PWG3 Analysis

–Gines Martinez
–for the PWG3 : Heavy Flavours

PWG3 code in AliRoot

- 20 classes in the module;
- 7 analysis tasks and 18 macros;
- 1 library PWG3base;
- Dependencies (to be cross-checked) on libANALYSIS, libSTEERbase, libAOD, libESD, libANALYSISalice and libANALYSISRL (some cases);
- Two independent softwares related to the same physics: HF vertexing in central barrel, HF via the muon channel;

AliAnalysisVertexingHF class

single-track cuts on p_t and d_0, (PID...)
[need common cuts for all analyses (to be studied)]

build all (+,-) pairs and compute secondary vtx

for **$D^0 \to K\pi$, J/ψ (from B)$\to ee$**

create AliAODRecoDecayHF2Prong
apply reco cuts

tight D0 mass cut
loop on all tracks
for D* candidates

store D^0

store D^{*+}

store J/ψ

for **D^+, D_s^+, Λ_c^+**

loop on all tracks (+ & -):
build triplets,
create AliAODRecoDecayHF3Prong,
apply reco cuts (common for the 3 particles?)

store D^+, D_s^+, Λ_c

for **$D^0 \to K\pi\pi\pi$**

loop on all tracks (+ & -) …
create AliAODRecoDecayHF
…

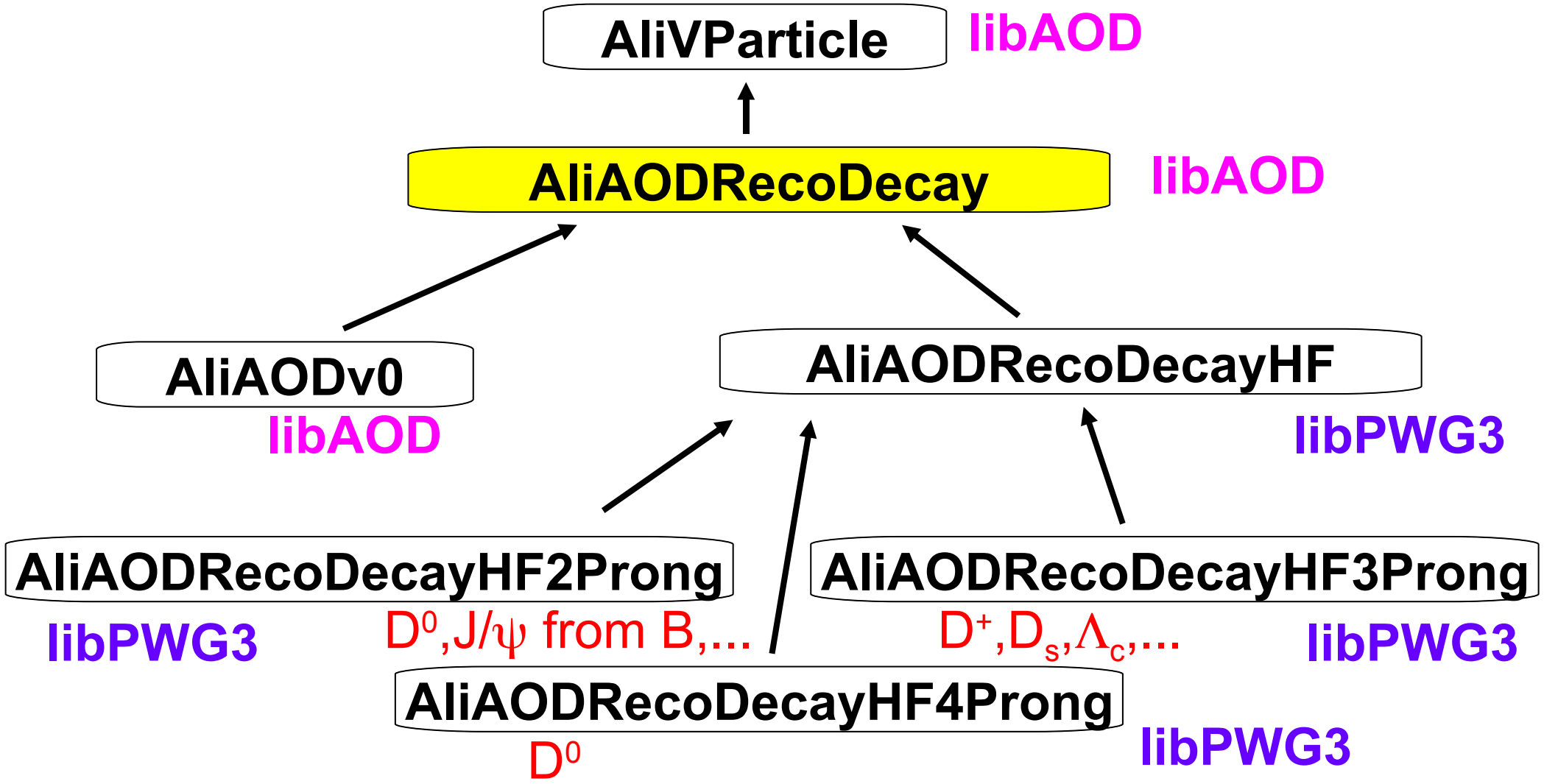
store D^0

AliAnalysisVertexingHF

- Default vertexer is AliVertexerTracks
