Key4hep

Core, Sim and Rec Tools for Future Colliders

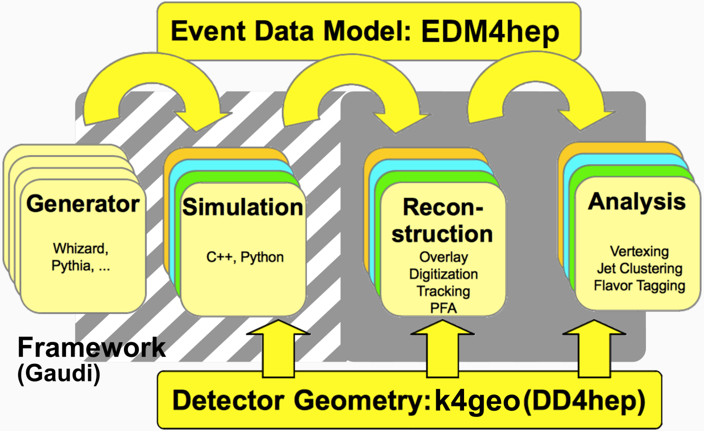
Thomas Madlener

FH Future Collider Day & SciComp Workshop

Jul 1, 2024

This project has received funding from the European Union’s Horizon 2020 Research and Innovation programme under grant agreement No 101004761.

From generation to analysis - the general workflow

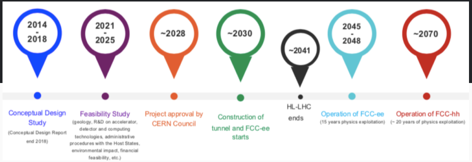


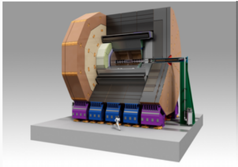
- Many steps involved from generating events to analyzing them
- Hundreds of SW packages
  - Building & deploying
  - Consistency
  - Reproducibility
- Try to give an overview of the **Key4hep** SW ecosystem

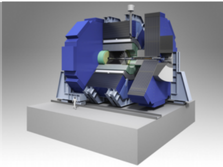
Look for major DESY contributions on earmarked slides

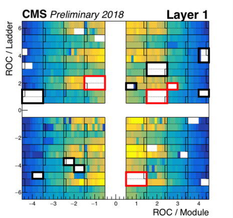
Software Challenges in HEP

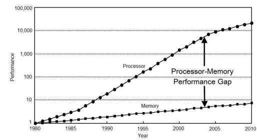
- Long lifetimes of experiments
- Shift of priorities throughout the evolution of an experiment
  - Conceptual and design work
  - Production and the real world
  - Continuous upgrades
  - **Avoid amassing "technical debt"**
- New technological developments potentially lead to new paradigms
  - Optimize resource usage 🌱
- Data preservation and ability to look at data in the future

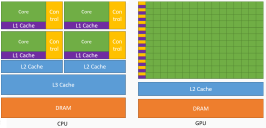

2014 - 2018
2021 - 2025
~2028
~2030
~2041
2045 - 2048
~2070
Conceptual Design Study
(Conceptual Design Report end 2018)
Feasibility Study
(geology, R&D on accelerator, detector and computing technologies, administrative procedures with the Host States, environmental impact, financial feasibility, etc.)
Project approval by CERN Council
Construction of tunnel and FCC-ee starts
HL-LHC ends
Operation of FCC-ee
(15 years physics exploitation)
Operation of FCC-hh
(~ 20 years of physics exploitation)



