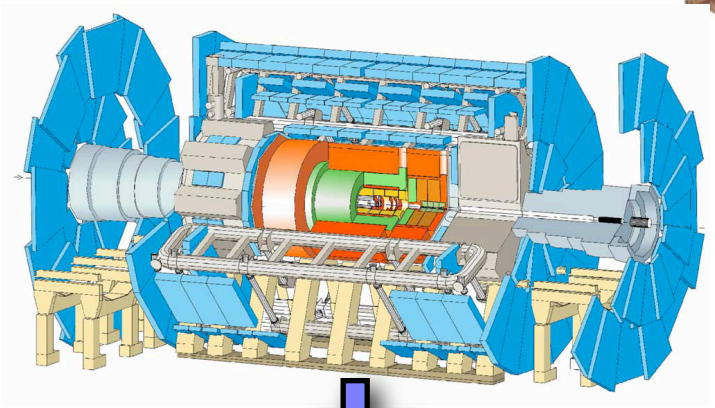
ATLAS Trigger Event Data Model & EF Tracking

A. Hamilton, F. Bucci

Andrew Hamilton
Université de Genève

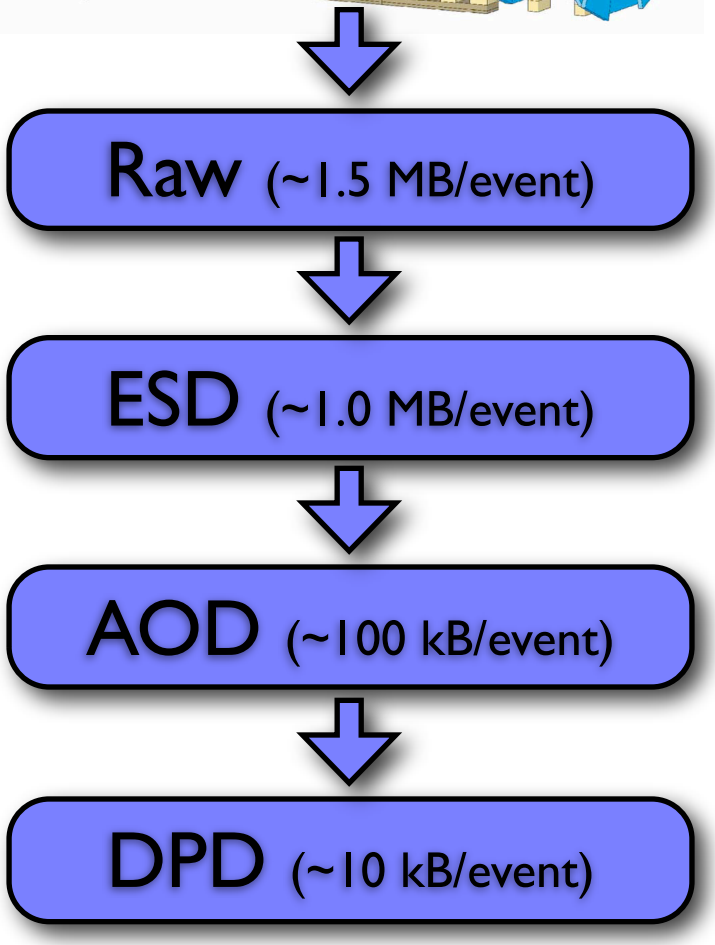
ATLAS Event Data Model

The ATLAS Event Data Model (EDM) defines the content and format of the ATLAS data.

