CDF Offline

Kevin McFarland
Rochester/Fermilab
for the CDF Offline Group

CDF Week

May 31, 2001

---

1. Reconstruction Overview
2. Infrastructure Issues
   - User Support
   - Code Release Management
   - Calibrations
   - Build/Link Issues
   - I/O
3. Performance Metrics
   - Examples

Reconstruction Scorecard

Good First Pass
Needs Work
Just Starting

	D→"E"	Calib./Align.	Algorithms	Transient Rep.	PAD Rep.
Calorimeter					
COT					
Muons					N/A?
Strips/Wires					
Si					

CDF Offline Customer Relations

- `cdf_software_help` mailing list
  - ↪ Offline ACEs answer questions, coordinate documentation
  - ↪ Serves as offline "customer comment" mechanism
    - ⋆ Identifies issues most pressing issues
    - ⋆ Helps to set offline priorities

```
CDF_SOFTWARE_HELP archives - May 2001

23.Problem with CprWire::getSide (PR#308)
         Problem with CprWire::getSide (PR#308) (67 lines)
         From: Pasha Murat (630)840-8237@169G <murat@NCDF41.FNAL.GOV>
28.Pythia-MC-problem
         Pythia-MC-problem (23 lines)
         From: Carsten Rott <carott@PHYSICS.PURDUE.EDU>
         Re: Pythia-MC-problem (42 lines)
         From: A. Stan Thompson <thompson@A5.PH.GLA.AC.UK>
67.getting started with offline
         getting started with offline (38 lines)
         From: James Russ <russ@CMUHEP2.PHYS.CMU.EDU>
         Re: getting started with offline (39 lines)
         From: DongHee Kim <dkim@FNAL.GOV>
85.no ces
         no ces (138 lines)
         From: Steve Kuhlmann <stk@CDF.HEP.ANL.GOV>
         Re: no ces (152 lines)
         From: Marjorie Shapiro <Mdshapiro@LBL.GOV>
93.problem with getting 36x36 data from tape (PR#351)
         problem with getting 36x36 data from tape (PR#351) (126 lines)
         From: Pasha Murat (630)840-8237@169G <murat@NCDF41.FNAL.GOV>
```

CDF Offline Customer Relations (cont'd)

Agenda for Computing Institutional Representatives Board Meeting

Thursday, May 31, 1:30-3:30pm, Auditorium

Organization and Introduction of Chair		(5 min)	
Code Distribution Report	C. DeBaun	(10 min)	
Central Systems Status	Task Force	(10 min)	(TBC)
Run II Tape Technology		(10 min)	
Report on GCC port	L. Sexton-Kennedy	(10 min)	
Computer Security/Kerberos Status		(10 min)	
Discussion	Chair		

* Network Bandwidth to Remote Institutions
* Trailer Computing
* AOB

Code Release Management

- Significant new initiative (Sexton-Kennedy)
  - ↪ Adopt *fixed* weekly timetable for "integration" releases, e.g., `3.16.0int2`
    - ⋆ Last month of integration releases available on central systems and for distribution
  - ↪ Package librarians (code experts) are identified by reconstruction SPLs who control content for each release
  - ↪ Designated integration releases (plus bug fixes) are promoted to frozen releases, e.g., `3.17.0`
- Why do you care?
  - ↪ Frozen releases are more frequent (less effort to assemble)
    - ⋆ Lessens reliance on `development` or patched releases
  - ↪ Release quality is higher (accountability)
  - ↪ Aids production and Level-3 integration
    - ⋆ Quasi-real time Production achieved
    - ⋆ Significant Level-3 reconstruction

