Data processing in HEP experiments

Oleynik D. JINR LIT

A long times ago….

Tycho Brahe

1546 - 1601

Danish astronomer who designed and constructed greatly improved astronomical instruments. This increased the accuracy of measurements.

18 years of astronomy observations, precision measurement and systematisation of collected data by Tycho Brahe were complexly analysed by Johannes Kepler, which finally allowed to formulate "Kepler's laws of planetary motion" and eventually paving the way for Isaac Newton theory of universal gravitation.

Johannes Kepler

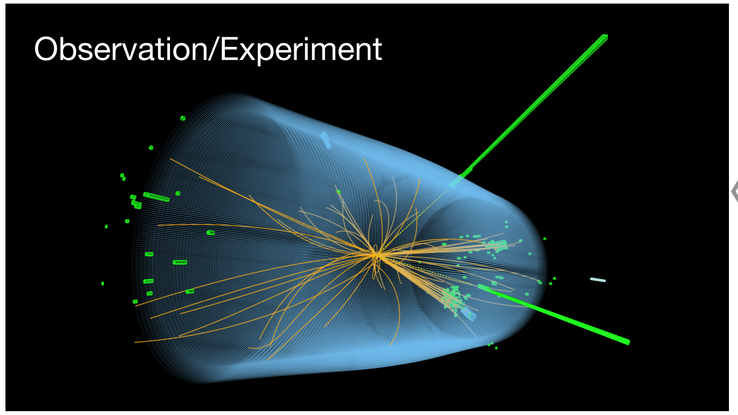
1570 - 1601

Mathematician who used Tycho's observations of the heavens to validate the Copernican model.

Data Acquisition

Data Analysis

Where computing in physics research?

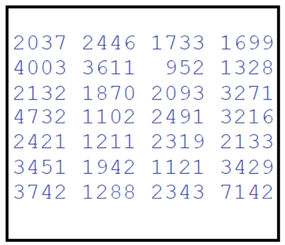


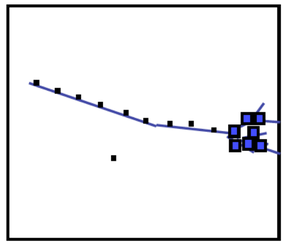
Computation / Analysis

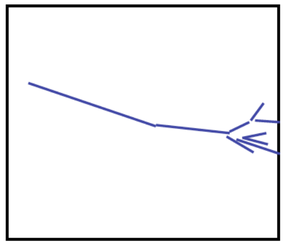
Theory

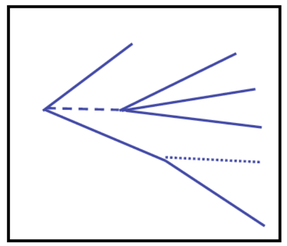
$$\mathcal{S}[\varphi_i] = \int \mathcal{L}[\varphi_i(s)]\, d^n s,$$

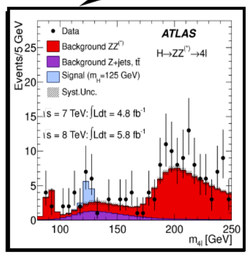
Base streams of data processing











RAW DATA

CONVERT TO PHYSICS QUANTITIES

DETECTOR RESPONSE APPLY CALIBRATION, ALIGNMENT,

INTERACTION WITH DETECTOR MATERIAL PATTERN, RECOGNITION, PARTICLE IDENTIFICATION

FRAGMENTATION, DECAY PHYSICS ANALYSIS

BASIC PHYSICS

RESULTS

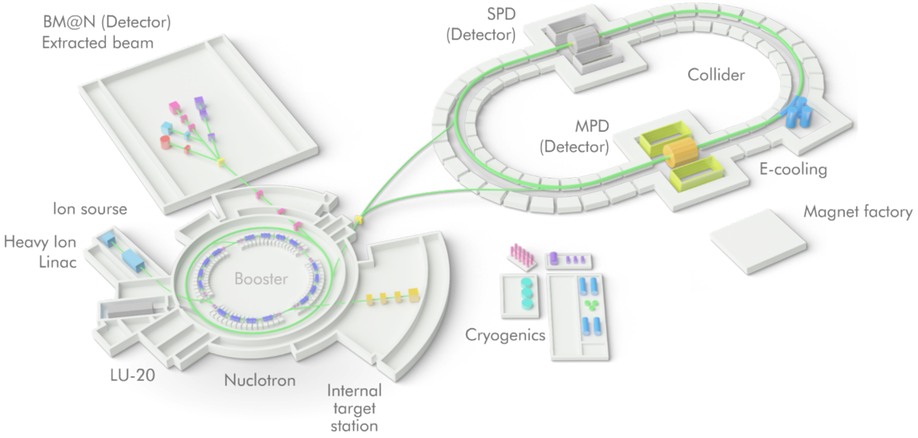
RECONSTRUCTION

ANALYSIS

SIMULATION (MONTE-CARLO)

What is HEP?

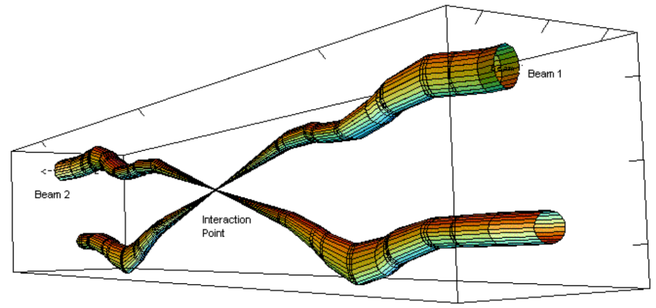
HEP - High Energy Physics



- **Particle physics** or **high energy** physics is the study of fundamental particles and forces that constitute matter and radiation (Wikipedia)
- A **particle accelerator** is a machine that uses electromagnetic fields to propel charged particles to very high speeds and energies, and to contain them in well-defined beams
- A **particle detector**, also known as a **radiation detector**, is a device used to detect, track, and/or identify ionizing particles. Detectors can measure the particle energy and other attributes such as momentum, spin, charge, particle type, in addition to merely registering the presence of the particle.

Data granularity

What is “Event”?



Relative beam sizes around IP1 (Atlas) in collision

In physics, and in particular relativity, an event is the instantaneous physical situation or occurrence associated with a point in spacetime (wikipedia).

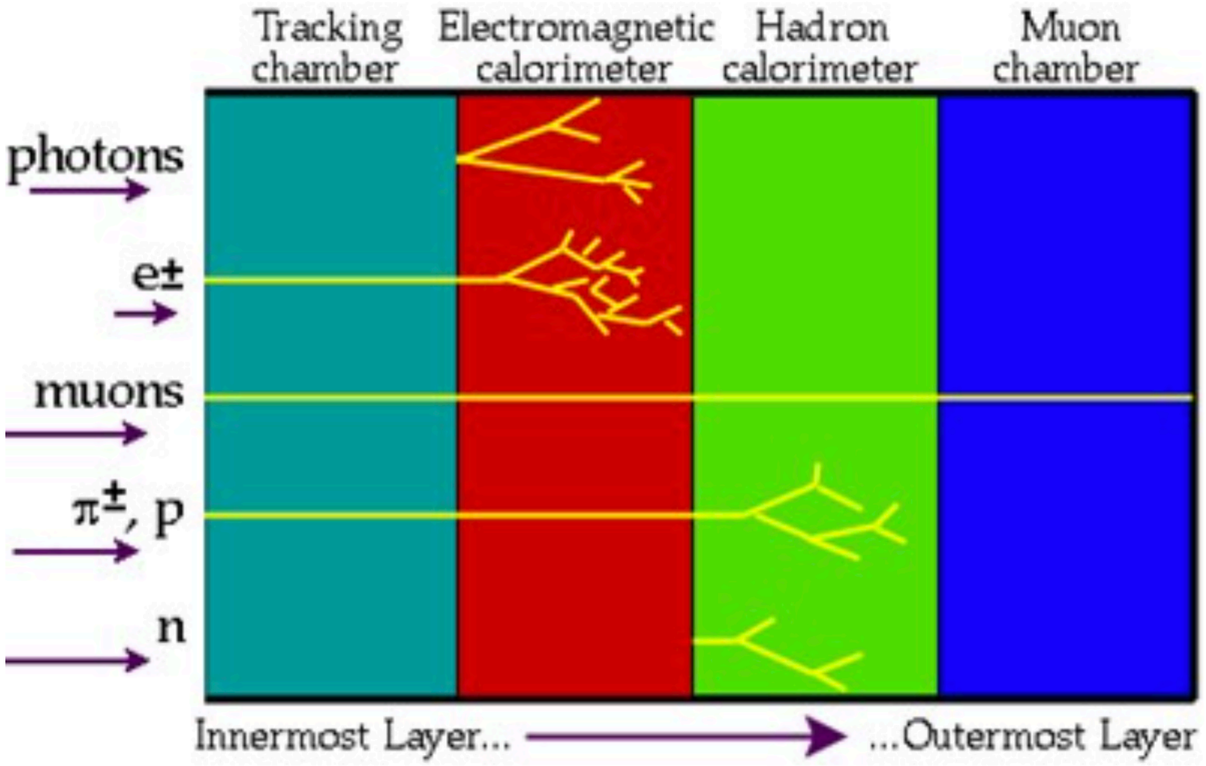
For HEP:

