Finding the Higgs boson: the story from the software and computing perspective

Ken Bloom

CERN openlab summer lectures

12 July 2016

Why is this man smiling?

Because software and computing enabled
the discovery of the Higgs boson!

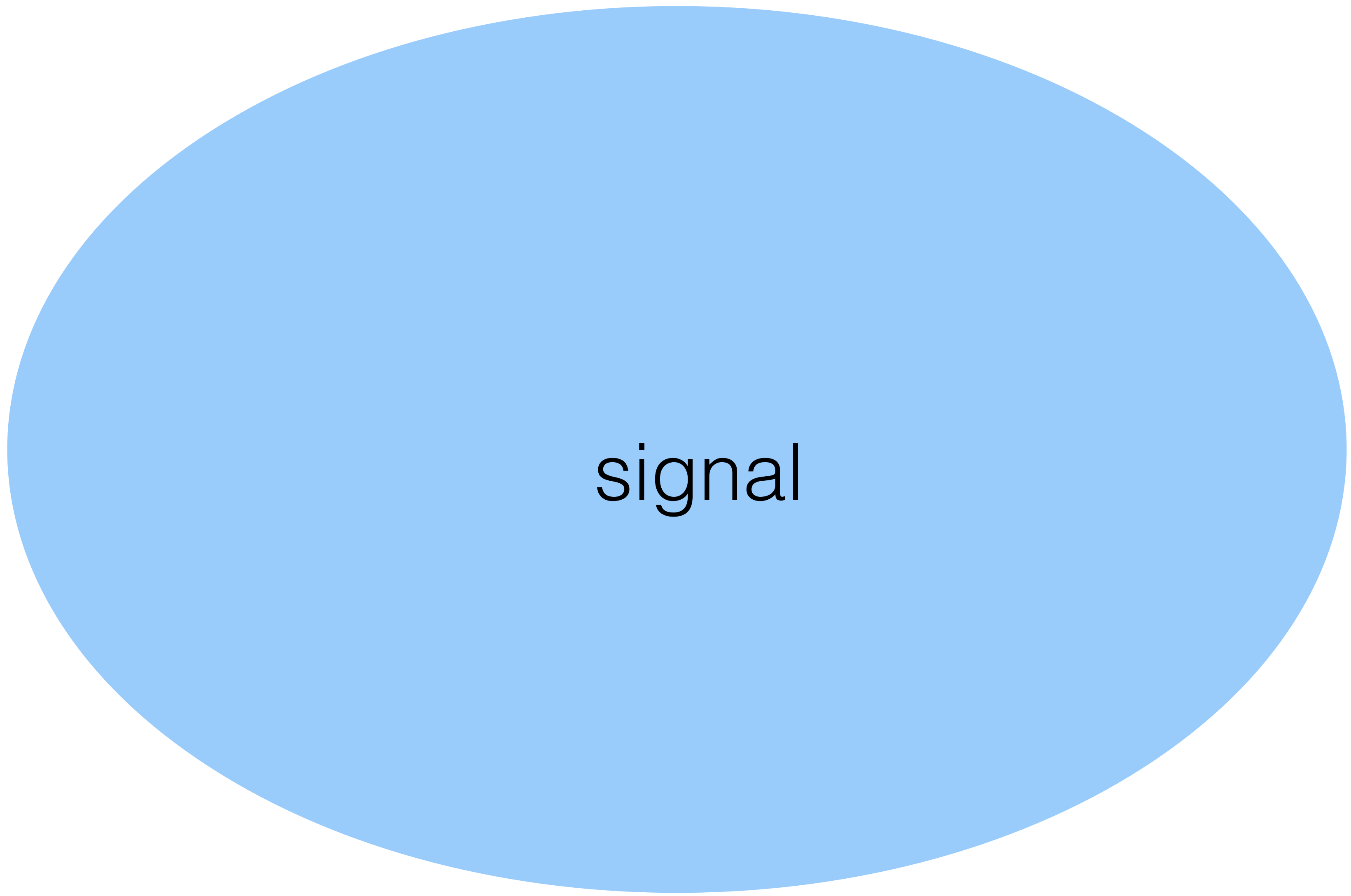
Why is this man talking?

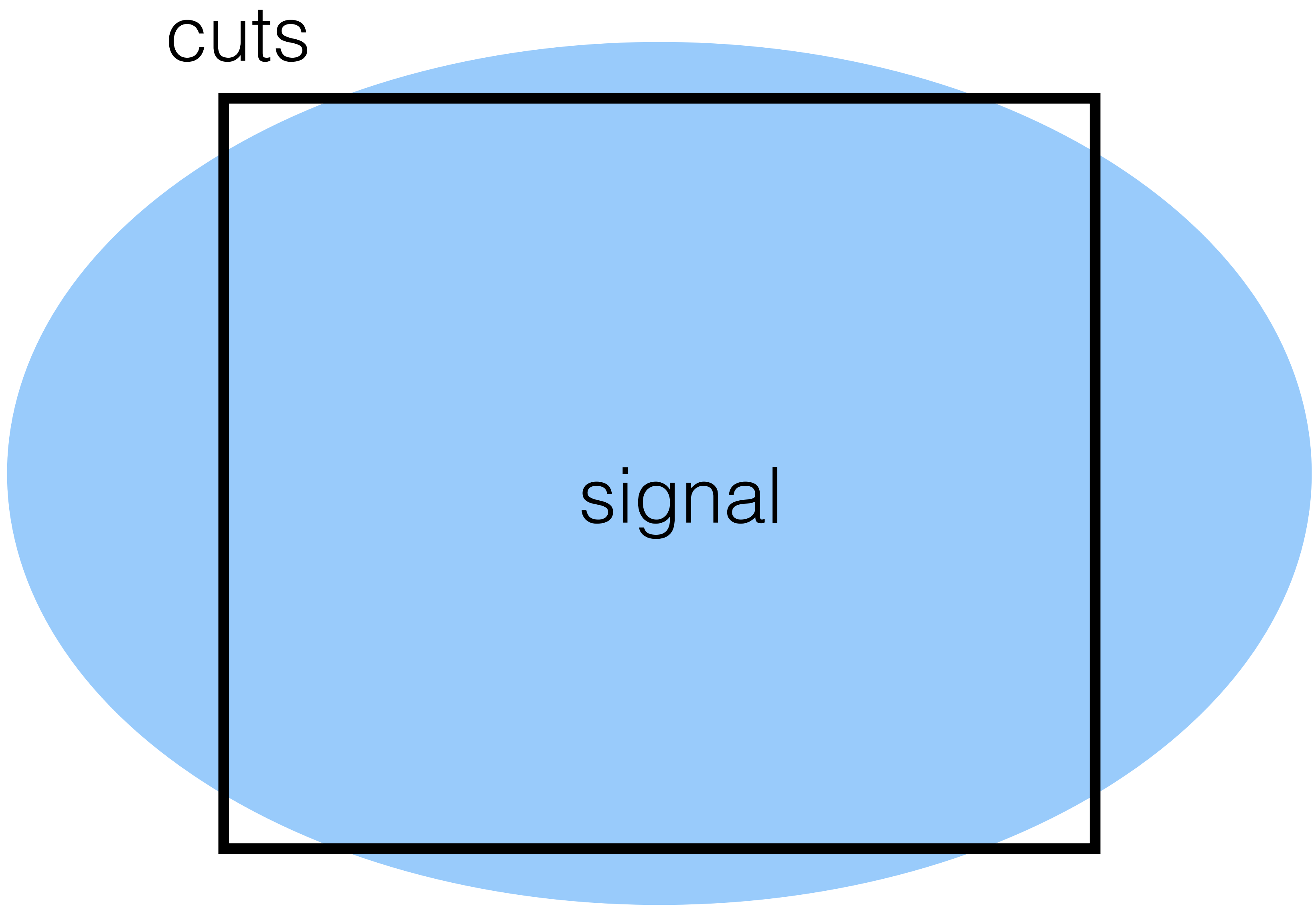
- I am a professor in Physics and Astronomy at the University of Nebraska-Lincoln

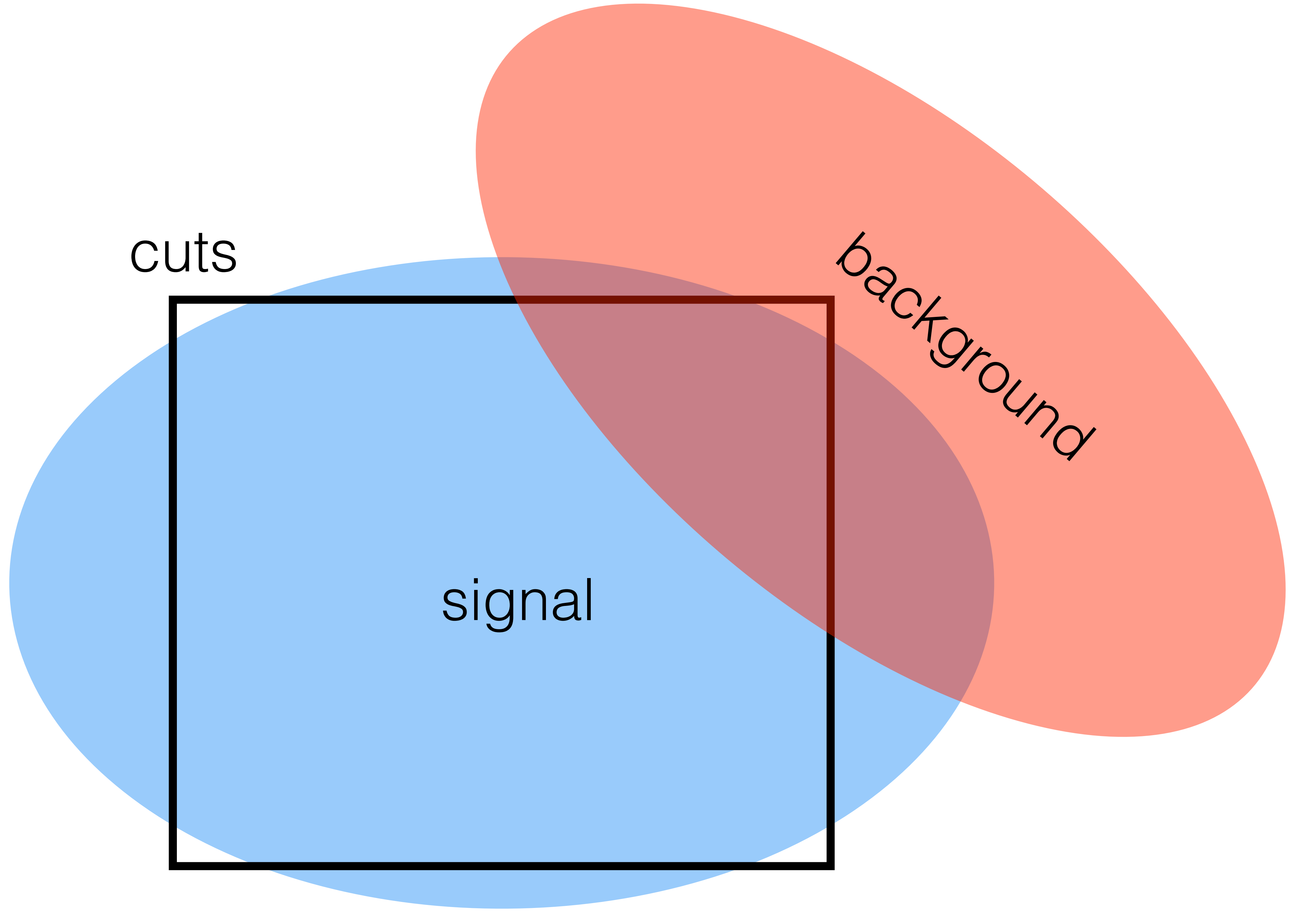
- I have no formal training in computer science!
- But I've gotten a lot of on-the-job training:
  - Designed and implemented particle reconstruction algorithms in C++ when C++ was new in HEP
  - Co-led the development and operation of a computing center for the Compact Muon Solenoid (CMS) experiment
  - Project leader for "Any Data, Anytime, Anywhere", the CMS world-wide data federation
  - For past 1.5 years, software and computing manager for the U.S. CMS Operations Program
- For me, computing is a tool to get my science done — and to make it easy for my collaborators to do the same

