Parallel Session - Data Management, Access and Organization / Data Streaming - HEP Introduction

Oliver Gutsche

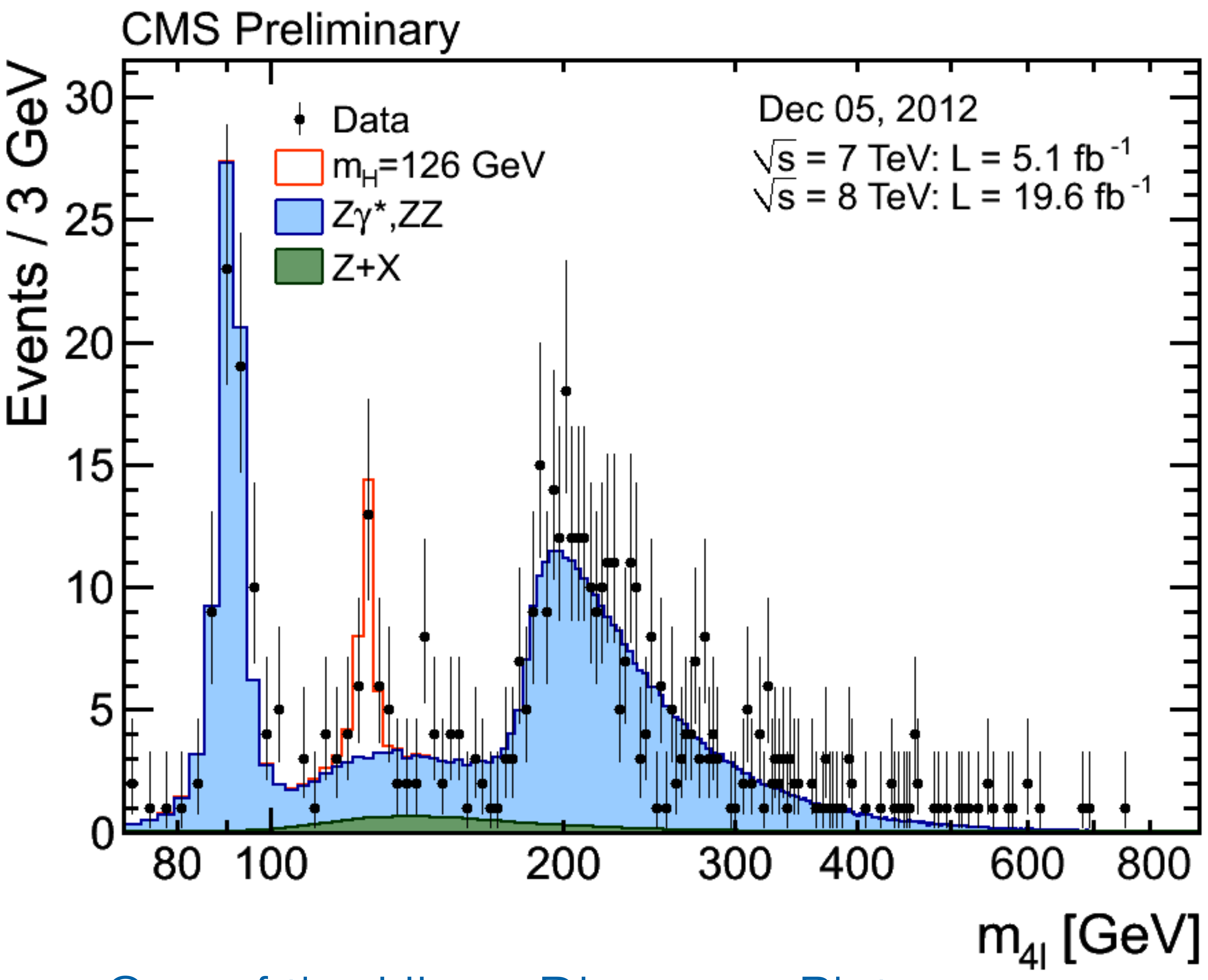
2nd S2I2 HEP/CS Workshop, Princeton

02. May 2017

Outline

- My personal conclusion from the first day: it is difficult for CS and HEP people to speak the same language
- A good outcome from this parallel session (in this order)
  - Talk about data management, define terminology and try to speak the same language
  - Define a list of questions for HEP to define the data management problem of the HL-LHC for CS
  - Talk about possible common projects
- I don't know what is known of the data management problem of the HL-LHC, I try to define what the LHC is currently doing and then try to define the HL-LHC problem

