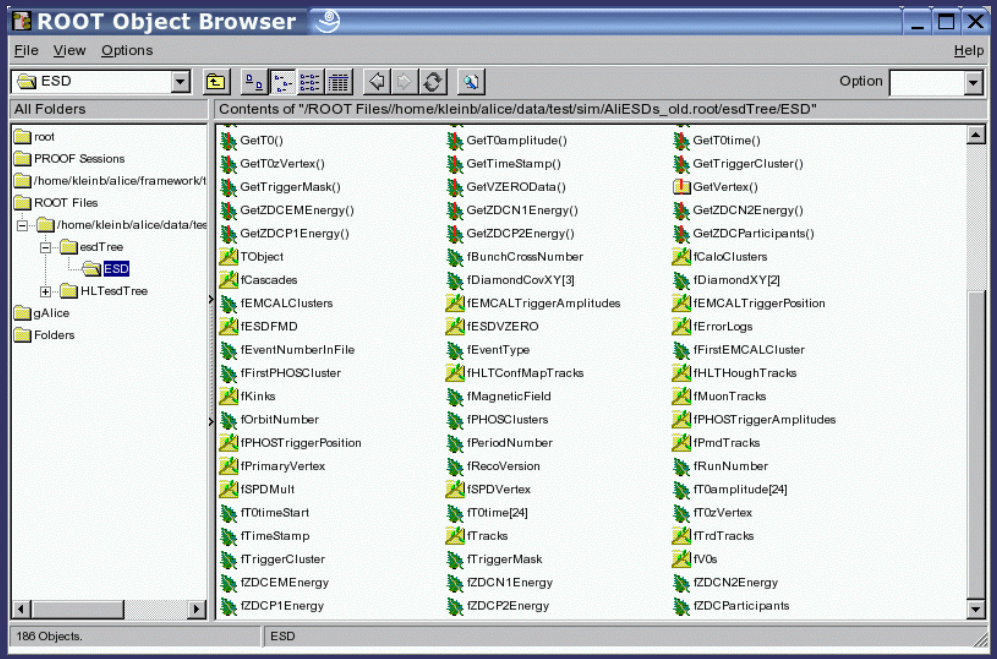
ESD Status

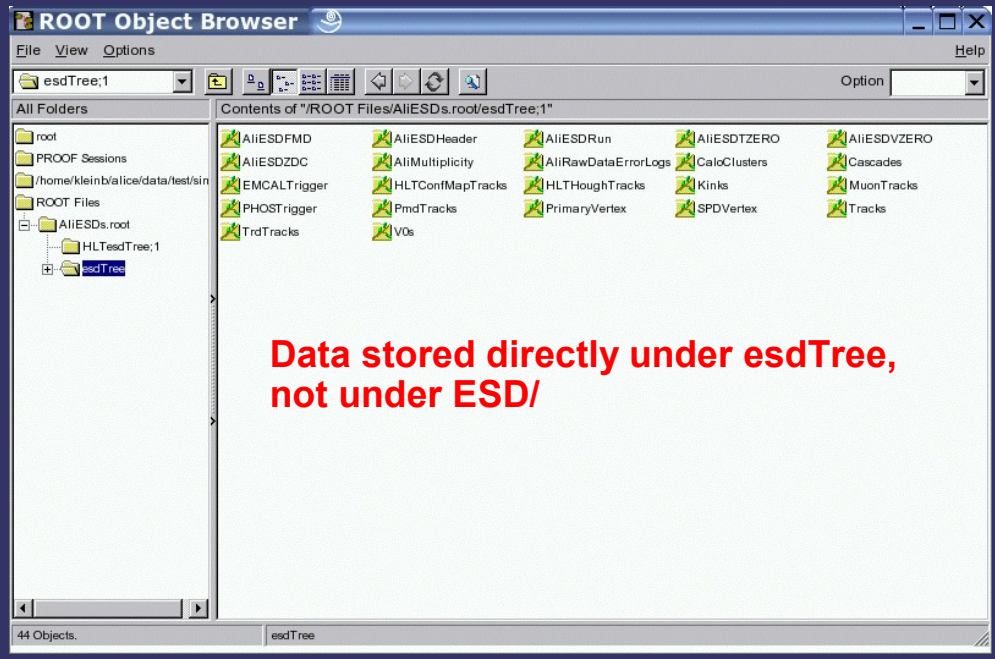
Offline Week
October 10th 2007

Christian Klein-Bösing

Reminder

- **AliESD has been replaced by AliESDEvent in July**
  - v4-06 first release with AliESDEvent
- **Content of AliESDEvent is managed by a TList**
  - Standard content tracks, clusters etc.
  - Only Branches, no single leaves





AliESDEvent 101

- **Writing:**

```
AliESDEvent *esd = new AliESDEvent();
esd->CreateStdContent();
esd->WriteToTree(tree);
```

creates tree structure

  - Usual TTree::Fill() etc.

- **Reading:**

```
AliESDEvent *esd = new AliESDEvent();
esd->ReadFromTree(fChain);

for(int i = 0;i<fChain->GetEntries();i++){
  Int_t nBytes;
  nBytes = fChain->GetEntry(i);
  cout << nBytes << endl;
  if(esd->GetAliESDOld())esd->CopyFromOldESD();
```

Sets the branch addresses depending on tree structure

Populate new structure from old AliESD

  - Backward compatible, with patched ESD.par even PDC06 data readable
  - Same for AOD-Reading/Writing, common base class AliVEvent (see Markus talk)

Let's talk about Size...

