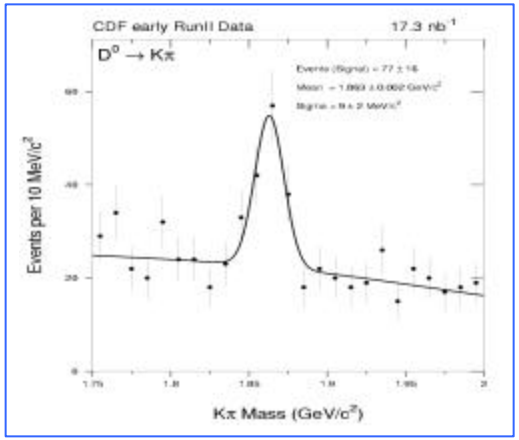
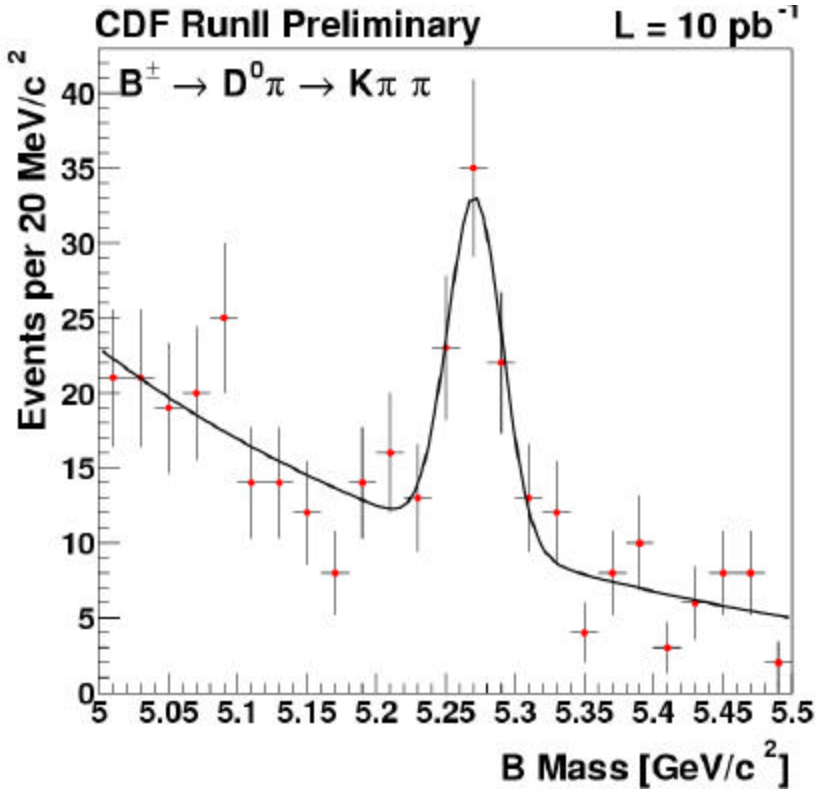
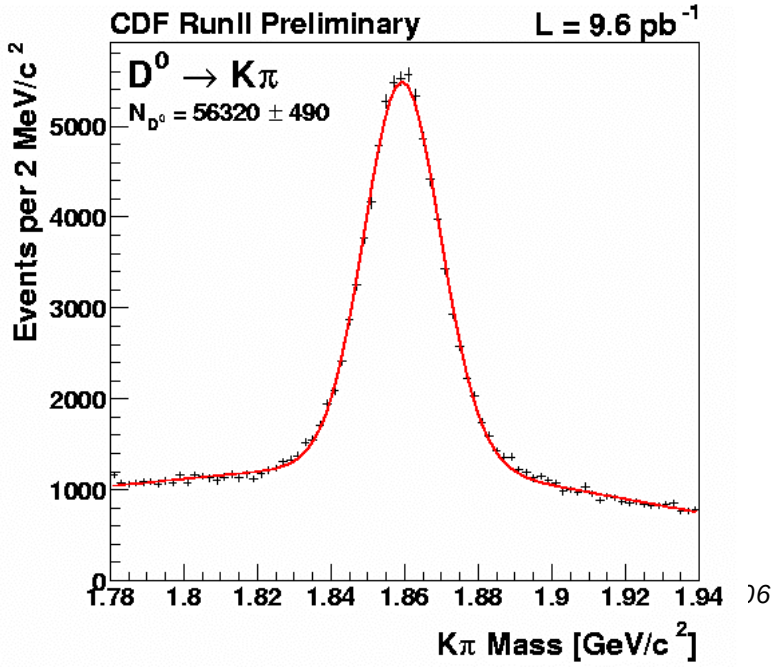
SVT Progress Report

- When you heard from us back in January, this was news ….
  - ➥ **We had just gone into the trigger**
  - ➥ **s(d) was ~75mm; SVT used ~36/72 wedges**
  - ➥ **Timing tails went to 100's of msec**
- We've made some progress since then ….