Experimental Particle Physics

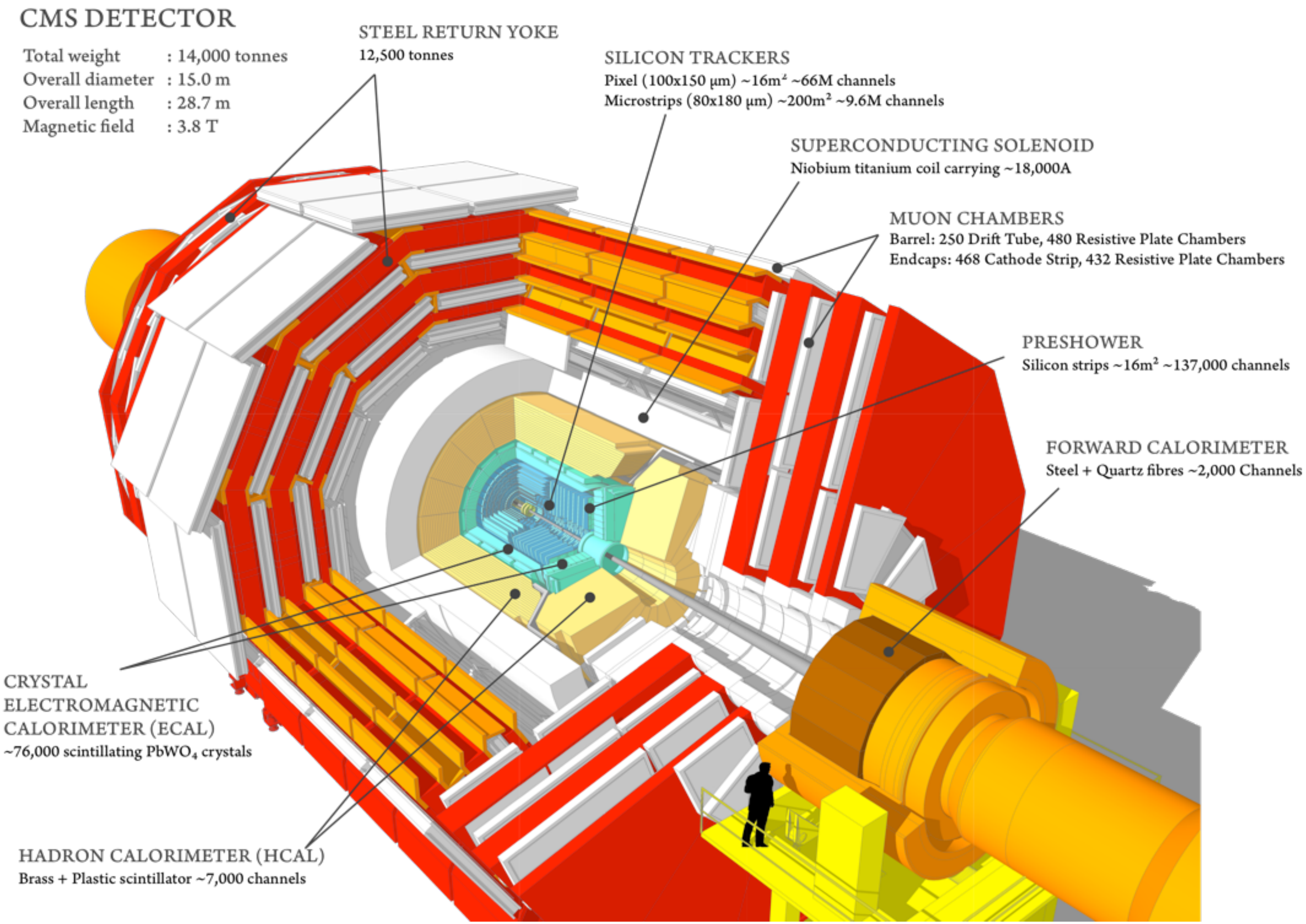


One of the Higgs Discovery Plot:

Higgs Simulation is RED

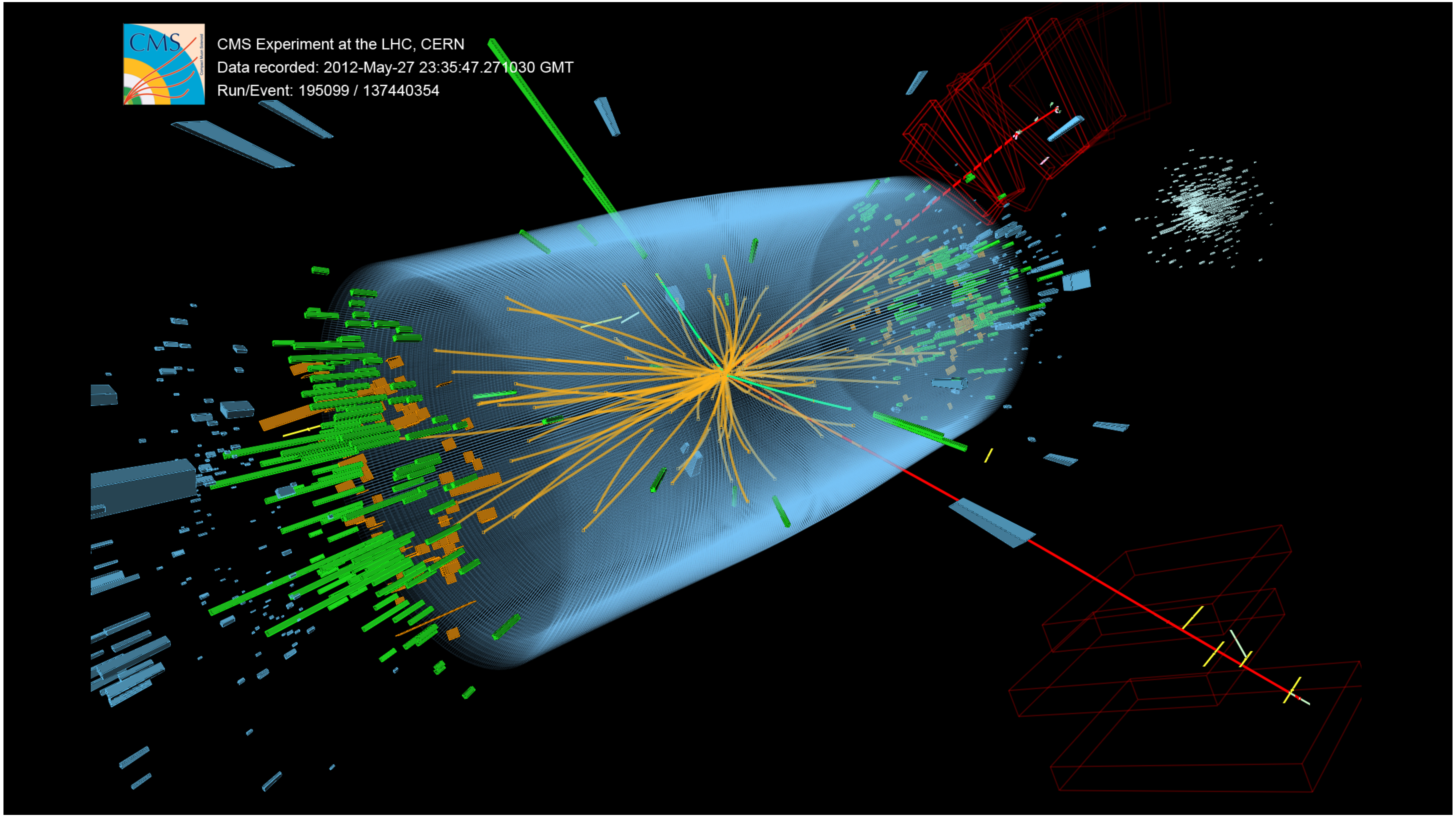
- Detect particle interactions and compare to Standard Model
  - Black dots: measurement
  - Blue shape: simulation of Standard Model
  - Red shape: simulation of new theory (in this case the Higgs)