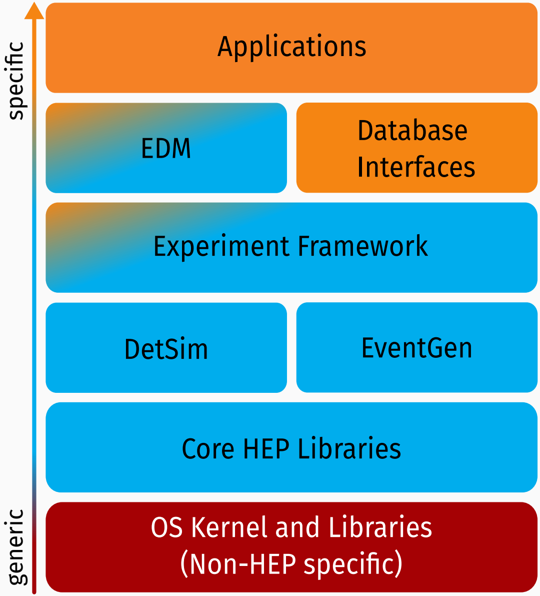








HEP Software Stack



specific

generic

Applications

EDM

Database Interfaces

Experiment Framework

DetSim

EventGen

Core HEP Libraries

OS Kernel and Libraries (Non-HEP specific)

Dedicated algorithms (reconstruction, analysis, …)

Data access & representation

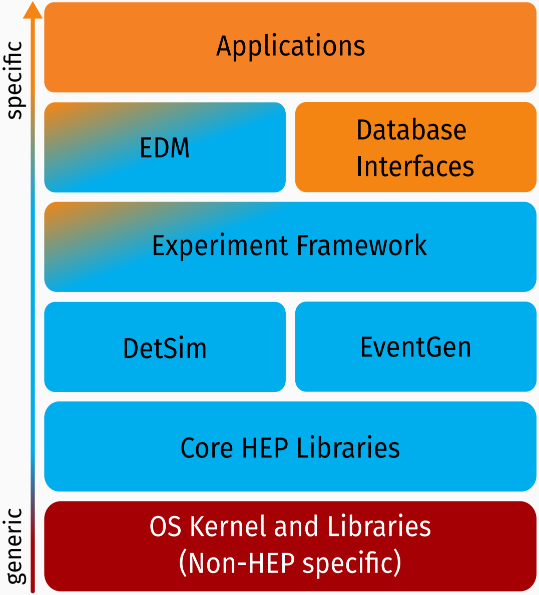
Algorithm / workflow orchestration

Specific components reused by many experiments (DD4hep, Delphes, Pythia, …)

Commonly used HEP core libraries (ROOT, Geant4, CLHEP, …)

Python, Compilers, CMake, boost, …

HEP Software Stack



- Pieces of software are not living in isolation
- Ecosystem of interacting components
- Compatibility between different elements doesn’t come for free
  - Common standards can help a lot
- Building a consistent stack of software for an experiment is highly non-trivial
  - Benefits can be gained from using common approaches

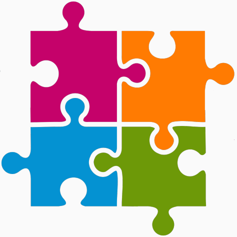
Key4hep Motivation & Brief History

- Future detector studies require well maintained software
- Existing scattered landscape of HEP software
  - Dedicated tools for specific tasks
  - Integrated frameworks tailored to specific experiments
- Available person power for maintenance and development is limited
- Consensus to develop a common software stack @ "2019 Bologna Workshop"
  - Originally by **FCC, ILC, CLIC, CEPC**
  - By now also with contributions from **EIC, MuonCollider, ...**
- Support from major R&D initiatives
  - CERN R&D for Future Experiments, AIDAinnova WP12, ECFA

Key4hep goals

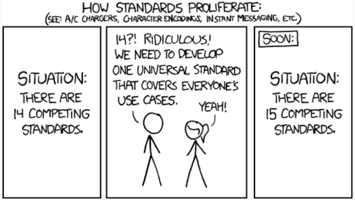
- Provide and maintain a consistent SW stack that allows to do physics studies for **all projects**
- Ensure interoperability of the necessary building blocks
- Reuse existing solutions where possible
  - A lot of experience from LHC experiments and LC communities
- Focus new developments on EW/Higgs factory specifics
- Share knowledge, processes, workflows and resources
  - Best practices, tutorials, documentation, …

