

Persistence and Data Handling Models

David Brown, LBNL

SuperB Computing Workshop
10 March 2010

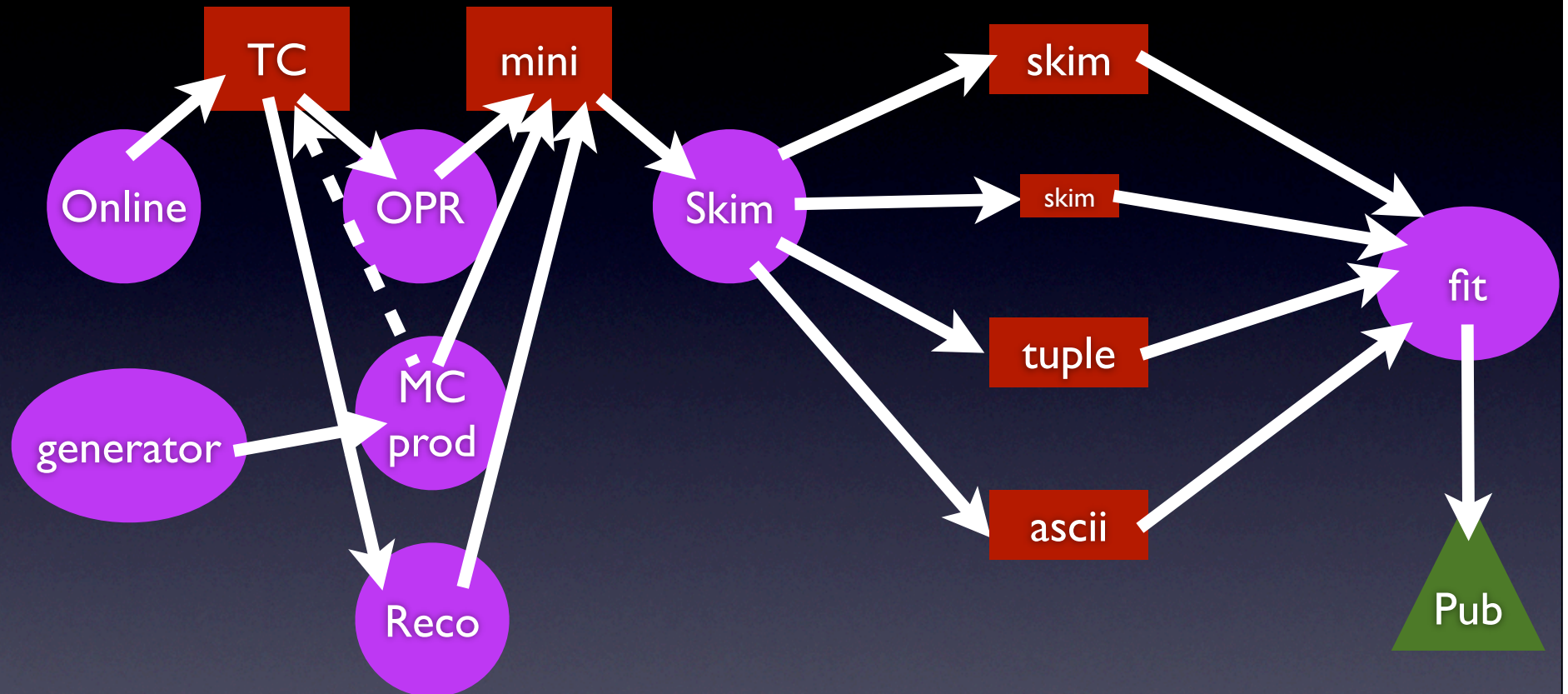
Definitions

- Persistence
 - Read/write bits to/from 'disk' (NVM)
 - Interface between transient objects and NVM
 - schema evolution, platform compatibility, staging, compression, ...
- Data Handling
 - Procedural model of data transitions
 - Supported data structures
 - Data storage and movement strategy
 - NOT storage technology

Persistence/data R&D issues

- Persistence technology (technologies?)
- Interface with transients
- dynamic conversion?
 - smart pointers
- Interface with detector (online)
- Direct access?
- Files structure
- Processing model
- Interface with database (bookkeeping)
- Interface with framework

SuperB Processing?