Detector Simulation: Compact Muon Solenoid (CMS)

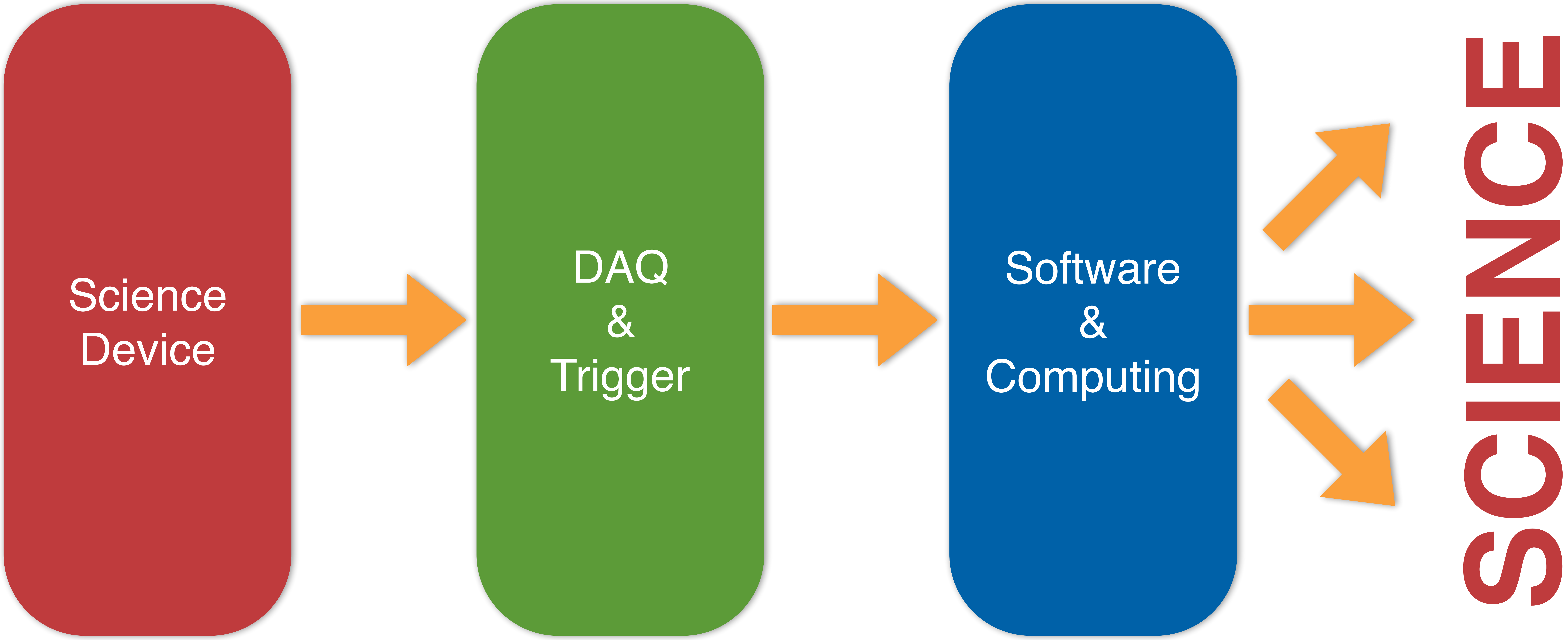


- Detector built around collision point
  - One of four detectors at the LHC
- Records flight path and energy of all particles produced in a collision
- 100 Million individual measurements (channels)
  - Grouped by detector component
- All measurements of a collision together are called: **event**

