ATL-SOFT-CONF-2007-001
09 October 2007

ATLAS Simulation readiness for first data at LHC

A. Dell' Acqua [(1)], A.Di Simone [(2)], M.V. Gallas [(1)], A.Rimoldi [(3)], V. Tsulaia [(4)], I. Ueda [(5)], S. Vahsen [(6)], Z. Marshall[(7)]
[(1)] CERN
[(2)] CERN & CNAF
[(3)] University of Pavia & INFN
[(4)] University of Pittsburgh, USA
[(5)] University of Tokyo
[(6)] LBNL, Berkeley (USA)
[(7)] Columbia University (USA)

E-mail: adele.rimoldi@cern.ch

Abstract. The commissioning phase for the ATLAS experiment, in preparation for the new LHC machine to switch on, has presented challenges to nearly every aspect of the software development. The ATLAS simulation program, as a part of this phase, is now operational and fully functional within the ATLAS common software framework, Athena. The latest developments are directed towards enhanced versatility to cope with the increasing needs of developers and users and ease of use for the large ATLAS community, now with more than 2000 potential users. Emphasis in this talk is on recently added functionality recently added, validation and production strategy, and improved robustness and maintainability.

1. Introduction

A common framework for event processing, Athena [1], is the basis for all ATLAS applications. An application is a set of services and algorithms assembled and configured at runtime, steered using jobOption scripts written in Python, an Object Oriented scripting language which adds interactivity to the primarily C++ based applications. It is simple and intuitive, robust when used in an interactive manner, and allows introspection mechanisms so that any user can interrogate the object about type and internals.

The simulation program for ATLAS has been developed in this environment. The main results, recent developments, the validation and production strategies in terms of performance figures, and robustness and maintainability are summarized in this paper.

2. G4Atlas

The application used in ATLAS to setup the simulation is named G4Atlas. It is the only application required and supported by the experiment for simulation, and is written entirely in C++. This application is a full featured OO GEANT4 simulation suite based on dynamic loading and action on demand, so that all user requested functionality is added using plug-in modules.

The Python application that sets up the appropriate conditions to run the simulation is called PyG4Atlas, and its role is to add flexibility for configuring the different setups, interactivity, and introspection for all settings needed at runtime.

With these tools we are able to handle all daily user requests and to set different geometry configurations by setting parameters at runtime with no code manipulations.

The robustness of the resulting applications has been proven in many Grid productions with negligible failures rates. Table 1 shows the amount of data simulated in big productions since 2004.

The different geometry setups implemented are handled similarly: the full ATLAS detector, all available cosmic-ray setups, and the combined test beam geometries are simultaneously available and usable for immediate simulation purposes.

Table 1: ATLAS event productions

Year	Millions of Events	Production Type (full simulation)
2004	12	Large scale production (DC2)
2005	4 + 8.6	Combined Test Beam for performance studies + ATLAS workshop production
2006	~1	Test productions on Grid
2007	~1	Test productions on Grid, ongoing

Consistency and functionality are maintained throughout all the applications so that the user can switch among them with minimal effort. A non-ideal detector description is available to describe the geometry for the detector as installed, introducing misalignments, extra material, services, etc. Algorithms and tests tools are also in place (e.g. G4AtlasTest application) to access the detector "hits", to perform material scans and to allow computations of radiation and interaction lengths along a selected slice of the detector [2].

3. ATLAS Detector Description

The description of the complex geometry of ATLAS is decoupled from the simulation framework (G4Atlas), and two hierarchical trees are present in memory at the same time ("GeoModel" and "Geant4). GeoModel provides a transient geometry representation built from primary numbers and alignment constants. The database solution adopted is Oracle and versioning is in place. As a consequence, the simulation, digitization, and reconstruction applications all use the same geometry built at runtime.

The GeoModel description is optimized for a large numbers of volumes (~10^6) with extensive use of parameterized volume-based solutions. In the initialization phase, this geometry is translated into the GEANT4 geometry and placed into resizable and moveable GEANT4 envelopes. Despite major optimization, the total amount of memory required currently exceeds 90 MB. The single contributions from the different detectors are shown in Table 2.

