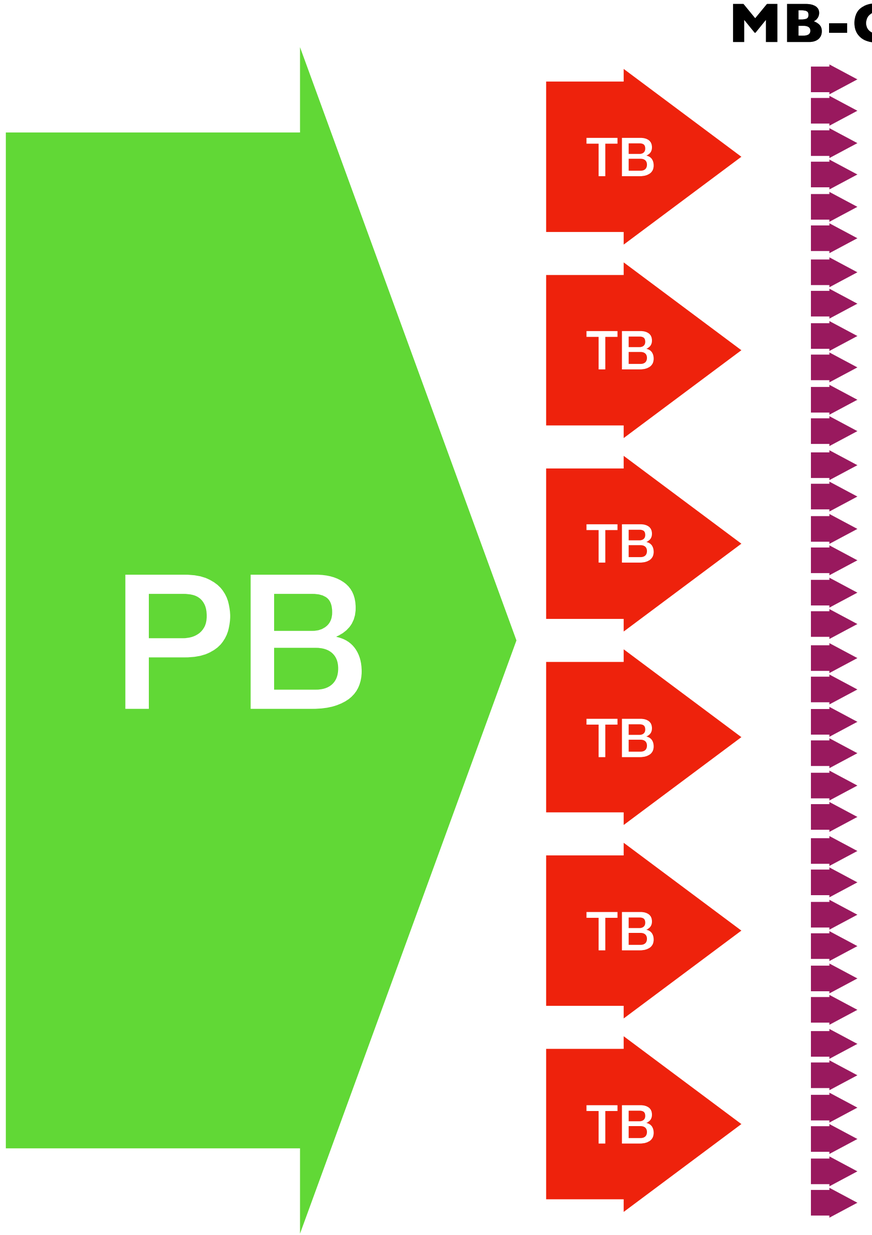
Introduction to the Derivation Framework

James Catmore (Oslo)