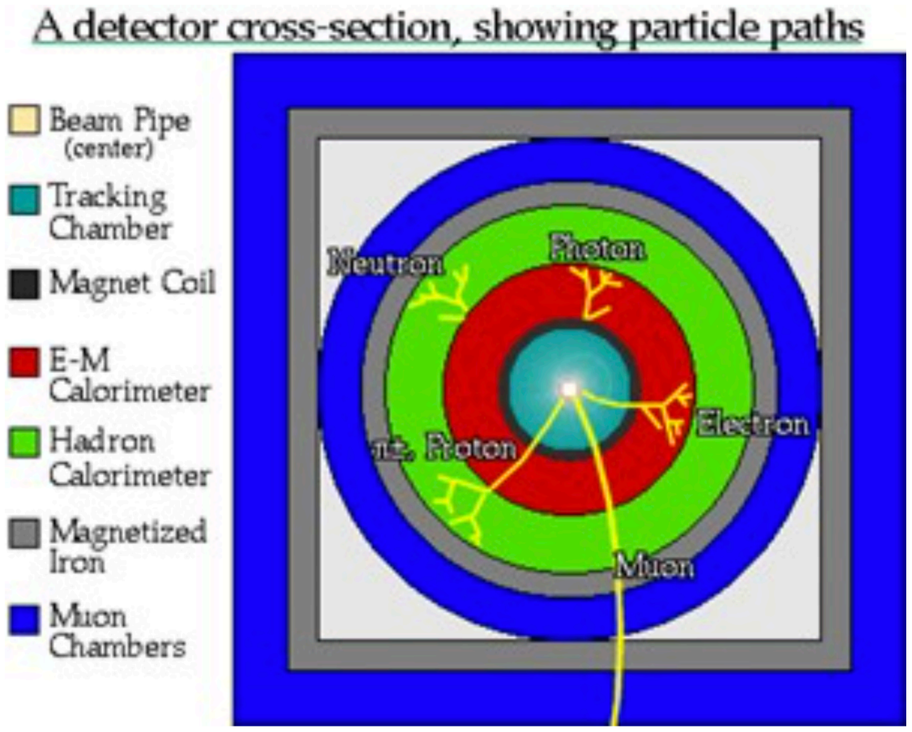
- **Interaction:** Two particles interact and somehow produce a change, be it in energy, trajectory or identity.
- **Collision:** Two particles are made to approach one another and actually undergo an interaction.
- **Beam crossing:** Two beam bunches pass through one another in the center of a detector.
- **Event:** During a beam crossing, one pair or multiple pairs of particles undergo a collision. In an event, often one collision dominates the signature in the detector.

Each event may be processed independently, so for HEP computing - event is the least data unit.

By knowing the size of event, complicity of event processing and expected number of event for processing, you can estimate requirements for computing system.

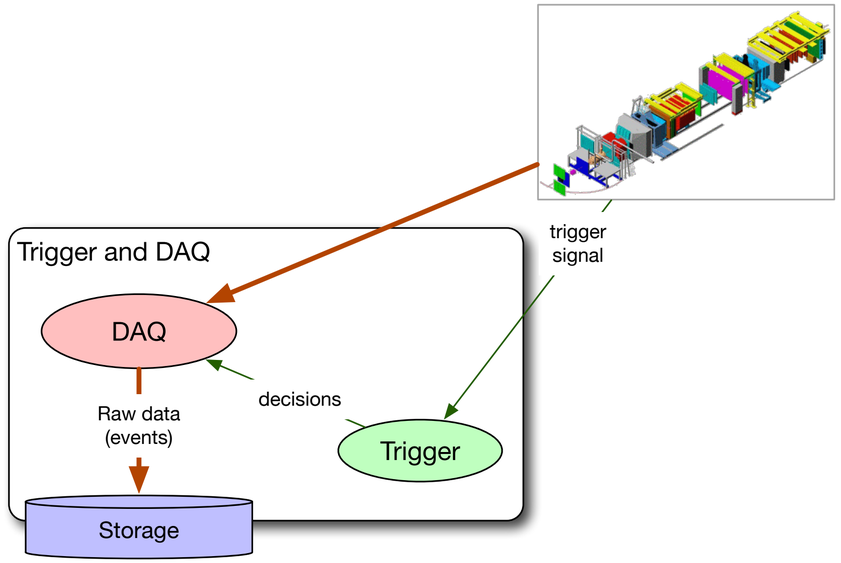
Generic HEP detector





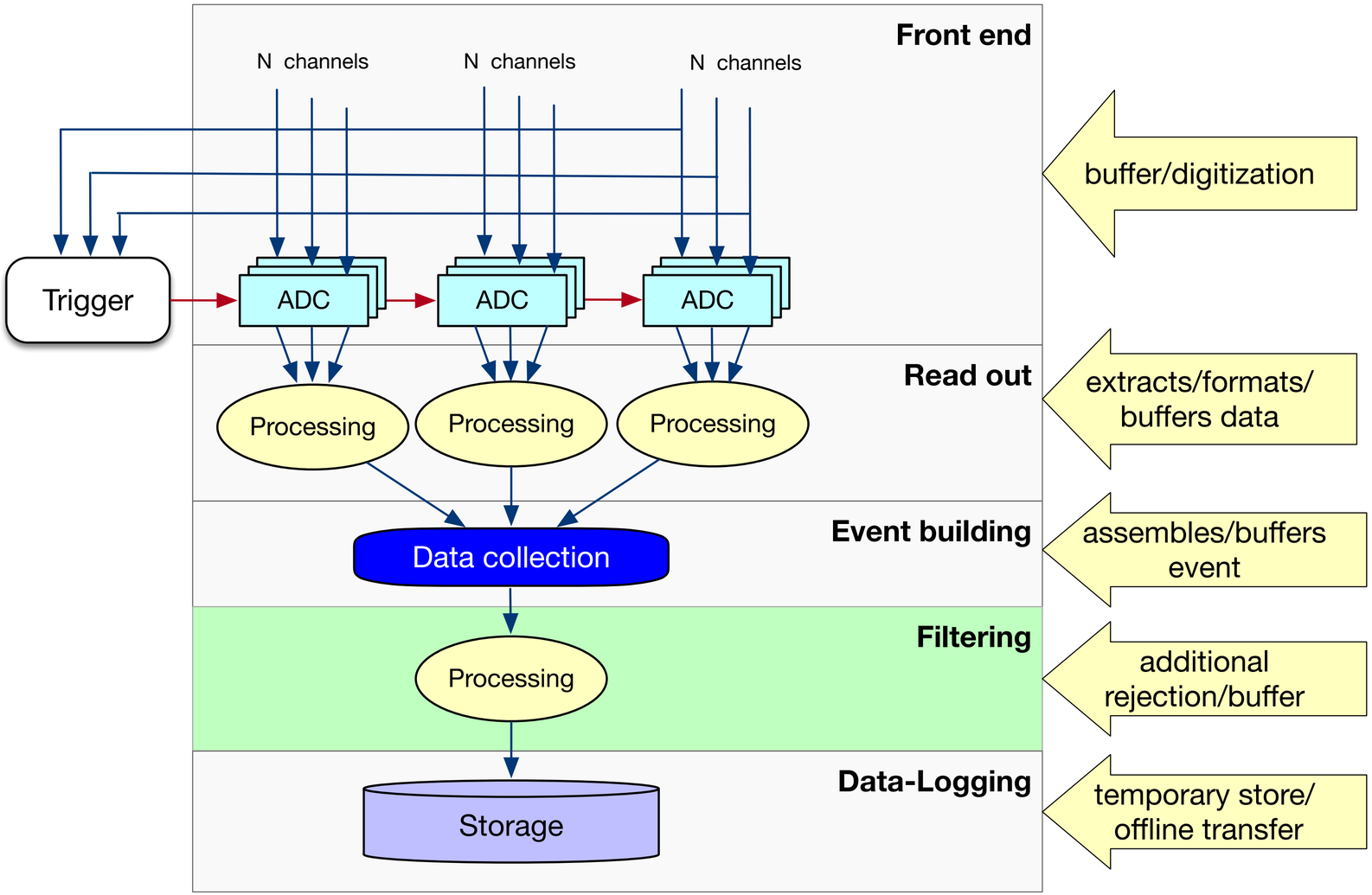
Data Acquisition System

What is trigger?



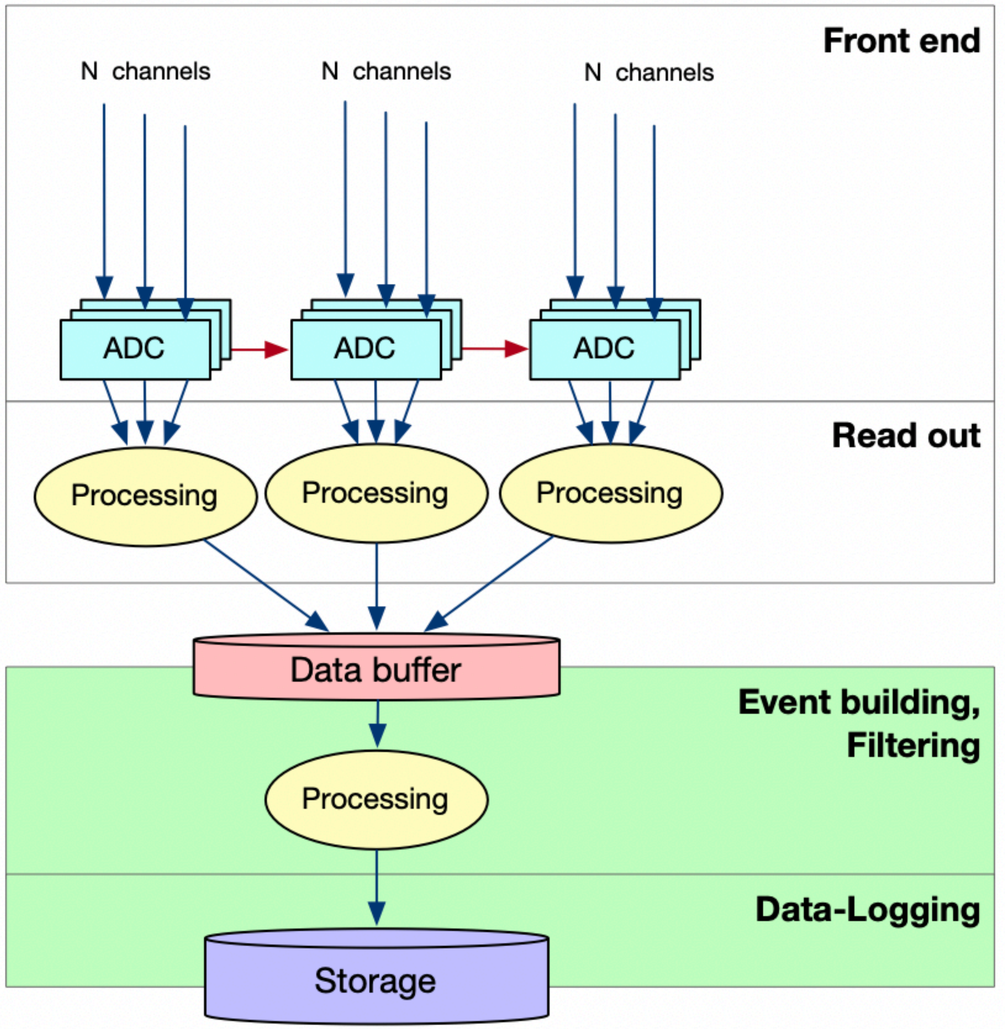
- Main role of Trigger & Data acquisition (DAQ):
  - process the signals generated in the detectors
  - Select the ‘interesting’ events and reject the ‘boring’ ones
  - save interesting ones on mass storage for future processing or analysis
- Trigger, in general, something which tells you when is the “right” moment to take your data
- Trigger – process to very rapidly decide if you want to keep the data if you can’t keep all of them. The decision is based on some ‘simple’ criteria