Sizes

ESD	CTDR	v4-04	v4-06
p+p/event	0.04	0.189	0.035
Pb+Pb/ev	2.5		6.9

*All sizes in MB

Different Approaches

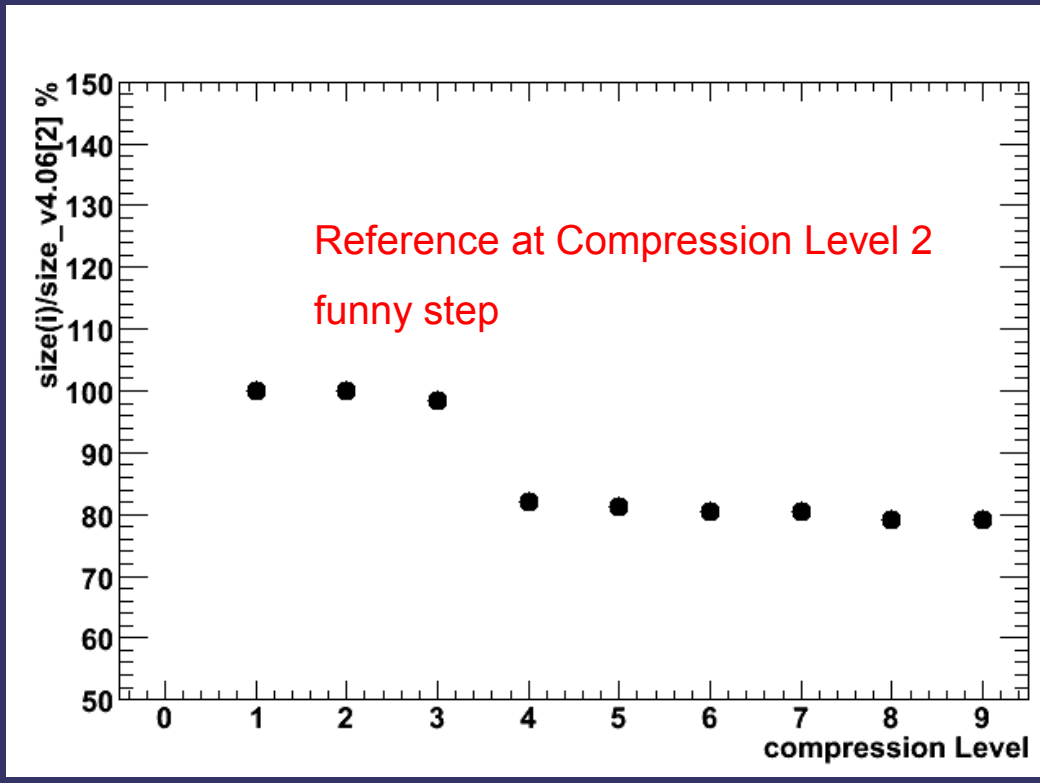
- **Reduce overall size of data**
  - Clean up: Remove variables/redundant information
  - Move from Int_t to Short_t or Char_t when possible
  - Move from Double_t and Float_t to Double32_t
- **Improve compression**
  - Use of Double32_t range and precision (with some reasoning...)
  - Grouping of data
  - Think about default values (0 is best)
- **Physics motivated clean-up (See Youri's talk)**
  - Remove e.g. certain tracks, V0s...

Nothing of the above affects the backward compability

(though “redundancy” is often a matter of opinion)

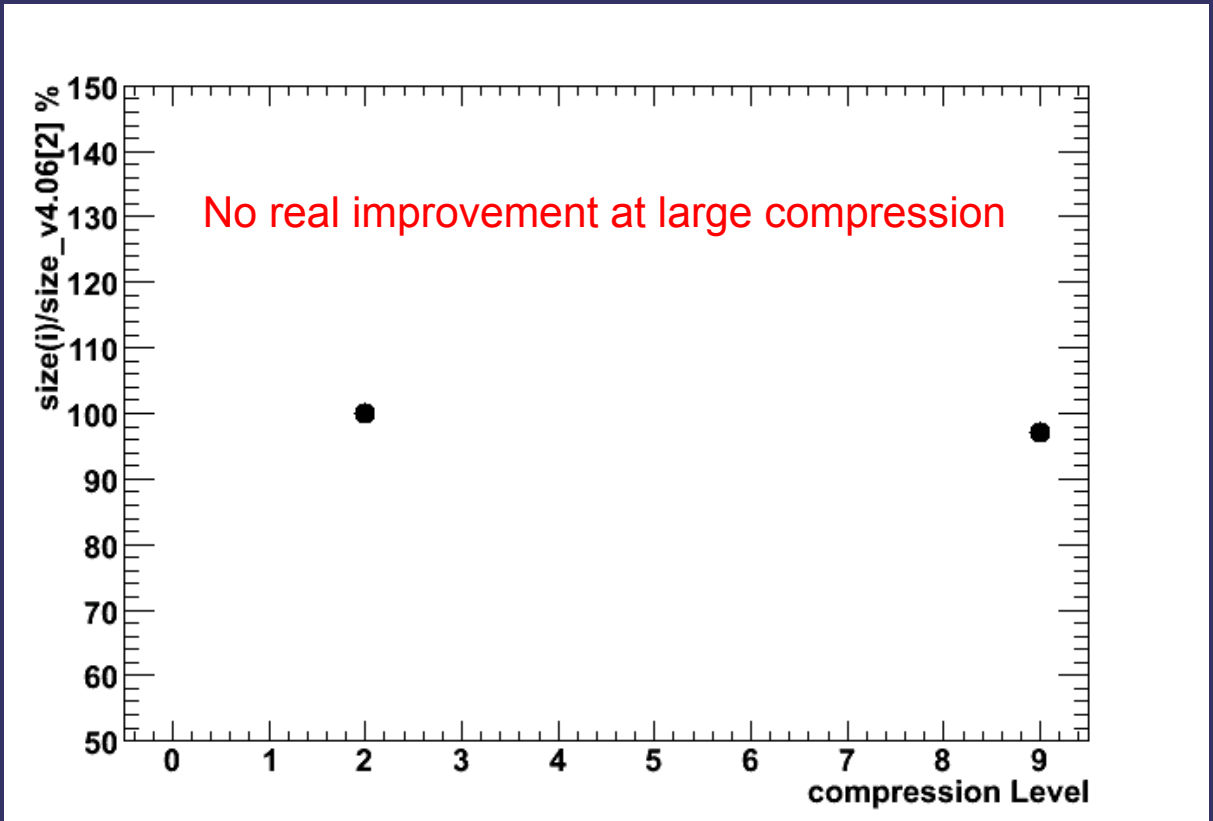
The Situation in p+p

- **v4-06-Release**
  - 50 MB events
  - 28 kB per Minbias p+p event (Compressionlevel 2 default, same with 1, uncompressed 525kB)
  - Check compression



