Nano AODs and how to use them on Analysis Trains

Markus Zimmermann

08.11.2017

Motivation

Make analysis more efficient on the Grid
Statistics per month averaged over the last three months

ESD data analysis	AOD data analysis	Nano AOD data analysis
• 160 train runs	• 560 train runs	• 10 train runs
• avg 3.2 CPU years/train run	• avg 1.2 CPU years/train run	• avg 80 CPU days/train run
• 45% of LEGO train run time	• 55% of LEGO train run time	• <<1% of LEGO train run time

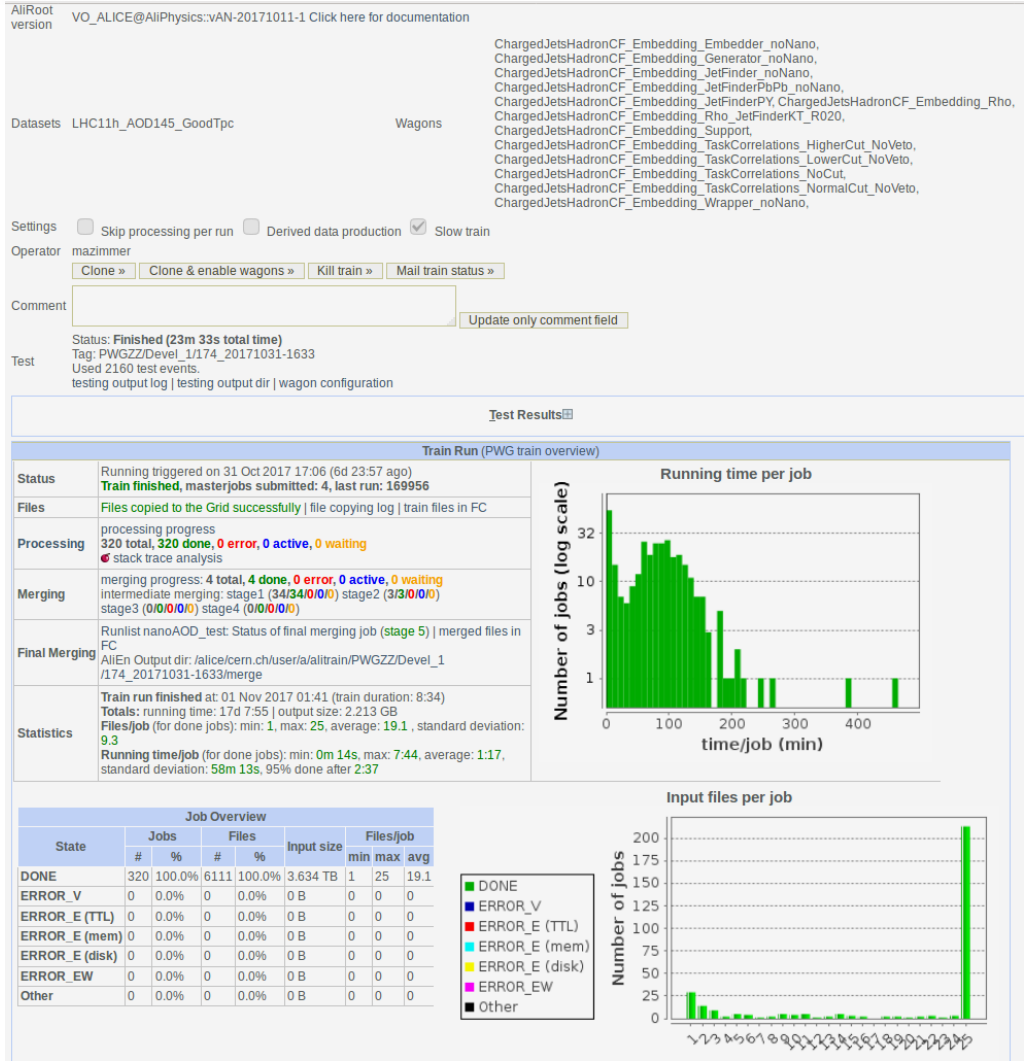
ESD Analysis → **AOD Analysis** → **Nano AOD Analysis**

What are Nano AODs

- Nano AODs duplicate a dataset on the Grid
  - Nano AODs file size is smaller than for AODs
  - Reduce track information to what is really needed
- Nano AODs are created in a dedicated LEGO train run as derived datasets
  - No merging step for Nano AODs
- Analysis of nano AODs
  - Behave like normal datasets in the train interface
  - Analysis code needs a few code adjustments

Nano AODs trade memory for CPU time

Creating Nano AODs



AliRoot version: VO_ALICE@AliPhysics::vAN-20171011-1 Click here for documentation

Datasets: LHC11h_AOD145_GoodTpc

Wagons: ChargedJetsHadronCF_Embedding_Embedder_noNano, ChargedJetsHadronCF_Embedding_Generator_noNano, ChargedJetsHadronCF_Embedding_JetFinder_noNano, ChargedJetsHadronCF_Embedding_JetFinderPbPb_noNano, ChargedJetsHadronCF_Embedding_JetFinderPY, ChargedJetsHadronCF_Embedding_Rho, ChargedJetsHadronCF_Embedding_Rho_JetFinderKT_R020, ChargedJetsHadronCF_Embedding_Support, ChargedJetsHadronCF_Embedding_TaskCorrelations_HigherCut_NoVeto, ChargedJetsHadronCF_Embedding_TaskCorrelations_LowerCut_NoVeto, ChargedJetsHadronCF_Embedding_TaskCorrelations_NoCut, ChargedJetsHadronCF_Embedding_TaskCorrelations_NormalCut_NoVeto, ChargedJetsHadronCF_Embedding_Wrapper_noNano,

Settings: Skip processing per run | Derived data production | Slow train

Operator: mazimmer

Clone » | Clone & enable wagons » | Kill train » | Mail train status »

Comment | Update only comment field

Test: Status: **Finished (23m 33s total time)**
Tag: PWGZZ/Devel_1/174_20171031-1633
Used 2160 test events.
testing output log | testing output dir | wagon configuration

Test Results

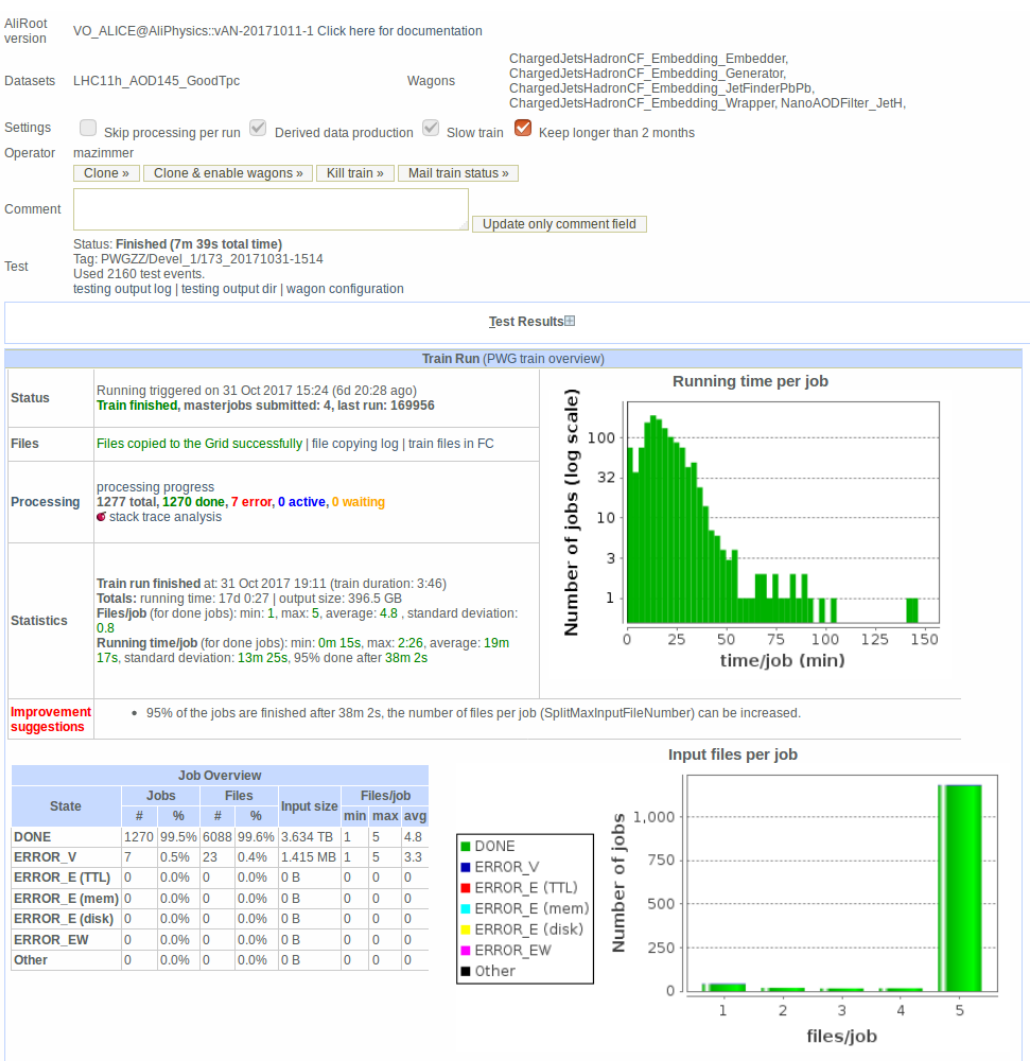
Train Run (PWG train overview)

Status	Running triggered on 31 Oct 2017 17:06 (6d 23:57 ago) **Train finished, masterjobs submitted: 4, last run: 169956**
Files	Files copied to the Grid successfully | file copying log | train files in FC
Processing	processing progress **320 total, 320 done, 0 error, 0 active, 0 waiting** stack trace analysis
Merging	merging progress: **4 total, 4 done, 0 error, 0 active, 0 waiting** intermediate merging: stage1 (**34/34/0/0/0**) stage2 (**3/3/0/0/0**) stage3 (**0/0/0/0/0**) stage4 (**0/0/0/0/0**)
Final Merging	Runlist nanoAOD_test: Status of final merging job (**stage 5**) | merged files in FC AliEn Output dir: /alice/cern.ch/user/a/alitrain/PWGZZ/Devel_1/174_20171031-1633/merge
Statistics	**Train run finished** at: 01 Nov 2017 01:41 (train duration: 8:34) **Totals:** running time: 17d 7:55 | output size: 2.213 GB **Files/job** (for done jobs): min: 1, max: 25, average: 19.1 , standard deviation: 9.3 **Running time/job** (for done jobs): min: 0m 14s, max: 7:44, average: 1:17, standard deviation: 58m 13s, 95% done after 2:37

Job Overview

State	Jobs #	Jobs %	Files #	Files %	Input size	Files/job min	Files/job max	Files/job avg
DONE	320	100.0%	6111	100.0%	3.634 TB	1	25	19.1
ERROR_V	0	0.0%	0	0.0%	0 B	0	0	0
ERROR_E (TTL)	0	0.0%	0	0.0%	0 B	0	0	0
ERROR_E (mem)	0	0.0%	0	0.0%	0 B	0	0	0
ERROR_E (disk)	0	0.0%	0	0.0%	0 B	0	0	0
ERROR_EW	0	0.0%	0	0.0%	0 B	0	0	0
Other	0	0.0%	0	0.0%	0 B	0	0	0

AOD analysis



AliRoot version: VO_ALICE@AliPhysics::vAN-20171011-1 Click here for documentation

Datasets: LHC11h_AOD145_GoodTpc

Wagons: ChargedJetsHadronCF_Embedding_Embedder, ChargedJetsHadronCF_Embedding_Generator, ChargedJetsHadronCF_Embedding_JetFinderPbPb, ChargedJetsHadronCF_Embedding_Wrapper, NanoAODFilter_JetH,

Settings: Skip processing per run | Derived data production | Slow train | Keep longer than 2 months

Operator: mazimmer

Clone » | Clone & enable wagons » | Kill train » | Mail train status »

Comment | Update only comment field

Test: Status: **Finished (7m 39s total time)**
Tag: PWGZZ/Devel_1/173_20171031-1514
Used 2160 test events.
testing output log | testing output dir | wagon configuration

Test Results

Train Run (PWG train overview)

Status	Running triggered on 31 Oct 2017 15:24 (6d 20:28 ago) **Train finished, masterjobs submitted: 4, last run: 169956**
Files	Files copied to the Grid successfully | file copying log | train files in FC
Processing	processing progress **1277 total, 1270 done, 7 error, 0 active, 0 waiting** stack trace analysis
Statistics	**Train run finished** at: 31 Oct 2017 19:11 (train duration: 3:46) **Totals:** running time: 17d 0:27 | output size: 396.5 GB **Files/job** (for done jobs): min: 1, max: 5, average: 4.8 , standard deviation: 0.8 **Running time/job** (for done jobs): min: 0m 15s, max: 2:26, average: 19m 17s, standard deviation: 13m 25s, 95% done after 38m 2s
Improvement suggestions	• 95% of the jobs are finished after 38m 2s, the number of files per job (SplitMaxInputFileNumber) can be increased.

Job Overview

State	Jobs #	Jobs %	Files #	Files %	Input size	Files/job min	Files/job max	Files/job avg
DONE	1270	99.5%	6088	99.6%	3.634 TB	1	5	4.8
ERROR_V	7	0.5%	23	0.4%	1.415 MB	1	5	3.3
ERROR_E (TTL)	0	0.0%	0	0.0%	0 B	0	0	0
ERROR_E (mem)	0	0.0%	0	0.0%	0 B	0	0	0
ERROR_E (disk)	0	0.0%	0	0.0%	0 B	0	0	0
ERROR_EW	0	0.0%	0	0.0%	0 B	0	0	0
Other	0	0.0%	0	0.0%	0 B	0	0	0

Nano AOD creation

Analysing Nano AODs

AliRoot version: VO_ALICE@AliPhysics::vAN-20171011-1 Click here for documentation

Datasets: LHC11h_AOD145_GoodTpc

Wagons: ChargedJetsHadronCF_Embedding_Embedder_noNano, ChargedJetsHadronCF_Embedding_Generator_noNano, ChargedJetsHadronCF_Embedding_JetFinder_noNano, ChargedJetsHadronCF_Embedding_JetFinderPbPb_noNano, ChargedJetsHadronCF_Embedding_JetFinderPY, ChargedJetsHadronCF_Embedding_Rho, ChargedJetsHadronCF_Embedding_Rho_JetFinderKT_R020, ChargedJetsHadronCF_Embedding_Support, ChargedJetsHadronCF_Embedding_TaskCorrelations_HigherCut_NoVeto, ChargedJetsHadronCF_Embedding_TaskCorrelations_LowerCut_NoVeto, ChargedJetsHadronCF_Embedding_TaskCorrelations_NoCut, ChargedJetsHadronCF_Embedding_TaskCorrelations_NormalCut_NoVeto, ChargedJetsHadronCF_Embedding_Wrapper_noNano,

Settings: Skip processing per run | Derived data production | Slow train

Operator: mazimmer

Clone » | Clone & enable wagons » | Kill train » | Mail train status »

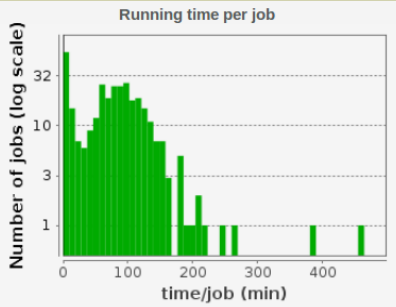
Comment | Update only comment field

Test: Status: **Finished (23m 33s total time)**
Tag: PWGZZ/Devel_1/174_20171031-1633
Used 2160 test events.
testing output log | testing output dir | wagon configuration

Test Results

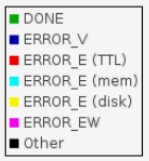
Train Run (PWG train overview)

