

# Validation of the GEANT4-Based Full Simulation Program for the ATLAS Detector: An Overview of Performance and Robustness

D. Costanzo  
BNL (USA)

A. Dell'Acqua, M. Gallas  
CERN (Switzerland)

A. Nairz  
Univ. of Innsbruck (Austria)

N. Benekos  
MPI Munich (Germany)

A. Rimoldi  
INFN and Univ. of Pavia (Italy)

J. Boudreau, V. Tsulaia  
Univ. of Pittsburgh (USA)

March 11, 2005

## Abstract

This paper gives an overview of the validation tests of the ATLAS GEANT4 simulation package G4ATLAS, which were performed in the period from fall 2003 until the end of 2004. This spans the time from first comprehensive tests after G4ATLAS had been fully embedded into the ATLAS common framework in its full functionality and detail, the time of its development into a highly reliable, performant and robust tool, up to its extremely successful usage in ATLAS Data Challenge 2 (DC2) and (Combined) Testbeam simulation productions. During that validation phase, G4ATLAS became widely accepted as *the* simulation package for ATLAS.

The paper describes in some detail the testing period from its beginning (ATLAS offline release 7.1.0) until arriving at a stable version of G4ATLAS for DC2 (release 8.0.5), and presents results from performance tests carried out after DC2 and in the context of (Combined) Testbeam simulation.

# 1 Introduction

The ATLAS detector simulation programs have been heavily based on the GEANT3 (G3) simulation package and infrastructure since the inception of the experiment. These programs were used in the preparation of the experiment's Letter of Intent, in detector optimisation studies, for the various subdetectors' Technical Design Reports, and they were stress-tested in 'Phase 1' (i.e., the event-generation and simulation phase) of Data Challenge 1 (DC1), which was run during summer 2002. GEANT3 has been a powerful, reliable and successful tool within ATLAS for about ten years.

With the development and implementation of the GEANT4 (G4) toolkit [1, 2], starting from the year 2000, ATLAS prepared for moving its simulation suite to the object-oriented (OO) paradigm. GEANT3 and GEANT4 were run together for a while in order to validate the new suite against the previous one. The switch-over [3] eventually happened in 2003, in the early preparation phase of the upcoming Data Challenge 2 (DC2). Currently GEANT3 is still maintained, but not further developed any more; GEANT4 has become the main simulation engine of ATLAS, and all new developments are being carried out in the new environment.

The GEANT4-based detector simulation programs (G4ATLAS) are based on criteria like dynamic loading and action-on-demand, and all user-requested functionality has been added by means of plug-in modules. An interactive environment is provided, and a python-based user interface, which is supposed to replace the GEANT4 macro language within G4ATLAS, is currently under deployment. The simulation suite is now fully implemented for both the detector and testbeam simulation. The same code is used in all cases, the actual detector configuration is chosen at run-time: this fact ensures consistency throughout all applications.

G4ATLAS has been intensively exploited in running production tests, in running simulation for the ATLAS Combined Testbeam facility, in heavy ions simulation and ultimately in 'Phase 1' of DC2, which took place in the period from late spring to late fall 2004, where more than 12 million full physics events were successfully simulated.

Continuous comparisons with experimental results and previous versions of the simulation software, constant improvements and upgrades, and regularly carried-out performance tests have made G4ATLAS a highly dependable and robust tool, whose development will of course be on-going during current and future data taking preparation activities.

The main purpose of this paper is to give an overview of the validation activities performed in preparation of DC2 (started in late fall 2003 and lasting until the start-up of DC2 in late spring 2004), and during the (Combined) Testbeam period in 2004. A focus will be on describing the extensive test programme carried out in order to achieve the robustness and performance necessary for running G4ATLAS in (DC2) 'production mode'.

Section 2 outlines the testing activities done in preparation of DC2, from first tests with offline release 7.1.0 in late fall 2003, via releases 7.5.0, 7.7.0 and 7.8.0, to the work invested into the dedicated DC2 8.0.X (X = 0, ..., 5) branch releases during spring 2004. Section 3 summarises G4ATLAS performance in DC2; it gives results of performance checks done in fall 2004 with release 8.0.7, whose version of G4ATLAS is identical to the one run in large-scale DC2 production. Section 4 highlights some activities, out of many, performed after DC2, up to release 8.7.0 (fall 2004). Section 5 is dedicated to performance measurements done in the Combined Test-Beam (CTB) simulation area. And Section 6, finally, addresses

in detail a few more general, still open issues and problems which have been encountered during the testing period.

## A Note on Units

Most of the actual test jobs were run on the CERN LSF batch farm, whose large number of available machines allowed to run many jobs in parallel in different configurations. LSF also has the advantage of providing both standardised information about memory usage and normalised timing measurements.

Throughout this paper results of timing measurements will therefore be given in *Normalised CPU Seconds* (NCU-seconds) as provided by the CERN LSF batch facility, which also facilitates the comparison. The conventions changed at the beginning of 2005, but during the whole testing period one NCU-second roughly corresponded to one CPU-second on a 500 MHz PIII machine.<sup>1</sup> On the fastest machines available at the time when the tests were done (2.4 GHz PIV dual-processor machines, with 512 MB memory per CPU), the actual processing time was shorter by a factor of 3.3.

## 2 Performance Tests in Preparation of DC2

Intensive stress-testing of G4ATLAS with respect to performance and robustness started end of October 2003 and it was carried on at same intensity throughout the first half of 2004, until the start-up of DC2. This section gives an overview of those activities.

### 2.1 “Warming Up” — First Performance Studies with Rel. 7.1.0 and 7.2.0

This phase covers basically the first two months of testing activity, from its beginning until the presentation of first results at the ATLAS Software Workshop in December 2003.

After a huge development effort of almost three years, both in core framework and in sub-detector simulation, the successful integration of the individual subdetector simulation packages into G4ATLAS, and the integration of G4ATLAS into the ATLAS software framework Athena, a GEANT4-based simulation suite for the whole ATLAS detector had become available. The goal of this first phase of larger-scale tests was to get an idea about the performance (wrt. timing and memory usage) of this new product, and, where possible, to compare it with the corresponding figures for GEANT3.

To facilitate the comparison between GEANT4 and GEANT3, it was agreed on reading in and processing the same generated input events in both simulation programs.<sup>2</sup> For this purpose, several 100k single-particle events ( $p_T = 50$  GeV muons, electrons and pions for

---

<sup>1</sup>Since 2005, a new NCU-second equals one kilo-SpecINT2000-second (kSI2k-second). The ratio (old NCU-second):(new NCU-second) is about 0.180:1 [4].

<sup>2</sup>Originally it was also intended to compare the tracking performances of GEANT3 and GEANT4 in the magnetic field on a track-by-track basis. Generating exactly the same sequence of single particles on-the-fly within GEANT3 and GEANT4, using built-in single-particle guns, could have been problematic because of the different random-number generators. This potential difficulty could have been avoided through the usage of pre-generated single-particle events.

various geometrical configurations, like within the whole detector acceptance, barrel/end-cap regions only, calorimeter acceptance, etc.) were generated with a single-particle gun in HepMC format and persistified via ROOT. A large variety of generator HepMC/ROOT files for whole physics events had already been produced for DC1 and were ready to be used.

At this early stage of G4ATLAS performance tests, it was agreed that no persistent output (i.e., ROOT files with hits and generator truth information) was written. This decision was also due to the upcoming migration to POOL persistency, which would soon have made results based on ROOT obsolete.

In releases 7.1.0 and 7.2.0, the underlying GEANT4 version was v5.2, with slight ATLAS-specific adaptations (v5.2.ref04). The G4ATLAS physics list chosen in these tests was the default one, `Atlas_Physics`, which was based on the GEANT4 list `ExN04`<sup>3</sup>. Although new GEANT4 physics lists with improved hadronic physics (QGSP, LHEP) were just about to become available, it was decided to stick to the outdated `Atlas_Physics` list, and use its performance as a reference for future tests.

