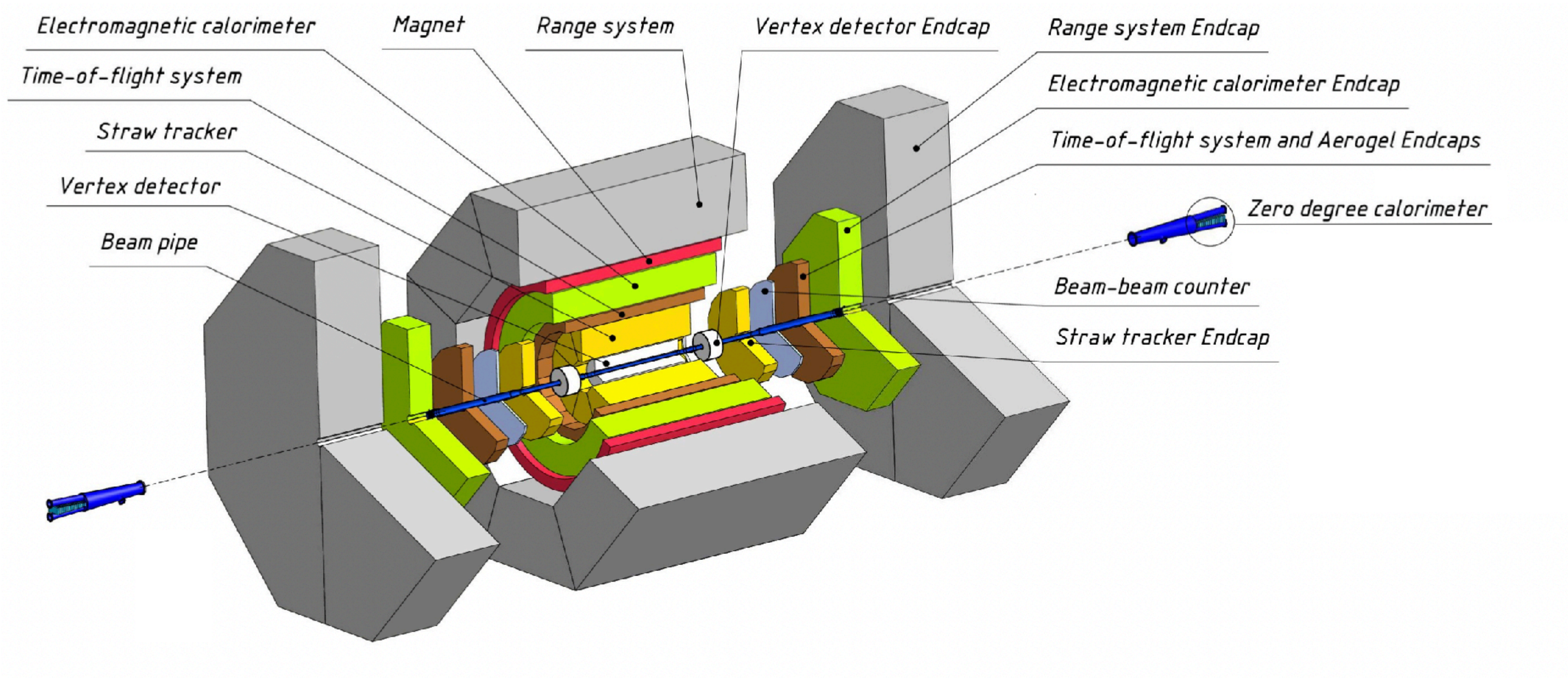
SPD Online filter

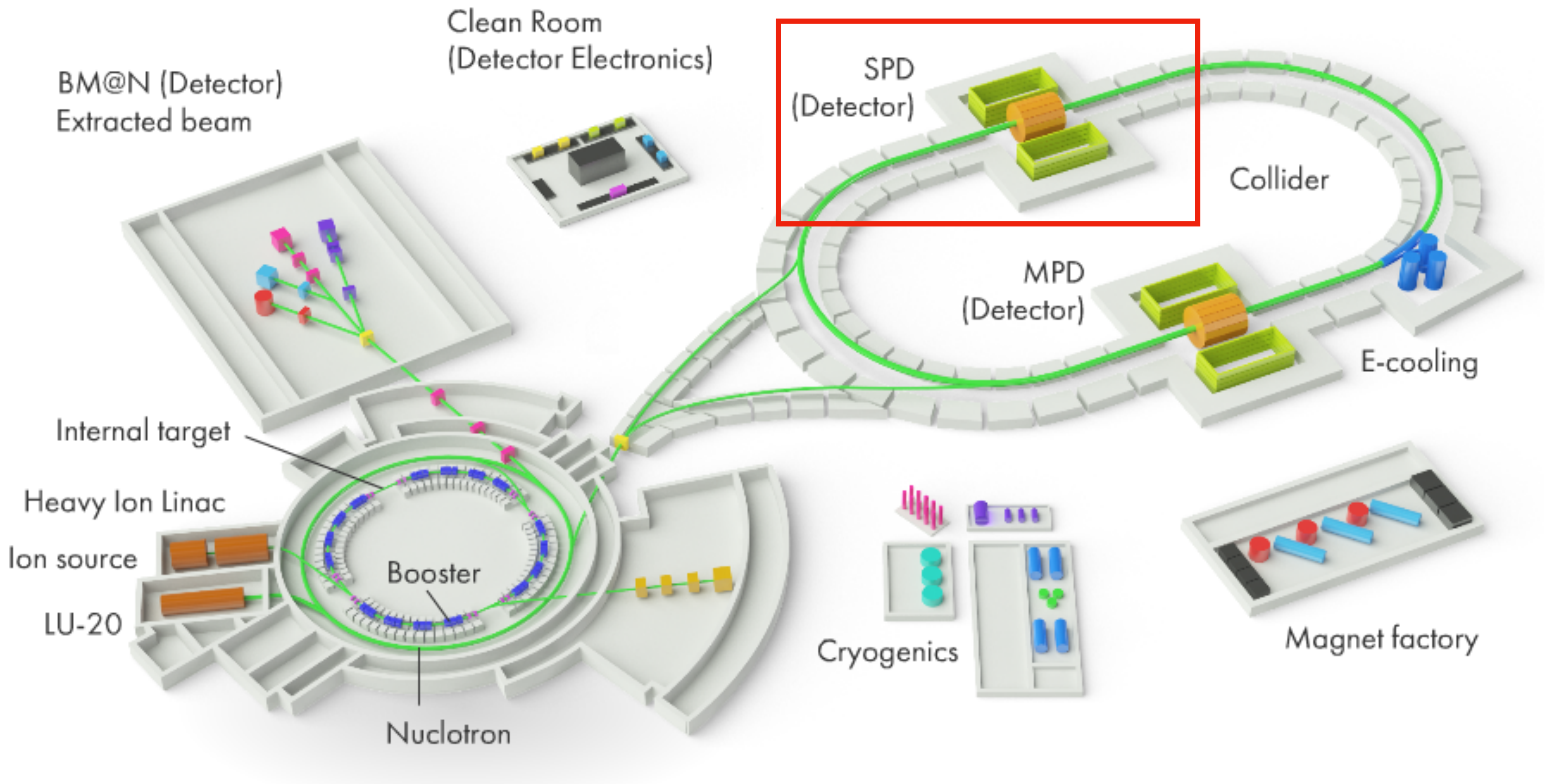
Система первичной обработки данных эксперимента SPD

