

# Easybuild CERN Site Talk

---

EUM 2026

Valentin Volkl

**Is this about  
CernVM-FS?**

**CERN uses  
EasyBuild now?**



# ~~Easybuild CERN Site Talk~~

## Build tools in High Energy Physics

---

EUM 2026

- Biased, personal opinions on these slides
- Take with a grain of salt

Valentin Volkl

# Large Hadron Collider



- ATLAS, LHCb are using **LCGCMake**
  - in-house tool based on CMake ExternalProject Scripts
- CMS, ALICE use **cmsBuild/aliBuild** which shares some DNA
  - Also developed in-house, implemented in python with recipes in bash
  - After an evaluation of tools in 2016, Spack has been tried out, and adopted by some labs, but not at CERN
  - LHCb investigating Conda (with caches on CernVM-FS)
  - A major initiative to establish BITS <https://github.com/bitsorg/bits> (an evolution of aliBuild) as a common tool underway.

# Particularities of HEP Software Stacks / Builds

1. Balance between using common tools and diversified approaches
2. Long timescales
3. Workloads are data intensive and well suited for cloud computing
4. The sheer number of packages
5. [lcgcmake] A set of system packages is always assumed to be installed
6. [lcgcmake] source ../setup.sh instead of of modules

# Evaluation of build tools for HEP in 2016

|            | Linux | MacOS X | Windows | Xcompiler |
|------------|-------|---------|---------|-----------|
| aliBuild   | +     | +       | -       | +         |
| cmsBuild   | +     | +       | -       | +         |
| Contractor | +     | +       | -       | +         |
| Homebrew   | o     | +       | -       | -         |
| LCGCMake   | +     | +       | o       | o         |
| Nix        | +     | +       | o       | o         |
| SciSoft    | +     | +       | -       | -         |
| Spack      | +     | +       | -       | +         |
| Worch      | +     | +       | -       | o         |

|            | Multi-Rel | Multi-BuildVar | MultiShell-RTE | Relocation |
|------------|-----------|----------------|----------------|------------|
| aliBuild   | +         | +              | +              | +          |
| cmsBuild   | +         | +              | +              | +          |
| Contractor | +         | +              | NA             | -          |
| Homebrew   | -         | -              | NA             | -          |
| LCGCMake   | +         | +              | +              | +          |
| Nix        | +         | -              | +              | -          |
| SciSoft    | +         | +              | +              | +          |
| Spack      | +         | +              | NA             | -          |
| Worch      | +         | +              | +              | +          |

|            | Depends-On     | Ease-Add-Pkg  | Ease-Bootstrap | Documentation |
|------------|----------------|---------------|----------------|---------------|
| aliBuild   | Python         | spec-file     | +              | o             |
| cmsBuild   | Python,rpm,apt | spec-file     | o              | -             |
| Contractor | Python         | python file   | git checkout   | -             |
| Homebrew   | Ruby           | Formula       | +              | +             |
| LCGCMake   | Python, Cmake  | Cmake-macro   | +              | o             |
| Nix        | Perl           | expression    | -              | +             |
| SciSoft    | Cmake          | Cmake-macro   | o              | -             |
| Spack      | Python         | auto template | git checkout   | +             |
| Worch      | Waf,Python     | Text-files    | +              | o             |

|            | Performance | Sys-Reuse | Community | Unique-IDs | VCS-Support |
|------------|-------------|-----------|-----------|------------|-------------|
| aliBuild   | +           | o         | -         | +          | +           |
| cmsBuild   | +           | o         | -         | +          | +           |
| Contractor | +           | +         | -         | -          | +           |
| Homebrew   | +           | +         | +         | -          | +           |
| LCGCMake   | +           | +         | +         | +          | +           |
| Nix        | +           | -         | +         | +          | ?           |
| SciSoft    | o           | o         | +         | +          | +           |
| Spack      | o           | o         | +         | +          | +           |
| Worch      | +           | +         | -         | o          | +           |