“Classical" DAQ



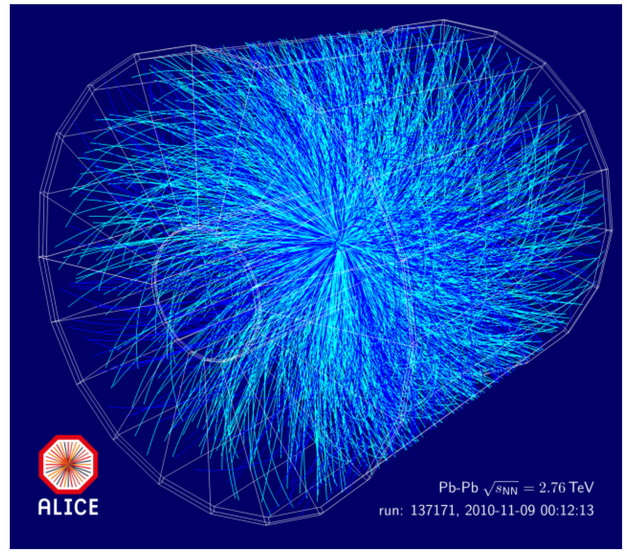
Trigerless DAQ

- Triggerless DAQ, means that the output of the system will not be a dataset of raw events, but a set of signals from sub-detectors organized in time slices
- To get data in proper format for future processing (reconstruction) and filtering of ‘boring’ events special computing facility named "Online Filter" in progress



How to estimate expected data volume

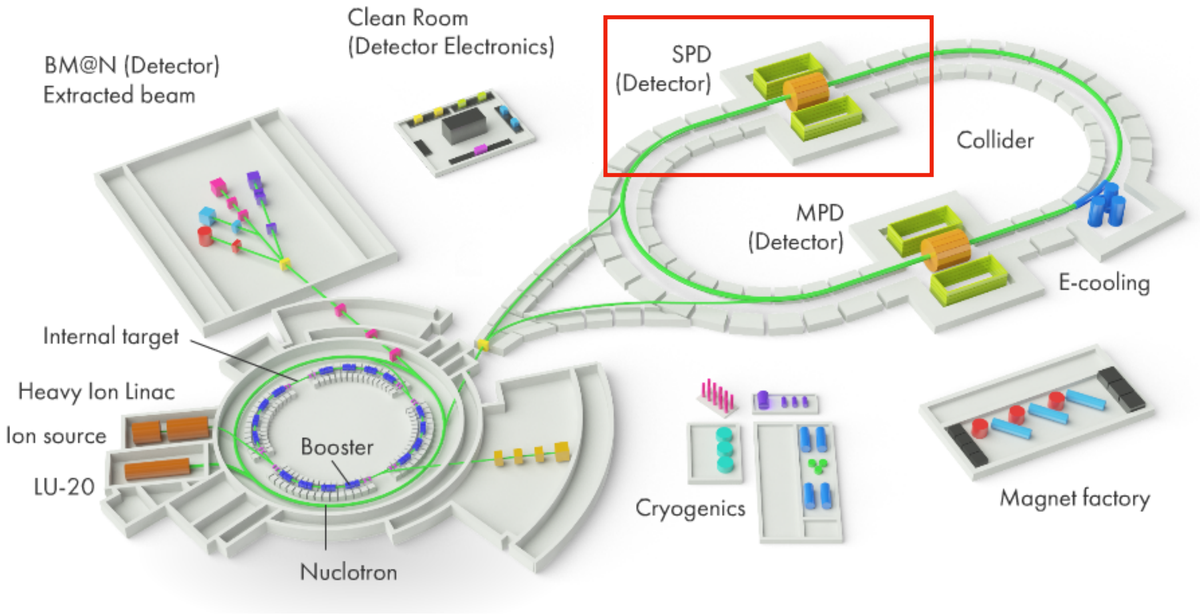
- Accelerator parameters: luminosity
  - NICA: $10^{27}\ cm^2 \cdot s^{-1}$ (nucleon-nucleon) – $10^{34}\ cm^2 \cdot s^{-1}$ (proton-deuteron)
- Expected efficiency of detector and DAQ: how many signals and how often they and be collected, how many events can be stored

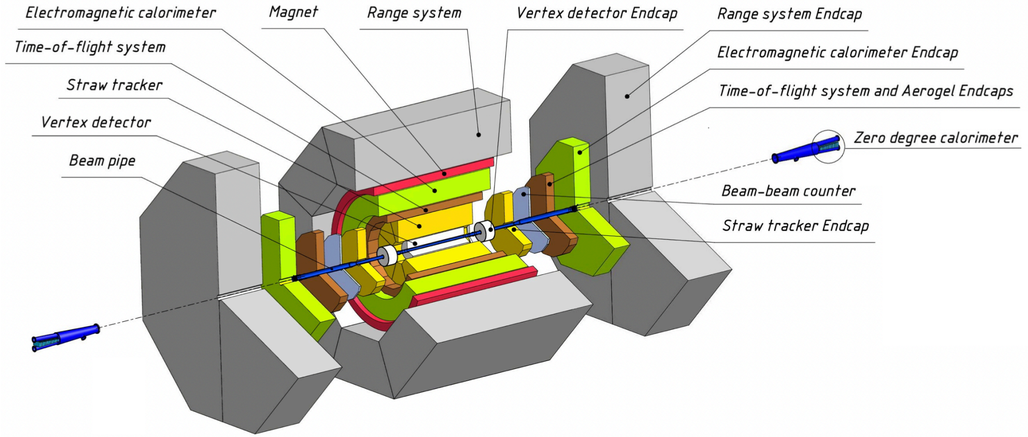


Experiment	Production rate (event/sec)	Raw event size (KB)
SPD	150 000	50
MPD	7 000	1500
BM&N	5 000	500

SPD Spin Physics Detector

Study of the nucleon spin structure and spin-related phenomena in polarized *p-p*, *d-d* and *p-d* collisions





SPD - a universal facility for comprehensive study of gluon content in proton and deuteron

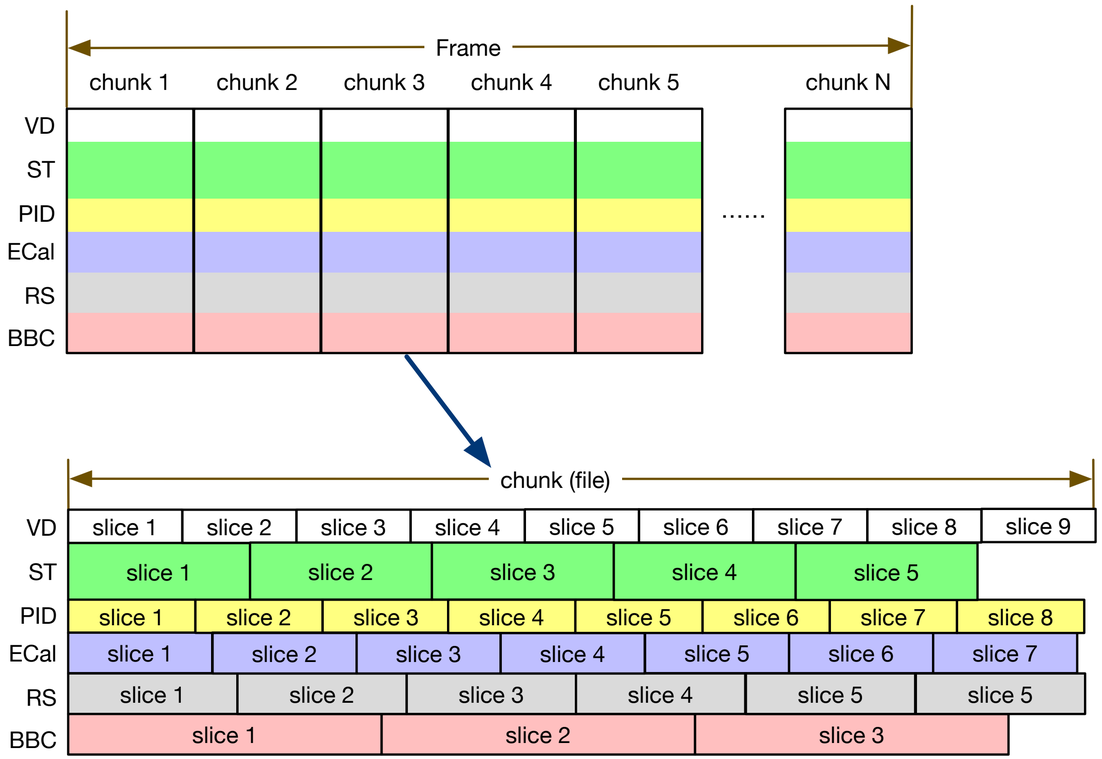
SPD as data source

- Bunch crossing every 80 ns = crossing rate 12.5 MHz
  - ~ 3 MHz event rate (at 10^{32} $cm^{-2}s^{-1}$ design luminosity) = **pileups**
- **20 GB/s** (or **200 PB/year** "raw" data, **~3*10^{13}** events/year)
  - Selection of physics signal requires momentum and vertex reconstruction → no **simple trigger** is possible
- Comparable amount of simulated data