- Added possibility to use Kalman-filter vertexer (AliKFParticle)
- Current idea for data analysis (discussion with Silvia):
  - use AliVertexerTracks for first pass on data and production of candidates (loose reconstruction cuts)
  - KF vertexer (including constraints) can be used during analysis on candidates, by recomputing the vertex on the fly

AliAODRecoDecay

- common base class for candidates -



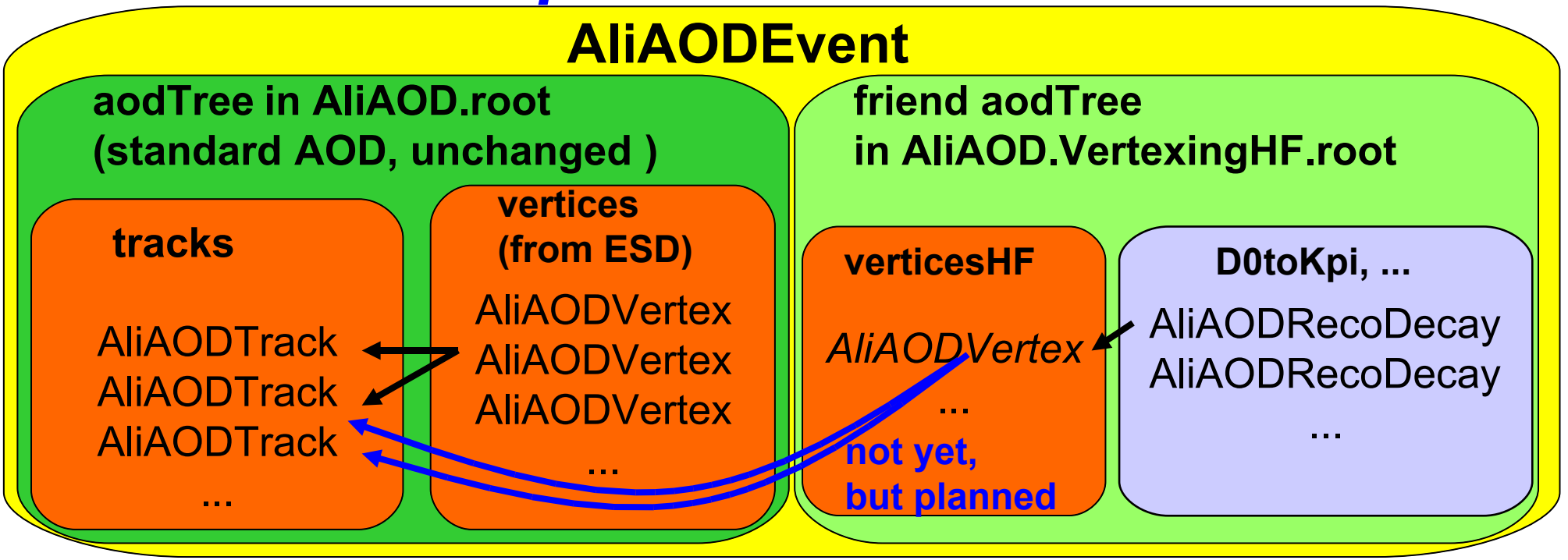
Analysis input: status & plans

- *Current input:* **AliESDEvent** (single event analysis)
  - AliAnalysisVertexingHF uses AliESDtrack (that derives from AliExternalTrackParam, that derives from AliVParticle)
  - Vertex fit done by AliVertexerTracks, that works on array of AliExternalTrackParam
- *Plan:* use **AliVEvent** as input (i.e. both ESD and AOD!) and work with AliVParticle tracks
- *Requirements:*
  - AliVertexerTracks: change it to work on array of AliVParticle
    - this will also allow to reconstruct primary vertex from AliAODEvent
  - AliVParticle: provide abstract methods
    - GetStatus()
    - GetITSClusterMap()
    - RelateToVertex()
    - GetImpactParameters()

