Practical introduction to reconstruction

Tutorial: Monte Carlo in ATLAS

Christian Ohm

September 28, 2015

BERKELEY LAB
Lawrence Berkeley National Laboratory

About this talk

- This talk is a brief practical introduction to reconstruction in ATLAS
- It will be from a *user* perspective, i.e. how to *run* reconstruction (devs may need to know *a lot* more details)
- I will try to give general advice and point to documentation and e-groups for more info and support

Let's get going!

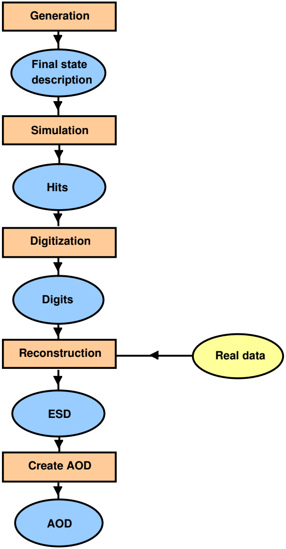
About this talk

- This talk is a brief practical introduction to reconstruction in ATLAS
- It will be from a *user* perspective, i.e. how to *run* reconstruction (devs may need to know *a lot* more details)
- I will try to give general advice and point to documentation and e-groups for more info and support

Let's get going!

"An expert is a person who has made all the mistakes that can be made in a very narrow field"
– Niels Bohr

Overview and data formats



Generation

Final state description

Simulation

Hits

Digitization

Digits

Reconstruction

Real data

ESD

Create AOD

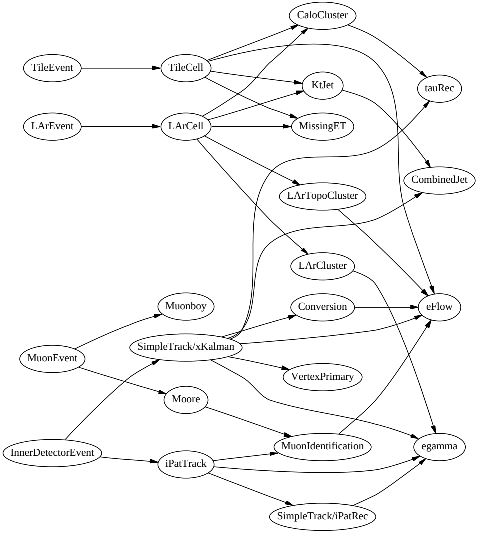
AOD

- Input: `RAW` (data) or `RDO` (MC)
- Outputs:
  - `(x)AOD`: Analysis Object Data
  - `ESD`: Event Summary Data
  - `HIST`: files with histograms from monitoring algorithms
  - `TAG`: lightweight flat ntuple with summary info[a]
  - `DAOD/DESD`: dedicated derived formats for physics and performance

NB! Reconstruction is the same for data and MC! ⇒ Reco does not need to know about what kind of physics process it's being fed

[a] Used in TADA for fast physics monitoring

What happens in the reconstruction step?



- The reconstruction in ATLAS is divided in many steps, e.g.
  1. read-out signals ⇒ tracker hits, local energy deposits
  2. several hits in ID ⇒ tracks
  3. calo cells ⇒ clusters
  4. tracks + clusters ⇒ electron candidates
  5. calibration and corrections are applied

Figure on left shows the “reco flow” for release. . . 7.4.0 (!!!)

Configuration

Many settings can/need to be specified:

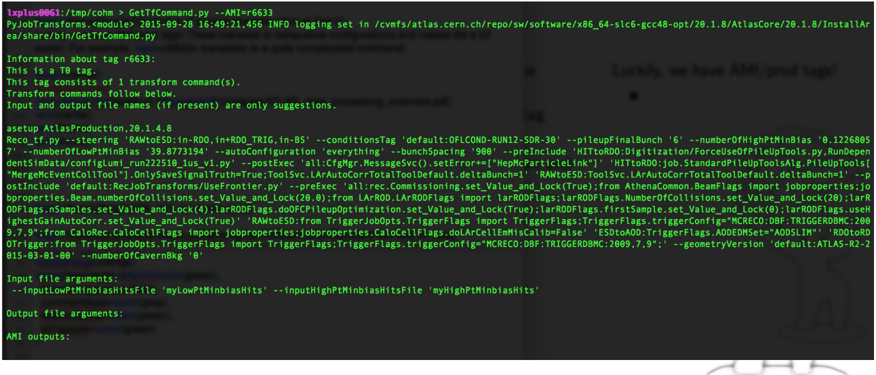
- Conditions data global tag
- Detector geometry
- Pileup settings
- Trigger menu settings
- Special detector settings, usually via `preExec` and `postExec` options (and `preInclude` and `postInclude`)