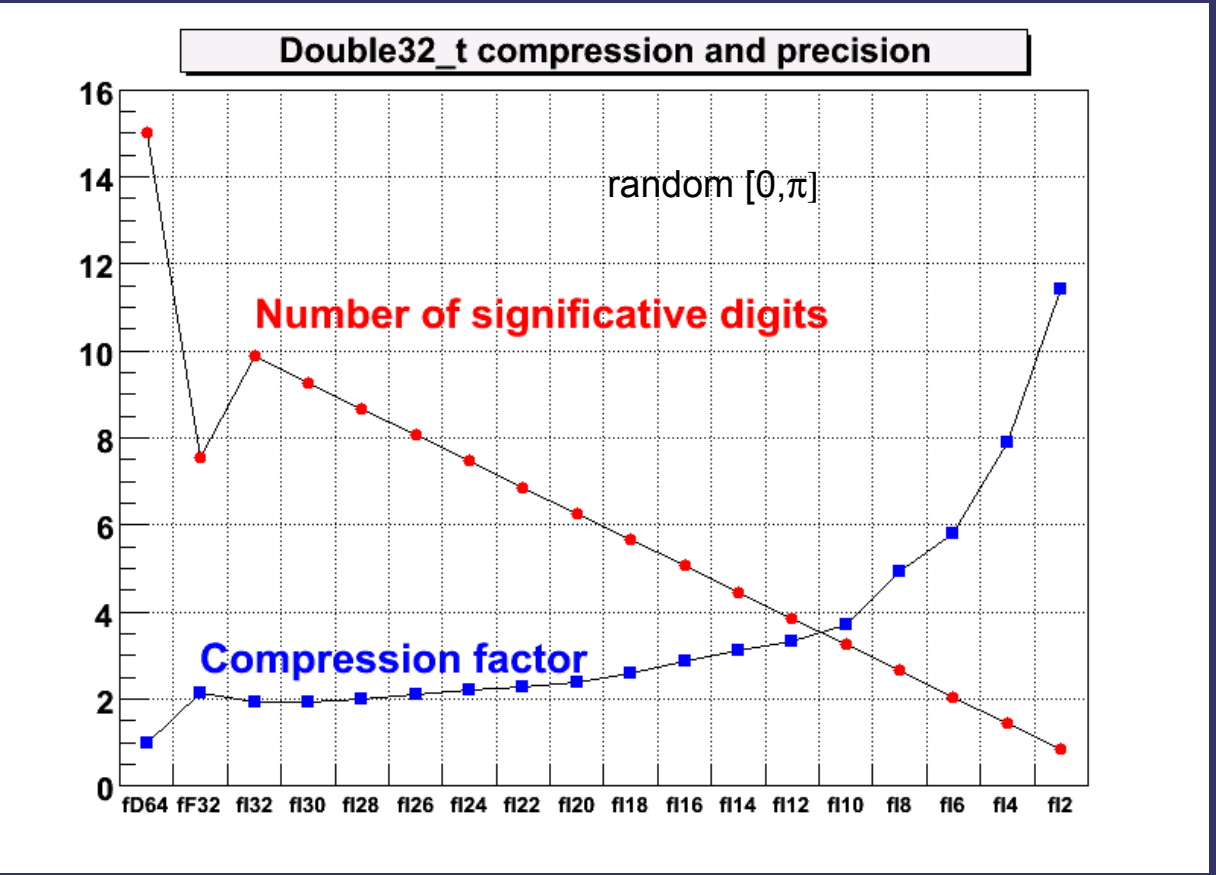
The Situation in Pb+Pb

- **v4-06-Release**
  - Standard Hijing (per2) one Event (N_{coll} = 975, ~14000 tracks)
  - 13930 kB



A Word on Double32_t

- **Double_t in memory**
- **Written as Float_t (32bit)**
- **With ranges:**
  - e.g. //[0,pi,12]
  - Written as UInt_t
  - Encoded with 12bit precision
  - Compression due to leading zeros and “binning”
  - Only usefull for rather **flat distributions**
  - **Absolute** errror is uniform, not the relative
- **NOT usefull for integer values**
  - Don't [0,1024,10], use Short_t



Reducing data size

- **Clean up: Remove variables/redundant information**
  - Only done in very few cases (e.g. AliESDtrack::fStopVertex)
  - AliESDtrack: Contains all kinds of detector related things
    - Only filled/used for a fraction of tracks
    - E.g. HMPID hits in LORS, perhabs this is unavoidable...
  - Need more feedback from tracking expert(s) on V0s, kinks and cascades on what is reduntant
- **Move from Int_t to Short_t or Char_t when possible**
  - Some cases done, #ITScluster #TPCclusters etc.
- **Move from Double_t to Double32_t**
  - Already done prior to v4-06 in many cases
- **In some cases this may seem a little overdone (e.g. Muons, Cascades) but better to have a common conception of the problem**

Improving Compression

- **Use same default values as uniform as possible**
  - Init to “0” wherever possible
  - E.g. chi2 init values seen: -1,0,7,31,1E+31
  - If you need large numbers 4096 is better than 1234
- **Use Double32_t with ranges**
  - Only some angles [-2pi,2pi,16] and PID [0,1,8] for now
- **Sort in the header by data size and type**
  - Largest first, treat Double32_t with range as UInt_t
  - However, consider also which variables are likely to be similar (angle, MC-labels, unused track-variables, TRD,TOF,HMPID)
  - Should slightly improve memory usage as well
- **Changes in CVS**

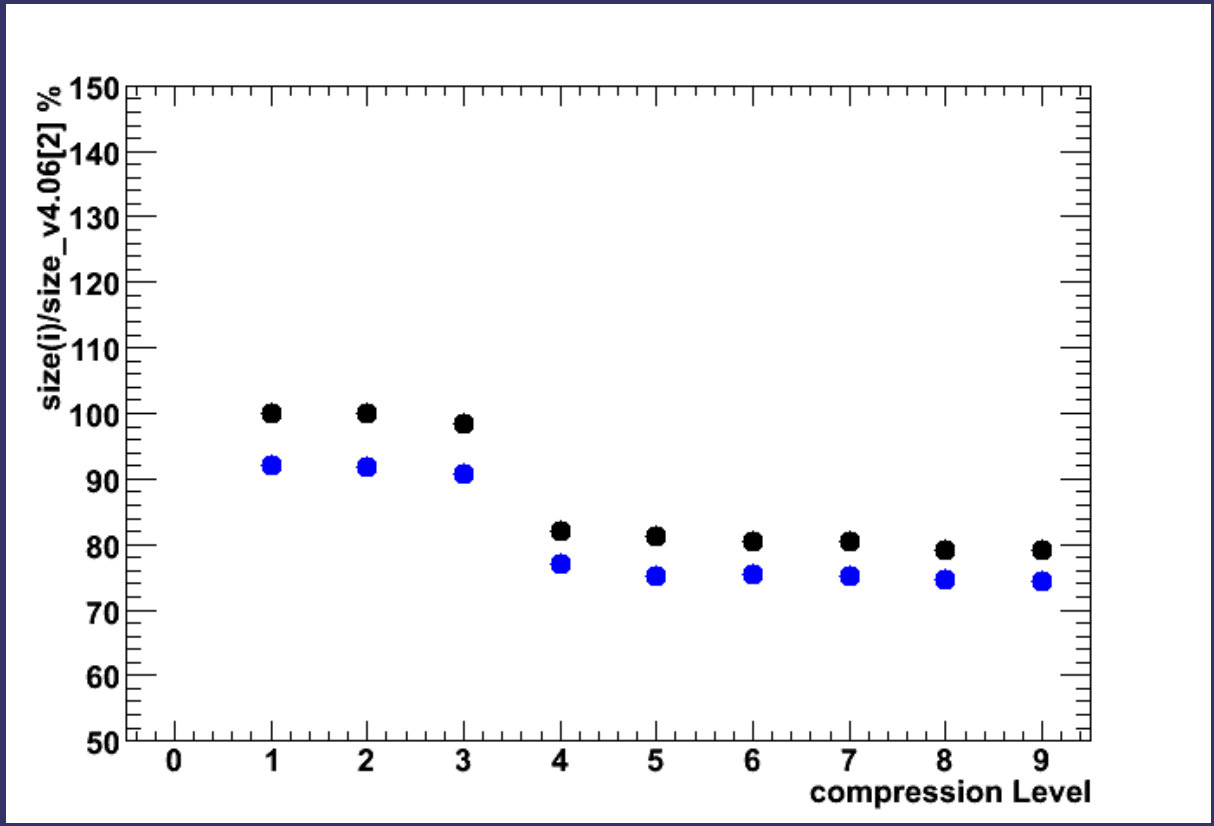
Up to now nothing breaks backward compability