Calibrations

- Major development on infrastructure for selecting calibrations to be used in analysis (Jack C., Jim K.)
  - ↪ Individual subsystems can mark calibrations as "valid" for a run or set of runs
  - ↪ These "valid" calibrations by sub-system can now be merged
  - ↪ *Crucial* for, e.g., incorporating stage 0 at Level-3 where the luxury of human intervention is not there
- Human part of the infrastructure: Rob Snihur (UCL) is offline calibration coordinator
  - ↪ Responsible for working with sub-system experts to ensure correct calibrations are used at Level-3 and in Production
- Calibration Export
  - ↪ Review in January endorsed freeware database solution for export to remote sites
  - ↪ Implementation work ongoing (but behind schedule)
  - ↪ Meantime, remote access to Oracle database is still valid *(performance for UK, Italy, Japan?)*

Building and Linking Issues

- It would be safe to assert that most people are aware that there are serious "quality of life" issues resulting from performance here
- gcc vs KAI
  - ↪ We have completed substantial work on a port to gcc, but concluded we have to wait for gcc3.0, "first half of 2001" (see L. Sexton-Kennedy talk in Comp. Rep. meeting)
  - ↪ Performance under gcc will not necessarily be adequate
  - ↪ KAI licenses now site-wide at FNAL
- Dynamic loading (Ashmanskas, Calafiura, Sexton-Kennedy)
  - ↪ The most time consuming step in assembling binaries is linking
  - ↪ Can avoid linking repeatedly when developing code by dynamically linking user modules
  - ↪ Infrastructure in and mostly functioning
    - ⋆ Working on reliability problems seen under IRIX
  - ↪ Example: modifying `ExampleTrackAnalysis` PII/400MHz/512MB, NFS disk (Tony Vaiciulis)
    - ⋆ Standard Build: Recompile 55 sec, Relink 190 sec
    - ⋆ Dynamic Build: Recompile 64 sec, no Relink required

Building and Linking (cont'd)

- Reducing Infrastructure Code
  - ↪ We have been linking in a fair amount of code not being used by most users
    - ⋆ Extra database interfaces (textDB, Oracle OCI) not typically used
    - ⋆ Unused GEANT4 code
  - ↪ Recent work by Joe Boudreau to remove G4 will result in 5% decrease in production exe size
  - ↪ Similar results per database interface (Jim Kowalkowski)
- Reducing Symbols
  - ↪ Debugging symbols *dominate* size of executable
  - ↪ Size of executable has a significant effect on link speed
  - ↪ Working to develop ways to link debug-symbol free infrastructure libraries against debug-laden user code

	Default Link Time	Opt/NoDebug Link Time
`ProductionExe`	6:26	2:39
`CdfSim`	13:42	2:51

(Marjorie Shapiro) Linux PC/256MB

Building and Linking (cont'd)

- Understanding `fcdfsgi2` performance
  - ↪ Link speed on `'sgi2` is dramatically slower than, say, my laptop
  - ↪ This has been investigated, but not understood
  - ↪ We have negotiated with D0 to increase our fraction of Jim Kowalkowski (CD "C++ guru") to address these performance issues
  - ↪ Will the new Sun have similar "issues"?
    Paul Keener (Penn) has offered to donate time to investigate performance of new Sun central SMP

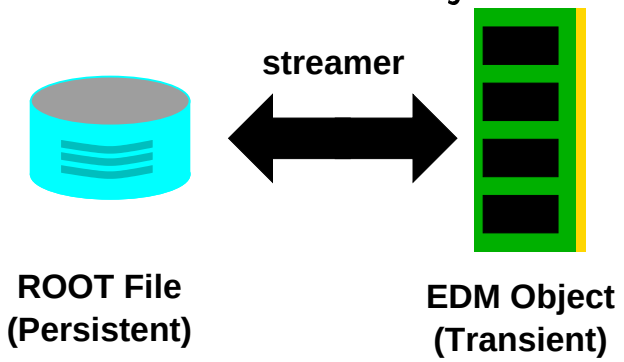
"I/O" Performance

- A common complaint is:

  "DHInput takes too much #!&*% time to run!"

  And, yes, offline management is aware of this. . .