Table 2: breakdown of ATLAS memory allocation at runtime for the different subdetectors (MB)

Subdetector type	Memory consumption (MB)
Pixel	5.6
SCT	9.1
TRT	3.1
Inner Detector material	1.0
LAr	54.4
Tile	1.1
Muon System (including toroids)	21.3

GeoModel is also used for the Combined Test Beam description (2004 setup). From 2006 onwards, since the past productions were carried out using ideal detectors with nominal positions, the new productions include a revised description of the detector "as installed." All information about detector deformation is imported into GeoModel with time variation associated to the run number. Presently three flavours of detector descriptions are simultaneously available: the ideal detector, the misaligned detector, and the misaligned detector with material distortions. In addition special descriptions for the commissioning and combined testbeam setups are available. The simulation application can deal with the different configurations with setting at runtime.

4. G4Atlas applications

The simulation framework itself offers a set of pre-configured applications for full ATLAS simulation, combined test beam, cosmic ray setups and old standalone test-beams. For each application several layouts are available so that subdetector-specific studies and user customizations can be easily achieved from the pre-configured applications. These applications are exercised nightly in automatic tests. Feedback from users as well as improvements and new features are all included in the preconfigured applications and they are maintained centrally for the community [3].

Example 1 – Cosmic Ray simulation

Since 2006 priority has been put in ATLAS on the cosmic ray commissioning data. Full support for the cosmic ray simulation is in place, from the description of the experimental area (rock overburden and surface buildings) to primary cosmics using a dedicated CosmicGenerator able to produce cosmic muons [4]. Each detector envelope is used as a scoring layer so that particles at its entrance are recorded. The most external envelope (Muon System) saves particles propagated through the rock overburden before entering the ATLAS detector so that at the next loop the simulation could be restarted from that point.
The ATLAS cavern description, with shafts and muon system, is completely described by the simulation application.

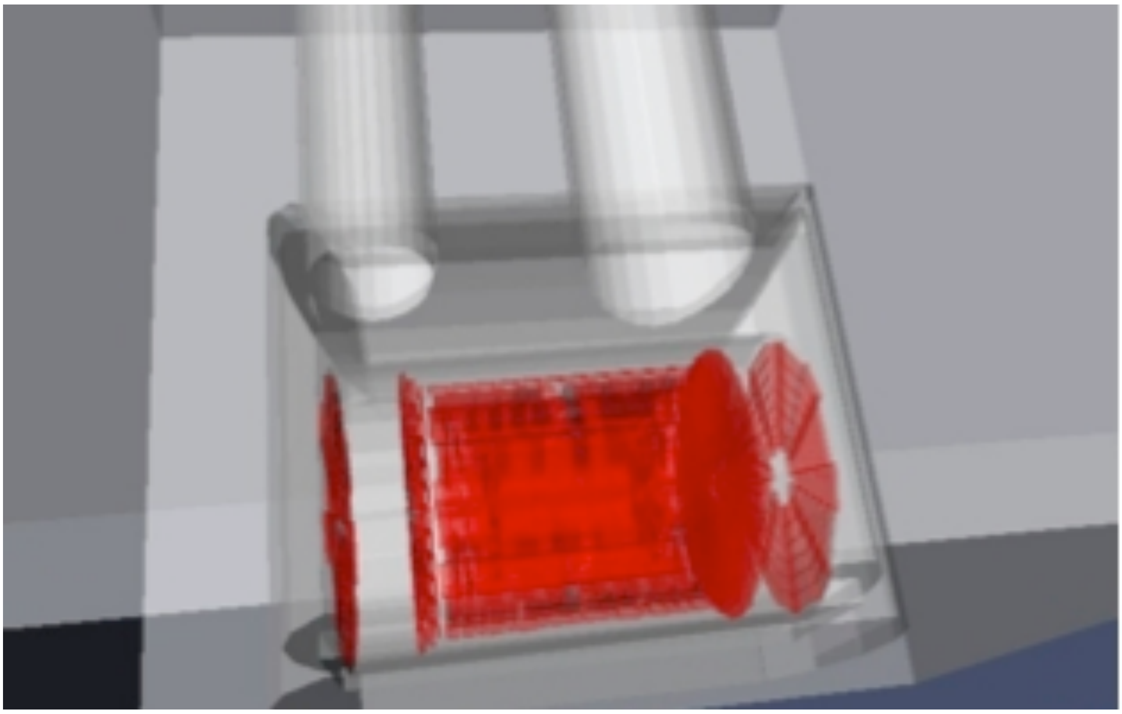


Figure 1: 3D view of the ATLAS “cavern” setup. The muon system is positioned inside the ATLAS cavern

