Track 5: Software Development

Highlights

Tina Cartaro

for the Track 5 conveners: Alberto Aimar, C.C., Florian Uhlig
and with contributions from the session chairs
**Track 5 Sessions**

- 42 talks split in 7 sessions very roughly organized by topic
- 38 poster contributions in the 2 poster sessions
- Track 5 charge:
  - Software development process and tools; analysis tools and techniques; algorithms in the software development; visualization technology; frameworks for event processing; event data models; processor architectures; vectorization and parallelization; programming techniques and tools; software testing and quality assurance; configuration management; software build, release and distribution tools; documentation.

<table>
<thead>
<tr>
<th>5.1 Physics application</th>
<th>5.2 ROOT</th>
<th>5.3 Performace, metrics, etc</th>
<th>5.4 Processors and architectures</th>
<th>5.5 Software Development</th>
<th>5.6 Visualization</th>
<th>5.7 Big Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>Chair: Steffen Luitz (SLAC)</td>
<td>Chair: Andreas Petzold (KIT)</td>
<td>Chair: Jakob Blomer (CERN)</td>
<td>Chair: Malachi Schram (PNNL)</td>
<td>Chair: Alberto Aimar (CERN)</td>
<td>Chair: Andrea Dotti (SLAC)</td>
<td>Chair: Marcus Ebert (U. of Edinburgh)</td>
</tr>
</tbody>
</table>
AN IMPRESSIVE WEALTH OF CONTRIBUTIONS

61 - Model-independent partial wave analysis using a massively-parallel fitting framework
   Liang Sun et al.

100 - POGIO - Applying plain-old-data for defining physics data models
   Benedikt Riegler et al.

237 - Tracking Machine Learning Challenge
   David Rousseau et al.

952 - Using computing models from particle physics to investigate dose-toxicity corre...
   Karin Harrison et al.

954 - Deep-Learning Analysis Pipelines on Raw HEP Data from the Daya Bay Neutrino Experiment at NERSC
   Mr. Evan Rachal et al.

952 - Neutrino Identification With A Convolutional Neural Network in the NOVAta Detectors
   Adam Aurisano et al.

157 - Status and Evolution of ROOT
   Axel Naumann

156 - ROOT and new programming paradigms
   Philippe Coval et al.

110 - Optimizing ROOT's Performance Using C++ Modules
   Vassil Georgiev Vasilev et al.

258 - Expressing Parallelism in ROOT
   Danilo Piparo et al.

250 - The New ROOT Interface: Jupiter Notebooks
   Danilo Piparo et al.

311 - New Machine Learning Developments in ROOT
   Dr. Sergei Gleyzer et al.

502 - Exploring Compression Techniques for ROOT IO
   Brian Paul Bockelman et al.

9 - RootJS: Node.js Bindings for ROOT 6
   Dr. Marek Szuba et al.

77 - Identifying memory allocation patterns in HEP software
   Sami Karna et al.

158 - Computing Performance of Geant4 Physics Models
   Soon Yong Jun et al.

149 - How To Review 4 Million Lines of ATLAS Code
   Graeme Stewart et al.

222 - Application of econometric and ecology analysis methods in physics software
   Mincheol Han et al.

229 - First results from a combined analysis of CERN computing infrastructure metrics
   Dirk Duettmann et al.

62 - CERN openlab Research Technologies That Might Become Game Changers in Software Development
   Fons Rademakers

306 - Software Quality Control at Belle II
   Timothy Gebhard et al.

316 - Design and Execution of make-like distributed Analyses based on Spotify’s Pipelining Package Igu
   Marcel Rieger et al.

393 - Accelerating Navigation in the VceGeom Geometry Modeler
   Sandro Christian Werner et al.

413 - Machine Learning with TensorFlow as an alternative to TMVA
   Prof. Martin Sevor et al.

436 - Parallel Monte Carlo search for Hough transform
   Raul Cardoso Lopes et al.

114 - Event visualization in ALICE - current status and strategy for Run 3
   Jeremi Nedelcheva et al.

117 - An interactive and comprehensive working environment for high-energy physics software w...
   Mr. Nils Braun et al.

216 - Event visualization in ATLAS: current software technologies / future prospects and trends
   Riccardo Maria Bianchi et al.

449 - A browser-based event display for the CMS Experiment at the LHC using WebGL
   Thomas Mc Cauley

551 - Integrating Visualization Applications, such as ParaView, into HEP Software Framw...
   Adam Lyon et al.

470 - Everware toolkit - supporting reproducible science and challenge driven education
   Alexander Tsihov et al.

80 - A Comparison of Deep Learning Architectures with GPU Acceleration and Their Applications
   Dr. Jin Huang et al.

161 - Multi-threaded Geant4 on Intel Many Integrated Core architectures
   Andrea Dotti et al.