Example

```
AliExternalTrackParam *fCp; // Track parameters constrained to the primary vertex
AliExternalTrackParam *fIp; // Track parameters at the first measured point (TPC)
AliExternalTrackParam *fTPCInner; // Track parameters at the first measured point (TPC) - first itteration
AliExternalTrackParam *fOp; // Track parameters at the last measured point (TPC or TRD)
AliESDfriendTrack *fFriendTrack; //! All the complementary information

TBits    fTPCClusterMap; // Map of clusters, one bit per padrow; 1 if has a cluster on given padrow
TBits    fTPCSharedMap;  // Map of clusters, one bit per padrow; 1 if has a shared cluster on given padrow



ULong_t   fFlags;          // Reconstruction status flags
Int_t     fID;             // Unique ID of the track
Int_t     fLabel;          // Track label
Int_t     fITSLabel;       // label according TPC
Int_t     fTPCLabel;       // label according TPC
Int_t     fTRDLabel;       // label according TRD
Int_t     fTOFLabel[3];    // TOF label
Int_t     fTOFCalChannel;  // Channel Index of the TOF Signal
Int_t     fTOFindex;       // index of the assigned TOF cluster
Int_t     fHMPIDqn;         // 1000000*QDC + number of photon clusters
Int_t     fHMPIDcluIdx;     // 1000000*chamber id + cluster idx of the assigned MIP cluster
Int_t     fEMCALindex;     // index of associated EMCAL cluster (AliESDCaloCluster)


Int_t     fKinkIndexes[3]; // array of indexes of posible kink candidates
Int_t     fV0Indexes[3];   // array of indexes of posible kink candidates

Double32_t   fR[AliPID::kSPECIES]; //[0.,1.,8] combined "detector response probability"
Double32_t   fITSr[AliPID::kSPECIES]; //[0.,1.,8] "detector response probabilities" (for the PID)
Double32_t   fTPCr[AliPID::kSPECIES]; //[0.,1.,8] "detector response probabilities" (for the PID)
Double32_t   fTRDr[AliPID::kSPECIES]; //[0.,1.,8] "detector response probabilities" (for the PID)
Double32_t   fTOFr[AliPID::kSPECIES]; //[0.,1.,8] "detector response probabilities" (for the PID)
Double32_t   fHMPIDr[AliPID::kSPECIES];//[0.,1.,8] "detector response probabilities" (for the PID)

Double32_t fHMPIDtrkTheta;//[-2*pi,2*pi,16] theta of the track extrapolated to the HMPID, LORS
// how much of this is needed?
Double32_t fHMPIDtrkPhi;     //[-2*pi,2*pi,16] phi of the track extrapolated to the HMPID, LORS
Double32_t fHMPIDsignal;  // HMPID PID signal (Theta ckov, rad)

Double32_t   fTrackTime[AliPID::kSPECIES]; // TOFs estimated by the tracking
Double32_t   fTrackLength;   // Track length
Double32_t   fD;             // Impact parameter in XY plane
Double32_t   fZ;             // Impact parameter in Z
Double32_t   fCdd,fCdz,fCzz; // Covariance matrix of the impact parameters

Double32_t   fCchi2; // chi2 at the primary vertex
Double32_t   fITSchi2;        // chi2 in the ITS
Double32_t   fTPCchi2;        // chi2 in the TPC
Double32_t   fTRDchi2;        // chi2 in the TRD
Double32_t   fTOFchi2;        // chi2 in the TOF
Double32_t fHMPIDchi2;       // chi2 in the HMPID
```

