LHCb Software Deployment

pre-GDB
05/05/2020

C.Bozzi, B. Couturier, M. Clemencic,
(for the LHCb Computing team)

Software deployment locations

Location	Characteristics
LHCb trigger farm	LCG supported OS, root access, easy to install all required dependencies
LCG Grid	LCG supported OS, CVMFS Installed, HEPOS_libs installed, singularity available (nearly) everywhere
Institutes clusters	(typically) LCG compatible OS, CVMFS or local installation
HPC Sites	Each one is a different case
Developer machines	Likely a non supported OS, container or VM technology available
Outreach exercises	All OS (including windows) have to be supported, trying to accomodate all hardware (including old machines)

LHCb Production Stack

- Based on LCG releases, with the same OS requirements
  - centos7/gcc9 is currently our main platform
  - Not against the use of spack if adopted by the LCG releases
- Supported instruction sets:
  - x86_64 > SSE4.2
  - Custom build for avx2+fma prepared for use in the farm

    (or any node with that instruction set)
  - Working towards a aarch64 port (being validated, dependent on LCG of course)
- C++17 built with CMake/make/ninja
- Allen trigger application: Cuda 10 with gcc >= 8.2.0

LHCb Production Stack deployment

- Packaging as LCG RPMs (i.e. with fake dependencies)
- Requires HEPOS_libs on top of standard OS
- Installation
  - Local Installation possible using the lbinstall custom tool or yum
  - CVMFS installation is the main deployment platform (c.f. Next slide)
- Satisfied with the deployment of software to CVMFS, but some weak points:
  - Performance of the deployment to the stratum-0

    Good results with tests of a multiple release manager setup backed by S3 for CI
  - Loading many small files in clients can be slow

    e.g. when setting up a conda environment

LHCb CernVM-FS repositories

	Size	GC	Role	Comments
/cvmfs/lhcb.cern.ch	1.5 TB	No	• Production software (Online/Offline)	• Used by Grid and LHCb Trigger farm • A few transactions/day in average • **Crucial for long term software preservation**
/cvmfs/lhcbdev.cern.ch	2.6 TB	Yes	• Continuous integration artifacts and installation • Software under test • Clone of GIT repositories	• Deduplication is really helpful • High turnover: ~3.8 million files per day (~200 GB/day) ***Testing S3 backend + multiple managers to handle the load*** • **Short term interest** • **Crucial for development team**
/cvmfs/lhcb-condb.cern.ch	~ GB	Yes	• LHCb Conditions	• Frequent releases • New volume to decouple the release of the GIT conditions from the long term preservation repository.

* As of 24/01/2018

Matching software and platforms

The LCG software is released for several platforms (combination of architecture, instruction set, system, compiler, compilation flags)

Custom python package performs hardware detection and matches OS with platforms:

- LbPlatformUtils: https://pypi.org/project/LbPlatformUtils/
  - Used by LHCbDirac and the LHCb Environment scripts (LbEnv)
  - Also detects the presence of singularity (>=3)

LHCb Environment (LbEnv) and LHCbDirac use it to select the best platform of an application (and decide whether the use of containers is needed)

Use of VMs and containers in production

Both Virtual Machines and Containers are used for LHCb productions:

- Some sites are configured to run the jobs with A. McNab's Vac or VCycle
  - CernVM is used to run the jobs in that case
- Singularity used to run jobs not supported on the host

  (e.g. SLC5 applications on Centos7, om when the sites uses non RH based linux)
  - Use of singularity v3 with user namespaces
  - OS taken from:
    - /cvmfs/cernvm-prod.cern.ch/cvm3/
    - /cvmfs/cernvm-prod.cern.ch/cvm4/

This ensures consistency with the environment found by jobs running in VMs

Development environment

Developer machines are unlikely to run centos7.

- CernVMs or Docker/Singularity can fill the gap

  Depending on the OS
- Local installations are possible with **lbinstall**

  But CVMFS is available on many OSs making it the popular choice

Specific deployments

- HPC Sites
  - Easy if CVMFS can be installed
  - Harder if not as cvmfs_shrinkwrap may not be the solution
  - We can provide containers, but they are large (10s of GBs)
- “LHCb trigger on USB key”, for use during the HLT farm tender
  - Need to fit the trigger stack in a few GB, no network access are allowed
  - However just the system + Hepos libs are already taking 0.5GB…
  - Used lbinstall in the past (will consider cvmfs_shrinkwrap for the next round)
    - Performed a full install, then removed a lot of unnecessary packages

Analysis environments

- ROOT is at the core of LHCb data analysis
- But we see a trend in the use of python packages to perform/complete the analysis (facilitated by tools such as uproot)
- ROOT is now available in conda-forge

Conda environments offer the possibility users to start the analysis on their laptops, then run the job on the grid

  - Deploying Conda Environments on CVMFS can help in this case (20+ users have already been using an experimental deployment)

User containers/environments

LHCb users use Ganga to submit their jobs. It makes it possible to download and extract a container before running the job.

- Not efficient if each job downloads its own container…
- Publication of the container in CVMFS would be more efficient

We are also thinking about making conda environment available on demand on CVMFS

- Lighter than extracting whole containers
- Not as generic, of course...

LHCb Software Preservation

- We keep on /cvmfs/lhcb.cern.ch all releases of the software ever used to generate productions
- Counting on containers/VMs to able to rerun old versions of the software

CernVM3 and CernVM4, as well as the associated CVMFS repositories in the system, are crucial to that strategy

Open data & Outreach Software

Constraints are very different from the production and analysis

- For open data, VMs with tools can be provided with the necessary tools
- For outreach we want to reach schools with very different machines/OS with limited capacity
  - Current masterclass exercise is a ROOT application that they have to install locally (on any platform supported by ROOT including windows)
  - We have a WebGL version that has been developed but not yet validated by the IPPOG

Outlook & conclusion

- We intend to **follow the evolution of the LCG stack** deployment
  - Will use spack for deployment if adopted for the LCG stack
- **CVMFS is crucial to our operations and preservation** of the software
- Using **containers as much as possible**
  - Keen to build **small** containers with our stack, to run where CVMFS does not reach
- **We use CernVM3 and CernVM4** to run current and old jobs
  - OS installation on CVMFS used by singularity
- **Investigating analysis in conda environment**
  - As well as the automation of their deployment in CVMFS
- Outreach using either VMs or Web technologies when possible