LAr Software Status and Plans

ATLAS Canada Review

4-5 November 2003

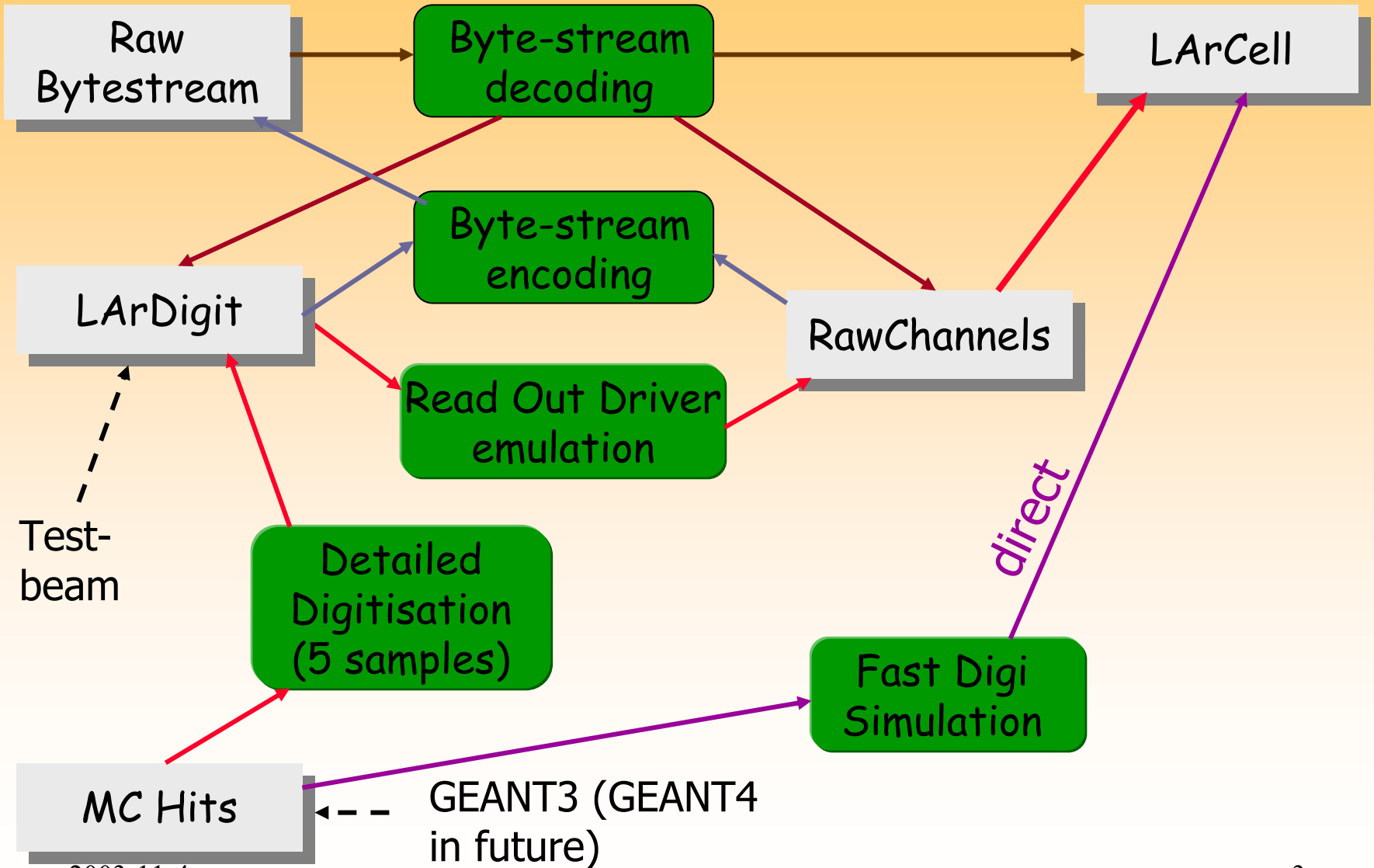
Michel Lefebvre

Victoria

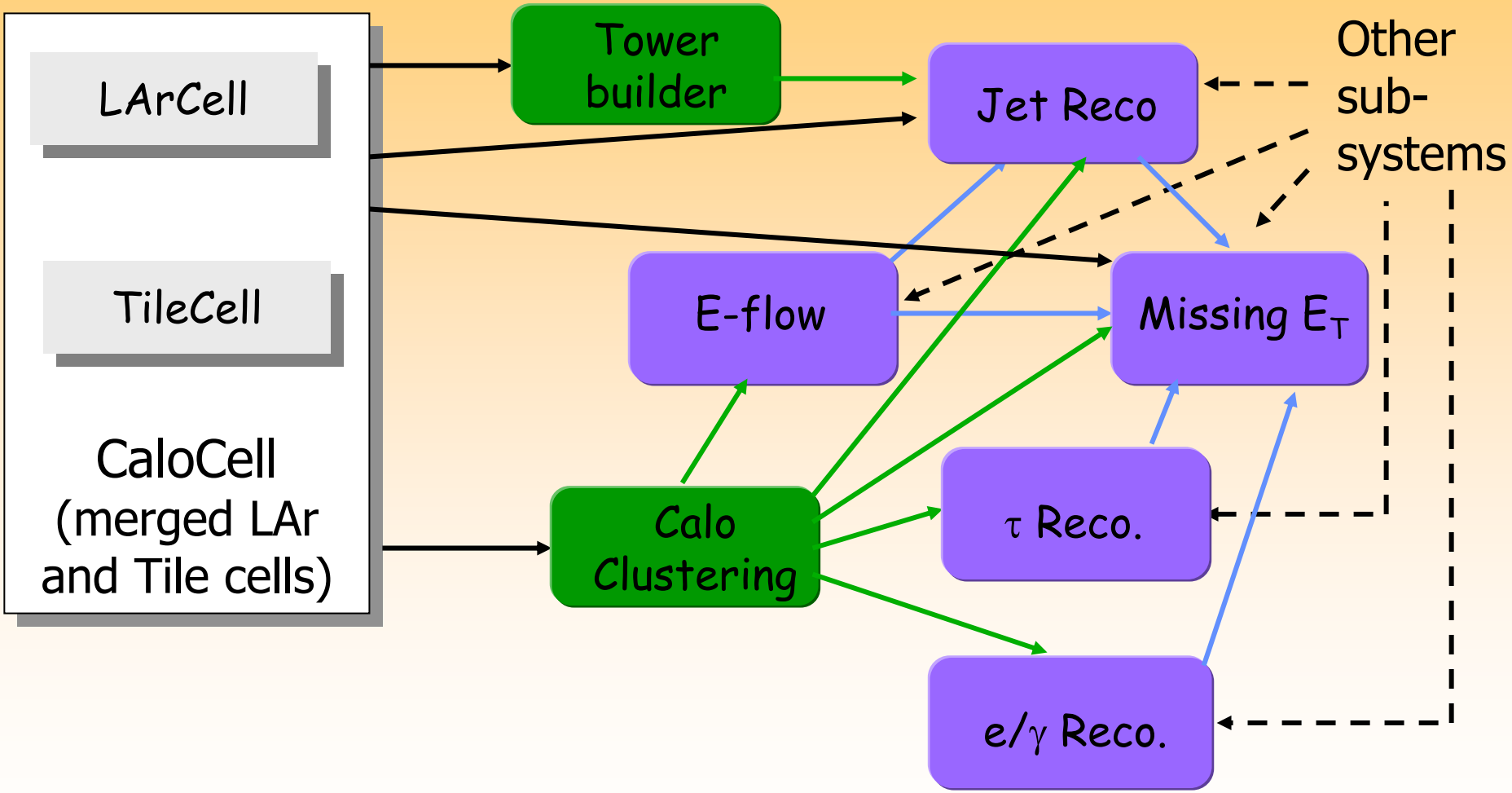
Overview of LAr SW

- **Developed for MC simulation**
  - Often developed/used/tested only with fast MC (ATLFAST)
  - Even with full MC, extreme simplifications built in at a low level
    - Complete inability to handle cell-by-cell calibrations
- **"ATHENA" Reconstruction algorithms available for**
  - Detailed simulation of LAr detector response from GEANT hits
  - Cluster finding and their properties (e.g. shower shapes)
  - Calibration of e.m. cluster, jets, missing E_T
- **Collaborating closely together with Tile group**
- **Strong contributions within combined performance groups**
  - e, γ, τ, jet, $E_T$miss reconstruction and identification
- **Major contributions for trigger community**
  - Simulation of data formats as seen by the trigger

LAr Athena Reconstruction (1)



LAr Athena Reconstruction (2)

