Blurring Online and Offline

Peter Hristov

CHEP 2016

San Francisco

Introduction

- There are O(100) relevant presentations that appeared after CHEP 2015
  - I learned a lot
  - I am grateful to all the authors of original works or reviews
  - This 20 min. talk certainly cannot include all the interesting topics
  - I apologize if I misunderstood or misrepresented any subject
- The old days when Online and Offline were completely separated probably did not exist at all
  - Pre-LHC: Offline components have been used online since decades
  - LHC: Most of the LHC experiments use their Offline frameworks also online
- What are the trends (from my personal offline point of view subject to ALICE bias)?

Online *Systems* vs Offline *Processing*

Online Systems: working during data-taking run

- Front-End Electronics (FEE): collect, digitize and process the signals from the detector(s)
- Detector control system/slow control (DCS): control and monitoring of high voltages, currents, temperatures, flows, pressures, etc.
- Experiment Control System: based on state machines − Init, Start, Pause, Stop the run
- Trigger: fast selection of events based on specific detector signals
- High level trigger (HLT): selection of events based on fast reconstruction, compression,…
- Data acquisition (DAQ): data transport, event building, optional compression, data storage

Offline Processing: working independently of the run

- Alignment: (infrequent) procedure to define the shifts and rotations of detector elements wrt the nominal position
- Calibration: calculation of time-dependent parameters of detectors (gains, dead/noisy channels, etc.). Usual granularity − per run
- Reconstruction: pattern recognition of tracks, calorimeter clusters, calculation of physics quantities (momenta, energies, particle identification probabilities)
- Monte-Carlo simulation: generation, geometry and materials, particle transport, detector response, etc.
- Analysis: searches, measurements, etc.

Online vs Offline: traditional tasks

Online: DAQ

- Analog signal processing
- Digitization
- Digital signal processing
- Readout
- Event building
- Raw data storage on DAQ buffer
- Data transfer and registration to Tier0
- Quality assurance
- Raw data = header + payload

=> Ideally minimal or no processing of detector "payload"

Offline

- Replicate the raw data from Tier0 to Tier1s
- Run calibration algorithms and update the offline conditions DB
- Run reconstruction, register and replicate Event Summary Data (ESD)
- Filter ESD to produce Analysis Objects Data (AOD), ntuples, specific samples (skimming), etc. and register the results
- Quality assurance

Online vs Offline: traditional tasks

Online: HLT

- Run fast reconstruction algorithms using approximate calibration
  - Good efficiency
  - Relatively high fake rate
  - Relatively bad resolution
- Run fast selection of interesting events
  - "Loose" selection criteria
- Quality assurance

Offline

- Run full reconstruction using precise calibration
  - Good efficiency
  - Low fake rate
  - Good resolution
- Obtain "physics quality" results
- Quality assurance

Online vs Offline: traditional requirements

Online

- Reliable algorithms
- Predictability
- High throughput
- Low latency
- Fixed time budget
- Fast algorithms
- Limited memory footprint

=> Avoid data losses, they cannot be recovered

Offline

- Focus on physics quality: high efficiency, low fake rate, good resolution
- The limits on the resources (CPU, memory) come mainly from the available (GRID) infrastructure
- The processing can (in theory) be repeated

=> Get the best "physics quality" with "reasonable" resources

Online vs Offline: technology

Online

- Use of accelerators (FPGA, GPGPU, etc.)
- Parallel processing with many attributes
  - Multithreading
  - Multiprocessing
  - Shared memory & DMA
  - Pipelining and buffers
- Hardware components:
  - Network: cards, switches
  - Special components

Offline

- Accelerators are almost not used (the GRID sites do not provides them by default)
- Mostly sequential processing: one raw file is reconstructed in one process
- The hardware components are "hidden":
  - Keep under control memory and CPU usage

Online vs Offline: sociology

Online

- Smaller groups mainly consisting of hardware and computing experts
- Compact location
- Common computing science language
- Sometimes non-public code, repositories containing also proprietary software, medium size (~100 KLOC)