Danila Oleynik, 20.12.2023 JINR MLIT

SPD Spin Physics Detector

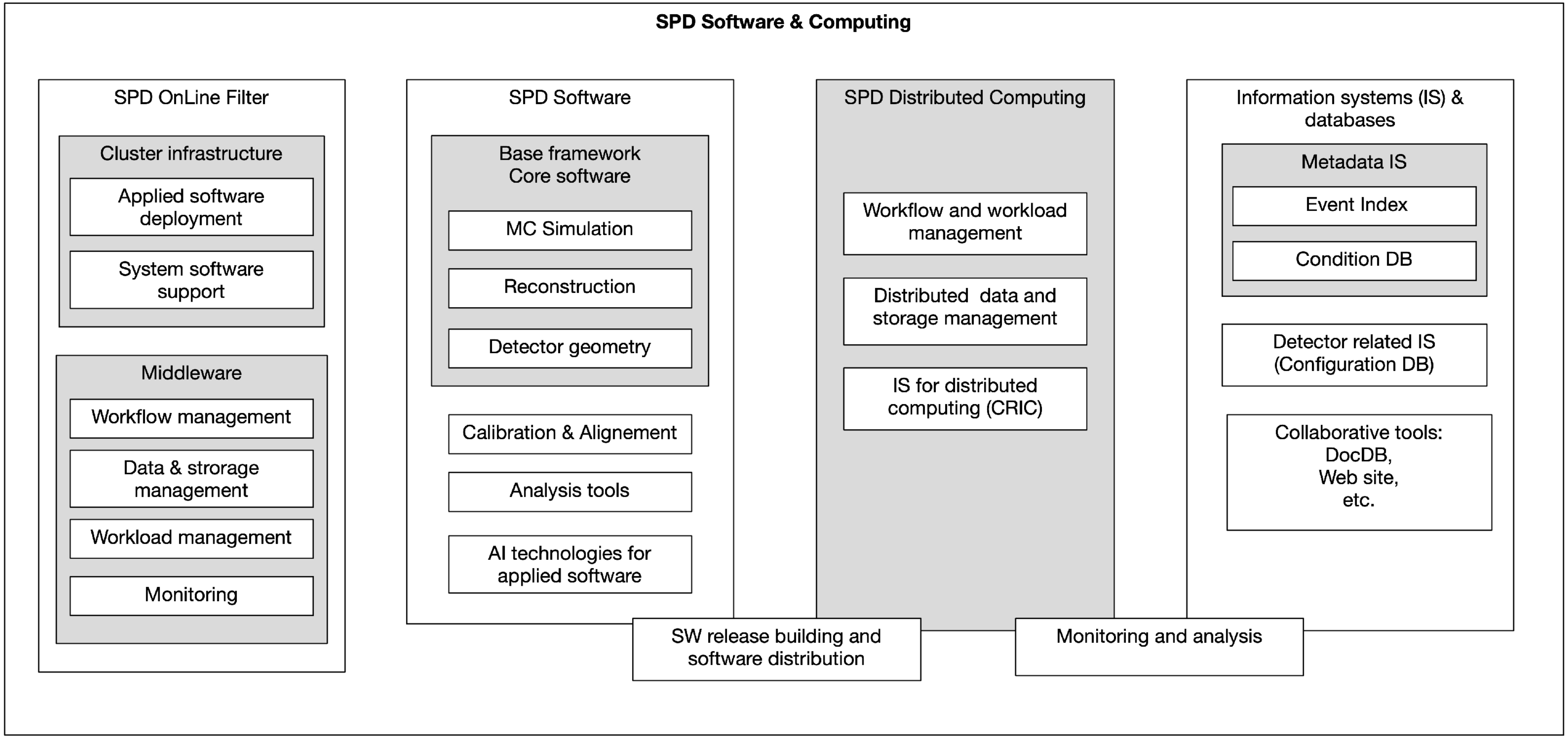
Study of the nucleon spin structure and spin-related phenomena in polarized *p-p*, *d-d* and *p-d* collisions