Particle physics measurements

- All measurements are ultimately “counting experiments” — in a given dataset of discrete “events”, how many times do you observe events of a type representing a particular physics process?
  - Quantum mechanics predicts how often different processes occur, but only as a probability
  - Set criteria (“cuts”) to identify “signal” events, count them
- But:
  - Efficiency: Cuts might exclude some signal events
  - Background: Other events might look similar to the signal events, contaminating the sample
- Larger efficiency typically implies more background; selection must be optimized for the most accurate estimate of the event rate (maximize $S^2/(S+B)$)


signal

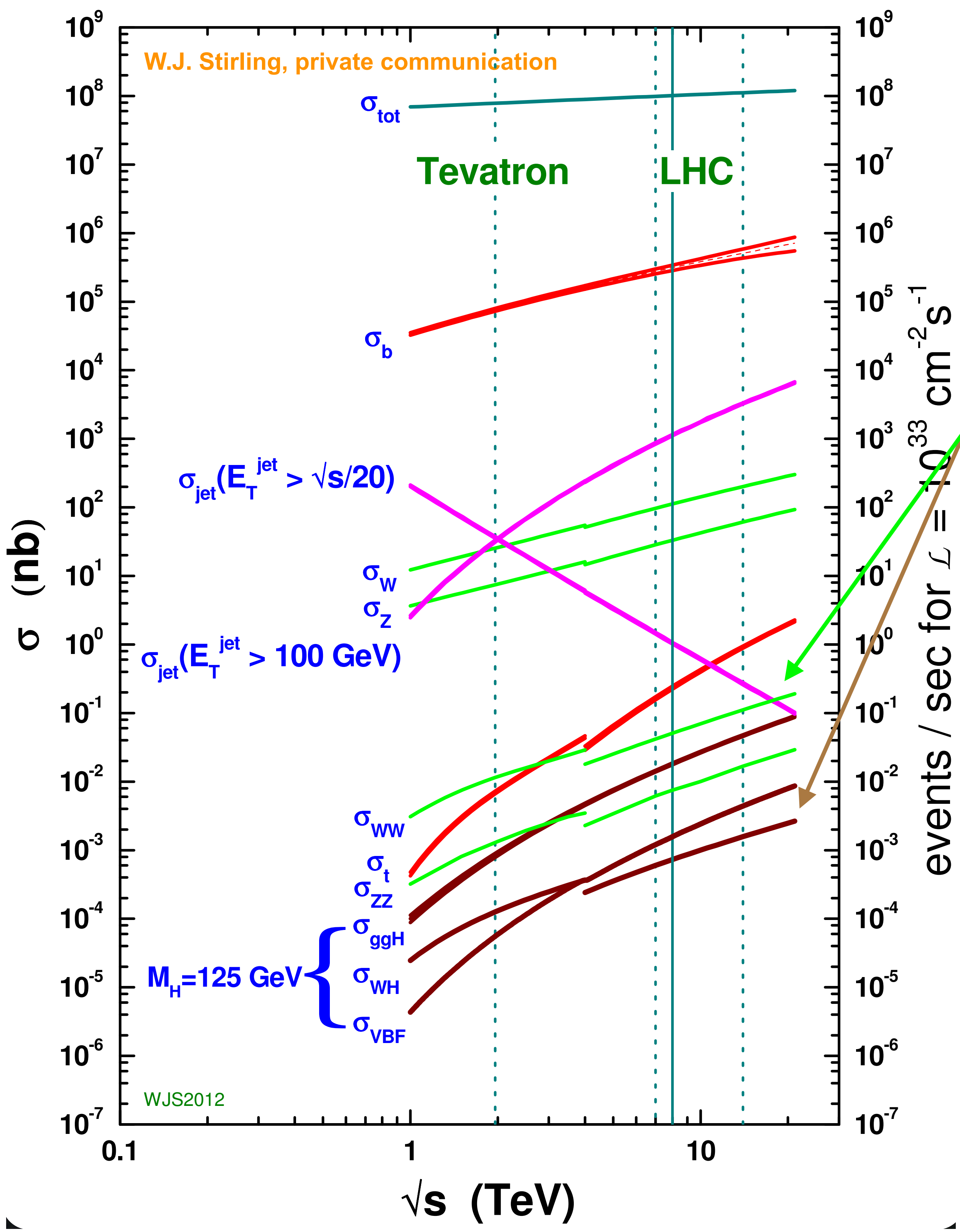

cuts
signal


cuts
background
signal

Particle physics measurements

- All measurements are ultimately “counting experiments” — in a given dataset of discrete “events”, how many times do you observe events of a type representing a particular physics process?
  - Quantum mechanics predicts how often different processes occur, but only as a probability
  - Set criteria (“cuts”) to identify “signal” events, count them
- But:
  - Efficiency: Cuts might exclude some signal events
  - Background: Other events might look similar to the signal events, contaminating the sample
- Larger efficiency typically implies more background; selection must be optimized for the most accurate estimate of the event rate (maximize $S^2/(S+B)$)

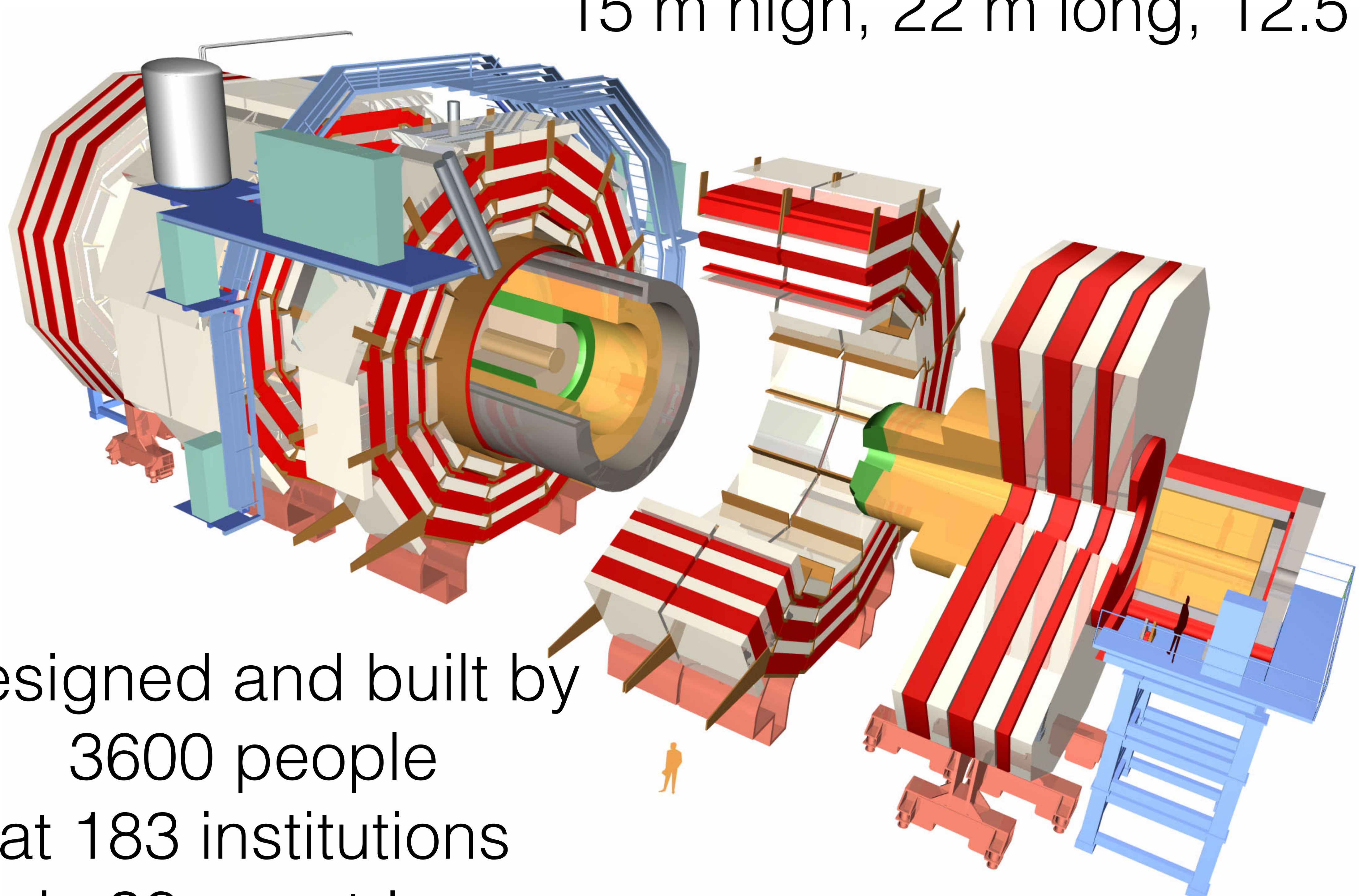
Needles in a needle factory



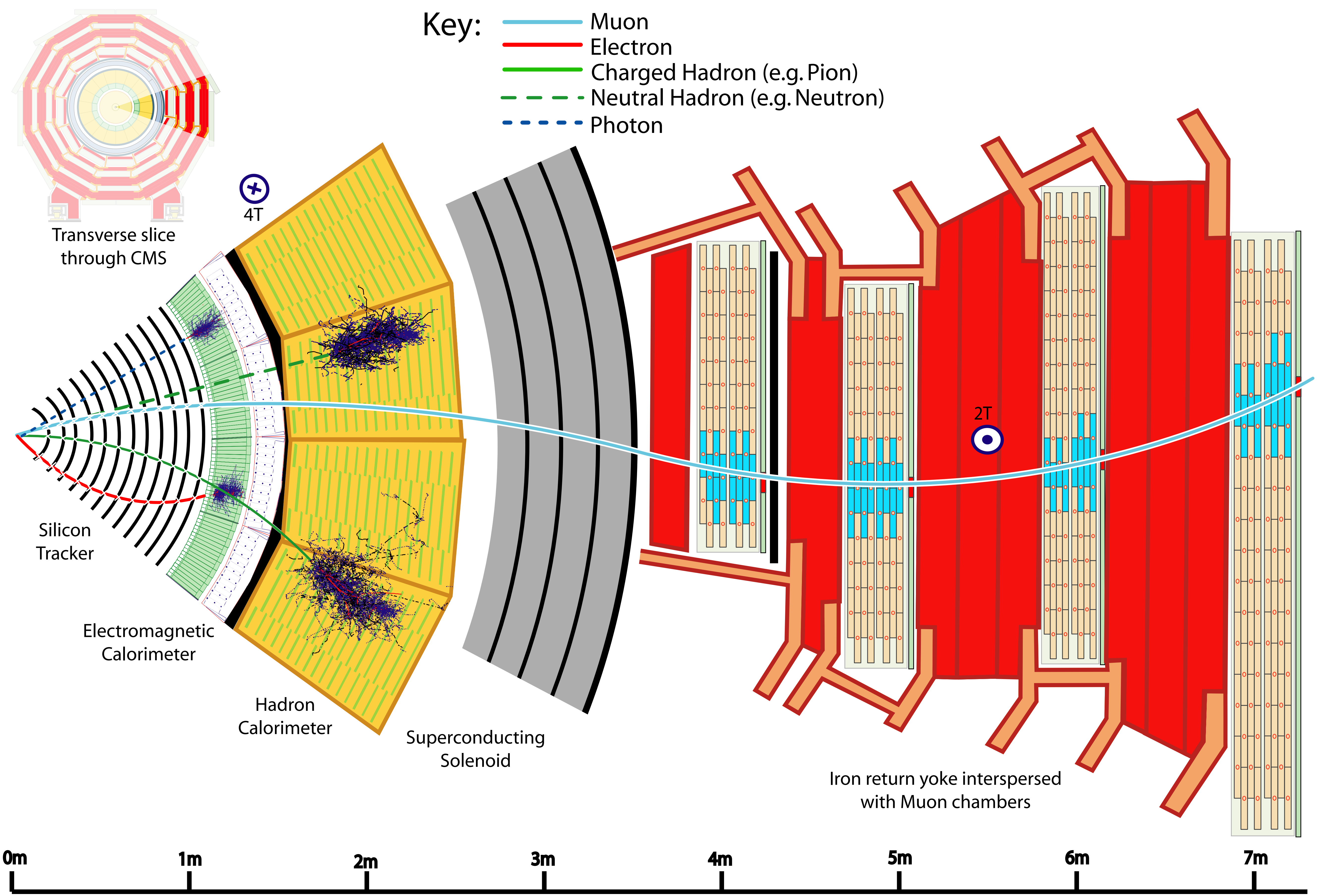
- These are the Higgs events
- But these are the Higgs background events!
- We build our detectors to allow us to make the greatest distinction between signal and background while maintaining efficiency, but software and computing are needed to realize that

Compact Muon Solenoid (CMS)

15 m high, 22 m long, 12.5 tons



Designed and built by
3600 people
at 183 institutions
in 38 countries
over about 20 years



Key:
Muon
Electron
Charged Hadron (e.g. Pion)
Neutral Hadron (e.g. Neutron)
Photon
Transverse slice through CMS
4T
2T
Silicon Tracker
Electromagnetic Calorimeter
Hadron Calorimeter
Superconducting Solenoid
Iron return yoke interspersed with Muon chambers
0m
1m
2m
3m
4m
5m
6m
7m

Basic ingredients

- Record data from the detector
  - Data quanta are “events”, single LHC beam crossings, statistically independent of each other
- Determine the particles produced in each event, and their kinematic properties
- Make selection cuts on the basis of the above info
- Make estimates of background rates and detection efficiencies
  - This often relies heavily on simulations
- Compare results with predictions from theories

The data path

- CMS has about 80 million readout channels
  - Each can produce ~40 bytes/event
  - LHC beams collide at 40 MHz
  - 128 PB/s of data? Uh oh.
- Don’t read out every channel
  - Most channels are empty, or electronic noise
  - Eliminate them from readout using algorithms within the electronics
  - Read out only ~20,000 channels per event
  - 0.8 MB/event x 40 MHz = 32 TB/s?

