Imperial College London	WP500 SOFTWARE USERS' MANUAL

Doc. Type: SUM				DRD N°: NA
Doc. N°: **WP500-SUM-ICL-002**	Issue: **1.0**	Date: **05/05/2004**	Page **1** Of **21**	
Title: Simulation of Test-Mass Charging in the LISA Mission				

	Name & Function	Signature	Date
Prepared by:	Araújo H. Wass P. Sumner T.		
Approved by:	Araújo H. Wass P. Sumner T.		
Application authorised by:			
Customer / Higher Level Contractor			
Accepted by:			
Approved by:			

DISTRIBUTION LIST	
Internal Araújo H. Wass P. Sumner T.	
External Lei F. (QinetiQ)	

Data Management: -- Signature Date	File: WP500_SUM_ICL_002_v1.0.doc

CHANGE RECORD			
ISSUE	DATE	CHANGE AUTHORITY	REASON FOR CHANGE AND AFFECTED SECTIONS
001-1.0	05/01/2004		Draft Issue
002-1.0	05/05/2004	H. Araújo	Reissue with minor modifications for Geant4 advanced example

LIST OF VALID PAGES									
PAGE	ISSUE	PAGE	ISSUE	PAGE	ISSUE	PAGE	ISSUE	PAGE	ISSUE
1-21	**001-1.0**								
1-21	**002-1.0**								

TABLE OF CONTENTS

1 Introduction

1.1 Contractual

This document has been issued by the Astrophysics Group at Imperial College London for QinetiQ, under contract CU009-0000028631, for the SEPTIMESS Project (Space Energetic Particle Transport and Interaction Modelling for ESA Science Studies). It describes work undertaken in Work Package 500 – Simulation of Test-Mass Charging in the LISA Mission.

1.2 Purpose of the Document

This document is the Software Users' Manual (SUM) for the simulation code developed for modelling electrostatic charging of the test masses onboard the LISA Science Modules using the GEANT4 toolkit.

1.3 Scope of the Software

This code is exclusively intended for use with the GEANT4 toolkit for the simulation of the stochastic charging of the LISA test masses due to energetic protons and helium nuclei from galactic and solar origin. When compiled and linked into the GEANT4 toolkit, the code will track the primary cosmic rays and any secondary particles produced in the spacecraft geometry, in particular those entering or leaving the test masses.

1.4 Applicable Documents and References

[AD 1]	WP500-URD-ICL-001 (LISA Users' Requirements Document)
[AD 2]	WP500-SSD-ICL-001 (LISA Software Specification Document)
[AD 3]	WP500-SVVP-ICL-001 (LISA Software Verification and Validation Plan)
[AD 4]	WP500-TN-ICL-001 (Report on Test-Mass Charging in the LISA Mission)
[AD 5]	WP600-TN-ICL-001 (Report on Test-Mass Charging in LISAPathfinder Mission)
[AD 6]	WP400-RP-ICL-001 (Report on Charging from Solar Energetic Particles)
[AD 7]	Geant4 website: http://cern.ch/geant4
[AD 8]	Geant4 Installation Guide: http://geant4.web.cern.ch/geant4/G4UsersDocuments/UsersGuides/InstallationGuide/html/index.html
[AD 9]	Geant4 Physics Reference Manual http://geant4.web.cern.ch/geant4/G4UsersDocuments/UsersGuides/PhysicsReferenceManual/html/PhysicsReferenceManual.html
[AD 10]	Geant4 User's Guide for Application Developers http://geant4.web.cern.ch/geant4/G4UsersDocuments/UsersGuides/ForApplicationDeveloper/html/index.html
[AD 11]	LISA Integrated Solid Model (NASA/GSFC) Rev 1, Sep 2003.
[AD 12]	LISA Inertial Sensor Design Report (Carlo Gavazzi Space) LTP-RP-CGS-001.
[AD 13]	Vannuccini *et al.*, 27th Int. Cosmic Ray Conf., Hamburg, Germany, 2001
[AD 14]	GPS Documentation : http://www.space.qinetiq.com/geant4/gspm.html
[AD 15]	AIDA – Abstract Interfaces for Data Analysis (http://aida.freehep.org)