High-throughput computing for SPD data processing

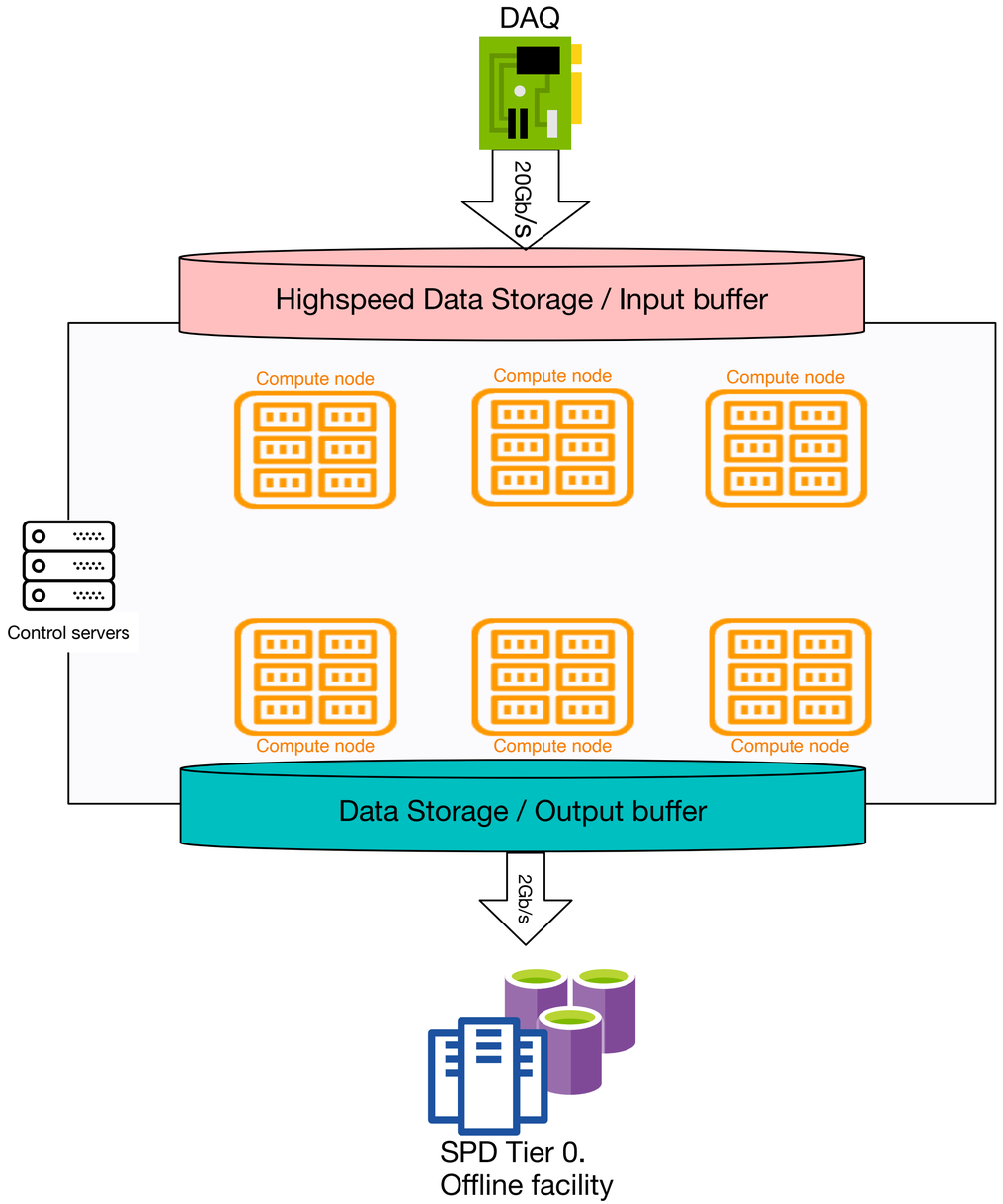
High-throughput computing (HTC) involves running many independent tasks that require a large amount of computing power.

- DAQ provide data organized in time frames and sliced to files with reasonable size (a few GB)
- Each of these file may be processed independently as a part of top-level workflow chain
- No needs to exchange of any information during handling of each initial file, but results of may be used as input for next step of processing.



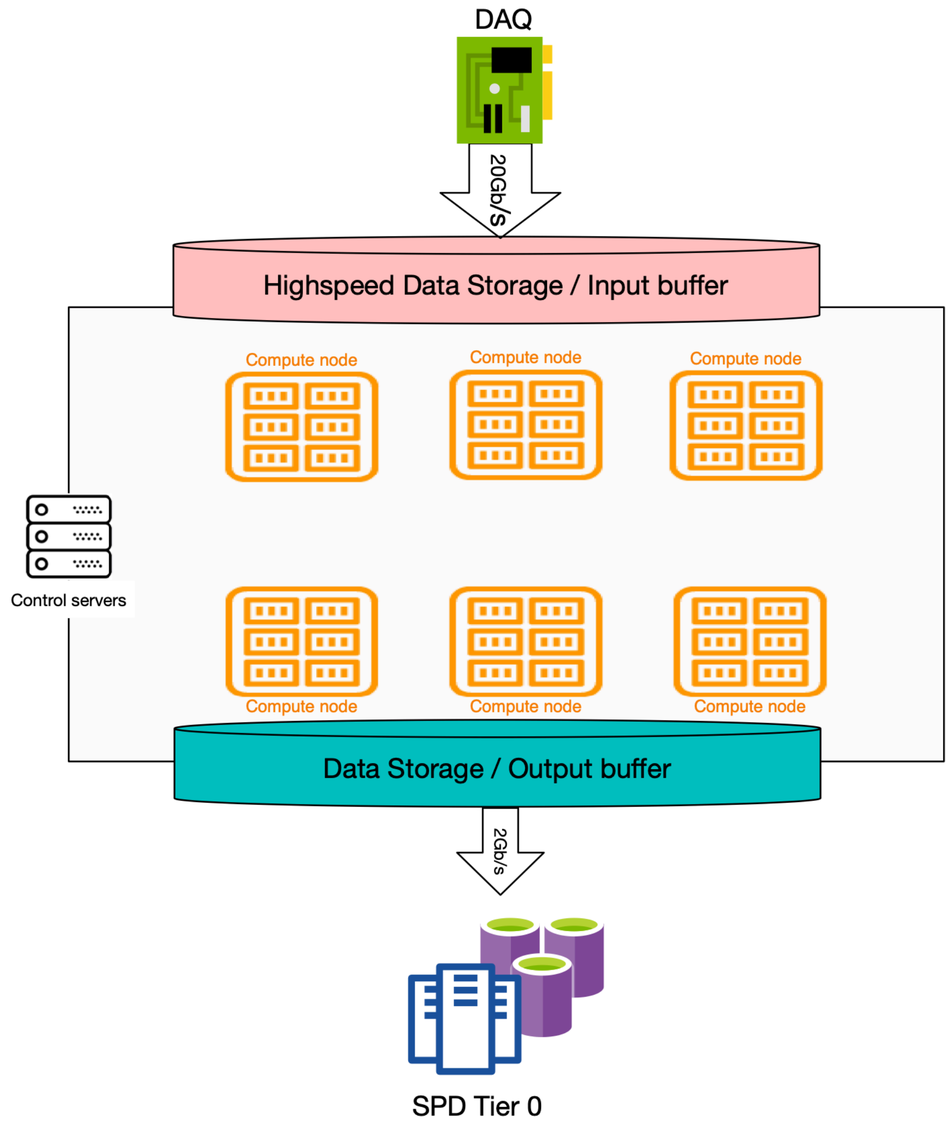
Online filter

- SPD Online Filter is a high performance computing system for high throughput processing
- This computing system should carry out next transformation of data: identify physics events in time slices; reorganize data (hits) in event's oriented format; filter 'boring' events and leave only 'hot'; settle output data, merge events into files and files in datasets for future processing