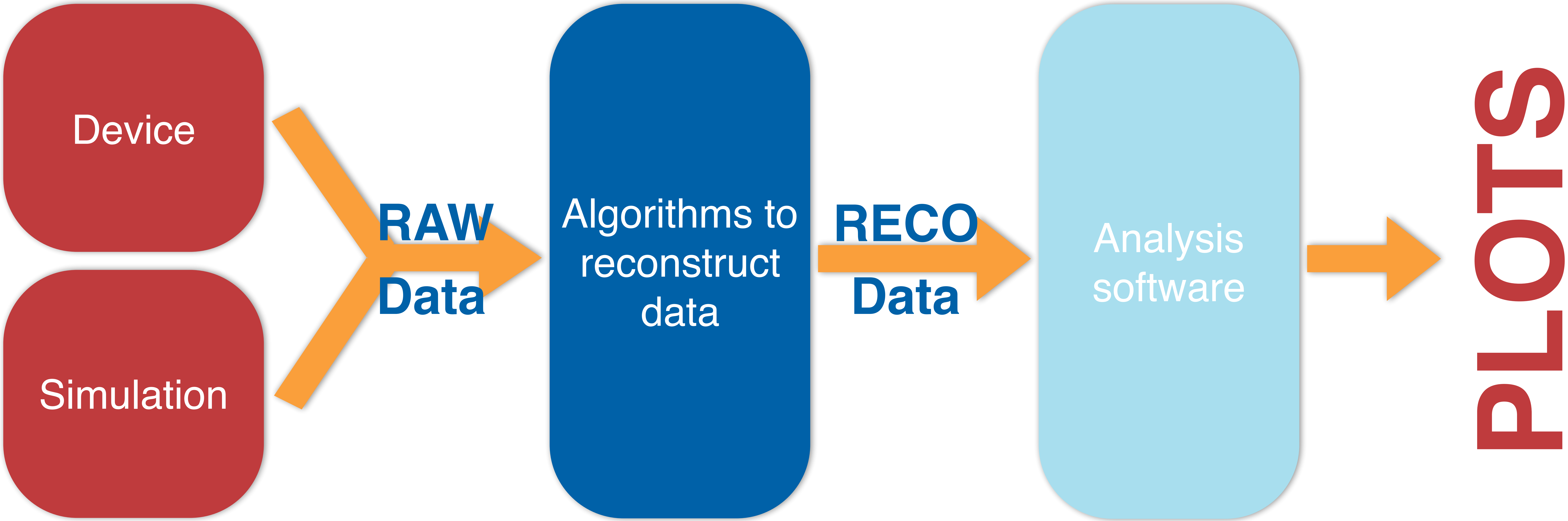
The Detector Information



The Scientific Process in HEP

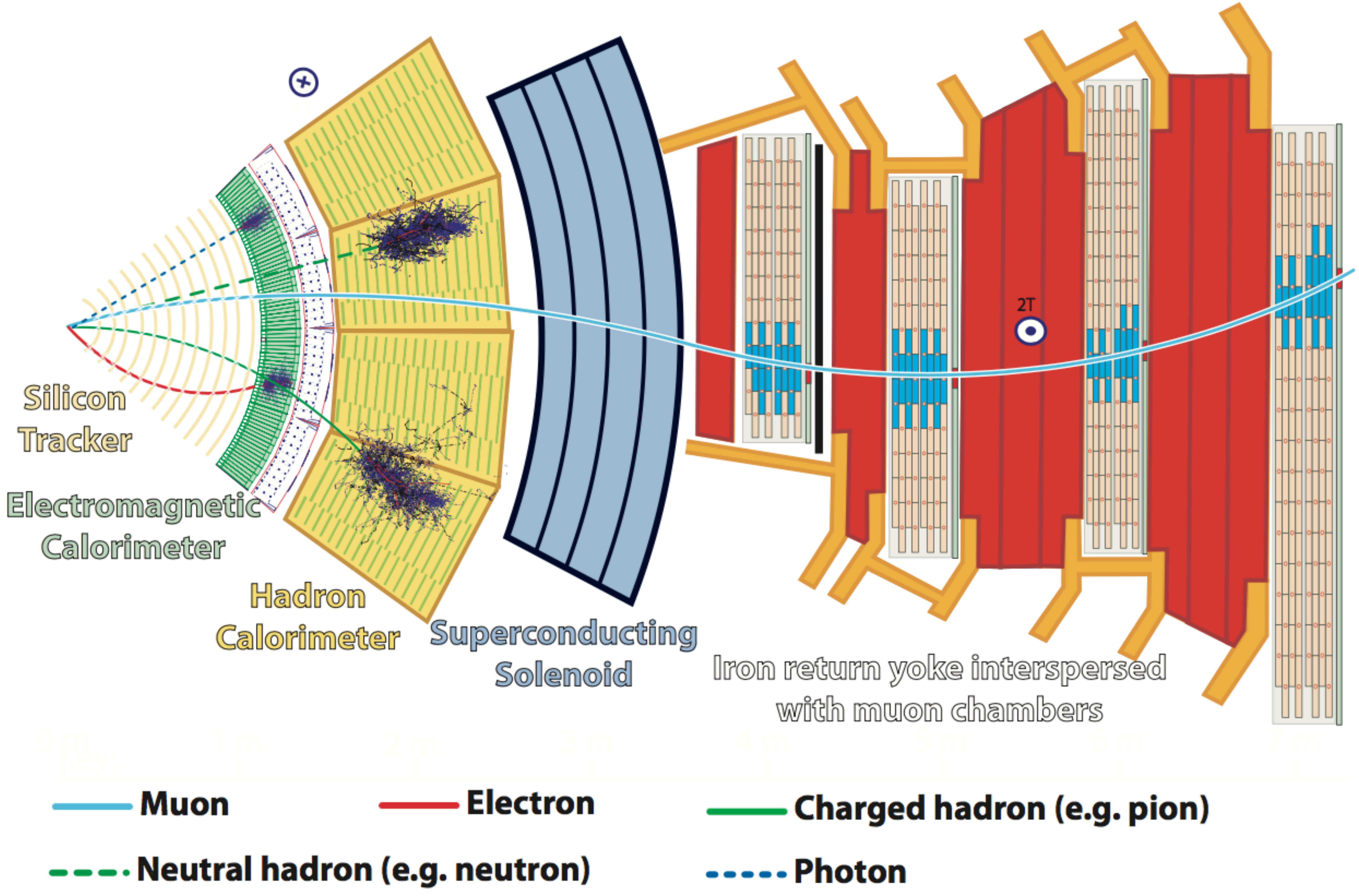


Software & Computing



- Detector signals (and equivalent simulated signals) need to be reconstructed to learn about the particles that produced them