1.5 Acronyms

AIDA	Abstract Interfaces for Data Analysis
ANAPHE	C++ Implementation of AIDA (formerly called LHC++)
CAD	Computer Aided Design
CERN	European Organisation for Nuclear Research
CMS	Charge Management System
ESA	European Space Agency
G4	Geant4
GCR	Galactic Cosmic Rays
GNU	Gnu is Not Unix
GPS	G4 General Particle Source module
GUI	Graphical User Interface
HBOOK	CERN Data Presentation and Analysis Package
IA	Interactive Analysis
ICL	Imperial College London
LISA	Laser Interferometer Space Antenna
LTP	LISA Technology Package onboard SMART-2
OpenGL	Graphical rendering and acceleration software libraries
PAW	Physics Analysis Workstation (CERN)
ROOT	Object Oriented Data Analysis Framework
SMART	ESA Missions for Advanced Research in Technology
UI	User Interface
XMotif	Graphical User Interface Toolkit (IEEE 1295 specification)

2 General Description

The `CRCharging` software application tracks cosmic-ray particles and their secondaries in the LISA spacecraft, and keeps a record of events in which a net particle charge results in one of the two test masses [AD 1-6]. The following sections describe the usage of the software in some detail, assuming that the user has a basic familiarity with Geant4 [AD 7-10]. Some guidance is provided in Section 2.5 for novice users.

2.1 Geometry implementation of the LISA Science Module

The LISA G4 geometry model, shown in Figure 1, consists of nearly 200 placed volumes (85% of total spacecraft mass), and is based on the LISA Integrated Solid Model [AD 11]. Each LISA spacecraft includes two interferometer telescopes mounted inside a Y-shaped payload structure. The outer spacecraft structure is approximately 2.7 m in diameter and 0.5 m in height. Two test masses are contained in inertial sensors located in each of the two interferometer arms. The coordinate system is the same as in the LISA solid model: the origin is located at the midpoint between the two TMs, the x-axis bisects the 60^o angle between the interferometer arms and the *z* axis points upwards perpendicularly to the spacecraft plane. The IS implementation was based on the LTP engineering model described in [AD 12]. The G4 model of the sensor is shown in Figure 2. The 46-mm TM is represented by the yellow cube at the centre. The surrounding white and orange structures represent the sensing and injection electrodes, respectively.

