

I/O in the ATLAS multithreaded framework

Tuesday 10 July 2018 11:00 (15 minutes)

Scalable multithreading poses challenges to I/O, and the performance of a thread-safe I/O strategy may depend upon many factors, including I/O latencies, whether tasks are CPU- or I/O-intensive, and thread count.

In a multithreaded framework, an I/O infrastructure must efficiently supply event data to and collect it from many threads processing multiple events in flight.

In particular, on-demand reading from multiple threads may challenge caching strategies that were developed for serial processing and may need to be enhanced.

This I/O infrastructure must also address how to read, make available, and propagate in-file metadata and other non-event data needed as context for event processing.

We describe the design and scheduling of I/O components in the ATLAS multithreaded control framework, AthenaMT, for both event and non-event I/O.

We discuss issues associated with exploiting the multithreading capabilities of our underlying persistence technology, ROOT,

in a manner harmonious with the ATLAS framework's own approach to thread management.

Finally, we discuss opportunities for evolution and simplification of I/O components that have successfully supported ATLAS event processing for many years

from their serial incarnations to their thread-safe counterparts.

Authors: VAN GEMMEREN, Peter (Argonne National Laboratory (US)); MALON, David (Argonne National Laboratory (US)); CRANSHAW, Jack (Argonne National Laboratory (US)); NOWAK, Marcin (Brookhaven National Laboratory (US))

Presenter: CRANSHAW, Jack (Argonne National Laboratory (US))

Session Classification: T5 - Software development

Track Classification: Track 5 –Software development