What Is Upgrade Simulation?

Monte Carlo in ATLAS Tutorial
29 September 2015

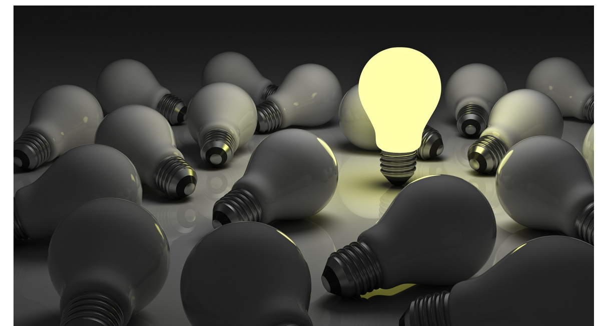
Jochen Meyer

Nikhef

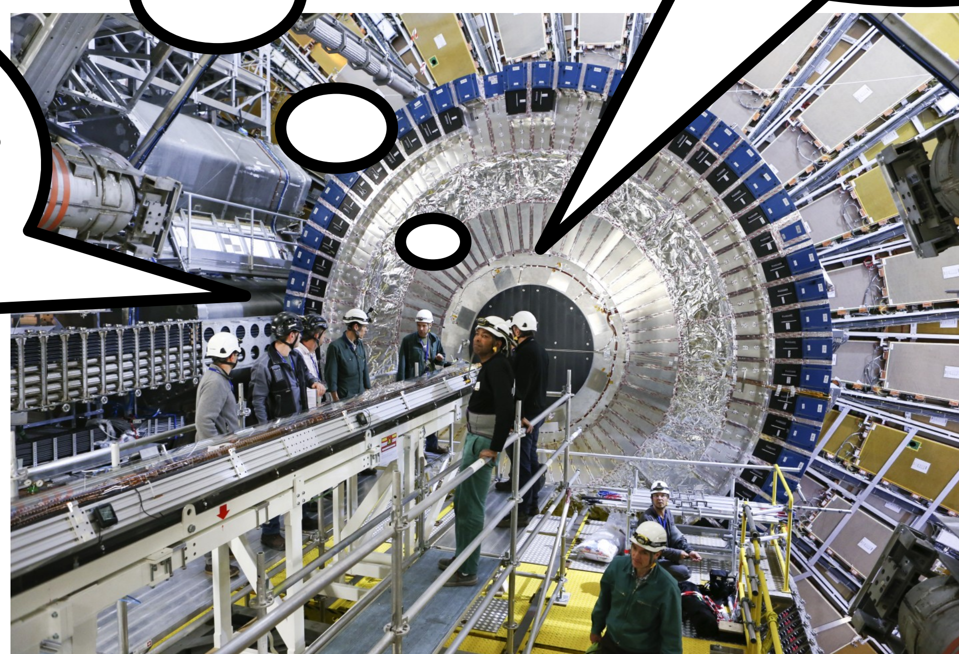
from viewpoint of “hardware”

Many ideas/solutions!

Let's ask the software folks to hit a bottom and find the right answer so we can start to build it “tomorrow”!



The detectors may reach some limitations when conditions change.



Typical questions in context of upgrade to be answered by simulation studies:

- Which rates and fluxes are expected in the region of the new detectors (preferably for several LHC upgrade scenarios)?
- What is the acceptance and efficiency of the new detectors?
- Which (combined) resolutions can be achieved by using the new detectors?
- How should the detector be (geometrically) designed to make the best use of it?
- What is the interplay of the new detector and the already existing one?

from viewpoint of “software”

Finally there is the chance to write new code and introduce new programming techniques!

There are a mildly interesting questions from the hardware folks.

Answers are expected in that time scale?!?! Seriously?!?!

Typical environment to start from:

- even though current software chain is running for already installed technologies an upgrade can be seen as chance to improve it

  → time constraints are limiting such efforts

- requested studies put different demands (which have been solved at some point for existing detectors) on required modifications

  → to speed up usually one needs to do things in parallel

- fundamental components (like geometry) change rapidly

  → either code is written in a flexible way (usually not) or needs to be adjusted often (including filling of databases)

- there is a variety of releases to develop upgrade software in

What kind of project are we talking about?