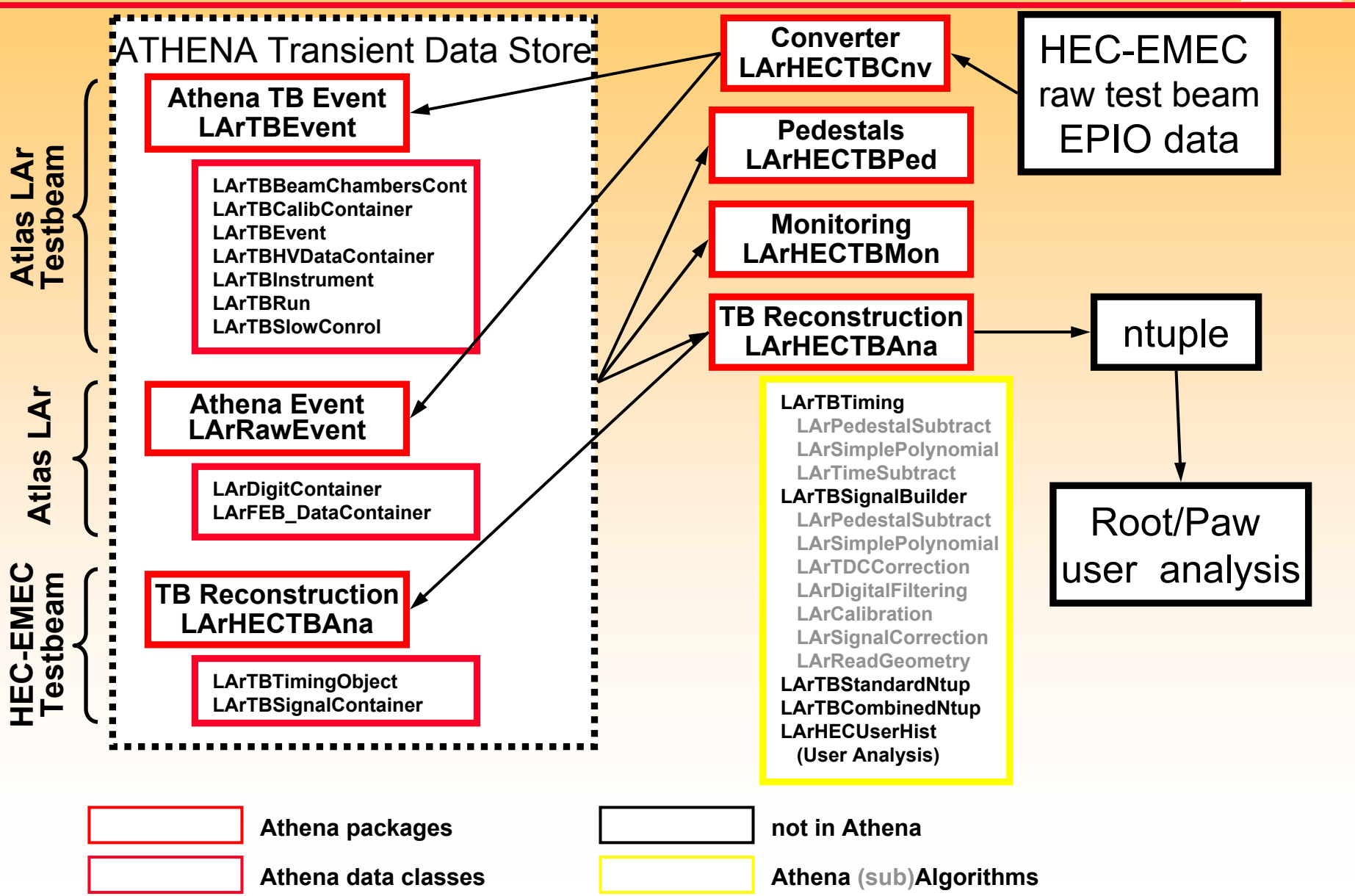


Testbeam Software

- **First applications of ATHENA to real data:**
  - 2002 EMEC+HEC and 2003 FCAL beam test data
  - Pioneered by Victoria: Kanaya *et al.*
  - Several rude shocks to the offline community:
    - Different calibrations per channel
      - Pedestals, pulse shapes, gain, ...
    - Calibrations change with time
    - Want to compare different types of reconstruction, clustering, ...
      - Non-Gaussian, variable noise, ...
  - Pragmatic solution for the older testbeams:
    - Use minimal offline existing class structure, write rest of code in testbeam specific packages

HEC-EMEC Test Beam Software

schematic view (May 2003)



Compromises w.r.t. "ideal ATHENA" for previous LAr Testbeams

- **DAQ systems not TDAQ**
  - No standard eformat/ByteStream data
  - Needed custom event readers (EventSelector/EventIterator)
- **Identifiers / Geometry services incomplete**
  - Custom TB classes introduced for analysis (hack ...)
  - Even for ATLAS, need some way of handling mis-alignments, swapped channels, etc.
- **Insufficient Database support in ATHENA**
  - Heavy use of input ASCII files for geometries, calibration constants, etc., hidden in new custom TB classes
- **Testbeam specific information introduced**
  - In principle, only "asynchronous trigger" is different
