source (SE))

Presenter: MANDRYSCH, Rocco (University of Iowa (US))

Session Classification: Software Engineering, Parallelism & Multi-Core

Track Classification: Software Engineering, Parallelism & Multi-Core