Detector upgrades may be …

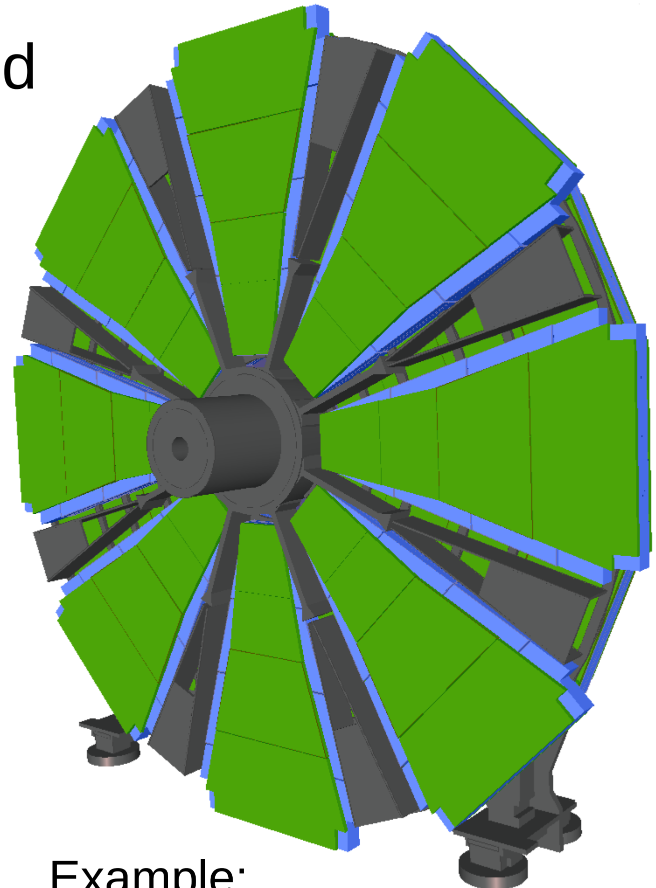
… something completely new, not yet seen (real **upgrade**)
(*example*: New Small Wheel)

… a “simple” **addition** of more detectors in currently empty spots
(*example*: elevator chambers added to muon system)

… a **replacement** of existing detectors with slightly modified new ones which use the same technology though
(*example*: replacement of MDT inner barrel layer)

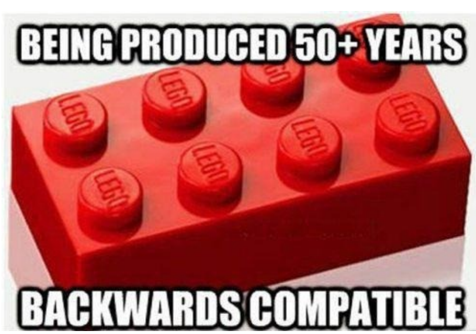
Solutions and complications differ a lot depending on the scope!

- new detectors need to be implemented in GeoModel
  - ➡ geometrical shape may not (yet) be used
  - ➡ dead regions inside detecting volumes could be complex
- association of “sensitive detectors” to simulated volumes to record hits
  - ➡ new hit classes are required to hold all information for subsequent steps or very fast studies/validation
  - ➡ first sim hit content can be too comprehensive for background studies
- supporting structures need to be adjusted
- identifiers need to be extended and reviewed
- all existing volumes in spots of new structures need to be removed
  - ➡ simulation configuration could become tricky