The data path

- Don’t read out every event
  - Most events are not interesting from a physics standpoint anyway (remember that plot!)
  - Must make fast decisions about which events to keep, using limited information
    - Combination of electronics and software
    - Select on detector patterns indicative of single or multiple high-energy particles
  - Two-stage trigger reduces rate from 40 MHz to < 100 kHz and then 1 kHz
    - 39,999 of every 40,000 collisions are discarded without any human intervention
- 0.8 GB/s data rate, or 5 PB/year
  - Partitioned into “datasets” by the detector patterns selected on
  - Trigger rate is set not by the limits of the DAQ system, but by how much data the computing systems can accommodate!

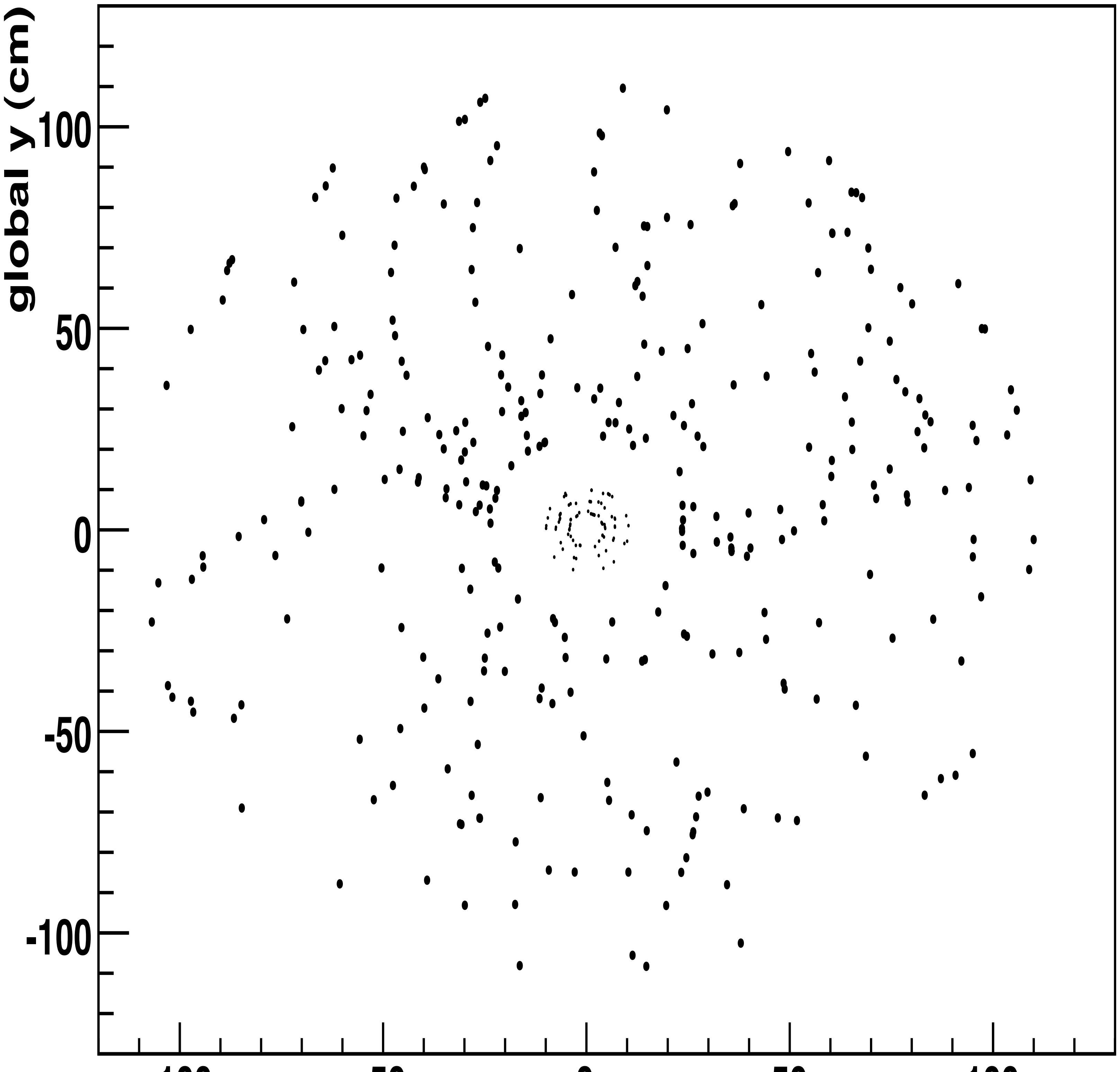
From bits to particles

- What you get: each readout channel gives the amount of charge deposited on an amplifier, and/or the time that the charge arrived
- What you want: the energy and momentum of each of hundreds of particles produced in each collision, and the identity of each of those particles
- The big gap between the two is bridged by *event reconstruction* which is in turn supported by *alignment and calibration*

Example: charged particle tracks

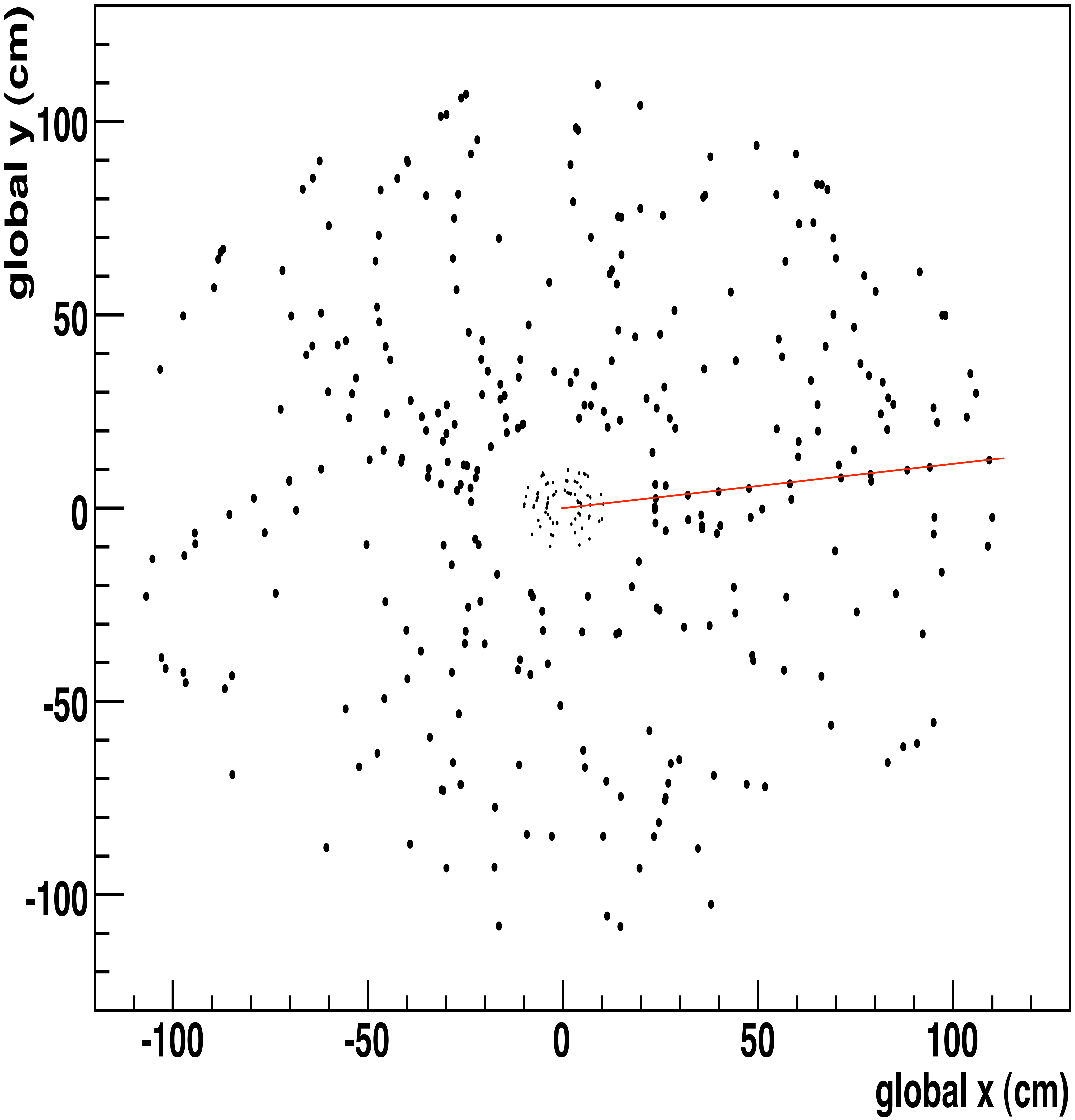
- Tracking detectors record locations in space ("hits") where charged particles have passed by
- Identify collections of hits that are consistent with the path of a single particle
- The curvature of the path is related to the momentum of the particle
- Harder than it sounds:
  - Huge pattern recognition challenge to identify the right collections of hits
  - Particle momentum varies as the particle travels and loses energy through interaction with matter

Can you find the high-momentum particle?



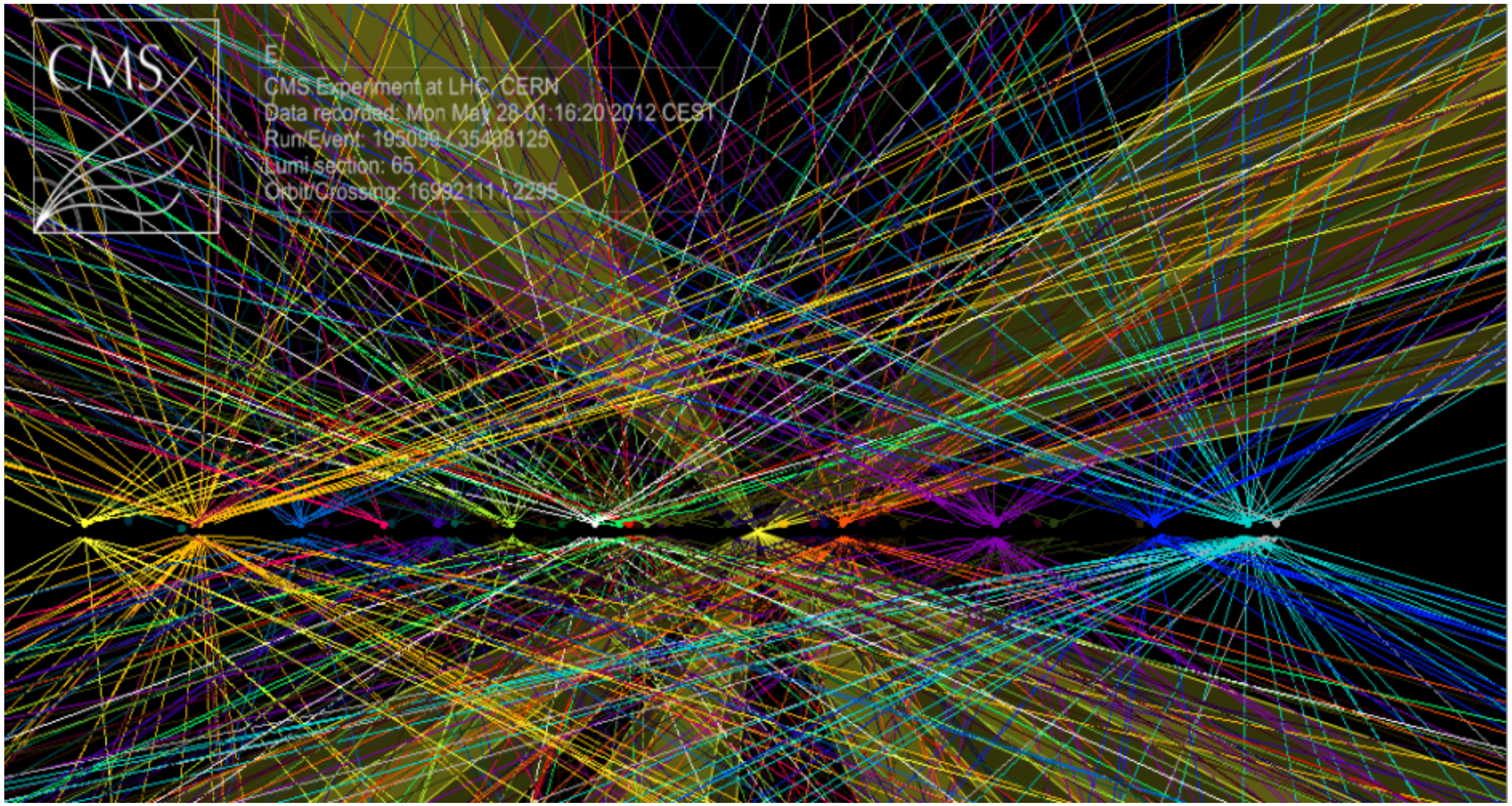
- Hint: charged particles travel in helical paths, with the radius of the helix proportional to the particle momentum

Can you find the high-momentum particle?



