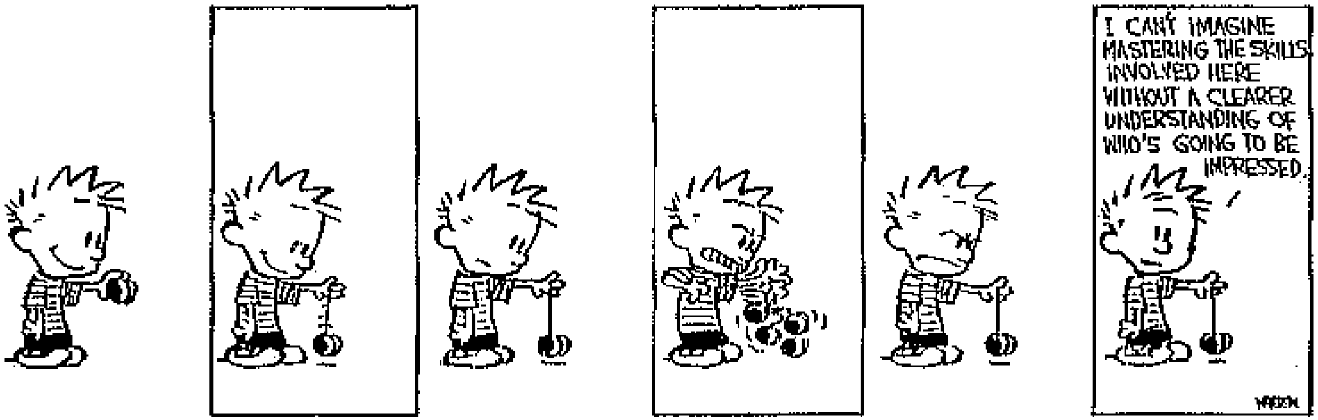
From Raw Data to Physics: Reconstruction and Analysis

Reconstruction: Tracking

Analysis: Measuring a lifetime



Why does tracking need to be done well?

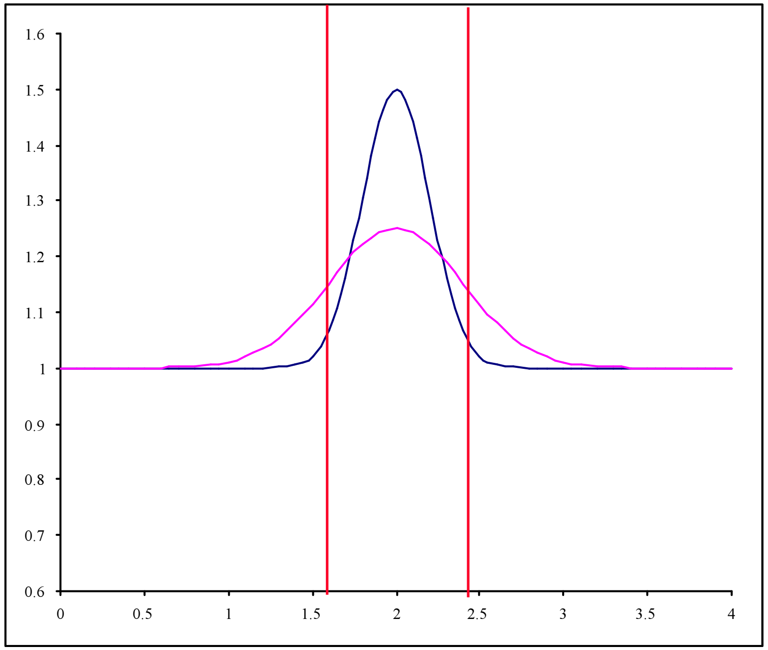
1) Tells you particles were created in an event

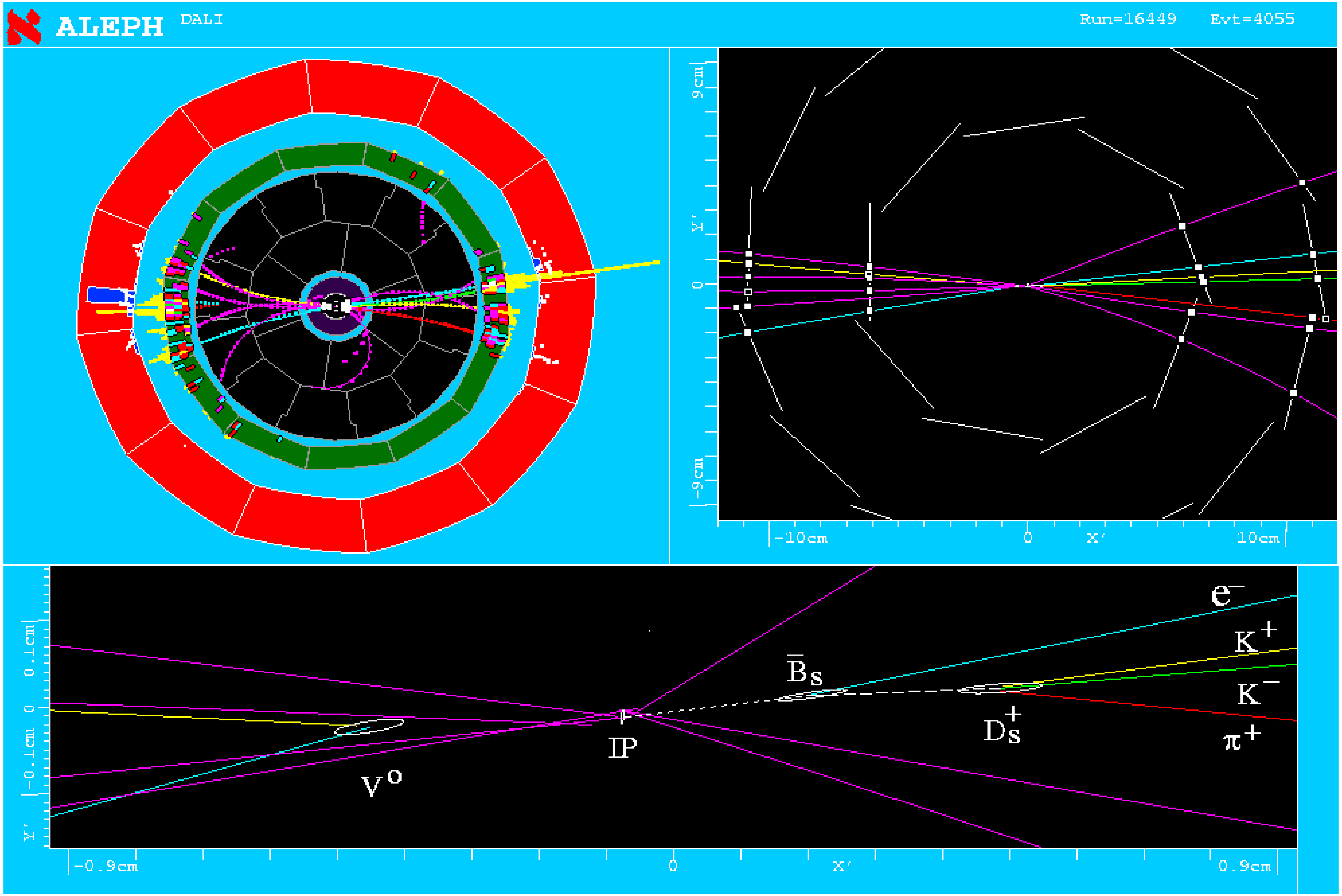
2) Allows you to measure their momentum

- Direction and magnitude
- Combine these to look for decays with known masses
- Only final particles are visible!