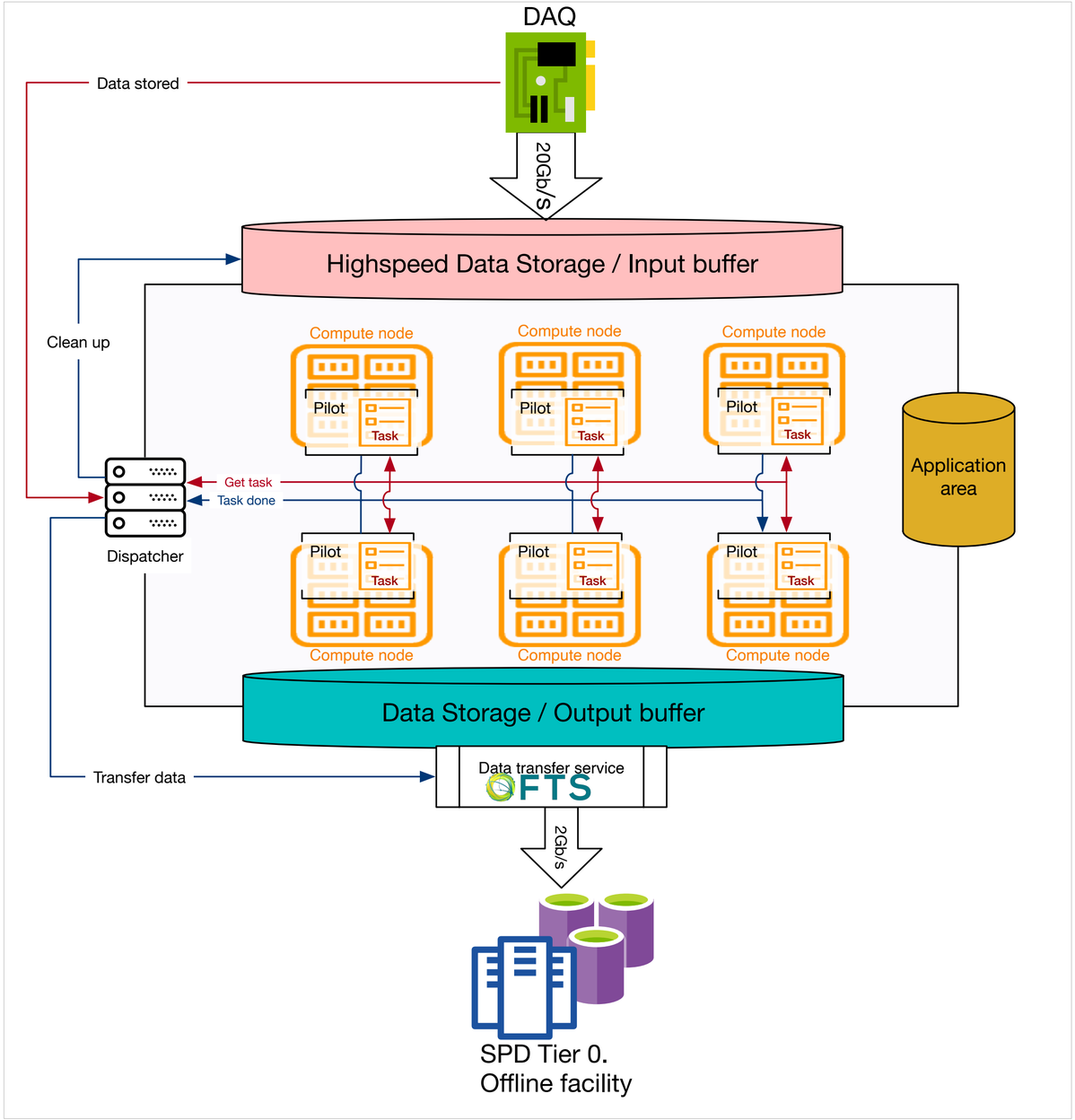
Online filter infrastructure

- High speed (parallel) storage system for input data written by DAQ.
- Compute cluster with two types of units: multi-CPU and hybrid multi CPU + Neural network accelerators (GPU, FPGA etc.) because we are going to use AI ;-).
- A set of dedicated servers for managing of processing workflow, monitoring and other service needs.
- Buffer for intermediate output and for data prepared for transfer to long-term storage and future processing.



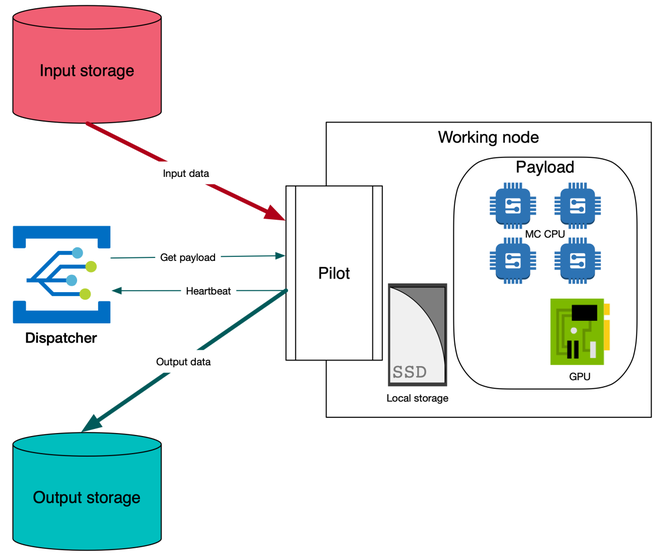
Online filter middleware

- HTC processing will be managed by a special software system which will automate processing workflow to achieve required performance.
- Software system have two main components:
  - **Dispatcher**, which control workflow execution
  - **Pilot** an application, which working on compute node, executes task generated by dispatcher



Online filter computing facility

- *Online computing facility should provide high-throughput data processing, by managing of handling of small parts of data on each compute node.*
- *Special service, which will manage processing workflow and dispatch jobs across compute nodes is required.*
  - *Pilot - the execution environment for compute jobs*
  - *Pilot applications continuously run on each compute node*
  - *A message queue technology is going to be used for communication*



Dispatcher required functionality

Data management;

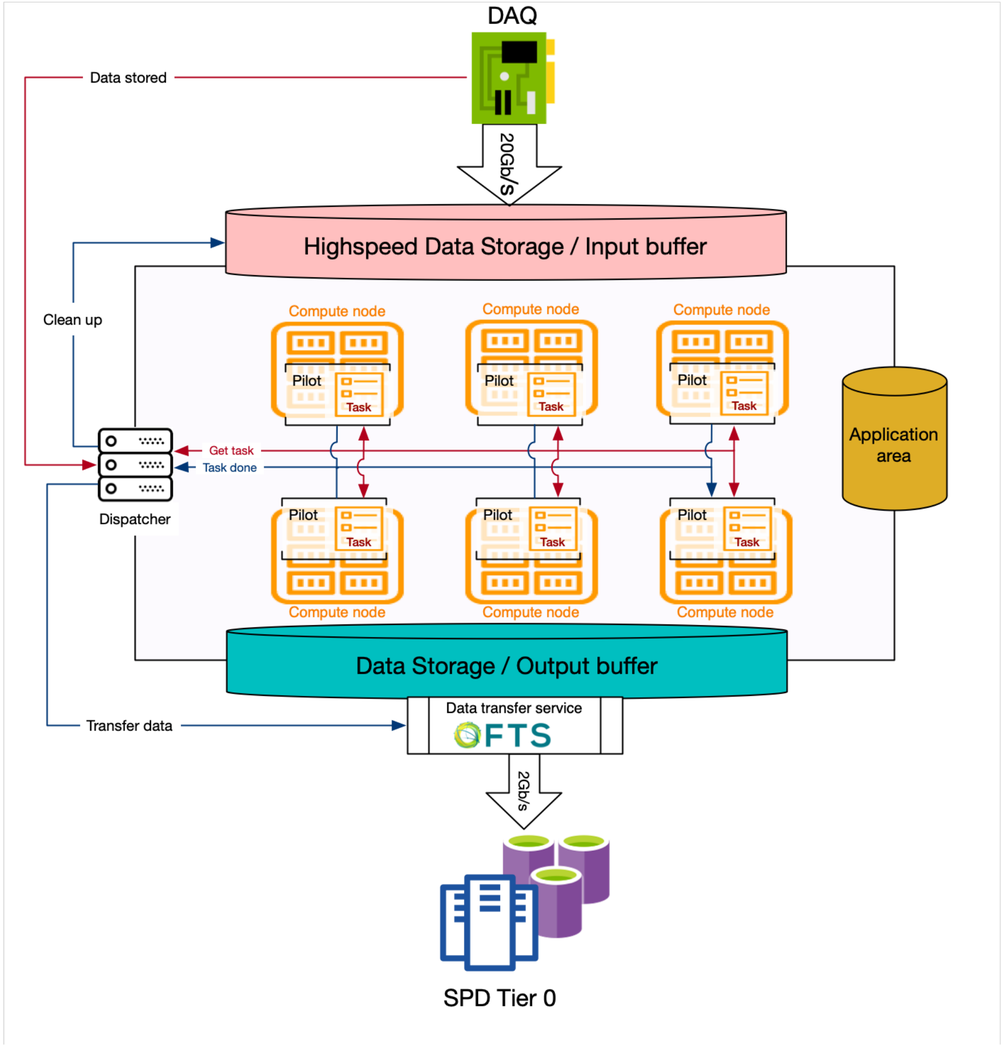
- *Support of data lifetime (registering, global transfer, cleanup);*

Processing management;

- *Generate jobs for each type of processing:*
  - *Events identification (building);*
  - *Verifying of processing results (AI vs traditional processing);*
  - *Select (Filter) events;*
  - *Pack (merge) output data for transferring to “offline”;*

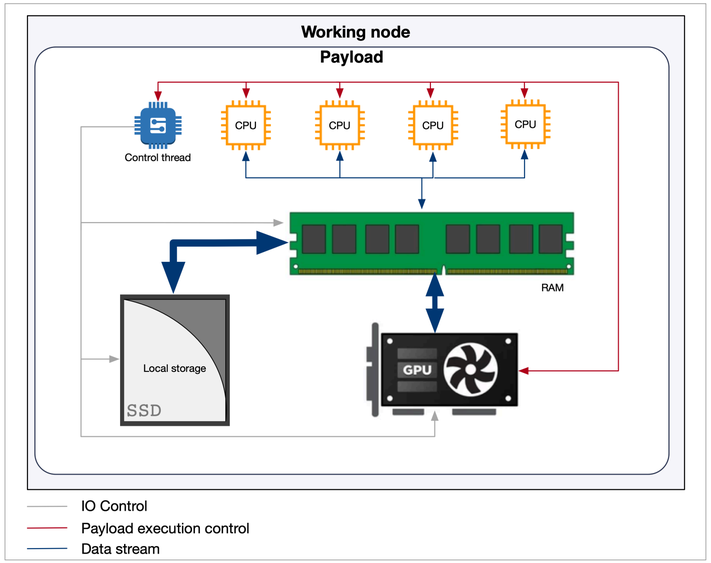
Workload management:

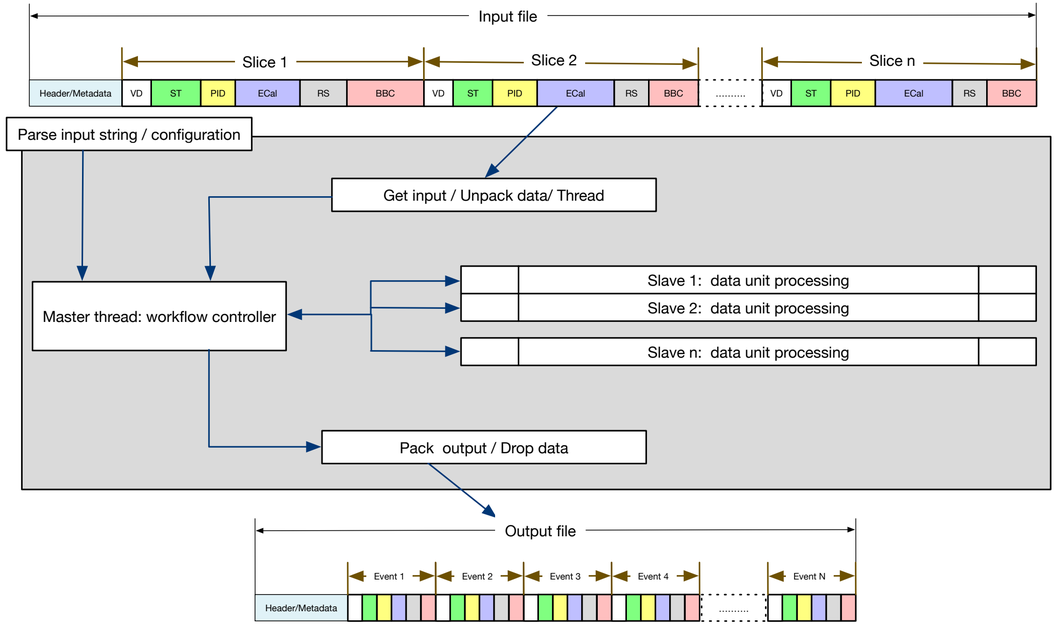
- *Dispatch jobs to pilots;*
- *Control of jobs executions;*
- *Control of pilots (identifying of "dead" pilots)*



Multithread processing

- Multicore computers already reality
  - Efficient usage requires multithreading processing
  - A lot of algorithms in HEP software stack does not support multithread execution (yet)
- We tries to explore multithread processing on data layer (each thread process own piece of data)





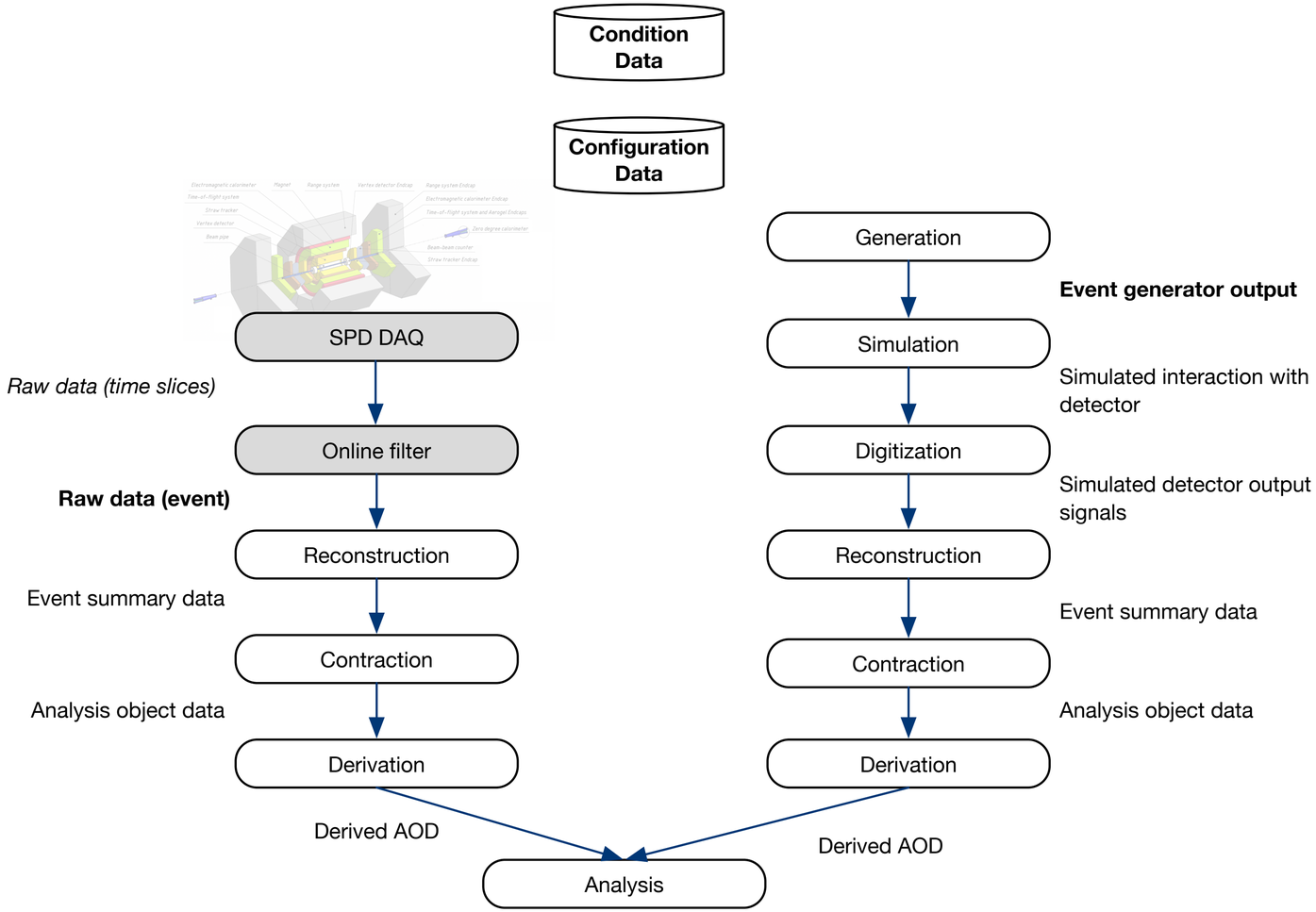
Event index & problem oriented databases

- Event index - is the set of special information systems which allows to store and navigate across all produced events
  - In simple words Event index allows identify dataset or even file where particular event is stored.
- Quite important system as only you start to use hundreds of thousands files
- Condition database - stores data which is not related with event production itself, but status of environment during data tacking
- Configuration database - stores detector hardware setup and other hardware related information

Offline processing

Reconstruction, Simulation

- Amount of data reduced, but data is not ready for analysis yet
  - Events contain raw or partially reconstructed data
  - Calibration and alignment is not applied yet
- Simulation - pure computation processing (will start mach early than apparatus will be ready)
  - Will require significant amount of computing resources



- Another type of computing facility required for routine offline processing – distributed data processing system (aka grid)

grid computing

Grid computing is the collection of computer resources from multiple locations to reach a common goal. The grid can be thought of as a distributed system with non-interactive workloads that involve a large number of files. Grid computing is distinguished from conventional high performance computing systems such as cluster computing in that grid computers have each node set to perform a different task/application.

We are in progress with a distributed computing system for offline processing of SPD data by incorporation of resources of experiment collaborators.

Basis of grid infrastructure

- Agreed rules of usage, shares of provided resources and level of participation
- Common authentication and authorization across infrastructure
- Common set of protocols and instruments for access to compute resources and data
- Information system with all required information about infrastructure
- A service which takes care of proper data catalog and data distribution
- A service which manage jobs execution

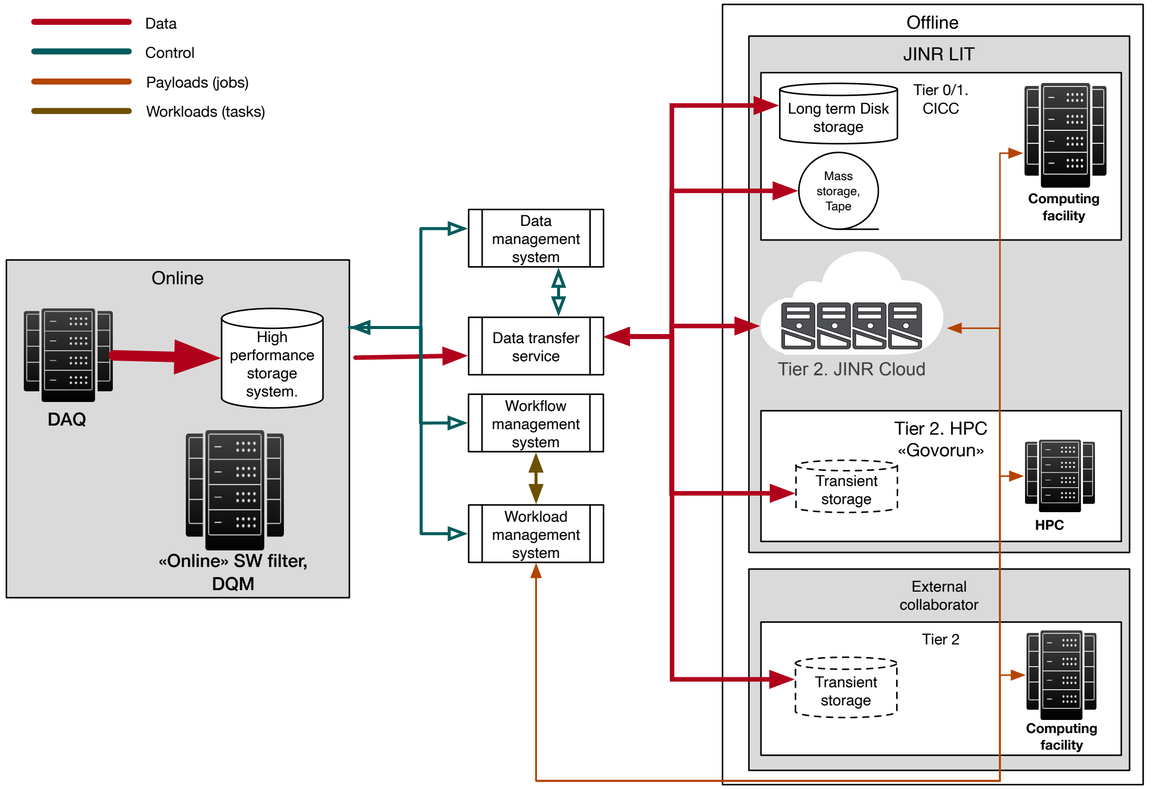
Estimated data volumes in numbers

Why we need grid?