Analysis output: status & plans

- *AliAODRecoDecay is flexible*
- *Can be stand-alone → store candidates in a normal TTree*
  - AliVertexerTracks::FindCandidates(AliESDEvent, TTree*)
  - Integrated in ANALYSIS framework by AliAnalysisTaskVertexingHF (by J. Faivre)
    - used on Grid
    - tested in Analysis Train by Mihaela: OK
- *Can use TRef technology and be stored inside AliAODEvent*
  - AliVertexerTracks::FindCandidatesESDtoAOD(AliESDEvent, TClonesArray*)
  - Integrated in ANALYSISalice framework by AliAnalysisTaskSEVertexingHF (many thanks to Andreas)
    - TClonesArray's with candidates added as branches of a TTree that is attached as friend of aodTree and stored in a separate file AliAOD.VertexingHF.root
    - tested in Analysis Train by Mihaela: OK

Output in AliAODEvent

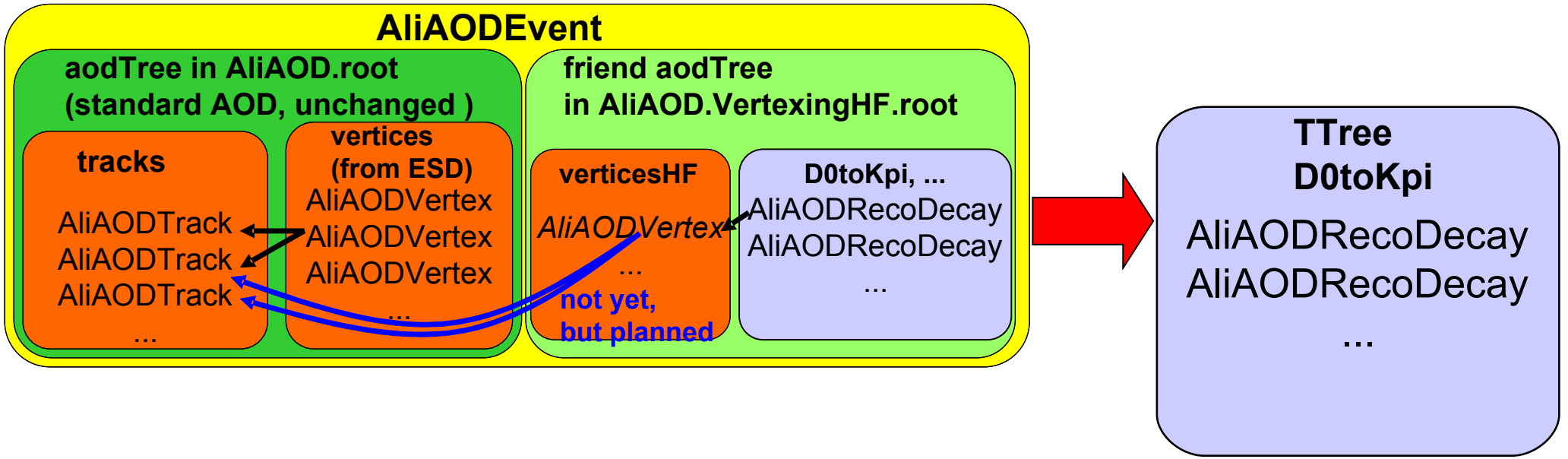


- All the I/O nicely managed by AliAODHandler & AliAODInputHandler
- Macro to read the candidates: ReadAODVertexingHF.C
- Warning: on Grid, the friend file(s) should be on the same Storage Element as standard AOD file (or a replica of it)
  - e.g.: in a given Tier-2 there is a replica of a sample of std AODs; we send the analysis specifying to use the CE and SE of that Tier-2

Candidates analysis from AliAODEvent

NEW

- AliAnalysisTaskSESelectHF:
  - read candidates from AliAODEvent (from friend tree)
  - do some analysis/selection (can also recompute vertex with Kalman filter and use topological/mass constraints)
  - write candidates to a new branch of the AOD
  - OR convert candidates to stand-alone and write them to a standard TTree (to be copied on a laptop for making the plots or the invariant mass analysis)



