



## Session Summary

# Libraries and Toolkits for a Parallel World

[https://indico.fnal.gov/sessionDisplay.py?  
sessionId=3&confId=6138#20130204](https://indico.fnal.gov/sessionDisplay.py?sessionId=3&confId=6138#20130204)

Benedikt Hegner  
PH/SFT CERN

# Parallelization Potential in ROOT Math libraries

- **Vectorization of math libraries shows big potential**
  - Distributing Vc library with ROOT soon
  - would have an immediate impact on experiments' code performance
  - Interfaces to support auto-vectorization
- **Plans to make tools and algorithms thread-safe and concurrent**
  - Prototypes based on OpenMP
  - Would like to take advantage of C++11
  - At the same time, code improvement planned
- **Concrete ideas and plans, but a lot of work ahead**
  - RooFit may be one of the more complicated problems

# ROOT, Threading and Parallelism

- **ROOT has it's own coarse-grained threading model**
  - Big threading-obstacle CINT replaced by Cling soon
  - OS abstraction for thread handling; to be replaced by C++11
  - More for protection rather than for exploiting multiple threads
- **Concrete ideas for exploiting parallelism for I/O**
- **Ideas to provide similar functionality as TBB**
  - E.g. concurrent containers, task management
  - Might result in a lost investment
- **Concrete requirement agreement with experiments needed**
  - Compatibility with experiments' approaches to be proven
  - Let's take advantage of the discussion session!
- **PROOF as orthogonal approach for filling multiple cores**
  - Very successful

# Geant4 Parallelism

- **Lots of effort invested into concurrency**
- **Geant4-MT prototype to become part of production version by Dec 2013**
  - Including recommendations, guidelines and examples for users
- **Infrastructure to a good fraction made thread-safe or on its way**
  - Tools for code and runtime checking
  - Impact on user-code rather little (compared with the changes underneath)
- **Adaption for TBB for well in phase with experiments**
  - Rather concrete plans for integration with experiments' frameworks
- **Component internal parallelism and GPU usage**
  - Discussed in more detail in the other sessions