

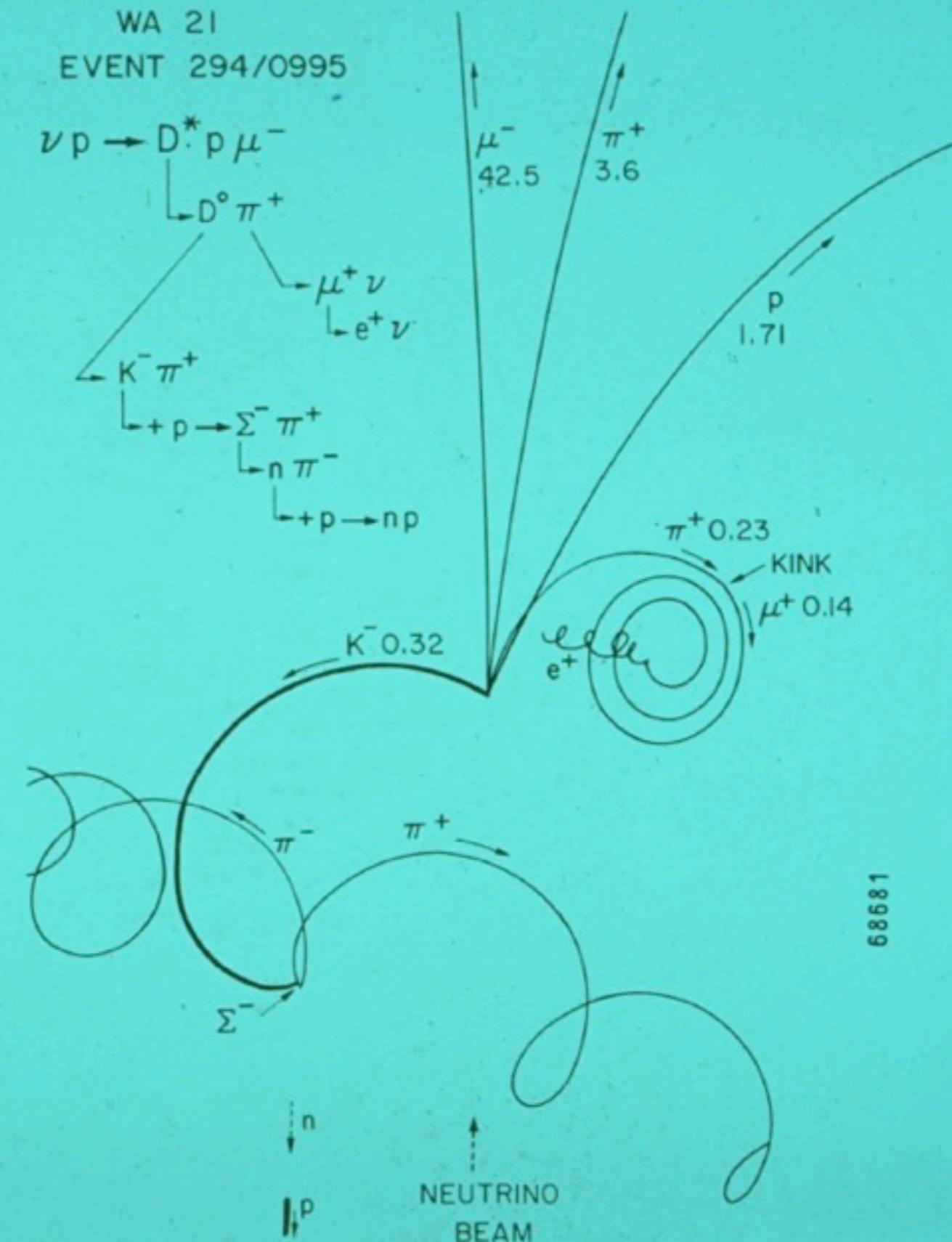
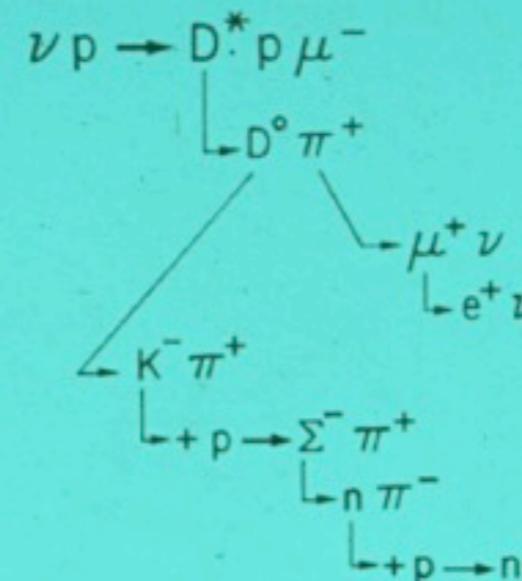