- **General inability of ATHENA to handle time-varying data from imperfect detectors**

Current LAr software development

Large rewrite of LAr offline software, driven by 2004 combined beam tests (Beam test software coordinator: McPherson)

1) **New "Bytestream converters" to read data from LAr RODs and ATLAS DAQ**
2) **Offline packages to emulate ROD usecases**
   - Eg: analyze FADC data for testbeam and ATLAS commissioning stages, normally "crunched" in RODs
3) **Cell identifier and geometry mapping**
   - Read from databases
   - Handle "nearest neighbours" *etc.* for clustering (was impossible)
   - Cell properties (volumes, ...) needed for energy weighting
4) **Calibration databases with "run/event" validity**
   - Used both online and offline
5) **Rewrite of detector monitoring for compatibility with other systems**
   - Close collaboration with Event Filter team (C. Padilla, CERN; M. Bosman, Barcelona)

LAr Readout Drivers: real and emulated

- **Read FADC time samples (Digits) from front end boards**
- **Perform pulse reconstruction (optimal filter, cubic, ...)**
  - **Produce Energy and Time (RawChannel)**
- **ROD → Bytestream → DAQ → Persistency (file) → ATHENA**
- **Raw time samples also sometimes sent → Bytestream**
  - **Testbeam, ATLAS commissioning, subset of pulses ATLAS running**
- **RODs must also handle shape/gain calibration runs**