- This kind of pattern recognition is one of the most expensive computations we do
- This is actually a relatively simple event….

Pileup



- A typical LHC event has 20 proton collisions on average!
- Most collisions aren't interesting, but you need to sort out everything to get to the interesting stuff
- Gather sets of tracks that originate from the same collision

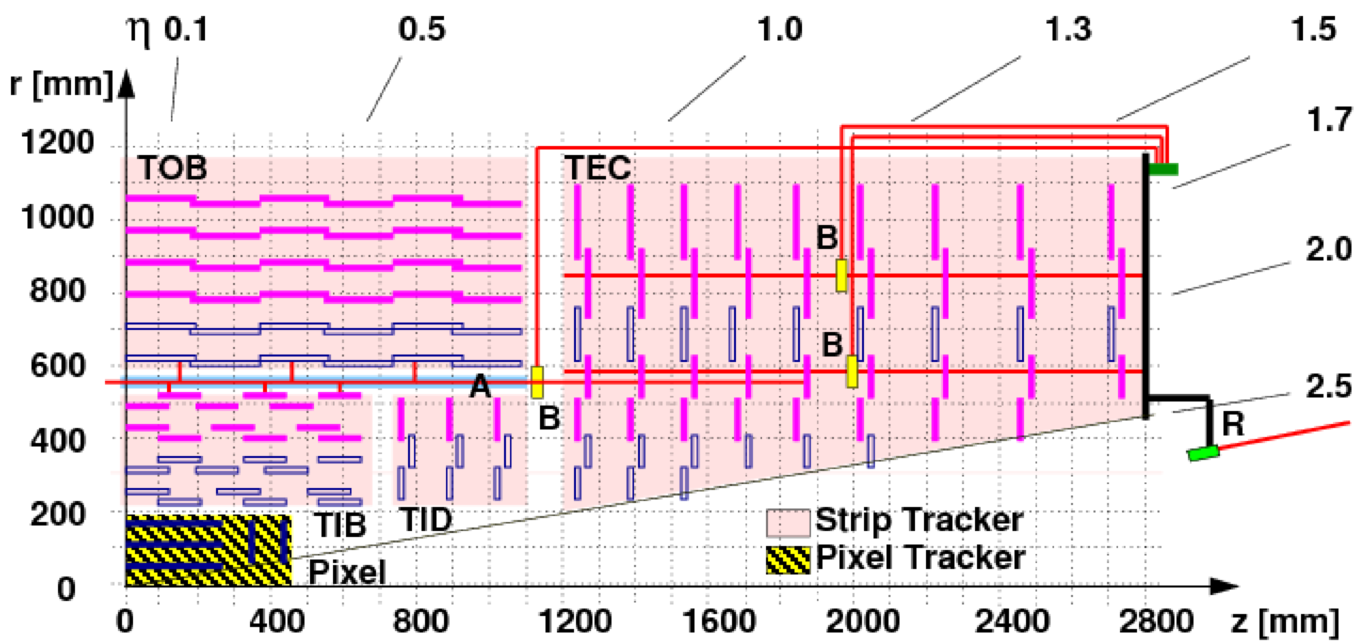
Other examples

- Track finding is the most computationally-intense event reconstruction problem, but not the only one:
  - A single particle can deposit energy in multiple elements of the calorimeter — how to decide which elements should be clustered together?
  - Some short-lived particles produce sprays of many longer-lived particles (“jets”) — how to decide which reconstructed particles belong in the same jet?
  - Some particles travel some distance before they decay into other particles — how to gather those particles to reconstruct the location of decay?

Alignment and calibration

- Track-finding algorithms rely on knowing the locations of the hits
- The ~16K physical elements of the tracker have nominal locations, but their actual placement is not known as accurately as the 10-30 micron intrinsic resolution individual hits
- Thus the elements need to be aligned *in situ*, using actual particles from proton collisions
  - A bootstrapping problem!

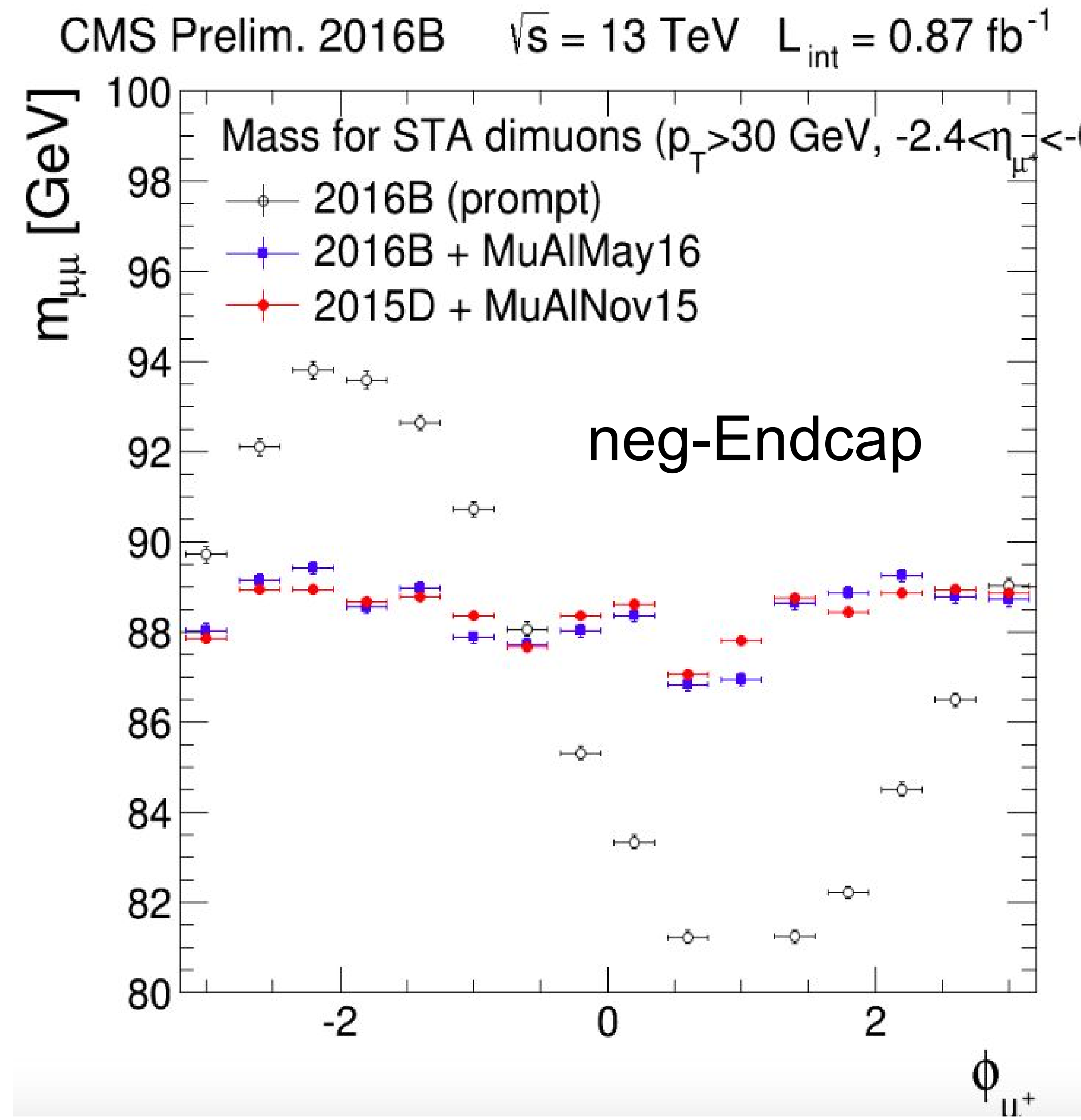
Alignment and calibration



- Each element is a rectangular wafer; each needs three coordinates and three angles to locate and orient it, and also account for potential wafer bowing — 200K parameters to determine
- Big matrix inversion problem!
    - Attacked through clever linear algebra, and also parallel computing with multiple threads and shared memory

Alignment and calibration

- Real impacts on the quality of the physics!
- Improved alignment → more accurate measurement of track parameters → better resolution on kinematic quantities used for event selection → better signal to noise



Alignment and calibration

- Other examples
  - Other elements of the detector must also be aligned, e.g. calorimeter pieces
  - Each element of the calorimeter has a slightly different response to a particle of a given energy → each needs to be calibrated
- The information derived is stored in a database that can be accessed by CMS software
- Calibration and alignment can change over time, so database must be keyed on that

Simulations

- Experimental measurements rely extensively on simulations. Why?
  - You need to know what to look for!
  - Even if you know what to look for, how it manifests itself in the detector depends on many assumptions, which must be tested
- Goal: simulation samples should look as if they had been recorded by the detector
  - This requires multiple steps

Simulation steps

- Model the physics that takes place in a collision
  - Requires a theory that describes the interactions being studied, and a model for the initial protons
    - Theory might have undetermined parameters, protons aren't perfectly understood — these can be varied in the simulation as a test
  - It's quantum mechanics — sample a probability distribution describing the interaction
  - Output per event is a list of particles that emerges from the collision, and their momenta
  - Usually not the limiting factor in computation time

Simulation steps

- Model how each emerging particle would interact with the detector
  - Detailed models that depend on the type of incoming particle, type of material, kinematics….
  - And also a careful description of the detector material itself — quantity, geometry….
  - Standard codes for this in HEP (GEANT4), usually the most computationally expensive piece of the simulation
  - Extensive verification against well-understood data samples, tuning of simulation as needed

Simulation steps

- Model how these interactions are recorded by the electronics
  - Requires good understanding of the electronics themselves
  - Output format is that of real detector data
- Reconstruct this “data” just as one reconstructs the data
  - As the LHC beam intensities increase in coming years, this step will take more computation time than the simulation of particle interactions

Using simulation

- Most important job: modeling the efficiency of event selection
  - What fraction of events from a given physics process are actually detectable?
    - Some events won't have all objects within the detector volume
  - Number of events observed must be corrected for this
- Unavoidable uncertainties come from physics models
- Also have uncertainties in modeling detector response, controlled by comparisons to data

Using simulation

- Can also use simulation to model backgrounds
- Out of the box: simulate a process, assume that the rate and kinematic properties are correct
  - This can carry substantial uncertainties from physics modeling, especially on the rate
  - Safest for relatively small backgrounds
- Or, use it in conjunction with real data:
  - Create a data control sample that is dominated by a background process
  - Use simulation to estimate how many background events would be selected in the signal sample, given the number of events in the control sample
  - And use simulation to model the kinematics of the background events that appear in the signal sample

Computing perspectives: data

- But how is this all done from a computing standpoint??
  - (Note: this is how things are done on CMS, other experiments differ)
- First, there is the data
  - The fundamental unit of data is the event, representing a single LHC beam crossing
  - Events are grouped into ~1 GB files
    - These files are essentially the computing quanta
    - Detector events are grouped into files based on the triggers used to collect them
    - Simulated events are grouped based on the physics process being simulated

Computing perspectives: data

- Files are then grouped into “datasets”
  - A given dataset can have from several to thousands of files, so bookkeeping is required
- Need databases that track files and datasets
  - Which files are in which datasets, and what are their attributes?
  - Processing history of datasets?
  - Parent and derived datasets?
  - Location(s) of datasets?