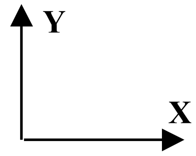
3) Allows you to measure spatial trajectories

- Combine to look for separated vertices, indicating particles with long lifetimes




ALEPH
DALI
Run=16449
Evt=4055
9cm
Y'
0
-9cm
-10cm
0
X'
10cm
e⁻
K⁺
$\bar{B}_s$
K⁻
D_s^+
π^+
IP
V⁰
0.1cm
0
-0.1cm
Y'
-0.9cm
0
X'
0.9cm

Track Fitting



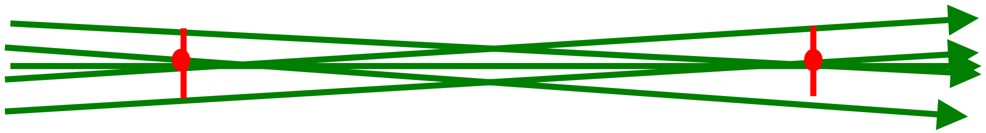
1D straight line as simple case

Two perfect measurements

- Away from interaction point
- With no measurement uncertainty
- Just draw a line through them and extrapolate

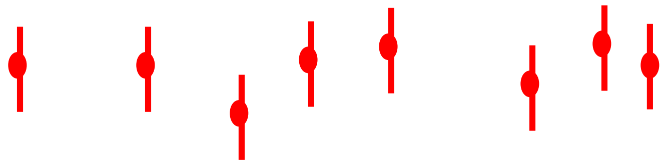
Imperfect measurements give less precise results

- The farther you go, the less you know



Smaller errors, more points help constrain the possibilities

How to find the best track from a large set of points?



How to fit quantitatively?

Parameterize track: $y(x) = \theta x + d$

- Two measurements, two parameters => OK

Best track?

- Consistency with measurements represented by

  Sum of normalized errors squared

$$\chi^2 = \sum_{i=1}^{n_{hits}} \frac{\left(y_i - y(x_i)\right)^2}{\sigma_i^2}$$

Position of i[th] hit — y_i

Predicted track position at i[th] hit — $y(x_i)$

Accuracy of measurement — σ_i

- This is directly a function of our parameters:

$$\chi^2 = \sum_{i=1}^{n_{hits}} \frac{\left(y_i - \theta x_i - d\right)^2}{\sigma_i^2}$$

- The best track has the smallest normalized error
- So minimize in the usual way:

$$\frac{\partial \chi^2}{\partial \theta} = 0 \qquad \frac{\partial \chi^2}{\partial d} = 0$$

$$\frac{\partial \chi^2}{\partial \theta} = 2\sum \frac{(y_i - \theta x_i - d)}{\sigma_i^2}(-x_i)$$

$$0 = \left(\sum \frac{y_i x_i}{\sigma_i^2}\right) - \left(\sum \frac{x_i}{\sigma_i^2}\right) d - \left(\sum \frac{x_i^2}{\sigma_i^2}\right) \theta$$

$$\frac{\partial \chi^2}{\partial d} = 2\sum \frac{(y_i - \theta x_i - d)}{\sigma_i^2}(-1)$$

$$0 = \left(\sum \frac{y_i}{\sigma_i^2}\right) - \left(\sum \frac{1}{\sigma_i^2}\right) d - \left(\sum \frac{x_i}{\sigma_i^2}\right) \theta$$

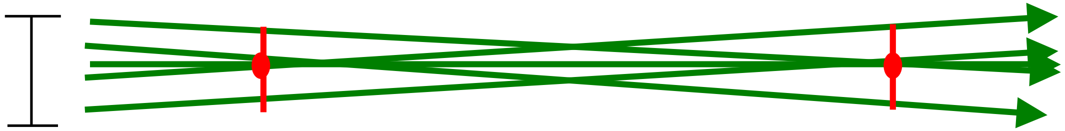
Two equations in two unknowns

- Terms in () are constants calculated from measurement, detector geometry

Generalizes nicely to 3D, helical tracks with 5 parameters

- Five equations in five unknowns

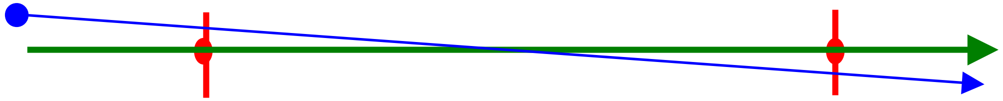
With a little more work, can calculate expected errors on θ, d



"Most likely" that <u>real</u> d (Y intercept) is within this band of $\pm\sigma_d$

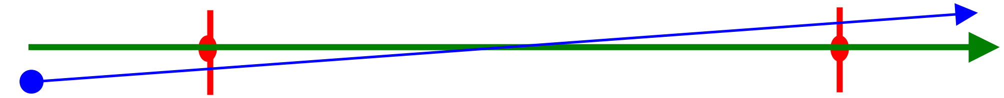
Similar θ error, where θ_{real} is most likely within $\pm\sigma_\theta$ of best value

Note that the errors are <u>correlated</u>:



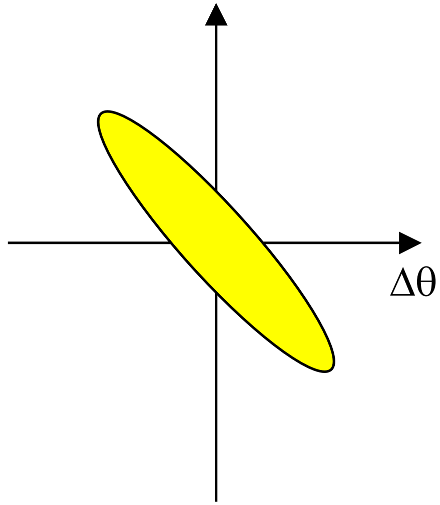
Δd = "+" - 0 > 0

$\Delta\theta$ = "-" - 0 < 0

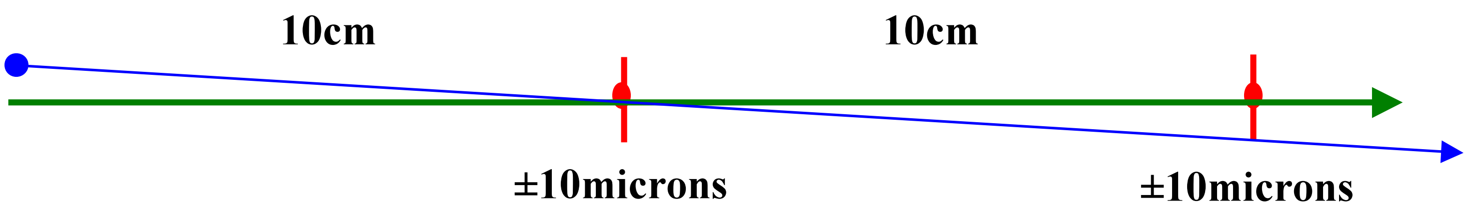


Δd = "-" - 0 < 0

$\Delta\theta$ = "+" - 0 > 0



Typical size of errors



Error on position is about ±10 microns

By similar triangles

Error on angle is about ±0.1 milliradians (±0.002 degrees)

Satisfyingly small errors!

Allows separation of tracks that come from different particle decays

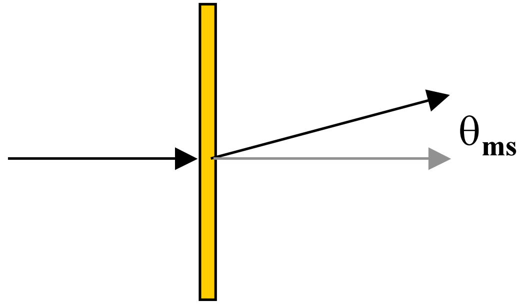
But how to we "see" particles?

- Charged particles pass through matter,
- ionize some atoms, leaving energy
- which we can sense electronically.

