PRS muon Meeting,

CERN, 4 February 2003

New geometry architecture for Muon DT

Stefano Lacaprara, INFN and Universitá di Padova

Summary:

- DDD transition status,
- New geometry architecture,
- new interface to access DT geometry,
- release plan,

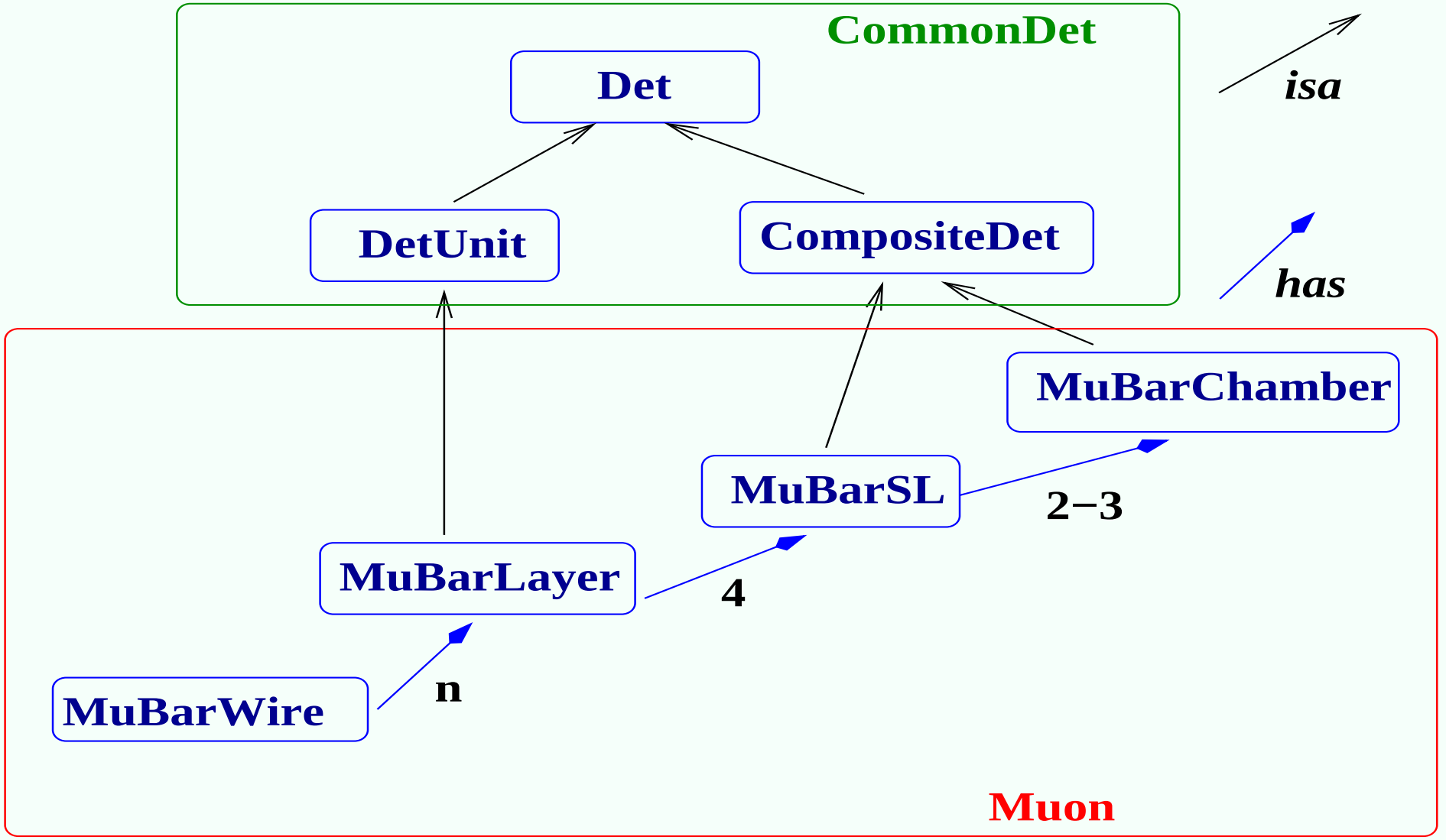
You're looking at these slide because they are published in a portable format (pdf), if yours are not (e.g. ppt) I won't be able to do the same! *(slide viewable by anyone campaign)*

DDD transition status:

READY!!