Summary of requirements

input also from S.Masciocchi

- We would like to use AliVEvent & AliVParticle as interfaces
- AliVEvent should provide abstract method GetPrimaryVertex()
- AliVParticle should provide abstract methods needed for:
  - track selection
    - GetStatus()
    - GetITSClusterMap()
  - creation of AliKFParticle from AliVParticle
    - getter for covariance matrix
  - propagation and calculation of impact parameters
    - PropagateTo()
    - RelateToVertex()[to be implemented in AliAODTrack]
    - GetImpactParameters() [to be implemented in AliAODTrack]
    - other possibility (quick & dirty?) is to always create a AliExternalTrackParam from AliVParticle and then use the methods of AliExternalTrackParam to do the propagation and vertexing
- An abstract class (AliVVertex?) for AliAODVertex and AliESDVertex is probably needed
- Btw, thanks to Andreas, Markus, Mihaela for the good support provided

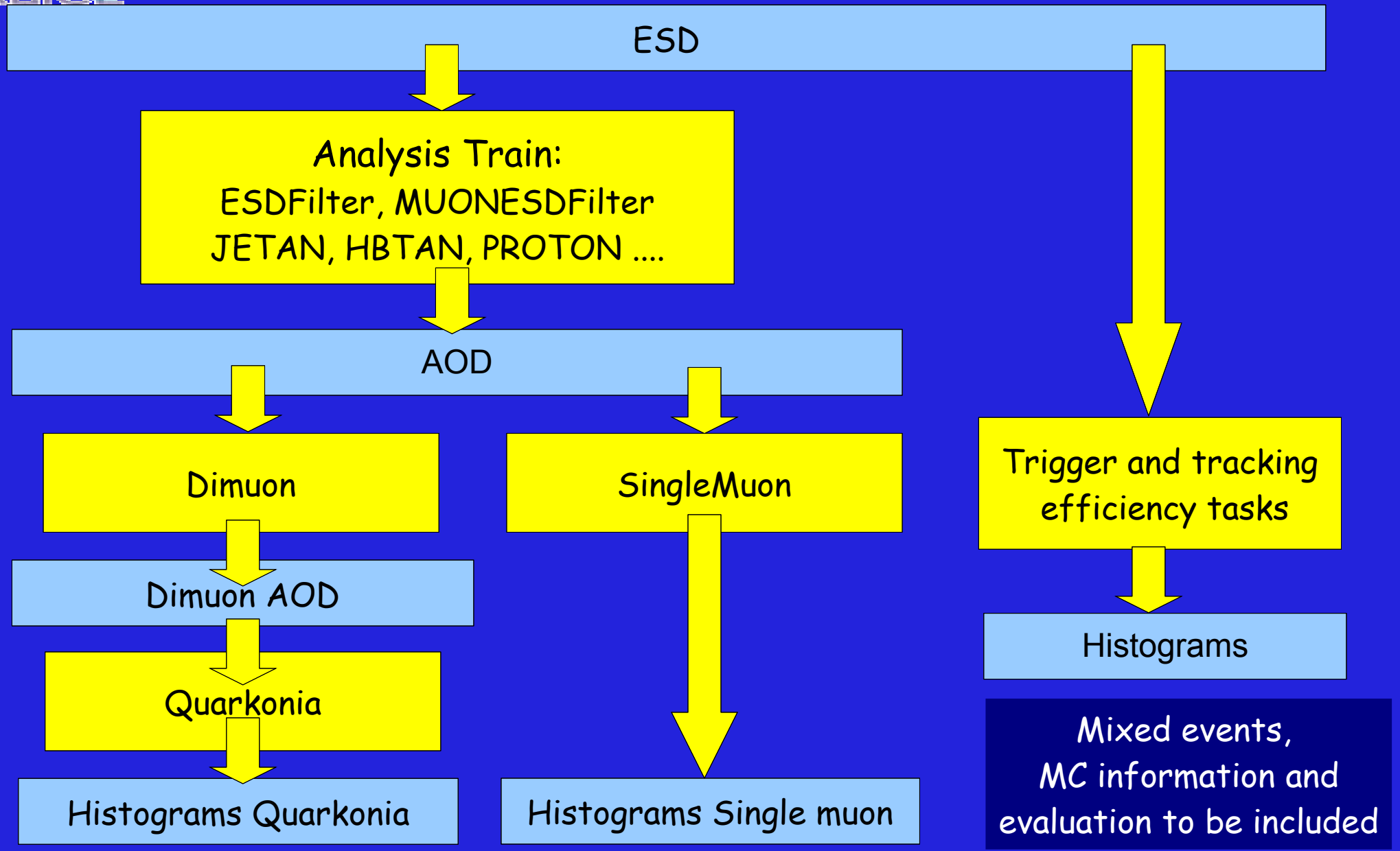
Next steps in HF-vertexing

- Use AliESDtrackCuts for track selection
  - need possibility to check ITS cluster map!;
- Develop tools for comparison with MC, using AliMCInputHandler;
- Develop tools for corrections, using the Correction Framework;

