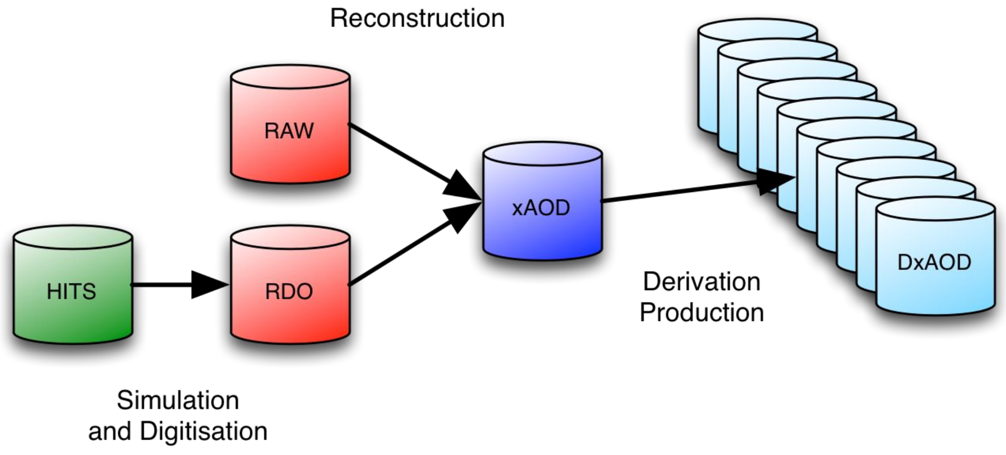
Data Analysis in ATLAS

Graeme Stewart
with thanks to Attila Krasznahorkay and Johannes Elmsheuser

University of Glasgow | School of Physics & Astronomy

ATLAS Data Flow into Analysis



- RAW detector data and simulated RDO data are reconstructed into our xAOD format
  - Called “xAOD” to make it clear that it has our Run 2 EDM
- From these xAOD events a number of derived formats are made, called DxAODs
  - There are about 100 of these derived formats, each targeting a particular group of analyses, each targeting 1% of the total size of the primary xAODs

xAOD to DxAOD

- Production of DxAODs is done via a train model
  - This is to amortise the costs of staging in the xAOD input, and repeating analysis tasks
  - About 14 trains are run, each producing ~7 outputs
  - Trains are organised based on the output size of the DxAODs (merging is easier that way)
    - Size is strongly correlated to production time; optimise delivery time to physics groups
- Each DxAOD is produced from the xAOD applying any/all of
  - Skimming - remove events that don’t pass some selection criteria
  - Slimming & Thinning - remove of whole/partial containers from events that are uninteresting
  - Enhancing - add additional content to the DxAOD
    - If multiple derivations all require the same enhancements they are done only once
    - In fact for 2017 ATLAS is moving most jet finding and flavour tagging into the derivations step - to save space in the primary xAOD and to catch final code improvements

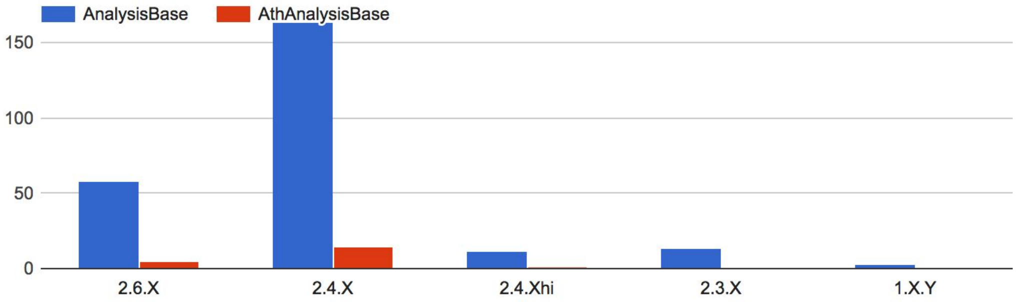
Conditions and Calibration

- First pass calibration is done in our calibration loop of 48 hours
  - Usual calibrations for beam spot, alignment, noisy channels, etc. derived from express stream
  - Then bulk reconstruction proceeds to produce xAODs at the Tier-0
  - These are lower level calibrations, stored in the database, used within Athena
- Refined calibrations from combined performance groups are prepared only once the bulk reconstruction is ready
  - This is rather a long process, taking several months
  - Serialised down the reconstruction chain (tracking, jet finding, MET, flavour tagging, …)
  - This process is fine tuned to maximise the physics performance
    - E.g., b-tagging requires several weeks to train their neural networks
    - So it is out of the question to do that in any calibration loop
  - Generally these calibrations may be used in DxAOD production itself (even applying on the fly fixes to xAOD)