More ionization => more signal => more precision

=> more energy loss

Multiple Scattering

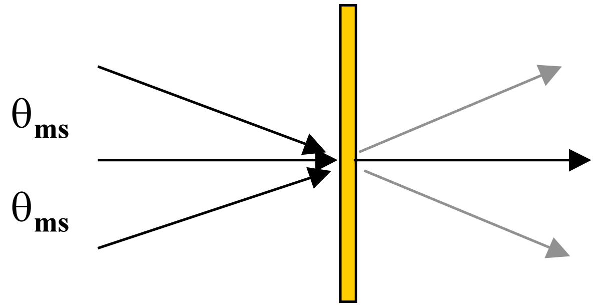


Charged particles passing through matter "scatter" by a random angle

$$\sqrt{\langle \theta_{ms}^2 \rangle} = \frac{15\, MeV/c}{\beta p} \sqrt{\frac{\text{thickness}}{X_{rad}}}$$

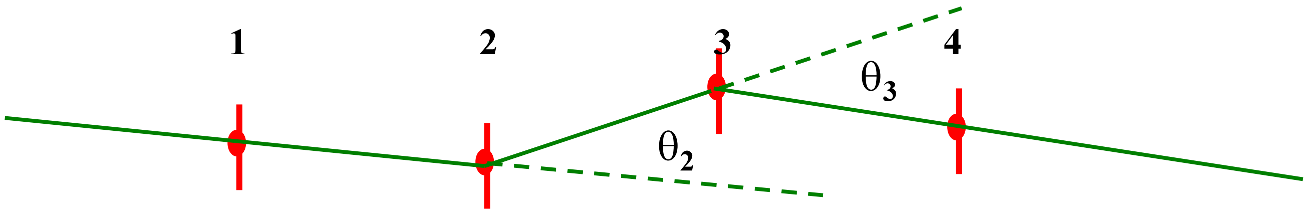
300μ Si   RMS = 0.9 milliradians / βp

1mm Be RMS = 0.8 milliradians / βp



Also leads to position errors

So?



Fitting points 3 & 4 no longer measures angle at IP

Track already scattered by random angles θ_1, θ_2,θ_3

Track has more parameters

1 if $x-x_3 > 0$, otherwise 0

$$y(x) = d + \theta x + \theta_1 (x - x_1)\Theta(x - x_1)$$
$$+\theta_2 (x - x_2)\Theta(x - x_2) + \theta_3 (x - x_3)\Theta(x - x_3) + \ldots$$

If we knew θ_1, $\theta_{2,\ldots}$ we'd know entire trajectory

Can we measure those angles?

θ_2 roughly given by y_1, y_2, y_3

Just a more complex χ^2 equation?

$\sqrt{\langle \theta_{ms}^2 \rangle}$ **acts like a measurement**

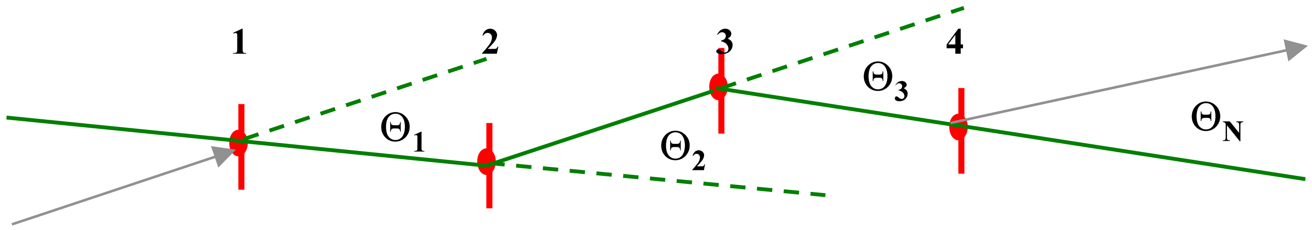
"I'd be surprised if it was larger than $0 \pm \frac{15 MeV/c}{\beta p} \sqrt{\frac{L}{X_{rad}}}$

"Add information" to fit by adding new terms to χ^2

$$\chi^2 = \chi^2_{old} + \sum_i \frac{\theta_i^2}{\sigma_{ms}^2}$$

N measurements from planes (say 100)

N+2 unknowns (d, θ, plus N scattering angles)



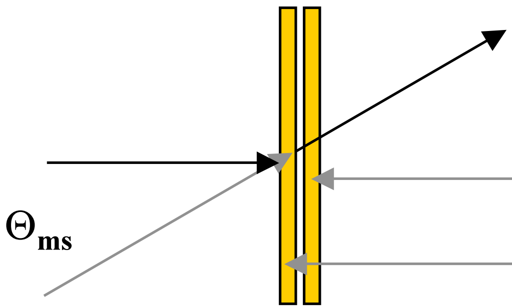
Can't see first, last scattering angles; can only extrapolate outside

Hence ignore θ_1, θ_N

Now all we have to do is solve 100 equations in 100 unknowns...

Nobody cares about θ_N

But θ_1 effects accuracy of d



θ_{ms} => 1.2 milliradian/βp error on θ

@10 cm, leads to 120μ/βp error on d

$$\sigma_d \approx 10\mu \oplus \frac{120\mu}{\beta p}$$

In spite of

N=100 chambers,

complicated programs

and inverting 100x100 matrices

Some problems, the programs can't fix!

"Kalman fit"?

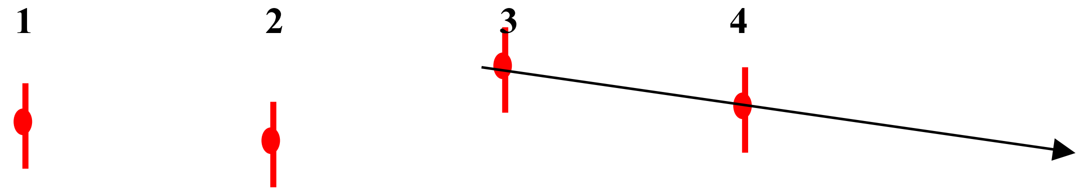
(ref: Brillion)

Computational expensive to calculate solutions with 100 angles

Computer time grows like O(N^3), with N large

And we're not really interested in all those angles anyway

Instead, approximate, working inward N times:



“Kalman fit”?

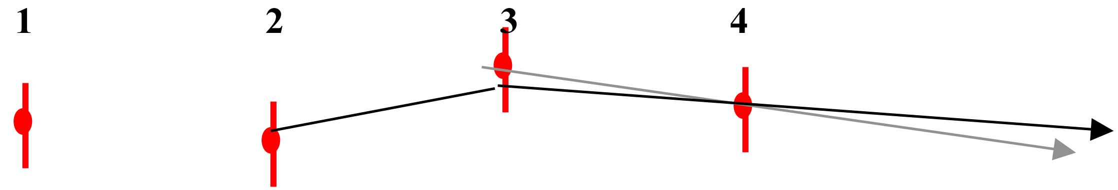
(ref: Brillion)

Computational expensive to calculate solutions with 100 angles

Computer time grows like O(N^3), with N large

And we’re not really interested in all those angles anyway

Instead, approximate, working inward N times:



“Kalman fit”?