Offline

- Larger heterogeneous groups including many physicists
- Spread around the world
- Common language from particle physics
- As a rule public repositories with millions of LOC

Online vs Offline: programming

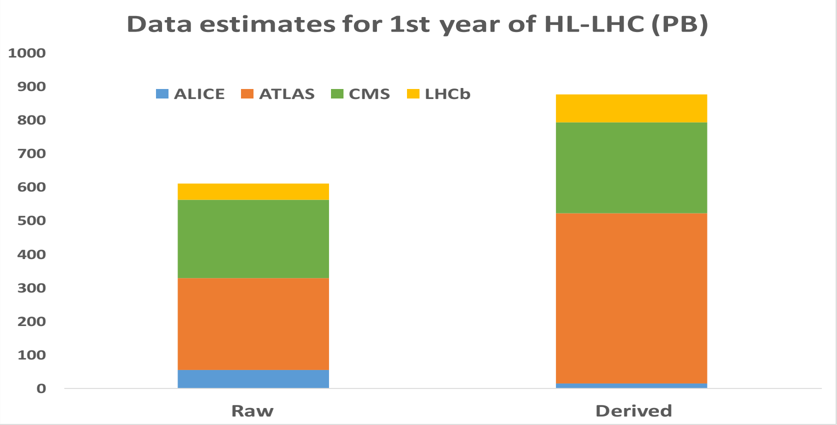
Online

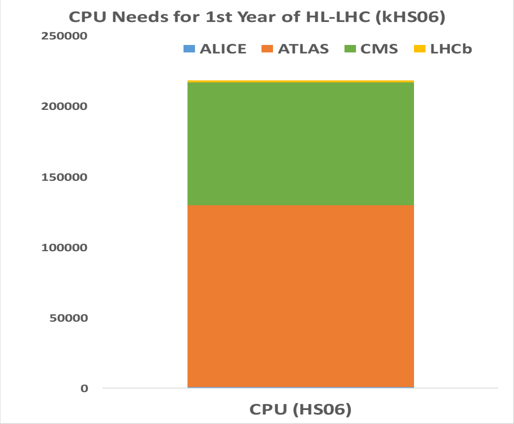
- Programming language
  - General purpose: C/C++
  - FPGA: VHDL, Verilog, OpenCL
  - GPGPU: CUDA, OpenCL, …
- Mostly "C-style" design
  - POD structures
  - Avoid deep inheritance and virtual methods. Static polymorphism.
- ROOT may be used only at the latest stages of processing
- Sometimes statically linked executables

Offline

- Programming languages: C++, Fortran, Python
- OO design with full list of features
  - Deep inheritance chains
  - Virtual methods and polymorphism
  - Templates and STL
  - Complex objects
- ROOT is used almost at each stage
- As a rule dynamically linked executables

Estimates of resource needs for HL-LHC





Data:
- Raw 2016: 50 PB → 2027: 600 PB
- Derived (1 copy): 2016: 80 PB → 2027: 900 PB

CPU:
- x60 from 2016

Presented by Ian Bird 21/09/2016 @ LHCC

By far the most quoted slide @ CHEP2016

Technology at ~20%/year will bring x6-10 in 10-11 years

- Simple model based on today's computing models, but with expected HL-LHC operating parameters (pile-up, trigger rates, etc.)
- At least x10 above what is realistic to expect from technology with reasonably constant cost

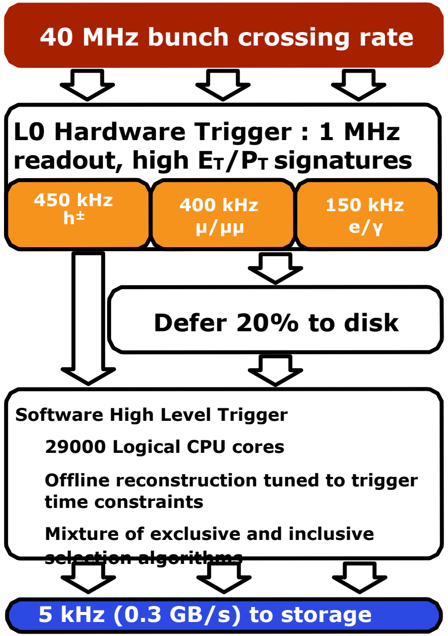
Trigger (if you can)