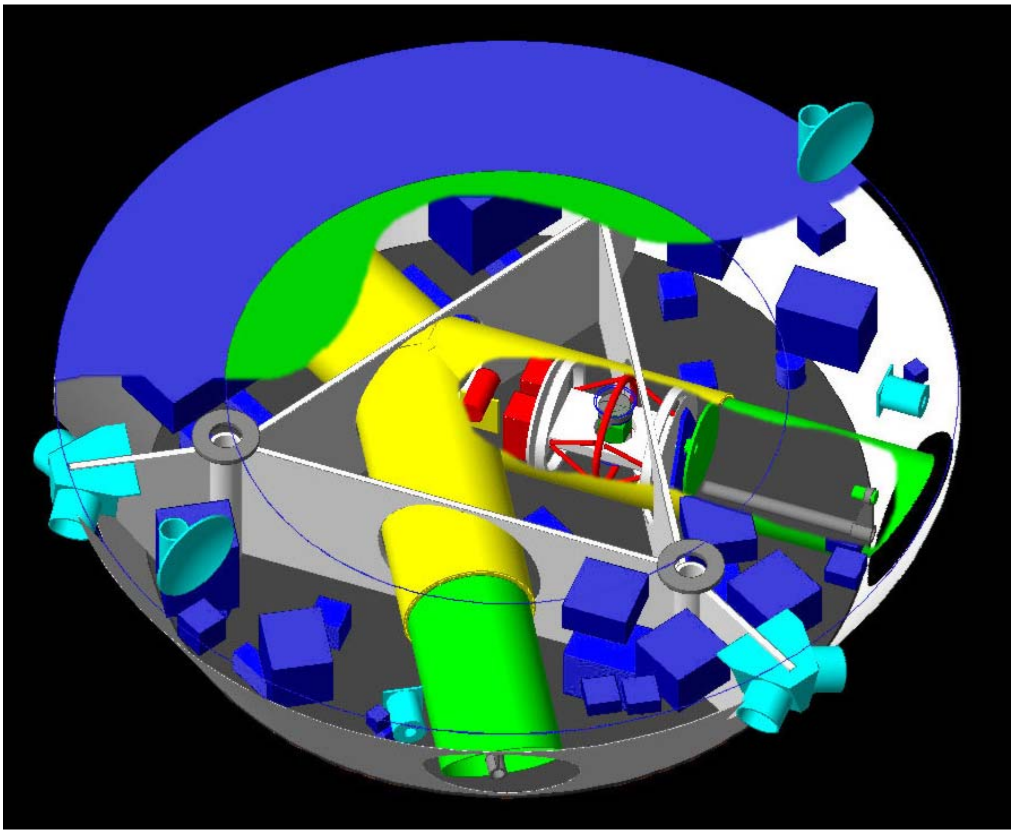

Figure 1: Geometry model of LISA Science Module spacecraft implemented in Geant4.

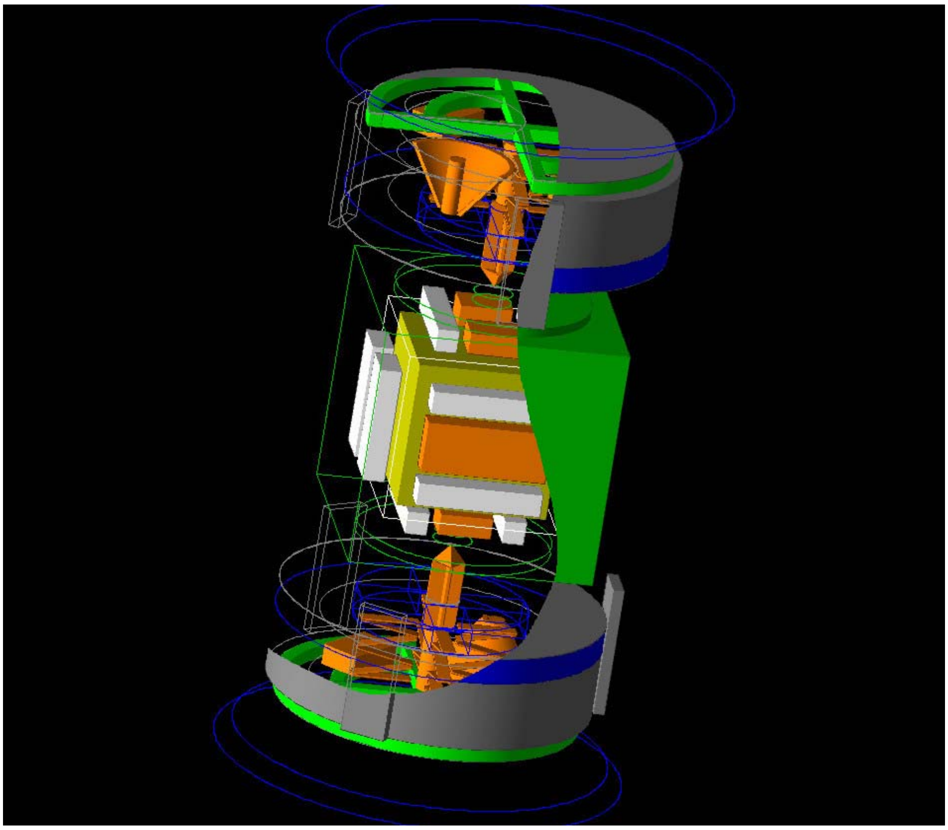

Figure 2: LTP Inertial Sensor design

2.2 Physics processes and models

In Geant4, individual physics processes and production thresholds (cuts) for secondaries are attached to each particle, which are tracked independently across the geometry. A considerable number of physics models existing in G4 have been implemented in the LISA application, which we group in four broad categories:

1. Electromagnetic Physics
2. Hadronic Physics
3. Electronuclear Physics
4. Decays in Flight

Section 4.1 explains how to generate lists of particles and processes implemented in the `CRCharging` application. Their choice is justified in [AD 4], where this issue is discussed in some detail. For a complete description of the Geant4 physics models we refer to [AD 9].

