Opened discussion

4th workshop on …

(J. Lauret)

Workshop goals

- We had a few goals
  - Have status reports on the common package
  - Bring more (if not all) the LHc experiments on board
  - Bring or keep other communities in (Geant 5, ROOT, …)
- Goals overall met
  - CMS, LHCb joined ATLAS who joined the ALICE/CBM/STAR core activities from past workshops
  - Common package and use seem to converge
    - KFParticles used with success in multiple experiments as well as offline/online – *unified ALICE/CBM package*
    - CA/HLT and CA/offline in use – *possible similarities ALICE/STAR*

Cross collaboration (KFParticles) obviously benefitting all. This stresses & raise again the issue of and need for code repository, common package/toolkit, horizontal communication

KFParticles pushed down – how experiment independent can it be? Do we need scalar double, vector … Can it be a compilation switch via header? The integration will not be that trivial BUT some recent bugs and issues shows (as noted above) that a common package must be the goal.

Workshop goals

- Some issues remain opened
  - Vc in ROOT
  - Documentation, mailing lists, code repository
  - common platforms for testing
- Some seemed "closed"
  - All understand SIMD is important
    - Still confusions between vectors to help SIMDize code versus long vectors versus libraries of vector mathematics

Common platform tried – will require POC from experiments for installing packages / software stacks (non trivial efforts). SFT package possible? (J. will follow up with Pere/Federico → done after workshop and no tight relation for now but each team will follow the other's workshops and efforts and we will see).

Vectors – SSE to AVX improvements not always clear / memory bandwidth, restructuring of code needed? Blocking for bandwidth "impedance match". Also, do more calculations.

Vc in ROOT → Jerome disappeared on afternoon during the workshop / issue is settled (ROOT team will include; workforce for long term maintenance secured)

Opened discussions items

- Vendor locking, library locking still came as topic
- Architecture concerns: Intel Xeon/Phi MIC, ARM, …
- Tuning, compiler options an "art" – is this what all should go through? Common wisdom possible?

- Code needs to run everywhere (LHCb, …)
- Distributed computing and libraries (GPU on Grid …)

- Auto-vectorization – is this really possible in the short or medium term? (stability of coding)

- Geant 5 presented a complex design optimizing resources – is this compatible with CA approach?
- Reproducibility of results (concerns that out of sequence approach may be hard to debug, …)

Even if we have a diversity, still minor variation along the same "theme". We have multiple dimensions and for now, threading & vectors where the sole focus but architecture should be flexible enough for other dimensions. Cannot grow indefinitely – cache, bandwidth, core comm., … Intel commitment on x86 and AMD – no magic recipes but … Intel recommends highly // and highly vectorized (very general guidance).

Auto-vectorization: still needs to be prepared for it (but instabilities may remains / compiler instabilities were discussed and coding variation changing results as well). Questioning short term path. Amount of work required to make code stable via auto-vectorization is large: one needs to think of the data structure (possible gather/scatter) and code flow.

Geant 5 – geani / KF – project for Geant 5. Right now, vector and TGeo not ready. Push down needed (but Tgeo is thread safe).

Next workshop

- Should we find a name for those workshop?
  - "International workshop for future Challenges in Tracking and Trigger Concepts" is long (and not mnemonic)
- What should be our focus?
  - Asking this last time did lead to a focused effort
  - Would be good to have feedback from the "newcomers"
  - Would be good to also retune what the first comers want to see as drivers
- More communities? More topics?
  - Can we scale beyond this team size?
    - My own take: risk of expertise running thin – must have shared efforts
  - Should we expand beyond tracking?
    - Parallelization strategies?
    - Models (Geant 5 example) and global architecture designs?
    - Should we expand the workshops and bring training back?

Title: Not only the long title issue – acronym WFCTTC seem fine with a few (for the lack of a better one).

Do we expand beyond reco? Geant 5 is one example. IO is becoming important … Architecture designs may also come back as a focus (choices are not all compatible / ambitious development in Geant 5 and integration in software stacks as example). General feel is that we should remain focused but invite teams (like Geant5 talk).

Feedback from experiments: Jerome collected many many feedback from all experiments. This would be worth a separate summary.