# Trigger in HEP

Tiehui Ted Liu

Fermilab

Tropical Seminar on Frontier of Particle Physics 2011 –
Detector and Electronics

Aug 18-22, 2011, Beijing

# Outline of lectures

- Trigger in HEP (I): the view from Physics
- Trigger in HEP (II): the view from Instrumentation
- Trigger in HEP (III): the view into the Future

Decided to take a very different approach to the lectures:

cover much less, explain more, with only a few selected topics,
bias towards collider experiments

FEE/Trigger/DAQ go hand-in-hand.
Have to drop a lot interesting topics in my lectures, but Patrick Le Du will cover all the rest. Please see his talks/slides to get a more complete view.

# Lecture II

- Trigger in HEP II: the view from instrumentation
  - A reminder on what we learned in Lecture I
  - Concept of triggering on interesting events
  - Then take a closer look at *tracking trigger* the rest of lecture
  - Case study: L1 tracking trigger
    - Babar L1 Drift Chamber Track Trigger (Zeus track trigger)
    - CDF L1 Drift Chamber Track Trigger (D0 fiber track trigger)
  - Case study:  Silicon Track Trigger
    - CDF L2 Silicon Vertex Trigger (SVT)
    - ATLAS Fast Track Trigger (FTK)
  - Comments on tracking trigger in the future (next lecture)

# Challenges at different cases
# → simplified view

e+e- ⟶ $\Phi$ ⟶ K $\bar{K}$

KLOE/DAΦNE

e+e- ⟶ ψ'' ⟶ D $\bar{D}$

BES/BEPC

e+e- ⟶ ϒ(4S) ⟶ B $\bar{B}$

No pile up, but beam related background

CLEO/CESR, BaBar/PEP-II, Belle/KEK-B

e+e- ⟶ Z

SLD/SLC, (Aleph,Delphi,Opal,L3)/LEP

ep (H1, ZEUS, HERMES, HERA-B) /HERA

pp-bar (CDF,D0)/Tevatron

pp (Atlas, CMS, ALICE, LHCb)/LHC

Pile up at high lumi

# Multi-Level Trigger Systems

- Can't achieve necessary rejection in a single triggering stage
- Reject in steps with successively more complete information

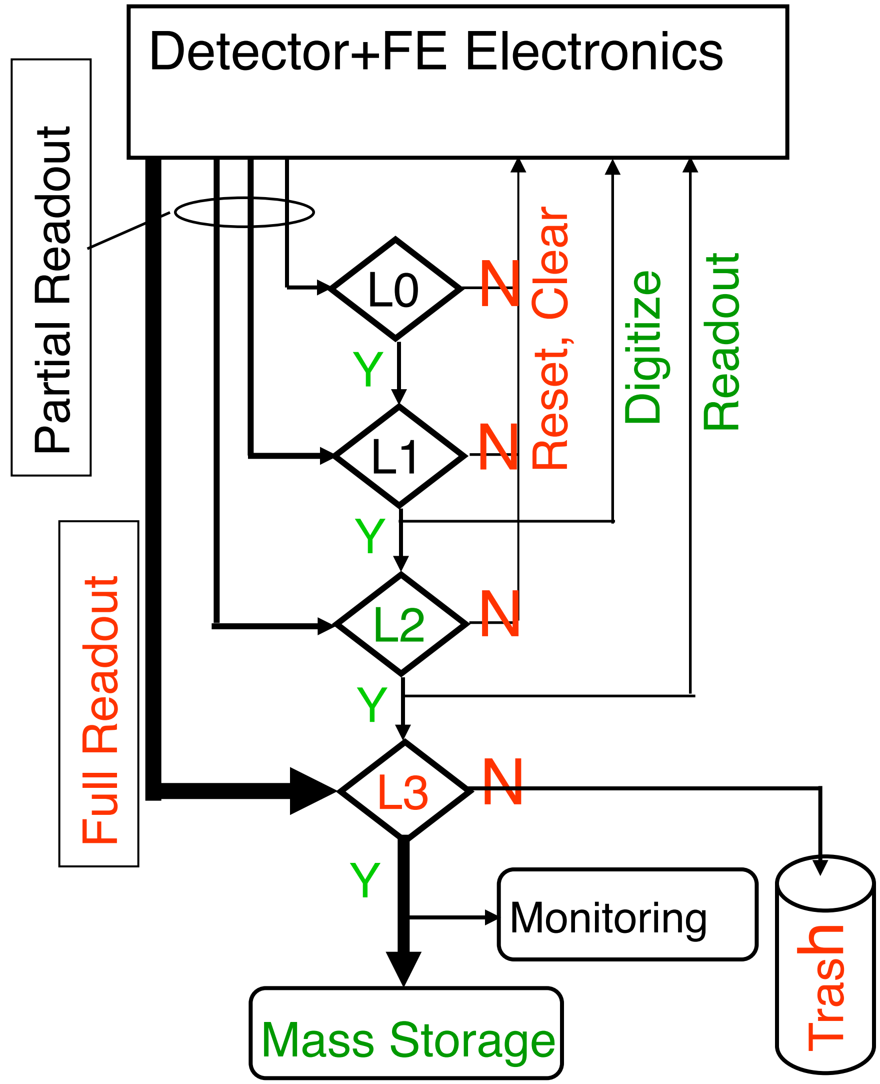
  L0 – very fast & simple

  (often not needed)

  L1 – fast (~few μs) with limited information, hardware

  L2 – moderately fast (~10s of μs), hardware and sometimes software (sometimes not needed)

  L3 – Commercial processor(s)

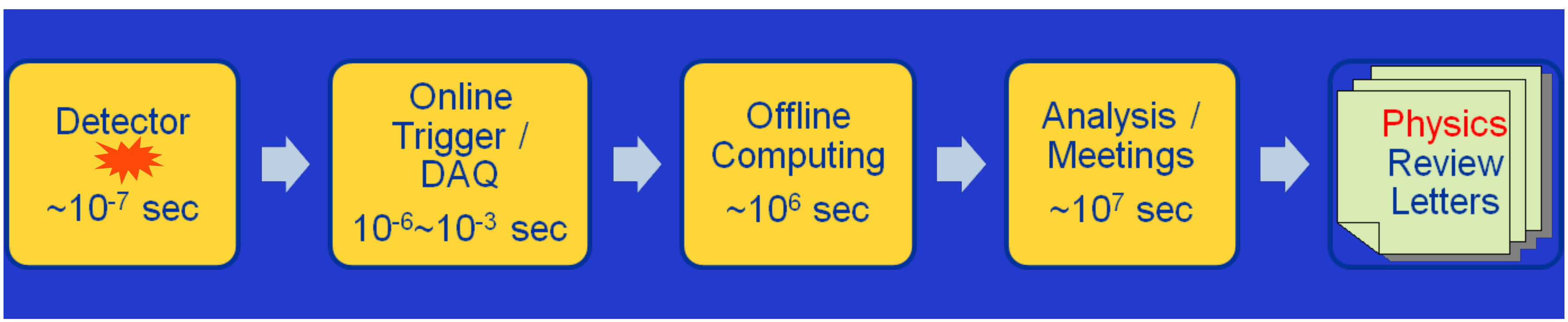


# Luminosity and bandwidth

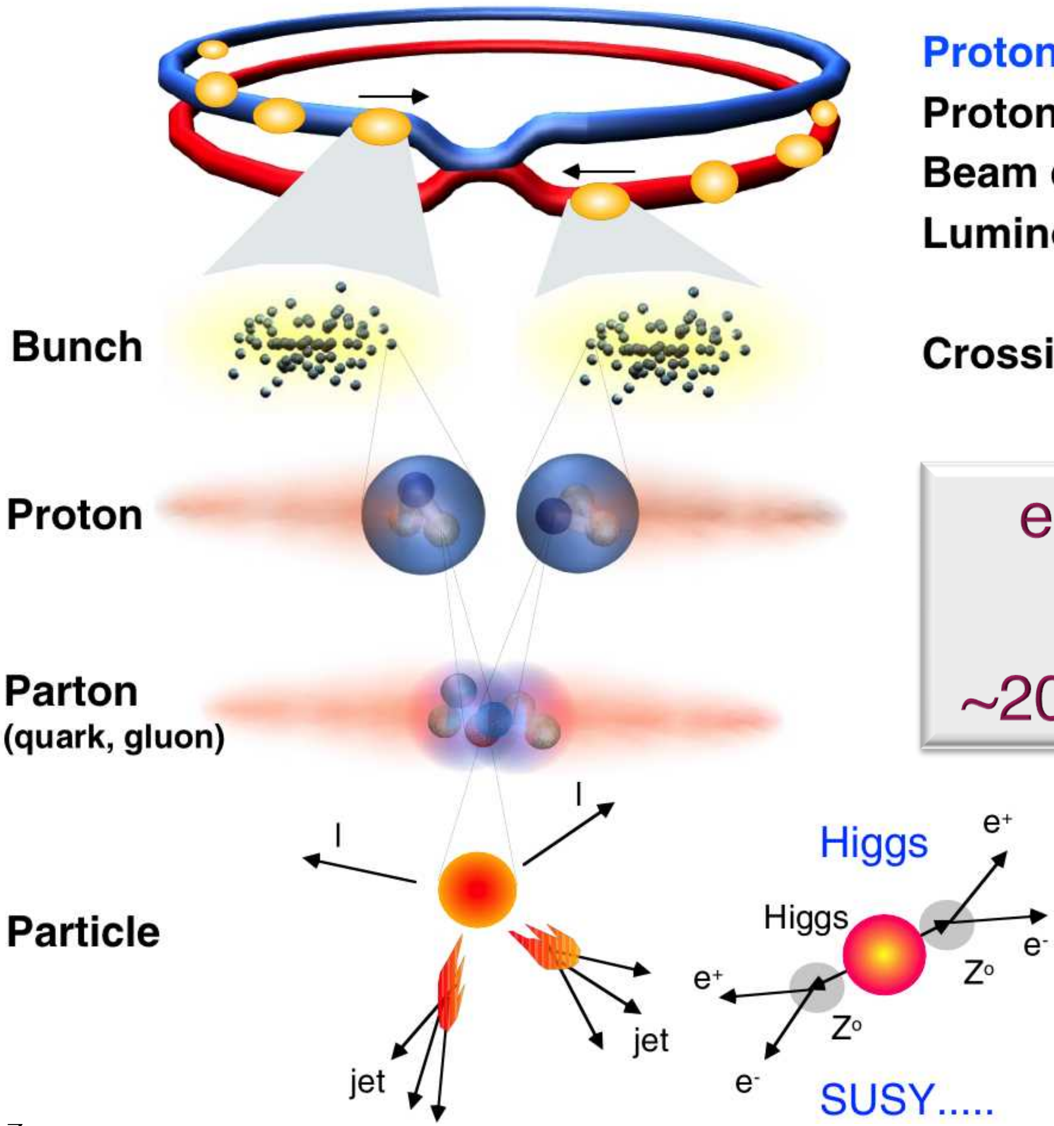
- Not only the luminosity has to be increased, but also the bandwidth …
- from collision point all the way to Physics Review Letters editor's office
- Increase the Trigger/DAQ bandwidth:

*improve latency and purity, and system flexibility, with final physics goals in mind (systematic control etc)*

For each beam crossing, you only trigger once



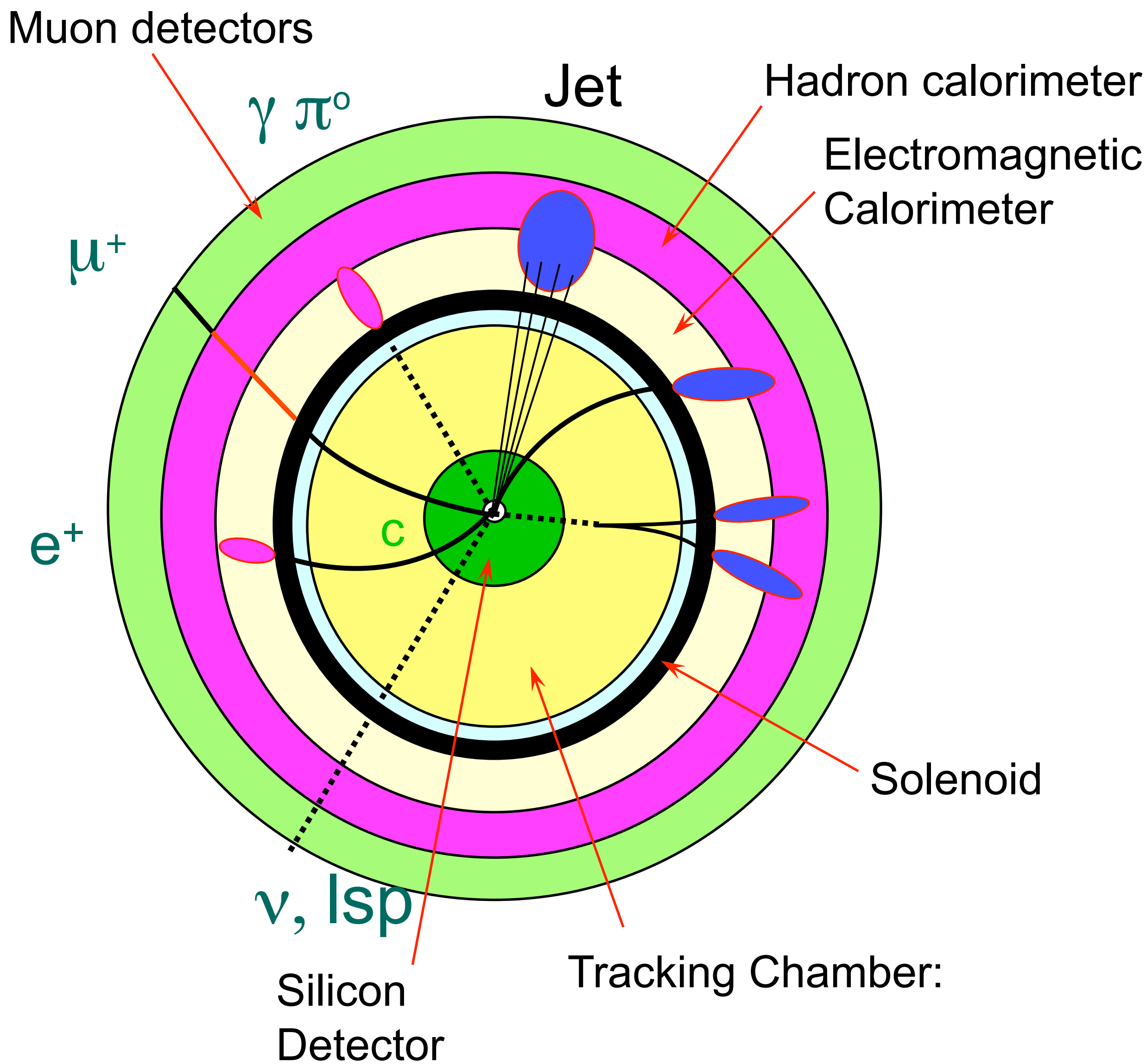
# LHC Collisions



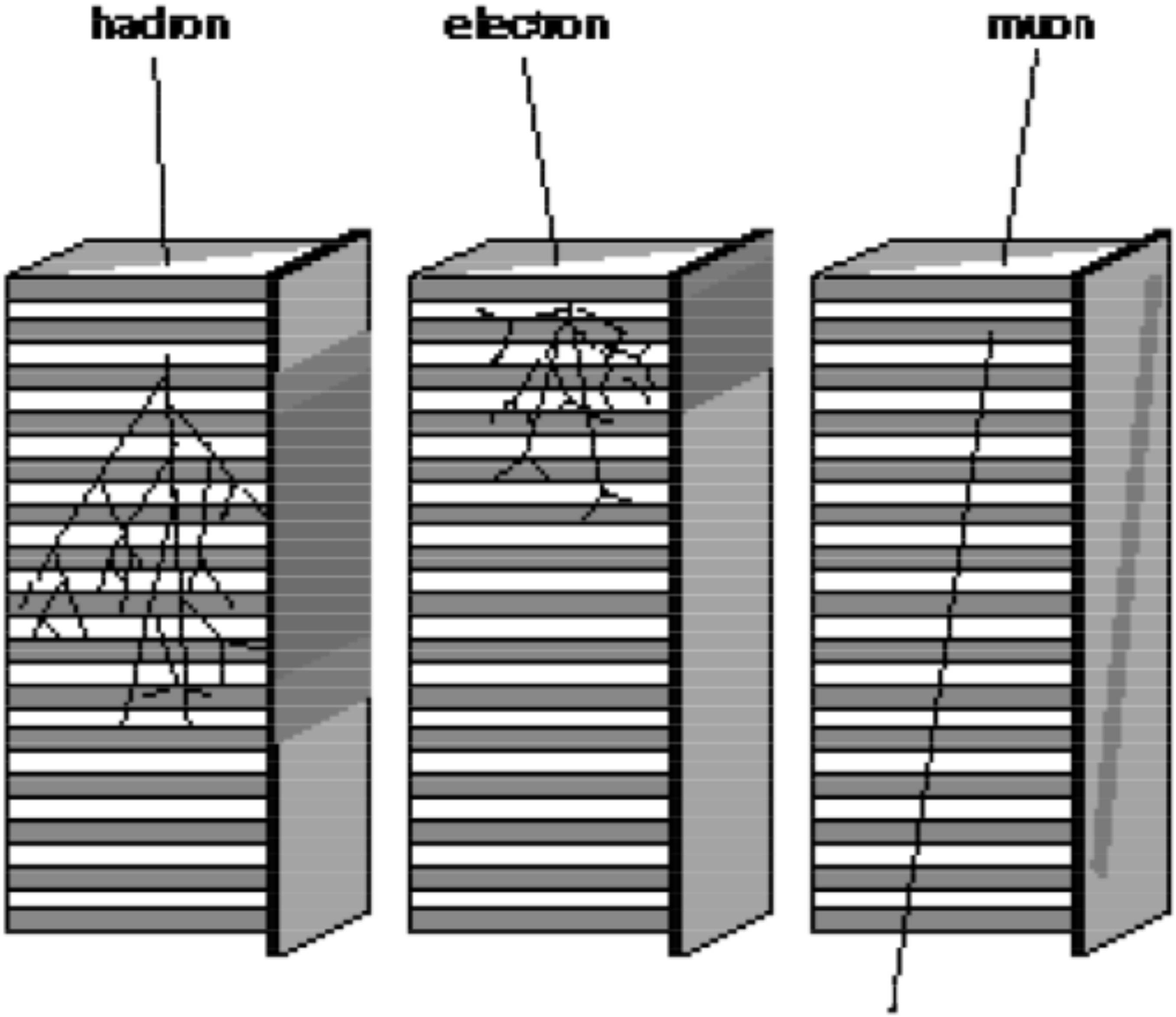

| | |
|---|---|
| Proton-Proton | 2835 bunch/beam |
| Protons/bunch | $10^{11}$ |
| Beam energy | 7 TeV ($7 \times 10^{12}$ eV) |
| Luminosity | $10^{34}$ cm$^{-2}$ s$^{-1}$ |
| Crossing rate | 40 MHz |

every bunch crossing
~ 25 collisions
~2000 particles produced

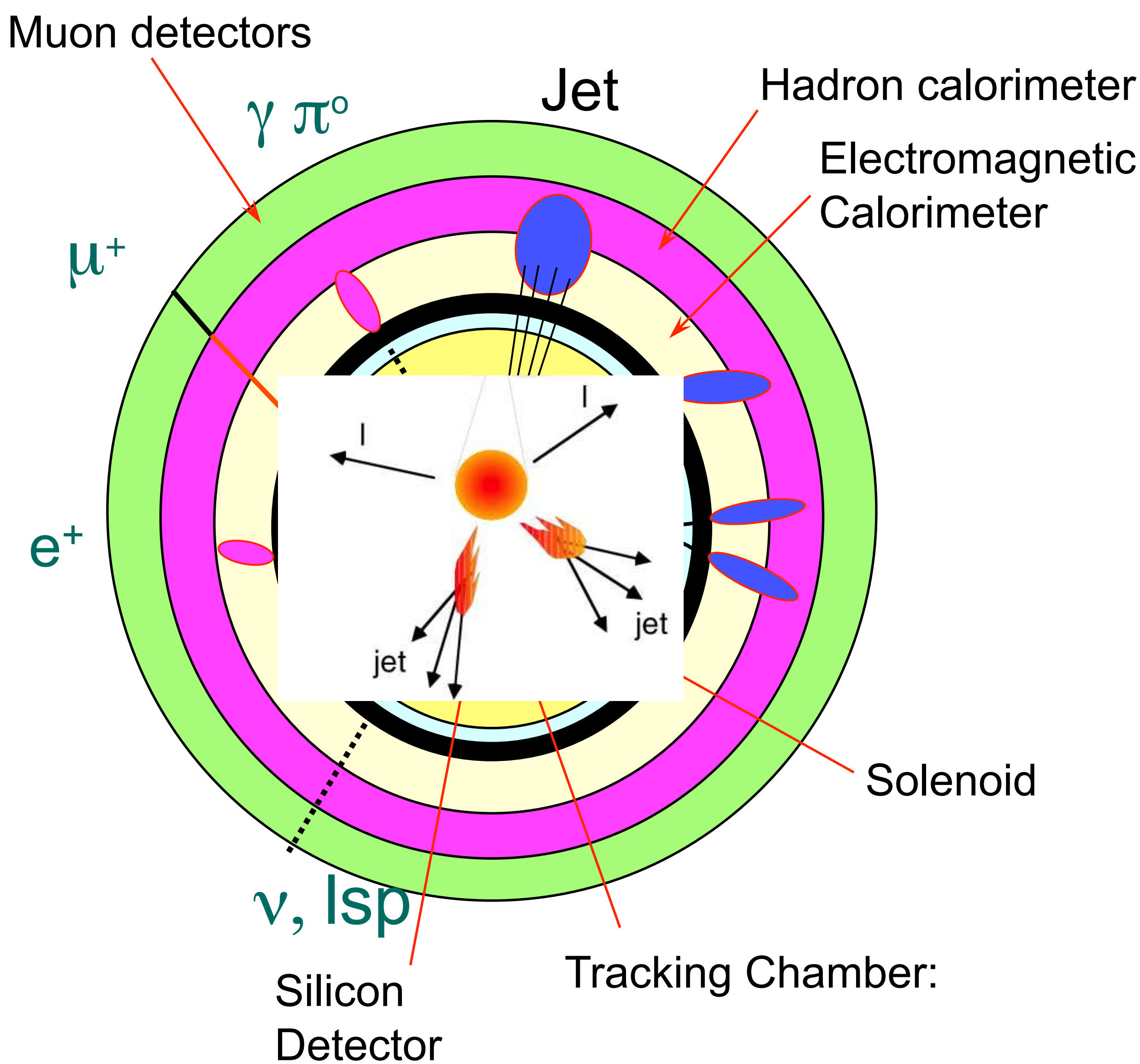
# Collider Detector Schematic



# Interaction of Particles with Matter



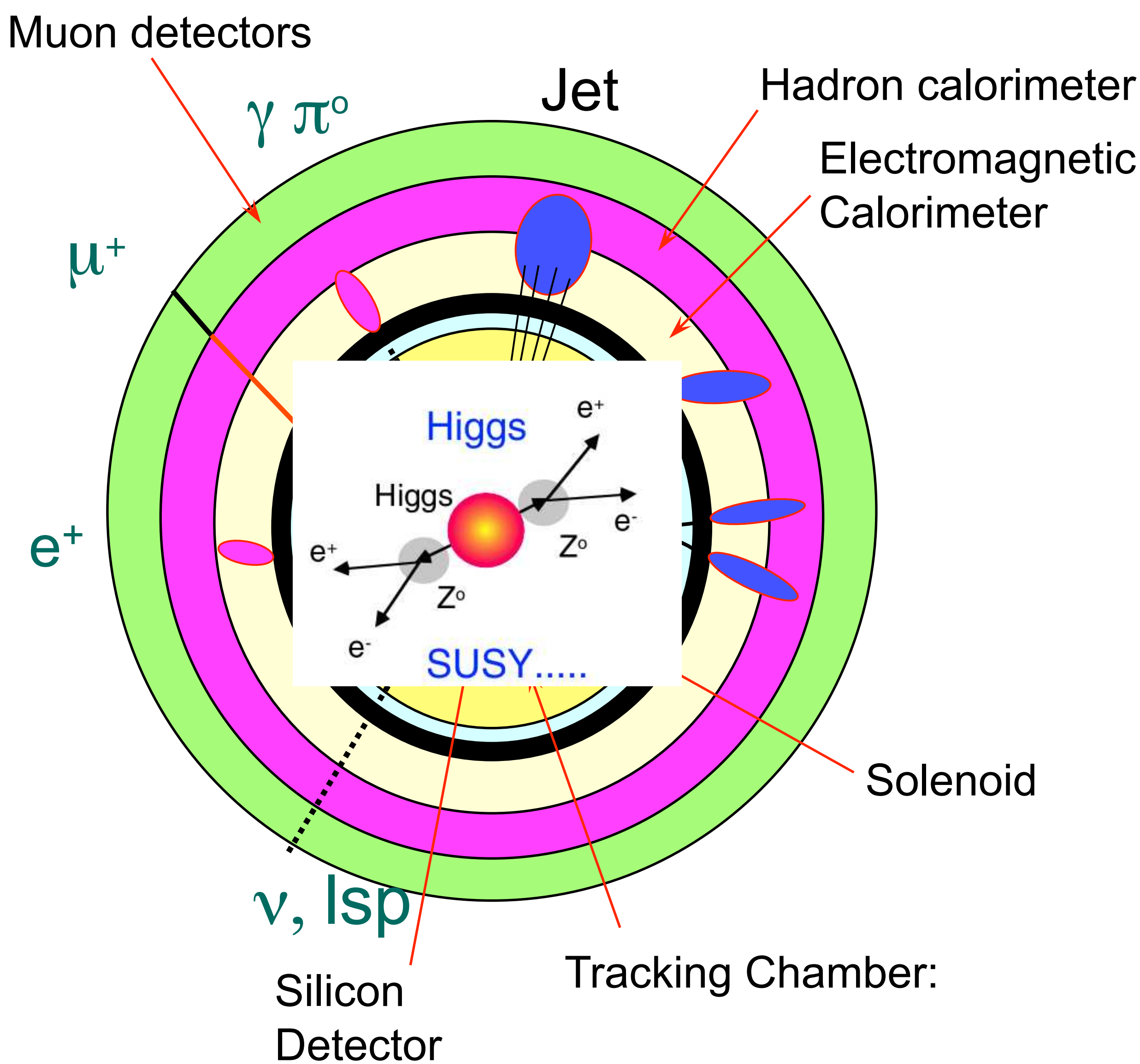
# How to trigger on this event?






Fermilab

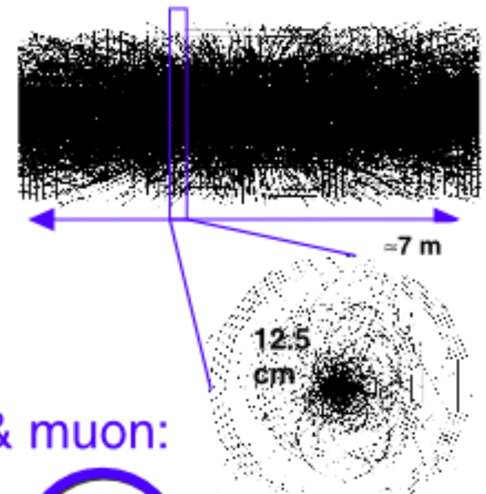
# How to trigger on this event?



# ATLAS and CMS Strategy

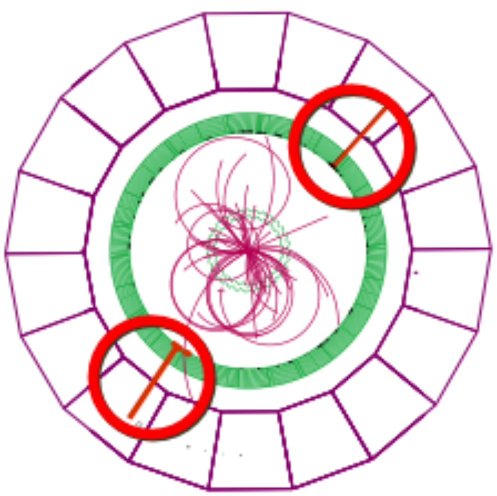
## Level-1 : only calorimeters & muons ....

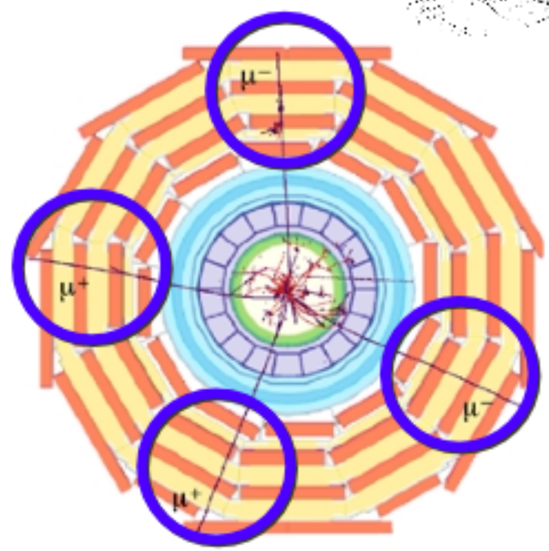
Compare to Central tracking at $L = 10^{34}$
(50 ns integration, ≈1000 tracks)




Algorithm Complexity
+
huge amount of data

Pattern recognition much easier on calo & muon:



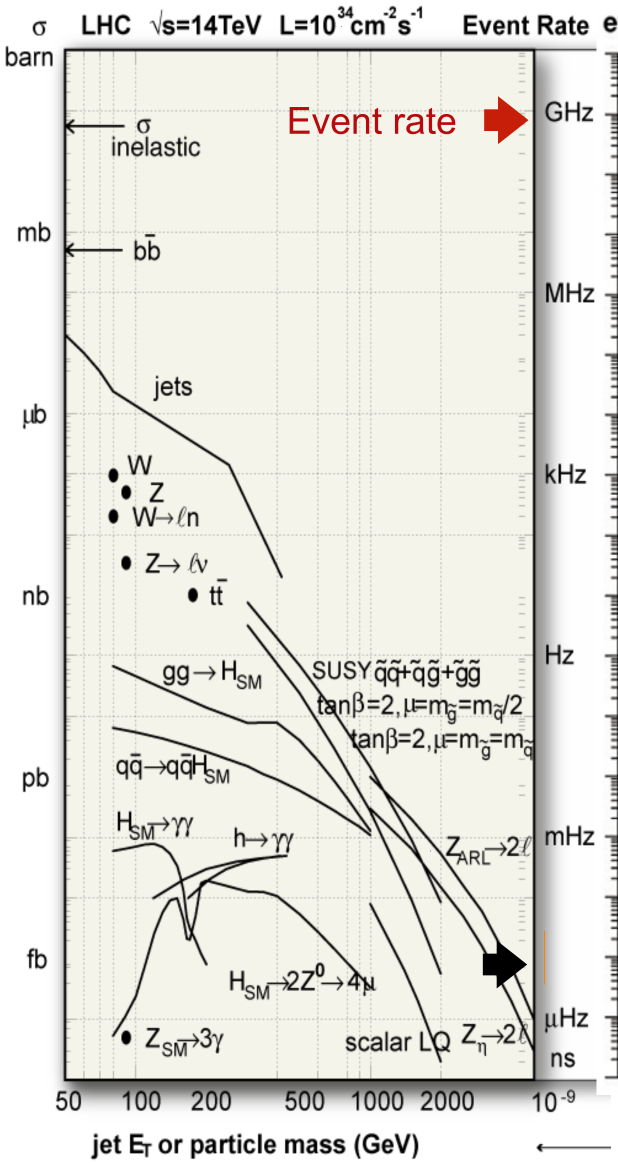


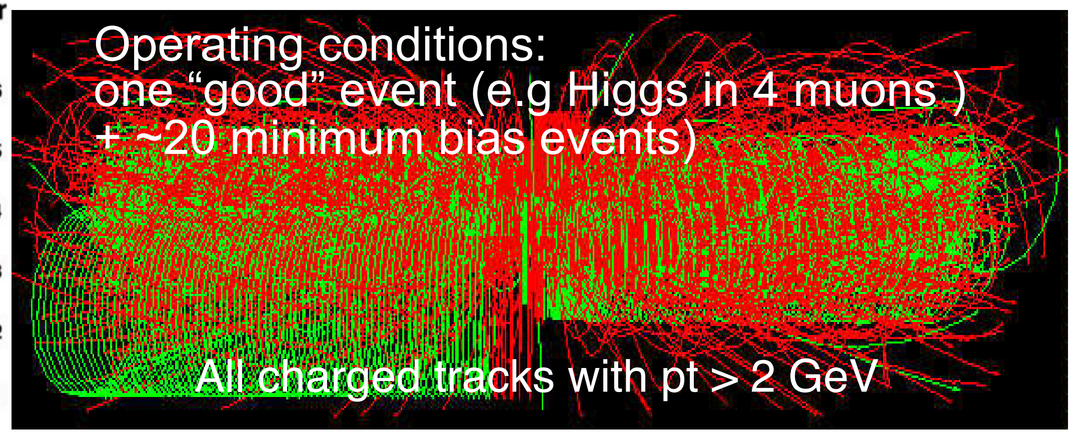

Complexity handled in software on CPUs

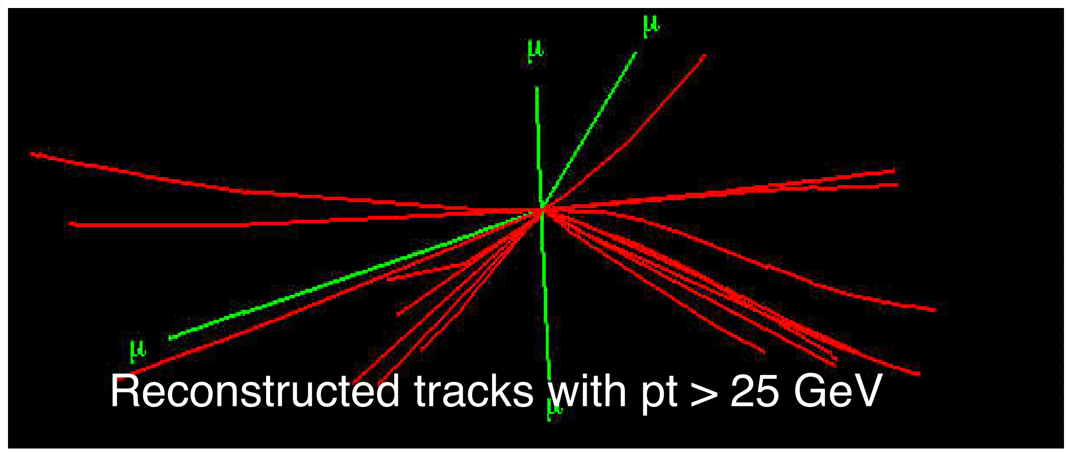
at high level trigger

The approach works well at low luminosity

# Collisions (p-p) at LHC



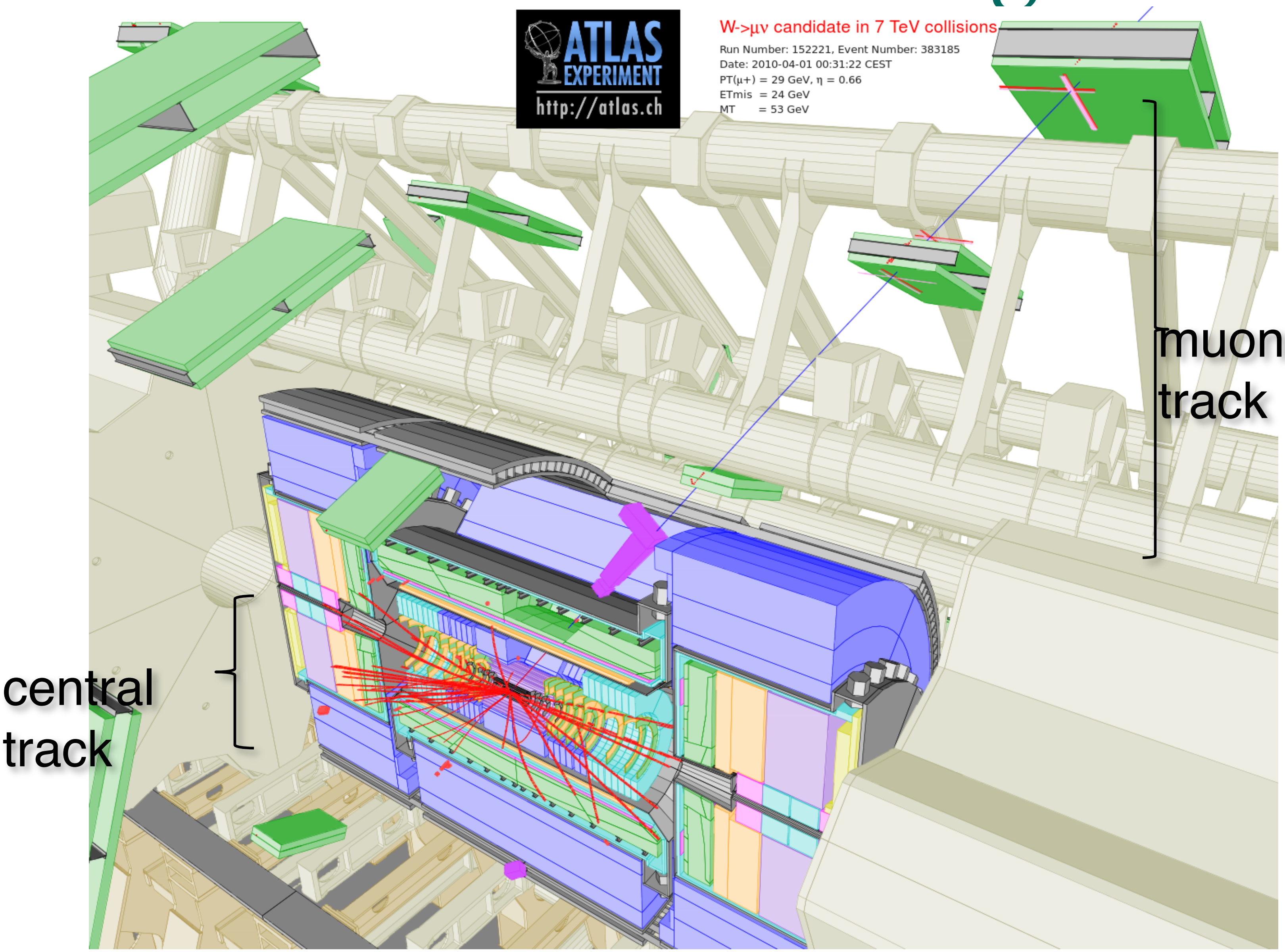



# The importance of individual tracks

- Many/most new physics scenarios produce final states containing heavy elementary particles *(b* quarks & $\tau$ leptons).
  - must be separated from an enormous background of light quarks and gluons produced through the strong nuclear force
    - *b*-jets: displaced vertices from *B* meson with picosecond lifetime
    - $\tau$-jets: 1 or 3 tracks in a narrow cone with a surrounding isolation region due to the decay of a relatively low mass object.

