LIM, January 25, 2021

Connected: Andre Sailer (chair), Dmitri Konstantinov (chair), Ewelina Lobodzinska, Johannes Elmsheuser, Pere Mato, Omar Zapata, Charles Delort, Ari Kraut, Reiner Hauser, Marco Clemencic, Giulio Eulisse, Jakob Blomer, Ben Couturier, Gerri Ganis, Andrei Kazarov, Guilherme Amadio,

Apologies:

Indico: https://indico.cern.ch/event/993184/

Next meeting: February 8, 2021

Invitation for the CVMFS Workshop

Jakob Blomer invites attendees to the virtual CVMFS workshop next week.

The event's indico URL is https://indico.cern.ch/e/cvm21

Status of things

Nightlies

- Upgraded packages for the dev[34] and dev[34]python2 development builds

    ○ delphes (3.4.3pre8) (following change in ROOT), cool/coral 3.3.4

- New package:

    ○ qcachegrind

        ▪ Guilherme notes that kcachegrind can be used from /cvmfs/sft.cern.ch/lcg/contrib/gentoo/linux/startprefix made available via gentooprefix

- Tests:

    ○ Gaudi: 2 (out of ~272) tests are still failing

- dev[34]Cuda

    ○ enabled a Cuda option in vecGeom

- devArm

    ○ Added a number of packages requested by ATLAS (SPI-1785), removed gPerfTools (SPI-1747)

Pere: Gaudi dependency on gPerfTools optional, can it be removed?

Marco: I believe it is optional, disabled with "cmake -DGAUDI_USE_GPERFTOOLS=FALSE"

Johannes: Definitely optional on ARM

Releases

LCG_99:
- Deployed: cc7 (gcc8/10,clang10), cc8 (gcc10), ubuntu2004 (gcc9), mac1015 (clang12), all using python 3.8.6,

LCG_99cuda:
- Deployed for cc7 with gcc8 and python 3.7.6

LCG_98python3_ATLAS_4 with gcc10 (SPI-1784)
- Not working because of jsonmcpp in ROOT

LCG_98python3_ATLAS_5 with gcc10
- jsonmcpp 3.9.3 for gcc10 not compatible with ROOT

- Johannes: not too much pain caused by this missing layer with gcc10

LCG_98python3_ATLAS_6 (for generators) (SPI-1787)
- Ewilina: Standard build with modified generator versions like in Release 21 to run the validation

Comments for the Releases

- Marco: Checks of LCG_99 in LHCb. Did not manage to build the whole stack, found a couple of issues in ROOT, a bug still exists that is fixed in ROOT-master. Issues with minor changes from python 3.7 to 3.8.
  - Surprising: Why clang 10.0.0 instead of clang 10.0.1. troubles with clang 10.0.0 build with gcc8 gave troubles with debug build. With Dmitri agreed to have clang 10.0.01 based on gcc10. Problem is the gcc version to build clang. Gcc10 must be used to build clang.
  - (Also the clang 10.0.0.1 is not being used in dev4)
  - Gerri: The problem is that clang10 is not linking to the newer clang release
  - Pere: Are you testing against dev4 in your nightlies?
  - Pere: Problem that only 1 digit is used for compiler identification
  - Marco: clang 10.0.0 to 10.0.0.1 not binary compatible because of major version change in underlying gcc.
  - Pere: add dependency of clang to gcc to change the hash and force rebuild
  - Marco: ideally have a build with clang8 and clang10, could also jump to clang11 or 12.
  - Pere: Would be good to have a test reproducing the issue to avoid it in the future.

- Marco: undefined symbol in debug build in std::filesystem
- Gerri: Only python3 OK?
- Marco: One simulation application not yet ported, but working on it. One application with issues from py37 to py38. Need better understanding of the issues before making a definitive statement about python2 build requirement
- Johannes: LCG_99 based on gcc9 works with a substack of ATLAS, not yet tested with full Athena

Operating Systems et al.

There was an IT-ASDF meeting discussing the way forward https://indico.cern.ch/event/992276/

Discussion:

- Marco: For physics application the underlying OS does not matter so much.
- Giulio: Hardware support might be an issue. GPUs only really work on CentOS8 because of the drivers. AMD device virtualization does not work inside docker
- Ben: To decrease dependence on OS, one could install only the needed parts for given application from the LCG stack and HEPOSLibs into containers to move from one OS to another, not the whole LCG stack

Other issues:

- Andre: Closing many old issues, please let us know if closing an issue that isn't solved. Please close issues that you know are solved.
- SPI-695: Potential VSIZE reduction for static python linking, about 100 MB of memory difference
- Omar: Request for the installation of the Bokeh library for plotting in notebooks. When exporting Plots to another format, dependency Firefox and geckodriver.
  - Pere: needs libraries or the full browser?
  - Omar: needs the browser, which renders the javascript to create the plots. It is executed on the SWAN container side, not in the browser of the client.
  - → Schedule a meeting in the near future, once some investigations could be done to understand the full requirements
- Marco: For profiling, it might be better to have "-fno-omit-framepointer", which is enabled in debug build. Would it be possible to have an OPT build with this flag? Should be possible to do via the compiler wrapper
  - E.g.: Platform OPT+WFP (with frame pointer)
  - Giulio: For which profiler? libunwind should be able to do things even with omit-framepointer
  - Guilherme: Yes, must set no-omit-framepointer for "perf". Very limited performance penalty.
  - Marco: A dedicated build for profiling would be very useful

- Giulio: Why not keep dwarf information? Debug information should be better nowadays.

- Marco: Need to check with the requester in LHCb, with debug information would need split debug information instead.

- Guilherme: Having both, frame-pointer, and debug information is best. Sometimes perf doesn't find the debug information. Have some slides LINK:

    - https://indico.cern.ch/event/974382/

    - https://amadio.web.cern.ch/amadio/flamegraphs/

    - For flags, "-O2 -fno-omit-frame-pointer -g" is what I've been using for profiling (with optional -march=native)

- Andre: Please open a JIRA ticket

- Pere: Should create a new stack instead

Stakeholder feedback discussion

Johannes: Sounds good. Should give hints about what kind of feedback you would like

Stakeholder feedback

Alice: NTR

Atlas: NTR

CMS: No Connection

LHCb: NTR

SWAN: Working on integration of LCG_99

BE/NXCals:

AOB: