Developing the Roadmap for HL-LHC Software

ECFA High Luminosity LHC Experiments Workshop - 2016
October 3-6 2016, Aix-Les-Bains

Pere Mato / CERN

October 3-6 2016
Coping with HL-LHC Needs

To cope with the expected enormous computing demands for HL-LHC we have two solutions:

* Invest on more computing: more hardware, more centers, …
* Invest on better software
  * or a combination of both

What is better software?

* Better algorithms
* Better adapted to the current and future hardware architectures
* Better optimisations
* Better quality
* Better sustainability
CPU Technology Trends

- Until ~2004 we have had an easy life in HEP software and computing
  - Year after year up to 2x increase in computing capacity thanks to the #transistor/chip (Moore’s law) and higher clock frequencies
  - The same program that in year 1995 was needing 10 seconds, would need 1 second in 2002
- The “easy life” is now over
  - The available transistors are used for adding new CPU cores while keeping the clock frequency basically constant thus limiting the power consumption
- We need to introduce parallelism into applications to fully exploit the continuing exponential CPU throughput gains
Technical Challenges

* Big chunks of the LHC software is more than 20 years old, and some parts require re-engineering and modernization
  * Need to exploit modern hardware (many-core, GPU, etc.) to boost performance
  * Modernize implementations (C++11/14 constructs, use more modern and performant libraries, etc.)
* Many algorithms will need to be re-designed to be run in parallel but integrating them to run in a single application is highly non-trivial
  * It will require new levels of expertise that need to be acquired by the community
* Changes in the code of running experiments must be done gradually whilst preserving the correctness of the physics output
Paradigm Shift

* Most of the scientific software and algorithms was designed for sequential processor in use for many decades and will require significant re-engineering

* Migrating sequential applications to multi-threaded is highly non-trivial
  * Difficult to develop: we not only need to code what needs to be done but also how this is done in parallel
  * Difficult to debug: nasty data race conditions will be difficult to reproduce, and so to fix
  * Difficult to maintain: latent threading bugs may take years to be visible

* The community needs to develop expertise in concurrent programming
  * Similarly to the OOP migration, training will be eagerly needed
Collaboration Challenges

* LHC experiments **cannot afford to undertake this software [r]evolution independently**
  * There are many common software packages that would require common efforts, thus coordination
  * General wish to increase the level of commonality and re-use

* Require the collaboration of the whole HEP community to ensure evolution and sustainability
  * Show a common and coherent roadmap to funding agencies
  * Establish structures to facilitate contributions to the HEP software stack

* The adoption of a **collective response** will help to meet the challenges using available expertise and resources and within the required timescale
Prospects for HEP Software

* Potential gains can be made by exploiting features of today’s CPUs’ micro architecture
  * by making use of vector registers, instruction pipelining, multiple instructions per cycle
  * by improving data and code locality and making use of hardware threading

* New architectures to off-load large computations to accelerators (GPGPUs, Xeon Phi™) or the new integrated architectures with heterogenous processors (AMD)
  * specific memory models will force explicit memory programming
  * new programming languages (Cuda, OpenCL, etc.)
Prospects for HEP Software (2)

* Today multi-core architectures employing $O(10)$ cores are well exploited using a multi-process model (1 job/core)

* However this performance will **not scale** to future generations of many-core architectures employing $O(100)$ cores due to memory issues

  * there are technical issues related to connecting many cores to shared memory that will reduce the amount of memory available to each core
  
  * whereas the memory footprint of HEP code is increasing due to increasing event complexity as the energy and luminosity of the LHC is increased
  
  * in addition, we may see new architectures with non-uniform memory access
Addressing the Challenges

- **HEP Software Foundation (HSF)** as the umbrella for addressing these challenges together!
  - Collection of ideas and proposals in 2014 and **startup-team** formed
  - Kick-off workshop Jan 2015 at SLAC established concrete activities
  - Workshop in May 2016 at LAL to review progress and setting directions

- In addition, the HSF aims at
  - Support **career development** for software and computing specialists
  - Provide a framework for **attracting effort and support** to S&C common projects
  - Provide a **structure to set priorities** and goals for the work
  - Facilitate wider connections; it should be open enough to form the basis for **collaboration with other sciences**
Current Status and Activities