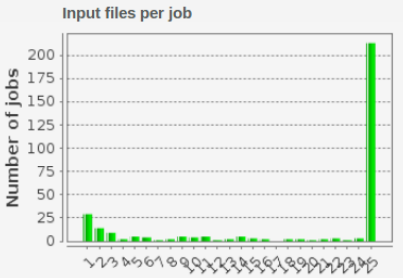
Status	Running triggered on 31 Oct 2017 17:06 (6d 23:57 ago) **Train finished, masterjobs submitted: 4, last run: 169956**
Files	Files copied to the Grid successfully | file copying log | train files in FC
Processing	processing progress **320 total, 320 done, 0 error, 0 active, 0 waiting** stack trace analysis
Merging	merging progress: **4 total, 4 done, 0 error, 0 active, 0 waiting** intermediate merging: stage1 (**34/34/0/0/0**) stage2 (**3/3/0/0/0**) stage3 (**0/0/0/0/0**) stage4 (**0/0/0/0/0**)
Final Merging	Runlist nanoAOD_test: Status of final merging job (stage 5) | merged files in FC AliEn Output dir: /alice/cern.ch/user/a/alitrain/PWGZZ/Devel_1 /174_20171031-1633/merge
Statistics	**Train run finished** at: 01 Nov 2017 01:41 (train duration: 8:34) **Totals:** running time: 17d 7:55 | output size: 2.213 GB **Files/job** (for done jobs): min: 1, max: 25, average: 19.1 , standard deviation: 9.3 **Running time/job** (for done jobs): min: 0m 14s, max: 7:44, average: 1:17, standard deviation: 58m 13s, 95% done after 2:37



Job Overview

State	Jobs #	Jobs %	Files #	Files %	Input size	Files/job min	Files/job max	Files/job avg
DONE	320	100.0%	6111	100.0%	3.634 TB	1	25	19.1
ERROR_V	0	0.0%	0	0.0%	0 B	0	0	0
ERROR_E (TTL)	0	0.0%	0	0.0%	0 B	0	0	0
ERROR_E (mem)	0	0.0%	0	0.0%	0 B	0	0	0
ERROR_E (disk)	0	0.0%	0	0.0%	0 B	0	0	0
ERROR_EW	0	0.0%	0	0.0%	0 B	0	0	0
Other	0	0.0%	0	0.0%	0 B	0	0	0





AOD analysis

AliRoot version: VO_ALICE@AliPhysics::vAN-20171013-1 Click here for documentation

Datasets: LHC11h_AOD145_GoodTpc_nano

Wagons: ChargedJetsHadronCF_Embedding_JetFinder_nano, ChargedJetsHadronCF_Embedding_JetFinderPbPb, ChargedJetsHadronCF_Embedding_JetFinderPY_nano, ChargedJetsHadronCF_Embedding_Rho_JetFinderKT_R020_nano, ChargedJetsHadronCF_Embedding_Rho_nano, ChargedJetsHadronCF_Embedding_Support_nano, ChargedJetsHadronCF_Embedding_TaskCorrelations_HigherCut_NoVeto_noNano, ChargedJetsHadronCF_Embedding_TaskCorrelations_LowerCut_NoVeto_noNano, ChargedJetsHadronCF_Embedding_TaskCorrelations_NoCut_noNano, ChargedJetsHadronCF_Embedding_TaskCorrelations_NormalCut_NoVeto_noNano, NanoAOD_Wrapper,

Settings: Skip processing per run | Derived data production | Slow train

Operator: mazimmer

Clone » | Clone & enable wagons » | Kill train » | Mail train status »

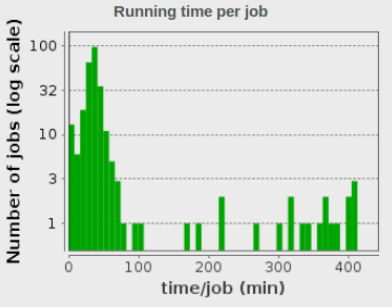
Comment | Update only comment field

Test: Status: **Finished (42m 11s total time)**
Tag: PWGZZ/Devel_1/176_20171101-0957
Used 5760 test events.
testing output log | testing output dir | wagon configuration

Test Results

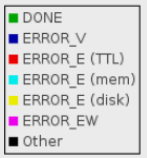
Train Run (PWG train overview)

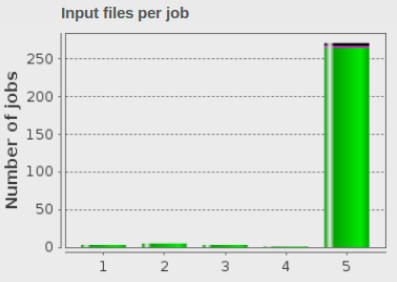
Status	Running triggered on 01 Nov 2017 10:54 (6d 6:16 ago) **Train finished, masterjobs submitted: 4, last run: 169956**
Files	Files copied to the Grid successfully | file copying log | train files in FC
Processing	processing progress **283 total, 277 done, 6 error, 0 active, 0 waiting** stack trace analysis
Merging	merging progress: **4 total, 4 done, 0 error, 0 active, 0 waiting** intermediate merging: stage1 (**13/13/0/0/0**) stage2 (**0/0/0/0/0**) stage3 (**0/0/0/0/0**) stage4 (**0/0/0/0/0**)
Final Merging	Runlist 1: Status of final merging job (stage 5) | merged files in FC AliEn Output dir: /alice/cern.ch/user/a/alitrain/PWGZZ/Devel_1 /176_20171101-0957/merge
Statistics	**Train run finished** at: 01 Nov 2017 20:52 (train duration: 9:58) **Totals:** running time: 10d 20:38 | output size: 828.6 MB **Files/job** (for done jobs): min: 1, max: 5, average: 4.9 , standard deviation: 0.6 **Running time/job** (for done jobs): min: 1m 36s, max: 6:52, average: 56m 27s, standard deviation: 1:19, 95% done after 5:13



Job Overview

State	Jobs #	Jobs %	Files #	Files %	Input size	Files/job min	Files/job max	Files/job avg
DONE	277	97.9%	1351	97.8%	700.8 GB	1	5	4.9
ERROR_V	0	0.0%	0	0.0%	0 B	0	0	0
ERROR_E (TTL)	0	0.0%	0	0.0%	0 B	0	0	0
ERROR_E (mem)	0	0.0%	0	0.0%	0 B	0	0	0
ERROR_E (disk)	0	0.0%	0	0.0%	0 B	0	0	0
ERROR_EW	2	0.7%	10	0.7%	31.04 GB	5	5	5
Other	4	1.4%	20	1.4%	5.576 GB	5	5	5





