Analysis software in the wider HEP/nuclear community :

@ LHCb

N. Skidmore on behalf of the LHCb collaboration

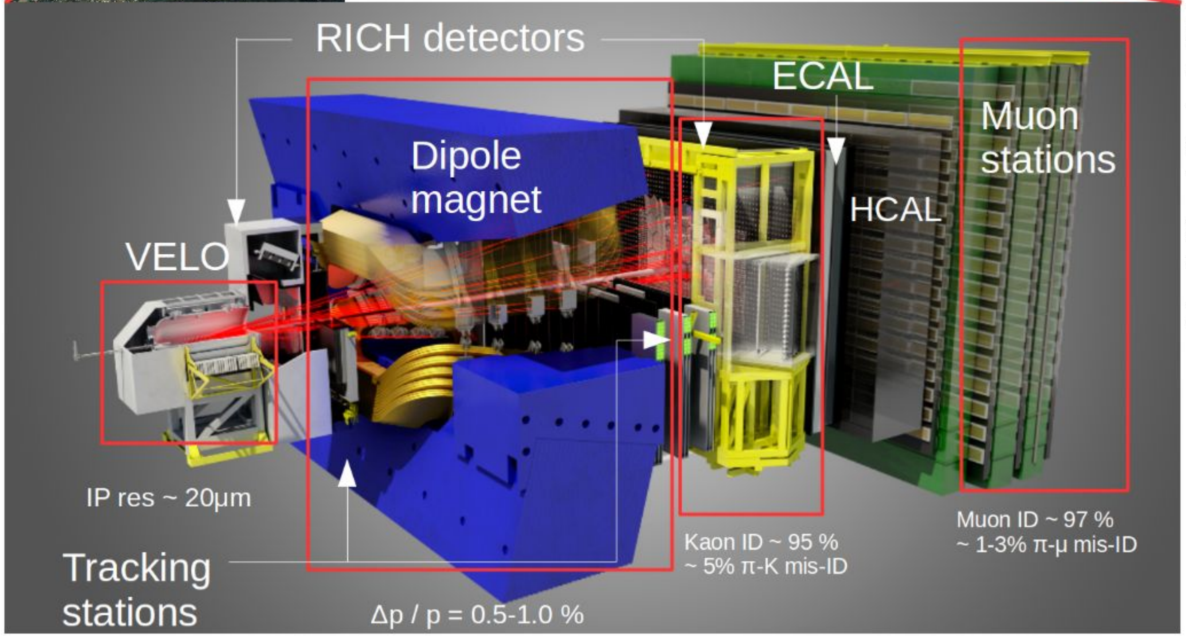
BEAUTY2CHARM
European Research Council
Established by the European Commission

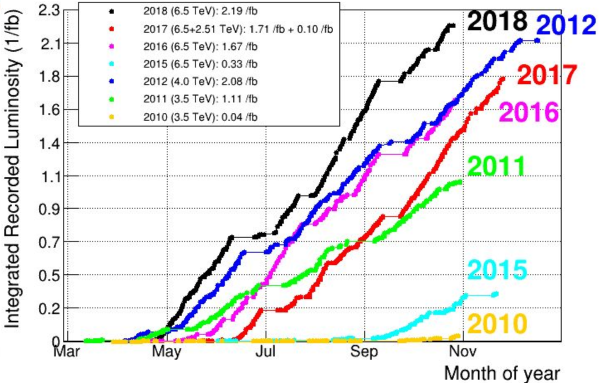
June 2021

LHCb Run 1+2

LHCb experiment

LHCb is a forward arm spectrometer specialising in decays of beauty and charm hadrons





9 fb^{-1} collected in run 1 + 2
$>10^{12}$ $b\bar{b}$ pairs in acceptance

LHCb persistency model - Turbo vs. Full

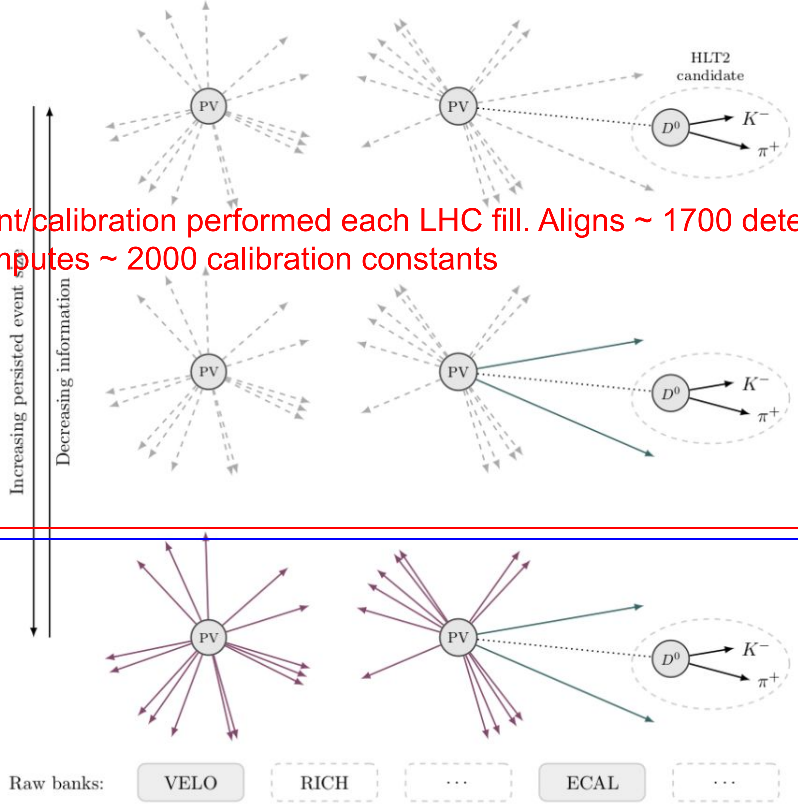
LHCb Run 1+2

Turbo

(2018 event size)