- Even for the traditional workhorse trigger, an isolated high energy electron or muon, tracking is essential at very high accelerator intensity: The usual isolation (calorimeter) deteriorates badly in its efficiency because it integrates over the 25-75 *pp* collisions per beam crossing. Reconstructed tracks each point back to the beam. Isolation only using those close to the muon or electron at the beamline largely removes the effect of the "pile-up".

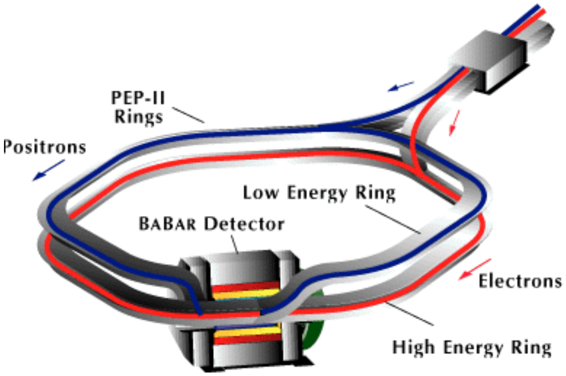
# Tracking

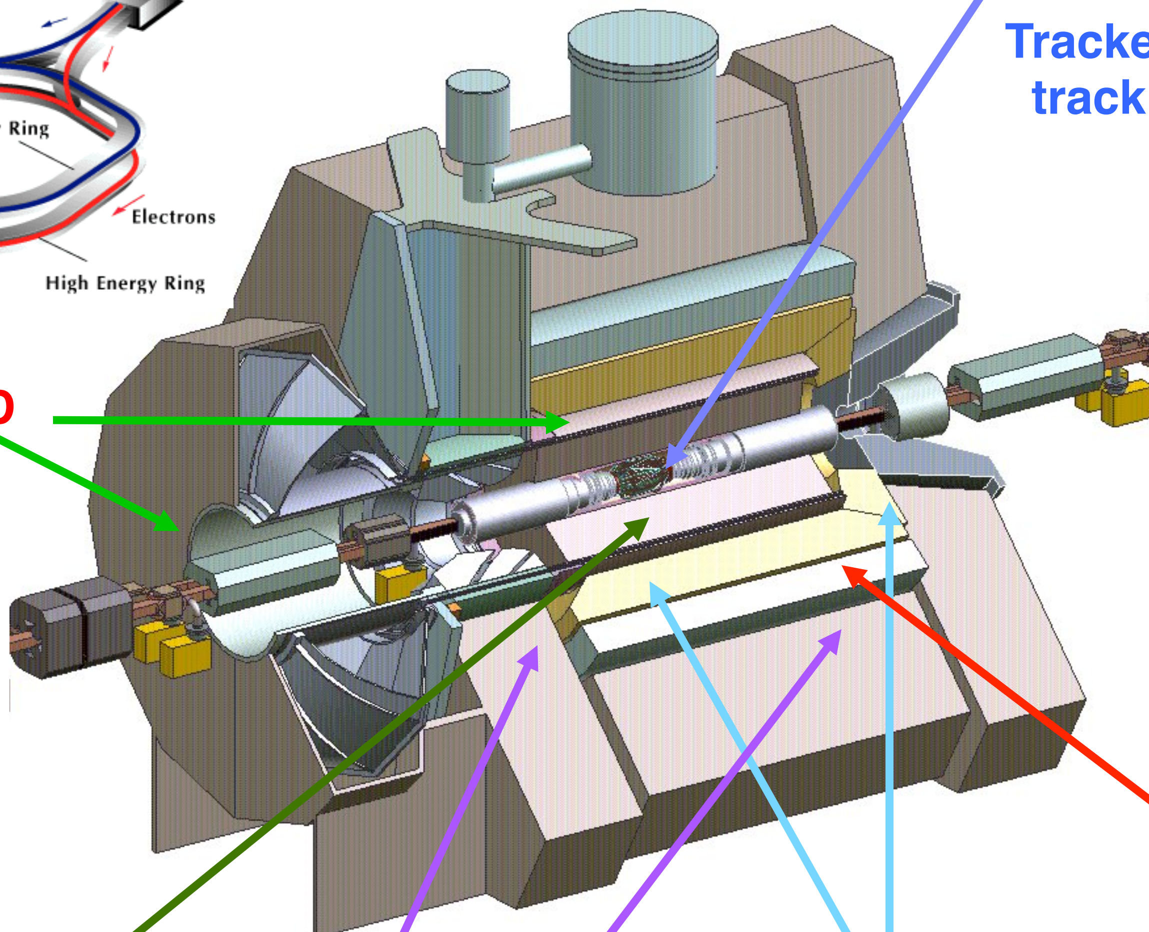

ATLAS EXPERIMENT
http://atlas.ch
W->μν candidate in 7 TeV collisions
Run Number: 152221, Event Number: 383185
Date: 2010-04-01 00:31:22 CEST
PT(μ+) = 29 GeV, η = 0.66
ETmis = 24 GeV
MT = 53 GeV
muon track
central track

# Case Studies:
# → Let's learn a bit more about track trigger with examples

- Case Studies: track trigger
  - Babar L1 Drift Chamber Track Trigger
  - A few words on D0 fiber tracker trigger
  - CDF L2 Silicon Vertex Trigger
  - Atlas L2 Fast Track Trigger
  - Challenges in track trigger (next lecture)

# The BaBar Detector



**Silicon Vertex Tracker:**
**tracking, $\Delta z$**

**3.1 GeV e+**

**DIRC PID**
***p/K/$\pi$***

**9 GeV e-**

**1.5T Solenoid**

**Drift Chamber**
**tracking, dE/dx**

**Electromagnetic Calorimeter:**
**for $\pi^0$, $\gamma$'s, e ID**

**Instrumented Flux Return**
**$\mu$, neutral hadron ID**

# Detector requirements

***Technology:***

**Vertex detection:**

$\sigma(\Delta z) \sim 1/2$ average$(\Delta z) \sim 125\ \mu m$
single vertex $\sigma < 80\ \mu m$
Low Pt (< ~ 100 MeV) Tracking (SVT) : D* --> D $\pi$ ...

*Double sided silicon micro-strip*

**Main Tracking:**

$\sigma(p_t) \sim 0.3\%$ Pt (P > 1 GeV), up to 4 GeV
$\sigma(dE/dx) \sim 7\%$ for PID at low P

*Small-cell cylindrical drift chamber*

**EM Calorimeter: high resolution ~ *O(1%)***

Detect $\gamma$ down to ~ 20 MeV : $\pi$0 asymmetric decays
electron ID: 0.5 GeV --- 9 GeV (kinematics limit)

*CsI (Tl) + silicon pin diode readout*

**Muon & neutral hadron detector:**

*RPC + Fe*

$\mu$ Id down to ~ 0.6 GeV, $K_L$ (flight-direction) catcher from B ->J/$\Psi$ $K_L$

**PID**

*DIRC*

K/$\pi$ separation: up to 2 GeV for tagging
up to 4 GeV for B rare decay (B --> $\pi\pi$ vs K$\pi$...)

# BABAR Trigger System ( L1 + L3 )

***Design challenges*** → ***Data taking environment at PEP-II:***

**Beam crossings occur at 238 Mhz**

**Have severe beam background**

**Event time must be determined by trigger**

***Design goal:***

***High eff for ALL physics events***

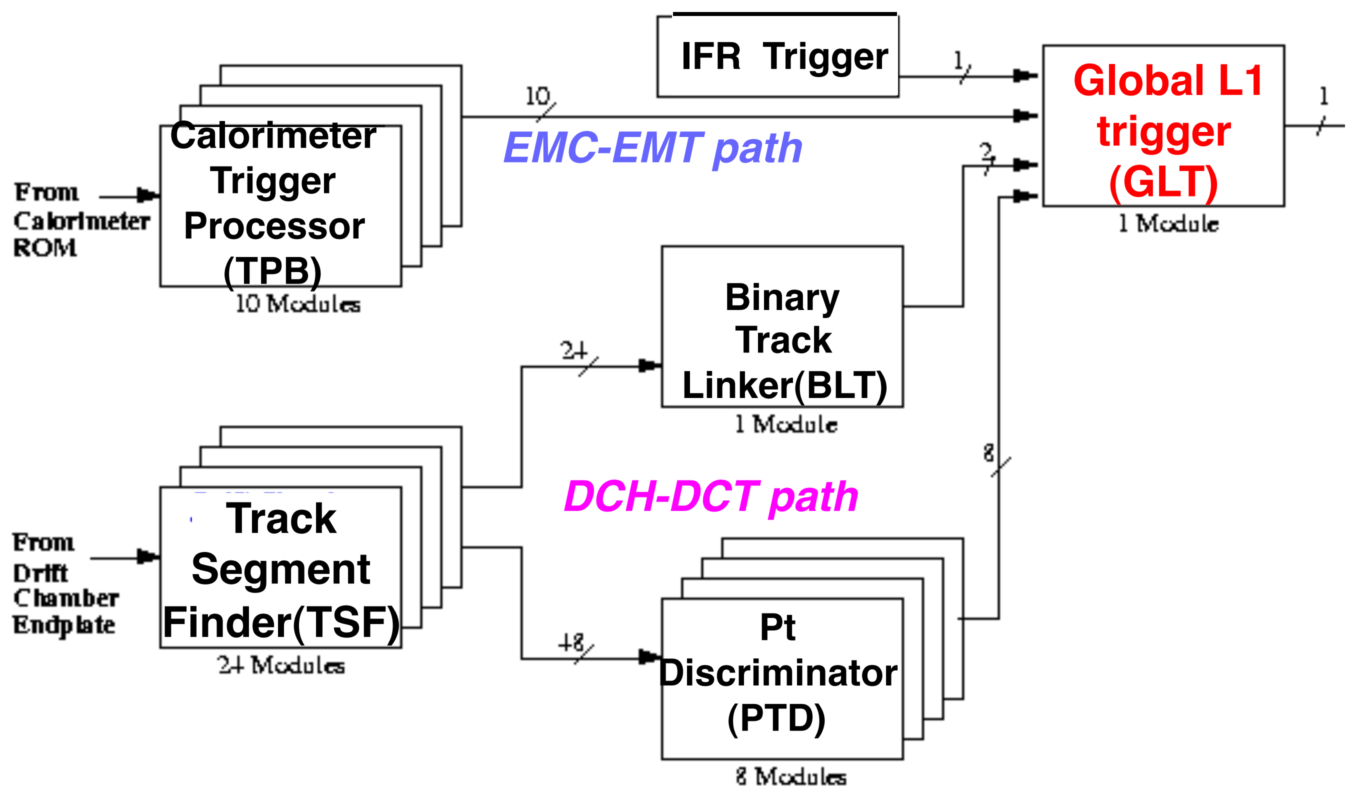
***Keep L1 rate below 2 kHz***

***Trigger latency < 12 μs***

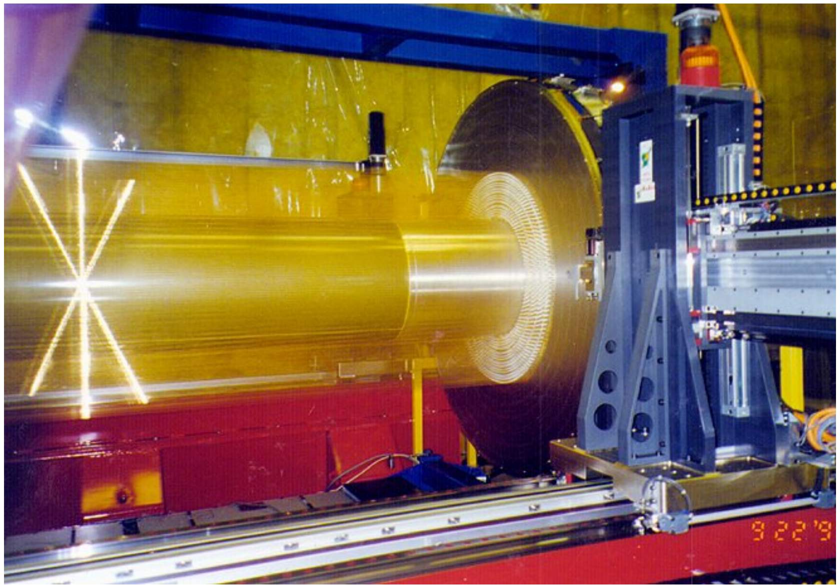
***L1: hardware-based***

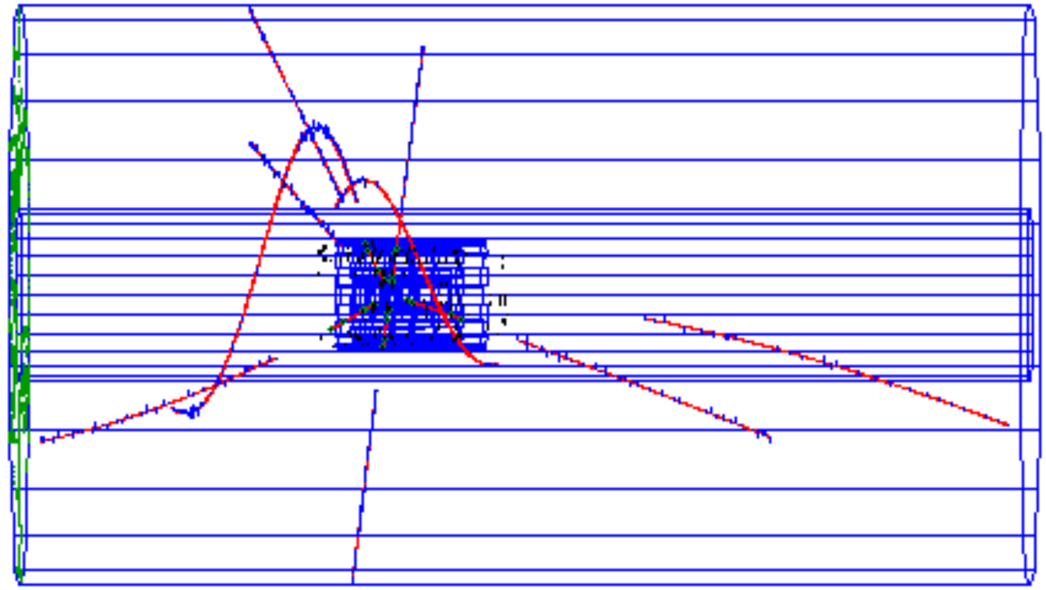
***L3: software-based***

## Level 1 Trigger Implementation



# *The Drift Chamber ( DCH )*





***10 superlayers of 4 layers each***

***Axial and stereo alternate***

***Field: 1.5 T***

***7104 signal wires***

**Cell size:**
***1.2* x *1.8* $cm^2$**

***Helium-isobutane:***

**80%:20%**

***R: 23.6 to 80.9 cm***

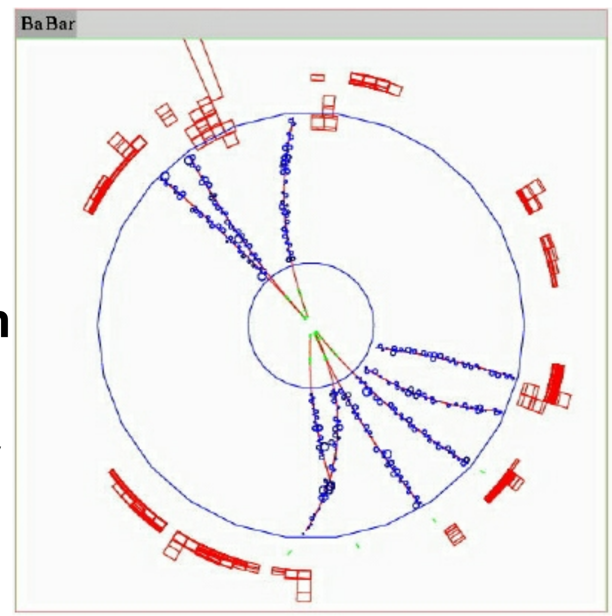
**Design goal:**

**Average hit resolution: 140 um**

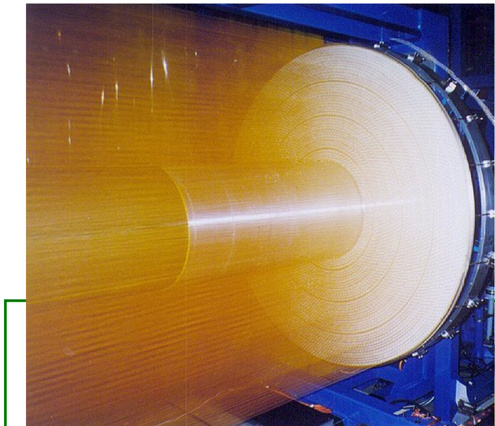
**$\sigma$ (dE/dx) ~ *7 %***

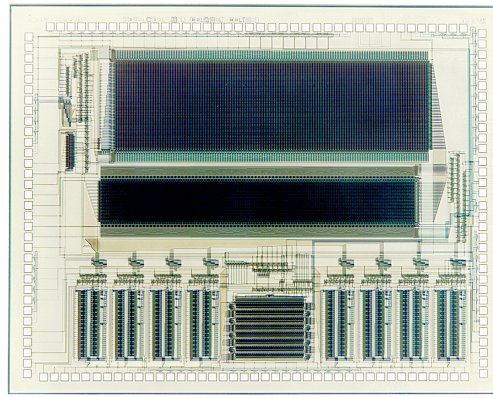
**$\sigma$(pt) / pt ~ 0.3 % for pt > 1 GeV**

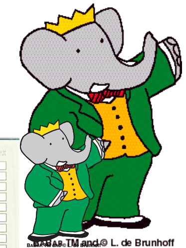
**Material (X0) at $90^0$: 2.08% X0 total**



*Drift Chamber Readout:* *ELEFANT chip*

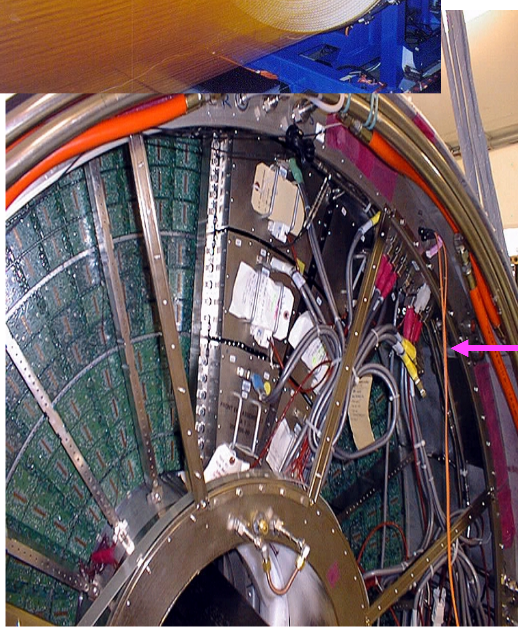







preamp
+ shaper
+ disc.

TDC/FADC
+
trigger output



***Drift chamber endcap***

**ELEctronics For Amplitude 'N Timing**

**0.25M gates, fabricated in 0.8 um CMOS process**

**DCH cable plant: 4 fibers + HV cables**

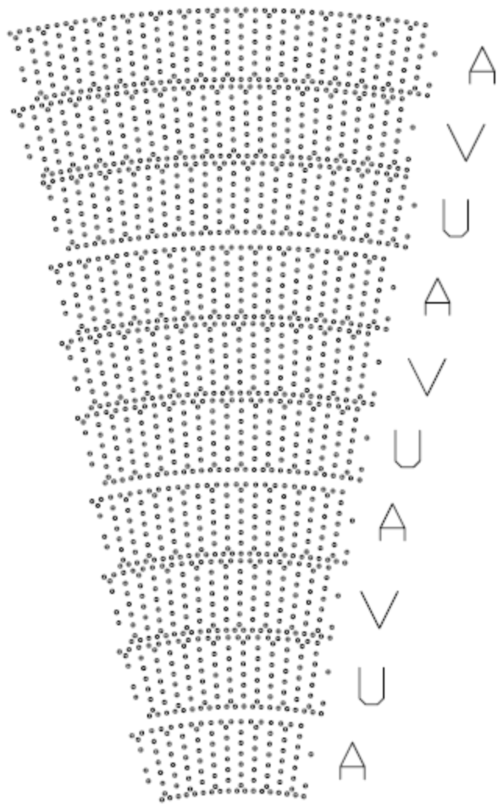
**first complete timing and amplitude system for drift chambers on a chip**

**allows complete DCH readout system to be mounted on the DCH endcap**

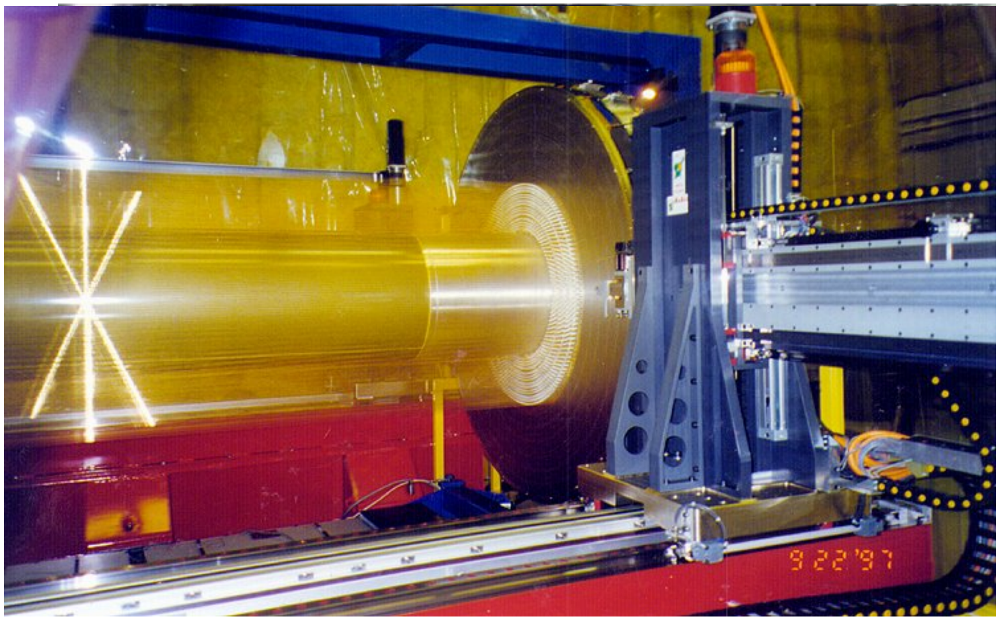
**eliminating >280 off-detector VME cards**

# BaBar Drift Chamber

- Optimized for high-rate *and* low-mass
  - helium-based gas mixture (80% He, 20% isobutane)
  - gas + wires only gives 0.3% $X_0$ at 90
  - small cells (short drift times) allow use in trigger
  - also used for dE/dx measurement

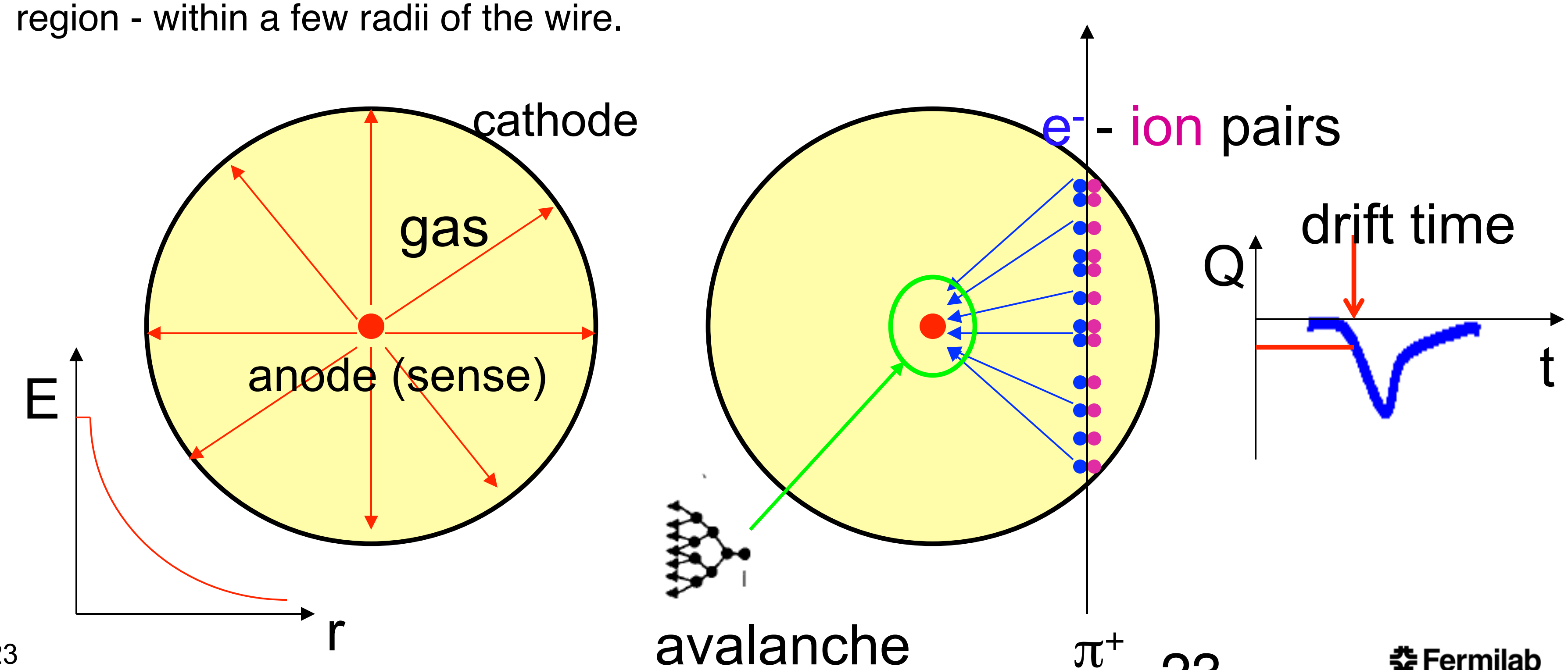



**Cell size:**
***1.2 cm* x *1.8 cm***

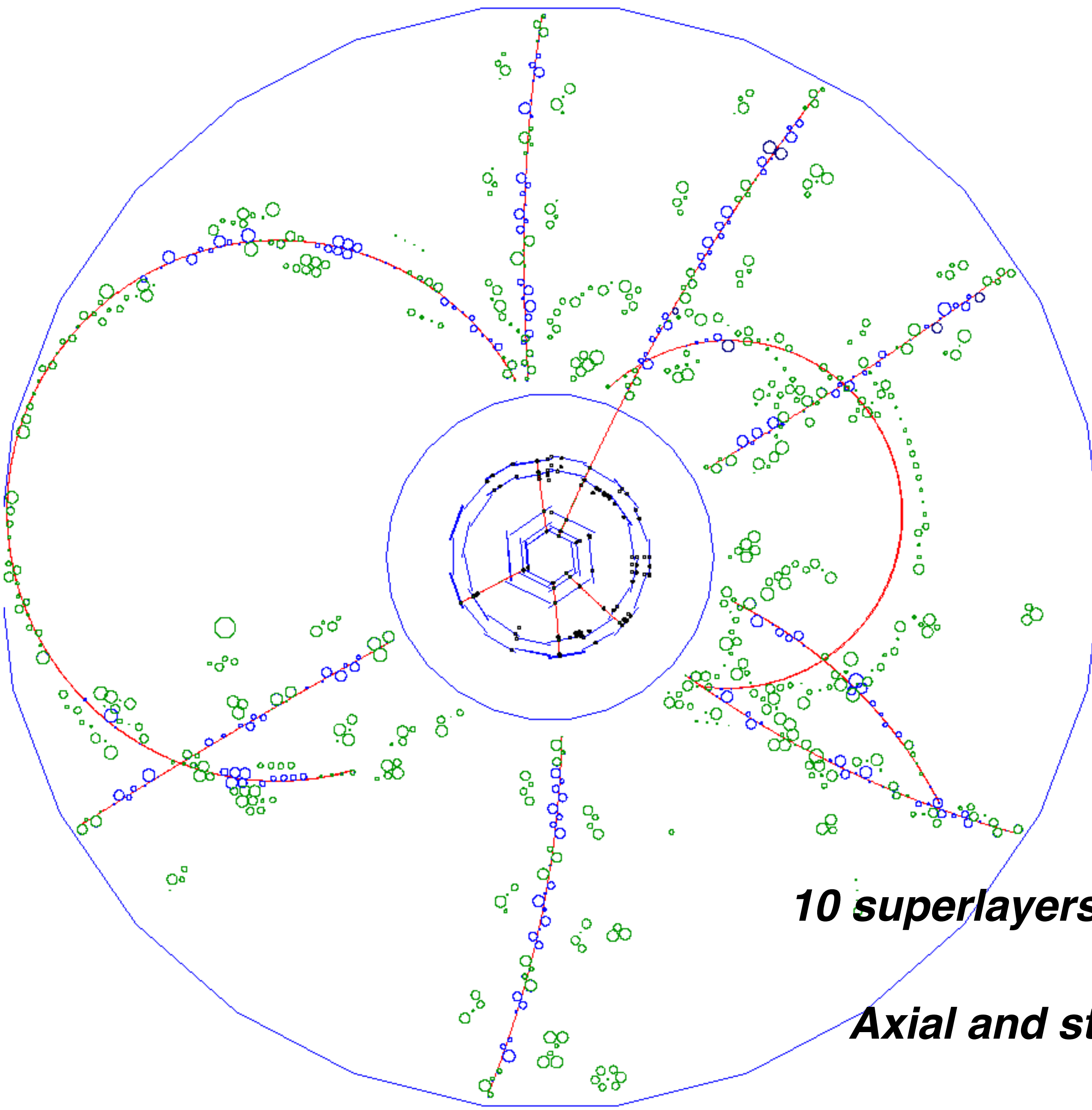


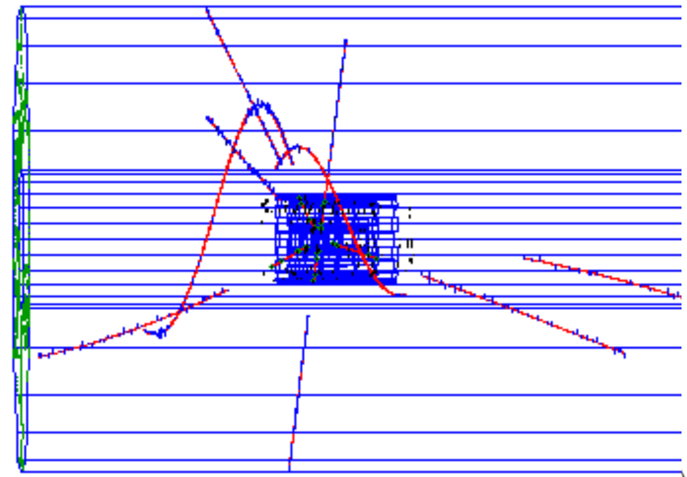
# E-field,   Drift-Time Relation

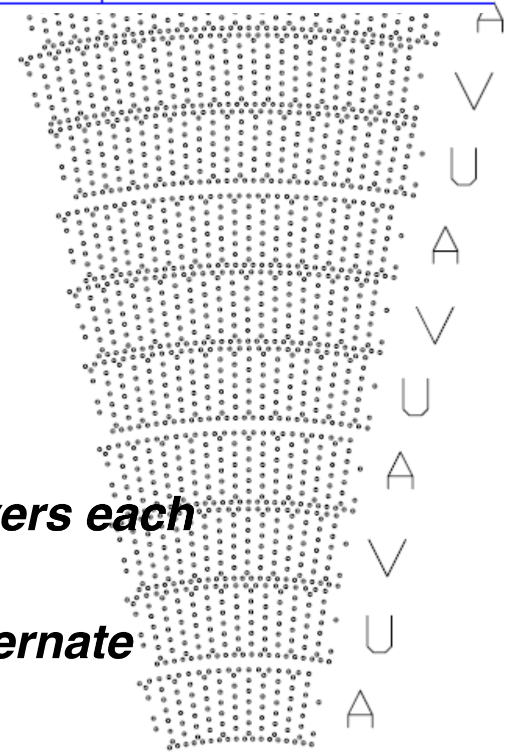
- A charged particle enters gas in E-field, Ionizes gas, produces $e^-$-ion pairs
  - ~300 μm / pair
  - Ionization E ~30 eV / pair
- Primary ionization electrons drift toward anode (sense) wire (low E field region)
- Avalanche multiplication of charges by electron-atom collision in high E field region - within a few radii of the wire.
- Signal induced via motion of charges.
- Measure "drift time", $\Delta t_{drift}$, (first arrival time) of electrons at sense wire relative to a time $t_0$



## *The Drift Chamber ( DCH )*








***10 superlayers of 4 layers each***

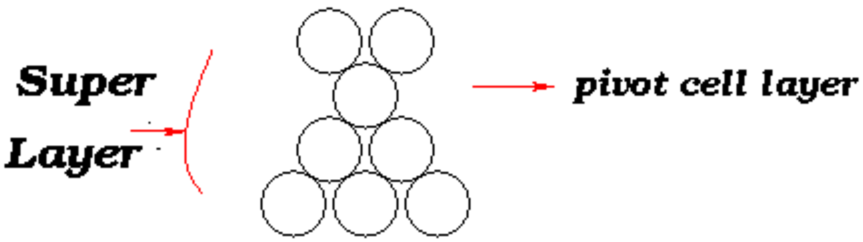
***Axial and stereo alternate***

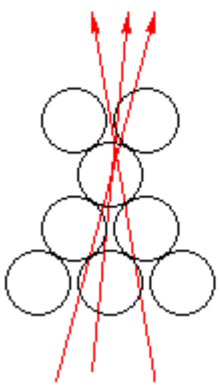
# *Track Segment Finder Concept*

* The search for track segments is organized in terms of pivot cells

* Each pivot cell and seven neighboring cells constitute a pivot group

## *One Pivot Group*



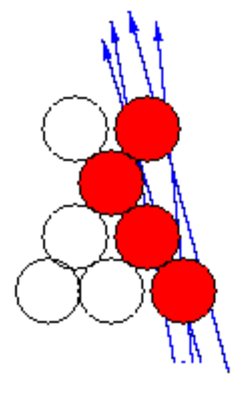
tracks from IP

* Why only 8 cells in one pivot group ?