Photo by Stewart B. / CC-BY



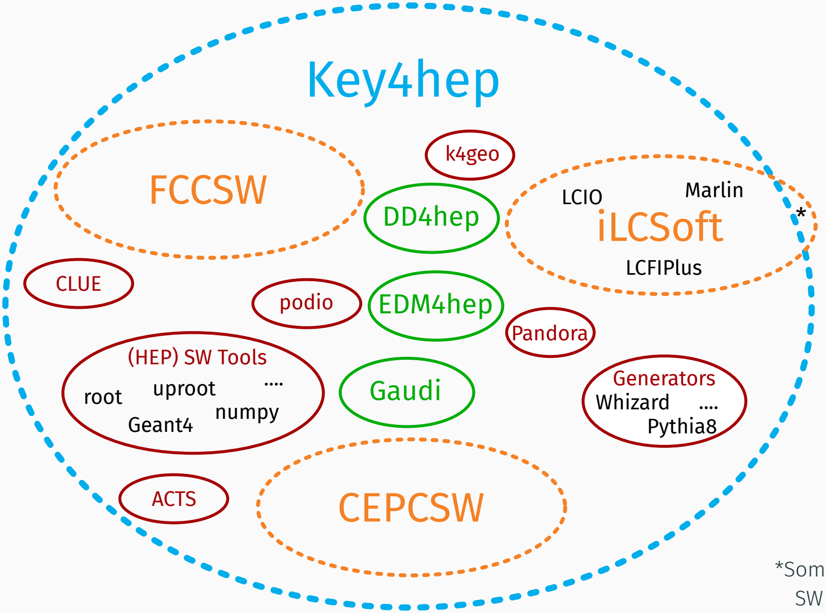
Non-goal

- Develop and maintain project specific software and workflows


HOW STANDARDS PROLIFERATE:
(SEE: A/C CHARGERS, CHARACTER ENCODINGS, INSTANT MESSAGING, ETC)
SITUATION: THERE ARE 14 COMPETING STANDARDS.
14?! RIDICULOUS! WE NEED TO DEVELOP ONE UNIVERSAL STANDARD THAT COVERS EVERYONE'S USE CASES.
YEAH!
SOON:
SITUATION: THERE ARE 15 COMPETING STANDARDS.

xkcd.com/927

Key4hep (simplified) overview

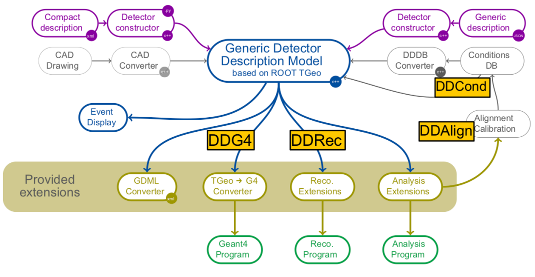


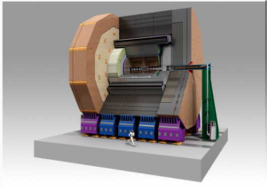
*Some testbeam related
SW not yet included

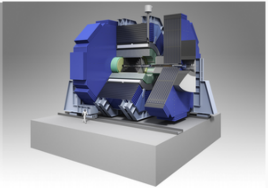
DD4hep - Detector description

dd4hep.web.cern.ch

- Complete detector description
  - Geometry, materials, visualization, readout, alignment, calibration, ...
- From a **single source of information**
  - Simulation, reconstruction, analysis
- Comes with a powerful plug-in mechanism that allows customization
- More or less “industry standard” now
  - FCC, ILC, CLIC, EIC, LHCb, CMS, ODD, ...
- `ddsim` - standalone simulation executable

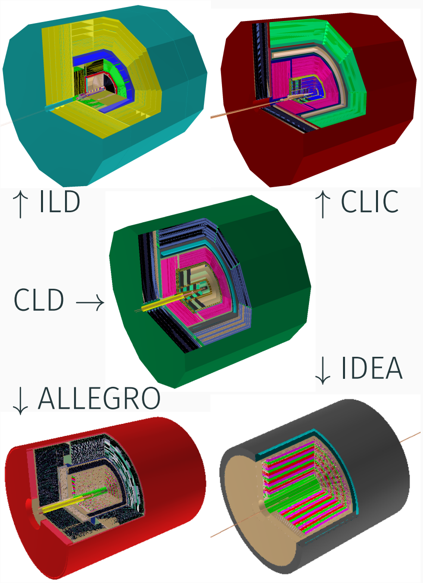






k4geo - The detector geometry repository

- Central repository for detector models
- Many existing detector models from LC studies
- Many recent developments for FCC detector concepts
- “Plug and play” approach for subdetectors
    - Use CLD inner tracker in ILD for TPC studies at FCC