- Possible = selective AND efficient
  - High PT physics
  - High energy e/γ
  - Jets
  - “What’s possible is done!”
- Not possible = not selective OR inefficient
  - “Soft” new physics
  - Complex signatures: displaced secondary vertices, particle identification, etc.
  - Need for full reconstruction to select interesting events

- Trigger-less DAQ becomes popular
  - Run3 LHCb ~4 TB/s
  - Run3 ALICE ~3.4 TB/s
  - CBM ~ 1 TB/s (in 2020+)
  - Panda ~300 GB/s (in 2020+)
  - LSST ~3 GB/s
  - mu2e ~ 30 GB/s
  - DUNE ~ 1 TB/s (in 2020+)

LHCb: A Working Model for Future Experiments

LHCb 2012 Trigger Diagram



40 MHz bunch crossing rate

L0 Hardware Trigger : 1 MHz readout, high E_T/P_T signatures

450 kHz h±	400 kHz μ/μμ	150 kHz e/γ

Defer 20% to disk

Software High Level Trigger

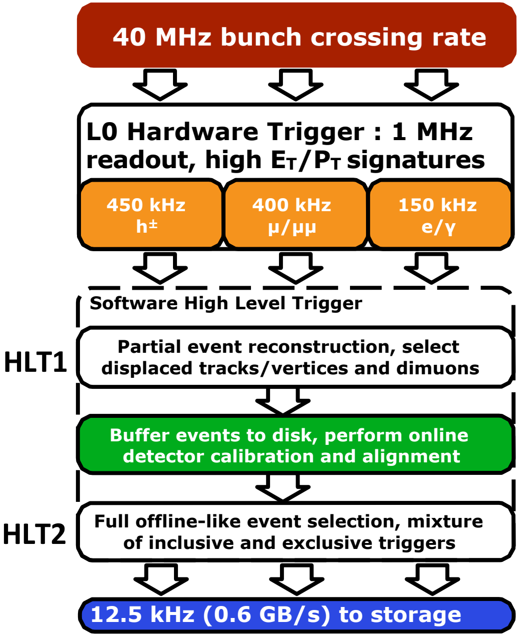
29000 Logical CPU cores

Offline reconstruction tuned to trigger time constraints

Mixture of exclusive and inclusive selection algorithms

5 kHz (0.3 GB/s) to storage

LHCb 2015 Trigger Diagram



40 MHz bunch crossing rate

L0 Hardware Trigger : 1 MHz readout, high E_T/P_T signatures

450 kHz h±	400 kHz μ/μμ	150 kHz e/γ

Software High Level Trigger

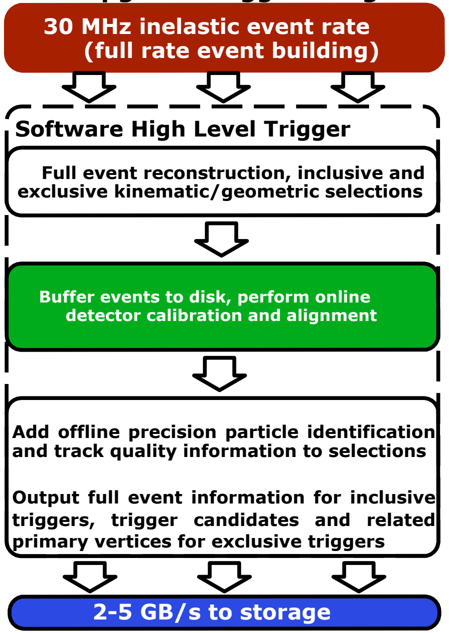
HLT1 Partial event reconstruction, select displaced tracks/vertices and dimuons

Buffer events to disk, perform online detector calibration and alignment

HLT2 Full offline-like event selection, mixture of inclusive and exclusive triggers

12.5 kHz (0.6 GB/s) to storage

LHCb Upgrade Trigger Diiagram



30 MHz inelastic event rate (full rate event building)

Software High Level Trigger

Full event reconstruction, inclusive and exclusive kinematic/geometric selections

