Session Summary: Track 5
Software Development

P. Canal, P. Mendez, M. Pedersen, B. von Haller
Generalities

- High attendance: 80-100 people per session
- 46 talks, 21 posters
- Strong presence of ALICE and LHCb (imminent upgrade)
- Non-LHC presence: Cosmology, NICA, AMS, NOvA, DUNE and an important number of multidisciplinary generic tools
- CMS and ATLAS ramping up towards their respective upgrades and the corresponding computing challenges (resources, performance)
- No blockchain, no Machine Learning, no Artificial Intelligence, no IoT
  But we have CI!
## The sessions in a glance

<table>
<thead>
<tr>
<th>Internal T5 categorization</th>
<th>Convener</th>
<th>CHEP18 Session</th>
</tr>
</thead>
<tbody>
<tr>
<td>Experiments</td>
<td>Patricia Mendez</td>
<td>Monday - 1</td>
</tr>
<tr>
<td>Software Development and Improvements</td>
<td>Barthelemy von Haller</td>
<td>Monday - 2</td>
</tr>
<tr>
<td>Parallelism</td>
<td>Barthelemy von Haller</td>
<td>Tuesday - 1</td>
</tr>
<tr>
<td>Frameworks</td>
<td>Patricia Mendez</td>
<td>Tuesday - 2</td>
</tr>
<tr>
<td></td>
<td>Maiken Pedersen</td>
<td>Wednesday - 1</td>
</tr>
<tr>
<td>Software packaging and build systems</td>
<td>Maiken Pedersen</td>
<td>Wednesday - 1</td>
</tr>
<tr>
<td>Continuous integration</td>
<td>Maiken Pedersen</td>
<td>Thursday - 1</td>
</tr>
<tr>
<td>Performance</td>
<td>Patricia Mendez</td>
<td>Thursday - 2</td>
</tr>
</tbody>
</table>
Session 1 - Experiments: Monday morning

- ALFA: ALICE-FAIR new message queuing based framework (Mohammad Al-Turany)
- The ALICE analysis framework for LHC Run 3 (Dario Berzano)
- From catalogs to cosmology: an automated weak gravitational lensing analysis pipeline (Eric Vaandering)
- Software development for the NICA experiment: MdpRoot&BmnRoot (Konstantin Gertsenberger)
- ATLAS technical coordination expert system (Ignacio Asensi Tortajada)
- The AMI (Atlas Metadata Interface) 2.0 metadata ecosystem: new design principles and features (Jerome Odier)

POSTERS

- A new approach for ATLAS Athena job configuration (Walter Lampl)
- Belle II Documentation Effort using Sphinx (Martin Ritter)
Session 1 - Experiments: Monday morning

- Software experimental frameworks being adapted to their specific Run3 requirements
  - ASPECT REPEATED IN MOST OF THE LHC TALKS - Some highlights during this session:
    - Continuous improvement implemented in a transparent way to the users without service interruption (ALICE)
    - New python implementation accommodate $O(10^{-3})$ elements available in Athena/Gaudi (ATLAS)

- High level of commonalities visible at “vertical” level (different areas of the same experiment)
  - Maximal 1+1 experiment
    - AMI (ATLAS Metadata Interface) 2.0 for ATLAS and ROSETTA
    - ALFA: the new ALICE-FAIR concurrency framework for high quality parallel data processing and reconstruction on heterogeneous computing systems
Session 1 - Experiments: Monday morning (cont.)

- **Homogeneity as consequence of Multi-areas escalation:**
  - Similar abstraction levels, even in online-offline (ALICE)
  - Generic approach implementation for HW multi-platform and multi-languages
  - Selection of specific frameworks towards this aim: e.i. PEGASUS in Cosmology

- **Common multidisciplinary base tools implemented at different levels**
  - Base classes of ROOT application also beyond LHC (NICA presentation)
  - Message-passing structures and memory optimization as common elements
Session 2 - Software development / improvements - Monday afternoon

- A tale of two programs: driving innovation in HEP computing (Federico Carminati)
- Software training for the next generation of Physicists: joint experience of LHCb and ALICE (Dario Berzano)
- Exploring polyglot software frameworks in ALICE with FairMQ and fer (Sebastien Binet)
- HEPCon - A Cross-Platform Mobile Application for HEP Events (Martin Vasilev)
- Application extensibility and customization - Indico's case (Pedro Ferreira)
- Extending ROOT through Modules (Oksana Shadura)
- A Python upgrade to the GooFit package for parallel fitting (Henry Fredrick Schreiner)
- Vectorization of ROOT Mathematical Libraries (Lorenzo Moneta)

POSTERS

- Evolution of the VISPA-project (Benjamin Fischer)
- A Historic Data Quality Monitor (HDQM) tool for the CMS TRACKER Detector (Dimitrios Loukas)
- Improvements to the LHCb software performance testing infrastructure using message queues and big data technologies (Maciej Pawel Szymanski)
Session 2 - Software development / improvements

