

# HEP\_OSlibs

## status and outlook

Andrea Valassi (CERN IT-SC-RD)

PreGDB Meeting on Software Deployment, 5<sup>th</sup> May 2020  
(and next week: Librarian and Integrators Meeting, 12<sup>th</sup> May 2020)

See also my talks at the May 2018 LIM workshop, June 2017 LIM and July 2016 LIM

Further in the past: see Pere Mato's and Oliver Keeble's talks at the June 2009 GDB

# One-slide overview and summary

Present status and *open questions* both at PreGDB and LIM

This talk provides some background information

## HEP\_OSlibs for Grid deployment (discussion at WLCG meetings, PreGDB)

- New context: container deployment on Grid sites
  - *Do Grid sites still need a HEP\_OSlibs? For which experiments?*

## HEP\_OSlibs for the LCG software stack (discussion at recent LIM meeting)

- New O/S platforms upcoming (CentOS8, Ubuntu20)
  - *New versions for el8, ubuntu2004? Drop el6, ubuntu1604?*
- Improve contents: runtime, developer packages, EPEL?
  - *Should we split runtime and developer meta-rpms?*
  - *Should we allow the inclusion of EPEL packages?*
- New model for software deployment? (Spack?)
  - *Do developers still need a HEP\_OSlibs (or an LCG stack)?*

# What is HEP\_OSlibs?

- **A meta-package**: *'yum install HEP\_OSlibs'* on RedHat is equivalent to *'yum install alsa-lib-devel.x86\_64 ... zsh.x86\_64'*
  - a list of packages
  - without a specific version number
  - taken from the official O/S repos (not from EPEL)
  - “the O/S level dependencies of the LHC experiment software stacks”
- **A more precise definition of what a “supported O/S” is**
  - Recent Linux default O/S's are pretty “thin” (e.g. CC7 thinner than SLC6)
- Latest version on RedHat7: 7.2.14-1.el7
  - [https://gitlab.cern.ch/linuxsupport/rpms/HEP\\_OSlibs/-/blob/7.2.14-1.el7/HEP\\_OSlibs.spec](https://gitlab.cern.ch/linuxsupport/rpms/HEP_OSlibs/-/blob/7.2.14-1.el7/HEP_OSlibs.spec)
  - [https://gitlab.cern.ch/linuxsupport/rpms/HEP\\_OSlibs/-/blob/7.2.14-1.el7/dependencies/HEP\\_OSlibs.x86\\_64.dependencies-installed-size.txt](https://gitlab.cern.ch/linuxsupport/rpms/HEP_OSlibs/-/blob/7.2.14-1.el7/dependencies/HEP_OSlibs.x86_64.dependencies-installed-size.txt)
  - 694 MB (direct dependencies 185 packages, plus indirect dependencies)

# How is HEP\_OSlibs maintained?

- Since 2009 to 2017: built and maintained manually
  - Credits: O. Keeble, F. Furano, A.V.
  - Platforms: SLC5, SLC6, CC7 (all x86\_64)
- **Since October 2017: fully built and maintained in the gitlab CI**
  - Credits: A.V. (current maintainer), T. Oulevey
  - Platforms: *SLC6, CC7, Ubuntu16.10, Ubuntu18.10*
    - All on x86\_64 (one CC7 was released on ARM too)
  - See [https://gitlab.cern.ch/linuxsupport/rpms/HEP\\_OSlibs](https://gitlab.cern.ch/linuxsupport/rpms/HEP_OSlibs)
    - Source code repo, build and test (CI, koji), documentation (dependency lists)
  - Easy to add a package, *minimal maintenance (if there are no new features)*
    - How to add a package? Edit spec file, commit, run ./pkg.sh, let the CI do its work...
- **NB: *maintenance easy because there is a single meta-rpm***
  - One for all experiments (union of all required packages)
  - No distinction between runtime and buildtime (and no EPEL included)

# HEP\_OSlibs – two overlapping use cases

- **Initial motivation: Grid deployment**
  - Provide a uniform O/S *runtime* environment for landing of Grid jobs
    - En passant: this is also deployed on CERN Ixplus
  - First version in 2009 on SLC5 (e.g. SLC4 compatibility packages)
  - **Forum for discussion: WLCG operations and grid deployment meetings**
    - Grid deployment discussion in 2013 led to decision to *exclude EPEL packages*
- **Natural evolution: the substrate of the LCG software stack**
  - Runtime: Grid jobs run software from /cvmfs, on top of the LCG stack
  - Slowly moved to including also *buildtime* packages (e.g. headers)
  - Now also the substrate of SWAN software deployment
  - **Forum for discussion: LIM meetings**
    - Overall strategy clarified in a few LIM meetings in 2016-2018 (see title slide)

# Where is HEP\_OSlibs deployed?

- Current users from LHC: ALICE, ATLAS, LHCb – is this still the case?
  - Use HEP\_OSlibs directly (ALICE) or through the LCG stack (ATLAS, LHCb)
  - CMS has moved to a different model rebuilding much more than the others
- Grid nodes (runtime) for /cvmfs software of the above experiments
  - the initial motivation of HEP\_OSlibs
  - at CERN: lxbatch
  - some of the Docker containers for benchmarking too (e.g. ALICE workloads)
- Interactive user environments (development and runtime)
  - at CERN: lxplus; SWAN (through the LCG stack)
- Build nodes (development packages)
  - in particular, the build nodes for the LCG stack and the experiments IIUC

# Example

*ATLAS rXX x86\_64-centos7-gcc62-opt*

...