*for a track from IP passing through the pivot cell,*

*the only cells which can be possibly hit by this track*

*are these 8 cells*

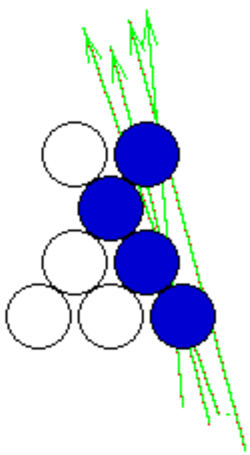


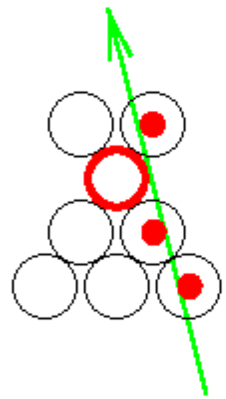
One-Shot Segment Finder

# *Track Segment Finder Concept*

**use drift time information to better determine track position and event time**

**One - shot** *versus* *Counter - Based*





***Event time jitter window:***

**( 99% C.L. )**

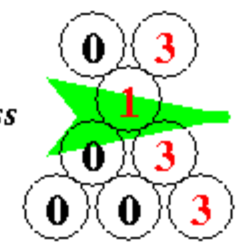
**180 ns @ L 1**

***Position Resolution:***

**0.05 cell width**

**→ 1 mm**

***Look-Up-Table address***




***position and time***

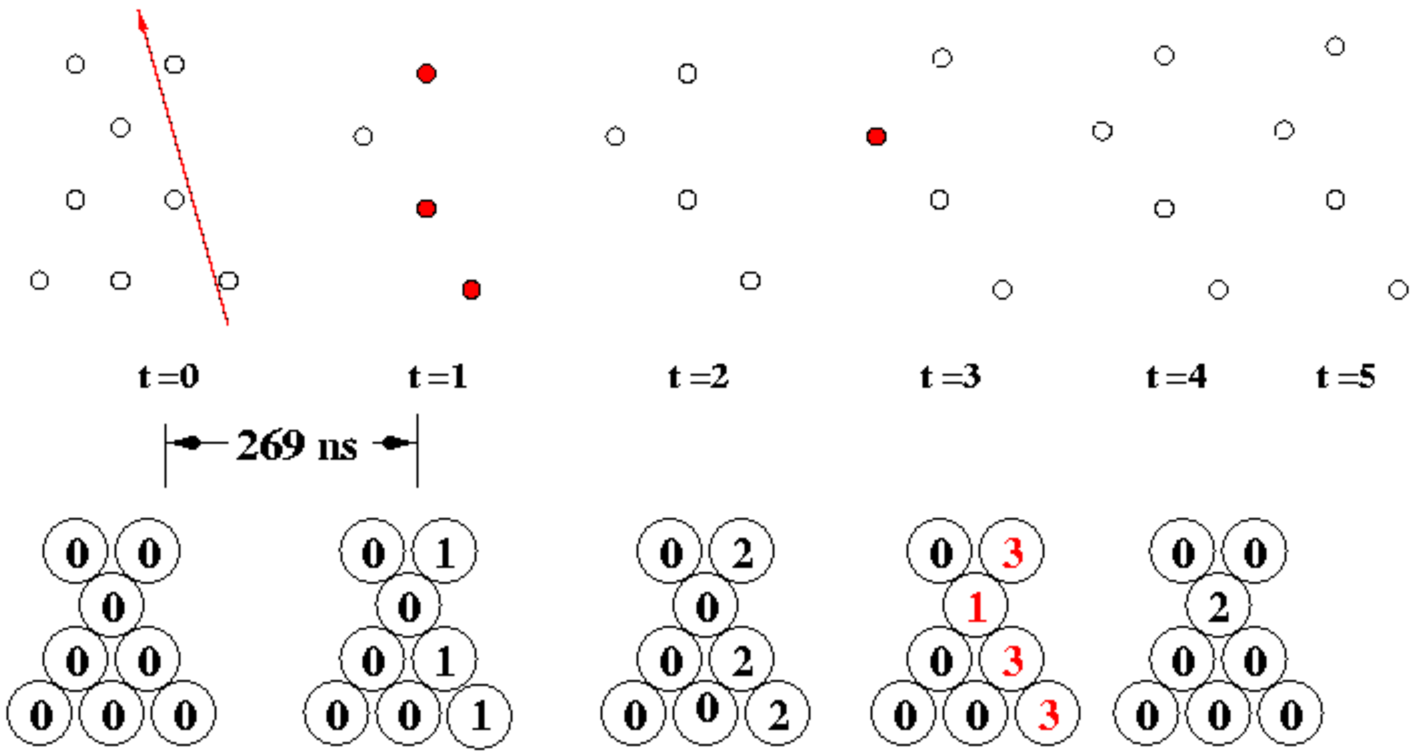


Fermilab

# Track Segment Finder Concept

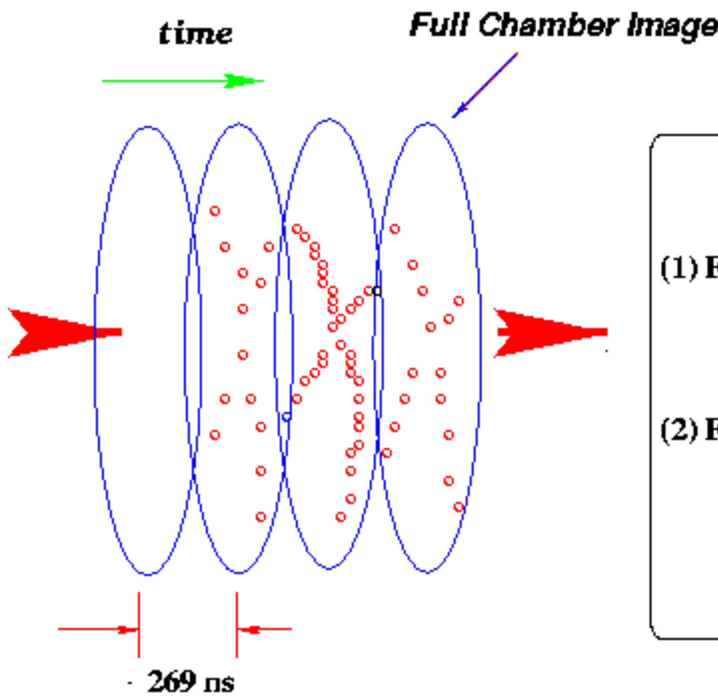
- use drift time information to better determine track position and event time

** Cell hits development in time* → **Counter - Based Segment Finder**




*Look-Up-Table address <-----> track position and time*

# *Track Segment Finder: continuously live image processor*




*TSF Algorithm:*

(1) Find track segments with sub-us time resolution; → **BLT**

(2) Find segment position with sub-cell space resolution; → **PTD**

***Use Drift Time***

*24 TSFs*

24 Gb/s data from Drift Chamber → *via 24 fiber optic Gigalinks*

***10 four-layer superlayers: 7104 cells, max. drift time: 600 ns***

# *Drift Chamber Trigger ( DCT )*

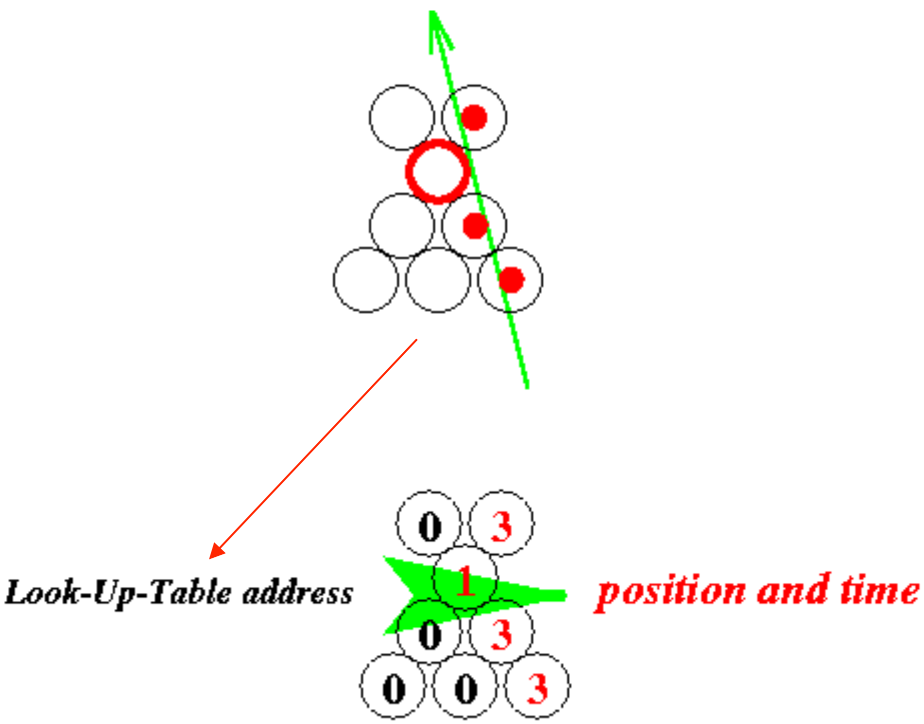
***The heart of DCT is the Track Segment Finder***

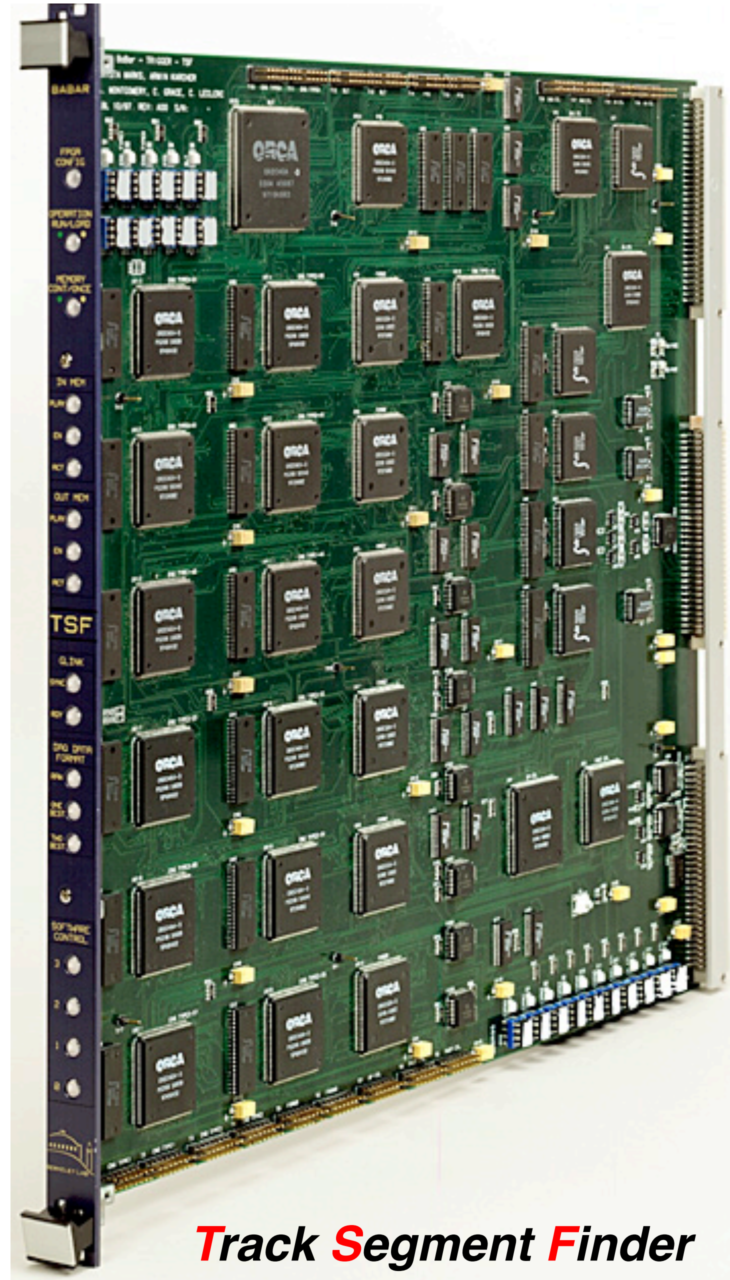
***TSF → continuously live image processor →***

***The method**: using both occupancy and drift-time information, to find track segments continuously with:*

*time resolution of ~ 100 ns,*

*spatial resolution ~ 1 mm*



# *Drift Chamber Trigger ( DCT )*

***The heart of DCT is the Track Segment Finder***

***TSF → continuously live image processor***

***A novel method: using both occupancy and drift-time information, to find track segments continuously with:***

***time resolution of ~ 100 ns,***

***event-time jitter window ~ 100 ns***

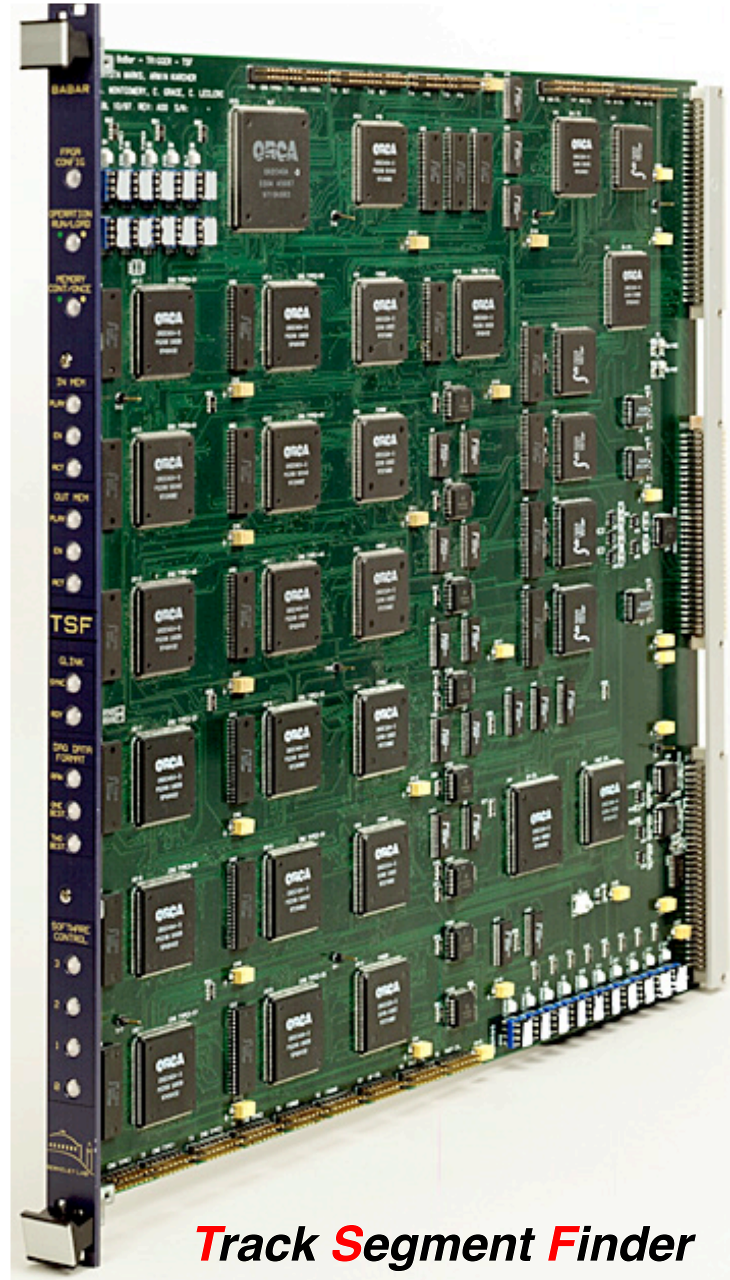
***spatial resolution ~ 1 mm***

***used for track Pt Discrimination***

***(1) send segment patterns downstream for L1 trigger decision making***

***Upon a L1 accept:***

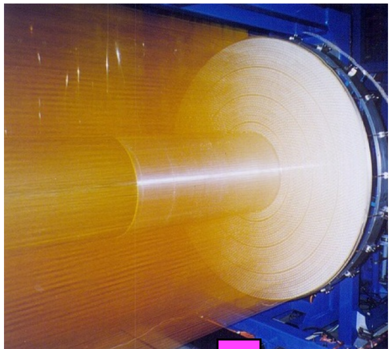
***(2) send segment patterns to the DAQ system for use in Level 3 ...***




***Track Segment Finder***

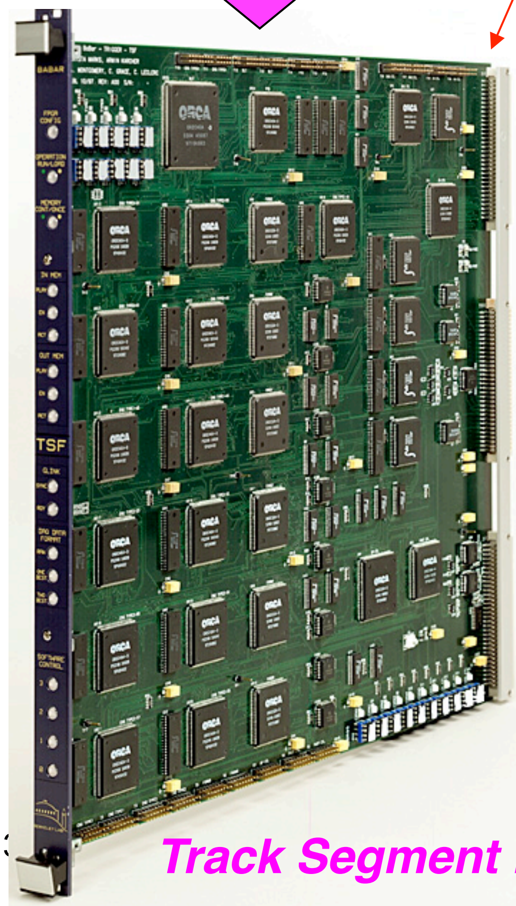
# L1 Drift Chamber Trigger Hardware (LBNL)

*Drift Chamber*



*Trigger data*
*24 Gbits/s*

*Track segments finding*

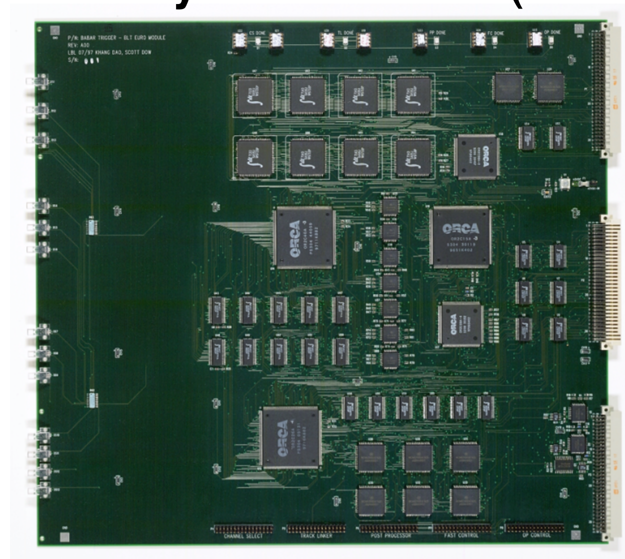


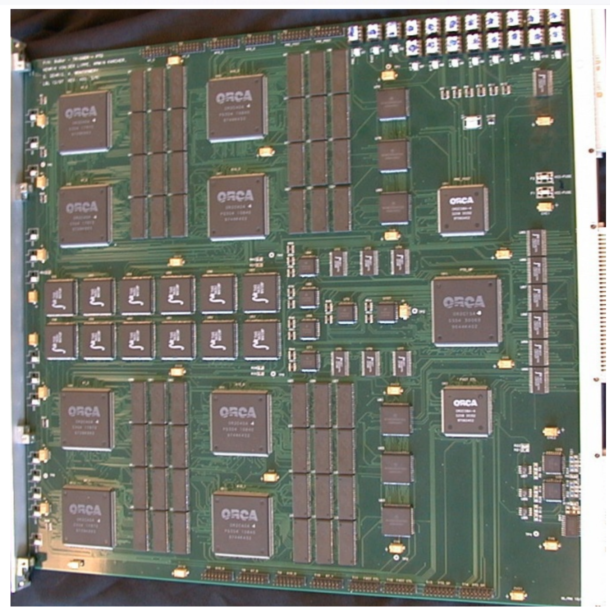
*Track Segment Finder (TSFx24)*

*Coarse data for all supercell hits*

*Fine position data for segments found for <u>axial</u> SLs*

**Binary Track Linker (BLT x1)**





**PT Discriminator (PTD x 8)**

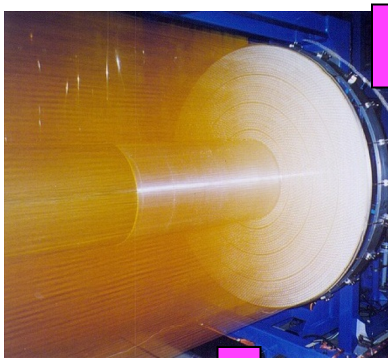
**tracks**

**Global Trigger**

**High Pt tracks**

## Upon a L1 Accept …

Drift Chamber



DAQ

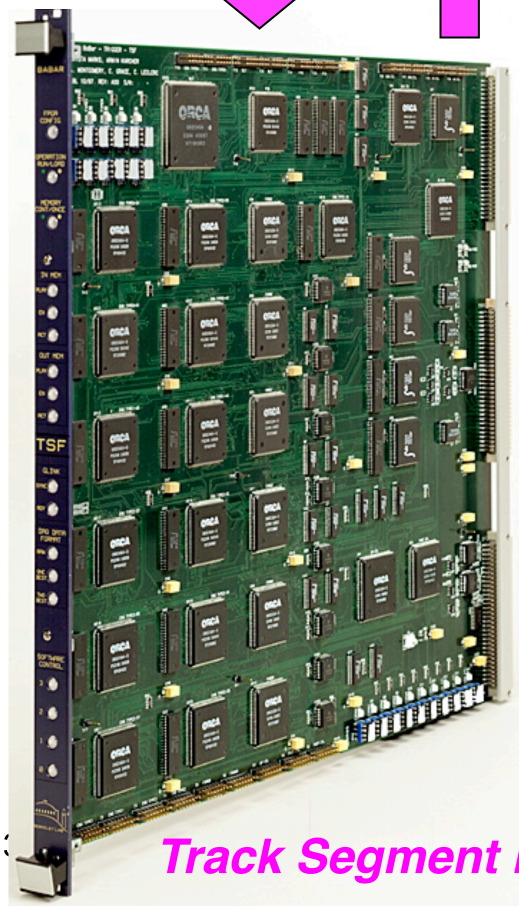
DAQ

**Level 3 Filter**

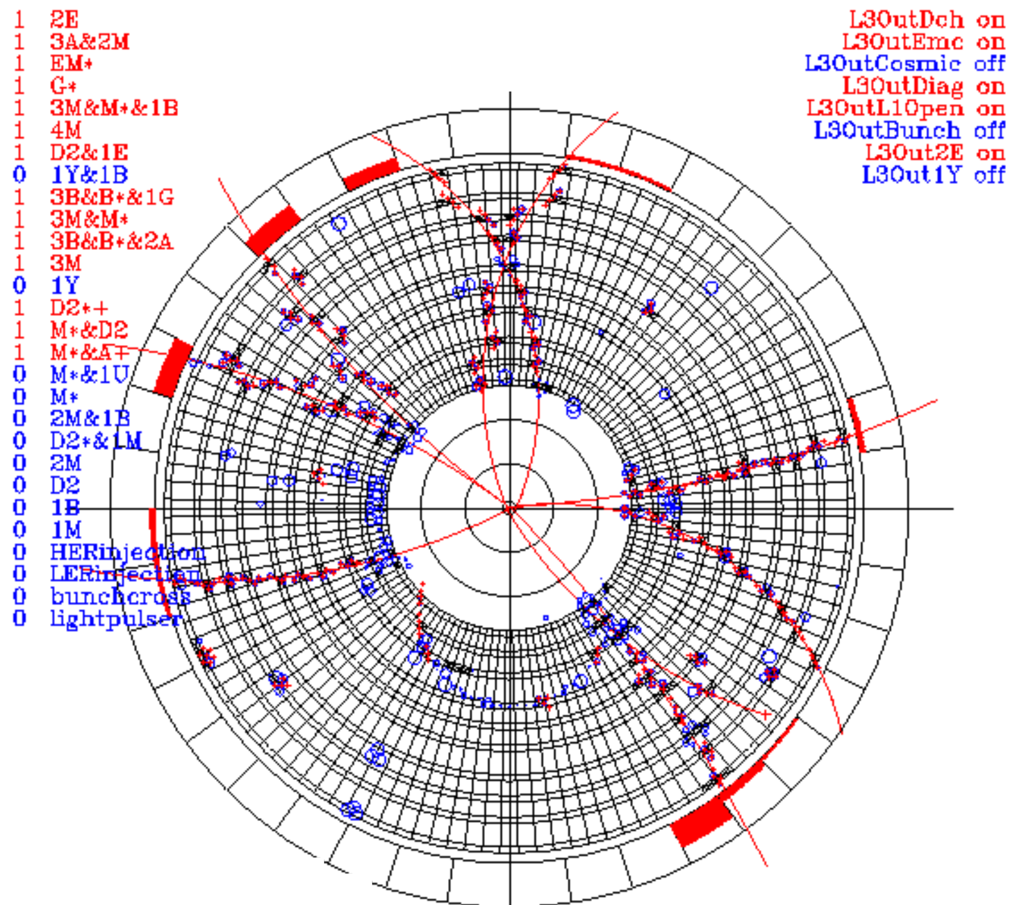
***Using segments found by TSFs as seeds, fast algorithm applied to DCH data to further reject background... reduce the rate from 2 kHz to 100 hz.***

*Trigger data*
*24 Gbits/s*

*Track segments found*

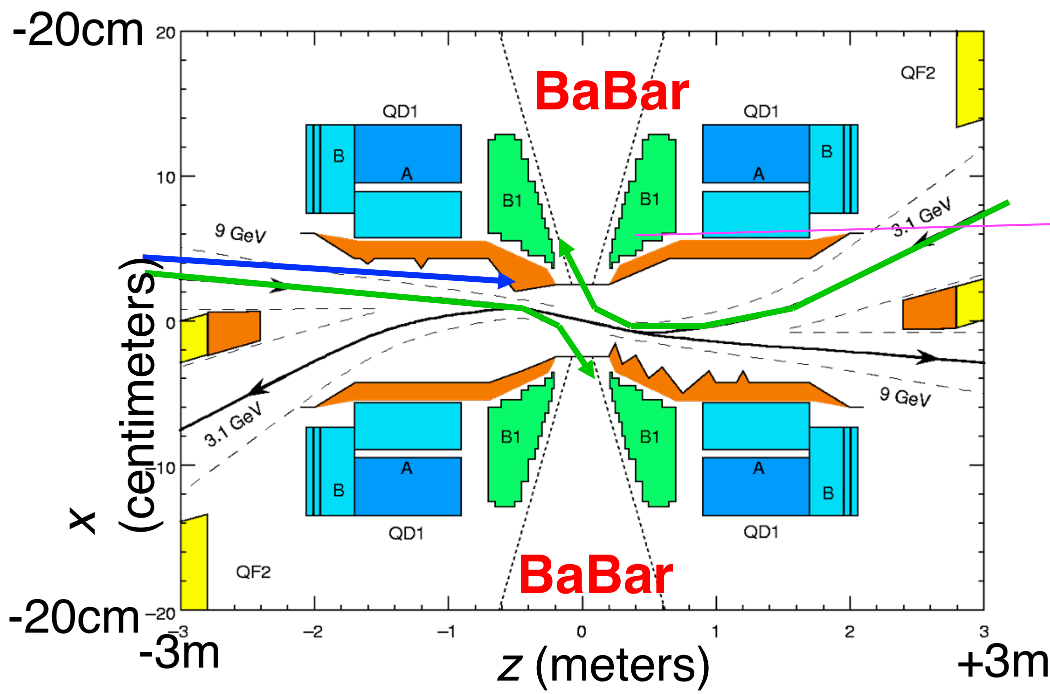


*Track Segment Finder (x24)*




**L3 Event Display**

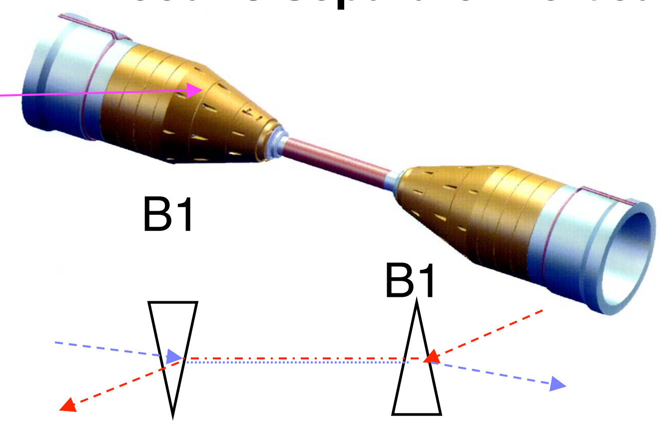
# Interaction Region and High Background

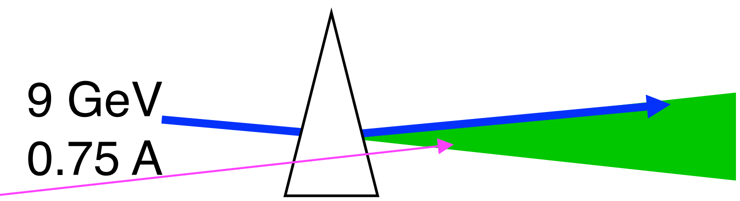



**Beam Crossing @ 238 MHz**

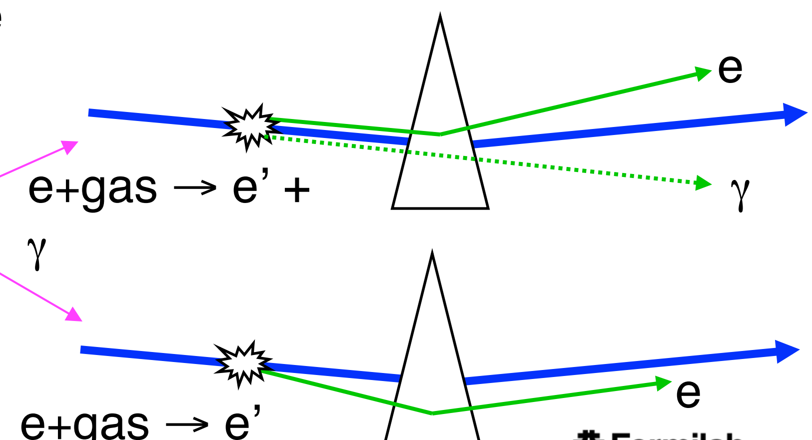
**Head-on collision**

**Need to separate the beam**



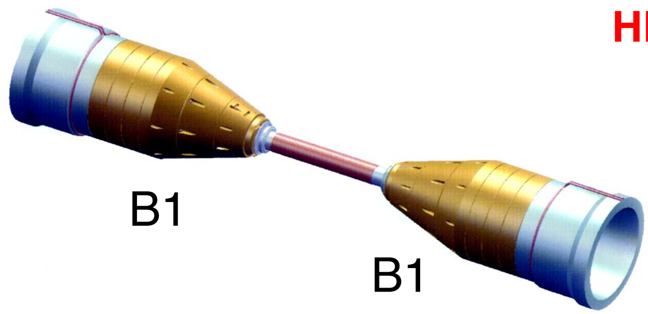




**beams passing through bending magnets generate a "fan" of X-rays in bending plane**

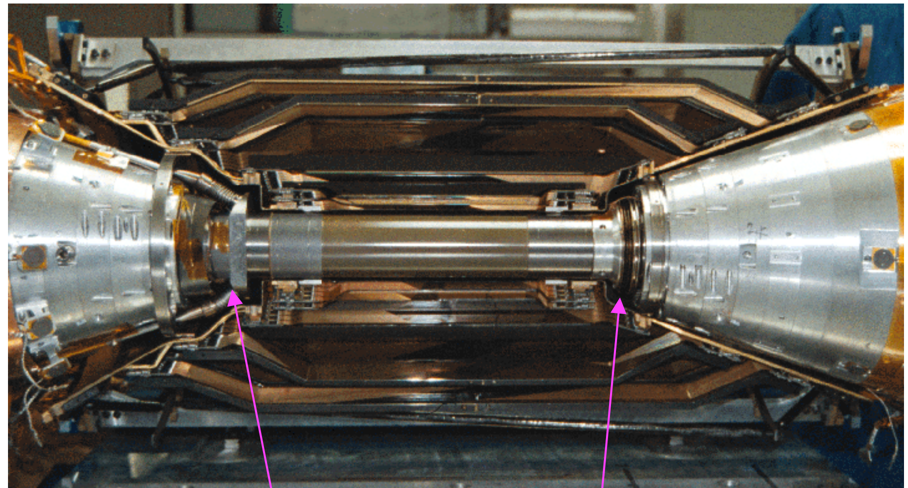



**Bremsstrahlung & Coulomb interactions produce energetic e, γ along beam line**

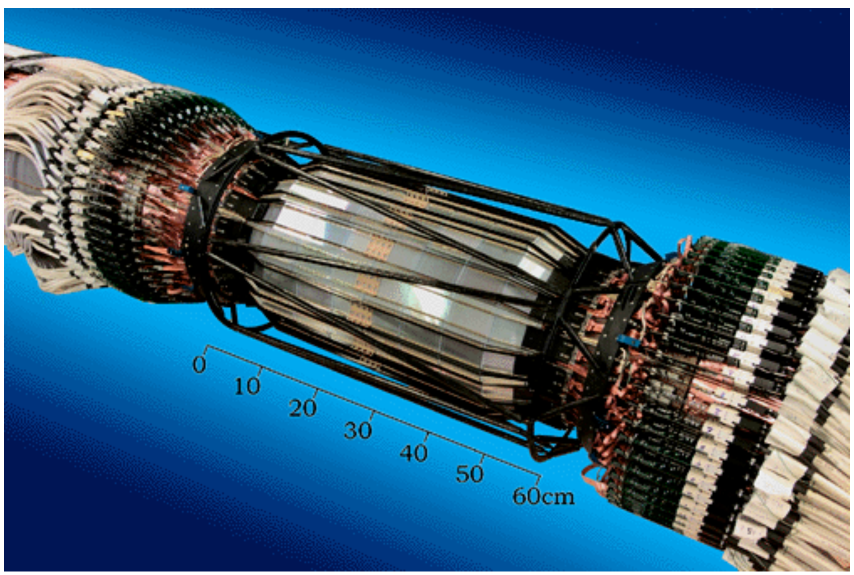
***High Luminosity*** ➡ ***High Background***