Computing perspectives: distribution

- An LHC experiment produces many petabytes of detector data, information derived from it, and associated simulations
- As a practical matter, not stored all in one place, and by extension not processed all in one place
- Data is distributed to dozens of sites around the world, so need infrastructure to manage transfers of datasets and keep track of locations

Computing perspectives: processing

- Datasets are distributed all over the world
- Users are distributed all over the world
- Thus users want to access datasets that might be in a great variety of locations → get processing jobs to the right locations
- This is the realm of grid computing (different lecture)
  - Mechanisms to move jobs to sites hosting data, authenticate users at each site, retrieve outputs
- Or: run jobs locally and stream data to jobs
  - Data federations that allow jobs to identify data locations and then stream data with low latency

Computing perspectives: processing

- All LHC beam crossings are statistically independent, making both data processing and simulation embarrassingly parallel computing problems
- Can split a given task into many parallel jobs that can run simultaneously/independently
  - Typically create one task of many jobs per dataset, merge the job outputs once all jobs in the task are done
- But then need to manage all the tasks and jobs, across ~125K job slots available
  - Significant centralized infrastructure for this, to manage both the centrally-controlled production of simulation files, and the user-controlled processing of data for physics analysis

Computing perspectives: analysis

- A physicist must process many datasets:
  - Actual detector data, events collected with suitable trigger
  - Simulated sample of physics process of interest to estimate efficiency
  - Simulated samples of multiple other physics processes to estimate backgrounds
  - (Note that these samples are typically fully reconstructed already)

Computing perspectives: analysis

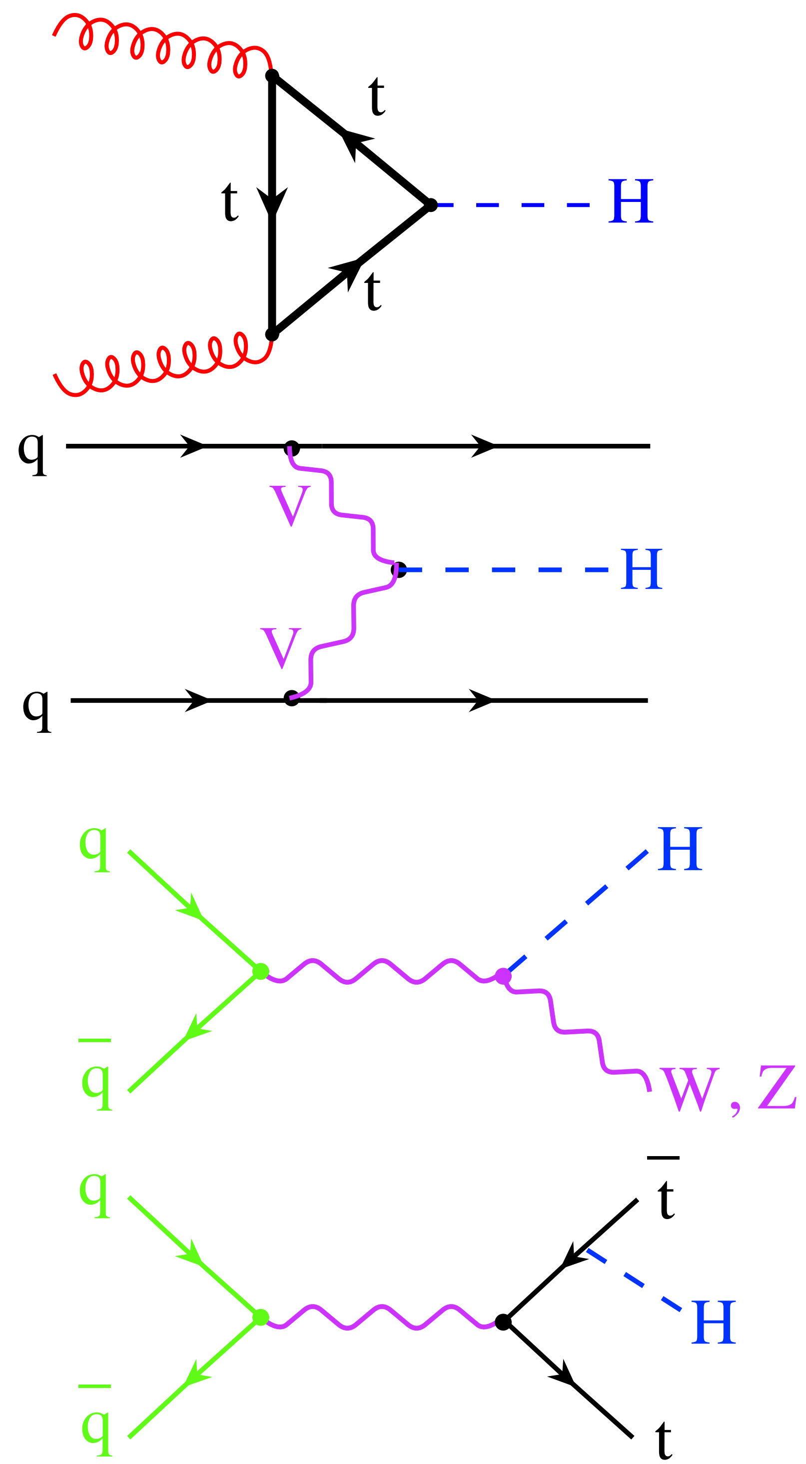
- This can amount to 100's of TB to process, cumbersome to do it frequently
- Data reduction is useful:
  - Run over all input datasets on the grid, once every few months, write smaller outputs to local computing
  - Run over those once/week as research questions are refined
  - Can then make very small outputs that can be processed on a desktop machine in minutes to quickly generate plots, make calculations

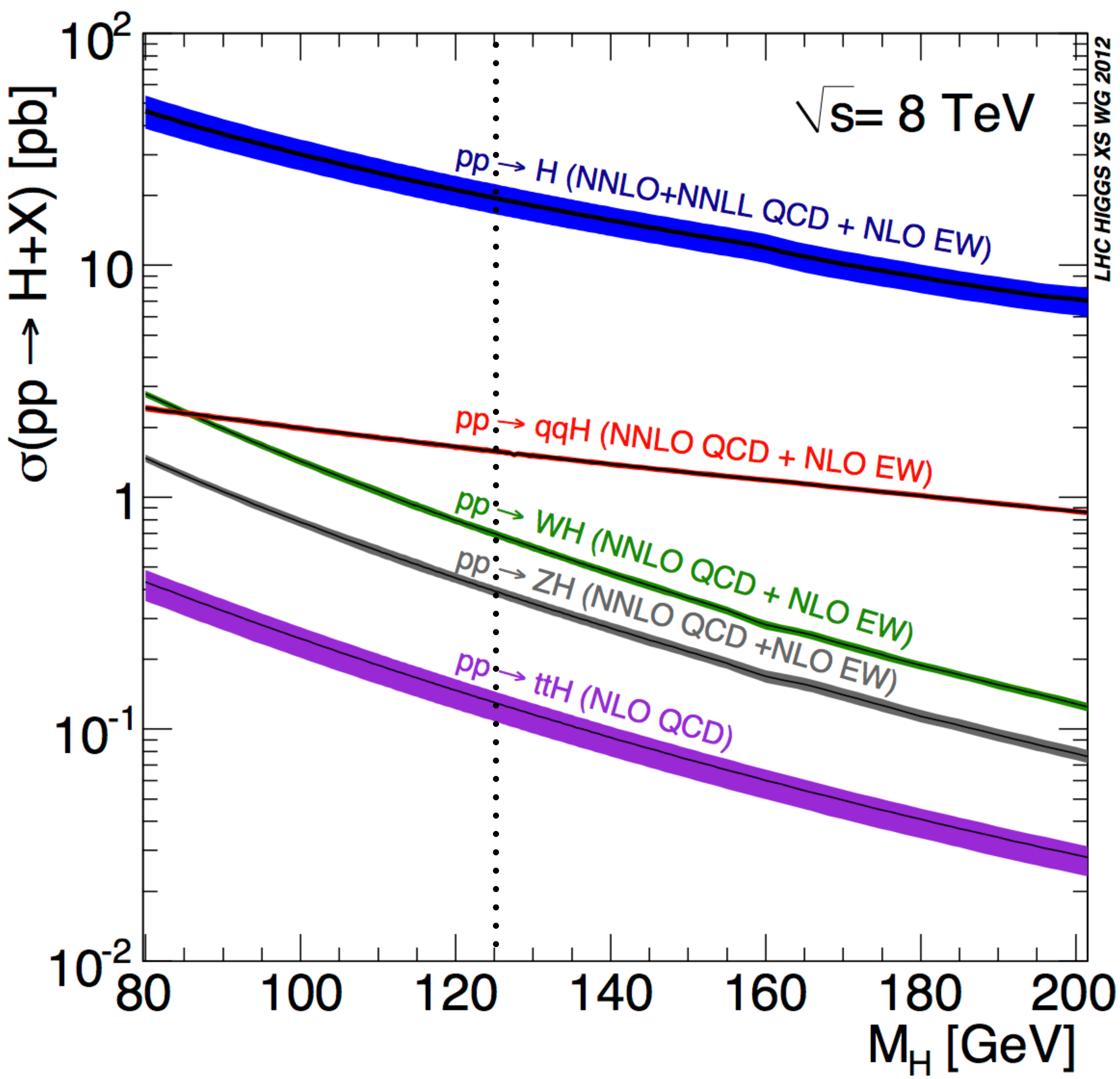
Higgs boson!

- Since the title advertised the Higgs boson, let's look at that:
  - Introduction to Higgs experiments
  - Higgs with large signal to background
  - Higgs with small signal to background
  - with software and computing considerations

How to recognize a Higgs

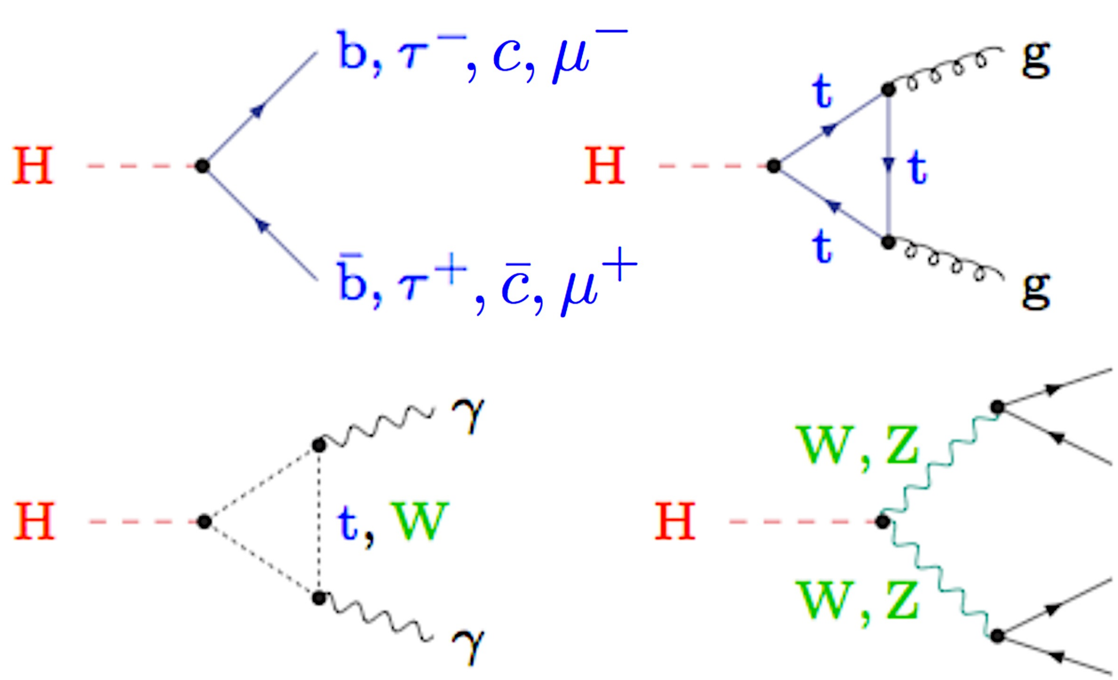
- Higgs mass determines rates of production mechanisms