*ATLAS rYY x86\_64-centos7-gcc7-opt*

...

*LCG\_93/x86\_64-centos7-gcc62-opt*

*LCG\_93/ROOT/6.12.06/x86\_64-centos7-gcc62-opt*  
*LCG\_93/Python/2.7.13/x86\_64-centos7-gcc62-opt*  
*LCG\_93/Boost/1.66.0/x86\_64-centos7-gcc62-opt*  
*LCG\_93/expat/2.2.5/x86\_64-centos7-gcc62-opt*

...

*LCG\_93python3/x86\_64-centos7-gcc62-opt*

*LCG\_93python3/ROOT/6.12.06/x86\_64-centos7-gcc7-opt*  
*LCG\_93python3/Python/3.6.3/x86\_64-centos7-gcc7-opt*  
*LCG\_93python3/Boost/1.66.0/x86\_64-centos7-gcc7-opt*  
*LCG\_93python3/expat/2.2.5/x86\_64-centos7-gcc7-opt*

...

*HEP\_OSlibs for CentOS 7*

*libX11.x86\_64 (→ libX11-1.6.5-1.el7.x86\_64 for CentOS 7.5)*

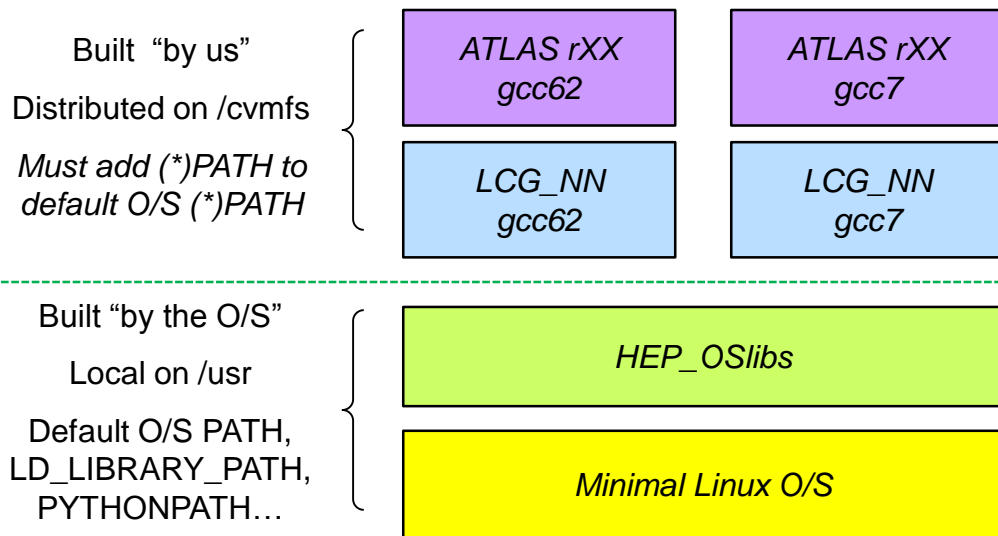
...

*“Minimal” CentOS 7.5.1804*

*kernel-3.10.0-862.3.2.el7.x86\_64.rpm*

*– and “a few” more packages (much fewer for CC7 than previously for SLC6) –*

# Why HEP\_OSlibs?



When a new dependency appears:  
- should it be built in the LCG stack?  
- should it be taken from the O/S?

Where do you put the boundary?

*IMO this question will continue to apply in the future regardless of technology (e.g. even with Spack)*

- My opinion in short:
  - **IF you can take a package from the O/S, why build it yourself?**
  - Lower risk of conflicts (we did observe many conflicts in the past – expat, uuid...)
  - Lower overhead to build and distribute



# Some new scenarios

- Grid nodes: only provide Docker/Singularity and /cvmfs?
  - Experiments ship their own containers including their favorite “O/S”?
    - Containers with *granular selection of O/S packages* (no HEP\_OSlibs)?
    - In any case: easy to cache O/S and HEP\_OSlibs Docker layers if desired
- Replace “LCG stack” machinery by Spack?
  - The question remains: *boundary between O/S and non-O/S packages?*
    - Choosing which O/S packages to use may still be needed
    - Can Spack be used to trace dependencies on individual O/S packages?
- Are there any scenarios left where HEP\_OSlibs is still needed?
  - Should clarify this, before deciding whether to split runtime/devel/EPEL
- Deploy HEP\_OSlibs on /cvmfs itself?
  - I do not think it makes sense, unless you use CernVM (O/S on /cvmfs)