A feature common to most physics analyses

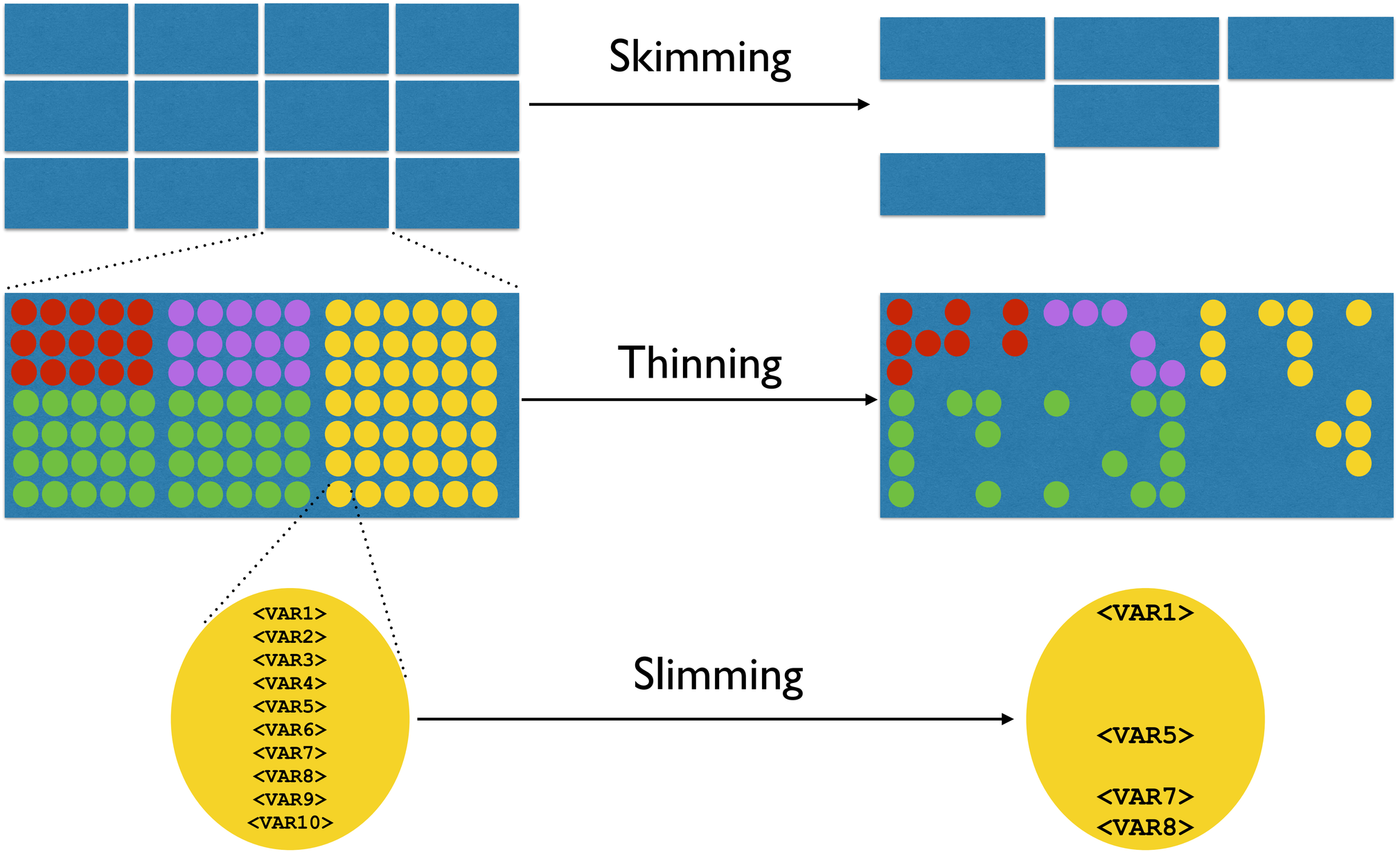


MB-GB

Full output of reconstruction, ~PB size	One format
Intermediate analysis format ~TB size	~100 formats
Final n-tuple ~MB-GB size	~1000 formats

- These formats tend to be specific to a single analysis or group of analyses
- Calibrations and common object selections are often applied as they are made
- They generally need to contain all variables needed for calculating systematics
- **Producing them is the purpose of the Derivation Framework and the topic of this talk**