   Ariz Bross Lartza et al.

528 - Challenges in Scaling NLO Generators to Leadership Computers
   Taylor Childers et al.

549 - A programming framework for data streaming on the Xeon Phi
   Sylwah Chapeland

360 - “Big Data” in HEP: A comprehensive use case study
   Oliver Gutsche et al.

69 - Big Data Analytics for the Future Circular Collider Reliability and Availability Studies
   Andrea Aplonova et al.

176 - XRootD Popularity on Hadoop Clusters
   Marco Monni et al.

231 - Hadoop and friends - first experience at CERN with a new platform for high throughput analysis steps
   Dirk Duettmann et al.

215 - Big Data Analytics Tools as Applied to ATLAS Event Data
   Ilja Vukotic et al.

283 - Developing and optimizing applications for the Hadoop environment
   Prasanth Kotturi et al.
The Path to the Next Level Up

• The CPU clock speed flat-lined years ago, the level of parallelization in hardware has increased, and the complexity of the HEP (and non-HEP) applications has grown. Even more growth is expected in the coming years together with the growth of the datasets scale

• HEP community is having a deep look at what industry is doing and the tools and the hardware/platforms that are available on a wider scale outside of HEP and in doing so more fields are looking into what we have to offer
  – New possibilities also for outreach, education, and preservation

• The tools we are most familiar with have become more mature all the while trying new ways to improve and interface with the latest technologies
  – Sustainable software development
5.1 Physics Applications and Machine Learning

- **GooFit**: a massively-parallel fitting framework for partial wave analysis on GPUs
- **New simplified event data aiming at FCC, PODIO**: supports vectorization, automated code generation, C++ and Python bindings, thread-safe, different I/O layers.
- **Medicine**: dose-toxicity correlations in cancer radiotherapy
  - Online measurement of delivered dose at millimeter-scale volumes and correlate with side-effects
  - **Software framework inspired by Gaudi/Athena**, job management with Ganga and started to work on **GEANT-based simulation** of treatment setup
  - Large processing requirements by cancer research standards
- **A.I. techniques can help to solve extremely complex problems**
  - First use of **convolutional neural networks** for a HEP result in neutrino appearance analysis at NOvA
  - Raw data feature extraction at Daya Bay with **deep machine learning**
  - **The TrackML Challenge**
5.2 ROOT EVOLUTION

- **ROOT is evolving** along two main directions: development (with a substantial rewriting) of ROOT core and its components and integrating ROOT with other tools and frameworks

- "doxygen", i.e. documentation, is the main keyword in commits

- New C++/programming paradigms will improve performance and user experience (e.g. the compilers will be more efficient in detecting errors)

- Gradual transition to new interfaces for **ROOT7**
  - New **ROOT::Experimental** name space for testing and development in closer collaboration with users

- **Using C++ modules** to improve compilation time, in collaboration with Google engineers
  - Preliminary result: **21% compile time reduction**. A lot of old code is being revisited and fixed or improved.

- **Multi-threading/multi-processing support**: parallel processing of branches and entries supported via high-level interface to implicit parallelism

- **Exploring new technologies** like Apache Spark and CERN SWAN service for parallel processing

- **ROOT integration with Jupyter** notebooks: focus on interactivity for machine learning and analysis
  - ROOT tutorial available as notebooks.

- **TMVA** with interfaces to many external toolkits (R, scikit-learn, TensorFlow, ...) and high-performance **GPU integration for Deep Learning**

- **node.js integration with ROOT** allows access to all **ROOT features from complex web applications**
5.3 PERFORMANCE, ANALYTICS, ...

- Even our **largest codes are not unmanageable**: ATLAS performed a multi-months design review of the entire code base.

- We can treat our large **code bases as data analytics problem**. There are powerful tools to gather software metrics both statically and at runtime; it helps us to decide where to put coding effort.

- New "FOM (Find Obsolete Memory) Tools" is a powerful utility to understand **memory allocation patterns in real-world applications**: they point out a number of seemingly low-hanging fruits such as frequent, short-lived, and small allocations that can probably be avoided.

- **Systematic data analytics** of running jobs in data centers can possibly replace synthetic benchmarks (which not necessarily reflect application usage patterns); we should be soon able to somewhat predict job performance from job features.
5.4 Processors, Architectures, ...