How is SVT doing?

- We're working well enough to see $B^+ \rightarrow D^0\pi^+$, $B \rightarrow h^+h^-$, lotsa D's
  - **but rate of triggered track pairs (for ICHEP data) was down by factor ~3 to 4 w.r.t. expectations**
    - SVT workshop July 12; follow-up workshop September 10&11
    - recent/forthcoming acceptance/efficiency improvements in next slides
- We do in a ~15μs pipeline what would traditionally take ~hundreds of milliseconds in software
  - **but SVX+SVT path limits L1A rate to ~20-30kHz**
    - workshop September 4 to discuss speed-up options (but it got diverted)
- We have monitoring tools that should make you drool
  - **but we need more automatic problem detection/reporting**
- Beam is fitted every ~30 seconds, written to bank, fed to ACNET
  - **but relationship to CDF global coordinates is not yet perfect**
  - **and we don't yet dare to bug MCR to steer the beam**
- ➔ We should be ecstatic that we made it to this point
  - **but we'll always find something to complain about**

Digression: algorithm overview

- clustering combines adjacent channels that are above a programmable strip threshold, finds centroid
  - **option to suppress hot strips, not currently used**
- pattern recognition looks for coincidence of 5/5 of the following (optionally (4+1)/6)
  - **hits in silicon layers, binned to O(few hundred um)**
  - **XFT track phi at R=10.5, binned to O(1 degree)**
  - **bin = "superstrip," coincidence="pattern"**
- linear fit: 6 measurements $\rightarrow$ 3 parameters plus χ^2

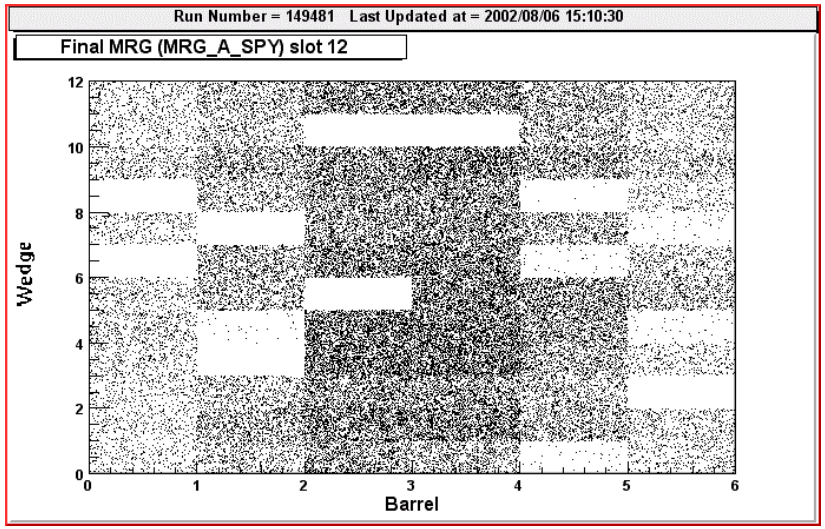
$$(c, \phi, d, x_1, x_2, x_3) = \vec{p} = \vec{p}_0 + V \cdot \vec{x}$$

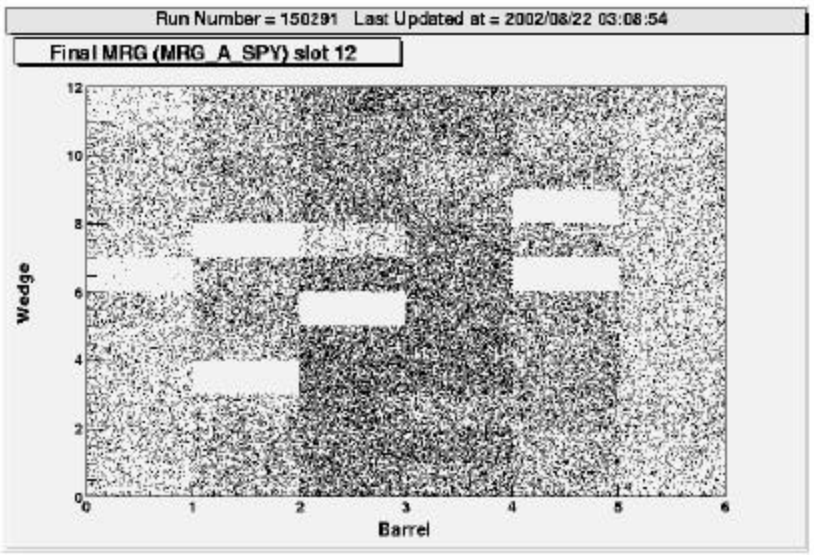
- if there are multiple hits within a superstrip, track fitter iterates over combinations
- ghost removal board picks just one SVT track (currently best χ^2) per XFT track
- beam line computed every ~30 seconds and subtracted dynamically from d vs phi

- constants are generated by Monte Carlo: intersect helix with detector planes
- not all available knobs have been adjusted. Perhaps you have good ideas to suggest?