HER



LER





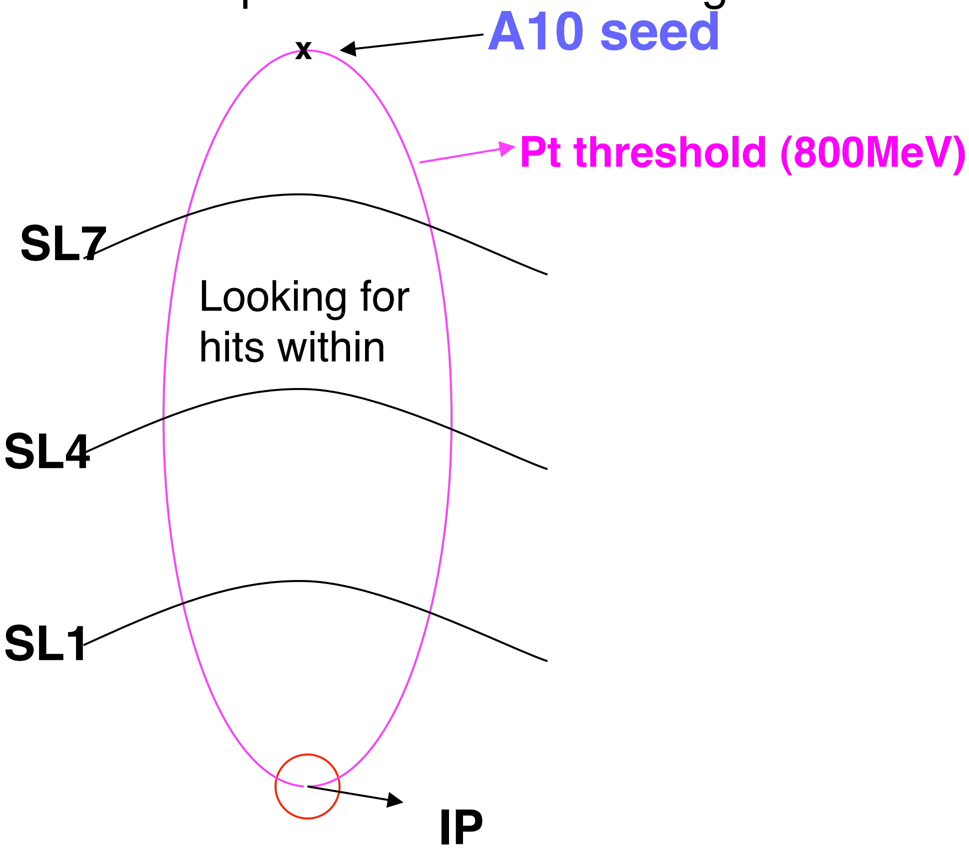
*Lost particle interact with beampipe flange*

*real collision*

*due to a step in the synchrotron mask*

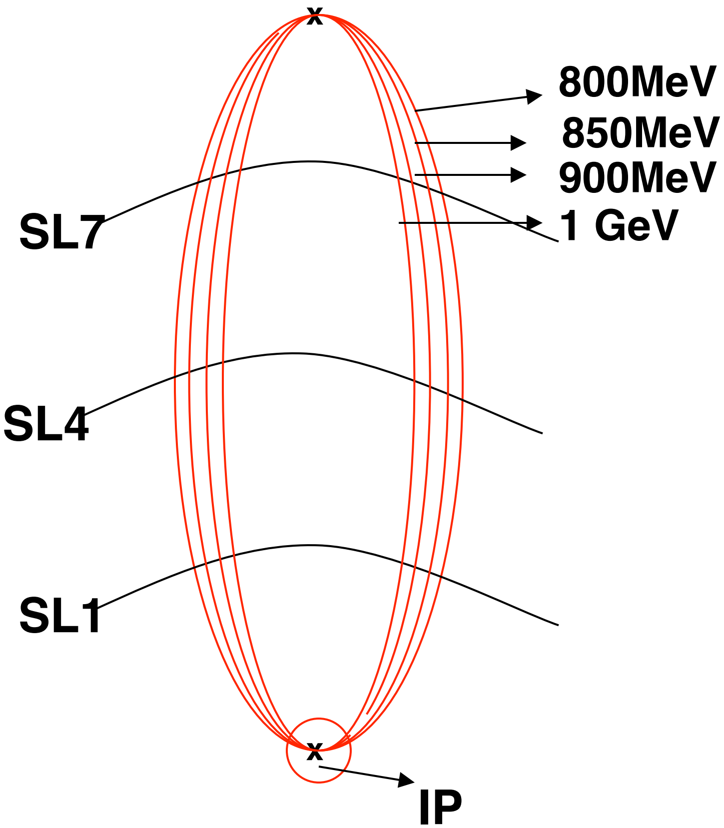
**Track z distribution for L1 passthrough events**

Original PTD algorithm is very simple:
→Works very well at design luminosity
→Needs improvement at much higher luminosity




4 hits on one track with σ~1mm, should be able to tell whether the track is from IP

One easy way to Improve PTD

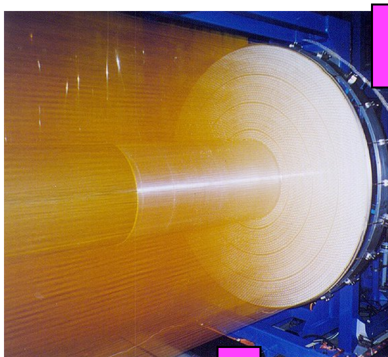



**Tracks (above threshold) coming from IP should leave all the hits in one of the slices**

**Tracks not coming from IP will likely leave hits in different slices.**

*Would be even better to use stereo layer info to determine z*

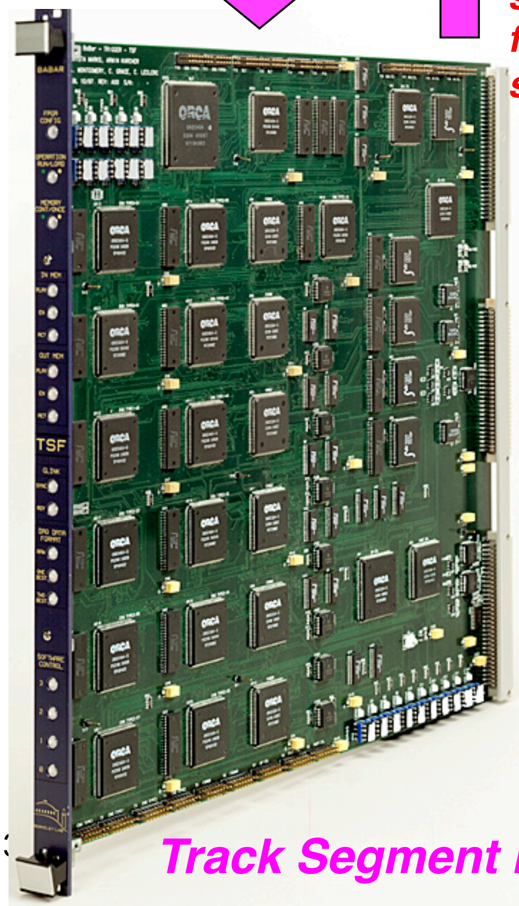
## Drift Chamber



DAQ

DAQ

Trigger data
24 Gbits/s

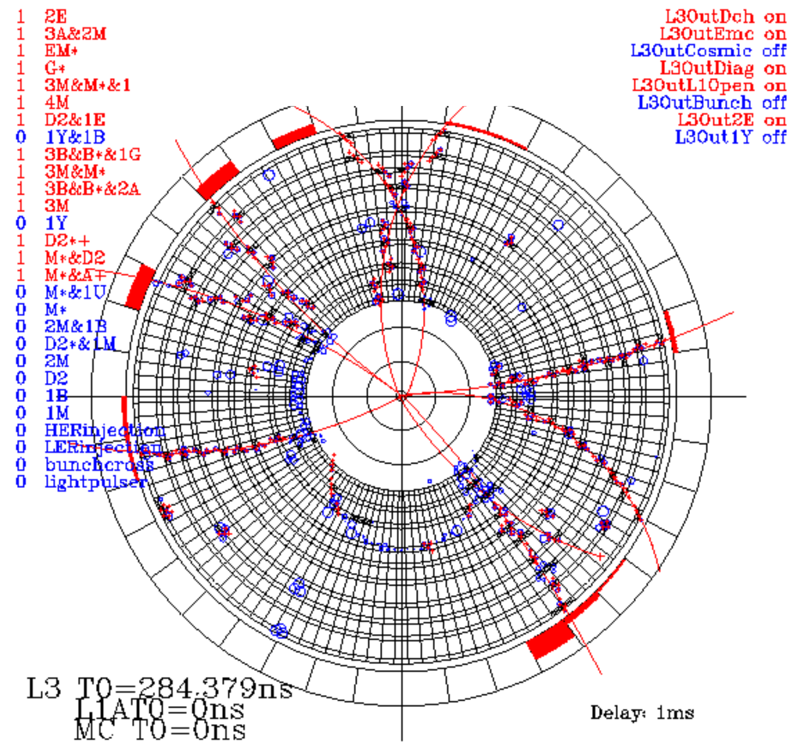


Fine position info for segments found for all superlayers

# Upon a L1 Accept …

## Level 3 Filter

***Using segments found by TSFs as seeds, fast algorithm applied to DCH data to further reject background... reduce the rate from 2 kHz to 100 hz.***

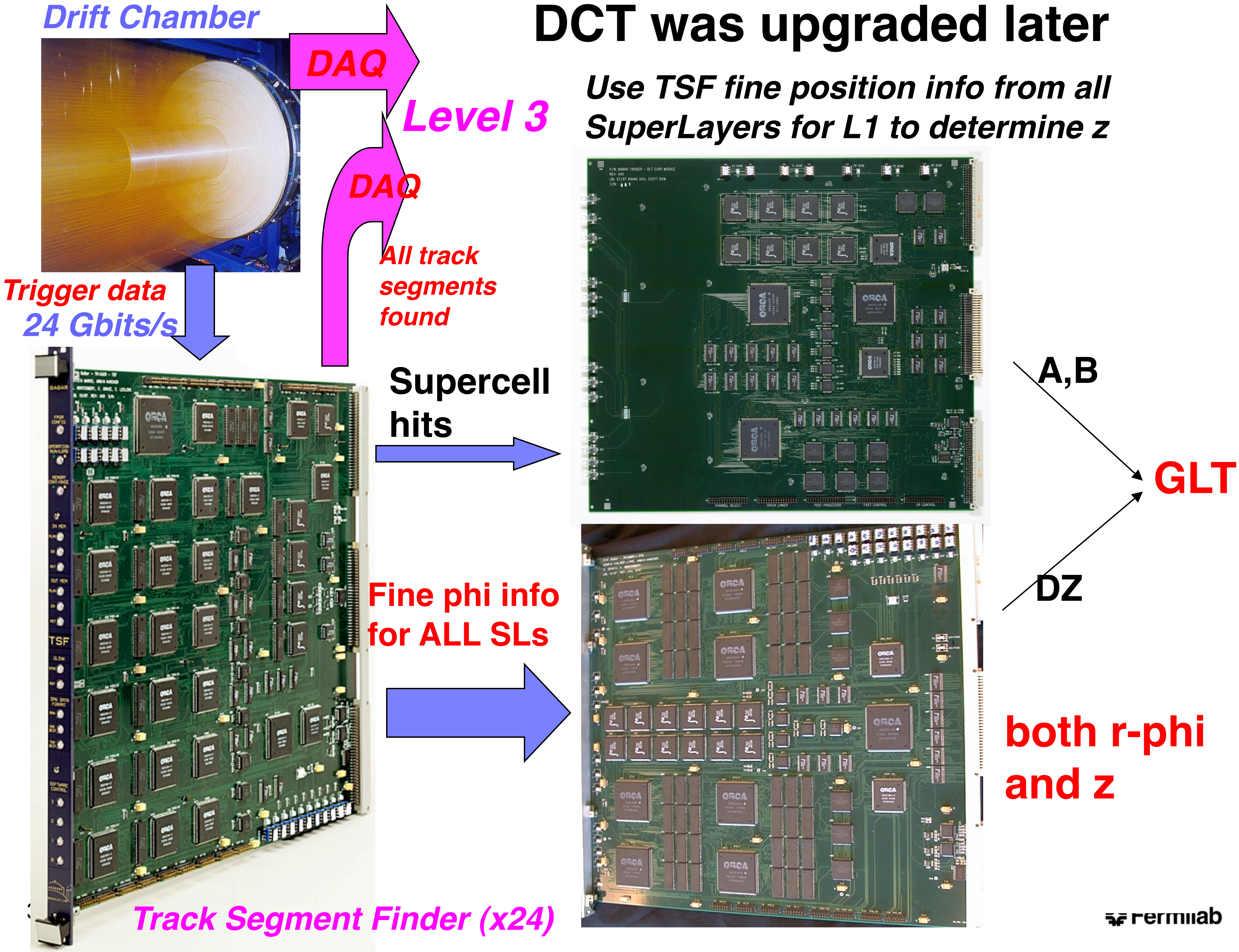



***original TSF provides fine position info for all SuperLayers to L3, only axial SLs info passed to PTD.***

Track Segment Finder (x24)

# DCT was upgraded later

*Use TSF fine position info from all SuperLayers for L1 to determine z*



# Rejecting Beam-Gas at Zeus and H1

Primary task is rejecting Beam-Gas background:

- Timing of TOF hits (H1) rejects out of time events
- Track processors reject events with large impact parameter in r-$\phi$ and r-z planes to remove beam-wall and beam-gas backgrounds
- Example: Look for patterns in r/z across layers:
  - Good tracks constant r/z
  - Tracks not from interaction region will have wrong pattern

Zeus Track Z0 Finding

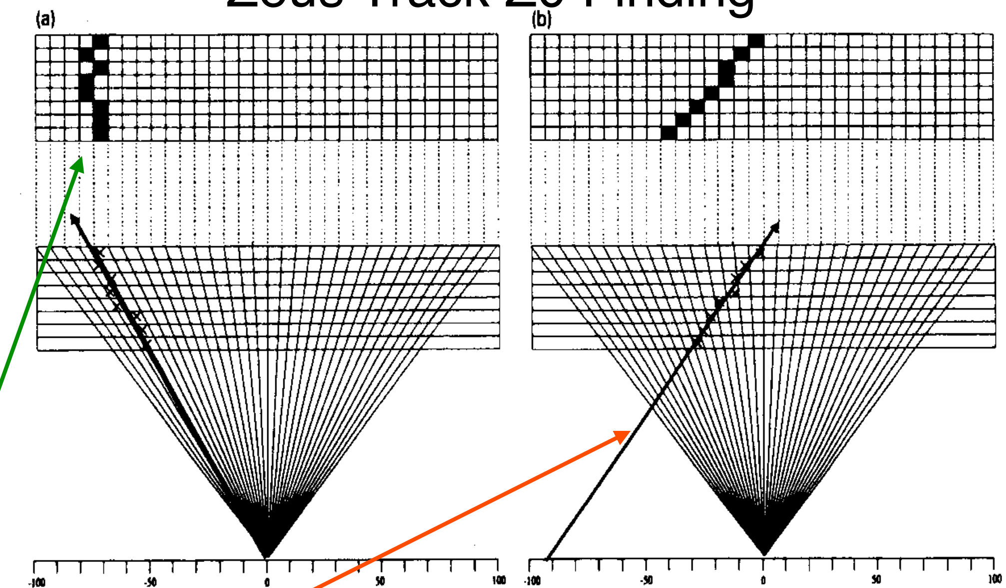
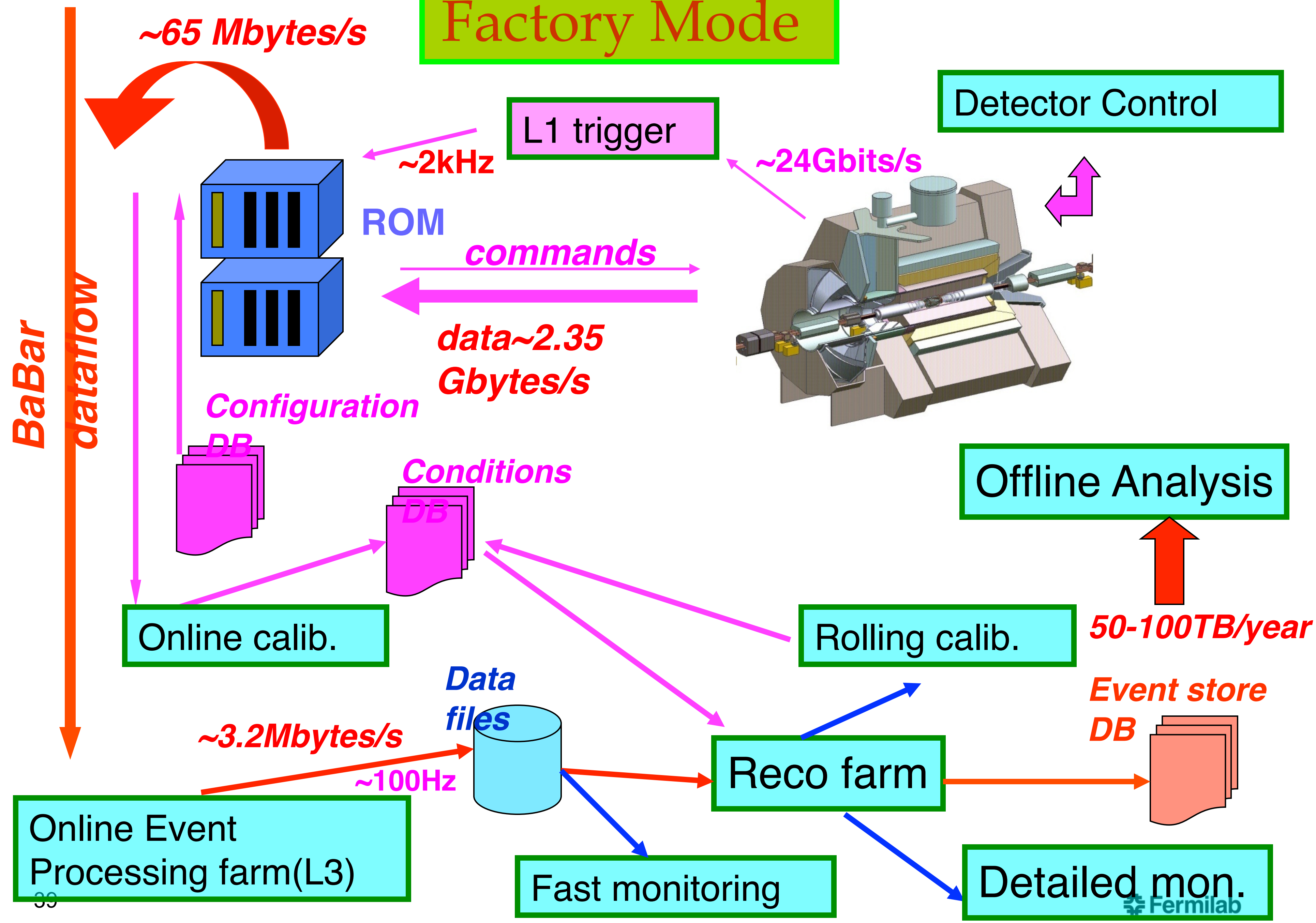



Fig. 2. The formation of the $z/r$ table in the segment finding processor for (a) a track from the vertex region and (b) a track from outside the vertex region.

GP Heath etal, NIMA 315(1992) 431.

ep collider case

# Factory Mode



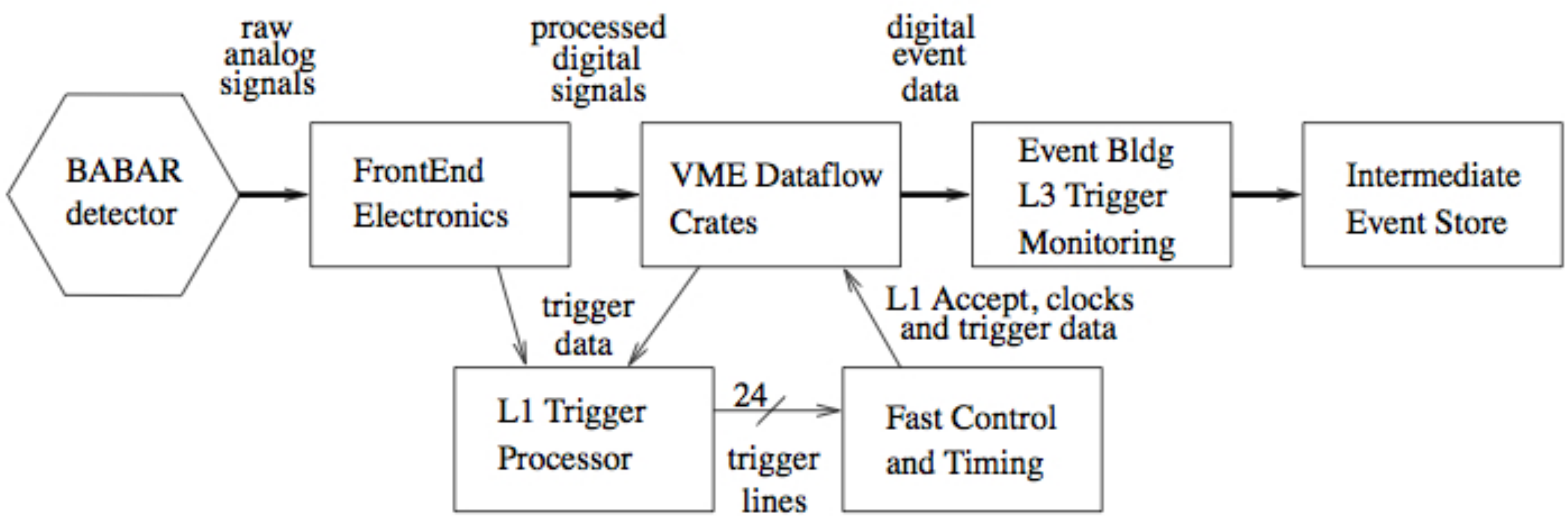
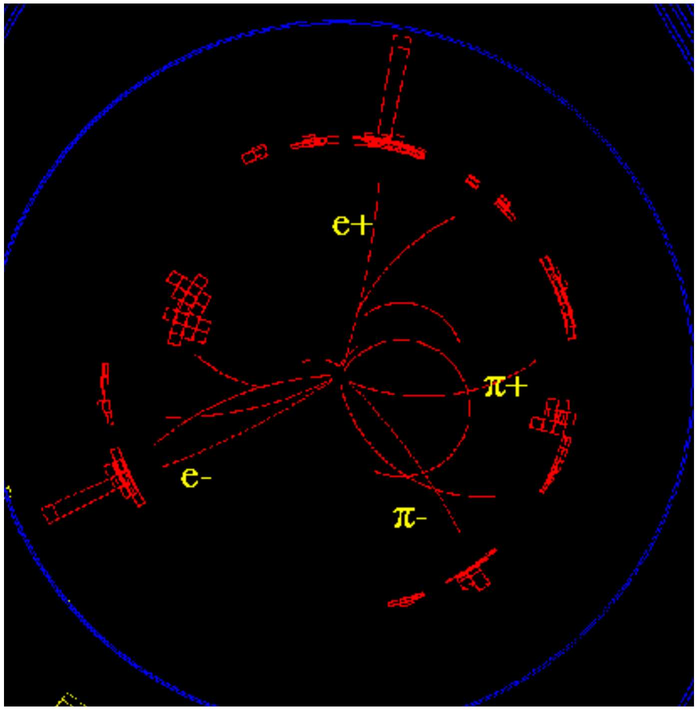
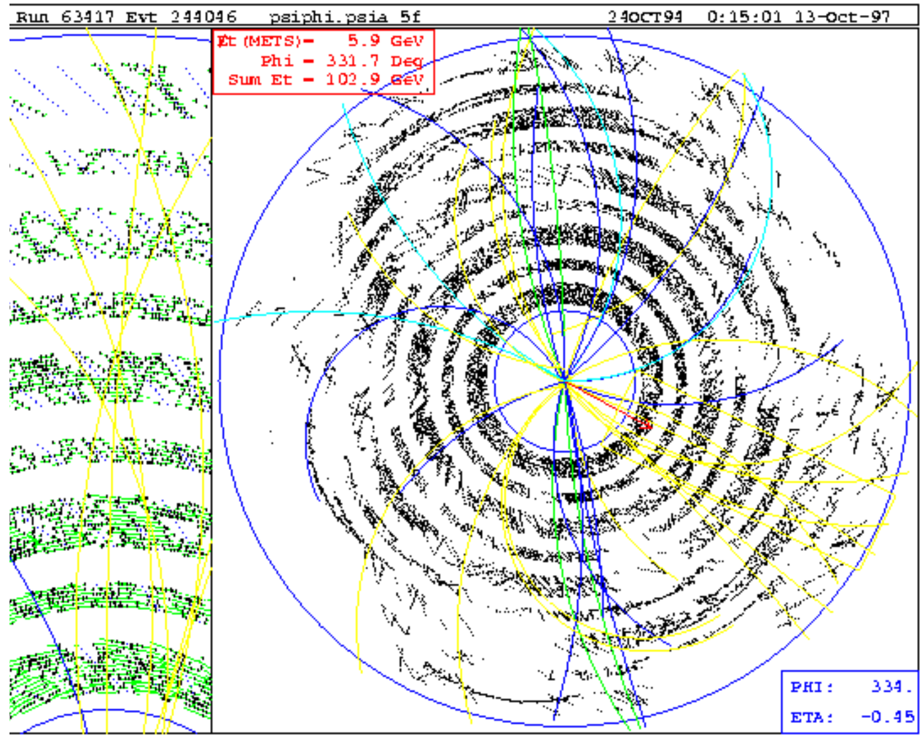



Figure 5. Schematic diagram of the data acquisition.