Data reduction operations

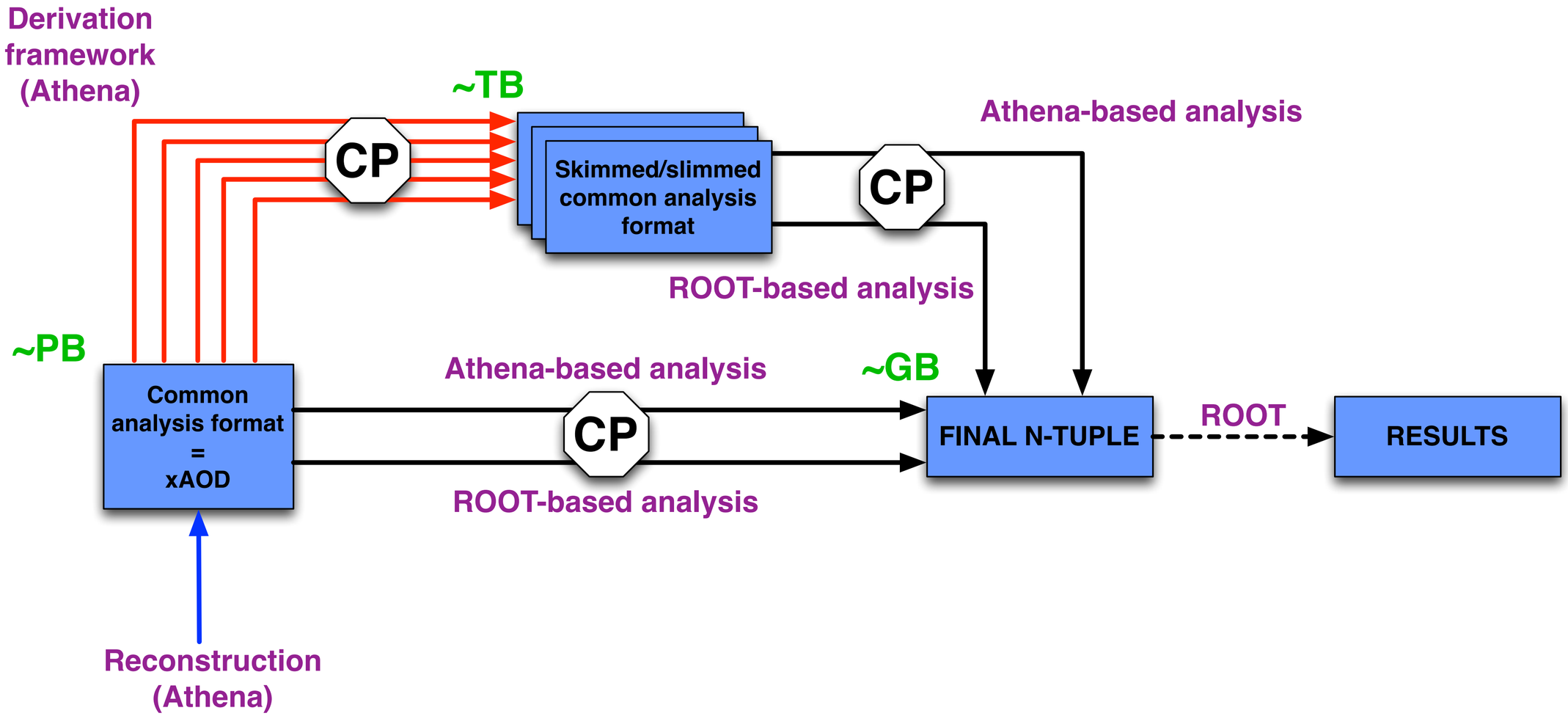


Skimming: removal of whole events based on pre-set criteria

Thinning: removal of whole objects within events based on pre-set criteria

Slimming: removal of variables within objects uniformly across events

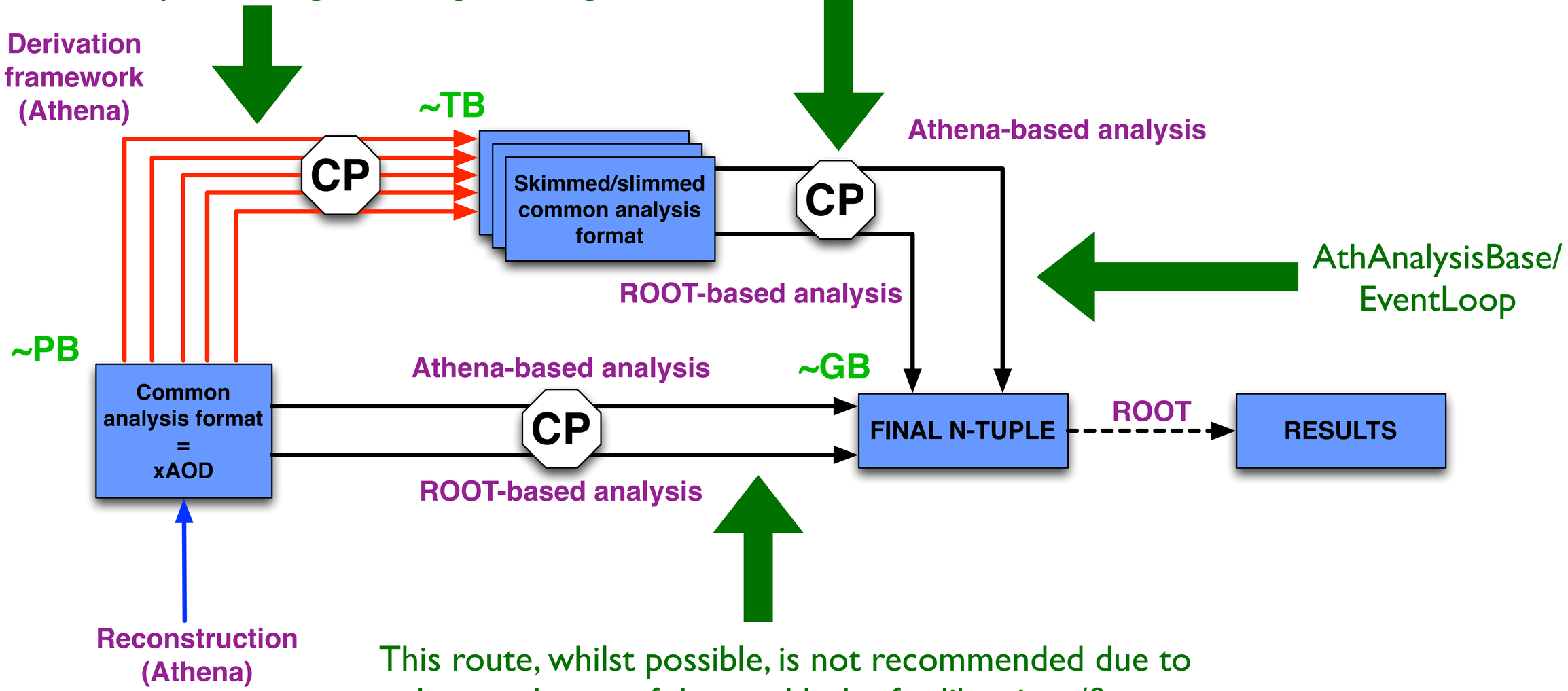
Run-II analysis model



Run-II analysis model

Topic of this talk: the "heavy lifting" to get from PB sized to TB sized datasets. Output remains in xAOD format but reduced by skimming, slimming, thinning

"CP" = calibrations and common object selections
These need to be applied to xAOD objects