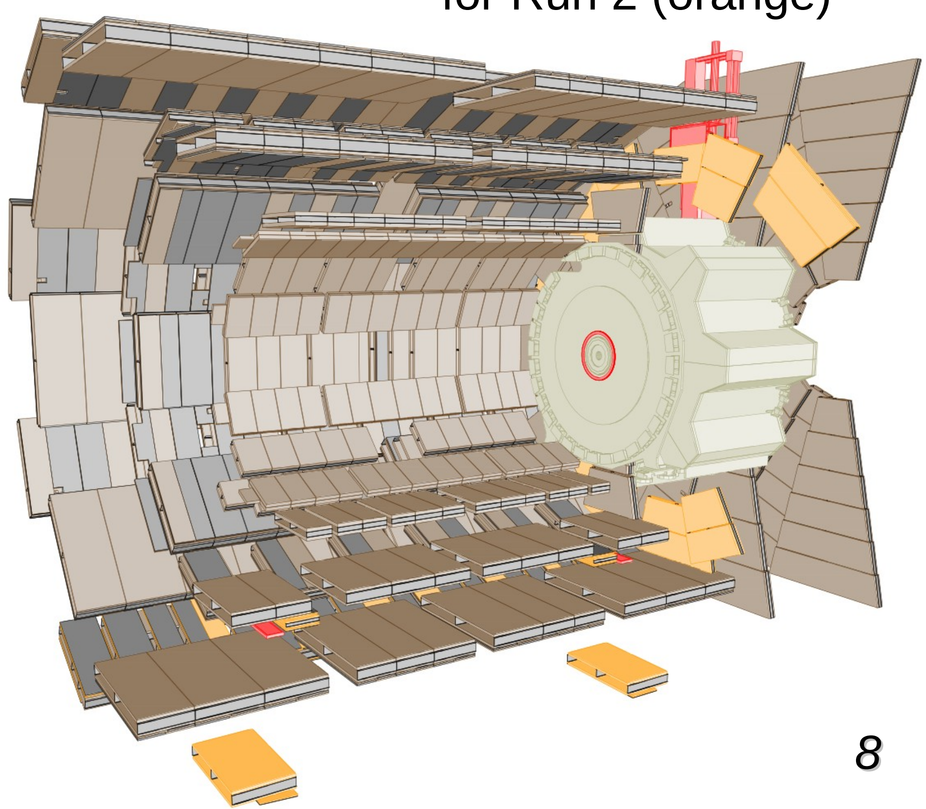


Example:
New Small Wheels (NSW)
for Muon Spectrometer (MS)

- extension of existing geometries with further chambers/detectors using already established coding techniques
  - ➡ check for clashes with other structures present in simulation
  - ➡ possible adjustment of simulated volume
- increase of identifier ranges used to address sensitive volumes uniquely
  - ➡ identifiers are highly packed bitmasks utilized also for conditions data and during previous real data recordings
  - ➡ backward compatibility is a major concern



Example:
chambers added to MS
for Run 2 (orange)



upgrade digitization

- porting and validation of stand alone code emulating detector response to Athena framework

  → usually written by detector experts and therefore improvable in terms of CPU/memory usage

- new digit classes need to be introduced (at first overloaded with information similar like sim hits) along with converters

- readout mechanisms, raw data formats, cabling, … unknown or changing a lot since depending on detailed detector configuration (e.g. number of strips or such)

→ fast/parametrized digitization to bypass this step

digitization of detector extension

- decoding and cabling needs to be adjusted

  → changed readout can introduce complications

- reconstruction suffers from same issues like simulation/digitization
  - ➡ unknown geometrical shapes of tracking surfaces
  - ➡ readout system and positions not fixed
  - ➡ detector output and therefore PrepRawData format may change with time (cluster, drift time, …)
- overlay of simulated data and (scaled) real data could enter combined reconstruction
- promising approach for fast studies is to approximate one or the other step
  - smearing sim hits to skip digitization
  - start reconstruction from digits rather than PrepRawData
  - ...

upgrade software development is a chance to …

- … learn something about already existing code
- … solve shortcomings of current implementations
- … contribute to detector design and software decisions
- … get in touch with a lot of code written by people on various levels in programming experience
- … face new challenges every other day
- … see how hardware ideas develop