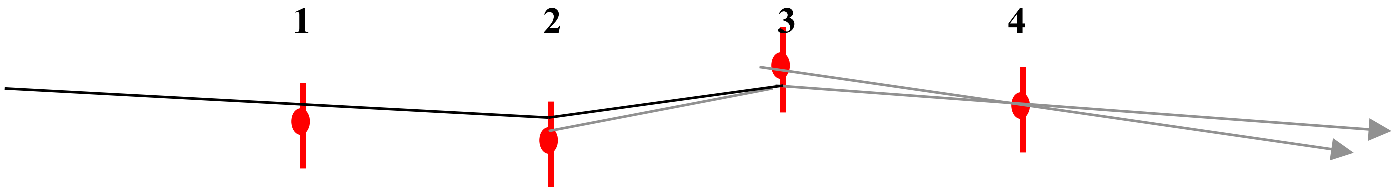
(ref: Brillion)

Computational expensive to calculate solutions with 100 angles

Computer time grows like $O(N^3)$, with N large

And we’re not really interested in all those angles anyway

Instead, approximate, working inward N times:



This is O(N) computations

May need to repeat once or twice to use good starting estimate

Each one a little more complex

But still results in a large net savings of CPU time

Moral: Consider what you <u>really</u> want to know

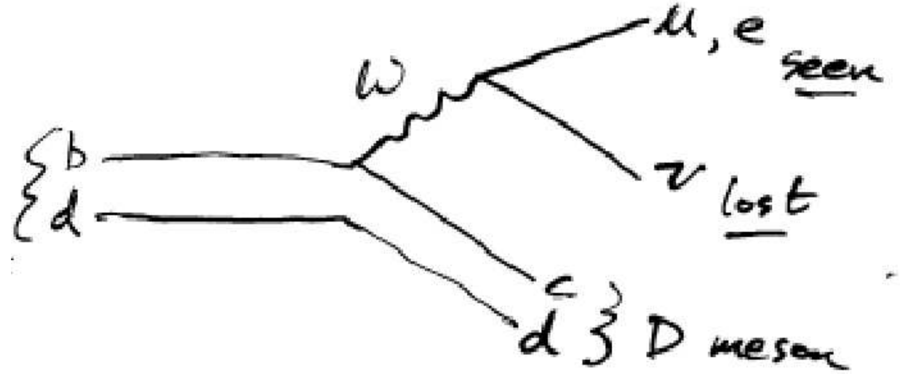
Analysis: Lifetime measurement

Why bother?

Standard model contains 18 parameters, a priori unknown

Particle lifetimes can be written in terms of those

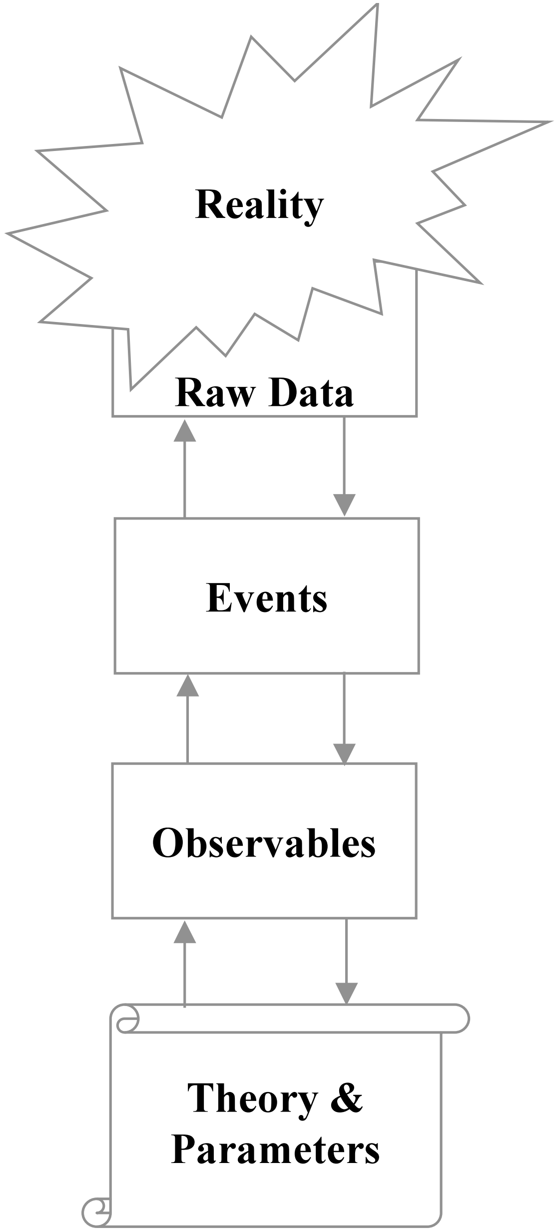
$$\Gamma_Q^{sl} \equiv \Gamma(Q \to q l \upsilon) = \frac{G_F^2}{192\pi^3} m_Q^5 f \left| V_{Qq} \right|^2$$



"Measure once to determine a parameter

Measure in another form to check the theory"

Measure lots of processes to check overall consistency



A model of how physics is done.

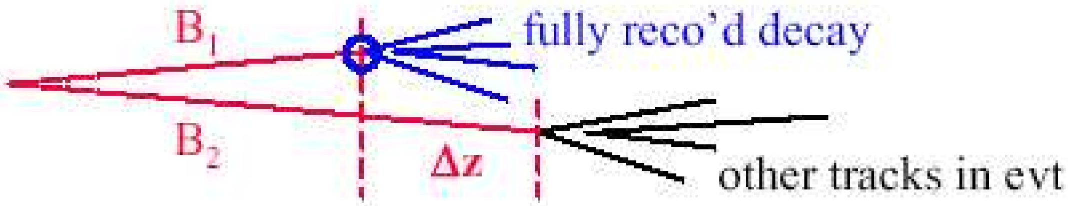
The imperfect measurement of a (set of) interactions in the detector

A unique happening: Run 21007, event 3916 which contains a J/psi -> ee decay

Specific lifetimes, probabilities, masses, branching ratios, interactions, etc

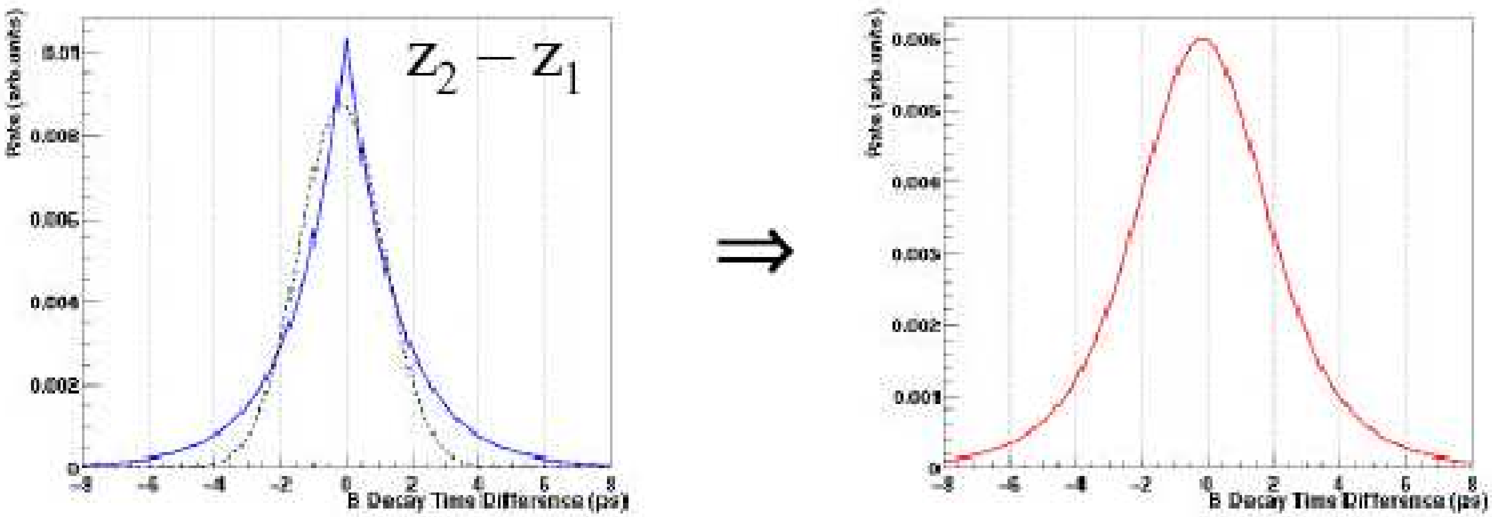
A small number of general equations, with specific input parameters (perhaps poorly known)

B lifetime: What we measure at BaBar:



$$dN \propto \exp(-|\Delta z| / \beta_z \gamma c\, \tau_B)$$

Unfortunately, we can't measure Δz perfectly:



First, you have to find the B vertex

To reconstruct a B, you need to look for a specific decay mode

(Un)fortunately, there are lots!