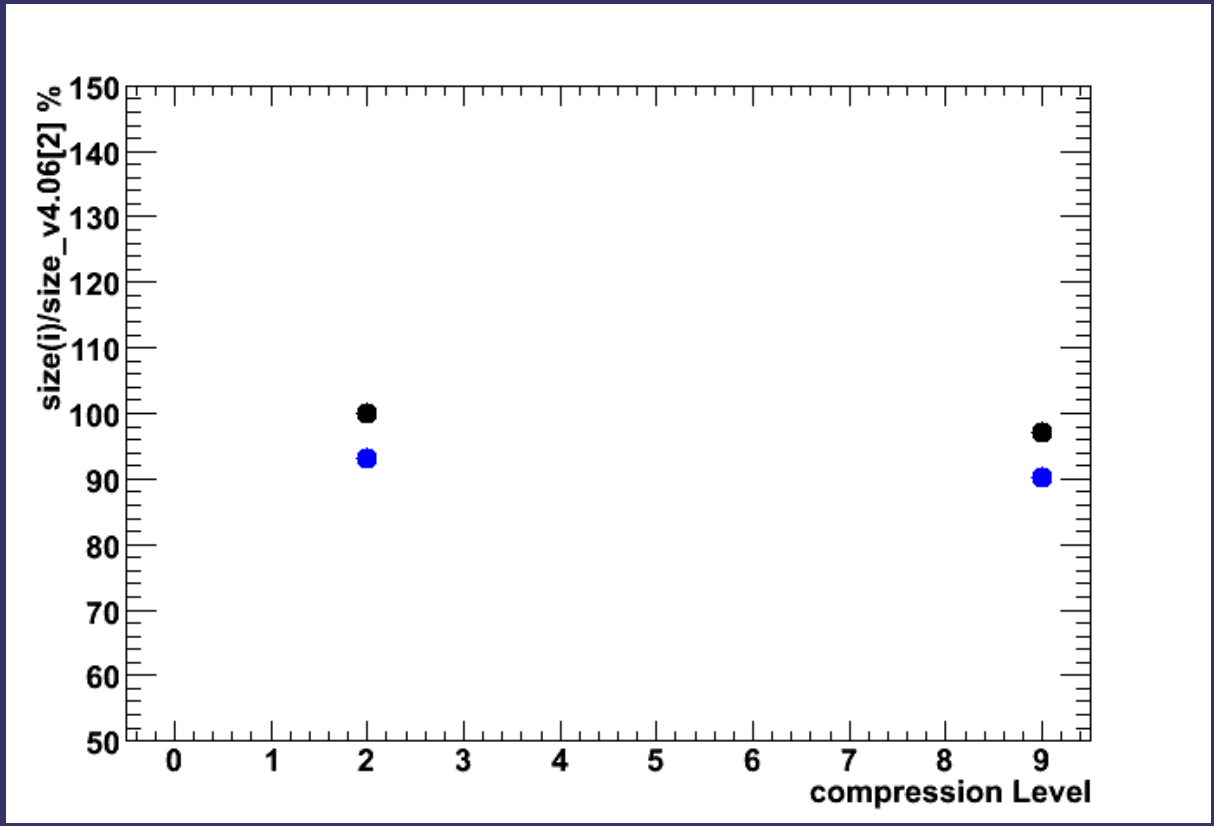
What do we get in p+p

- Some ~8% improvement
- Compression 5 gains total > 20%



What do we get in Pb+Pb

- Some ~8% improvement
- Not much gain with higher compression 5

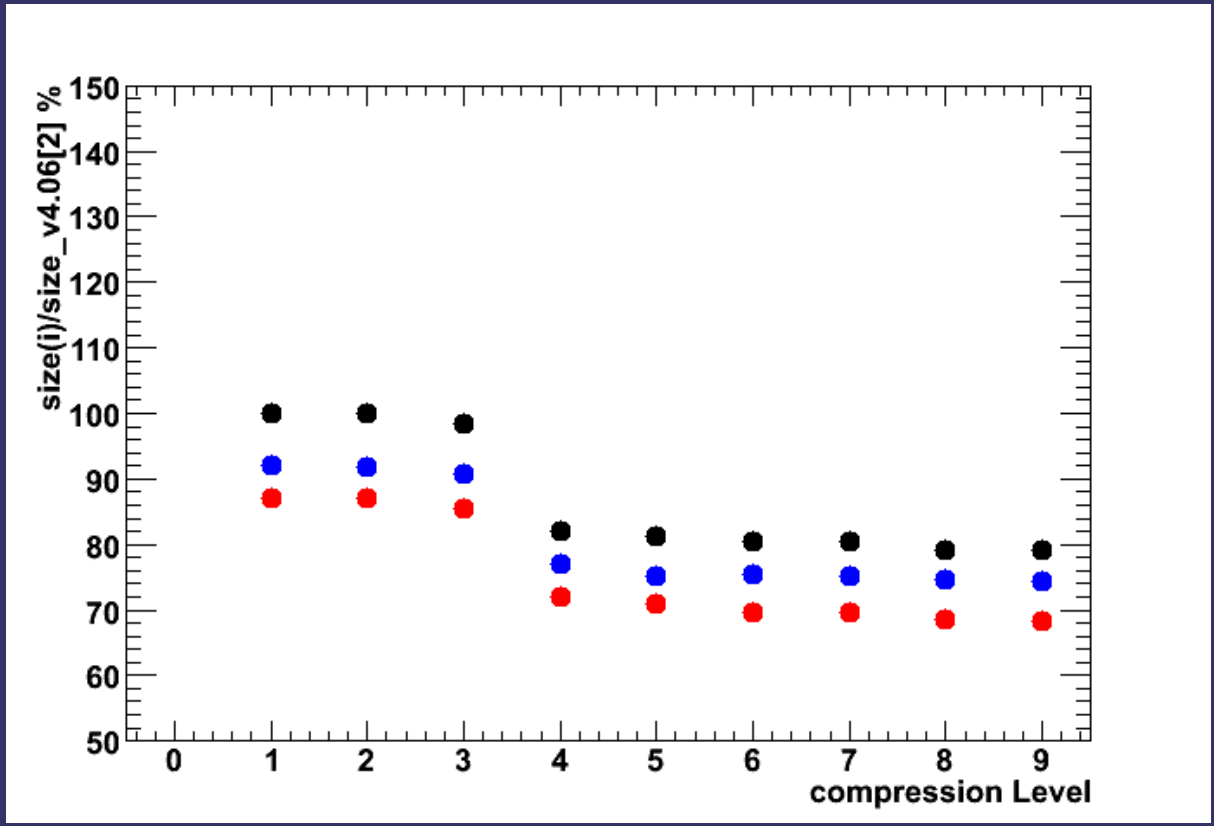


Further possible improvements

- **Some minor changes after discussion with detector/tracking experts**
- **Purification of ESD tracks/V0 (Youri)**
- **AliExternalTrackParam**
  - 5 times in AliESDtrack
  - 2 times in AliESDv0
  - 2 times in AliESDkink
  - Reduce precision for off-diagonal elements of covariance matrix
    - Hack with AliAODRedCov<6>
    - [-1,1,8]
    - Don't stream fC[21]
  - Disclaimer: This was just a test what can be done, implementation details not sorted out yet, Marian's point of CPU-time needs to be considered