Underlying questions and trends quite in the same line as the previous session

- How to handle changes in software
- Towards common tools and software

Important conclusions/highlights from these questions

- can we apply a disruptive innovation? → Geant4 vs GeantV (when and how)
- RECURRENT MESSAGE:
  - Education now become crucial as the price to so many fancies innovations (“multi-everything” and modularization approaches)
  - Big, common, effort from ALICE and LHCb → GOOD INITIATIVE
Session 2 - Software development / improvements (cont)

- **Need for Modularization**
  - Extensible and customizable towards a better handling of future changes and reach of new communities
    - Achievable through plug-in systems for both developers and users
      - Indico plugins: extensible by users, UN started using Indico!, we give back to the community
      - ROOT modules & package manager

- **Language generic implementations/interoperability and integration with existing frameworks**
  - Fer example: Go implementation of FairRoot “device”.
  - LHCb software performance testing: Code quality assurance towards a multithreading approach
  - VecCore: simpler programming model for SIMD. It is now integrated in ROOT!
Session 3 - Parallelism - Tuesday morning

- I/O in the ATLAS multithreaded framework (Jack Cranshaw)
- Hydra: A framework for data analysis in massively parallel platforms (Antonio Augusto Alves Junior)
- Supporting Future HEP Data Processing with a Parallelised ROOT (Danilo Piparo)
- Implementing Concurrent Non-Event Transitions in CMS (Christopher Jones)
- Writing ROOT Data in Parallel with TBufferMerger (Guilherme Amadio)
- A modular software framework for test-beam data analysis (Bartlomiej Rachwal)

POSTERS

- Conditions Data Handling In The Multithreaded ATLAS Framework (Charles Leggett)
- Optimising Lattice QCD for GPUs (Waseem Kamleh)
Session 3 - Parallelism

- We conclude during the session that Parallelism is:
  - A “prerequisite” (Danilo’ statement)
    - For code delivering in Run3 and beyond
  - A necessity
  - An ubiquitous problem: Every experiment mentioned it

- However!
  - Nobody says it is easy to implement/use
    - Implicit sometimes in those cases when the code it does it for you
    - Explicit often
  - Definitely it has a pay off in the end (ok, maybe years)
Session 3 - Parallelism

Principal Highlights of the session

- Multi-thread vs multi-process:
  - No separation, People are using both
- Parallelism must be applied at every stage of the dataflow (I/O, processing, analyses)
- Data model has to be adapted to parallelism, including conditions data and their access
- Common solutions are appearing (ROOT, Hydra) but no standard (yet?)
- Dissemination of knowledge on latest programming techniques is key: multi-threading, vectorization, latest C++ standard and many others (again, it already appeared in the previous session)
Session 4 - Frameworks. Tuesday afternoon

- The core software framework for the LHCb Upgrade (Sebastien Ponce)
- Evolution of the ALICE software framework for LHC Run 3 (Giulio Eulisse)
- The ATLAS multithreaded offline framework (Scott Snyder)
- GNA: new framework for statistical data analysis (Anna Fatkina)
- Physics data processing with micro-services (Vandan Gyurjyan)
- A generic data quality monitoring framework for HEP (Remi Ete)
- Performance optimization of the air shower simulation code for the Cherenkov Telescope Array (Johan Bregeon)
- Floating-Point profiling of ACTS using Verrou (Hadrien Benjamin Grasland)
- Beautiful, simple and remote ROOT graphics and GUI (Sergey Linev)
- Exploring server/web-client event display for CMS (Alja Mrak Tade)

POSTERS

- A simplified computing framework for FPGA accelerated workloads (David Nonso)
- RDMA-accelerated data transport in ALFA (Dennis Klein)
- Dcache development and testing in Openstack (Christian Voss)
Session 4 - Frameworks. Tuesday afternoon

- Frameworks modernization as follow up of the previous session towards Run3 for all LHC experiments
  - Application of the parallelization concepts to the frameworks
    - Multithreading, parallelization, expanded use to modern HW and memory use reduction
    - Blending the offline/online traditional roles (ALICE)
    - Code clean-up
    - Knowledge dissemination

- Generic infrastructures for data analysis presented:
  - GNA (Generic Neutrino Analysis) with multicore CPU+GPU support used by JUNO
  - CLARA: SW framework for physics data analysis using (more and more homogeneous) heterogeneous SW/HW structures
    - Similarity with ALFA explained at the 1st session
    - Based on “A Roadmap for HEP Software and Computing R&D for the 2020s”
  - Data quality monitoring for (again) online/offline events → DQM4hep
Session 4: Graphical Frameworks. Wednesday morning

- Full rethinking of the current graphics in ROOT towards a more innovative approach
  - multithreading, remote displays, multiple view, portable
- CMS successfully explored the new web-based ROOT graphics - implemented with Fireworks and ROOT EVE.
Sessions 5.2&6 Software packaging, Build systems and Continuous integration - Wednesday and Thursday morning