# One-slide overview and summary

Present status and *open questions* both at PreGDB and LIM

This talk provides some background information

## HEP\_OSlibs for Grid deployment (discussion at WLCG meetings, PreGDB)

- New context: container deployment on Grid sites
  - *Do Grid sites still need a HEP\_OSlibs? For which experiments?*

## HEP\_OSlibs for the LCG software stack (discussion at recent LIM meeting)

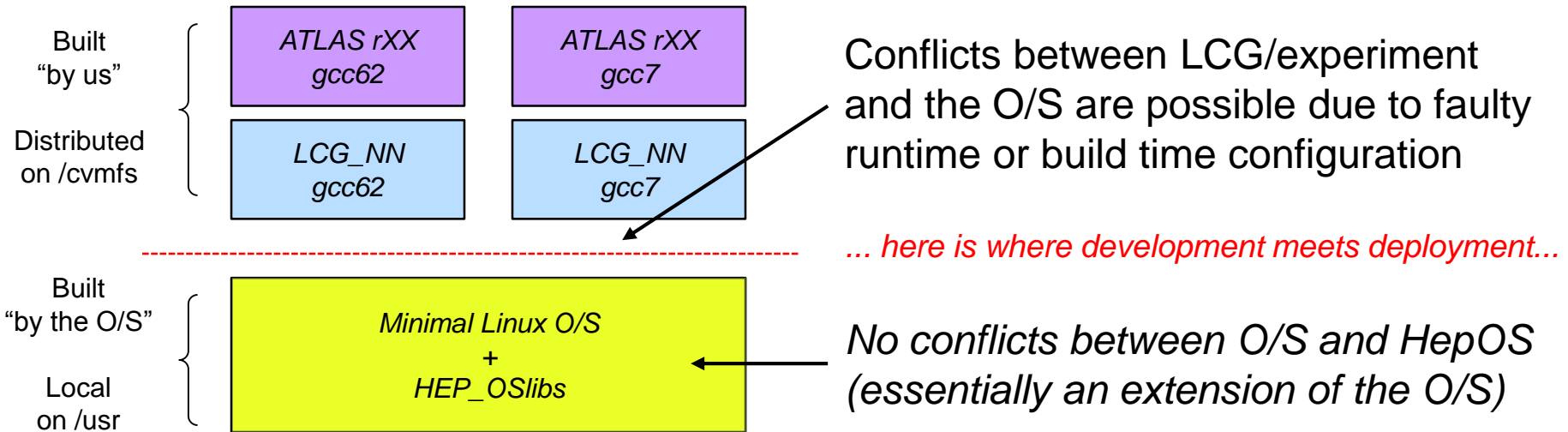
- New O/S platforms upcoming (CentOS8, Ubuntu20)
  - *New versions for el8, ubuntu2004? Drop el6, ubuntu1604?*
- Improve contents: runtime, developer packages, EPEL?
  - *Should we split runtime and developer meta-rpms?*
  - *Should we allow the inclusion of EPEL packages?*
- New model for software deployment? (Spack?)
  - *Do developers still need a HEP\_OSlibs (or an LCG stack)?*

# BACKUP SLIDES

# Which packages cannot be in HEP\_OSlibs?

- On Linux, I see only 3 reasons to build a package ourselves (LCG stack):
  - 1. Need a different C++ version than the O/S compiler (e.g. Boost)
  - 2. Need a newer package version than the O/S version (e.g. Python)
  - 3. Need a HEP or EPEL package not available in the O/S (e.g. ROOT)
  - In all other cases, I would install missing O/S packages via HEP\_OSlibs
- On MacOS, system directories (/bin, /sbin, /usr) may not be modified
  - Linux is an open system, MacOS a closed one (the “Apple walled garden”)
  - Three different policies and locations for Ports, Brew and Fink
  - No HEP\_OSlibs on MacOS: all external dependencies should be rebuilt
- *NB: if a package is missing on MacOS but can be included from the O/S on Linux, IMO it should be added to the LCG stack only on MacOS*
  - *do not add unnecessary risks of conflicts on Grid deployment (100% Linux)*

# Conflicts?



- Build time configuration: deploy HEP\_OSlibs on build nodes
- Runtime configuration: use dependencies from LCG/experiment stack if an O/S package has been duplicated there (e.g. LD\_LIBRARY\_PATH)
  - responsibility of LCG/experiment stack
  - *cannot solve these issues by uninstalling packages from the O/S or HepOS*
  - BTW: are the experiments using LCG views or cherry-picking packages?

# Is HEP\_OSlibs too fat?

- Most frequent user criticism on the deployment side
- By design, so far there has been only ONE HEP\_OSlibs version
  - dependencies of ALL experiments
  - dependencies for both runtime and build time
  - easier management...
  - easier deployment (e.g. lxplus)... and disk space is cheap after all?
- Of course this could be improved
  - but it would take someone's effort