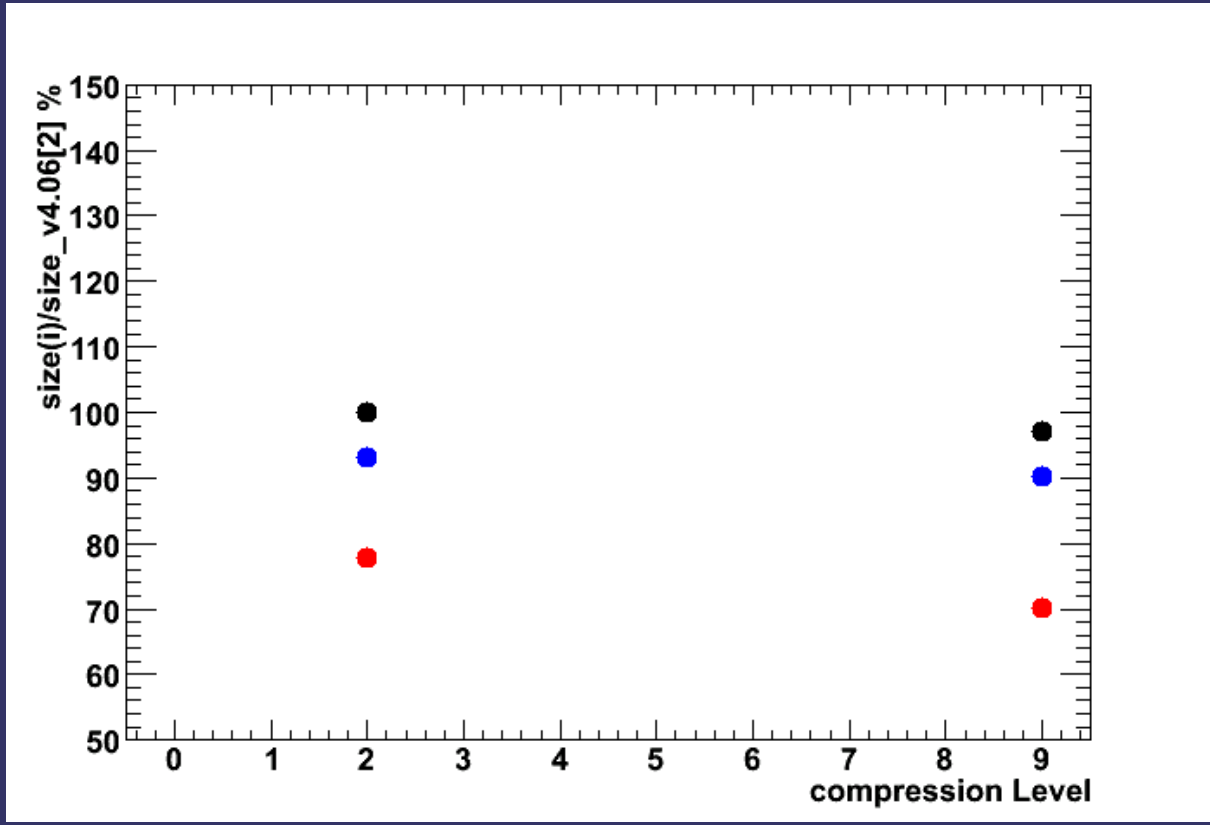
What can we get in p+p

- Some 15% improvement
- Compression 5 gains ~ 30%



What can we get in Pb+Pb

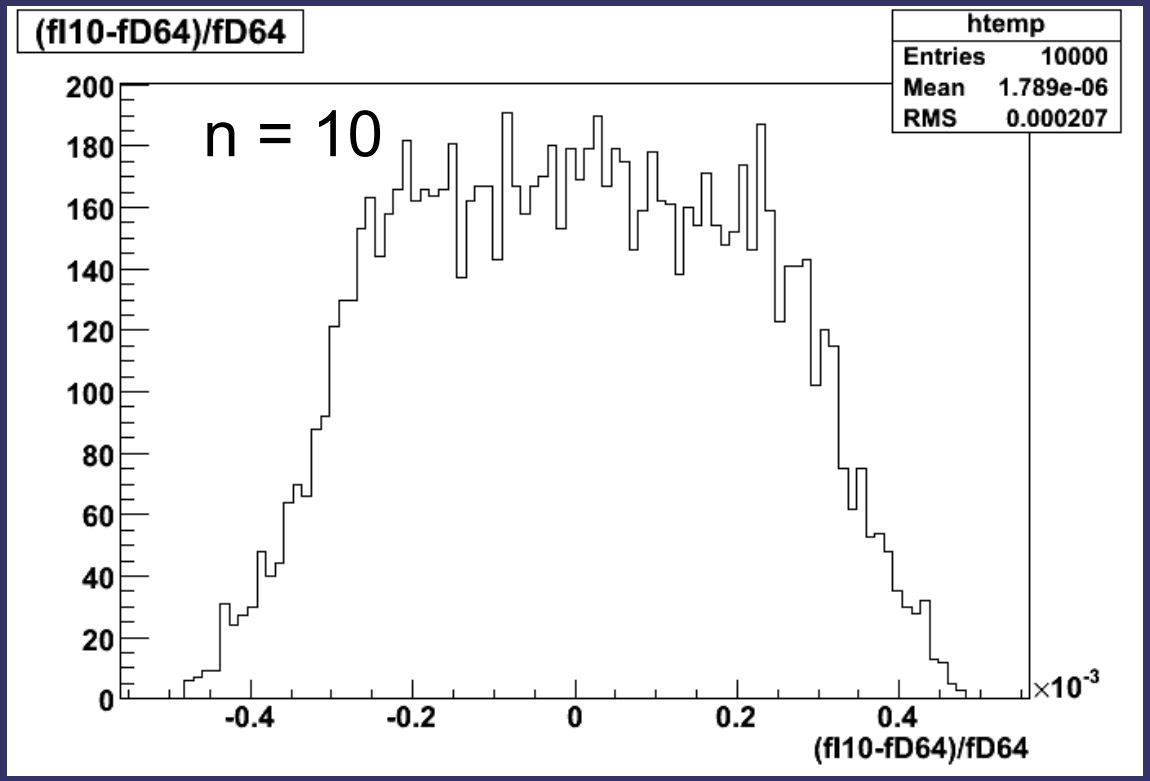
- **Significant gain of 25% to 30%**
  - The catch: Modify AliESDExternalTrackParam cannot read back old data with aliroot (only with ESD.par)