2.3 Low energy thresholds

Although secondary particles produced by electromagnetic processes are tracked until they lose all their energy (i.e. there are no *tracking* cuts), they are only produced above a certain energy threshold (*production* cuts). These cuts are specified by a range for a given particle, which is translated by G4 into an energy threshold for that particle in a particular material. The lowest cut in G4 corresponds to an energy threshold of 250 eV in all materials.

There is, however, one exception to these rules: a higher production cut intrinsically applied to secondary electrons emitted from proton tracks (or other hadrons). These are limited to a minimum energy equal to the mean excitation energy of the material (the quantity *I* in the Bethe-Block equation), which is 790 eV for gold. This arises from the lack of good quality experimental data describing the influence of atomic structure on hadron-induced ionisation. No such problem exists for electron-induced ionisation, which is well parameterised down to 250 eV.

Recent G4 releases allow the user to specify regions of the geometry for which different cuts can be specified. In spite of the dangers associated to the existence of different range cuts across a boundary between two regions, which can lead to inconsistent physics results, this is an extremely powerful feature for applications such as LISA. These "cuts-by-region" allow efficient tracking of higher energy particles away from the test masses, by avoiding the production of copious amounts of low energy secondaries which could never reach into the IS, while keeping the shortest possible production cuts in the immediate vicinity of the TMs. Two regions have been defined in the LISA geometry: the default (world) region, in which a range cut of 2 mm has been assigned to electrons, positrons and gammas, and the inertial sensor, where the minimum 250 eV energy threshold applied to those particles. The standard G4 UI allows the modification of the default cuts for any particle and region.

2.4 Generation of the LISA radiation environment

2.4.1 Radiation environment

During the science phase of the LISA mission, two particle fluxes will dominate the charging rates observed in its test masses: a relatively steady flux of cosmic-ray protons and light nuclei of *galactic* origin and particles originating in *solar* events such as solar flares and coronal mass ejections.

The galactic cosmic-ray (GCR) spectrum varies along the 11-year long solar cycle. The interplanetary magnetic field is disturbed the least at the minimum of solar activity; as a consequence, a higher cosmic ray flux is expected in the solar cavity and in the LISA Earth-like orbit in particular. At solar minimum, approximately 90% of the particle flux consists of protons, 8% of helium nuclei (^{3}He and ^{4}He), 1% of heavier nuclei and 1% of electrons. The smallest galactic cosmic-ray fluxes are expected at solar maximum. The three most abundant primary nuclei (p, ^{3}He, ^{4}He) at solar minimum and solar maximum fluxes are considered the main input scenarios for the LISA simulation. The corresponding differential energy spectra are depicted in Figure 3 ([AD 13]).