- Here's the explanation:
  - ↪ Native ROOT I/O speed is fast, $\gtrsim 20$ MB/s
    - ⋆ this is, after all, how we log data!
  - ↪ "streamers": how objects are read into memory



    - ⋆ Reconstructed objects, e.g., `CdfTrack`, are optimized for their transient (not persistent) representation
    - ⋆ The streamer is essentially unable to do block transfers for such a complicated data structure
      (but streaming of StorableBanks is fast!)
    - ⋆ Reconstructed objects are big, and there are lots of them
  - ↪ "post-read" / "pre-write": housecleaning tasks
    - ⋆ But often they do a full spring cleaning instead of a mild dusting

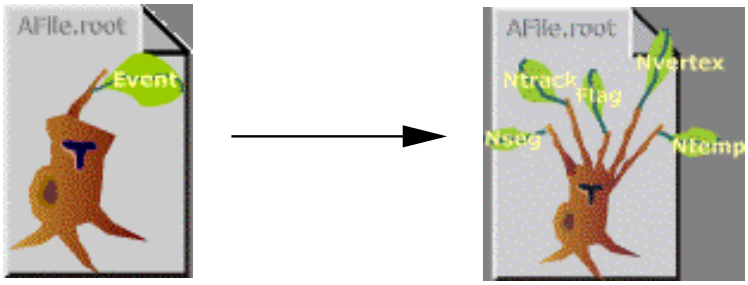
"I/O" Solutions

- PADs
  - ↪ One could propose to clean up the persistent form of our reconstructed objects
    - ⋆ but that's a bit silly when we know we don't want to read/write those anyway
  - ↪ Production writes PADs $\Longrightarrow$ *focus is on design of PAD objects and on optimizing their performance* (Yagil talk, Wednesday pm)
- Example: `QTRK` replacement (Ivan Furic)

	Time per event (msec)	
	`CdfTracks+LRIH`	`QTRKs+LRIH`
I/O	17	2.7
Puffing	n/a	1.5

"I/O" Solutions (cont'd)

- Multi-branch ROOT I/O
  - ↪ ROOT supports a feature of the file structure called "branches"
  - ↪ I/O can be performed on each branch *independently*
  - ↪ Our events can be stored in multiple branches



- Analysis examples:

  A An event skim that wants to form a new dataset by L1/L2/L3 triggers
  - (a) Read in "header branch" ($\sim 1\%$ of event)
  - (b) Make trigger selection
  - (c) Write out dataset
    - ↪ Dominated by tape staging and output speed, not reading!

  B Need to redo jet clustering in calorimeter with new algorithm
  - (a) Read in "calorimeter branch" ($\sim 10\%$ of event)
  - (b) Redo clustering; replace PAD objects
  - (c) Write out dataset

"I/O" Solutions (cont'd)

- Status of multi-branch ROOT I/O
  - ↪ Prototype example exists (Fedor Ratnikov)
  - ↪ Design stage for implementation in EDM (Kennedy) / Framework (Sexton-Kennedy)
  - ↪ Organizational stage for PADs group (Yagil/Rolli)
    - ⋆ this last work will need approval by physics groups, TDWG

Analysis Benchmarks

- Access to the **LATEST** data

Data Handling:

↪ Latency for all raw data to be available to users through the tape system is approximately 24 hours

↪ Latency for all data through production is $\gtrsim 48$ hours

↪ LOOK area files (~ 5000 events per run per stream)