CP Calibration Storage and Access

- Once CP calibrations are prepared they are stored on CVMFS in a group data area
  - Individual groups are responsible for the mechanics of versioning and ensuring that their software packages will find the correct version of the calibrations
  - In general a set of final calibrations are provided for each major software release, with a set of pre-recommendations coming from MC and then revised once detector data arrives
- These calibrations are accessed both in the xAOD→DxAOD step, and during the DxAOD analysis
- The vast majority of groups use these calibrations ‘as is’
  - In some special ‘bleeding edge’ cases the CP group and the analysis group may work together on refinements, e.g, the H→ZZ*→4l and egamma CP

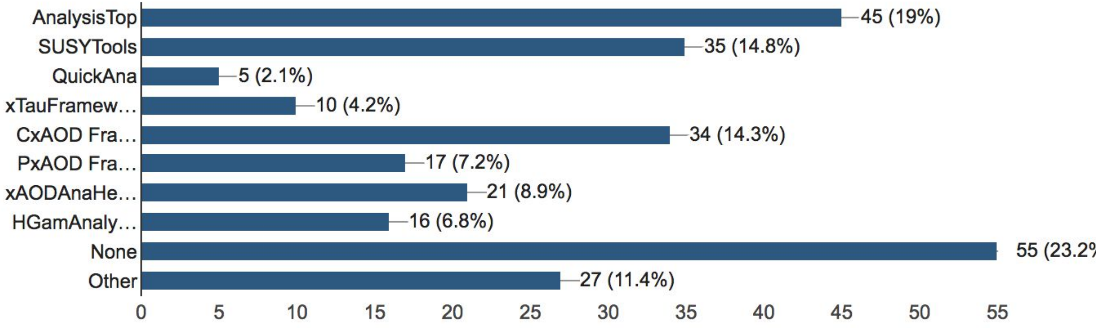
Analysis Software Provision

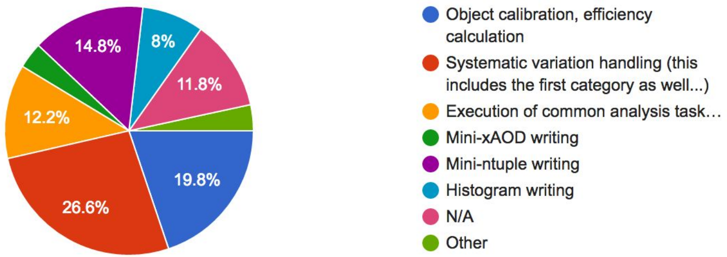


- ATLAS provides two base releases to support analysis
  - AthAnalysisBase - a stripped down build of Gaudi+Athena framework, containing only the packages needed by analysts
  - AnalysisBase - a ROOT based analysis toolkit, with a much smaller footprint
    - Perceived lower barrier to entry; does lack some features
- Of the two, AnalysisBase is by far the most popular
- Both releases can read the xAOD format (big improvement from Run 1)