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

SPD Software & Computing

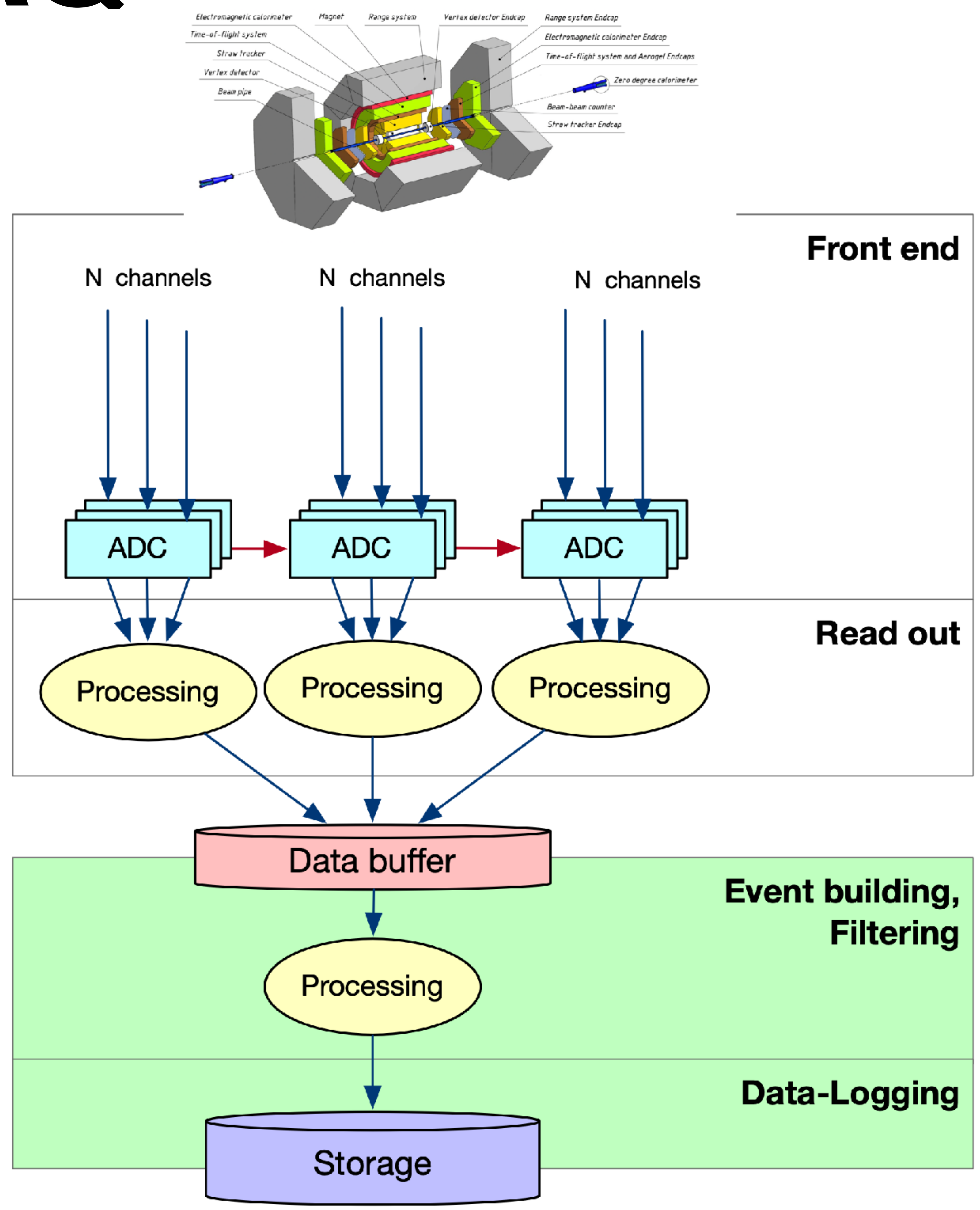


SPD detector as data source

- 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

SPD Free-running DAQ

- With a free-running DAQ, there is no hardware trigger system, therefore the readout process is not controlled by a trigger anymore. All data that exceed the thresholds in the front-end electronics are read out together with their timing marks. Then in DAQ the data are grouped on the basis of timing and detector affiliation
- To get data in proper format for future processing (reconstruction) and filtering of ‘boring’ events and other initial processing processing a special computing facility named “SPD Online Filter" in progress



