Status of the Current STAR Tracker

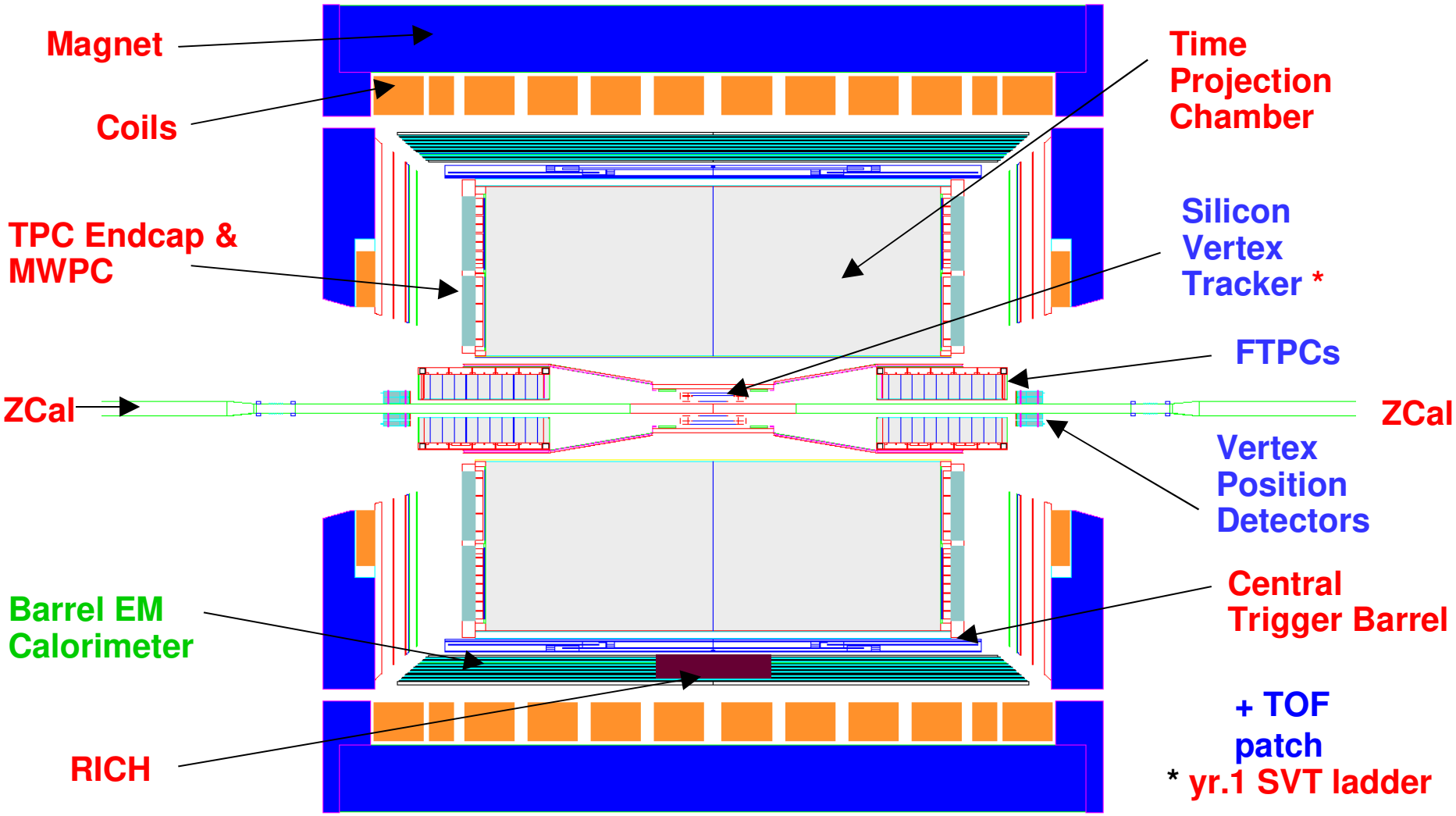
Lee Barnby

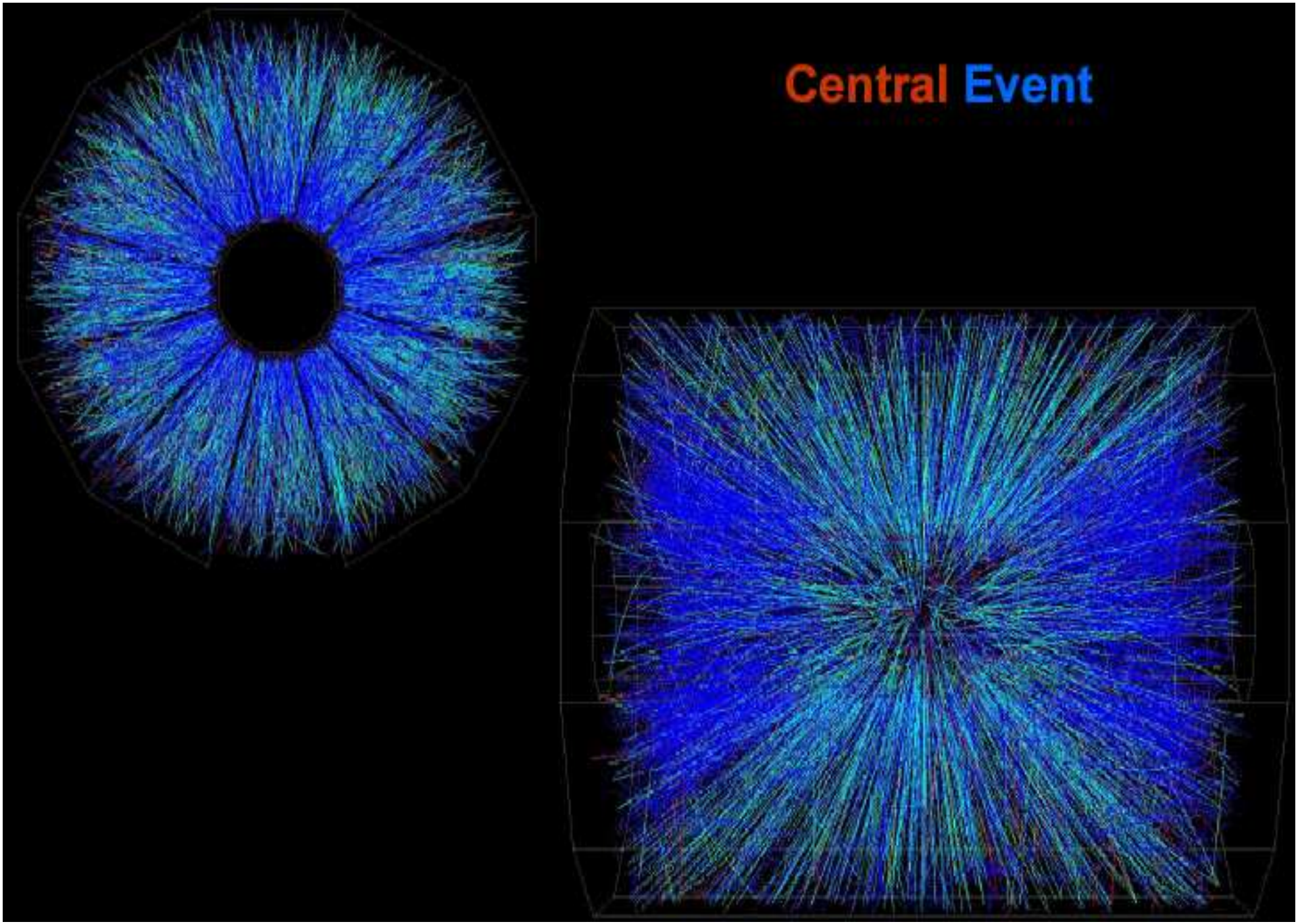
KENT STATE UNIVERSITY

Introduction

- STAR introduction
- Overview of current packages
- TPC tracker
- Global tracking
- Infrastructure, stability & performance issues

The STAR Detector




Central Event

Jargon – terms you will meet

StEvent: C++/OO event model, star DST format

Maker: C++ reconstruction (or analysis) package

PAM: FORTRAN based reco. code from pre-ROOT days. PAMs are called as functions from Makers

Tables: old DST format, current means of communication between PAMS/Makers. Tables are currently used to fill StEvent.