Analysis Software Frameworks

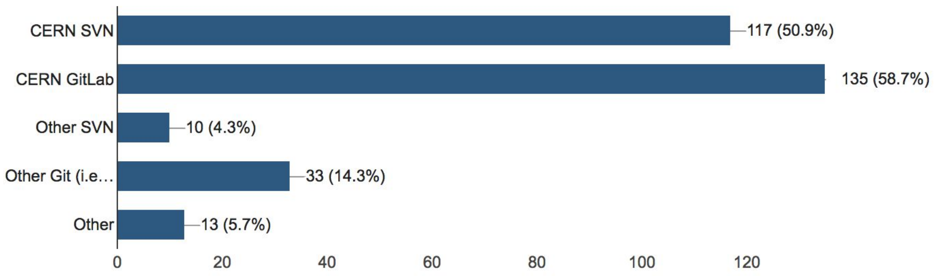
- Many analysis groups built on the AnalysisBase software
  - Adding extra layers to apply the correct calibrations, deal with systematics and luminosity, etc.
- Over time these additions were generalised to more analyses and became more generic
- There are now a lot of these and their use is widespread
- Management frequently ask if these could be rationalised
  - Despite the fact that effort is being duplicated, these are very successful and well established frameworks - the effort to rationalise them would be substantial

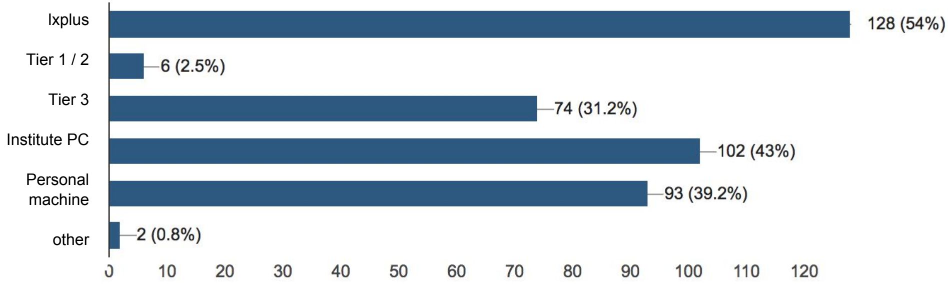




Code Management and Development

- Git is the future, SVN is the past
- Most groups do use CERN hosted services
    - SSO integration is probably the ‘killer feature’
- A surprising amount of development is still lxplus based
    - Everyone’s reference platform



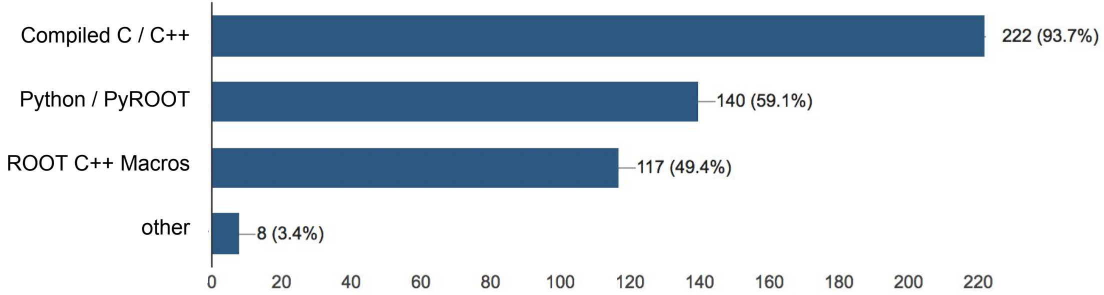


Software Validation

- ATLAS software is well validated up to the xAOD level
  - This is physics validation, comparing new outputs with reference samples
  - Centrally organized effort in CP and detector SW groups
    - Every small or major change in reconstruction or simulation SW is validated with hundreds of histogram comparisons in each sub-domain
    - Happens almost on a weekly or bi-weekly basis
  - As the xAOD content is largely what is in the DxAODs then these are implicitly validated
- DxAOD production is constantly monitored for performance also
  - E.g., sudden degradation in CPU time would be noticed
- Analysis group validation is performed on a more ad-hoc basis
  - Internal validations of recommendations in the analysis of recent data/MC performance
  - Usually driven by major SW releases or conferences

Analysis Data Access and Tools

- ATLAS analysis is using C++ and Python, very little else
- No widespread use of non-HEP tools
  - However, there is some trend towards looking at newer machine learning techniques (i.e., beyond what is in TMVA or other traditional HEP machine learning)
  - Such tools are usually used to prepare something that goes back into ROOT, e.g., a trained network
  - Here there is some back and forth format translation: xAOD→ROOT Ntuple→JSON, txt or CSV