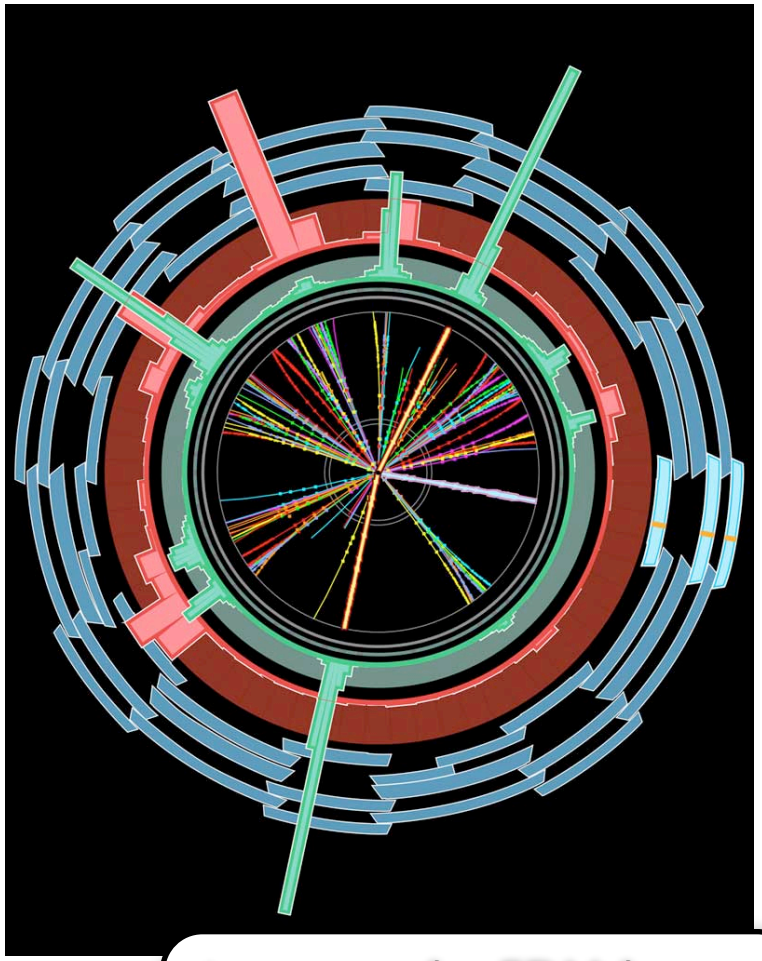


There are 4 levels of detail in the ATLAS EDM:

- **Raw Data:** saved directly from detector
- **ESD:** Raw data is processed (ie. calibrated) and put into "Event Summary Data" format
- **AOD:** ESD format is reduced to the "Analysis Object Data" format
- **DPD:** AOD is further reduced to the "Derived Physics Data" format



The Trigger EDM



important for EDM format

The Trigger Event Data Model (EDM) defines what trigger information is saved in each event.

Trigger Menu:

- defines all the trigger chains in an event

Feature Extraction (FEX) Algorithms:

- constructs 'objects' like electrons, jets, ...

Hypothesis (HYPO) Algorithms:

- applies selection criteria to results of FEX

Linking Information:

- an electron object must be linked to the track and cluster objects that seeded it

The trigger is divided into 9 'slices' (Electron, Muon, Jet, Cosmic, etc), each slice has it's own FEX and HYPO algorithms and it's own sub-menu

Issues in Trigger EDM

The Trigger EDM must consider:

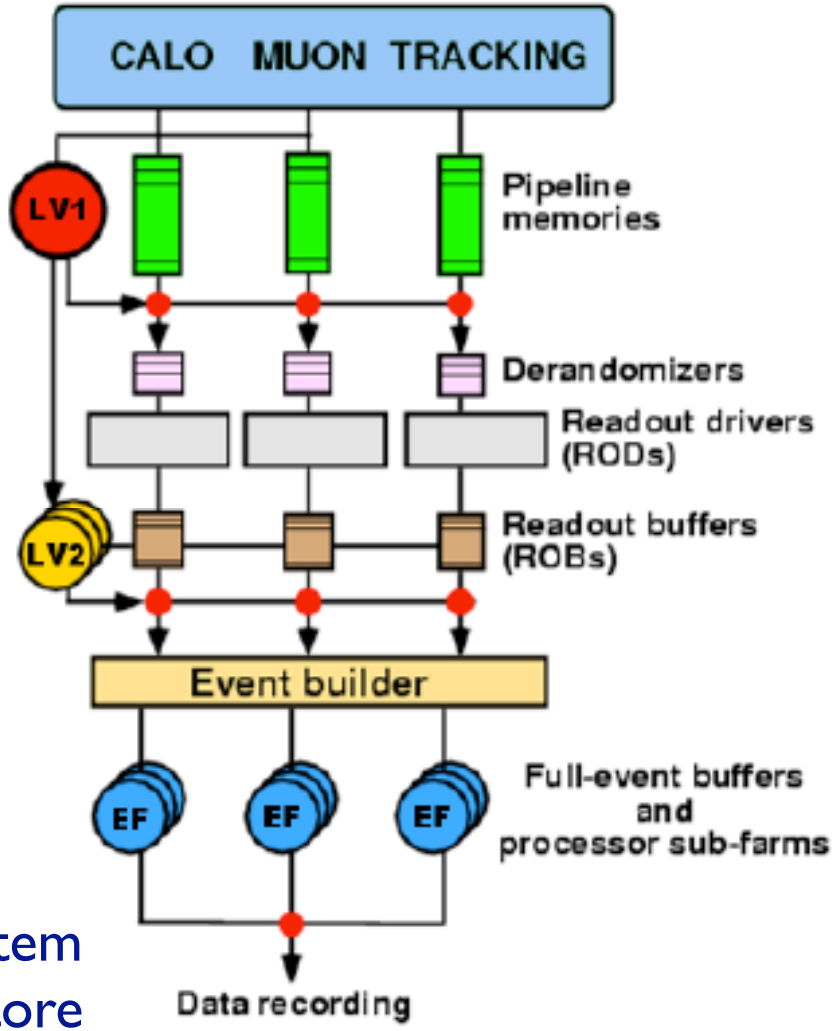
- the 9 trigger slices (Electron, Jet, ...)
- the 4 event formats (ESD, AOD, ...)
- the 3 trigger levels (L1, L2, EF)