* Sharing expertise
  * Schools, trainings and courses (not always easy to find)
  * Adopting wikiToLearn a platform for training material
  * HEP S&C Knowledge Base
    * Database of software packages, categories, experiments, organisations, languages, meetings, workshops, etc.
  * HSF Technical notes
  * Pursuing a journal on "SW&C for Big Science"
  * Topical fora and working groups in the HSF web

* New hardware architectures and technologies
  * Concurrency forum, evolved into a general software technology forum
  * Usage of resources provided on best-effort basis by e.g. CERN’s TechLab / Openlab
  * Porting to new architectures efforts within the LHC experiments
Software performance

- Simulation: parallelisation of Geant4; GeantV R&D activity
  - HSF is organising "software community meeting" to review the progress made in simulation R&D
- Reconstruction: HSF common tracking SW forum + IML forum
- I/O: parallel ROOT I/O, key-value-store evaluations
- Mathematics: MetaLibm, parallelisation of fitting, etc.
- Ad-hoc improvements and parallelisation in various SW projects
- Performance tools (e.g. igprof, FOM tools)
Current Status and Activities III

- Supporting developers and participating projects
  - Providing best practices to facilitate integration into HEP eco-system
  - Project templates for bootstrapping new projects
  - Development services
  - Help in selecting the proper SW license

- Quite some activity in HSF, even though participation in the startup-team is on volunteer/best-effort level

- Need to allocate soon some dedicated resources to keep momentum and ensure continuity
## HSF Working Groups

<table>
<thead>
<tr>
<th>Working Group</th>
<th>Objectives</th>
<th>Forum - Mailing list</th>
</tr>
</thead>
<tbody>
<tr>
<td>Communication and information exchange</td>
<td>Address communication issues and building the knowledge base Technical notes</td>
<td>hep-sf-tech-forum</td>
</tr>
<tr>
<td>Training</td>
<td>Organization of training and education, learning from similar initiatives</td>
<td>hep-sf-training-wg</td>
</tr>
<tr>
<td>Software Packaging</td>
<td>Package building and deployment, runtime and virtual environments</td>
<td>hep-sf-packaging-wg</td>
</tr>
<tr>
<td>Software Licensing</td>
<td>Recommendation for HSF licence(s)</td>
<td>hep-sf-tech-forum</td>
</tr>
<tr>
<td>Software Projects</td>
<td>Define incubator and other project membership or association levels. Easy-start project templates</td>
<td>hep-sf-tech-forum</td>
</tr>
<tr>
<td>Development tools and services</td>
<td>Access to build, test, integration services and development tools</td>
<td>hep-sf-tech-forum</td>
</tr>
</tbody>
</table>
HSF Topical Fora

- Software Technology Forum
  - Technical issues to embrace new technology in our software
  - Ongoing activity

- Reconstruction Algorithms Forum
  - All matters of event reconstruction and pattern recognition software
  - Several in-person meetings, “Connecting the Dots” workshop

- Machine Learning Forum
  - ML discussions and code development in the context of HEP
  - Development of relevant tools, methodology and applications
Cross-experiments Collaborations

- Experiment frameworks
  - Gaudi (ATLAS, LHCb, FCC)
  - FAIRRoot (FAIR, ALICE)
  - ART (CMS, Neutrino programme)

- Common Conditions Data Project
  - Discussion/cooperation between ATLAS, Belle II, CMS and LHCb

- Common Software Build and Packaging Tool efforts
  - Working group of HSF comparing HEP and non-HEP solutions
  - Starting point was LCG’s Librarians and Integrators Meeting

- Cooperation on Reconstruction Software
  - “Connecting the Dots” tracking workshop extended by HSF session about common tracking implementations

- AIDA2020 (EU funded)
  - DD4hep for detector description (LCD, FCC, potentially LHCb)
  - PODIO data model library (FCC, LCD, potentially LHCb)

- DIANA (Data Intensive ANAlysis) (NSF funded)
  - 4-year project on analysis software, including ROOT and its ecosystem

Examples of cross-experiment collaborations, with involvement or moderation of the HSF - going beyond LHC
Defining Longer-term Strategy