- Expected that $2*10^{12}$ events per year (EPY) should be processed
  - One trillion of reconstruction and one trillion of simulation (yep, this is BigData)
- With processing rate of one event per second per CPU - we will need to have more than 63000 fully loaded CPUs during the year
- To handle load of such level, distributed system will require to deal with any available computing resource like remote cluster, cloud infrastructure or HPC
- It’s quite hard to estimate requirements for storage resources for the moment, but even with size of event in few KB required storage will be on the level of tens of PB

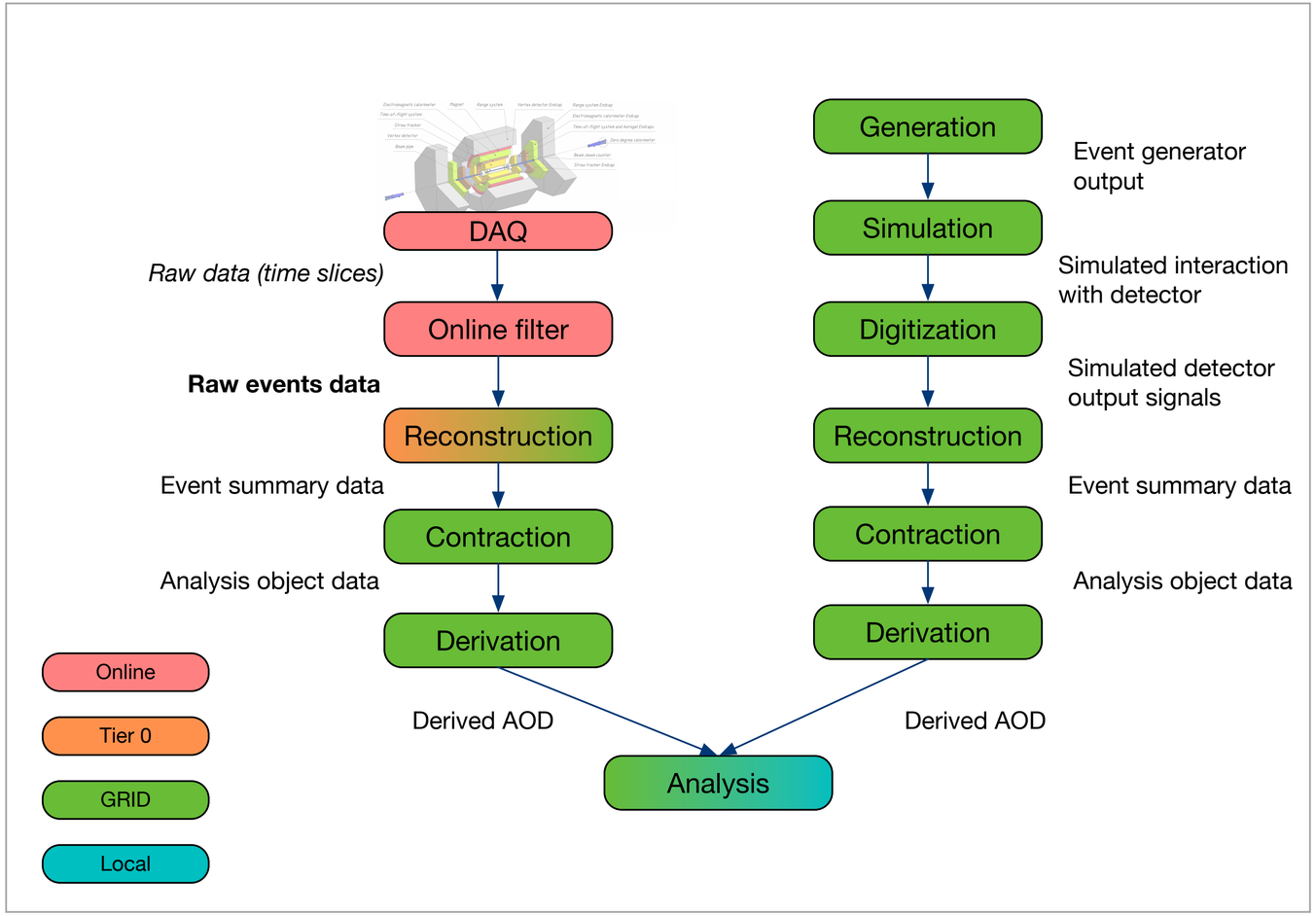
Managing of data processing
in heterogenous distributed computing system

- Key middleware components required for efficient processing in grid:
  - **Workflow management system** - control the process of processing of data on each step of processing. Produce tasks, which required for processing of certain amount of data, manages of tasks execution.
  - **Workload management system** - processes tasks execution by the splitting of the task to the small jobs, where each job process a small amount of data. Manage the distribution of jobs across the set of computing resources. Takes care about generation of a proper number of jobs till task will not be completed (or failed)
  - **Data management system** - responsible for distribution of all data across computing facilities, managing of data (storing, replicating, deleting etc.)
  - **Data transfer service**: takes care about major data transfers. Allow asynchronous bulk data transfers.



Processing steps and data types

- As reconstruction as simulation – are multistep workflows
  - Each step produces own data type, which correspond to different representation of events
  - So size of event will be different in different data type
- Why we need different types?
  - Some types of processing, like raw data, quite expensive or unique, producing of other types is resource consuming, another types good for long term storage but not optimal for final analysis because of redundancy



- Tier 0 – entry point to offline processing

Data for users analysis.

Final step

- Huge dataset of collected and simulated data are not well fit for usage of particular used due to its size
- Usually, only a set of particular properties from a subset of events is needed
  - Derivation - data reduction process which achieved by slimming, skimming and thinning procedures.
    - Slimming - subset of files form datasets
    - Skimming - subset of events from files
    - Thinning - subset of only required properties of events

Application software deployment

How to deliver software to particular site

- Software repositories acceptable, in case:
  - Software framework is not huge
  - No needs to have multiple version of software
  - No needs support of thousands simultaneous jobs
- Shared file system
  - Works good for single cluster
- CVMFS - wide shared read-oriented filesystem
- Containers to cover OS and environment differences

Scientific digital services

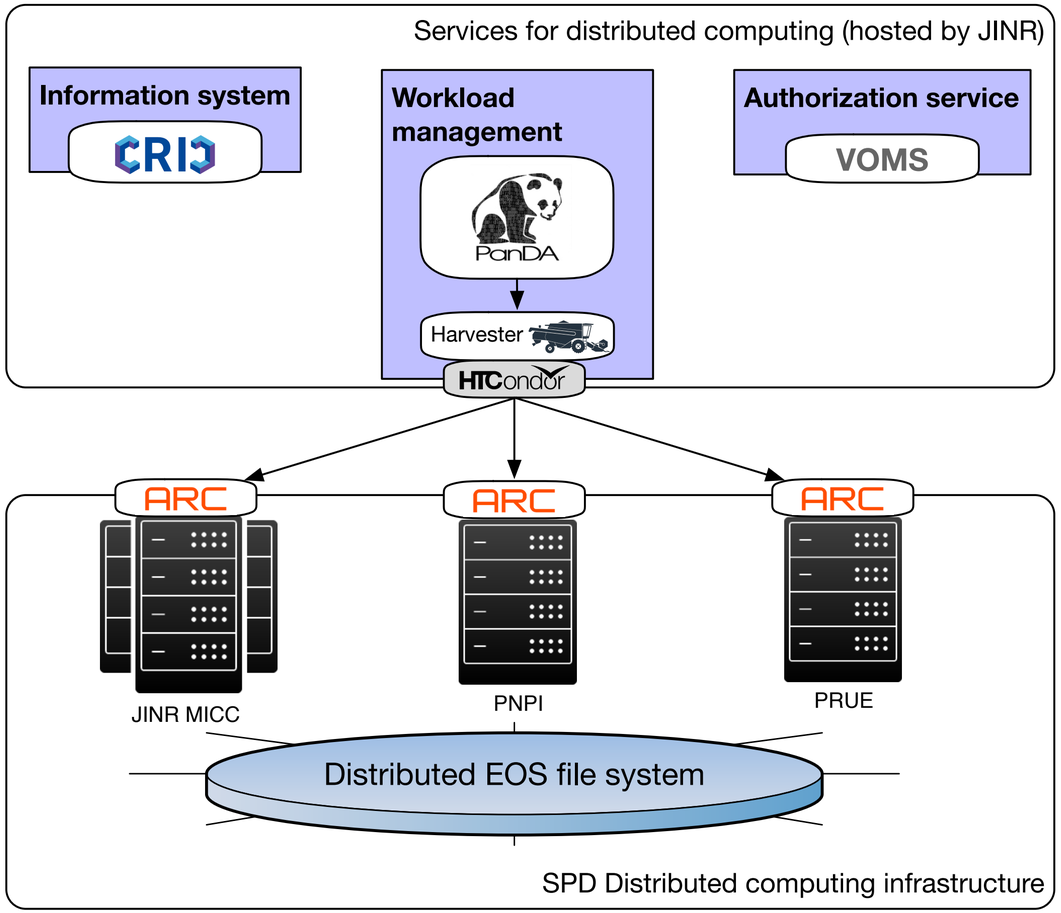
- Collaborative tools:
  - E-log - digital shifter logbook
  - Project management and issue tracking system
  - Digital libraries for managing of documentation and publication
  - Group calendars, meetings support etc…