Issues include:

- deciding which details are necessary
- size of the event on disk
- read/write speed of the data
- logical structure of event format
- backwards compatibility

Choices will determine:

- how easily we can debug the trigger system
- how many events we can process and store
- how we measure a trigger's efficiency

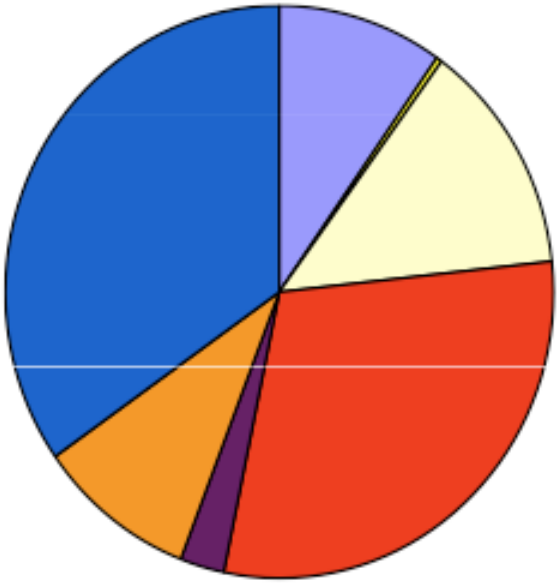


A Recent Issue :: Size on Disk

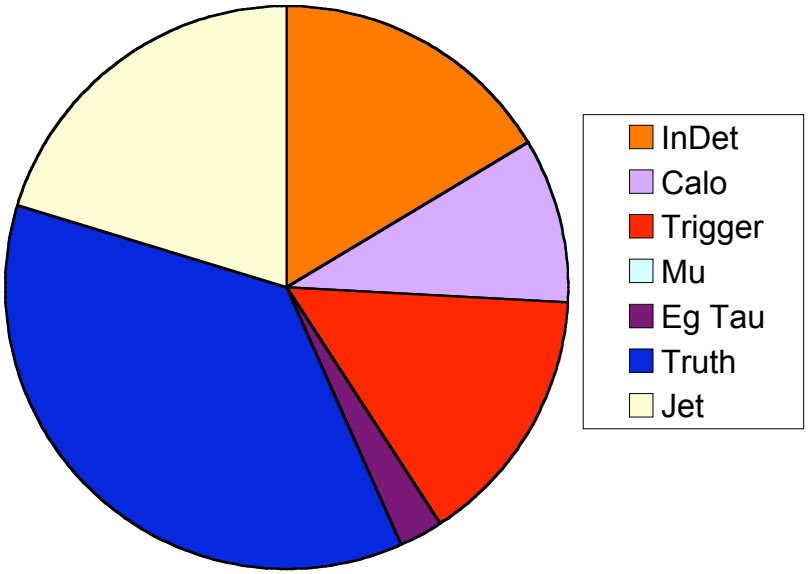
Six months ago, the AOD trigger EDM required ~115kB per event

- it occupied more than the projected size of the entire AOD (100 kB/evt)