Free-run SPD DAQ

20 GB*sec ~(3 MHz event rate)

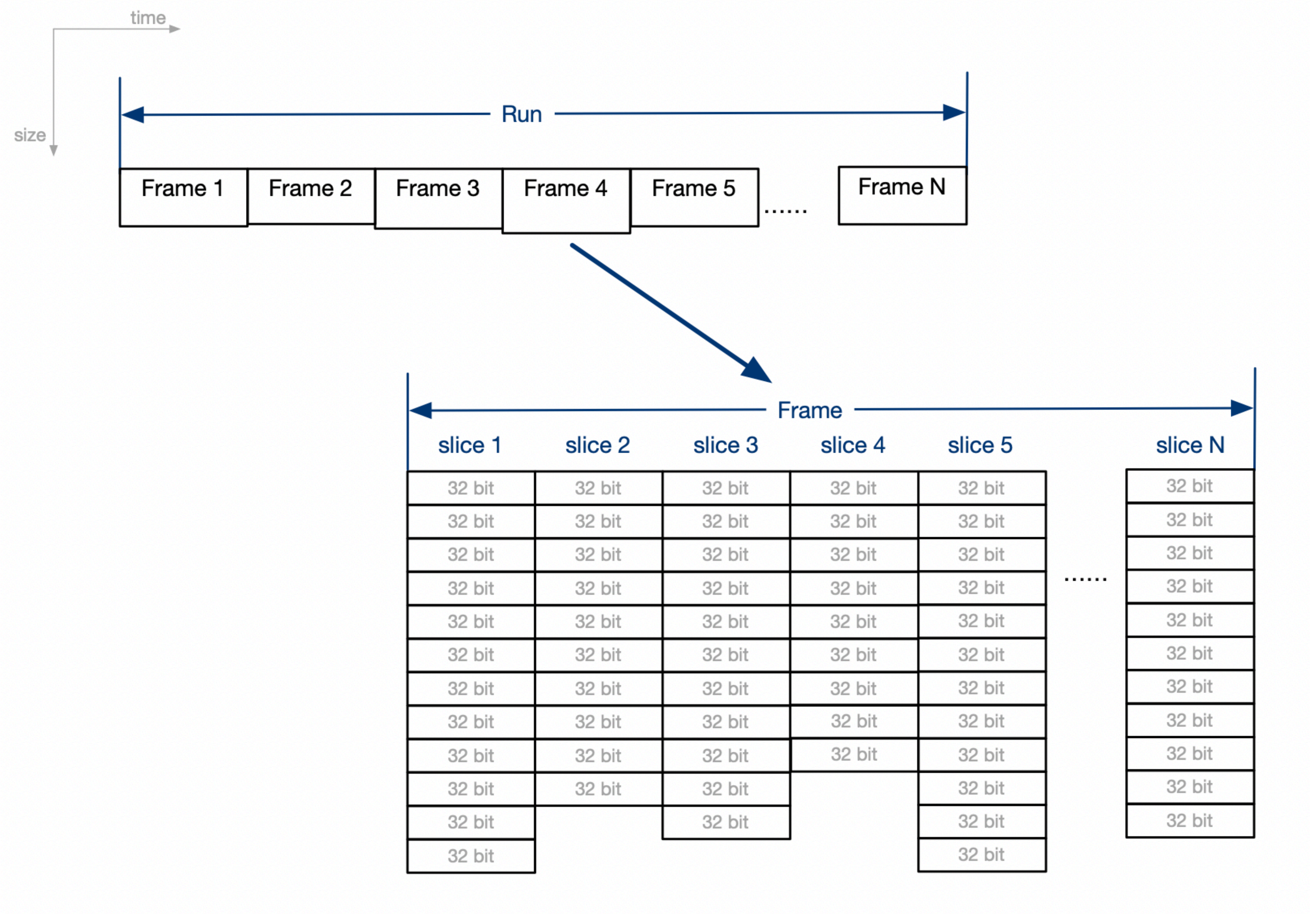
SPD On-line Filter

150 kHz event rate

Collected data

data retrieved from DAQ

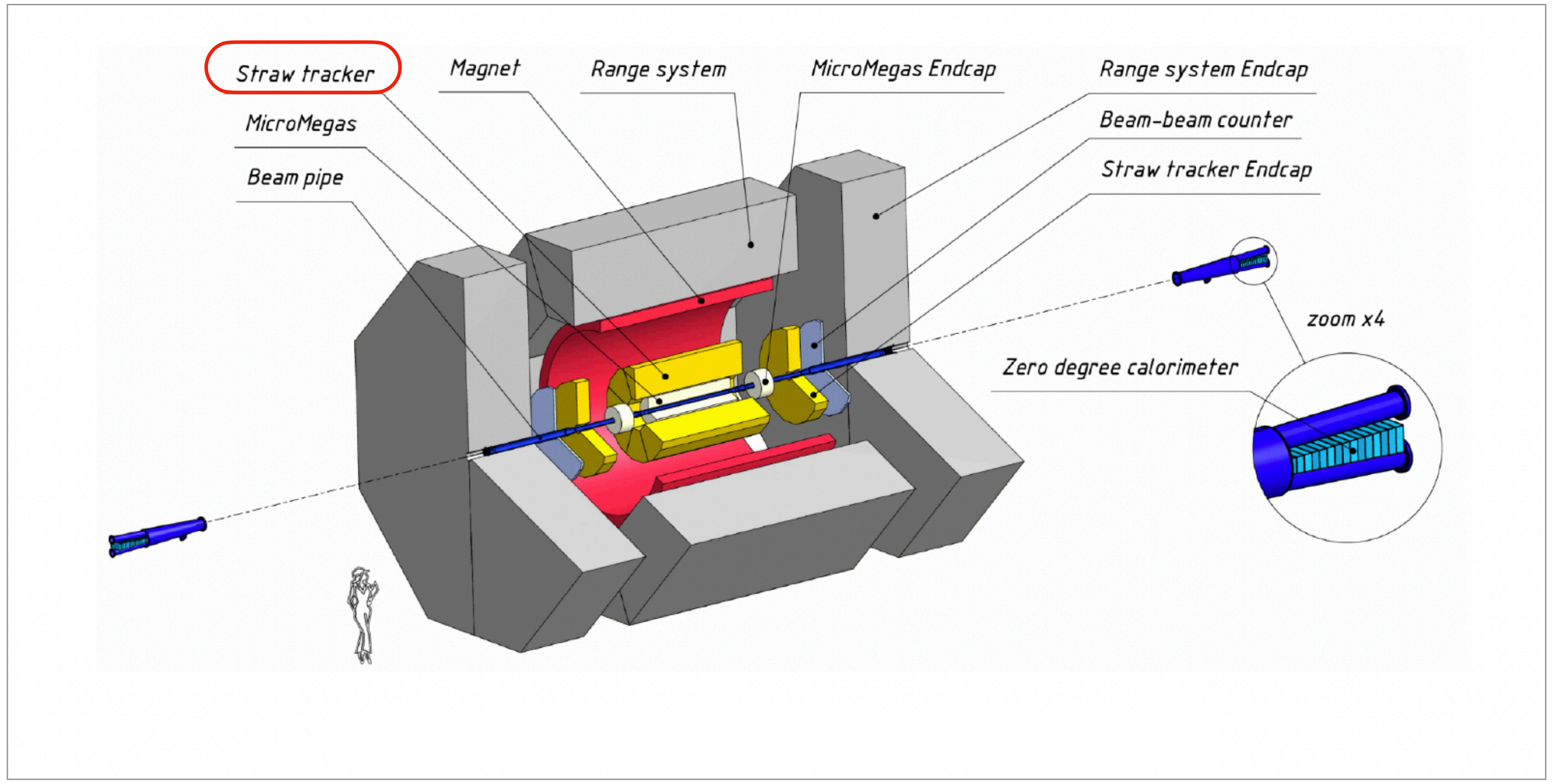
- DAQ provide data organized in time frames and sliced to files with reasonable size (a few GB)
- Each of these file may be processed independently as a part of top-level workflow chain
- No needs to exchange of any information during handling of each initial file, but results of may be used as input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