Conclusions

- HEP Data processing requires significant efforts from IT side
- Amount and structure of HEP data may be easy qualified as BigData
  - Efficient algorithms and methods required for data processing
  - Sophisticated information systems is needed for managing of processing in distributed and high performance computing systems
- Collaborative scientific activities requires at supporting by different digital services

BackUp

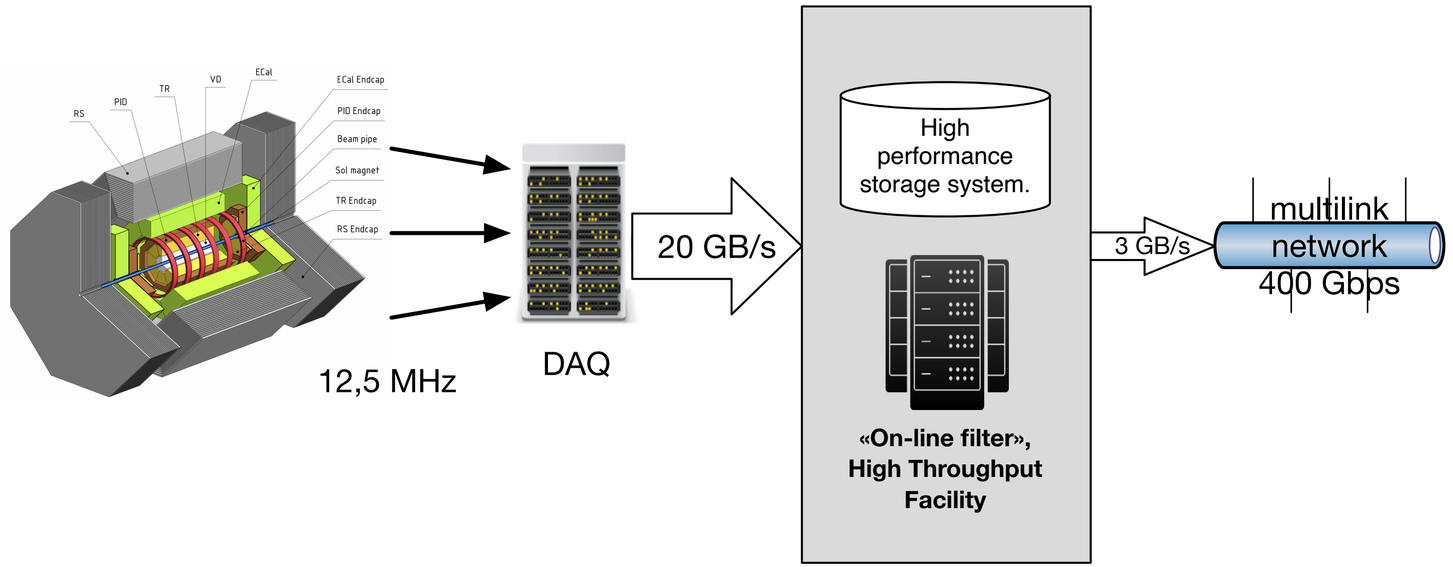
Current status of SPD offline computing



- A lot of middleware required for building of distributed computing system already exist and well supported
  - Thanks to LHC experiments
- In quite short period of time with limited manpower we were able to deploy functional prototype of the system and cover a few data processing centres
- Right now we are in process of definition of processing chains for SPD experiment, tuning and development some experiment specifc tools
- Big work foreseen for next years to move this system to full scale, but in short period of time we will be ready for processing of quite big simulation tasks

SPD as data source

On-line facility

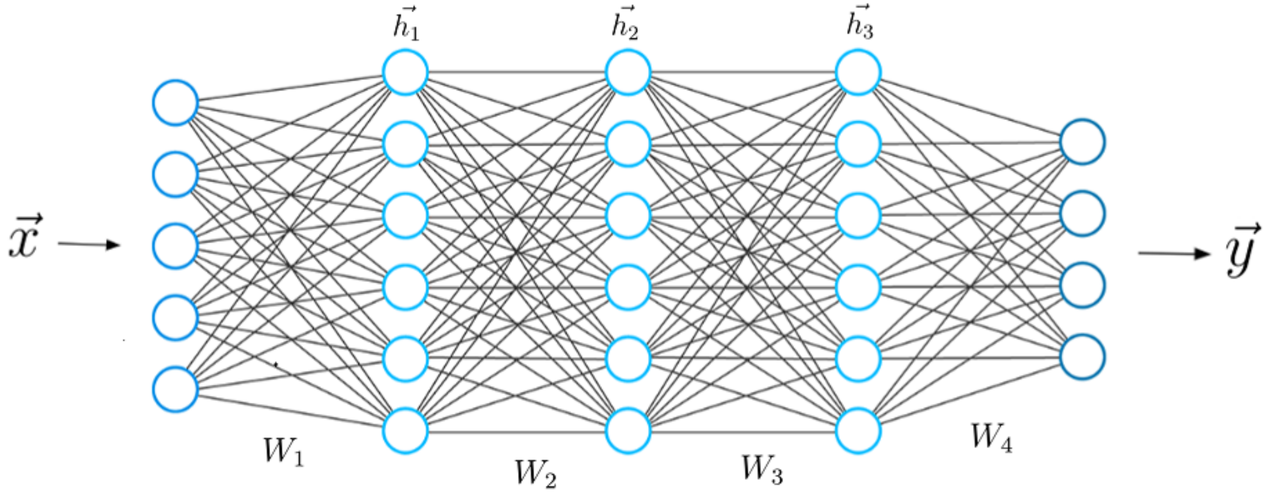


- “On-line filter” - dedicated high throughput computing facility with integrated high performance storage system for:
  - Intelligent data reduction
  - Initial data organization

On-line filter

details

- Partial reconstruction
  - Fast tracking
  - Fast ECAL clustering
- Event unscrambling
- Software trigger
  - several data streams
- Monitoring and Data quality assessment
- Local polarimetry

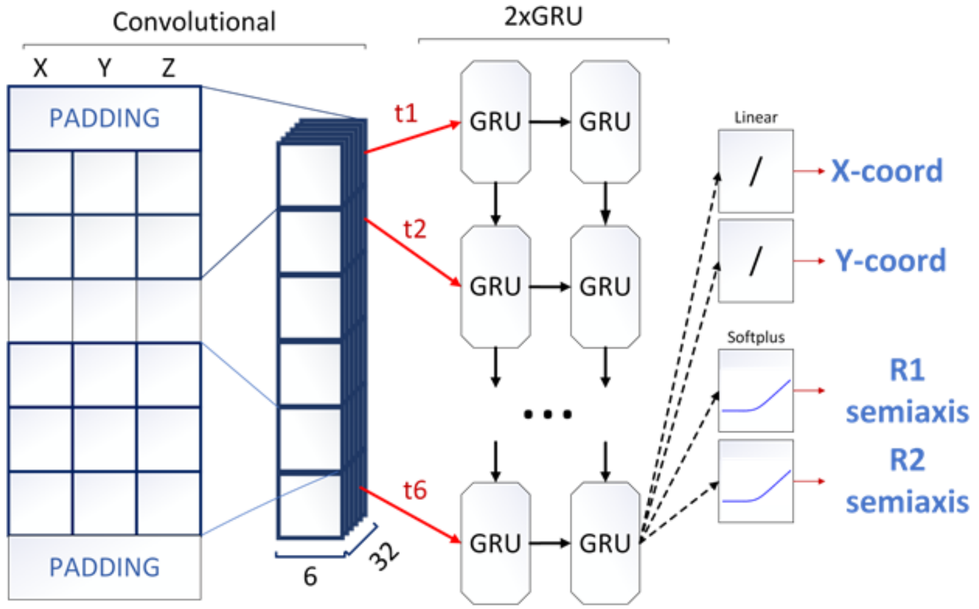


Machine learning is a promising technology

Machine learning for SPD

Under research: TrackNETv2

- works like learnable version of the Kalman filter
- for the starting part of a track predicts an elliptical area at the next station where to search for the continuation
- if there is not continuation candidate track is thrown away
- Results (Based on BM@N experiment data):
  - 12K tracks/sec on Intel Core i3-4005U @1.70 Ghz
  - 96% of tracks were reconstructed without any mistake



P.Goncharov, G. Ososkov, D. Baranov
AIP Conf 2163, 040003 (2019)

Work supported by the RFBR-NFSC project No. 19-57-53002

Distributed computing for HEP in Russia

JINR, PNPI, SPbSU, IHEP – already have experience of supporting own Data Processing Centers and participation in the distributed computing for LHC

- *JINR – Data Center for LHC with ~23000 CPU and 25PB Disk storage and 55PB Tape storage*
- *PNPI – Data Center for own experiments and LHC with ~15000 CPU and 5Pb Disk storage*
- *SPbSU – Tier2 ALICE Computing facility*
- *IHEP – Tier2 WLCG site*