DDP Computing Project

Project overview

46th CBM Collaboration Meeting | 22.10.2025

Bartosz Soból

DDP - Distributed Data Processing

- New CBM Computing Project
- Filling gaps between other projects involved in **Online Processing (OP)**

Main focus areas:

- Orchestration, deployment and control interface for OP pipeline on HPC
- Efficient utilization of compute resources
- Intra- and inter-node communication for OP pipeline stages on HPC

DDP - Distributed Data Processing

Current (mCBM) solutions (examples)

- **Deployment -** Set of custom shell script for managing slurm jobs and configuration
- **Control, fail recovery -** Limited custom solutions, scripts
- **OP pipeline -** Monolithic executable, partially parallelized

They serve their purposes in prototype setup, but

- Have limited capabilities (features)
- Are not scalable
- Not structured, hard to maintain

Connections with other projects

ODM (Online Data Management)

- Interfacing on the timeslice-forwarding level

DPF (Data Processing Framework)

- Efficient resource utilization
- Communication between processing stages and elements within cluster

EDC (Experiment and Detector Control)

- Integration of the cluster deployment/control system with EDC
    - SCA (sub-system control agent) implementation

DDP in context

DDP

- Deployment on Virgo
  - OP pipeline, timeslice receiver
  - Resource-efficient topology
  - Interface to cluster management
- Connecting: OP stages, QA, …
  - Transport layer, buffering
- SCA implementation
  - Expose OP health status (nodes, processes)
  - Interface for EDC to take actions

ODM

- FLES operation
- Timeslice building
- Timeslice management
- …

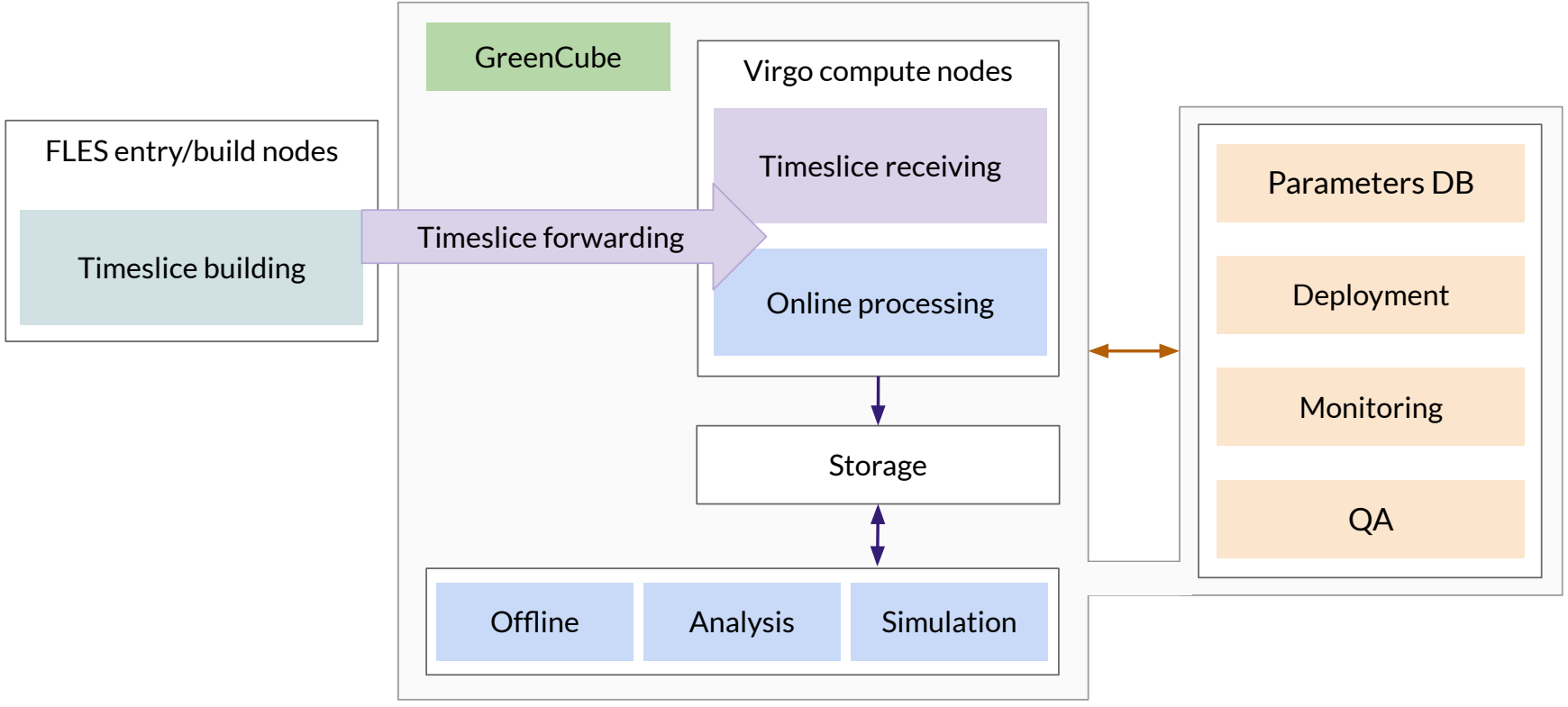
DPF

- Implementation of OP pipeline stages
- Timeslice processing
- Parameters database
- …