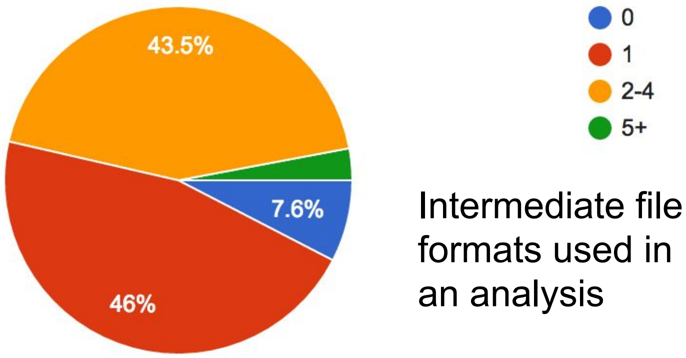


Conditions and Calibration

- Lower level conditions needs the full Athena framework to be accessed
    - Just because the tools that interpret the database entries only exist in Athena and rely on quite fat libraries, like COOL
    - ATLAS are considering simplifying this for Run 3 (as a common project)
- Analysis calibration files are mostly plain ROOT Ntuples
    - Accessing this data is mainly done via tools written by the CP groups themselves
    - There is special support for these tools to ensure they can be compiled into an Athena release and into a ROOT release
    - Insofar as they are usually quite simple data structures, reading them on your own is possible
        - But highly discouraged

Analysis Bifurcation Points

- Almost all analysis groups pass through at least one additional format between DxAOD and final plot data
  - “Flat” ntuples are still quite widely used in later stages
- Almost all analyses use one of the common framework layers previously shown
  - These are common
- Beyond that, code usually becomes more specialised



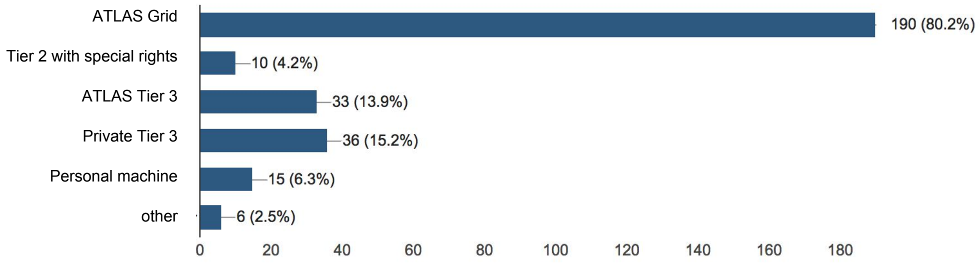
Intermediate file formats used in an analysis

Grid Use

- Almost everyone starts on the grid, or at least in some substantial private cluster
  - That's just the reality of LHC datasets!
  - Metadata: ATLAS Metadata Interface (AMI) used for discovery of appropriate data
    - Container = Many Datasets
    - Dataset = many files
  - ATLAS tools for good run list and luminosity built into the system
- Really substantial amounts of data are downloaded as well

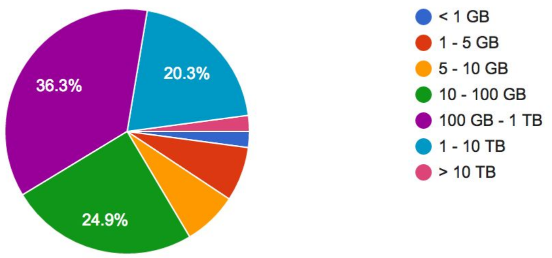
How do you run your "largest" jobs, the ones that you run as the first step of your analysis, on the largest datasets?

237 responses



What amount of data do you download to local/private resources during your analysis?

237 responses



For the Future - worries and mitigations

- Dramatic increases in LHC luminosity are not going to be matched by resources
- For analysis the time to complete one run over increasingly large datasets is only going up
  - N.B. here we suffer more than other workflows as the IO rate per TB is decreasing (larger disks, no increase in read rate)
- Analysis could lose a lot of flexibility, less scope to try something different
- Less copies of data available mean that remote reading of data will only grow
  - Work to do that efficiently
- Can we provide concurrency plugins that support analysis on modern CPUs easily?
  - We can’t expect analysis to become experts in parallel programing