Event Reconstruction



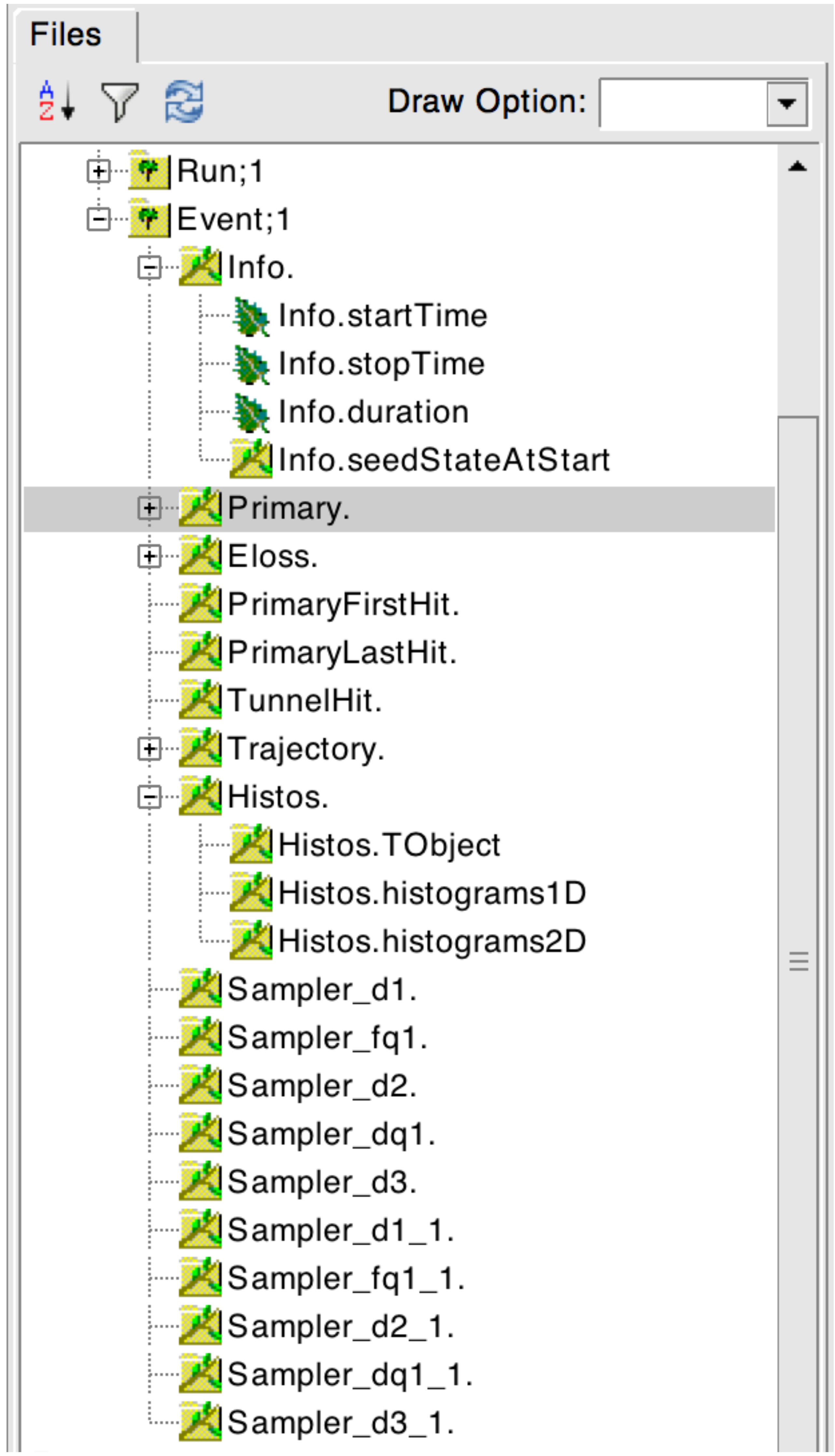
- The reconstructed events are then used for analysis

Software & Computing

- **Central:** organized processing/production, one software stack/framework per experiment (C++), one or few output sets, shared by large parts of the experiments for analysis
- **Chaotic:** smaller groups down to individuals explore the data for analysis, individually implemented analysis code, because of data volumes requires data reduction to keep notion of interactivity

The files and file content: The Event Data Model (EDM)

- An Event starts as a collection of the RAW data from a detector or MC event, stored as a single entity in memory, a C++ type-safe container called edm::Event.
  - Any C++ class can be placed in an Event, there is no requirement on inheritance from a common base class.
- As the event data is processed, products (of producer modules) are stored in the Event as reconstructed (RECO) data objects.
- The Event thus holds all data that was taken during a triggered physics event as well as all data derived from the taken data.
- The Event also contains metadata describing the configuration of the software used for the reconstruction of each contained data object and the conditions and calibration data used for such reconstruction.
- The Event data is output to files browsable by ROOT. The event can be analyzed with ROOT and used as an n-tuple for final analysis.
- Products in an Event are stored in separate containers, organizational units within an Event used to collect particular types of data separately. There are particle containers (one per particle), hit containers (one per subdetector), and service containers for things like provenance tracking.



https://root.cern.ch/input-and-output

The data tiers and workflows

- Data Tiers (collections of classes with specific meaning)
  - RAW: RAW detector data
  - RECO: Reconstructed Information
  - AOD: Analysis Object Data, subset of RECO
  - MINIAOD: Slimmed subset of AOD
- Simulation Data Tiers
  - GEN: event generator
  - GEN-SIM: simulated information
  - GEN-SIM-RECO: reconstructed information of simulation
  - AODSIM
  - MINIAODSIM
- Special files with flat structure (no classes)
  - NTuples

- Data Workflows:
  - Reconstruction: input RAW, output AOD/MINIAOD
  - produce MINIAOD: input AOD, output MINIAOD
  - AOD analysis: input AOD, output NTuple
  - MINIAOD analysis: input MINIAOD, output NTuple
  - User Analysis: input NTuple, output plots, tables
- Simulation Workflows
  - Generation: input nothing, output GEN
  - Simulation: input GEN, output GEN-SIM
  - Digitization/Reconstruction: input GEN-SIM, output AODSIM/MINIAODSIM
    - Digitization reads one to hundreds of additional events from secondary files
  - produce MINIAODSIM: input AODSIM, output MINIAODSIM
  - AODSIM analysis: input AODSIM, output NTuple
  - MINIAODSIM analysis: input MINIAODSIM, output NTuple
  - User Analysis: input NTuple, output plots, tables