New Double32_t Option

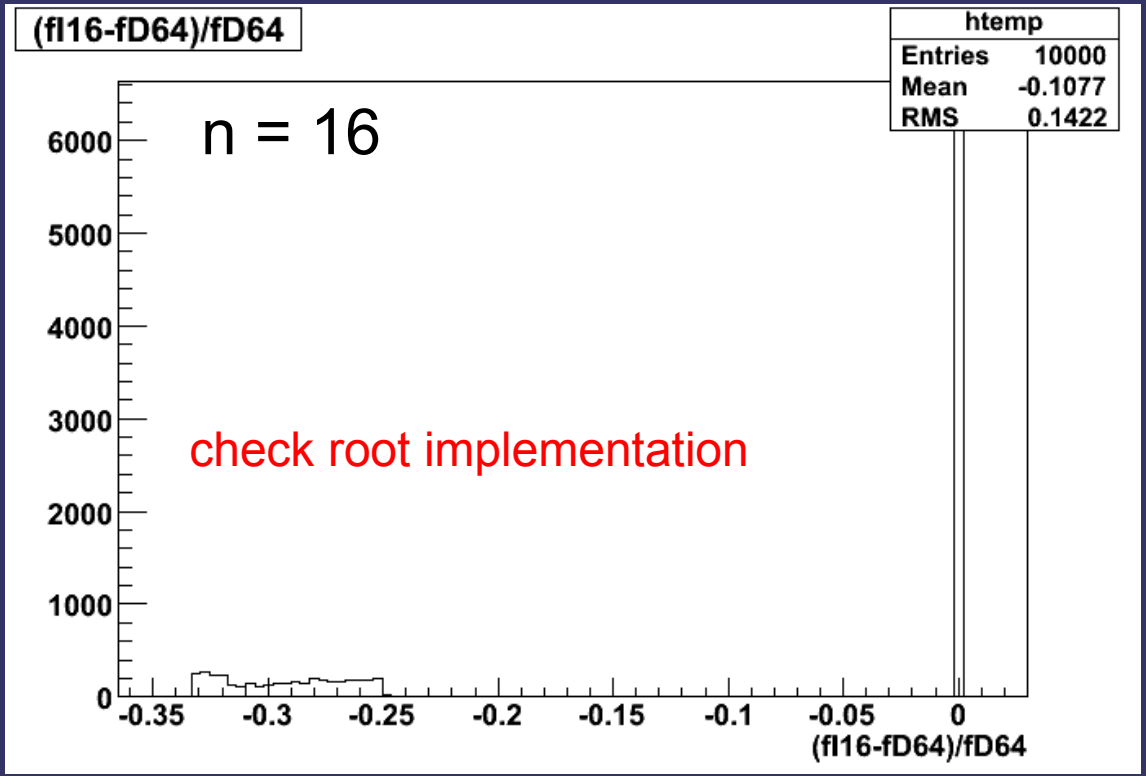
- **Double32_t fChi2 //[0,0,n] root Head version**
  - Streams mantissa and exponent separately mantissa with n-Bit precision
  - Gives a uniform RELATIVE error
  - No need for limiting the range
  - Sizeable compression expected

Could help in many cases where a limited precision is enough.
Smooth transition since //[0,0,n] has no meaning in older root-versions...



New Double32_t Option

- **Double32_t fChi2 //[0,0,n] root Head version**
  - Streams mantissa and exponent separately mantissa with n-Bit precision
  - Gives a uniform RELATIVE error
  - No need for limiting the range
  - Sizeable compression expected



Summary

- **Common interface for AOD/ESD**
- **ESD Size**
  - Some improvement already reached by simple means
    - Grouping, use Double32_t
    - Int_t to Short_t/Char_t
    - Common initializations
      - Some standards would be nice ...
    - ~ 8% gain, in CVS
  - Of course breaking can give large improvements
    - Better concentrate on condensing to relevant physics information
  - Explore usage of new Double32_t features
  - Carry on clean-up, some input from experts needed...

Sizes

ESD	CTDR	v4-04	v4-06	Minor-Mod**	AliExternalTrackParam-Mod**
p+p/event	0.04	0.189	0.035	0.032	0.030
Pb+Pb/ev	2.5		6.9	6.4	5.3

*All sizes in MB
**projected

Some Observations on Parameter Ranges

```
AliESDCaloCluster

Double32_t   fDispersion;       // cluster dispersion, for shape analysis
> -1 to ~4 no clear cut...
Double32_t   fChi2;             // chi2 of cluster fi
> -1
Double32_t   fPID[AliPID::kSPECIESN]; //"detector response probabilities" (for the PID)
> 0-1
Double32_t   fM20;              // 2-nd moment along the main eigen axis
> 0-2.5 NCC
Double32_t   fM02;              // 2-nd moment along the second eigen axis
> 0-12 NCC
Double32_t   fM11;              // 2-nd mixed moment Mxy
> -1 or 0...
Double32_t   fEmcCpvDistance;   // the distance from PHOS EMC rec.point to the closest CPV rec.point
> -1 or 10000
Double32_t   fDistToBadChannel; // Distance to nearest bad channel
> -1 or 10000
UShort_t  fClusterType;      // Flag for different cluster type/versions
> -1,0,2
UShort_t  fNExMax ;          // number of (Ex-)maxima before unfolding
> 0,1,2



AliESDcascade

Double32_t fChi2Xi;         // chi2 value
> always 7
Double32_t fPosCovXi[6];    // covariance matrix of the vertex position
> all 0
Double32_t fBachMomCov[6];  // covariance matrix of the bachelor momentum.
> 0 or 1E+10;


AliESDFmd.

Maps initialized to 1000 bad...
Why is multiplicity a Float?
```

Some Observations on Parameter Ranges

```
AliESDTZERO

Float_t      fT0timeStart;     // interaction time estimated by the T0
> -150 - 0 all ints
Float_t      fT0time[24];      // best TOF on each T0 PMT
> -150 - 1000 all ints
Float_t      fT0amplitude[24]; // number of particles(MIPs) on each T0 PMT
> 0 -300

AliESDv0

  // CKB PDGCode really needed in real data
  Int_t    fPdgCode;             // reconstructed V0's type (PDG code)
  > 310 or -3122
  // No need to stream this, just update when called first... make virtual
  // for calls in AliESDkinks
  Double32_t  fEffMass;          // reconstructed V0's effective mass
  Double32_t  fDcaV0Daughters;   // dca between V0's daughters
  Double32_t  fPointAngle;       //[-1,1,16] cosine of the pointing angle
  > 0 - 1 ...
  // CKB never modified!! skip it?? is 31 always...
  Double32_t  fChi2V0;           // V0's chi2 value
  Double32_t fPos[3];         // V0's position (global)
  // CKB zero at the moment... is only used for filling in AOD
  Double32_t fPosCov[6];      // covariance matrix of the vertex position
  > all zero
  // CKB the getter is never used... in
  Double32_t        fNormDCAPrim[2];  // normalize distance to the priary vertex
  //
  // CKB only used in AliITS/TPCTrackerMI don't stream...
  Double32_t        fAngle[3];   //[-2*pi,2*pi,16]three angles
  > 0-pi

  // Made Short_t only values seen 0 and 100? checked in AliITStrackerMI.cxx
  Int_t             fStatus;     //status
  > 0 - 100
  Double32_t        fDistSigma; //sigma of distance
  // CKB only used in AliITSTrackerMI don't stream...??
  Double32_t        fCausality[4];  //[0,1,6] causality information - see comments in SetCausality
  > 0 - 1
  // can be reduced?? getters are never uses....
  Short_t           fNBefore;      // number of possible points before V0
  > 0-6 identical to fNAfter
  Double32_t        fChi2After;   // chi2 of the tracks after V0
  > -200 , -2, -1 , 0 -50
  Short_t           fNAfter;      // number of possible points after V0
  > 0 -6
```