- 7 kB
- 16 kB

Real-time alignment/calibration performed each LHC fill. Aligns ~ 1700 detector components & computes ~ 2000 calibration constants



Turbo - save only **reconstructed** objects involved in the trigger decay candidate

Turbo Selective Persistence (SP) - save **additional** reconstructed objects from event such as other tracks from PV

No offline recon./slimming required

Full

- 48 kB
- 69 kB

Full - **whole raw event** is saved. Re-reconstruction performed **offline**

Offline recon./slimming required

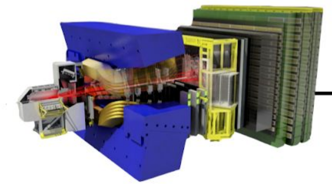
LHCb Run 1+2 data flow

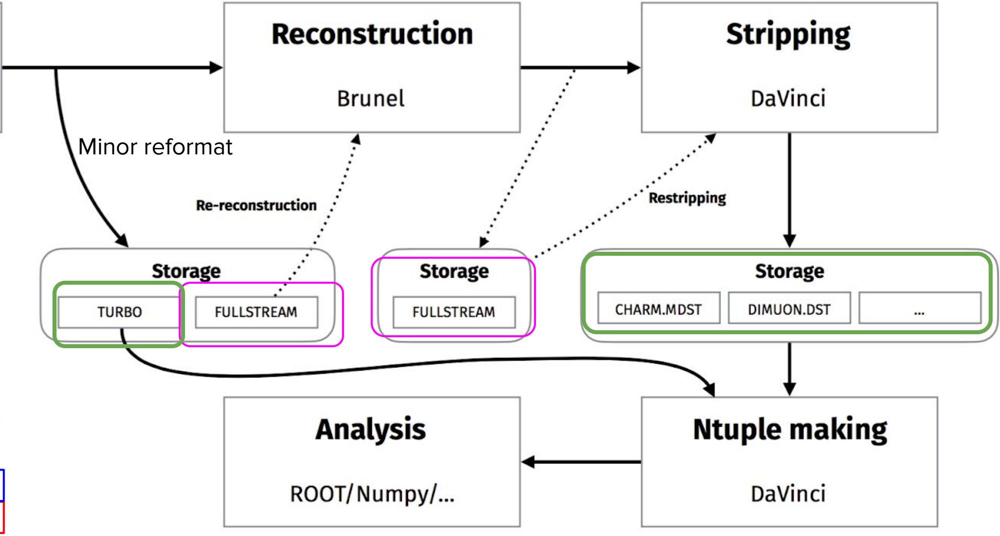
LHCb Run 1+2

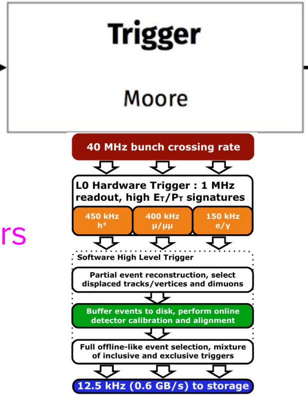
In Run 1+2 most physics uses full persistency model

- Offline re-reconstruction
- Skimming/slimming required - stripping
- Happens concurrently with datataking

In run 2 Turbo (+SP) model adopted for some physics







Tape storage - not accessible to users

Disk storage - available to users

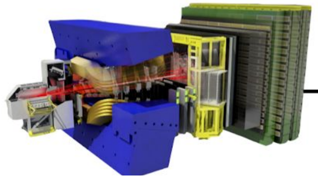
stream	event size (kB)	event rate (kHz)	rate fraction	throughput (GB/s)	bandwidth fraction
FULL	70	7.0	65%	0.49	75%
Turbo	35	3.1	29%	0.11	17%
TurCal	85	0.6	6%	0.05	8%
total	61	10.8	100%	0.65	100%