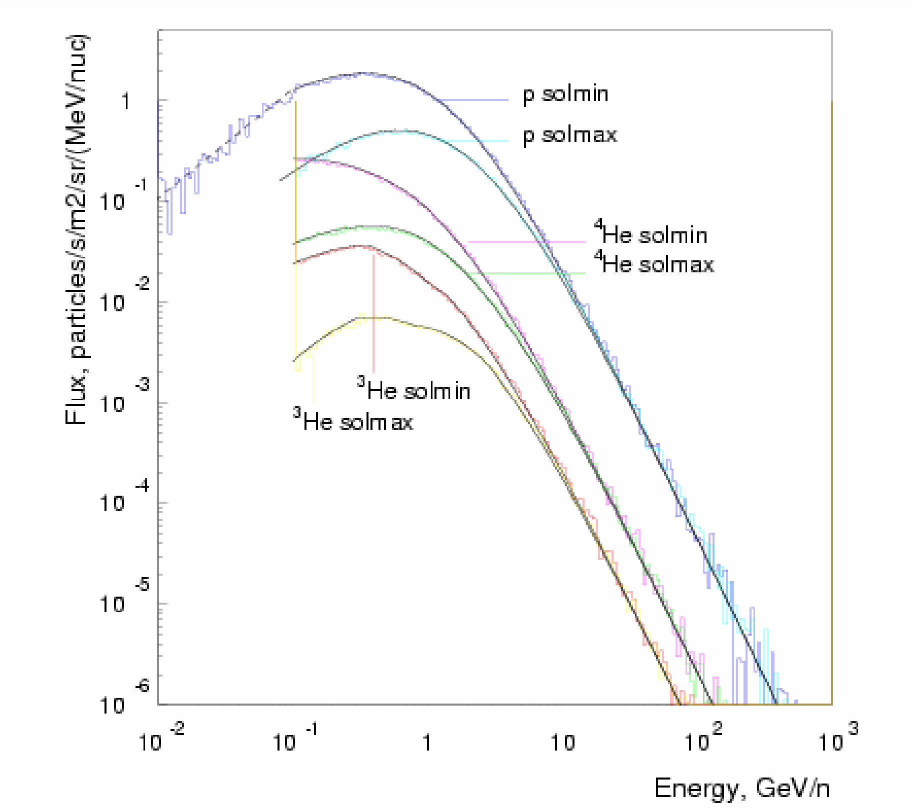

Figure 3: Primary spectra for galactic cosmic rays.

The second main radiation flux has its origin in solar events. Although, to first approximation, the same particles are involved, their relative abundance differs from the galactic cosmic-ray component and their distribution and intensity varies a great deal from event to event. Here we must distinguish between solar events of two types: shorter, 'impulsive' events associated with smaller solar flares, which typically last for hours, and longer, less frequent ones attributed to more dramatic coronal mass ejections. The latter have considerably harder spectra and can upset the LISA science measurements, whereas the spectrum of smaller solar-flare events is too soft to penetrate the shielding provided by the LISA spacecrafts and cause significant effects. Due to the very disparate nature of solar flare events, it is not possible to select a representative spectrum for detailed modelling. In any case, an estimate of the charging rates expected from solar flare events can be obtained if a "spectral charging efficiency" is known for a particular input spectrum. The issue of solar events in the context of test-mass charging is discussed in more detail in [AD 6].

2.4.2 Generic specification of primary particles

The LISA software uses the Geant4 General Particle Source Module (GPS) as a particle gun for specification of the primary vertex. The GPS implements generic particle sources of varying complexity, from simple monoenergetic, unidirectional pencil beams to more sophisticated angular, spatial and energy distributions. A description of this module is given in [AD 14].

All GPS-related commands can be found under the '`gps/`' directory in the LISA user interface. For example, a simple source configuration for 100 MeV α-particles emitted isotropically from the origin could be defined with:

```
/gps/particle alpha
/gps/position 0 0 0 cm
/gps/angtype iso
/gps/energy 100 MeV
```

2.4.3 Generator surface and normalisation

The GPS can easily generate the *isotropic* and *uniform* galactic cosmic-ray flux. The simplest way to do this is to emit from points uniformly chosen over a generator sphere which comprises the whole geometry, using a cosine-law angular distribution around the local surface normal.

The integral flux of particles crossing a unit area per unit time which are expected from N_0 primaries is equal to $\Phi=(N_0/T)/(2\pi R^2)$ anywhere inside a generator sphere of radius R, where T is the exposure time. Note that this expression includes hits from both sides of the unit surface. Using the integral fluxes obtained from the differential spectra in Figure 3, one can calculate the Monte Carlo conversion factors for a generator sphere with R=1.5 m; the exposure times corresponding to 1 million events (1 Mevt) simulated are given in the table below:

Table 1: Integral fluxes and Monte Carlo conversion factors for cosine-law generation from 1.5 m radius sphere