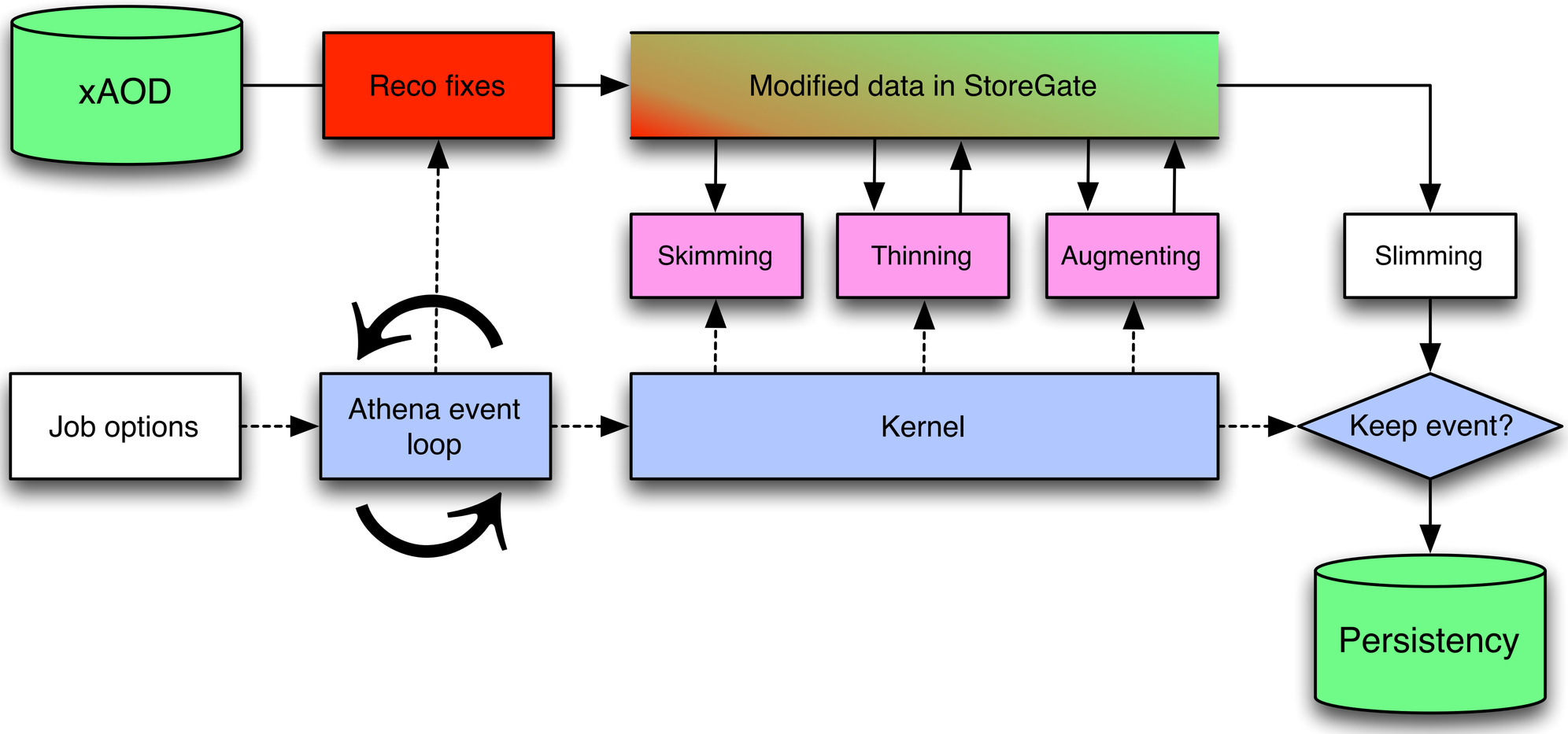


This route, whilst possible, is not recommended due to large volumes of data and lack of calibrations/fixes

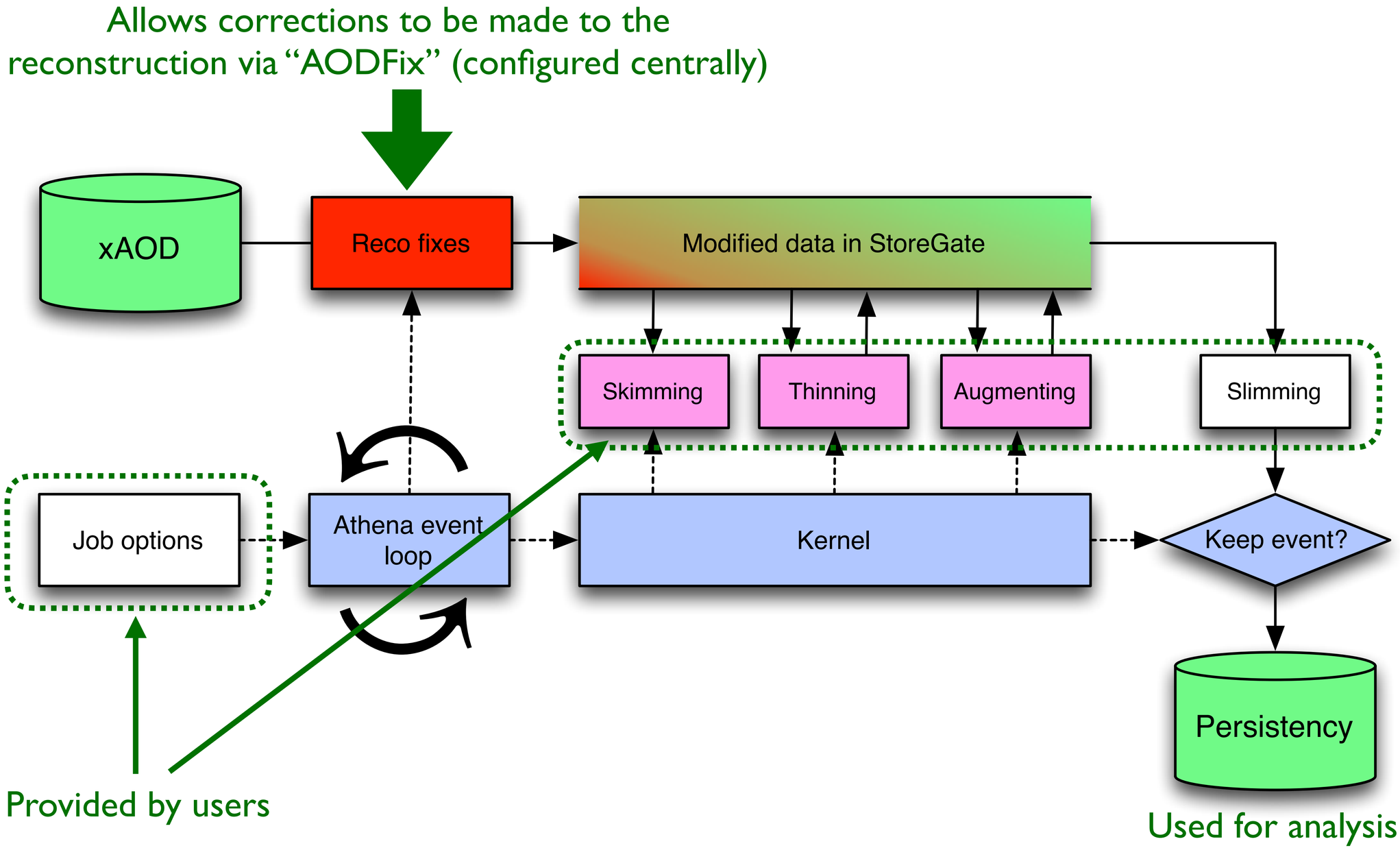
Derivation framework features

- Built on the main ATLAS data processing framework (Athena)
  - Enables re-running of parts of the reconstruction
  - Benefits from the Athena core software (algorithms and tools, whiteboard, thinning service, metadata handling, multiple outputs etc)
- Provides
  - interfaces for users to implement tools for skimming, thinning and augmenting their data
  - … and a set of central tools for commonly needed selections
  - a text-based event/object selection mechanism to minimise user-developed C++
  - built-in lists of variables required for each calibration/object selection/systematic tool used in the analyses (“smart slimming”)
  - detailed monitoring of CPU, skimming rates, overlaps between formats