In addition to the measurements taken by LSF (see above), also tools available within G4ATLAS were used: `G4SimTimer`, which provides timing measurements and averages after each processed event and at the end-of-run; and `MemorySnooper`, a tool to monitor the current size (virtual memory usage) of a job, which can be invoked at various stages during job execution (begin-of-run, end-of-run, begin-of-event, end-of-event) through environment variables. Consistency checks showed that LSF and G4ATLAS measurements were compatible: the size reported by `MemorySnooper` corresponds roughly to the *Swap* quantity given by LSF; and the times measured by `G4SimTimer` agree with the NCU time (modulo the known LSF normalisation factor of the individual machine).

In the first round of tests with release 7.2.0, measurements with `MemorySnooper` were also taken at several places during the G4ATLAS and GEANT4 initialisation phase. Results of these and later measurements are summarised in Table 1.

### 2.1.1 Encountered Problems

In the out-of-the-box configuration of G4ATLAS which was available at the beginning of the performance tests, there was a sizeable failure rate (i.e., jobs crashed or were aborted). This was no surprise, because the subdetector simulation packages had been tested only individually and had been assembled and integrated only shortly before. It was also the first time that larger-scale performance tests, based on a statistics of thousands of single-particle events, were done, which made it also possible to detect rarely occurring problems.

#### Geometry-Related Problems

- Probably due to small volume overlaps or problems in the surface definitions of the LAr EM Barrel ‘tailored’ version (an optimised geometry description specially developed for fast testing purposes), the affected volumes being

```
LAr::EMB::Tailored::ThinAbsorber
LAr::EMB::Tailored::ActiveLiquidArgon
LAr::EMB::Tailored::Electrode
```

---

<sup>3</sup>A customised, user-defined physics list as in the GEANT4 `examples/novice/N04` package (until v6.2).

Processing Step	Proc. Size @ Run-Time
G4 initialisation (application alone)	50 MB
Building ATLAS envelopes	0.9 MB
Building ID Geometry and Sensitive Detector	2.2 MB + 0.3 MB
Building LAr Calorimeter	83.7 MB
Building Tile Geometry and Sensitive Detector	0.6 MB + 0.3 MB
Building Muon Geometry and Sensitive Detector	8.0 MB + 0.2 MB
Geometry optimisation	30 MB
Magnetic field	12 MB
Loading of full (default) Physics Lists	49 MB
<b>External Dependencies</b>	
Reading generator events from ROOT files	27 MB
Declaring POOL in jobOptions (for hit persistification)*	$\sim$ 40 MB
ID GeoModel at initialisation (overhead)*	$\sim$ 18 MB
LAr hit calibration*	$\sim$ 100 MB
Using Oracle DetDescr database (due to Instant Client)*	$\sim$ 100 MB
Using MCTruth*	$\sim$ 10 MB

Table 1: The measured process size at run-time for various processing steps during G4ATLAS simulation run. The bulk of the measurements was performed on Rel. 7.2.0, using the ATLAS `MemorySnooper` tool. The items marked with an asterisk are from later measurements (done up to recently) and added for completeness.

tracks (mainly  $\nu_\mu$ ,  $\bar{\nu}_\mu$ ) got trapped in the boundary regions of adjacent volumes, which resulted in a long process of (almost) zero-length stepping.

Although most of the particles eventually managed to escape the problematic regions and most of the jobs succeeded, the pathologically long execution times for these individual events were responsible for unexpectedly high average processing times per (single-electron) event (about 10 times longer than for a GEANT3 event).

The problem turned out to be related with the tailored version only, because it disappeared when the so-called ‘*standard*’ or ‘*complete*’ version was taken instead. Therefore the tailored version was abandoned (there was no manpower to maintain it and develop it further), and the standard version was chosen as a default.

- Incompatibilities in the ATLAS and Muon System envelope definitions, such that the Muon System envelope was not fully contained in the global one, led to sporadic crashes. Particles escaped the envelopes and were tracked up to kilometres to the outside; the crashes finally took place when secondaries created outside the ATLAS boundaries were tried to be propagated.

A simple redefinition of the Muon System envelopes made the problem disappear.

## GEANT4-Related Problems

- The usage of outdated hadronic physics (physics list `Atlas_Physics`, see above) led to rare abortions of single-pion jobs by GEANT4:

```
G4ReactionDynamics::NuclearReaction: inelastic reaction kinematically not possible
*** G4Exception: Aborting execution ***
```

It was decided to ignore these problems and not to track them down further. As mentioned above, `Atlas_Physics` was anyway to be replaced by new, improved physics lists in the next round of performance tests.

- A combination of geometry and GEANT4 problems whose effects were difficult to entangle: geometry problems (i.e., small volume overlaps as described above; here they occurred in the beam-pipe) caused particles to miss the detector boundaries and reappear at the outside; these particles were propagated again throughout kilometres. Eventual termination, however, did not occur through a crash like in the above case, but through LSF itself, after the job size had continuously grown and finally reached a maximum allowed value of 1.5 GB. This strange behaviour was found to be due to a mistake in GEANT4 navigation in magnetic field: millions of secondaries were produced during the propagation process and added to the memory, hence the continuous increase of memory.

After running tests of  $\mathcal{O}(10000)$  single electron and pion events without magnetic field, where no problems of this kind showed up, the problem was reported back to the GEANT4 group. A fix to the `G4Navigator` which was provided soon afterwards was implemented into G4ATLAS after Rel. 7.2.0, so did not come into effect any more in this first phase of performance tests.

- Memory allocated by GEANT4 to new track stacks (e.g. when massive EM showers occur) did not get freed any more, which resulted in a step-like increase of the job size (cf. Figure 1). This is rather a ‘feature’ of GEANT4 than a problem; it did not lead to crashes, but could result in a slow-down of job execution due to swapping, once the job size had become too big.

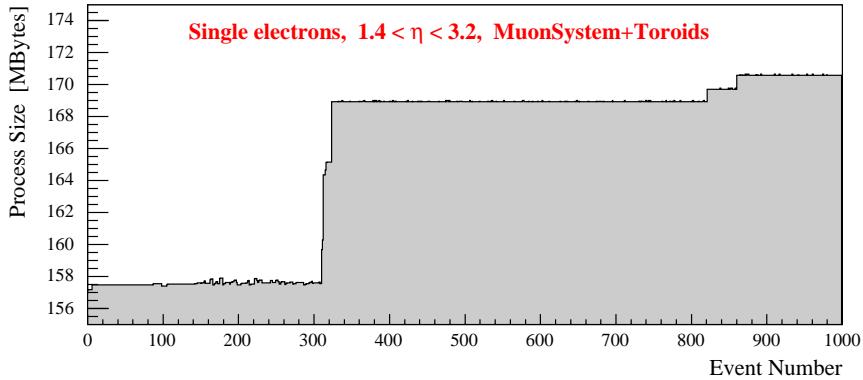


Figure 1: The memory usage monitored during a G4ATLAS simulation job of 1000 single-electron events ( $p_T = 50$  GeV,  $1.4 < \eta < 3.2$ ), run with Rel. 7.2.0 in the configuration “Muon System and Toroids only”. The step-like increase of process size is a feature of GEANT4 memory allocation: memory required for new track stacks, once allocated, is not released any more. In this (unrealistic) example, electrons are dumped into the EC Toroid coils, where they undergo massive showering.

### 2.1.2 Results

At the end of this first exercise, after having fixed some of the problems described above, G4ATLAS was already in a quite stable, robust and performant state. Several 100k single-muon events, about 100k single-electron and single-pion events each, and several thousand full physics events in different configurations could be run without crashes. Execution times were stable and under control. Compared to GEANT3, with all the necessary caveats (GEANT4 and GEANT3 simulation were run in different contexts and within different scopes<sup>4</sup>), the ratios between G4ATLAS and GEANT3/atlsim processing times per event were about 1.5 for single muons and 1.8 for single electrons and pions. Memory requirements were comparable: about 450 MB both for G4ATLAS and GEANT3/atlsim.

## 2.2 Achieving Production Quality — Performance Studies with Rel. 7.5.0

The second series of performance tests started end of December 2003 and lasted about two months. Results were presented at the ATLAS Software Workshop beginning of March 2004.

There were two main goals of this phase: to test and establish the new POOL persistency mechanism, and to test the new GEANT4 physics lists; both of which had been implemented and become available for ATLAS testing towards the end of 2003. Apart from these two major changes, the configuration, setup, used tools, etc. were similar to the ones in the first phase of performance tests described in Section 2.1.