Autoconfiguration:

- Some of the settings above can be configured automatically
- To do this, the metadata in the input file is used (e.g. conditions and geo tags)

Configuration

Luckily, we have AMI/prodsys tags! These translate to exhaustive configurations and make life a lot easier.
For example, the `r6633` tag commonly used in MC15 production translates to a quite complicated command:



```
lxplus0061:/tmp/cohm > GetTfCommand.py --AMI=r6633
PyJobTransforms.<module> 2015-09-28 16:49:21,456 INFO logging set in /cvmfs/atlas.cern.ch/repo/sw/software/x86_64-slc6-gcc48-opt/20.1.8/AtlasCore/20.1.8/InstallAr
ea/share/bin/GetTfCommand.py

Information about tag r6633:
This is a T0 tag.
This tag consists of 1 transform command(s).
Transform commands follow below.
Input and output file names (if present) are only suggestions.

asetup AtlasProduction,20.1.4.8
Reco_tf.py --steering 'RAWtoESD:in-RDO,in+RDO_TRIG,in-BS' --conditionsTag 'default:OFLCOND-RUN12-SDR-30' --pileupFinalBunch '6' --numberOfHighPtMinBias '0.1226805
7' --numberOfLowPtMinBias '39.8773194' --autoConfiguration 'everything' --bunchSpacing '900' --preInclude 'HITtoRDO:Digitization/ForceUseOfPileUpTools.py,RunDepen
dentSimData/configLumi_run222510_1us_v1.py' --postExec 'all:CfgMgr.MessageSvc().setError+=["HepMcParticleLink"]' 'HITtoRDO:job.StandardPileUpToolsAlg.PileUpTools[
"MergeMcEventCollTool"].OnlySaveSignalTruth=True;ToolSvc.LArAutoCorrTotalToolDefault.deltaBunch=1' 'RAWtoESD:ToolSvc.LArAutoCorrTotalToolDefault.deltaBunch=1' --p
ostInclude 'default:RecJobTransforms/UseFrontier.py' --preExec 'all:rec.Commissioning.set_Value_and_Lock(True);from AthenaCommon.BeamFlags import jobproperties;jo
bproperties.Beam.numberOfCollisions.set_Value_and_Lock(20.0);from LArROD.LArRODFlags import larRODFlags;larRODFlags.NumberOfCollisions.set_Value_and_Lock(20);larR
ODFlags.nSamples.set_Value_and_Lock(4);larRODFlags.doOFCPileupOptimization.set_Value_and_Lock(True);larRODFlags.firstSample.set_Value_and_Lock(0);larRODFlags.useH
ighestGainAutoCorr.set_Value_and_Lock(True)' 'RAWtoESD:from TriggerJobOpts.TriggerFlags import TriggerFlags;TriggerFlags.triggerConfig="MCRECO:DBF:TRIGGERDBMC:200
9,7,9";from CaloRec.CaloCellFlags import jobproperties;jobproperties.CaloCellFlags.doLArCellEmMisCalib=False' 'ESDtoAOD:TriggerFlags.AODEDMSet="AODSLIM"' 'RDOtoRD
OTrigger:from TriggerJobOpts.TriggerFlags import TriggerFlags;TriggerFlags.triggerConfig="MCRECO:DBF:TRIGGERDBMC:2009,7,9";' --geometryVersion 'default:ATLAS-R2-2
015-03-01-00' --numberOfCavernBkg '0'

Input file arguments:
 --inputLowPtMinbiasHitsFile 'myLowPtMinbiasHits' --inputHighPtMinbiasHitsFile 'myHighPtMinbiasHits'

Output file arguments:

AMI outputs:
```

Ok, let's run some reco!