## $e^+e^- \longrightarrow \Upsilon(4S)$

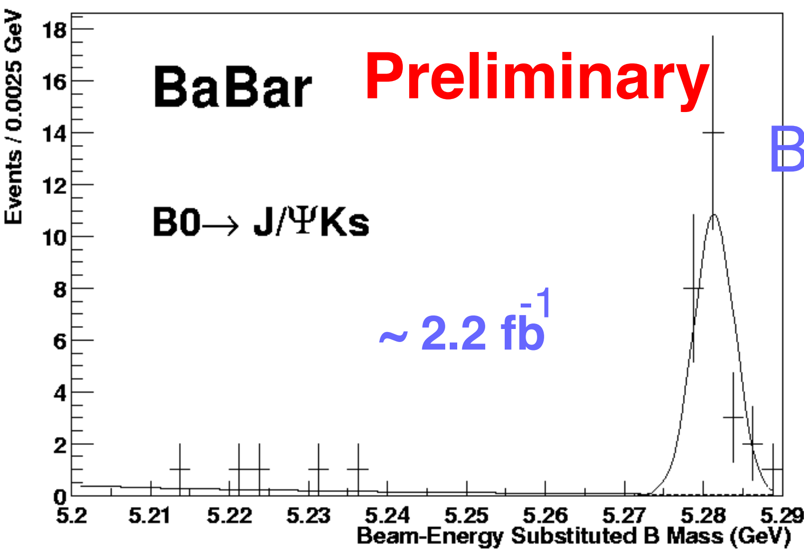
## Hadron collider

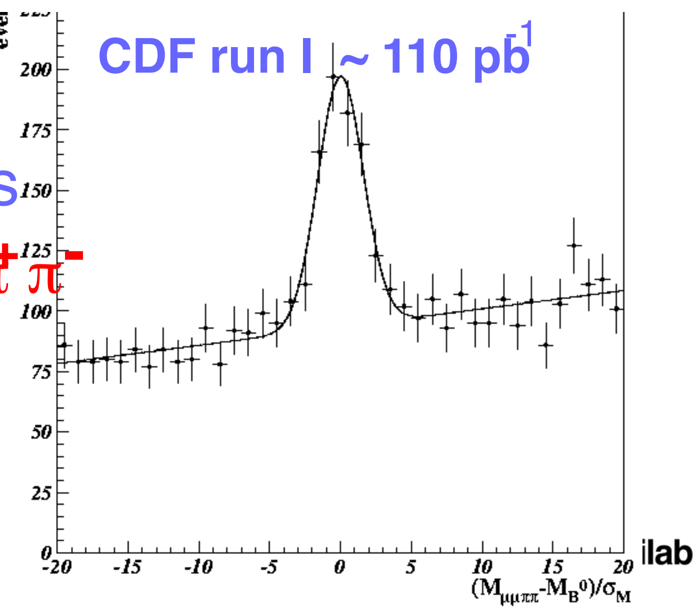


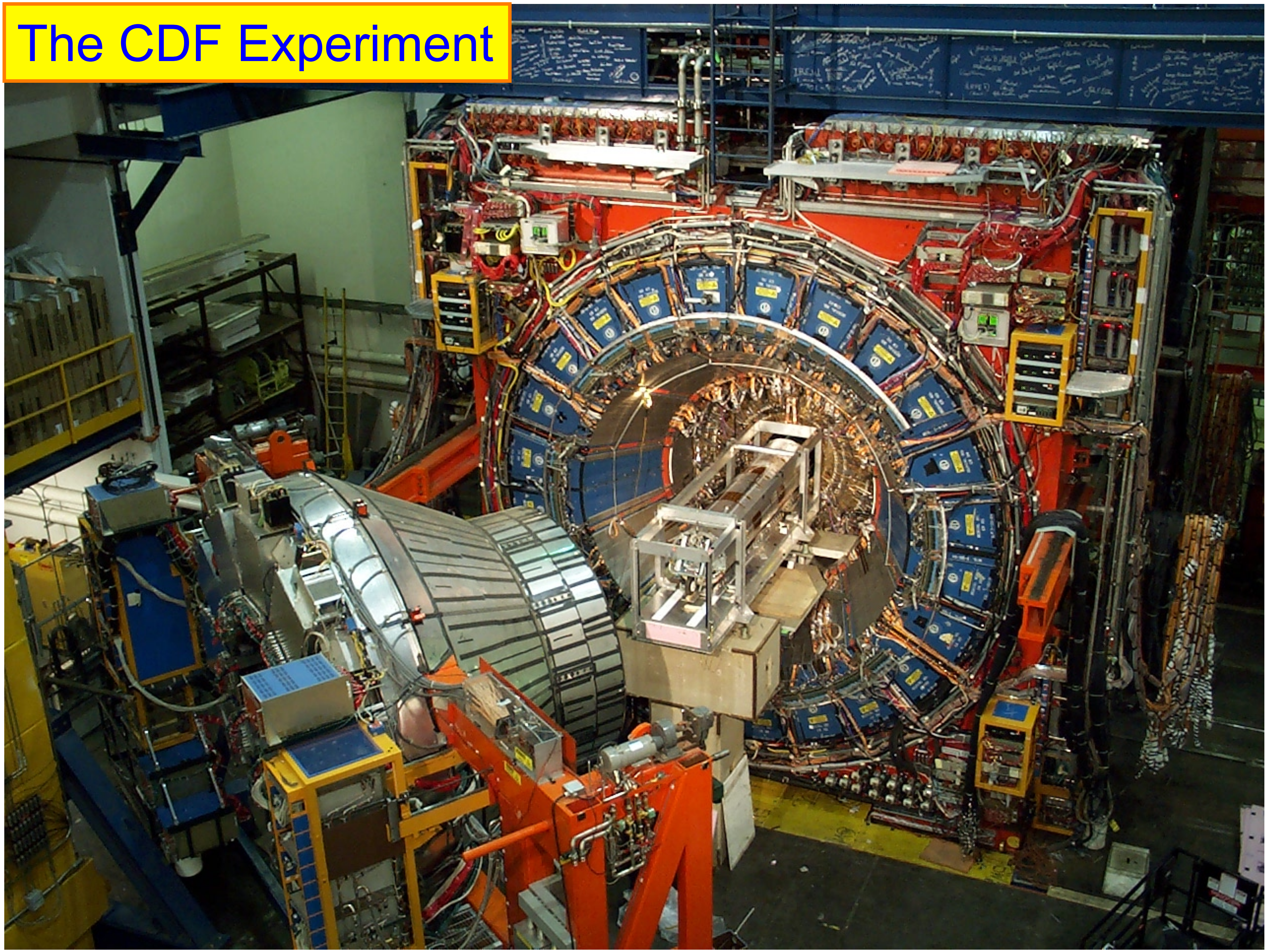

$B^0 \longrightarrow J/\Psi \; Ks$

$\quad \llcorner\!\to \mu^+\mu^- \quad \llcorner\!\to \pi^+\pi^-$

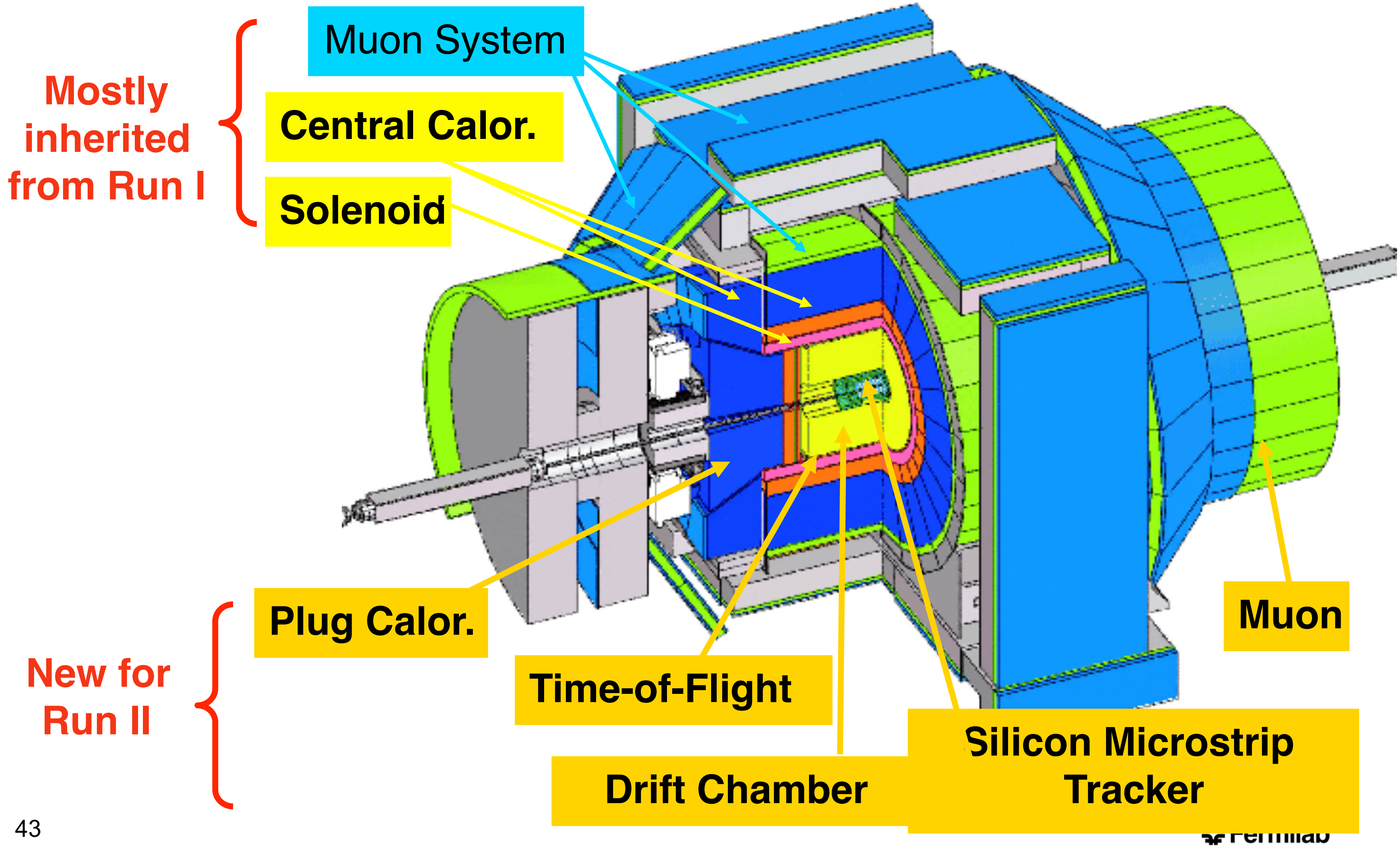


# The CDF Experiment

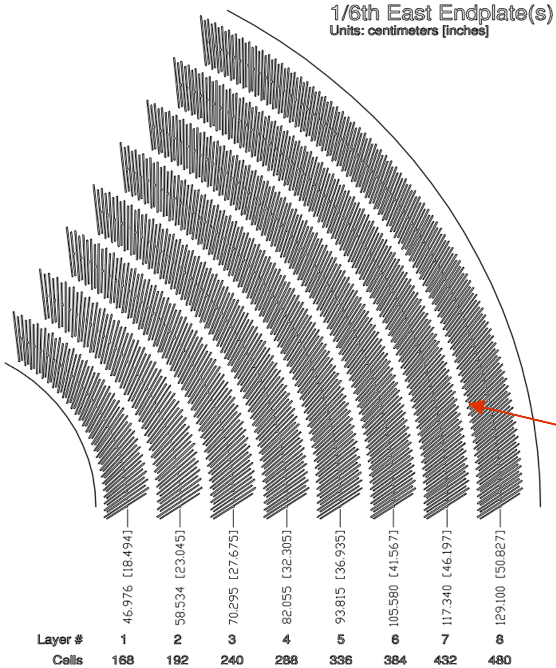


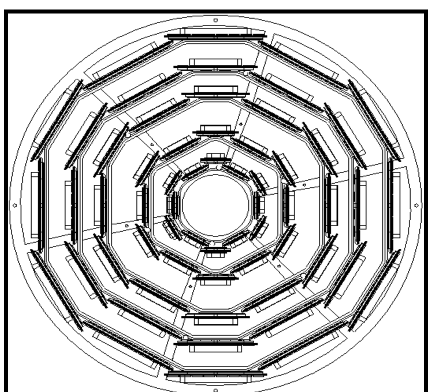
# The CDF detector



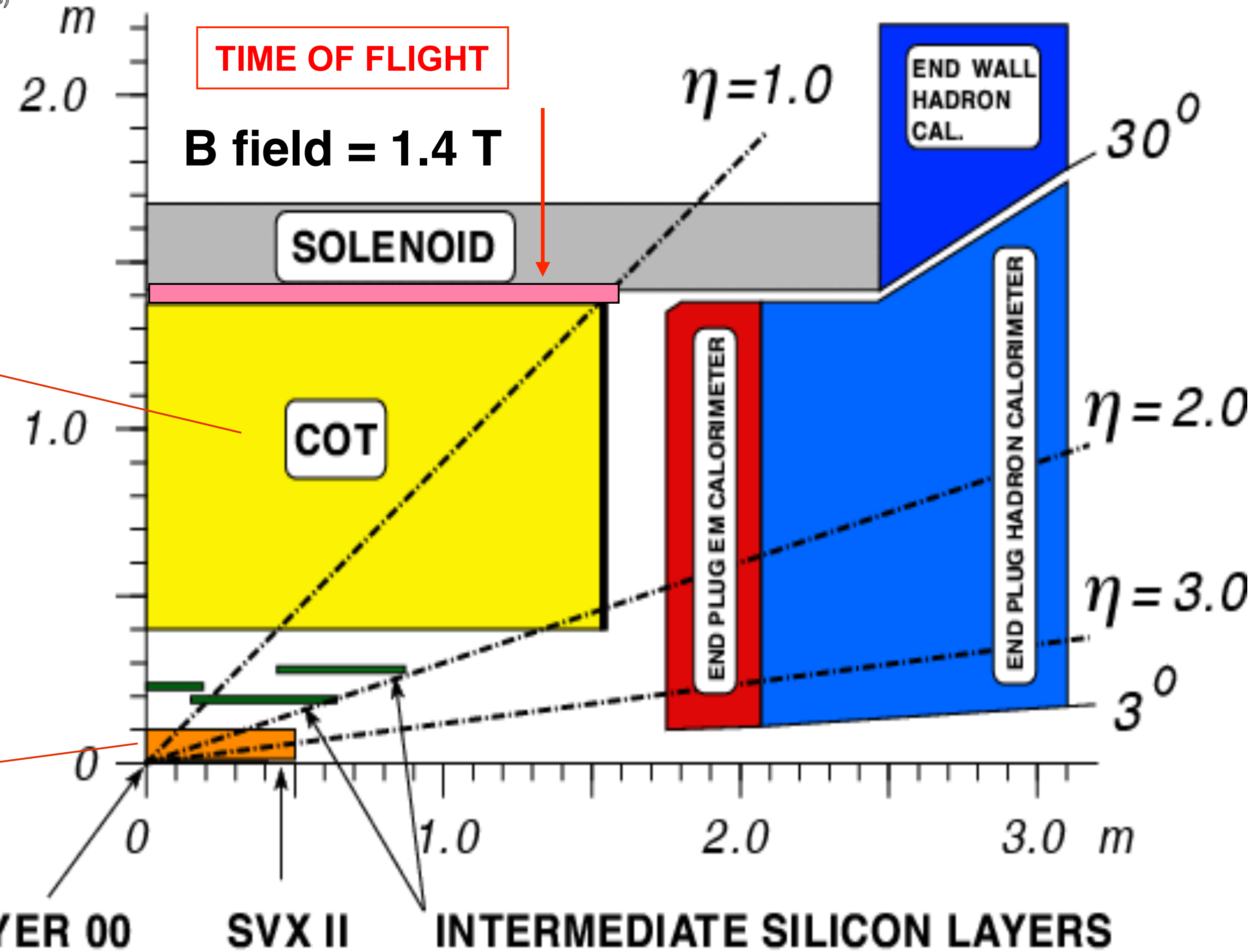
# The CDF Tracker

## Transverse view

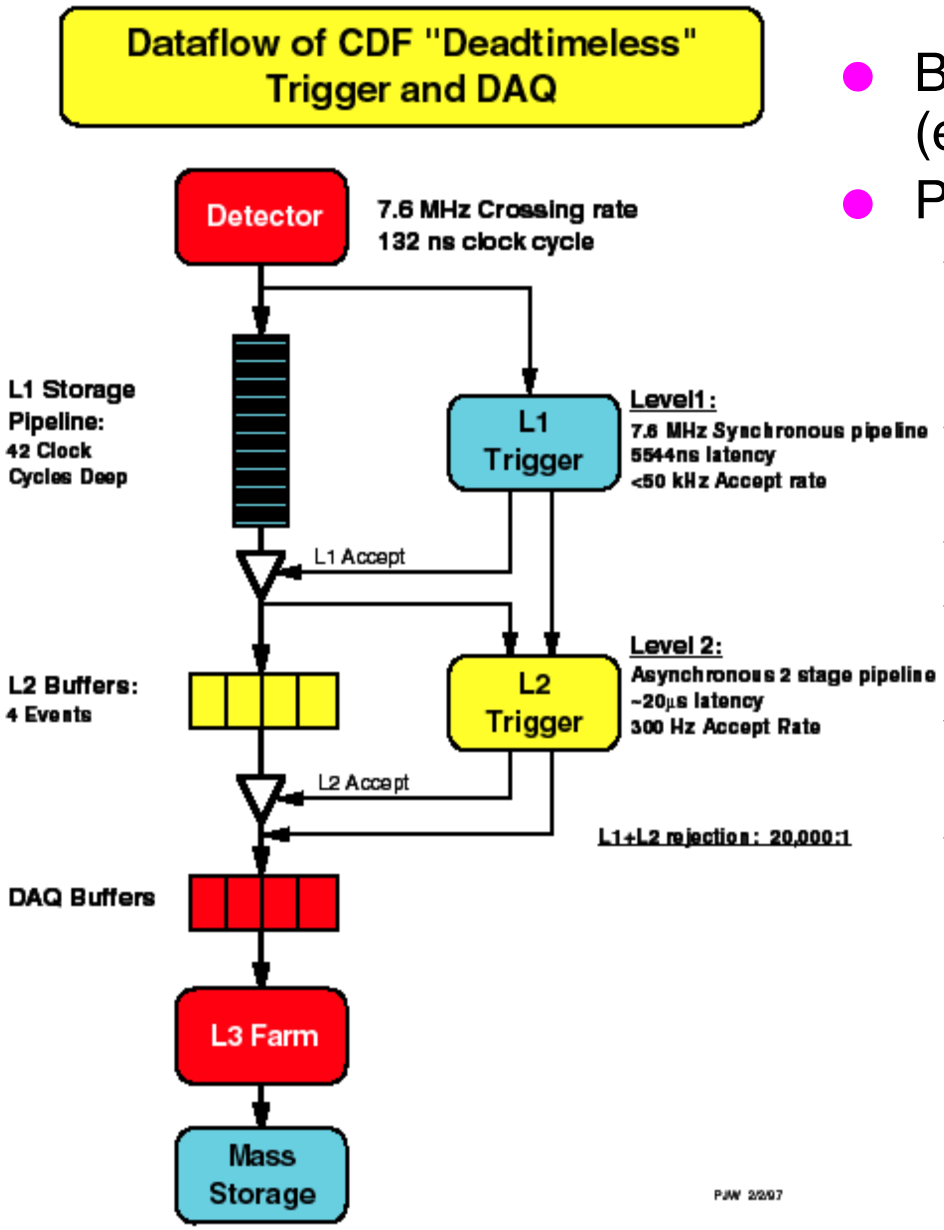


## Longitudinal view



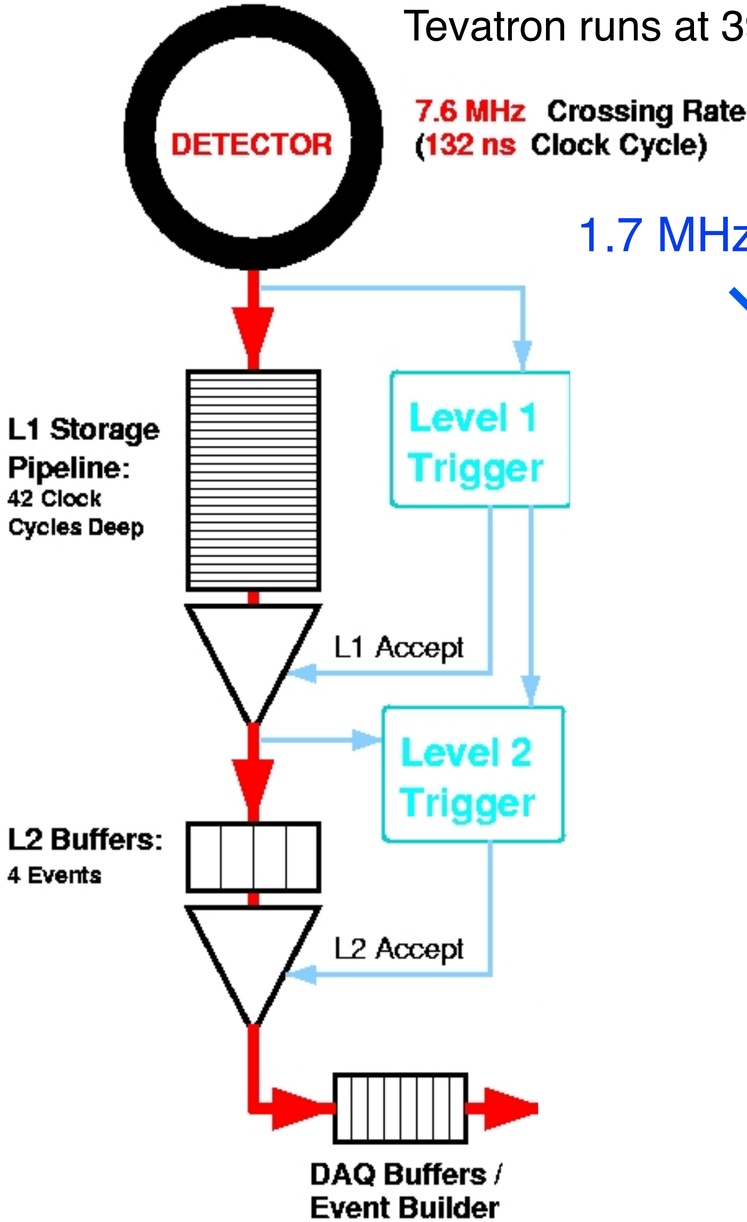
# CDF/D0 Pipelined Trigger/DAQ

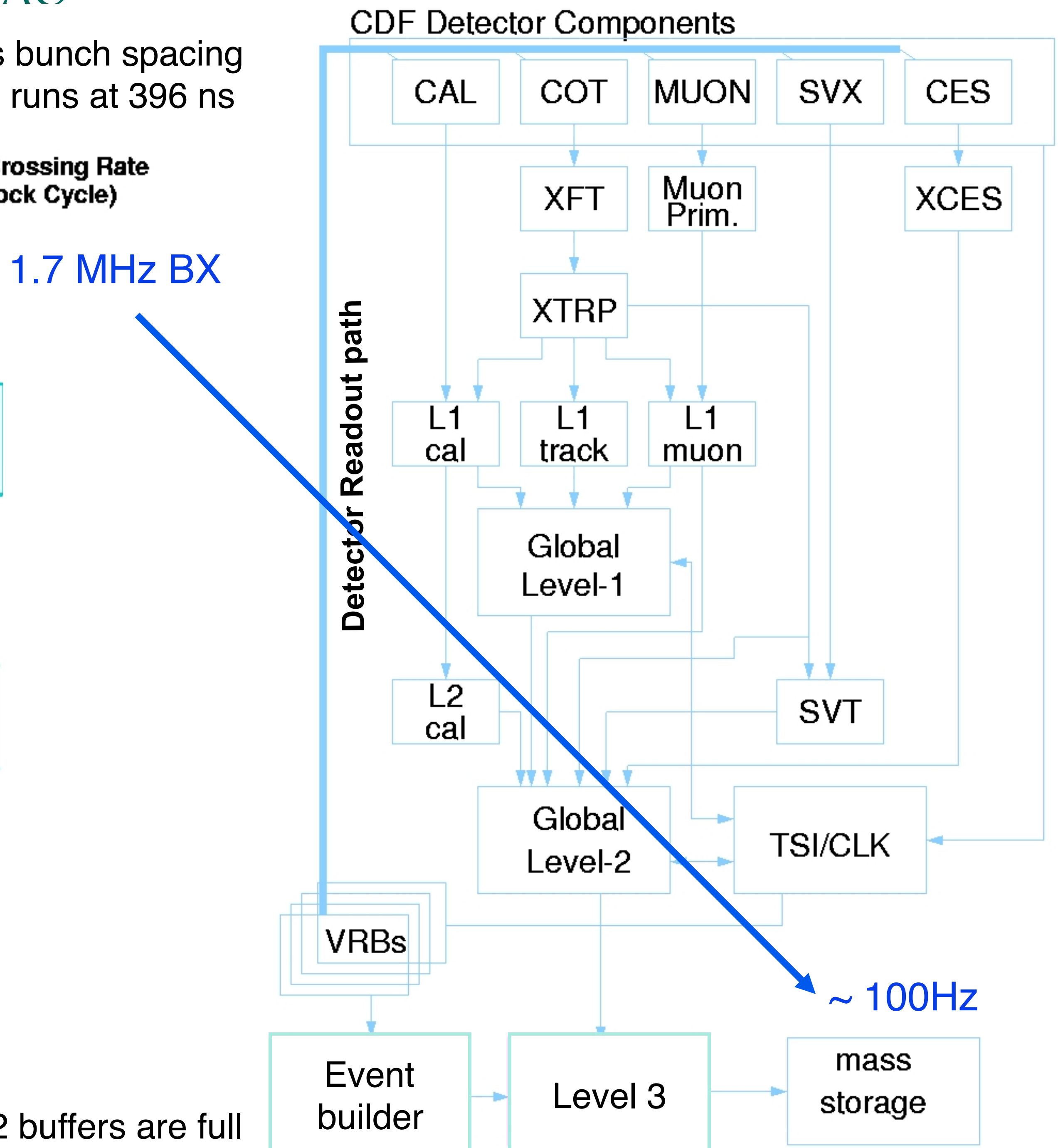



- Beam x-ing always multiple of 132ns (either 132 or 396)
- Pipelined DAQ+Trigger:
  - Every front-end system stores data for at least 42 clock cycles during L1 decision
  - All L1 trigger processing in parallel pipelined operation
  - L1 decision is made every 132 nsec
  - On L1 accept, data is moved from L1 Pipeline and stored in L2 buffers
  - On L1 reject data is dropped off end of pipeline
  - On L2 accept data is read into VME Readout Buffer (VRB) modules awaiting readout via switch to L3

# CDF RunII Trigger/DAQ

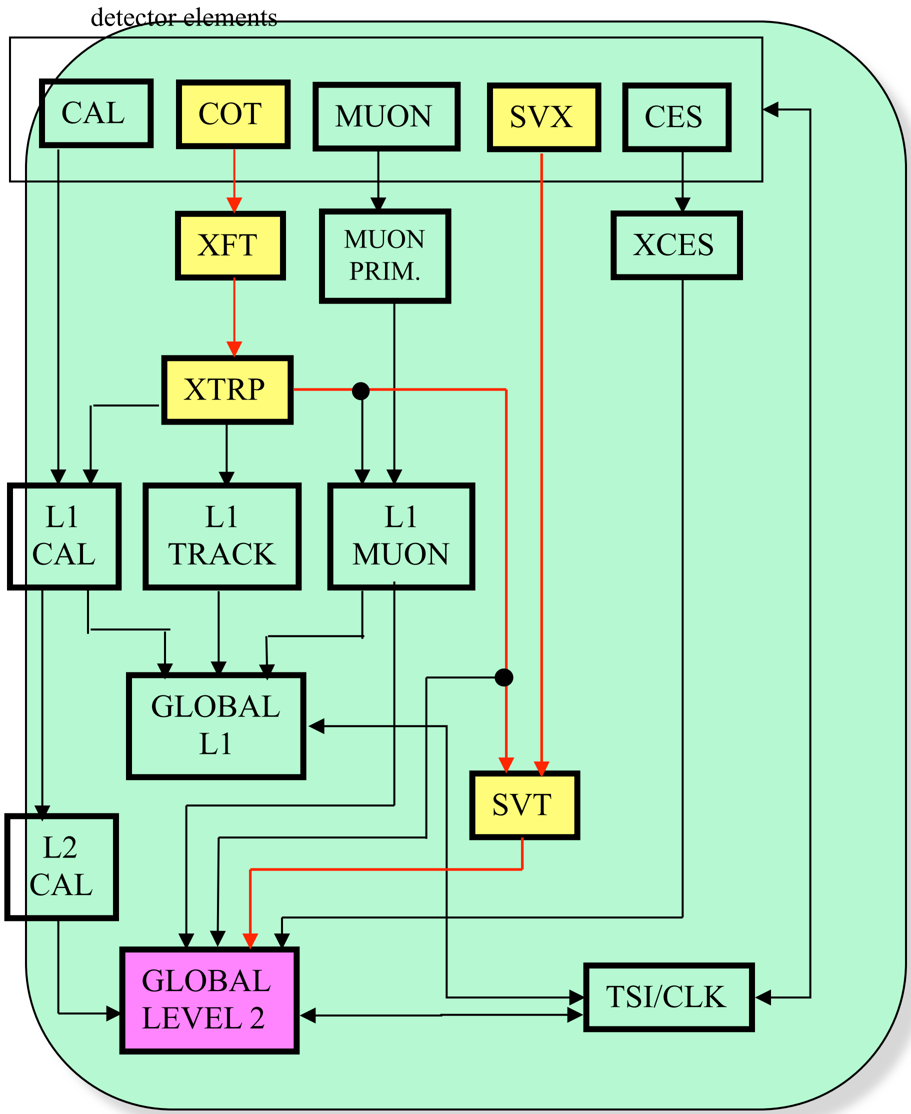
Designed for 132 ns bunch spacing
Tevatron runs at 396 ns



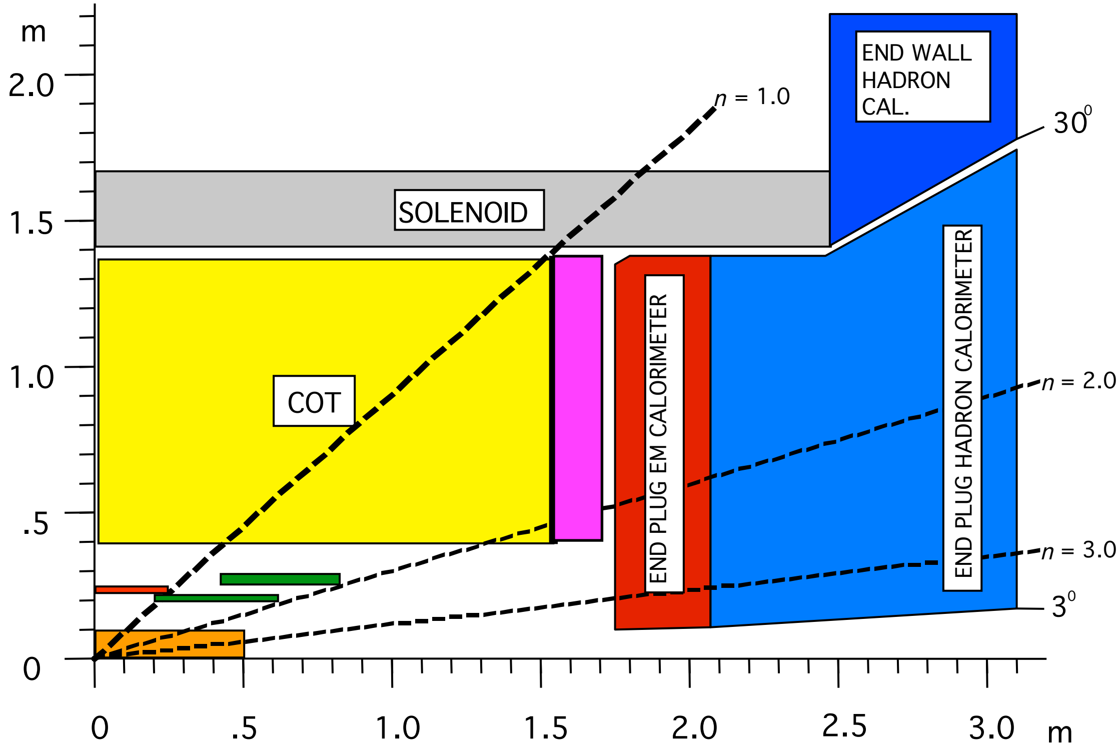



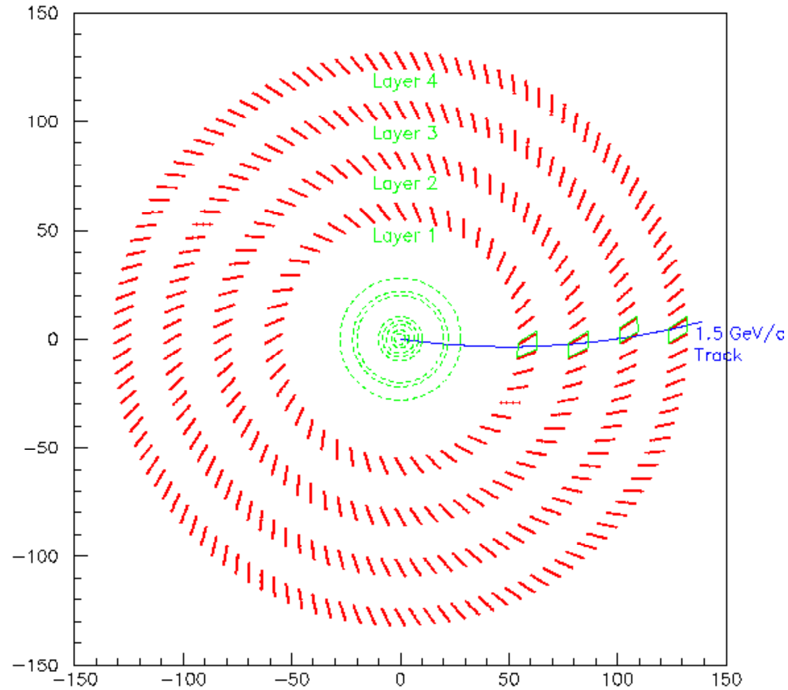

Dead-time occurs when all L2 buffers are full

# CDF Tracking Trigger in Run II



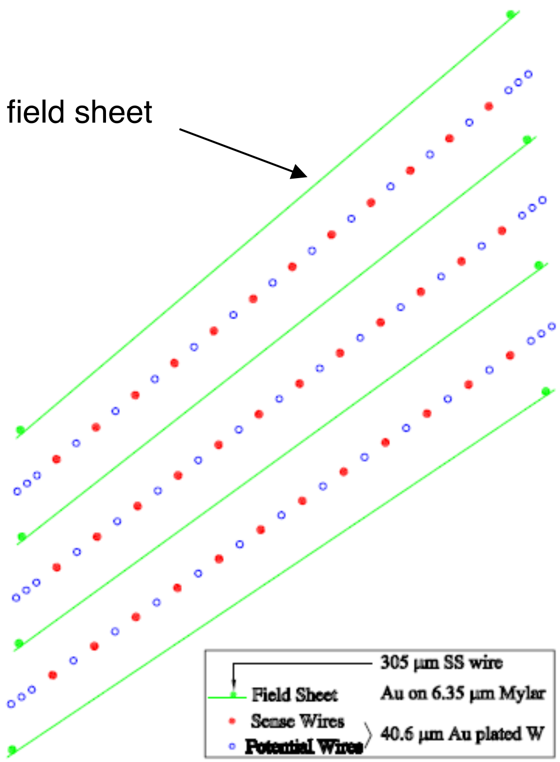
# CDF Central Outer Tracker



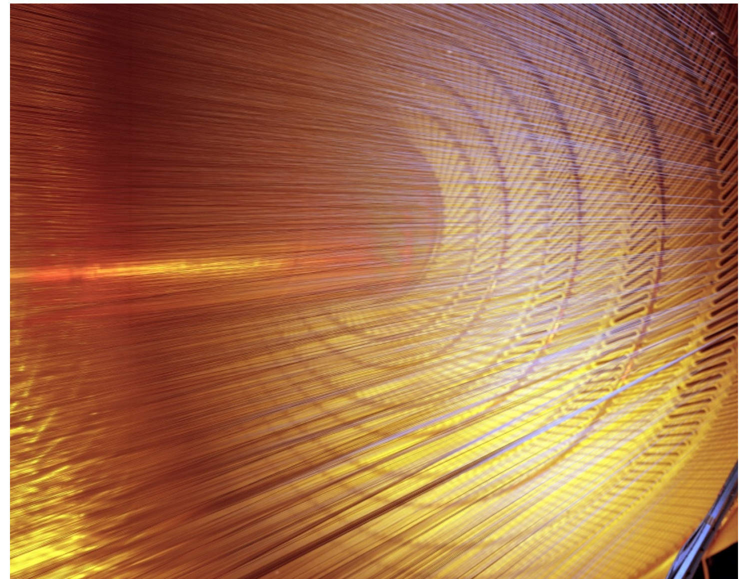



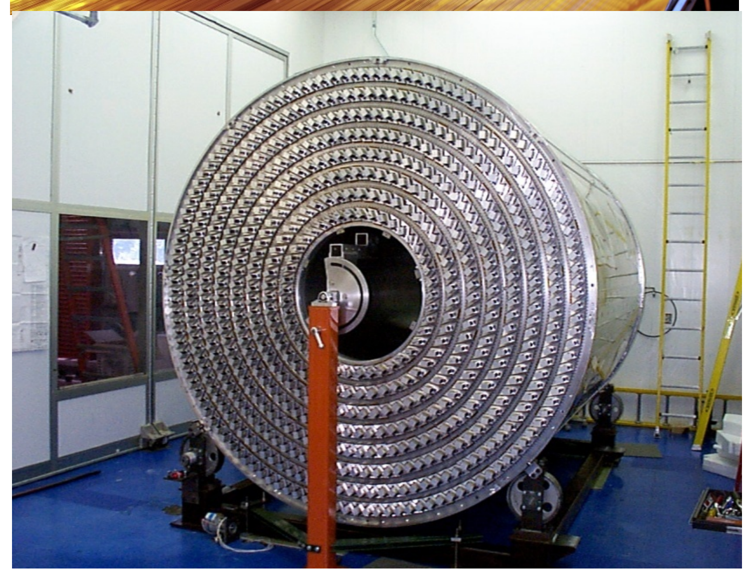

- Small drift cells, ~ 2 cm wide, a factor of 4 smaller than in the Run I tracker
- Fast gas, drift times short ~130 ns
- COT cell has 12 sense wires oriented in a plane, at ~ 35° with respect to radial direction for Lorentz drift
- A group of such cells at given radius forms a superlayer (SL)
- 8 alternating superlayers of 4 stereo (~3°) and 4 axial wire planes

# CDF COT




- narrow drift cells insure short collection times: trigger input





typical resolutions:

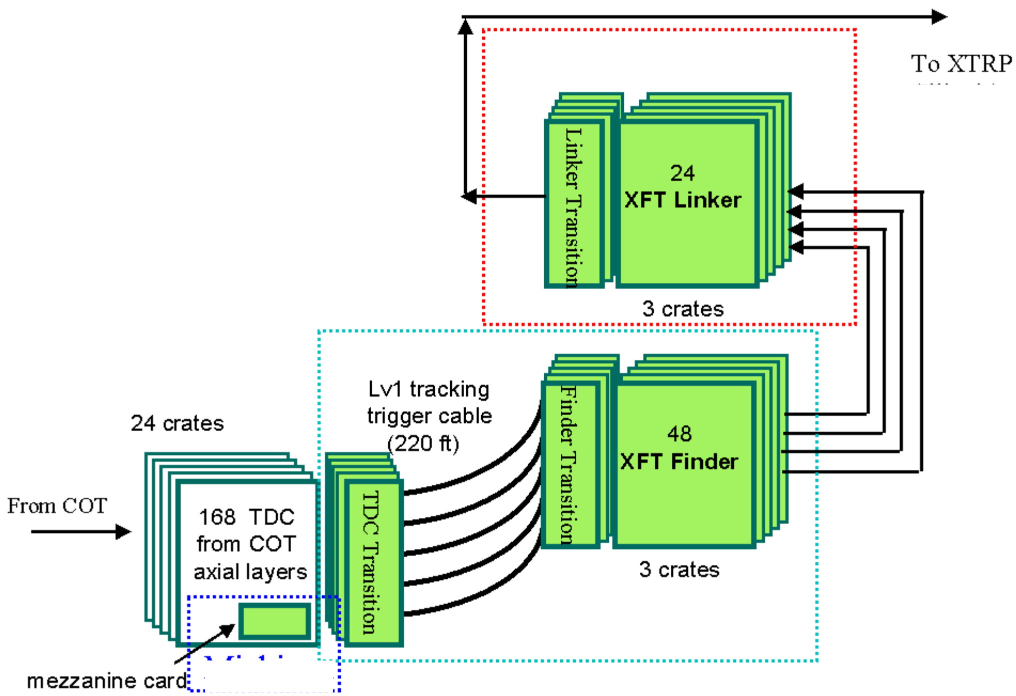
$\sigma_{xy}$ ~100μm

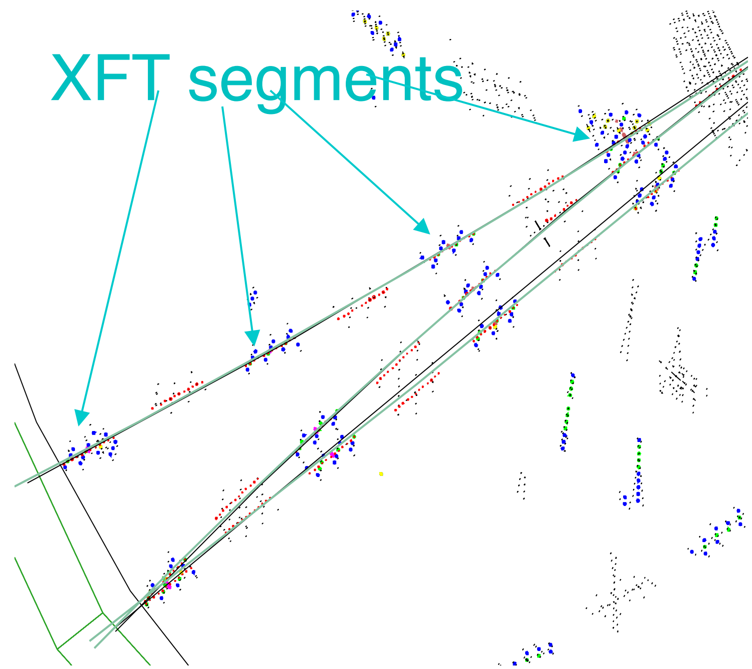
$\sigma_z$~<1mm

# CDF L1 Track trigger: XFT

- XFT uses COT hits from four axial SLs to reconstruct tracks

- Track finding is done in stages:
  (1) Digitize raw COT hits
  (2) Form track segments from the digitized hits
  (3) Link segments into track

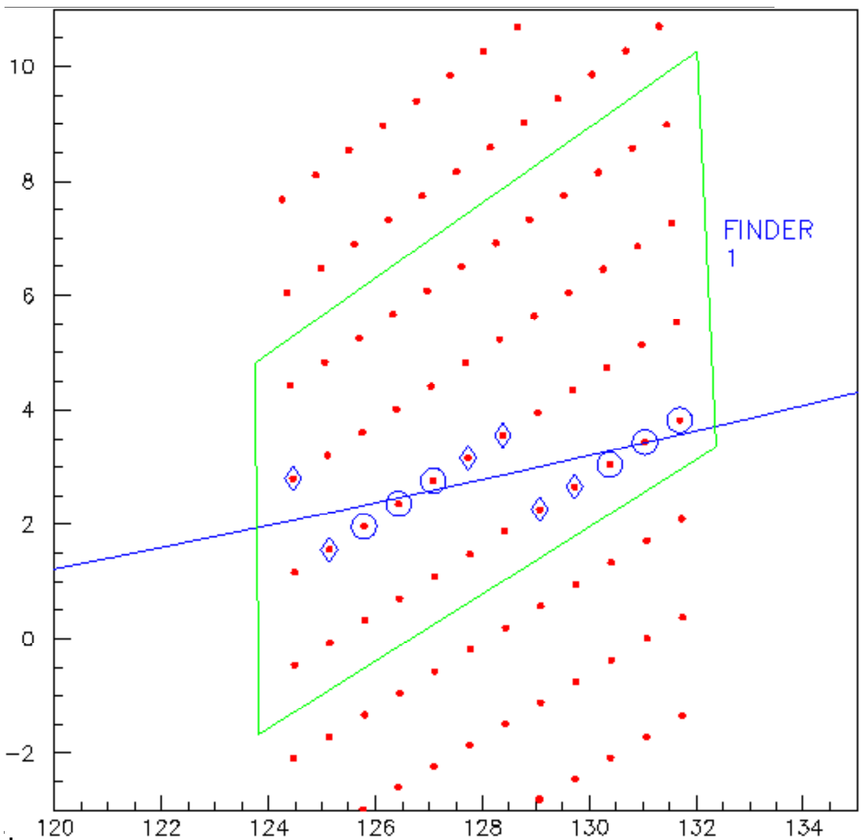



Fermilab

# XFT Hit Digitization and Segment Finding

- Each raw COT hit is classified as either prompt or delay hits which provide crude (two bin) timing information on the hits (now the bunch crossing time is known)
- The hits are used to find segment in a given SL.