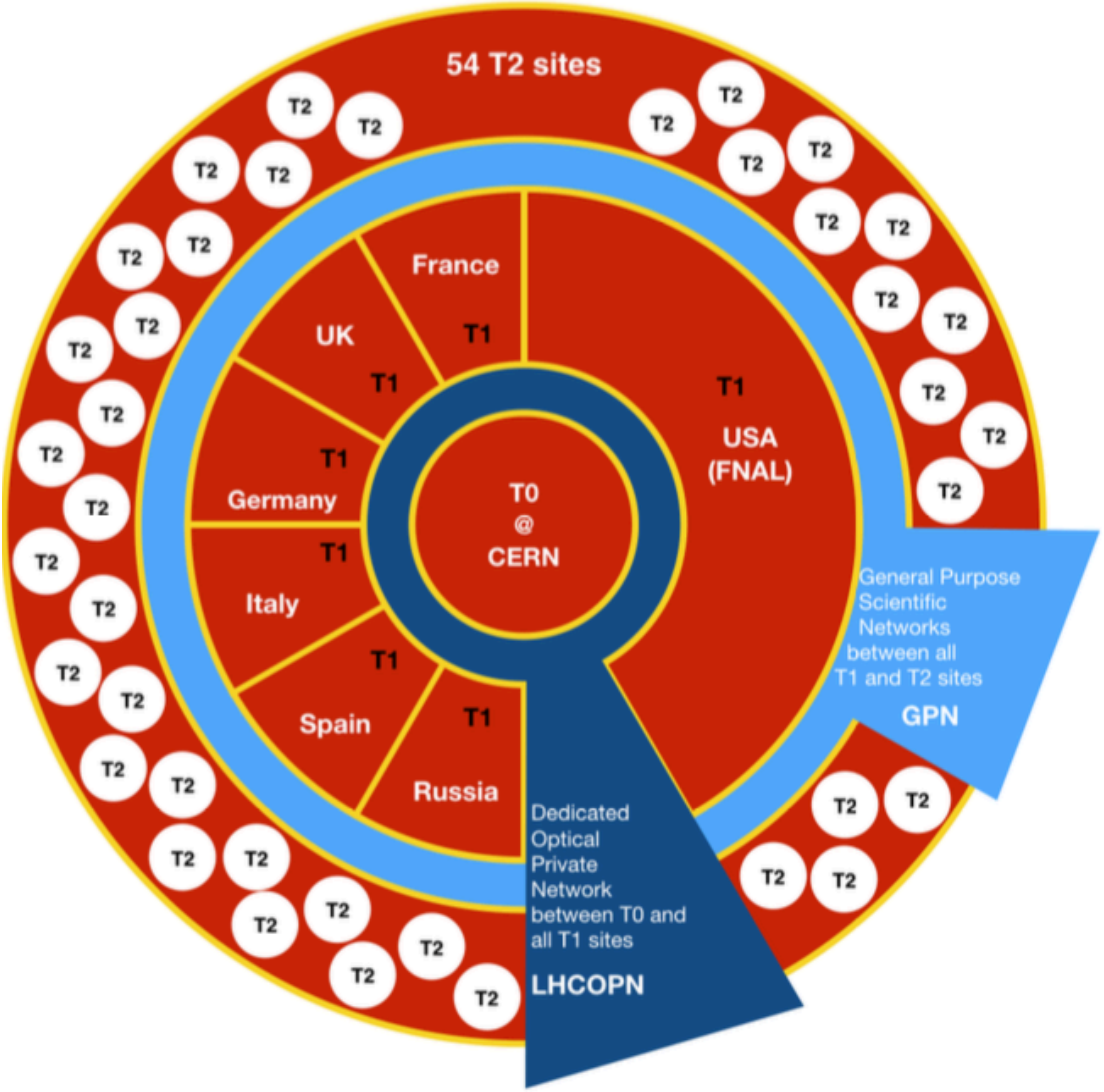
Computing Infrastructure for CMS

- Over 70 sites distributed across the world

Distributed Computing Model

- CMS resources world-wide
  - ~150,000 cores
  - ~75 Petabyte Disk
  - ~100 PB used tape space

- Strong networks connecting the individual sites
  - Weekly transfer volume between all sites: 4-6 Petabyte
  - Total LHC Trans-Atlantic network capacity: 340 Gigabits per second