Some Observations on Parameter Ranges

```
 // only modified in AliV0::Update no getter
 // skip... don't stream...
 Double32_t        fPointAngleFi; //[-2*pi,2*pi,16]point angle fi
 > 0 or close to 1
 Double32_t        fPointAngleTh; //[-2*pi,2*pi,16]point angle theta
 > 0 or 1
 //

 Double32_t fCovXX,fCovXY,fCovYY,fCovXZ,fCovYZ,fCovZZ;  // vertex covariance matrix
 > 0.05**2  , 0 ,0 0.05**2, 0, 0, ~2.5E-05
Double32_t fSNR[3];  // S/N ratio
 > 0
 Double32_t fChi2;  // chi2 of vertex fit
 > 0
```

Some Observations on Parameter Ranges

```
Short_t          fMultiple[2];
> 0 at the moment
Double32_t        fTPCdensity[2][2];  //tpc cluster density before and after kink
> 0-1
Double32_t        fTPCdensity2[2][2];  //tpc cluster density before and after
> -1!!
 Double32_t       fAngle[3]; //three angles
> 0-0.16 NCC
Char_t         fStatus[12]; /status of kink - first 4 mother (ITS,TPC,TRD,TOF)  other daughter
> NAN??

Int_t          fRow0;              // critical pad row number
> 40-140
Int_t          fTPCncls[2];     //number of clusters for mother particle
> 0-140

AliESDMuonTrack.

Non found...

AliESDPmdTrack

Short_t fDet;      // Detector, 0:PRE, 1:CPV
> 0 or 1
Float_t fX;      // Cluster X position
> -80 - 80
Float_t fY;      // Cluster Y position
> -90 - 90
Float_t fZ;      // Cluster Z position (vertex uncorrected)
> 359. or 363.
Float_t fCluADC; // Cluster Energy in ADC
> 0 - 8000 NCC
Float_t fNcell;  // Cluster cells
> 0 -83 NCC
Float_t fCluPID; // Cluster probability, 1: Photon, 0: Hadron
> 0 or 1 is this just a flag??
```

Some Observations on Parameter Ranges

```
Short_t          fMultiple[2];
> 0 at the moment
Double32_t        fTPCdensity[2][2];  //tpc cluster density before and after kink
> 0-1
Double32_t        fTPCdensity2[2][2];  //tpc cluster density before and after
> -1!!
 Double32_t       fAngle[3]; //three angles
> 0-0.16 NCC
Char_t          fStatus[12]; /status of kink - first 4 mother (ITS,TPC,TRD,TOF)  other daughter
> NAN??

Int_t           fRow0;              // critical pad row number
> 40-140
Int_t           fTPCncls[2];     //number of clusters for mother particle
> 0-140

AliESDMuonTrack.

Non found...

AliESDPmdTrack

Short_t fDet;      // Detector, 0:PRE, 1:CPV
> 0 or 1
Float_t fX;      // Cluster X position
> -80 - 80
Float_t fY;      // Cluster Y position
> -90 - 90
Float_t fZ;      // Cluster Z position (vertex uncorrected)
> 359. or 363.
Float_t fCluADC; // Cluster Energy in ADC
> 0 - 8000 NCC
Float_t fNcell;  // Cluster cells
> 0 -83 NCC
Float_t fCluPID; // Cluster probability, 1: Photon, 0: Hadron
> 0 or 1 is this just a flag??
```

```
Bool_t   fOnFlyStatus;    // if kTRUE, then this V0 is recontructed
                          // "on fly" during the tracking
Int_t    fPdgCode;          // reconstructed V0's type (PDG code)
Float_t  fEffMass;          // reconstructed V0's effective mass                // CKB calculable on the fly
Float_t  fDcaV0Daughters;   // dca between V0's daughters                       // CKB calculabe on the fly from tracks (GetDCA),
Float_t  fPointAngle;       //cosine of the pointing angle                     // CKB there is even a member function to calc it, calcu
lable on the fly
Float_t  fChi2V0;           // V0's chi2 value                                  // CKB no setter, not modified in the AliESv0
Double32_t fPos[3];         // V0's position (global)                           // CKB  Makes sense, but this is set in the constructor
based on the AliExternalTrackParam, no setter!!
Double32_t fPosCov[6];      // covariance matrix of the vertex position         // CKB No setter no getter, never modified
Int_t fNidx;                // index of the negative daughter                   // CKB track reference?
Double32_t fNmom[3];        // momentum of the negative daughter (global)       // CKB not needed with track ref, complete overkill anyw
ay since fParamN is stored!
AliExternalTrackParam fParamN;  // external parameters of negative particle     // CKB not needed with track reference
Int_t fPidx;                // index of the positive daughter                   // same as with negatives
Double32_t fPmom[3];        // momentum of the positive daughter (global)       // same as with negatives
AliExternalTrackParam fParamP;  // external parameters of positive particle     // same as with negatives


// **** The following data members need to be revised ***                       // CKB DO IT!!!

Int_t          fClusters[2][6]; //! its clusters                               // CKB  not set in aliv0vertexer
Float_t        fNormDCAPrim[2];  // normalize distance to the priary vertex     // Set only in AliITSVertexMI for AliV0 object
Double32_t     fAngle[3];   //three angles                                      // no setter! not modified in AliESDv0, used in AliV0?
Float_t        fRr;         //rec position of the vertex                       // no setter! but used for calculations in AliESDV0, upd
ated in AliV0....
Int_t          fStatus;       //status                                         // CKB of what?
Float_t        fDistSigma; //sigma of distance                                 //  Set only in AliITSTrackerMI for AliV0
Float_t        fCausality[4];  // causality information - see comments in SetCausality  // Set only in AliITSTrackerMI for AliV0, used i
n calculation in AliESDv0...
Float_t        fChi2Before;   //chi2 of the tracks before V0                        // CKB Set only in AliITSTrackerMI for AliV0
Float_t        fNBefore;      // number of possible points before V0               // CKB Set only in AliITSTrackerMI for AliV0
Float_t        fChi2After;   // chi2 of the tracks after V0                       // CKB Set only in AliITSTrackerMI for AliV0
Float_t        fNAfter;      // number of possible points after V0                // CKB Set only in AliITSTrackerMI for AliV0
Float_t        fPointAngleFi; //point angle fi                                   // CKB No Setter! No getter! Calculated in AliV0
::Update
Float_t        fPointAngleTh; //point angle theta                               // CKB No Setter! No getter! Calculated in AliV0
::Update
```