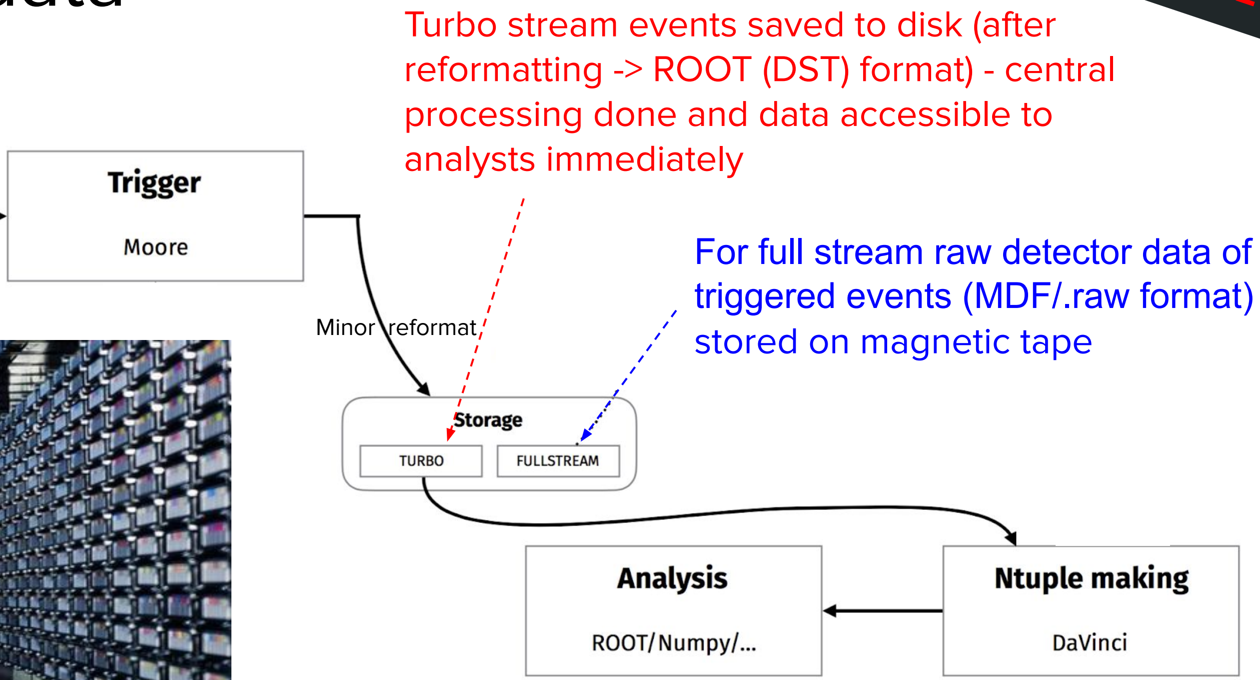
Online

Offline

LHCb Run 1+2

LHCb Run 1+2 data flow





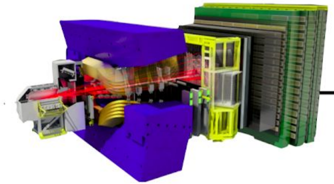
Automated magnetic tape vault at CERN computer center, 2008.

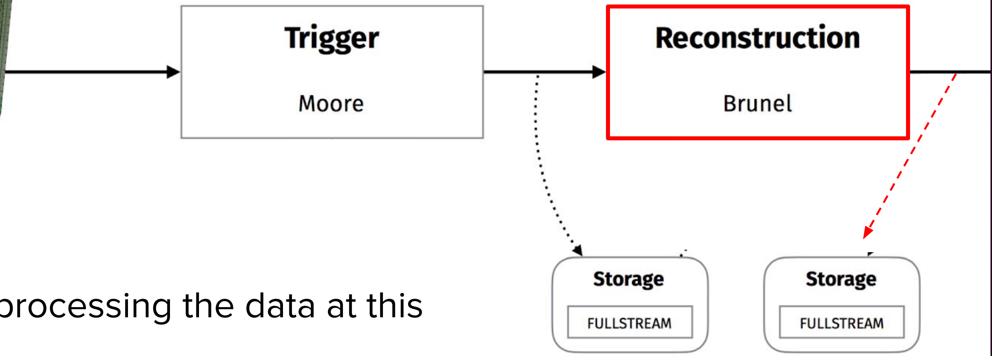
Why store to tape?

- Tape storage is cheap compared to disk
- No electricity consumption when tapes are not being accessed
- For data that does not need to be available immediately

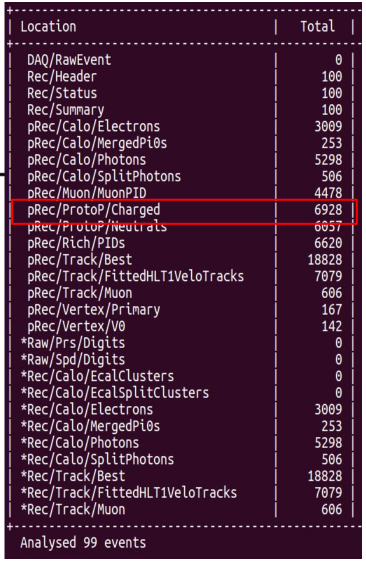
LHCb Run 1+2

LHCb full stream - re-reconstruction





Reconstructed events (.rdst)
Stored on magnetic tape



Location	Total
DAQ/RawEvent	0
Rec/Header	100
Rec/Status	100
Rec/Summary	100
pRec/Calo/Electrons	3009
pRec/Calo/MergedPi0s	253
pRec/Calo/Photons	5298
pRec/Calo/SplitPhotons	506
pRec/Muon/MuonPID	4478
pRec/ProtoP/Charged	6928
pRec/ProtoP/Neutrals	6057
pRec/Rich/PIDs	6620
pRec/Track/Best	18828
pRec/Track/FittedHLT1VeloTracks	7079
pRec/Track/Muon	606
pRec/Vertex/Primary	167
pRec/Vertex/V0	142
*Raw/Prs/Digits	0
*Raw/Spd/Digits	0
*Rec/Calo/EcalClusters	0
*Rec/Calo/EcalSplitClusters	0
*Rec/Calo/Electrons	3009
*Rec/Calo/MergedPi0s	253
*Rec/Calo/Photons	5298
*Rec/Calo/SplitPhotons	506
*Rec/Track/Best	18828
*Rec/Track/FittedHLT1VeloTracks	7079
*Rec/Track/Muon	606

Analysed 99 events

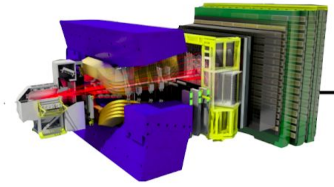
Why can't analysts start processing the data at this point?

- This is a LOT of data (events are whole) - must be saved to tape
- Even if we were to periodically stage this data to disk all analysts would be trying to individually access and run over ALL the triggered data!

Run 1 + 2 RAW+RECO > 20PB!

LHCb full stream - particle building/selection

LHCb Run 1+2



Trigger	Reconstruction	Stripping
Moore	Brunel	DaVinci

The Stripping (skimming/slimming application) runs sets of selection criteria (lines) over the reconstructed physics objects.
The first step is building the particles from the protoparticles...

Location	Total
DAQ/RawEvent	0
Rec/Header	100
Rec/Status	100
Rec/Summary	100
pRec/Calo/Electrons	3009
pRec/Calo/MergedPi0s	253
pRec/Calo/Photons	5298
pRec/Calo/SplitPhotons	506
pRec/Muon/MuonPID	4478
pRec/ProtoP/Charged	6928
pRec/ProtoP/Neutrals	6057
pRec/Rich/PIDs	6620
pRec/Track/Best	18828
pRec/Track/FittedHLT1VeloTracks	7079
pRec/Track/Muon	606
pRec/Vertex/Primary	167
pRec/Vertex/V0	142
*Raw/Prs/Digits	0
*Raw/Spd/Digits	0
*Rec/Calo/EcalClusters	0
*Rec/Calo/EcalSplitClusters	0
*Rec/Calo/Electrons	3009
*Rec/Calo/MergedPi0s	253
*Rec/Calo/Photons	5298
*Rec/Calo/SplitPhotons	506
*Rec/Track/Best	18828
*Rec/Track/FittedHLT1VeloTracks	7079
*Rec/Track/Muon	606

Analysed 99 events

→ NoPIDsParticleMaker.py
↓
StdAllNoPIDsPions.py
↓
Phys/StdAllNoPIDsPions/Particles

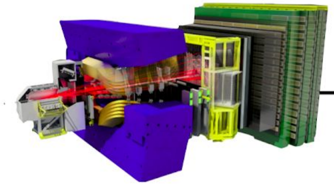
Each stripping line then builds these (composite)-particles into decay chains and runs loose selection criteria.