Primary	Condition	Flux, /s/cm^2	Time, s/Mevt
protons	solar minimum	4.29	1.651
protons	solar maximum	1.89	3.748
He-3	solar minimum	0.0591	119.6
He-3	solar maximum	0.0236	299.9
He-4	solar minimum	0.315	22.4
He-4	solar maximum	0.142	49.9

2.4.4 Pre-defined input macros

The six particle/flux combinations for galactic cosmic rays described in the previous section are implemented as macros distributed in the '`primaries/`' directory:

```
1. protons_solmin.mac
2. protons_solmax.mac
3. helium3_solmin.mac
4. helium3_solmax.mac
5. helium4_solmin.mac
6. helium4_solmax.mac
```

These macros set-up the GPS, the levels of verbosity and other input parameters, and offer a convenient way to start a batch run (without visualisation); for example:

```
CRCharging primaries/protons_solmin.mac
```

Figure 4 describes the macro for protons at solar minimum flux. The differential energy spectra generated in the six cases are those previously shown in Figure 3.

2.5 Default control functionality

A description of the standard G4 User Interface is outside the scope of this document. We refer to [AD 10] for this purpose. However, some basic G4 commands are briefly introduced, as a way of allowing users new to G4 to get quickly familiarised with the LISA application.

2.5.1 Geometry visualisation

Geant4 supports a number of tools for detector and event visualisation. The one used by default in LISA is one of the simplest, OpenGL, which is freely distributed with most Linux packages. It can offer some display interactivity when combined with XMotif, but it is otherwise quite plain. Visualisation stretches system resources, and should be disabled for most modelling work. The following is a list of some useful, self-explanatory visualisation commands:

```
/vis/enable
/vis/disable
/vis/viewer/refresh
/vis/viewer/viewpointThetaPhi 30 30
/vis/viewer/zoom 2
/vis/viewer/pan 0 2 cm
/vis/viewer/reset
```

The physical volume list and hierarchy implemented in the LISA geometry can be found by invoking the ASCII Tree viewer:

```
/vis/open ATree
/vis/ASCIITree/verbose 2
/vis/viewer/refresh
```

```
####################################################################
# Differential energy spectrum for solar minimum PROTONS at 1 AU
####################################################################
#
/vis/disable
/tracking/storeTrajectory 0
/run/verbose 1
/event/verbose 0
#
/control/suppressAbortion 1
#
# Random seeds
/run/autoSeed 1
#/random/resetEngineFrom seeds.rndm
#
# particle surveys
#/surveys/surveyParticles 1
#/surveys/surveyPrimaries 1
#/surveys/surveyTestMasses 1
#
/gps/particle proton
#
# Isotropic distribution
/gps/type Surface
/gps/shape Sphere
/gps/radius 1499. mm
/gps/centre 0. 0. 0. mm
/gps/angtype cos
#
# Differential energy spectrum for solar minimum
# proton flux (/m2/s/sr/MeV) from Vannuccini 2001
# INTEGRAL FLUX = 4.29 p/cm2/s
/gps/energytype Arb
/gps/histname arb
#
/gps/histpoint   1.0000e+01   1.0886e-01
/gps/histpoint   1.4874e+01   1.6753e-01
…
…
/gps/histpoint   1.0000e+06   7.6479e-08
#
/gps/arbint Lin
#
/run/beamOn 10000000
```

Figure 4: Macro protons_solmin.mac

2.5.2 Particles and processes

The list of available particles can be generated with one of the two following commands:

```
/particle/list
/gps/List
```

Registered processes can like-wise be listed with:

```
/process/list
```

A process can be made inactive (although it will remain in memory) or re-activated at any stage during the session:

```
/process/inactivate MultipleScattering
```

Processes are often made inactive in order to study one interaction at a time. In general, it is helpful to look at processes attached to a specific particle, in this case an electron:

```
/particle/select e-
/particle/process/dump
```

It is also possible to inactivate/activate processes attached to individual particles:

```
/particle/select e-
/particle/process/dump
/particle/process/inactivate 1
```

This inactivates multiple scattering for electrons (the process labelled `1' in the electron process list) but not, for example, for positrons.

