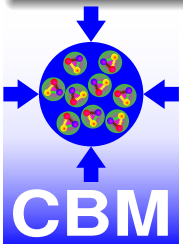
CBM Reconstruction Module



Sergei Zharko

CBM Department
GSI Helmholtzzentrum für Schwerionenforschung
Darmstadt, Germany

December 11, 2025

Introduction

- Parameters and configuration for cbmreco (currently in master):
  - readout: individual yaml-files
  - local reconstruction: individual yaml-files, configuration in MainConfig.yaml (STS)
  - tracking: binary file ca.par, reco-setup binary file, configuration in MainConfig.yaml
  - event-builder (trigger, selector): configuration in MainConfig.yaml, reco-setup binary file
  - **true parameters:** MainConfig.yaml, readout
  - **derived parameters:** ca.par, reco-setup, local reconstruction yaml-files
- Parameters for offline reconstruction:
  - ROOT-geometry
  - diverse digitization parameters (derived from `FairGenericParSet`, registered in `FairRuntimeDb`)
- **There is no proper service to produce the derived parameters**
  - the parameter service cbm-online-par-dump: mCBM-beamtime oriented

Outline

- MainConfig.yaml
- Service cbm-reco-par-dump
- Parameters for offline mCBM
- Parameters for VT25
- Missing features and development prospects

MainConfig.yaml

- Paths to parameter files, settings for different algorithms
- Supposed to be modified by a user
- The paths to reconstruction parameters are updated with the `cbm-reco-par-dump` (see next slide)

Service cbm-reco-par-dump

- Generates parameter files for reconstruction algorithms on-the-fly from ROOT-geometry and digitization parameters
- mCBM-beamtime oriented:
  - requires runId and its registration in CbmRunDatabase.yaml
  - calls hard-coded setters depending on the runId (e.g., STS, TOF)

- Parameters generation for offline mCBM:
  - command:[1]

    ```
    $ cbm-online-par-dump -r <runId> -o <outDir> -g <geoDir>
    ```

  - generates reconstruction parameters for a given `<recoParTag>` (see CbmRunDatabase.yaml)

- Parameters generation for online processing:
  - command:

    ```
    $ cbm-online-par-dump -r <runId> -o <output directory> -g <directory with .geo.root> --experimental
    ```

  - copies the contents of `parameters/online/recoPar/<recoParTag>` in the `<outputDir>` (`MainConfig.yaml`, readout)
  - generates the reconstruction parameters (hitfinder parameters, CA-parameters, reco-setup)
  - updates the paths of the generated parameters in the `<outputDir>/MainConfig.yaml` (the node `parFiles:`)
  - see CI-test OnlineRecoParDump

[1] `<outDir>` is a directory with pre-generated `*.geo.root` file

Parameters for offline mCBM

Issue #3826

A. TSA files unpacking, digi-timeslices as the output
   - done with a help of `reco_mcbm.sh`
   - requires a proper definition of readout parameters, accessible in `parameters/online/recoPar/<recoParTag>` (see CbmRunDatabase.yaml)

B. TSA files unpacking and event building, digi-events as the output
   - **experimental usage, not included in the** `reco_mcbm.sh`
   - requires a reco-setup input
   - the parameters have to be generated in the same way as for online processing by hand and copied in `parameters/online/recoPar/<recoParTag>` (see previous slide)

Parameters for VT25

Issue #3595

- Input: TSA, produced by the `vtfeeder` service from a `*.digi.root` file
- Output: digi events
- Requires **arbitrary** valid readout parameters
- Requires `MainConfig.yaml` (the nodes `trigger`, `eventbuilder`, `selector`)
- **Usage diverges between the master and jul25_patches branches**:
  - **jul25_patches**: uses mCBM definition of STS and TOF station, requires TofHitfinderPar.yaml to define a TOF station (can be ignored with no selection based on a minimal number of layers)
  - **master**: requires a valid `cbm::RecoSetup` binary (must be provided)

Missing features and development prospects

- **Target: VT26**
- A new parameters service is needed:
  - no mCBM specific options
  - **a full list of input parameter files**: geometry, readout, digitization, calibration, alignment...
  - **a proper format of the "hand-written" parameters**: YAML, no magic numbers (e.g., `iCalSet` for TOF)
- What is to be done with the `FairRuntimeDb`?
  - widely used in simulations
  - required to generate reconstruction parameters
  - uses either ROOT, or ASCII input (**supported not by all detectors, e.g. STS**)
  - issues with loading a valid set of parameters, which were created in simulations (VT25 experience)
- mCBM hard-coded features must be removed from the codebase (CbmRoot) **entirely** and replaced with a proper configuration files