↑ ILD

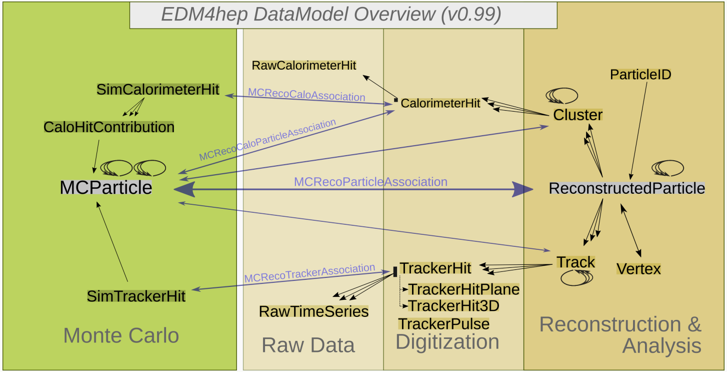
↑ CLIC

CLD →

↓ IDEA

↓ ALLEGRO

EDM4hep - The common EDM for Key4hep



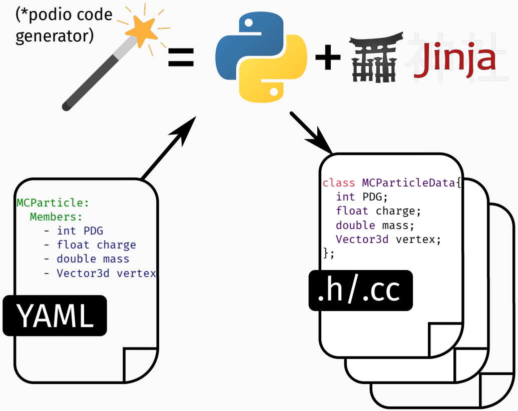
- Based on `LCIO` and `FCC-edm`
  - Focus on usability in reconstruction & analysis
- Currently finalizing v1.0
- Can easily be extended
  - Used by EDM4eic
  - Prototyping!
- Generated via `podio`

key4hep/EDM4hep

edm4hep.web.cern.ch

The podio EDM toolkit

- Implementing a performant event data model (EDM) is non-trivial
- Use `podio` to generate code starting from a high level description
- Provide an easy to use interface to the users
- **Version 1.0 just released!**



```yaml
MCParticle:
  Members:
    - int PDG
    - float charge
    - double mass
    - Vector3d vertex
```

```cpp
class MCParticleData{
  int PDG;
  float charge;
  double mass;
  Vector3d vertex;
};
```

AIDASoft/podio

key4hep.web.cern.ch/podio

Experiment Framework & Core components

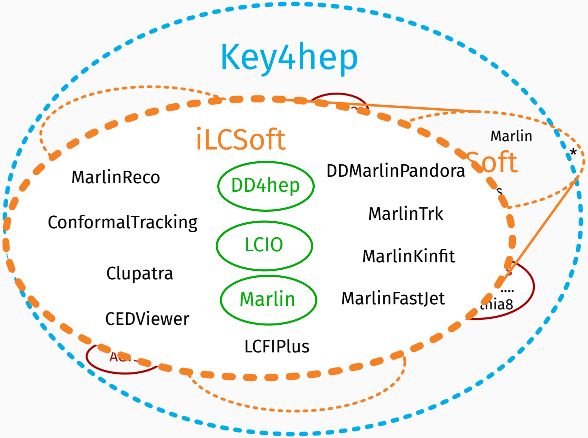
- `Gaudi`, originally developed by LHCb, now also used by ATLAS, FCCSW and smaller experiments
  - Supports concurrency
  - “Battle-proven” from data taking during LHC operations
- Key4hep has decided to adapt `Gaudi` as its experiment framework
  - Contribute to its development where necessary