POOL [5] is the persistency model developed by LCG and adopted by ATLAS. In ATLAS, POOL persistency replaced plain ROOT, and a combination of ROOT and ZEBRA which had been co-existing for roughly a year<sup>5</sup>. In the context of G4ATLAS simulation for DC2 and beyond, POOL (through its ATLAS-specific interface and implementation **AthenaPOOL**) is used both for reading in generator HepMC/POOL events and for writing out the produced SimHits. The necessary **AthenaPOOL** converters had become available already before release 7.5.0, together with the individual **AthenaPOOL** functionalities to *either* read from *or* write to POOL files. The *combined* functionality for simultaneous reading and writing became possible only with release 7.5.0, the necessary skipping of generator input events only as patch shortly after release 7.5.0.

In preparation of the G4ATLAS tests, generator HepMC/POOL input events had to be produced. Single-particle events were generated through **ParticleGenerator**, physics events were taken from already existing DC1 ROOT samples and converted to POOL. In total about six million single  $\mu^\pm$ ,  $e^\pm$ ,  $\pi^\pm$  events (with  $p_T$  ranging from 1 GeV to 2 TeV (for muons up to 10 TeV), all within  $|\eta| < 3.2$ ), and 140k physics events (minimum bias;  $Z \rightarrow \ell\ell$  (where  $\ell = e, \mu, \tau$ ); di-jet events;  $H(130) \rightarrow ZZ^* \rightarrow 4\ell$ ; SUSY/SUGRA events) were eventually available.

The new GEANT4 physics lists to be tested (**QGSP\_GN**, **QGSP\_BERT**, **LHEP\_GN**, **LHEP\_BERT**, for details see [2]) make use of improved models for hadronic interactions (as compared to the outdated **Atlas\_Physics** list which was to be replaced). The physics validation of these lists

---

<sup>4</sup>As an example, G4ATLAS jobs did not write output, whereas GEANT3/atlsim jobs wrote hits and other information to ZEBRA files.

<sup>5</sup>The persistency model of DC1 simulation consisted in reading generator events from ROOT and writing the hits and digits produced by GEANT3/atlsim to ZEBRA.

had already been done before, (timing) performance tests within G4ATLAS remained to be done. After a small series of comparative tests it turned out that QGSP and LHEP lists gave comparable timings, and that the BERT (Bertini cascade model) versions were about 30% slower than their GN (gamma-nuclear) counterparts. Compared to `Atlas_Physics`, the usage of GN lists resulted in only 10% slower processing times per event. Following a suggestion by F. Gianotti, it was then decided to choose `QGSP_GN` as the default.

With release 7.5.0, also the underlying GEANT4 version changed (v6.0 replaced v5.2).

### 2.2.1 Encountered Problems

The out-of-the-box configuration of G4ATLAS as in release 7.5.0 did not work satisfactorily. The original job failure rate was about 10%. A couple of problems could be identified and, like in the previous round of performance tests (cf. Section 2.1), tracked down both to bugs in the ATLAS-specific geometry implementation and to ones in GEANT4 proper.

#### Geometry-Related Problems

- The implementation of the complex toroid magnet geometry makes heavy use of boolean operations (mainly subtractions, using simpler base shapes). It occurred frequently that runs had to be aborted because particles got caught in the end-cap toroid region, ‘ping-ponging’ between the surfaces of two vessel segments. This did happen, however, *outside* the toroid proper, involving only the ‘virtual’ volumes used for *boolean subtraction*. This behaviour was not observed before (with GEANT4 v5.2) and most likely due to a changed, less tolerant treatment of intersecting or adjacent volumes within GEANT4 v6.0.

This problem could be fixed by re-doing the toroid geometry implementation, by using larger, more generous subtraction shapes than originally foreseen.

- Crashes in the course of the GEANT4 navigation process (involving `G4Navigator`) could be tracked down to new incompatibilities in the ATLAS and Muon System envelope definitions (as already described in Section 2.1). The problem could again be fixed by a redefinition of the envelopes.

#### GEANT4-Related Problems

- Segmentation violations in single-pion events, where the stack trace started with

```
G4StringChipsParticleLevelInterface::Propagate( ... )
```

sometimes happening in conjunction with

```
G4ExcitedStringDecay::EnergyAndMomentumCorrector - Warning
Scale not unity at end of iteration loop: nan nan nan
```

The problem was reported back to the GEANT4 group (H.P. Wellisch). To provide more feedback, the affected jobs were re-run with increased verbosity, using a modified GEANT4 library. Thanks to that, H.P. Wellisch found a problem in dividing through the transverse mass (i.e., zero) for gammas (on-shell, in projectile-side diffractive events). After changes to the class `G4QGSMSSplittableHadron` and their implementation into G4ATLAS, the problem disappeared.

- Crashes with core dump involving “G4Quasmons”

```
*** G4Exception : 027
    issued by : G4QEnvironment::Fragment
*General CHIPSEException
*** Fatal Exception *** core dump ***
```

The problem was again reported back to H.P. Wellisch, who provided a fast fix.

- Runs on very-high- $p_T$  single pions ( $p_T = 2$  TeV,  $|\eta| < 3.2$ ) sometimes crashed with core dumps and the following diagnostics (example):

```
In src/G4EnergyRangeManager.cc, line 110:
====> GetHadronicInteraction: No Model found
Unrecoverable error for:
- Particle energy[GeV] = 11613.4
- Material = Gten
- Particle type = anti_proton
```

This foreseeable problem happened because the hadronic models in the physics lists are restricted to energies up to 10 TeV, except for the models involving pion, kaon and nucleon reactions.<sup>6</sup> Given the LHC beam energy of 7 TeV, these crashes were therefore not a GEANT4 problem, but rather a ‘feature’, and due to the chosen unrealistic scenario, where the pion energy exceeds 7 TeV for  $\eta > 1.9$ , and where secondaries might have energies of more than 10 TeV for  $\eta > 2.3$ .

### 2.2.2 Results

After having implemented all the fixes to the problems described above, the patched version of G4ATLAS (including patches both on top of ATLAS release 7.5.0 and of GEANT4 v6.0) turned out to be already in a production-like quality. It was run very stably and robust, without crashes, on more than 700k single-particle events (muons, electrons, pions within  $|\eta| < 3.2$ ) and about 36k full physics events (mostly in the challenging setup of  $|\eta| < 6$ ), simulating the full ATLAS detector geometry and response.

Table 2 summarises the processing times and event sizes for single-particle and full physics events, as obtained in this round of performance studies.

POOL persistency was successfully used for the first time, both for reading in generator events and writing out the simulated hits. Using POOL turned out to involve no significant additional CPU requirements; also writing persistent output via *AthenaPOOL* took a only negligible fraction of the total processing time (as expected). As shown already in Table 1, the ‘penalty’ on memory consumption due to POOL was about 40 MB.

## 2.3 Getting Ready for DC2

This section summarises the testing activities carried out during spring 2004, starting in March with offline releases 7.7.0 and 7.8.0, progressing through the various ‘bug-fixing’ steps in the dedicated DC2 release branch 8.0.X (X = 0, ..., 4), and leading to the finalisation

---

<sup>6</sup>According to H.P. Wellisch, the threshold could be pushed up to 15 TeV at most in full shower simulation.