- HL-LHC computing requires a major ‘software upgrade’
- A **Community White Paper** (CWP) on the overall strategy and roadmap for software and computing has been proposed
  - The scope should not be restricted only to HL-LHC
  - However, it can be used to identify research required to prepare the LHC experiment’s TDRs in advance of HL-LHC
  - Some early software components could be built, tested and used by experiments in LHC Run3
- Organised by the HEP Software Foundation (HSF)
- Paper to be delivered by Summer 2017, important ingredient for
  - Future funding opportunities, including a US opportunity for an NSF-funded Software Institute
  - WLCG computing roadmap for HL-LHC
The CWP should **identify and prioritise** the software research and development investments required

- to achieve **improvements in software** efficiency, scalability and performance and to make use of the advances in CPU, storage and network technologies
- to enable **new approaches to computing and software** that could radically extend the physics reach of the experiments
- to ensure the **long term sustainability** of the software through the lifetime of the HL-LHC

We need to **engage the HEP community** in this process through a series of workshops

- Aiming for a broader participation (LHC, neutrino program, Belle II, linear collider so far)
CWP: Getting Organised

* First organisational discussion took place two weeks ago
  
  * Reviewed draft charge, initial **working group organisation** and asked attendees to encourage people in their communities to join up and participate
  
  * Working groups will self-organise, the “**do-oocracy**” determining the proactive people who emerge as conveners
  
  * Created a single CWP mailing list
    
    * [https://groups.google.com/forum/#!forum/hsf-community-white-paper](https://groups.google.com/forum/#!forum/hsf-community-white-paper)
    
    * Please subscribe if you want to participate or follow progress
  
  * A GoogleDoc page will be setup for each WG to start planning and writing
CWP: Getting Organised II

- The next step: Sun Oct 9 during the pre-CHEP WLCG meeting
  - The afternoon is an HSF session, to be devoted mainly to CWP getting organised
  - Flesh out the charges, the initial ideas, plans for the WGs
  - Ideally with early volunteers in at least some WGs having brought some initial written ideas
  - Only a subset of the interested community will be present, asked for Vidyo

- The real launch: a workshop at UCSD San Diego Jan 23-26
  - Start real writing after a few months post-CHEP gestation in the WGs
  - Discussions on more controversial topics, reach consensus
  - Detailed plans and responsibilities for delivering white paper by summer 2017
CWP: Working Groups

<table>
<thead>
<tr>
<th>Working Group</th>
<th>Challenges and Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>Computing models, facilities, technology evolution</td>
<td>range of possible models, costing</td>
</tr>
<tr>
<td>Physics generators</td>
<td>better models, better precision, code optimisations</td>
</tr>
<tr>
<td>Detector simulation</td>
<td>full and fast simulations, hi-pileup environments</td>
</tr>
<tr>
<td>Triggering</td>
<td>algorithms, GPUs and/or FPGAs</td>
</tr>
<tr>
<td>Event reconstruction</td>
<td>new approaches to event reconstruction</td>
</tr>
<tr>
<td>Data access and management</td>
<td>scaling to the exabyte level</td>
</tr>
<tr>
<td>Workflow and resource management</td>
<td>millions of jobs in heterogenous systems</td>
</tr>
<tr>
<td>Data analysis and interpretation</td>
<td>efficient use of many-core, modern techniques</td>
</tr>
<tr>
<td>Software development, deployment and validation/verification</td>
<td>improved modularity and quality, easy contribution</td>
</tr>
<tr>
<td>Data and software preservation</td>
<td>preservation and reuse of data and software</td>
</tr>
<tr>
<td>Visualization</td>
<td>tools for data analysis, education, and outreach</td>
</tr>
<tr>
<td>Careers, staffing, training</td>
<td>perhaps in a separate concurrent white paper</td>
</tr>
</tbody>
</table>

This list will evolve. Additional working groups could be formed if it makes sense (e.g. on specific technology issues)
Summary

- We need as a community to **invest on better software** to cope with the high demands of the HL-LHC
- Existing software needs to be re-engineered, and a lot of new software needs to be developed using new ways: **paradigm shift**
  - The community needs to develop expertise in concurrent programming
- Initiated the **HEP Software Foundation (HSF)** as the umbrella for addressing these challenges together!
  - Need to put dedicated resources soon to keep momentum
- Working on a **Community White Paper (CWP)** to define the strategy and roadmap for long-term software and computing
  - Input for funding opportunities and WLCG roadmap for HL-LHC
  - Call for participation to the defined WGs