- Processors trends
  - Many-core and vector CPSs, Next generation ARM, PowerPC, Nvidia P100 (after K40)

- ...And applications
  - Geant4 multithread on MIC
  - Alice processing pipeline on MIC
  - NLO Generators scaling up to top computers

Multi-threaded Geant4 on Intel Many Integrated Core architectures

Starting from Geant4 Version 10.x: focus on parallelism
- Multi-threading: memory reduction (factor 10 w.r.t. multi-process) achieved
- Load-free event loop: almost perfect scalability (90% linearity) with large number of threads
- Hybrid MPI/MT jobs: achieved on very large number of workers (286k total threads)
- Test jobs on hybrid host (KNL) show good results

KNC supported since few years, preliminary tests on KNL confirm scaling and memory reduction
- Factor 3 speedup w.r.t. KNC achieved
- Simplified workflow and access to large DDR3 memory: more complex jobs possible

Started testing on SuperComputers with massive parallelism (MIRA@ANL):
- With more than 100K workers U/G's limiting scaling (not specific to G4): plans to mitigate effect are in place
- Next generation of SuperComputers based on KNL: Theta@ANL, Cori@NERSC
- Getting ready to run complex simulations

http://www.geant4.org

THE NEW ERA: THE CHANGE

The on-chip power density limitations are driving the computing market towards a greater variety of solutions
- ARM announced ARMv8.2(0,1,2) which covers server market
- Designs with up to 64 cores announced & to be used in supercomputers
- ARM recently announced Scalable Vector Extension (SVE) for HPC
- Up to 2K vector sizes & binary compatibility
- IBM announced OpenPOWER Foundation which allows other companies to build PowerPC CPUs
- POWER9 (based on PowerISA 3.0) is the first one after OpenPOWER Foundation was launched
- IBM recently announced POWER8+ with NVLink 1.0 support
- Intel recently announced KNL (many-core and vector CPU) with legacy support for x86_64

Motivation
- As we have seen in many talks at CHEP, LHC computing needs are expected to grow by more than a factor of 60 through the HL-LHC.
- Any core cycle is added to an HPC cores up a cycle on the Grid.
- NLO generators like Sherpa & MadGraph5_AMC@NLO, involve computationally intensive integrations.
- In Sherpa, W+S+J with 1% precision requires a 16-core Intel Xeon 6.5days to compute.
- Increased multiplicity and going to NNLO will continue to increase the compute time of these calculations.
- Scaling codes on leadership machines reduces the time to get results and these optimizations typically result in better efficiency on all systems.

What’s a Xeon Phi ?
- Device from Intel
- Intel Many Integrated Cores (MIC)
  - KNC (2013), KNL (2016)
- Form factor
  - (big) PC-E "co-processor" (KNL, KNC, etc)
  - Standalone systems, integrated fabric (KNL)
- Specifications highlights
  - 57-72 physical cores
  - 40 cores, based on Pentium (KNC) / Atom (KNL)
  - 4-way multithreading
  - Moderate core clocking (1.0-1.5 GHz)
  - Large vectors (512)
  - Fast on-chip MCDRAM (KNL, 8-16GB)
  - Power 200-300 W
- Potentially interesting for some processing tasks

> cat/proc/cpuinfo | grep processor | wc -l

C. Cartaro

SLAC National Accelerator Laboratory
5.5 **SOFTWARE**

- Software and algorithm optimization, and modern technologies together could make a difference
  - 3D Xpoint NVRAM, XeonPhi, NVidia, ...
  - Structure of Arrays to Array of Structures, ...
  - Vectorization
  - Machine learning with TensorFlow
  - New algorithms (track reconstruction with Hough transform)

- New **computing models** for experiments
  - Modern code management, efficient, robust

- New analysis **workflows**
  - Scalability and complexity must be addressed with new set of tools
5.6 Visualization

- Clear trends toward using **less “in-house solutions” and use best practice:**
  - WebGL and web standards, jupyter / python, Paraview

- Experiments **event display**
  - ALICE: based on AliEVE, very attractive possibilities for outreach, including **augmented reality**
  - Belle II: **jupyter** based notebooks **for full analysis in your browser** (experiment agnostic tools developed)
  - ATLAS: Two separate solutions, ATLANTIS and VP1. VP1-light under development for **standalone “laptop” visualization (and outreach)**
  - CMS: WebGL event display based on web-technologies. Independent of CMSSW allowing usage in **education** context (even w/ limited resources)

- Experiment agnostic tools
  - Paraview: being developed for g-2 a scientific application with advanced capabilities (client scriptable via **python/numpy**), client-server (catalyst) data processing framework (art) pushing objects for visualization
  - Everware: addressing problem of **reproducibility of scientific results**. Use best practice to increase open access by publishing full analysis “containerized” systems
Example, Muon g-2 Geant4 Steps
5.7 Big Data

- Big data tools available in industry come in focus for physics analysis
- Studies for example on usability of data storage in Hadoop, Elasticsearch for indexing of events, and Spark/Scala applications
THANK YOU!

- Thank you to all the speakers, poster presenters, session chairs, session helpers, program committee, and to the conference organizers!