Buffer events to disk, perform online detector calibration and alignment

Add offline precision particle identification and track quality information to selections

Output full event information for inclusive triggers, trigger candidates and related primary vertices for exclusive triggers

2-5 GB/s to storage

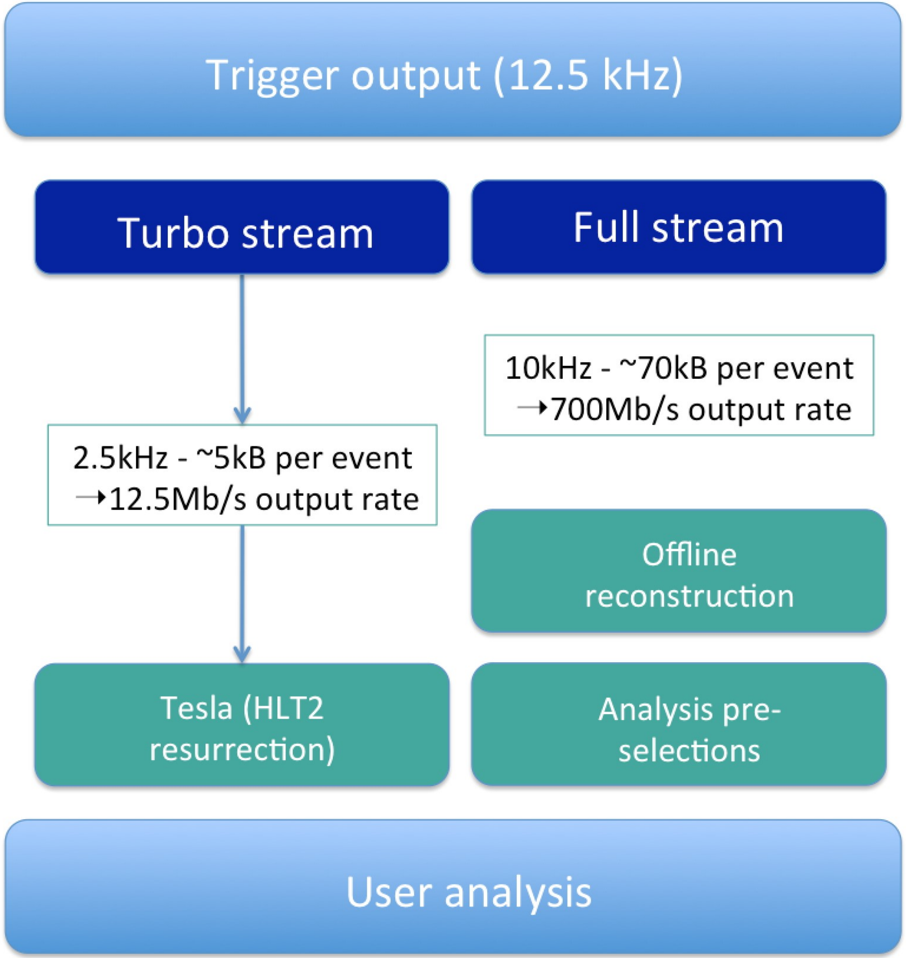
Buffering and automation: Run2 real time alignment and calibration:

- Alignment sequence ~O(10 min): Velo, Tracker, Muon, RICH1, RICH2
- Calibration ~O(10 min): RICH (refraction, HPD), Outer tracker (drift time)

Run3: only software trigger

LHCb Turbo Stream: use trigger information in analysis

- For charm physics, must rely (mainly) on exclusive triggers to limit rate
- By *construction*, trigger information is sufficient for most charm analysis
- 2016: 150 out of 420 HLT2 ‘lines’ are Turbo
- Purity and resolution for charged particles equivalent to best Run1 offline results
- The offline reconstruction becomes redundant - the best (or “good enough”) reconstruction is already done online
- Turbo++: enable additional analysis
- At the end: keep only analysis specific information for each trigger class



Trigger output (12.5 kHz)