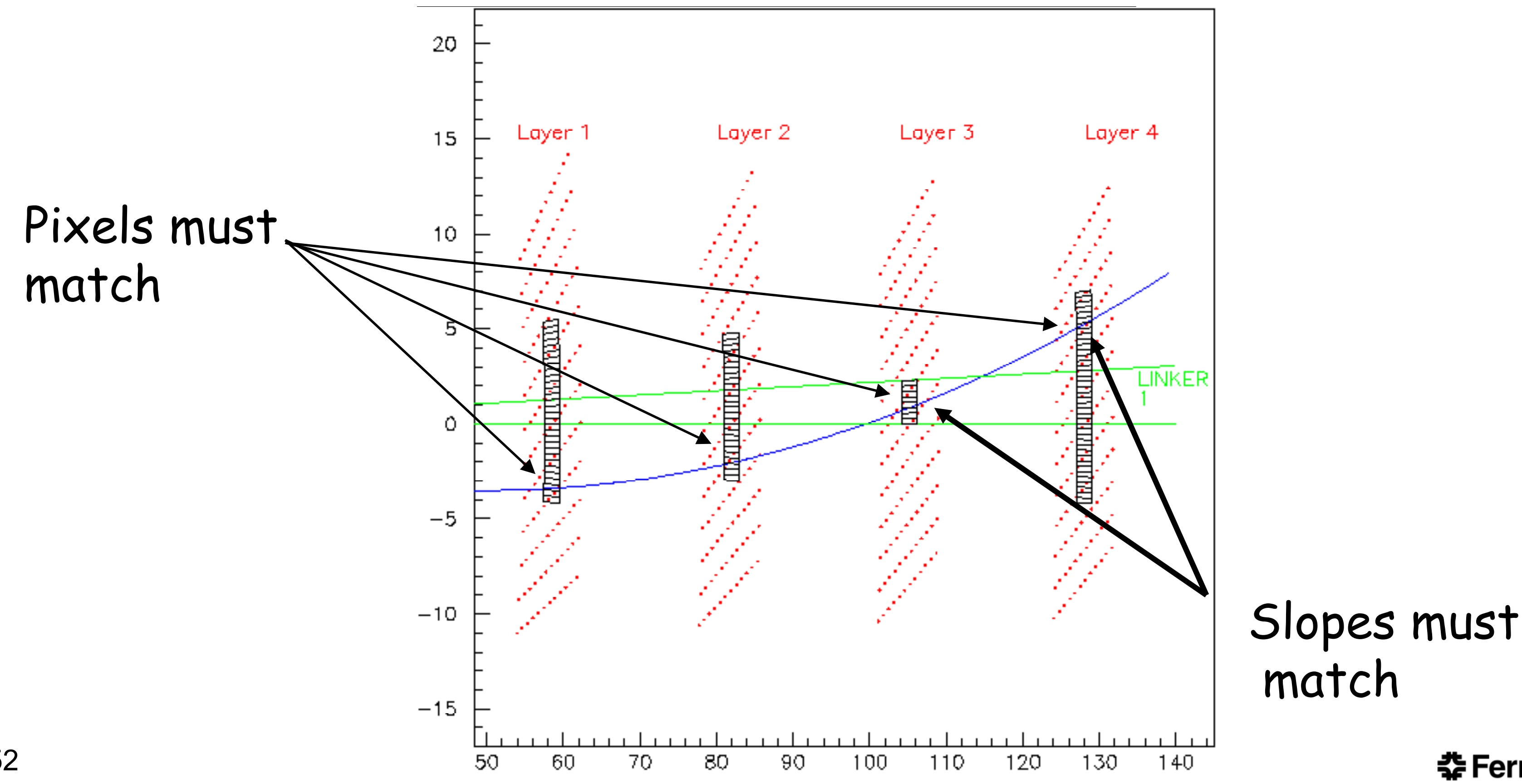



- When a segment is found, reports the position of the segment
- For the outer two layers, also reports the slope (low pT+, low pT-, high pT)

# Track Linking

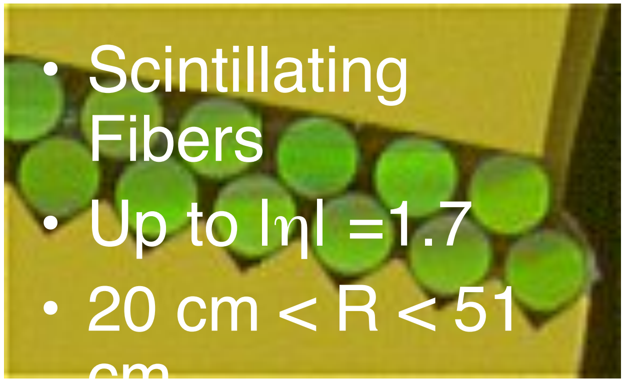
- compares the pixels in all four layers to a list of valid pixel patterns (or "roads") to find tracks.

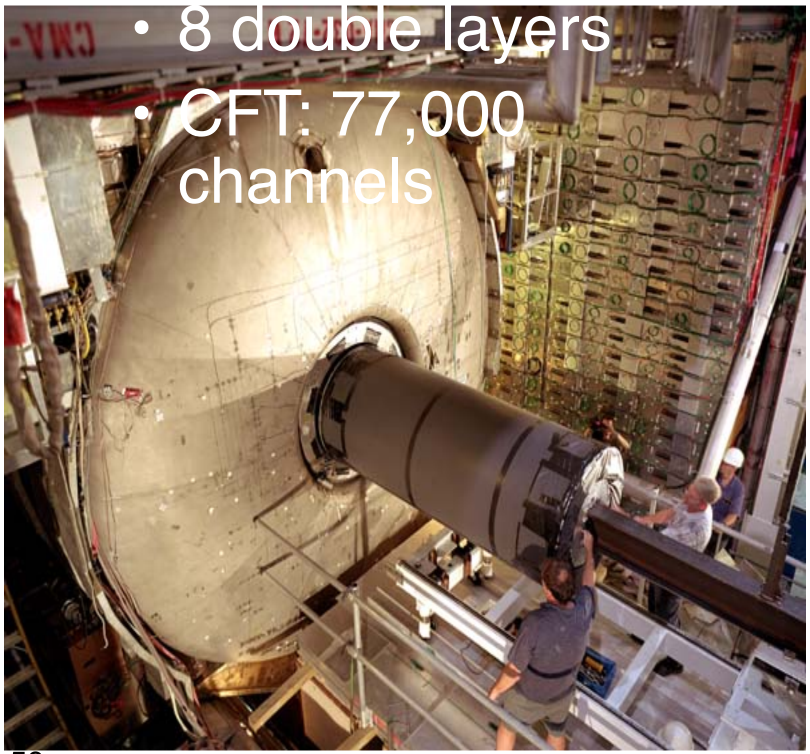
- A valid track is required to match pixels locations as well as the slope of the segments at the outer two SLs.



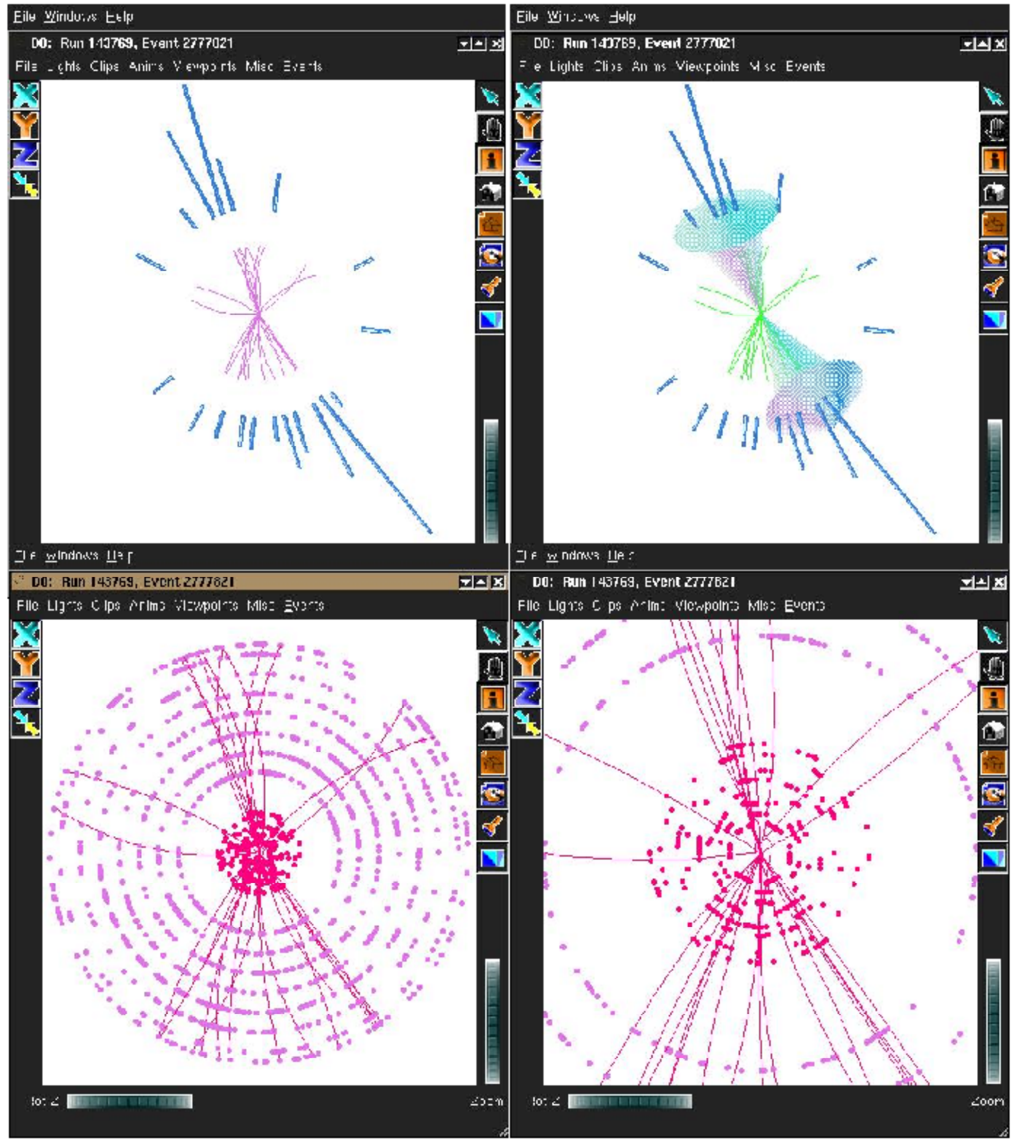
# The DØ Central Fiber Tracker

- Scintillating Fibers
- Up to |η| =1.7
- 20 cm < R < 51 cm
- 8 double layers
- CFT: 77,000 channels



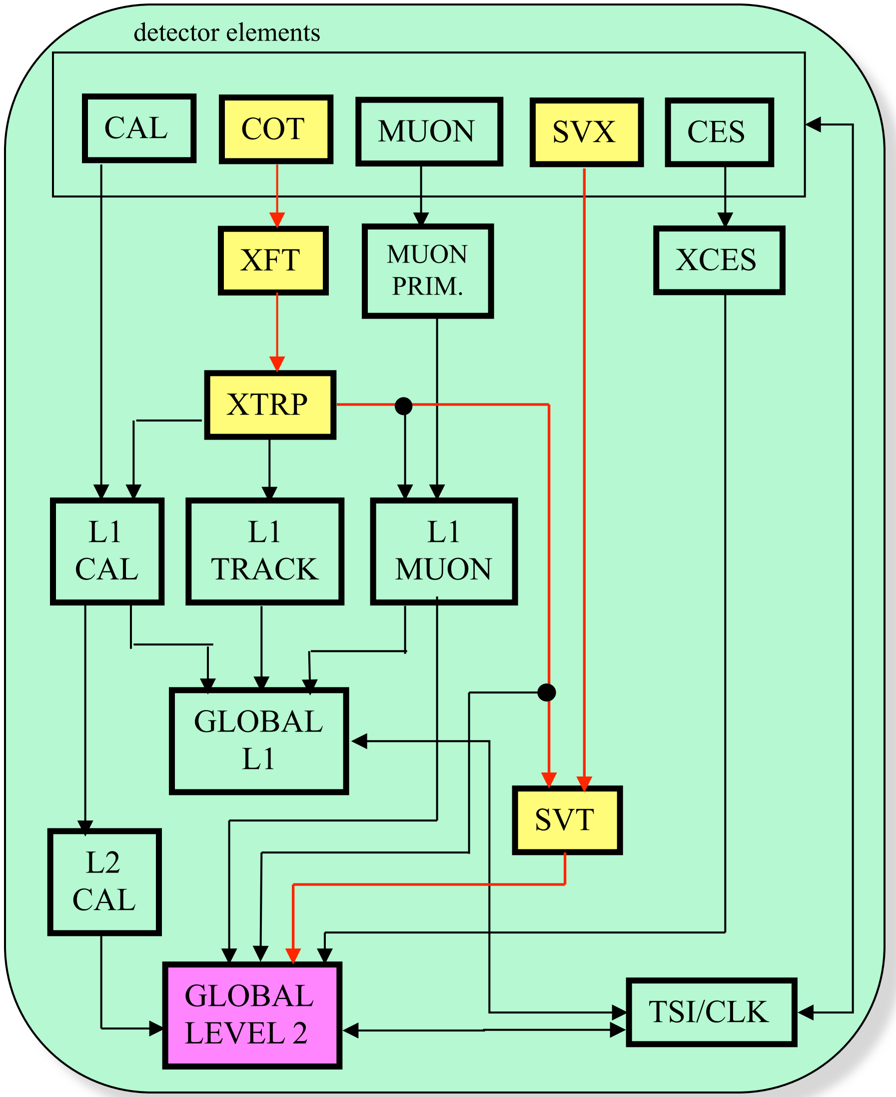


Zoom in to run 143769, event # 2777821

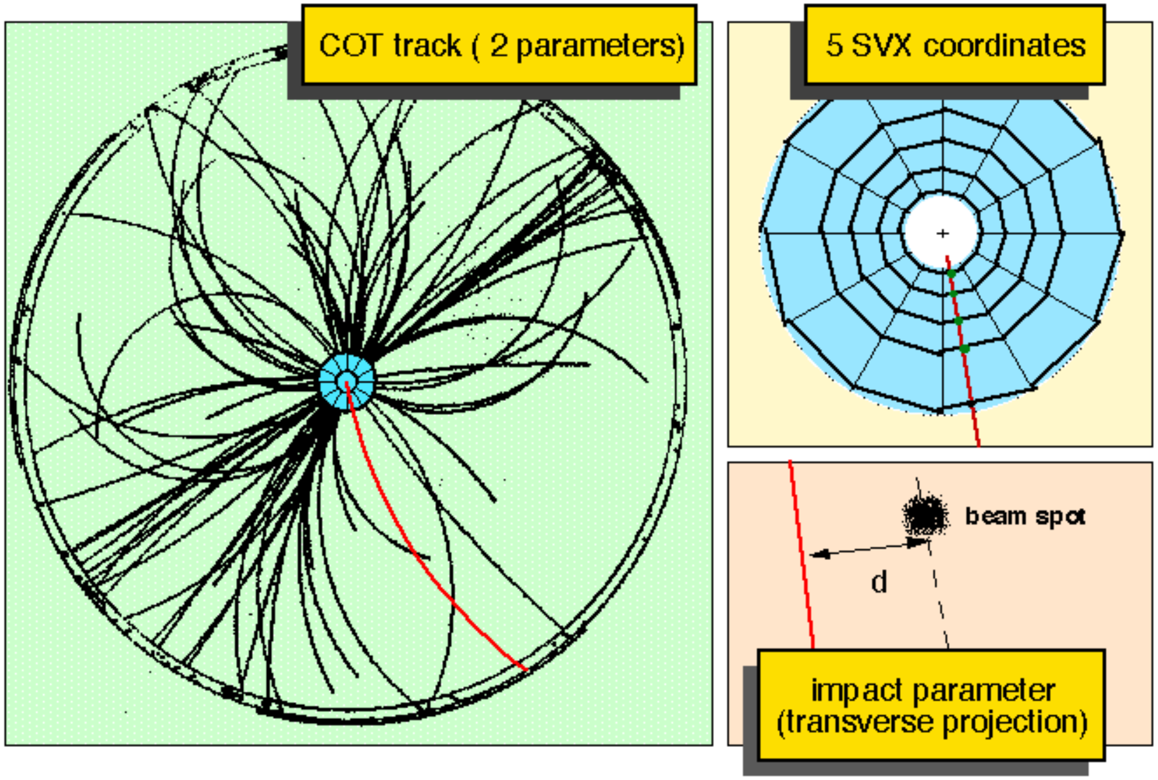


L1 Fiber tracker Tracking trigger done using track equations

# CDF Tracking Trigger in Run II



# SVT: Input & Output




**Inputs:**

- **L1 tracks from XFT ($\phi$, $p_T$)**
- **digitized pulse heights from SVX II**

**Functionalities:**

- **hit cluster finding**
- **pattern recognition**
- **track fitting**

**Outputs:**

- **reconstructed tracks ($d$, $\phi$, $p_T$)**

Online tracking with offline quality

# $b\bar{b}$ cross section comparison

| | $\sigma(b\bar{b})$ | $b\bar{b}$/All |
|---|---|---|
| LHC-B ( pp ) | ~ 500 μb | ~0.002 |
| Tevatron ( $p\bar{p}$ ) | ~ 50 μb | ~0.001 |
| LEP(e+e- -> Z->$\bar{b}$b) | ~ 6 nb | ~0.22 |
| e+e- -> Υ(4S) -> BB | ~ 1 nb | ~0.25 |

Impact para from b/c decays:
~ 100 um independent from Pt
Typical offline d resolution
From silicon: 10-50 um

Trigger: online resolution~40um
Fast enough

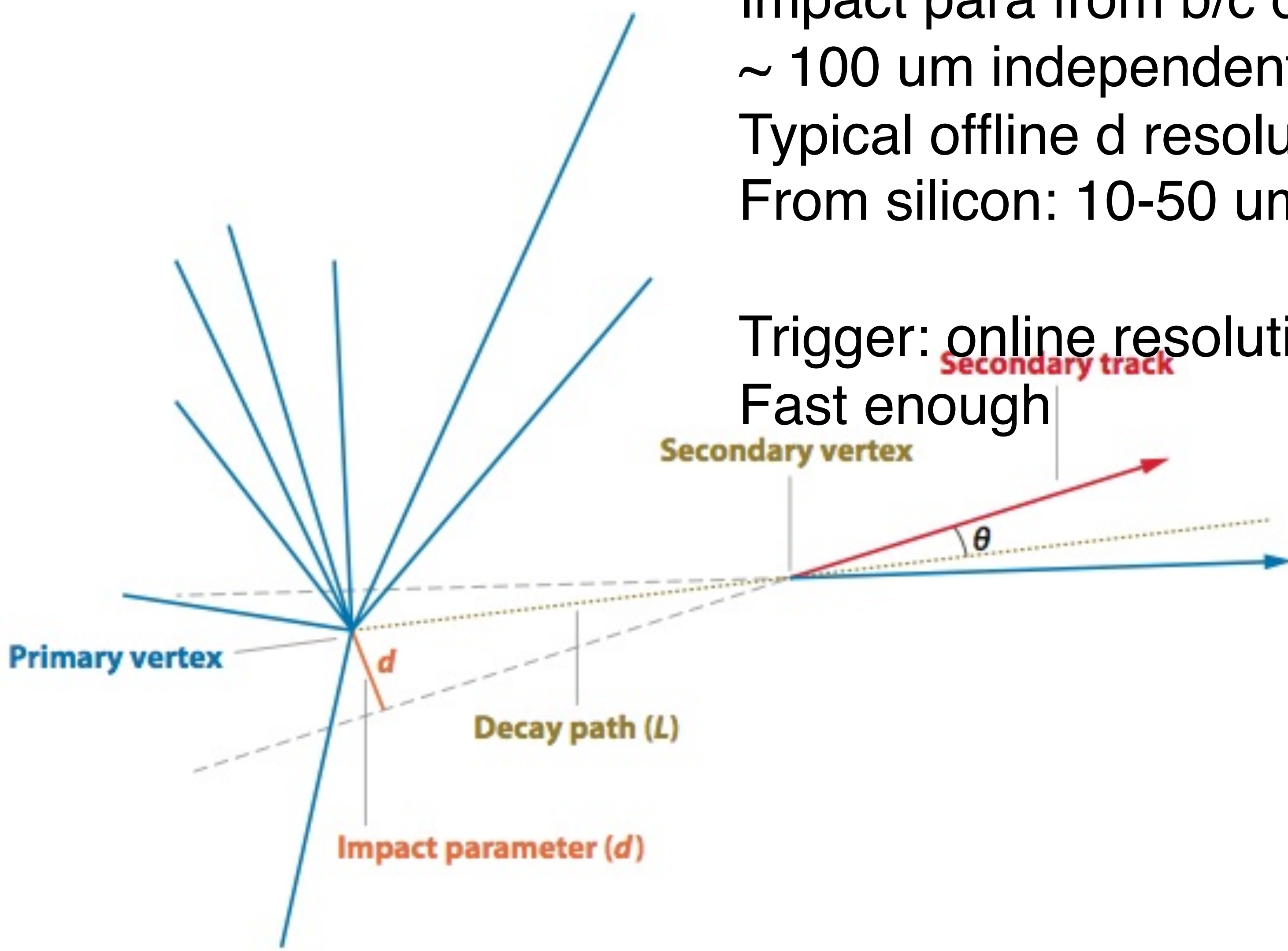
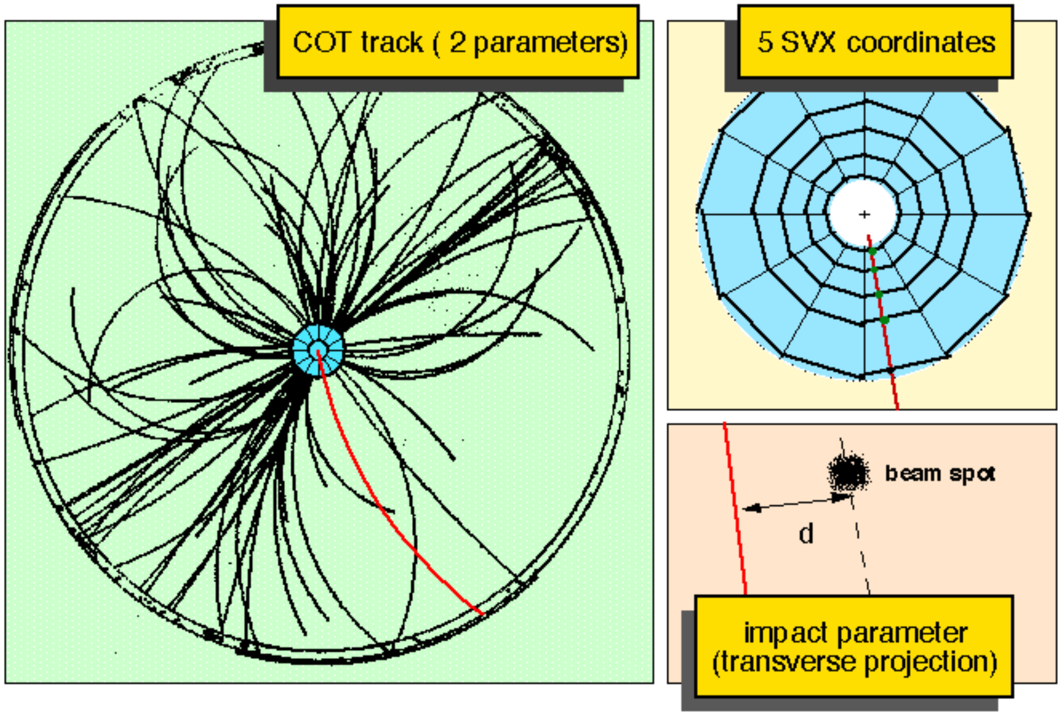



**Figure 1**

When a long-lived particle decays after traveling some distance, the trajectories of the decay products do not point back to the collision point. The distance of closest approach of the extrapolated trajectory to the collision point is known as the impact parameter.

# SVT: Silicon Vertex Tracker




**SVX II geometry :**

- **12 φ-slices (30°each) "wedges"**
- **6 modules in z ("semi-barrels")**

➡ Reflected in SVT architecture

## SVT inputs:

- COT tracks from Level 1 **XFT** ($\phi$, $Pt$)
  $\sigma(q/P_T)$=1.7%/GeV, $\sigma(\phi)$=5mrad
- Digitized pulse height in **SVXII** strips

⬇

Performs tracking in a two-stage process:

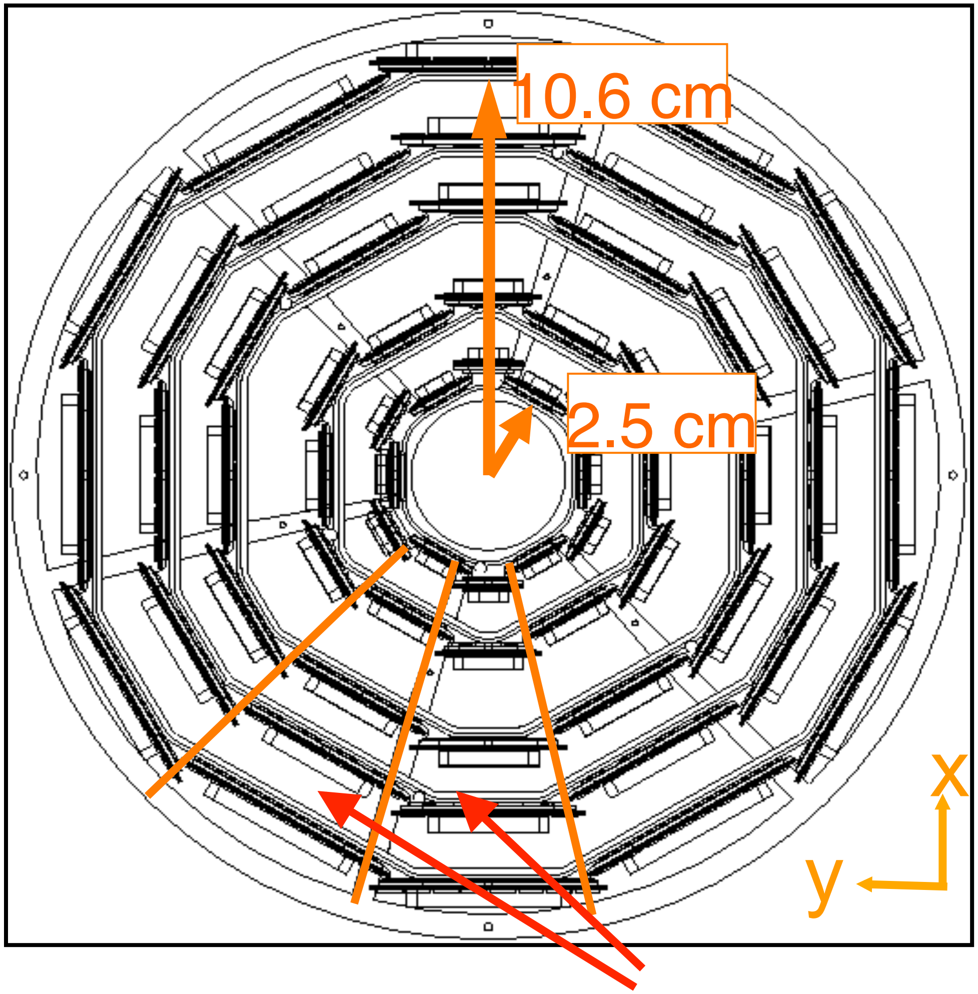
1. Pattern recogniton:

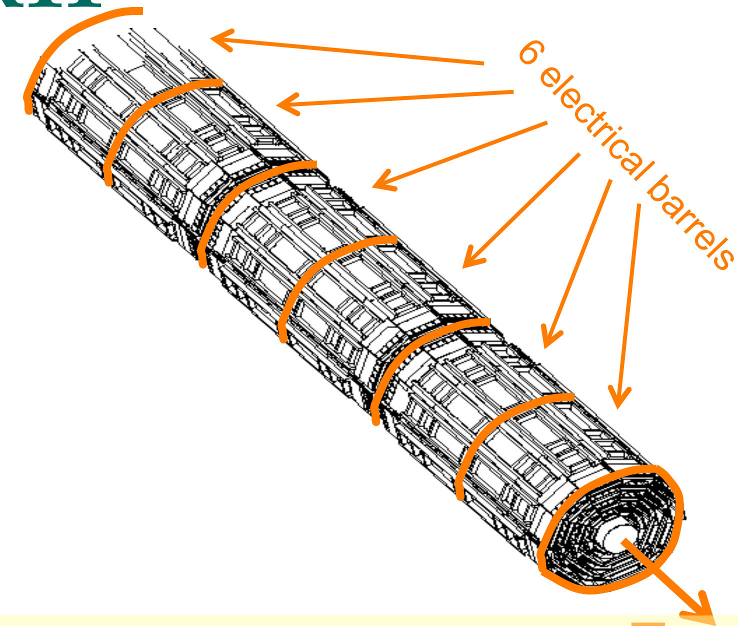
   Search "candidate" tracks (**ROADS**) @ low resolution

2. Track fitting:

   Full resolution 2-D track fit
   $\sigma(q/P_T)$=1.0%/GeV, $\sigma(\phi)$=1.5mrad , $\sigma(d)$=35um

# CDF SVXII




- 5 double sided layers
  - 5 axial + 3 x 90$^{o}$, 2 x 1.2$^{o}$
- Very compact
- Tight alignment tolerances
  - For the trigger
- Very symmetric
  - 12 fold in $\Phi$
  - 6 barrels in Z

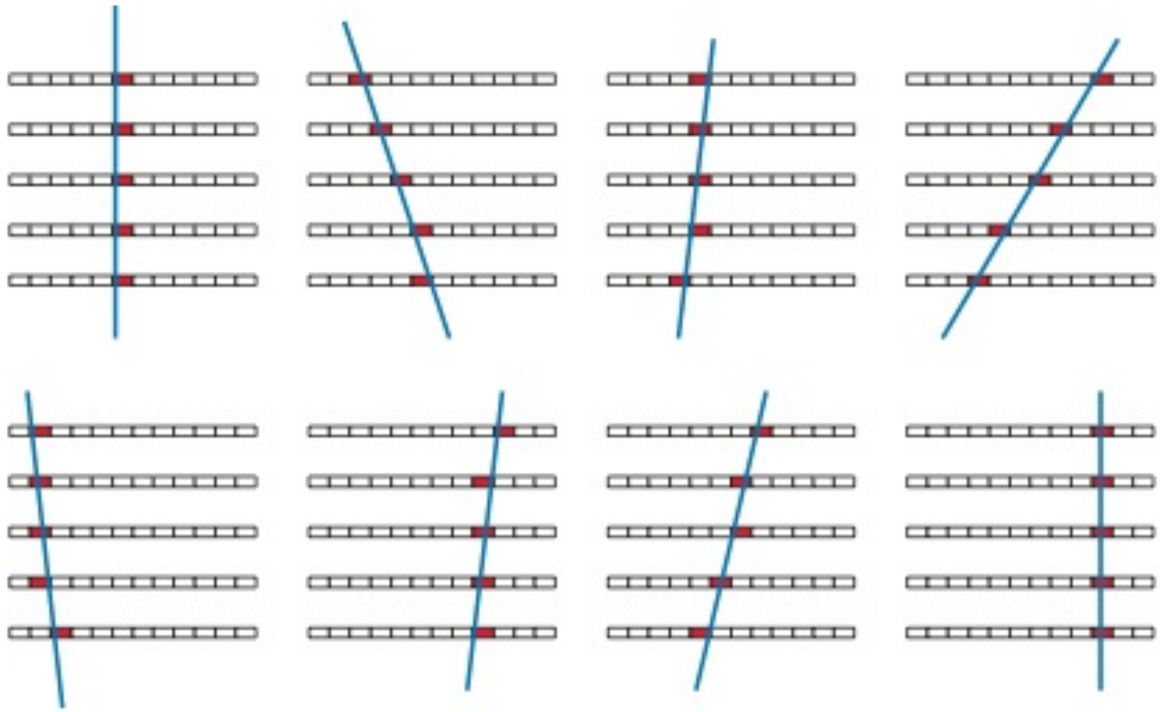


**Figure 2**

An ideal detector with five sensitive layers, each of which is divided into a number of segments. A charged track crossing the five layers fires one and only one segment per layer, producing a pattern of hits. If we let the track parameters span a certain volume of the phase space, a corresponding finite set of distinct patterns is generated.