Implementation



Implementation



Expression evaluation

- To avoid large numbers of C++ tools being written the event and object selection is configured where possible from the Python job options alone
  - Execution in C++ is performed by a single "expression evaluation" tool
- Allows arbitrarily complex selections of the following type to be made:
  - Events (slimming):
    - `count(Muons.pt > 25.0*GeV && Muons.eta < 2.5) >= 4`
  - Objects (thinning):
    - `InDetTrackParticles.pt > 5.0*GeV`
- Can access any variables in StoreGate (either from the input file or added by other tools) - also supports operators, unary mathematical functions, constants
- Text parsing is done in the initialise step (once per job) minimising the CPU overhead

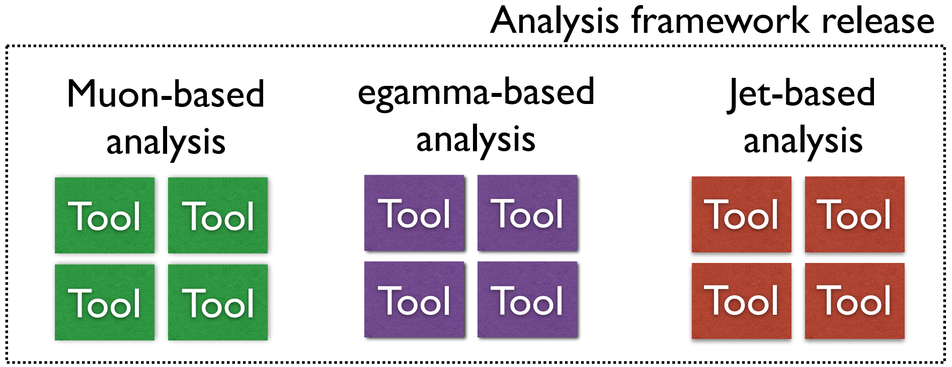
Slimming

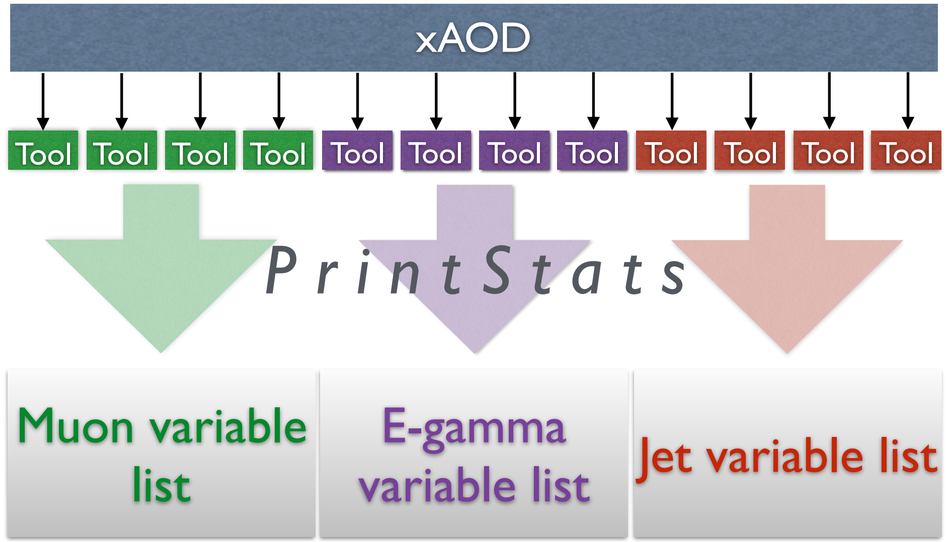
- Aim is to keep the variables needed for analysis and nothing more

1. We need to keep variables required by any of the tools in the analysis framework

2. Run over an xAOD, calling each of the tools in turn. The PrintStats service generates lists of all accessed variables.

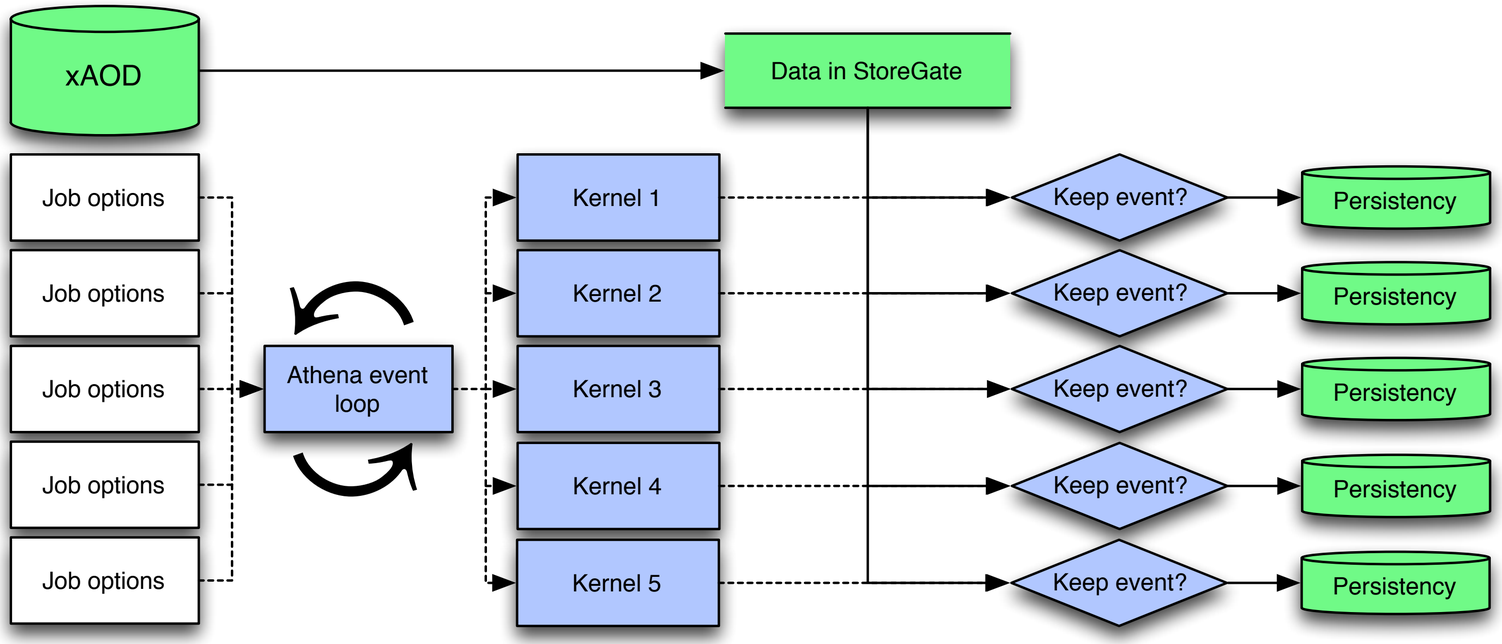
3. The variable lists are installed in the Derivation Framework release and automatically included. Extra variables can be added as needed.