Turbo stream

Full stream

10kHz - ~70kB per event
→700Mb/s output rate

2.5kHz - ~5kB per event
→12.5Mb/s output rate

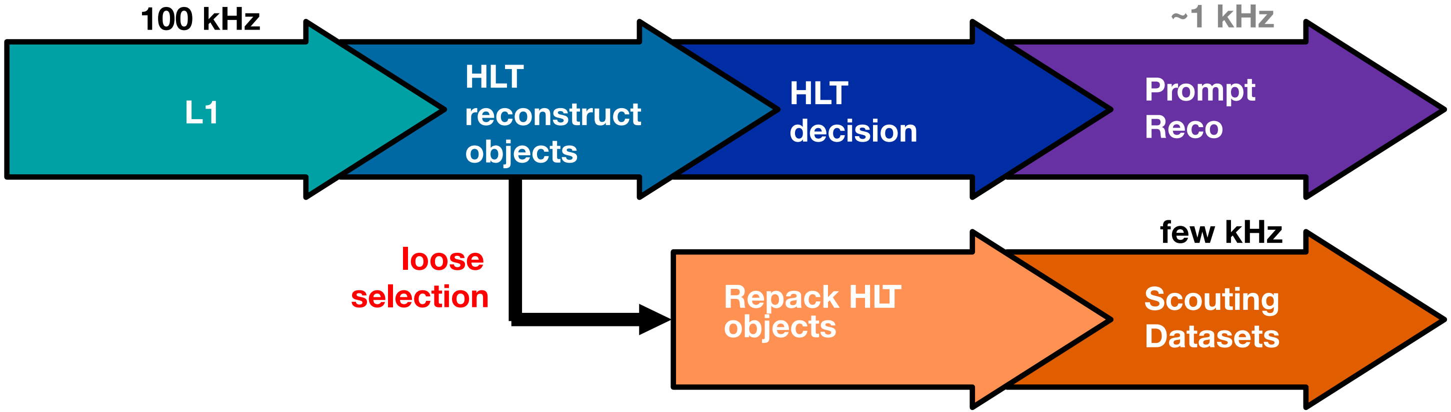
Offline reconstruction

Tesla (HLT2 resurrection)

Analysis pre-selections

User analysis

CMS Scouting



100 kHz

L1

HLT reconstruct objects

HLT decision

~1 kHz

Prompt Reco

loose selection

few kHz

Repack HLT objects

Scouting Datasets

- Scouting allows to workaround the limitations of HLT rate and to lower thresholds
  - Resources for Prompt Reco → save directly HLT objects, including particle flow candidates!
  - DAQ bandwidth → event size O(1-10) kB compared to ordinary O(1) MB
  - CPU resources at HLT farm → run in shadow, use objects already reconstructed by other paths
- Run4 scouting: extended analysis on federated detector/trigger data

ALICE Online-Offline (O^2)

Requirements

1. LHC min bias Pb-Pb at 50 kHz
   ~100 x more data than during Run 1
2. Physics topics addressed by ALICE upgrade
   - Rare processes
   - Very small signal over background ratio
   - Needs large statistics of reconstructed events
   - Triggering techniques very inefficient if not impossible
3. 50 kHz > TPC inherent rate (drift time ~100 μs)
   Support for continuous read-out (TPC)
   - Detector read-out triggered or continuous

New computing system

- Read-out the data of all interactions
- ➔ Compress these data intelligently by online reconstruction
- ➔ One common online-offline computing system: O^2
- Paradigm shift compared to approach for Run 1 and 2

Unmodified raw data of all interactions shipped from detector to online farm in trigger-less continuous mode

HI run 3.4 TByte/s ⇩

Baseline correction and zero suppression
Data volume reduction by cluster finder. No event discarded.
Average compression factor 6.6

500 GByte/s ⇩

Data volume reduction by online tracking.
Only reconstructed data to data storage.
Average compression factor 5.5