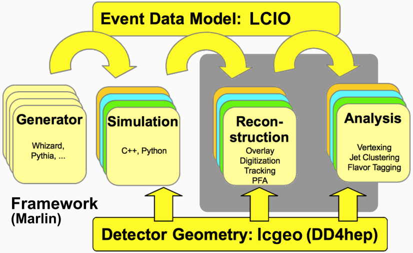
The `k4FWCore` package

key4hep/k4FWCore

- Providing core functionality, e.g.
  - Data Service for EDM4hep / podio inputs
  - `k4run` for running options files

iLCSoft - The trove of existing solutions

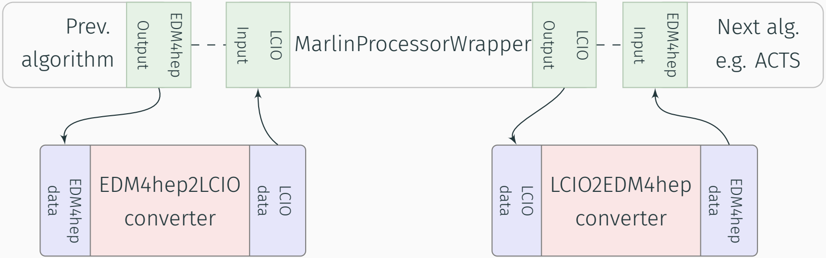




- Full suite of reconstruction & analysis tools
- Used in numerous productions for LC studies

k4MarlinWrapper

- Wraps Marlin processor in a Gaudi algorithm and allows to run them unchanged
- Automatic, on-the-fly conversion between LCIO and EDM4hep
- Allows to "mix and match" existing reconstruction algorithms with new developments
  - Working horse for many full simulation studies at the moment

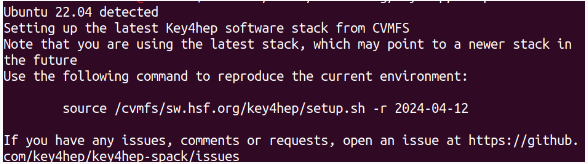


Keyhep releases and nightlies

- (Rolling) latest release of the complete Key4hep software stack
  - Full stacks for AlmaLinux9, Ubuntu22.04, (CentOS7)

```
/cvmfs/sw.hsf.org/key4hep/setup.sh
/cvmfs/sw-nightlies.hsf.org/key4hep/setup.sh
```

- Documentation
  - key4hep.github.io/key4hep-doc
  - Includes tutorials & How-tos
- **Release early and release often**
  - Make fixes available early
  - Discover problems and collect feedback as early as possible
- Biweekly, alternating meetings for Key4hep & EDM4hep
  - indico.cern.ch/category/11461/



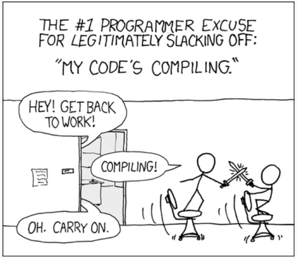
```
Ubuntu 22.04 detected
Setting up the latest Key4hep software stack from CVMFS
Note that you are using the latest stack, which may point to a newer stack in
the future
Use the following command to reproduce the current environment:

        source /cvmfs/sw.hsf.org/key4hep/setup.sh -r 2024-04-12

If you have any issues, comments or requests, open an issue at https://github.
com/key4hep/key4hep-spack/issues
```

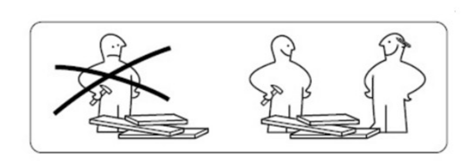
Summary

- Key4hep aims to provide a common software stack for all future collider projects
- **Very successful in bringing together communities and focusing on common approaches**
- In use for physics studies by several communities already
- Finalizing first versions of core components
- Ongoing efforts towards more “native Key4hep algorithms”
- No shortage of work
    - Contact us (T. Madlener, F. Gaede) if you want to get involved