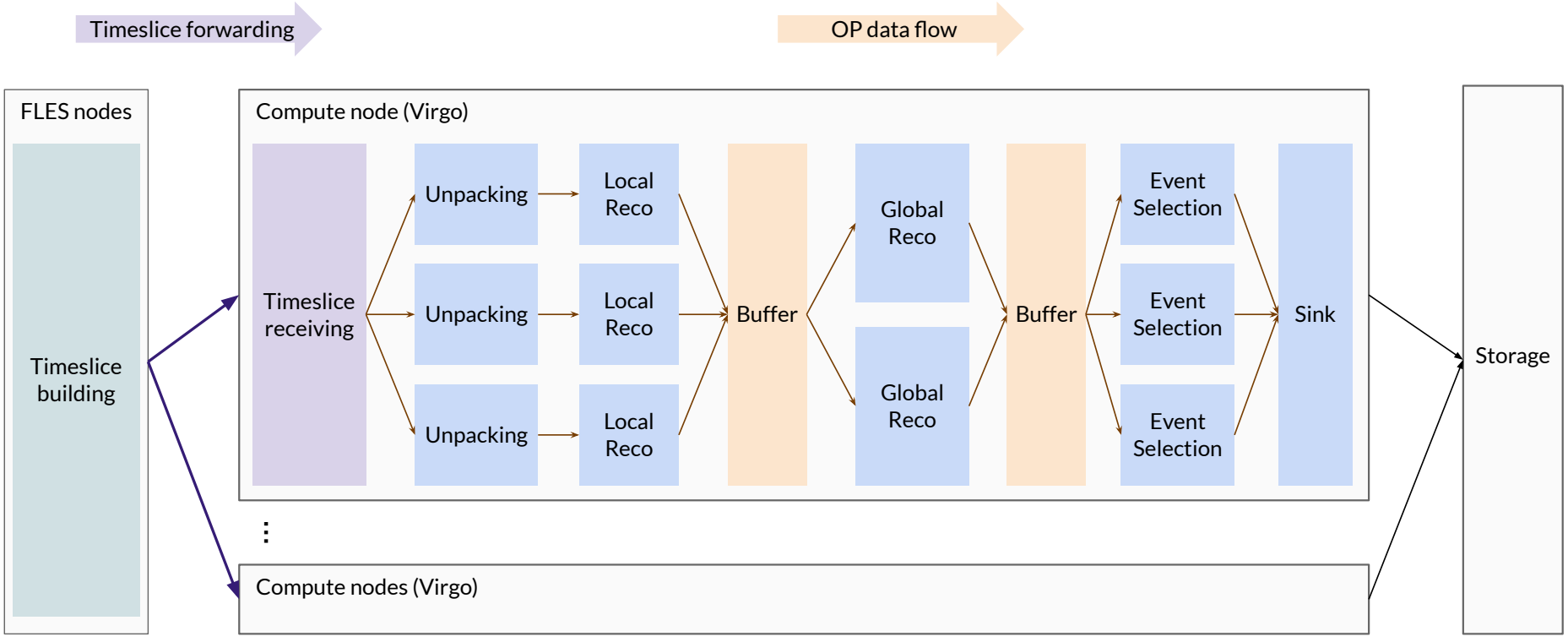
EDC

- Run controls (detector subsystems and OP)
- Recovery actions (node fail, software crash)
- …

CBM online setup

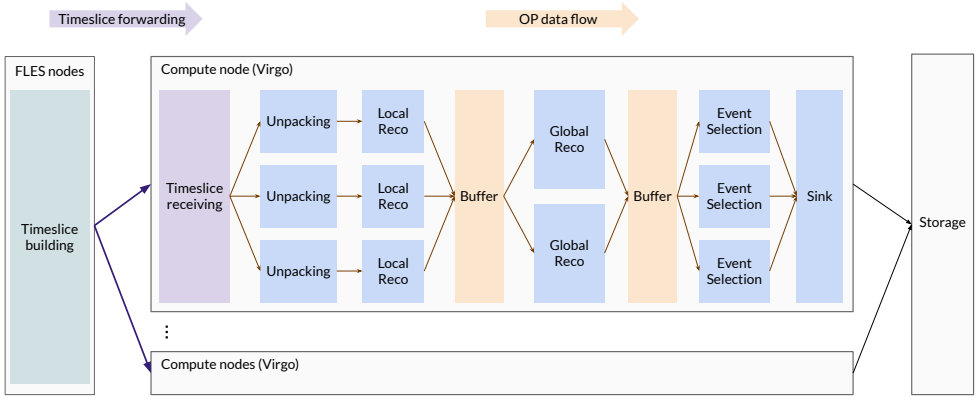


Main online data flow (concept)



Main online data flow (concept)

- Online binary split into stages and/or subsystem dedicated tasks
- Potential buffering between stages
- Many benchmarks required to define the final configuration
  - Efficient task separation
  - Communication scheme between tasks
  - Depends on final algorithms implementations, which are W.I.P
- Connection to ECS, QA, …



Cooperation with GSI IT and SDE group

- GSI IT operates the Virgo cluster
  - CBM can use a subset of Virgo hardware during beamtime
  - Separate cluster environment, same setup as Virgo
- Planned usage of solutions developed at GSI for community (FairMQ, ODC, DDS)
  - SDE is dedicated to further support and develop the toolset
  - Adaptations and new features for CBM needs
  - Potential introduction of other (standard) solutions

Initial activities (DDP + SDE)

- Adaptation of OP elements to work as FairMQ devices
  - **cbmreco -** monolithic OP pipeline executable
  - **tsclient -** current version of timeslice-sender/receiver
- Deployment of mCBM online setup using ODC + DDS
- Preparations of slurm-enabled OP Apptainer container based on VAE
- *Virtual* Slurm *cluster* setup deployable with Docker Compose for development and testing
  - Self-contained testing environment

Detailed project description in prep.