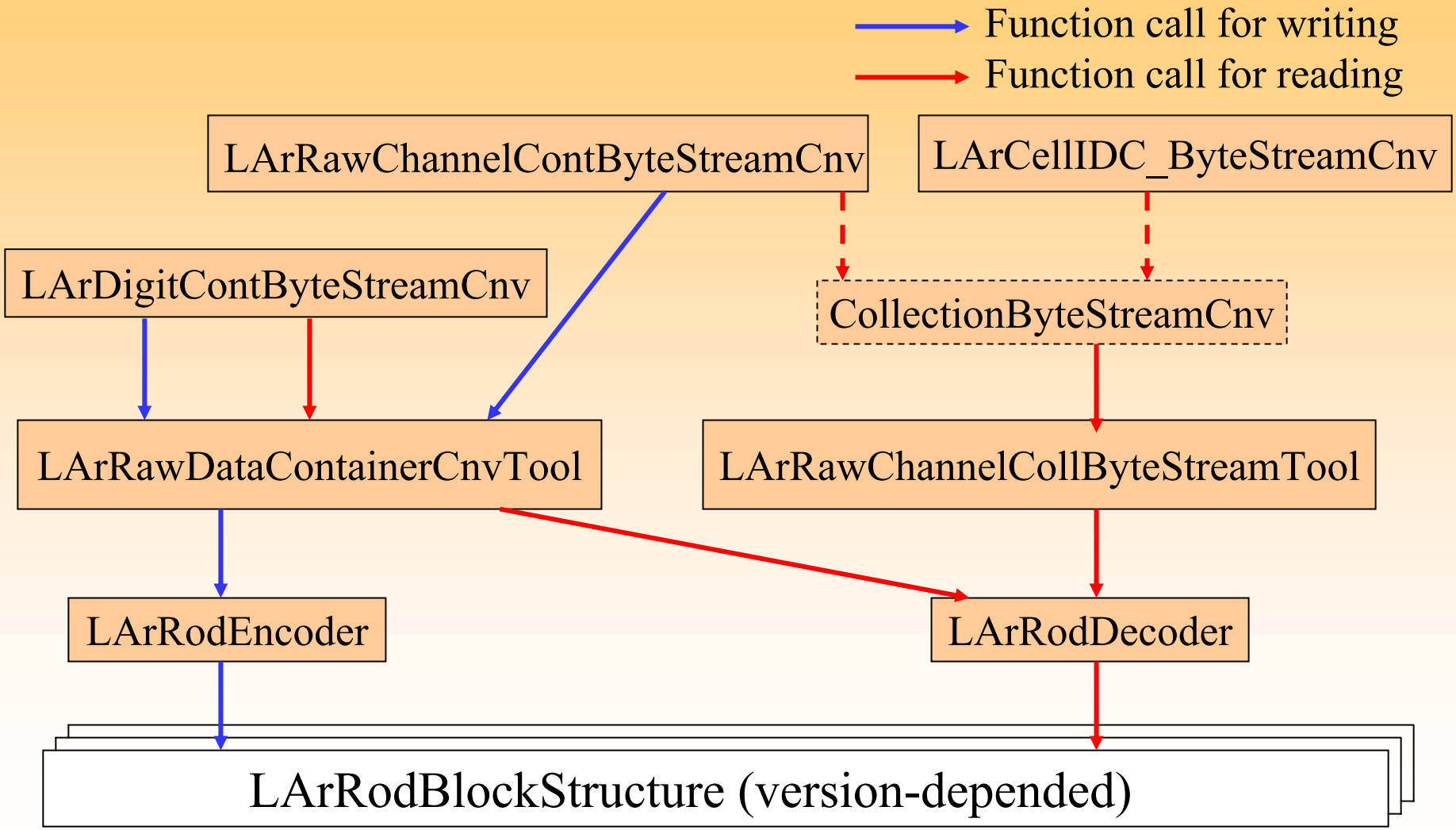
<u>Physics</u>
LArDigits
⇓
OF or cubic
⇓
LArRawChannel
⇓
LArCells
⇓
Clusters

<u>Calibration</u>
LArDigits
⇓
Avg. Triggers
⇓
Averages (x 3 gains)
⇓
(1) Pulse Shapes
(2) Gain Coeffs.