- Still private code, not yet released nor committed in CVS;
- careful tested for geometry related quantities (chamber, SL, layer and wires position and orientation);
- thorough comparison for SimHits and Digis wrt old G3 geometry;
- fully working for local chamber reconstruction (hit and segments);
- fully working also for global reco (tried L2 internal seeded, resolution and efficiency are ok).

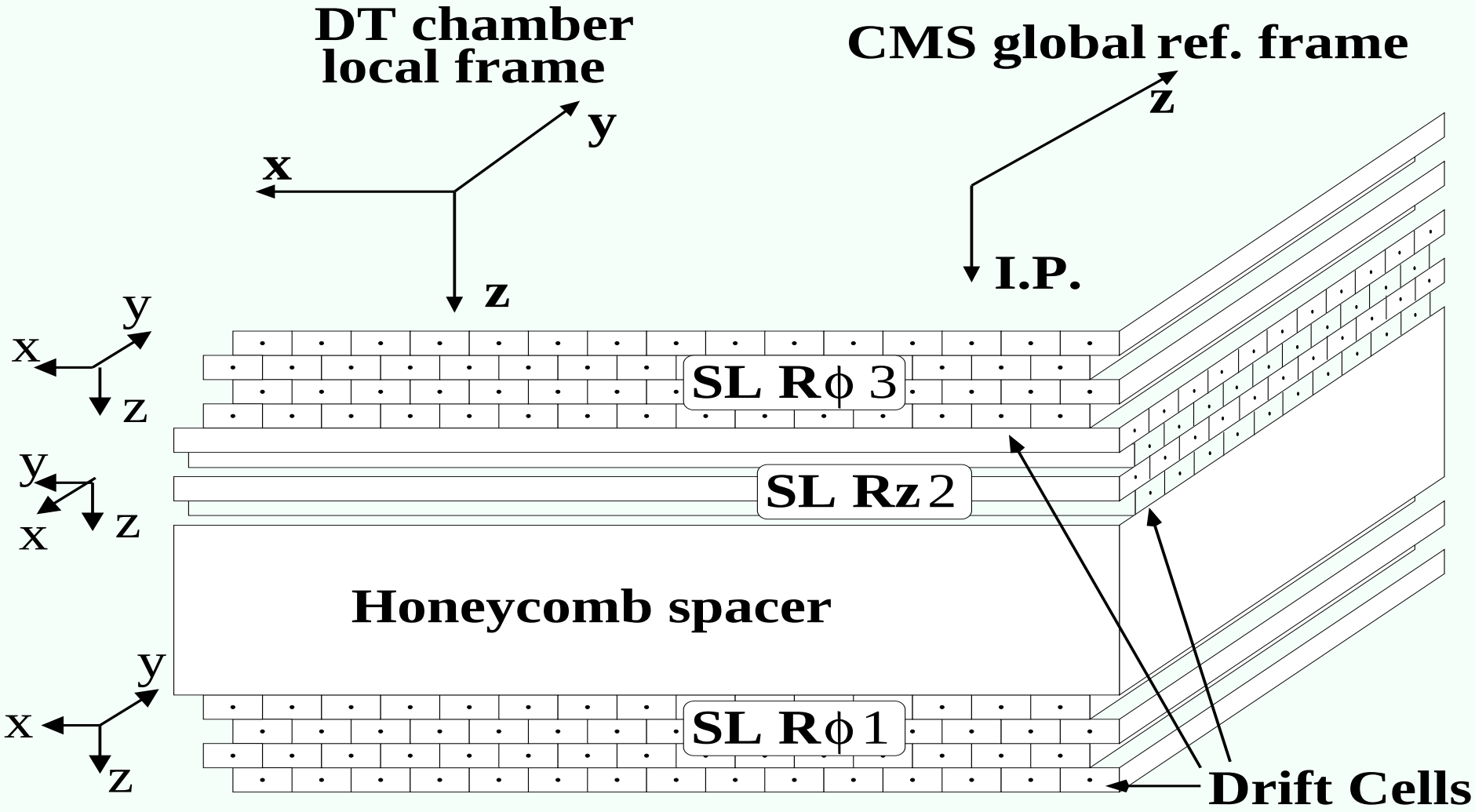
New DT geometry architecture:



New DT geometry architecture (2):

- Every DT "detector" (chamber, superlayer, layer, not wire) *is a* **Det**,
- unified access to position, rotation, frame transformation,
- Local coordinate system defined at every level, not just for chamber,
- several choices possible, I decide to maintain the old chamber frame as far as possible (can be discussed),
  - ⋆ **DTLayer ref frame:** x is perpendicular to the wire, the coordinate measured by the DT cell, y is along the wire, z is toward I.P. **for both $R-\phi$ and $R-Z$ layers**,
  - ⋆ **DT superlayer frame** follows the layer one, so $R-\phi$ and $R-Z$ SL have ref frame rotated by 90°,
  - ⋆ **DT chamber ref frame** is identical to the old one: z is toward I.P., y is parallel to global CMS z and x close the dextro-rotary tern: **this frame correspond to the $R-\phi$ SL's one**