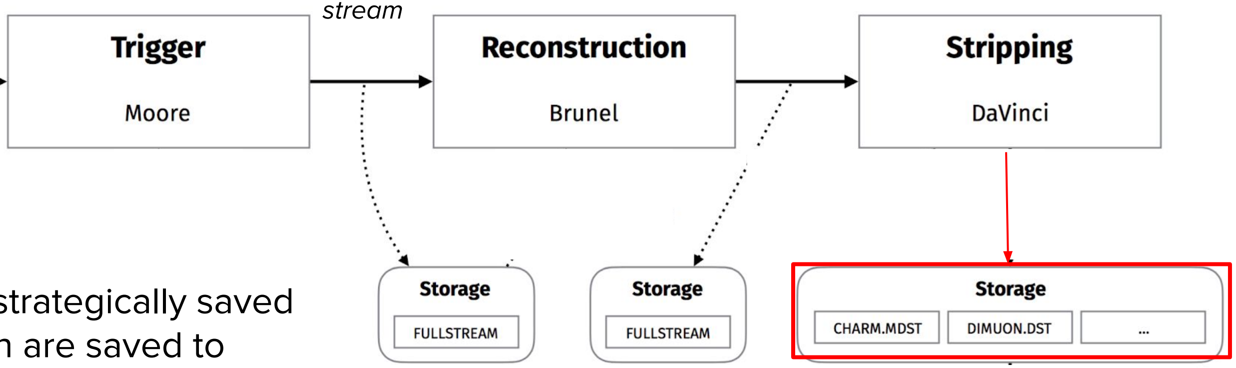
StrippingInclusiveCharmBaryons_Xic0Line
DecayDescriptor: [Xi_c0 -> pi+ K- K- p+]cc

LHCb Run 1+2

Full stream - Streaming

DST - Save full reconstructed event for events that pass any line in the stream
MDST - Only save the signal candidates for events that pass any line in the stream





Stripped events (.(m)DST)
Stored to disk - available to analysts

The events selected by each line are strategically saved to different output files (streams) which are saved to disk

Streaming is optimised considering:

- Do not want same event saved in many streams (reduce streams)
- Efficient access to your data (increase streams)

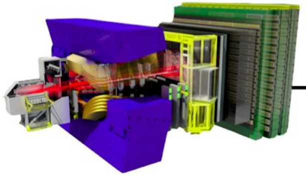
Send lines with common physics to the same stream to maximise event overlap

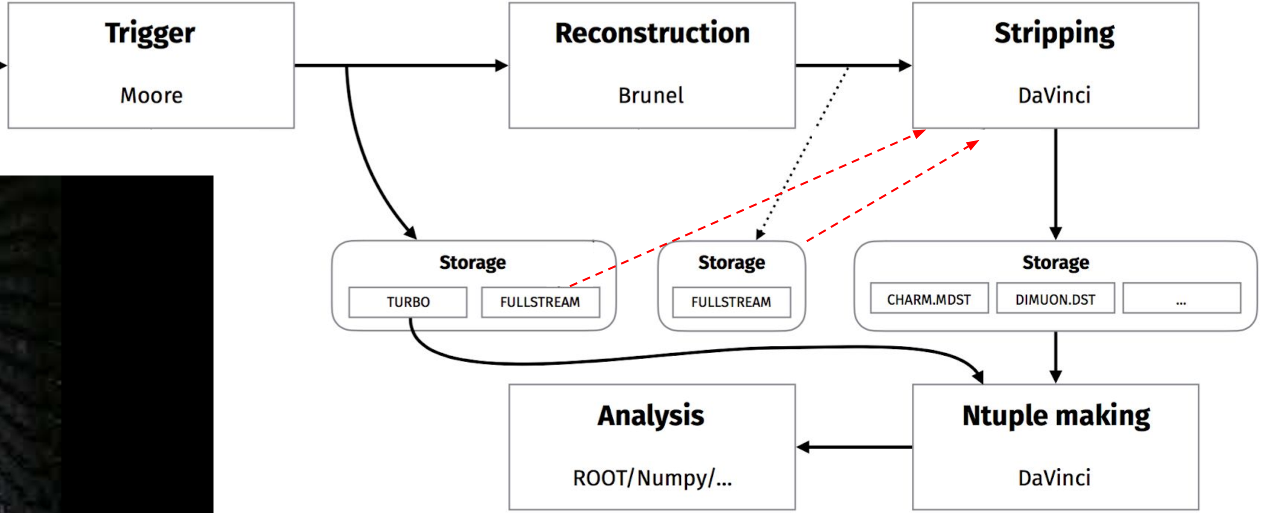
BhadronCompleteEvent.dst	Bhadron.mdst
CharmCompleteEvent.dst	Charm.mdst
Dimuon.dst	EW.dst
Semileptonic.dst	Leptonic.dst

2016 BhadronCompleteEvent stream ~ 100TB

LHCb Run 1+2

Re-stripping campaigns





Click on me :)

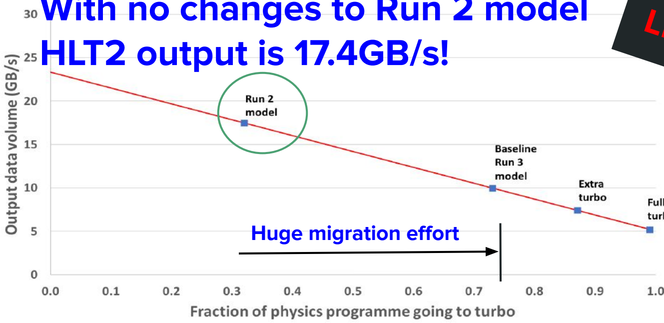
In End of Year Shutdowns we re-strip the data

