A few issues about MuonGeoModel

2/7/08

stefania spagnolo

Reorganization of ReadoutElements:

goal – optimize data access (algorithm timing!)

Mdt example

- now:
  - MdtReadoutElement = 1 multilayer
  - MdtPrepDataCollection contains data corresponding to 2 MdtReadoutElements

First implementation

- for rel. 14
  - keep MdtReadoutElements
    - purely readout geometry – no link to raw geometry volumes
  - add MdtDetectorElements
    - holding pointers to 2 MdtReadoutElements
    - MdtPrepDataCollection contains data corresponding to 1 MdtDetectorElement

Reorganization of ReadoutElements:

goal – optimize data access (algorithm timing!)

- before implementing fully Mdt and start with Rpc/Tgc/Csc *try to estimate the gain:*
  - Test algorithm: **MuonGeoModelTest/MuonGMTestOnPrd**
  - logic: loop over MdtPrepDataCollections
    - for each collection: processCollection
      - get collection hashId + ask the manager for MdtDetectorElement(hashID)
      - for each prepData:
        - get prepData Identifier
        - ask MdtDetectorElement->center(Identifier)
          - will decode Identifier to get ml,tl,tube
          - ask MdtReadoutElement[ml-1]->center(ml,tl,tube))
    - for each collection: processCollectionOld
      - for each prepData:
        - get prepData Identifier + ask the manager for MdtReadoutElement (Identifier) + ask MdtReadoutElement->center(Identifier)

Reorganization of ReadoutElements:

goal – optimize data access (algorithm timing!)

MuonGeoModel in dbg mode gain 37%

- **use *callgrind* for algorithm profiling**
  - 5.95% of Job cpu for processCollectionOld
    - 2.7% manager->getMdtReadoutElement [identifier decoding]
    - 2.4% MdtReadoutElement->center(Identifier) [identifier decoding]
  - 3.77% of job cpu for processCollection
    - 3.3% MdtDetectorElement->center(Identifier)
      - 0.42% x 3 MdtIdHelper->multilayer/tubeLayer/tube(Identifier)
      - 1.16% MdtReadoutElement->center(tubeLayer, tube)
- callgrind output in
  - www.fisica.unile.it/~spagnolo/allow_listing/AtlasMuonSoftware/callgrind.out.REvsDE

Reorganization of ReadoutElements:

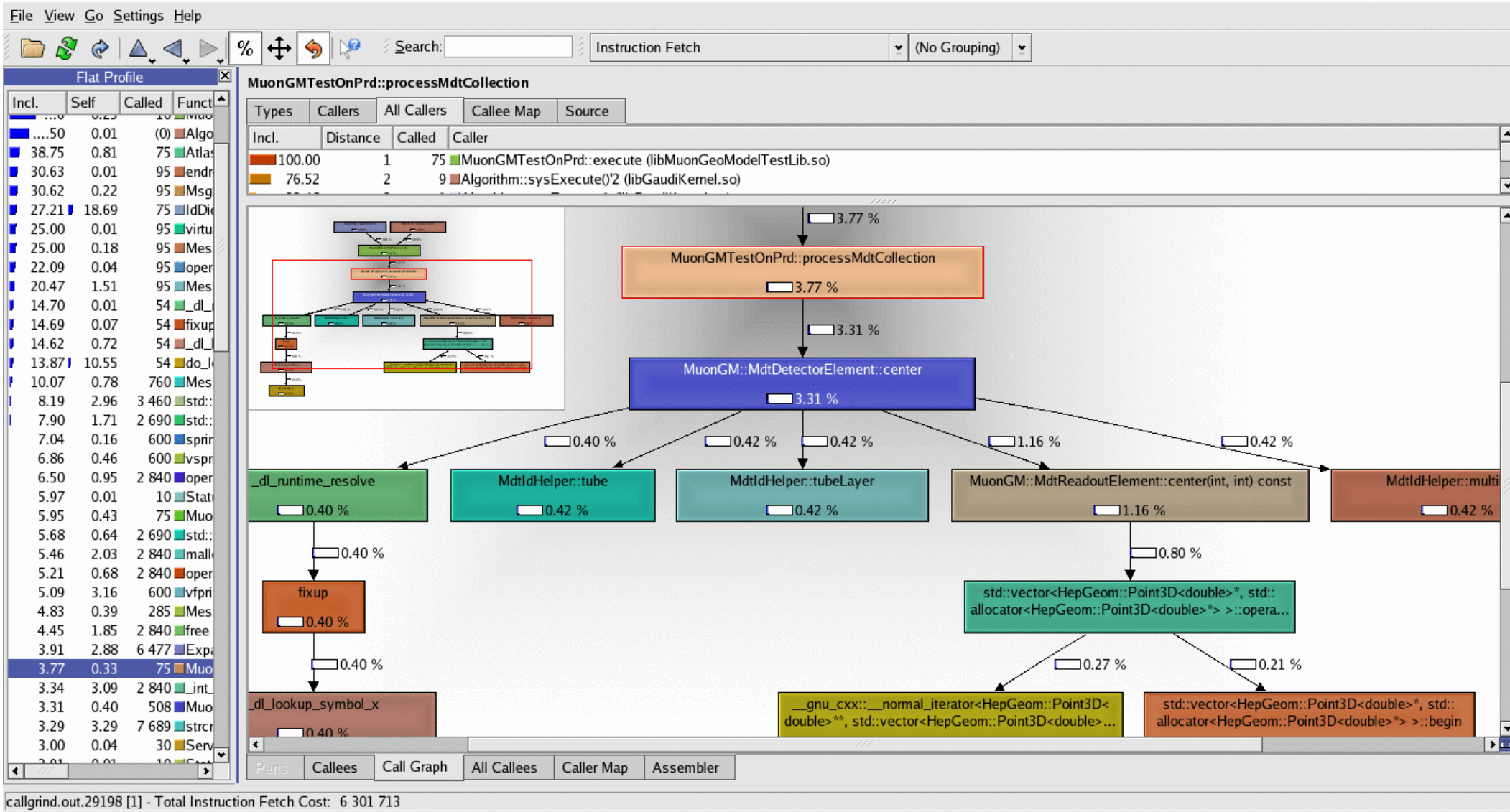
goal – optimize data access (algorithm timing!)

MuonGeoModel in opt mode gain 50%

- **use *callgrind* for algorithm profiling**
  - 3.04% of Job cpu for processCollectionOld
    - 1.24% manager->getMdtReadoutElement [identifier decoding]
    - 0.83% MdtReadoutElement->center(Identifier) [identifier decoding]
  - 1.49% of job cpu for processCollection
    - 1.01% MdtDetectorElement->center(Identifier)
      - 0.xx% x 3 MdtIdHelper->multilayer/tubeLayer/tube(Identifier)
      - 0.34% MdtReadoutElement->center(tubeLayer, tube)
- callgrind output in
  - www.fisica.unile.it/~spagnolo/allow_listing/AtlasMuonSoftware/callgrind.out.REvsDE

Reorganization of ReadoutElements:
goal – optimize data access (algorithm timing!)

processCollection profile



Reorganization of ReadoutElements:

goal – optimize data access (algorithm timing!)

- **NOTE:** all timing tests are insensitive unless
  - no use of MsgStream is done
  - the MuonGeoModel cache is not built in advance (init time)
    - from MuonGeoModel.MuonGeoModelConf import MuonDetectorTool
    - MuonDetectorTool = MuonDetectorTool()
    - MuonDetectorTool.FillCacheInitTime = 1 # default is 0
    - *should we turn this to default option ???* (most likely this is going to be the default option in HLT)
- ***Shall we start moving MuonGeoModel with the partial implementation of XxxDetectorElement in 14.0.0 nightlies ???***
  - no disruption of clients (standard usage of XxxReadoutElements is kept)

Other pending issues

- no test yet of simulation reading cond. data
  - need to re-tag the test folder in COOL with alignment data
- *We should define clearly the strategy (Geometry tags and Condition tags) for our non official exercise of simulation + reconstruction with condition data*
- resume clash survey
  - in cooperation with Alexander K.
  - Laurent C. recently spot a small clash to be investigated
  - *cutouts*