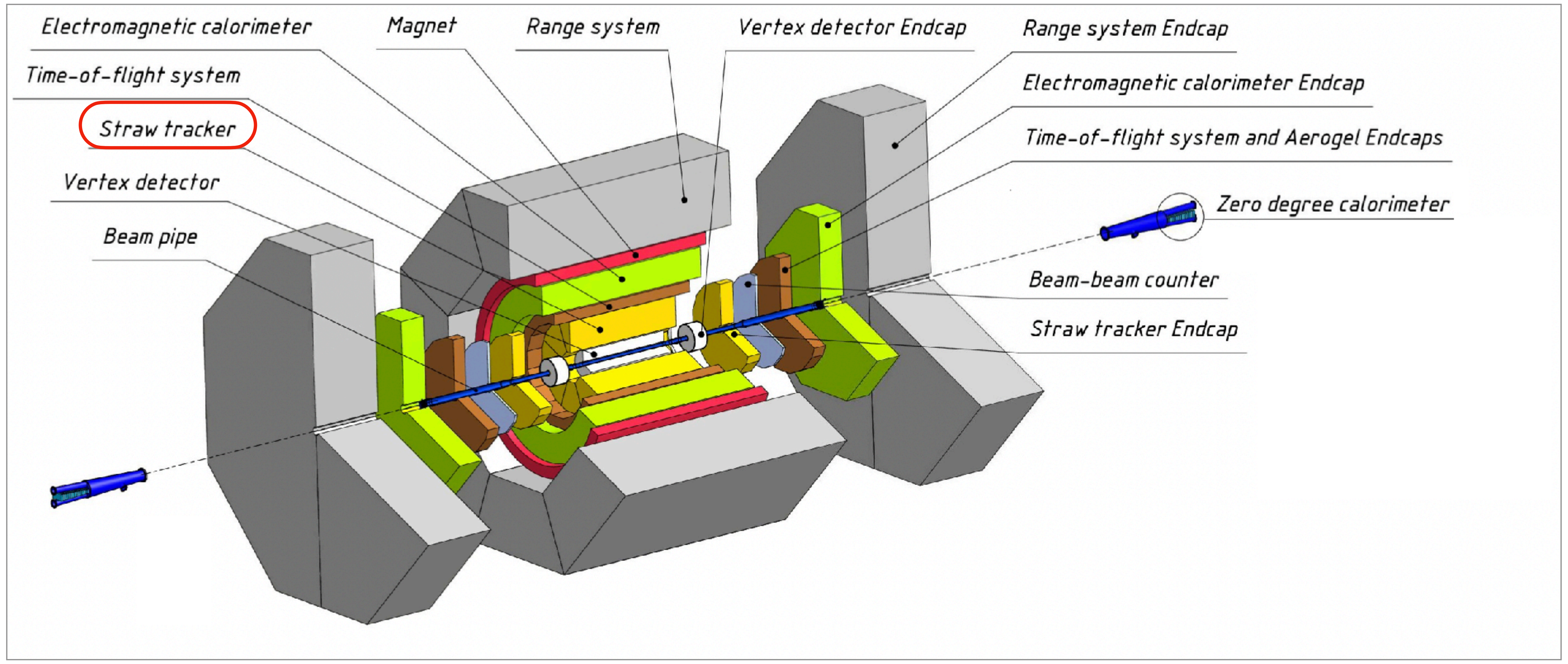


Event unscrambling

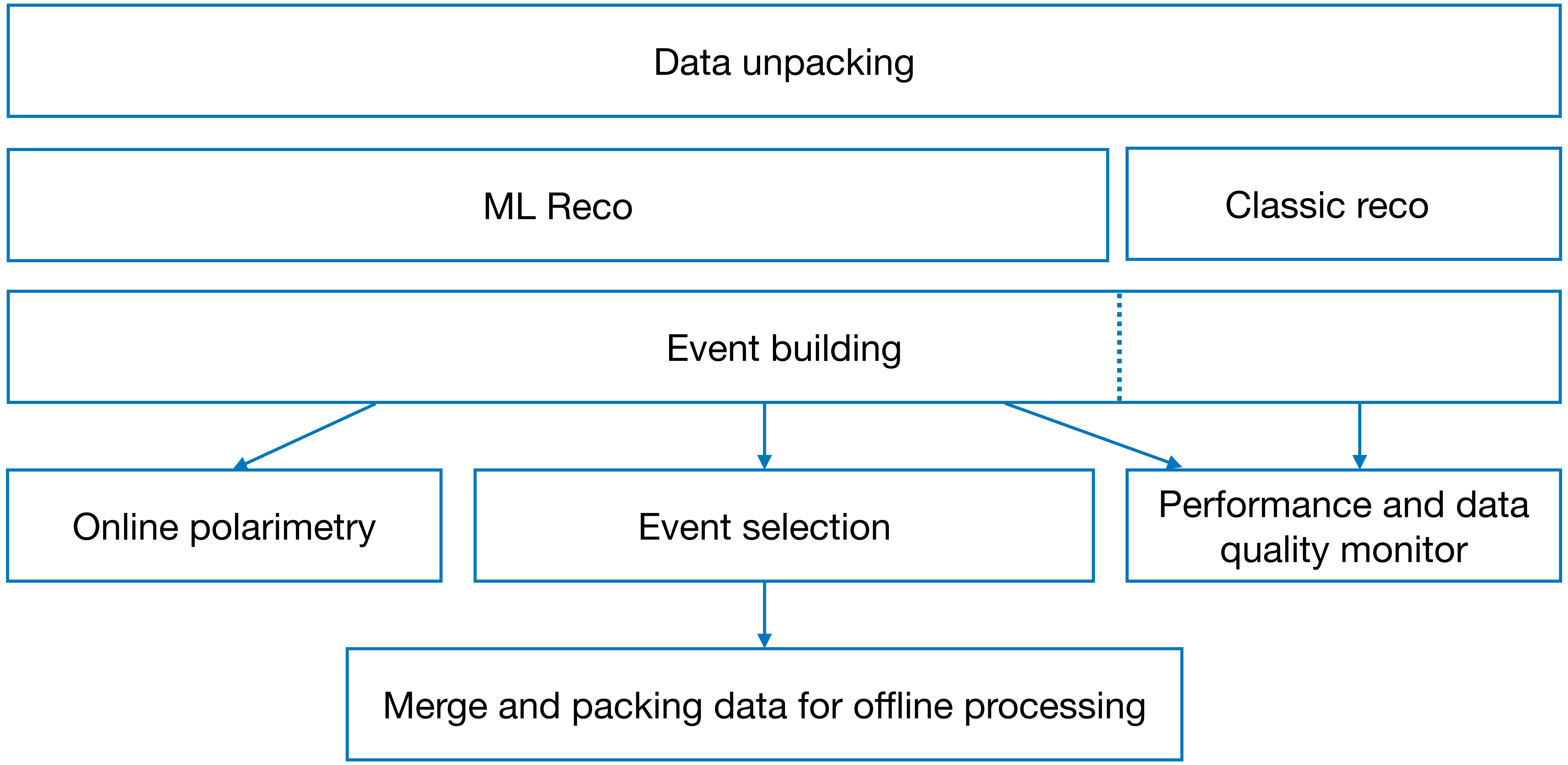
For each time slice

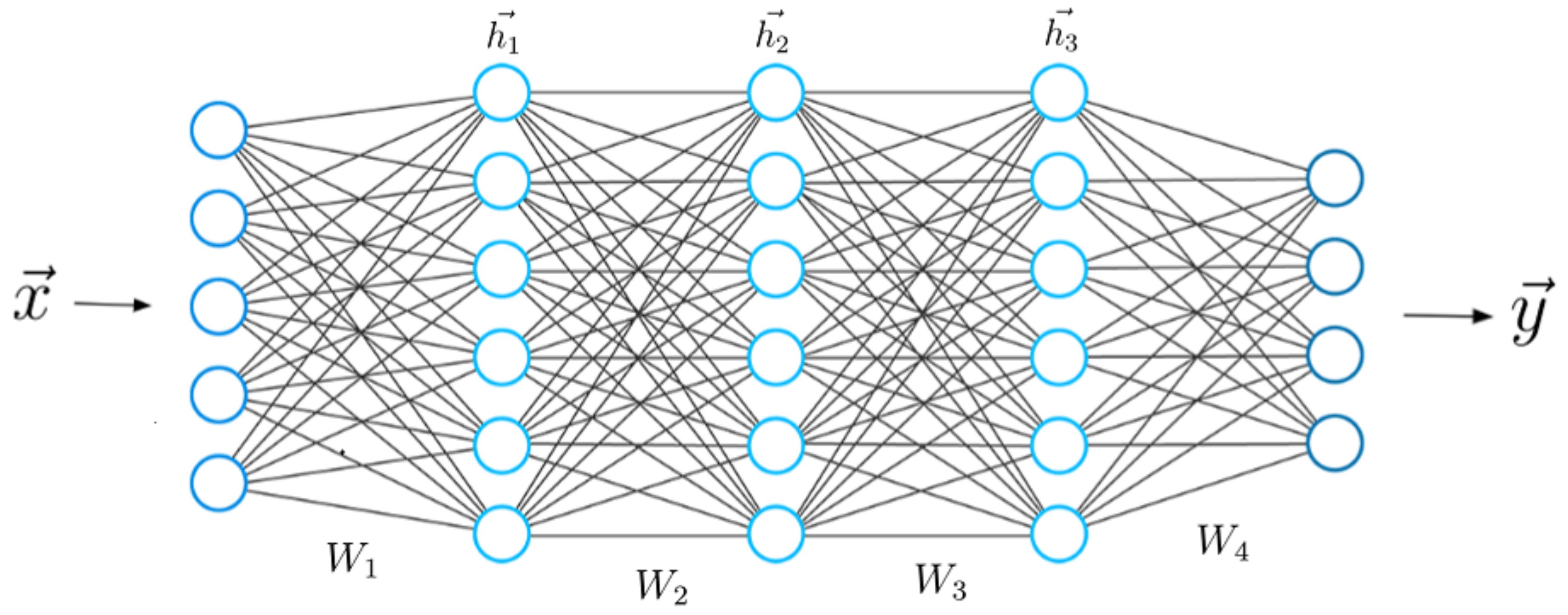
- Reconstruct tracks and associate them with vertices
- Determine bunch crossing time for each vertex
- Associate ECAL and RS hits with each vertex (by timestamp)
- Attach unassociated tracker hits in a selected time window according to bunch crossing time
- Attach raw data from other subdetectors according to bunch crossing time
- Name the block of information associated with each vertex an event
- Store reconstructed events