Nano AOD analysis

Use Case of Nano AODs

- Analysis uses
  - **Limited subset** of ESD/AOD information
  - Events which pass a certain **event cut**
  - Tracks which pass a certain **track cut**
  - Derived variables which take a long time to calculate them in a future reprocessing
- Analysis which take a lot of computing time
- Analysis are often repeated

When to NOT use Nano AODs

- Analysis uses
  - most of the ESD/AOD information
  - Event and track cuts change often
- Analysis is never repeated and no suitable Nano AODs exist
- Analysis finishes quickly (<50 CPU days)
  - Nano AODs will barely speed up the analysis

Statistics	**Train run finished** at: 22 Jul 2017 00:04 (train duration: 13:03) **Totals:** running time: 29d 21:31 output size: 390.6 MB **Files/job** (for done jobs): min: 1, max: 3, average: 2.9 , standard deviation: 0.4 **Running time/job** (for done jobs): min: 0m 2s, max: 5:01, average: 23m 6s, standard deviation: 32m 23s, 95% done after 1:24

NanoAOD Productions

- Roughly 20 NanoAOD productions per month
- Only 10 analysis of NanoAODs per month

- Derived datasets are used to
  - Create tree structures on the trains
  - Avoid merging errors
  - Analysis is continued on local machines or clusters
- Different use case than originally planned

New Nano AOD analysis

Improved Instructions

- Detailed twiki article with instructions
  - https://twiki.cern.ch/twiki/bin/viewauth/ALICE/NanoAOD
- Working implementation since summer for a correlation task
  - Two times faster than AOD analysis
  - Unfortunately no further analysis after summer

Jet embedding task

- The jet embedding has the following workflow
  - Embed pythia jets in PbPb events
  - Use jet finder to find jets in these PbPb events
  - Do analysis
- In the AOD analysis the embedding step is repeated in every train run
- Can create Nano AODs with embedded events
- Needs a wrapper class to keep AOD/Nano AOD usage transparent for the user

Resource consumption

AOD analysis

- takes 218 CPU days
- 3200 jobs
- Average analysis time 1:40 hours
- Train finished after 12:24 hours

Nano AOD analysis

- Takes 87 CPU days
- 3100 jobs
- Average analysis time 0:40 hours
- Train finished after 9:22 hours

On the Grid produced Nano AODs contain additional data

Problem in the implementation

- On the Grid produced results from Nano AODs contain roughly the double statistics compared to the analysis of AODs
- Locally Nano AODs and normal AODs result in same analysis results
  - This makes debugging very difficult
- This was never observed in any other Nano AOD production
  - Could be effect of the additional wrapper class?

Implementing a task with Nano AODs

- Changing a task is easy and straight forward
  - Normal AOD functions are AliVTrack information are usable the same way
  - Event information like centrality are accessed as a single double value instead of a centrality object
  - Additional custom variables can be implemented in a dedicated Nano AOD creation task
- Overall people hesitate to use the framework
- Debugging new productions require some knowledge about the Nano AOD framework and the analysis task

Summary

- New sets of Nano AODs were produced
- Nano AODs can be analyzed in less than half the time used for normal AOD productions
- Nano AODs are still poorly used
- For the jet embedding framework Nano AODs contain a not understood error
  - Further investigations are going on
- Debugging Nano AODs requires a person who understands the task and an expert for Nano AODs

BACKUP

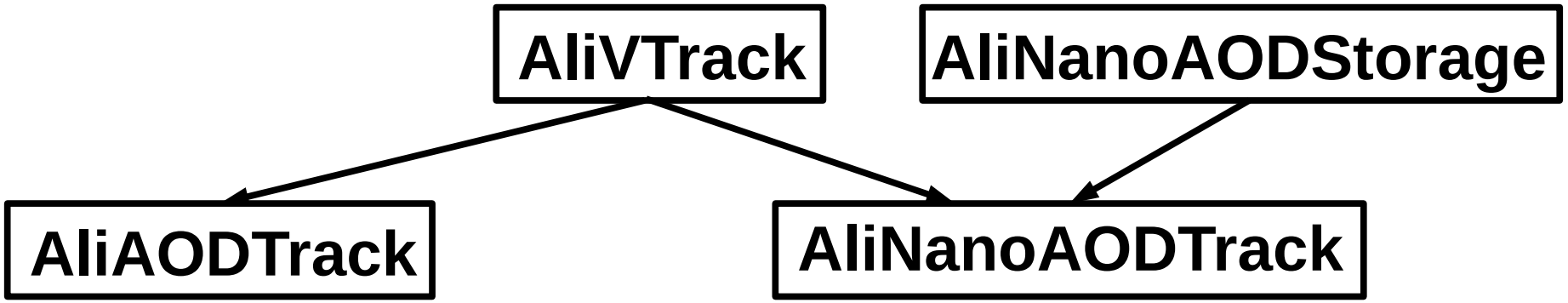
Analysis Types

AOD Analysis

- AliAODEvent
- AliAODTrack

Nano AOD Analysis

- AliAODEvent
- AliNanoAODTrack



- Calculates and stores basic track information

- Reimplements AOD functions
- Stores AOD information in local variables in AliNanoAODStorage

Event Information in Nano AODs

- Nano AODs contain the same AliAODEvent type as AOD events
  - No code change is necessary in the analysis for the event analysis
  - Not necessarily all AliAODevent content is available
- Centrality
  - V0M
  - Under development (TRK, CL1, CL2)
- Vertices (are always copied)
- Magnetic Field
- More Event properties can be implemented
  - Pile Up
  - ...