B0-> D*+ pi-
D*+ rho-
D*+ a1-
D+ pi-
D+ rho-
D+ a1-
J/Psi K*0bar

Each involves additional long-lived particles, which have to be searched for:

D*+ -> D0 pi+
D*0 -> D0 pi0

D0 -> K- pi+, K- pi+ pi0,
K- pi+ pi- pi+, K0S pi+ pi-

D+ -> K- pi+ pi+, K0S pi+

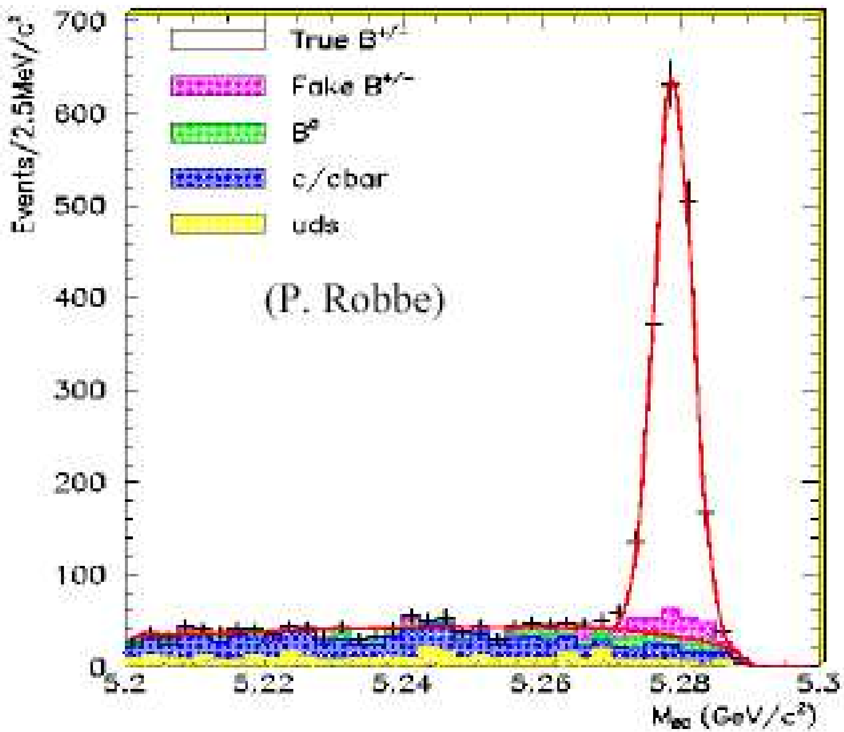
K0S -> pi+ pi-

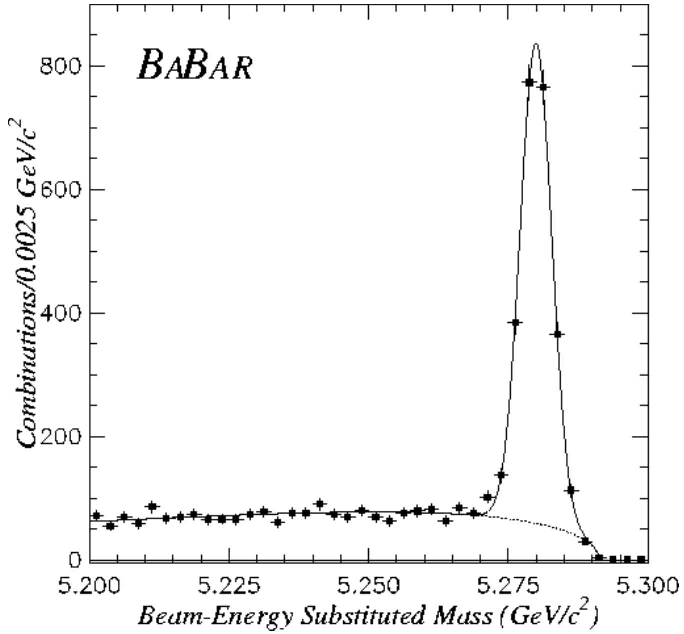
a1- -> rho0(-> pi+ pi-) pi-
rho- -> pi- pi0
pi0 -> gamma gamma

Psi(2S) -> J/Psi pi+ pi-, mu+ mu-, e+ e-
J/Psi -> mu+ mu-, e+ e-

K*0bar -> K- pi+,

And some will be wrong:



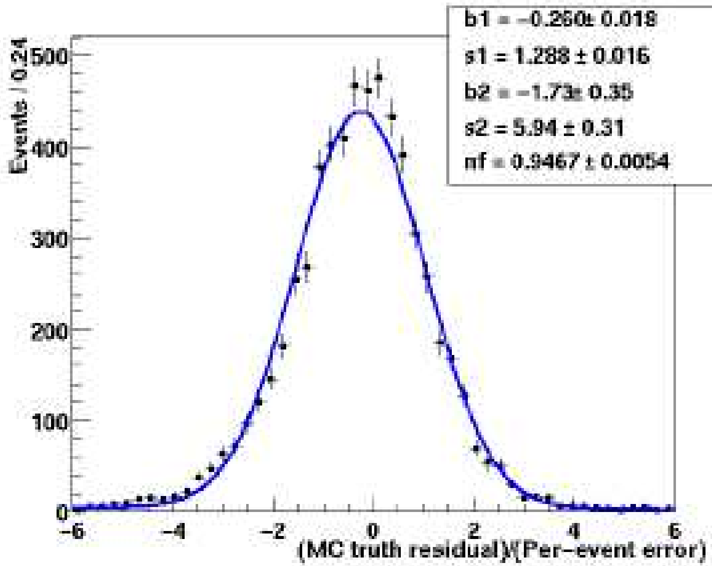


Have to correct for effects of these when calculating the result

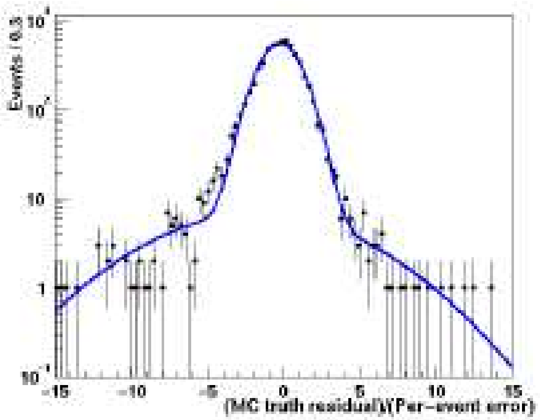
Including a term in systematic error for limited understanding

Next, have to understand the resolution:

Studies of resolution seen in Monte Carlo simulation:



G+G fit to Δt "pulls"
in D*lν signal MC



But how do you know the simulation is right?

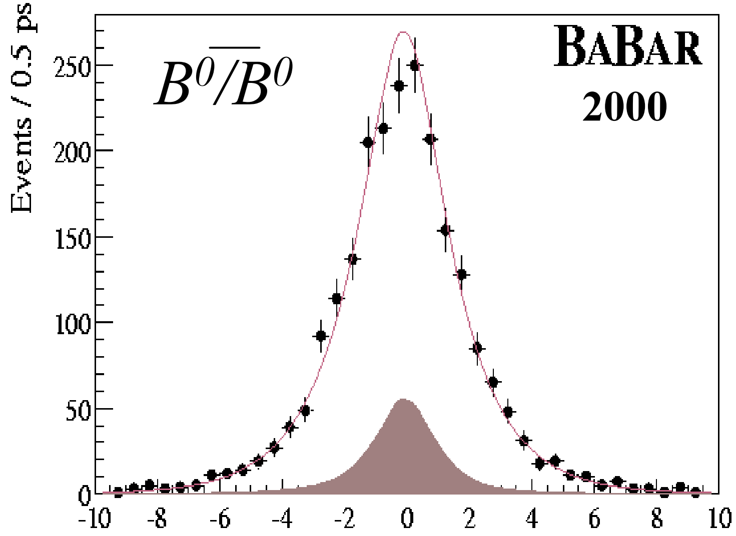
- Find ways to compare data and Monte-Carlo predictions
- Watch for bias in your results!