- Software Packaging and Deployment in HEP (Benjamin Morgan)
- Spack-Based Packaging and Development for HEP Experiments (Kyle Knoepfel)
- IceCube CVMFS Software and Spack (David Schultz)
- dCache as open-source project showcase for education (Tigran Mkrtchyan)
- Building, testing and distributing common software for the LHC experiments (Patricia Mendez Lorenzo)
- Testing of complex, large-scale distributed storage systems: a CERN disk storage case study (Andrea Manzi)
- Robust Linux Binaries (Guilherme Amadio)
- LHCb continuous integration and deployment system: a message based approach (Stefan-Gabriel Chitic)
- Monitoring LHCb Trigger developments using nightly integration tests and a new interactive web UI (Robert Andrew Currie)
- Continuous Integration service at Fermilab (Vito Di Benedetto)

Posters

- Software packaging and distribution for LHCb using Nix (Chris Burr)
- Automation and Testing for Simplified Software Deployment (Andre Sailer)
- Open Science Grid distributed software modules using Spack (Benedikt Riedel)
Software packaging, Build systems, Continuous integration

Some Keywords of the session

● automation, testing, validation, lots of -ity (scalability, flexibility, reliability, reproducibility, relocatability)

Packaging and deploying needs in HEP

● Tools to build and install packages and all dependencies needed at build and runtime, taking care of inter-dependencies
● Allow multiple versions of software active at the same time
● Packages must be supplied in various formats, tar, rpm, containers and for a wide range of platforms. So far x86_64 architecture focused.
  ○ All communities willing to have stable arm64 builds

HOWEVER: Many different solutions (full or partial) → And that’s the bad news

● Spack, Nix, aliBuild, LCGCMak, Portage (but not Easybuild)
● Everyone is doing different things
● No convergence really seen yet :(
● Lots of hopes in HSF
Common approaches partially achieved:

- IceCube+Spack = True – BUT! dependencies and correct versioning is tricky
- Fermilab+Spack = True – although in early and careful start of relationship – R&D continuing
- SFT + SPACK = TRUE
- LHCb-Nix-Hydra= True and success

HSF (HEP Software Foundation Packaging Group) really needed

- Goal: Common practice, experience and knowledge sharing
- Not yet at point of common practice! But getting there?

Software Quality → Some common aspects to all communities

- Application of modern tools for version control, continuous integration, testing and deployment/orchestration such as Jenkins, GitLab/Hub ++
- Code is being tested for performance, style, they include unit tests, regression tests, functional tests, and some even validation tests
- Some have even gone further and increased user interaction possibilities of the test-outcomes in a web-based test tracking system
Session 7 - Performance: Thursday afternoon

- Retroactive sustainability improvements in the MERLIN particle code (Scott Rowan)
- Data mining techniques for software quality prediction in open source SW: an initial assessment (Mara Canaparo)
- Porting the LHCb Stack from x86 to arch64 (Laura Promberger)
- Implementation of Feldman-Cousins correction and oscillation calculations in the HPC environment for the NOvA & DUNE experiments (Alexandre Sousa)
- The event buffer management for MT-Sniper (Jiaheng Zou)
- A plug-in based approach to data analysis for the AMS experiment of the ISS (Valerio Formato)

POSTERS

- Development of profiling system for low energy physics (Kihyeon Cho, Insung Yeo)
- Shared memory transport for ALFA (Alexey Rybalchenko)
- Performance Analysis of effective symbolic methods for solving band matrix (Milena Veneva)
- Geant4 validation web application (Ivan Razumov, Witold Pokorski)
- Continuous performance benchmarking framework for ROOT (Oksana Shadura)
Session 7 - Performance: Thursday afternoon

- **General message:** The more elements we include in the SW/FW the more quality controls are needed
  - Generic external sources as UK “Software Sustainability Institute” → Merlin++
  - Software quality prediction and data mining studies applied to free data sets (NASA) arriving to ROOT and Geant4

- **Quite interesting talk about LHCb HTL SW portability to ARM64 (important for all communities)**
  - Promising results for arm64
  - Still need to resolve the problem of missing cross-platforms support of vectorization libs

- **Generic approach as baseline of the performante application**
  - Optimization on the same data set operations by several users → AMS approach
  - Central process data loading → user’s code dispatching as plug-ins (nightly analysis trains a la ALICE)
Final messages to bring with

- The experiences acquired during Run1-2 are the basis for the new Run3 software framework designs → Major task for all experiments
  - Optimization of the codes to exploit multithreading, memory consumption, multi-core and heterogeneous SW/HW capabilities
  - No interference with users while developing new frameworks → IN THE SENSE OF NO INTERRUPTING THEIR WORK

- SW dissemination becomes crucial at this stage → Common efforts ongoing

- Generic solutions in terms of data analysis and quality under development
  - Applicable to both online/offline (different roles being abandoned)

- In view of Run 3 and 4 software has to evolve. The change will be mostly incremental with limited impact on the users. Testing and CI are key to secure performance and stability

- Common efforts and software, amongst LHC exp. but also between LHC and non-LHC experiments

- Significant shift toward data centric design and declarative interfaces (ALFA, CLARA, ROOT Data Frame, Task based frameworks)
● **THANKS** to all the contributors to the Track 5 for their great talks and posters and their contributions to this summary

● **THANKS** to the organizers and to the conveners committee for this great CHEP2018 conference

● **THANK YOU** Sofia for your hospitality this week

*See you all at CHEP19 in... ????*