Summer acceptance improvements

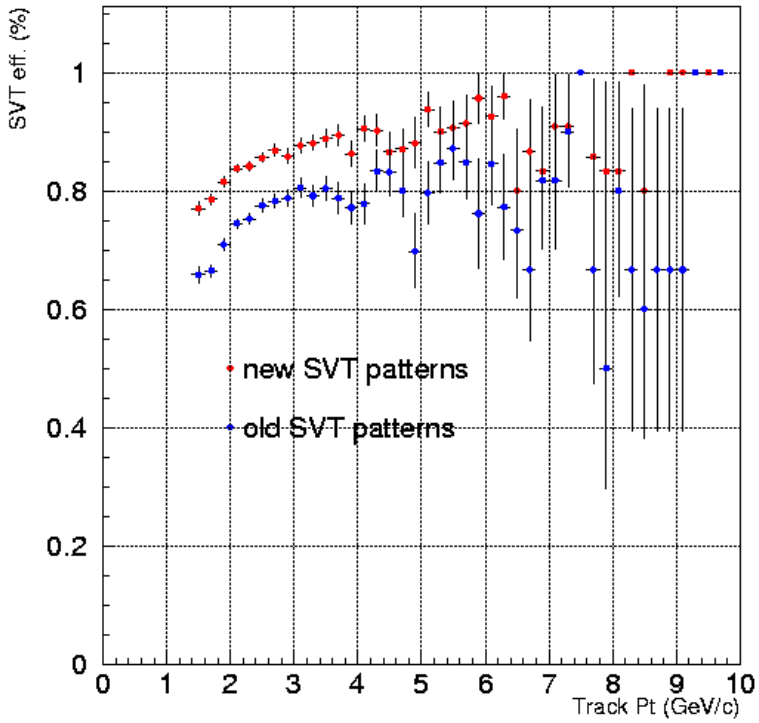
- SVT choice of 4 layers is now per 1/72 of SVX (was 1/12 before April)
- went from 58/72 to 66/72 on August 8
  - **this is every wedge with L0 + 3 others**
  - **a few of these 66 are inefficient; investigate?**
  - **could get B1W3 and B2W5 by skipping L0**
    - predict resolution ~ 55um ⊕ beam
  - **could get others (except B1W7) by using 3 layers**
    - predict 35-40um ⊕ beam if include L0; fake rate??
- Of course the real credit here goes to our SVX friends
  - **lots of work to recover the goods**
  - **they now look at some of our online plots, and they tell us which ladders to use per wedge**





Summer efficiency improvements

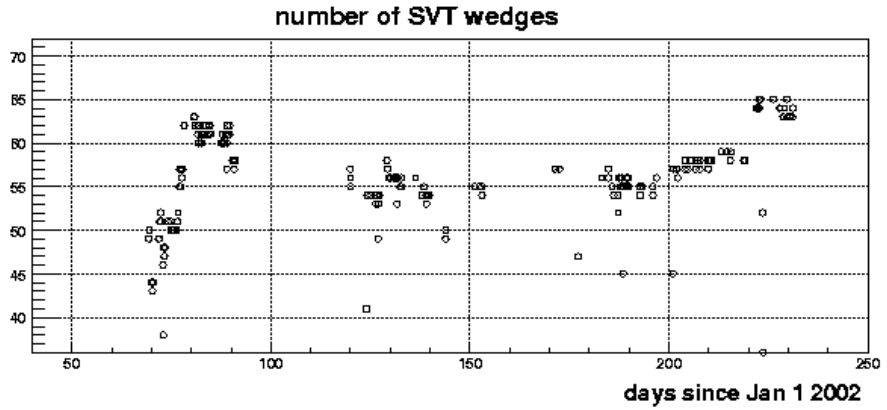
- Pattern generation geometry includes global alignment offsets
- Changes in parent distribution flatten out efficiency vs impact parameter
- Widen roads to increase fraction of desired patterns that fit in finite associative memory
  - **set of desired patterns is made larger by beam offset**
    - old: 6,4,4,6,8 = 360,248,240,360um,1.25°
    - new: 8,4,4,8,12 = 480,248,240,480um,1.875°
  - **allow 4 XFT tracks per ss (was 1)**
- Side effect: if new patterns are needed, we're set up to remake them in a matter of hours