# [notes] HEP Software and Computing

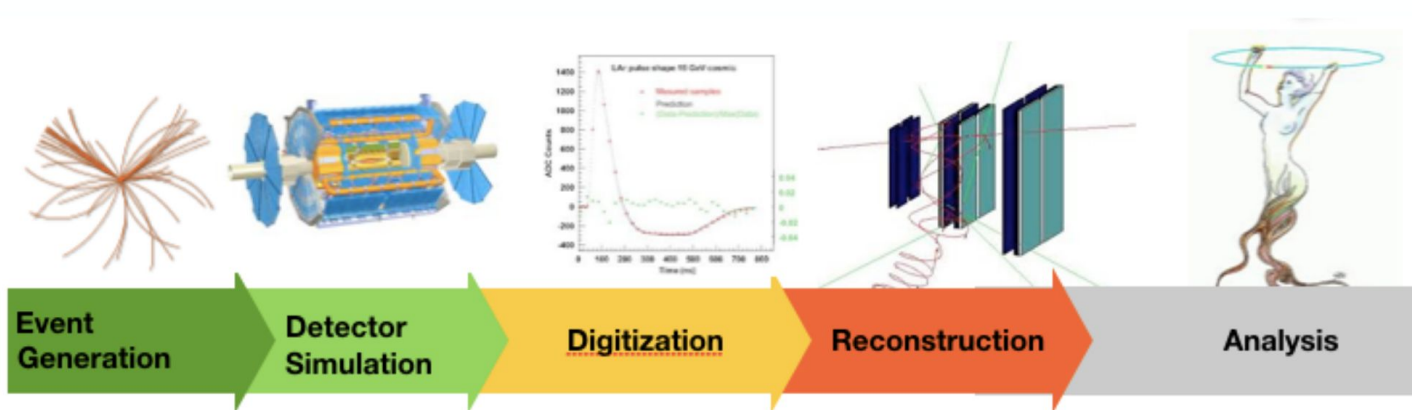
- High Energy Physics has a vast investment in software
  - Estimated to be around 50M lines of C++ (almost all open source!)
  - Which would cost more than 500M\$ to develop commercially
- It is a critical part of our physics production pipeline, from triggering the event readout all the way to analysis and final plots as well as simulation
- LHC experiments use about 1M CPU Cores every hour of every day, we have around 1000PB of data with 1000PB of data transfers per year (10-100Gb links)
  - We are in the exabyte era already
- This is a *huge* and *ongoing* cost in hardware and human effort
- With significant challenges ahead of us to support our ongoing physics programme

## Particularities of HEP Software Stacks / Builds

1. Balance between using common tools and diversified approaches
2. Long timescales
- 3. Workloads are data intensive and well suited for cloud computing**
4. The sheer number of packages
5. [lcgcmake] A set of system packages is always assumed to be installed
6. [lcgcmake] source .../setup.sh instead of of modules

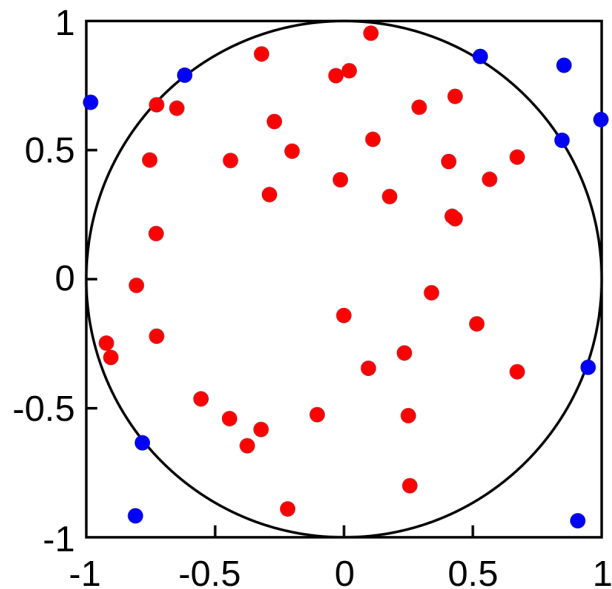
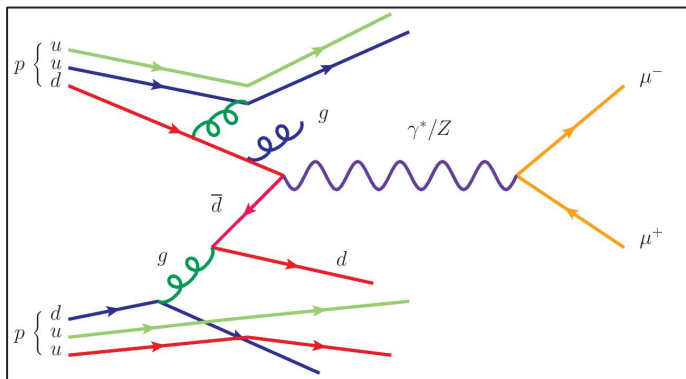
# Computing Workloads in HEP