Central
Computing

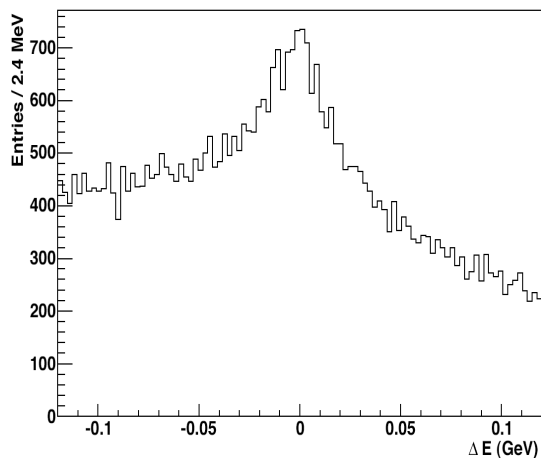
Analysis

BaBar Persistence

- root based
- Trees = 'components'
 - Hdr, Tag = event header, coordination
 - Raw, Sim = low level data
 - Esd = processed detector data (hits)
 - Aod = high-level reco objects (tracks, clusters)
 - Cnd = 4-vectors ,vertices, errors, genealogy, + PID
 - Usr = user-customized cache of event, candidates
- Trees can be written to same or different files
 - mini = Esd + micro
 - micro = Hdr, Tag, Cnd, Aod
 - skim = micro + Usr, or mini + Usr

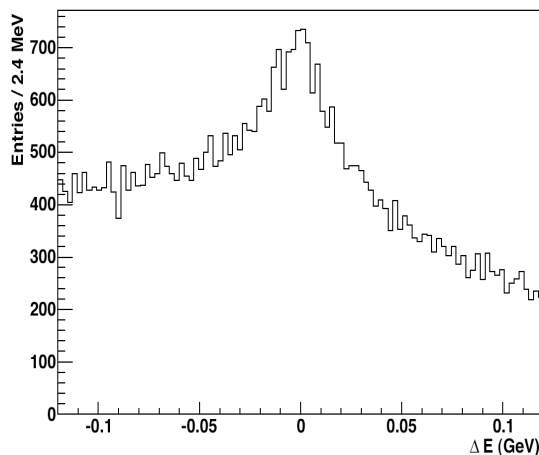
BaBar Model Access

ΔE for $B \rightarrow D^{(*)} X$ “Semi Exclusive” Selection
43,000 Selected Candidates from 2.5 M Events



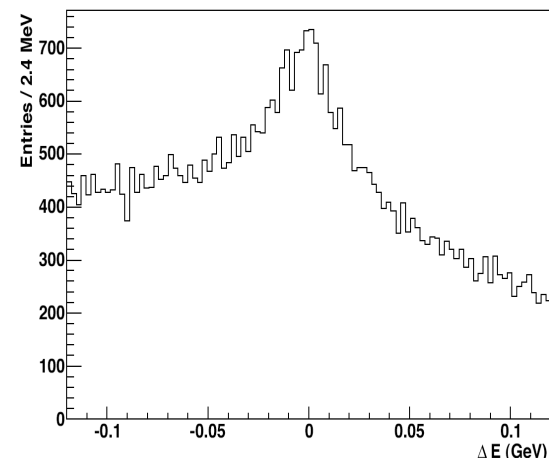
10,000 sec.

Rerun combinatorics
and compute ΔE



1,000 sec.

Compute ΔE from
persistent composites



10 sec.