90 GByte/s ⇩

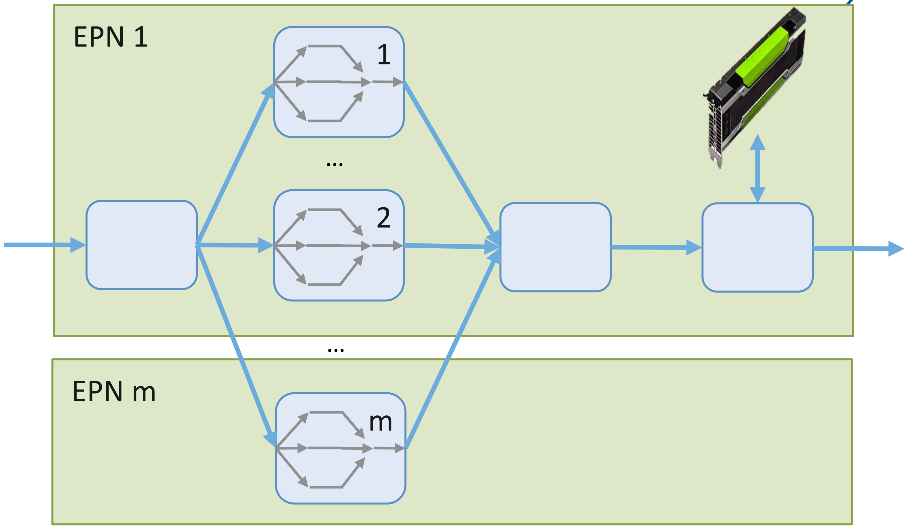
Data Storage: 1 year of compressed data
- Bandwidth: Write 170 GB/s Read 270 GB/s
- Capacity: 60 PB

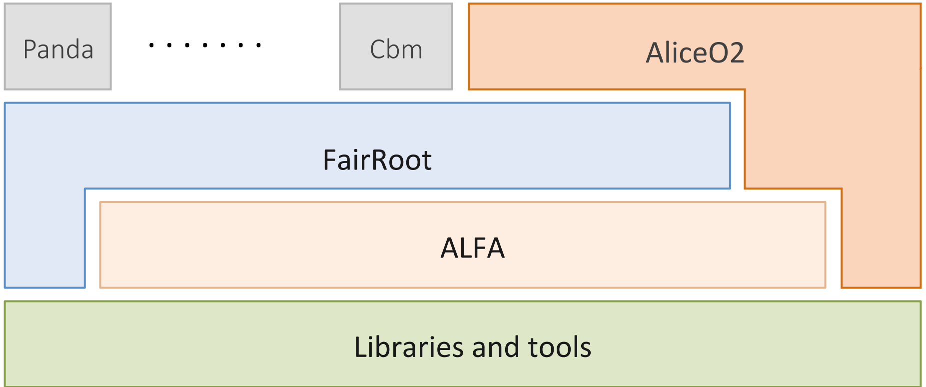
20 GByte/s ⇳ Tier 0, Tiers 1 and Analysis Facilities

⇳ Asynchronous (few hours) event reconstruction with final calibration

ALICE O2 Software Design

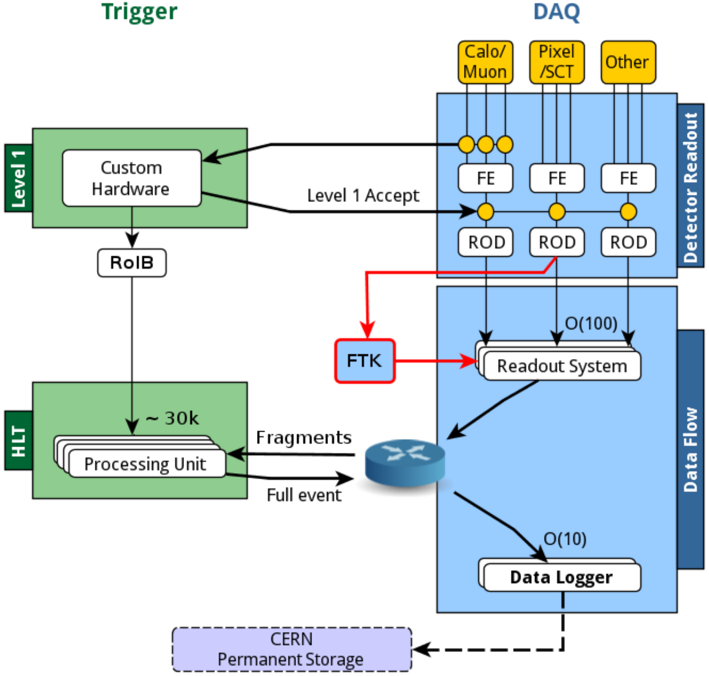
- Message-based multi-processing
  - Ease of development
  - Ease to scale horizontally
  - Possibility to extend with different hardware
  - Multi-threading possible within processes
- ALFA : ALICE-FAIR concurrency framework
  - Data transport layer
  - ZeroMQ
  - Multi-process
  - Steady development
- AliceO2
  - Prototyping
  - Development started