- Generation
- Simulation
- Reconstruction
- Analysis



# Event Generation

$$\sigma_{ab \rightarrow N} = \int_0^1 dx_a dx_b f_a(x_a, \mu_F) f_b(x_b, \mu_F) \int_{\text{cuts}} d\Phi_N \frac{1}{2\hat{s}} |\mathcal{M}(\Phi_N, \mu_F, \mu_R)|^2$$



# The LHC Computing Grid

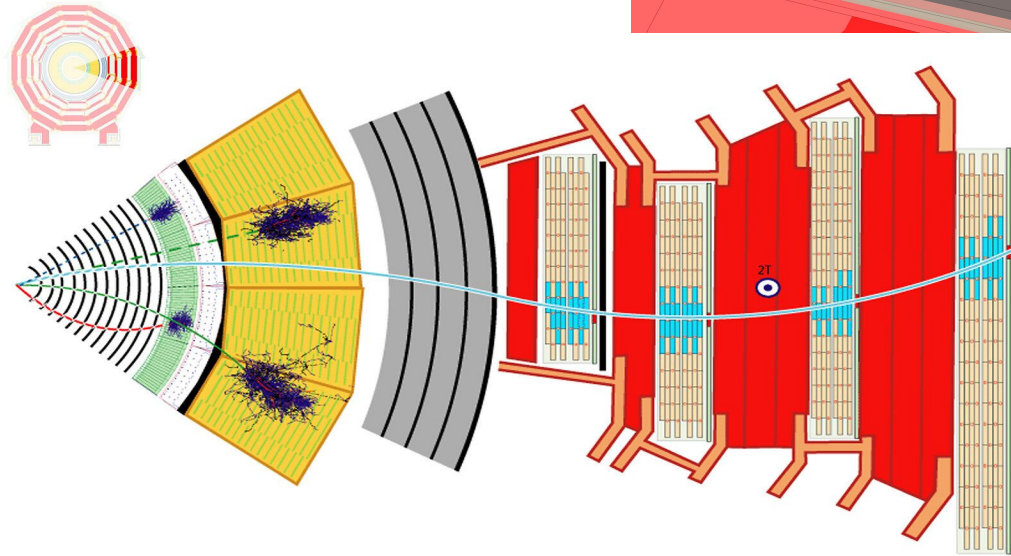
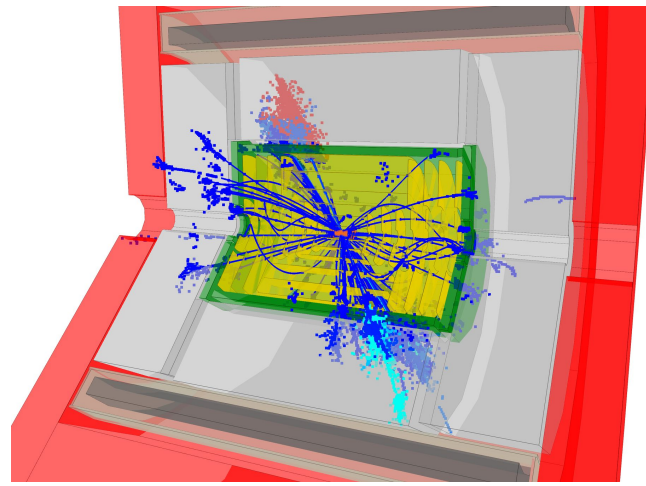
Started in 2002  
Provide processing power and data  
access to physicists  
~170 centres in 42 countries  
Running 24/7/365