HF-muon software

- Muon tracks are stored in the AOD via the AliAODTrack class;
- Muon AOD information is included in the Anaysis train via the ESDfilter analysis taks;
  - Standalone task MUON will be committed soon (R. Arnaldi);
  - New information for DCA analysis ready in the AOD (P. Pillot)
- MUON analysis tasks for the trigger and tracking efficiencies under progress (N. Le Bris and D. Stocco);
- A dimuon-AOD like (reduced size) is created via the task MuonAOD from general and special classes for storage have been defined (P. Cortese)
- New task for the determination of the LUT committed (B. Vulpescu)

Tasks for HF-muon



Next steps in HF-muon

- Commit of the efficiency classes
- Commit of the new task MUONESDfilter; To be included in the official analysis train;
- Update of new ESD data into AOD data: DCA analysis (already agreed);
- Test script to be defined (data from pp muon minb id=17); README files to be created;
- Dimuon analysis task to be adapted to the new schema and included in the analysis train;
- Production with Tfluka for the DCA analysis;

Trigger Efficiency task

Trigger chamber efficiency can be calculated via an Analysis Task:

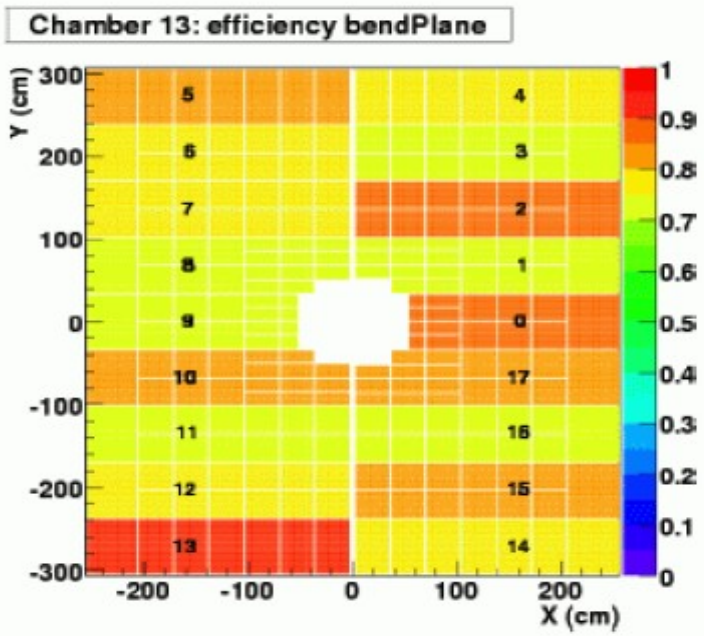
- run `AliAnalysisTaskTrigChEff` (to be committed): saves histograms with $N_{4/4}$ and $N_{3/3}$ into a user defined *filename.root*.
- Get efficiency map object (and display results):

```
AliMUONTriggerEfficiencyCells effCells("filename.root")
effCells.DisplayEfficiency(Bool_t perSlat)
```

Diego Stocco
Lyon March 18-20

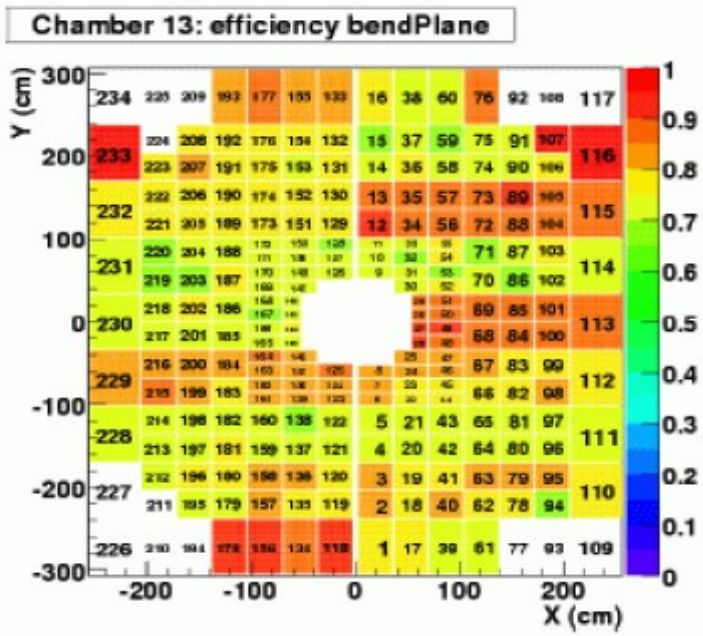
Examples of results:

effCells.DisplayEfficiency(kTRUE)



Slat	Input eff.(%)
0	88
1	72
2	87
3	73
4	73
5	83
6	75
7	77
8	72
9	73
10	84
11	72
12	77
13	92
14	76
15	81
16	74
17	80

effCells.DisplayEfficiency(kFALSE)



PWG3 productions in AliEn

21	PDC 07/LHC07g	Completed	PWG3	p-p	14 TeV	20,000,000	25,446,300
20			PWG3	p-p	14 TeV	200,000	0
19			PWG3	p-p	14 TeV	260,000	0
18			PWG3	p-p	14 TeV	5,000,000	0
17	PDC 08/LHC08t	Running	PWG3	p-p	14 TeV	125,000,000	46,248,510

- 21 : Dimuon production in pp @ 14 TeV;
- 17: m minb in pp @ 14 TeV;
- 20: Beauty, with B->J/psi->ee decay forced, 19: Beauty, charm hadronic decays forced, and 18: Charm with forced hadronic decays;

These productions were ubmitted in nov 07. We propose:

- produce them in specific Tiers (Italy + GSI) and use thier SEs
- analyse them using new AliAnalysisTaskSE's:
  - 1st step: produce friend AOD with candidates and store it in same SE as std AOD
  - 2nd step: analysis jobs to run on the friend AOD

Muon minb production (I)