Evts.	Event Type	Time/Evt. [s]	Evt. Size [kB]
40000	single muons, $p_T = 5 \text{ GeV}$ , $ \eta  < 3.2$	1.3	11
40000	single muons, $p_T = 50 \text{ GeV}$ , $ \eta  < 3.2$	3.4	12
40000	single muons, $p_T = 200 \text{ GeV}$ , $ \eta  < 3.2$	9.1	14
20000	single electrons, $p_T = 5 \text{ GeV}$ , $ \eta  < 3.2$	28	15
20000	single electrons, $p_T = 50 \text{ GeV}$ , $ \eta  < 3.2$	275	24
20000	single electrons, $p_T = 200 \text{ GeV}$ , $ \eta  < 3.2$	1088	41
20000	single pions, $p_T = 5 \text{ GeV}$ , $ \eta  < 3.2$	14	18
20000	single pions, $p_T = 50 \text{ GeV}$ , $ \eta  < 3.2$	134	36
20000	single pions, $p_T = 200 \text{ GeV}$ , $ \eta  < 3.2$	562	72
5000	minimum-bias, $ \eta  < 6.0$	640	910
5000	$Z \rightarrow e^+ e^-$ , $ \eta  < 6.0$	2207	2489
5000	$Z \rightarrow \mu^+ \mu^-$ , $ \eta  < 6.0$	1718	2499
5000	$Z \rightarrow \tau^+ \tau^-$ , $ \eta  < 6.0$	1916	2569
5000	$H(130) \rightarrow ZZ^* \rightarrow 4\ell$ , $ \eta  < 6.0$	2446	2691
5000	SUSY/SUGRA, $ \eta  < 6.0$	2177	2569
5000	di-jets, $E_T(\text{hard}) > 17 \text{ GeV}$ , $ \eta  < 6.0$	1877	2631
1000	di-jets, $E_T(\text{hard}) > 17 \text{ GeV}$ , $ \eta  < 2.7$	549	2154

Table 2: Processing NCU times and event sizes for single-particle and full physics events, obtained with Rel. 7.5.0. All samples were simulated within the full ATLAS detector and with magnetic field switched on. Output files are located at `/castor/cern.ch/atlas/project/dc2/preprod/g4sim750`, (tar'ed) logfiles at `/castor/cern.ch/user/a/atlsprod/DC2_preproduction`.

of the G4ATLAS version to be used in DC2 simulation production (release 8.0.5) in late spring 2004. The individual tests performed during this period were all of relatively small scale, and there was rather a more-or-less continuous testing process, checking and trying out new features when they became available.

The main purpose of the tests with releases 7.7.0 and 7.8.0 was to introduce and evaluate the new *Geo2G4* mechanism, which builds the GEANT4 geometry tree from the one implemented within the ATLAS detector description package *GeoModel* — an approach which clearly has the advantage of concentrating on one, single source of geometry description, thus avoiding duplicated development and maintenance effort and the danger of inconsistencies between the two independent versions. Until release 7.7.0, the GEANT4 geometry had been built directly within the G4ATLAS FADS/Goofy framework. A GeoModel description of the Inner Detector became available with release 7.7.0, one of the Muon System with release 7.8.0. During DC2, the calorimeters (LAr and Tile) still used the direct, ‘hand-coded’ way of describing and building their GEANT4 geometries.

The purpose of the tests carried out within the 8.0.X release branch obviously was to arrive at a usable, stable and robust version of G4ATLAS for DC2.

Since release 7.5.0, G4ATLAS had been based on GEANT4 v6.0. When applying this version in the context of ATLAS Combined Testbeam simulation, it turned out, however, that its new multiple-scattering model gave unrealistic, too large predictions of this effect. Soon after GEANT4 v6.1 came out, which was supposed to cure the multiple-scattering problem, this version was also implemented in G4ATLAS; this happened together with offline release 8.1.0, and was also tried in parallel in the dedicated DC2 release branch, starting with release 8.0.1. Due to a serious problem with the muon pair-production model (`G4MuPairProduction`) in GEANT4 v6.1 (job failure rates of  $\mathcal{O}(20\%)$ ; for more details see below), for DC2 production it was decided to return to a combination of GEANT4 v6.0 (whose stability and robustness had already been shown in the previous series of tests with release 7.5.0; cf. Section 2.2) and the multiple-scattering model as used in GEANT4 v5.2.ref04 (which did not have problems).

### 2.3.1 Encountered Problems

Besides the above-mentioned muon pair-production problem in GEANT4 v6.1, there were only minor G4ATLAS geometry problems which could be fixed quickly.

Listed below as “geometry-related problems”, there are also two examples which are intended to illustrate the often tedious and difficult process of identifying, tracking down, disentangling and ruling out various sources of problems. In both cases, by the way, the observed behaviour (increase of execution time after introducing the Inner Detector and Muon System GeoModel descriptions) eventually turned out to be not strictly a problem, but rather a ‘feature’ of the used testing configurations.

#### Geometry-Related Problems

- First tests with release 7.7.0, using the GeoModel description of the Inner Detector geometry and its ‘translation’ to GEANT4 via `Geo2G4`, and switching off all detectors but the Inner Detector, indicated an average increase of the execution time per event by  $\mathcal{O}(30\%)$  as compared to release 7.5.0. Subsequent runs with release 7.7.0 using GeoModel and, alternatively, ‘hand-coded’ Inner Detector geometry gave compatible execution times, which therefore ruled out GeoModel as a cause of this behaviour.

A systematic series of checks could finally localise the problematic region to be in the Muon System: switching off the Muon System without modifying the detector envelopes accordingly, left just the bare toroidal magnetic field in the else empty Muon System envelope. After leaving the Inner Detector, particles were then propagated through the toroid field, which accounted for the extra processing time.

- First timing results obtained with release 7.8.0, where GeoModel was used both for the Inner Detector and the Muon System, gave up to  $\mathcal{O}(10\%)$  longer execution times for single particles, but a 50% increase for minimum-bias events, as compared to release 7.5.0. This therefore hinted at a problem in the forward region of the detector.

Finally it turned out that in the GeoModel default Muon System configuration of release 7.8.0, the passive material (toroids, feet) was not built, whereas this was done in release 7.5.0; so the compared configurations were not identical. The situation was very similar to the one described above: once the toroids (in particular the end-cap ones) are not present, particles have to be propagated through the strong magnetic field, which

is very time consuming. With material, most of those particles get dumped into the material (e.g. the coils), and are finally stopped.

- There also occurred a real geometry problem in the beam-pipe, where particles got stuck in some surface regions. The problem could be identified and fixed.

To have a better handle on geometry problems of this sort, a request to the GEANT4 group was made to have protection against zero-length stepping, when particles get trapped in surfaces. According modifications to GEANT4 tracking came into effect with GEANT4 v6.2.

## GEANT4-Related Problems

The main problem encountered in this phase of validation and performance studies consisted in crashes due to `G4MuPairProduction`, after the new GEANT4 v6.1 had been introduced.

The validation of GEANT4 v6.1 within G4ATLAS started with release 8.0.1, by simulating about 5000 full physics events (SUSY/SUGRA, minimum-bias,  $Z \rightarrow e^+e^-$ ,  $Z \rightarrow \tau^+\tau^-$  events). Out of the about 100 run jobs,  $\mathcal{O}(20\%)$  failed due to segmentation violation. The stack trace pointed at a problem in `G4MuPairProduction`, which was reported to the GEANT4 group (through J. Apostolakis).

In an attempt to collect and provide more information about the problem, it turned out that the crashes were usually not reproducible (i.e., re-running the job usually led to successful completion). Temporary instabilities in the run environment or network (e.g. AFS problems) could soon be ruled out (the same failure rate was observed in tests run at LBL, which did not rely on AFS access to CERN). The focus then turned towards reproducibility tests: since there were no known problems with reproducibility in GEANT4, potential problems could lie either in its ATLAS-specific usage through G4ATLAS or in other, non-GEANT4-related ‘modules’ (e.g. Athena, POOL, …). A test suite was then launched, involving the running of *valgrind* to spot memory leaks, the checking of the random-number generator status at various times during the job (e.g. begin-of-run, begin-of-event, end-of-event, end-of-run), the re-running of jobs for several times using the same random seeds, etc.

The irreproducible configuration finally turned out to be not the cause of the crashes. The reason for the irreproducibility could unfortunately not be found and has remained an open issue.

In addition it was also tried to spot possible problems in the `G4MuPairProduction` model itself. For that purpose, the source code was modified to re-introduce a protection which had been there in a previous version (without success: the problem remained), and to increase the verbosity of diagnostics printout (provided by V. Ivantchenko). It turned out that the problem always occurred in compound materials (like `FCal2MixAbsorber` in the LAr calorimeter), and this finding helped the author of the `G4MuPairProduction` model, V. Ivantchenko, to identify and fix problems in the code (in particular, fixes to a formula calculating  $t_{\max}$  for compound materials). This made the problem eventually disappear.