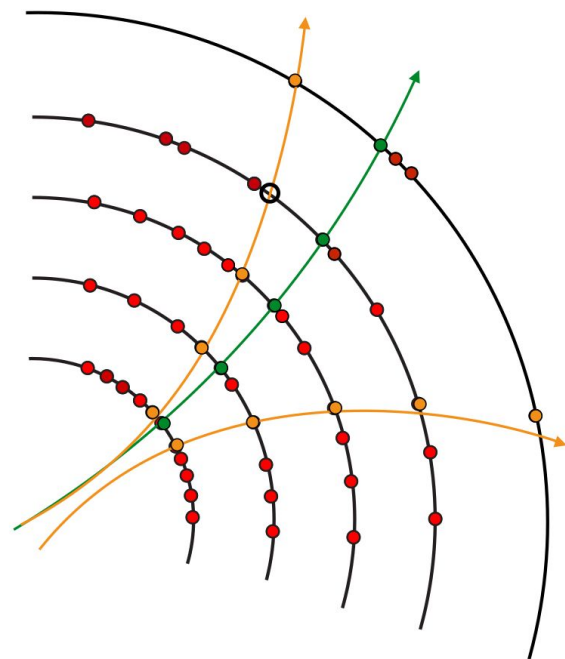
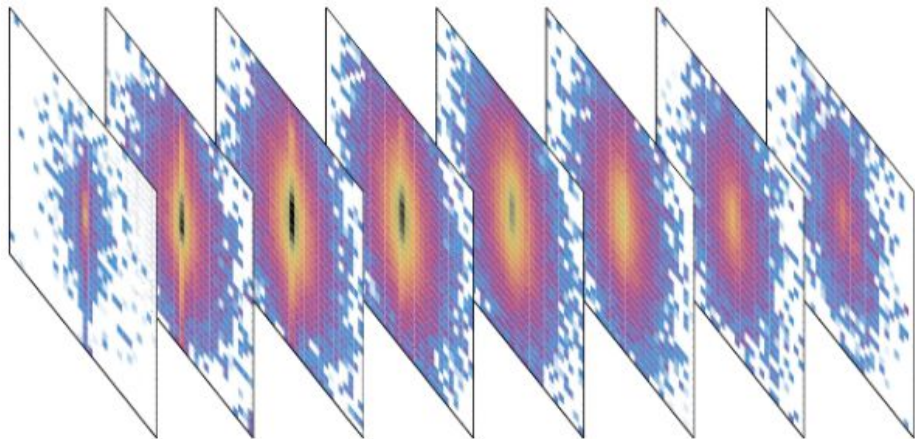
# [notes] Event Generation

- Our hypotheses what goes on in a collision of particles are expressed in QFTs
  - Observable quantities can be calculated from high-dimensional integrals
- “Feynman diagrams” are actually representation of perturbation terms of these integrals