Example 2 – Combined Test Beam

The Combined test Beam environment is a big and natural source for performance studies and physics validation at the LHC (Figure 2). It has been completely simulated with all active and passive components in place.
The simulation infrastructure deals with all the following different configurations:

- Combined mode
- Photon beams
- Material studies
- Pseudorapidity scans
- Calibration
- Ancillary detectors

In the data-taking period (24 weeks in 2004) the layout had frequent and sudden evolutions; simulation of these different and time-dependent layouts was handled by specifying the run number when needed. Single particle generator was used in most cases, while Hijing generator was used to speed-up material studies (jet and hadronic processes).
The total data available after the data-taking period is 90M events (4.5 TB) with 22M events in combined mode. With simulation we produced 4 M events (electrons pions and muons) in a momentum range from 1 to 350 GeV. GRID facilities were extensively used in production of 200 validated runs.

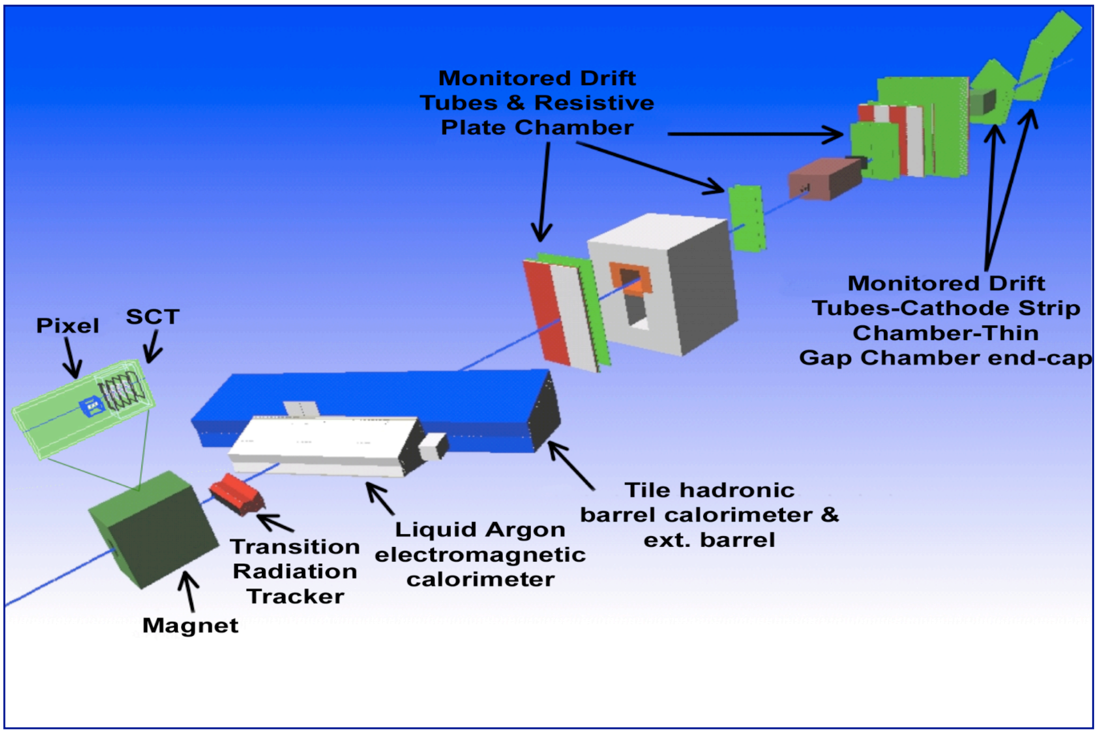


Figure 2 : 3D view of the 2004 combined test-beam setup. The beam line is drawn in blue.