- Stage raw/recon. data from tape
- Can rerun reconstruction if improved calibration available
- Add new lines/bug fixes

LHCb Run 3 data flow

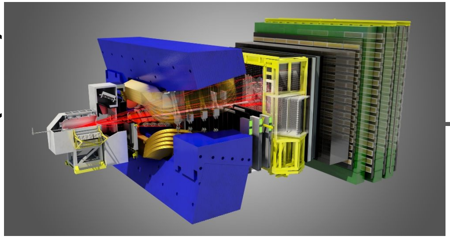
Move bulk of selections Online = More Turbo!

With no changes to Run 2 model HLT2 output is 17.4GB/s!



LHCb Upgrade

Default model Turbo (+SP)



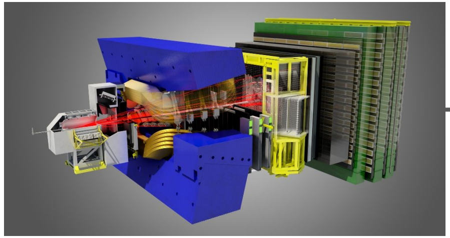
HLT2 Moore → Minor reformat → **Data analysis** DaVinci

Default model - >70% of physics

To tape

To disk

Sprucing model FULL and TURCAL



HLT2 Moore → **Sprucing** Moore → **Data analysis** DaVinci

Use cases - topological, inclusive triggers, datamining

A further offline stage of data reduction/selection between tape and disk storage when HLT2 line throughput is too large to go straight to disk. **Utilise same selection framework as HLT2**

Moore
Project ID: 465

Online

Offline

Analyst data tuples

DaVinci

Project ID: 544

In Run 1 + 2 (all) analysts create **nTuples individually** from (m)DST data using DaVinci

DaVinci provides **TupleTools** - tools for the creation and saving of variable branches for typical use cases eg. TupleToolTrackInfo

- Very **easy to implement** but adds lots of **redundant branches** - can easily save 500+ variables
- 500GB - 10TB of data for a single Run 1+2 analysis - nTuples tend to be only used for one analysis
- **Redesign** of tools for Run 3 such that this redundancy is minimised

Jobs submitted to the grid typically using Ganga

- Time consuming - O(weeks) for Run 1 + 2 tuples - failed jobs re-submitted manually by user
- **No analysis preservation** infrastructure
- In run 3 submit jobs centrally using **DIRAC transformation System** (Analysis Productions)
    - Does not require analyst to “babysit” jobs
    - Jobs **tested automatically** with GitLab CI
    - Job details/configuration/logs **automatically preserved** in bookkeeping/EOS

LHCb analysis tools

LHCb as a collaboration exploits a wide range of tools

- Gaudi and ROOT are central pieces of the software stack
- Run 3 calibration packages (PIDCalib) use Data Science tools (Scikit-HEP, SciPy)
- Stack development increasingly involves only python interaction - all underlying C++ has python wrappers. All application configuration and running is in python
- Extensive use of conda in LHCb environment

LHCb analyst tools

LHCb has a wide range of analyses with different offline requirements:

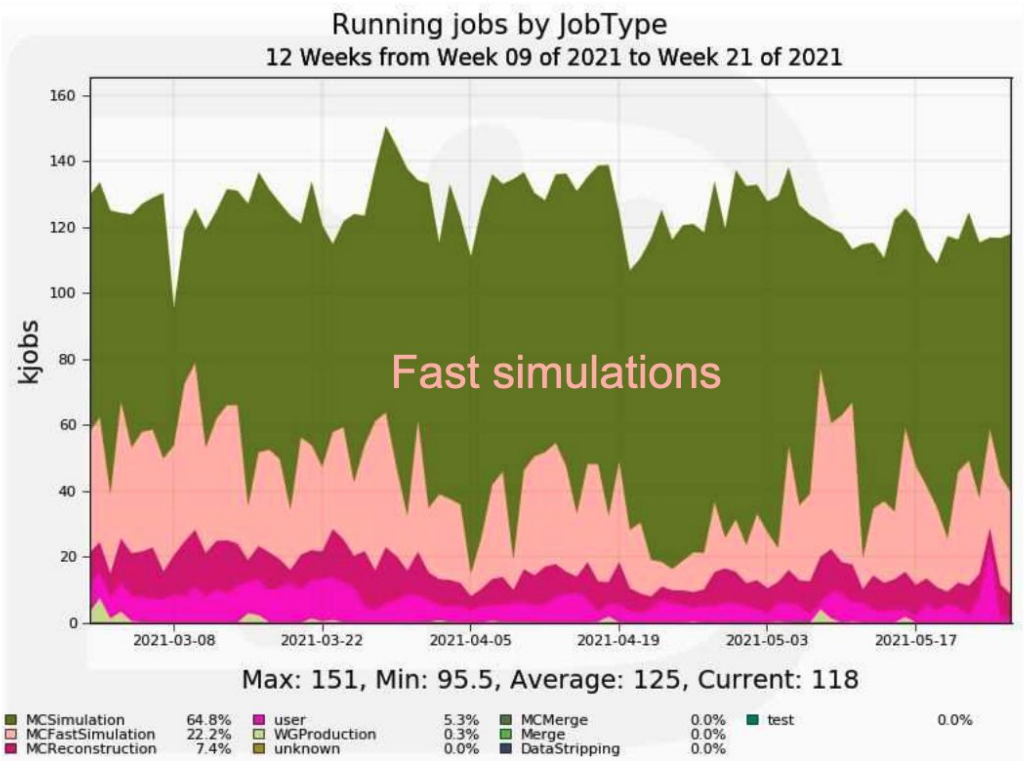
[18kB/evt : 83kB/evt] on disk for [CEP : B to open charm] event in Run 2