Payload

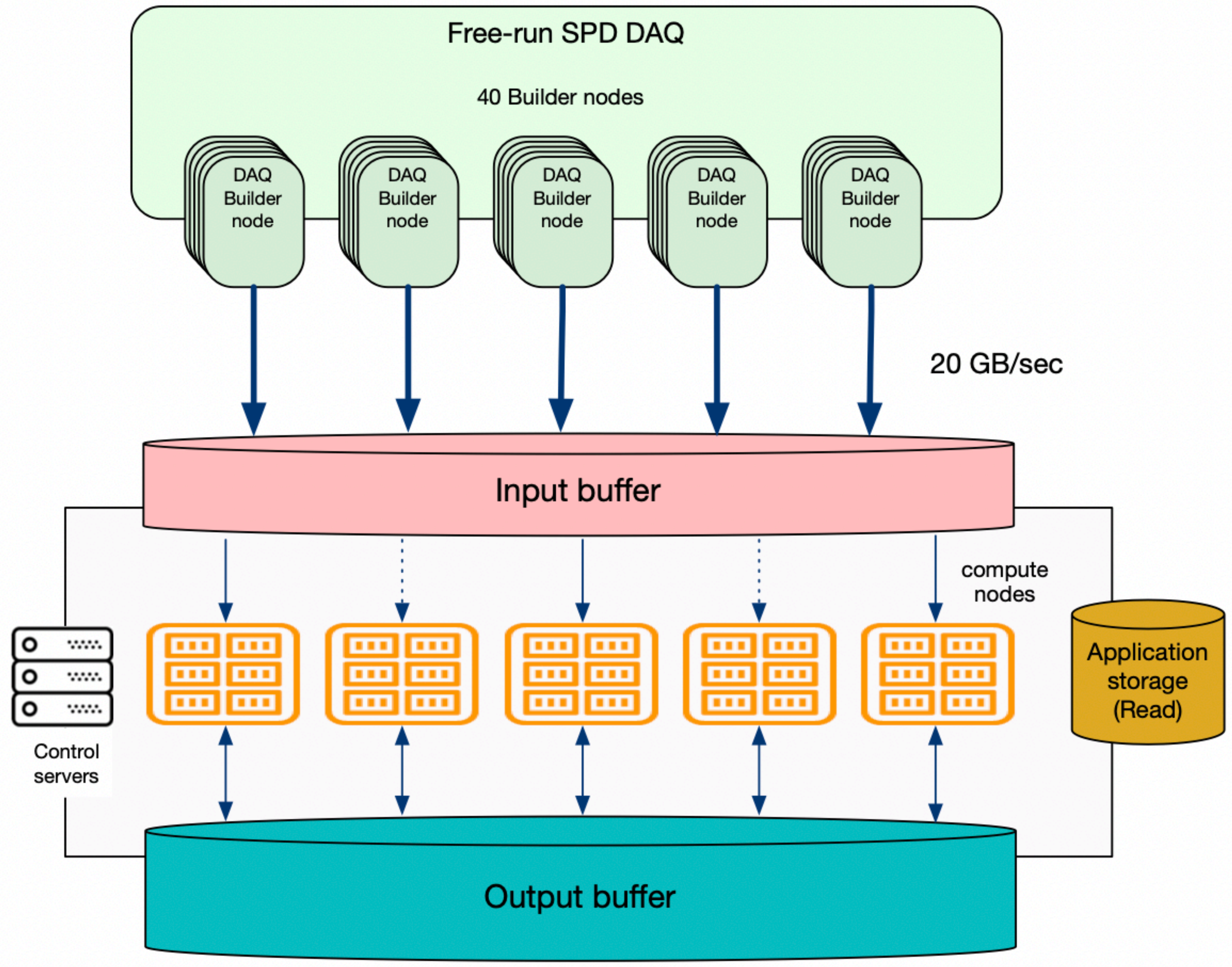




Machine learning is a promising technology

SPD Online filter infrastructure

- SPD Online Filter is a high performance computing system for high throughput processing
  - 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 middleware which will manage 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
Software part

- **Middleware** - software complex for management of multistep data processing and efficient loading (usage) of computing facility
  - Workflow management
  - Data management
  - Workload management
- **Applied software** - performs actual data processing
  - Framework - responsible for unified algorithm interfaces, IO, multithreading etc.
  - Algorithms - responsible for a single pieces of processing

Middleware

Current status

- Each subsystem were engineered and partially prototyped
- Microservice architecture with domain driven design was chosen
  - Flexibility, scalability, easy for long-term support
- Data management
  - dsm-register – responsible for registration of input data from DAQ in the catalogue
  - dsm-manager – realise interfaces to the catalogue for subsystems
  - dsm-inspector – realise auxiliary tools for storage management (consistency check, cleanup, dark data identification)

Middleware

Current status 2

- Workflow management
  - “Chain definer” - user oriented application which allow define sequences of processing steps
  - “Processing starter” - microservice responsible for triggering of processing chains
  - “Chain executor” - microservice responsible for control of execution of processing chain
- Workload management
  - Realize a task execution process by shredding a required number of jobs to provide controlled loading to compute facility, tacking into account priority of tasks and associated jobs. A task is one step in a processing chain of a block of data. Job is a processing of a single piece of data (file or few files).
  - Microservices: task manager, task executor, job manager, job executor
  - Base architecture and initial functionality of pilot application is defined. It is a multithread application with interactions between threads through queues

SPD Online filter middleware

next steps

- Manpower: two PHD students, one full time researcher
  - A couple of master students recently joined
- Development, testing, integration infrastructure
- Deployment procedures etc.
- Integration with applied software (framework)

What else?

in middleware

- **Monitoring**
- Sources of auxiliary data: mapping, geometry, etc.
- Applied software repository management
- Middleware deployment machinery