# Tracking

Eugenio Paoloni

AACHEN-BONN-CERN-MUNICH-OXFORD COLLABORATION

WA 21  
EVENT 294/0995



MOMENTUM IN GeV/c

# Talk Outline

- ◆ Una breve introduzione pedagogica ad uso degli sviluppatori dei tools di analisi
- ◆ Novità sul fronte tracking dall'ultimo meeting Belle2 Italia
  - ◆ Abbiamo salutato Trasan
  - ◆ Niente più assistenza Monte Carlo nel merging

Tracking



2

Roma dicembre 2015

# Tracking MDST Objects

- ◆ La ricostruzione delle tracce cariche fornisce due StoreArrays scritti negli mdst:
  - ◆ `StoreArray<Track>` contiene le informazioni riguardanti le tracce cariche.
  - ◆ `StoreArray<V0>` contiene le informazioni riguardanti Ks, Lambda e conversioni di fotoni.  
Questa lista viene digerita e proposta all'utente dal ParticleLoader.

# mdst/dataobjects/Track.h

## Public Member Functions

**Track ()**

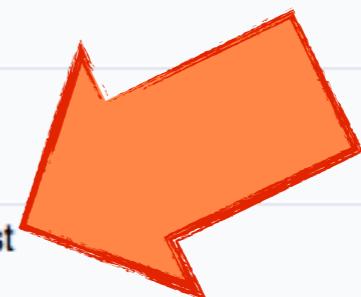
Constructor without arguments; needed for I/O. More...

**~Track ()**

Destructor.

**const TrackFitResult \* getTrackFitResult (const Const::ChargedStable &chargedStable) const**

Access to TrackFitResults. More...



**void setTrackFitResultIndex (const Const::ChargedStable &chargedStable, short index)**

Set an index (for positive values) or unavailability-code (with negative values) for a specific mass hypothesis.

More...

**unsigned int getNumberOfFittedHypothesis () const**

Returns the number of fitted hypothesis which are stored in this track. More...

1 per ora

**void addRelationTo (const RelationsInterface< BASE > \*object, double weight=1.0) const**

Add a relation from this object to another object (with caching). More...

**void addRelationTo (const TObject \*object, double weight=1.0) const**

Add a relation from this object to another object (no caching, can be quite slow). More...

MC info

**void copyRelations (const RelationsInterface< BASE > \*sourceObj)**

Copies all relations of sourceObj (pointing from or to sourceObj) to this object (including weights). More...

# Perché Più Ipotesi Di Massa?

- ◆ in un futuro (prossimo) miglioreremo la ricostruzione applicando diverse ipotesi sul tipo di particella:
  - ◆ per convertire il tempo dell'hit nella CDC in distanza occorre sapere in che istante la particella entra nella cella, a parità di impulso la velocità scende all'aumentare della massa.
  - ◆ Modello della perdita di energia (anche se al momento gli elettroni ci vengono meglio se trattati come pioni).

# mdst/dataobjects/TrackFitResult.h

TVector3 [\*\*getPosition \(\) const\*\*](#)

Getter for vector of position at closest approach of track in r/phi projection. More...

TVector3 [\*\*getMomentum \(\) const\*\*](#)

Getter for vector of momentum at closest approach of track in r/phi projection. More...

TLorentzVector [\*\*get4Momentum \(\) const\*\*](#)

Getter for the 4Momentum at the closest approach of the track in the r/phi projection. More...

double [\*\*getEnergy \(\) const\*\*](#)

Getter for the Energy at the closest approach of the track in the r/phi projection. More...

double [\*\*getTransverseMomentum \(\) const\*\*](#)

Getter for the absolute value of the transverse momentum at the perigee.

TMatrixDSym [\*\*getCovariance6 \(\) const\*\*](#)

Position and Momentum Covariance Matrix. More...

**Const::ParticleType** [\*\*getParticleType \(\) const\*\*](#)

Getter for ParticleType of the mass hypothesis of the track fit. More...

int [\*\*getPDG \(\) const\*\*](#)

Getter for PDG of the mass hypothesis of the track fit. More...

