20th International Conference on Computing in High Energy and Nuclear Physics (CHEP2013)

Contribution ID: **70** Type: **Poster presentation**

go-hist: a multi-threaded Go package for histogramming

Monday, 14 October 2013 15:00 (45 minutes)

Current HENP libraries and frameworks were written before multicore systems became widely deployed and used.
From this environment, a 'single-thread' processing model naturally emerged but the implicit assumptions it encouraged are greatly impairing our abilities to scale in a multicore/manycore world.

Thanks to C++11, C++ is finally slowly catching up with regard to concurrency constructs, at the price of further complicating the language and its standard library.
Instead, the approach of the Go language is to provide simple concurrency enabling building blocks, readily integrated into the language, in the hope that these goroutines and channels will help to design, write and maintain concurrent applications.

To further investigate whether Go is a suitable C++ and python replacement language for HENP, we developed go-hist, a multi-threaded Go package for histogramming.

We will first present the overall design of the go-hist package and the various building blocks (goroutines, channels and wait-groups) provided by the Go language which are then leveraged by this library. A special emphasis will be put on the current SIMD support available in Go and thus how vectorization can be leveraged in histogramming code: a cornerstone for automatic performance scalability on the ever-wider-registers architectures the future has in store.

Then, I/O considerations, such as read/write performances, ease of serialization and disk sizes will be discussed, for each of the currently implemented backends (protobuf, gob and json.)

Finally, comparisons with ROOT, Java ROOT and inlib/exlib (an AIDA implementation in C++) performances will be presented.

Summary

Primary author: Dr BINET, Sebastien (IN2P3/LAL)

Presenter: Dr BINET, Sebastien (IN2P3/LAL)

Session Classification: Poster presentations

Track Classification: Software Engineering, Parallelism & Multi-Core