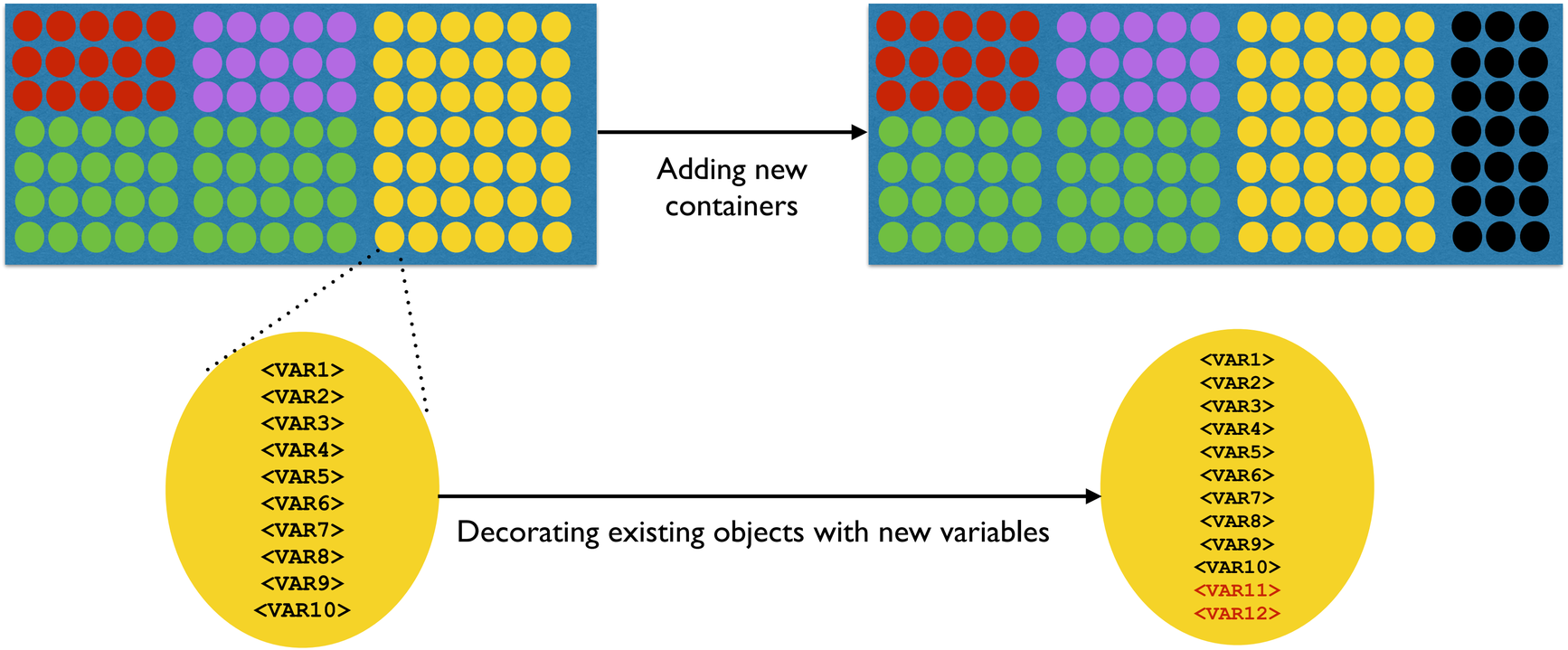
Multiple output streams

- Multiple outputs per input file produced in a single job
  - Steered by the existing multiple stream manager in Athena
  - Each output stream is independent of the others
- This allows a "train model" of central production: several outputs per input
- Each train typically handles 5-10 output formats from a single physics group



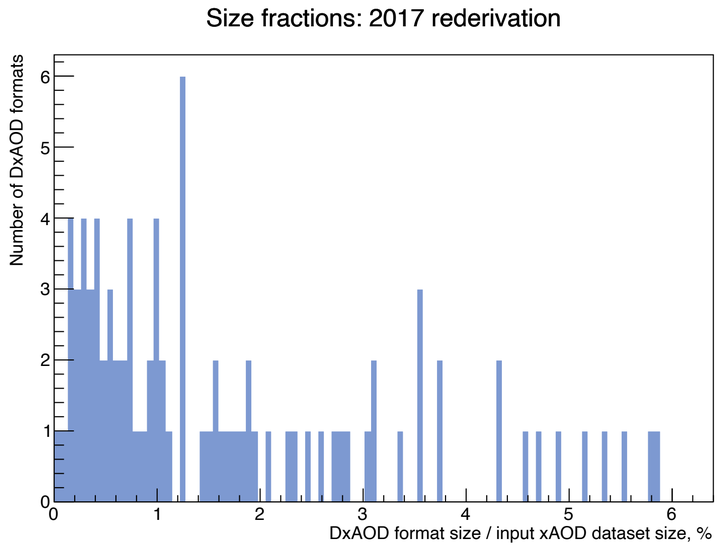
Augmentation

- New information (augmentation) is typically done in two ways:
  - Adding new reconstructed object containers: typically jets made with a modified algorithm.
  - Decorating existing objects with extra variables: typically the results of object selection by combined performance tools (e.g. "this is a good muon")
- Augmentation can be shared across a train, saving CPU