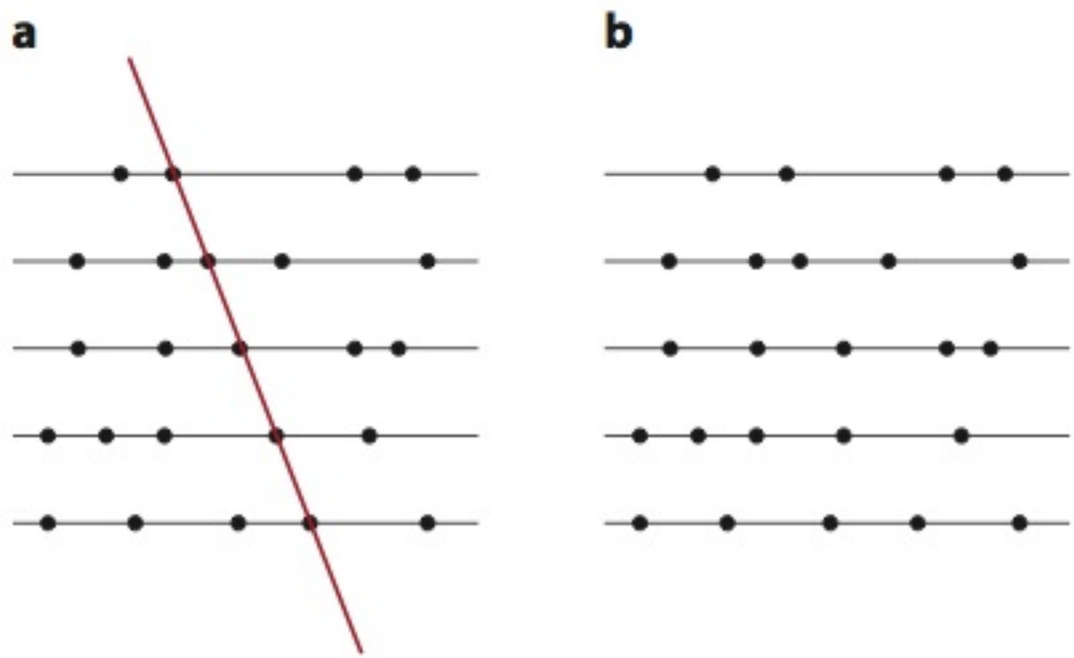
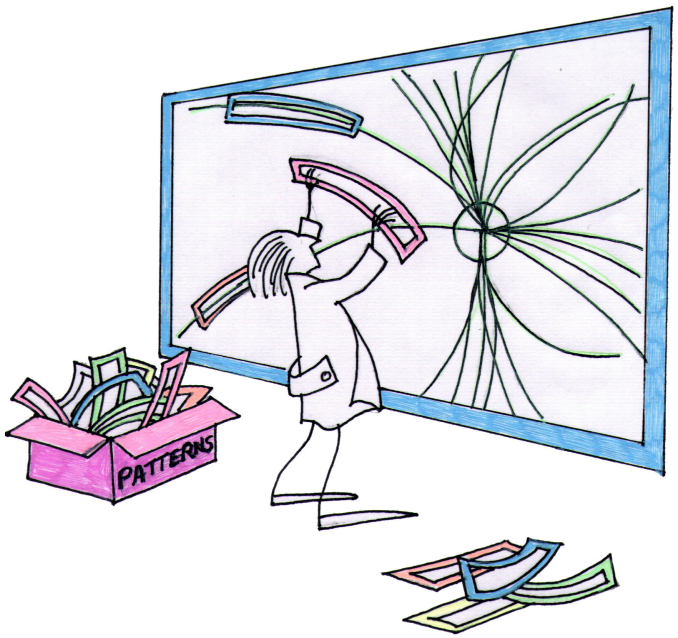


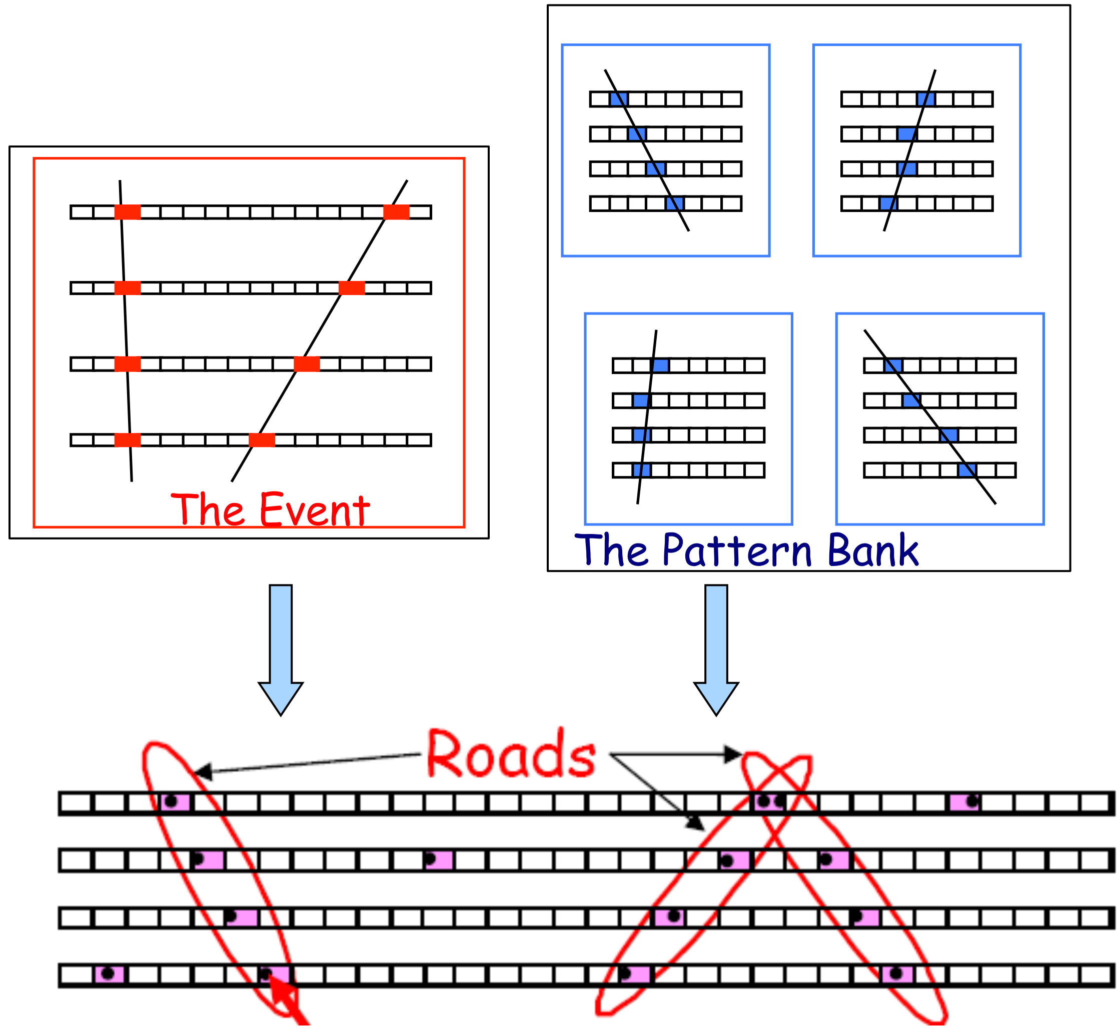

**Figure 3**

A typical pattern-recognition problem, consisting of noise hits superimposed on a track. The problem is to implement an algorithm that can tell the difference between the two cases shown in panels *a* and *b*, namely that in panel *a* we can find a combination of hits, one per detector layer, that can be produced by a single track, whereas in panel *b* no such combination exists.

# CDF Silicon Vertex Trigger (SVT) for RunII

Pattern Matching using Associative memory
(M.Dell' Orso and L. Ristori: initial idea in 1985)



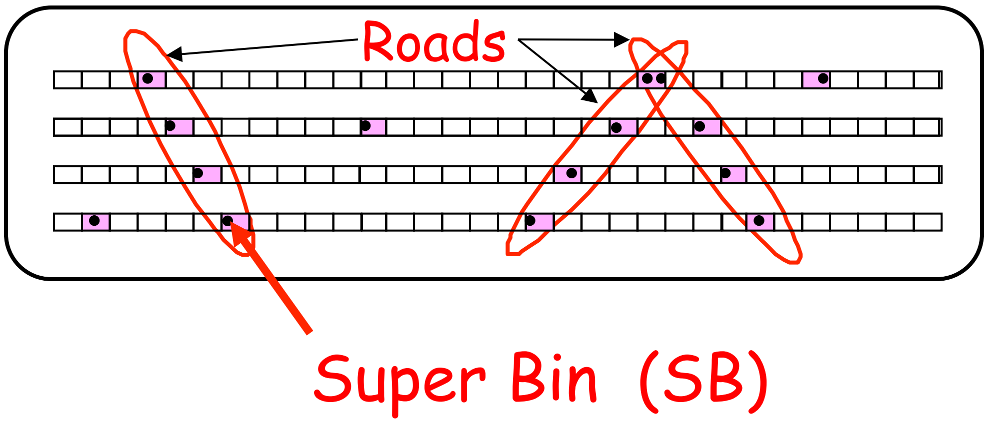

**Finds low resolution track candidates first (roads)**

**Then perform track fitting downstream**

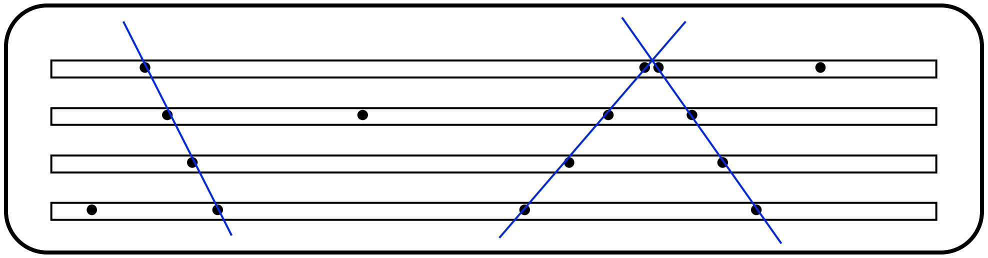
# Tracking in 2 steps

- Pattern recognition and track fitting done separately and pipelined

1. Find low resolution track candidates called "roads". Solve most of the pattern recognition




2. Then fit tracks inside roads. Thanks to 1st step it is much easier





Fermilab

# Track Fitting

- To complete the tracking in 10 μs, must do a fit at a rate of 1 per ns!
- It is not possible to do a fit of hits to a helical path in 1 ns.
- However if a small region of the detector is considered, a linear approximation gives near ideal precision within the required execution time.

$$p_i = \sum_{j=1}^{14} a_{ij} x_j + b_i$$

  - **$p_i$'s are the helix parameters and $\chi^2$ components.**
  - **$x_j$'s are the hit coordinates in the detector layers.**
  - **$a_{ij}$ & $b_i$ are prestored constants.**
  - **This is VERY fast in FPGAs (multiply & accumulate)**
  - **1 ns/fit is achievable (many DSPs within the FPGA)**

# RAM vs CAM from wiki

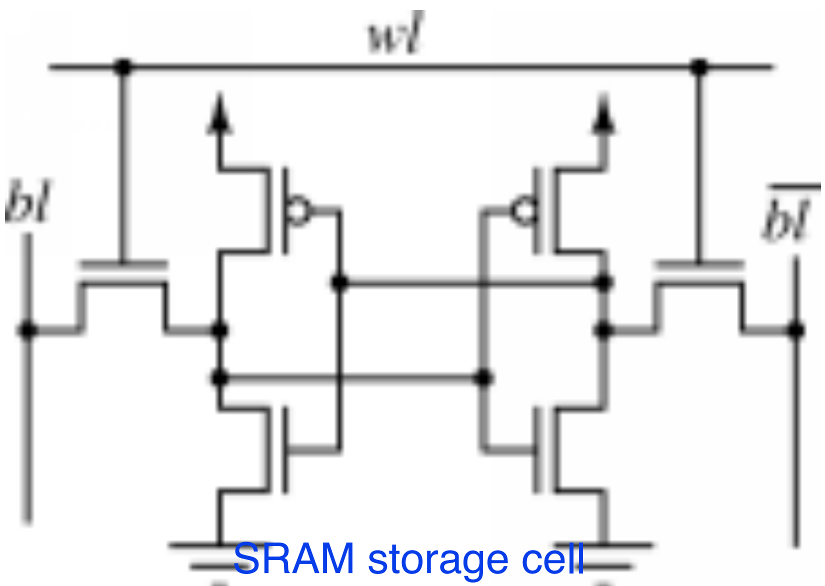
- With random access memory or RAM, the user supplies a memory address and the RAM returns the data word stored at that address
- CAM is designed such that the user supplies a data word and the CAM searches its entire memory to see if that data word is stored anywhere in it. In other words, it is accessed by virtue of its contents, not its location.
  - If the data word is found, the CAM returns a list of one or more storage addresses where the word was found
  - In some architectures, it also returns the data word, or other associated pieces of data
- In essence, CAM == “Inverse RAM”

# RAM vs CAM

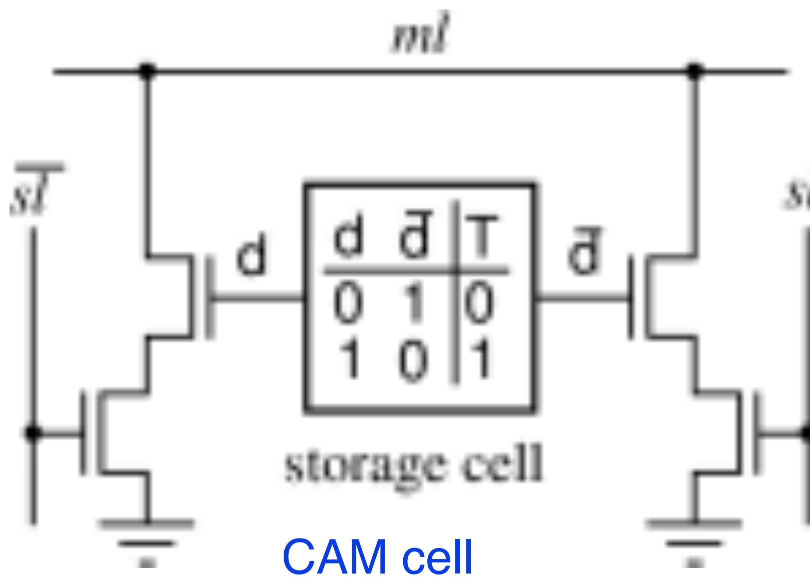
- Why CAM is useful? It is designed to search its entire memory in a single operation, much faster than RAM
  - HEP application example: CDF SVT (Silicon Vertex Trigger)
- However, no pain no gain:
  - each individual CAM memory bit must have its own associated comparison circuit to detect a match
  - match outputs from each cell must be combined to yield a complete data word match signal.
  - The extra circuitry also increases power dissipation since every comparison circuit is active on every clock cycle
  - Consequently, CAM/AM is only used in specialized applications where searching speed cannot be accomplished using a less costly method.

# CAM = SRAM + comparator

- Each memory bit in a CAM must have its own comparison circuit to detect match
- SRAM cell (bottom left) as basic storage cell for CAM (Binary CAM bottom right)
  - The comparison circuitry attached to the storage cell performs a comparison between the data on the search lines (sl and sl-bar) and the data in the cell
  - The matchline (ML) was pre-charged high first
  - A mismatch in a CAM cell creates a path to ground for the matchline (ML)
  - A multi-bit CAM word is a row of adjacent cells with their MLs connected…

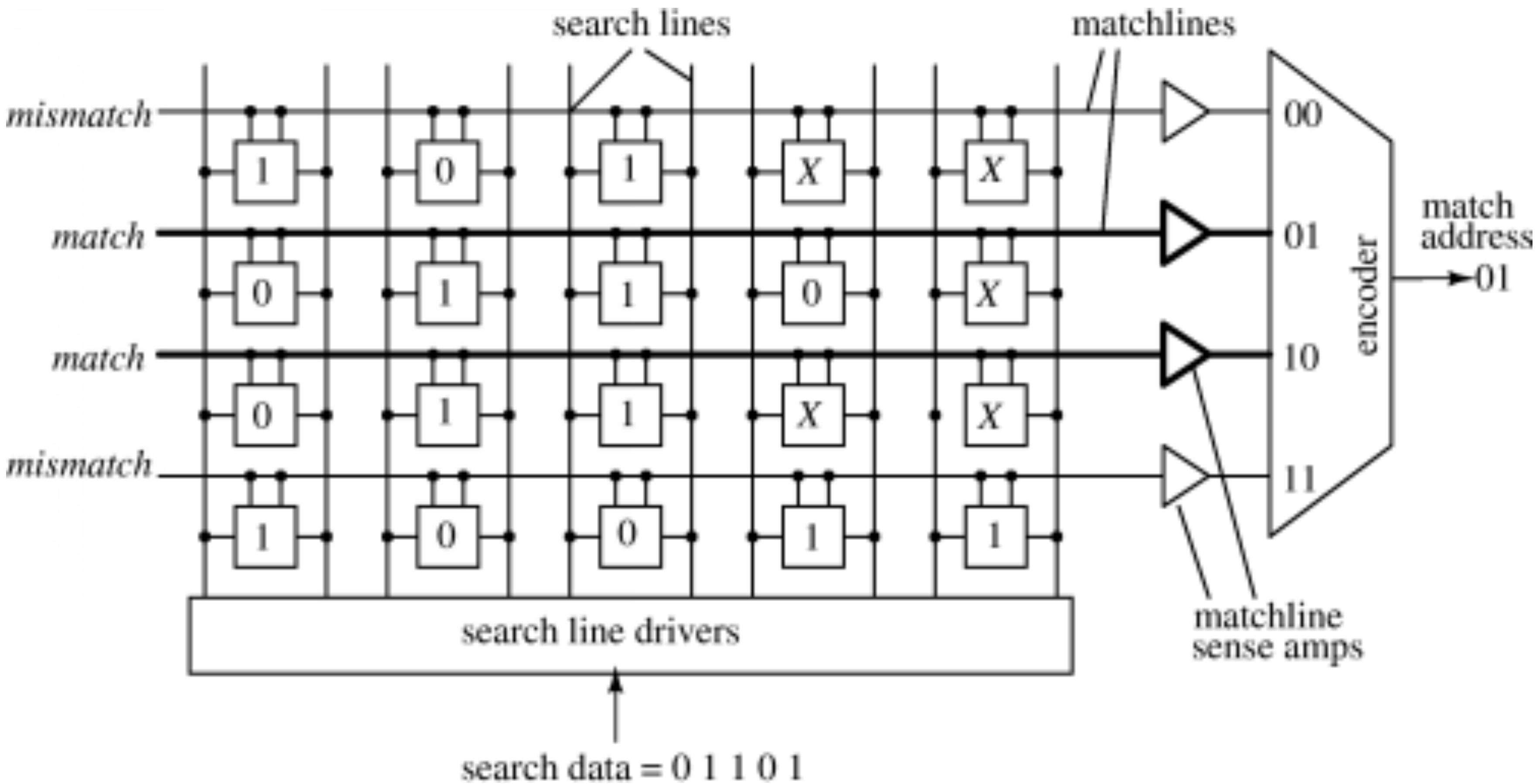



SRAM storage cell

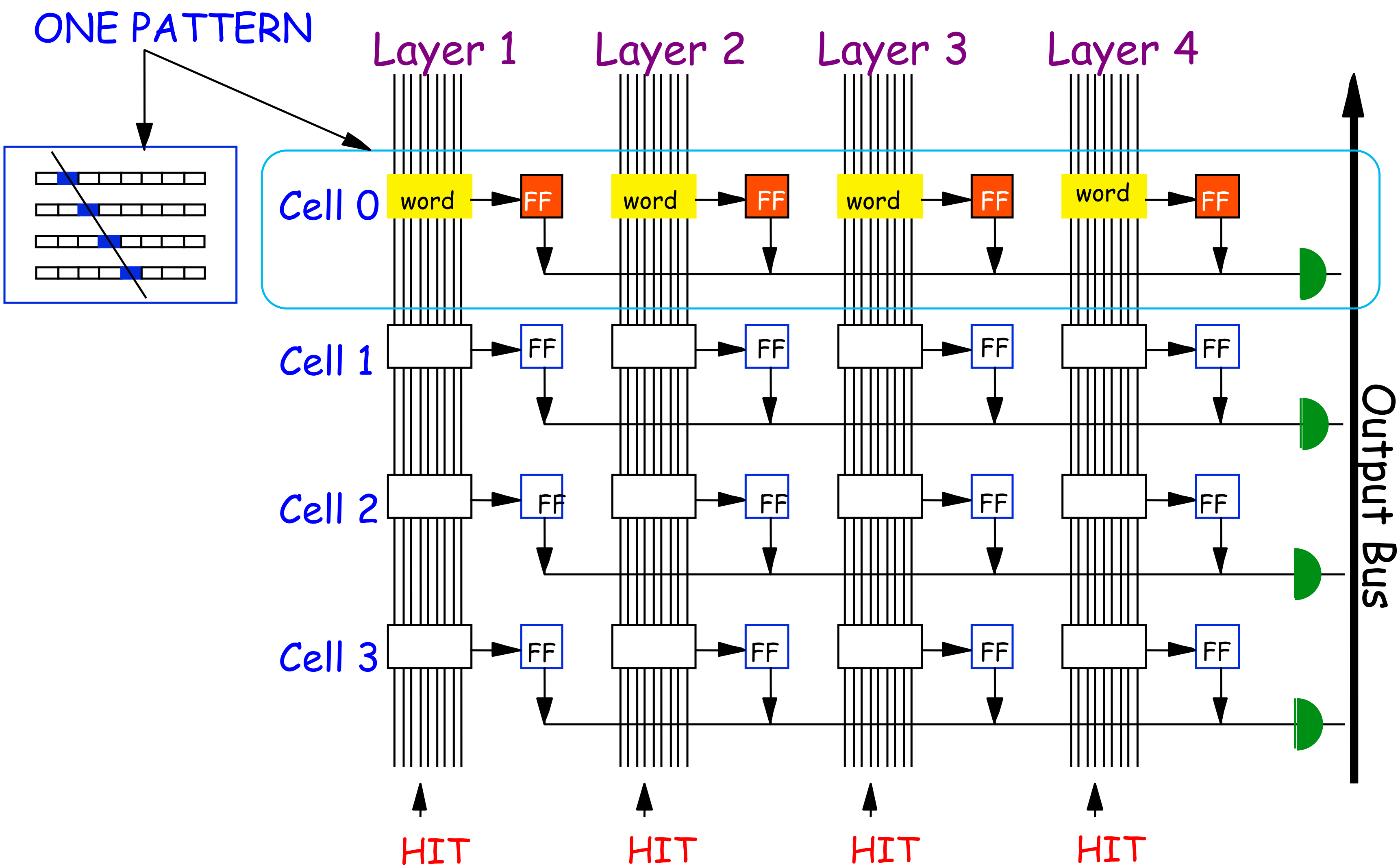



CAM cell

# CAM/AM

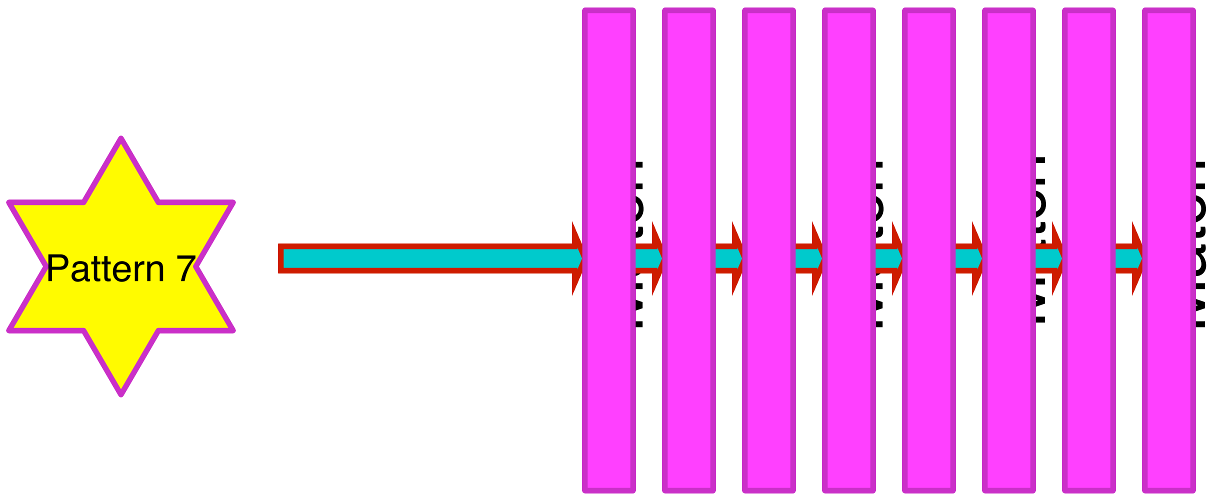

search lines
matchlines
mismatch
match
match
mismatch
1
0
1
X
X
0
1
1
0
X
0
1
1
X
X
1
0
0
1
1
00
01
10
11
encoder
match
address
01
search line drivers
matchline
sense amps
search data = 0 1 1 0 1

# Back to the basics of Associative Memory



# How CAM works

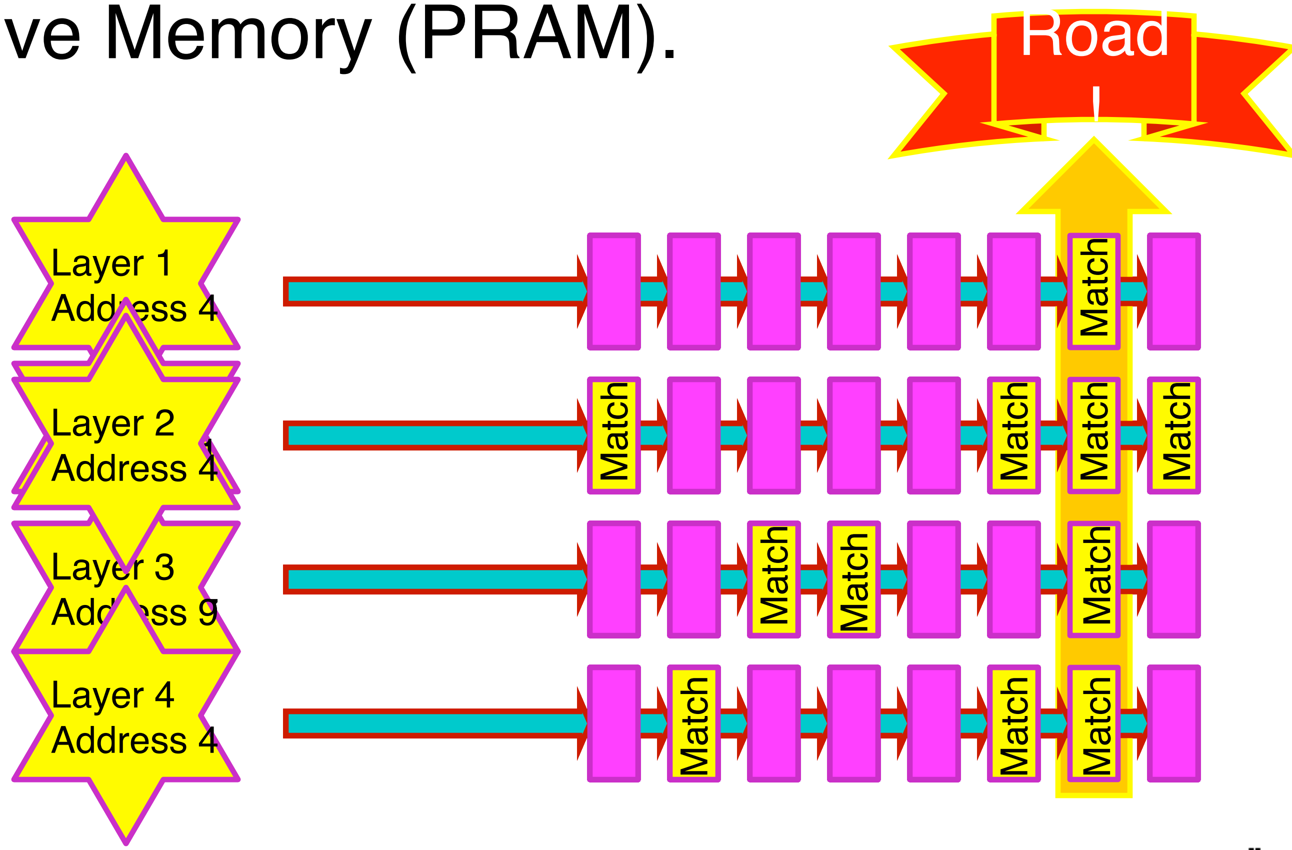
- A CAM (Content Addressable Memory) is a classical digital system building block




- •One pattern at a time
- •Each CAM cell responds or does not respond to the current pattern
- •There is no memory of previous matches

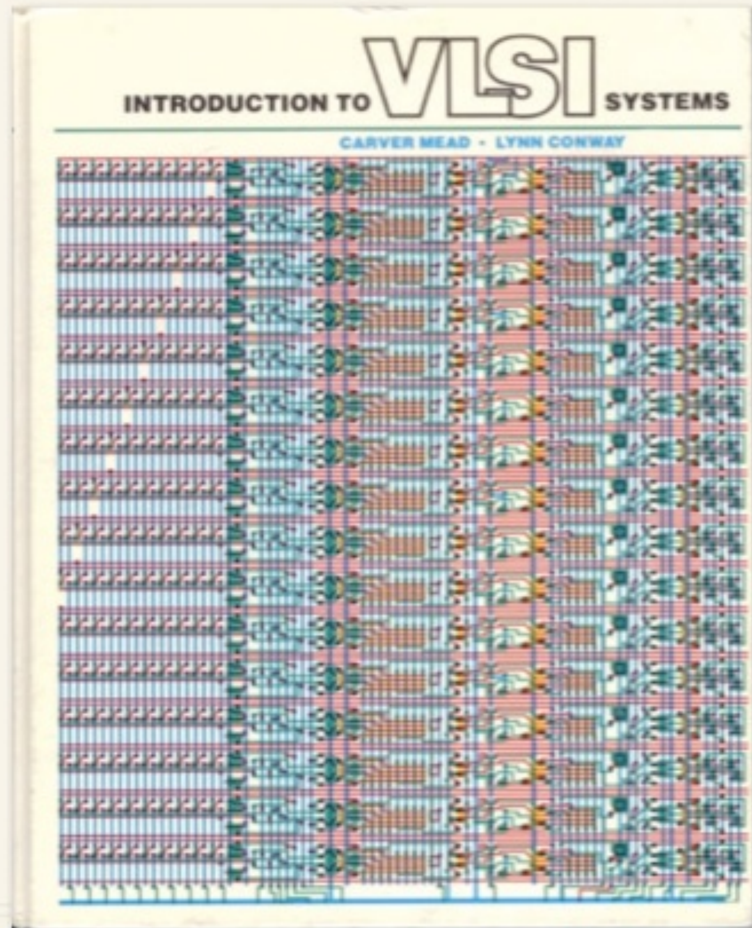
# How PRAM works

- A PRAM on the other hand is a Pattern Recognition Associative Memory (PRAM).



# Very Large Scale Integration
## the revolution

Carver Mead & Lynn Conway




in the ‘80s the technology of VLSI design becomes available to the universities and to small research projects

Slides from Luciano Ristori
at TIPP 2011 conference

436

Nuclear Instruments and Methods in Physics Research A278 (1989) 436–440
North-Holland, Amsterdam

October 24, 1988

# VLSI STRUCTURES FOR TRACK FINDING

Mauro DELL'ORSO

*Dipartimento di Fisica, Università di Pisa, Piazza Torricelli 2, 56100 Pisa, Italy*

Luciano RISTORI

*INFN Sezione di Pisa, Via Vecchia Livornese 582a, 56010 S. Piero a Grado (PI), Italy*

Received 24 October 1988

We discuss the architecture of a device based on the concept of *associative memory* designed to solve the track finding problem, typical of high energy physics experiments, in a time span of a few microseconds even for very high multiplicity events. This "machine" is implemented as a large array of custom VLSI chips. All the chips are equal and each of them stores a number of "patterns". All the patterns in all the chips are compared in parallel to the data coming from the detector while the detector is being read out.

## 1. Introduction

The quality of results from present and future high energy physics experiments depends to some extent on the implementation of fast and efficient track finding algorithms. The detection of *heavy flavor* production, for example, depends on the reconstruction of secondary vertices generated by the decay of long lived particles, which in turn requires the reconstruction of the majority of the tracks in every event.

Particularly appealing is the possibility of having detailed tracking information available at trigger level even for high multiplicity events. This information could be used to select events based on impact parameter or secondary vertices. If we could do this in a sufficiently short time we would significantly enrich the sample of events containing heavy flavors.

Typical events feature up to several tens of tracks each of them traversing a few position sensitive detector layers. Each layer detects many hits and we must correctly correlate hits belonging to the same track on different layers before we can compute the parameters

## 2. The detector

In this discussion we will assume that our detector consists of a number of layers, each layer being segmented into a number of *bins*. When charged particles cross the detector they *hit* one bin per layer. No particular assumption is made on the shape of trajectories: they could be straight or curved. Also the detector layers need not be parallel nor flat. This abstraction is meant to represent a whole class of real detectors (drift chambers, silicon microstrip detectors etc.). In the real world the coordinate of each hit will actually be the result of some computation performed on "raw" data: it could be the center of gravity of a cluster or a charge division interpolation or a drift-time to space conversion depending on the particular class of detector we are considering. We assume that all these operations are performed upstream and that the resulting coordinates are "binned" in some way before being transmitted to our device.



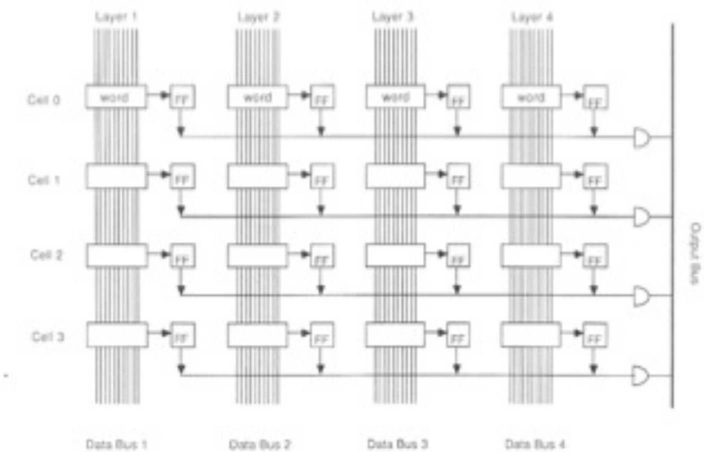



Fig. 3. Associative memory architecture.

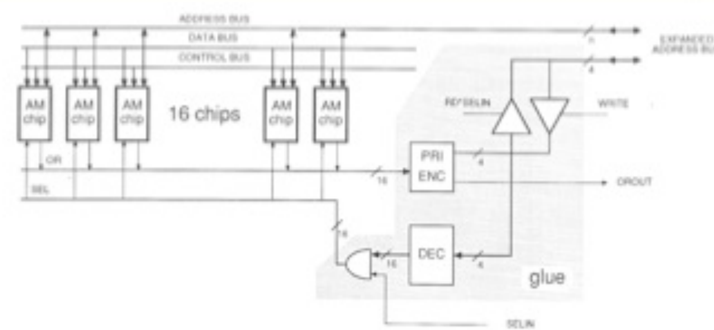
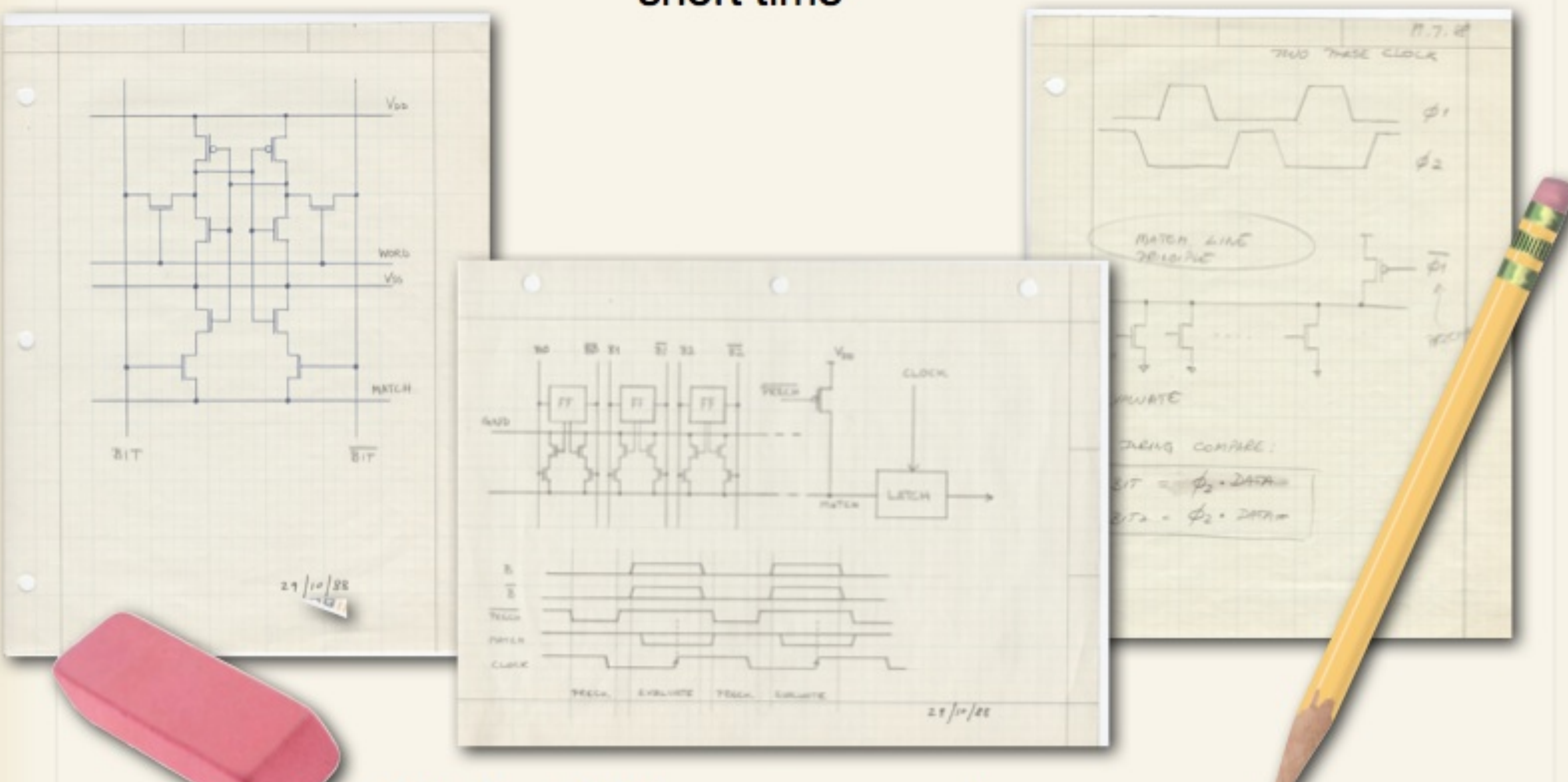



Fig. 5. 16 AM chips tied by the "glue".