Track Information in Nano AODs

- p_T, θ, φ, X^2/ndf,
  pos X, pos Y, pos Z, pos DCA x, posDCA y, pos DCS Z,
  p_x at DCA, p_y at DCA, p_z at DCA, R at Absorber End,
  TPC Ncls, TPC NclsF, TPC N Crossed Rows,
  Track φ on EmCal, Track θ on EmCal, Track p_T on EmCal,
  ITS signal, TPC signal, TPC signal Tuned, TPC signal N,
  TPC momentum, TPC Tgl, TOF signal, integrated length, TOF signal Tuned, HMPID signal, HMPID occupancy,
  TRD signal, TRD X^2, TRD N Slices
- Train Operators define included variables of a Nano AOD
- User Code can directly access nano AOD track information if the track is interpreted as AliVTrack
- Track can also be casted to AliNanoAODTrack

Track Information in Nano AODs

```
AliVTrack particle;
particle = (AliAODTrack) aodEvent->GetTrack(0);
particle = (AliNanoAODTrack) aodEvent->GetTrack(0);
double pT = particle→Pt();
```

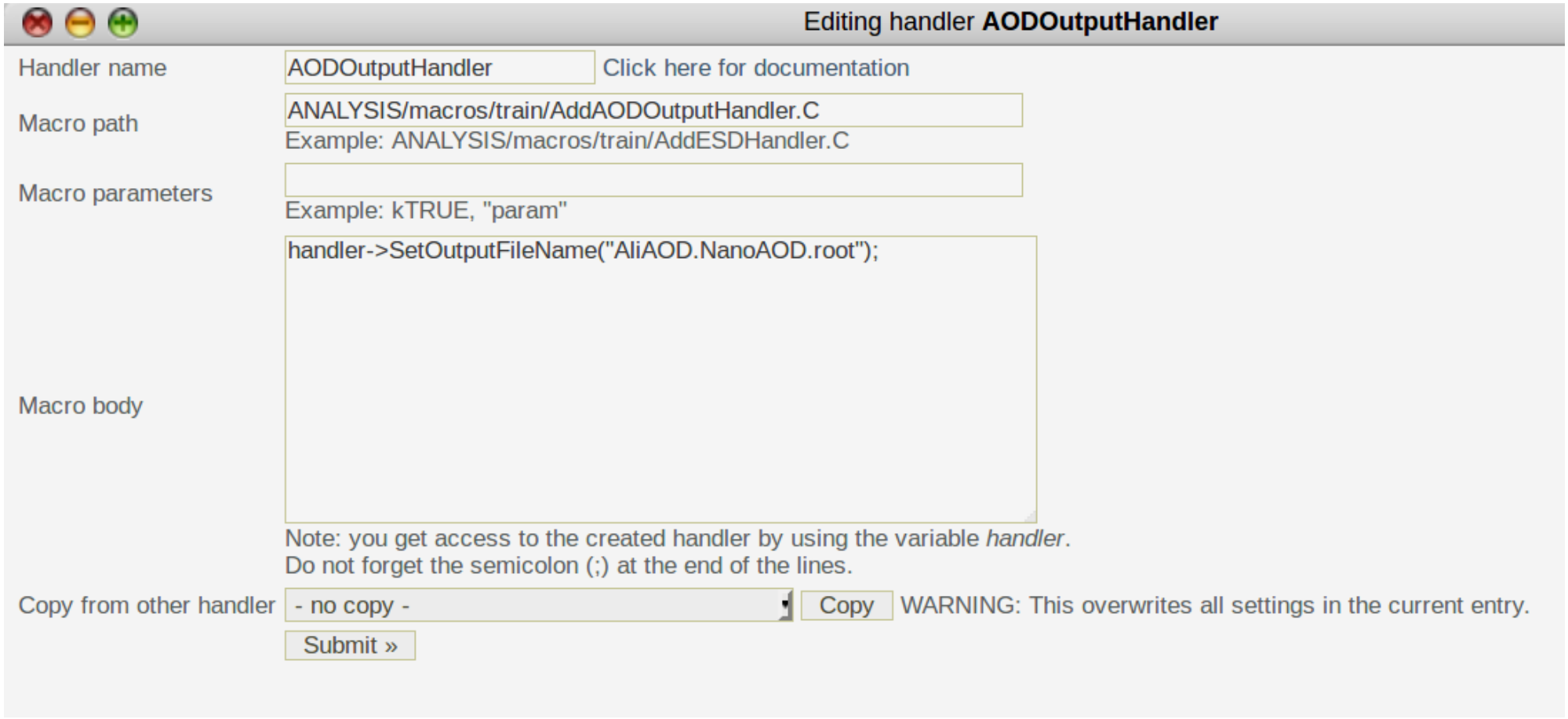
- Check if a certain method is implemented in AliNanoAODTrack
  - Implement it yourself if you need it
  - New derived variables can be stored in the Nano AODs
  - Contact m.zimmermann@cern.ch

Generating Nano AODs on the LEGO trains

Generation of Nano AODs

- Normal train train run
- Define AOD Output handler additionally to the AOD handler
- PWG/DevNanoAOD/AddTaskNanoAODFilter.C
- Output files are not merged
- Output is kept for 2 months
- Operator can define output to be kept for longer

AOD Output Handler



Nano AOD Wagon

Basic settings | Advanced settings | Subwagon configuration | Testing statistics | Click here for documentation

Wagon name: NanoAODFilter_Data_2plus1
NB. Only alphanumerical characters (a-z, A-Z, 0-9) and underscore (_) allowed.