5. Baseline for the simulation application as of August 2007
The recently adopted version of Geant4 for productions of full physics samples throughout 2007 is 8.3.patch01.
The standard physics list adopted for productions is QGSP_EMV. This physics list is known for having a new model of Multiple Scattering which is not allowed to limit the step. Physics-wise this list behaves like QGSP_GN under G4 7.1, from the CPU viewpoint we observe an increasing degradation (in 8.2 QGSP_EMV is about 20% slower than QGSP_GN in 7.1). Table 4a shows the results of preliminary studies performed to compare the performance of different available physics lists when of full physics samples are run. In the table the CPU time/event for the different samples (SUSY events, Z->ee, Z-> tt, H(130)->4

leptons, Minimum Bias events, jets) is shown. Comparisons for the same figures with different Geant4 flavours are also shown.
Table 4b shows performance ratios for different physics lists.

a)

CPUtime per event (kSI2K)					
physics channels	G4.7/QGSP_GN	G4.8/QGSP_EMV	G4.8/QGSP	G4.8/QGSP 1mm	G4.8/QGSP_BERT
susy	921.64	1123.82	1956.42	1560.52	2594.16
Zee	949.58	1107.58	1944.05	1546.41	2432.79
Ztautau	668.64	831.19	1429.71	1361.49	2129.3
H(130)4l	776.72	1067.55	1793.55	1468.79	2334.59
MB	263.35	332.66	584.2	509.29	805.98
jets	765.06	920.77	1480.34	1328.76	1957.11

b)

	Ratios				
physics channels	QGSP_EMV/QGSP_GN	QGSP/QGSP_EMV	QGSP1mm/QGSP_GN	QGSP_BERT/QGSP	QGSP_BERT/QGSP_EMV
susy	1.22	1.74	1.69	1.33	2.31
Zee	1.17	1.76	1.63	1.25	2.2
Ztautau	1.24	1.72	2.04	1.49	2.56
H(130)4l	1.37	1.68	1.89	1.3	2.19
MB	1.26	1.76	1.93	1.38	2.42
jets	1.2	1.61	1.74	1.32	2.13

Table 4 – a) CPU time per event for different physics channels in kSI2Ksec and different physics list flavours b) for the same physics channels ratios with different physics lists

A coherent revision of range cuts in subdetectors has been recently considered for performance optimization. The adopted range cuts are presently set to 50 μm in the Inner Detector and Muon Spectrometer, and to 30 μm in the Calorimeter System. These settings seem to offer the optimal balance between performance and good physics description of all samples. Dedicated studies are ongoing.

6. Validation

This process is parallel to the simulation development. The aim of validation is to spot as soon as possible any non-optimal performance, internal inconsistency, or even inaccurate description of the detectors or physical processes. The validation process uses single particle and physics events in restricted samples for quick feedback or larger samples for validation activities (using also reconstruction) ([5],[6]).

Figure 3 - Total memory allocation vs. Athena release number (MB)