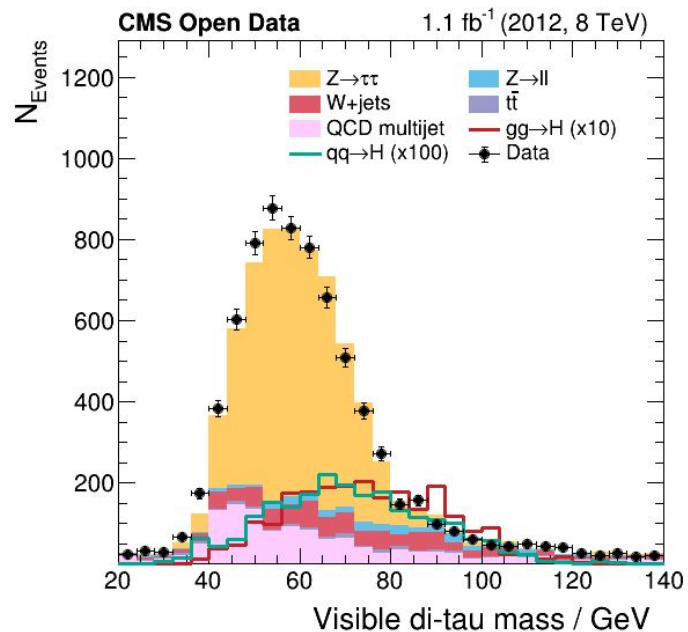
# (Detector) Simulation



# Reconstruction



# Analysis



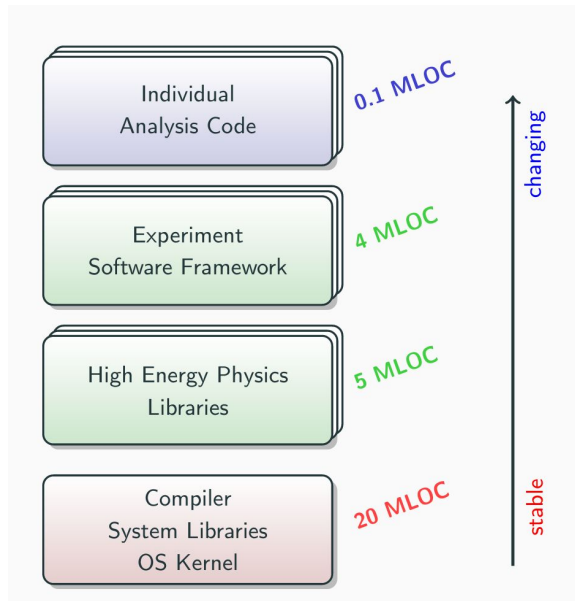
# Particularities of HEP Software Stacks #2

The sheer number of packages

## Particularities of HEP Software Stacks / Builds

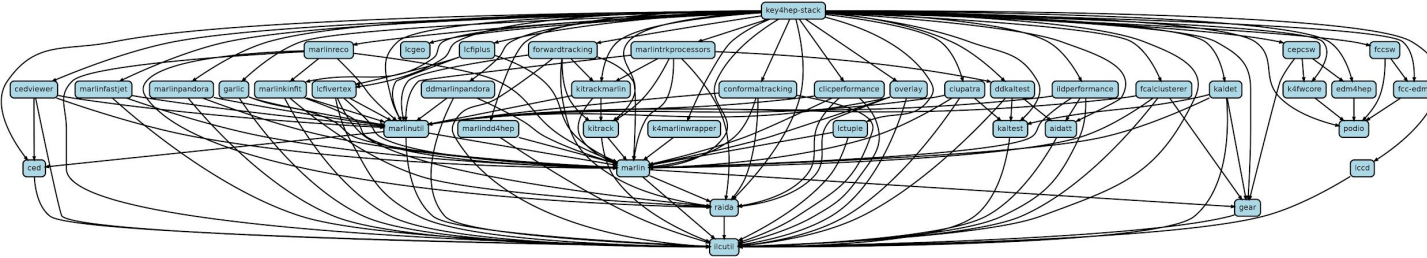
1. Balance between using common tools and diversified approaches
- 2. Workloads are data intensive and well suited for cloud computing**
3. The sheer number of packages
4. [lcgmake] A set of system packages is always assumed to be installed
5. [lcgmake] source ../setup.sh instead of of modules