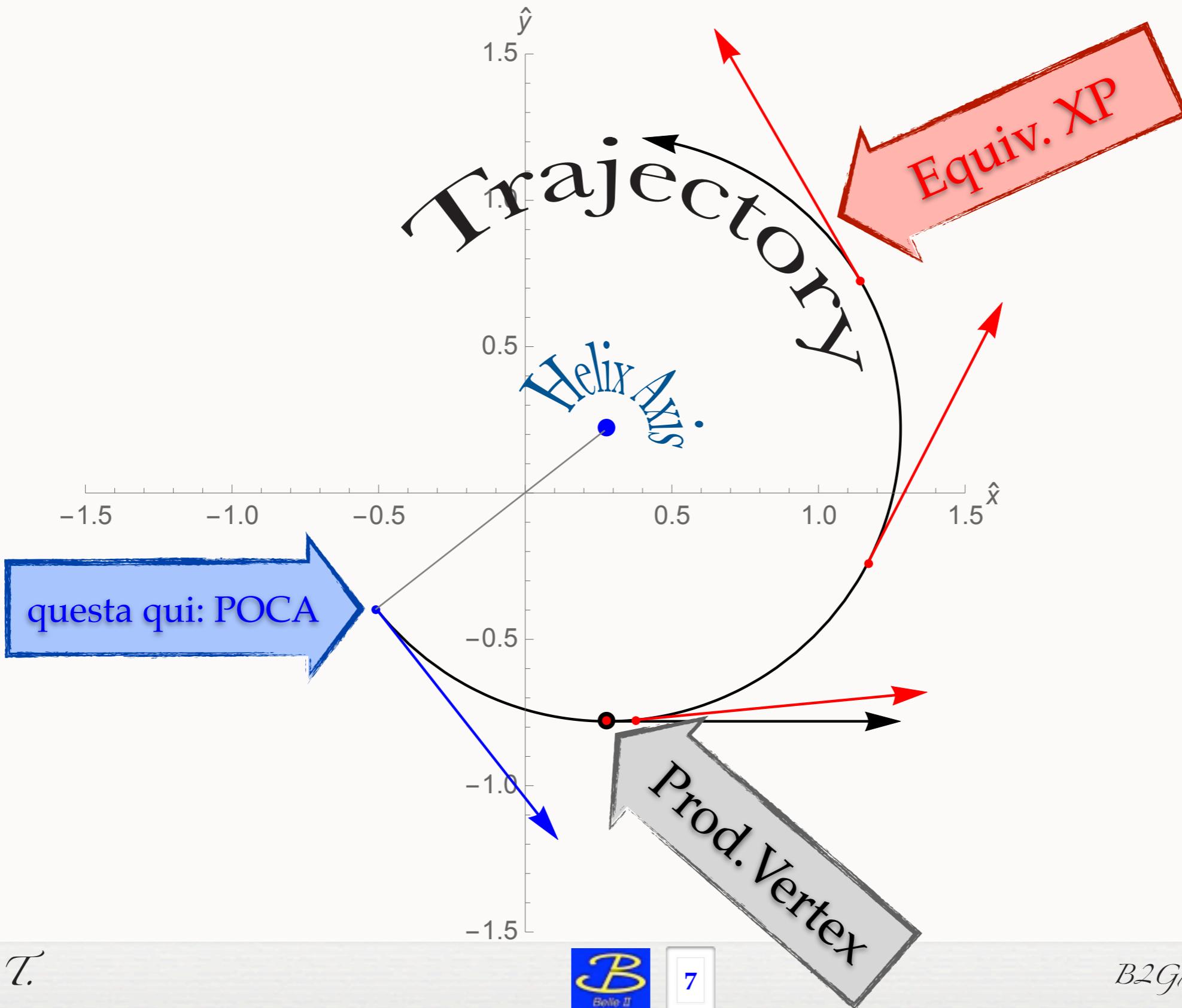
short [\*\*getChargeSign \(\) const\*\*](#)

Return track charge (1 or -1). More...

double [\*\*getPValue \(\) const\*\*](#)

Getter for Chi2 Probability of the track fit. More...