The fix of the `G4MuPairProduction` problem unfortunately came too late for DC2: it had been decided already before to return to GEANT4 v6.0 (with the multiple-scattering model from v5.2.ref04) in the 8.0.X release branch dedicated to DC2 production.

### 2.3.2 Results

Resulting from all the tests outlined and discussed above, finally a stable, robust and performant version of G4ATLAS could be released for large-scale DC2 simulation production.

New functionalities could be successfully integrated into G4ATLAS: the mechanism of building the GEANT4 geometry tree from the GeoModel one through `Geo2G4` passed the performance tests, both for the Inner Detector and the Muon System. No significant differences in execution times compared to ‘hand-coded’ GEANT4/FADS/Goofy geometry could be detected.

## 3 G4ATLAS Performance in DC2

The main purpose of DC2 was to test essential parts of the ATLAS Computing Model, targeting at the so-called *Tier-0 Exercise*, whose goal in turn was to set up and simulate the conditions corresponding to one day of ATLAS data-taking at the LHC. In a preparatory data production phase (DC2 ‘Phase 1’), which was carried out as a world-wide effort in a distributed way, about ten million full physics events had to be generated, simulated with GEANT4, superimposed with minimum-bias background at high-luminosity (‘*pile-up*’), digitised, mixed (‘*event mixing*’) and finally replicated to CERN. In the Tier-0 exercise proper, these events were reconstructed and the produced ESD’s (Event Summary Data) and AOD’s (Analysis Object Data) shipped to remote Tier-1 centres.

Distributed production in ‘Phase 1’ was mainly done on the Grid, involving three different flavours (LCG, Grid3, NorduGrid), and making use of a (semi-) automated production and data management system. In particular the simulation production was fully run on the Grid(s) in a distributed way.

DC2 ‘Phase 1’ was executed during the second half of 2004. A first, scaled-down attempt of a Tier-0 exercise was run in fall 2004, the final exercise in the first quarter of 2005.

Despite of DC2 being mainly a *computing* exercise, the composition of the about ten million events needed for the Tier-0 exercise was carefully discussed and chosen in order to serve also the needs of the various physics groups; for details on the sample composition see [6]. In addition, and outside the scope of DC2 proper but taking advantage of the already set-up production environment, also about two million events of ‘calibration samples’ [7] were produced; they had been requested by several subdetector groups to validate their software.

Thus in total about 12 million events had to be passed through full detector simulation within G4ATLAS, almost all of them within the full detector acceptance  $|\eta| < 6$ . Concerning simulation, this was the largest, most demanding exercise run in ATLAS until then.

From the perspective of G4ATLAS and GEANT4, DC2 ‘Phase 1’ was a striking success. In total, the production ran smoothly, stably and without major problem. As an impressive example, the NorduGrid team was able to complete the full sample of one million  $Z \rightarrow e^+e^-$  events, which had been assigned to them, without failure.

For the whole about 12 million events, only one crash due to GEANT4 hadronic physics (in `G4QEnvironment`) was recorded by Grid3. Here is an excerpt from the diagnostics output issued before the crash:

```

Exception thrown passing through G4ChiralInvariantPhaseSpace
  targetPDGCode = 90018022
  Dumping the information in the projectile list
  Incoming 4-momentum and PDG code of 0'th hadron:
(4.6638e-15,1.87567e-15,22.9108;22.9108) 22
In src/G4QEnvironment.cc, line 3728:
===> *Fragment:Exception Up to Hadronics
  G4HadronicProcess failed in ApplyYourself call for
  - Particle energy[GeV] = 0.0229108
  - Material = LiquidArgon
  - Particle type = gamma

*** G4Exception : 007
  issued by : G4HadronicProcess

```

The problem was reported to the GEANT4 group (H.P. Wellisch).

Tables 3 and 4 summarise results of performance measurements taken in September 2004 with Rel. 8.0.7, whose G4ATLAS version is identical to the one used in ‘Phase 1’ of DC2 (Rel. 8.0.5); these figures can therefore serve as references for G4ATLAS performance in DC2.

Configuration	Process Size	
	Max Swap (LSF)	MemorySnooper @ EOR
Full ATLAS detector (DC1 layout)	406 MB	397 MB
Full ATLAS detector, no $\mathbf{B}$ -field	386 MB	379 MB
Inner Detector only	213 MB	202 MB
LAr Calorimeter only	308 MB	299 MB
Tile Calorimeter only	171 MB	158 MB
Muon System (with Toroids) only	218 MB	206 MB

Table 3: The maximum job sizes of G4ATLAS simulation runs in various configurations, obtained with Rel. 8.0.7. As tools, the standard memory monitoring as provided by the CERN LSF batch system (“*Max Swap*”) and the ATLAS *MemorySnooper* (invoked at “*End-Of-Run*”) were used.

Evts.	Event Type	Time/Evt. [s]	Evt. Size [kB]
10000	single muons, $p_T = 50$ GeV, $ \eta  < 3.2$	4.4	11
1000	single electrons, $p_T = 50$ GeV, $ \eta  < 3.2$	228	23
1000	single pions, $p_T = 50$ GeV, $ \eta  < 3.2$	178	40
200	$Z \rightarrow e^+ e^-$ , $ \eta  < 5.0$	2282	1890
200	$H(130) \rightarrow ZZ^* \rightarrow 4\ell$ , $ \eta  < 5.0$	2192	2200
200	SUSY/SUGRA, $ \eta  < 5.0$	2546	1950

Table 4: Processing times and event sizes for single-particle and full physics events, obtained with Rel. 8.0.7. All samples were simulated within the full ATLAS detector. Note that the increase in execution time for physics events with respect to Rel. 7.5.0 (cf. Table 2) is mainly due to a modified G4 physics cut in the LAr (30  $\mu$ m instead of 1 mm).

The data produced during DC2, however, turned out later to be only of limited usefulness for physics studies. When the simulation code for mass production had to be delivered by release 8.0.5, development in the reconstruction area was still on-going; feedback could therefore not be provided to the necessary extent. One of the main problems encountered was the missing implementation of the *forward shielding* in the DC2 detector layout.

The problem became manifest because events in DC2 were simulated by default within the full detector acceptance  $|\eta| < 6$ : for  $|\eta| \geq 5$ , particles started hitting the beam-pipe, and the secondaries produced were scattered back into the detector. This effect resulted in a massive excess of hits especially in the Muon CSC, TGC and end-cap MDT chambers (see Figure 2), which spoilt the performance of (or, at least, posed a big additional challenge on) the Muon reconstruction programs.<sup>7</sup> The missing shields were then implemented with release 9.0.0.