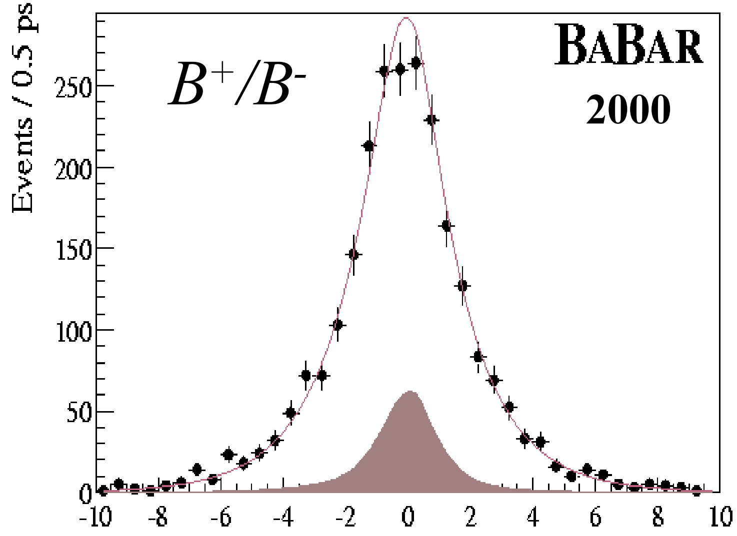
Combined fit to the data gives the lifetime:

You can't extract a lifetime from one event - it's a distribution property

$$N(t) = f(t;\tau) \otimes G(a,b,c,d) + b(t;e,f,g)$$

Try different values until you 'best' fit the data





$\tau_{B0} = 1.506 \pm 0.052$ (stat) $\pm$ 0.029 (syst) ps [PDG= 1.548 ± 0.032]

$\tau_{B+} = 1.602 \pm 0.049$ (stat) $\pm$ 0.035 (syst) ps [PDG= 1.653 ± 0.028]

Note that systematic errors are not so much smaller than statistical ones:
2001 data reduces the statistical error; only improved understanding reduces systematic

What about the computing behind this?

BaBar records about 70k B events per day

- Hidden in 7 million events recorded/day
- Take data about 300 days per year

'Prompt processing'

- Want data available in several days
- Reconstruction takes about 3 CPU seconds/evt
- Processed multiple times
  - E.g. new algorithms, constants, etc

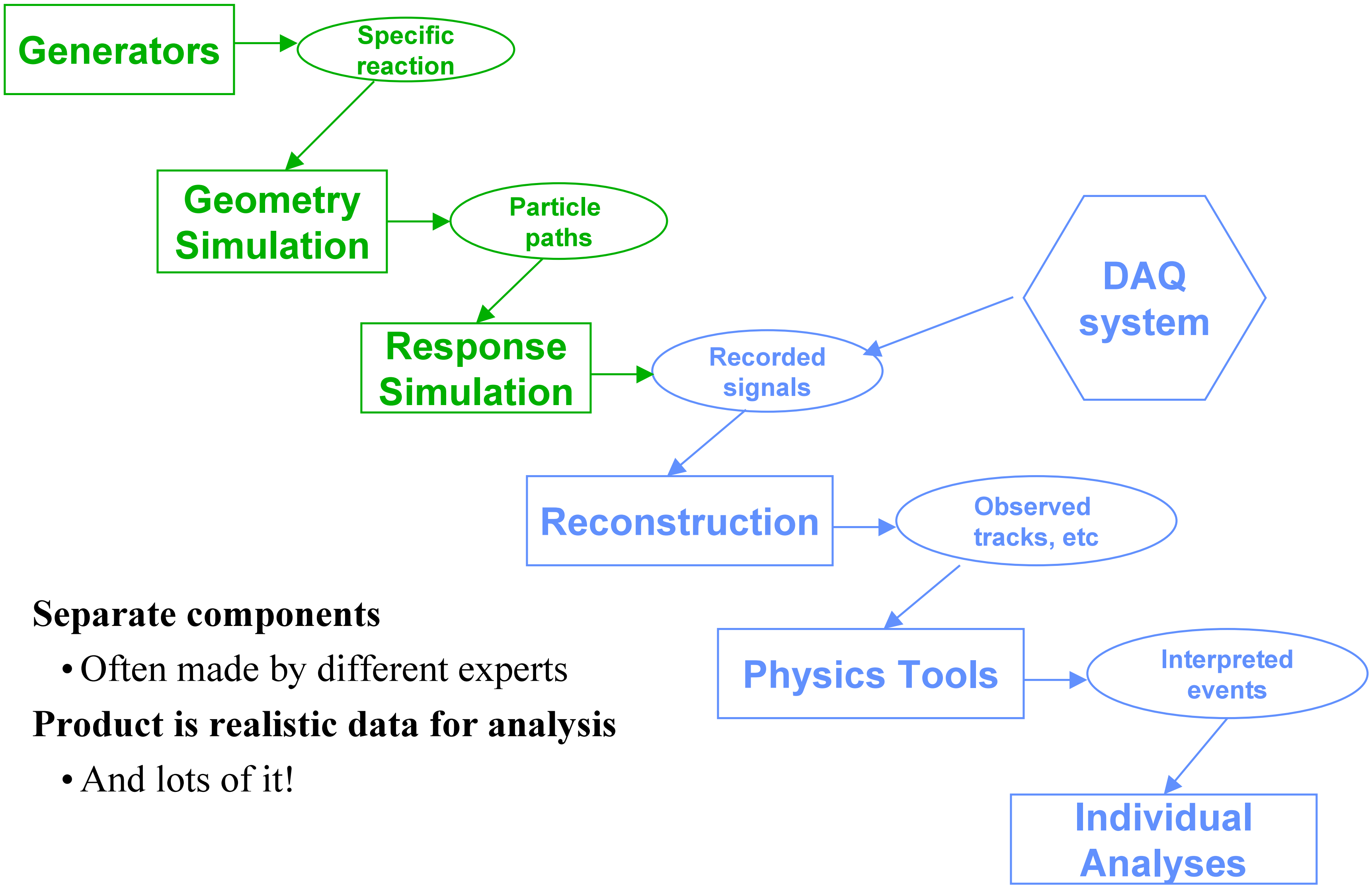
We have about 600 million simulated events to study