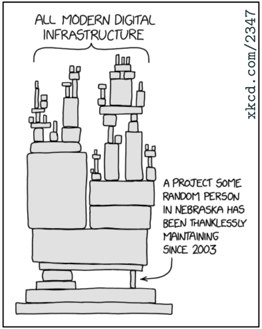


xkcd.com/303

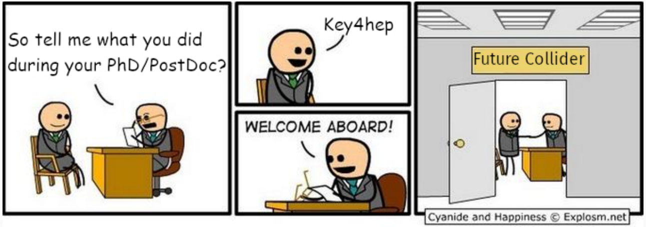
A few convincing arguments



Collaboration is “The Right Thing”^{TM}

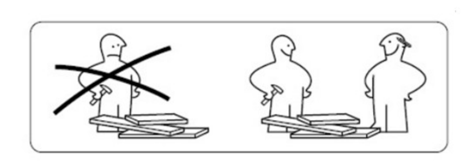

ALL MODERN DIGITAL INFRASTRUCTURE
A PROJECT SOME RANDOM PERSON IN NEBRASKA HAS BEEN THANKLESSLY MAINTAINING SINCE 2003
xkcd.com/2347

- Managing large SW stacks
- Research Software Engineering
  - Continuous Integration (CI)
  - Containerization
  - ML/AI (and running it in production SW)
  - 🐝 *more buzz words

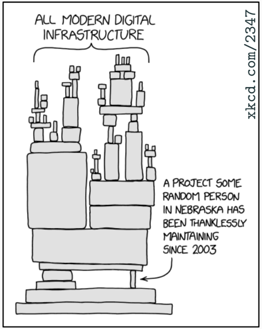

So tell me what you did during your PhD/PostDoc?
Key4hep
WELCOME ABOARD!
Future Collider
Cyanide and Happiness © Explosm.net

- More work than people
  → Create your own area of work
- A lot of visibility
- **Simply a cool project**