<u>Physics Pulse Shapes</u>
LArDigits
⇓
Pedestals
Avg. Pulse Shapes

<u>OF Computation</u>
LArPedestals
Physics Pulse Shapes
Calib. Pulse Shapes
Gain Coefficients
⇓
OF coefficients

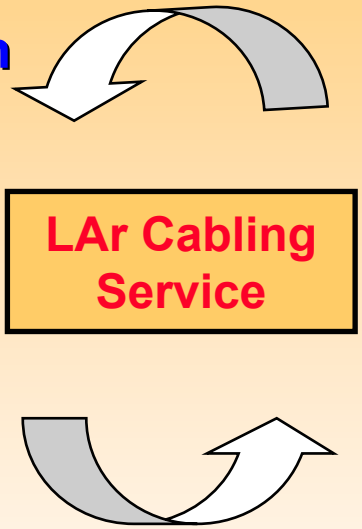
LAr Converters: architecture



LAr ATHENA Identifiers

OFFLINE

- 32-bit integer
- Give detector cell position
- Access via "helpers"
  - LArEM_ID (HEC, FCAL)
  - ieta, iphi, sampling, ...
  - Nearest neighbour lists
- Helpers built from (XML) dictionary
  - Hash tables built for fast access



ONLINE

- 32-bit integer
- Hardware identifier
  - Feedthrough, slot, channel, ...

(unconnected channels annoying ...)

- 2002 EMEC+HEC and FCAL TB
  - Used private TB index
  - Made dummy (valid) offline ID
  - LArCabling → dummy online ID
  - Breaks: hashing, neighbours, ...
- 2004 combined TB and ATLAS
  - XML dictionary for geometries
  - Needs detector input for LArCabling
  - Developing database

LAr ATHENA Geometry services

- **Current LArTB in ATHENA**
  - Reads from ASCII files (private indexing schemes) for geometry information
  - Private analysis routines, would not work transparently with, eg, offline clustering routines
- **Cell Geometry being developed based on testbeam experience**
  - Online + Offline Identifiers
  - Cell position (x/y/z , η/ϕ/depth)
  - Cell volume
    - Essential for cell weighting techniques
  - Both IDEAL and TRANSFORMED coordinates
    - Implemented in database
- **Geometry subdetector implementation:**
  - EMEC+HEC: Lefebvre
  - FCAL: Loch
  - EMB: McPherson (+ Boonekamp)

Databases (both offline and online)

1) **Configuration DB**
   - DAQ, ROD, FEB, Calib Board configurations
2) **Online Book-keeping DB**
   - Beam energy, table position, beam type, ...
   - Previous LAr TB used LArBookkeeping (Grenoble)
   - TDAQ uses a private “OKB” DB
   - LAr moving to single conditions DB including book-keeping
3) **Conditions DB: Online**
   - Calibration constants (OF weights etc.)
   - Also collects monitoring histograms, etc.
   - Can collect calibrations from offline (synchronization??)
   - Must be available 24/7, probably in beam hut area
4) **Conditions DB: Offline**
   - Mirror online, but accessible from more places

Other work in progress

- **Monte Carlo (both ATLAS and testbeam):**
  - Moving towards same geometry and persistency as offline (lagging slightly behind)
- **Online Monitoring**
  - Previous LAr testbeams: completely homegrown
  - Currently working closely with Event Filter groups to write LAr monitoring in HLT compatible algorithms
    - < 10 msec for LVL2, < 100 msec for LVL3 (C. Padilla, CERN)
    - And use of HLT real-time histogramming (M. Bosman, Barcelona)

Summary

- **LAr software in a phase of rapid evolution**
  - Algorithms being added / improved
- **Testbeams acting as benchmark**
  - Experience from 2002/2003 being built upon
  - Missing pieces in offline code identified, being developed
  - 2004 testbeam being used as goal for restructure
- **Canadian groups strongly involved**
  - Pioneered the use of ATHENA for testbeam analysis
  - Used in 2002 EMEC+HEC and 2003 FCAL run
  - Current impact muted somewhat by resources (RAs, travel)
- **Looking forward to 2004 and beyond**