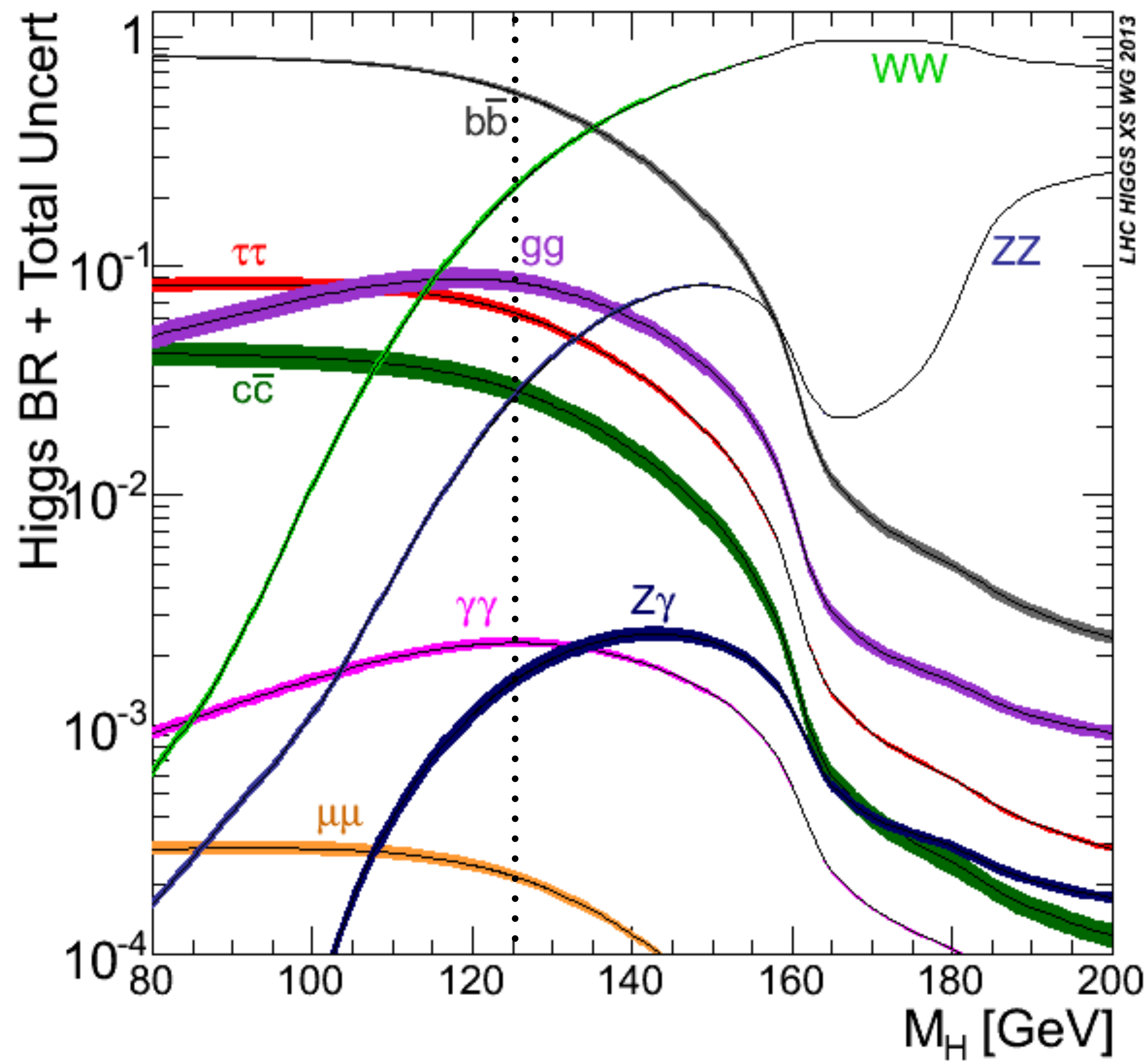




How to recognize a Higgs

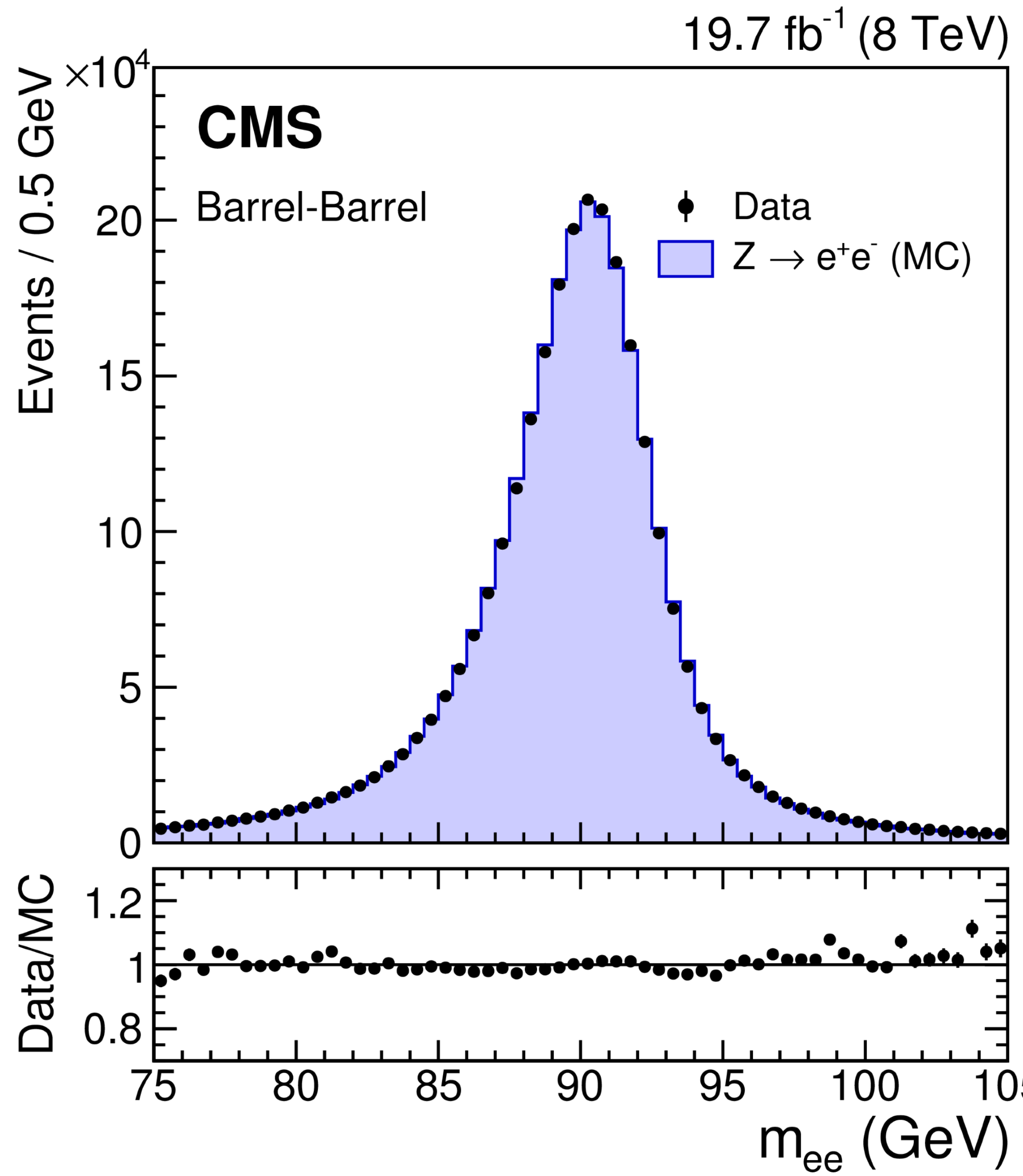
- Higgs mass determines decay rates to each final state





Higgs→γγ

- The Higgs decays to a pair of photons 0.3% of the time, fairly rare
- But CMS can measure photon energies to great precision, straightforward background estimation
- Calibration of photon energy measurement is critical
    - Response depends on variables such as temperature, radiation environment
    - Reconstruct particles of known masses
    - Each one of 75,458 lead tungstate crystals is calibrated to precision of a few per mille

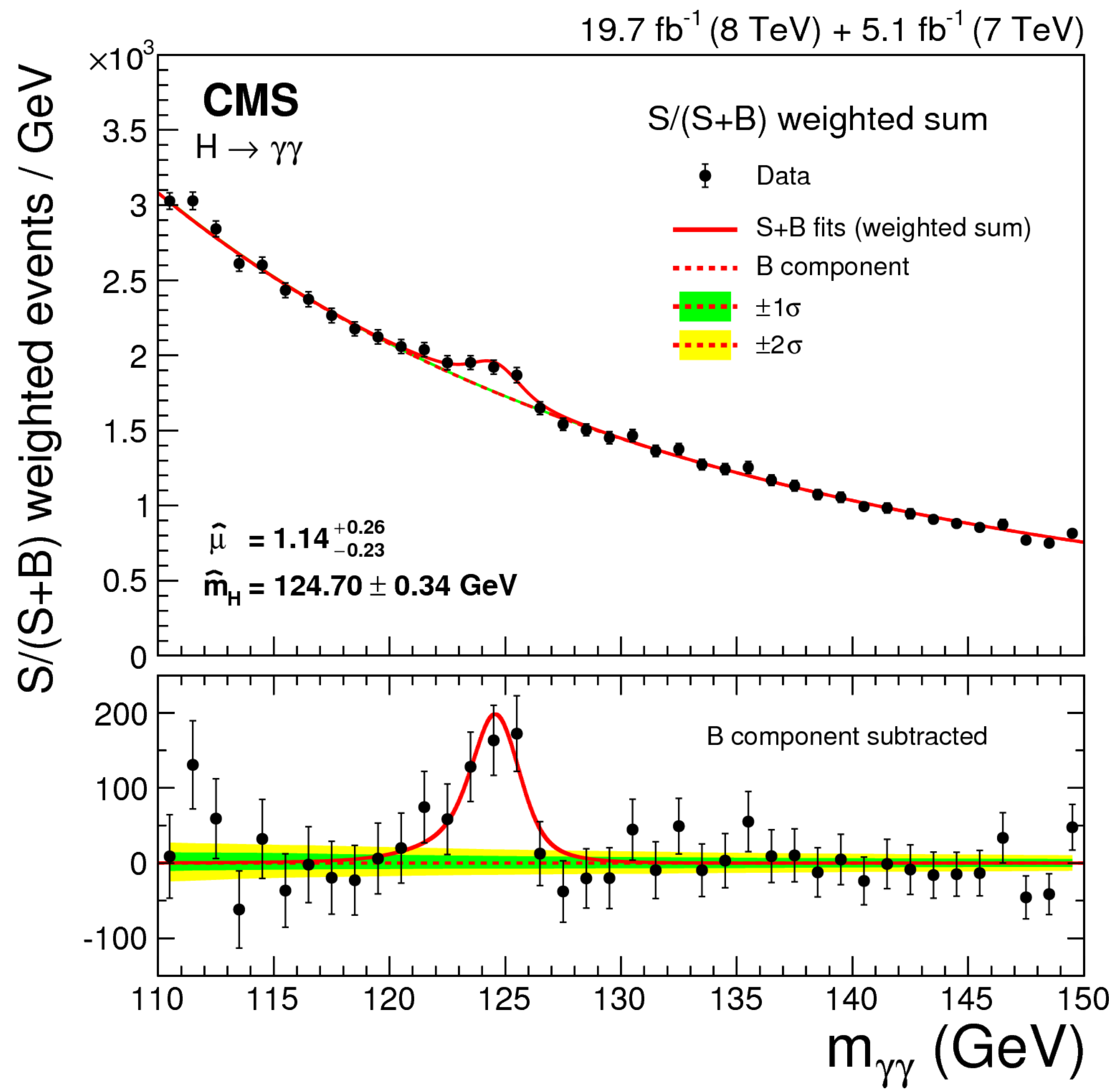


Higgs→$\gamma\gamma$

- Also critical: associating photons to the correct proton collision
  - Goal: correct to 1 cm, with collisions spread over 10 cm
  - Photons don't leave tracks; need to infer correct collision from recoiling tracks
- Diphoton events are classified into four groups based on quality, with highest quality events given the greatest weight in the measurement
- Heavy use of multivariate classifiers such as boosted decision trees that make optimal use of multiple pieces of information

Higgs→γγ

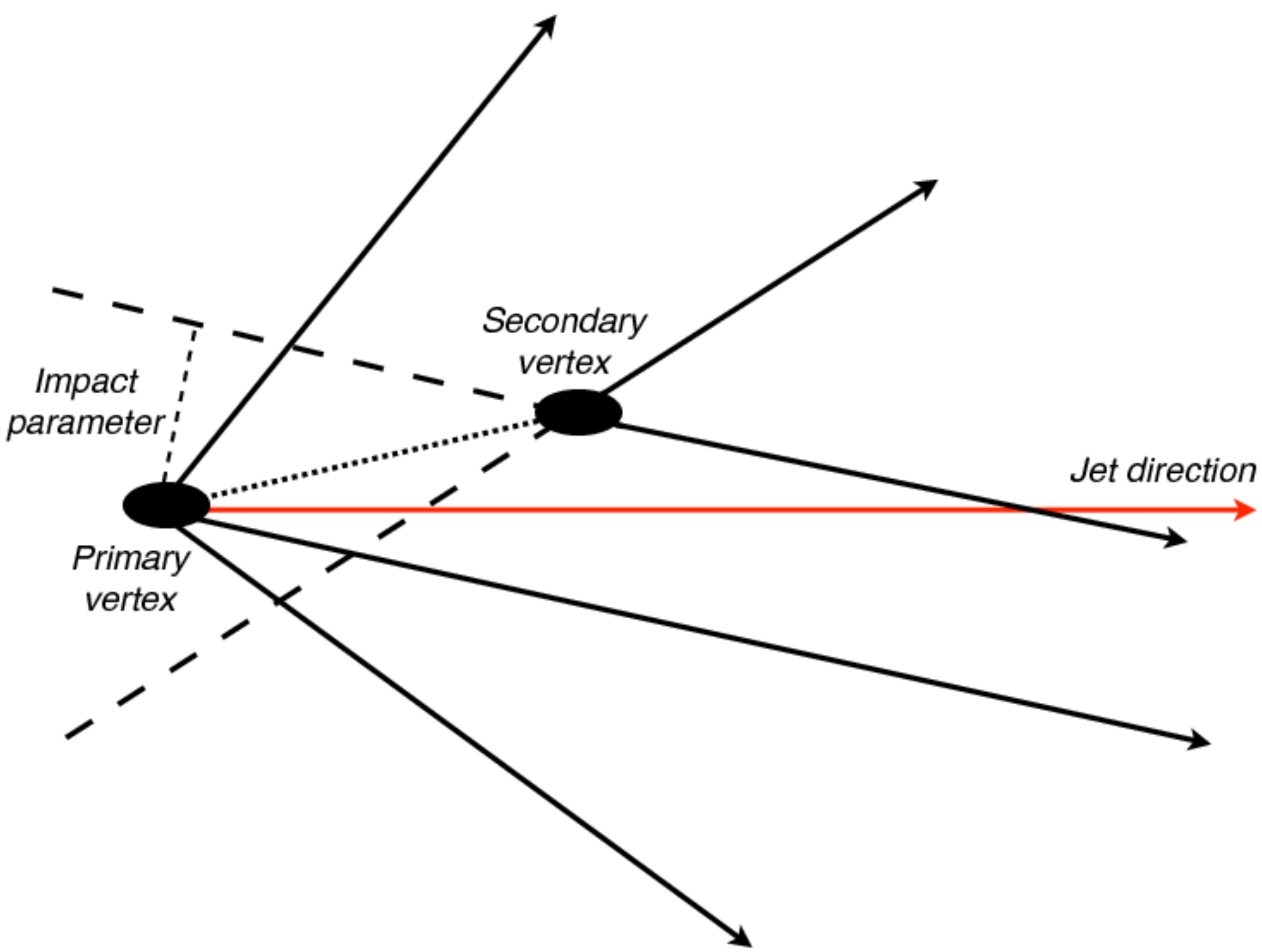
- Assume a fifth-order polynomial for background shape
- Most background events are from real photons, some from misidentified particle jets
- Observe a bump at ~125 GeV with 5.6 SD significance
    - Width of the bump is determined by photon energy resolution