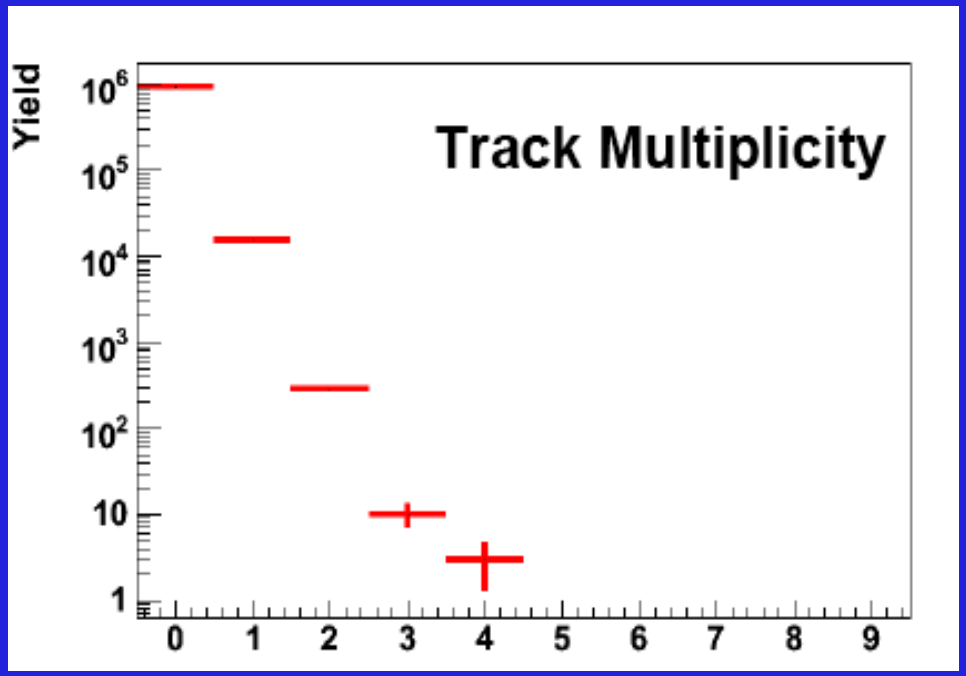
Nicole Bastid
Lyon March 18-20

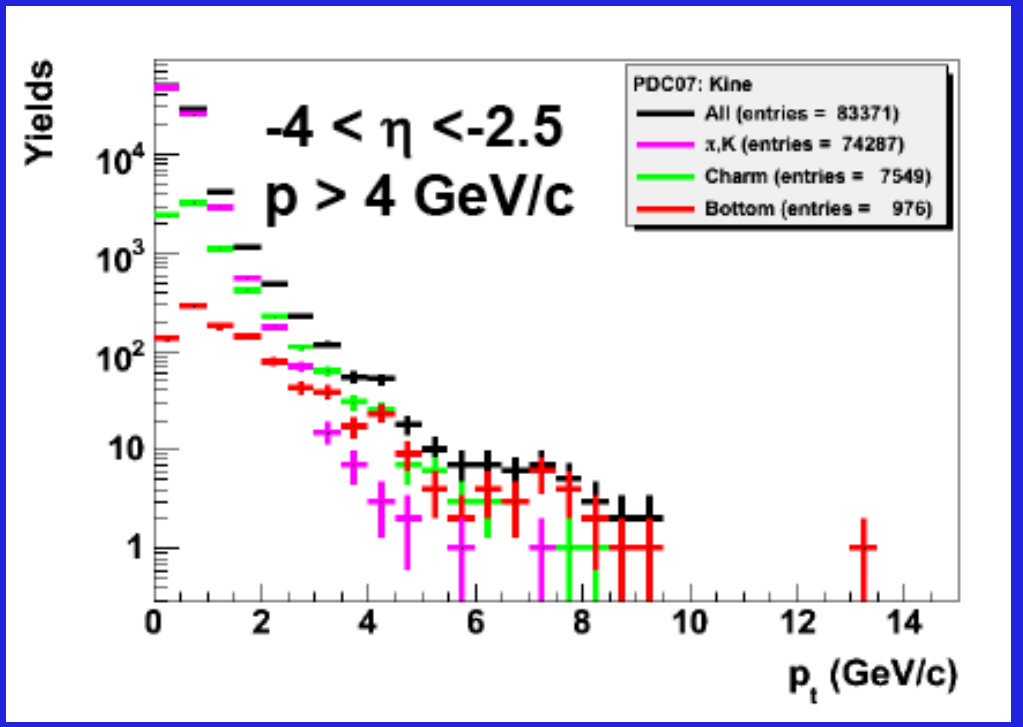
- pp minb muon production (pp at 14 TeV) running smothly in the grid since March 20th 2008;
- Production id=17 (pcalimonitor.cern.ch/PWG)
- 50% of the production is ready on disk to be analyse (LUT with trigger cuts lpt = 1GeV, hpt = 1.7GeV);
- Second half of the production was restarted last week (LUT with trigger cuts lpt = allpt = 0.5GeV, hpt = 1.7GeV);

Muon minb production (II)

Nicole Bastid
Lyon March 18-20

- Full slow simulation of minb pp collisions;
- Agreement with NLO HQ cross-sections;





Tri-muon analysis (I)

Mercedes Lopez

$$B \xrightarrow{BR= 0.1\%} J/\psi + K$$

$J/\psi \rightarrow \mu^+\mu^-$ (BR= 5.9%)

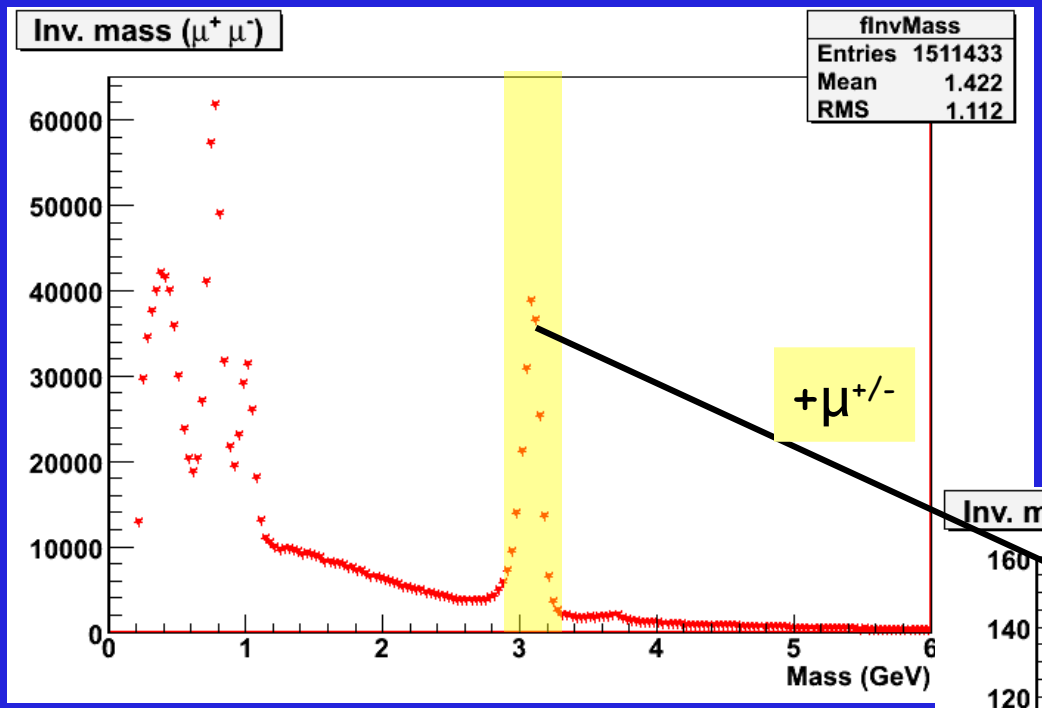
$K \rightarrow \mu$ (BR> 60%)