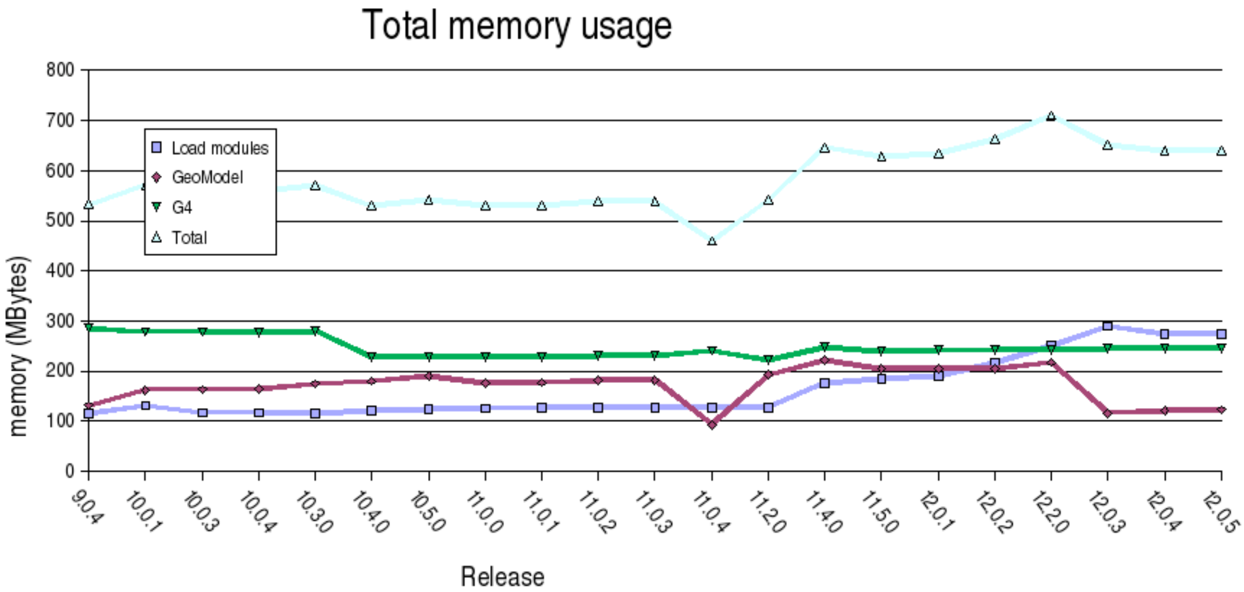


Figure 3 shows the memory consumption at runtime (in MB) as a function of the Athena release number over the last two years. The histogram shows the total amount of memory needed to run the simulation (about 550 MB) with the breakdown of the contributions from GeoModel, Geant4 itself, and loaded modules.

The overall approach for validation is threefold:

- Continuous measurement of the performance in terms of the CPU time and memory consumption.
- Comparisons with real data from old standalone test beams for the different subdetectors, ATLAS combined testbeam and cosmic ray tests.
- Physics performance studies by reconstruction of full physical events.

CPU time per event and memory consumption at runtime is monitored daily. Detailed measurements of these quantities for single particle and full physics events are performed in each new release.

6.1. Particles in Calorimetry

Electrons in the ATLAS Liquid Argon barrel calorimeter are well described by Geant4 both for energy response and resolution. Longitudinal and radial profiles are also well described. In the forward calorimeter a recent steady improvement was observed, even making the resolution a bit too good when compared to the data.

Muons are also well described in the ATLAS barrel calorimeter by simulation: we observe that the mean energy deposits in Tile and LAr calorimeter stay within 2% of their expected values.

Pions and protons are studied with the physics list QGSP and LHEP. Adding the Bertini nuclear cascade model to the Geant4 physics list, a better description of longitudinal and lateral shower profiles has been observed.

6.2. Tile calorimeter and hadronic physics

Comparison between Geant4 and Fluka with respect to real data was recently done using a common source of geometry and the same format of digitization output, allowing common digitization and analysis. GDML+FLUGG+FLUKA (with a Fortran-C++ interface) was used to create FLUKA-hits with the material and the geometry extracted from the G4 simulation of the test-beam.

Electrons with Geant4 seem to have a better agreement with data. Pions agree (2%) with data when using Geant4 with Bertini or FLUKA. Bertini and FLUKA give reasonable agreement with data in longitudinal shower shapes, and after adding Bertini, higher energy in lateral shower halos has been observed. In conclusion, it seems that Geant4 needs Bertini for good data reproducibility in hadronic showers.

7. Fast Simulation

Shower development before the calorimeter is handled by full simulation. Each particle entering the calorimeter is then considered separately by fast simulation mechanism.

Only high-energy electrons and positrons are parameterized. High-energy photons are followed by full simulation up to conversion, and the parameterization then considers the new electron and positron separately. The parameterized shower's energy must be >95% contained within the calorimeter in both the longitudinal and radial directions. In the case they aren't, full simulation is used. If all boundary conditions are satisfied, the track is killed. Fake steps are then simulated along the initial particle trajectory. The number of spots per step is calculated and fake GEANT4 steps are generated to reproduce the radial shower shape. Sampling fluctuations are calculated according to calorimeter resolution. The fake steps are filled up and given to standard Sensitive Detector classes to generate hits. The standard simulation chain is used to process hits until the total shower energy is deposited.

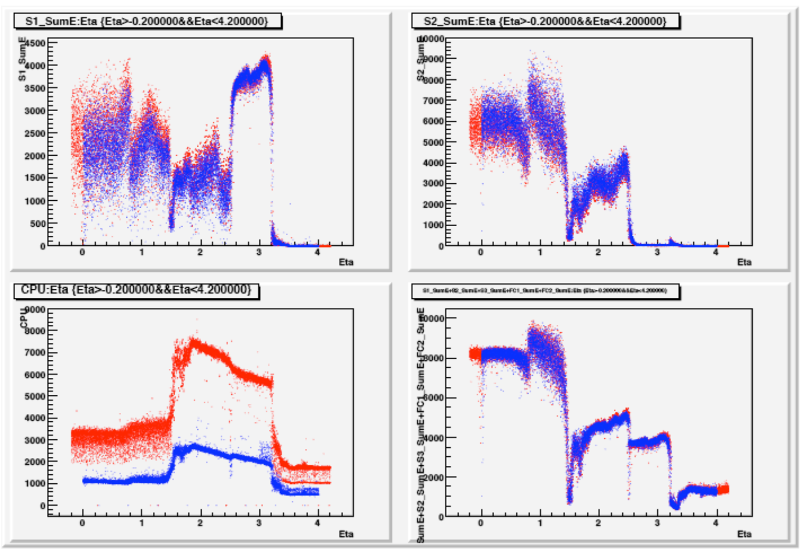
Shower libraries are generated and applied to electrons, positrons, and photons below 1 GeV; the high- and low-energy cut-offs restrict memory consumption. A low-energy cut-off for the parameterization is then generated to minimize CPU time by region.