Basics – pre-tracking

- **Cluster finding is performed before tracking**
  - This will not be discussed in much detail as is common to old and new trackers
  - TPC, SVT, FTPC have own cluster finders (most natural)
  - Space points are produced and stored in tables, later can be put into StEvent (as StHit) although not usually persistent.

Tracking strategy

- TPC is main tracking device so TPC tracking comes first
  - tpt: track finding and helix fit
- Global tracking tasks follow
  - egr: optionally add SVT hits (using est) and do Kalman refit
  - Vertex finding with evr
  - egr: fit tracks to primary vertex

Tracking strategy cont.

- **FTPC is independent since phase space does not overlap.**
  - Uses own separate (C++) code StFtpcGlobalMaker, StFtpcPrimaryMaker, StFtpcTrackMaker but writes into same format (tables).
- **No standalone SVT tracking is performed**
  - SVT not so far used for physics production

Post-tracking

Tracks the used for:

- strangeness reconstruction V0, cascade, kink
- Extrapolation and matching to non-tracking outer detectors (RICH, EMC)
- Particle ID via dE/dx calculation

These tasks are not discussed further

TPT – the TPC tracker

Algorithm

- Proceeds in identical(?) 2 passes
- After 1st pass hits used are removed from consideration
- Each pass:
  - Divides TPC into (overlapping subvolumes)
  - Forms segments/track ‘roots’
  - Extends segments
  - Merge Spirals

TPT tracking pass

- Free points identified as tracks seeds
- Extended to connecting hits (outer sector)
  - Low momentum inner sector hits don't overlap (TPC geometry) so seed followed toward vertex (z=0 assumption)
- Two hits make linear prediction to add 3rd (no assumption now)
- Roots from a point extended and best segment kept according to χ^2
- Gaps allowed (dead FEE/ sector boundaries) but limited in size

TPT track fit

Fast 2 component helix fit

- Circle fit in x-y plane
- Line fit in s-z co-ordinates (s = path length)
- No vertex constraint

Global Tracking

- Main components are Kalman fit and primary refit (both are saved to DST) and optional adding of SVT points to track
  - Kalman fit allows for deviation from helix due to average energy loss in TPC gas for global track and inner field cage + beampipe for primary
  - Points may be (are) removed from fit
  - Pion hypothesis used

SVT tracker - EST

- Projects TPC tracks to SVT ladders
- Proceeds in several passes starting with higher p_T tracks first
  - Successful projections remove hits
  - Can also be used with SSD – so called 4th layer (hardware not installed yet)
- Status: ready to use in pp data production

Primary Track Fit

- Constrained fit where track is forced through vertex
  - OK for Au+Au, probably not a good strategy for pp?
  - Applied to all tracks with DCA<3cm
    - Cut off should really depend in track parameters

Infrastructure (problems)

- Most code (tpt, egr) is in FORTRAN.
  - Difficult to debug currently (and in future?)
  - Expertise is waning
- Often (at least for egr) tracks have several representations the deeper down the code you go.
- Some service tasks (e.g. updating dca) done in C++
  - Requires temporary creation of StTrack (StEvent component) which is then used to fill table, which ultimately used to create StTrack objects!

Stability

- TPT had no major changes since STAR started taking data
- EGR has been stable for ~18+ Months
  - Some upgrades tried but not implemented, e.g. alternative PID hypotheses
  - At least one known bug where exact track parameters depend on order events run
    - fortunately does not affect physics
  - SVT tracking parameter tuning has been going on as part of SVT software+calibration 'commissioning'

Performance

- No detailed assessment here
  - I am hoping each individual compares current and ITTF
- Generically efficiencies from embedding (tracking + cluster finding) are 70-80% with small dependence on p_T once maximum reached.
- Time-wise a central Au+Au event takes a few cpu minutes for *all* reconstruction.

Summary

Results are good and acceptable for physics. However, infrastructure is inefficient and (lack of) design, especially for global tracking part, making new development almost impossible.

The most exciting phrase to hear in science, the one that heralds new discoveries, is not 'Eureka!' (I found it!) but 'That's funny ...'

Isaac Asimov