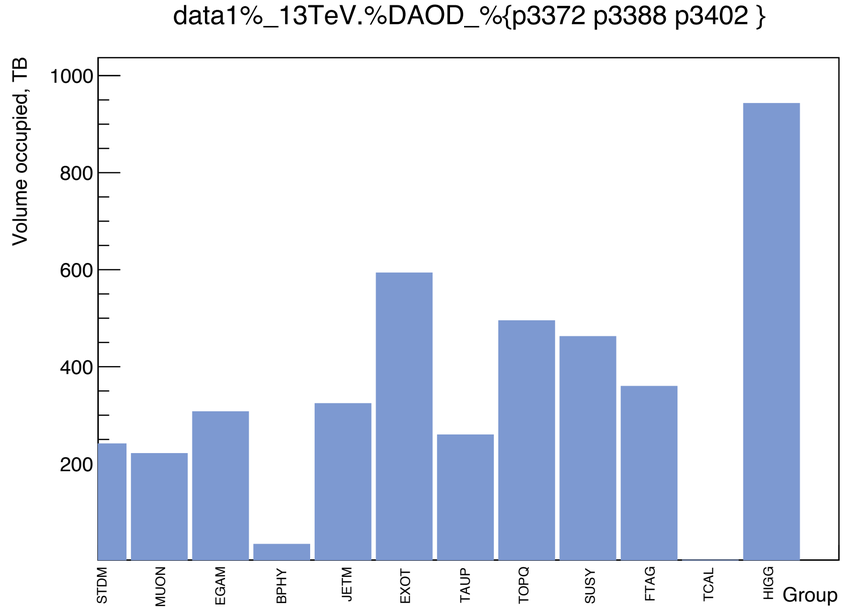


Implemented derivations

- No limit on the number of derivations: only on the total size
- It should be possible to analyse a derivation dataset on the grid with normal user privileges in approximately 1 day
- Budget:
  - total derivations size ≤ total xAOD size
  - With ~100 formats, each DAOD should aim to be ~1% of its input xAOD size

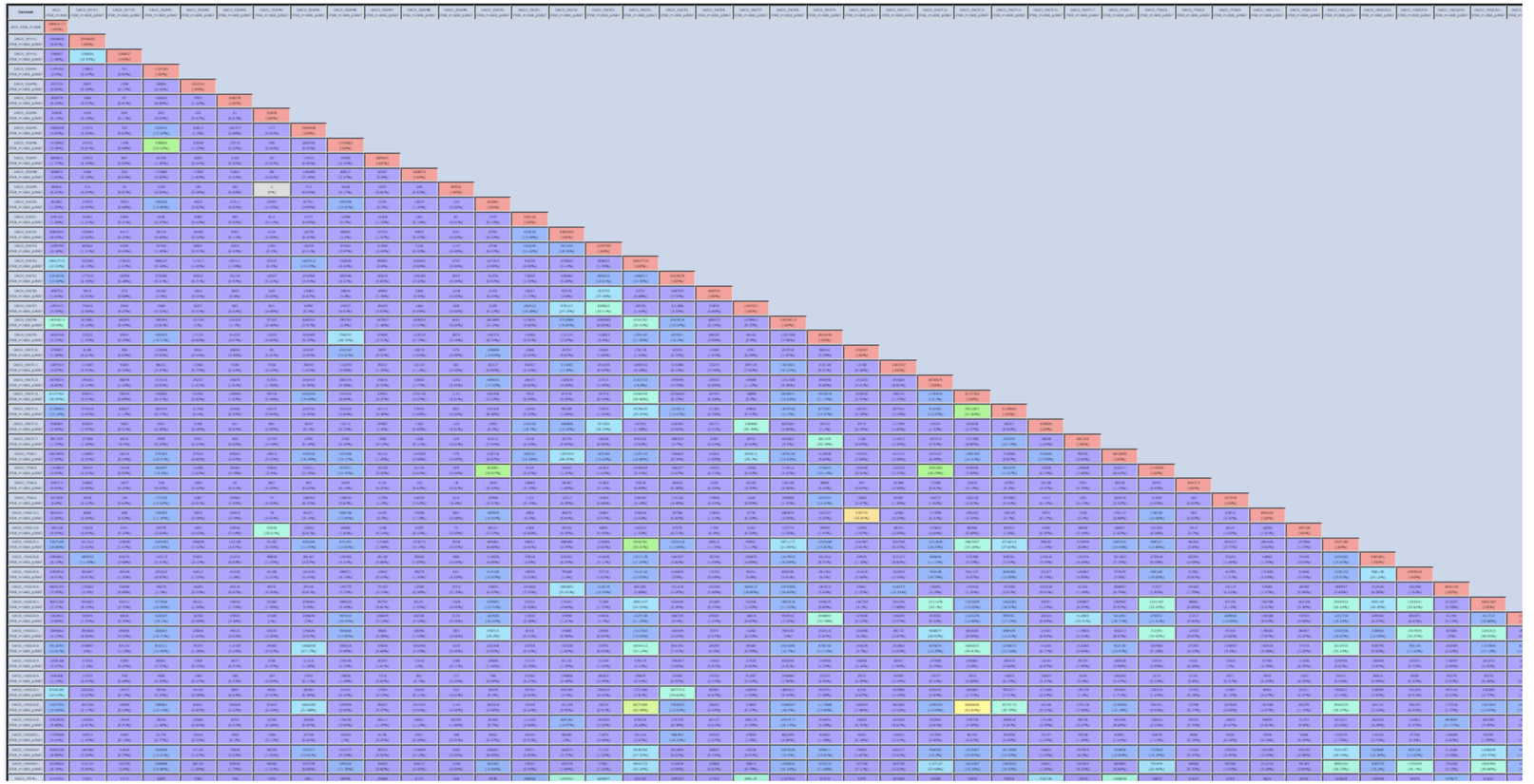


Full list of DAOD formats
(please do not edit)