Standard dev test on data

```
# run a standard reco test on early 2015
# stable-beams data
asetup 20.1.8.1
Reco_tf.py --AMI=q431
```

More general example, running on MC RDO instead:

General (but unrealistic) example for MC

```
# run reco of MC RDO file
Reco_tf.py --inputRDOFile=your.RDO.pool.root \
--outputESDFile=myESD.pool.root \
--outputAODFile=myAOD.pool.root \
--outputHISTFile=myHIST.root
```

Where to look for more information

Twiki pages:

- ReconstructionIntegration - overview of organization and activities for combined reconstruction

egroups - when you need to ask for help:

- `hn-atlas-recoIntegration@cern.ch` - a good starting point for general reco issues

JIRA projects, for tracking bugs feature requests etc:

- ATLASRECTS JIRA tracker - for general reconstruction issues
- ATLASJT JIRA tracker - for issues specific to the Job Transforms

Meetings, usually the quickest way for finding out plans for releases etc:

- Weekly ATLAS Reconstruction Meeting (Tue 16:00 CET) - long-term development, reports from production usage
- Software & Computing Coordination (Thu 16:00 CET) - release planning, etc

Debugging

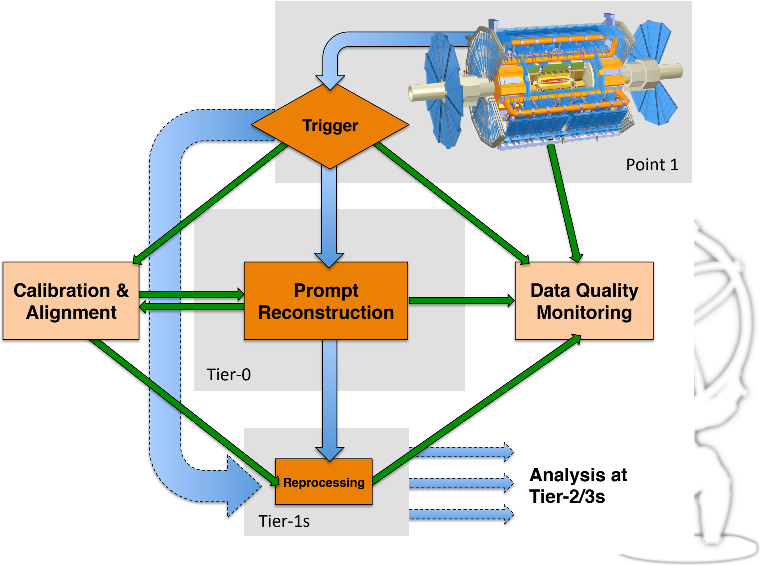
- The log file is your friend! Read it carefully.
- If the job crashes completely, the stacktrace usually provides clues
- If the transform recovers, look at the message summary at the end for `ERROR` or `FATAL` messages
- Before reporting, try to find a *complete* minimal working example (ideally running over only one event) - the more precise the report, the quicker you will get expert help!
- SoftwareDevelopmentWorkBookDebuggingCode twiki contains lots of useful info

Concluding remarks & advice

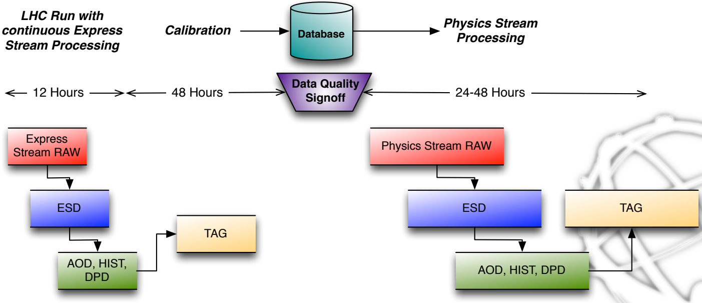
- The reconstruction treats the data and simulated detector signals in the MC in the same way
- The reconstruction is done in many steps, from low-level detector reconstruction to higher-level calibrated physics objects
- Reconstruction is a sweeping term used to wrap many tools and algorithms
- The configuration is (in principle) flexible, but has proven to be fragile at times
- Advice: if you want to run reconstruction yourself, always start with a configuration you know works
- If you run into problems, seek information (and help if needed) on the pages linked above
- Important: if the documentation is insufficient, constructive feedback is always very welcome

Back-up

Overview of prompt reconstruction at Tier-0



Overview of prompt reconstruction at Tier-0



LHC Run with continuous Express Stream Processing
Calibration
Database
Physics Stream Processing
12 Hours
48 Hours
Data Quality Signoff
24-48 Hours
Express Stream RAW
ESD
AOD, HIST, DPD
TAG
Physics Stream RAW
ESD
TAG
AOD, HIST, DPD