- About half in specific decay modes
- Half 'generic' decays to all modes

About 4 million lines of code in simulation and reconstruction programs

- Plus the individual analyses

Traditional flow of data - real and simulated



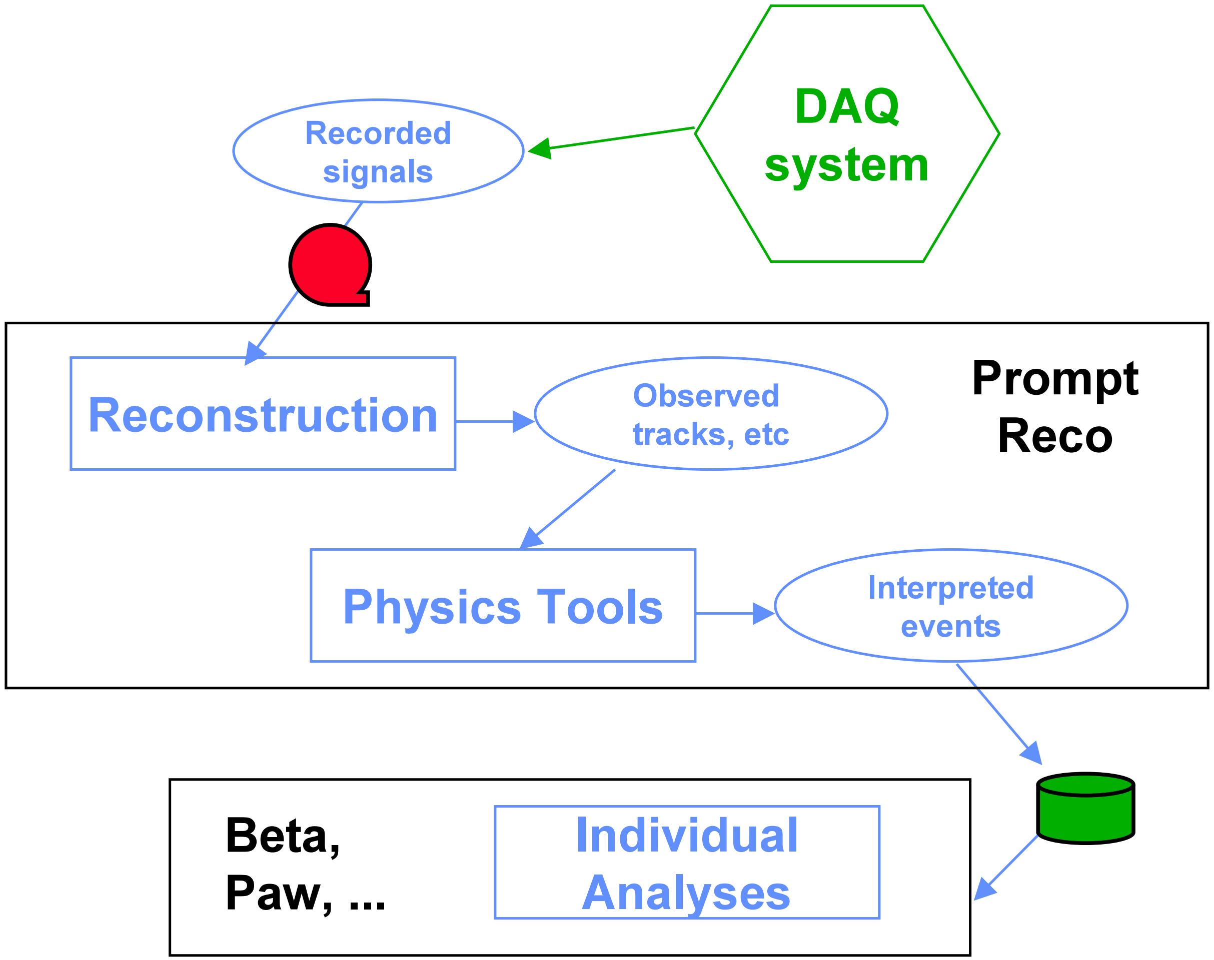
Separate components

- Often made by different experts

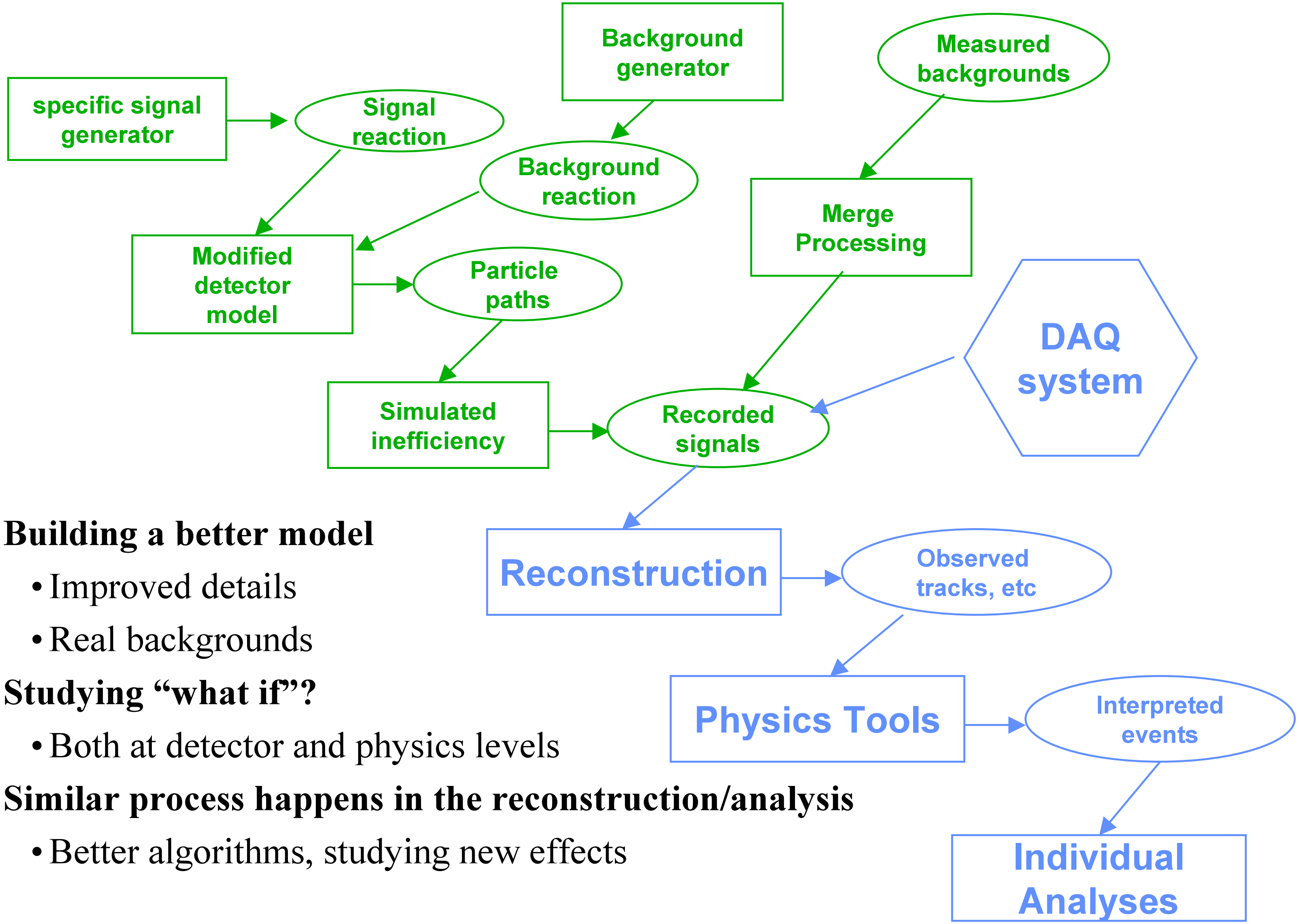
Product is realistic data for analysis

- And lots of it!