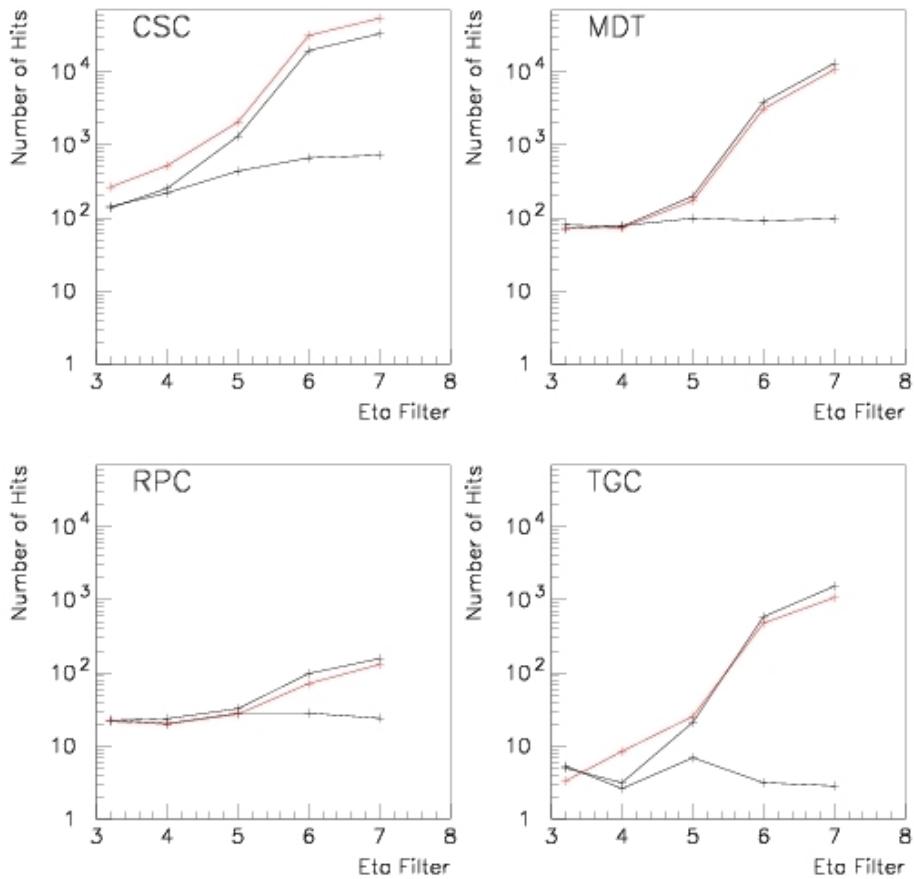


Figure 2: The numbers of CSC, MDT, RPC and TGC hits as a function of the simulation acceptance cut on  $\eta$ . The red curves are for DC2 (final) layout without shielding, the black ones for initial layout once with shielding (lower curve), once without (upper curve). As described in the text, the steep increase in the number of hits at about  $|\eta| \simeq 5$  in the configurations without shielding is due to secondaries back-scattering from the beam-pipe into the detector.

<sup>7</sup>There was no shielding available in DC1, either, but most of the production there was done with an acceptance cut of  $|\eta| < 3$ , so avoided the problem.

## 4 Post-DC2 Tests

This section describes some selected testing activities (out of many) in the period from summer to fall 2004, after the production release for DC2 simulation (8.0.5) had been delivered and DC2 production started, up to release 8.7.0. Testing went on more or less continuously, series of tests were launched once new functionalities had become available.

One important activity was related with validation of the new Monte-Carlo truth (**MCTruth**) within G4ATLAS, which provides information about secondary particles produced during the GEANT4 tracking process through the detector, updated information about the primaries, etc. for most of the relevant physics processes (including transition radiation in the TRT). After first successful tests with release 8.4.0, the GEANT4-related MC-truth information (originally called **G4Truth**), was merged with the one coming from the generator (**GEN\_EVENT**) to a unified description (**TruthEvent**), which facilitated the treatment of MC-truth related information for ‘down-stream’ clients (like reconstruction). The unified MC-truth became available with release 8.7.0.

The detector layout used in DC2 was the same as in DC1 (‘Complete Layout’). It was steadily updated and developed further by the Detector Description group, in particular in view of the start-up phase of ATLAS, where only an incomplete (‘staged’) detector will be in place. Validation of this so-called *Initial Layout* was another important post-DC2 activity (important also for the preparation of the simulation software for the ATLAS Physics Workshop in Rome in June 2005, whose main goal will be to evaluate the physics performance during the ATLAS start-up phase).

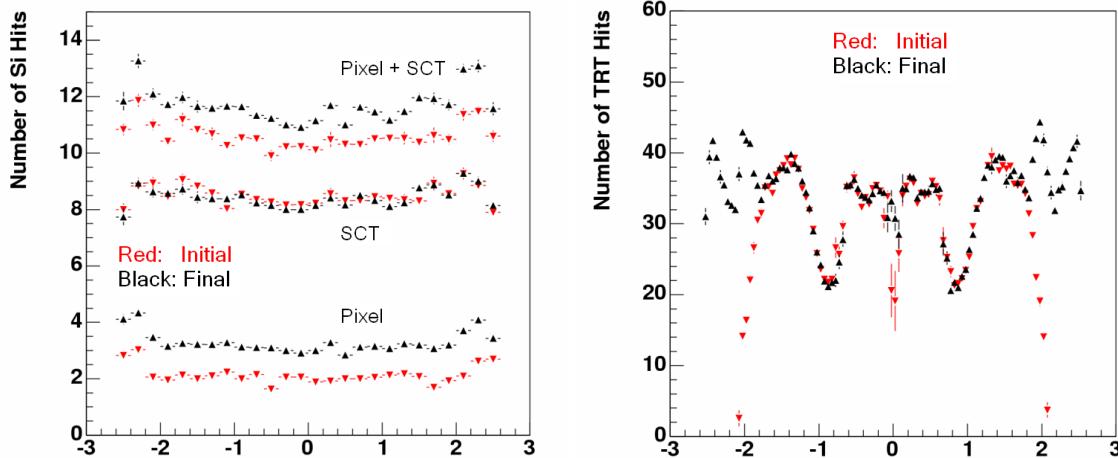


Figure 3: Results from Inner Detector studies on Initial Layout validation, performed with single-muon events ( $p_T = 50$  GeV,  $|\eta| < 3.2$ ). The events were simulated with Rel. 8.0.5, for both Initial and Final Layout, and reconstructed with Rel. 8.7.0. The left plot displays the number of Pixel and SCT hits, the right plot the number of TRT hits as a function of  $|\eta|$ , for both layouts. The effect of the missing Pixel layer (left plot) and of the missing TRT C-wheels (right plot) in the Initial Layout is clearly visible.

Figure 3 illustrates results from studies on Initial Layout validation for the Inner Detector. Figure 4 is taken from a series of studies done by F. Gianotti [8, 9] which combines both of

the above-mentioned activities (validation of MC-truth and the Initial Layout).

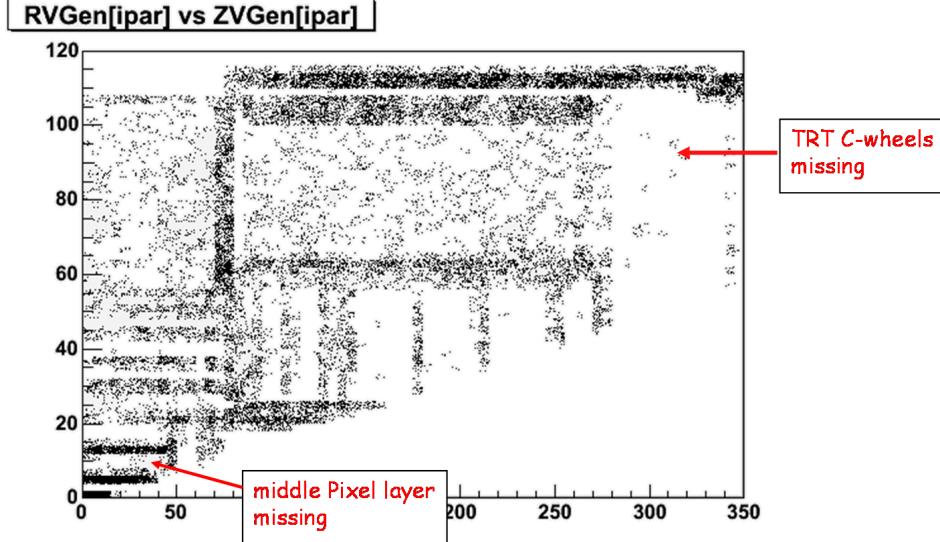


Figure 4: Results from Inner Detector studies on combined Initial Layout and G4ATLAS MC-truth validation, using 2000 single-electron events ( $p_T = 50$  GeV,  $|\eta| < 2.5$ ). The events were simulated with Rel. 8.0.7 and reconstructed with Rel. 8.7.0. The picture (“*Tracker Radiography*”) shows (in an  $r$ - $z$  representation) the vertices of secondary particles produced during the passage of primary electrons through the Inner Detector. The Initial Layout geometry (in particular the missing second Pixel layer and TRT C-wheels) is nicely outlined. (Plot by F. Gianotti, cf. [9].)

## 5 Tests and Validation of Testbeam Simulation

Development and validation activities in the Testbeam and Combined Testbeam (CTB) simulation areas went on in parallel and in close collaboration with the ones for ATLAS; the two ‘streams’ were of course not independent, since they are based on the same core G4ATLAS software.

Doing full justice to the activities in the TB and CTB simulation areas, by trying to give a comprehensive overview, would be beyond the scope of this paper. Most of them have been presented at meetings and material can be retrieved via the CERN CDS system; cf. [10]. Here only three activities can be picked out and described, intended to serve as examples: results from CTB performance studies, and from two physics-oriented validation studies in the Muon and Calorimeter (HEC) areas.

