

Online Reconstruction

(Prompt Offline Reconstruction at P2)

C. Cheshkov
on behalf of

L. Betev, S. Chapeland, J.F. Grosse-Oetringhaus, R.
Grosso, F. Roukoutakis, M. Tadel and C. Zampolli

ALICE Offline Week
7-11 April 2008

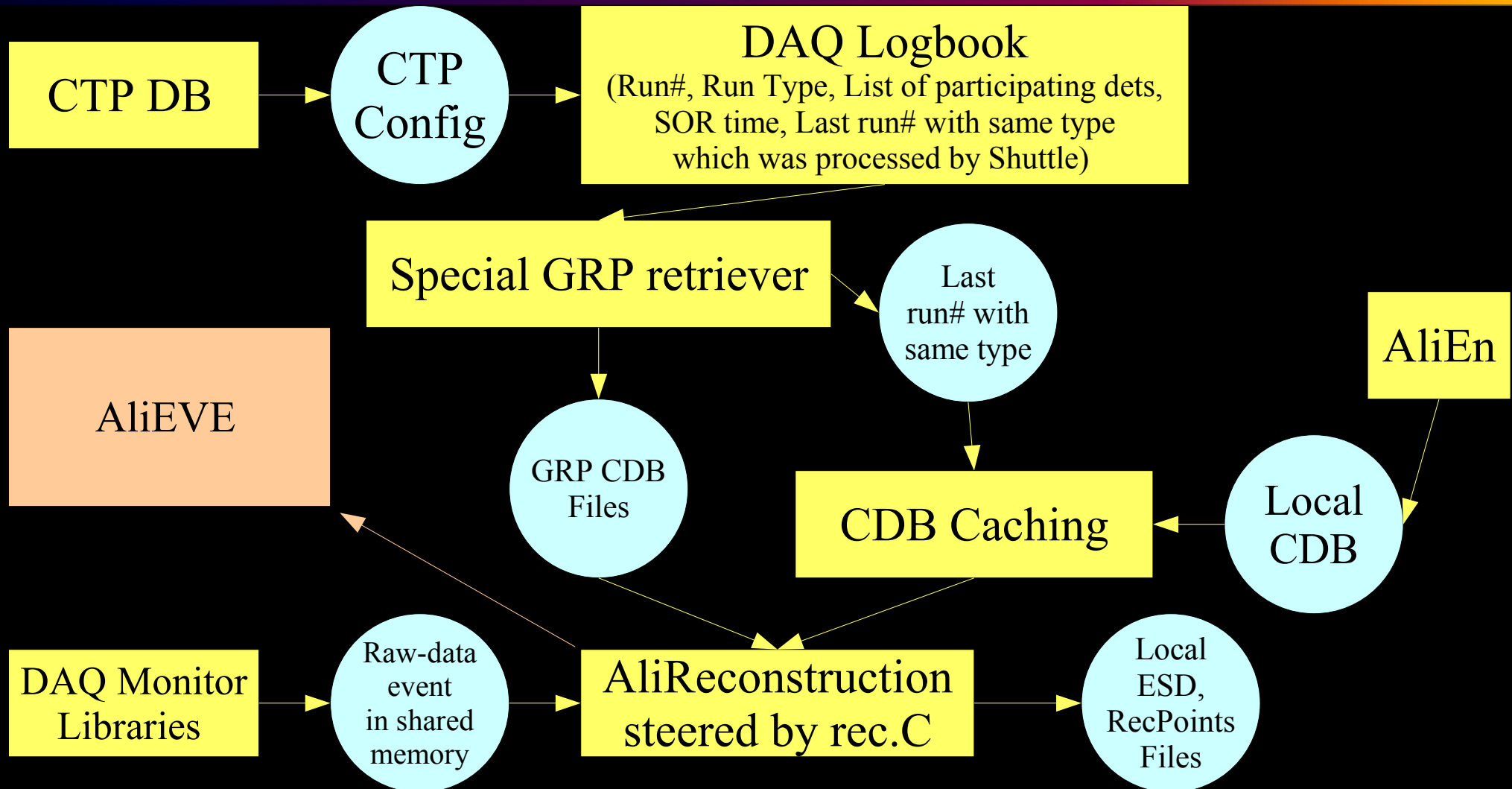
Contents

- Overview
- Implementation layout
- Details on various components
- Conclusions
- Interface with AliEVE

Overview

- The idea is to run Offline reconstruction directly while data is being taken
- Why: considered as very important since it will provide global high-level QA
- What is it: standard Offline reconstruction featuring all its functionality
- Where it will run: on DAQ monitoring machine(s), DQM nodes
- Who will operate it: QA shifter

Implementation Layout



Special GRP Retriever

- It queries the DAQ Logbook and extracts the GRP information shortly after the start of run:
 - run type
 - participating detectors
 - SOR time
 - Last run which has the same type and has been already processed by Offline Shuttle
 - CTP configuration
- Stores local file with GRP CDB entries for the current run

CDB Caching

- DQM node accesses AliEn via CDB API, cache the CDB into a local files
- Special GRP retriever provides the run# that is needed
- The location of the local CDB is then send to the reconstruction process

Reco of events in shared memory

- The access to DATE events in shared memory is provided by the DATE monitor libraries
- New raw-reader class:
 - Derives from AliRawReaderDate
 - NextEvent() - wait and gets an event from shared memory
 - In a separate library
- New static AliRawReader::Create(<uri>) method:
 - Depending on the 'uri' creates the corresponding raw-reader object
 - New raw-reader class is instantiated in case the <uri> starts with 'mem://':
 - 'mem://' - events from shared memory
 - 'mem://<date_filename>' - emulation mode, events are taken from date file via DATE mon libs

Reco of events in shared memory

- AliReconstruction::Run() method split in 3 pieces:
 - InitRun
 - Initialize the run-loaders and create the event-header tree, no events are added
 - RunEvent
 - Add the event in the event-header tree
 - FinishRun
 - 'Close' the run-loaders
- Reconstruction is run via normal rec.C by providing the correct <uri>

Conclusions

- Most of the components are tested and available
- Now they have to be assembled and tried in the online environment
- Open issue: GRP entries will be valid for the current run#, while the other CDB – for the last run with same type. This kind of functionality does not exist in CDB manager code, moreover the run# is taken from raw-data by default. Two solutions:
 - Add the possibility to set run# for (entire or part of) CDB – not good
 - Change run# while caching the CDB entries locally - preferable solution
- Interface online reco with AliEVE (possibility to visualize the reconstructed events really online)

Interface with AliEVE

- First step would be to move EVE event manager to STEER and use it inside AliReconstruction to steer and synchronize run-loader and raw-readers
- Add hooks to AliEVE inside the event loop, based on the event manager
- Run default chain of AliEVE macros