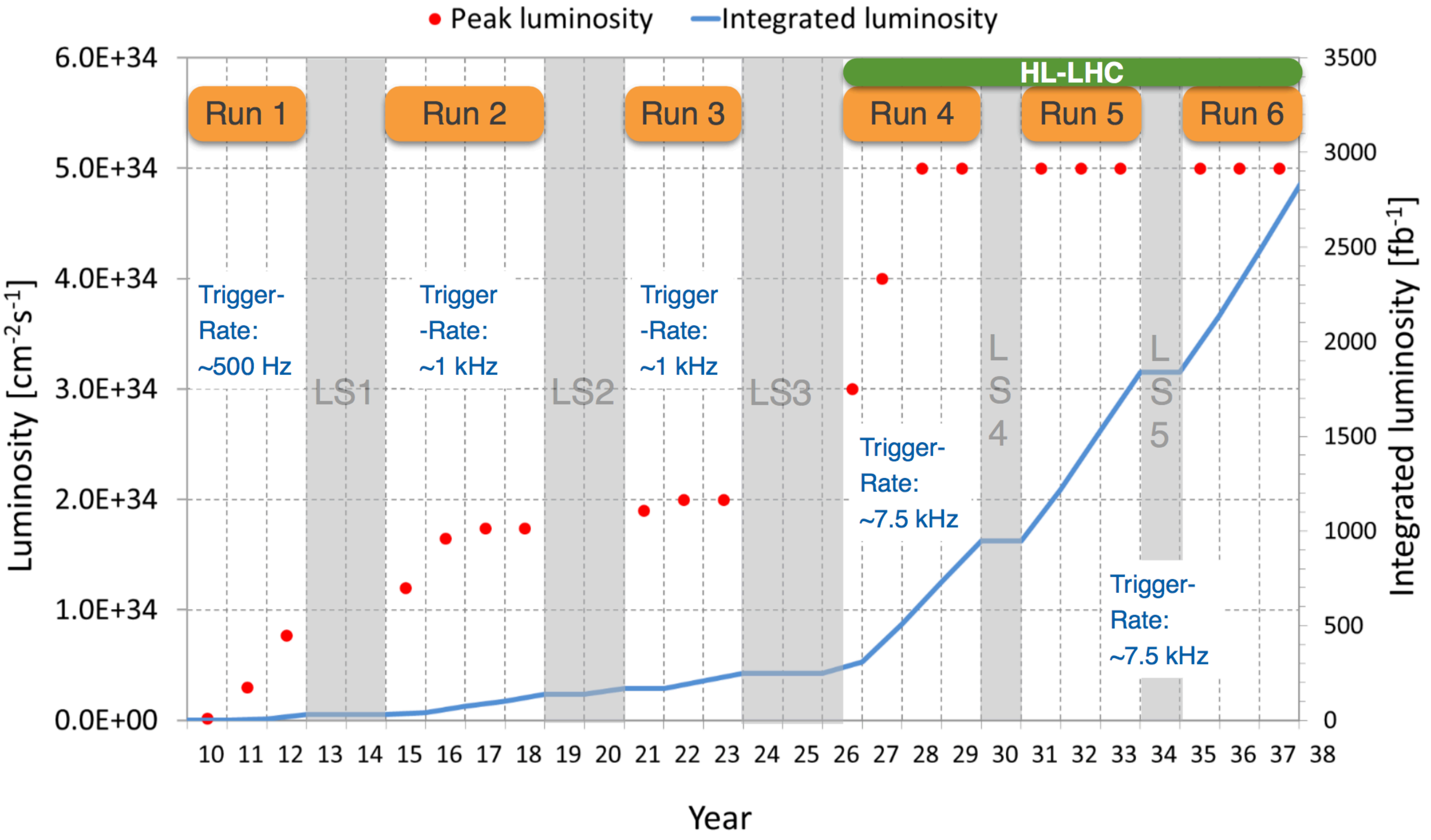


The sites, the transfers and the access methods

- Files are grouped in datasets
  - Datasets have same physics content (trigger or simulated collision type)
- Datasets are distributed across disk at all the sites automatically
  - Balanced and replicated according to popularity (the more in demand a dataset is, the more popular it is, the more it is replicated at different sites) ➜ more resources for more popular datasets
- Jobs are sent to sites which hold input dataset

- Additional access method: data federation
  - xrootd based data federation can discover files stored at all sites and stream file content over WAN (EDM optimized for WAN access with file content caching, etc.)