- Every analysis chain looks different ie. no “accepted way” to apply selections/fit data (ROOT/python/pandas...)
  - Analysts use C++ /Python/ ROOT/ uproot/ numpy/ pandas/... (50% did analyses in python in 2018 survey)
  - **Many** custom fitting packages written and used by individual institutes. Some adoption of GPU: GooFit, TensorFlowAnalysis, zfit etc. particularly for complex amplitude analyses
- Approx. all analyses use GitLab for **version control of analysis code** with *some* use of the CI
- Significant number of analyses use Snakemake for **automating analysis workflows**. Else use bash scripts
- Basic **analysis preservation** already enforced at LHCb

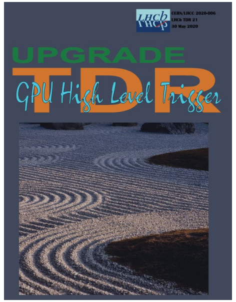
In particular, the analysis code should be preserved in a long-term archive such as a physics analysis gitlab group, the input ntuples should also be preserved in a long-term archive and sufficient documentation to enable a (technically competent) LHCb member to run the code in a standard environment such as lxplus should be included with the code.

- Making template workflows, analysis examples etc to make analysis preservation more robust in Run 3

LHCb Resources



In Run 3 LHCb's **HLT1 will run on GPUs**



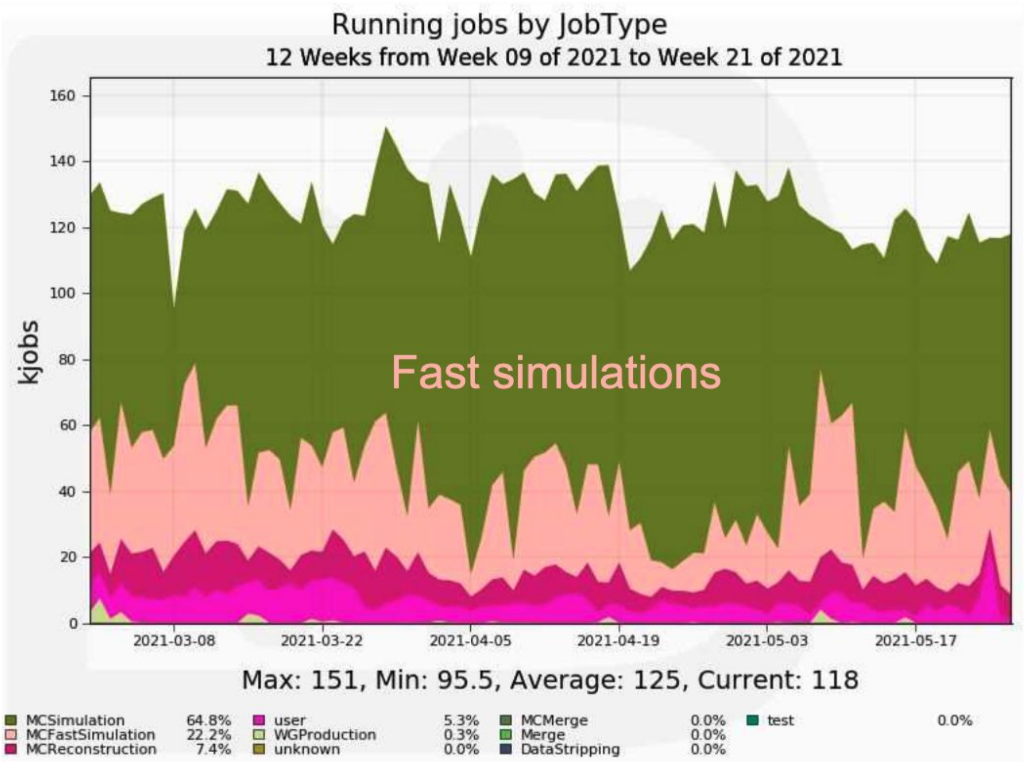
[See dedicated HSF talk]

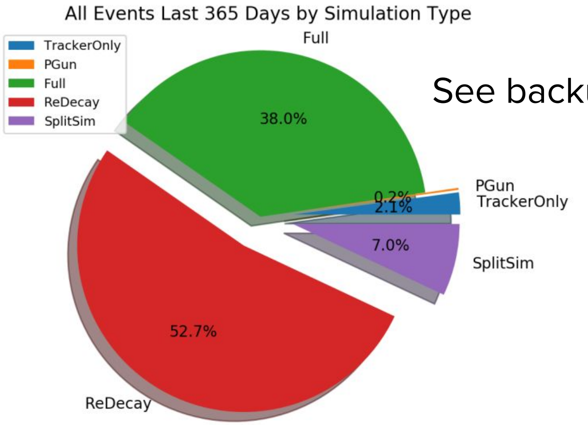
Potential to **utilise HLT1 GPU farm** during detector downtime

Need development such that **significant LHCb payloads can run on GPUs**

- User analysis utilising eg. TensorFlow for ML and fitting but small share of LHCb's CPU
- Full detector simulation main payload but not yet able to run on GPUs...