# The HEP Software Stack



# Dependency Graph of HEP software Stack

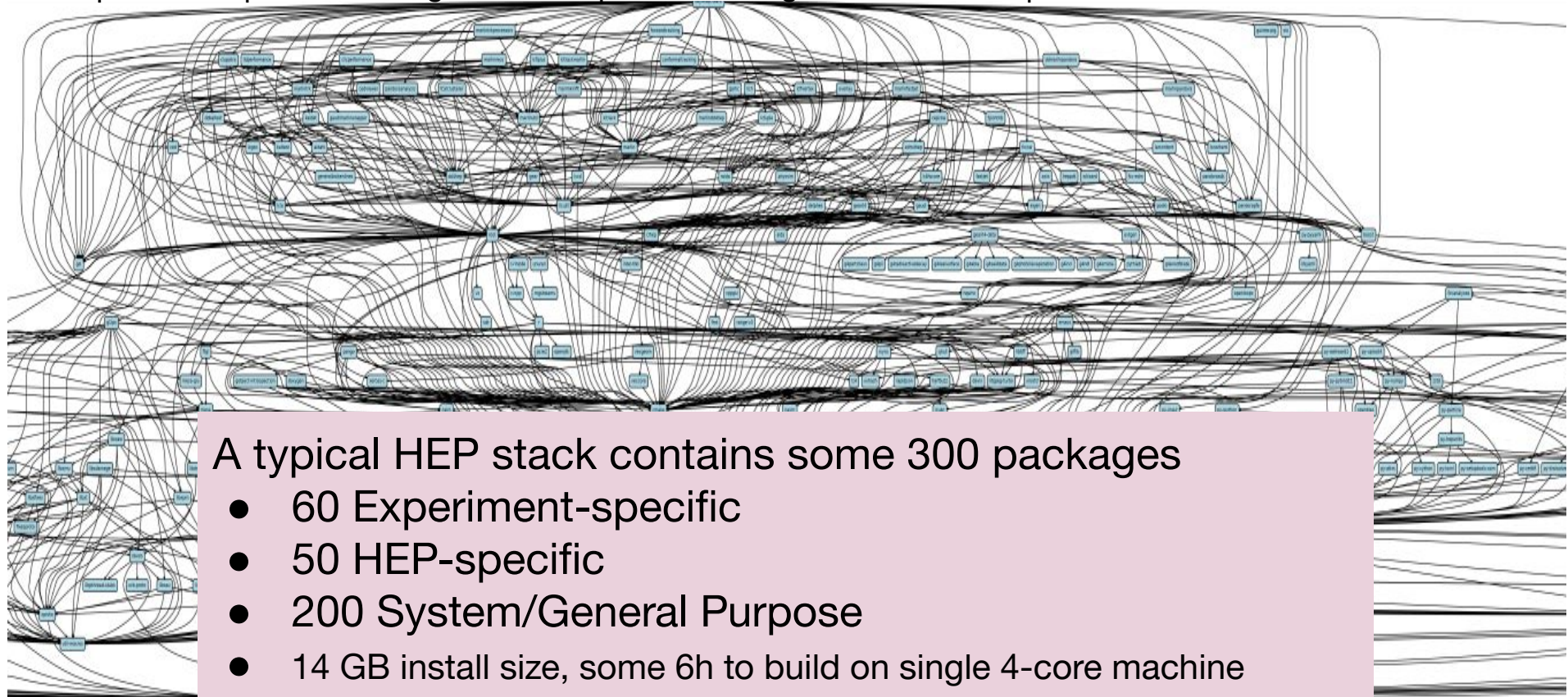
## Experiment-specific Packages



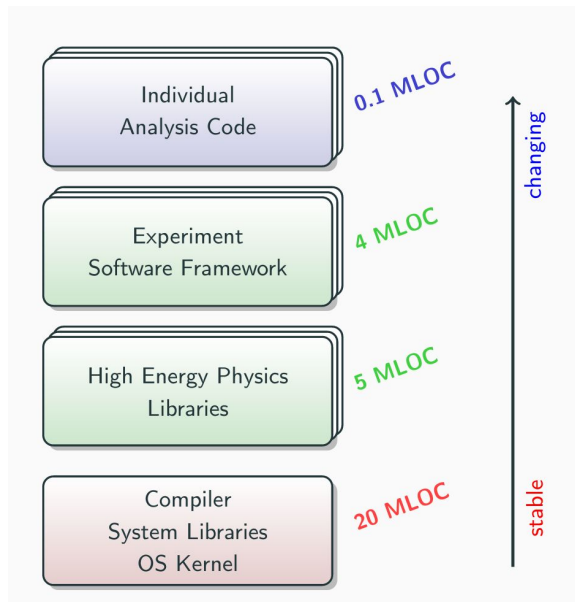


# Dependency Graph of HEP software stack

Experiment-specific Packages + HEP-specific Packages + General Purpose Libraries



# The Anatomy of a HEP Software Stack



## Key Figures for LHC Experiments

- Hundreds of (novice) developers
- > 100 000 files per release
- 1TB / day of nightly builds
- ~100 000 machines world-wide
- Daily production releases, remain available “eternally”

## Particularities of HEP Software #3

- Ubiquitous source `/cvmfs/.../setup.sh`
- Modules would be certainly nicer, but need to work with 500+ packages