Wagon owner: mazimmer — Only users who have a wagon in some train are displayed.

Wagon group: Default

Macro path: PWG/DevNanoAOD/AddTaskNanoAODFilter.C
By default macros are searched within $ALICE_PHYSICS. If you need a macro in $ALICE_ROOT, please specify $ALICE_ROOT in front of the path.
Examples: PWGCF/macros/AddTaskPhiCorrelations.C or $ALICE_ROOT/ANALYSIS/macros/AddTaskPIDResponse.C

☐ AddTask macro needs AliEn connection

Macro parameters: 0, 0
Example: kTRUE, "param"

Macro customization:

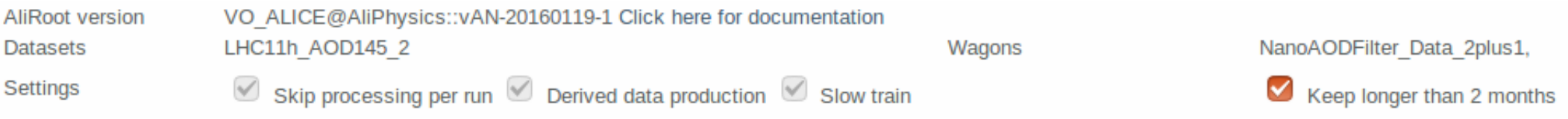
```
AliAnalysisNanoAODTrackCuts* trk = new AliAnalysisNanoAODTrackCuts;
trk->SetBitMask((1 << 8) | (1 << 9)); // hybrid 2011
//trk->SetBitMask((1 << 5) | (1 << 6)); // global 2011
//trk->SetBitMask((1 << 4) | (1 << 8)); // hybrid 2011 for pp
trk->SetMaxEta(0.9);
trk->SetMinPt(1.0);

AliAnalysisNanoAODEventCuts* evt = new AliAnalysisNanoAODEventCuts;
evt->SetVertexRange(8);

__R_ADDTASK__->SetTrkCuts(trk);
__R_ADDTASK__->SetEvtCuts(evt);
__R_ADDTASK__->SetSetter(new AliNanoAODSimpleSetter);
__R_ADDTASK__->SetVarList("pt,theta,phi");
__R_ADDTASK__->SetVarListHead("cstCentr,cstMagField");
//TODO centrality, vtx, magnetic field

__R_ADDTASK__->SelectCollisionCandidates(AliVEvent::kMB| AliVEvent::kCentral | AliVEvent::kSemiCentral);
```

Note: you get access to the created task by using the variable *__R_ADDTASK__*.
Do not forget the semicolon (;) at the end of the lines.
Example: __R_ADDTASK__->SelectCollisionCandidates(AliVEvent::kAnyINT);

Libraries:
Note: separate libraries with comma (,); do not specify *lib* in front
Example: CORRFW,EMCALUtils

Submit »

Train Run Configuration



- Derived data production → defines this train run as Nano AOD dataset
- Slow train → forbids to kill the last train jobs to speed up the analysis
- Keep longer than 2 months → analysis files will not be deleted after 2 months

Generation of Nano AODs

AliRoot version	VO_ALICE@AliPhysics::vAN-20160119-1 Click here for documentation		
Datasets	LHC11h_AOD145_2	Wagons	NanoAODFilter_Data_2plus1,
Settings	Skip processing per run, Derived data production, Slow train		Keep longer than 2 months
Operator	mazimmer		
	Clone » | Clone & enable wagons » | Kill train » | Mail train status »		
Comment	full nano AOD production with min Pt 1GeV — Update only comment field		
Test	Status: **Finished (5m 31s total time)** Tag: PWGZZ/Devel_1/106_20160120-1236 Used 6920 test events. testing output log | testing output dir | wagon configuration		

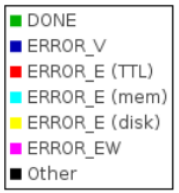
Test Results

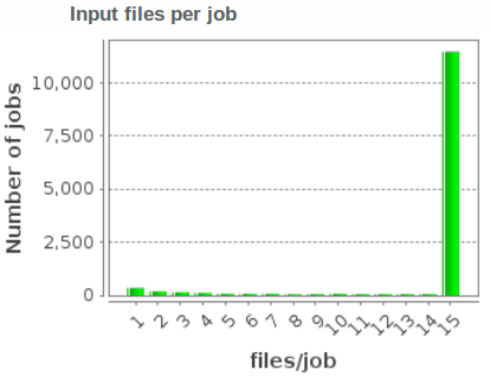
Train Run (PWG train overview)

Status	Running triggered on 20 Jan 2016 12:47 (1y 181d ago) **Train finished, masterjobs submitted: 12, last run: 170036**
Files	Files copied to the Grid successfully | file copying log | train files in FC
Processing	processing progress **12925 total, 12889 done, 36 error, 0 active, 0 waiting**
Statistics	**Train run finished** at: 21 Jan 2016 11:58 (train duration: 23:11) **Totals:** running time: 1y 16d | output size: 4.45 TB **Files/job** (for done jobs): min: 1, max: 15, average: 13.9 , standard deviation: 3.4 **Running time/job** (for done jobs): min: 0m 18s, max: 18:23, average: 42m 37s, standard deviation: 49m 19s, 95% done after 2:12