An extensive series of performance tests of CTB simulation software was carried out in the second half of 2004. CTB simulation described the complete setup as in the H8 testbeam area, consisting of Pixels, SCT, TRT, LAr and Tile barrel segments, the various Muon barrel and endcap chambers, magnets, etc. The performance with respect to timing and memory usage was carefully monitored and compared for three different releases (8.6.0, 8.8.0, 9.0.0). No peculiarities were encountered, the measured CPU times and memory usage per event stayed within expectations. For each of the three releases, 300k single-particle events (muons,

electrons, pions of various energies) were processed, which amounted to running 300 jobs à 1000 events each. All the jobs ran stably, no crash was observed. Results of these performance tests are displayed and summarised in Figure 5.

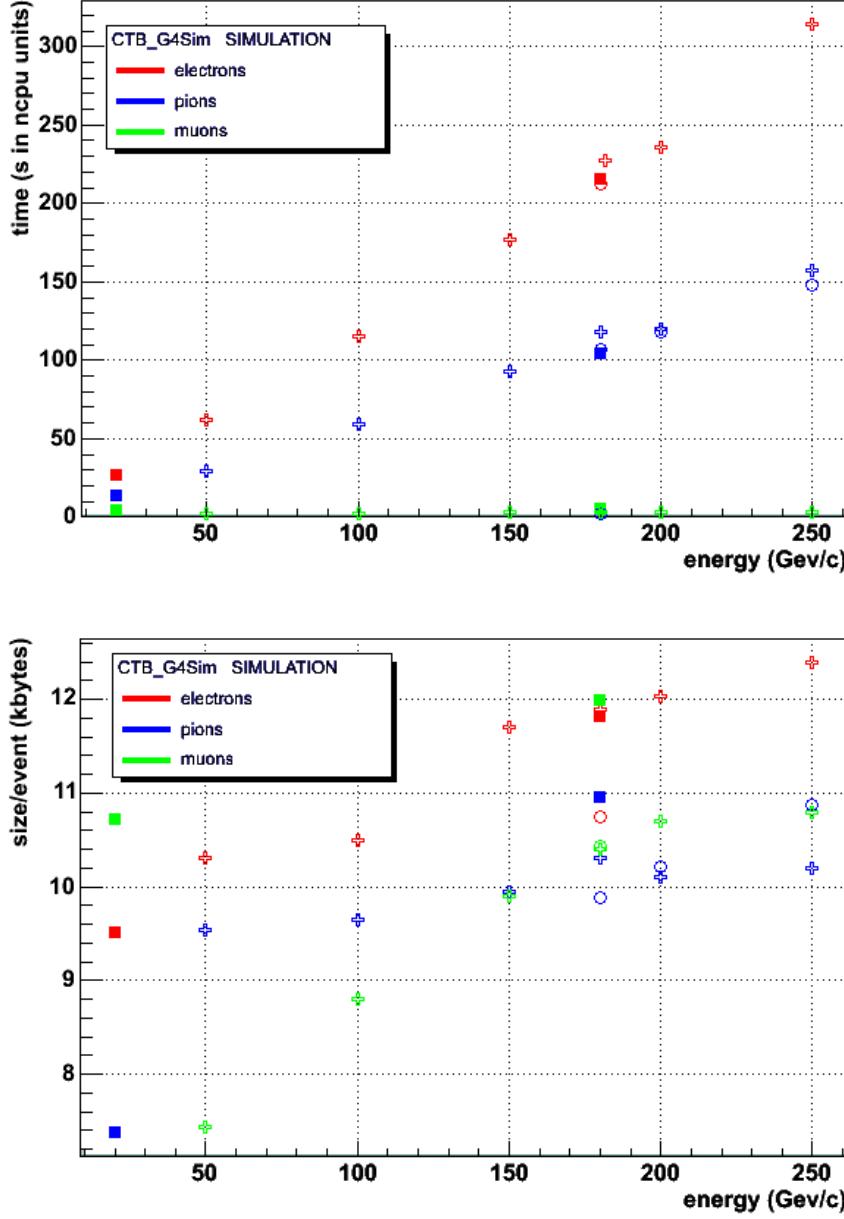


Figure 5: Results from CTB Simulation performance studies carried out with Rel. 8.6.0 (crosses), Rel. 8.8.0 (circles) and Rel. 9.0.0 (boxes), using single-particle events of various energies ( $20 \text{ GeV} \leq E \leq 250 \text{ GeV}$ ). The upper figure shows the average processing time per event in NCU-seconds, the lower one the average event size in kilobytes, as functions of the particles' energy. (Colour coding: red: electrons, blue: pions, green: muons.)

An illustrative example from the TB subdetector simulation area are the *track sagitta* studies in the Muon TB, which also provided valuable feedback both on G4ATLAS and GEANT4-

related issues. A detailed summary can be found e.g. in Ref. [11]. For each of the three barrel chambers BIL, BML, BOL in the TB setup, track segments were reconstructed, whose centres defined so-called ‘super-points’. The sagitta was then defined as the distance of the middle BML super-point to the straight line connecting the super-points in the outer chambers BIL and BML. The sagitta residual is, among other parameters, dependent on the intrinsic tube resolution, on multiple scattering, etc.

In first tests, carried out on 2002 TB data at the end of 2003, the sagitta residuals from simulation turned out to be much overestimated. Although there were known problems with GEANT4 multiple scattering (in GEANT4 v6.0, cf. Section 2.3; another problem was found in GEANT4 v6.2), they could be ruled out as the main source: fixing the GEANT4 problems only slightly decreased the sagitta residuals. After a long, careful study, in which the effects of every single constituent in the TB simulation setup on the sagitta was examined, the problem could finally be tracked down to a wrong material description in RPC support structures. Too high a density resulted in too much multiple scattering compared to the real TB configuration. Fixing the material density eventually led to good agreement between data and simulation; cf. Figure 6.

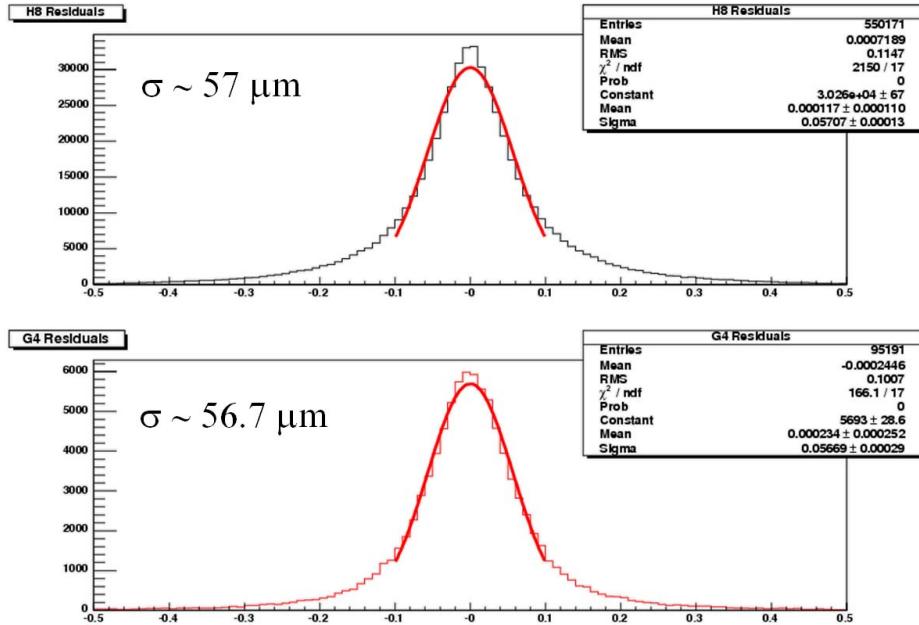


Figure 6: The sagitta residual for the CTB Muon barrel chambers (BIL/BML/BOL), obtained from data (top) and simulation (bottom). The whole study (simulation, digitisation, reconstruction) was performed with Rel. 8.8.0. See the text for more details. (Plot by D. Rebuzzi *et al.*, cf. [11].)