Very low-energy electrons and positrons are killed, and their energy is deposited in a single step to recreate the average detector response.

Samples produced for most physics channels paired with full simulation samples are in preparation and comparison figures should be ready shortly.

Figure 4 shows performance comparisons for 50 GeV electrons as a function of pseudorapidity for the full and fast samples. A significant gain in CPU performance is seen in the lower left corner of the figure for the whole eta range while the energy deposition in the different compartment is the same for the full and fast simulations.

Figure 4 – Fast- full simulation comparisons vs. eta in the ATLAS colorimeter system. Clockwise from top left: sampling layer one energy deposition, sampling layer two energy deposition, total energy deposition, and total CPU time



- 50 GeV electrons - Full simulation Fast simulation (kill only)

8. Performance

Studies towards an improved optimization are underway. Currently 15-20% of simulation time is spent on in tracking through the magnetic field, including plans to study the possibility of tracking in calorimeters with magnetic field off. Library profiling is also under control and monitored through releases. At each release a set of robustness tests are performed for different physics samples, from single particle to a compilation of physics sets (SUSY events, Z ->ee, Z-> tautau, Z-> mumu, H-> 4 leptons, jets, minimum bias). Memory consumption per event is also measured, comparisons figures are discussed, and the impact of different set of simulation parameters is carefully evaluated through dedicated tests.

9. Conclusions

The ATLAS simulation application is a mature project that is flexible, robust, and successful, but which still needs to be revised, tested and validated before the LHC turn-on. The different subdetector implementations are continuously updated and optimized. Beyond the present release we have planned for the LAr Calorimeter a revision of the entire endcap region for what concerns material, positions, dimensions and contraction. The sagging of the calorimeter is to be implemented as well as the description of imperfections throughout it. The HV imperfections are to be described and a uniform interface for all the LAr subdetectors should be implemented. The Tile Calorimeter needs a careful description of distortions while the muon system should provide a cure to revising active and passive material overlaps. Finally the forward detectors, already in implementation status, should finalize their simulation and integrate it in the full ATLAS simulation.

10. References

[1] Athena framework:*http://atlas.web.cern.ch/Atlas/GROUPS/SOFTWARE/OO/architecture/-General/Documentation/AthenaDeveloperGuide-8.0.0-draft.pdf*.

[2] ATL-SOFT-2004-006 . The Simulation of the ATLAS Experiment: present Status and Outlook. by:Rimoldi, A; Dell'Acqua, A; Gallas, M; Nairz, A; Boudreau, J; Tsulaia, V; Costanzo, D;; Geneva : CERN, 15 Nov 2004

[3] http://atlas-computing.web.cern.ch/atlas/computing/packages/simulation/geant4/G4AtlasApps-/doc/index.html

[4] Cosmics simulation: https://uimon.cern.ch/twiki/bin/view/Atlas/CosmicSimulation

[5] ATL-SOFT-PUB-2005-004, “Validation of the GEANT4-Based Full Simulation Program for the ATLAS Detector” by D. Costanzo, A. Dell'Acqua, M. Gallas, A.Nairz, N. Benekos, A. Rimoldi, J. Boudreau, V. Tsulaia

[6] Simulation validation page: http://atlas-computing.web.cern.ch/atlas/computing /packages-/simulation/geant4/validation/Validation.html

Acknowledgments

The authors want to thank all ATLAS developers in the detector groups that contributed to the successful and constant improvement of their implementation quality, and all organizations that gave financial support for the realization of this project. We are also extremely grateful to colleagues from the Geant4 Collaboration for helping in problem solving and for sharing their experiences with us, in particular J. Apostolakis.