Higgs→bb

- Higgs decays to a pair of bottom quarks 58% of the time, x60 more than to photons
- But:
  - Bottom production rate from other processes much larger than photon production rate
  - Poor resolution on b-pair mass (10%)
  - A bump hunt won't work
- Search for Higgs produced in conjunction with W or Z bosons, which are easy to trigger on and identify

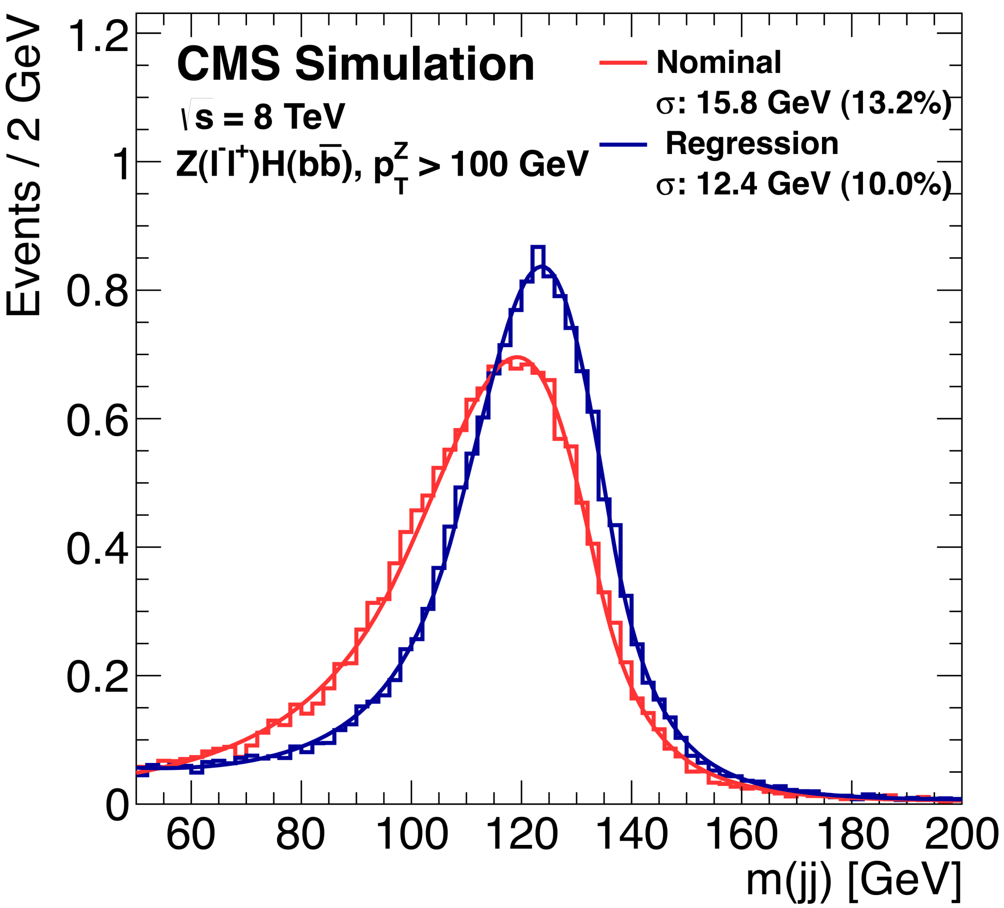
Higgs→bb



- Key software technology is "secondary vertex reconstruction"
    - Particles containing b quarks have relatively long lifetimes, can travel millimeters before decaying
    - Accurate alignment and track reconstruction are needed to separate primary and secondary vertices

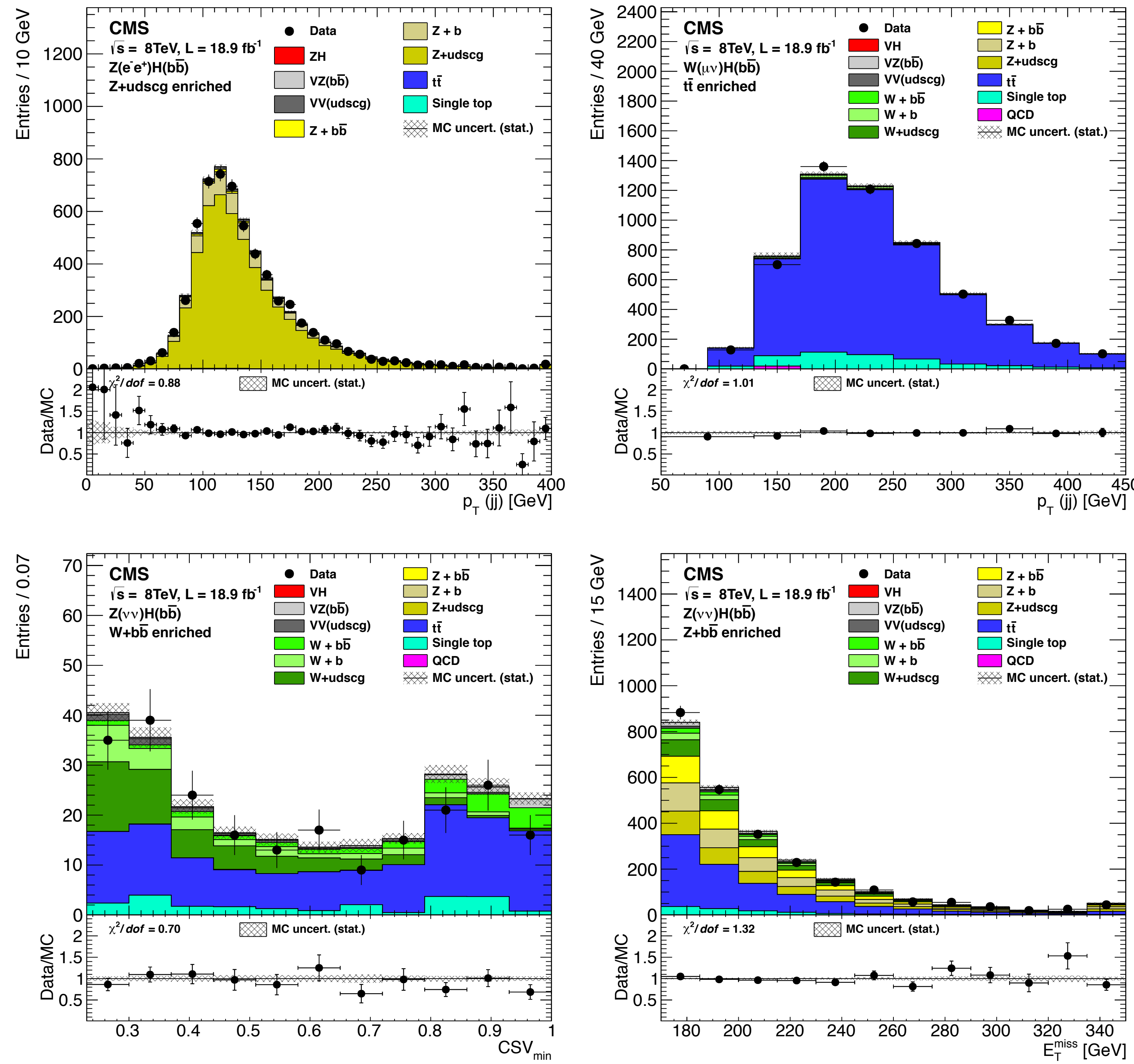
Higgs→bb

- Backgrounds producing W/Z and b jets have rates several orders of magnitude above Higgs production
  - Higgs production enhanced by selecting W/Z with very large transverse momentum
- b jet kinematics are taken into account to apply a simulation-derived correction to the measured energy, yielding 15% resolution improvement



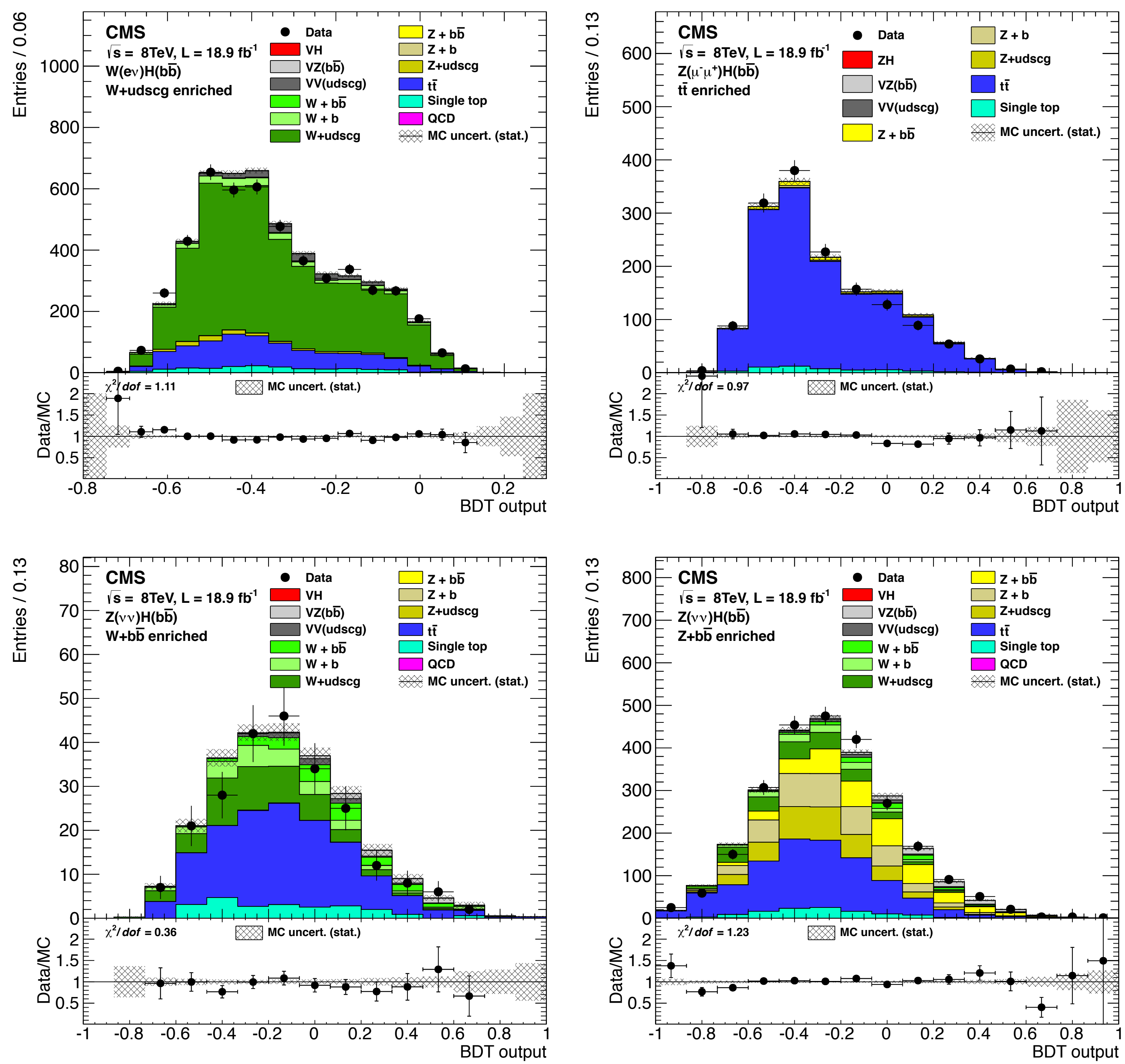
Higgs→bb

- Background-enriched control samples are defined and kinematic quantities are validated there
- These quantities are then combined into single variables used to discriminate signal from background