We discuss the architecture of a device based on the concept of *associative memory* designed to solve the track finding problem, typical of high energy physics experiments, in a time span of a few microseconds even for very high multiplicity events. This "machine" is implemented as a large array of custom VLSI chips. All the chips are equal and each of them stores a number of "patterns". All the patterns in all the chips are compared in parallel to the data coming from the detector while the detector is being read out.

it is at this point in time that we came up with the idea to use VLSI technology to solve the pattern recognition problem and reconstruct tracks in the detector in a very short time

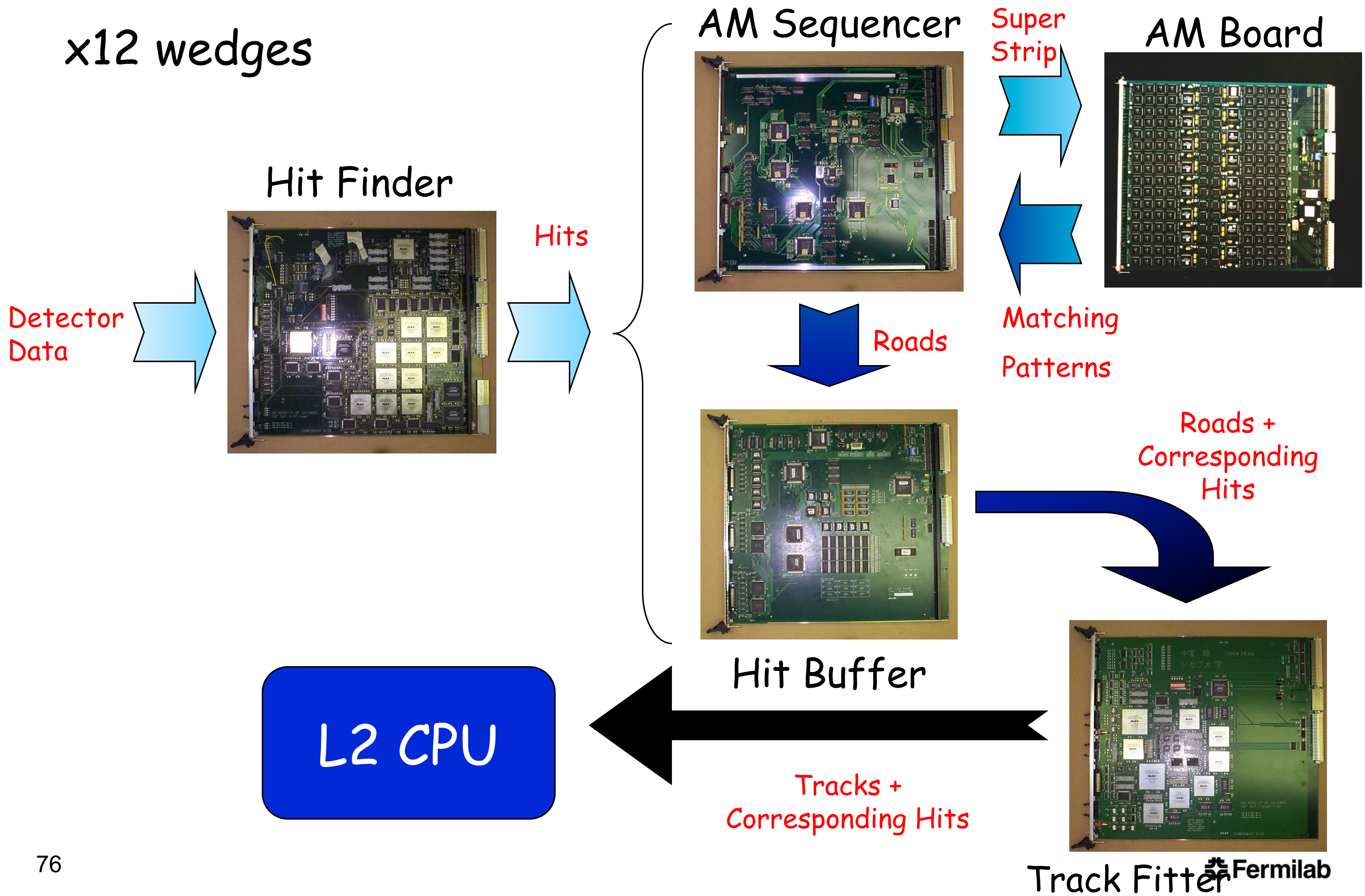



*October 1988: paper, pencil, eraser....*

# Pattern Recognition

- Hit combinations that form possible tracks are precalculated and stored ("pattern bank")
- To make the bank small, low resolution bins are used
- Every hit is compared with each stored pattern in parallel
- Small bins -> large bank->lower background->faster fit
- Large bins-> small bank->higher background->slower fit

# Original SVT system



# SVT System

2 meters

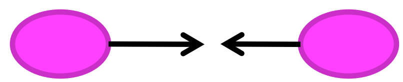

Reduces gigabytes/second to megabytes/second

Peak (avg): 20 (0.5) GB/s ⟶ 100 (1.5) MB/s



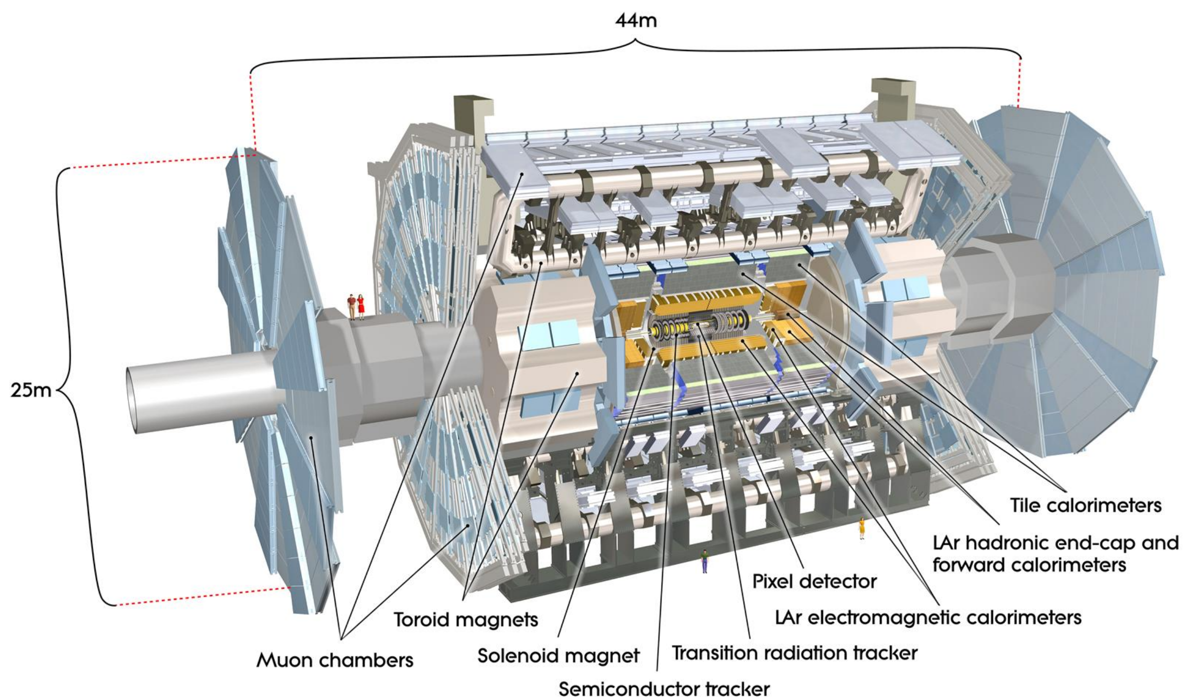
Fermilab

# The LHC Challenge



40 MHz accelerator bunch crossing rate

25-75 *pp* collisions per bunch crossing



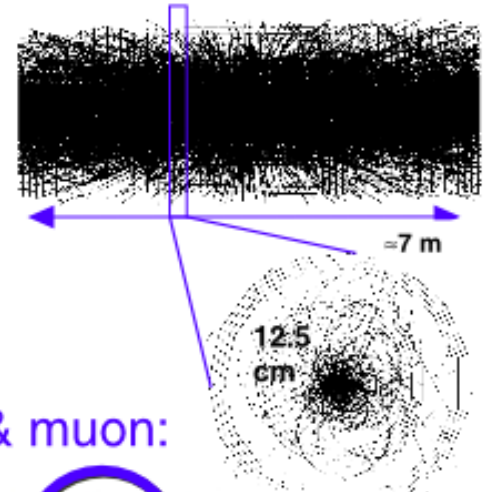

- 85M detector channels
- ~ 1 MB of data/event
- ⇒ can store 200 events/sec

**LHC ATLAS detector**

# ATLAS and CMS Strategy

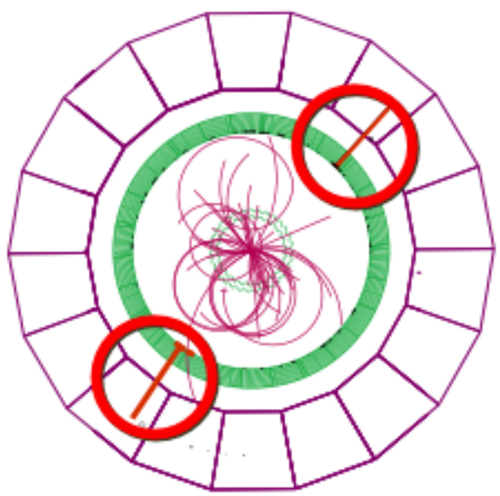
## Level-1 : only calorimeters & muons ....

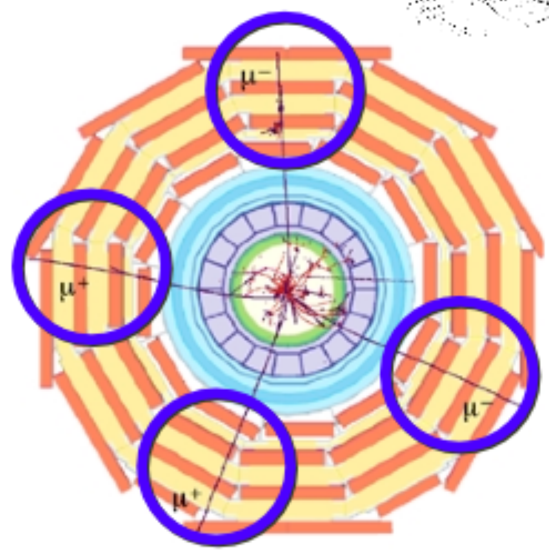
Compare to Central tracking at L = $10^{34}$
(50 ns integration, ≈1000 tracks)




Algorithm Complexity
+
huge amount of data

Pattern recognition much easier on calo & muon:



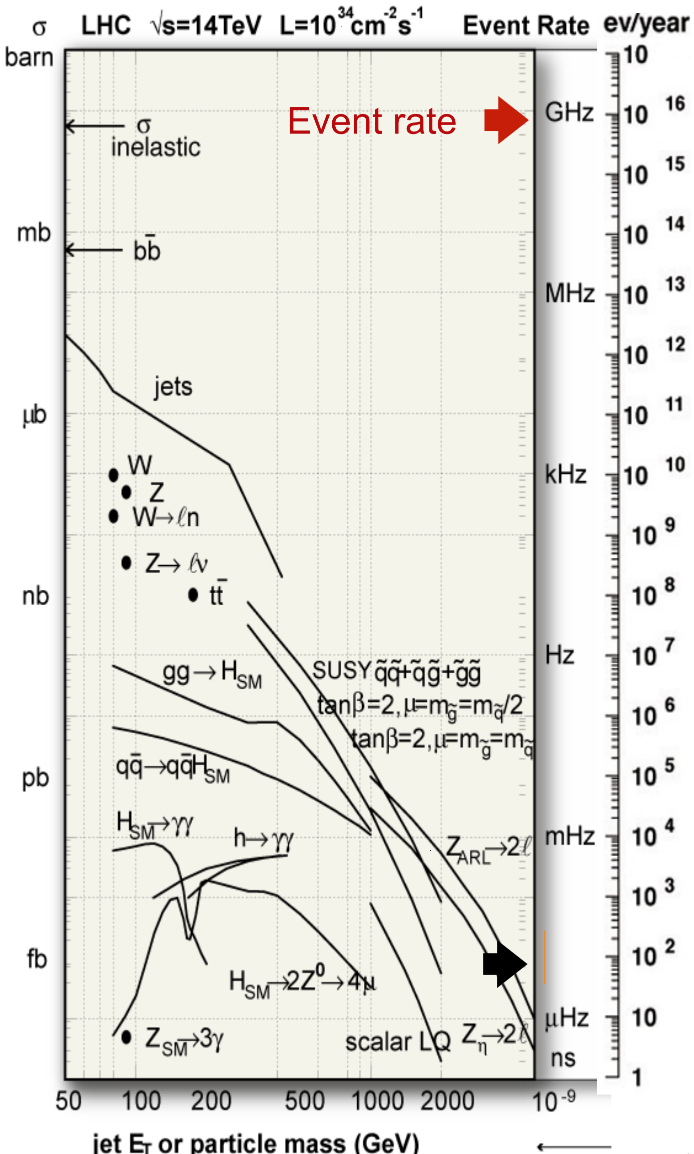


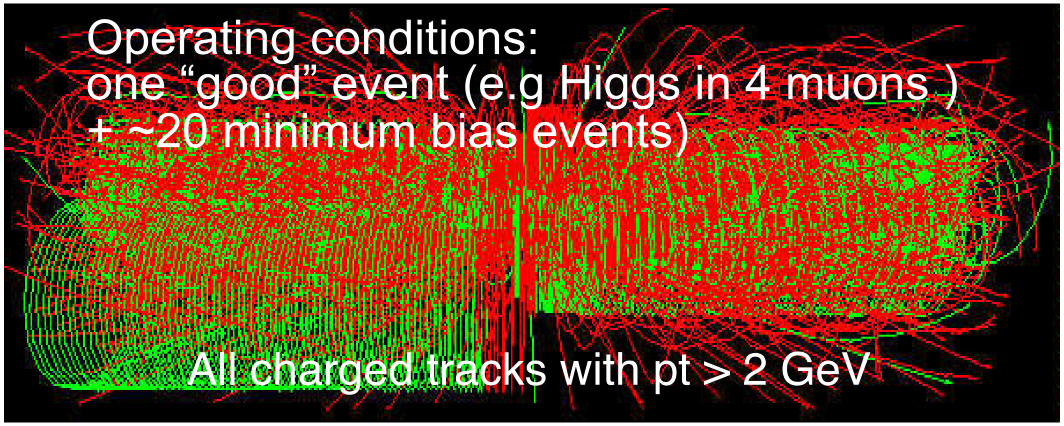

Complexity handled in software on CPUs

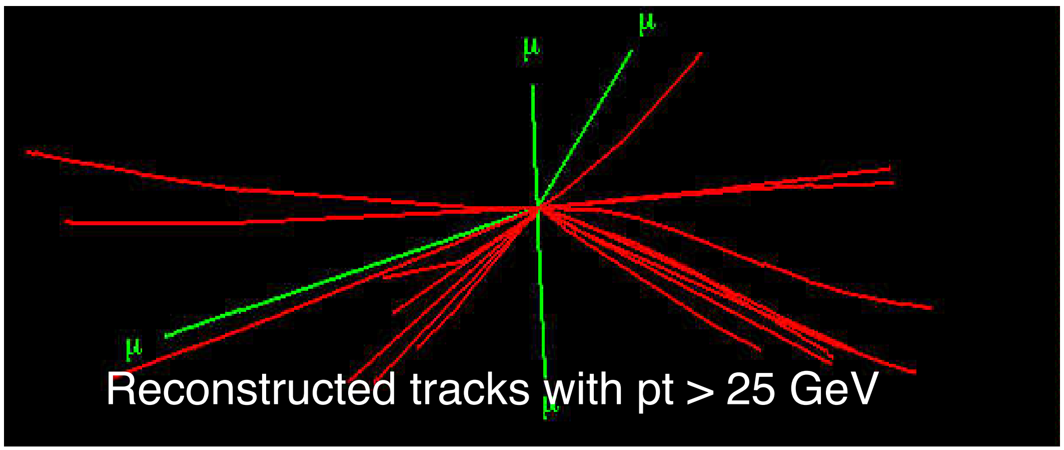
at high level trigger

The approach works well at low luminosity

# Collisions (p-p) at LHC



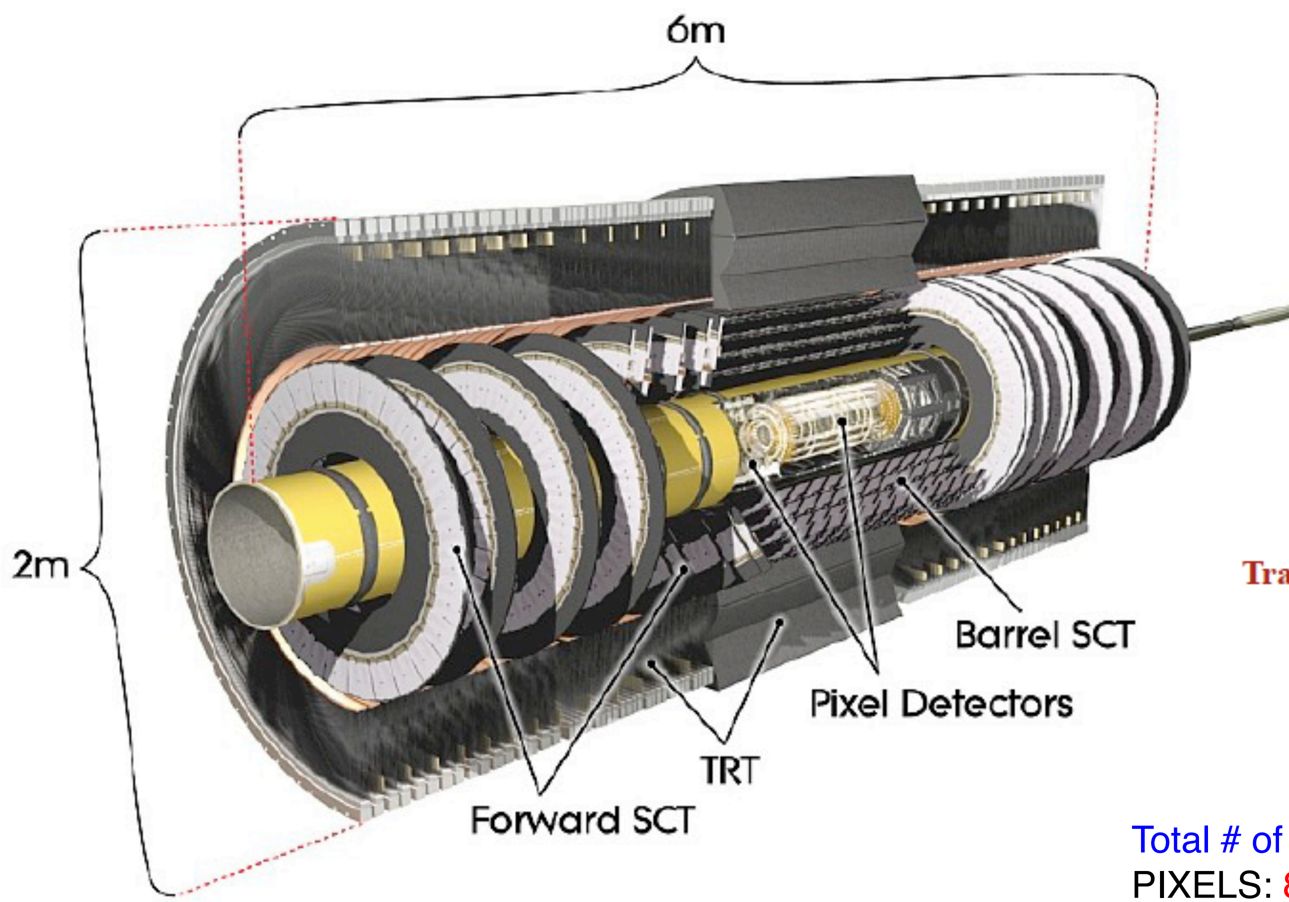



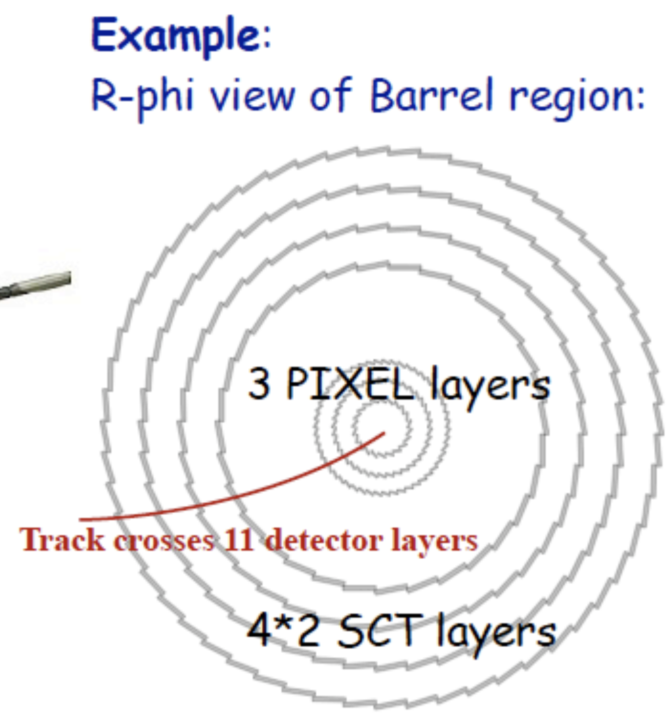

S. Cittolin CERN/EP-CMD LECC Workshop 2002



Fermilab

# Fast tracking with pixel and SCT det.




Total # of readout channels:
PIXELS: 80 millions
SCT: 6 millions

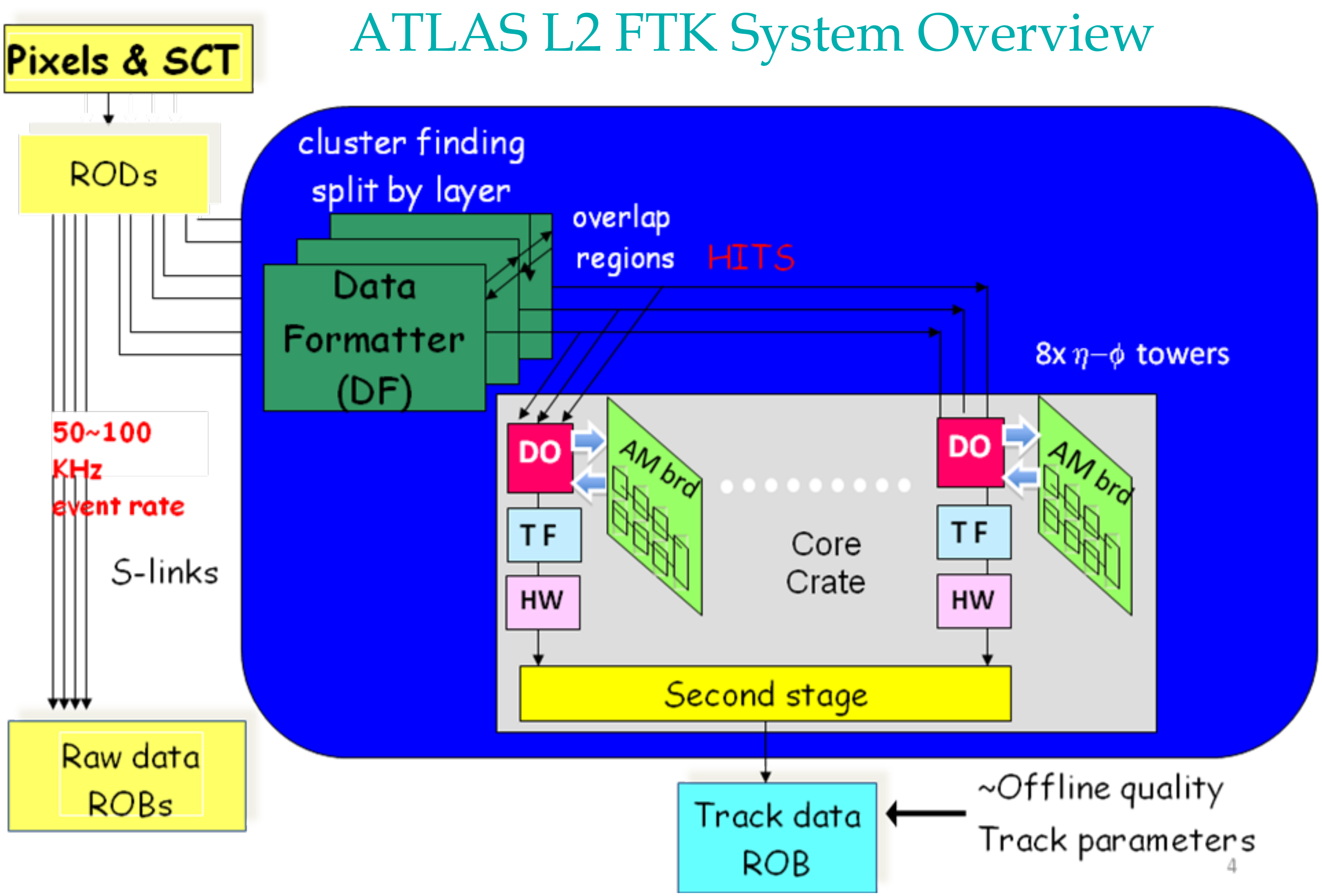


Fermilab

# The importance of individual tracks

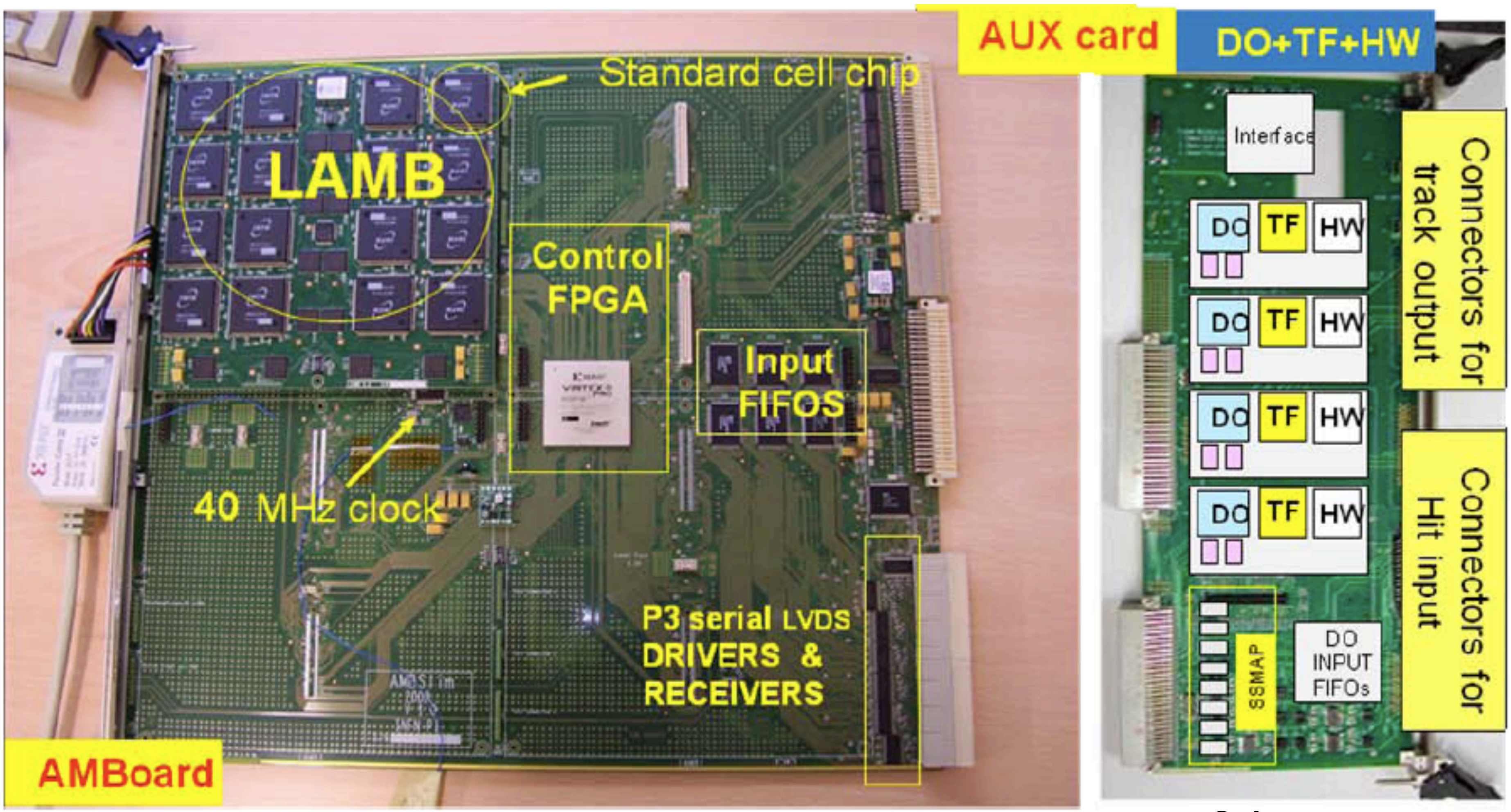
- Many/most new physics scenarios produce final states containing heavy elementary particles *(b* quarks & $\tau$ leptons).
  - must be separated from an enormous background of light quarks and gluons produced through the strong nuclear force
    - *b*-jets: displaced vertices from *B* meson with picosecond lifetime
    - $\tau$-jets: 1 or 3 tracks in a narrow cone with a surrounding isolation region due to the decay of a relatively low mass object.

- Even for the traditional workhorse trigger, an isolated high energy electron or muon, tracking is essential at very high accelerator intensity: The usual isolation (calorimeter) deteriorates badly in its efficiency because it integrates over the 25-75 *pp* collisions per beam crossing. Reconstructed tracks each point back to the beam. Isolation only using those close to the muon or electron at the beamline largely removes the effect of the "pile-up".

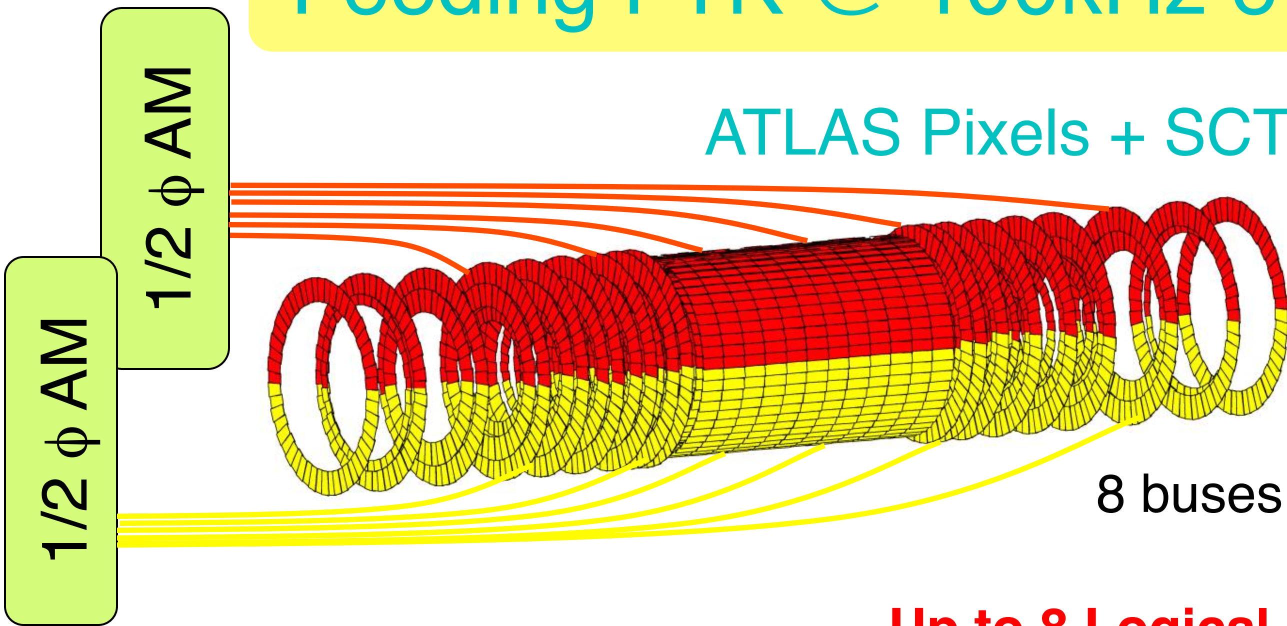



- Highly parallel data flow: 64 $\eta$ -$\phi$ towers in 8 core crates and 4-fold parallelism within each tower (for $3{\times}10^{34}$)

# Procrocessing Unit



# Feeding FTK @ 100kHz event rate



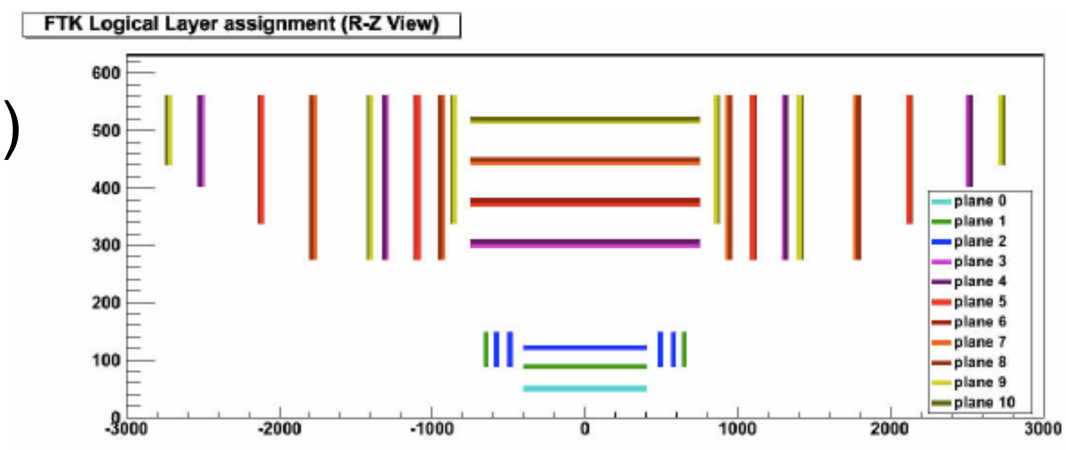

Divide into
more than 2 sectors

Allow a small overlap
for full efficiency

8 buses 100MHz/bus

**Up to 8 Logical Layers: full η coverage**

- 8 φ regions each with
- 8 sub-regions (η−φ towers)
  - δφ~22.5°, δη~1.25
  - bandwidth for up to 3*10E34 $cm^{-2}s^{-1}$






Fermilab

# The technical difficulties

- # of hits in the tracking chamber per beam crossing: 200k
  Must transfer to FTK each 10 μs (100 kHz level-1 trigger rate)
  ⇒ ~ 20 gigawords per second transfer
- This much data makes both stages in tracking very challenging: pattern recognition and track fitting
- There are several hundred good tracks per beam crossing.
  - 10 μs / event ⇒ < 100 ns/track for pattern recognition plus track fitting

# Summary of Lecture II

- Trigger in HEP II: the view from instrumentation
  - A reminder on what we learned in Lecture I
  - Concept of triggering on interesting events
  - Then take a closer look at *tracking trigger* the rest of lecture
  - Case study: L1 tracking trigger
    - Babar L1 Drift Chamber Track Trigger (Zeus track trigger)
    - CDF L1 Drift Chamber Track Trigger (D0 fiber track trigger)
  - Case study:  Silicon Track Trigger
    - CDF L2 Silicon Vertex Trigger (SVT)
    - ATLAS Fast Track Trigger (FTK)
  - Comments on tracking trigger challenges in the future
  - Next lecture (III): future challenges in tracking trigger