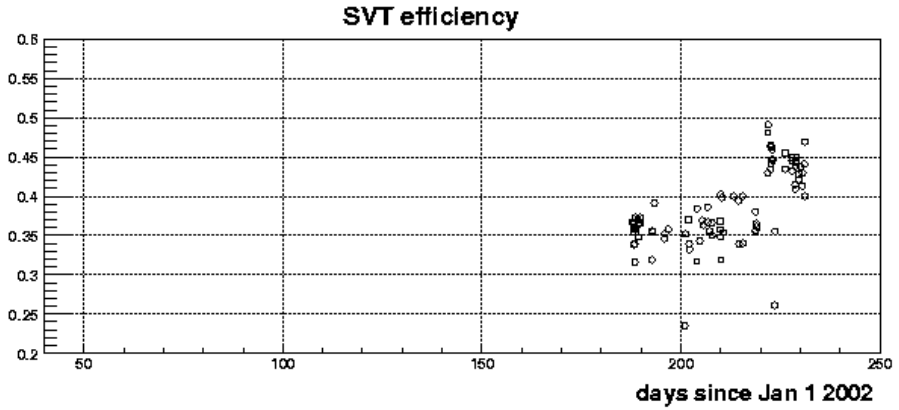


From Marco Rescigno: denominator = OI track using same 4 layers as SVT

Rate/yield results

- Number of wedges used by SVT increased by ~10%
- Pattern efficiency increased by ~10+%
- Rolf Oldeman measures:
  - **SVT track efficiency up 22%**
  - **B_CHARM xsec up 56%**
  - **D0 ® Kp yield up 66%**
  - **Ks ® pp yield up 65%**



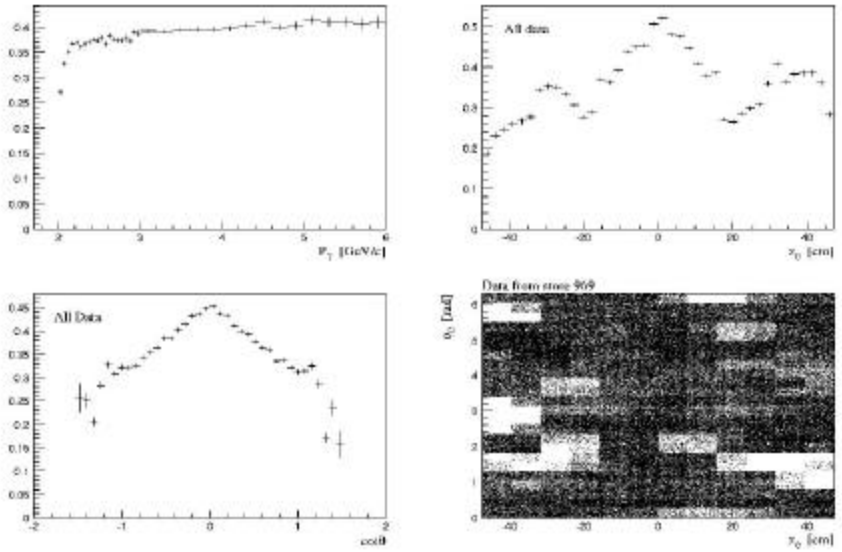


From Rolf Oldeman:
efficiency denominator
= COT track, |z|<zmax

Changing the denominator

- 4/5 majority logic?
  - **5 good layers available in ~2/3 of detector wedges**
  - **in these wedges, the per-track efficiency gain could be as much as ~30% (Barry Wicklund's estimate), depending on per-layer hit efficiency**
    - see e.g. Simon Waschke's CDF 6088
    - need to look more closely at hit-finding efficiency, for specific case of SVT clustering algorithm
  - **some cost in execution time**
    - from Giovanni Punzi's SVT workshop talk, one can optimistically estimate ~2-3usec
    - roughly, if efficiency**2 increases by more than the maximum L1A rate decreases, it's probably a winner
      - **it seems likely to be a winner**

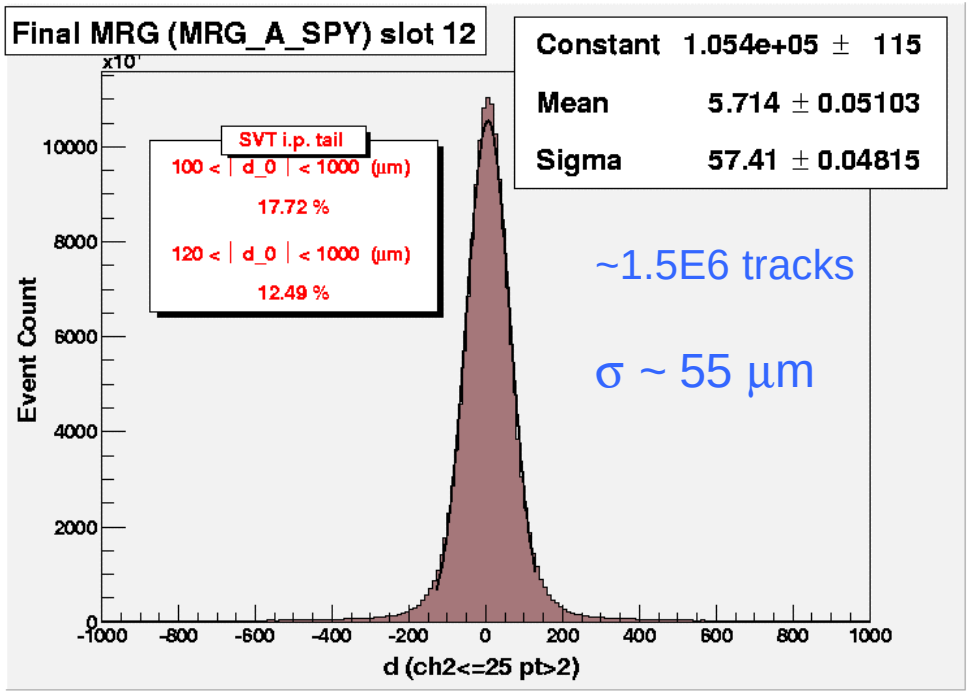
- Barrel crossers?