- ⋆ In chamber frame, in order of increasing local z, so going outside-in, we find $R-\phi$ SL (no.3), $R-Z$ SL (no.2 if present), honeycomb spacer and then $R-\phi$ SL (no.1)



New DT geometry architecture (3):

- DT layer *is a* **DetUnit** (previously Chambers were DU),
- SimHit and Digi access are now at DT layer level,
- G3/4 SimHits are given in Wire frame, now only a shift is needed to put them in layer frame,
- This simplify considerably the MuBarSimHit interface, which must return SimHit information in the DetUnit ref-frame
- Simplification also for Digitizer, since meaning of local coordinate of SimHit does not depend any more on SL number: the coordinate measured is **always** x
- Modification to SimHit formatter, MuBarSimHit and MuBarDigitizer already done and tested,
- Plan to use *ShiftingSimHit* and *ShiftingSimDet* (as in the Tracker) to avoid the need to made the shifting in the MuBarSimHit interface

How to access the geometry: new interface

- Two possible navigation pattern: via **CommonDet** interface or via **MBDetector** interface,
- The former used when accessing objects as Dets, as in global reconstruction,
- The latter used to access objects as MuBarObjects, to use the specific MB interface, e.g. to get MuBarSegments (as such and not as RecHits) or to get number of wire in a DT Layers or similar...,
- Concrete examples will be provided in the documentation:,

  Formal promise: no release without complete documentation
- Some anticipation:

- CommonDet access to SimHits:,
  - get barrel DetLayers (as before):

```
MuBarrelSetup* muon_setup =
      Singleton<MuBarrelSetup>::instance();
CMSMuonBarrel* muon = muon_setup->MBarrel();
vector<DetLayer*> detLayers=muon->barrelLayers();
```

  - loop through DetLayer and get all DetUnit via `DetLayer::detUnits()` method,
  - get SimDet from `DetUnit::simDet()`,
  - get SimHits via `SimDet::simHits()`
- MBDetector access to MuBarLayer:,
  - get barrel detector map (new);

```
MuBarrelSetup* muon_setup =
        Singleton<MuBarrelSetup>::instance();
const MuBarDetectorMap& map=muon_setup->map();
vector<MuBarLayer*> layers=map.layers();
```

- More complex: CommonDet access to SuperLayer (as Det):
  - get barrel DetLayer as before:
  - loop through DetLayers and get the Dets which compose them (namely the DT chamber) via `DetLayer::dets()`,
  - these dets are CompositeDets as well, since they are DT chambers which are composed of DTSuperLayer, so loop through Dets and ask again for the det which compose them (again with `CompositeDet::dets()`)
  - nasty technicality, since Det does not have dets() methods, we should first `dynamic_cast<CompositeDet*>(detPointer)`: to check with Teddy if possible to extend Det interface to avoid that...
  - the dets obtained like that are DTSuperLayer: only the Det interface is available (unless we dynamic_cast again), so we can access position(), rotation(), recHits() (return empty vector so far) and so on...

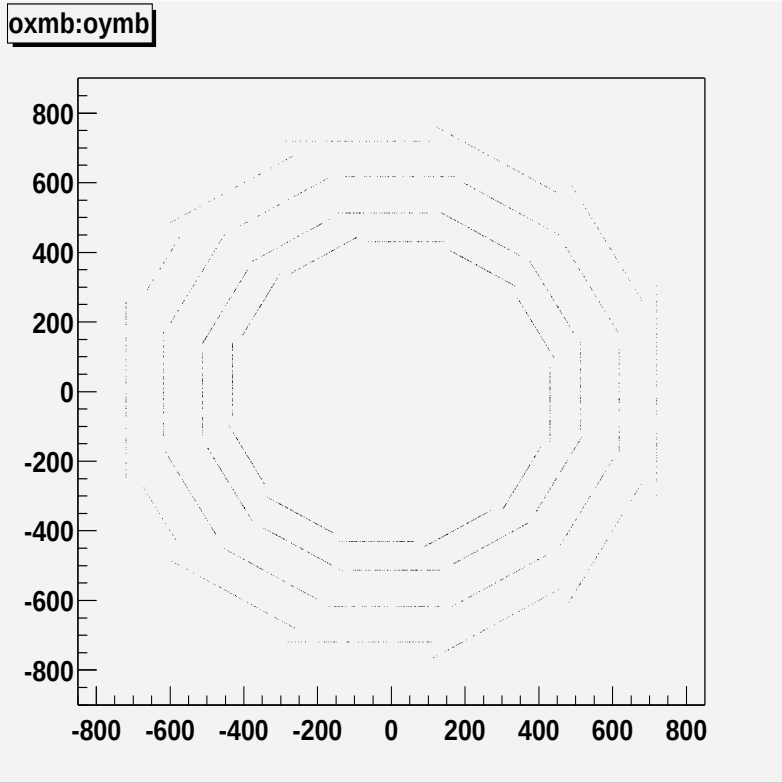
New DT geometry architecture (5) technicality:

- The map which allows the access to the DT detector also *owns* the detector, and it's responsible for the destruction at the end of the jobs;
- MuBarChamber *owns* the MuBarSL, SL *owns* the MuBarLayers,
- No more ReferenceCounted objects, careful check that all the objects are really deleted at the end of the job (was not properly done in the past, maybe responsible for jobs which did not exit after completion);
- Still room for optimization, and improving of user interface: should check with clients (mainly DT Trigger and TestBeam application) is the interface should be extended and/or improved.

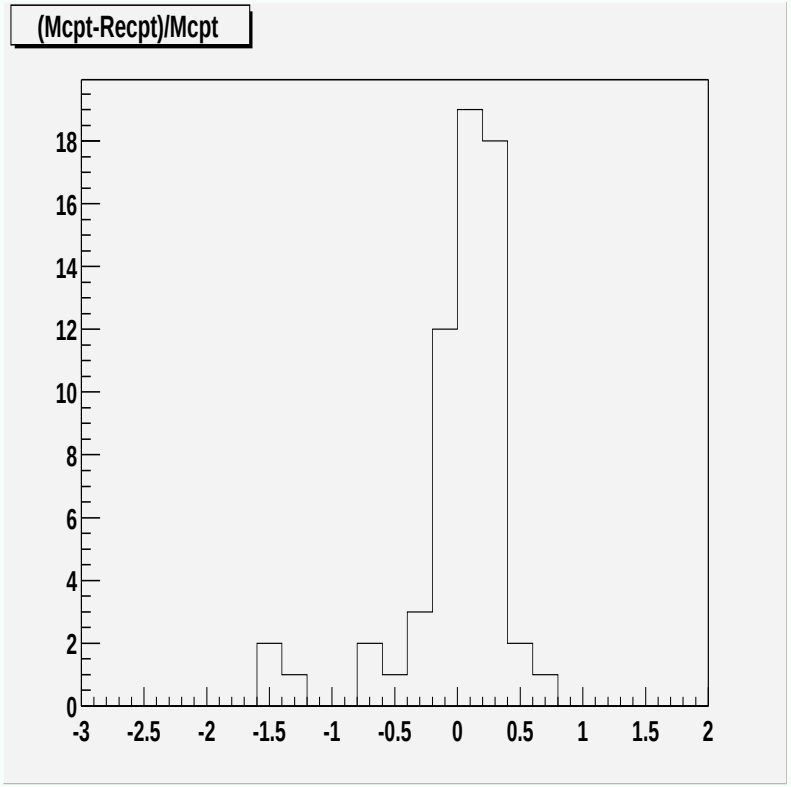
New DT geometry architecture (5) RecHits:

- So far the only RecHits available are the MuBarSegment, which are bounded to the MuBarChamber;
- Also MuBarRecHit (namely the hits at layer level) are built, but they are not (yet) attached to the MuBarLayer (as should be);
- First concentrate on guarantee backward compatibility in order to allow global reconstruction;
- Global reco is fully working;
- now concentrate on MuBarRecHit and modification in MuBarSegment building.

Propaganda plots:



Position of MBsegments



Resolution of MuonInternalReco (DT only), (very last minute plot...)

Not cheating, it's really working!

Release plan:

- Current developing based on ORCA 711, with gcc 2.95, plus integration with small modification introduced in ORCA 720;
- Must check with gcc 3.2;
- Integrate new geometry architecture with new readout by Giacomo (should be really easy, since the coupling is loose and well defined);
- New interface should be tested by DT trigger;
- thorough Doxygenation and Documentation;
- further optimization, improvement, test...
- Will be committed soon (this week), tagged, so that people can start playing;
- Will go in next ORCA release (730), timescale: after CHEP .