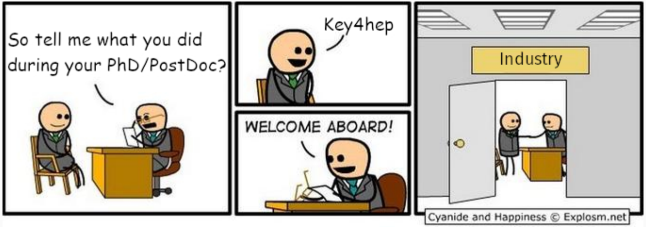
A few convincing arguments



Collaboration is “The Right Thing”^{TM}

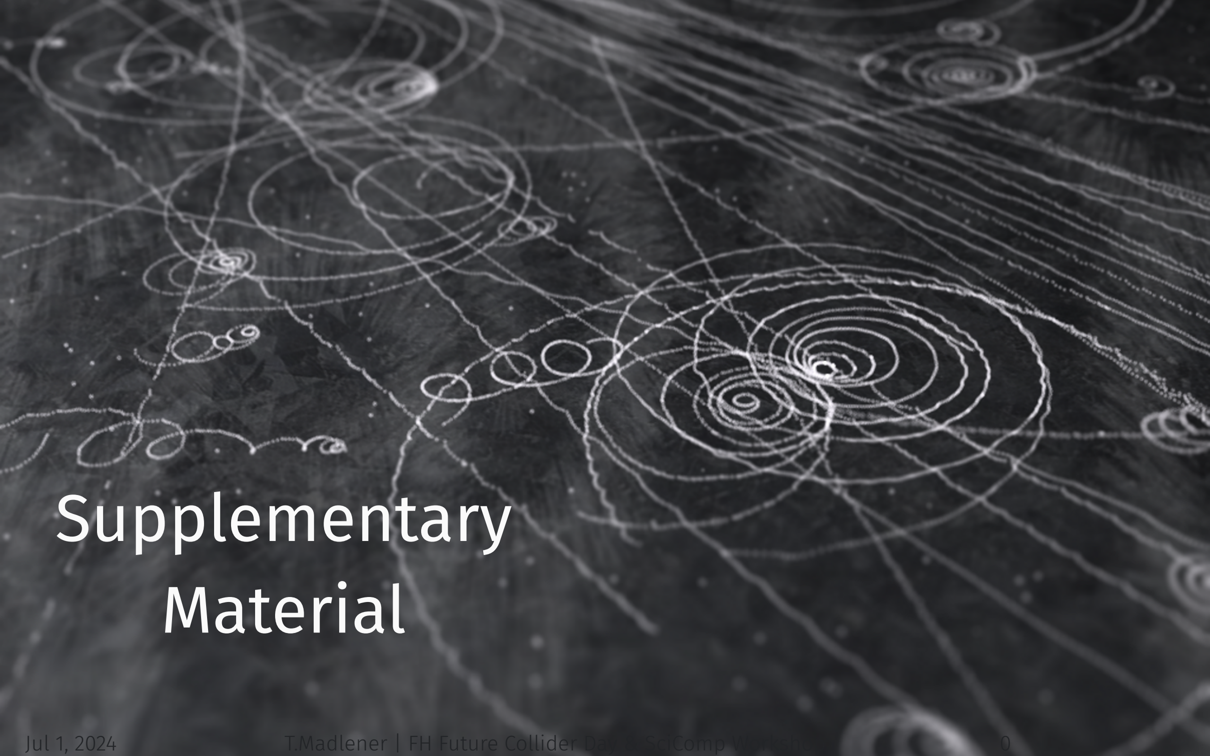

ALL MODERN DIGITAL INFRASTRUCTURE
A PROJECT SOME RANDOM PERSON IN NEBRASKA HAS BEEN THANKLESSLY MAINTAINING SINCE 2003
xkcd.com/2347

- Managing large SW stacks
- Research Software Engineering
  - Continuous Integration (CI)
  - Containerization
  - ML/AI (and running it in production SW)
  - *more buzz words


So tell me what you did during your PhD/PostDoc?
Key4hep
WELCOME ABOARD!
Industry
Cyanide and Happiness © Explosm.net

- More work than people

  → Create your own area of work
- A lot of visibility
- **Simply a cool project**



Supplementary Material

Pointers to software (re)sources

- Key4hep
  - key4hep.github.io/key4hep-doc
  - key4hep - github organisation
- EDM4hep
  - key4hep/EDM4hep
  - cern.ch/edm4hep
- DD4hep
  - AIDASoft/DD4hep
  - dd4hep.web.cern.ch
- podio
  - AIDASoft/podio
  - key4hep.web.cern.ch/podio
- FCCSW
  - HEP-FCC - github organisation

xkcd.com/138

Spack for Key4hep

- Spack is a package manager
  - Independent of operating system
  - Builds all packages from source
- Originaly developed by the HPC community
  - Emphasis on dealing with **multiple configurations** of the same package
- Basic building block is a formalized build procedure → **spack recipe**
  - Build instructions, dependencies, versions and location of source code
  - ∼ 8000 packages currently available from spack
  - Many Key4hep packages in key4hep/key4hep-spack
- The whole Key4hep software stack can be built from scratch using spack

```
spack install key4hep-stack
```

Key4hep packages

- `k4FWCore` key4hep/k4FWCore
  - Core Key4hep framework providing core functionality, e.g.
    - Data Service for EDM4hep / podio inputs
    - Overlay for backgrounds
- `k4SimDelphes` for Delphes fast simulation key4hep/k4SimDelphes
- `k4MarlinWrapper` Marlin proc. wrapper key4hep/k4MarlinWrapper
- Many packages migrated from FCCSW to Key4hep
  - `k4SimGeant4` for Geant4 simulation integration HEP-FCC/k4SimGeant4
  - `k4Gen` for generic generator interface HEP-FCC/k4Gen
  - ...
- Ongoing work to integrate more components
  - ACTS tracking framework acts-project/acts | key4hep/k4ActsTracking
  - CLUE fast clustering algorithms .cern.ch/kalos/CLUE | key4hep/k4CLUE