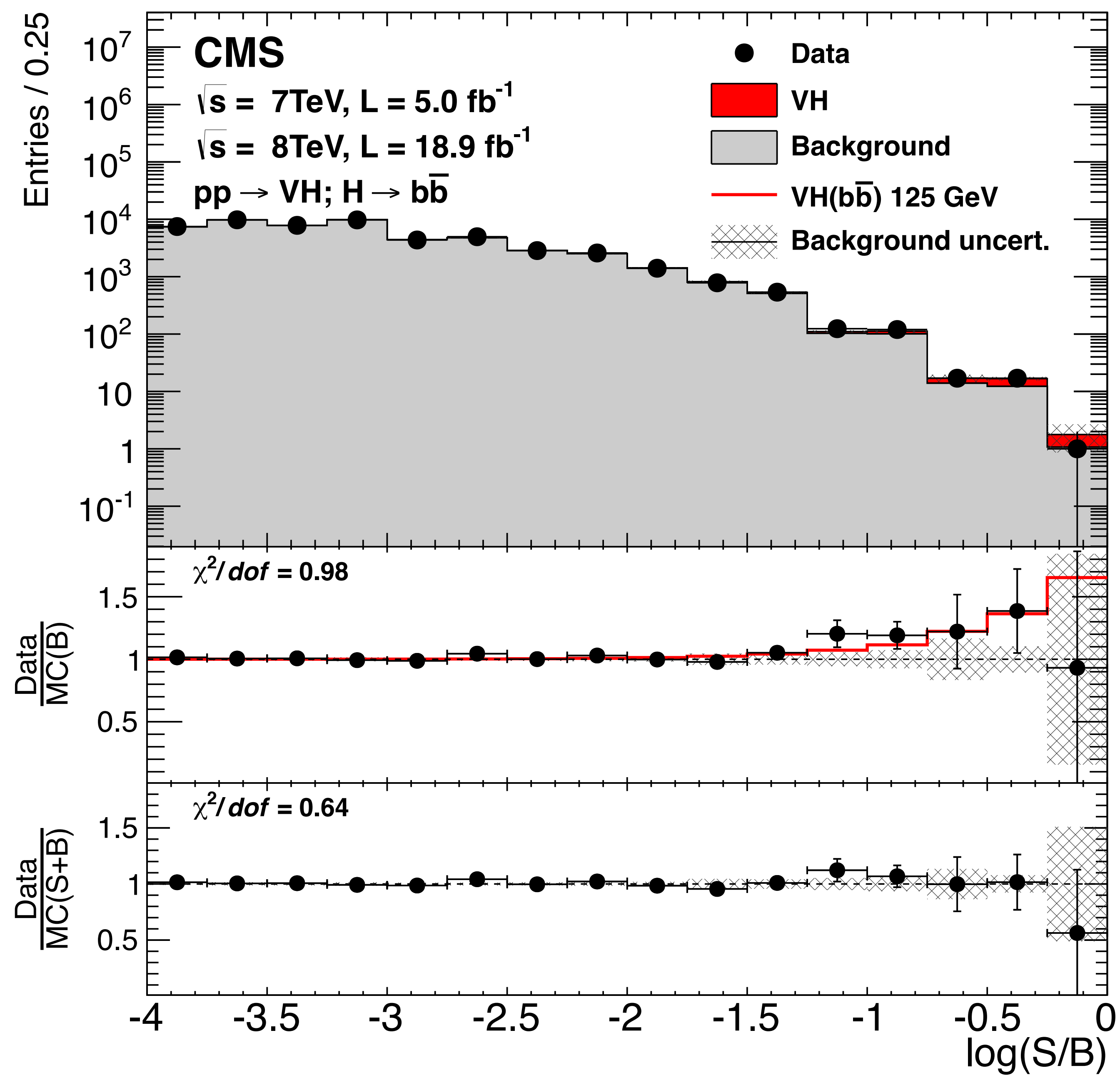
Higgs→bb

- Background-enriched control samples are defined and kinematic quantities are validated there
- These quantities are then combined into single variables used to discriminate signal from background

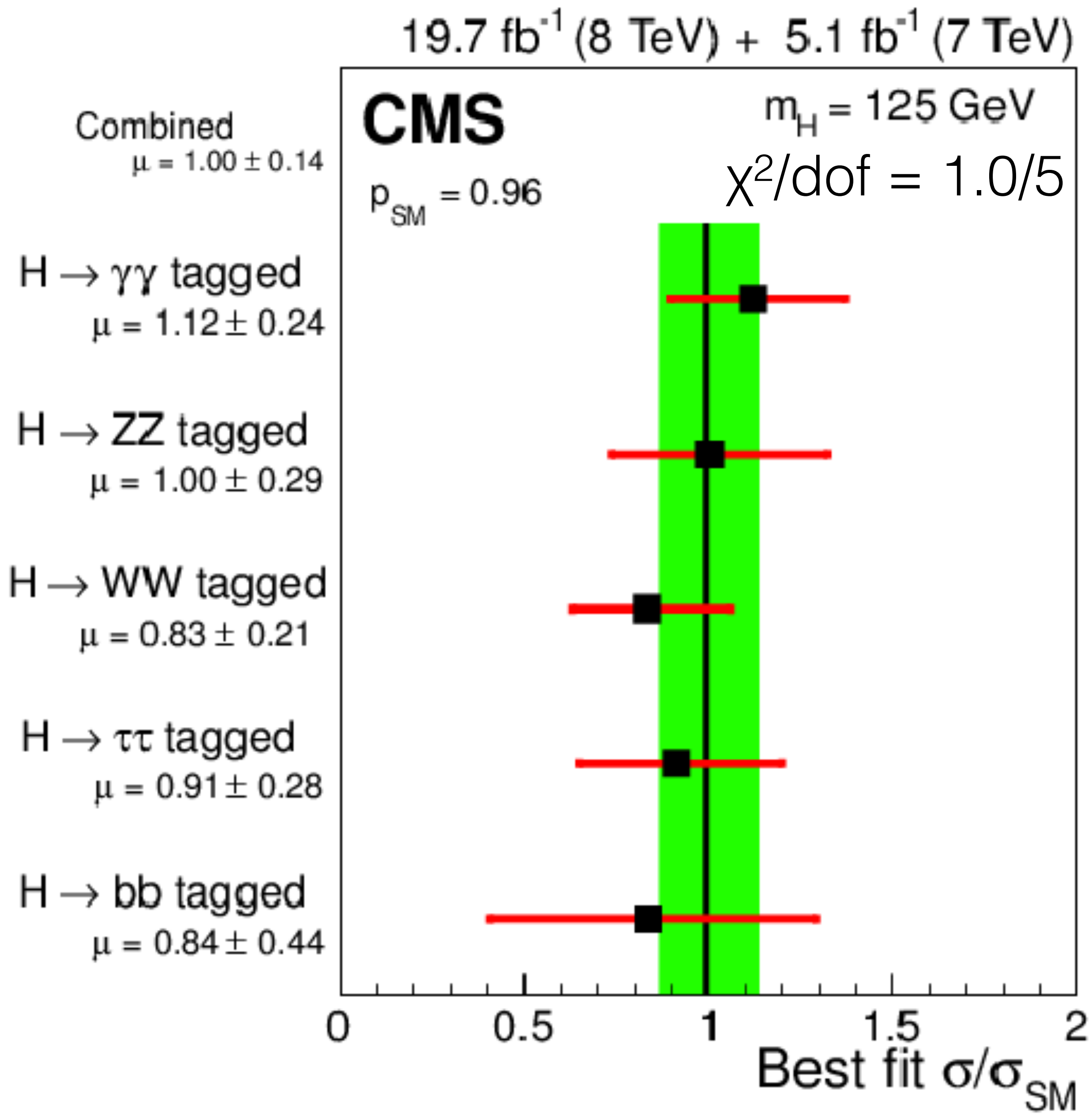


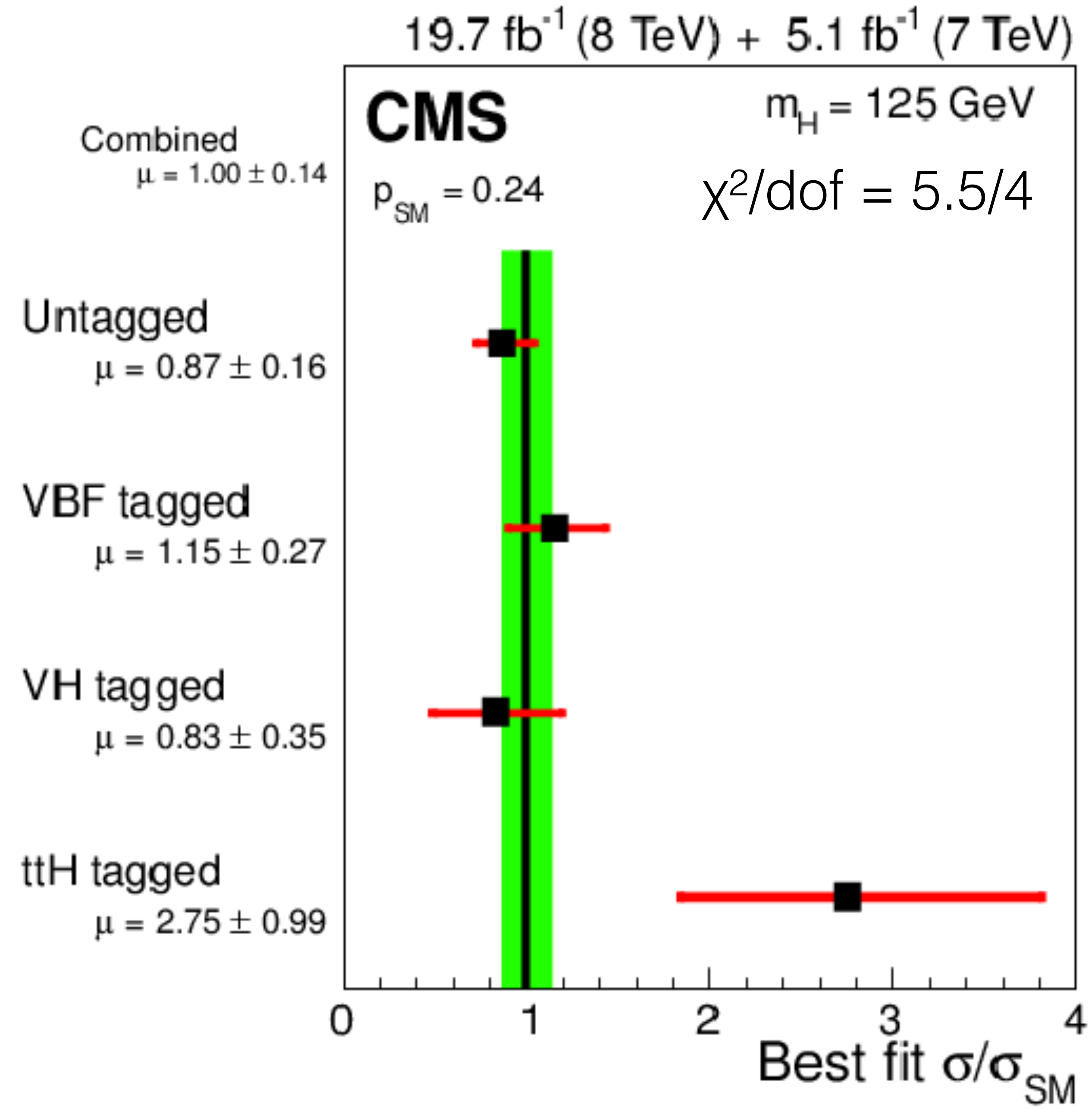
Higgs→bb

- There is a great digestion of all of this information into a single variable
- Higgs signal appears with 2.0 SD significance — not enough to claim any observation



Decay and production rates





$$\sigma/\sigma_{\mathrm{SM}} = 1.00 \pm 0.13 \left[\pm 0.09(\mathrm{stat.})^{+0.08}_{-0.07}(\mathrm{theo.}) \pm 0.07(\mathrm{syst.})\right]$$

Totally consistent with expectations! (so far)

The End

- Particle physics experiments are designed to study rare phenomena that occur in a very noisy environment
- Software and computing tools are necessary to fulfill the promise of the experiments through data processing, simulations and analysis to learn more about our physical world