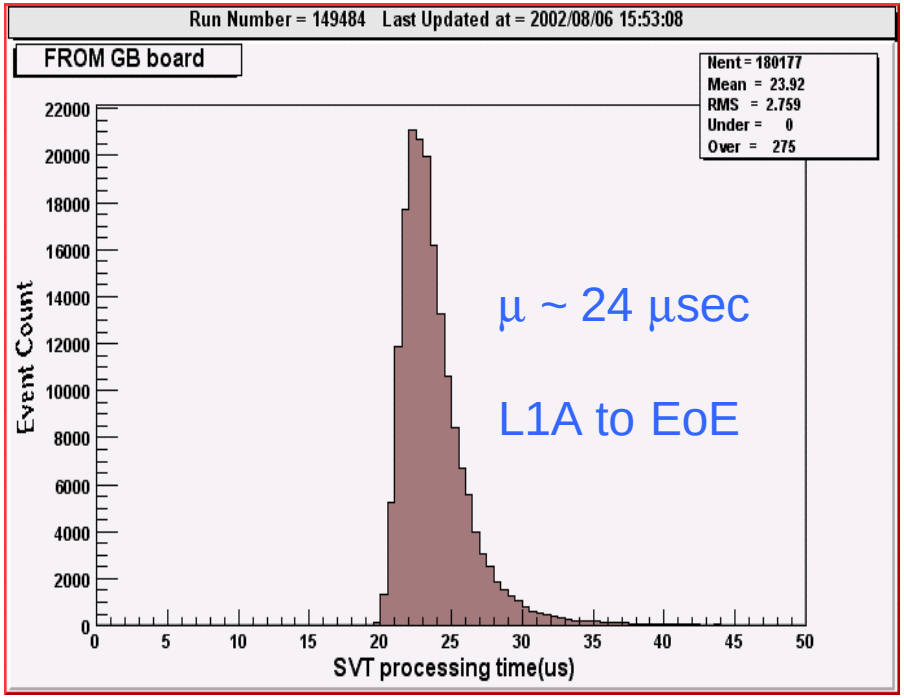


(from Chunhui Chen)

Rolf estimates that we could regain ~15% per track if we had a (no-cost) way to cover (mechanical) barrel-crossers

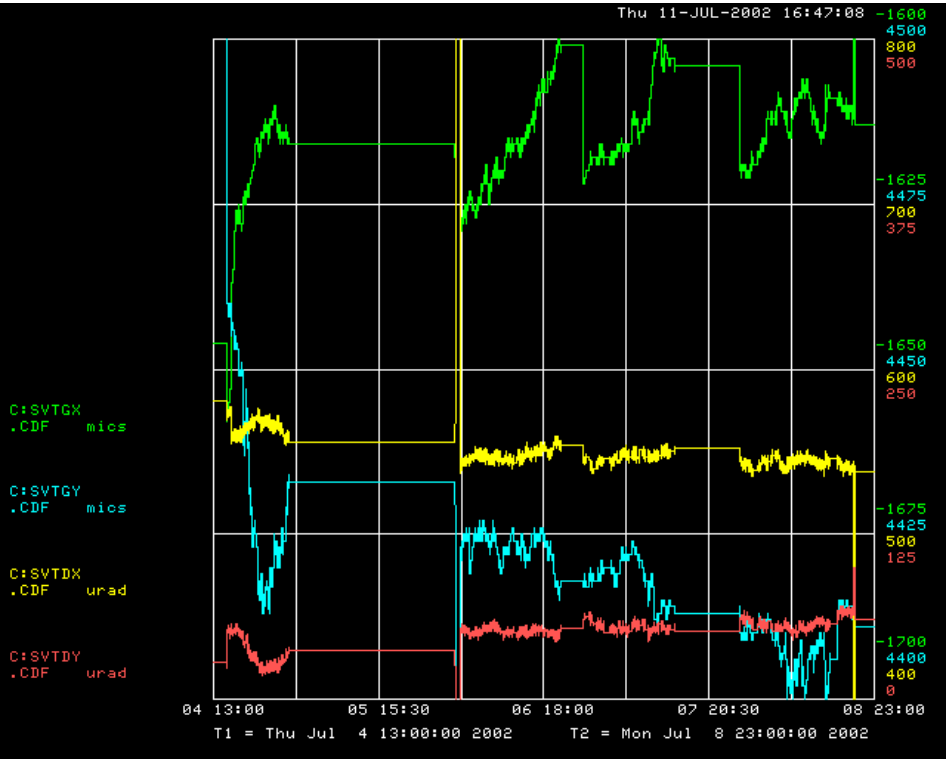
On-crate monitor (biased only by L1)