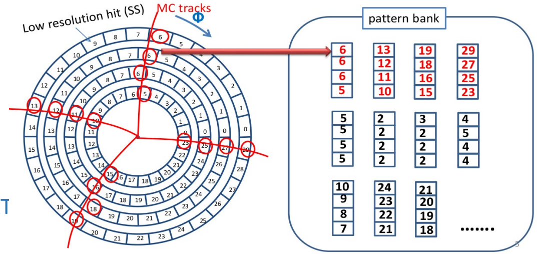




Hardware can help: ATLAS Fast Tracker (FTK)

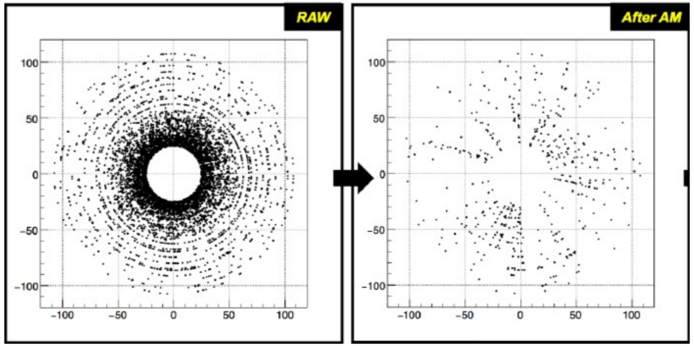
- A co-processor for the ATLAS HLT
  - Based on CDF's Silicon Vertex Tracker (SVT)
  - High throughput (40M tracks/ s) and low latency (100 µs)
  - Tracks for full event available to HLT
  - Fully installed (up to $\mu=40$) by end of 2016
- Design
  - Parallelism: 64 independent towers (4 in η x 16 in φ)
  - Hardware: custom ASICs and FPGAs
  - Two stages:
    1. Pattern matching with 8 detector layers
       - Uses Associative Memory (AM): 1 billion patterns
       - Reduced granularity: Pixels/Strips grouped to super strips
    2. Extension to 12 layers
  - Track parameters extracted on FPGA using Principle Component Analysis => Sum rather than fit

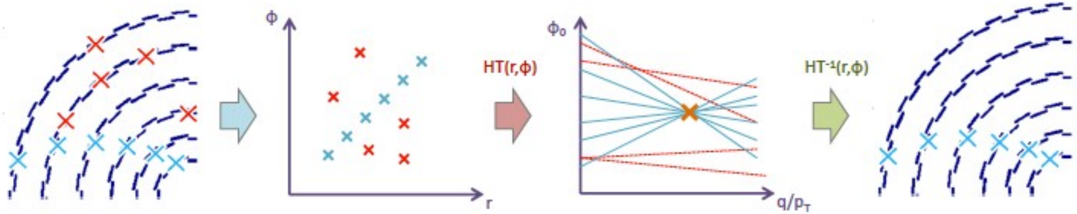




Hardware can help: CMS L1 track finding

- ASIC-assisted approach: Associative memory + FPGA, similar to the ATLAS FTK
- Purely FPGA-based
    - Hough transform:
        - geometric processor (GP) sorts stubs in 36 subdivisions of the octant
        - coarse HT ran on the stubs
        - stubs from HT track candidates not consistent with the track in the r-z plane are filtered out
        - duplicates are removed
        - final TF is performed to accurately determine track parameters
    - Combined Tracklet Builder & linearized track fit





GPGPU can help: ALICE GPU Track finder

- TPC Volume is split in 36 sectors.
  - The tracker processes each sector individually.
  - Increases data locality, reduce network bandwidth, but reduces parallelism.
  - Each sector has 160 read out rows in radial direction.