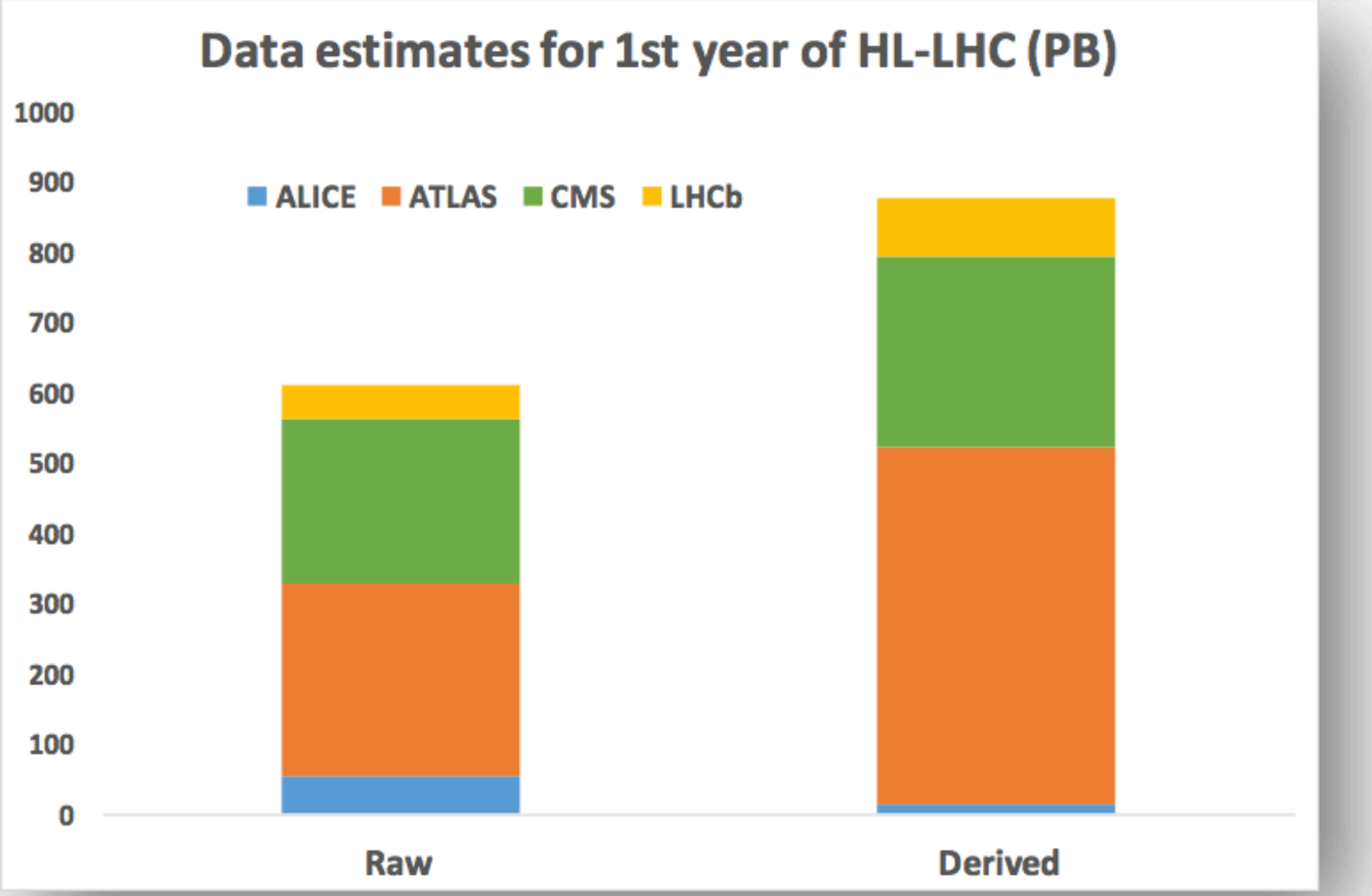
HL-LHC

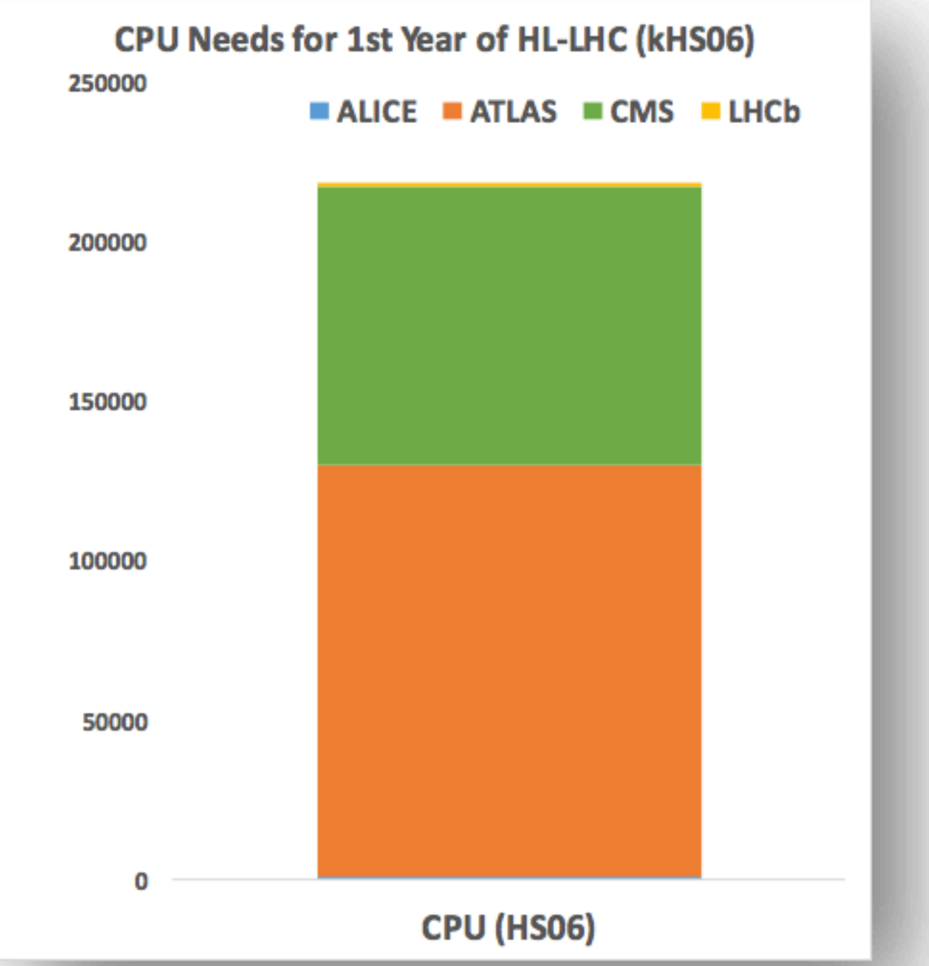


Peak luminosity
Integrated luminosity
HL-LHC
Run 1
Run 2
Run 3
Run 4
Run 5
Run 6
LS1
LS2
LS3
LS4
LS5
Trigger-Rate: ~500 Hz
Trigger-Rate: ~1 kHz
Trigger-Rate: ~1 kHz
Trigger-Rate: ~7.5 kHz
Trigger-Rate: ~7.5 kHz
6.0E+34
5.0E+34
4.0E+34
3.0E+34
2.0E+34
1.0E+34
0.0E+00
Luminosity [cm-2s-1]
3500
3000
2500
2000
1500
1000
500
0
Integrated luminosity [fb-1]
10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38
Year

The HL-LHC problem

Estimates of resource needs for HL-LHC





Data:
- Raw 2016: 50 PB → 2027: 600 PB
- Derived (1 copy): 2016: 80 PB → 2027: 900 PB

CPU:
- x60 from 2016

Technology at ~20%/year will bring x6-10 in 10-11 years

- Simple model based on today's computing models, but with expected HL-LHC operating parameters (pile-up, trigger rates, etc.)
- At least x10 above what is realistic to expect from technology with reasonably constant cost

CERN WLCG 8 October 2016 Ian Bird 10

- The goal of HL-LHC is unchanged:
  - Record collisions
  - Simulate collisions
  - Reconstruct everything
  - Analyze reconstructed information
- What is changed
  - More data (higher trigger rate, larger events)
  - More complicated reconstruction (new detectors with higher granularity, more intense collisions)