2.5.3 Running and Tracking

After setting up the primary generator as described in Section 2.4.2, the user can start the simulation of *n* events with:

```
/run/beamOn n
```

Specification of the primary vertex and other parameters can be given from a macro file:

```
/control/execute file.mac
```

The tracking of individual particles can be followed in the display, provided that visualisation is enabled, or more comprehensively in the terminal by typing:

```
/tracking/verbose 1
```

When running a very large number of events, it is inevitable that the application will sometimes terminate abnormally. This can be due, for example, to a fault in a particular physics process, which becomes apparent only for a particular combination of particle type, incident energy, material and reaction channel, therefore eluding the comprehensive G4 testing procedure. To prevent the application terminating in most of these instances (i.e. errors which generate a G4Exception), the follow flag can be set:

```
/control/suppressAbortion n
```

where n=0,1,2 reflects the level of G4Exception the user is prepared to ignore. Naturally, the results obtained in these particular events should no longer be trusted; in practice, due to the low probability for TM charging per event, this should not pose a significant problem.

2.6 Random seeds

LISA makes use of the *Ranecu* random engine [AD 10]. Seed handling by the Ranecu generator is slightly different than for other engines available in G4. In particular, the internal status of the engine is completely specified by only two seeds (long integers), while other generators require a lot more information. This makes it particularly easy to store the internal status for each event for later reproduction. The two random seeds are written to the output file for events in which net charging has occurred.

LISA provides automatic generation of the random seeds based on the system clock. This is useful for submission of multiple jobs in a distributed computing environment. This feature is activated with the command:

```
/run/autoSeed [0] 1
```

In the auto-seed mode, the system time at the beginning of a run is used to seed the random engine, and also to specify the output filename as described in Section 2.8. If the user wishes to specify a particular couple of seeds, such as those corresponding to a charging event already simulated, these should be inserted into an ASCII file and loaded with the following command:

```
/random/resetEngineFrom seeds.rndm
```

2.7 Particle surveys

Some functionality is provided to enable the application to output energy spectra for different particles in the geometry. These particle surveys are gathered in the `/surveys/` group of commands. The commands and output files are listed below.

2.7.1 Primary particles

The command below is used for quick inspection of the primary energies (MeV), which are written to the ASCII file specified. The primary is killed immediately after generation. Data in Figure 3 was obtained in this way.

```
/surveys/surveyPrimaries 1 [0]
```

```
1. primaries.out
```

2.7.2 All particles in geometry

The following command writes the energy (MeV) of all particles created anywhere in the geometry to the ASCII files listed below. It is mainly used to check particle abundances and whether physics models are available for each particle and energy range.

```
/surveys/surveyParticles 1 [0]
```

```
1. electrons.out
2. gammas.out
3. muons.out
4. kaons.out
5. pions.out
6. neutrons.out
7. fragmemts.out
8. nuclei.out
9. others.out
```

Note that only particles above 1 MeV are written to files 1 and 2, and that energies in files 7 and 8 are expressed in MeV/nucleon.

2.7.3 Test mass spectra

The following command writes the energies (eV) of electrons and hadrons entering and leaving any of the test masses to the ASCII files listed above, regardless of whether or not they lead to charge deposition.

```
/surveys/surveyTestMasses 1 [0]
```

```
1. electrons_in.out
2. electrons_out.out
3. hadrons_in.out
4. hadrons_out.out
```

2.8 Output

Besides the files produced by the particle surveys just described, the basic output of the LISA application is an ASCII file which summarises the TM charging registered in a single run for those events in which a non-zero net charge has occurred. If the `autoSeed` flag has *not* been activated, this run summary file is named

```
charge.out
```

When running in `autoSeed` mode, LISA generates output filenames on the fly for each run, using the random numbers that will seed the first event in the run (obtained from the system date). One such example is:

```
run00_1066247777_482071154.out
```

The same information is written to a HBOOK file (same filename but with extension `hbook`) if the code has been compiled with the environmental variable `G4_ANALYSIS_USE`, which specifies the use of the AIDA interface [AD 15]. This file will contain a HBOOK ntuple which can be read directly into PAW or ROOT:

```
run00_1066247777_482071154.hbook
```