# GetPosition (Si Ma Quale?)



# mdst/dataobjects/TrackFitResult.h

double [getD0 \(\) const](#)

    Getter for d0. [More...](#)

double [getPhi0 \(\) const](#)

    Getter for phi0. [More...](#)

double [getPhi \(\) const](#)

    Getter for phi0 with CDF naming convention. [More...](#)

double [getOmega \(\) const](#)

    Getter for omega. [More...](#)

double [getZ0 \(\) const](#)

    Getter for z0. [More...](#)

double [getTanLambda \(\) const](#)

    Getter for tanLambda. [More...](#)

double [getCotTheta \(\) const](#)

    Getter for tanLambda with CDF naming convention. [More...](#)

std::vector< float > [getTau \(\) const](#)

    Getter for all perigee parameters. [More...](#)

std::vector< float > [getCov \(\) const](#)

    Getter for all covariance matrix elements of perigee parameters. [More...](#)

TMatrixDSym [getCovariance5 \(\) const](#)

    Getter for covariance matrix of perigee parameters in matrix form. [More...](#)

Helix [getHelix \(\) const](#)

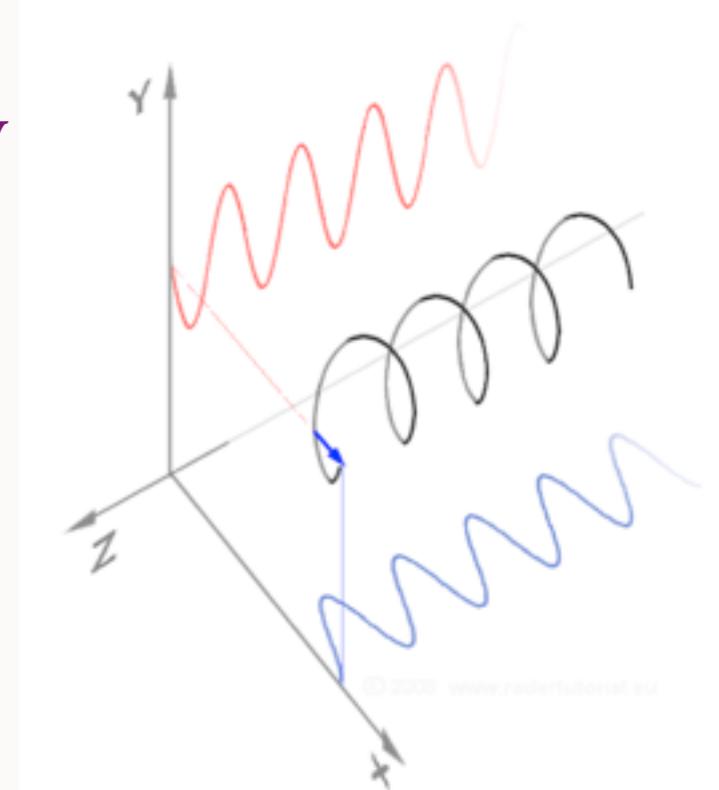
    Conversion to framework Helix (without covariance). [More...](#)

UncertainHelix [getUncertainHelix \(\) const](#)

    Conversion to framework Uncertain Helix (i.e., with covariance). [More...](#)

# Helix Parameters: MDST

- ◆ P.O.C.A. = point of closest approach to the z axis and closest to the xy plane.
- ◆ We do store these 5 parameters (in my favorite order):
  - ◆  $\varphi_0 \in [-\pi, +\pi]$ : is the angle defined by the  $p_t$  at the P.O.C.A. and the x axis
  - ◆  $\omega \in [-\infty, +\infty]$ : is the inverse of the radius of curvature signed with the assumed charge of the particle



```
#define M_PI           3.14159265358979323846264338327950288 /* pi */  
#define INFINITY       HUGE_VALF
```

# Helix Parameters: MDST

- ◆ We do store (part II):
  - ◆  $\tan \lambda \in [-\infty, +\infty]$ : the tangent of the angle defined by the momentum at the P.O.C.A. and the xy plane (a.k.a. dip angle)
  - ◆  $z_0 \in [-\infty, +\infty]$ : the z coordinate of the P.O.C.A.
  - ◆  $d_0 \in [-\infty, +\infty]$ : the signed distance of the P.O.C.A. with respect to the z axis
- ◆ Questions from the audience:
  - ◆ “Signed? Why have you decided to sign a positive number?”
  - ◆ “Signed? Which is the sign of the damned thing?”