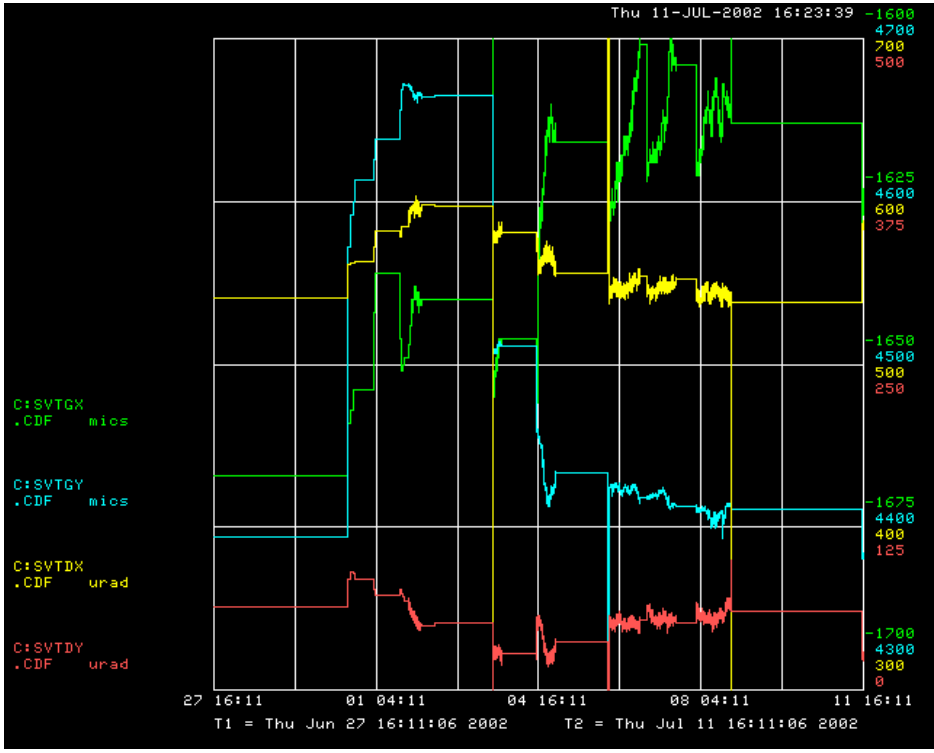




- http://www-cdfonline.fnal.gov/svt
  - ➥ **(follow link from there for up-to-the-minute plots)**
- this also illustrates current performance (resolution, timing)
- the same program runs the beam fit every ~30 seconds

Beam feedback to ACNET (page C82)