Interactive plot
of `UsrData` ΔE



Plots by Eric Charles

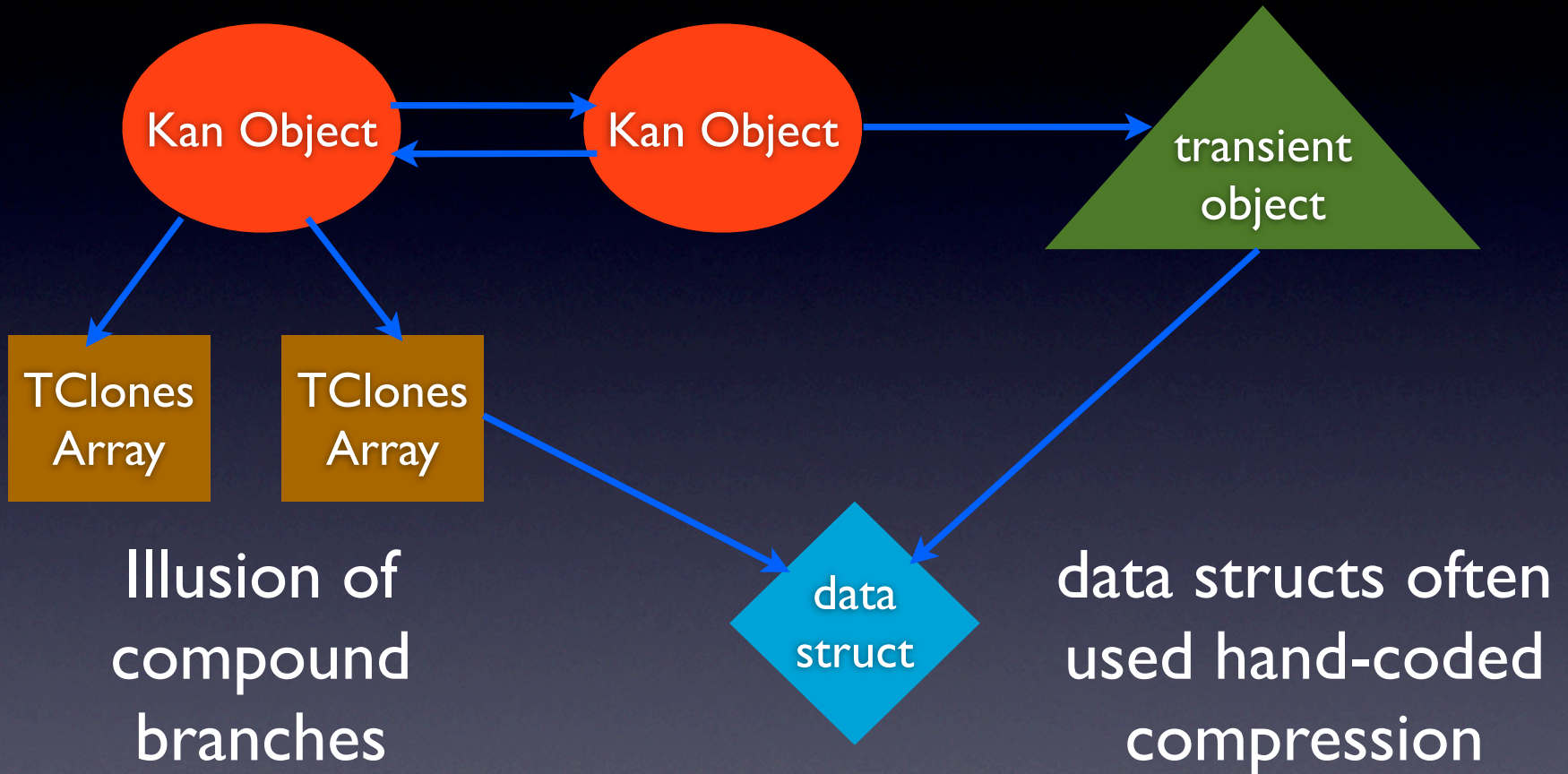
David Nathan Brown

BaBar Collaboration meeting

Dec. 8, 2003



BaBar object model



Dynamic Conversion (DNA)

- Dynamically read + process data as requested
 - Requires detailed dependency knowledge
- Transient version used successfully in analysis
 - SimpleComposition (candidate lists)
- Persistent version used very little in CM2
 - ‘track promotion’
- Can improve processing efficiency
- Introduces unpredictable resource load
 - mitigate through resource management?