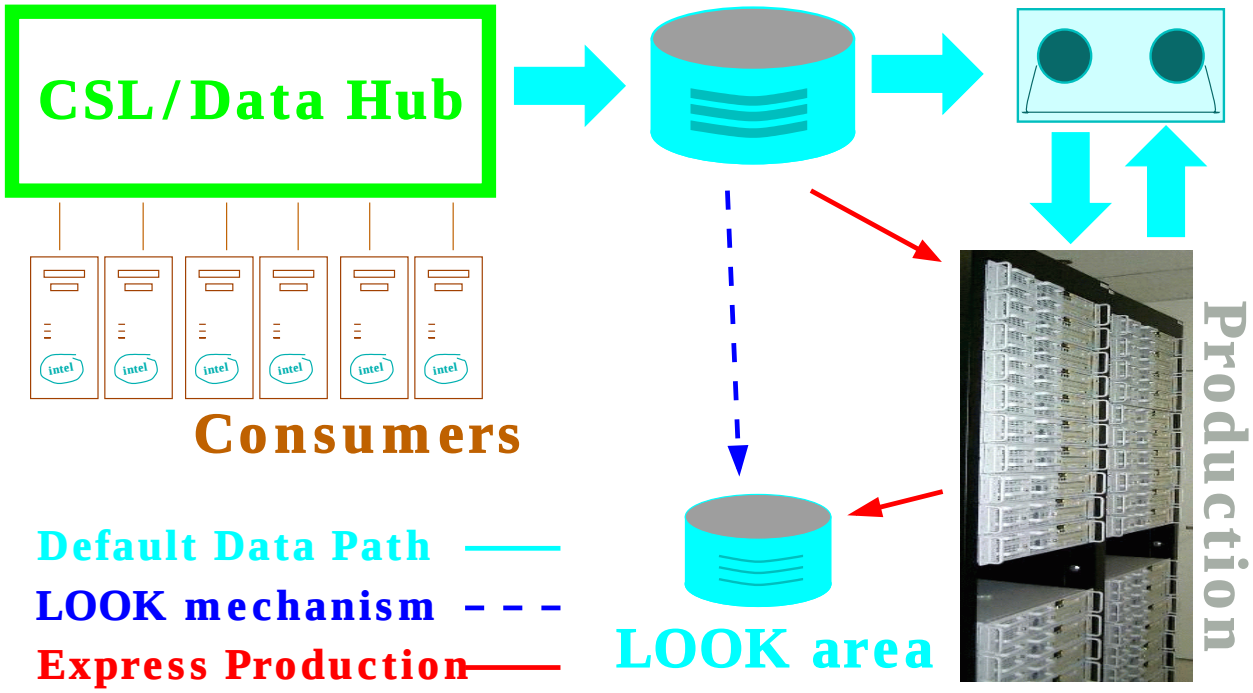
Reconstruction/Analysis:

↪ Currently it is slow to spin through raw data or production output, pending PADs/multi-branch

- ⋆ "I/O" currently limits rate to ~ 1–2 MB/s per job
- ⋆ c.f., re-tracking COT, ≈ 0.6 MB/s

Analysis Benchmarks (cont'd)

- Express Production



↪ For Stream A (3%) of data, reduce latency of production to approximately 8 hours

↪ Purpose is to monitor detector, data-taking, physics rates

↪ Datasets in this Express stream will *not* feed physics analyses

↪ TDWG/Physics groups working to define content

↪ Hope to be running in July
(caveat: details of output TBD)

Analysis Benchmarks (cont'd)

- Re-analyzing 10^7 events ($\sim 0.5\%$ of Run IIa) to produce a smaller dataset or N-tuple
  - ↪ Requires PADs (3 TB raw $\rightarrow 1$ TB PAD input data)
  - ↪ Tape reading speed (single drive) is 2–3 days
  - ↪ Retracking COT would require ~ 2 CPU-months (single CPU)
  - ↪ Skimming data *NOW* would require ~ 3 CPU-weeks!
    - ⋆ Highlights need for multi-branch events/PADs

Conclusions

- Many successes of the offline
  - ↪ Reconstruction algorithms and infrastructure basically functioning
  - ↪ Level-3 and Production running in real time
  - ↪ Data handling from tape enabled,
    PLEASE EXERCISE!
- Much to do
  - ↪ Improving speed of data access
    - ⋆ PADs, multi-branch ROOT
  - ↪ Improving compile/link speeds
    - ⋆ Many efforts underway
  - ↪ Will be bringing up infrastructure for secondary/tertiary datasets in summer
    (Litvintsev, Watts, Wednesday pm)