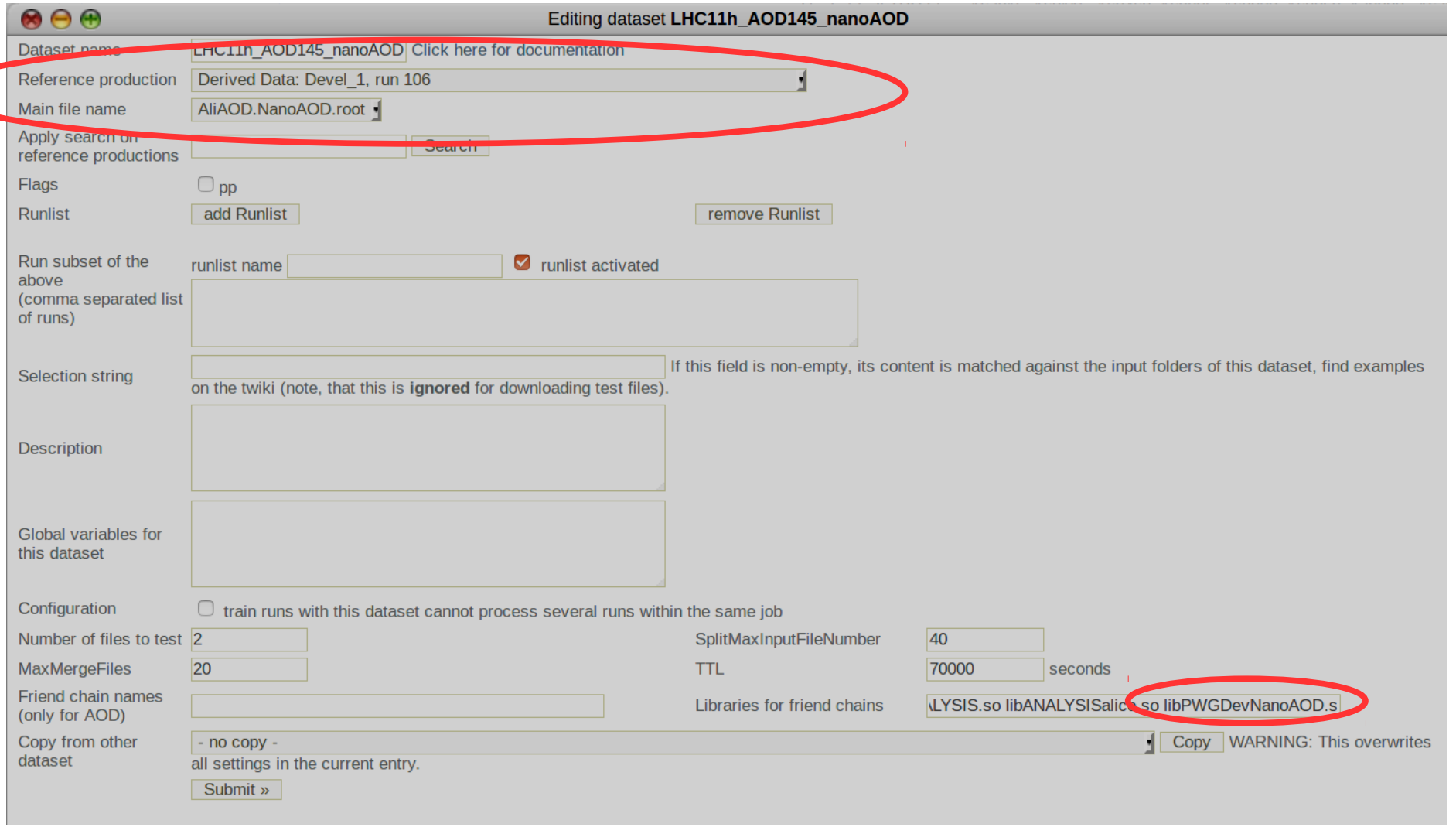
Job Overview

State	Jobs #	Jobs %	Files #	Files %	Input size	Files/job min	Files/job max	Files/job avg
DONE	12889	99.7%	179392	99.8%	107.1 TB	1	15	13.9
ERROR_V	17	0.1%	89	0.0%	61.03 GB	1	15	5.2
ERROR_E (TTL)	0	0.0%	0	0.0%	0 B	0	0	0
ERROR_E (mem)	0	0.0%	0	0.0%	0 B	0	0	0
ERROR_E (disk)	0	0.0%	0	0.0%	0 B	0	0	0
ERROR_EW	0	0.0%	0	0.0%	0 B	0	0	0
Other	19	0.1%	281	0.2%	172.7 GB	11	15	14.8





Derived Dataset



Derived Dataset

- Derived datasets look like normal datasets
- Wagons are activated for them in the same way

- Dataset is ready to be used in the analysis

Workflow with Nano AODs

- Create Nano AODs with many variables
  - Do event and track cuts in the analysis
  - Multiple analyses can use the same Nano AODs
- Create Nano AODs only out of events and tracks which pass the correspondent cuts
  - Nano AODs are very efficient
- Nano AODs can be recreated from time to time
  - Multiple analyses of the same Nano AODs
  - It is a waste of resources to create Nano AODs which are only analyzed once

Nano AODs with your Cuts

- Create your own Event and track cuts for the nano AOD generation
- Define your own Cuts in a file analogue to DevNanoAOD/AliAnalysisNanoAODCuts.cxx
  - This cannot inherit from any user class
  - use classes from $ALICE_PHYSICS/PWG/DevNanoAOD
- Create these track and event cuts in the wagon defintion of the Nano AOD generation

Try out Nano AODs locally

- Try out nano AODs locally
- Generate them with the local train test
  - https://twiki.cern.ch/twiki/bin/viewauth/ALICE/AnalysisTrains#Local_train_test
- Analyze these Nano AODs with another local train test by changing the input file to the newly produced nanoAOD