# TRASAN



# The New CDC Track Finder



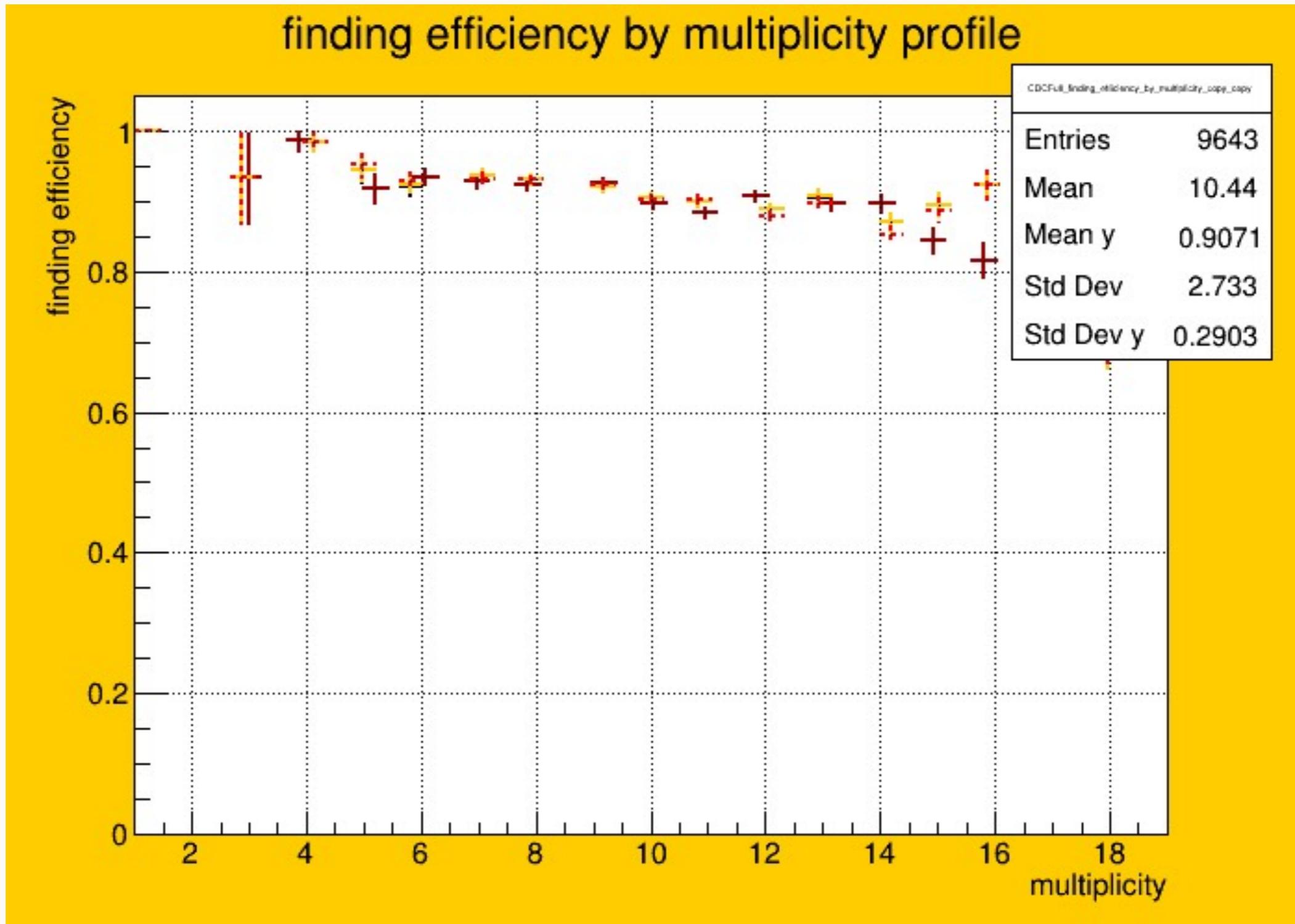
Tracking



# The New Track Finder

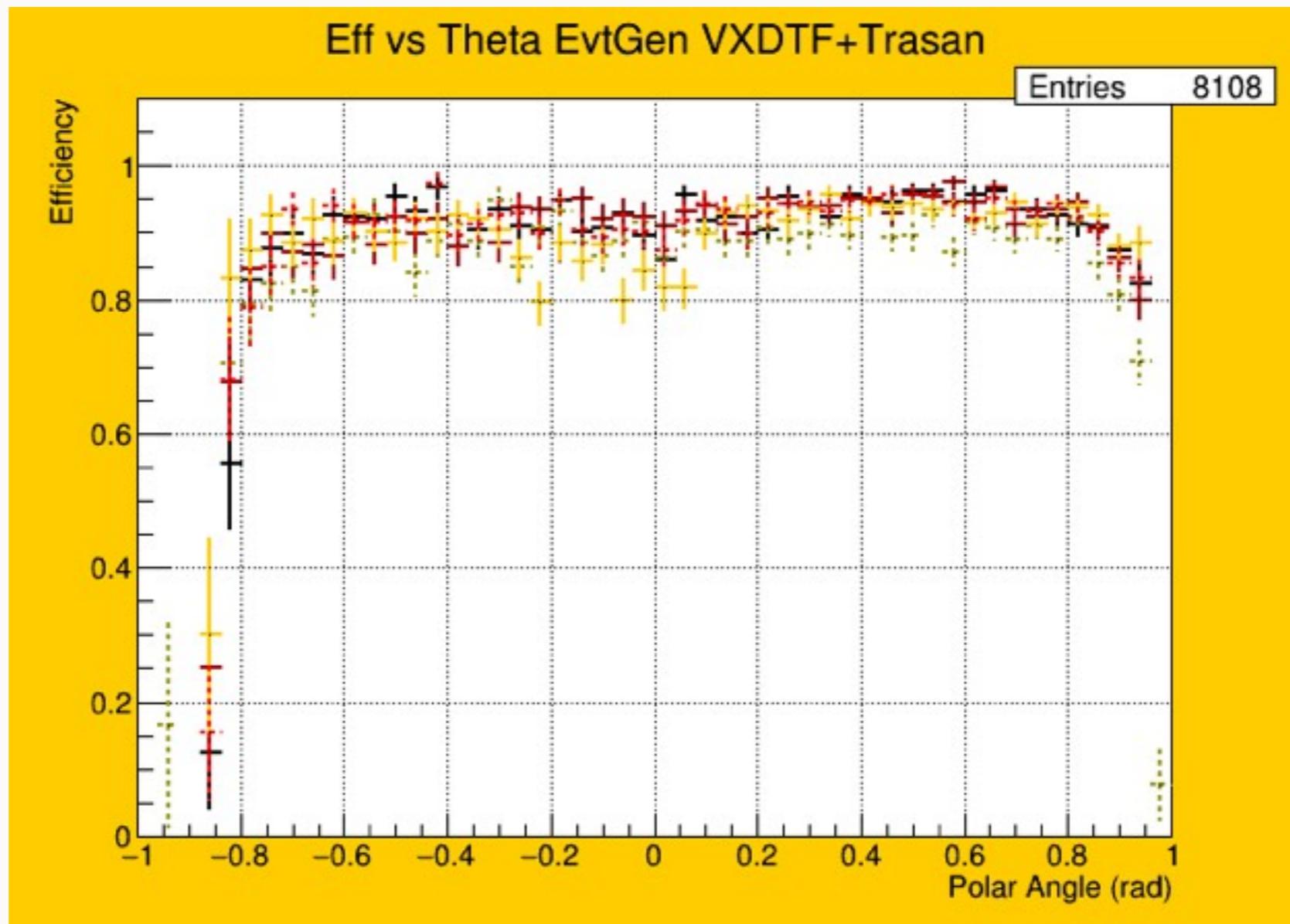


# The New Track Finder



# We Are Free!! (from MC)

- ◆ Al termine dell'ultimo B2GM ci siamo finalmente affrancati dall'MCTrackMerger (Grazie Ben)



Tracking

# Efficienza (Giulia)

efficiency  
& purity

## Integrated Efficiency

- this study is based on a sample of 10k Y(4S) generic decays reconstructed with the official standard reconstruction with the addition of the MC information

definition	in %	release-00-05-03	r22944*
$\frac{\# \text{MCParticles with at least one associated Track}}{\# \text{MCParticles}}$	physical efficiency	85.6±0.1	84.8±0.1
$\frac{\# \text{MCTrackCards with at least one associated Track}}{\# \text{MCTrackCards}}$	geom. accept. & det. ineff. factored out	94.0±0.1	93.3±0.1
$\frac{\# \text{MCTrackCard with at least one associated TrackCard}}{\# \text{MCTrackCards}}$	pattern recognition efficiency	96.6±0.1	94.5±0.1

- no striking changes in the standard tracking reconstruction in r22944 (MC-free Track Merging) w.r.t to release-00-05-03 (MC-assisted Track Merging)

Tracking

**GRAZIE**

**PER LA CORTESE  
ATTENZIONE!**

# Big Surprises! The Good One

## SVN History

### Finding Efficiency