12.0.6 EDM size breakdown



13.0.2 EDM size breakdown



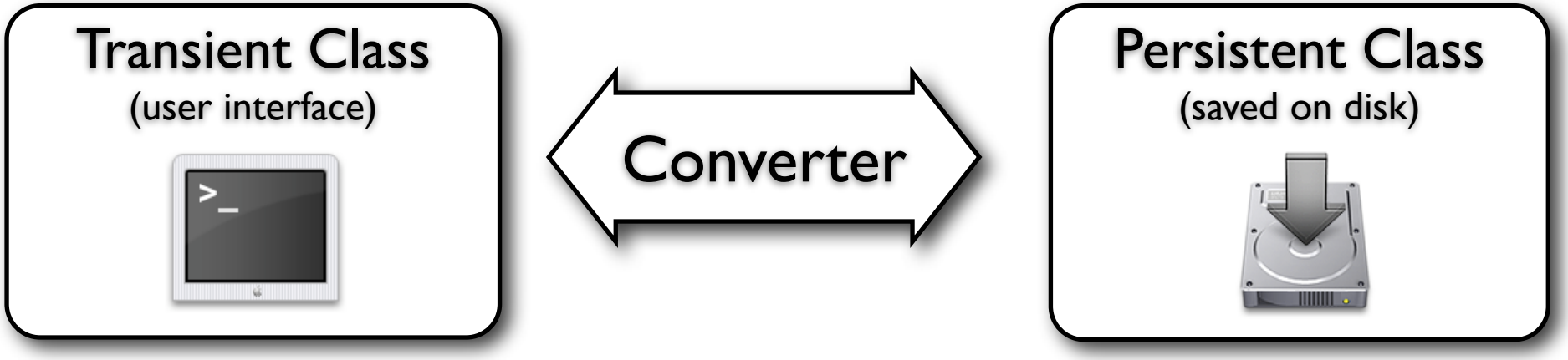
- reduction achieved by reducing detail and restructuring event format
- size is now ~50kB per event (acceptable)

Current Issue :: Backwards Compatibility

ATLAS software evolution will continue well into data taking

- with no backwards compatibility 'old' data will either need to re-processed or analyzed in blocks based on the software release

To ensure backwards compatibility, we separate the transient and persistent data objects by writing converters for each class:



This Transient-Persistent (TP) separation gives us explicit control over what is saved on disk and how that is interpreted by the user interface.

TP Converters in Trigger EDM

Most converters for the trigger classes are complete:

- TrigDecision
- HLTResult
- TrigRoiDescriptor
- TrigInDetTrackCollection
- TrigInDetTrack
- TrigInDetTrackFitPar
- MuonFeature
- CombinedMuonFeature
- TrigMuonEFContainer
- TrigEMCluster
- TrigTauCluster
- TrigCaloCluster
- TrigMissingET
- TrigElectronContainer
- TrigElectron
- TrigPhotonContainer
- TrigPhoton
- TrigTau
- TrigL2Bphys
- TrigL2BphysContainer
- TrigEFBphysContainer
- TrigVertex
- TrigVertexCollection
- TrigL2BjetContainer
- TrigEFBjetContainer
- TrigT2Jet

complete
incomplete

...completing them is our current focus.

Event Filter Tracking

Next project is Event Filter Tracking:

Milestones for release 14 (timescale 6-9 months)

Priority is to ensure that the code is ready for commisioning and first running. So must target online integration issues including robustness, ensuring no initialisation in event processing, ensuring final cabling maps are in place, etc.

- LVL2 stand-alone TRT reconstruction (TrigTRTSegFinder) working in slices
- IDScan and SiTrack optimisations available for all RoI types
- All algorithms migrated to new RegionSelector interface & using RoiDescriptor for RoI size and shape information.
- Minbias slice implemented
- FTK emulator implemented
- RTT tests implimented for all slices from RDO and from BS including tests for timing and memory leaks.
- Code robust against data errors - missing/truncates RoB info, data corruption etc.
- Possible to use Offline vertexing tools at LVL2
- Transient/persistent split for LVL2 classes → almost done
- Common Truth association LVL2 & EF
- back tracking usable in EF
- migrated iPatRec usable in seeded mode in EF
- Algorithms meet targets for memory leaks, robustness (crashes & errors) and timing
- Algorithms meet requirements for online integration

.... lots of work still to do!

Backup Slides

Breakdown of the Trigger EDM (12.0.6)

Top 10 offenders (~90% of size):

	No./Evt	Disk Size / Evt (kB)	Class
EF: 1 per EM/HA/MU/JT20 RoI	12.2	25.1	Rec::TrackParticleContainer
L2: 3 track coll'ns per EM + 1 per HA+MU RoI + 2 per JT20	22.6	22.4	TrigInDetTrackCollection
	1	18.0	EFResult
EF: 1 per EM/HA/MU/JT20 RoI	12.2	14.6	VxContainer
L2: ~5 HA RoIs per event; includes copied tracks & cluster	4.7	6.4	TrigTau
	1	5.8	L2Result
L2: 3.6 RoIs x 0-4 thresholds; includes copied track & cluster	8.2	4.1	TrigElectron
	1	3.8	DataHeader
EF: ~7 Jet RoIs per event + kt for JT20	9.0	2.7	JetCollection
	1	2.3	TrigInDetTrackTruthMap

100 top events: mc11004100.T1_McAtNLO_top.digit.RDO.v11000401._00001

Total size ~116 kB/event in trigger EDM (tops are larger than "typical" events)