A section of an output file is shown in Figure 5, where the column labels have the following meanings:

`Event`	event number in a run
`TM`	test mass number (0,1)
`Energy`	primary energy (MeV)
`Qnet`	net event charge (+e)
`Qin`	total charge entering TM (+e)
`Qout`	total charge leaving TM (+e)
`Seed1`	random seed 1
`Seed2`	random seed 2

```
Event   TM  Energy     Qnet  Qin   Qout  Seed1       Seed2

3424    1   52916.3    -1    0     1     557247848   1301175715
6765    0   3074.44    -1    -1    0     131735253   182683243
14057   0   2739.53    1     1     0     2101024993  945850202
17500   0   2048.68    2     1     -1    1172719212  2143550912
17868   1   1623.45    1     0     -1    1591157630  380789242
```

Figure 5: Example of text output.

3 Installation and Compilation

The LISA software is distributed as C++ source code and should be compiled and linked with the Geant4 toolkit. A previous installation of Geant4 is therefore required [AD 8]. Installation of the LISA code is very similar to that of the G4 examples; a configuration suitable for this task is therefore assumed.

The LISA software is distributed in a GNU tar file named `LISA-V04_YYMMDD.tar.gz`, where YYDDMM corresponds to the "release" date (i.e. when the code was last updated). The user should create a directory in which the code is to be installed, copy the code to that directory, make it his/hers current directory, and finally unpack the compressed tar file:

```
mkdir LISA
mv LISA-V04_YYMMDD.tar.gz LISA/
cd LISA
gtar zxfv LISA-V04_YYMMDD.tar.gz
```

A number of files and three subdirectories should appear, including a `LISA.cc` file containing the main function, a `GNUmakefile` for compiling the code, the initialisation macro `init.mac`, the directories `src/` and `include/` with the source code, and a directory `primaries/` containing the primary macros. The code is compiled by typing:

```
gmake
```

in the directory containing the `GNUmakefile`. If an installation of ANAPHE exists and the environment variable `G4ANALYSIS_USE` is set, the HBOOK output will be produced during execution.

4 Execution and Commands

4.1 Execution

After successful compilation and linking, the code is executed by typing

```
CRCharging <[macrofile]>
```

from the directory where the Geant4 execution files are installed, or from any other place if that directory is included in the PATH environment variable. If a macro file is provided, a session is executed in batch mode, with no visualisation. Otherwise, the macro `init.mac` is loaded to prepare for an interactive session.

4.2 Standard G4 UI commands

These will not be described.

4.3 Command directory /surveys/

Commands in this directory set the flags for output of the particle surveys described in Section 2.7.

4.3.1 /surveys/surveyPrimaries

```
Format:     /surveys/surveyPrimaries <flag>
Arguments:  G4Bool flag
Function:   If flag=true primary energies (MeV) are written to
            ASCII file primaries.out.
Default:    0
```

4.3.2 /surveys/surveyParticles

```
Format:     /surveys/surveyParticles <flag>
Arguments:  G4Bool flag
Function:   If flag=true particle energies (MeV, or
            MeV/nucleon) are written to particle-specific
            ASCII files.
Default:    0
```

4.3.3 /surveys/surveyTestMasses

```
Format:     /surveys/surveyTestMasses <flag>
Arguments:  G4Bool flag
Function:   If flag=true particle energies (eV) for electrons
            and hadrons entering or leaving a test mass
            are written to ASCII files electrons_in.out
            electrons_out.out, hadrons_in.out,
            hadrons_out.out.
Default:    0
```

4.4 Command directory /run/

One application-specific command has been included in this default G4 command directory, intended to switch automatic seeding of the random engine on/off as described in Section 2.6 and Section 2.8.

4.4.1 /run/autoSeed

```
Format:     /run/autoSeed <flag>
Arguments:  G4Bool flag
Function:   If flag=true, automatic (time-based) seed
            generation and filenaming are switched on.
Default:    0
```