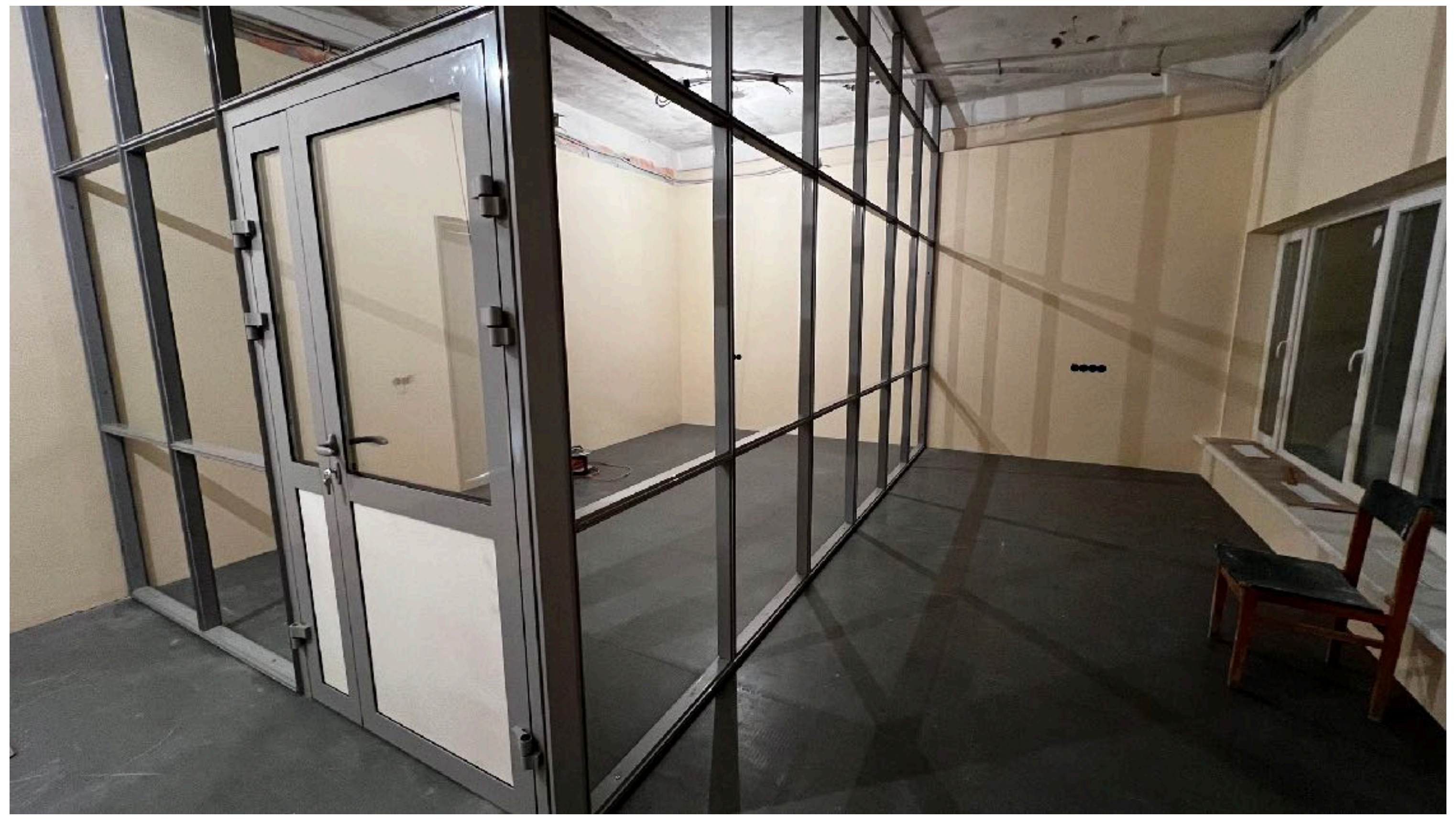
Debugging requirements

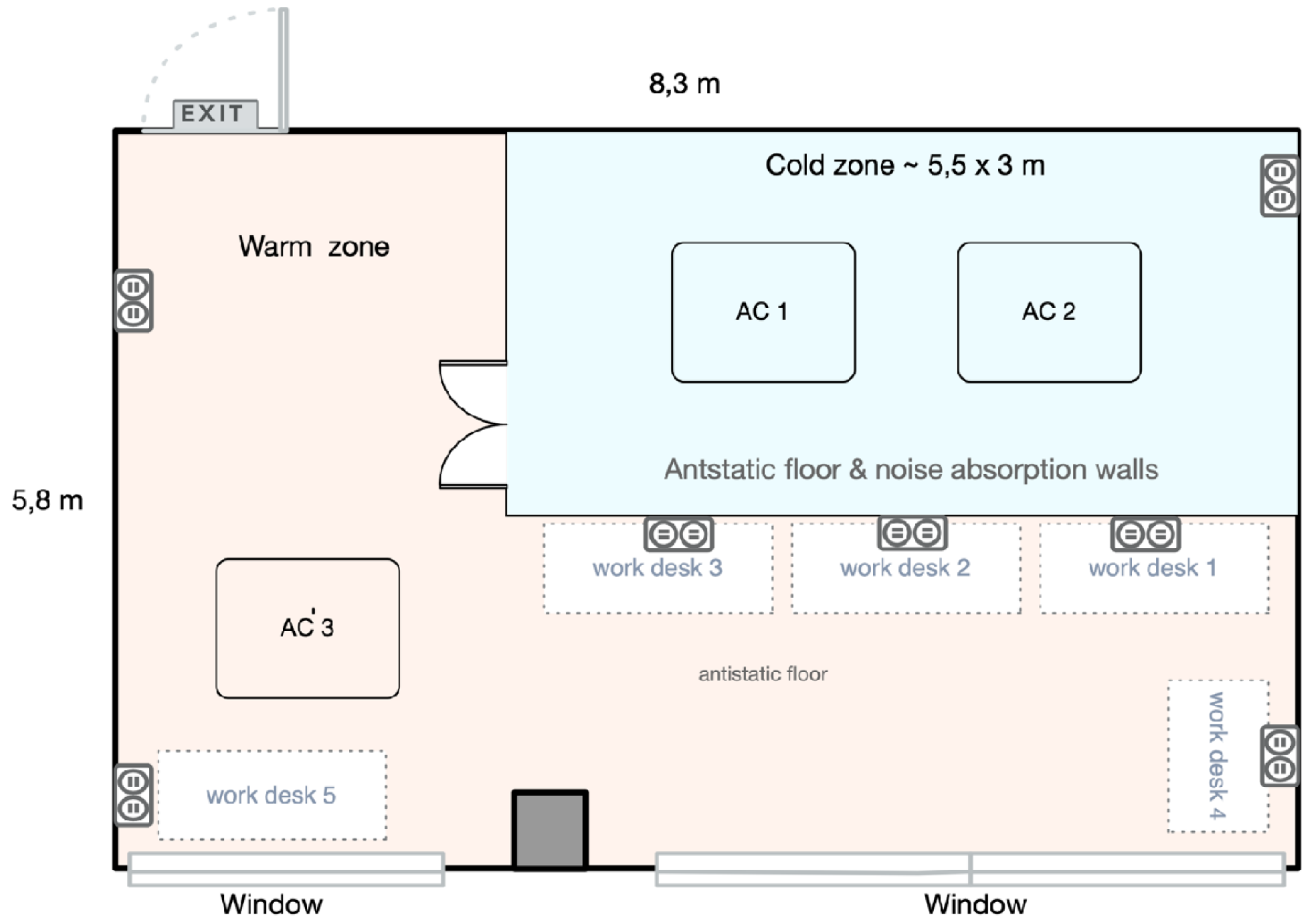
- Initial testing:
    - Agreed interfaces and data formats
    - Simplified simulated data: properly packed “white noise”
    - Low amount of data (<<0,1% of expected average)
- Functional testing:
    - Simulated data partially close to real data, which will allows debugging of some algorithms, and some workflows
    - Data amount (0,1 - 1% of expected average)
- Pre-production testing:
    - Simulated data of whole systems
    - Data amount (1 - 10% of expected average)

Debugging workflow

- Offline system:
  - MC production with incremental growth of simulated data
  - Agreed data organisation (to allow different types of debugging)
  - Physics group: algorithms and data production control
- Step by step improvement of Online filter prototype
  - Estimation of required set of services: software distribution and deployment
  - Subsidiary data sources: mapping, geometry etc.
  - VM to real HW (on small scale)

DAQ SOF testbed





General plan for next 6 months

- OnLine filter
  - More attention to framework and reco. algorithms: simulated data is needed
    - MC production workflow in offline system:
      - Agreed data model and data organization
      - Data management system in place
      - MC data production policies
  - Definition and implementation of obvious data processing pipelines
- Running up of the SOF-DAQ testbed