LHCb Resources





See backup

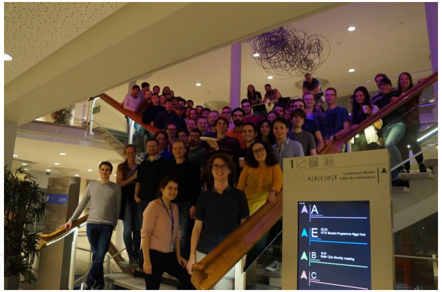
Simulation takes 95% of CPU resources

Already very successful adoption of fast simulation methods

Decrease in time required to simulate events crucial to fully exploit the larger datasets

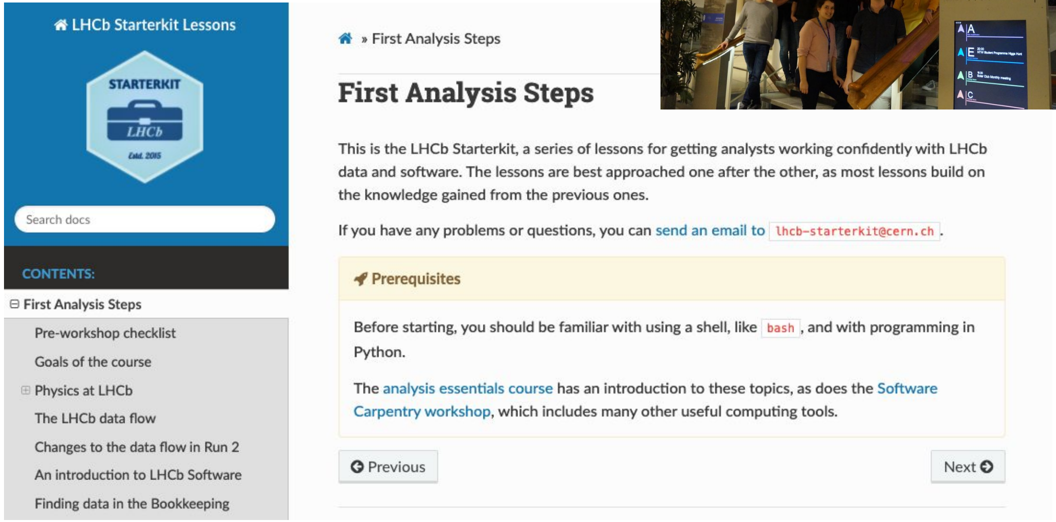
- Measurements hinting at SM tensions have systematics dominated by **limited MC statistics**
- These fast simulation options are crucial to exploit the run 3 dataset

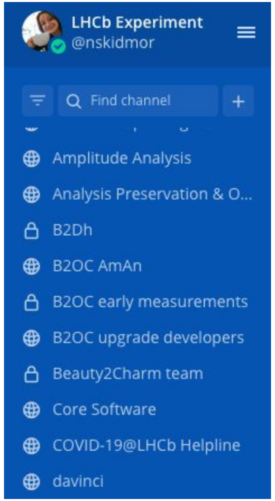
Training and getting help at LHCb



LHCb Starterkit aimed at newcomers

- Live event @ CERN
- Well maintained documentation used by whole collaboration





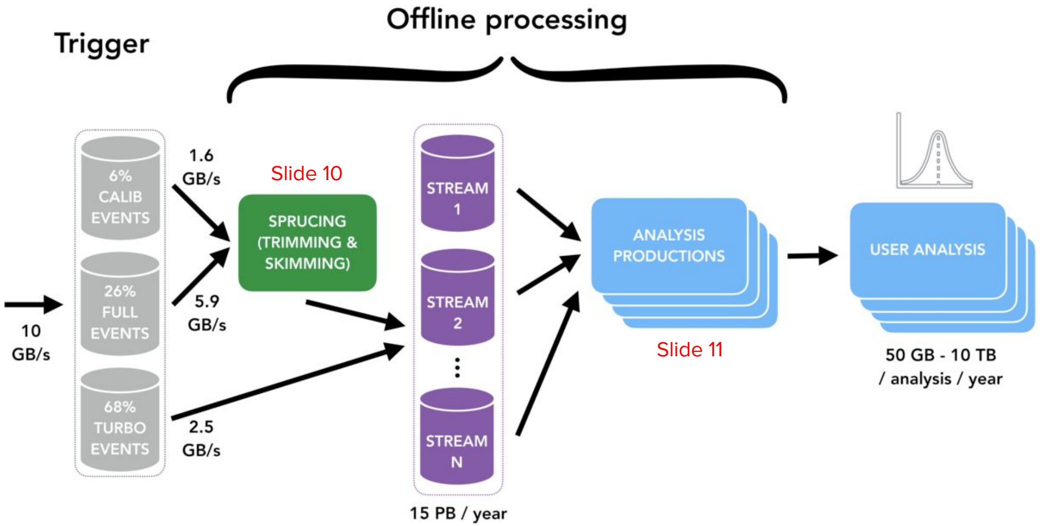
Day to day support mainly provided via dedicated MatterMost channels

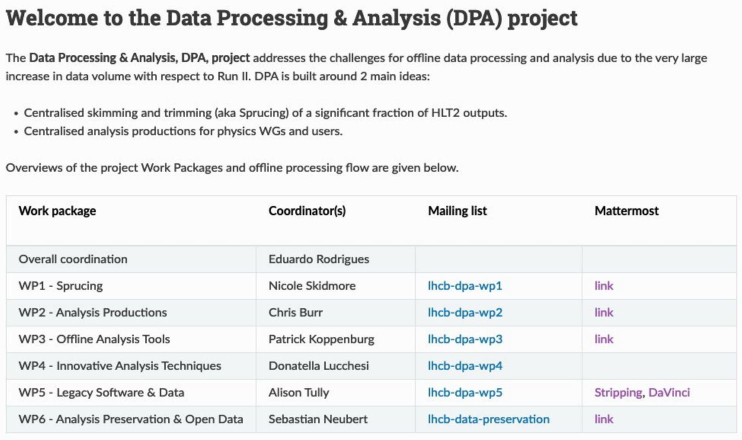
- Instant support
- No “nice” long term preservation of problem+solutions

Data Processing and Analysis (DPA) project

Run 3 offline data volume necessitates a **more coordinated approach** to offline data processing

- New DPA software project carries same status as sub-detector projects with institutional commitment





Welcome to the Data Processing & Analysis (DPA) project

The **Data Processing & Analysis, DPA, project** addresses the challenges for offline data processing and analysis due to the very large increase in data volume with respect to Run II. DPA is built around 2 main ideas:

- Centralised skimming and trimming (aka Sprucing) of a significant fraction of HLT2 outputs.
- Centralised analysis productions for physics WGs and users.