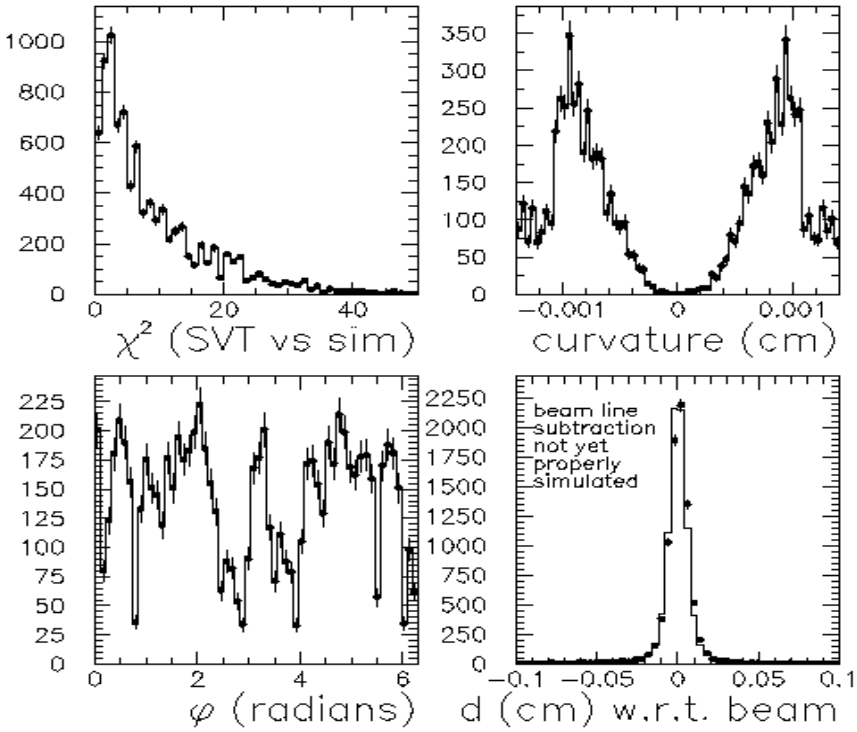
Full scale ~ 100um,400urad x 4 days

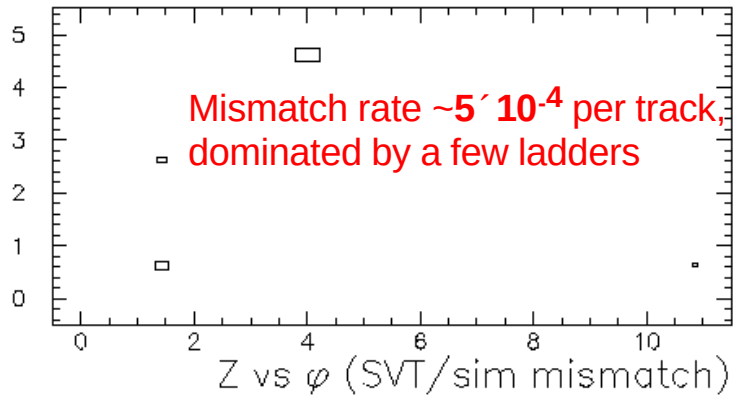
Full scale ~ 200um,400urad x 2 weeks

Motion within store ~ 20um

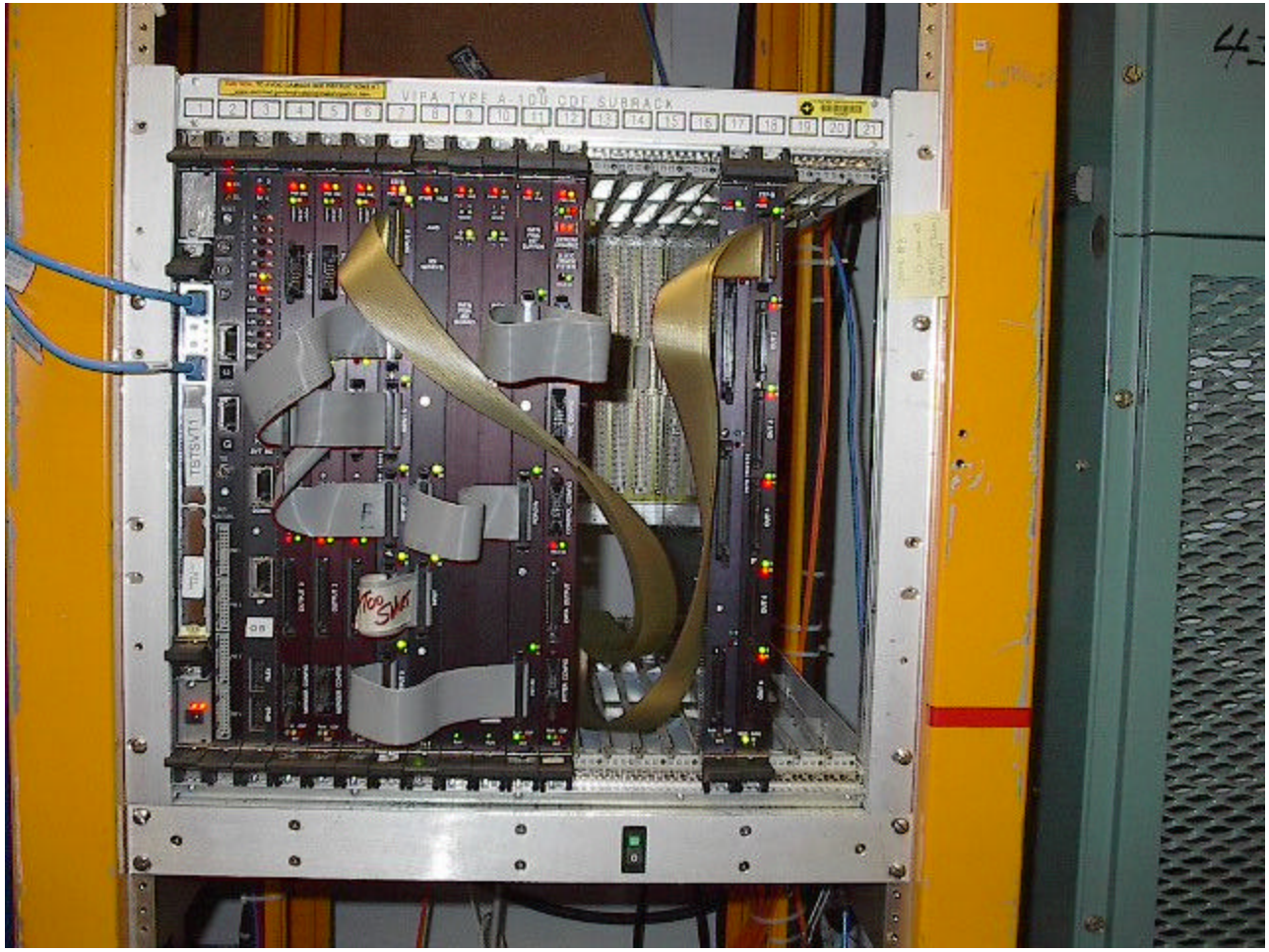
SVT emulation software (svtsim)