Overlap monitoring

- We do not want to write out the same selections in multiple formats
  - If two formats strongly overlap, and they are large, we should merge them
- Event-wise overlaps can be monitored both in production using the EventIndex database and offline by Athena
  - Content-wise overlaps can also be monitored



An event-wise overlap plot: enables us to determine whether a pair of formats might be so similar that they can be merged:
https://atlas-tagservices.cern.ch/tagservices/RunBrowser/runBrowserReport/EventIndex.php?runs=300655&EIAction=DatasetOverlaps&stream=physics_Main

What do you need to know about the derivation framework?

- Users do not usually need to produce derivations themselves - it is done centrally
- Each group has a derivation production contact who is the link between physicists and the production team. They will handle
  - requests for new production of defined formats (e.g. new MC)
  - requests for definitions of new derivation formats
- Derivation contacts + production/software teams meet every Wednesday at 3pm - open meeting
- Production team documentation:
  - https://twiki.cern.ch/twiki/bin/view/AtlasProtected/DerivationProductionTeam
- Software documentation (including how to run example code):
  - https://twiki.cern.ch/twiki/bin/viewauth/AtlasProtected/DerivationFramework

Finding derivations on the grid

- (Almost) All derivations have the following name:
  - DAOD_<GROUP><NUMBER>
    - where GROUP is the group name and NUMBER is an integer
    - e.g. DAOD_TOPQ1
- Exception: Higgs
  - DAOD_HIGG<HSG number>D<NUMBER>
    - e.g. DAOD_HIGG5D1 (HSG 5)
- Information on derivation versions are on the DAOD production team page: https://twiki.cern.ch/twiki/bin/view/AtlasProtected/DerivationProductionTeam
  - Once you have the p-tag, getting the dataset names can be done easily using either AMI or DDM
- Information about the contents of each derivation is a bit limited at the moment:
  - look in the job options in GitLab for the precise definition

Package layout in Git

- https://gitlab.cern.ch/atlas/athena/tree/21.2/PhysicsAnalysis/DerivationFramework
  - DerivationFrameworkCore → master job options fragments, kernel algorithm, format list
  - DerivationFrameworkInterfaces → definitions of tool interfaces
  - DerivationFrameworkTools → common tools (mainly the generic selector at the moment)
  - DerivationFrameworkExamples → example implementations of derivations and tools
  - DerivationFrameworkART → nightly tests of all DAOD formats
  - DerivationFramework{BPhys}{EGamma}{Exotics}{FlavourTag}{HI}{Higgs}{InDet}{JetEtMiss}{MCTruth}{Muons}{SM}{SUSY}{Tau}{Top}

    → location for derivation job options and tools developed by groups

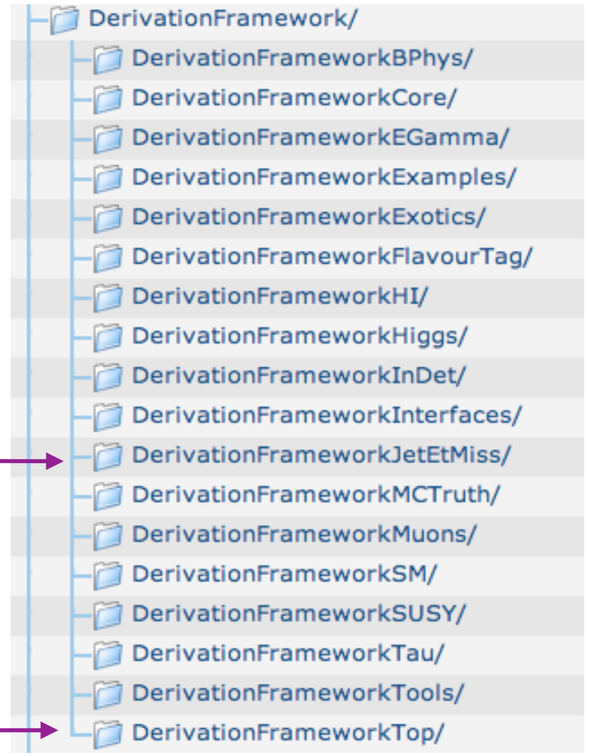
How is it run?


```
Reco_tf.py
--inputAODFile xaod.pool.root
--outputDAODFile test.pool.root
--reductionConf    JETM1 TOPQ1
```

Carriages

Output from the above command:

- log.AODtoDAOD
- runargs.AODtoDAOD.py
- runwrapper.AODtoDAOD.sh
- DAOD_JETM1.test.pool.root
- DAOD_TOPQ1.test.pool.root



Summary

- The derivation framework is Athena software to reduce PB-sized AOD datasets down to TB-sized formats for use by specific analyses (DAODs)
- This is done through three operations: skimming, slimming, thinning
- DAODs can also be "decorated" with information not found in the full AOD
  - Athena-only operations such as AODFix are carried out at this step as well
- Production is run centrally; physicists should go via their group contacts to request new production (having first discussed it in a physics group or subgroup)
- Each derivation should be approximately 1% of the input size - although MC will often be bigger
- Essentially all analysis should be done on DAOD

Bonus: changes for Run 3