➡ **There is always a not simulated upgrade!**

EXERCISE

- they are slightly meta stable since work in progress -

exercise – visualizing the NSW in VP1

- setting up environment: `setupATLAS`
- setting up Athena: `asetup 20.3.0.1,here`
- check out necessary package …

```
pkgco.py MuonGeoModelTest-04-00-10
cd MuonSpectrometer/MuonGeoModelTest/cmt
cmt config
gmake
cd $TestArea
```

- start VP1 with NSW configuration: `vp1 -nsw`
- check box “NSW” in “Geo”-tab

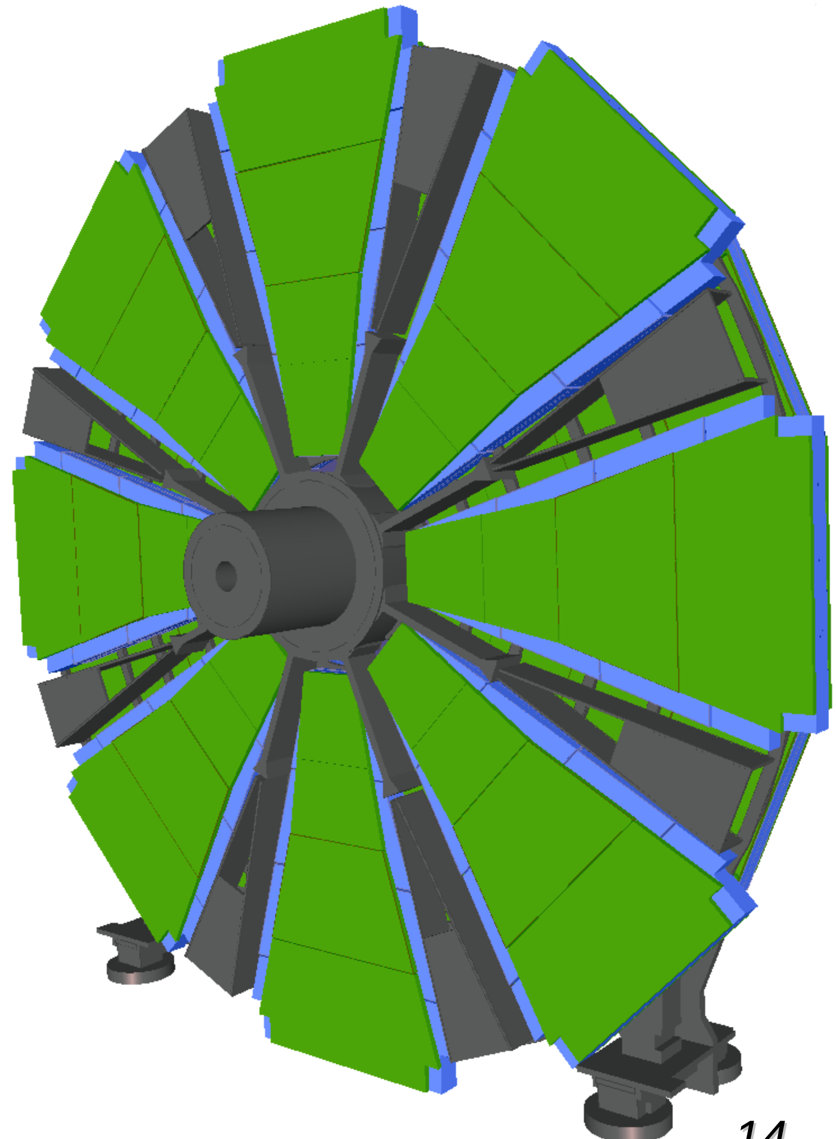
➡ **Do you spot a difference to the picture displayed here?**

➡ **Do you find inconsistencies when the MDTs are displayed to?**

➡ **What could be the implications?**

- Remark: VP1 does not just build GeoModel volumes, but also tracking surfaces!