1. Phase: **Sector-Tracking** (within a sector)

- Heuristic, combinatorial search for track seeds using a Cellular Automaton, GPU or CPU
  - Looks for three hits composing a straight line (link).
  - Concatenates links.
- Fit of track parameters, extrapolation of track, and search for additional clusters using simplified Kalman Filter: GPU or CPU

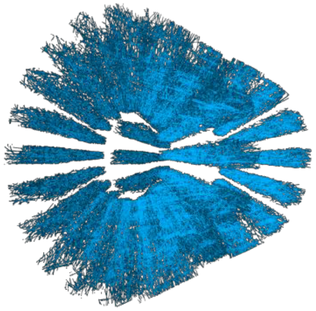
2. Phase: **Track-Merger**, CPU only
  - Combines the track segments found in the individual sectors.

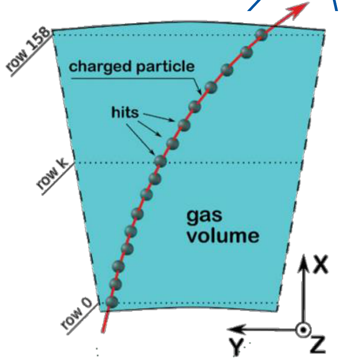
3. **Track fitter** using full Kalman filter: CPU (or GPU)

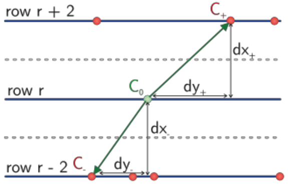
Runs on CUDA, OpenCL, OpenMP – one common shared source code

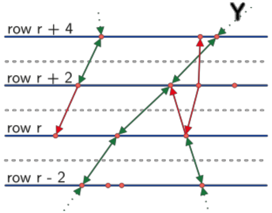
HLT tracking 15x faster on CPU wrt Offline

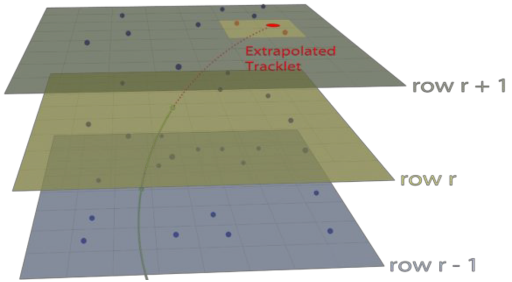
GPU speedup of 10 => speedup factor 150!











Online and Offline: towards "Great Unification"?

- The current and especially the future needs define a trend
- If Online is:
  - Moving towards "offline" quality of the results;
  - Carrying on "offline" tasks such as alignment and calibration;
  - Running "offline" algorithms;
  - Providing data for fast physics analysis.
- If Offline is exploring:
  - Multi-threading and message based multiprocessing like in "online";
  - Accelerators (FPGA, GPGPU);
  - Heterogeneous clusters;
  - "Online" algorithms.
- => the Online and the Offline converge to an Online-Offline system!
- Some tasks will remain online or offline specific

Online and Offline: towards "Great Unification"?

- The success of this process depends on several factors:
  - People
  - Software frameworks
  - Development process
  - Technology/Hardware availability
- Close collaboration is needed to achieve success!

References

- Connecting the Dots 2016
- ISOTDAQ 2016 - International School of Trigger & Data
- CERN Academic Training: Trigger/DAQ for Particle Physics Detectors
- ALICE, ATLAS, CMS & LHCb Second Joint Workshop on DAQ@LHC
- ECFA High Luminosity LHC Experiments Workshop - 2016
- CPAD Instrumentation Frontier Meeting 2016
- CHEP2016