Processing real data



More detailed studies via more detailed simulation



Building a better model

- Improved details
- Real backgrounds

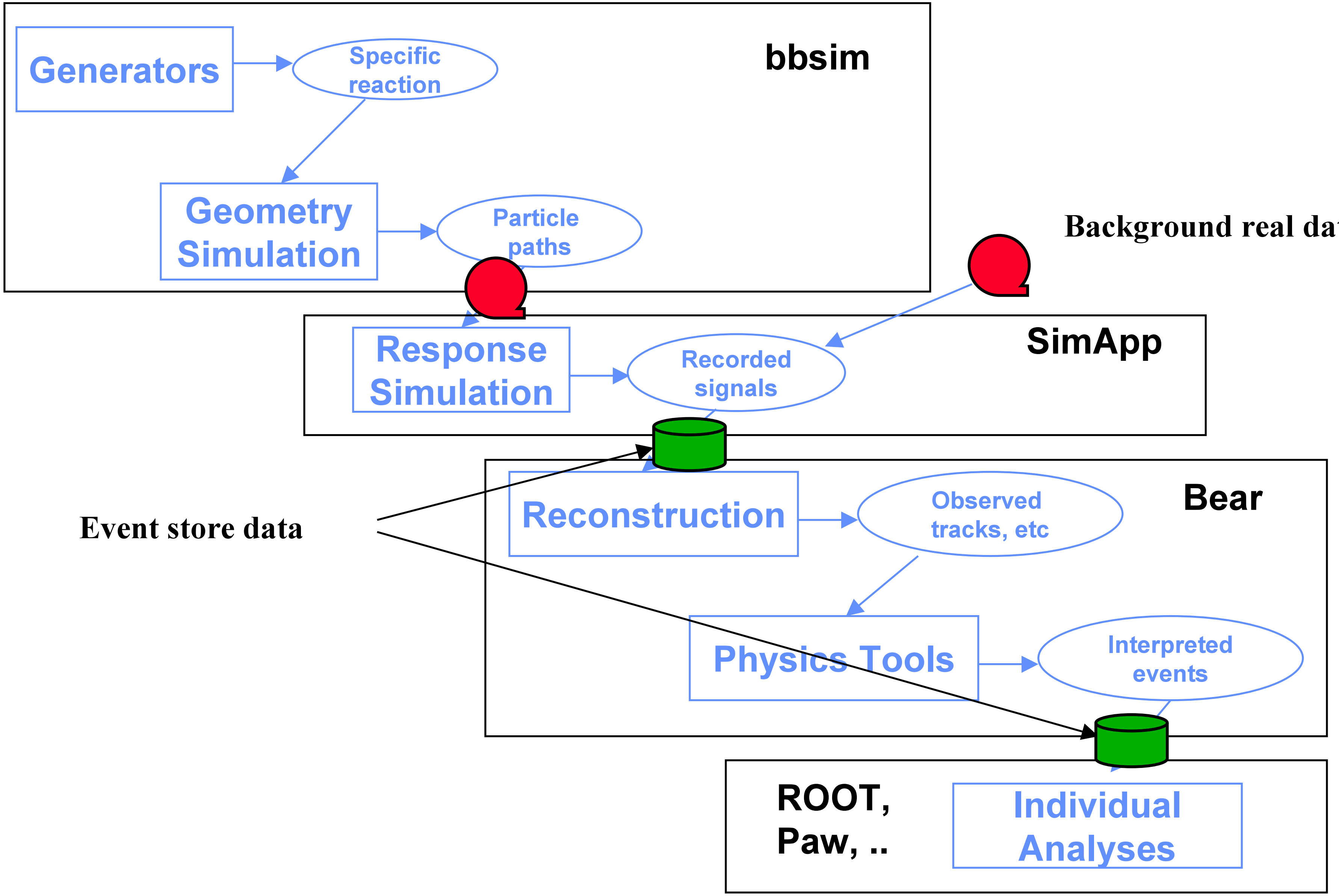
Studying "what if"?

- Both at detector and physics levels

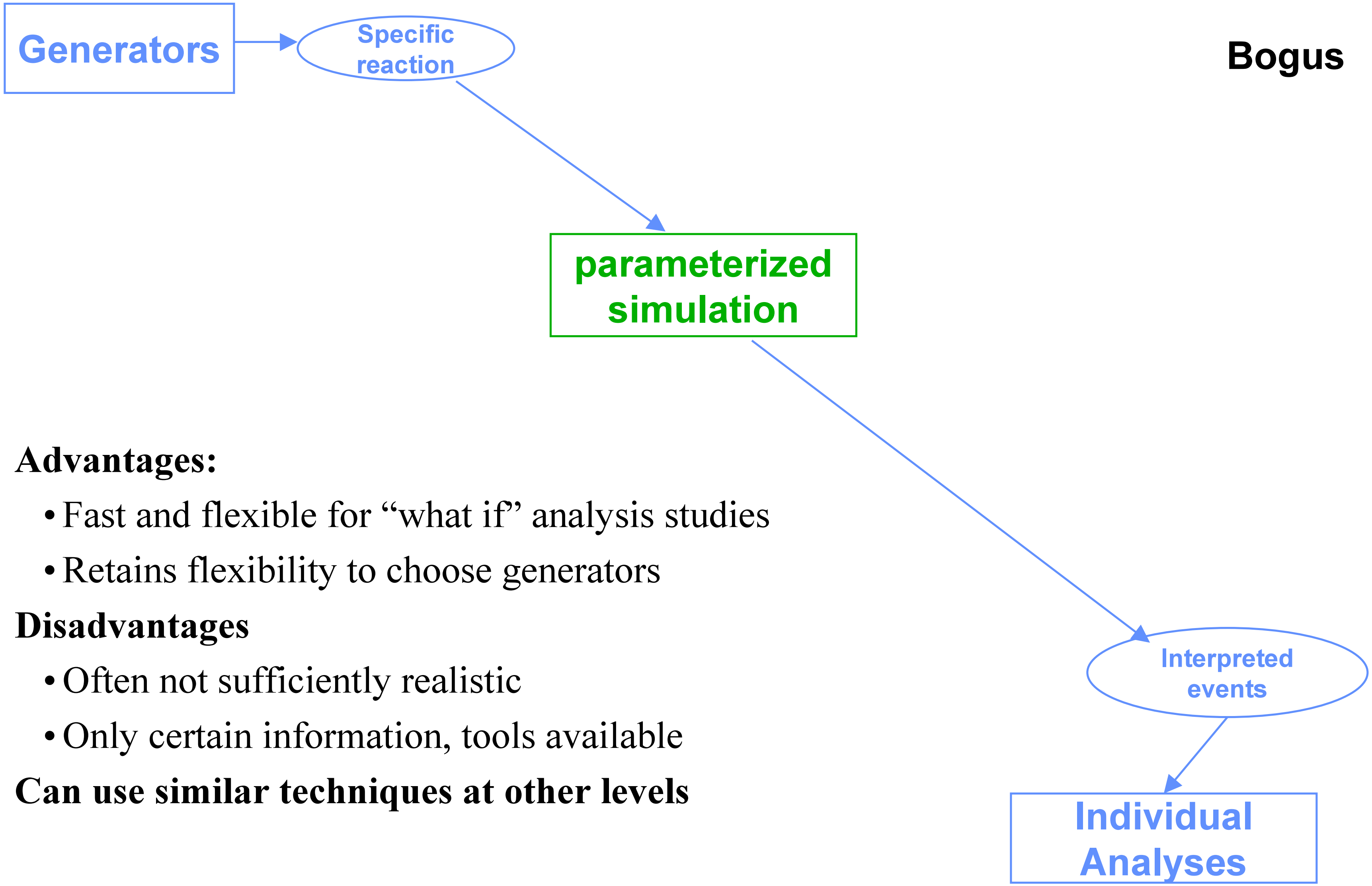
Similar process happens in the reconstruction/analysis

- Better algorithms, studying new effects

Partitioning production system into programs



Speed, simplify simulation by crossing levels



Advantages:

- Fast and flexible for "what if" analysis studies
- Retains flexibility to choose generators

Disadvantages

- Often not sufficiently realistic
- Only certain information, tools available

Can use similar techniques at other levels

Why do we do this?

Detailed simulations are part of HEP physics

- Simulations are present from the beginning of an experiment
  - Simple estimates needed for making detector design choices
- We build them up over time
  - Adding/removing details as we go along
- We use them in many different ways
  - Detector performance studies
  - Providing efficiency, purity values for analysis
  - Looking for unexpected effects, backgrounds

Why do we use such a structure?

- Flexibility - we have different versions of the pieces
  - Comparison forms an important cross check
- Efficiency
  - We build up collections of data at each step for repeated study
  - "I found this background effect in the Spring dataset…"
- Manageability
  - Large programs are hard to build, understand, use

Day 2 summary:

Track fitting as a sample reconstruction problem

- How to make “oh, just draw a line” more quantitative
- How realities of detector, computation effect solution

B lifetime as a sample analysis

- What it tells you
- What you need to know to make the measurement
- The roles of real and simulated data

Offline computing

- Why it’s not trivial
- A typical system organization

Tomorrow:

- How we try to tell particles apart
- What to do when theory isn’t precise
- How to deal with the real detector
- Summary