- MicroMegas (MM) modules in green
- small Thin Gap Chambers (sTGC) in blue
- support structures in gray



exercise – simulating the NSW

- setting up environment: `setupATLAS`
- setting up Athena: `asetup 20.3.3.1,here`
- simulation transform:

```
AtlasG4_tf.py \
--inputEVNTFile /afs/cern.ch/user/j/jomeyer/public/NSW/EVNT/MC12.107209.ParticleGenerator_dimu_Pt10_100.EVNT.pool.root \
--preExec simFlags.SimulateNewSmallWheel=True \
--postInclude /afs/cern.ch/user/j/jomeyer/public/NSW/NSW.config.simu.py,/afs/cern.ch/user/j/jomeyer/public/NSW/NSWPRDValAlg.simu.py \
--conditionsTag OFLCOND-RUN12-SDR-22 \
--DataRunNumber 222250 \
--geometryVersion ATLAS-R2-2015-02-01-00_VALIDATION \
--outputHITSFile test.hits.pool.root \
--maxEvents 120
```

triggers NSW geometry building:

- turns on Geant4 volumes of MM's and sTGC's
- enables sensitive detectors
- turns off current small wheel

override to use NSW layout MuonSpectrometer.R.07.00-NSW :

- get the right identifier dictionary
- passive material not overlapping with NSW detectors

attaches validation ntuple dumper for simhits

➡ **Do you get the `NSWPRDValAlg.sim.ntuple.root` output?**

➡ **Browse the flat ntuple! (see *Variables.h for content)**

exercise – digitize the generated hits

- digization transform:

```
Reco_tf.py \
--postInclude /afs/cern.ch/user/j/jomeyer/public/NSW/NSW.config.digi.py,/afs/cern.ch/user/j/jomeyer/public/NSW/NSWPRDValAlg.digi.py \
--inputHITSFile test.hits.pool.root \
--outputRDOFile test.rdos.pool.root
```

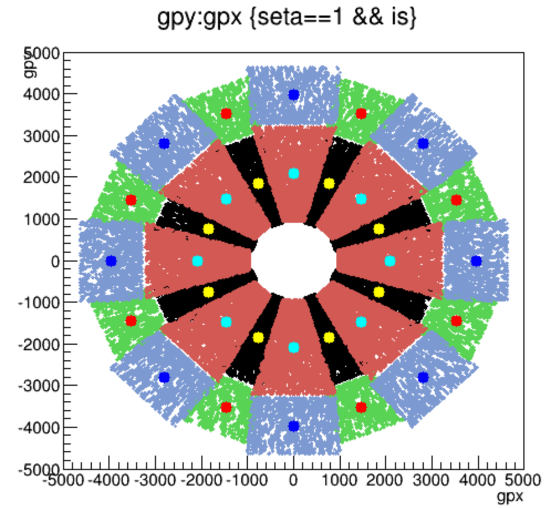
set up NSW geometry

- override to use MuonSpectrometer.R.07.00-NSW
- configure geometry algorithms/services (there is no `SimulateNewSmallWheel` here)

attaches validation ntuple dumper for simhits + digits

➡ Do you get the `NSWPRDValAlg.digi.ntuple.root` output?

➡ Produce some views in the XY plane and RZ plane of the sim hits / digits! (advanced level: “clown plot” on the right)



- important output to validate shapes, identifiers, …
- even trigger algorithms can already be written on that output

APPENDIX

some of the components implemented for the NSW

➡ completely new technologies are used, namely small thin gap chambers (sTGC) and MicroMegas (MM)

- generic muon sensitive detector (link)
- generic muon sim hit (link)
- fast digitization (link)
- full digitization (digits [MM, sTGC], digitizer [MM, sTGC])
- ntuple dumper for validation purposes (link)
- 13 geometry related packages: MuonAGDD, MuonAGDDBase, MuonAGDDDescription, MuonReadoutGeometry, MuonGeoModel, MuonGeoModelTest, NSW_Sim, AGDD2GeoSvc, AGDDKernel, AGDDControl, AGDDModel, AGDDHandlers, IdDictParser