A second example from TB subdetector simulation are the studies done by the Hadronic End-Cap (HEC) calorimeter group at MPI in Munich on GEANT4 hadronic physics validation. They have been following the developments in GEANT4 hadronic physics lists since GEANT4 v5, monitoring the performance in comparison with testbeam data, and providing feedback to the GEANT4 Collaboration.

In the study presented here (cf. [12]) simulation results obtained both with GEANT3 and GEANT4 were compared to testbeam data. The versions of the simulation software were:

GEANT3 v3.21 (using GCALOR hadronic shower code); GEANT4 v5.2.p02 (using the physics lists LHEP v3.6, QGSP v2.7), GEANT4 v6.1 (with LHEP v3.7, QGSP v2.8), and GEANT4 v6.2.p02 (with LHEP v3.7, QGSP v2.8). Problems were found and reported in GEANT4 v6.1, and the already good performance of GEANT4 v5 could eventually be recovered in v6.2. There was particularly good agreement between data and GEANT4 on the  $e/\pi$  ratio (see Figure 7) and the pion energy resolution. Still some improvement on the GEANT4 hadronic physics models turned out to be necessary with respect to e.g. the energy resolution of electrons and the longitudinal development of hadronic showers.

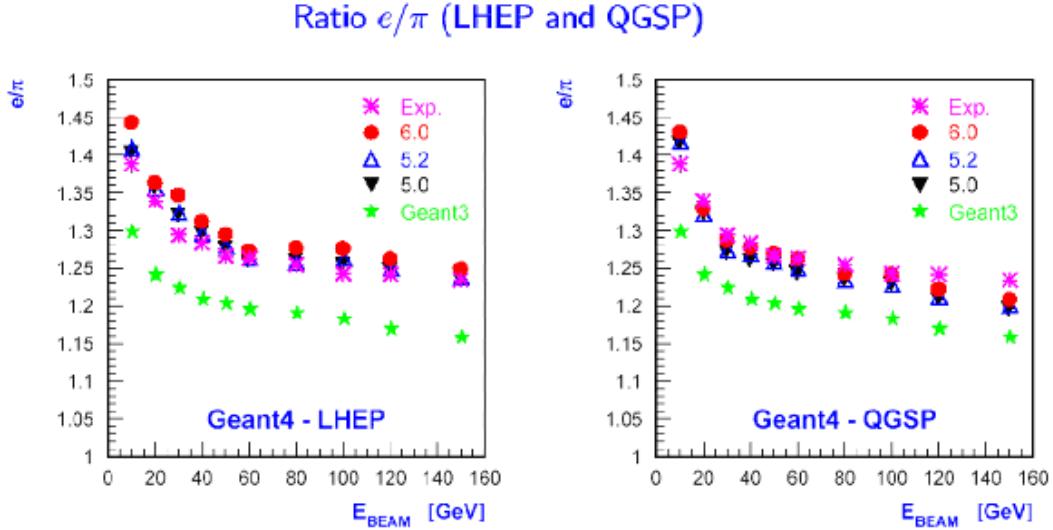


Figure 7: The  $e/\pi$  ratio in the HEC calorimeter measured in the testbeam as function of the beam energy, compared to simulation results obtained with different versions of GEANT4 (v5.2, v6.1, v6.2) and GEANT3 (v3.21). The two plots differ in the used GEANT4 physics list: LHEP in the left, QGSP in the right plot. Results from GEANT4 are in better agreement with data. (Studies by A. Kiryunin *et al.*, cf. [12].)

## 6 Miscellaneous Open Issues

As already mentioned in Section 2.3, there seems to be some inherent irreproducibility in G4ATLAS simulation runs, which has been first encountered during the preparation tests for DC2 and has never been fully understood. Studies done so far indicate that the first event of a job is reproducible, and that instability builds up only during the run. This might be due to yet undetected memory problems, problems in the interplay between GEANT4, G4ATLAS and external packages like POOL, etc. More systematic investigations on this issue will have to be done soon.

Further open issues which will need attention in the coming months are

- a careful investigation on the *memory requirements* of G4ATLAS jobs and (if possible) work on memory optimisation. The goal must be to bring the job size down to

about 500 MB, such that G4ATLAS jobs can also be run on relatively modest machines. Until DC2, memory usage met this requirement (cf. Section 2.3); in later checks with release 8.7.0, memory consumption rose to more than 600 MB, due to inefficiencies in the LAr geometry (which could be partly eliminated) and to the migration to Oracle as detector description database. Currently, in the running simulation production with release 9.0.4 for the 2005 Rome Physics Workshop, the memory requirements are at about 600 MB.

- the *timing optimisation of the tracking in the magnetic field*. This is not an urgent issue (all the timing measurements done so far do not indicate that tracking through the magnetic field is inefficient), but might become so in view of foreseen updates in the magnetic field description, where there are plans to increase the accuracy and detailedness.

## Acknowledgements

We would like to thank all the ATLAS core software developers for the robust, versatile and complete code provided; our colleagues working on subdetector simulation for their prompt implementation and validation of any new proposed functionality, both in G4ATLAS and GEANT4; and all the contact people from the GEANT4 Collaboration itself (in particular J. Apostolakis, J.-P. Wellisch, V. Ivantchenko, G. Cosmo and G. Folger) for their important role in the success of the G4ATLAS project.

Special thanks go to Fabiola Gianotti for her continuous interest in the project, her support and advice, and, in particular, her careful studies on MC-truth within G4ATLAS.

## References

- [1] The GEANT4 Collaboration (S. Agostinelli *et al.*), *GEANT4 — A Simulation Toolkit*, Nuclear Instruments and Methods in Physics Research, NIM **A** 506 (2003), 250–303.
- [2] GEANT4 webpage, <http://wwwasd.web.cern.ch/wwwasd/geant4/geant4.html>.
- [3] D. Barberis, G. Polesello and A. Rimoldi, *Strategy for the Transition from Geant3 to Geant4 in ATLAS*, Internal Note ATL-SOFT-2003-013 (Nov. 13, 2003).
- [4] [http://computingcourier.web.cern.ch/ComputingCourier/CNL\\_2004Nov/Batch-Time-Units.doc](http://computingcourier.web.cern.ch/ComputingCourier/CNL_2004Nov/Batch-Time-Units.doc).
- [5] LCG Applications Area/POOL webpage, <http://lcgapp.cern.ch/project/persist/>.
- [6] *Physics Generation for ATLAS Data Challenges* webpage, maintained by I. Hinchliffe, <http://www-theory.lbl.gov/~ianh/dc/dc2.html>.
- [7] *Software Validation* webpage, maintained by D. Costanzo, [http://atlas.web.cern.ch/Atlas/GROUPS/PHYSICS/SOFT\\_VALID/soft\\_valid.html](http://atlas.web.cern.ch/Atlas/GROUPS/PHYSICS/SOFT_VALID/soft_valid.html).

- [8] F. Gianotti, *First Look at G4 Truth*, presentation at the “ATLAS Physics Validation” meeting, CERN, Sept. 1, 2004;  
agenda <http://agenda.cern.ch/fullAgenda.php?ida=a043782>.
- [9] F. Gianotti, *More on the G4 Truth*, presentation at the “ATLAS Physics Validation” meeting, CERN, Sept. 15, 2004;  
agenda <http://agenda.cern.ch/fullAgenda.php?ida=a044013>.
- [10] ATLAS Combined Testbeam meetings,  
agendas at <http://agenda.cern.ch/displayLevel.php?fid=211>.
- [11] D. Rebuzzi, S. Rosati and G. Avolio, *Studies on H8 MDT Barrel Simulation*, presentation at the “V<sup>th</sup> Workshop on H8/X5 Muon Test”, CERN, Dec. 16–17, 2004;  
agenda <http://agenda.cern.ch/fullAgenda.php?ida=a044309>.
- [12] A. Kiryunin, D. Salihagić, P. Schacht and P. Strizenec, *Validation of Geant4 Version 6.2 with HEC Testbeam Data*, presentation at the “LAr Software and Performance” meeting, CERN, Nov. 17, 2004;  
agenda <http://agenda.cern.ch/fullAgenda.php?ida=a045052>.