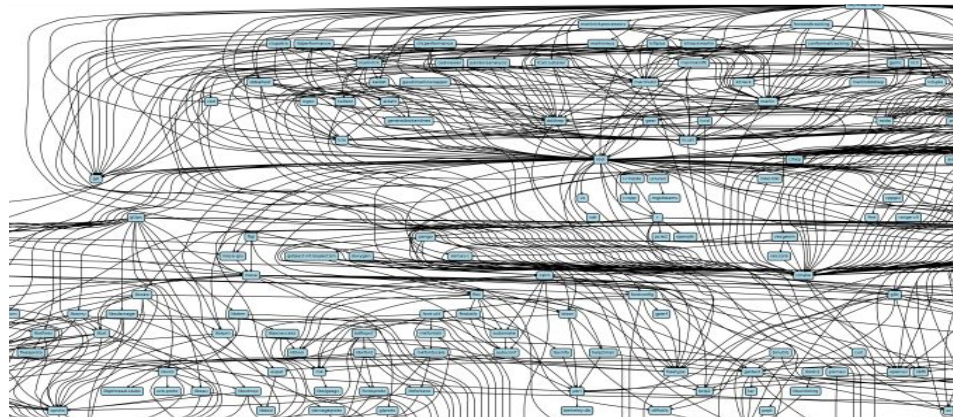
## Particularity #4

Not everything is taken from CernVM-FS, a set of system packages is always assumed to be installed (meta-package available for RHEL and Debian).

- Compatibility layer would be very helpful, but needs to allow building software on top outside of the build tool.
  - ROOT makes this particularly complicated

The recipes are more  
important than the build  
tool

Personal Opinion



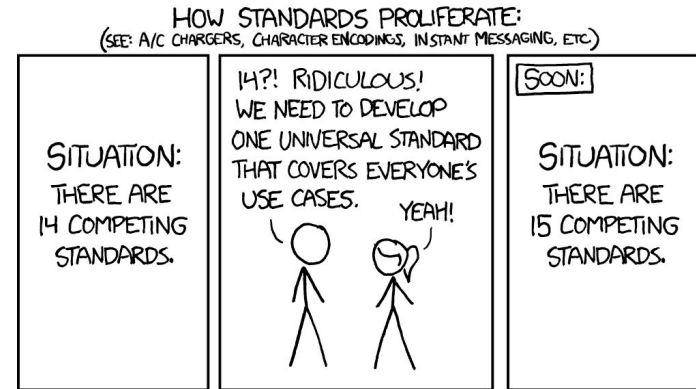
- If it's not possible to agree on one build tool, maybe it's possible to agree on one package recipe standard?

- Dream: derive any possible package format from a central source of truth

Obvious issues:

- Core build tool logic bound up in build tool (for example CMakePackage).
- Complex tool chains
- Stability and Governance

... Probably many more



# Conclusions

High Energy Physics faces some fairly unique challenges...

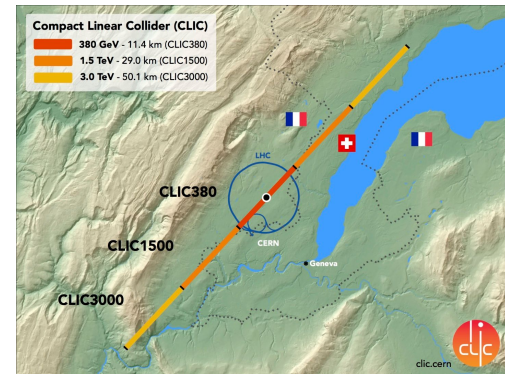
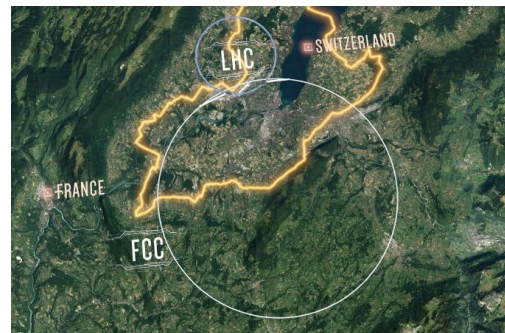
- And innovated on tools to solve them (see CVMFS in this workshop)

... but also many common ones, and would probably not be possible without the wider OSS ecosystem.

- HEP ❤️ Open Source!

- Open Data projects
- Software Projects:
  - ROOT
  - Geant4
  - Indico
  - ...

Future Collider Projects are collaborating on Software (Key4hep)!



FCC, CLIC, ILC, CEPC, ...