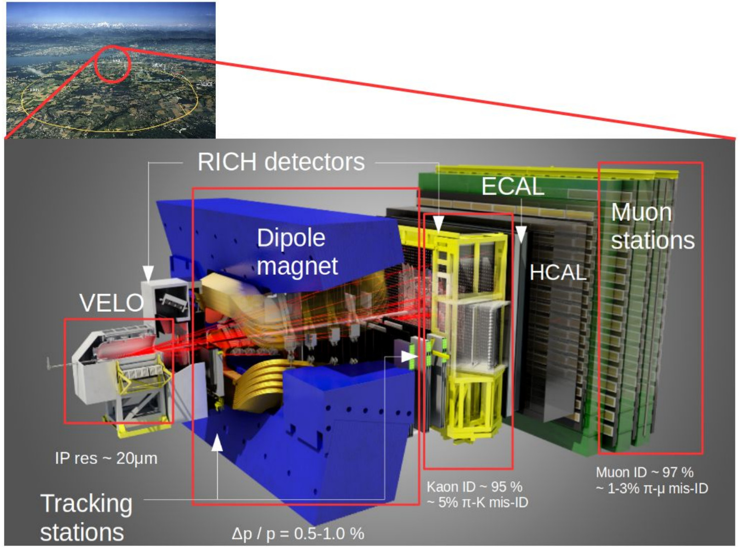
Overviews of the project Work Packages and offline processing flow are given below.

Work package	Coordinator(s)	Mailing list	Mattermost
Overall coordination	Eduardo Rodrigues		
WP1 - Sprucing	Nicole Skidmore	lhcb-dpa-wp1	link
WP2 - Analysis Productions	Chris Burr	lhcb-dpa-wp2	link
WP3 - Offline Analysis Tools	Patrick Koppenburg	lhcb-dpa-wp3	link
WP4 - Innovative Analysis Techniques	Donatella Lucchesi	lhcb-dpa-wp4	
WP5 - Legacy Software & Data	Alison Tully	lhcb-dpa-wp5	Stripping, DaVinci
WP6 - Analysis Preservation & Open Data	Sebastian Neubert	lhcb-data-preservation	link

https://lhcb-dpa.web.cern.ch/lhcb-dpa/

Summary

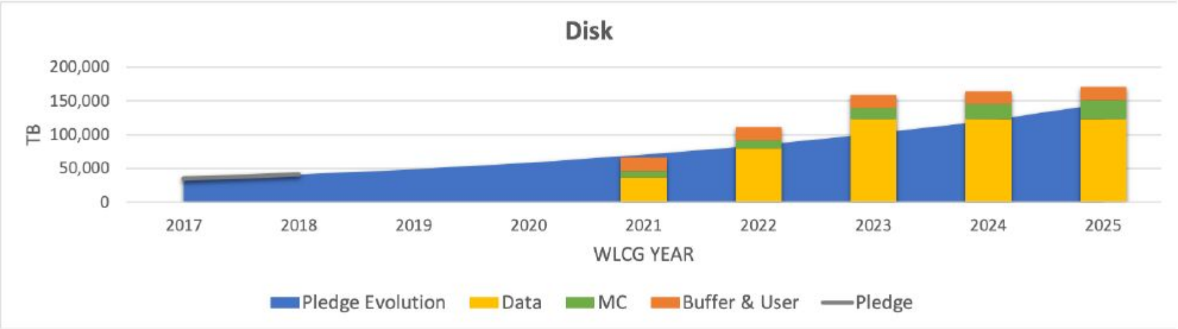
- LHCb has a very diverse range of analyses - no “one size fits all” analysis chain
    - As a result there are a lot of different workflows in the collaboration
- LHCb analysts are making good use of wider community software tools
- Big effort in use of GPUs for all stages of analysis
- Huge effort upgrading the offline data processing/handling to exploit higher data rate in Run 3



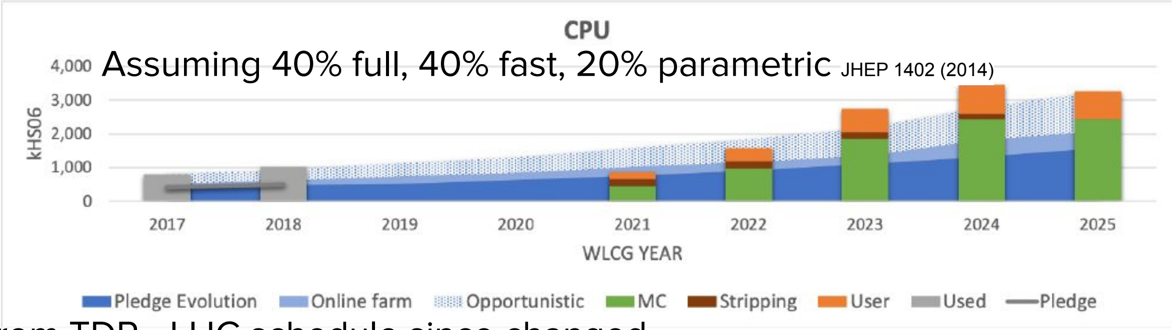
Backup

Resources at LHCb in Run 3

In Run 3 LHCb will produce ~ 15PB of user accessible data per year



Real data dominates disk storage



Assuming 40% full, 40% fast, 20% parametric JHEP 1402 (2014)

But simulation dominates CPU (90% of total CPU) - mitigation strategies using **fast simulation.**

*from TDR - LHC schedule since changed

Simulation

Successful adoption of fast simulation in Run 1 and 2

Full - full detector simulation

PGun - single signal particle spawned with kinematics configured to follow distribution (no full Pythia event) Factor 50 speed increase

ReDecay - re-use the underlying event but generate and simulate new signal decays every time Eur. Phys. J. C 78 (2018) 1009 Factor 10-20 speed increase

TrackerOnly simulation **-** Factor 10 speed increase

SplitSim - only simulate full event if required condition is passed eg. if a photon converts to $e^+ e^-$ Speed up depends on condition