- SVT is a fully digital system ⇒ can emulate bit-for-bit
  - **but we don't emulate the beamline subtraction yet**
    - (expect to do it this month)
  - **and $ a few ladders that give high rate of bit errors (worth problem-stalking?)**
  - **and we need to do better at systematically monitoring SVT vs svtsim**
    - checks that database / book-keeping / download code is OK
      - **I recently made a booboo that went unnoticed for 3 weeks …**
      - **fortunately, this booboo is easily emulated**
    - checks that SVT hardware is OK; without it, we're flying blind
    - in our best laid plans, we will even do this in real time on the VME crate CPU!!
- Use for predicting physics signal rates beginning (A. Cerri, Roma group, et al)





B0 test stand keeps spares hot



We have a test stand in which a set of spare boards is continuously flowing fake data, checking against svtsim.

Two examples of debugging tools

- Using spy buffers to hunt for glitches
  - **During Tuesday evening's pandemonium, we did not manage to figure out why SVT was sometimes not sending data to L2**
  - **But the next day, we were able to go back and study the SVT spy buffer dumps and catch SVT in the act by examining BC sequence**

```
b0svt00/11 .. .. 21 76 56 06 47 4a
b0svt00/11 03 18 21 76 56 06 47 4a
b0svt00/11 03 18 21 76 56 06 47 4a
b0svt00/12 03 18 21 76 56 06 47 4a
b0svt00/20 .. .. .. 76 56 06 47 4a
b0svt00/20 03 18 21 76 56 06 47 4a
b0svt00/20 03 18 21 76 56 06 47 4a
b0svt00/21 .. .. .. .. 03 18 21 76
b0svt01/11 .. .. .. 76 56 06 47 4a
b0svt01/11 03 18 21 76 56 06 47 4a
b0svt01/11 03 18 21 76 56 06 47 4a
b0svt01/12 03 18 21 76 56 06 47 4a
b0svt01/20 .. .. .. .. 56 06 47 4a
b0svt01/20 03 18 21 76 56 06 47 4a
b0svt01/20 03 18 21 76 56 06 47 4a
b0svt01/21 03 18 21 76 56 06 47 4a
```

- Using test stand to understand difficult SVT/svtsim mismatches
  - **What we can see offline are SVT input (silicon & XTRP raw data) and output (SVTD bank)**
  - **The emulator can predict the output of each intermediate board**
  - **If real SVTD doesn't match emulated SVTD, we can go to the test stand, download the exact same configuration that Run_Control used, and compare all of the intermediate SVT stages' data with the emulator**
  - **This was used to figure out the booboo mentioned on an earlier page, where we had downloaded something that svtsim didn't handle properly**

Summary

- Overall, SVT is doing its job well
  - **Despite low luminosity, CDF has some new (high-xsec) signals to study**
- Work is ongoing to optimize tracking performance (workshops July, September)
  - **acceptance Ü our industrious SVX friends**
  - **efficiency Ü better patterns**
  - **more acceptance Ü use “1234” and “3-layer” wedges (reach 71/72 wedges?)**
  - **more efficiency Ü use “4/5” track-finding algorithm (buy back hit efficiency)**
  - **remove fit parameter biases, improve resolution, tighten C^2 distribution**
    Ü improve SVT alignment, perhaps try to import offline alignment data
  - **consider barrel-crossers as a further acceptance improvement?**
- Effort is ramping up to consider SVX+SVT timing improvements
  - **SVX group made some important changes already last spring, e.g. changing FIB logic to eliminate long tails in readout time**
  - **workshop this past Wednesday, likely follow-up soon**
  - **there are some ideas to squeeze a ms here or there; need reality checks**
- A few loose ends to tie up on operations
  - **problem detection, validation, documentation, reducing our DAQ nuisance rate**