How:

- Invariant mass for 2 muons, if they are in the mass range of the J/ψ:
  - Invariant mass with a third muon (it should be smaller than the B meson mass)
  - Angular distribution, between the J/ψ and the third muon. Some kind of correlation expected.

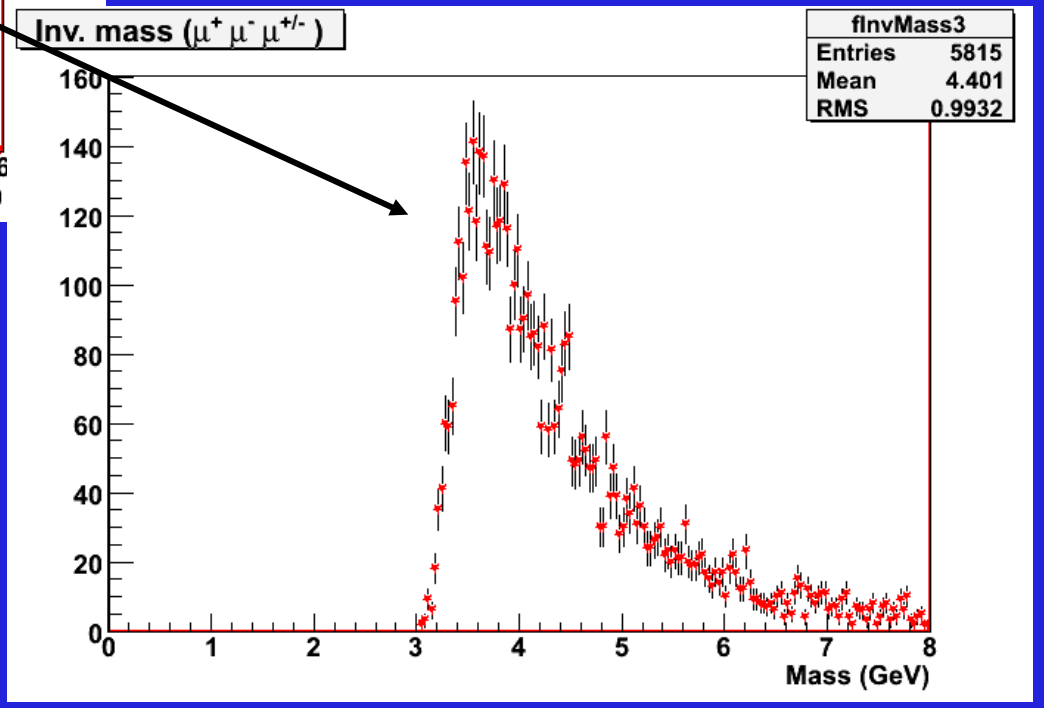
Tri-muon analysis (II)

Mercedes Lopez



Reconstructed events

$+\mu^{+/-}$



Preliminary study

Open items

- New libraries in the PWG3 module:
  - PWG3muon and PWG3vertexing ?
- HF verterxing from AOD;
  - HF vertexing performed form ESD;
- AOD splitting ?
  - One single AOD could slow down analysis based on rare probes;
  - Splitting based on "physics cases" would be optimal if entropy is kept low (selection based on trigger, event cluster and metadata);
  - Merging several ESD in one single AOD will be needed;